

IDENTIFICATION  
-----

PRODUCT CODE: AC-6527D-MA  
PRODUCT NAME: AIKLAD0 KL8-J/K LOOP PACK TST  
PRODUCT DATE: JANUARY 1979  
MAINTAINER: DIAGNOSTIC GROUP  
AUTHORS: BRUCE HANSEN, R. MOORE, DON RICE,  
AND HEATHER CZIRIA

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 - 1979, 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
10.1	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 5637 OF FIELD 0 AND 6600 TO 7377 OF FIELD 1.

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 M8655'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
0	0	0	1	150	BAUD B1=0	B2=0	B3=1
0	0	1	0	300	BAUD B1=0	B2=1	B3=0
0	0	1	1	600	BAUD B1=0	B2=1	R3=1
0	1	0	0	1200	BAUD B1=1	B2=0	R3=0
0	1	0	1	2400	BAUD B1=1	B2=0	R3=1
0	1	1	0	4800	BAUD B1=1	B2=1	R3=0
0	1	1	1	9600	BAUD B1=1	B2=1	R3=1
1	0	0	0	19.2K	BAUD B1=1	B2=1	R3=1
1	0	0	1	56.8	BAUD B1=0	B2=0	B3=0
1	0	1	0	66.7	BAUD B1=0	B2=0	B3=0
1	0	1	1	1050	BAUD B1=1	B2=0	B3=0

\*

\*19.2 KILO 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 2330
- 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, OR 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 SMD JUMPER IS INSTALLED TYPE Y IF NOT TYPE N  
THE PROGRAM WILL THEN TYPE SWD=1? IF ANSWER WAS YES, OR SWD=0? IF ANSWER WAS NO. SMD=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, OR 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  
(XXX 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 SR JUMPER IS NOT INSTALLED TYPE Y IF IT IS TYPE N.  
THE PROGRAM WILL THEN TYPE SB=0? IF ANSWER WAS YES,  
OR 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 2330
- 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-RE 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 SFTUP 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 C,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 (CURBRD) 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 2211. 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 2211 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (CAFXTM) 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 2211 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 2211 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 2211 WITH THE AC EQUAL TO THE PC WHERE THE ERROR WAS DETECTED.

THE NEXT TEST (ACMSKP) CHECKS THE EFFECT OF THE IOT ON THE AC AND ALSO CHECKS THAT THE IOT'S DO NOT SKIP. TPC AND TIS ARE NOT TESTED. AN ERROR HALT AT LOCATION 2231 INDICATES THAT AN IOT SKIPPED THAT SHOULD'N'T, THE AC CONTAINS THE PC WHERE THE ERROR WAS DETECTED. AN ERROR HALT AT LOCATION 2253 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 TIS 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 TIS 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 1454 WITH THE AC EQUAL TO THE WORD EXPECTED. PRESS "CONTINUE" AND THE PROGRAM WILL HALT AT LOCATION 1457 WITH THE WORD RECEIVED IN THE AC. PRESS "CONTINUE" AGAIN AND THE PROGRAM WILL HALT AT LOCATION 1462 WITH THE AC EQUAL TO THE NEW WORD TRANSMITTED. THIS WORD MAY BE THE SAME AS THE WORD EXPECTED DEPENDING ON 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 EQUALS 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 (BAUDTW) 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 21 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 04 AND A RECEIVE IOT OF 03.

ADDRESS 21 (HARDWARE CONFIGURATION WORD 1)  
-----

THE FOLLOWING IS THE BIT DEFINITION FOR HARDWARE CONFIGURATION WORD 1.

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  
 BIT2=1 PPROCESSOR NOT OF THE PDP 8/E FAMILY.  
 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 CONSOLE TERMINAL PACKAGE
- BIT4=1 IF FILLER CHARACTER JUMPER INSTALLED FIL=1

BIT5	BIT6	BIT7	BIT8	BAUD RATE	ROCKER SWITCHES	JUMPERS	VARIATION
0	0	0	0	110	B2=0	W2=1 W5=0	KL8-JA & KA
0	0	0	1	150	B2=0	W2=1 W5=0	KL8-JA & KA
0	0	1	0	300	B2=1	W2=1 W5=0	KL8-JA & KA
0	0	1	1	600	B2=1	W2=1 W5=0	KL8-JA & KA
0	1	0	0	1200	B2=0	W2=1 W5=0	KL8-JA & KA
0	1	0	1	2400	B2=0	W2=1 W5=0	KL8-JA & KA
0	1	1	0	4800	B2=1	W2=1 W5=0	KL8-JA & KA
0	1	1	1	9600	B2=1	W2=1 W5=0	KL8-JA & KA
* 1	0	0	0	19.2K	B2=1	W2=0 W5=1	KL8-JA & KA
1	0	0	1	56.8	B2=0	W2=1 W5=0	KL8-KD (M8655-YC)
1	0	1	0	66.7	B2=0	W2=1 W5=0	KL8-KC (M8655-YB)
1	0	1	1	1050	B2=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	NB1=1 NB2=1
0	1	6	NB1=0 NB2=1
1	0	7	NB1=1 NB2=0
1	1	8	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 SUNCHCKS 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  
-----

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

TO CONVEY THAT THE PSEUDO SWITCH REGISTER, ADDRESS 20, IS BEING ENABLED IN PLACE OF THE HARDWARE SWITCH REGISTER, ADDRESS 21 (HCW1) BIT 0 MUST BE SET TO A ZERO.

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 IS 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 ERPOP 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:

END OF PASS XXXX.

XXXX IS THE OCTAL NUMBER FOR THE PASS JUST COMPLETED.

11.0 LISTING  
-----



```

1 /KL8-JA & KL8-KA/KB/KC/KD LOOP BACK TEST.
2 /AC=6527D-MA
3
4 /COPYRIGHT (C) 1973-1979 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 / REV D AIDS P.R. CC 1004 - HEATHER CZIRIA
10 /
11 /THE IOT'S ARE USED ON THE APT8A TEST SYSTEM
12 /WHEN MULTIPLE DEVICES ARE TO BE TESTED.
13
14 6370 APTIO0=6370 /LOAD THE SELECT COUNTER FROM AC BITS
15 /8-11 AND CLEAR THE AC, THIS IS USED
16 /TO INITIALIZE TO DEVICE ZERO
17 6371 APTIO1=6371 /INCREMENT THE SELECT COUNTER
18 6372 APTIO2=6372 /SET THE FAIL FLAG DESCRIBED BY THE
19 /SELECT COUNTER
20 6373 APTIO3=6373 /CLEAR SELECT COUNTERS AND ALL FLAGS
21 6374 APTIO4=6374 /SKIP IF FAILURE FLAG DESCRIBED
22 6375 APTIO5=6375 /BY THE SELECT COUNTER IS SET.
23 /READ SELECT COUNTERS INTO AC BITS 8-11
24 /
25
26
27
28 /RECEIVE IOTS FOR KL8-JA,KA,KB,KC & KD
29
30 6007 CAF=6007 /CLEAR ALL FLAGS
31 6030 KCF=6030 /CLEAR RECEIVE FLAG,DON'T SET READER RUN F/F
32 6031 KSF=6031 /SKIP ON RECEIVE FLAG
33 6032 KCC=6032 /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
34 6033 KRS=6034 /READ RECEIVE BUFFER AND STATUS
35 6035 KIE=6035 /AC 11=1 SET INTERRUPT ENABLE
36 /AC 11 = 0 CLEAR INTERRUPT ENABLE F/F
37 6035 KSE=KIE /AC10=1 SET STATUS WORD ENABLE
38 /AC10=0 CLEAR STATUS WORD ENABLE
39 6036 KRB=6036 /CLEAR AC AND RECIEVE FLAG,SET READER RUN AND
40 /HEAD RECEIVE BUFFER AND STATUS
41
42 /TRANSMIT IOTS FOR KL8-JA,KA,KB,KC & KD
43
44 6040 TFL=6040 /SET THE TRANSMIT FLAG
45 6041 TSF=6041 /SKIP ON TRANSMIT FLAG
46 6042 TCF=6042 /CLEAR THE TRANSMIT FLAG
47 6044 TPC=6044 /LOAD TRANSMIT BUFFER AND TRANSMIT
48 6045 SPI=6045 /SKIP IF TRANSMIT OR RECEIVE FLAG IS SET AND
49 /INTERRUPT ENABLE FLIP/FLOP IS SET
50 6046 TLS=6046 /LOAD TRANSMIT BUFFER,TRANSMIT AND CLEAR TRANSMIT FLAG
51 7402 HALT=7402
52
53
54 /SWITCH REGISTER SETTINGS
55 /SR0=1 DON'T HALT ON ERROR

```

```

56 /SR1=1 LOOP ON ERROR OR DATA PATTERN
57 /SR2=1 LOOP ON TEST SEQUENCE
58 /SR10=1 PROCESSOR NOT A PDP-8E
59 /SR11=0 STARTING ADDRESS 200 ONLY-SETUP PROGRAM LIMITS
60 / BY WAY OF THE SWITCH REGISTER
61 /SR11=1 STARTING ADDRESS 200 ONLY-SETUP PROGRAM LIMITS
62 / BY WAY OF THE TELETYPE(DEVICE CODES OF 03&04)
63 /
64 7002 BSW=7002
65
66
67
68
69
70
71
72
73
74 0000 *0
75 0000 0304 304 /REVISION D
76 0001 5001 JMP 1
77 0002 0002 2
78 0003 0003 3
79
80 0011 *11
81 0011 0000 A11, 0 /FORUSE ON APT-8
82 0020 *20
83 /THE FOLLOWING 4 LOCATIONS ARE RESERVED FOR USE BY THE APT 8 TEST SYSTEM.
84 /
85 0020 0000 A20, 0
86 0021 4000 A21, 4000
87 0022 0000 A22, 0
88 0023 0000 A23, 0
89 /
90 0024 *24
91 /
92 0024 0000 SAVPNT, 0
93 0025 0002 K2, 2
94 0026 0115 K115, 115
95 0027 0117 K117, 117
96 0030 0033 K33, 33
97 0031 0037 K37, 37
98 0032 0077 C77, 77
99 0033 0177 C177, 177
100 0034 0377 K377, 377
101 0035 0304 DEVCOD, 0304
102 0036 0007 SAVETS, 0007
103 0037 0004 BITNO, 0004
104 0040 0377 DATBIT, 0377
105 0041 0000 BAUDNO, 0000
106 0042 0000 XMTDAT, 0
107 0043 0000 XMTDT1, 0
108 0044 0000 ERRFLG, 0
109 0045 0000 RECDAT, 0
110 0046 0000 LOOPPC, 0

```

```

111 0047 0000 NDELAY, 0
112 0050 0000 CNT1, 0
113 0051 0000 CNT2, 0
114 0052 0000 TSTCNT, 0
115 0053 0000 SAV2, 0
116 0054 7000 M1000, -1000
117 0055 0252 K0252, 0252
118 0056 0125 K0125, 0125
119 0057 0212 K212, 212
120
121
122 4460 LOAD=JMS I .
123 0060 2160 XLOAD
124 4461 DELAY=JMS I .
125 0061 2620 XDELAY
126 4462 STLPPC=JMS I .
127 0062 4032 XPCRET
128 4463 LOOP=JMS I .
129 0063 4042 XSR2
130 4464 EHLTLP=JMS I .
131 0064 2200 HLTLOP
132 4465 SW10NE=JMS I .
133 0065 4050 NOT8E
134 4466 TSFSKP=JMS I .
135 0066 2671 WATTSF
136 4467 KSFSKP=JMS I .
137 0067 2707 WATKSF
138 4470 LISN=JMS I .
139 0070 3314 XLISN
140 4471 MESSAGE=JMS I .
141 0071 3400 MESAGX
142 4472 ONEOCT=JMS I .
143 0072 3201 ONEOCK
144 4473 TWOOCT=JMS I .
145 0073 3211 TWOOCK
146 4474 FOROCT=JMS I .
147 0074 3464 FOROCK
148 4475 PRNT1=JMS I .
149 0075 3513 XPRNT1
150 4476 PRNT2=JMS I .
151 0076 3224 XPRNT2
152 4477 PRNT4=JMS I .
153 0077 3500 XPRNT4
154 4500 SPACE2=JMS I .
155 0100 3306 SPACK2
156 4501 TYPE=JMS I .
157 0101 3235 XTYPE
158 4502 CRLF=JMS I .
159 0102 3244 XCRLF
160 4503 MIOT=JMS I .
161 0103 3442 XMIOT
162 4504 XOR=JMS I .
163 0104 3261 XORS
164 4505 YESRNO=JMS I .
165 0105 3273 YFSRNX
    
```

/SUBROUTINE CALLS

```

166 4506 RANDOM=JMS I .
167 0106 3537 XRAND
168 4507 SAVGEN=JMS I .
169 0107 3555 XSAVGN
170 4510 RESGEN=JMS I .
171 0110 3564 XRESGN
172 4511 BSWAP=JMS I .
173 0111 3522 XRSW
174 4512 AERROR=JMS I .
175 0112 3671 XAERRO
176 4513 APT8=JMS I .
177 0113 3600 XAPT8
178 4514 TICK=JMS I .
179 0114 4013 XTICK
180 4515 CHEK22=JMS I .
181 0115 4123 XCHK22
182 4516 NERROR=JMS I .
183 0116 3744 XNEPRO
184 4517 LAS=JMS I .
185 0117 4070 XLAS
186 4520 HLT=JMS I .
187 0120 4102 XHALT
188
189 0121 0000 CLKCNT, 0
190 0122 0000 COUNT, 0
191 0123 4000 K4000, 4000
192 0124 0000 TEMP, 0
193 0125 0200 K200, 0200
194
195 0200 *200
196
197 0200 4513 BGNINT, APT8
198
199 0201 5250 NOINTR, JMP START
200 0202 5777 JMP BAUDTM
201 0203 4517 LAS
202 0204 7012 RTR
203 0205 7710 SPA CLA
204 0206 5776 JMP TYINTR
205 0207 4520 HLT
206 0210 4517 LAS
207 0211 3035 DCA DEVCOD
208 0212 4520 HLT
209 0213 4517 LAS
210 0214 3036 DCA SAVBTS
211 0215 1036 TAD SAVBTS
212 0216 0375 AND (3
213 0217 3037 DCA BITNO
214 0220 1374 TAD (TAD K37
215 0221 1037 TAD BITNO
216 0222 3223 DCA ,+1
217 0223 4520 HLT/TAD K37+(X)
218 0224 3040 DCA DATBIT
219 0225 1036 TAD SAVBTS
220 0226 0373 AND (4
    
```

/APT ERROR REPORTER.

/TEST FOR APT.

/TEMP STORAGE FOR APT-8

/TEST FOR APT OR

/BEGIN INTERROGATION FOR SETUP

/GO TO START OF TEST NO INTERROGATION REQUIRED

/BAUD RATE TIMING TEST (THE PROGRAM SHOULD HALT IN 30 SECONDS

/LOOK AT SR11 FOR DESIRED TYPE OF INTERROGATION

/PUT BIT 11 INTO AC0

/IF AC11=1 USE TELETYPE FOR INPUT, OTHERWISE USE THE SR

/GO TO TELETYPE FOR INTERROGATION

/SET BITS 0-5 TO THE RECEIVE IOT AND BITS 6-11 TO TRANSMIT IOT

/GET DEVICE CODE FROM SWITCH REGISTER

/SAVE IT FOR IOT MODIFICATION

/SET PARITY=STATUS-FILLER-BAUD RATE-STOP BITS-AND # OF DATA BITS IN SR

/GET THE SR AND CALCULATE THE RESULTS

/SAVE THEM

/SETUP THE NUMBER OF DATA BITS

/THIS NUMBER=37,77,177,377 FOR 5,6,7 OR 8 DATA BITS

/SET UP LENGTH OF CHARACTER FROM # OF STOP BITS

/IF BIT 9=1 2 STOP BITS;IF 0 ONLY 1 STOP BIT

```

221 0227 7640          SZA CLA
222 0230 2037          ISZ BITNO /ADD 1 MORE TO CHARACTER LENGTH
223 0231 1036          TAD SAVRTS /DOES IT HAVE PARITY
224 0232 7710          SPA CLA
225 0233 2037          ISZ BITNO /YES BUMP THE CHARACTER LENGTH BY 1
226 0234 1036          TAD SAVRTS /SET UP FOR BAUD RATE
227 0235 7012          RTR
228 0236 7010          RAR
229 0237 0372          AND (17
230 0240 3041          DCA BAUDNO /SAVE THE BAUD NUMBER POINTER
231 0241 1041          TAD BAUDNO /IS THE BAUD NUMBER WITHIN LIMITS
232 0242 1371          TAD (-13
233 0243 7740          SMA SZA CLA
234 0244 5212          JMP SETUP-3 /NO,BAUD NUMBER OUT OF BOUNDS GO BACK TO STATUS SETUP
235 0245 4515          CHEK22 /TEST FOR ACTIVE CONSOLE.
236 0246 4770          JMS XC8PSW /ASK SWITCH REGISTER QUESTION.
237 0247 4520          HLT /SET SR TO DESIRED SWITCH SETTINGS
238 0250 1035          START, TAD DEVCOD /GET THE DEVICE CODE
239 0251 7012          RTR /PUT THE RECEIVE DEVICE CODE IN BITS 3-8
240 0252 7010          RAR /
241 0253 4503          MIOT /GO MODIFY THE IOTS
242 0254 5241          RFCPNT /RECEIVE IOT TABLE POINTER
243 0255 1035          TAD DEVCOD /GET THE DEVICE CODES
244 0256 7006          RTL /PUT THE TRANSMIT DEVICE CODE IN BITS 3-8
245 0257 7004          RAL /
246 0260 4503          MIOT /GO MODIFY THEM
247 0261 5352          XMTIOT /POINTER TO TRANSMIT IOT TABLE
248 0262 1367          TAD (JMP I 2 /SET UP INTERRUPT RETURN LOCATIONS
249 0263 3001          DCA 1
250 0264 1366          TAD (INTRET
251 0265 3002          DCA 2
252 0266 5271          JMP CLRBRD /GO START TEST
253
254 0267 7240          INTRET, CLA CMA
255 0270 5400          JMP I 0
256
257
258 /INITIALIZE THE MODULE WITH A CAF INSTRUCTION AND CHECK THAT THE
259 /RECEIVE AND TRANSMIT FLAGS ARE NOT STUCK ON AND THAT KSF,TSF
260 /AND SPI DONT SKIP AND THAT THE INTERRUPT REQUEST LINE
261 /IS NOT PULLED LOW.
262
263 0271 4460          CLRBRD, LOAD /LOAD TIMING VALUE FOR APT IF REQUIRED.
264 0272 4462          STLPCC /SET LOOPING PC FOR TEST AND ERROR LOOPING
265 0273 4465          SW10NE /CHECK SR10 TO SEE IF PROCESSOR A PDP-8E
266 0274 0545          ACNSKP /PROCESSOR NOT A PDP-8E
267 0275 6007          CAF /CLEAR THE BOARD-CHECK THE SKIP IOT'S NOT TO SKIP
268 0276 6031          KSF0, KSF /SKIP ON RECEIVE FLAG
269 0277 7410          SYP
270 0300 4464          EHLTLP /ERROR,RECEIVE FLAG SET OR KSF SKIPPED
271 0301 6041          TSF0, TSF /SKIP ON TRANSMIT FLAG
272 0302 7410          SKP
273 0303 4464          EHLTLP /ERROR,TRANSMIT FLAG SET OR TSF SKIPPED
274 0304 6045          SPI0, SPI /SKIP IF XMIT/RECEIVE FLAG SET WITH INT ENB.
275 0305 7410          SKP

```

```

276 0306 4464          EHLTLP /SPI SKIPPED OR XMIT/RECEIVE FLAG SET WITH INT ENB
277 0307 6001          ION /CHECK THAT INT REQ IS NOT PULLED BY INT. ENB.
278 0310 7000          NOP /AND TRANSMIT/RECEIVE FLAG BEING SET
279 0311 6002          IOF
280 0312 7710          SPA CLA
281 0313 4464          EHLTLP /INT REQ LINE PULLED LOW OR TRANSMIT/RECEIVE FLAG SET
282 0314 4516          NERROR
283 0315 0273          CLRBRD+2
284 0316 4463          LOOP /LOOP IF SR2=1
285
286 /CHECK THAT THE TRANSMIT FLAG CAN BE SET AND CLEARED BY TFL,TSF AND TCF
287
288 0317 4462          SCXMIT, STLPCC /SET LOOPING PC FOR TEST AND ERROR LOOPING
289 0320 6030          KCF0, KCF /CLEAR RECEIVE FLAG FOR SCOPE LOOPING
290 0321 7410          SKP /SAFETY SKIP IN CASE KCF SKIPPED
291 0322 4464          EHLTLP /KCF SKIPPED
292 0323 6040          TFL0, TFL /SET THE TRANSMIT FLAG
293 0324 7410          SKP /SAFETY SKIP TO CHECK TFL NOT TO SKIP
294 0325 4464          EHLTLP /ERROR,TFL SKIPPED
295 0326 6041          TSF1, TSF /SKIP IF TRANSMIT FLAG=1
296 0327 4464          EHLTLP /TRANSMIT FLAG NOT SET OR TFL FAILED
297 0330 6042          TCF0, TCF /CLEAR THE TRANSMIT FLAG
298 0331 7410          SKP /SAFETY SKIP TO CHECK TCF NOT TO SKIP
299 0332 4464          EHLTLP /ERROR, TCF SKIPPED
300 0333 6041          TSF2, TSF /SKIP ON TRANSMIT FLAG
301 0334 7410          SKP
302 0335 4464          EHLTLP /ERROR,TCF FAILED TO CLEAR TRANSMIT FLAG
303 0336 6031          KSF1, KSF /CHECK TO SEE IF RECEIVE FLAG GOT SET
304 0337 7610          SKP CLA
305 0340 4464          EHLTLP /RECEIVE FLAG SET BY SETTING TRANSMIT FLAG
306 0341 4516          NERROR
307 0342 0320          SCXMIT+1
308 0343 4463          LOOP /LOOP IF SR2=1
309
310 /CHECK THAT TRANSMIT FLAG CAN BE CLEARED BY CAF
311
312 0344 4462          CAFXMT, STLPCC /SET LOOPING PC FOR TEST AND ERROR LOOPING
313 0345 6040          TFL1, TFL /SET THE TRANSMIT FLAG
314 0346 6041          TSF3, TSF /SKIP ON TRANSMIT FLAG
315 0347 4464          EHLTLP /ERROR,TRANSMIT FLAG FAILED TO SET
316 0350 6007          CAF /CLEAR ALL FLAGS
317 0351 6041          TSF4, TSF /SKIP ON TRANSMIT FLAG
318 0352 7410          SKP /OK FLAG NOT SET
319 0353 4464          EHLTLP /ERROR,CAF FAILED TO CLEAR TRANSMIT FLAG
320 0354 6031          KSF2, KSF /CHECK TO SEE IF RECEIVE FLAG GOT SET
321 0355 7610          SKP CLA /NO, IT DIDN'T
322 0356 4464          EHLTLP /ERROR RECEIVE FLAG GOT SET
323 0357 4516          NERROR
324 0360 0345          CAFXMT+1
325 0361 4463          LOOP /LOOP IF SR2=1
326 0362 5765          JMP INTXMT
327
328 0365 0400
329 0366 0267
330 0367 5402

