IDENTIFICATION

PRODUCT CODE:	MAINDEC-08-DHKMC-B-D
PRODUCT NAME:	PDP-8E EXTENDED MEMORY ADDRESS TEST (EA8E)
DATE: RELEASED:	MAY 1976
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	VERNON FREY D. MACOMBER B. HANSEN

1. ABSTRACT

MODIFIED TO RUN ON APT 8A SYSTEMS, APRIL 1975.
SEE NOTES AT END OF DOCUMENT.

MODIFIED TO RUN WITH A NEW CONSOLE PACKAGE MAY 1976.
SEE SECTION 10 FOR CONSOLE ADDENDUM.

MODIFIED TO RUN WITH NO CONSOLE TERMINAL MAY 1976.
SEE SECTION 11 FOR OPERATING PROCEDURES ON A NON CONSOLE SYSTEM.

THE PDP-8E EXTENDED MEMORY ADDRESS TEST IS DESIGNED TO DETECT ANY LOCATION THAT CANNOT BE UNIQUELY ADDRESSED. THIS IS PERFORMED BY A SERIES OF FOUR TEST ROUTINES WHICH WILL TEST SYSTEMS EQUIPPED WITH FROM 8K TO 32K WORDS OF CORE MEMORY. AUTOMATIC PROGRAM RELOCATION IS PROVIDED IN ORDER TO TEST ALL MEMORY FIELDS FROM EACH MEMORY FIELD. TELETYPE PRINT-OUTS ARE PROVIDED FOR ERROR IDENTIFICATION, AND THE OPERATOR IS GIVEN A DEGREE OF CONTROL OVER THE PROGRAM BY VARIOUS SR SETTINGS.

2. REQUIREMENTS

2.1 EQUIPMENT

A PDP-8E OR A PDP-8A COMPUTER EQUIPPED WITH A MINIMUM OF 8K WORDS OF CORE MEMORY.

2.2 STORAGE

THE PROGRAM OCCUPIES CORE LOCATIONS 0000 TO 4777, WITH LOCATIONS 5000 TO 5177 USED AS A BUFFER AREA.

2.3 PRELIMINARY PROGRAMS

THE BINARY LOADER MUST BE IN MEMORY. ALSO, ALL DIAGNOSTICS FOR A BASIC 4K PDP-8E OR PDP-8A MUST HAVE BEEN PREVIOUSLY RUN SUCCESSFULLY.

3. LOADING PROCEDURE

LOAD THE PROGRAM WITH THE BINARY LOADER (BIN). THE PROGRAM MAY BE LOADED INTO ANY DESIRED CORE STACK BY HAVING BIN IN THAT CORE STACK.

4. OPERATING PROCEDURE

4.1 PROGRAM AND OPERATOR ACTION

A. SET THE SR TO THE INSTRUCTION FIELD AND DATA FIELD OF THE STACK WHICH CONTAINS THE PROGRAM.

B. PRESS KEY EXTD ADDR LOAD.

C. SET THE SR FOR DESIRED STARTING ADDRESS ACCORDING TO THE FOLLOWING TABLE.

ADDRESS	TEST EXECUTION
0200	RUN ALL TESTS
0201	RUN ONLY TEST 1
0202	RUN ONLY TEST 2
0203	RUN ONLY TEST 3
0204	RUN ONLY TEST 4

D. PRESS KEYS ADDR LOAD, CLEAR, AND CONT. A SETUP SR MESSAGE WILL BE PRINTED.

E. SET THE SR FOR DESIRED OPERATION ACCORDING TO THE FOLLOWING TABLE.

SWITCH	0 (DOWN)	1 (UP)
SR00	CONTINUE AFTER ERROR	HALT AFTER ERROR
SR01	TYPEOUT ERRORS	INHIBIT ERROR TYPEOUTS
SR02	NORMAL	TTY BELL ON ERROR
SR03	RELOCATE PROGRAM	INHIBIT PROGRAM RELOCATION
SR04	NORMAL	CHANGE STACK LIMITS
SR05	NORMAL	HALT AFTER CURRENT TEST
SR06-08	STARTING STACK LIMIT (0-7)	
SR09-11	ENDING STACK LIMIT (0-7)	

F. PRESS KEY CONT.

4.2 DETAILED SR EXPLANATION

SR00-02 SR02, WILL RING THE TTY BELL ONCE FOR EACH ERROR.
SR00 AND SR01 HAVE NO EFFECT WITH SR02 SET.
SR03 SR03 MAY BE SET OR RESET AT ANY TIME AND THE PROGRAM
WILL ACT ACCORDINGLY
SR04 SR04 ALLOWS THE OPERATOR TO CHANGE THE STACK LIMITS AS
DEFINED BY SR06-11.
SR05 SR05 IS NORMAL HALT FOR PROGRAM
SR06-08 THESE SWITCHES DEFINE THE STARTING STACK LIMIT
(NORMALLY 0).
SR09-11 THESE SWITCHES DEFINE THE ENDING STACK LIMIT
(NORMALLY 7)

4.3 EXAMPLE OF SELECTING STACKS FOR TEST

EXAMPLE 1: SR = 0007, 28K SYSTEM

STACKS SELECTED FOR TESTING ARE 6,5,4,3,2,1,0

EXAMPLE 2: SR = 0004, 28K SYSTEM

STACKS SELECTED FOR TESTING ARE 4,3,2,1,0

EXAMPLE 3: SR = 0022 28K SYSTEM

STACKS SELECTED FOR TESTING ARE 2
(NO RELOCATION WILL OCCUR)

EXAMPLE 4: SR = 0041 28K SYSTEM

STACKS SELECTED FOR TESTING ARE 6,5,4,1,0

NOTE 1: STACKS NOT IN THE SYSTEM ARE AUTOMATICALLY DE-SELECTED
AS IS EXAMPLE 1. STACK 7 IS NOT PRESENT THEREFORE NOT
SELECTED.

NOTE 2: A SINGLE STACK CAN BE SELECTED FOR TESTING PROVIDING
THE PROGRAM IS NOT IN THAT STACK AS IN EXAMPLE 3.

NOTE 3: ANY STACK OR GROUP OF STACKS CAN BE BY-PASSED AS IN
EXAMPLE 4. STACKS 2 AND 3 ARE NOT SELECTED, STACK 7
IS NOT PRESENT.

5. ERRORS

THE CONTENTS OF A GIVEN MEMORY TEST LOCATION SHOULD ALWAYS BE EQUAL TO ITS ADDRESS OR THE COMPLEMENT OF ITS ADDRESS. IF IT IS NOT, A TEST ERROR WILL RESULT. A RELOCATION ERROR WILL OCCUR IF THE RELOCATION COMPARISON CHECK FAILS.

5.1 TEST ERROR TYPEOUTS

FOR THE FIRST ERROR ENCOUNTERED A HEADER WILL BE TYPED OUT FOLLOWED BY THE PERTINENT DATA. FOR ALL SUBSEQUENT ERRORS, ONLY THE PERTINENT DATA WILL BE TYPED. THE FORMAT IS AS FOLLOWS:

```
PR  LOC  ADDR   GOOD  BAD   TEST
```

PR LOC = THE PROGRAM ADDRESS WHERE THE ERROR JMS OCCURRED, (INCLUDES FIELD)
ADDR = THE ADDRESS OF THE LOCATION IN ERROR. (INCLUDES FIELD)

GOOD = WHAT THE DATA SHOULD BE.

BAD = WHAT THE DATA IS.

TEST = THE TEST (1-4) RUNNING WHEN THE FAILURE OCCURRED.

5.2 RELOCATION ERROR TYPEOUTS

ALL RELOCATION ERRORS ARE IN THE FOLLOWING FORMAT:

XXXXX RELOCATION ERROR AT LOCATION YYYYY

XXXXX = THE PROGRAM ADDRESS WHERE THE ERROR JMS OCCURRED, (INCLUDES FIELD)

YYYYY = THE ADDRESS OF THE LOCATION IN ERROR (INCLUDES FIELD)

NOTE: AFTER EACH ERROR PRINT-OUT THE PROGRAM CONTINUES ON WITH THE NEXT SEQUENTIAL MEMORY LOCATION.

6. RESTRICTIONS

6.1 STARTING RESTRICTIONS

THE PROGRAM MAY BE RESTARTED AT ANY TIME FROM LOCAION 0200 OF THE STACK THE PROGRAM IS PRESENTLY IN.

6.2 OPERATING RESTRICTIONS

NONE

7. EXECUTION TIME

THE TIME TO RUN ALL 4 TESTS IN ONE CORE STACK IS APPROXIMATELY 1/2 SECOND. DURING PROGRAM EXECUTION A 5 WILL BE TYPED ON THE TTY APPROXIMATELY EVERY 5 MINUTES OF PROGRAM RUN TIME. THIS ALLOWS THE OPERATOR TO DETERMINE APPROXIMATE RUN TIME BEFORE A FAILURE OCCURRED.

8. SCOPE LOOPS

TWO SPECIAL SCOPE LOOPS HAVE BEEN PROVIDED IN THIS PROGRAM.

8.1 SCOPE LOOP 1

THIS SCOPE LOOP WRITES THE VALUE EQUAL TO THE ADDRESS SPECIFIED BY THE SR INTO THE ADDRESS SPECIFIED BY THE SR. IT THEN LOOPS DOING A WRITE-READ. THE ADDRESS BEING LOOPED ON CAN BE CHANGED SIMPLY BY CHANGING THE SWITCH SETTING.

A. SET THE SR TO THE INSTRUCTION FIELD THAT THE PROGRAM IS IN AND THE DATA FIELD WANTED TO TEST.

B. PRESS KEY EXTD ADDR LOAD.

C. SET THE SR EQUAL TO 3200.

D. PRESS KEY ADDR LOAD.

E. SET THE SR EQUAL TO THE ADDRESS TO TEST.

F. PRESS KEYS CLEAR, AND CONT.

8.2 SCOPE LOOP 2

THIS SCOPE LOOP IS THE SAME AS SCOPE LOOP 1 EXCEPT THAT A GROUP OF ADDRESSES MAY BE SPECIFIED. THE STARTING ADDRESS SPECIFIED MUST BE LESS THAN THE ENDING ADDRESS SPECIFIED.

A. SET THE SR TO THE INSTRUCTION FIELD THAT THE PROGRAM IS IN AND THE DATA FIELD WANTED TO TEST.

B. PRESS KEY EXTD ADDR LOAD.

C. SET THE SR EQUAL TO 3207.

D. PRESS KEY ADDR LOAD.

E. SET THE SR EQUAL TO THE FIRST ADDRESS OF THE GROUP.

F. PRESS KEYS CLEAR AND CONT. A HALT WILL OCCUR AT ADDRESS 3211.
G. SET THE SR EQUAL TO THE LAST ADDRESS OF THE GROUP.

H. PRESS KEY CONT.

NOTE 1: THE ADDRESS(S) SPECIFIED WILL BE LOOPED UNTIL STOPPED
BY THE OPERATOR WITH KEY HALT. NO ERROR CHECKING IS
DONE. TO RESUME NORMAL OPERATION, RESTART PROGRAM
AT ADDRESS 0200-0204 OF THE CURRENT INSTRUCTION
FIELD.

9. PROGRAM DESCRIPTION

9.1 GENERAL

THE PDP-8E EXTENDED MEMORY ADDRESS TEST IS INTENDED FOR USE
WITH A PDP-8E EQUIPPED WITH THE EXTENDED MEMORY OPTION.
A TOTAL OF FOUR TEST ARE EXECUTED BY THE PROGRAM. (SEE 9.2 THRU
9.5). EACH TEST WRITES A UNIQUE PATTERN INTO CORE MEMORY AND
THE CHECKS FOR ERROR. THE PATTERNS WERE CHOSEN TO AID THE
OPERATOR IN THE EVENT OF ADDRESSING ERRORS.

THE PROGRAM AUTOMATICALLY RELOCATES ITSELF TO EACH MEMORY
FIELD UNDER TEST TO ENSURE THAT ALL FIELDS MAY BE CORRECTLY
REFERENCED FROM ANY FIELD. FIELDS NOT PRESENT IN THE SYSTEM WILL
AUTOMATICALLY BE DE-SELECTED FROM TESTING. (SEE 9.6)

CONTROL OF THE PROGRAM IS GIVEN TO THE OPERATOR BY MEANS OF THE
SR. THE OPERATOR MAY HALT AFTER ERROR, INHIBIT ERROR PRINTOUTS,
SUBSTITUTE TTY BELL FOR ERROR INDICATION, HALT AFTER TEST,
CHANGE FIELD TEST LIMITS, SELECT ALL OR ANY ONE OF FOUR TESTS,
INHIBIT PROGRAM RELOCATION, AND AT ANY TIME RESTART THE
PROGRAM AT LOCATION 0200 THRU 0204.

9.2 TEST 1

TEST 1 WRITES THE VALUE OF EACH LOCATION INTO ITSELF IN THE
FOWARD DIRECTION. THEN EACH LOCATION IS READ AND CHECKED IN
THE FORWARD DIRECTION.

9,3 TEST 2

TEST 2 WRITES THE COMPLEMENT VALUE OF EACH LOCATION INTO ITSELF IN THE FORWARD DIRECTION. THEN EACH LOCATION IS READ AND CHECKED IN THE FORWARD DIRECTION.

9,4 TEST 3

TEST 3 WRITES THE VALUE OF EACH LOCATION INTO ITSELF IN THE REVERSE DIRECTION. THEN EACH LOCATION IS READ AND CHECKED IN THE REVERSE DIRECTION.

9,5 TEST 4

TEST 4 WRITES THE COMPLEMENT VALUE OF EACH LOCATION INTO ITSELF IN THE REVERSE DIRECTION. THEN EACH LOCATION IS READ AND CHECKED IN THE REVERSE DIRECTION.

9,6 PROGRAM RELOCATION

PROGRAM RELOCATION IS GOVERNED BY THE STATUS OF SR BIT 3 OR BY THE FACT THAT ONLY ONE STACK IS SELECTED FOR TESTING. WITH SR BIT 3 DOWN (0 POSITION) PROGRAM RELOCATION OCCURS EACH TIME THE TEST PATTERN AND ITS COMPLEMENT HAVE BEEN COMPLETELY TESTED IN EACH SELECTED STACK. THE PROGRAM FIRST RELOCATES TO THE HIGHEST ORDER 4K STACK UNDER TEST. THE PROGRAM KEEPS RELOCATING TO THE NEXT LOWER STACK UNDER TEST UNTIL IT REACHES THE LOWEST ORDER STACK UNDER TEST. THE TESTING AND RELOCATION CYCLE IS THEN REPEATED. THE CONTENTS OF THE ENTIRE STACK ARE RELOCATED WHICH ENABLES ANY OTHER INFORMATION (RIM-BIN) TO BE CARRIED WITH THE PROGRAM.

THE PROGRAM PROVIDES A DEGREE OF PROTECTION FOR ITSELF BY REMEMBERING ALL STACKS WHERE ERRORS OCCUR. WHEN A FAULTY STACK IS NEXT IN SEQUENCE TO CONTAIN THE PROGRAM, THE PROGRM WILL SKIP THE FAULTY STACK AND RELOCATE TO THE FIRST LOWER ORDER STACK WHICH IS ERROR FREE. IF ALL OTHER SELECTED STACKS ARE FAULTY, PROGRAM RELOCATION WILL NOT TAKE PLACE.

DURING RELOCATION A COMPARISON CHECK IS MADE TO INSURE NO PROGRAM LOSS

FOR FURTHER UNDERSTANDING OF HOW THE TESTS ARE PERFORMED, REFER TO THE LISTING.

10.0 CONSOLE PACKAGE ADDENDUM

10.1 DESCRIPTION

A CONSOLE PACKAGE HAS BEEN ADDED TO THIS DIAGNOSTIC TO ALLOW THE PROGRAM TO RUN WITH NO HARDWARE SWITCH REGISTER AND TO HAVE COMMUNICATIONS WITH THE DIAGNOSTIC VIA A TERMINAL. THE DIAGNOSTIC CAN BE RUN IN TWO MODES WITH THE CONSOLE PACKAGE. 1). RUNNING WITH THE CONSOLE PACKAGE ACTIVE-THIS ALLOWS THE OPERATOR CONTROL OF THE DIAGNOSTIC THROUGH THE TERMINAL. THE DIAGNOSTIC WILL ASK FOR THE VALUE OF THE PSEUDO SWITCH REGISTER BEFORE CONTINUING WITH THE EXECUTION OF THE DIAGNOSTIC. ALL ERRORS AND PASS COMPLETES WILL BE PRINTED AT THE TERMINAL. NO HALTS WILL BE EXECUTED. 2). CONSOLE PACKAGE NOT ACTIVE- THIS WILL RESULT IN THE NORMAL STANDALONE OPERATION OF THE PROGRAM AS DESCRIBED IN SECTIONS 1 THROUGH 9 OF THIS DOCUMENT.

10.2 RESTRICTIONS

A. RUNNING THE CONSOLE PACKAGE REQUIRES THAT THE PSEUDO SWITCH REGISTER BE USED. HOWEVER, IF THE PROGRAM IS INITIALIZED TO USE THE HARDWARE SWITCH REGISTER THEN THE PROGRAM WILL USE THE HARDWARE SWITCH REGISTER REGARDLESS OF ANY CHANGES MADE USING THE CONSOLE TERMINAL.

B. THE TWO SCOPE LOOPS, SECTIONS 8.1 AND 8.2 OF THIS DOCUMENT, CAN NOT BE EXECUTED USING THE CONSOLE PACKAGE. TO RUN THE SCOPE LOOP SECTIONS OF THE PROGRAM, THE HARDWARE SWITCH REGISTER SHOULD BE USED.

10.3 INITIALIZATION

THE PROGRAM WHEN LOADED IS INITIALIZED TO USE THE HARDWARE SWITCH REGISTER AND TO RUN WITH THE CONSOLE PACKAGE NON ACTIVE. THIS WAS THE NORMAL STANDALONE CONFIGURATION OF THE PROGRAM. TO MAKE THE CONSOLE PACKAGE ACTIVE AND TO USE THE PSEUDO SWITCH REGISTER DO THE FOLLOWING:

A. SET LOCATION 0021 BIT 0=0 TO INDICATE TO THE PROGRAM TO USE THE PSEUDO SWITCH REGISTER (LOCATION 0020)

B. SET LOCATION 0022 BIT 3=1 TO INDICATE TO THE PROGRAM THAT THE CONSOLE PACKAGE IS ACTIVE.

10.4 STARTING THE PROGRAM

A. LOAD THE INSTRUCTION AND DATA FIELD TO THE FIELD THAT CONTAINS THE PROGRAM

B. LOAD ADDRESS TO 0200 AND PRESS "CLEAR", THEN "RUN"

C. THE PROGRAM WILL TYPEOUT A HEADING AND THEN PRINT SR=XXXX. XXXX IS THE VALUE OF THE SWITCH REGISTER USED. SELECT A VALUE TO USE FOR THE SWITCH REGISTER FROM THE SWITCH REGISTER TABLE IN PARAGRAPH 4.1 OF THIS DOCUMENT AND INPUT THE NUMBER USING THE CONSOLE KEYBOARD. TYPING IN FOUR DIGITS WILL CAUSE THE PROGRAM TO ECHO THE NEW VALUE OF THE SWITCH REGISTER AND START THE PROGRAM. TYPING A CARRIAGE RETURN AFTER INPUTTING A DIGIT WILL ALSO CAUSE THE PROGRAM TO ECHO THE NEW VALUE OF THE SWITCH REGISTER AND START THE PROGRAM. IF IT IS NOT DESIRED TO CHANGE THE SWITCH REGISTER VALUE, A CARRIAGE RETURN CAN BE TYPED AND THE PROGRAM WILL BE STARTED WITHOUT ECHOING THE SWITCH REGISTER VALUE.

   REFER TO PARAGRAPH 4.2 OF THIS DOCUMENT FOR A DETAILED EXPLANATION OF THE SWITCH REGISTER SETTINGS.

10.5 CONTROL CHARACTERS

10.5.1 CONTROL G

TO GAIN CONTROL OF THE CONSOLE SWITCH REGISTER PACKAGE WHILE RUNNING THE PROGRAM, A CONTROL G MUST BE TYPED IN ON THE CONSOLE TERMINAL. WHEN CONTROL G IS TYPED THE PROGRAM WILL RESPOND BY TYPING A UP ARROW THEN G FOLLOWED BY SR=XXXX. XXXX IS THE PRESENT CONTENTS OF THE PSEUDO SWITCH REGISTER. THE OPERATOR CAN NOW CHANGE THE SETTING OF THE SWITCH REGISTER BY TYPING IN A NEW NUMBER OR NOT CHANGE IT BY TYPING IN A CARRIAGE RETURN. TYPING IN FOUR DIGITS WILL CAUSE THE PROGRAM TO ECHO THE SWITCH REGISTER VALUE TYPED AND CONTINUE THE PROGRAM USING THE VALUE TYPED AS THE NEW SWITCH REGISTER. TYPING IN ONE TO THREE DIGITS FOLLOWED BY A CARRIAGE RETURN WILL ALSO CAUSE THE PROGRAM TO ECHO THE NEW VALUE OF THE SWITCH REGISTER TYPED AND CONTINUE THE PROGRAM. TYPING ONLY A CARRIAGE RETURN WILL CAUSE THE PROGRAM TO CONTINUE WITHOUT CHANGING THE SWITCH REGISTER, AND THE PROGRAM WILL NOT ECHO THE SWITCH REGISTER VALUE. BY TYPING A LINE FEED, THE PROGRAM WILL BE RESTARTED AT ADDRESS 0200. TYPING A CONTROL C WILL CAUSE THE PROGRAM TO RETURN TO THE OPERATING SYSTEM (OS/8) AT 7600 IN FIELD 0. ILLEGAL CHARACTERS TYPED WILL RESULT IN THE CHARACTER BEING ECHOED FOLLOWED BY A QUESTION MARK AND THE SWITCH REGISTER MESSAGE BEING RETYPED.

10.5.2 CONTROL S

THIS IS A CONTROL CHARACTER TO STOP SENDING DATA TO A TERMINAL. IT IS USUALLY A FUNCTION OF THE TERMINAL AND IS AUTOMATICALLY SENT WHEN THE TERMINAL BUFFER IS FULL. THE BUFFER IS EMPTIED AS THE DATA IS PRINTED. AFTER ALL THE DATA IN THE TERMINAL BUFFER IS PRINTED THE TERMINAL SENDS A CONTROL Q (START SENDING DATA TO TERMINAL). THE CONTROL S IS NOT ECHOED.

BY TYPING A CONTROL S WHILE RUNNING THE DIAGNOSTIC, THE PROGRAM, WHEN THE CONTROL S IS RECOGNIZED, WILL WAIT FOR A CONTROL Q TO CONTINUE THE PROGRAM, A LINE FEED TO RESTART THE PROGRAM, OR A CONTROL C TO RETURN TO THE OPERATING SYSTEM.