```

```

331 0370 6703
332 0371 7765
333 0372 0017
334 0373 0004
335 0374 1031
336 0375 0003
337 0376 3000
338 0377 2264
339 0400 PAGE
340 /
341 /USING THE TRANSMIT FLAG-CHECK THAT INTERRUPT ENABLE CAN BE SET
342 /AND CLEARED AND THAT THE PROGRAM CAN INTERRUPT. INTERRUPT
343 /ENABLE IS SET AND CLEARED BY DATA BIT 11 AND KIE COMMAND.
344
345 0400 4462 INTXMT, STLPPC /SET LOOPING PC FOR TEST AND ERROR LOOPING
346 0401 6030 KCF1, KCF /CLEAR RECEIVE FLAG FOR SCOPE LOOPING
347 0402 7300 CLA
348 0403 6035 KIE0, KIE CLL /CLEAR INTERRUPT ENABLE
349 0404 7410 SKP /SAFETY SKIP TO CHECK KIE NOT TO SKIP
350 0405 4464 EHLTLP /ERROR, KIE SKIPPED
351 0406 6040 TFL2, TFL /SET THE TRANSMIT FLAG
352 0407 6041 TSF5, TSF /SKIP ON TRANSMIT FLAG
353 0410 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET
354 0411 6045 SPI1, SPI /SKIP IF INTERRUPT ENABLE AND XMIT FLAG SET.
355 0412 7410 SKP /OK, INTERRUPT ENABLE NOT SET
356 0413 4464 EHLTLP /ERROR, INT ENB, SET, FAILED TO BE CLEARED BY KIE
357 0414 6001 ION /CHECK THAT INTERRUPT REQUEST IS NOT PULLED
358 0415 7000 NOP /INTERRUPT HERE IF SET
359 0416 6002 IOF /TURN THE INTERRUPT OFF
360 0417 7710 SPA CLA
361 0420 4464 EHLTLP /PROGRAM INTERRUPTED-CHECK INT. ENB.
362 0421 7001 IAC
363 0422 6035 KIE1, KIE /SET INT. ENB, WITH DATA BIT 11 AND KIE
364 0423 7610 SKP CLA
365 0424 4464 EHLTLP /PRROR, KIE SKIPPED
366 0425 6041 TSF6, TSF /SKIP ON TRANSMIT FLAG
367 0426 4464 EHLTLP /ERROR, TRANSMIT FLAG GOT CLEARED
368 0427 6045 SPI2, SPI /SKIP ON INT ENB AND TRANSMIT FLAG
369 0430 4464 EHLTLP /SPI FAILED TO SKIP OR INT ENB NOT SET
370 0431 6001 ION /CHECK THAT INTERRUPT REQUEST IS PULLED
371 0432 7000 NOP /SHOULD INTERRUPT HERE
372 0433 6002 IOF /TURN IT OFF
373 0434 7700 SMA CLA
374 0435 4464 EHLTLP /DID IT INTERRUPT?
375 0436 6042 TCF1, TCF /FAILED TO INTERRUPT-CHECK XMIT AND INT ENB
376 0437 6041 TSF7, TSF /CLEAR THE TRANSMIT FLAG
377 0440 7410 SKP /CHKCK TO SEE IF IT CLEARED
378 0441 4464 EHLTLP /IT FAILED TO CLEAR
379 0442 6045 SPI3, SPI /SKIP ON INT ENB AND TRANSMIT FLAG
380 0443 7410 SKP
381 0444 4464 EHLTLP /TRANSMIT FLAG IS GONE IT SHOULDN'T SKIP
382 0445 6001 ION /CHECK THAT IT DOESN'T INTERRUPT
383 0446 7000 NOP /
384 0447 6002 IOF /

```

```

385 0450 7710 SPA CLA
386 0451 4464 EHLTLP /PROGRAM INTERRUPTED WITHOUT TRANSMIT FLAG
387 0452 6040 TFL3, TFL /SET THE FLAG AGAIN
388 0453 6041 TSF8, TSF /SKIP ON THE TRANSMIT FLAG
389 0454 4464 EHLTLP /FLAG FAILED TO SET
390 0455 6045 SPI4, SPI /SKIP ON XMIT AND INT. ENB.
391 0456 4464 EHLTLP /FAILED TO SKIP ON INT ENB AND XMIT FLAG
392 0457 6035 KIE2, KIE /CLEAR INTERRUPT ENABLE WITH KIE AND DATA BIT 11
393 0460 6045 SPI5, SPI /SKIP IF INT ENB=1 WITH XMIT FLAG
394 0461 7410 SKP
395 0462 4464 EHLTLP /KIE FAILED TO CLEAR INTERRUPT ENABLE
396 0463 6001 ION /CHECK THAT THE PPROGRAM DOESN'T INTERRUPT
397 0464 7000 NOP
398 0465 6002 IOF
399 0466 7710 SPA CLA
400 0467 4464 EHLTLP /PROGRAM INTERRUPTED WITHOUT INT ENB
401 0470 6042 TCF2, TCF /CLEAR TRANSMIT FLAG
402 0471 6031 KSF3, KSF /CHECK TO SEE IF RECEIVE FAG GOT SET
403 0472 7610 SKP CLA
404 0473 4464 EHLTLP /RECEIVE FLAG SET BY ABOVE CODE
405 0474 4516 NERROR
406 0475 0401 INTXMT+1
407 0476 4463 LOOP /LOOP ON TEST IF SR2=1
408 0477 5300 JMP CAFINT
409
410
411
412 /CHECK THAT CAF WILL SET INTERRUPT ENABLE USING THE TRANSMIT
413 /FLAG TO SKIP AND INTERRUPT ON.
414
415 0500 4462 CAFINT, STLPPC /SET THE LOOPING PC FOR TEST AND ERROR LOOPING.
416 0501 6030 KCF2, KCF /CLEAR RECEIVE FLAG
417 0502 6035 KIE3, KIE /CLEAR INTERRUPT ENABLE
418 0503 6040 TFL4, TFL /SET THE TRANSMIT FLAG
419 0504 6041 TSF9, TSF /SKIP ON THE TRANSMIT FLAG
420 0505 4464 EHLTLP /FLAG FAILED TO SET
421 0506 6045 SPI6, SPI /SKIP ON TRANSMIT FLAG AND INT ENB.
422 0507 7410 SKP
423 0510 4464 EHLTLP /SPI SKIPPED OR INT ENB IS SET
424 0511 6007 CAF /CLEAR TRANSMIT FLAG AND SET INT ENB.
425 0512 6041 TSF10, TSF /SKIP IF TRANSMIT FLAG = 1
426 0513 7410 SKP
427 0514 4464 EHLTLP /CAF FAILED TO CLEAR XMIT FLAG
428 0515 6045 SPI7, SPI /SKIP ON TRANSMIT FLAG AND INT ENB.
429 0516 7410 SKP
430 0517 4464 EHLTLP /SPI SKIPPED WITHOUT TRANSMIT FLAG
431 0520 6040 TFL5, TFL /SET THE TRANSMIT FLAG
432 0521 6041 TSF11, TSF /SKIP IF XMIT FLAG IS SET
433 0522 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET
434 0523 6045 SPI8, SPI /SKIP ON INTERRUPT ENABLE AND TRANSMIT FLAG
435 0524 4464 EHLTLP /CAF FAILED TO SET INTERRUPT ENABLE
436 0525 6001 ION /CHECK THAT THE PROGRAM WILL INTERRUPT
437 0526 7000 NOP /GO AND INTERRUPT
438 0527 6002 IOF /TURN THE INTERRUPT OFF IF IT DIDN'T
439 0530 7700 SMA CLA

```



```

440 0531 4464 EHLTLP /PROGRAM FAILED TO INTERRUPT WITH XMIT AND INT ENB
441 0532 6042 TCF3, TCF /CLEAR THE TRANSMIT FLAG
442 0533 6041 TSF12, TSF /SKIP IF TRANSMIT FLAG IS SET
443 0534 7610 SKP CLA /IS FLAG SET
444 0535 4464 EHLTLP /FLAG FAILED TO CLEAR
445 0536 6035 KIF4, KIE /CLEAR INTERRUPT ENABLE
446 0537 6031 KSF4, KSF /CHECK TO SEE IF RECEIVE FLAG IS SET
447 0540 7610 SKP CLA
448 0541 4464 EHLTLP /RECEIVE FLAG GOT SET DURING TEST
449 0542 4516 NERROR
450 0543 0501 CAFINT+1
451 0544 4463 LOOP
452
453 /THE FOLLOWING TEST CHECKS THE EFFECT OF THE IOT ON THE AC
454 /AND ALSO CHECKS THAT THE IOTS DON'T SKIP. TPC AND TLS ARE NOT TESTED.
455
456 0545 4462 ACNSKP, STLPPC /SET THE LOOPING PC FOR TEST AND ERROR LOOPING
457 0546 4465 SW10NE /CHECK TO SEE IF PROCESSOR A PDP8E
458 0547 0600 KSF5-1 /NOT A PDP-8E GO TO NEXT SUBTEST
459 0550 7240 CLA CMA /SET AC TO ALL ONE'S
460 0551 6030 KCF3, KCF /CLEAR THE RECEIVE FLAG
461 0552 7410 SKP
462 0553 4777* JMS HLTLP1 /ERROR,KCF SKIPPED
463 0554 7040 CMA /SET THE AC BACK TO ZEROES
464 0555 7440 SZA
465 0556 4776* JMS HLTLP2 /ERROR,KCF CHANGED THE AC
466 0557 5775* JMP KSF5-1
467
468 /
469 0575 0600
470 0576 2241
471 0577 2217
472 0600
473 0600 7240 CLA CMA /SET THE AC BACK TO 1'S
474 0601 6031 KSF5, KSF /SKIP ON RECEIVE FLAG
475 0602 7410 SKP
476 0603 4777* JMS HLTLP1 /ERROR,RECEIVE FLAG SHOULD NOT BE SET
477 0604 7040 CMA /SET THE AC BACK TO ZEROES
478 0605 7440 SZA
479 0606 4776* JMS HLTLP2 /ERROR,KSF CHANGED THE AC
480 0607 7240 CLA CMA /SET THE AC TO ALL 1'S
481 0610 6032 KCC0, KCC /CLEAR AC AND RECEIVE FLAG AND SET HEADER RUN
482 0611 7410 SKP
483 0612 4777* JMS HLTLP1 /ERROR, KCC SKIPPED
484 0613 7440 SZA
485 0614 4776* JMS HLTLP2 /ERROR,KCC FAILED TO CLEAR AC
486 0615 1375 TAD (=4 /SET AC TO ALL 1'S EXCEPT BITS 10 AND 11
487 0616 6035 KIE5, KIE /CLEAR INTERRUPT AND STATUS ENABLE
488 0617 7410 SKP
489 0620 4777* JMS HLTLP1 /ERROR, KIE SKIPPED
490 0621 1374 TAD (3 /ADD 3 TO AC AND THEN COMPLEMENT IT
491 0622 7040 CMA
492 0623 7440 SZA
493 0624 4776* JMS HLTLP2 /ERROR, KIE CHANGED THE AC

```

```

494 0625 7240 CLA CMA /SET THE AC = TO ALL 1'S
495 0626 6034 KRS0, KRS /READ RECEIVE BUFFER STATIC AND STATUS
496 0627 7410 SKP
497 0630 4777* JMS HLTLP1 /ERROR, KRS SKIPPED
498 0631 7040 CMA /SET THE AC BACK TO ZEROES
499 0632 7440 SZA
500 0633 4776* JMS HLTLP2 /ERROR,KRS CHANGED THE AC
501 0634 1177 TAD (7400 /SET AC BITS 0-3
502 0635 6036 KRB0, KRB /CLEAR AC AND RECEIVE FLAG AND READ RECEIVE BUF
503 0636 7410 SKP
504 0637 4777* JMS HLTLP1 /ERROR, KRB SKIPPED
505 0640 0177 AND (7400
506 0641 7440 SZA
507 0642 4776* JMS HLTLP2 /KRB FAILED TO CLEAR THE AC
508 0643 4465 SW10NE /CHECK TO SEE IF PDP-8E
509 0644 0654 TCF4-1 /PROCESSOR NOT A PDP-8E GO DO NEXT SUBTEST
510 0645 7240 CLA CMA /SET AC EQUAL TO ALL ONES
511 0646 6040 TFL6, TFL /SET THE TRANSMIT FLAG
512 0647 7410 SKP
513 0650 4777* JMS HLTLP1 /ERROR, TFL SKIPPED
514 0651 7040 CMA /SET THE AC BACK TO 0
515 0652 7440 SZA
516 0653 4776* JMS HLTLP2 /TFL CHANGED THE AC
517 0654 7240 CLA CMA /SET THE AC TO 1'S
518 0655 6042 TCF4, TCF /CLEAR THE TRANSMIT FLAG
519 0656 7410 SKP
520 0657 4777* JMS HLTLP1 /ERROR,TCF SKIPPED
521 0660 7040 CMA /SET THE AC BACK TO 0
522 0661 7440 SZA
523 0662 4776* JMS HLTLP2 /TCF CHANGED THE AC
524 0663 7240 CLA CMA /SET THE AC TO ALL 1'S
525 0664 6041 TSF13, TSF /SKIP ON TRANSMIT FLAG
526 0665 7410 SKP
527 0666 4777* JMS HLTLP1 /TRANSMIT FLAG IS SET
528 0667 7040 CMA /SET THE AC BACK TO 0
529 0670 7440 SZA
530 0671 4776* JMS HLTLP2 /ERROR,TSF CHANGED THE AC
531 0672 7240 CLA CMA /SET THE AC TO ALL 1'S
532 0673 6045 SPI9, SPI /SKIP IF XMT/REC + INT ENB =1
533 0674 7410 SKP
534 0675 4777* JMS HLTLP1 /ERROR,SPI SKIPPED OR XMT/REC AND INT ENB =1
535 0676 7040 CMA /SET THE AC BACK TO ZERO
536 0677 7440 SZA
537 0700 4776* JMS HLTLP2 /ERROR,SPI CHANGED THE AC
538 0701 4516 NERROR
539 0702 0546 ACNSKP+1
540 0703 4463 LOOP
541
542
543
544 /START OF LOOP BACK TEST
545 /CHECK THAT THE TRANSMIT FLAG CAN BE SET BY TPC AND THAT DATA
546 /AVAILABLE WILL SET THE RECEIVE FLAG. CHECK THAT THE FLAGS CAN
547 /BE CLEARED BY TCF AND KCC. CHECK THAT THE FLAGS CAN CAUSE AN
548 /INTERRUPT BY MANIPULATING INTERRUPT ENABLE.

```

```

549
550 0704 4462 STFLGS, STLPPC /STORE LOOPING PC FOR TEST AND SCOPE LOOPING
551 0705 1176 TAD [-3720 /SET UP A DELAY OF 200MS TO ALLOW FLAGS TO SETTLE
552 0706 3047 DCA NDELAY
553 0707 4461 DELAY /WAIT FOR 200MS
554 0710 6035 KIE6, KIE /CLEAR INTERRUPT ENABLE
555 0711 7610 SKP CLA /SAFETY SKIP TO CHECK KIE NOT TO SKIP
556 0712 4464 EHLTLP /ERROR,KIE SKIPPED
557 0713 6032 KCC1, KCC /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
558 0714 7610 SKP CLA /SAFETY SKIP TO CHECK KCC NOT TO SKIP
559 0715 4464 EHLTLP /ERROR,KCC SKIPPED
560 0716 6042 TCF5, TCF /CLEAR TRANSMIT FLAG
561 0717 7610 SKP CLA /SAFETY SKIP TO CHECK TCF NOT TO SKIP
562 0720 4464 EHLTLP /ERROR,TCF SKIPPED
563 0721 6031 KSF6, KSF /CHECK THE RECEIVE FLAG TO BE ZERO
564 0722 7610 SKP CLA
565 0723 4464 EHLTLP /ERROR,RECEIVE FLAG =1 OR KSF SKIPPED
566 0724 6041 TSF14, TSF /SKIP IF TRANSMIT FLAG =1
567 0725 7610 SKP CLA
568 0726 4464 EHLTLP /ERROR,TRANSMIT FLAG=1 OR TSF SKIPPED
569 0727 6044 TPC0, TPC /LOAD TRANSMIT BUFFER AND TRANSMIT
570 0730 7610 SKP CLA /SAFETY SKIP TO CHECK TPC NOT TO SKIP
571 0731 4464 EHLTLP /ERROR,TPC SKIPPED
572 0732 4466 TSFSKP /WAIT FOR A SECOND FOR TRANSMIT FLAG TO SET
573 0733 4464 EHLTLP /ERROR,TPC FAILED TO SET XMIT FLAG OR TSF FAILED
574 0734 6031 KSF7, KSF /CHECK THE RECEIVE FLAG TO STILL BE A 0
575 0735 7610 SKP CLA
576 0736 4464 EHLTLP /RECEIVE FLAG GOT SET TO SOON
577 0737 6045 SPI10, SPI /SKIP IF XMIT/RECEIVE FLAG=1 AND INT ENB SET
578 0740 7610 SKP CLA
579 0741 4464 EHLTLP /ERROR,SPI SKIPPED OR INTERRUPT ENABLE SET
580 0742 6001 ION /CHECK THE PROGRAM NOT TO INTERRUPT
581 0743 7000 NOP
582 0744 6002 IOF
583 0745 7710 SPA CLA
584 0746 4464 EHLTLP
585 0747 7301 KIE7, KIE /FROR,INT ENB SET OR INT REQ PULLED LOW
586 0750 6035 SPI11, SPI /SET INTERRUPT ENABLE TO A 1
587 0751 6045 EHLTLP /AC11=1 AND KIE SET INTERRUPT ENABLE
588 0752 4464 ION /SKIP IF XMIT/RECEIVE FLAG=1 WITH INT ENABLE
589 0753 6001 NOP /INTERRUPT ENABLE FAILED TO SET OR KIE FAILED
590 0754 7000 IOF /CHECK THE PROGRAM TO INTERRUPT
591 0755 6002 IOF /IT SHOULD INTERRUPT HERE
592 0756 7700 SMA /TURN IT OFF
593 0757 4464 EHLTLP CLA
594 0760 6035 KIE8, KIE /ERROR PROGRAM FAILED TO INTERRUPT WITH XMIT AND INT ENABLE
595 0761 6045 SPI12, SPI /SET INT ENB=0 WITH AC11=0 AND KIE COMMAND
596 0762 7610 SKP CLA /CHECK THAT INT ENB CLEARED BY KIE
597 0763 4464 EHLTLP
598 0764 6001 ION /ERROR,INT ENB FAILED TO CLEAR OR SPI SKIPPED
599 0765 7000 NOP /CHECK THE PROGRAM NOT TO INTERRUPT
600 0766 6002 IOF
601 0767 7710 SPA CLA
602 0770 4464 EHLTLP /ERROR PROGRAM INTERRUPTED WITHOUT INT ENABLE
603 0771 5773 JMP TSF16

```

```

604
605 0773 1000 /
606 0774 0003 /
607 0775 7774 /
608 0776 2241 /
609 0777 2217 /
610 / PAGE
611 1000 6041 TSF16, TSF /CHECK THE TRANSMIT FLAG TO STILL = 1
612 1001 4464 EHLTLP /SOMETHING CLEARED THE TRANSMIT FLAG
613 1002 6042 TCF6, TCF /CLEAR THE TRANSMIT FLAG
614 1003 6041 TSF17, TSF /SKIP IF TRANSMIT FLAG =1
615 1004 7610 SKP CLA
616 1005 4464 EHLTLP /ERROR,TCF FAILED TO CLEAR XMIT FLAG
617 1006 4467 KSF5SKP /WAIT FOR ABOUT A SECOND FOR RECEIVE FLAG TO SET
618 1007 4464 EHLTLP /ERROR,RECEIVE FLAG=0 OR DATA AVAILABLE FAILED TO SET RECEIVE FLAG
619 1010 6034 KRS1, KRS /CHECK THAT KRS DOESN'T CLEAR RECEIVE FLAG
620 1011 7610 SKP CLA /SAFETY SKIP TO CHECK KRS NOT TO SKIP
621 1012 4464 EHLTLP /ERROR,KRS SKIPPED
622 1013 6031 KSF9, KSF /SKIP ON RECEIVE FLAG
623 1014 4464 EHLTLP /KRS CLEARED RECEIVE FLAG
624 1015 6045 SPI13, SPI /SKIP IF XMIT/RECEIVE FLAG AND INT ENABLE=1
625 1016 7610 SKP CLA
626 1017 4464 EHLTLP /ERROR SPI SKIPPED OR INT ENABLE=1
627 1020 6001 ION /CHECK THE PROGRAM NOT TO INTERRUPT
628 1021 7000 NOP
629 1022 6002 IOF
630 1023 7710 SPA CLA
631 1024 4464 EHLTLP /PROGRAM INTERRUPT WITHOUT INTERRUPT ENABLE
632 1025 7301 KIE9, KIE /SET INT ENB F/F=1
633 1026 6035 SPI14, SPI /SKIP IF RECEIVE AND INT ENB=1
634 1027 6045 EHLTLP /ERROR,SPI FAILED OR RECEIVE/INT ENB NOT= TO A 1
635 1030 4464 ION /CHECK THE PROGRAM TO INTERRUPT
636 1031 6001 NOP
637 1032 7000 IOF
638 1033 6002 SMA CLA
639 1034 7700 EHLTLP
640 1035 4464 EHLTLP /ERROR,FAILED TO INTEPRUPT WITH INT ENB AND RECEIVE FLAG = A 1
641 1036 6032 KCC2, KCC /CLEAR AC AND RECEIVE FLAG AND SET READER RUN
642 1037 6031 KSF10, KSF /SKIP IF RECEIVE FLAG =1
643 1040 7610 SKP CLA
644 1041 4464 EHLTLP /ERROR,KCC FAILED TO CLEAR RECEIVE FLAG
645 1042 6045 SPI15, SPI /SKIP IF INT ENB AND RECEIVE FLAG =1
646 1043 7610 SKP CLA
647 1044 4464 EHLTLP /ERROR,SPI SKIPPED WITHOUT RECEIVE FLAG = 1
648 1045 6001 ION /CHECK THE PROGRAM NOT TO INTERRUPT
649 1046 7000 NOP
650 1047 6002 IOF
651 1050 7710 SPA CLA
652 1051 4464 EHLTLP /INTERRUPTED WITHOUT RECEIVE FLAG SET
653 1052 6035 KIE10, KIE /CLEAR INTERRUPT ENABLE
654 1053 7300 CLA
655 1054 4516 NERROR
656 1055 0710 STFLGS+4
657 1056 4463 LOOP /LOOP ON TEST IF SR2=1

```

```

658 /
659 /THE FOLLOWING TEST CHECKS THAT A TPC COMMAND WILL SET THE
660 /TRANSMIT FLAG AND THAT A TIS WILL CLEAR THE FLAG AND THEN RESET
661 /IT. CHECK THAT THE RECEIVE FLAG WILL GET SET FROM A TPC AND TIS
662 /COMMAND AND THAT IT CAN BE CLEARED BY A KRB AND KCC OR KCF COMMAND
663
664 1057 4462 XMTREC, STLPPC /STORE LOOPING PC FOR TEST AND ERROR LOOPING
665 1060 1176 TAD [-3720 /DELAY 200MS FOR SCOPE LOOPING TO LET
666 1061 3047 DCA NDELAY /FLAGS SETTLE
667 1062 4461 DELAY /GO DELAY 200MS
668 1063 6032 KCC3, KCC /CLEAR AC AND RECEIVE FLAG
669 1064 6042 TCF /CLEAR THE TRANSMIT FLAG
670 1065 6031 KSF11, KSF /CHECK THE RECEIVE FLAG TO BE 0
671 1066 7610 SKP CLA
672 1067 4464 EHLTLP /RECEIVE FLAG STILL = 1 AFTER A KCC COMMAND
673 1070 6041 TSF18, TSF /SKIP IF TRANSMIT FLAG = 1
674 1071 7610 SKP CLA
675 1072 4464 EHLTLP /TRANSMIT FLAG STILL A 1 AFTER A TCF COMMAND
676 1073 6044 TPC1, TPC /LOAD TRANSMIT BUFFER AND TRANSMIT
677 1074 7200 CLA
678 1075 4466 TSFSKP /WAIT FOR THE FLAG TO SET
679 1076 4464 EHLTLP /ERROR XMIT FLAG FAILED TO SET BY TPC
680 1077 6031 KSF12, KSF /CHECK THE RECEIVE FLAG TO STILL BE 0
681 1100 7610 SKP CLA
682 1101 4464 EHLTLP /RECEIVE FLAG SET TO SOON
683 1102 6046 TLS0, TIS /LOAD TRANSMIT BUFFER AND TRANSMIT AND CLEAR FLAG
684 1103 7200 CLA
685 1104 7610 SKP CLA /SAFETY SKIP TO CHECK TIS NOT TO SKIP
686 1105 4464 EHLTLP /ERROR, TIS SKIPPED
687 1106 6041 TSF20, TSF /SKIP IF TRANSMIT FLAG = 1
688 1107 7610 SKP CLA
689 1110 4464 EHLTLP /ERROR, TIS FAILED TO CLEAR TRANSMIT FLAG
690 1111 4467 KSF5KP /WAIT FOR RECEIVE FLAG TO SET FROM FIRST XMIT
691 1112 4464 EHLTLP /ERROR, REC FLAG FAILED TO SET FROM FIRST XMIT
692 1113 6036 KRB1, FRB /CLEAR AC AND RECEIVE FLAG AND READ RECEIVE BUFF
693 1114 7610 SKP CLA /SAFETY SKIP TO CHECK KRB NOT TO SKIP
694 1115 4464 EHLTLP /ERROR, KRB SKIPPED
695 1116 6031 KSF14, KSF /SKIP ON RECEIVE FLAG
696 1117 7610 SKP CLA
697 1120 4464 EHLTLP /ERROR, KRB FAILED TO CLEAR RECEIVE FLAG
698 1121 4466 TSFSKP /WAIT FOR TRANSMIT FLAG TO SET FROM 2ND XMIT
699 1122 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET FROM TIS COMMAND
700 1123 6042 TCF8, TCF /CLEAR THE TRANSMIT FLAG
701 1124 6041 TSF22, TSF /SKIP IF TRANSMIT FLAG SET
702 1125 7610 SKP CLA
703 1126 4464 EHLTLP /ERROR, TCF FAILED TO CLEAR FLAG
704 1127 4467 KSF5KP /WAIT FOR RECEIVE FLAG TO SET FROM TIS COMMAND
705 1130 4464 EHLTLP /ERROR, RECEIVE FLAG FAILED TO SET FROM 2ND TRANSMIT
706 1131 4517 LAS /CHECK TO SEE IF PDP-8E
707 1132 7010 RAR
708 1133 7012 RTR
709 1134 7700 SMA CLA
710 1135 5343 JMP .+6 /PDP8E
711 1136 6032 KCC4, KCC /CLEAR THE AC AND RECEIVE FLAG
712 1137 6031 KSF16, KSF /SKIP IF RECEIVE FLAG = 1

```

```

713 1140 7610 SKP CLA
714 1141 4464 EHLTLP /ERROR, KCC FAILED TO CLEAR RECEIVE FLAG
715 1142 5347 JMP .+5
716 1143 6030 KCF4, KCF /CLEAR THE RECEIVE FLAG
717 1144 6031 KSF17, KSF /SKIP IF RECEIVE FLAG SET
718 1145 7610 SKP CLA
719 1146 4464 EHLTLP /ERROR, KCF FAILED TO CLEAR FLAG
720 1147 4516 NERROR
721 1150 1063 XMTREC+4
722 1151 4463 LOOP /LOOP IF SR2=1
723 /
724 /START OF DATA TEST-TRANSMIT 1 WORD AND THEN WAIT FOR THE
725 /RECEIVE FLAG TO SET
726
727 /DATA TEST 1 = TRANSMIT A ROTATING BIT AND CHECK THAT IT CAME BACK
728
729 1152 4462 SDTST1, STLPPC /STORE LOOPING PC FOR TEST LOOPING
730 1153 3042 DCA XMTDAT
731 1154 4460 LOAD /LOAD TIMING FOR APT IF REQUIRED.
732 1155 7120 CLL CML /SET LINK
733 1156 1042 TAD XMTDAT
734 1157 7004 RAL /AND SHIFT LEFT ONE
735 1160 3042 DCA XMTDAT /SET TRANSMIT WORD
736 1161 4777 JMS SLWDAT /GO TRANSMIT AND RECEIVE 1 WORD
737 1162 4516 NERROR
738 1163 1156 .-5
739 1164 4463 LOOP /LOOP IF SR2=1
740 1165 5766 JMP I .+1
741 1166 1200 SDTST2
742 /
743 /
744 1177 1333 PAGE
745 1200 /
746 /DATA TEST 2 = TRANSMIT ALL ONES AND CHECK THAT 1'S COME BACK
747 /
748 1200 4462 SDTST2, STLPPC /STORE LOOPING PC FOR TEST LOOPING
749 1201 7240 CLA CMA
750 1202 0040 AND DATBIT /MASK OUT FOR THE NUMBER OF DATA BITS
751 1203 3042 DCA XMTDAT /SAVE THE WORD FOR TRANSMITTING
752 1204 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
753 1205 4516 NERROR
754 1206 1204 .-2
755 1207 4463 LOOP /LOOP ON TEST IF SR2=1
756 /
757 /DATA TEST 3 = TRANSMIT ONES AND ZEROES
758 /
759 1210 4462 SDTST3, STLPPC /STORE LOOPING PC FOR TEST LOOPING
760 1211 3042 DCA XMTDAT /SAVE WORD
761 1212 7120 CLL CML /SET LINK
762 1213 1042 TAD XMTDAT
763 1214 7004 RAL /AND SHIFT LEFT
764 1215 3042 DCA XMTDAT /SET TRANSMIT WORD
765 1216 4333 JMS SLWDAT /GO TRANSMIT AND RECEIVE ONE WORD
766 1217 1042 TAD XMTDAT /COMPLEMENT DATA WORD

```

```

767 1220 7040          CMA
768 1221 0040          AND   DATBIT
769 1222 3042          DCA   XMTDAT      /SET TRANSMIT WORD
770 1223 4333          JMS   SLWDAT      /GO TRANSMIT AND CHECK IT
771 1224 1042          TAD   XMTDAT
772 1225 7040          CMA
773 1226 0040          AND   DATBIT
774 1227 3042          DCA   XMTDAT
775 1230 4516          NERROR
776 1231 1213          SDTST3+3
777 1232 4463          LOOP
778
779
780
781 1233 4462          SDTST4, STLPCC      /STORE LOOPING PC FOR TEST LOOPING
782 1234 4460          LOAD
783 1235 1040          TAD   DATBIT      /SET UP WORD COUNTER FROM THE # OF DATA BITS
784 1236 7040          CMA
785 1237 3052          DCA   TSTCNT      /SAVE IT
786 1240 3042          DCA   XMTDAT      /CLEAR THE TRANSMIT WORD
787 1241 4333          JMS   SLWDAT      /GO TRANSMIT AND RECEIVE ONE WORD
788 1242 2042          ISZ   XMTDAT      /INCREMENT THE TRANSMIT WORD
789 1243 1042          TAD   XMTDAT
790 1244 4575          JMS I [FILCHK      /GO CHECK FILLER CHAP FOR LF IF SELECTED
791 1245 5242          JMP   ,=3          /FIL IS SELECTED AND ITS A LF GO GET NEW WORD
792 1246 4516          NERROR
793 1247 1241          ,=-6
794 1250 4463          LOOP
795
796
797
798 1251 4462          SDTST5, STLPCC      /STORE LOOPING PC FOR TEST LOOPING
799 1252 4460          LOAD
800 1253 1040          TAD   DATBIT      /TIMING VALUE FOR APT.
801 1254 7040          CMA
802 1255 3052          DCA   TSTCNT      /SET UP WC FROM NUMBER OF DATA BITS
803 1256 3042          DCA   XMTDAT      /SAVE IT
804 1257 4333          JMS   SLWDAT      /CLEAR THE TRANSMIT WORD
805 1260 1042          TAD   XMTDAT      /GO TRANSMIT AND RECEIVE ONE WORD
806 1261 7040          CMA
807 1262 0040          AND   DATBIT      /SET THE TRANSMIT WORD TO ITS COMPLEMENT
808 1263 3042          DCA   XMTDAT
809 1264 1042          TAD   XMTDAT      /SAVE THE NEW WORD
810 1265 4575          JMS I [FILCHK
811 1266 7410          SKP
812 1267 4333          JMS   SLWDAT      /GO TRANSMIT AND RECEIVE ONE WORD
813 1270 1042          TAD   XMTDAT      /RESET THE WORD BACK AND ADD ONE
814 1271 7041          CIA
815 1272 0040          AND   DATBIT
816 1273 3042          DCA   XMTDAT
817 1274 1042          TAD   XMTDAT
818 1275 4575          JMS I [FILCHK
819 1276 5257          JMP   SDTST5+6
820 1277 4516          NERROR
821 1300 1256          SDTST5+5

```

```

822 1301 4463          LOOP
823
824
825
826 1302 4462          SDTST6, STLPCC      /STORE THE LOOPING PC FOR TEST LOOPING
827 1303 1055          TAD   K0252
828 1304 0040          AND   DATBIT      /MASK OUT FOR NUMBER OF BITS
829 1305 3042          DCA   XMTDAT      /SAVE IT
830 1306 4333          JMS   SLWDAT      /GO TRANSMIT AND CHECK THE WORD
831 1307 1056          TAD   K0125
832 1310 0040          AND   DATBIT      /GET THE COMPLEMENTING WORD
833 1311 3042          DCA   XMTDAT      /MASK OUT FOR NUMBER OF BITS
834 1312 4333          JMS   SLWDAT      /SAVE IT
835 1313 4516          NERROR
836 1314 1303          SDTST6+1
837 1315 4463          LOOP
838
839
840
841 1316 4462          SDTST7, STLPCC      /STORE LOOPING PC FOR TEST LOOPING
842 1317 4460          LOAD
843 1320 1054          TAD   M1000
844 1321 3052          DCA   TSTCNT      /TIMING & OR APT
845 1322 4506          RANDOM
846 1323 1042          TAD   XMTDAT
847 1324 4575          JMS I [FILCHK      /CHECK FOR FILLER CHARACTERS
848 1325 5320          JMP   ,=5
849 1326 4333          JMS   SLWDAT      /GO TRANSMIT THE RANDOM NUMBER AND CHECK IT
850 1327 4516          NERROR
851 1330 1322          ,=-6
852 1331 4463          LOOP
853 1332 5574          JMP I [FDATAT
854
855
856
857 1333 0000          SLWDAT, 0
858 1334 7326          CLA CLL CML RTL      /SET STATUS WORD ENABLE
859 1335 6035          KSE0,
860 1336 7200          CLA
861 1337 1042          TAD   XMTDAT
862 1340 6046          SLWTL5, TLS
863 1341 6031          KSF24, KSF
864 1342 7410          SKP
865 1343 5350          JMP   SLWREC      /GET THE RECEIVE WORD
866 1344 6041          TSF32, TSF
867 1345 5341          JMP   ,=-4
868 1346 6042          TCF14, TCF
869 1347 5341          JMP   SLWTL5+1      /SKIP IF TRANSMIT FLAG SET
870
871 1350 7240          SLWREC, CLA
872 1351 6036          KRB11, KRB
873 1352 3045          DCA   RECDAT      /PEAD RECEIVE BUFFER AND CLEAR RECEIVE FLAG
874 1353 1045          TAD   RECDAT      /SAVE THE WORD RECEIVED
875 1354 7041          CIA
876 1355 1042          TAD   XMTDAT      /COMPARE THE WORD WITH THE WORD TRANSMITTED

```

```

877 1356 7650 SNA CLA
878 1357 5372 JMP SLOLAS /GO GET THE NEXT WORD
879 1360 4512 AFRORR /TEST FOR APT ERROR CONDITION
880 1361 5372 JMP SLOLAS /ACTIVE CONSOLE
881 1362 4517 LAS
882 1363 7700 SMA CLA /HALT ON ERROR SWITCH SET?
883 1364 5372 JMP SLOLAS /NO, GO TEST FOR LOOP CONDITION
884 1365 1042 TAD XMTDAT
885 1366 7402 HALT /GOOD DATA WORD IN AC=WORD TRANSMITTED
886 1367 7200 CLA
887 1370 1045 TAD RECDAT
888 1371 7402 HALT /AC=RAD DATA WORD=WORD RECEIVED
889 1372 4517 SLOLAS, LAS /CHECK SRI FOR LOOP ON ERROR
890 1373 7004 RAL /PUT BIT 1 IN AC0
891 1374 7710 SPA CLA /LOOP?
892 1375 5372 JMP SLWILS-1 /YES,GO TRANSMIT AND RECEIVE SAME WORD
893 1376 5733 SLOLST, JMP I SLWDAT /RETURN FOR THE NEXT WORD
894
895
896 1400 PAGE
897
898 /FASTER DATA TEST - TRANSMIT-TRANSMIT-RECEIVE-TRANSMIT-RECEIVE-ETC
899
900 1400 4462 FDATAT, STLPPC /STORE LOOPING PC FOR TEST LOOPING
901 1401 4460 LOAD /TIMING FOR APT=8
902 1402 1176 TAD [-3720
903 1403 3047 DCA NDELAY /SETUP A DELAY OF 200 MS FOR SCOPE LOOPING
904 1404 1054 TAD M1000
905 1405 3052 DCA TSTCNT /SETUP A TEST LOOP OF 1000 WORDS
906 1406 3044 DCA ERRFLG /CLEAR THE PROGRAM ERROR FLAG
907 1407 4461 DELAY /DELAY 200MS
908 1410 6036 KRB2, KRB /ISSUE A KRB TO CLEAR ANY RECEIVE FLAG SET
909 1411 7200 CLA
910 1412 6042 TCF9, TCF /CLEAR THE TRANSMIT FLAG IF SET
911 1413 4506 RANDOM /GENERATE A RANDOM NUMBER
912 1414 1042 TAD XMTDAT
913 1415 4575 JMS I IFILCHK /CHECK TO SEE IF FILL CHAR OPTION SELECTED
914 1416 5211 JMP .-5 /GO GET NEW WORD,IT WAS A LF AND FILL WAS SELECTED
915 1417 1042 TAD XMTDAT /GET THE WORD TO TRANSMIT
916 1420 6046 XMIT, TLS /TRANSMIT THE WORD
917 1421 6041 TSF23, TSF /WAIT FOR THE FLAG
918 1422 5221 JMP .-1
919 1423 5230 JMP .+5 /GO TRANSMIT ANOTHER CHARACTER,
920 1424 6046 TLS10, TLS /WAIT READER FLAG
921 1425 6031 KSF10, KSF
922 1426 5225 JMP .-1
923 1427 5236 JMP RECEVE
924 1430 4771 RANDM1 /GO GENERATE ANOTHER WORD (XMTDT1)
925 1431 1043 TAD XMTDT1
926 1432 4575 JMS I IFILCHK /CHECK TO SEE IF FILL=1 AND THAT IT WAS A LF
927 1433 5226 JMP .-5 /YES IT WAS,GO GENERATE A NEW WORD
928 1434 1043 TAD XMTDT1 /GET THE WORD AND PRINT IT
929 1435 5224 JMP TLS10 /
930 1436 7240 RECEVE, CLA CMA
931 1437 6036 KRB3, KRB /READ THE BUFFER AND CLEAR THE FLAGS

```

```

932 1440 3045 DCA RECDAT /SAVE THE WORD
933 1441 1045 TAD RECDAT /GET THE WORD AND COMPARE IT TO THE WORD
934 1442 7041 CIA /TRANSMITTED
935 1443 1042 TAD XMTDAT
936 1444 7650 SNA CLA /ARE THEY EQUAL?
937 1445 5265 JMP UPDATE /YES,GO CHECK LOOP SWITCH
938 1446 4512 AFRORR /REPORT TO APT?
939 1447 5263 JMP UPDATE-2 /ACTIVE CONSOLE
940 1450 4517 LAS
941 1451 7700 SMA CLA /HALT ON ERROR CONDITION
942 1452 5263 JMP UPDATE-2 /NO, CHECK FOR LOOP ON ERROR
943 1453 1042 TAD XMTDAT /GET THE FIRST WORD TRANSMITTED
944 1454 7402 HALT /AC=THE 1ST WORD TRANSMITTED
945 1455 7200 CLA
946 1456 1045 TAD RECDAT /
947 1457 7402 HALT /AC=WORD RECEIVED
948 1460 7200 CLA
949 1461 1043 TAD XMTDT1
950 1462 7402 HALT /AC=2ND WORD TRANSMITTED IF PROGRAM GOT THAT FAR
951 1463 7240 CLA CMA
952 1464 3044 DCA ERRFLG /SET ERROR FLAG FOR SCOPE LOOPING
953 1465 4517 UPDATE, LAS /IS SP1=1
954 1466 7004 RAL
955 1467 7710 SPA CLA
956 1470 5310 JMP ERRLOP /LOOP ON DATA PATTERN
957 1471 1043 TAD XMTDT1 /PUT SECOND WORD IN FIRST WORD FOR COMPARISON
958 1472 3042 DCA XMTDAT /OF NEXT READ
959 1473 4516 NERROR
960 1474 1476 JMP .+2
961 1475 5302 JMP END /END OF TEST
962 1476 1044 TAD ERRFLG /CHECK THE ERROR FLAG FOR RETURN POINTER
963 1477 7640 SZA CLA
964 1500 5206 JMP FDATAT+6 /ERROR GO START TEST OVER
965 1501 5221 JMP XMIT+1 /GO TRANSMIT NEXT CHARACTER AND WAIT FOR RECEIVE
966 1502 4467 END, KSFSKP
967 1503 4464 EHLTLP /LAST FLAG FAILED TO SET
968 1504 6036 KRB4, KRB /CLEAR THE FLAG
969 1505 6042 TCF10, TCF /CLEAR THE TRANSMIT FLAG
970 1506 4463 LOOP /LOOP ON TEST IF SRI=1
971 1507 5573 JMP I ICHARLG
972
973
974 1510 4461 ERRLOP, DELAY /DELAY 200MS TO ALLOW FLAGS TO SETTLE
975 1511 6036 KRB5, KRB /CLEAR RECEIVE FLAG IF SET
976 1512 6042 TCF11, TCF /CLEAR TRANSMIT FLAG IF SET
977 1513 7200 CLA
978 1514 1042 TAD XMTDAT /GET THE FIRST WORD TO TRANSMIT
979 1515 6046 TLS1, TLS /LOAD AND TRANSMIT IT
980 1516 6041 TSF24, TSF
981 1517 5316 JMP .-1 /WAIT FOR THE FIRST TRANSMIT FLAG
982 1520 7200 CLA
983 1521 1043 TAD XMTDT1 /GET THE SECOND WORD TO TRANSMIT
984 1522 6046 TLS2, TLS /LOAD AND TRANSMIT IT
985 1523 6031 FDTLOP, KSF /WAIT FOR THE FIRST RECEIVE FLAG
986 1524 5323 JMP .-1

```

```

987 1525 7240          CLA      CMA
988 1526 6036          KRB      /READ THE FIRST WORD
989 1527 3045          DCA      RECDAT /SAVE THE FIRST WORD RECEIVED
990 1530 1045          TAD      RECDAT /COMPARE IT TO THE FIRST WORD TRANSMITTED
991 1531 7041          CIA
992 1532 1042          TAD      XMTDAT
993 1533 7640          SZA      CLA
994 1534 5364          JMP      XSR1LP /ERROR GO CHECK SR1
995 1535 6041          TSF25, TSF
996 1536 5335          JMP      .-1 /WAIT FOR 2ND TRANSMIT FLAG
997 1537 1042          TAD      XMTDAT /GET THE FIRST WORD AGAIN
998 1540 6046          TLS3,  TLS /LOAD AND TRANSMIT IT
999 1541 6031          KSF19, KSF
1000 1542 5341          JMP      .-1 /WAIT FOR SECOND RECEIVE FLAG
1001 1543 7240          CLA      CMA
1002 1544 6036          KRB7,  KRB /READ THE SECOND WORD
1003 1545 3045          DCA      RECDAT /IS IT EQUAL TO SECOND TRANSMIT
1004 1546 1045          TAD      RECDAT
1005 1547 7041          CIA
1006 1550 1043          TAD      XMTDT1
1007 1551 7640          SZA      CLA
1008 1552 5364          JMP      XSR1LP /ERROR,GO CHECK SR1=1
1009 1553 6041          TSF26, TSF
1010 1554 5353          JMP      .-1 /WAIT FOR THE TRANSMIT FLAG
1011 1555 1043          TAD      XMTDT1 /GET 2ND WORD AND TRANSMIT IT
1012 1556 6046          TLS4,  TLS /LOAD AND TRANSMIT
1013 1557 4517          LAS
1014 1560 7004          RAL
1015 1561 7710          SPA      CLA
1016 1562 5323          JMP      FDTLOP
1017 1563 5206          JMP      FDATAT+6
1018
1019 1564 4517          XSR1LP, LAS
1020 1565 7004          RAL
1021 1566 7710          SPA      CLA
1022 1567 5310          JMP      ERRLOP
1023 1570 5271          JMP      UPDATF+4
1024
1025 1571 4771          RANDM1=JMS I
1026 1571 4133          XRAND1
1027 /
1028 /
1029 1600          PAGE
1030
1031 /THE FOLLOWING TEST CHECKS THAT THE NUMBER OF DATA BITS WERE
1032 /SETUP CORRECTLY. TRANSMIT 377 AND TAKE THE 1'S COMPLEMENT
1033 /OF THE DATA BIT MASK WORD AND CHECK THAT THE AC CAME BACK
1034 /AS ZEROES.
1035
1036 1600 4462          CHARLG, STLPPC /STORE THE LOOPING PC FOR ERROR AND TEST LOOPING
1037 1601 4460          LOAD
1038 1602 1040          TAD      DATBIT /TIMING FOR APT.
1039 1603 7040          CMA      /SETUP 1'S COMPLEMENT OF SELECTED DATA
1040 1604 3042          DCA      XMTDAT /BIT CHARACTER LENGTH
1041 1605 1034          TAD      K377 /SAVE IT FOR COMPARISON