10.5.3 CONTROL Q

THIS CONTROL CHARACTER IS TO RESUME SENDING DATA TO THE TERMINAL. IT IS USUALLY SENT AUTOMATICALLY BY A TERMINAL WHEN IT IS READY TO ACCEPT MORE DATA. THIS CONTROL CHARACTER IS NOT ECHOED.

BY TYPING A CONTROL Q AFTER A CONTROL S HAS BEEN TYPED, THE PROGRAM WILL CONTINUE FROM WHERE IT WAS INTERRUPTED FROM.

10.5.4 CONTROL C

THIS CONTROL CHARACTER IS USED TO RETURN CONTROL BACK TO AN OPERATING SYSTEM KEYBOARD MONITOR. THE OPERATING SYSTEM SELECTED FOR THE PDP-8 IS THE OS/8 SYSTEM WITH ITS BOOTSTRAP LOCATED IN THE LAST PAGES OF FIELD 0 AND 1. WHEN THIS CONTROL CHARACTER IS RECOGNIZED, THE PROGRAM WILL ECHO THE CONTROL CHARACTER AS AN UP ARROW THEN C, RESTORE THE LAST PAGE OF FIELD 0 AND 1, AND JUMP TO LOCATION 7600 IN FIELD 0.

10.5.5 ILLEGAL CHARACTERS

A CHARACTER TYPED ON THE KEYBOARD, OTHER THAN A CONTROL G, CONTROL S, OR A CONTROL C, WILL RESULT IN THE CHARACTER BEING ECHOED FOLLOWED BY A QUESTION MARK AND THE PROGRAM WILL BE CONTINUED.

10.6 END OF PASS REPORTING

THESE WILL BE AN END OF PASS MESSAGE CONTAINING THE DIAGNOSTIC NAME, THE WORD PASS AND AN OCTAL NUMBER OF PASSES. THE PROGRAM WILL ALSO TYPE THE ORIGINAL END OF PASS MESSAGE ALONG WITH THE ABOVE MESSAGE.

10.7 ERRORS

THE STANDARD ERROR REPORTS AS DESCRIBED IN SECTION 5 OF THIS DOCUMENT WILL BE USED.

10.8 SWITCH REGISTER SETTINGS

THE STANDARD SWITCH SETTINGS AS DESCRIBED IN SECTION 4 OF THIS DOCUMENT WILL BE USED. THE ONLY DIFFERENCE BEING THAT INSTEAD OF HALTING THE PROGRAM ON A SWITCH SETTING, THE PROGRAM WILL ENTER THE CONSOLE SWITCH REGISTER PACKAGE AND WAIT FOR OPERATOR RESPONSE.

10.9 FILLER CHARACTERS

IF THE CONSOLE TERMINAL REQUIRES FILLER CHARACTERS, DEPOSIT THE NUMBER OF FILLER CHARACTERS INTO LOCATION 4400 OF THE FIELD THAT THE PROGRAM IS LOCATED IN.

11.0 NON CONSOLE TERMINAL SYSTEM ADDENDUM

11.1 DESCRIPTION

THE PROGRAM HAS BEEN MODIFIED TO RUN ON THOSE SYSTEMS WITHOUT A CONSOLE TERMINAL. THIS IS DONE BY SETTING A LOCATION IN THE PROGRAM TO NON ZERO, ALL ERRORS AND FIELD LIMIT CHANGES WILL RESULT IN A HALT OR HALTS INSTEAD OF TYPEOUTS ON THE CONSOLE TERMINAL.

11.2 RESTRICTIONS

A. FIELD LIMITS MUST BE SET AT PROGRAM START, OTHERWISE, THE PROGRAM WILL HALT TO ALLOW THE OPERATOR TO SET THE FIELD LIMITS IN THE SWITCH REGISTER.

B. TO RUN THIS PROGRAM, A MINIMUM OF 8K OF MEMORY IS REQUIRED.

C. MEMORIES TO BE TESTED MUST BE IN SEQUENTIAL ORDER STARTING AT FIELD 0.

11.3 INITIALIZATION

A. SET LOCATION 0024 IN THE PROGRAM FIELD TO A NUMBER OTHER THAN 0000. SETTING THIS LOCATION TO NON-ZERO SIGNIFIES TO THE PROGRAM THAT A CONSOLE TERMINAL IS NOT AVAILABLE.

B. THE PROGRAM WHEN LOADED IS INITIALIZED TO USE THE HARDWARE SWITCH REGISTER. IF NO HARDWARE SWITCH REGISTER IS AVAILABLE, DO THE FOLLOWING TO SELECT THE SOFTWARE PSEUDO SWITCH REGISTER (LOCATION 0020).

1. SET BIT 0 EQUAL TO A 0 IN LOCATION 0021 TO INDICATE TO THE PROGRAM THAT LOCATION 0020 WILL BE USED AS THE PSEUDO SWITCH REGISTER. THE PROGRAM WHEN STARTED WILL THEN SET THE PSEUDO SWITCH REGISTER TO THE FIELD LIMITS FOR A NORMAL SYSTEM STARTUP. THE PSEUDO SWITCH REGISTER WILL EQUAL XX07 WHERE Xx EQUALS SWITCH REGISTER BITS PREVIOUSLY SET IN THE PSEUDO SWITCH REGISTER, 0 EQUALS STARTING FIELD LIMIT AND 7 EQUALS ENDING FIELD LIMIT.

   IF IT IS DESIRED TO INITITIALIZE THE FIELD LIMITS TO OTHER THAN THE ABOVE DO THE NEXT STEP.

2. SET LOCATION 0021 TO 00XX WHERE XX IS THE MEMORY SIZE. XX=07=8K; XX=13=12K; XX=17=16K; XX=37=32K. THE PROGRAM WHEN STARTED WILL THEN ADJUST THE PSEUDO SWITCH REGISTER TO THE APPROPRIATE FIELD LIMITS SELECTED IN LOCATION 0021.

11.4 OPERATING PROCEDURES

TO START THE PROGRAM:

A. SET THE IF AND DF TO THE FIELD THAT CONTAINS THE PROGRAM.

B. LOAD ADDRESS TO 0200

C. IF THE PROGRAM WAS INITIALIZED TO USE THE HARDWARE SWITCH REGISTER, SET THE SWITCH REGISTER TO 0007.

D. PRESS "INIT" AND THEN "RUN".

E. THE PROGRAM WILL NOW RUN UNTIL AN ERROR IS ENCOUNTERED OR A SWITCH REGISTER OPTION IS SELECTED TO CAUSE THE PROGRAM TO HALT. REFER TO THE LISTING FOR ALL HALTS.

F. SETTING THE SWITCH REGISTER TO 01XX (XX=FIELD LIMITS), WILL CAUSE THE PROGRAM TO HALT AFTER CURRENT TEST. REFER TO LISTING FOR ADDRESS OF HALT.

G. SETTING THE SWITCH REGISTER TO 02XX (XX=FIELD LIMITS), WILL CAUSE THE PROGRAM TO HALT FOR FIELD LIMIT CHANGES VIA THE SWITCH REGISTER. REFER TO LISTING FOR ADDRESS OF HALT.

11.5 SWITCH REGISTER SETTINGS

SR0=1	HALT AFTER ERROR
SR1=1	INHIBIT ERROR HALTS EXCEPT HALT AFTER ERROR SWITCH
SR2=1	INHIBIT OPERATION OF SR0 AND SR1
SR3=1	INHIBIT PROGRAM RELOCATION
SR4=1	HALT PROGRAM FOR FIELD LIMIT CHANGES VIA SR 6-11.
SR5=1	HALT AFTER CURRENT TEST
SR6-8	STARTING FIELD LIMIT (0-7)-NORMALLY=0
SR9-11	ENDING FIELD LIMIT (0-7)-NORMALLY=7

11.6 ERRORS

ALL ERRORS ENCOUNTERED WILL RESULT IN AN ERROR HALT WITH ERROR INFORMATION IN THE AC. REFER TO THE LISTING FOR THE TYPE OF ERROR HALT AND GO TO THE APPROPRIATE PARAGRAPH BELOW. A TEST ERROR WILL RESULT IF THE CONTENTS OF A GIVEN MEMORY TEST LOCATION IS NOT EQUAL TO ITS ADDRESS OR THE COMPLEMENT OF ITS ADDRESS. A RELOCATION ERROR WILL OCCUR IF THE RELOCATION COMPARISON CHECK FAILS.

11.6.1 TEST ERROR HALTS

FOR ERRORS ENCOUNTERED TESTING MEMORY ADDRESSES, THE PROGRAM WILL HALT WITH PERTINENT INFORMATION IN THE AC. REFER TO THE STEPS BELOW FOR THE TEST ERROR INFORMATION.

A. THE PROGRAM WILL HALT AT ADDRESS 3473 WITH THE CONTENTS OF AC BITS 9-11 EQUAL TO THE PROGRAM FIELD.

B. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3476 WITH THE CONTENTS OF THE AC EQUAL TO THE PROGRAM ADDRESS WHERE THE ERROR JMS OCCURED.

C. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3502 WITH THE CONTENTS OF AC BITS 9-11 EQUAL TO THE FIELD CONTAINING THE ERROR.

D. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3505 WITH THE CONTENTS OF THE AC EQUAL TO THE ADDRESS OF LOCATION IN ERROR.

E. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3510 WITH THE CONTENTS OF THE AC EQUAL TO WHAT THE DATA SHOULD BE.

F. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3513 WITH THE CONTENTS OF THE AC EQUAL TO WHAT THE DATA WAS.

G. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3522 WITH THE CONTENTS OF THE AC EQUAL TO A TEST NUMBER (1-4) OF THE TEST RUNNING WHEN THE FAILURE OCCURED.

H. PRESS "CONT" TO CONTINUE THE PROGRAM ON TO THE NEXT SEQUENTIAL TEST MEMORY ADDRESS.

I. TEST ERROR HALTS MAY BE INHIBITED FROM HALTING BY SETTING SR1 TO A 1.

11.6.2 RELOCATION ERROR HALTS

ALL RELOCATION ERRORS WILL RESULT IN A HALT WITH PERTINENT INFORMATION IN THE AC. REFER TO THE STEPS BELOW FOR THE ERROR INFORMATION.

A. THE PROGRAM WILL HALT AT ADDRESS 3527 WITH THE CONTENTS OF AC BITS 9-11 EQUAL TO THE FIELD THE PROGRAM IS LOCATED IN.

B. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3532 WITH THE CONTENTS OF THE AC EQUAL TO THE ADDRESS WHERE THE ERROR JMS OCCURED.

C. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3536 WITH THE CONTENTS OF AC BITS 9-11 EQUAL TO THE FIELD THE PROGRAM IS RELOCATING TO.

D. PRESS "CONT". THE PROGRAM WILL HALT AT ADDRESS 3541 WITH THE CONTENTS OF THE AC EQUAL TO THE ADDRESS OF LOCATION IN ERROR.

E. PRESSING "CONT" AGAIN WILL RESULT IN THE PROGRAM CONTINUING WITH THE NEXT SEQUENTIAL MEMORY LOCATION. CONTINUING IS NOT ADVISABLE BECAUSE THE PROGRAM MAY HAVE BEEN CHANGED DURING RELOCATION.

11.6.3 HALT AFTER ERROR

THE PROGRAM WILL ONLY HALT HERE IF SWITCH REGISTER 0 EQUALS A ONE AND AN ERROR OCCURED. IF SWITCH REGISTER TWO EQUALS A ONE, NO HALT WILL OCCUR.

A. THE PROGRAM WILL HALT AT ADDRESS 2262 WITH THE CONTENTS OF THE AC EQUAL TO THE PROGRAM LOCATION WHERE THE ERROR JMS LOCATION.

B. PRESSING "CONT" WILL CAUSE THE PROGRAM TO CONTINUE FROM THE POINT WHERE THE ERROR WAS DETECTED.

APT NOTES

ALL OF THE FOLLOWING NOTES APPLY ONLY WHEN THE PROGRAM IS BEING RUN ON AN APT SYSTEM.

1. FOR MORE INFORMATION SEE THE FOLLOWING DOCUMENTS.

   A. STANDARD APT SYSTEM TO PDP8 DIAGNOSTIC INTERFACE.

   B. APT SYSTEM MANAGERS GUIDE.

2. FOR ANY DIFFERENCES BETWEEN THESE NOTES AND THE REST OF THE DOCUMENT, THESE NOTES WILL PREVAIL. (SEE THE LISTING ALSO.)

3. ALL CODE THAT HAS BEEN ADDED (INSERTED) FOR APT WILL CONTAIN THE EXPRESSION: /APT/

   ANY ORIGINAL CODE NEGATED FOR APT HAS BEEN 'REMOVED' BY PRECEDING IT WITH THE EXPRESSION: /*APT*/

4. IF BIT 0 OF HCW1 IS A '1' THEN THE HARDWARE SWITCH REGISTER WILL BE USED, REGARDLESS OF LOAD METHOD (SCRIPT OR DUMP).

THE FOLLOWING NOTES APPLY ONLY WHEN THE LOAD METHOD WAS SCRIPT LOAD.

5. SWITCHES:

   SR3, SR6-8 & SR9-11 ARE USED ASED DESCRIBED IN THE DOCUMENT.

   SR6-8 & SR9-11 MAY BE USED TO SPECIFY FIELD LIMITS INSTEAD OF HCW1. (WILL BE USED IF HCW1 BITS 7-11 = 0.)

   FIELD 7 CANNOT BE SPECIFED.

6. ERRORS:

   ALL ERRORS CALL APT.