```

```

1042 1606 6046          TLS5,  TLS /TRANSMIT 8 BITS OF ONES
1043 1607 4466          TSFSKP /WAIT FOR THE TRANSMIT FLAG TO SET
1044 1610 4464          EHLTLP /TRANSMIT FLAG FAILED TO SET
1045 1611 6042          TCF12, TCF /CLEAR THE FLAG
1046 1612 4467          KFSFKP /WAIT FOR THE RECEIVE FLAG TO SET
1047 1613 4464          EHLTLP /ERROR,RECEIVE FLAG FAILED TO SET
1048 1614 7240          CLA      CMA
1049 1615 6036          KRB8,  KRB /READ THE WORD AND SAVE IT
1050 1616 3045          DCA      RECDAT
1051 1617 1045          TAD      RECDAT
1052 1620 6042          AND      XMTDAT
1053 1621 7450          SNA
1054 1622 5242          JMP      FILERT-3 /TEST FOR LOOP ON TEST
1055 1623 3043          DCA      XMTDT1 /SAVE THE BITS THAT WEREN'T SUPPOSED TO BE SELECTED
1056 1624 4512          AERROR
1057 1625 5236          JMP      CHRLLAS /CONSOLE ACTIVE
1058 1626 4517          LAS
1059 1627 7700          SMA CLA /HALT ON ERROR
1060 1630 5236          JMP      CHRLLAS /NO, TEST FOR LOOP ON TEST CONDITON
1061 1631 1043          TAD      XMTDT1
1062 1632 7402          HALT
1063 1633 7200          CLA
1064 1634 1040          TAD      DATBIT /AC=BITS THAT WEREN'T SUPPOSED TO BE SELECTED
1065 1635 7402          HALT /AC=DATA BITS THAT OPERATOR HAD TOLD THE PROGRAM
1066 1636 4517          CHRLLAS, LAS /THAT WERE SELECTED
1067
1068 1637 7004          RAL
1069 1640 7710          SPA      CLA
1070 1641 5204          JMP      CHARLG+4
1071 1642 4516          NERROR
1072 1643 1604          CHARLG+4 /LOOP ON ERROR
1073 1644 4463          LOOP /LOOP ON TEST IF SR2=1
1074
1075 /FILLER CHARACTER TEST-DO THIS TEST IF OPERATOR HAS SELECTED
1076 /THE FILLER CHARACTER OPTION. THE PROGRAM TRANSMITS A LINE
1077 /FEED AND CHECKS THAT 5 RECEIVE FLAGS COME BACK. THE DATA RECEIVED
1078 /SHOULD BE 1 WORD OF LINE FEED AND 4 WORDS OF FILLER CHARACTERS.
1079 /THE PROGRAM ALSO CHECKS THAT THE TRANSMIT FLAG DOES NOT GET SET
1080 /UNTIL ALL THE RECEIVE FLAGS ARE IN.
1081
1082 1645 1036          FILFRT, TAD /CHECK TO SEE IF FILLER CHARACTERS SELECTED
1083 1646 7006          RTL
1084 1647 7700          SMA CLA /WAS IT SELECTED BY THE OPERATOR
1085 1650 5777          JMP      STENAB /NO,GO CHECK FOR STATUS ENABLE
1086 1651 4462          STLPPC /STORE THE LOOPING PC
1087 1652 3047          DCA      NDELAY /SETUP PROGRAM DELAY OF 409MS.
1088 1653 4461          DELAY /DELAY TO ALLOW FLAGS TO SETTLE
1089 1654 6036          KRB9,  KRB /CLEAR THE RECEIVE FLAG IF SET
1090 1655 7200          CLA
1091 1656 6042          TCF13, TCF /CLEAR THE TRANSMIT FLAG IF SET
1092 1657 1376          TAD      (=4 /SETUP A COUNTER TO RECEIVE FOUR FLAGS
1093 1660 3043          DCA      XMTDT1 /SAVE IT
1094 1661 1057          TAD      K212 /GET LINE FEED
1095 1662 0040          AND      DATBIT /MASK OUT TO WORD LENGTH
1096 1663 3042          DCA      XMTDAT /SAVE IT FOR COMPARISON OF FIRST WORD

```

```

1097 1664 1042 TAD XMTDAT /GET THE LINE FEED CHARACTER
1098 1665 6046 TLS6, TLS /LOAD AND TRANSMIT IT
1099 1666 7200 CLA
1100 1667 1172 FILLOP, TAD [-100 /LOOP TO WAIT FOR THE RECEIVE FLAG AND CHECK XMIT
1101 1670 3051 DCA CNT2
1102 1671 3050 DCA CNT1
1103 1672 6041 TSF28, TSF /SKIP IF TRANSMIT FLAG=1
1104 1673 7610 SKP CLA
1105 1674 4464 EHLTLP /ERROR, TRANSMIT FLAG SET-SHOULD GET 4 RECEIVE FLAGS FIRST
1106 1675 6031 KSF21, KSF /SKIP IF RECEIVE FLAG SET
1107 1676 7610 SKP CLA
1108 1677 5305 JMP +6
1109 1700 2050 ISZ CNT1
1110 1701 5272 JMP -7
1111 1702 2051 ISZ CNT2
1112 1703 5272 JMP -11
1113 1704 4464 EHLTLP /ERROR, RECEIVE FLAG NOT SET OR MISSING SOME
1114 1705 1043 TAD XMTDT1 /SETUP TO COMPARE FOR EITHER A L.F. OR FILLER
1115 1706 1375 TAD (4 /WAS IT THE FIRST CHARACTER
1116 1707 7640 SZ CLA
1117 1710 3042 DCA XMTDAT /NO, THEN CLEAR COMPARE WORD FOR FILLER CHAR
1118 1711 6036 KRB10, KRB /READ THE WORD AND CLEAR THE FLAG
1119 1712 3045 DCA RECDAT /SAVE IT
1120 1713 1045 TAD RECDAT /COMPARE THE WORD RECEIVED WITH WORD EXPECTED
1121 1714 7041 CIA
1122 1715 1042 TAD XMTDAT
1123 1716 7650 SNA CLA
1124 1717 5337 JMP CNTREC /WORD COMPARES, GO BUMP RECEIVE COUNTER
1125 1720 4512 AERROR
1126 1721 5344 JMP CNTREC+5 /ACTIVE CONSOLE
1127 1722 4517 LAS
1128 1723 7700 SMA CLA /HALT ON ERROR?
1129 1724 5332 JMP CNTREC-5 /NO, CHECK FOR LOOP ON ERROR
1130 1725 1042 TAD XMTDAT /PRESS "CONTINUE" FOR EXPECTED CHARACTER
1131 1726 7402 HALT /AC=WORD EXPECTED X12=LF OR 0000=FIL CHAR.
1132 1727 7200 CLA
1133 1730 1045 TAD RECDAT /GET THE WORD RECEIVED
1134 1731 7402 HALT /AC=WORD RECEIVED SHOULD BE 212 OR 12 OR 000
1135 1732 4517 LAS /CHECK SR1 TO LOOP ON ERROR
1136 1733 7004 RAL
1137 1734 7710 SPA CLA
1138 1735 5251 JMP FILERT+4 /LOOP ON THE ERROR
1139 1736 5774 JMP FILEXT /EXIT THE TEST
1140 1737 2043 CNTREC, ISZ XMTDT1 /BUMP THE RECEIVE COUNTER
1141 1740 5267 JMP FILLOP /GO GET THE NEXT RECEIVE FLAG
1142 1741 1172 TAD [-100
1143 1742 3051 DCA CNT2
1144 1743 3050 DCA CNT1
1145 1744 6031 KSF22, KSF /CHECK THAT THE RECEIVE FLAG DOESN'T GET SET BEFORE XMIT
1146 1745 7610 SKP CLA
1147 1746 4464 EHLTLP /LAST RECEIVE FLAG SHOULDN'T GET SET UNTIL SOME
1148 / TIME AFTER THE TRANSMIT FLAG
1149 1747 6041 TSF29, TSF /WAIT FOR THE TRANSMIT FLAG TO GET SET
1150 1750 7610 SKP CLA
1151 1751 5357 JMP +6

```

```

1152 1752 2050 ISZ CNT1
1153 1753 5344 JMP -7
1154 1754 2051 ISZ CNT2
1155 1755 5344 JMP -11
1156 1756 4464 EHLTLP /TRANSMIT FLAG FAILED TO SET AFTER 5 RECEIVE FLAGS
1157 1757 6032 KCC5, KCC /CLEAR THE FLAG
1158 1760 1172 TAD [-100
1159 1761 3051 DCA CNT2
1160 1762 6031 KSF23, KSF
1161 1763 7610 SKP CLA
1162 1764 5372 JMP +6
1163 1765 2050 ISZ CNT1
1164 1766 5362 JMP -4
1165 1767 2051 ISZ CNT2
1166 1770 5362 JMP -6
1167 1771 4464 EHLTLP /LAST RECEIVE FLAG FAILED TO SET
1168 1772 5773 JMP KCC6
1169 /
1170 1773 2000 PAGE
1171 1774 2003 /
1172 1775 0004 KCC6, KCC /CLEAR THE RECEIVE FLAG
1173 1776 7774 NERROR
1174 1777 2005 FILLOP-10 /GO TRANSMIT ANOTHER FILLER CHARACTER
1175 / FILEXT, LOOP /LOOP ON TEST IF SR2=1
1176 2004 5205 JMP STENAB
1177 /
1178 /
1179 /
1180 /
1181 /
1182 /
1183 /
1184 /
1185 /
1186 /
1187 /
1188 /
1189 /
1190 /
1191 /
1192 2005 1036 STENAB, TAD SAVBTS /CHECK TO SEE IF STATUS ENABLE WAS SELECTED
1193 2006 7004 RAL
1194 2007 7700 SMA CLA /WAS IT SELECTED BY THE OPERATOR
1195 2010 5333 JMP SR4HLT /NO, GO CHECK END OF PROGRAM HALT
1196 2011 4462 STLPPC /STORE THE LOOPING PC FOR TEST AND SCOPE LOOPING
1197 2012 3047 DCA NDELAY /SETUP A DELAY OF 409MS
1198 2013 4461 DELAY /DELAY TO ALLOW FLAGS TO SETTLE FOR SCOPE LOOPING
1199 2014 6036 KRB12, KRB /CLEAR THE RECEIVE FLAG IF SET
1200 2015 6042 TCF15, TCF /CLEAR THE TRANSMIT FLAG IF SET
1201 2016 7346 CLA CLL CMA RTL /SETUP A COUNT OF 3 TO TRANSMIT 3 TIMES
1202 2017 3043 DCA XMTDT1 /SAVE IT
1203 2020 7001 IAC
1204 2021 3042 DCA XMTDAT /SET THE FIRST WORD TO BE TRANSMITTED=1
1205 2022 7326 CLA CLL CML RTL /SET AC BIT 10 TO A 1

```

```

1206 2023 6035 KSE1, KSE /AND TRY TO SET STATUS ENABLE TO A 1
1207 2024 7200 CLA
1208 2025 1042 TAD XMTDAT /GET THE WORD (1,2 OR 3)
1209 2026 6046 TLS9, TLS /TRANSMIT IT
1210 2027 7200 CLA
1211 2030 4466 TSFSKP /WAIT FOR THE TRANSMIT FLAG
1212 2031 4464 EHLTLP /ERROR, TRANSMIT FLAG FAILED TO SET
1213 2032 2042 ISZ XMTDAT /BUMP THE WORD TO A 2 THEN 3
1214 2033 2043 ISZ XMTDT1 /TRANSMITTED 3 WORDS YET
1215 2034 5225 JMP *-7 /NO, GO TRANSMIT NEXT WORD
1216 2035 6042 TCF16, TCF /CLEAR THE TRANSMIT FLAG
1217 2036 6031 KSF25, KSF /SKIP ON THE RECEIVE FLAG
1218 2037 4464 EHLTLP /ERROR, RECEIVE FLAG FAILED TO SET AFTER 3 XMTS
1219 2040 6034 KRS2, KRS /DO A STATIC READ OF STATUS AND RECEIVE BUFFER
1220 2041 3045 DCA RECDAT /AND SAVE IT
1221 2042 1377 TAD (4402 /GET EXPECTED WORD (ERROR-OVERRUN-DATA OF 2)
1222 2043 3042 DCA XMTDAT /AND SAVE IT FOR COMPARISON
1223 2044 4301 JMS STERR /GO CHECK THE WORDS FOR ERRORS
1224 2045 6035 KSE2, KSE /CLEAR STATUS ENABLE AND CHECK BUFFER FOR A 2
1225 2046 6034 KRS3, KRS /DO A STATIC READ OF THE RECEIVE BUFFER
1226 2047 3045 DCA RECDAT /SAVE THE WORD
1227 2050 7326 CLA CLL CML RTL /SETUP FOR WORD EXPECTED
1228 2051 3042 DCA XMTDAT /SAVE IT FOR COMPARISON
1229 2052 4301 JMS STERR /GO CHECK FOR ERRORS
1230 2053 7326 CLA CLL CML RTL /SET AC BIT 10 TO A 1
1231 2054 6035 KSE3, KSE /RESET STATUS ENABLE
1232 2055 7200 CLA
1233 2056 1377 TAD (4402 /RESET EXPECTED DATA TO (ERROR-OVERRUN-DATA OF 2)
1234 2057 3042 DCA XMTDAT /NO A DYNAMIC READ OF STATUS AND RECEIVE BUFFER
1235 2060 6036 KRB13, KRB /SAVE THE WORD READ FOR COMPARISON
1236 2061 3045 DCA RECDAT /GO CHECK FOR ERROR CONDITIONS
1237 2062 4301 JMS STERR /CHECK THAT RECEIVE FLAG = 0
1238 2063 6031 KSF26, KSF
1239 2064 7610 SKP CLA /ERROR, RECEIVE FLAG SHOULD NOT BE SET YET
1240 2065 4464 EHLTLP /GO WAIT FOR THE LAST TRANSMITTED CHARACTER
1241 2066 4467 KSF5KP /THE THIRD TRANSMIT DID NOT SET RECEIVE FLAG
1242 2067 4464 EHLTLP /READ THE CHARACTER
1243 2070 6036 KRB14, KRB /SAVE IT FOR COMPARISON
1244 2071 3045 DCA RECDAT
1245 2072 1376 TAD (3 /SETUP FOR WORD EXPECTED
1246 2073 3042 DCA XMTDAT /GO CHECK FOR ERRORS
1247 2074 4301 JMS STERR
1248 2075 4516 NERROR
1249 2076 2012 STENAB+5 /NO GO DO TEST AGAIN
1250 2077 4463 LOOP /LOOP ON TEST IF SR2=1
1251 2100 5333 JMP SR4HLT /GO CHECK END OF PROGRAM HALT
1252
1253 2101 0000 STERR, 0 /CHECK TO SEE IF ERROR EXIST IN STATUS REGISTER AND RECEIVE BUFFER
1254 2102 1045 TAD RECDAT /GET THE WORD RECEIVED AND COMPARE IT WITH
1255 2103 7041 CIA /THE WORD EXPECTED
1256 2104 1042 TAD XMTDAT /
1257 2105 7650 SNA CLA /ARE THEY EQUAL?
1258 2106 5701 JMP I STERR /YES, CONTINUE TESTING
1259 2107 7340 CLL CLA CMA /AC=-1
1260 2110 1301 TAD STERR /ESTABLISH ERROR PC

```

```

1261 2111 4512 AERROR
1262 2112 5326 JMP STLAS /ACTIVE CONSOLE SELECTED
1263 2113 4517 LAS
1264 2114 7700 SNA CLA /HALT ON ERROR
1265 2115 5326 JMP STLAS /NO, GO TEST FOR LOOP ON ERROR
1266 2116 1010 TAD 10 /GET BACK ERROR PC
1267 2117 7402 HALT /AC=PC WERE ERROR WAS DETECTED AT
1268 2120 7200 CLA
1269 2121 1042 TAD XMTDAT /GET THE WORD EXPECTED
1270 2122 7402 HALT /AC=WORD EXPECTED
1271 2123 7200 CLA
1272 2124 1045 TAD RECDAT /GET THE WORD RECEIVED
1273 2125 7402 HALT /AC=WORD RECEIVED
1274 2126 4517 STLAS, LAS /LOOP ON ERROR?
1275 2127 7004 RAL
1276 2130 7710 SPA CLA /YES GO LOOP
1277 2131 5211 JMP STENAB+4 /NO, GO GET NEXT ERROR
1278 2132 5701 JMP I STERR
1279
1280
1281 /HALT AT END OF PROGRAM IF SWITCH REGISTER 4 EQUALS A ONE
1282
1283 2133 4515 SR4HLT, CHEK22
1284 2134 4775 JMS XC8PAS /PRINT END OF PASS IF ON CONSOLE
1285 2135 7000 NOP
1286 2136 4517 LAS
1287 2137 0374 AND (200
1288 2140 7650 SNA CLA
1289 2141 5346 JMP BAUDST /SET UP BUAD TABLE.
1290 2142 4515 CHEK22
1291 2143 4773 JMS XC9PSW
1292 2144 4520 HLT /END OF THE PROGRAM SR4=1
1293 2145 5772 JMP CLRBRD /PRESS CONTINUE TO GO ON
1294
1295 2146 1022 BAUDST, TAD 22
1296 2147 7004 RAL
1297 2150 7710 SPA CLA /SPECIAL TESTING BEING DONE
1298 2151 4771 JMS NEWDEV /YES UPDATE DEVICE NUMBER
1299 2152 1524 TAD I TEMP
1300 2153 3121 DCA CLKCNT /GET FIRST CLOCK COUNT,
1301 2154 7240 STA
1302 2155 1124 TAD TEMP /SET UP FOR AUTO INDEX.
1303 2156 3011 DCA A11
1304 2157 5772 JMP CLRBRD /EXIT AND CONTINUE PROGRAM
1305
1306 /
1307 /ROUTINE TO LOAD VALUE FOR APT TIMING.
1308 /
1309 2160 0000 XLOAD, 0
1310 2161 1411 TAD I A11 /GET TIMING VALUE.
1311 2162 7450 SNA /TEST FOR A ZERO.
1312 2163 5760 JMP I XLOAD /THERE IS ONE, USE SAME VALUE.
1313 2164 3122 DCA COUNT /SETUP NEW VALUE.
1314 2165 5760 JMP I XLOAD /EXIT.
1315 /

```



```

1316 2171 4000
1317 2172 0271
1318 2173 6703
1319 2174 0200
1320 2175 7241
1321 2176 0003
1322 2177 4402
1323 2200
1324 2200
1325 2200 0000
1326 2201 7240
1327 2202 1200
1328 2203 4512
1329 2204 7410
1330 2205 4517
1331 2206 7700
1332 2207 5212
1333 2210 1010
1334 2211 7402
1335 2212 4517
1336 2213 7004
1337 2214 7710
1338 2215 5446
1339 2216 5600
1340
1341 2217 0000
1342 2220 3240
1343 2221 7240
1344 2222 1217
1345 2223 4512
1346 2224 5232
1347 2225 4517
1348 2226 7700
1349 2227 5232
1350 2230 1010
1351 2231 7402
1352 2232 4517
1353 2233 7004
1354 2234 7710
1355 2235 5446
1356 2236 1240
1357 2237 5617
1358
1359 2240 0000
1360
1361 2241 0000
1362 2242 3240
1363 2243 7240
1364 2244 1241
1365 2245 4512
1366 2246 5254
1367 2247 4517
1368 2250 7700
1369 2251 5254

```

PAGE

```

HLTLOP, 0 /INHIBIT ERROR HALT IF SR0=1 AND LOOP ON ERROR IF SR1=1
CLA CMA
TAD HLTLOP /GET THE FAILING PC WHERE THE ERROR WAS DETECTED
AERROR /APT ERROR HANDLER.
SKP /ACTIVE CONSOLE
LAS /HALT ON ERROR?
SMA CLA
JMP .+3
TAD 10 /GET BACK ERROR PC
HALT /AC=FAILING PC WHERE ERROR WAS DETECTED
LAS /LOOP ON THE ERROR?
RAL
SPA CLA
JMP I LOOPPC
JMP I HLTLOP

HLTLP1, 0 /ROUTINE USED IN ACNSKP TEST TO SAVE AC FOR FALSE SKIPPING
DCA SAVAC /SAVE THE AC FOR NON LOOPING PURPOSES
CLA CMA
TAD HLTLP1 /GET THE FAILING PC WHERE ERROR WAS DETECTED
AERROR /REPORT ERROR TO APT=8 SYSTEM
JMP .+6 /ACTIVE CONSOLE
LAS /HALT ON ERROR?
SMA CLA
JMP .+3
TAD 10 /GET BACK ERROR PC
HALT /AC=FAILING PC WHERE ERROR WAS DETECTED
LAS /LOOP ON THE ERROR?
RAL
SPA CLA
JMP I LOOPPC
TAD SAVAC /NO, RESET THE AC AND CONTINUE
JMP I HLTLP1 /RETURN AND CHECK THE EFFECT OF THE IOT ON AC

SAVAC, 0
HLTLP2, 0 /THIS ROUTINE USED ONLY WHEN IOT EFFECTS CONTENTS OF AC
DCA SAVAC /SAVE THE AC FOR ERROR INDICATION
CLA CMA /GET THE FAILING PC WHERE ERROR WAS DETECTED
TAD HLTLP2 /
AERROR /REPORT ERROR TO APT.
JMP .+6 /ACTIVE CONSOLE
LAS /HALT ON ERROR?
SMA CLA
JMP .+3

```

```

1370 2252 1010 TAD 10 /GET BACK ERROR PC
1371 2253 7402 HALT /AC=FAILING PC WHERE ERROR WAS DETECTED
1372 2254 7200 CLA /PRESS "CONT" TO GET CONTENTS OF AC AFTER EXECUTION
1373 /AND COMPARISON OF THE IOT AND AC
1374 2255 1240 TAD SAVAC
1375 2256 7402 HALT /AC=BITS THAT WERE EFFECTED AFTER EXECUTION OF IOT
1376 2257 4517 LAS /LOOP ON THE ERROR?
1377 2260 7004 RAL
1378 2261 7710 SPA CLA
1379 2262 5446 JMP I LOOPPC /YES, GO LOOP ON THE ERROR
1380 2263 5641 JMP I HLTLP2 /RETURN AND CONTINUE THE TEST
1381
1382
1383
1384 /PAUD RATE TIMING TEST-LOAD AND START 202 AND WATCH YOUR WATCH
1385 /FOR 30 SECONDS. THE PROCESSOR SHOULD HALT IN 30 SECONDS, IF IT
1386 /DOESN'T CHECK THE BAUD RATE WITH A SCOPE OR CHECK THE BAUD RATE
1387 /SWITCHES
1388
1389 2264 1377 BAUDTM, TAD (JMP I 2
1390 2265 3001 DCA 1
1391 2266 1376 TAD (RETINT
1392 2267 3002 DCA 2 /SETUP RETURN POINTER FOR THE INTERRUPT
1393 2270 1041 TAD BAUDNO /GET THE BAUD RATE
1394 2271 1375 TAD (BAUDTB /GET THE ADDRESS OF THE BAUD RATE TABLE
1395 2272 3333 DCA BDPNTR /SAVE THE POINTER TO THE BAUD RATE TABLE
1396 2273 1037 TAD BITNO /GET THE CHARACTER LENGTH
1397 2274 7104 CLL RAL /MULTIPLY IT BY 2
1398 2275 1733 TAD I BDPNTR /ADD IN BAUD RATE ADDRESS
1399 2276 3333 DCA BDPNTR /ADDRESS OF BAUD RATE CONSTANTS ARE READY TO BE SETUP
1400 2277 4462 STLPPC /STORE THE LOOPING PC FOR TEST LOOPING
1401 2300 1333 TAD BDPNTR /GET POINTER ADDRESS TO THE CONSTANTS
1402 2301 3052 DCA TSTCNT /SAVE IT IN TEST COUNT
1403 2302 1452 TAD I TSTCNT /GET THE FIRST CONSTANT
1404 2303 3050 DCA CNT1
1405 2304 2052 ISZ TSTCNT /BUMP THE ADDRESS BY 1 FOR NEXT CONSTANT
1406 2305 1452 TAD I TSTCNT
1407 2306 3051 DCA CNT2
1408 2307 6046 TLS7, TLS /LOAD AND TRANSMIT FIRST CHARACTER-FLAG
1409 2310 6041 TSF30, TSF /COMES UP ALMOST IMMEDIATELY
1410 2311 5310 JMP .-1
1411 2312 5315 JMP .+3
1412 2313 6036 INTON, KRB /CLEAR THE RECEIVE FLAG
1413 2314 7610 SKP CLA
1414 2315 6046 TLS8, TLS /LOAD AND TRANSMIT NEXT CHARACTER AND CLEAR FLAG
1415 2316 6001 ION /TURN THE INTERRUPT ON
1416 2317 6031 RETINT, KSF /SKIP IF RECEIVE FLAG SET
1417 2320 7610 SKP CLA /FLAG NOT SET, CHECK TRANSMIT FLAG
1418 2321 5313 JMP INTON /RECEIVE FLAG SET, GO CLEAR IT
1419 2322 6041 TSF31, TSF /SKIP IF TRANSMIT FLAG SET
1420 2323 5317 JMP INTON+4 /NOT SET YET GO WAIT FOR A FLAG
1421 2324 2050 ISZ CNT1 /BUMP THE FIRST COUNTER
1422 2325 5315 JMP INTON+2 /GO TRANSMIT ANOTHER CHARACTER
1423 2326 2051 ISZ CNT2 /FIRST COUNTER OVERFLOWED
1424 2327 5315 JMP INTON+2 /GO DO ANOTHER 4095 INTERRUPTS