LISTING

```
                        /KM8E EXTENDED MEMORY ADDRESS TEST  MAINDEC-08-DHKMC-B-L
                        /EXTENDED ADDRESS TEST FOR KM8-E EXTENDED MEMORY (VER B)
                        /COPYRIGHT 1971, 1975,  1976, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        /PROGRAMMER, VERNON FREY

                        /       MODIFIED FOR APT        APRIL 1975      D. MACOMBER
                        /       MODIFIED TO RUN WITH NO CONSOLE - APRIL 1976 - B. HANSEN
                        /       MODIFIED FOR A NEW CONSOLE PACKAGE - APRIL 1976 - B. HANSEN

                        /
                        /
                        /SW0=1          HALT AFTER ERROR
                        /SW1=1          INHIBIT ERROR TYPEOUT
                        /SW2=1          BELL ON ERROR (USEFUL FOR MAINTENENCE)
                        /SW3=1          INHIBIT PROGRAM RELOCATION
                        /SW4=1          CHANGE STACK LIMITS
                        /SW5=1          HALT AFTER CURRENT TEST
                        /SW6-SW8        STARTING STACK LIMIT (0-7)
                        /SW9-SW11       ENDING STACK LIMIT (0-7)
                        /
                        /
                        /
                        /PROGRAM STARTING ADDRESS
                        /0200           RUN ALL TESTS
                        /0201           RUN ONLY TEST 1
                        /0202           RUN ONLY TEST 2
                        /0203           RUN ONLY TEST 3
                        /0204           RUN ONLY TEST 4
                        /
                        /
                        /
                        /

                        /
                        /IOT COMMANDS FOR THE MC8-E EXTENDED MEMORY & INTERRUPT
                        /
                6004    GTF=6004        /GET INTERRUPT FLAGS
                                        /AC0    LINK
                                        /AC1    GREATER THAN FLAG
                                        /AC2    INTERRUPT BUS
                                        /AC3    INTERRUPT INHIBIT FLIP-FLOP
                                        /AC4    INTERRUPT ON
                                        /AC5    USER FLAG
                                        /AC6-8  INSTRUCTION FIELD
                                        /AC9-11 DATA FIELD
                6005    RTF=6005        /RESTORE INTERRUPT FLAGS
                                        /AC0    LINK
                                        /AC1    GREATER THAN FLAG
                                        /1      INTERRUPT INHIBIT FLIP-FLOP
                                        /1      INTERRUPT ON
                                        /AC5    USER FLAG
                                        /AC6-8  INSTRUCTION BUFFER
                                        /AC9-11 DATA FIELD
                6201    CDF0=6201       /CHANGE TO DATA FIELD 0
                6211    CDF1=6211       /CHANGE TO DATA FIELD 1
```



```
                6221    CDF2=6221       /CHANGE TO DATA FIELD 2
                6231    CDF3=6231       /CHANGE TO DATA FIELD 3
                6241    CDF4=6241       /CHANGE TO DATA FIELD 4
                6251    CDF5=6251       /CHANGE TO DATA FIELD 5
                6261    CDF6=6261       /CHANGE TO DATA FIELD 6
                6271    CDF7=6271       /CHANGE TO DATA FIELD 7
                6202    CIF0=6202       /CHANGE TO INSTRUCTION FIELD 0
                6212    CIF1=6212       /CHANGE TO INSTRUCTION FIELD 1
                6222    CIF2=6222       /CHANGE TO INSTRUCTION FIELD 2
                6232    CIF3=6232       /CHANGE TO INSTRUCTION FIELD 3
                6242    CIF4=6242       /CHANGE TO INSTRUCTION FIELD 4
                6252    CIF5=6252       /CHANGE TO INSTRUCTION FIELD 5
                6262    CIF6=6262       /CHANGE TO INSTRUCTION FIELD 6
                6272    CIF7=6272       /CHANGE TO INSTRUCTION FIELD 7
                6203    CBF0=6203       /CHANGE TO DATA AND INSTRUCTION FIELD 0
                6213    CBF1=6213       /CHANGE TO DATA AND INSTRUCTION FIELD 1
                6223    CBF2=6223       /CHANGE TO DATA AND INSTRUCTION FIELD 2
                6233    CBF3=6233       /CHANGE TO DATA AND INSTRUCTION FIELD 3
                6243    CBF4=6243       /CHANGE TO DATA AND INSTRUCTION FIELD 4
                6253    CBF5=6253       /CHANGE TO DATA AND INSTRUCTION FIELD 5
                6263    CBF6=6263       /CHANGE TO DATA AND INSTRUCTION FIELD 6
                6273    CBF7=6273       /CHANGE TO DATA AND INSTRUCTION FIELD 7
                6204    CINT=6204       /CLEAR USER INTERRUPT (TIME SHARE)
                6214    RDF=6214        /READ DATA FIELD INTO AC BITS 6-8
                6224    RIF=6224        /READ INSTRUCTION FIELD INTO AC BITS 6-8
                6234    RIB=6234        /READ INTERRUPT BUFFER
                                        /AC6-8 INSTRUCTION FIELD IN USE BEFORE LAST
                                        /       PROGRAM INTERRUPT.
                                        /AC9-11 DATA FIELD IN USE BEFORE LAST
                                        /       PROGRAM INTERRUPT.
                6244    RMF=6244        /RESTORE MEMORY FIELD
                                        /INSTRUCTION FIELD LOADED FROM SAVE FIELD 0-2
                                        /DATA FIELD LOADED FROM SAVE FIELD 3-5
                6254    SINT=6254       /SKIP ON USER INTERRUPT (TIME SHARE)
                6264    CUF=6264        /CLEAR USER FLAG (TIME SHARE)
                6274    SUF=6274        /SET USER FLAG (TIME SHARE)



                0020    *20

          0020  0000    PSR,    0                       /APT/
          0021  4000    MCW1,   4000                    /APT/
          0022  0000    MCW2,   0                       /APT/

                4425    LISN=JMS I      XLISP
                4426    PRNTMS=JMS I    MESAGP
                4427    ONEOCT=JMS I    ONEOCP
                4430    TWOOCT=JMS I    TWOOCP
                4432    PRNT1=JMS I     XPRN1P
                4431    FOROCT=JMS I    FOROCP
                4433    PRNT2=JMS I     XPRN2P
                4434    PRNT4=JMS I     XPRN4P
```

```
                4435    SPACE2=JMS I    SPCX2P
                4436    TYPE=JMS I      TYPEP
                4437    CRLF=JMS I      CRLFP
                4440    GETSR=  JMS  I  GETSRX
                4441    CHKCON= JMS  I  CHKCAC

                4440    LAS=    GETSR

                0024    *24


0024    0000    NOTTY,  0                       /THIS FLAG SET TO NON ZERO SIGNIFIES THAT
                                                /NO CONSOLE TERMINAL IS AVAILABLE

0025    4475    XLISP,  XLISN
0026    4600    MESAGP, MESAGX
0027    4401    ONEOCP, ONEOCK
0030    4411    TWOOCP, TWOOCK
0031    4640    FOROCP, FOROCK
0032    4667    XPRN1P, XPRNT1
0033    4424    XPRN2P, XPRNT2
0034    4654    XPRN4P, XPRNT4
0035    4467    SPCX2P, SPACX2
0036    4435    TYPEP,  XTYPE
0037    4452    CRLFP,  XCRLF
0040    4240    GETSRX, SRGET
0041    4562    CHKCAC, CONCHK


                /
                /CONSTANTS AND POINTERS
                /
0042    4000    SW0,    4000            /HALT AFTER ERROR
0043    2000    SW1,    2000            /INHIBIT ERROR TYPEOUT
0044    1000    SW2,    1000            /BELL ON ERROR
0045    0400    SW3,    400             /INHIBIT PROGRAM RELOCATION
0046    0200    SW4,    200             /CHANGE STACK LIMITS
0047    0100    SW5,    100             /HALT AFTER CURRENT TEST
0050    0000    STACK0, 0               /
0051    0000    STACK1, 0               /
0052    0000    STACK2, 0               /
0053    0000    STACK3, 0               /STACKS CONTAIN 0 IF SELECTED FOR TESTING
0054    0000    STACK4, 0               /
0055    0000    STACK5, 0               /
0056    0000    STACK6, 0               /
0057    0000    STACK7, 0               /
0060    0000    STK0,   0               /
0061    0000    STK1,   0               /
0062    0000    STK2,   0               /
0063    0000    STK3,   0               /0 IF RELOCATE
0064    0000    STK4,   0               /
0065    0000    STK5,   0               /
0066    0000    STK6,   0               /
```



```
0067    0000    STK7,   0               /
0070    0000    NORELO, 0               /PROG RELOCATION CONTROL (0=INH)
0071    1715    KABOVE, ABOVE           /CONTROL UPPER STACKS NOT TESTED
0072    1734    KBELOW, BELOW           /CONTROL LOWER STACKS NOT TESTED
0073    0000    HEAD1,  0               /ERROR HEADING CONTROL
0074    0000    INSAME, 0               /PROG IN SEL STACK
0075    0000    LEGAL0, 0               /LEGAL STACK SELECTION
0076    0000    RUNTST, 0               /6003=ALL, 0001=1, 0002=2, 2000=3, 4000=4
0077    0000    TESTAD, 0               /TEST ADDRESS COUNTER
0100    0000    KBINT,  0               /HIGHEST ACTUAL STACK IN SYSTEM
0101    0000    SSL,    0               /STARTING STACK LIMIT 00X0
0102    0000    ESL,    0               /ENDING STACK LIMIT 000X
0103    0000    STKPIN, 0               /STACK PROG IS IN 00X0
0104    0000    STKTST, 0               /STACK SEL FOR TEST 00X0
0105    0000    BDATA,  0               /BAD DATA
0106    0000    GDATA,  0               /GOOD DATA
0107    0000    MOVE,   0               /RELOCATION ADDRESS
0110    1745    KDOWN,  DOWN            /CONTROL LOWER STACKS TESTED
0111    0000    TEMP,   0               /INDIRECT ADDRESS TEMP STORAGE - CHEXM
0112    0000    COUNT,  0               /CHECKERBOARD ERROR COUNTER
0113    0000    ERRLOC, 0               /CODERR
0114    7777    M1,     -1              /CODERR - TEST 3 & 4
0115    7776    M2,     -2              /MESAGE - LEGAL
0116    7775    M3,     -3              /MESAGE
0117    7774    M4,     -4              /MESAGE - 4 WORDS
0120    7744    M34,    -34             /MESAGE

                SW911,                  /C8/ENDING STACK LIMIT (0-7).

0121    0007    K7,     7               /CODERR - ERRC - STACKS
0122    0010    K10,    10              /CHEXN
0123    0020    K20,    20              /CHEXN
0124    0030    K30,    30              /CHEXN
0125    0040    K40,    40              /CHEXN
0126    0050    K50,    50              /CHEXN
0127    0060    K60,    60              /CHEXN

                SW68,                   /C8/STARTING STACK LIMIT (0-7).

0130    0070    K70,    70              /CHEXN
0131    0077    K77,    77              /SIXTY - MESAGE
0132    0207    K207,   207             /MESAGE - CODERR
0133    0212    K212,   212             /MESAGE
0134    0215    K215,   215             /MESAGE
0135    0240    K240,   240             /TOSEL
0136    4060    K4060,  4060            /CODERR - ERRC
0137    6201    K6201,  6201            /CDF 0
0140    2042    XMESAG, MESAGE          /TTY ROUTINE POINTER
0141    2000    XSIXTY, SIXTY           /SIXTY ROUTINE POINTER
0142    2201    XCODER, CODERR          /ERROR ROUTINE POINTER
0143    2200    XRETUR, RETURN          /ERROR RETURN POINTER
0144    2251    XSTOP,  STOP            /STOP ROUTINE POINTER
0145    2250    XADDER, ADDER           /ADDRESS OF ERROR TYPEOUT POINTER
0146    0000    MIN5,   0               /FIVE MINUTE CONTROL
```

```
                0000    *0
0000  0000              0
0001  5001              JMP
0002  0002              2
0003  0003              3

0004  4334    KSFCHK,   CHKKSF
0005  4026    IAPTER,   APTER           /APT/
0006  4000    IAPTOK,   APTOK           /APT/

                0200    *200
                /
                /KM8-E EXTENDED MEMORY ADDRESS TEST (EA8E)
                /

0200  5777'   START,    JMP     APTIZ           /SETUP FOR APT/SETUP FOR NO TTY/SETUP FOR CONSOLE
0201  5776'             JMP     RUN1            /TEST 1
0202  5775'             JMP     RUN2            /TEST 2
0203  5774'             JMP     RUN3            /TEST 3
0204  5773'             JMP     RUN4            /TEST 4
0205  3076    EXTAD0,   DCA     RUNTST          /TEST CONTROL
0206  6002              IOF
0207  6224              RIF
0210  1137              TAD     K6201
0211  3212              DCA     .+1
0212  6201              CDF 0                   /MAKE DATA FIELD=INST FIELD

                /APT/   IF UNDER APT CONTROL NEXT TWO LOCS WILL = NOP.

                APTN00,                         /APT/

0213  4772'             JMS     TITLE           /TYPEOUT PROGRAM TITLE

                APTN01,                         /APT/

0214  4771'   CHEXA,    JMS     SETSW           /TYPEOUT TO SETUP SWITCHES
0215  7240              STA
0216  3070              DCA     NORELO          /CLEAR INH RELOCATION
0217  3077              DCA     TESTAD          /CLEAR TEST ADDR COUNTER
0220  7240              STA
0221  3073              DCA     HEAD1           /RESET ERROR HEADING
0222  1146              TAD     MIN5
0223  3770'             DCA     FIVE            /SETUP COUNTER
0224  4767'             JMS     DOWN+2          /CLEAR STACK SELECTION CONTROLS
0225  4440              LAS
0226  0130              AND     SW68
0227  3101              DCA     SSL             /STARTING STACK LIMIT
0230  4440              LAS
0231  0121              AND     SW911
0232  3102              DCA     ESL             /ENDING STACK LIMIT
0233  4766'             JMS     MSSL            /OBTAIN -SSL IN AC BITS 9-11
0234  1102              TAD     ESL
0235  7640              SZA CLA                 /SKIP IF SSL EQUALS ESL
0236  5262              JMP     CHEXC           /CONTINUE CHECK
0237  6224              RIF                     /READ INSTRUCTION FIELD
```



```
0240  7041              CIA
0241  1101              TAD     SSL
0242  7650              SNA CLA
0243  5765'             JMP     PINF            /PROGRAM IS IN THE SELECTED FIELD
0244  3070              DCA     NORELO          /INHIBIT PROGRAM RELOCATION
0245  1364              TAD     (CHEXB
0246  3763'             DCA     ABOVE-1         /STORE RETURN ADDRESS
0247  1102              TAD     ESL
0250  1071              TAD     KABOVE
0251  3111              DCA     TEMP
0252  5511              JMP I   TEMP            /INCREMENT UPPER FIELDS NOT TESTED
0253  1362    CHEXB,    TAD     (CHEXE
0254  3761'             DCA     BELOW+1         /STORE RETURN ADDRESS
0255  1102              TAD     ESL
0256  7041              CIA
0257  1072              TAD     KBELOW
0260  3111              DCA     TEMP
0261  5511              JMP I   TEMP            /INCREMENT LOWER FIELDS NOT TESTED
0262  4766'   CHEXC,    JMS     MSSL            /OBTAIN -SSL IN AC BITS 9-11
0263  1102              TAD     ESL
0264  7710              SPA CLA
0265  5302              JMP     CHEXD           /STARTING FIELD IS GREATER THAN ENDING FIELD
0266  1360              TAD     (CHEXC1
0267  3763'             DCA     ABOVE-1         /STORE RETURN ADDRESS
0270  1102              TAD     ESL
0271  1071              TAD     KABOVE
0272  3111              DCA     TEMP
0273  5511              JMP I   TEMP            /INCREMENT UPPER FIELDS NOT TESTED
0274  1362    CHEXC1,   TAD     (CHEXE
0275  3761'             DCA     BELOW+1         /STORE RETURN ADDRESS
0276  4766'             JMS     MSSL            /OBTAIN -SSL IN AC BITS 9-11
0277  1072              TAD     KBELOW
0300  3111              DCA     TEMP
0301  5511              JMP I   TEMP            /INCREMENT LOWER FIELDS NOT TESTED
0302  1357    CHEXD,    TAD     (CHEXD1
0303  3761'             DCA     BELOW+1         /STORE RETURN ADDRESS
0304  4766'             JMS     MSSL            /OBTAIN -SSL IN AC BITS 9-11
0305  1072              TAD     KRELOW
0306  3111              DCA     TEMP
0307  5511              JMP I   TEMP            /INCREMENT ALL LOWER FIELDS
0310  1362    CHEXD1,   TAD     (CHEXE
0311  3767'             DCA     DOWN+2          /STORE RETURN ADDRESS
0312  1102              TAD     ESL
0313  7041              CIA
0314  1110              TAD     KDOWN
0315  3111              DCA     TEMP
0316  5511              JMP I   TEMP            /RESTORE LOWER FIELDS TESTED
0317  4756'   CHEXE,    JMS     HIGHST          /FIND SYSTEMS HIGHEST STACK
0320  1100              TAD     KBINT
0321  1355              TAD     (260
0322  3100              DCA     KBINT           /MAKE HIGHEST STACK 0-7 FOR TYPEOUT

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = NOP.

                APTN02,                         /APT/
```

```
0323  4754'          JMS     TSTSYS          /TYPEOUT # OF STACKS IN SYSTEM
0324  1353          TAD     (CHEXE2
0325  3763'         DCA     ABOVE-1         /STORE RETURN ADDRESS
0326  1100          TAD     KBINT
0327  0121          AND     K7
0330  1071          TAD     KABOVE
0331  3111          DCA     TEMP
0332  5511          JMP I   TEMP            /INCREMENT UPPER STACKS NOT IN SYSTEM
0353  0400
0354  3017
0355  0260
0356  2731
0357  0310
0360  0274
0361  1735
0362  0317
0363  1714
0364  0253
0365  2127
0366  1751
0367  1747
0370  1262
0371  2624
0372  2600
0373  1624
0374  1617
0375  1612
0376  1605
0377  3600
      0400          PAGE

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = NOP.

                APTN05,                                 /APT/

0400  4777'    CHEXE2, JMS     TOSEL           /TYPEOUT STACK TEST SELECTION
0401  4210          JMS     LEGAL
0402  1070          TAD     NORELO
0403  7650          SNA CLA
0404  5276          JMP     CHEXM
0405  4776'         JMS     CHKSW3          /CHECK PROG RELO SW
0406  5775'         JMP     CHEXO           /RELOCATE PROGRAM
0407  5332          JMP     CHEXN           /INHIBIT PROGRAM RELOCATION

                /
                /CHECK FOR LEGAL STACK SELECTION
                /
0410  0000    LEGAL,  0
0411  7300          CLA CLL
0412  3074          DCA     INSAME          /CLEAR SAME CONTROL
0413  1115          TAD     M2
0414  3075          DCA     LEGAL0          /SETUP LEGAL CONTROL
0415  3104          DCA     STKTST
```



```
0416  1050          TAD     STACK0
0417  4263          JMS     LEGALA
0420  1122          TAD     K10
0421  3104          DCA     STKTST
0422  1051          TAD     STACK1
0423  4263          JMS     LEGALA
0424  1123          TAD     K20
0425  3104          DCA     STKTST
0426  1052          TAD     STACK2
0427  4263          JMS     LEGALA
0430  1124          TAD     K30
0431  3104          DCA     STKTST
0432  1053          TAD     STACK3
0433  4263          JMS     LEGALA
0434  1125          TAD     K40
0435  3104          DCA     STKTST
0436  1054          TAD     STACK4
0437  4263          JMS     LEGALA
0440  1126          TAD     K50
0441  3104          DCA     STKTST
0442  1055          TAD     STACK5
0443  4263          JMS     LEGALA
0444  1127          TAD     K60
0445  3104          DCA     STKTST
0446  1056          TAD     STACK6
0447  4263          JMS     LEGALA
0450  1130          TAD     K70
0451  3104          DCA     STKTST
0452  1057          TAD     STACK7
0453  4263          JMS     LEGALA
0454  2075          ISZ     LEGAL0
0455  5774'         JMP     NOSTK           /NO STACK SELECTION
0456  1074          TAD     INSAME
0457  7640          SZA CLA
0460  5773'         JMP     PINF            /PROG IN SELECTED FIELD
0461  3070          DCA     NORELO          /ONLY 1 STACK SELECTED
0462  5610          JMP I   LEGAL
                /
                /LEGAL STACK SELECTION SUBROUTINE
                /
0463  0000    LEGALA, 0
0464  7640          SZA CLA
0465  5663          JMP I   LEGALA          /NOT SELECTED
0466  2075          ISZ     LEGAL0
0467  7410          SKP
0470  5610          JMP I   LEGAL
0471  6224          RIF
0472  3103          DCA     STKPIN
0473  4772'         JMS     SAME            /PROG IN SEL STACK
0474  2074          ISZ     INSAME          /YES
0475  5663          JMP I   LEGALA

                /
                /NO PROGRAM RELOCATION AND TEST ONLY 1 STACK
                /
```

```
0476  6224    CHEXM,  RIF
0477  3103            DCA     STKPIN          /STACK PROGRAM IS IN
0500  1371            TAD     (STACK0-1
0501  3017            DCA     17
0502  3111            DCA     TEMP
0503  1417    CHEXM1, TAD I   17              /FIND STACK SEL FOR TEST
0504  7650            SNA CLA
0505  5310            JMP     CHEXM2
0506  2111            ISZ     TEMP
0507  5303            JMP     CHEXM1
0510  1111    CHEXM2, TAD     TEMP
0511  7104            CLL RAL
0512  7006            RTL
0513  3104            DCA     STKTST          /STACK SEL FOR TEST

                /IF UNDER APT NEXT LOCATION WILL = NOP

                APTN06,

0514  4770'           JMS     PNOREL          /TYPEOUT NO RELOCATION
0515  4767'   CHEXM3, JMS     TEST            /TEST THE SELECTED STACK

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMP APTJ50.

                APTJ00,                         /APT/

0516  4440            LAS
0517  0047            AND     SW5             /HALT AFTER TEST
0520  7650            SNA     CLA
0521  5325            JMP     .+4             /DO NOT HALT AFTER TEST
0522  4441            CHKCON                  /CHECK TO SEE IF CONSOLE ACTIVE
0523  7402            HLT                     /CONSOLE INACTIVE-HALT AFTER TEST
0524  4766'           JMS     PSEUDO          /HALT AFTER TEST-ASK SR QUESTION
0525  4440            LAS
0526  0046            AND     SW4             /CHANGE STACK LIMITS?
0527  7640            SZA CLA
0530  5765'           JMP     CHEXA           /YES

                APTJ50,                         /APT/

0531  5315            JMP     CHEXM3          /NO


                /
                /NO PROGRAM RELOCATION BUT TEST ALL SELECTED STACKS
                /

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = NOP.

                APTN03,                         /APT/

0532  4770'   CHEXN,  JMS     PNOREL          /TYPEOUT NO RELOCATION
0533  4764'   CHEXN0, JMS     CHEXN1          /TEST SEL'D STACKS

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMP APTJ51.
```



```
                APTJ01,                         /APT/

0534  4440            LAS
0535  0047            AND     SW5             /HALT AFTER TEST
0536  7650            SNA     CLA             /
0537  5343            JMP     .+4             /NO=DO NOT HALT AFTER TEST
0540  4441            CHKCON                  /CHECK TO SEE IF CONSOLE ACTIVE
0541  7402            HLT                     /INACTIVE CONSOLE-HALT AFTER TEST
0542  4766'           JMS     PSEUDO          /ASK SR QUESTION IF CONSOLE ACTIVE
0543  4440            LAS
0544  0046            AND     SW4             /CHANGE STACK LIMITS?
0545  7640            SZA CLA
0546  5765'           JMP     CHEXA           /YES

                APTJ51,                         /APT/

0547  4776'           JMS     CHKSW3          /NO
0550  5775'           JMP     CHEXO           /RELOCATE
0551  5333            JMP     CHEXN0          /CONTINUE

0564  0600
0565  0214
0566  4247
0567  1200
0570  2647
0571  0047
0572  1640
0573  2127
0574  2152
0575  1000
0576  1647
0577  3077
      0600      PAGE
                /
                /TEST STACKS CONTROL
                /
0600  0000    CHEXN1, 0
0601  7200            CLA
0602  6224            RIF
0603  3103            DCA     STKPIN          /STACK PROGRAM IS IN
0604  1057            TAD     STACK7
0605  7640            SZA CLA
0606  5222            JMP     CHEXN2
0607  1130            TAD     K70
0610  3104            DCA     STKTST          /STACK SEL FOR TEST
0611  3112            DCA     COUNT
0612  4777'           JMS     SAME            /PROG IN SEL STACK?
0613  5222            JMP     CHEXN2          /YES
0614  4776'           JMS     TEST            /NO = TEST THE SEL STACK
0615  1112            TAD     COUNT
0616  7640            SZA CLA
0617  2067            ISZ     STK7
0620  7410            SKP
0621  5217            JMP     .-2
```

```
0622  1056    CHEXN2, TAD     STACK6
0623  7640            SZA CLA
0624  5240            JMP     CHEXN3
0625  1127            TAD     K60
0626  3104            DCA     STKTST          /STACK SEL FOR TEST
0627  3112            DCA     COUNT
0630  4777'           JMS     SAME            /PROG IN SEL STACK?
0631  5240            JMP     CHEXN3          /YES
0632  4776'           JMS     TEST            /NO - TEST THE SEL STACK
0633  1112            TAD     COUNT
0634  7640            SZA CLA
0635  2066            ISZ     STK6
0636  7410            SKP
0637  5235            JMP     .-2
0640  1055    CHEXN3, TAD     STACK5
0641  7640            SZA CLA
0642  5256            JMP     CHEXN4
0643  1126            TAD     K50
0644  3104            DCA     STKTST          /STACK SEL FOR TEST
0645  3112            DCA     COUNT
0646  4777'           JMS     SAME            /PROG IN SEL STACK?
0647  5256            JMP     CHEXN4          /YES
0650  4776'           JMS     TEST            /NO - TEST THE SEL STACK
0651  1112            TAD     COUNT
0652  7640            SZA CLA
0653  2065            ISZ     STK5
0654  7410            SKP
0655  5253            JMP     .-2
0656  1054    CHEXN4, TAD     STACK4
0657  7640            SZA CLA
0660  5274            JMP     CHEXN5
0661  1125            TAD     K40
0662  3104            DCA     STKTST          /STACK SEL FOR TEST
0663  3112            DCA     COUNT
0664  4777'           JMS     SAME            /PROG IN SEL STACK?
0665  5274            JMP     CHEXN5          /YES
0666  4776'           JMS     TEST            /NO - TEST THE SEL STACK
0667  1112            TAD     COUNT
0670  7640            SZA CLA
0671  2064            ISZ     STK4
0672  7410            SKP
0673  5271            JMP     .-2
0674  1053    CHEXN5, TAD     STACK3
0675  7640            SZA CLA
0676  5312            JMP     CHEXN6
0677  1124            TAD     K30
0700  3104            DCA     STKTST          /STACK SEL FOR TEST
0701  3112            DCA     COUNT
0702  4777'           JMS     SAME            /PROG IN SEL STACK?
0703  5312            JMP     CHEXN6          /YES
0704  4776'           JMS     TEST            /NO - TEST THE SEL STACK
0705  1112            TAD     COUNT
0706  7640            SZA CLA