```

```

1425 2330 4520 HLT /TRANSMITTED FOR 30 SECONDS????
1426 2331 4463 LOOP /LOOP ON TEST IF SR2=1
1427 2332 5330 JMP /END OF THE TEST
1428 2333 0000 BDPNTR, 0
1429
1430 /POINTERS TO BAUD RATE TABLE
1431 2334 2400 BAUDTB, BR110
1432 2335 2414 BP150
1433 2336 2430 BR300
1434 2337 2444 BR600
1435 2340 2460 BR1200
1436 2341 2474 BR2400
1437 2342 2510 BR4800
1438 2343 2524 BR9600
1439 2344 2540 B19200
1440 2345 2554 BR568
1441 2346 2570 BR667
1442 2347 2604 BR1050
1443
1444 /POINTERS TO BAUD RATE TYPEOUTS
1445 2350 4473 MESTAB, MESS6A
1446 2351 4520 MESS6B
1447 2352 4545 MESS6C
1448 2353 4572 MESS6D
1449 2354 4617 MESS6E
1450 2355 4645 MESS6F
1451 2356 4673 MESS6G
1452 2357 4721 MESS6H
1453 2360 4747 MESS6I
1454 2361 4776 MESS6J
1455 2362 5024 MESS6K
1456 2363 5052 MESS6L
1457
1458 /POINTERS TO DATA BIT TYPEOUTS
1459 2364 5145 MESTB1, MES10A
1460 2365 5164 MES10B
1461 2366 5203 MES10C
1462 2367 5222 MES10D
1463
1464
1465 2375 2334 PAGE
1466 2376 2317
1467 2377 5402
1468 2400
1469 /BAUD RATE CONSTANTS FOR 110 BAUD
1470
1471 2400 7051 BR110, -727
1472 2401 7777 -1
1473 2402 7143 -635
1474 2403 7777 -1
1475 2404 7221 -557
1476 2405 7777 -1
1477 2406 7266 -512
1478 2407 7777 -1

```

```

1479 2410 7324 -454 /11 BITS AT 10 CHAR/SEC=300 CHAR/30 SEC
1480 2411 7777 -1
1481 2412 7355 -423 /12 BITS AT 9.17 CHAR/SEC=275 CHAR/30 SEC
1482 2413 7777 -1
1483
1484 /BAUD RATE CONSTANTS FOR 150 BAUD
1485
1486 2414 6575 BR150, -1203
1487 2415 7777 -1
1488 2416 6715 -1063
1489 2417 7777 -1
1490 2420 7014 -764
1491 2421 7777 -1
1492 2422 7076 -702
1493 2423 7777 -1
1494 2424 7147 -631
1495 2425 7777 -1
1496 2426 7211 -567
1497 2427 7777 -1
1498
1499 /BAUD RATE CONSTANTS FOR 300 BAUD
1500
1501 2430 5372 BR300, -2406
1502 2431 7777 -1
1503 2432 5633 -2145
1504 2433 7777 -1
1505 2434 6030 -1750
1506 2435 7777 -1
1507 2436 6174 -1604
1508 2437 7777 -1
1509 2440 6316 -1462
1510 2441 7777 -1
1511 2442 6422 -1356
1512 2443 7777 -1
1513
1514 /BAUD RATE CONSTANTS FOR 600 BAUD
1515
1516 2444 2765 BR600, -5013
1517 2445 7777 -1
1518 2446 3466 -4312
1519 2447 7777 -1
1520 2450 4060 -3720
1521 2451 7777 -1
1522 2452 4370 -3410
1523 2453 7777 -1
1524 2454 4633 -3145
1525 2455 7777 -1
1526 2456 5044 -2734
1527 2457 7777 -1
1528
1529 /BAUD RATE CONSTANTS FOR 1200 BAUD
1530
1531 2460 5750 BR1200, -2030
1532 2461 7776 -2
1533 2462 7153 -625 /8 BITS AT 150 CHAR/SEC=4500 CHAR/30 SEC

```

1534 2463 7776 -2  
 1535 2464 0140 -7640  
 1536 2465 7777 -1  
 1537 2466 0760 -7020  
 1538 2467 7777 -1  
 1539 2470 1467 -6311  
 1540 2471 7777 -1  
 1541 2472 2110 -5670  
 1542 2473 7777 -1  
 1543  
 1544  
 1545  
 1546 2474 3720 BR2400, -4060  
 1547 2475 7775 -3  
 1548 2476 6326 -1452  
 1549 2477 7775 -3  
 1550 2500 0277 -7501  
 1551 2501 7776 -2  
 1552 2502 1737 -6041  
 1553 2503 7776 -2  
 1554 2504 3156 -4622  
 1555 2505 7776 -2  
 1556 2506 4217 -3561  
 1557 2507 7776 -2  
 1558  
 1559  
 1560  
 1561 2510 7640 BR4800, -140  
 1562 2511 7772 -6  
 1563 2512 4654 -3124  
 1564 2513 7773 -5  
 1565 2514 0575 -7203  
 1566 2515 7774 -4  
 1567 2516 3675 -4103  
 1568 2517 7774 -4  
 1569 2520 6332 -1446  
 1570 2521 7774 -4  
 1571 2522 0436 -7342  
 1572 2523 7775 -3  
 1573  
 1574  
 1575  
 1576 2524 7477 BR9600, -301  
 1577 2525 7765 -13  
 1578 2526 1530 -6250  
 1579 2527 7767 -11  
 1580 2530 1371 -6407  
 1581 2531 7770 -10  
 1582 2532 7571 -207  
 1583 2533 7770 -10  
 1584 2534 4664 -3114  
 1585 2535 7771 -7  
 1586 2536 1073 -6705  
 1587 2537 7772 -6  
 1588

/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  
 /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

1589  
 1590  
 1591 2540 7176 B19200, -602  
 1592 2541 7753 -25  
 1593 2542 3257 -4521  
 1594 2543 7756 -22  
 1595 2544 2761 -5017  
 1596 2545 7760 -20  
 1597 2546 7362 -416  
 1598 2547 7761 -17  
 1599 2550 1550 -6230  
 1600 2551 7763 -15  
 1601 2552 2165 -5613  
 1602 2553 7764 -14  
 1603  
 1604  
 1605  
 1606  
 1607 2554 7415 BR560, -363  
 1608 2555 7777 -1  
 1609 2556 7453 -325  
 1610 2557 7777 -1  
 1611 2560 7503 -275  
 1612 2561 7777 -1  
 1613 2562 7526 -252  
 1614 2563 7777 -1  
 1615 2564 7545 -233  
 1616 2565 7777 -1  
 1617 2566 7562 -216  
 1618 2567 7777 -1  
 1619  
 1620  
 1621  
 1622 2570 7342 BR667, -436  
 1623 2571 7777 -1  
 1624 2572 7406 -372  
 1625 2573 7777 -1  
 1626 2574 7442 -336  
 1627 2575 7777 -1  
 1628 2576 7470 -310  
 1629 2577 7777 -1  
 1630 2600 7512 -266  
 1631 2601 7777 -1  
 1632 2602 7531 -247  
 1633 2603 7777 -1  
 1634  
 1635  
 1636  
 1637 2604 7154 BR1050, -624  
 1638 2605 7776 -2  
 1639 2606 0236 -7542  
 1640 2607 7777 -1  
 1641 2610 1124 -6654  
 1642 2611 7777 -1  
 1643 2612 1662 -6116

/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  
 /BAUD RATE CONSTANTS FOR 56.8 BAUD  
 /7 BITS AT 8.11 CHAR/SEC =243 CHAR/30 SEC  
 /8 BITS AT 7.10 CHAR/SEC =213 CHAR/30 SEC  
 /9 BITS AT 6.31 CHAR/SEC =189 CHAR/30 SEC  
 /10 BITS AT 5.68 CHAR/SEC =170 CHAR/30 SEC  
 /11 BITS AT 5.16 CHAR/SEC =155 CHAR/30 SEC  
 /12 BITS AT 4.73 CHAR/SEC =142 CHAR/30 SEC  
 /BAUD RATE CONSTANTS FOR 66.7 BAUD  
 /7 BITS AT 9.53 CHAR/SEC =286/30 SEC  
 /8 BITS AT 8.35 CHAR/SEC =250/30 SEC  
 /9 BITS AT 7.41 CHAR/SEC =222/30 SEC  
 /10 BITS AT 6.67 CHAR/SEC =200/30 SEC  
 /11 BITS AT 6.06 CHAR/SEC =182/30 SEC  
 /12 BITS AT 5.56 CHAR/SEC =167/30 SEC  
 /BAUD RATE CONSTANTS FOR 1050 BAUD  
 /7 BITS AT 150 CHAR/SEC =4500/30 SEC  
 /8 BITS AT 131.25 CHAR/SEC =3938/30 SEC  
 /9 BITS AT 116.66 CHAR/SEC =3500/30 SEC  
 /10 BITS AT 105 CHAR/SEC =3150/30 SEC

1644 2613 7777 -1  
 1645 2614 2320 -5460  
 1646 2615 7777 -1  
 1647 2616 2677 -5101  
 1648 2617 7777 -1  
 1649  
 1650  
 1651 2620 0000 XDELAY, 0  
 1652 2621 7300 CLA CLL  
 1653 2622 1047 TAD NDELAY  
 1654 2623 3243 DCA DELAYN  
 1655 2624 1244 DELLOP, TAD CON100  
 1656 2625 3245 DCA US100  
 1657 2626 2245 ISZ US100  
 1658 2627 5226 JMP .-1  
 1659 2630 7200 CLA  
 1660 2631 7200 CLA  
 1661 2632 7200 CLA  
 1662 2633 7200 CLA  
 1663 2634 2243 ISZ DELAYN  
 1664 2635 7610 SKP CLA  
 1665 2636 5620 JMP I XDELAY  
 1666 2637 0620 AND I XDFLAY  
 1667 2640 0620 AND I XDELAY  
 1668 2641 0220 AND XDELAY  
 1669 2642 5224 JMP DELLOP  
 1670  
 1671 2643 0000 DELAYN, 0  
 1672 2644 7754 CON100, -24  
 1673 2645 0000 US100, 0  
 1674  
 1675  
 1676  
 1677

/11 BITS AT 95.45 CHAR/SEC =2864/30 SEC  
 /12 BITS AT 87.5 CHAR/SEC =2625/30 SEC

/IF FILLER CHARACTER OPTION IS SELECTED-DO NOT TRANSMIT A L,F,  
 /FILLER CHARACTERS WILL BE CHECKED LATER.

1678  
 1679 2646 0000 FILCHK, 0  
 1680 2647 3270 DCA CHKFIL  
 1681 2650 1036 TAD SAVBTS  
 1682 2651 7006 RTL  
 1683 2652 7700 SMA CLA  
 1684 2653 5263 JMP .+10  
 1685 2654 1270 TAD CHKFIL  
 1686 2655 1377 TAD (-12  
 1687 2656 7450 SNA  
 1688 2657 5265 JMP .+6  
 1689 2660 1376 TAD (-200  
 1690 2661 7650 SNA CLA  
 1691 2662 5265 JMP .+3  
 1692 2663 2246 ISZ FILCHK  
 1693 2664 5646 JMP I FILCHK  
 1694 2665 2052 ISZ TSTCNT  
 1695 2666 5646 JMP I FILCHK  
 1696 2667 5646 JMP I FILCHK  
 1697  
 1698 2670 0000 CHKFIL, 0

/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

1699  
 1700 2671 0000 WATTSF, 0  
 1701 2672 1172 TAD [-100  
 1702 2673 3051 DCA CNT2  
 1703 2674 3050 DCA CNT1  
 1704 2675 6041 TSF15, TSF  
 1705 2676 7610 SKP CLA  
 1706 2677 5305 JMP .+6  
 1707 2700 2050 ISZ CNT1  
 1708 2701 5275 JMP .-4  
 1709 2702 2051 ISZ CNT2  
 1710 2703 5275 JMP .-6  
 1711 2704 5671 JMP I WATTSF  
 1712 2705 2271 ISZ WATTSF  
 1713 2706 5671 JMP I WATTSF  
 1714  
 1715 2707 0000 WATKSF, 0  
 1716 2710 1172 TAD [-100  
 1717 2711 3051 DCA CNT2  
 1718 2712 3050 DCA CNT1  
 1719 2713 6031 KSF8, KSF  
 1720 2714 7610 SKP CLA  
 1721 2715 5323 JMP .+6  
 1722 2716 2050 ISZ CNT1  
 1723 2717 5313 JMP .-4  
 1724 2720 2051 ISZ CNT2  
 1725 2721 5313 JMP .-6  
 1726 2722 5707 JMP I WATKSF  
 1727 2723 2307 ISZ WATKSF  
 1728 2724 5707 JMP I WATKSF  
 1729  
 1730  
 1731 2776 7600  
 1732 2777 7766  
 1733 3000 PAGE

/ROUTINE TO WAIT FOR TRANSMIT FLAG,  
 /IF IT DOESN'T SET IN A SECOND OR SO  
 /TIMEOUT AND HALT  
 /TSF FAILED TO SKIP  
 /OK,TSF SKIPPED RETURN  
 /ROUTINE TO WAIT FOR THE RECEIVE FLAG  
 /IF IT DOESN'T SET IN A SECOND OR SO  
 /TIMEOUT AND HALT.  
 /ERROR,KSF FAILED TO SKIP  
 /OK,RECEIVE FLAG SKIPPED

1733  
 1734  
 1735 3000 4471 TYINTR, MESSAGE  
 1736 3001 4203 MESS1  
 1737 3002 4473 TWOCT  
 1738 3003 5200 JMP .-3  
 1739 3004 4511 BSWAP  
 1740 3005 3035 DCA DEVCOD  
 1741 3006 4471 MESSAGE  
 1742 3007 4213 MESS2  
 1743 3010 4473 TWOCT  
 1744 3011 5206 JMP .-3  
 1745 3012 1035 TAD DEVCOD  
 1746 3013 3035 DCA DEVCOD  
 1747 3014 4471 MESSAGE  
 1748 3015 4224 MESS3  
 1749 3016 4505 YESRNO  
 1750 3017 5214 JMP .-3  
 1751 3020 7610 SKP CLA  
 1752 3021 7330 CLA CLL CML RAR

/INTERAGATION SETUP FOR THE TELETYPE  
 /TYPE RECEIVE IOT?  
 /GET RECEIVE DEVICE CODE  
 /INPUT ERROR  
 /SWAP IT AROUND TO BITS 0-5  
 /SAVE THE RECEIVE DEVICE CODE  
 /TYPE TRANSMIT IOT  
 /GET TRANSMIT IOT  
 /INPUT ERROR  
 /ADD TRANSMIT IOT TO RECEIVE IOT  
 /SAVE THE IOTS  
 /TYPE PARITY(Y OR N)?  
 /WAIT FOR A YES OR NO  
 /NOT A Y OR N  
 /SET NO PARITY BIT  
 /SET THE PARITY BIT TO A 1

```

1753 3022 3036 DCA SAVBTS /SAVE THE PARITY BIT IN STATUS WORD
1754 3023 4471 MESSAGE
1755 3024 4236 MESS3A
1756 3025 1036 TAD SAVBTS
1757 3026 7710 SPA CLA
1758 3027 7001 IAC
1759 3030 4475 PRNT1
1760 3031 4471 MESSAGE /TYPE STATUS REGISTER(Y OR N)
1761 3032 4241 MESS4
1762 3033 4505 YESRNO /WAIT FOR A YES OR NO
1763 3034 5231 JMP .-3 /NOT A YES OR NO
1764 3035 7610 SKP CLA /NO STATUS REGISTER SELECTED
1765 3036 7332 CLA CLL CML RTR /STATUS REGISTER IS SELECTED
1766 3037 1036 TAD SAVBTS /ADD STATUS REGISTER BIT TO WORD
1767 3040 3036 DCA SAVBTS /AND SAVE IT
1768 3041 4471 MESSAGE
1769 3042 4302 MESS4A
1770 3043 1036 TAD SAVBTS
1771 3044 7004 RAL
1772 3045 7710 SPA CLA
1773 3046 7001 IAC
1774 3047 4475 PRNT1
1775 3050 4471 MESSAGE /TYPE FILLER CHARACTERS(Y OR N)?
1776 3051 4306 MESS5
1777 3052 4505 YESRNO /WAIT FOR A YES OR NO
1778 3053 5250 JMP .-3 /NOT A YES OR NO
1779 3054 7610 SKP CLA /NO FILLER CHARACTERS
1780 3055 1377 TAD (1000) /YES,FILLER CHARACTERS
1781 3056 1036 TAD SAVBTS /ADD THE FILLER CHARACTER BIT TO STATUS WORD
1782 3057 3036 DCA SAVBTS /
1783 3060 4471 MESSAGE
1784 3061 4326 MESS5A
1785 3062 1036 TAD SAVBTS
1786 3063 7006 RTL
1787 3064 7710 SPA CLA
1788 3065 7001 IAC
1789 3066 4475 PRNT1
1790 3067 4471 MESSAGE /TYPE BAUD RATE(00-13)?
1791 3070 4400 MESS6
1792 3071 4471 MESSAGE
1793 3072 4440 CMESS6
1794 3073 4473 TWOOCT /INPUT A NUMBER FROM 00-13
1795 3074 5267 JMP .-5 /INPUT ERROR
1796 3075 3364 DCA SAVIT /SAVE THE NUMBER TYPED BY OPERATOR
1797 3076 1364 TAD SAVIT /WAS THE NUMBER WITHIN BAUD RATE LIMITS
1798 3077 1376 TAD (-13)
1799 3100 7740 SMA SZA CLA
1800 3101 5267 JMP .-12 /NOT WITHIN LIMITS GO TYPE MESSAGE OVER
1801 3102 1364 TAD SAVIT /GET THE NUMBER AND PUT IN SAVBTS
1802 3103 7106 CLL RTL
1803 3104 7004 RAL
1804 3105 1036 TAD SAVETS /PUT NUMBER IN BITS 5 6 7 AND 8
1805 3106 3036 DCA SAVETS /ADD IT TO THE STATUS WORD
1806 3107 1036 TAD SAVETS /AND SAVE IT
1807 3110 7012 RTR

```

```

1808 3111 7010 RAP
1809 3112 0375 AND (17)
1810 3113 1374 TAD (MESTAB)
1811 3114 3320 DCA CHGMES
1812 3115 1720 TAD I CHGMES
1813 3116 3320 DCA CHGMES
1814 3117 4471 MESSAGE
1815 3120 4473 CHGMES, MESS6A
1816 3121 4471 MESSAGE
1817 3122 5100 MESS7
1818 3123 4505 YESRNO /TYPE TWO STOP BITS?
1819 3124 5321 JMP .-3 /WAIT FOR A Y OR N
1820 3125 7610 CLA /INPUT ERROR
1821 3126 1373 TAD (4) /ONLY 1 STOP BIT
1822 3127 1036 TAD SAVBTS /2 STOP BITS
1823 3130 3036 DCA SAVBTS /ADD THE NUMBER OF STOP BITS
1824 3131 4471 MESSAGE /TO THE STATUS WORD
1825 3132 5115 MESS7A
1826 3133 1036 TAD SAVETS
1827 3134 0373 AND (4)
1828 3135 7650 SMA CLA
1829 3136 7001 IAC
1830 3137 4475 PRNT1
1831 3140 4471 MESSAGE /TYPE # OF DATA BITS (0,1,2,3)
1832 3141 5120 MESS10
1833 3142 4472 ONEOCT
1834 3143 5340 JMP .-3 /WAIT FOR A NUMBER
1835 3144 3320 DCA CHGMES /NOT A NUMBER
1836 3145 1372 TAD (-3) /SAVE THE NUMBER
1837 3146 1320 TAD CHGMES /IS IT GREATER THAN 3
1838 3147 7740 SMA SZA CLA
1839 3150 5340 JMP .-10 /YES, TRY AGAIN
1840 3151 1320 TAD CHGMES /
1841 3152 1036 TAD SAVETS /ADD IT TO STATUS WORD
1842 3153 3036 DCA SAVETS /AND SAVE IT
1843 3154 1371 TAD (MESTB1)
1844 3155 1320 TAD CHGMES
1845 3156 3362 DCA .+4
1846 3157 1762 TAD I .+3
1847 3160 3362 DCA .+2
1848 3161 4471 MESSAGE
1849 3162 5145 MESS10A
1850 3163 5770 JMP SETUP /GO AND CALCULATE IT
1851
1852 3164 0000 SAVIT, 0
1853 3170 0215
1854 3171 2364
1855 3172 7775
1856 3173 0004
1857 3174 2350
1858 3175 0017
1859 3176 7765
1860 3177 1000
1861 PAGE 3200