0707  2063            ISZ     STK3
0710  7410            SKP
```



```
0711  5307            JMP     .-2
0712  1052    CHEXN6, TAD     STACK2
0713  7640            SZA CLA
0714  5330            JMP     CHEXN7
0715  1123            TAD     K20
0716  3104            DCA     STKTST          /STACK SEL FOR TEST
0717  3112            DCA     COUNT
0720  4777'           JMS     SAME            /PROG IN SEL STACK?
0721  5330            JMP     CHEXN7          /YES
0722  4776'           JMS     TEST            /NO - TEST THE SEL STACK
0723  1112            TAD     COUNT
0724  7640            SZA CLA
0725  2062            ISZ     STK2
0726  7410            SKP
0727  5325            JMP     .-2
0730  1051    CHEXN7, TAD     STACK1
0731  7640            SZA CLA
0732  5346            JMP     CHEXN8
0733  1122            TAD     K10
0734  3104            DCA     STKTST          /STACK SEL FOR TES1
0735  3112            DCA     COUNT
0736  4777'           JMS     SAME            /PROG IN SEL STACK?
0737  5346            JMP     CHEXN8          /YES
0740  4776'           JMS     TEST            /NO - TEST THE SFL STACK
0741  1112            TAD     COUNT
0742  7640            SZA CLA
0743  2061            ISZ     STK1
0744  7410            SKP
0745  5343            JMP     .-2
0746  1050    CHEXN8, TAD     STACK0
0747  7640            SZA CLA
0750  5361            JMP     CHEXN9
0751  3104            DCA     STKTST          /STACK SEL FOR TEST
0752  3112            DCA     COUNT
0753  4777'           JMS     SAME            /PROG IN SEL STACK?
0754  5361            JMP     CHEXN9          /YES
0755  4776'           JMS     TEST            /NO - TEST THE SEL STACK
0756  1112            TAD     COUNT
0757  7640            SZA CLA
0760  2060            ISZ     STK0
0761  5600    CHEXN9, JMP I   CHEXN1
0762  5360            JMP     .-2

0776  1200
0777  1640
      1000          PAGE


                    /
                    /CHECK ALL SELECTED STACKS FROM EACH SELECTED STACK
                    /

                    /APT/   IF UNDER APT CONTROL NEXT LOC WILL = NOP.

                    APTN04,                         /APT/
```

```
1000  4777'  CHEX0,  JMS     PREL            /TYPEOUT RELOCATION
1001  4776'          JMS     RESTK           /RESTORE STK(S)
1002  4775'          JMS     CHEXN1          /TEST FROM PRESENT STACK
1003  4774'          JMS     CHKSW3
1004  7410           SKP
1005  5773'          JMP     CHEXN
1006  6224   CHEX0A, RIF
1007  3103           DCA     STKPIN          /STACK PROGRAM IS IN
1010  1067           TAD     STK7
1011  7640           SZA CLA
1012  5224           JMP     CHEX00
1013  1130           TAD     K70
1014  3104           DCA     STKTST          /STACK SEL FOR MOVE TO
1015  4772'          JMS     SAME            /PROG IN MOVE STACK?
1016  7410           SKP                     /YES
1017  4771'          JMS     RELO            /NO = RELOCATE PROGRAM
1020  4775'          JMS     CHEXN1          /TEST ALL SEL STACKS
1021  4774'          JMS     CHKSW3
1022  7410           SKP
1023  5773'          JMP     CHEXN
1024  1066   CHEX00, TAD     STK6
1025  7640           SZA CLA
1026  5240           JMP     CHEX01
1027  1127           TAD     K60
1030  3104           DCA     STKTST          /STACK SEL FOR MOVE TO
1031  4772'          JMS     SAME            /PROG IN MOVE STACK?
1032  7410           SKP                     /YES
1033  4771'          JMS     RELO            /NO = RELOCATE PROGRAM
1034  4775'          JMS     CHEXN1          /TEST ALL SEL STACKS
1035  4774'          JMS     CHKSW3
1036  7410           SKP
1037  5773'          JMP     CHEXN
1040  1065   CHEX01, TAD     STK5
1041  7640           SZA CLA
1042  5254           JMP     CHEX02
1043  1126           TAD     K50
1044  3104           DCA     STKTST
1045  4772'          JMS     SAME
1046  7410           SKP
1047  4771'          JMS     RELO
1050  4775'          JMS     CHEXN1
1051  4774'          JMS     CHKSW3
1052  7410           SKP
1053  5773'          JMP     CHEXN
1054  1064   CHEX02, TAD     STK4
1055  7640           SZA CLA
1056  5270           JMP     CHEX03
1057  1125           TAD     K40
1060  3104           DCA     STKTST
1061  4772'          JMS     SAME
1062  7410           SKP
1063  4771'          JMS     RELO
1064  4775'          JMS     CHEXN1
1065  4774'          JMS     CHKSW3
```



```
1066  7410           SKP
1067  5773'          JMP     CHEXN
1070  1063   CHEX03, TAD     STK3
1071  7640           SZA CLA
1072  5304           JMP     CHEX04
1073  1124           TAD     K30
1074  3104           DCA     STKTST
1075  4772'          JMS     SAME
1076  7410           SKP
1077  4771'          JMS     RELO
1100  4775'          JMS     CHEXN1
1101  4774'          JMS     CHKSW3
1102  7410           SKP
1103  5773'          JMP     CHEXN
1104  1062   CHEX04, TAD     STK2
1105  7640           SZA CLA
1106  5320           JMP     CHEX05
1107  1123           TAD     K20
1110  3104           DCA     STKTST
1111  4772'          JMS     SAME
1112  7410           SKP
1113  4771'          JMS     RELO
1114  4775'          JMS     CHEXN1
1115  4774'          JMS     CHKSW3
1116  7410           SKP
1117  5773'          JMP     CHEXN
1120  1061   CHEX05, TAD     STK1
1121  7640           SZA CLA
1122  5334           JMP     CHEX06
1123  1122           TAD     K10
1124  3104           DCA     STKTST
1125  4772'          JMS     SAME
1126  7410           SKP
1127  4771'          JMS     RELO
1130  4775'          JMS     CHEXN1
1131  4774'          JMS     CHKSW3
1132  7410           SKP
1133  5773'          JMP     CHEXN
1134  1060   CHEX06, TAD     STK0
1135  7640           SZA CLA
1136  5344           JMP     CHEX07
1137  3104           DCA     STKTST
1140  4772'          JMS     SAME
1141  7410           SKP
1142  4771'          JMS     RELO
1143  4775'          JMS     CHEXN1

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMP APTJ52,

                APTJ02,                         /APT/

1144  4440   CHEX07, LAS
1145  0047           AND     SW5             /HALT AFTER TEST
1146  7650           SNA     CLA             /
1147  5353           JMP     .+4             /NO DO NOT HALT AFTER TEST
```

```
1150  4441          CHKCON                  /CHECK TO SEE IF CONSOLE ACTIVE
1151  7402          HLT                     /CONSOLE INACTIVE-HALT AFTER TEST
1152  4770'         JMS     PSEUDO          /GO ASK SR QUESTION IF CONSOLE ACTIVE
1153  4440          LAS
1154  0046          AND     SW4             /CHANGE STACK LIMITS?
1155  7640          SZA CLA
1156  5767'         JMP     CHEXA           /YES

                APTJ52,                     /APT/

1157  4774'         JMS     CHKSW3          /NO
1160  5206          JMP     CHEXOA          /RELOCATE THE PROGRAM
1161  5773'         JMP     CHEXN           /INHIBIT PROGRAM RELOCATION

1167  0214
1170  4247
1171  1655
1172  1640
1173  0532
1174  1647
1175  0600
1176  3054
1177  2706
      1200          PAGE


                /
                /RUN THE SELECTED TEST(S) ON THE SELECTED FIELD (STKTST)
                /
1200  0000      TEST,   0

                /APT/   IF UNDER APT CONROL NEXT LOC WILL = JMS I IAPTOK,

1201  4404      APTOK0, JMS  I  KSFCHK          /THIS LOC OVERLAID IF APT
1202  7000              NOP                     /
1203  7300              CLA CLL
1204  1104              TAD     STKTST          /UPDATE COF TEST DATA FIELDS
1205  1137              TAD     K6201
1206  3275              DCA     TDF1
1207  1275              TAD     TDF1
1210  3777'             DCA     TDF2
1211  1777'             TAD     TDF2
1212  3776'             DCA     TDF3
1213  1776'             TAD     TDF3
1214  3775'             DCA     TDF4
1215  3112              DCA     COUNT           /CLEAR ERROR COUNT
1216  1076              TAD     RUNTST
1217  7010              RAR
1220  7630              SZL CLA
1221  4273              JMS     TEST1           /EXECUTE TEST 1
1222  1076              TAD     RUNTST
1223  7012              RTR
1224  7630              SZL CLA
1225  4774'             JMS     TEST2           /EXECUTE TEST 2
```



```
1226  1076              TAD     RUNTST
1227  7006              RTL
1230  7630              SZL CLA
1231  4773'             JMS     TEST3           /EXECUTE TEST 3
1232  1076              TAD     RUNTST
1233  7004              RAL
1234  7630              SZL CLA
1235  4772'             JMS     TEST4           /EXECUTE TEST 4

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMP APTJ53,

                APTJ03,                     /APT/

1236  4440              LAS
1237  0046              AND     SW4             /CHANGE STACK LIMITS?
1240  7640              SZA CLA
1241  5771'             JMP     CHEXA           /YES
1242  2262              ISZ     FIVE

                APTJ53,                     /APT/

1243  5600              JMP I   TEST            /NOT 5 MINUTES YET
1244  1146              TAD     MINS
1245  3262              DCA     FIVE            /RESTORE TIMER
1246  4441              CHKCON                  /CHECK TO SEE IF ACTIVE CONSOLE
1247  5255              JMP     .+6             /INACTIVE CONSOLE-DO NOT PRINT PASSES
1250  4426              PRNTMS                  /PRINT END OF PASS MESSAGE
1251  1263              PASMES                  /POINTER TO MESSAGE
1252  2261              ISZ     PASCNT          /ADD 1 TO THE PASS COUNTER
1253  1261              TAD     PASCNT          /GET THE COUNTER
1254  4434              PRNT4                   /PRINT THE 4 OCTAL DIGITS
1255  4540              JMS I   XMESAG
1256  4543              4543
1257  6500              6500
1260  5600              JMP I   TEST

1261  0000      PASCNT, 0

1262  0000      FIVE,   0

1263  4304      PASMES, TEXT                "#DHKMC-B PASS "
1264  1313
1265  1503
1266  5502
1267  4020
1270  0123
1271  2340
1272  0000


                /TEST 1
                /
                /WRITE THE VALUE OF EACH LOCATION INTO ITSELF AND CHECK
                /
1273  0000      TEST1,  0
```

```
1274  3077            DCA     TESTAD          /CLEAR TEST ADDRESS COUNTER
1275  6201  TDF1,     CDF0                    /CHANGE TO TEST DATA FIELD
1276  1077  TEST1A,   TAD     TESTAD
1277  3477            DCA I   TESTAD          /WRITE MEMORY
1300  2077            ISZ     TESTAD
1301  5276            JMP     TEST1A          /4096 TIMES
1302  1077  TEST1B,   TAD     TESTAD          /READ AND CHECK
1303  7041            CIA
1304  1477            TAD I   TESTAD
1305  7640            SZA CLA

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMS I IAPTER.

                APTE01,                         /APT/

1306  5316            JMP     ADDER1          /ADDRESS ERROR
1307  2077  ADDRT1,   ISZ     TESTAD
1310  5302            JMP     TEST1B          /CONTINUE READ AND CHECK
1311  1103            TAD     STKPIN
1312  1137            TAD     K6201
1313  3314            DCA     .+1
1314  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1315  5673            JMP I   TEST1           /DONE

1316  1077  ADDER1,   TAD     TESTAD
1317  3106            DCA     GDATA           /GOOD
1320  1477            TAD I   TESTAD
1321  3105            DCA     BDATA           /BAD
1322  1103            TAD     STKPIN
1323  1137            TAD     K6201
1324  3325            DCA     .+1
1325  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1326  4770'           JMS     ERR1            /ADDRESS ERROR TEST1
1327  1104            TAD     STKTST
1330  1137            TAD     K6201
1331  3332            DCA     .+1
1332  6201            CDF0                    /CHANGE TO TEST DATA FIELD
1333  5307            JMP     ADDRT1

1370  2272
1371  0214
1372  1514
1373  1443
1374  1400
1375  1516
1376  1445
1377  1402
      1400            PAGE


                /TEST 2
                /
                /WRITE THE COMPLEMENT VALUE OF EACH LOCATION INTO ITSELF AND CHECK
                /
1400  0000    TEST2,  0
```



```
1401  3077            DCA     TESTAD          /CLEAR TEST ADDRESS COUNTER
1402  6201  TDF2,     CDF0                    /CHANGE TO TEST DATA FIELD
1403  1077  TEST2A,   TAD     TESTAD
1404  7040            CMA
1405  3477            DCA I   TESTAD          /WRITE MEMORY
1406  2077            ISZ     TESTAD
1407  5203            JMP     TEST2A          /4096 TIMES
1410  1077  TEST2B,   TAD     TESTAD          /READ AND CHECK
1411  7001            IAC
1412  1477            TAD I   TESTAD
1413  7640            SZA CLA

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMS I IAPTER.

                APTE02,                         /APT/

1414  5224            JMP     ADDER2          /ADDRESS ERROR
1415  2077  ADDRT2,   ISZ     TESTAD
1416  5210            JMP     TEST2B          /CONTINUE READ AND CHECK
1417  1103            TAD     STKPIN
1420  1137            TAD     K6201
1421  3222            DCA     .+1
1422  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1423  5600            JMP I   TEST2           /DONE

1424  1077  ADDER2,   TAD     TESTAD