```

```

1862
1863
1864
1865
1866
1867      3200      PAGE
1868
1869      3200 0000      FILLER, 0          /SET TO NUMBER OF FILLERS REQUIRED
1870
1871      /INPUT ONE OCTAL NUMBER TO AC 9 THRU 11
1872      /GOOD RETURN IS JMS+2
1873
1874      3201 0000      ONEOCK, 0          /CALL BY "ONEOCK"
1875      3202 4470      LISN
1876      3203 0001      1
1877      3204 3207      ,+3
1878      3205 0000      0
1879      3206 3210      ,+2
1880      3207 2201      ISZ      ONEOCK
1881      3210 5601      JMP I      ONEOCK
1882
1883      /INPUT TWO OCTAL NUMBERS TO AC 6 THRU 11
1884      /GOOD RETURN IS JMS+2
1885
1886      3211 0000      TWOOCK, 0          /CALL BY "TWOOCK"
1887      3212 4201      JMS      ONEOCK
1888      3213 5611      JMP I      TWOOCK
1889      3214 7104      CLL RAL
1890      3215 7006      RTL
1891      3216 3224      DCA      XPRNT2
1892      3217 4201      JMS      ONEOCK
1893      3220 5611      JMP I      TWOOCK
1894      3221 1224      TAD      XPRNT2
1895      3222 2211      ISZ      TWOOCK
1896      3223 5611      JMP I      TWOOCK
1897
1898      /PRINT THE TWO OCTAL NUMBERS IN THE AC 6 THRU 11
1899
1900      3224 0000      XPRNT2, 0          /CALL BY "PRNT2"
1901      3225 3211      DCA      TWOOCK
1902      3226 1211      TAD      TWOOCK
1903      3227 7012      RTR
1904      3230 7010      RAR
1905      3231 4475      PRNT1
1906      3232 1211      TAD      TWOOCK
1907      3233 4475      PRNT1
1908      3234 5624      JMP I      XPRNT2
1909
1910      /TYPE THE ASCII CHARACTER IN THE AC
1911
1912      3235 0000      XTYPE, 0          /CALL BY "TYPE"
1913      3236 6046      TLS
1914      3237 7200      CLA
1915      3240 6041      TSF
1916      3241 5240      JMP      ,=1

```

```

1917      3242 6042      TCF
1918      3243 5635      JMP I      XTYPE
1919
1920
1921
1922      /TYPE A CR AND LF WITH NUMBER OF FILLERS
1923      /AS DETERMINED BY LOCATION "FILLER"
1924
1925      3244 0000      XCRLF, 0          /CALL BY "CRLF"
1926      3245 7200      CIA
1927      3246 1260      TAD      K215
1928      3247 4501      TYPE
1929      3250 1200      TAD      FILLER
1930      3251 7040      CMA
1931      3252 3261      DCA      XORS
1932      3253 1057      TAD      K212
1933      3254 4501      TYPE
1934      3255 2261      ISZ      XORS
1935      3256 5254      JMP      ,=2
1936      3257 5644      JMP I      XCRLF
1937      3260 0215      K215, 0215
1938
1939      /PERFORM THE XOR OF THE AC AND THE CALL+1
1940      /RETURN TO CALL+2
1941
1942      3261 0000      XORS, 0          /CALL BY "XOR"
1943      3262 3273      DCA      YESRNX
1944      3263 1273      TAD      YESRNX
1945      3264 0661      AND I      XORS
1946      3265 7041      CIA
1947      3266 7044      CLL RAL
1948      3267 1273      TAD      YESRNX
1949      3270 1661      TAD I      XORS
1950      3271 2261      ISZ      XORS
1951      3272 5661      JMP I      XORS
1952
1953      /LOOK FOR "Y" OR "N" INPUT
1954
1955      3273 0000      YESRNX, 0          /CALL BY "YESRNO"
1956      3274 4470      LISN
1957      3275 7447      =Y
1958      3276 3303      ,+5          /RETURN TO CALL+3 IF "Y"
1959      3277 7462      =N
1960      3300 3304      ,+4          /RETURN TO CALL+2 IF "N"
1961      3301 0000      0
1962      3302 3305      ,+3
1963      3303 2273      ISZ      YESRNX
1964      3304 2273      ISZ      YESRNX
1965      3305 5673      JMP I      YESRNX
1966
1967      /PRINT 2 SPACES
1968
1969      3306 0000      SPACX2, 0          /CALL BY "SPACE2"
1970      3307 4471      MESSAGE
1971      3310 3312      ,+2

```

```

1972 3311 5706 JMP I SPACX2
1973 3312 4040 4040
1974 3313 0010 K10, 0010 /USED BY LISN
1975
1976
1977 /COMPARE INPUT TO LIST FOLLOWING CALL
1978 /INPUT ONE CHARACTER IF AC=0
1979 /USE LAST INPUT IF AC NON ZERO
1980
1981 3314 0000 XLISN, 0 /CALL BY "LISN"
1982 3315 7640 SZA CLA
1983 3316 5340 JMP LISN1 /USE LAST INPUT SINCE AC NOT ZERO
1984 3317 6031 KSF
1985 3320 5317 JMP .-1
1986 3321 6036 KRB
1987 3322 0372 AND K177
1988 3323 1125 TAD K200
1989 3324 3706 DCA I LISN1
1990 3325 1706 TAD I LISN1
1991 3326 1373 TAD M212
1992 3327 7450 SNA /IS IT A LF?
1993 3330 5334 JMP .+4 /YES
1994 3331 1374 TAD M3
1995 3332 7640 SZA CLA /IS IT A CR?
1996 3333 5336 JMP .+3 /NO
1997 3334 4502 CRLF
1998 3335 5340 JMP LISN1
1999 3336 1706 TAD I LISN1
2000 3337 4501 TYPE /PRINT THE CHARACTER
2001 3340 1714 LISN1, TAD I XLISN /GET COMPARE VALUE
2002 3341 2314 ISZ XLISN
2003 3342 7450 SNA
2004 3343 5351 JMP LISN3 /EXIT?
2005 3344 7500 SMA /YES
2006 3345 5361 JMP LISNUM /LOOK FOR OCTAL NUMBER
2007 3346 1706 TAD I LISN1 /COMPARE
2008 3347 7640 SZA CLA /EQUAL?
2009 3350 5356 JMP LISN2 /NO
2010 3351 3571 LISN3, DCA I [XPRNT1
2011 3352 1714 TAD I XLISN
2012 3353 3314 DCA XLISN
2013 3354 1571 TAD I [XPRNT1
2014 3355 5714 JMP I XLISN /AC IS ZERO UNLESS OCTAL NUMBER
2015 3356 7200 LISN2, CLA
2016 3357 2314 ISZ XLISN
2017 3360 5340 JMP LISN1 /LOOK FOR OCTAL NUMBER
2018 3361 7200 LISNUM, CLA
2019 3362 1706 TAD I LISN1
2020 3363 1367 TAD M270
2021 3364 7500 SMA /IS IT LESS THAN 8?
2022 3365 5356 JMP LISN2 /NO, SO NOT AN OCTAL NUMBER
2023 3366 1313 TAD K10
2024 3367 7510 M270, SPA /IS IT GREATER THAN ZERO?
2025 3370 5356 JMP LISN2 /NO, SO NOT A NUMBER
2026 3371 5351 JMP LISN3
    
```

```

2027 3372 0177 K177, 0177
2028 3373 7566 M212, 7566
2029 3374 7775 M3, 7775
2030 0170 LISN1=[SPACX2
2031 3400 PAGE
2032
2033
2034
2035 /PRINT PACKED ASCII TEXT TERMINATED BY
2036 /SIX-BIT 00
2037
2038 3400 0000 MESAGX, 0 /CALL BY "MESSAGE"
2039 3401 7200 CLA
2040 3402 1600 TAD I MESAGX
2041 3403 3264 DCA FOROCK
2042 3404 2700 ISZ MESAGX /SET UP RETURN
2043 3405 1664 TAD I FOROCK
2044 3406 7012 RTR
2045 3407 7012 RTR
2046 3410 7012 RTR
2047 3411 4216 JMS MESAGF
2048 3412 1664 TAD I FOROCK
2049 3413 4216 JMS MESAGF
2050 3414 2264 ISZ FOROCK
2051 3415 5205 JMP .-10
2052 3416 0000 MESAGF, 0
2053 3417 0235 AND K77
2054 3420 7450 SNA /TERMINATOR (00)?
2055 3421 5600 JMP I MESAGX /YES
2056 3422 1236 TAD M43
2057 3423 7450 SNA /CRLF?
2058 3424 5233 JMP .+7 /YES
2059 3425 1237 TAD K3
2060 3426 7510 SPA /200 OR 300
2061 3427 1240 TAD K100 /300
2062 3430 1241 TAD K240 /200
2063 3431 4501 TYPE
2064 3432 5616 JMP I MESAGF
2065 3433 4502 CRLF
2066 3434 5616 JMP I MESAGF
2067 3435 0077 K77, 0077
2068 3436 7735 M43, 7735
2069 3437 0003 K3, 0003
2070 3440 0100 K100, 0100
2071 3441 0240 K240, 0240
2072
2073 /MODIFY DEVICE CODE FOR A LIST OF IOT ADDRESSES
2074 3442 0000 XM10T, 0 /CALL BY "M10T"
2075 3443 0262 AND K770
2076 3444 3200 DCA MESAGX
2077 3445 1642 TAD I XM10T
2078 3446 2242 ISZ XM10T
2079 3447 3264 DCA FOROCK
2080 3450 1664 TAD I FOROCK /GET NEXT ADDRESS
2081 3451 7450 SNA /END OF LIST? (ZERO)
    
```

```

2082 3452 5642      JMP I  XMIOT      /YES
2083 3453 3300      DCA   XPRNT4
2084 3454 1700      TAD I  XPRNT4      /GET IOT
2085 3455 0263      AND   K7007        /REMOVE OLD DEVCIE CODE
2086 3456 1200      TAD   MESAGX       /ADD NEW DEVICE CODE
2087 3457 3700      DCA I  XPRNT4      /PUT BACK IOT
2088 3460 2264      ISZ   FOROCK
2089 3461 5250      JMP   .-11
2090
2091
2092
2093 3462 0770      K770, 0770
2094 3463 7007      K7007, 7007
2095
2096                /INPUT 4 OCTAL NUMBERS TO AC
2097                /GOOD RETURN IS CALL+2
2098
2099 3464 0000      FOROCK, 0          /CALL BY "FOROCT"
2100 3465 4473      TWOOCT
2101 3466 5664      JMP I  FOROCK
2102 3467 7106      CLL  RTL
2103 3470 7006      RTL
2104 3471 7006      RTL
2105 3472 3300      DCA   XPRNT4
2106 3473 4473      TWOOCT
2107 3474 5664      JMP I  FOROCK
2108 3475 1300      TAD   XPRNT4
2109 3476 2264      ISZ   FOROCK
2110 3477 5664      JMP I  FOROCK
2111
2112
2113                /PRINT FOUR OCTAL NUMBERS IN AC 11 THRU 0 FOLLOWED
2114                /BY TWO SPACES
2115
2116 3500 0000      XPRNT4, 0         /CALL BY "PRNT4"
2117 3501 3264      DCA   FOROCK
2118 3502 1264      TAD   FOROCK
2119 3503 7012      RTR
2120 3504 7012      RTR
2121 3505 7012      RTR
2122 3506 4476      PRNT2
2123 3507 1264      TAD   FOROCK
2124 3510 4476      PRNT2
2125 3511 4500      SPACE2
2126 3512 5700      JMP I  XPRNT4
2127
2128                /
2129                /PRINT THE OCTAL NUMBER IN AC 9 THRU 11
2130 3513 0000      XPRNT1, 0        /CALL BY "PRNT1"
2131 3514 0320      AND   K7
2132 3515 1321      TAD   K260
2133 3516 4501      TYPE
2134 3517 5713      JMP I  XPRNT1
2135 3520 0007      K7, 0007
2136 3521 0260      K260, 0260
2137

```

```

2137                /SWAP BITES IN THE AC, PRESERVE THE LINK
2138                /
2139 3522 0000      XBSW, 0          /CALL BY "BSWAP"
2140 3523 3337      DCA  XRAND
2141 3524 7012      RTR
2142 3525 7012      RTR
2143 3526 7012      RTR
2144 3527 1337      TAD   XRAND
2145 3530 0336      AND   K7700
2146 3531 1337      TAD   XRAND
2147 3532 7006      RTL
2148 3533 7006      RTL
2149 3534 7006      RTL
2150 3535 5722      JMP I  XBSW
2151 3536 7700      K7700, 7700
2152
2153                /
2154                /GENERATE RANDOM NUMBER
2155                /EXIT WITH NUMBER IN AC
2156
2157 3537 0000      XRAND, 0          /CALL BY "RANDOM"
2158 3540 7301      CLA  CLL IAC
2159 3541 1373      TAD   RAN1
2160 3542 1374      TAD   RAN2
2161 3543 7106      CLL  RTL
2162 3544 3373      DCA   RAN1
2163 3545 1374      TAD   RAN2
2164 3546 7012      RTR
2165 3547 1373      TAD   RAN1
2166 3550 3374      DCA   RAN2
2167 3551 1374      TAD   RAN2
2168 3552 0040      AND   DATBIT      /MASK CHAR FOR I/O
2169 3553 3042      DCA  XMTDAT      /STORE
2170 3554 5737      JMP I  XRAND
2171
2172                /SAVE RANDOM
2173                /GENERATOR PRIMES
2174
2175 3555 0000      XSAVGN, 0        /CALL BY "SAVGEN"
2176 3556 7200      CLA
2177 3557 1373      TAD   RAN1
2178 3560 3375      DCA  SAV1
2179 3561 1374      TAD   RAN2
2180 3562 3053      DCA  SAV2
2181 3563 5755      JMP I  XSAVGN
2182
2183                /RESTORE RANDOM
2184                /GENERATOR PRIMES
2185
2186 3564 0000      XRESGN, 0        /CALL BY "RESGEN"
2187 3565 7200      CLA
2188 3566 1375      TAD   SAV1
2189 3567 3373      DCA  RAN1
2190 3570 1053      TAD   SAV2
2191 3571 3374      DCA  PAN2

```



```

2192 3572 5764      JMP I  XRESGN
2193 3573 1234     RAN1, 1234
2194 3574 5670     RAN2, 5670
2195 3575 0000     SAV1, 0
2196
2197 /
2198 / PAGE
2199 /
2200 /ROUTINE TO DETERMINE IF ON APT-8 TEST SYSTEM.
2201 /IF ON APT-8 NO HALTS WILL BE RECOGNIZED AND ALL PERTINENT INFORMATION
2202 /WILL BE CONTAINED IN ADDRESSES 20 AND 22.
2203 /
2204 3600 0000     XAPT8, 0
2205 3601 1022     TAD 22
2206 3602 7104     CLL RAL
2207 3603 7520     SMA SNL
2208 3604 5210     JMP ONAPT-3      /ON APT8 OR MULTIPLE OPTION TESTER
2209 3605 7710     SPA CLA        /NO, GO CHECK FOR CONSOLE.
2210 3606 5216     JMP ONAPT+3     /SKIP IF NOT ON TESTER
2211 3607 5213     JMP ONAPT
2212 /
2213 3610 4515     CHEK22
2214 3611 5777     JMP SETUP-7
2215 3612 5776     JMP BGNINT+3
2216 3613 1020     ONAPT, TAD 20   /GET DEVICE CODE TO USE.
2217 3614 7440     SZA 20         /TEST TO SEE IF A DVICE CODE IS THERE
2218 3615 3035     DCA DEVCOD
2219 3616 1022     TAD 22
2220 3617 0375     AND (1000)
2221 3620 7104     CLL RAL
2222 3621 3010     DCA 10
2223 3622 1022     TAD 22
2224 3623 0125     AND K200
2225 3624 7006     RTL
2226 3625 1010     TAD 10
2227 3626 3010     DCA 10
2228 3627 1021     TAD 21
2229 3630 0374     AND (2000)
2230 3631 7104     CLL RAL
2231 3632 1010     TAD 10
2232 3633 3010     DCA 10
2233 /
2234 3634 1022     TAD 22
2235 3635 0033     AND C177
2236 3636 1010     TAD 10
2237 3637 3036     DCA SAVBTS
2238 3640 1022     TAD 22
2239 3641 0373     AND (6000)
2240 3642 3022     DCA 22
2241 3643 1036     TAD SAVBTS
2242 3644 7010     RAR
2243 3645 7012     RTR
2244 3646 0372     AND (17
2245 3647 3024     DCA SAVPNT
2246 3650 1024     TAD SAVPNT

```

```

2247 3651 1371     TAD (TABLE1
2248 3652 3124     DCA TEMP
2249 3653 1524     TAD I TEMP
2250 3654 3271     DCA XAERRO
2251 3655 1271     TAD XAERRO
2252 3656 3124     DCA TEMP
2253 3657 7240     STA
2254 3660 1271     TAD XAERRO
2255 3661 3011     DCA A11
2256 3662 1671     TAD I XAEPRO
2257 3663 3121     DCA CLFCNT
2258 3664 1022     TAD 22
2259 3665 7004     RAL
2260 3666 7710     SPA CLA
2261 3667 6370     APTIO0
2262 3670 5770     JMP SETUP
2263 /
2264 /
2265 /ROUTINE TO REPORT ERROR TO APT-8 IF REQUIRED.
2266 /ONLY THE ERROR PC WILL BE ESTABLISHED.
2267 /
2268 /
2269 3671 0000     XAERRO, 0
2270 3672 7440     SZA
2271 3673 5276     JMP .+3
2272 3674 7240     STA
2273 3675 1271     TAD XAERRO
2274 3676 3010     DCA 10
2275 3677 1022     TAD 22
2276 3700 7004     RAL
2277 3701 7520     SMA SNL
2278 3702 5324     JMP EXIT
2279 3703 6002     IOF
2280 3704 7700     SMA CLA
2281 3705 5316     JMP APTCHK
2282 3706 6372     APTIO2
2283 3707 7420     SNL
2284 3710 5314     JMP .+4
2285 3711 4517     LAS
2286 3712 7710     SPA CLA
2287 3713 5320     JMP GOTOUV
2288 3714 4767     JMS NEWDEV
2289 3715 5766     JMP CLRBRD
2290 /
2291 3716 7420     APTCHK, SNL
2292 3717 5324     JMP EXIT
2293 /
2294 3720 1010     GOTOUV, TAD 10
2295 3721 6272     CIF 70
2296 3722 5743     JMP I K6520
2297 3723 5323     JMP .
2298 /
2299 3724 7300     EXIT, CLL CLA
2300 3725 4515     CHEK22
2301 3726 7410     SKP

```

/TEST FOR ACTIVE CONSOLE  
/CONSOLE ACTIVE.

```

2302 3727 5341      JMP      .+12
2303 3730 4517      LAS
2304 3731 7710      SPA CLA      /HALT ON ERROR SELECTED?
2305 3732 5341      JMP      .+7      /NO, TAKE EXIT.
2306 3733 4471      MESSAGE
2307 3734 4200      MPC      /PRINT PC=
2308 3735 1010      TAD      10      /GET BACK ERROR PC
2309 3736 4477      PRNT4     /PPINT IT
2310 3737 4765*     JMS      XC8PSW   /ASK SR QUESTION
2311 3740 5671      JMP I    XAERRO
2312 3741 2271      ISZ     XAERRO
2313 3742 5671      JMP I    XAERRO
2314 3743 6520      K6520, 6520
2315
2316 /
2317 /ROUTINE TO LOOP ON NO ERROR FOR THE DESIRED NUMBER OF TIMES.
2318 /THIS ALSO CONTAINS THE APT TIMING SECTION.
2319 /
2320 XNERRO, 0
2321 TAD I    XNERRO   /GET TEST TO LOOP ON
2322 DCA     RETURN   /STORE FOR RETURN.
2323 ISZ     XNERRO   /UPDATE RETURN.
2324 TICK
2325 CHEK22 /APT TIMING
2326 SKP
2327 JMP     .+5      /CONSOLE ACTIVE
2328 KSF
2329 JMP     .+3      /YES, CHECK FOR KEYBOARD FLAG.
2330 KRB
2331 JMS     XC8CNT   /KEYBOARD FLAG SET?
2332 ISZ     TSTCNT  /NO
2333 JMP I    RETURN  /YES, READ THE CHARACTER
2334 JMP I    XNERRO  /CHECK FOR CONTROL CHARACTER
2335 /
2336 RETURN, 0
2337 /
2338 3764 6600
2339 3765 6703
2340 3766 0271
2341 3767 4000
2342 3770 0215
2343 3771 5500
2344 3772 0017
2345 3773 6000
2346 3774 2000
2347 3775 1000
2348 3776 0203
2349 3777 0206
2350 PAGE
2351 /THIS ROUTINE WILL UPDATE THE DEVICE NUMBER TO TEST IF
2352 /THE PROGRAM IS BEING RUN FOR MULTIPLE TESTING.
2353 /
2354 NEWDEV, 0
2355 CAF
2356 TAD      20

```

```

2356 4003 7002      BSW
2357 4004 7710      SPA CLA      /MOVE BIT 6 INTO BIT 0
2358 4005 6371      APTIO1     /TEST ALL 16 DEVICES?
2359 4006 6371      APTIO1     /NO, ONLY EVERY OTHER ONE
2360 4007 6374      APTIO4
2361 4010 7410      SKP
2362 4011 5201      JMP     NEWDEV+1 /IS ERROR INDICATOR SET ON THIS ONE
2363 4012 5600      JMP I    NEWDEV  /NO, JUST EXIT ROUTINE
2364 /
2365 /
2366 /ROUTINE TO NOTIFY APT THAT THE PROGRAM IS RUNNING.
2367 /
2368 XTICK, 0
2369 TAD      22
2370 AND     K4000
2371 SNA CLA      /ON APT??
2372 JMP I    XTICK  /NO, EXIT.
2373 ISZ     CLKCNT /READY.
2374 JMP I    XTICK  /NO, EXIT.
2375 TAD     COUNT  /INIT COUNTER
2376 DCA     CLKCNT
2377 IOF
2378 CDF
2379 CIF     70
2380 JMS     K6500
2381 JMP I    XTICK
2382 /
2383 K6500, 6500
2384 /
2385 /SETS UP LOOP COUNTER AND LOOP TEST POINTER.
2386 /
2387 XPCRET, 0
2388 CLA     CMA
2389 TAD     XPCRET
2390 DCA     LOOPPC
2391 TAD     M10
2392 DCA     TSTCNT
2393 JMP I    XPCRET
2394 /
2395 M10, -10
2396 /
2397 /
2398 /
2399 /LOOP ON TEST IF SR2=1
2400 /
2401 XSR2, 0
2402 LAS
2403 RTL
2404 SPA CLA
2405 JMP I    LOOPPC
2406 JMP I    XSR2
2407 /
2408 NOTBE, 0
2409 TAD     22
2410 SMA     CLA

```

```

2411 4053 5256      JMP      .+3
2412 4054 1021      TAD      21
2413 4055 7410      SKP
2414 4056 4517      LAS
2415 4057 7012      RTR
2416 4060 7010      RAR
2417 4061 7710      SPA      CLA
2418 4062 5265      JMP      .+3
2419 4063 2250      ISZ      NOT8E
2420 4064 5650      JMP I    NOT8E
2421 4065 1650      TAD I    NOT8E
2422 4066 3250      DCA      NOT8E
2423 4067 5650      JMP I    NOT8E
2424
2425
2426
2427
2428
2429 4070 0000      XLAS,    0
2430 4071 7300      CLL     CLA
2431 4072 1022      TAD     22
2432 4073 7710      SPA     CLA
2433 4074 5670      JMP I   XLAS
2434 4075 1021      TAD     21
2435 4076 7710      SPA     CLA
2436 4077 7614      TAD     7614
2437 4100 1020      TAD     20
2438 4101 5670      JMP I   XLAS
2439
2440
2441
2442
2443
2444 4102 0000      XHALT,  0
2445 4103 3270      DCA     XLAS
2446 4104 1022      TAD     22
2447 4105 0123      AND     K4000
2448 4106 7650      SNA     CLA
2449 4107 5311      JMP     XHLT
2450 4110 5702      JMP I   XHALT
2451
2452 4111 4515      XHLT,   CHEK22
2453 4112 5702      JMP I   XHALT
2454 4113 1270      TAD     XLAS
2455 4114 7440      SZA
2456 4115 5320      JMP     .+3
2457 4116 7340      CLL     CLA
2458 4117 1302      TAD     XHALT
2459 4120 7402      TAD     7402
2460 4121 7300      CLL     CLA
2461 4122 5702      JMP I   XHALT
2462
2463
2464
2465

```

```

2466
2467 4123 0000      XCHK22, 0
2468 4124 7300      CLL     CLA
2469 4125 1022      TAD     22
2470 4126 0332      AND     K400
2471 4127 7650      SNA     CLA
2472 4130 2323      ISZ     XCHK22
2473 4131 5723      JMP I   XCHK22
2474
2475 4132 0400      K400,   400
2476
2477
2478
2479
2480 4133 0000      XPRAND1, 0
2481 4134 7300      CLA     CLL
2482 4135 1042      TAD     XMTDAT
2483 4136 3345      DCA     TEMSAV
2484 4137 4506      RANDOM
2485 4140 1042      TAD     XMTDAT
2486 4141 3043      DCA     XMTDT1
2487 4142 1345      TAD     TEMSAV
2488 4143 3042      DCA     XMTDAT
2489 4144 5733      JMP I   XPRAND1
2490
2491 4145 0000      TEMSAV, 0
2492
2493
2494
2495
2496
2497 4200 4320      /MESSAGES
2498 4201 0375      MPC,    TEXT    "#PC= "
2499 4202 4000
2500 4203 4322      MESS1,  TEXT    "#RECEIVE IOT? "
2501 4204 0503
2502 4205 0511
2503 4206 2605
2504 4207 4011
2505 4210 1724
2506 4211 7740
2507 4212 4000
2508 4213 4324      MFSS2,  TEXT    "#TRANSMIT IOT? "
2509 4214 2201
2510 4215 1623
2511 4216 1511
2512 4217 2440
2513 4220 1117
2514 4221 2477
2515 4222 4040
2516 4223 0000
2517 4224 4320      MESS3,  TEXT    "#PARITY(Y OR N)? "
2518 4225 0122
2519 4226 1124
2520 4227 3150

```

4230 3140  
4231 1722  
4232 4016  
4233 5177  
4234 4040  
4235 0000  
2501 4236 4040 MESS3A, TEXT " NP="  
4237 1620  
4240 7500  
2502 4241 7743 MESS4, TEXT "?\*EVEN PARITY EVN=0? OOD PARITY EVN=1?\*STATUS ENABLED(Y OR N)? "  
4242 0526  
4243 0516  
4244 4020  
4245 0122  
4246 1124  
4247 3140  
4250 0526  
4251 1675  
4252 6077  
4253 4040  
4254 1704  
4255 0440  
4256 2001  
4257 2211  
4260 2431  
4261 4005  
4262 2616  
4263 7561  
4264 7743  
4265 2324  
4266 0124  
4267 2523  
4270 4005  
4271 1601  
4272 0214  
4273 0504  
4274 5031  
4275 4017  
4276 2240  
4277 1651  
4300 7740  
4301 4000  
2503 4302 4040 MESS4A, TEXT " SWD="  
4303 2327  
4304 0475  
4305 0000  
2504 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  
2505 4326 4040 MESS5A, TEXT " FIL="  
4327 0611  
4330 1475  
4331 0000  
2506 PAGE  
2507  
2508 4400 7743 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  
2509 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

2510  
 2511 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

2512 MESS6B, TEXT "#150 BAUD - B1=0? B2=0? B3=1? W2=1? W5=0?"

4520 4361  
 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

2513 MESS6C, TEXT "#300 BAUD - B1=0? B2=1? B3=0? W2=1? W5=0?"

4545 4363  
 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

2514 MESS6D, TEXT "#600 BAUD - B1=0? B2=1? B3=1? W2=1? W5=0?"

4572 4366  
 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  
 4616 7700

2515 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  
4644 0000  
2516 4645 4362 MESS6F, TEXT "#2400 BAUD - B1=1? B2=0? B3=1? W2=1? W5=0?"  
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  
4672 0000  
2517 4673 4364 MESS6G, TEXT "#4000 BAUD - B1=1? B2=1? B3=0? W2=1? W5=0?"  
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  
2518 4721 4371 MESS6H, TEXT "#9600 BAUD - B1=1? R2=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  
2519 4747 4361 MESS6I, TEXT "#19,200 BAUD - B1=1? B2=1? B3=1? W2=0? W5=1?"  
2520 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  
2521 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  
2522 5023 0000  
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  
2523 5051 0000  
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  
2524 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  
2525 5115 4040 MESS7A, TEXT " SB="   
5116 2302  
5117 7500  
2526 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  
2527  
2528 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		
2529	5164	MES10B, TEXT	"#6 DATA BITS - NR1=0? NB2=1?#"
	5165		
	5166		
	5167		
	5170		
	5171		
	5172		
	5173		
	5174		
	5175		
	5176		
	5177		
	5200		
	5201		
	5202		
2530	5203	MES10C, TEXT	"#7 DATA BITS - NR1=1? NB2=0?#"
	5204		
	5205		
	5206		
	5207		
	5210		
	5211		
	5212		
	5213		
	5214		
	5215		
	5216		
	5217		
	5220		
	5221		
2531	5222	MES10D, TEXT	"#8 DATA BITS - NR1=0? NB2=0?#"
	5223		
	5224		
	5225		
	5226		
	5227		
	5230		
	5231		
	5232		
	5233		
	5234		
	5235		
	5236		
	5237		
	5240		
2532	5241	RECPNT, KCF0	
2533	5242	KCF1	
2534	5243	KCF2	
2535	5244	KCF3	
2536	5245	KCF4	
2537	5246	KSF0	
2538	5247	KSF1	

2539	5250	0354	KSF2
2540	5251	0471	KSF3
2541	5252	0537	KSF4
2542	5253	0601	KSF5
2543	5254	0721	KSF6
2544	5255	0734	KSF7
2545	5256	2713	KSF8
2546	5257	1013	KSF9
2547	5260	1037	KSF10
2548	5261	1065	KSF11
2549	5262	1077	KSF12
2550	5263	1116	KSF14
2551	5264	1137	KSF16
2552	5265	1144	KSF17
2553	5266	1425	KSF18
2554	5267	1523	FDTL0P
2555	5270	1541	KSF19
2556	5271	1675	KSF21
2557	5272	1744	KSF22
2558	5273	1762	KSF23
2559	5274	1341	KSF24
2560	5275	2036	KSF25
2561	5276	2063	KSF26
2562	5277	0610	KCC0
2563	5300	0713	KCC1
2564	5301	1036	KCC2
2565	5302	1063	KCC3
2566	5303	1136	KCC4
2567	5304	1757	KCC5
2568	5305	2000	KCC6
2569	5306	0626	KRS0
2570	5307	1010	KRS1
2571	5310	2040	KRS2
2572	5311	2046	KRS3
2573	5312	1335	KSE0
2574	5313	2023	KSE1
2575	5314	2045	KSE2
2576	5315	2054	KSE3
2577	5316	0403	KIE0
2578	5317	0422	KIE1
2579	5320	0457	KIE2
2580	5321	0502	KIE3
2581	5322	0536	KIE4
2582	5323	0616	KIE5
2583	5324	0710	KIE6
2584	5325	0750	KIE7
2585	5326	0760	KIE8
2586	5327	1026	KIE9
2587	5330	1052	KIE10
2588	5331	0635	KRB0
2589	5332	1113	KRB1
2590	5333	1410	KRB2
2591	5334	1437	KRB3
2592	5335	1504	KRB4
2593	5336	1511	KRB5



2594	5337	1526	KRB6
2595	5340	1544	KPB7
2596	5341	1615	KRB8
2597	5342	1654	KRB9
2598	5343	1711	KRB10
2599	5344	2313	INTON
2600	5345	1351	KRB11
2601	5346	2014	KRB12
2602	5347	2060	KRB13
2603	5350	2070	KPB14
2604		/	
2605		/	
2606	5351	0000	0000
2607			
2608			
2609	5352	0323	XMTIOT, TFL0
2610	5353	0345	TFL1
2611	5354	0406	TFL2
2612	5355	0452	TFL3
2613	5356	0503	TFL4
2614	5357	0520	TFL5
2615	5360	0646	TFL6
2616	5361	0301	TSF0
2617	5362	0326	TSF1
2618	5363	0333	TSF2
2619	5364	0346	TSF3
2620	5365	0351	TSF4
2621	5366	0407	TSF5
2622	5367	0425	TSF6
2623	5370	0437	TSF7
2624	5371	0453	TSF8
2625	5372	0504	TSF9
2626	5373	0512	TSF10
2627	5374	0521	TSF11
2628	5375	0533	TSF12
2629	5376	0664	TSF13
2630	5377	0724	TSF14
2631	5400	2675	TSF15
2632	5401	1000	TSF16
2633	5402	1003	TSF17
2634	5403	1070	TSF18
2635	5404	1106	TSF20
2636	5405	1124	TSF22
2637	5406	1421	TSF23
2638	5407	1516	TSF24
2639	5410	1535	TSF25
2640	5411	1553	TSF26
2641	5412	1672	TSF28
2642	5413	1747	TSF29
2643	5414	2310	TSF30
2644	5415	2322	TSF31
2645	5416	1344	TSF32
2646	5417	0330	TCF0
2647	5420	0436	TCF1
2648	5421	0470	TCF2

2649	5422	0532	TCF3
2650	5423	0655	TCF4
2651	5424	0716	TCF5
2652	5425	1002	TCF6
2653	5426	1064	TCF7
2654	5427	1123	TCF8
2655	5430	1412	TCF9
2656	5431	1505	TCF10
2657	5432	1512	TCF11
2658	5433	1611	TCF12
2659	5434	1656	TCF13
2660	5435	1346	TCF14
2661	5436	2015	TCF15
2662	5437	2035	TCF16
2663	5440	0727	TPC0
2664	5441	1073	TPC1
2665	5442	0304	SPI0
2666	5443	0411	SPI1
2667	5444	0427	SPI2
2668	5445	0442	SPI3
2669	5446	0455	SPI4
2670	5447	0460	SPI5
2671	5450	0506	SPI6
2672	5451	0515	SPI7
2673	5452	0523	SPI8
2674	5453	0673	SPI9
2675	5454	0737	SPI10
2676	5455	0751	SPI11
2677	5456	0761	SPI12
2678	5457	1015	SPI13
2679	5460	1027	SPI14
2680	5461	1042	SPI15
2681	5462	1102	TL50
2682	5463	1420	XMIT
2683	5464	1515	TL51
2684	5465	1522	TL52
2685	5466	1540	TL53
2686	5467	1556	TL54
2687	5470	1606	TL55
2688	5471	1665	TL56
2689	5472	2307	TL57
2690	5473	2315	TL58
2691	5474	2026	TL59
2692	5475	1424	TL510
2693	5476	1340	SLWTL5
2694		/	
2695		/	
2696	5477	0000	0000
2697			/BRAUD RATE TABLES FOR USE ONLY UNDER APT-8 TEST SYSTEM.
2698			/
2699	5500	5514	TABLE1, B110
2700	5501	5523	B150
2701	5502	5532	B300
2702	5503	5541	B600
2703	5504	5550	B1200

2704	5505	5557	B2400	
2705	5506	5566	B4800	
2706	5507	5575	B9600	
2707	5510	5604	B19K2	
2708	5511	5613	B56P8	
2709	5512	5622	B66P7	
2710	5513	5631	B1050	
2711			/	
2712			/ACTUAL VALUES TO USE FOR THE APT TIMING SECTION,	
2713			/	
2714	5514	7677	B110,	-101
2715	5515	7750		-30
2716	5516	7765		-13
2717	5517	7771		-7
2718	5520	7765		-13
2719	5521	7765		-13
2720	5522	7770		-10
2721			/150 BAUD	
2722			/	
2723	5523	7677	B150,	-101
2724	5524	7750		-30
2725	5525	7760		-20
2726	5526	7765		-13
2727	5527	7761		-17
2728	5530	7761		-17
2729	5531	7770		-10
2730			/300 BAUD	
2731			/	
2732	5532	7677	B300,	-101
2733	5533	7750		-30
2734	5534	7740		-40
2735	5535	7752		-26
2736	5536	7744		-34
2737	5537	7744		-34
2738	5540	7770		-10
2739			/	
2740	5541	7647	/600 BAUD	
2741	5542	0000	B600,	-131
2742	5543	7700		0
2743	5543	7700		-100
2744	5544	7724		-54
2745	5545	7700		-100
2746	5546	7700		-100
2747	5547	0000		0
2748			/	
2749			/1200 BAUD	
2750	5550	7647	B1200,	-131
2751	5551	0000		0
2752	5552	7600		-200
2753	5553	7650		-130
2754	5554	7600		-200
2755	5555	7600		-200
2756	5556	0000		0
2757			/2400 BAUD	
2758			/	

2759	5557	7247	B2400,	-531
2760	5560	0000		0
2761	5561	0000		0
2762	5562	0000		0
2763	5563	7400		-400
2764	5564	7400		-400
2765	5565	0000		0
2766			/	
2767			/4800 BAUD	
2768	5566	7247	B4800,	-531
2769	5567	0000		0
2770	5570	0000		0
2771	5571	0000		0
2772	5572	7400		-400
2773	5573	7400		-400
2774	5574	0000		0
2775			/	
2776			/9600 BAUD	
2777	5575	6247	B9600,	-1531
2778	5576	0000		0
2779	5577	0000		0
2780	5600	0000		0
2781	5601	0000		0
2782	5602	0000		0
2783	5603	0000		0
2784			/	
2785			/19.2 KILO BAUD,	
2786	5604	6247	B19K2,	-1531
2787	5605	0000		0
2788	5606	0000		0
2789	5607	0000		0
2790	5610	0000		0
2791	5611	0000		0
2792	5612	0000		0
2793			/	
2794			/56.8 BAUD,	
2795	5613	7740	B56P8,	-40
2796	5614	7763		-15
2797	5615	7772		-6
2798	5616	7775		-3
2799	5617	7772		-6
2800	5620	7772		-6
2801	5621	7773		-5
2802			/	
2803			/66.7 BAUD	
2804	5622	7740	B66P7,	-40
2805	5623	7763		-15
2806	5624	7772		-6
2807	5625	7775		-3
2808	5626	7772		-6
2809	5627	7772		-6
2810	5630	7773		-5
2811			/	
2812			/1050 BAUD	
2813	5631	7647	B1050,	-131

```

2814 5632 0000 0
2815 5633 7600 -200
2816 5634 7650 -130
2817 5635 7600 -200
2818 5636 7600 -200
2819 5637 0000 0
2820
2821
2822 /STANDARD CONSOLE PACKAGE,
2823 /THE FOLLOWING PAPAMETERS MUST BE DEFINED PRIOR TO
2824 /ASSEMBLY.
2825 /THSFLD THIS DEFINES THE FIELD THE ROUTINE
2826 / WILL RESIDE IN
2827 /CALFLD THIS DEFINES CALLING FIELD.
2828 /ORIGIN STARTING ADDRESS OF ROUTINES
2829 /STRST DEFINES RESTART TEST FOR LINE FEED DETECTION.
2830 /
2831 0000 THSFLD=0 /FIELD 0
2832 0000 CALFLD=00 /CALLING FIELD
2833 6600 ORIGIN=6600 /START AT ADDRESS 6600
2834 0000 CONFLD=THSFLD*10
2835 0000 PRGFLD=CALFLD*10
2836 0271 STRTST=CLRRDR
2837 /
2838 0170 3306
2839 0171 3513
2840 0172 7700
2841 0173 1600
2842 0174 1400
2843 0175 2646
2844 0176 4060
2845 0177 7400
2846 0000 FIELD THSFLD

```

```

0000 11110000 01000000 11111111 11111111 11111111 11111111 11111111 11111111
0100 11111111 11111111 11111100 00000000 00000000 00000000 00000000 11111111

0200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0300 11111111 11111111 11111111 11111111 11111111 11111111 11100111 11111111

0400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0500 11111111 11111111 11111111 11111111 11111111 11111111 00000000 00000111

0600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11011111

1000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1100 11111111 11111111 11111111 11111111 11111111 11111111 11111110 00000001

1200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111110

1400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11000000

1600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

2000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2100 11111111 11111111 11111111 11111111 11111111 11111111 11111100 01111111

2200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 00000111

2400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

2600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2700 11111111 11111111 11111000 00000000 00000000 00000000 00000000 00000011

3000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3100 11111111 11111111 11111111 11111111 11111111 11111111 11111000 11111111

3200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111000

3400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111100

3600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

```

```

4000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4100 11111111 11111111 11111111 11111111 11111100 00000000 00000000 00000000
4200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4300 11111111 11111111 11111111 11000000 00000000 00000000 00000000 00000000
4400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

5000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

5200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

5400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

5600 11111111 11111111 11111111 11111111 00000000 00000000 00000000 00000000
5700 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

```

```

2847          6600  *ORIGIN
2848          /CONTROL CHARACTER DECODE ROUTINE.
2849          /ENTER WITH CHARACTER IN THE AC.
2850          XCBcnt, 0
2851          6601 0000 DCA C8CHAR /STORE CHARACTER.
2852          6602 6214 RDF /GET CALLING DATA FIELD.
2853          6603 1257 TAD KCDFIF /SET UP RETURN FIELD.
2854          6604 3241 DCA FLDRTN /STORE IT
2855          6605 3252 DCA INDEXA /SET DISPLACEMENT TO 0
2856          6606 1254 TAD XTABLA
2857          6607 3253 DCA GETDAT /SET UP POINTER TABLE.
2858          6610 6201 CDF CONFLD
2859          6611 1653 REDOA, TAD I GETDAT /GET CONTROL CHARACTER
2860          6612 7450 SNA /END OF TABLE.
2861          6613 5222 JMP DONEA /YES.
2862          6614 1360 TAD C8CHAR
2863          6615 7650 SNA CLA /GET A MATCH?
2864          6616 5243 JMP GOITA /YES.
2865          6617 2252 ISZ INDEXA /NO, UPDATE DISPLACEMENT
2866          6620 2253 ISZ GETDAT /UPDATE POINTER.
2867          6621 5211 JMP REDOA /GET NEXT COMPARE.
2868          6622 1360 DONEA, TAD C8CHAR /GET BACK CHARACTER.
2869          6623 1777 TAD C8M260
2870          6624 7700 SNA CLA
2871          6625 5234 JMP ,+7 /NO, THEN JUST PRINT IT.
2872          6626 1360 TAD C8CHAR
2873          6627 4776 JMS TSTCRL /GO TEST FOR CRLF.
2874          6630 5236 JMP FLDRTN-3 /CRLF FOUND.
2875          6631 1775 TAD ARROW
2876          6632 4774 JMS XC8TYP /PRINT "
2877          6633 1256 TAD C80100 /SET UP CHARACTER
2878          6634 1360 TAD C8CHAR
2879          6635 4774 JMS YC8TYP /PRINT IT.
2880          6636 1260 TAD C8K277
2881          6637 4774 JMS XC8TYP /PRINT ?
2882          6640 4773 JMS YC8CRL /AND <CRLF>
2883          6641 6203 FLDRTN, CIF CDF PRGFLD /RETURN FIELD.
2884          6642 5600 JMP I XC8CNT /EXIT BACK TO CALL.
2885          /
2886          6643 1255 GOITA, TAD XTABLB /GET CONTROL CHARACTER ROUTINES.
2887          6644 1252 TAD INDEXA /ADD IN OFFSET
2888          6645 3251 DCA GOTOA /STORE THIS
2889          6646 1651 TAD I GOTOA /GET ACTUAL POINTER.
2890          6647 3251 DCA GOTOA
2891          6650 5651 JMP I GOTOA /PERFORM FUNCTION.
2892          /
2893          /CONSTANTS USED BY THIS ROUTINE.
2894          /
2895          6651 0000 GOTOA, 0
2896          6652 0000 INDEXA, 0
2897          6653 0000 GETDAT, 0
2898          6654 7254 XTABLA, TABLA
2899          6655 7261 XTABLB, TABLB
2900          /
2901          /

```

```

2902 6656 0100 C80100, 100
2903 6657 6203 KCDFIF, CDF CIF
2904 6660 0277 C8K277, 277
2905 /
2906 /ABORT PRINT OUT ROUTINE.
2907 /ALL TERMINAL OUTPUT IS STOPPED UNTIL A "Q" IS INPUT
2908 /ALL OTHER CONTROL FUNCTIONS CAN BE PERFORMED.
2909 /
2910 6661 1302 CNTRLS, TAD INMODE
2911 6662 7640 SZA CLA /HAS ROUTINE BEEN ENTERED BEFORE.
2912 6663 5272 JMP .+7
2913 6664 1200 TAD XC8CNT
2914 6665 3301 DCA C8PTFN /STORE RETURN ADDRESS.
2915 6666 7240 STA
2916 6667 3302 DCA INMODE /SET UP FOR A REENTRY.
2917 6670 4772 JMS XC8TTY /GO BACK AND WAIT FOR NEXT CHARACTER.
2918 6671 4200 JMS XC8CNT /DECODE FOR CONTROL FUNCTION.
2919 6672 5270 JMP .-2 /NO CONTROL FUNTION. GO BACK AND WAIT.
2920 /
2921 /CONTROL Q FUNCTION.
2922 /CLEARS INMODE POINTER. AND SETS UP PROPER RETURN
2923 /TO CALLING FIELD.
2924 /
2925 6673 1302 CNTRLQ, TAD INMODE
2926 6674 7650 SNA CLA /HAS CONTROL S BEEN TYPED.?
2927 6675 5241 JMP FLDRTN /NO.
2928 6676 3302 DCA INMODE /CLEAR REENTRY POINTER
2929 6677 6203 ACTFLD, CIF CDF PRGFLD
2930 6700 5701 JMP I C8RTRN /AND TAKE A RETURN
2931 /
2932 6701 0000 C8RTRN, 0000 /GETS SETUP TO RETURN ADDRESS.
2933 6702 0000 INMODE, 0000
2934 /
2935 /LOADS PSEUDO-SWITCH REGISTER WITH NEW VALUE.
2936 /
2937 6703 0000 XC8PSW, 0
2938 6704 7300 CLA CLL
2939 6705 6214 RDF /GET CLING DATA FIELD
2940 6706 1257 TAD KCDFIF
2941 6707 3347 DCA PSWRTN /SET UP RETURNING FIELD
2942 6710 6201 CDF CONFLD /SET DATA FIELD TO FIELD OF CONSOLE.
2943 6711 7200 CLA
2944 6712 3357 DCA CHRTMP
2945 6713 4773 JMS XC8CRL /<CRLF>
2946 6714 4771 JMS XC8PNT
2947 6715 7265 MESA /PRINT "SR="
2948 6716 6201 CDF PRGFLD /SET UP FIELD OF PSW.
2949 6717 1770 TAD I (20 /GET PSEUDO SWITCH REGISTER.
2950 6720 6201 CDF CONFLD /RESTORE FIELDS.
2951 6721 3767 DCA C8STOR
2952 6722 1767 TAD C8STOR
2953 6723 4766 JMS XC8OCT /PRINT THE CURRENT VALUE.
2954 6724 1356 TAD C8M4
2955 6725 3765 DCA TPCNT /SET UP TO INPUT FOUR CHARACTERS.
2956 6726 1764 TAD C8K240
    
```

```

2957 6727 4774 JMS XC8TYP /PRINT A SPACE.
2958 6730 4772 JMS XC8TTY /GET A CHARACTER.
2959 6731 4763 JMS TSTCHA /CHECK FOR A VALID CHARACTER.
2960 6732 5304 JMP XC8PSW+1 /ERROR RETURN.
2961 6733 3360 DCA C8CHAR
2962 6734 1357 TAD CHRTMP
2963 6735 7106 CLL RTL
2964 6736 7004 RAL /MOVE OVER THREE BITS
2965 6737 1360 TAD C8CHAR
2966 6740 3357 DCA CHRTMP /AND STORE IT.
2967 6741 2765 ISZ TPCNT /DONE YET.
2968 6742 5330 JMP .-12 /NO, GET NEXT VALUE.
2969 6743 4773 JMS XC8CRL
2970 6744 6201 TSTRTN, CDF PRGFLD
2971 6745 1357 TAD CHRTMP /GET NEW SWITCH VALUE
2972 6746 3770 DCA I (20 /AND STORE IT IN PROGRAM FIELD
2973 6747 0000 PSWRTN, 0000
2974 6750 2354 ISZ LFFLAG /WAS A LINE FEED FOUND.
2975 6751 5703 JMP I XC8PSW
2976 6752 6203 CIF CDF PRGFLD /RETURN TO PROGRAM FIELD.
2977 6753 5755 JMP I RSTART /GO RSTART TESTING.
2978 /
2979 6754 0000 LFFLAG, 0
2980 6755 0271 RSTART, STRTST /STARTING TEST OF DIAGNOSTIC.
2981 6756 7774 C8M4, -4
2982 6757 0000 CHRTMP, 0
2983 6760 0000 C8CHAR, 0
2984 /
2985 6763 7022
2986 6764 7065
2987 6765 7066
2988 6766 7215
2989 6767 7075
2990 6770 0020
2991 6771 7076
2992 6772 7135
2993 6773 7146
2994 6774 7126
2995 6775 7011
2996 6776 7200
2997 6777 7072
    PAGE
    7000
2998 /
2999 /CONTROL C FUNCTION.
3000 /ROUTINE CAUSES RETURN TO MONITOR IF ON AN
3001 /OPERATING SYSTEM OR TO THE LOADER IF NOT.
3002 /
3003 7000 1211 CNTRLC, TAD ARROW
3004 7001 4326 JMS XC8TYP /PRINT AN ARROW
3005 7002 1210 TAD C
3006 7003 4326 JMS XC8TYP /AND A "C"
3007 7004 4346 JMS XC8CRL /NOW A <CRLF>
3008 7005 6203 CIF CDF PRGFLD /LOADER MUST BE IN FIELD 0
3009 7006 5607 JMP I K7600
3010 /
    
```

```

3011 /
3012 7007 7600 K7600, 7600
3013 7010 0303 C, "C
3014 7011 0336 ARROW, "
3015 7012 0307 G, "G
3016 /
3017 /CONTROL G FUNCTION
3018 /PRINT "SR=XXXX" WHERE XXXX IS THE CURRENT VALUE OF
3019 /THE PSEUDO-SWITCH REGISTER, IT THE WAITS FOR A NEW SETTING.
3020 /
3021 7013 1211 CNTRLG, TAD ARROW
3022 7014 4326 JMS XC8TYP /PRINT ""
3023 7015 1212 TAD G
3024 7016 4326 JMS XC8TYP /AND A "G
3025 7017 4346 JMS XC8CRL /AND FINALLY A CRLF
3026 7020 4777' XCNTLG, JMS XC8PSW /GO GET NEW SWITCH VALUE.
3027 7021 5776' JMP FLDRTN /EXIT ROUTINE.
3028 /
3029 /THIS ROUTINE WILL VERIFY CHARACTER IS A VALID OCTAL DIGIT
3030 /OR A LINE FEED OR CARRIAGE RETURN.
3031 /
3032 7022 0000 TSTCHA, 0
3033 7023 3274 DCA C8TEMP
3034 7024 1274 TAD C8TEMP
3035 7025 1375 TAD (-203 /TEST FOR CONTROL C
3036 7026 7650 SNA CLA /WAS IT?
3037 7027 5200 JMP CNTRLC /YES.
3038 7030 1274 TAD C8TEMP
3039 7031 4774' JMS TSTCRL /TEST FOR <CRLF>
3040 7032 5251 JMP TSTEXT /FOUND ONE
3041 7033 1274 TAD C8TEMP
3042 7034 4326 JMS XC8TYP
3043 7035 1274 TAD C8TEMP
3044 7036 1272 TAD C8M260
3045 7037 7710 SPA CLA /WAS GREATER THAN 0
3046 7040 5261 JMP ERREXT /NO, TAKE ERROR EXIT.
3047 7041 1274 TAD C8TEMP
3048 7042 1273 TAD C8M270
3049 7043 7700 SMA CLA /WAS 7 OR LESS
3050 7044 5261 JMP ERREXT /NO TAKE ERROR EXIT.
3051 7045 1274 TAD C8TEMP
3052 7046 0271 AND C8K7
3053 7047 2222 ISZ TSTCHA /UPDATE RETURN FOR VALID CHARACTER.
3054 7050 5622 JMP I TSTCHA
3055 7051 4346 TSTEXT, JMS XC8CRL
3056 7052 7307 CLL CLA IAC RTL /+4 TO AC
3057 7053 1266 TAD TPCNT
3058 7054 7640 SZA CLA /CHANGE SWITCH REGISTER.
3059 7055 5773' JMP TSTRTN /GET BACK TO CALL.
3060 7056 1275 TAD C8STOR
3061 7057 3772' DCA CRTMP
3062 7060 5773' JMP TSTRTN
3063 7061 1771' ERREXT, TAD C8K277
3064 7062 4326 JMS XC8TYP
3065 7063 4346 JMS XC8CRL /AND A <CRLF>

```

```

3066 7064 5622 JMP I TSTCHA /PRINT A "?" AND EXIT.
3067 7065 0240 C8K240, 240
3068 7066 0000 TPCNT, 0
3069 7067 7566 C8M212, -212
3070 7070 7775 C8M3, -3
3071 7071 0007 C8K7, 7
3072 7072 7520 C8M260, -260
3073 7073 7510 C8M270, -270
3074 7074 0000 C8TEMP, 0
3075 7075 0000 C8STOR, 0
3076 /
3077 /
3078 /MESSAGE TYPE ROUTINE, MESSAGE TO PRINT IS
3079 /CALL +1
3080 /
3081 7076 0000 XC8PNT, 0
3082 7077 7300 CLL CLA
3083 7100 1676 TAD I XC8PNT /GET POINTER TO MESSAGE.
3084 7101 3275 DCA C8STOR
3085 7102 2276 ISZ XC8PNT /UPDATE RETURN.
3086 7103 1675 C8D01, TAD I C8STOR /GET A SET OF CHARACTERS.
3087 7104 7012 RTR
3088 7105 7012 RTR
3089 7106 7012 RTR
3090 7107 4314 JMS C8DCOD
3091 7110 1675 TAD I C8STOR
3092 7111 4314 JMS C8DCOD
3093 7112 2275 ISZ C8STOR
3094 7113 5303 JMP C8D01
3095 /
3096 /
3097 /
3098 7114 0000 CRDCOD, 0
3099 7115 0356 AND C8K77
3100 7116 7450 SNA /END OF TEXT.
3101 7117 5676 JMP I XC8PNT /YES.
3102 7120 1357 TAD C8M40
3103 7121 7510 SPA
3104 7122 1360 TAD C8K100
3105 7123 1265 TAD C8K240 /ESTABLISHES CHARACTER TO PRINT.
3106 7124 4326 JMS XC8TYP /NOW PRINT IT.
3107 7125 5714 JMP I C8DCOD
3108 /
3109 /
3110 /TYPE ROUTINE.
3111 /
3112 7126 0000 XC8TYP, 0
3113 7127 6046 TLS
3114 7130 6041 TSF
3115 7131 5330 JMP ,.-1
3116 7132 6042 TCF
3117 7133 7200 CLA
3118 7134 5726 JMP I XC8TYP
3119 /
3120 /ROUTINE INPUTS A CHARACTER AND EXITS WITH IT IN THE AC

```

```

3121 /
3122 7135 0000 XCRTTY, 0
3123 7136 6031 KSF
3124 7137 5336 JMP .-1
3125 7140 6036 KRB
3126 7141 0344 AND C8K177
3127 7142 1345 TAD C8K200
3128 7143 5735 JMP I XCRTTY
3129 /
3130 7144 0177 C8K177, 177
3131 7145 0200 C8K200, 200
3132 /
3133 /
3134 7146 0000 XC8CRL, 0
3135 7147 1355 TAD C8K215
3136 7150 4326 JMS XC8TYP
3137 7151 1354 TAD C8K212
3138 7152 4326 JMS XC8TYP
3139 7153 5746 JMP I XC8CRL
3140 /
3141 7154 0212 C8K212, 212
3142 7155 0215 C8K215, 215
3143 /
3144 7156 0077 C8K77, 77
3145 7157 7740 C8M40, -40
3146 7160 0100 C8K100, 100
3147 /
3148 7171 6660
3149 7172 6757
3150 7173 6744
3151 7174 7200
3152 7175 7575
3153 7176 6641
3154 7177 6703
3155 7200 PAGE
3156 7200 0000 /
3157 7201 1777 TSTCRL, 0
3158 7202 7440 TAD C8M212
3159 7203 5207 SZA /IS IT A LINE FEED?
3160 7204 7240 JMP .+4 /NO
3161 7205 3776 DCA LFFLAG
3162 7206 5600 JMP I TSTCRL /EXIT TO PROPER POINT
3163 7207 1775 TAD C8M3
3164 7210 7640 SZA CLA /IS IT A CARRIAGE RETURN?
3165 7211 2200 ISZ TSTCRL /NO, UPDATE RETURN
3166 7212 3776 DCA LFFLAG /CLEAR LINEFEED FLAG
3167 7213 5600 JMP I TSTCRL /EXIT
3168 7214 5600 JMP I TSTCRL /MUST BE A CONTROL CHARACTER,
3169 /
3170 /
3171 /OCTAL TYPE ROUTINE,
3172 /
3173 7215 0000 XC8OCT, 0
3174 7216 3237 DCA C8OCT /STORE CHARACTER TO BE PRINTED.

```

```

3175 7217 1774 TAD C8M4
3176 7220 3773 DCA TMPCNT /SET UP FOR NUMBER OF SHIFTS.
3177 7221 1237 TAD C8OCT /GET VALUE
3178 7222 7004 RAL
3179 7223 7004 RAL
3180 7224 7006 RTL /MOVE INTO BITS 9-11
3181 7225 3237 DCA C8OCT /DON'T CLEAR LINK.
3182 7226 1237 TAD C8OCT
3183 7227 0772 AND C8K7
3184 7230 1240 TAD C8K260
3185 7231 4771 JMS XC8TYP /NOW TYP DIGIT.
3186 7232 1237 TAD C8OCT /GET BACK CHARACTER
3187 7233 2773 ISZ TMPCNT /DONE ALL FOUR YET.
3188 7234 5223 JMP .-11 /NO, GOBACK AND DO NEXT.
3189 7235 7300 CLL CLA
3190 7236 5615 JMP I XC8OCT /EXIT ROUTINE,
3191 /
3192 7237 0000 C8OCT, 0
3193 7240 0260 C8K260, 260
3194 7241 0000 XC8PAS, 0
3195 7242 6201 CDF CONFLD
3196 7243 2253 ISZ PASSES
3197 7244 4770 JMS XC8PNT /PRINT END OF PASS MESSAGE
3198 7245 7267 MESB
3199 7246 1253 TAD PASSES /PRINT NUMBER OF PASSES
3200 7247 4215 JMS XC8OCT
3201 7250 4767 JMS XC8CRL
3202 7251 6203 CIF CDF CALFLD
3203 7252 5641 JMP I XC8PAS /EXIT
3204 /
3205 7253 0000 PASSES, 0
3206 /
3207 7254 7575 TABLA, 100="C
3208 7255 7571 100="G
3209 7256 7557 100="Q
3210 7257 7555 100="S
3211 7260 0000 0000 /END OF TABLE
3212 /
3213 7261 7000 TABLB, CNTRLC
3214 7262 7013 CNTRLG
3215 7263 6673 CNTRLQ
3216 7264 6661 CNTRLS
3217 /
3218 7265 2322 MESA, TEXT "SR="
3219 7266 7500
3219 7267 0516 MESR, TEXT "END OF PASS "
3219 7270 0440
3219 7271 1706
3219 7272 4020
3219 7273 0123
3219 7274 2340
3219 7275 0000
3220 ss
3221 7367 7146
3222 7370 7076

```

3223	7371	7126
3224	7372	7071
3225	7373	7066
3226	7374	6756
3227	7375	7070
3228	7376	6754
3229	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 11111111  
6700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 10011111 11111111  
  
7000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111  
7100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 10000000 01111111  
  
7200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111100  
7300 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000001 11111111  
  
7400  
7500  
  
7600  
7700

A11	0011	C77	0032	DELLOP	2624	K4000	0123
A20	0020	C80100	6655	DEVCOD	0035	K6500	4031
A21	0021	C8CHAR	6760	DONEA	6622	K6520	3743
A22	0022	C8DCOD	7114	EHLTLP	4464	K7	3520
A23	0023	C8D01	7103	END	1502	K7007	3463
ACNSKP	0545	C8K100	7160	ERPEXT	7061	K7600	7007
ACTFLD	6677	C8K177	7144	ERRFLG	0044	K77	3435
AERROR	4512	C8K200	7145	ERRLOP	1510	K770	3462
APT8	4513	C8K212	7154	EXIT	3724	K7700	3536
APTCHK	3716	C8K215	7155	FDATAT	1400	KCC	6032
APTI00	6370	C8K240	7065	FUTLOP	1523	KCC0	0610
APTI01	6371	C8K260	7240	FILCHK	2646	KCC1	0713
APTI02	6372	C8K277	6660	FILERT	1645	KCC2	1036
APTI03	6373	C8K7	7071	FILEXT	2003	KCC3	1063
APTI04	6374	C8K77	7156	FILLER	3200	KCC4	1130
APTI05	6375	C8M212	7067	FILLOP	1667	KCC5	1757
ARROW	7011	C8M260	7072	FLDRTN	6641	KCC6	2000
B1050	5631	C8M270	7073	FPOCK	3464	KCDFIF	6657
B110	5514	C8M3	7070	FOROCT	4474	KCF	6030
B1200	5550	C8M4	6756	G	7012	KCF0	0320
B150	5523	C8M40	7157	GFTPAT	6653	KCF1	0401
B19200	2540	CROCT	7237	GCITA	6643	KCF2	0501
B19K2	5604	CBRTN	6701	GOTOA	6651	KCF3	0551
B2400	5557	C8STOR	7075	GOTOUV	3720	KCF4	1143
B300	5532	C8TEMP	7074	HALT	7402	KIE	6035
B4800	5566	CAF	6007	HLT	4520	KIE0	0403
B56P8	5613	CAFINI	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
BAUNDN	0041	CHEK22	4515	INMODE	6702	KIE4	0536
BAUDST	2146	CHGMES	3120	INTON	2313	KIE5	0616
BAUDTB	2334	CHKFIL	2670	INTRET	0267	KIE6	0710
BAUDTM	2264	CHPLAS	1636	INTXMT	0400	KIE7	0750
BDPNTR	2333	CHRTMP	6757	K0125	0056	KIE8	0760
BGNINT	0200	CLKCNT	0121	K0252	0055	KIE9	1026
BITNO	0037	CLRRRD	0271	K10	3313	KRB	6036
BR1050	2604	CMESS6	4440	K100	3440	KRB0	0635
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	CNTRLC	7000	K2	0025	KRB12	2014
BR300	2430	CNTRLG	7013	K200	0125	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	DATEIT	0040	K37	0031	KRB7	1544
C	7010	DELAY	4461	K377	0034	KRB8	1615
C177	0033	DELAYN	2643	K400	4132	KRB9	1654

KRS	6034	M43	3436	PSWRTH	6747	SPI9	0673
KRS0	0626	MES10A	5145	RAN1	3573	SR4HLT	2133
KRS1	1010	MES10B	5164	RAN2	3574	START	0250
KRS2	2040	MES10C	5203	RANDM1	4771	STENAR	2005
KRS3	2046	MES10D	5222	RANDOM	4506	STERR	2101
KSE	6035	MESA	7265	RECDAT	0045	STFLGS	0704
KSE0	1335	MESSAGE	4471	RECEVE	1436	STLAS	2126
KSE1	2023	MESAGF	3416	RECPNT	5241	STLPPC	4462
KSE2	2045	MESAGX	3400	REDOA	6611	STRST	0271
KSE3	2054	MESB	7267	RFSGEN	4510	SW10NE	4465
KSF	6031	MESS1	4203	RFTINT	2317	TABLA	7254
KSF0	0276	MESS10	5120	RFTURN	3763	TABLB	7261
KSF1	0336	MESS2	4213	RSTART	6755	TABLE1	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	SOTST1	1152	TCF15	2015
KSF21	1675	MESS6C	4545	SOTST2	1200	TCF16	2035
KSF22	1744	MESS6D	4572	SOTST3	1210	TCF2	0470
KSF23	1762	MESS6E	4617	SOTST4	1233	TCF3	0532
KSF24	1341	MESS6F	4645	SOTST5	1251	TCF4	0655
KSF25	2036	MESS6G	4673	SOTST6	1302	TCF5	0716
KSF26	2063	MESS6H	4721	SOTST7	1316	TCF6	1002
KSF3	0471	MESS6I	4747	SETUP	0215	TCF7	1064
KSF4	0537	MESS6J	4776	SLOLAS	1372	TCF8	1123
KSF5	0601	MESS6K	5024	SLOLST	1376	TCF9	1412
KSF6	0721	MESS6L	5052	SLWDAT	1333	TEMP	0124
KSF7	0734	MESS7	5100	SLWREC	1350	TEMSAV	4145
KSF8	2713	MESS7A	5115	SLWTL8	1340	TFL	6040
KSF9	1013	MESTAB	2350	SPACE2	4500	TFL0	0323
KSF8KP	4467	MESTB1	2364	SPACX2	3306	TFL1	0345
LAS	4517	MLOT	4503	SP1	6045	TFL2	0406
LPFLAG	6754	MPC	4200	SP10	0304	TFL3	0452
LISN	4470	NDELAY	0047	SP11	0411	TFL4	0503
LISN1	3340	NERROR	4516	SP110	0737	TFL5	0520
LISN2	3356	NEWDEV	4000	SP111	0751	TFL6	0646
LISN3	3351	NOINTR	0201	SP112	0761	TMSFLD	0000
LISNT1	0170	NOT8E	4050	SP113	1015	TICK	4514
LISNUM	3361	ONAPT	3613	SP114	1027	TL8	6046
LOAD	4460	ONEOCK	3201	SP115	1042	TL80	1102
LOOP	4463	ONEOCT	4472	SP12	0427	TL81	1515
LOOPPC	0046	ORIGIN	6600	SP13	0442	TL810	1424
M0	4041	PASSES	7253	SP14	0455	TL82	1522
M1000	0054	PRGFLD	0000	SP15	0460	TL83	1540
M212	3373	PRNT1	4475	SP16	0506	TL84	1556
M270	3367	PRNT2	4476	SP17	0515	TL85	1606
M3	3374	PRNT4	4477	SP18	0523	TL86	1665

TL87	2307	XAERRO	3671
TL88	2315	XAPT8	3600
TL89	2026	XBSW	3522
TMPCNT	7066	XCRCNT	6600
TPC	6044	XCBCRL	7146
TPC0	0727	XCROCT	7215
TPC1	1073	XC8PAS	7241
TSF	6041	XC8PNT	7076
TSF0	0301	XC8PSW	6703
TSF1	0326	XCRTTY	7135
TSF10	0512	XC8TYP	7126
TSF11	0521	XCHK22	4123
TSF12	0533	XCNILG	7020
TSF13	0664	XCRLF	3244
TSF14	0724	XDELAY	2620
TSF15	2675	XHALT	4102
TSF16	1000	XHLT	4111
TSF17	1003	XLAS	4070
TSF18	1070	XLISN	3314
TSF2	0333	XLOAD	2160
TSF20	1106	XMIOT	3442
TSF22	1124	XMIT	1420
TSF23	1421	XMTDAT	0042
TSF24	1516	XMTDT1	0043
TSF25	1535	XMTIOT	5352
TSF26	1553	XMTREC	1057
TSF28	1672	XNERRO	3744
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	XPAND1	4133
TSF7	0437	XRESGN	3564
TSF8	0453	YSAVGN	3555
TSF9	0504	XSR1LP	1564
TSF8KP	4466	XSR2	4042
TSCHA	7022	XTABLA	6654
TSCHNT	0052	XTABLB	6655
TSCHRL	7200	XTICK	4013
TSTEXT	7051	XTYPE	3235
STRIN	6744	YFSRNO	4505
TWOOCK	3211	YESRNX	3273
TWOOCK	4473		
TYINTR	3000		
TYPE	4501		
UPDATE	1465		
US100	2645		
WATKSF	2707		
WATTSF	2671		

ERRORS DETECTED: 0  
LINKS GENERATED: 81  
RUN-TIME: 8 SECONDS  
3K CORE USED

A11	81#	1303	1310	2255						
A20	85#									
A21	86#									
A22	87#									
A23	88#									
ACNSKP	266	456#	539							
ACTFLD	2929#									
AERROR	174#	879	938	1056	1125	1261	1328	1345	1365	
APT8	176#	197								
APTCHK	2281	2291#								
APTIO0	14#	2261								
APTIO1	17#	2358	2359							
APTIO2	18#	2282								
APTIO3	20#									
APTIO4	21#	2360								
APTIO5	23#									
ARROW	2875	3003	3014#	3021						
B1050	2710	2813#								
B110	2699	2714#								
B1200	2703	2750#								
B150	2700	2723#								
B19200	1439	1591#								
B19K2	2707	2786#								
B2400	2704	2759#								
B300	2701	2732#								
B4800	2705	2768#								
B56P8	2708	2795#								
B600	2702	2741#								
B66P7	2709	2804#								
B9600	2706	2777#								
BAUDNO	105#	230	231	1393						
BAUDST	1289	1295#								
BAUDTB	1394	1431#								
BAUDTM	200	1389#								
BDPNTR	1395	1398	1399	1401	1428#					
BGNINT	197#	2215								
BITNO	103#	213	215	222	225	1396				
BR1050	1442	1637#								
BR110	1431	1471#								
BR1200	1435	1531#								
BR150	1432	1486#								
BR2400	1436	1546#								
BR300	1433	1501#								
BR4800	1437	1561#								
BR568	1440	1607#								
BR600	1434	1516#								
BR667	1441	1622#								
BR9600	1438	1576#								
BSW	64#	2356								
BSWAP	172#	1739								
C	3005	3013#								
C177	99#	2235								
C77	98#									

SEQ 0096















.V7075	2951	2952	2989#				
.V7076	2946	2991#	3197	3222#			
.V7126	2876	2879	2881	2957	2994#	3185	3223#
.V7135	2917	2958	2992#				
.V7146	2882	2945	2969	2993#	3201	3221#	
.V7200	2873	2996#	3039	3151#			
.V7215	2953	2988#					
.V7241	1284	1320#					
.V7400	501	505	2845#				
.V7575	3035	3152#					
.V7600	1689	1731#					
.V7700	1100	1142	1158	1701	1716	2840#	
.V7765	232	332#	1798	1859#			
.V7766	1686	1732#					
.V7774	486	607#	1092	1173#			