1425  7040            CMA
1426  3106            DCA     GDATA           /GOOD
1427  1477            TAD I   TESTAD
1430  3105            DCA     BDATA           /BAD
1431  1103            TAD     STKPIN
1432  1137            TAD     K6201
1433  3234            DCA     .+1
1434  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1435  4777'           JMS     ERR2            /ADDRESS ERROR TEST 2
1436  1104            TAD     STKTST
1437  1137            TAD     K6201
1440  3241            DCA     .+1
1441  6201            CDF0                    /CHANGE TO TEST DATA FIELD
1442  5215            JMP     ADDRT2

                /TEST 3
                /
                /WRITE THE VALUE OF EACH LOCATION INTO ITSELF AND CHECK BACKWARDS
                /
1443  0000    TEST3,  0
1444  3077            DCA     TESTAD          /CLEAR TEST ADDRESS COUNTER
1445  6201  TDF3,     CDF0                    /CHANGE TO TEST DATA FIELD
1446  1077  TEST3A,   TAD     TESTAD
1447  1114            TAD     M1
1450  3077            DCA     TESTAD
1451  1077            TAD     TESTAD
1452  3477            DCA I   TESTAD          /WRITE MEMORY
1453  1077            TAD     TESTAD
1454  7640            SZA CLA
```

```
1455  5246            JMP     TEST3A          /4096 TIMES
1456  1077  TEST3B,   TAD     TESTAD
1457  1114            TAD     M1
1460  3077            DCA     TESTAD
1461  1077            TAD     TESTAD          /READ AND CHECK
1462  7041            CIA
1463  1477            TAD I   TESTAD
1464  7640            SZA CLA

                      /APT/   IF UNDER AP CONTROL NEXT LOC WILL = JMS I IAPTER.

                      APTE03,                 /APT/

1465  5276            JMP     ADDER3          /ADDRESS ERROR
1466  1077  ADDRT3,   TAD     TESTAD
1467  7640            SZA CLA
1470  5256            JMP     TEST3B          /CONTINUE READ AND CHECK
1471  1103            TAD     STKPIN
1472  1137            TAD     K6201
1473  3274            DCA     .+1
1474  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1475  5643            JMP I   TEST3           /DONE

1476  1077  ADDER3,   TAD     TESTAD
1477  3106            DCA     GDATA           /GOOD
1500  1477            TAD I   TESTAD
1501  3105            DCA     BDATA           /BAD
1502  1103            TAD     STKPIN
1503  1137            TAD     K6201
1504  3305            DCA     .+1
1505  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1506  4776'           JMS     ERR3            /ADDRESS ERROR TEST 3
1507  1104            TAD     STKTST
1510  1137            TAD     K6201
1511  3312            DCA     .+1
1512  6201            CDF0                    /CHANGE TO TEST DATA FIELD
1513  5266            JMP     ADDRT3
                      /TEST 4
                      /
                      /WRITE THE COMPLEMENT VALUE OF EACH LOCATION INTO ITSELF
                      /AND CHECK BACKWARDS
                      /
1514  0000  TEST4,     0
1515  3077            DCA     TESTAD          /CLEAR TEST ADDRESS COUNTER
1516  6201  TDF4,     CDF0                    /CHANGE TO TEST DATA FIELD
1517  1077  TEST4A,   TAD     TESTAD
1520  1114            TAD     M1
1521  3077            DCA     TESTAD
1522  1077            TAD     TESTAD
1523  7040            CMA
1524  3477            DCA I   TESTAD          /WRITE MEMORY
1525  1077            TAD     TESTAD
1526  7640            SZA CLA
1527  5317            JMP     TEST4A          /4096 TIMES
1530  1077  TEST4B,   TAD     TESTAD
```



```
1531  1114            TAD     M1
1532  3077            DCA     TESTAD
1533  1077            TAD     TESTAD          /READ AND CHECK
1534  7001            IAC
1535  1477            TAD I   TESTAD
1536  7640            SZA CLA

                      /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMS I IAPTER.

                      APTE04,                 /APT/

1537  5350            JMP     ADDER4          /ADDRESS ERROR
1540  1077  ADDRT4,   TAD     TESTAD
1541  7640            SZA CLA
1542  5330            JMP     TEST4B          /CONTINUE READ AND CHECK
1543  1103            TAD     STKPIN
1544  1137            TAD     K6201
1545  3346            DCA     .+1
1546  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1547  5714            JMP I   TEST4           /DONE

1550  1077  ADDER4,   TAD     TESTAD
1551  7040            CMA
1552  3106            DCA     GDATA           /GOOD
1553  1477            TAD I   TESTAD
1554  3105            DCA     BDATA           /BAD
1555  1103            TAD     STKPIN
1556  1137            TAD     K6201
1557  3360            DCA     .+1
1560  6201            CDF0                    /CHANGE TO PROGRAM DATA FIELD
1561  4775'           JMS     ERR4            /ADDRESS ERROR TEST 4
1562  1104            TAD     STKTST
1563  1137            TAD     K6201
1564  3365            DCA     .+1
1565  6201            CDF0                    /CHANGE TO TEST DATA FIELD
1566  5340            JMP     ADDRT4

1575  2457
1576  2441
1577  2323
      1600            PAGE


                      /
                      /SETUP 5 MINUTE TIMER & TEST SELECTED TO RUN
                      /

1600  7200  RUN0,     CLA                     /RUN ALL TESTS
1601  1231            TAD     MIN50
1602  3146            DCA     MIN5
1603  1236            TAD     K6003
1604  5777'           JMP     EXTAD0
1605  7200  RUN1,     CLA                     /RUN ONLY TEST 1
1606  1232            TAD     MIN51
1607  3146            DCA     MIN5
```

```
1610  7001            IAC
1611  5777'           JMP     EXTAD0
1612  7300    RUN2,   CLA CLL                 /RUN ONLY TEST 2
1613  1233            TAD     MIN52
1614  3146            DCA     MIN5
1615  7005            IAC RAL
1616  5777'           JMP     EXTAD0
1617  7200    RUN3,   CLA                     /RUN ONLY TEST 3
1620  1234            TAD     MIN53
1621  3146            DCA     MIN5
1622  7132            STL RTR
1623  5777'           JMP     EXTAD0
1624  7200    RUN4,   CLA                     /RUN ONLY TEST 4
1625  1235            TAD     MIN54
1626  3146            DCA     MIN5
1627  7130            STL RAR
1630  5777'           JMP     EXTAD0

1631  7100    MIN50,  -700
1632  3500    MIN51,  -4300
1633  3500    MIN52,  -4300
1634  6000    MIN53,  -2000
1635  6000    MIN54,  -2000
1636  6003    K6003,  6003
1637  6203    K6203,  6203

                /
                /RETURN IF PROGRAM IS IN SELECTED STACK
                /RETURN+1 IF PROGRAM IS NOT IN SELECTED STACK
                /
1640  0000    SAME,   0
1641  1103            TAD     STKPIN
1642  7041            CIA
1643  1104            TAD     STKTST
1644  7640            SZA CLA
1645  2240            ISZ     SAME            /PROG NOT IN SEL STACK
1646  5640            JMP I   SAME
                /
                /CHECK PROGRAM RELOCATION SWITCH
                /RETURN IF RELOCATE, RETURN+1 IF INHIBIT RELOCATION
                /
1647  0000    CHKSW3, 0
1650  4440            LAS
1651  0045            AND     SW3
1652  7640            SZA CLA
1653  2247            ISZ     CHKSW3          /INHIBIT RELOCATION
1654  5647            JMP I   CHKSW3
                /
                /RELOCATE THE PROGRAM
                /
1655  0000    RELO,   0
1656  7200            CLA
1657  3112            DCA     COUNT           /CLEAR ERROR COUNTER
1660  3107            DCA     MOVE
1661  1137            TAD     K6201
```



```
1662  1103            TAD     STKPIN
1663  3274            DCA     RELO2
1664  1137            TAD     K6201
1665  1104            TAD     STKTST
1666  3276            DCA     RELO3
1667  1274            TAD     RELO2
1670  3301            DCA     RELO4
1671  1237            TAD     K6203
1672  1104            TAD     STKTST
1673  3312            DCA     RELO5
1674  6201    RELO2,  CDF0                    /MOVE FROM DATA FIELD
1675  1507            TAD I   MOVE
1676  6201    RELO3,  CDF0                    /MOVE TO DATA FIELD
1677  3507            DCA I   MOVE
1700  1507            TAD I   MOVE
1701  6201    RELO4,  CDF0                    /MOVE FROM DATA FIELD
1702  7041            CIA
1703  1507            TAD I   MOVE
1704  7640            SZA CLA

                /APT/   IF UNDER APT CONTROL NEXT LOC WILL = JMS I IAPTER,

                APTE00,                         /APT/

1705  4776'           JMS     ERRM            /MOVE ERROR
1706  2107            ISZ     MOVE
1707  5274            JMP     RELO2
1710  1112            TAD     COUNT
1711  7650            SNA CLA                 /SKIP IF MOVE ERROR
1712  6203    RELO5,  CBF0                    /CHANGE TO NEW PROG FIELD
1713  5655            JMP I   RELO
                /
                /INCREMENT CONTROL OF UPPER STACKS NOT TESTED AND/OR
                /STACKS NOT IN THE SYSTEM
                /
1714  0000            0                       /RETURN ADDRESS
1715  2051    ABOVE,  ISZ     STACK1
1716  2052            ISZ     STACK2
1717  2053            ISZ     STACK3
1720  2054            ISZ     STACK4
1721  2055            ISZ     STACK5
1722  2056            ISZ     STACK6
1723  2057            ISZ     STACK7
1724  5714            JMP I   ABOVE+1

                /
                /INCREMENT CONTROL OF LOWER STACKS NOT TESTED
                /
1725  2056            ISZ     STACK6
1726  2055            ISZ     STACK5
1727  2054            ISZ     STACK4
1730  2053            ISZ     STACK3
1731  2052            ISZ     STACK2
1732  2051            ISZ     STACK1
1733  2050            ISZ     STACK0
```

```
1734  5735    BELOW,  JMP I   .+1
1735  0000            0
                      /
                      /CLEAR ALL STACKS OR STACKS TO BE TESTED
                      /
1736  3057            DCA     STACK7
1737  3056            DCA     STACK6
1740  3055            DCA     STACK5
1741  3054            DCA     STACK4
1742  3053            DCA     STACK3
1743  3052            DCA     STACK2
1744  3051            DCA     STACK1
1745  3050    DOWN,   DCA     STACK0
1746  5747            JMP I   .+1
1747  0000            0
1750  5336            JMP     .-12            /RETURN ADDRESS
                                              /CLEAR ALL STACK SELECTION CONTROLS
                      /
                      /OBTAIN -SSL (MINUS STARTING STACK LIMIT)
                      /
1751  0000    MSSL,   0
1752  1101            TAD     SSL
1753  7112            CLL RTR
1754  7010            RAR
1755  7041            CIA
1756  5751            JMP I   MSSL

1776  2475
1777  0205
      2000            PAGE

                      /
                      /CONVERT OCTAL NUMBERS FOR TYPEOUT
                      /
2000  0000    SIXTY,  0
2001  7300            CLA CLL
2002  1600            TAD I   SIXTY           /GET ADDRESS OF OPERAND
2003  3237            DCA     SIXTY0
2004  2200            ISZ     SIXTY
2005  1600            TAD I   SIXTY           /GET STORAGE ADDRESS
2006  3240            DCA     SIXTY1
2007  2200            ISZ     SIXTY           /CORRECT RETURN ADDRESS
2010  1131            TAD     K77
2011  7040            CMA                     /AC=7700
2012  0637            AND I   SIXTY0          /AND OPERAND FIRST 2 DIGITS
2013  7112            CLL RTR
2014  7012            RTR
2015  7012            RTR                     /POSITION FIRST 2 DIGITS
2016  4224            JMS     CNV             /CONVERT DIGITS FOR TYPEOUT
2017  2240            ISZ     SIXTY1          /INCREMENT STORAGE ADDRESS
2020  1131            TAD     K77
2021  0637            AND I   SIXTY0          /AND OPERAND SECOND 2 DIGITS
2022  4224            JMS     CNV             /CONVERT DIGITS FOR TYPEOUT
2023  5600            JMP I   SIXTY
2024  0000    CNV,    0
2025  3241            DCA     SIXTY2          /SAVE DIGITS
```



```
2026  1241            TAD     SIXTY2
2027  7106            CLL RTL
2030  7004            RAL
2031  0325            AND     K707            /AND LEFT DIGIT
2032  1241            TAD     SIXTY2
2033  0325            AND     K707            /AND RIGHT DIGIT
2034  1326            TAD     K6060
2035  3640            DCA I   SIXTY1          /STORE CONVERTED DIGITS
2036  5624            JMP I   CNV

2037  0000    SIXTY0, 0                       /ADDRESS OF OPERAND
2040  0000    SIXTY1, 0                       /STORAGE ADDRESS
2041  0000    SIXTY2, 0                       /TEMPORARY STORAGE

                      /

                      /
                      /TELETYPE OUTPUT ROUTINE WITH BELL
                      /
2042  0000    MESAGE, 0

2043  7200            CLA                     /APT/
2044  1022            TAD     HCW2            /APT/UNDER APT CONTROL?
2045  7700            SMA CLA                 /APT/SKP IF YES,
2046  5252            JMP     APT000          /APT/
2047  1242            TAD     MESAGE          /APT/FORCE AN ERROR CALL TO APT,
2050  3777'           DCA     APTER           /APT/
2051  5776'           JMP     APTER+1         /APT/

                APT000,                       /APT/

2052  7240            STA
2053  1242            TAD     MESAGE          /FIRST WORD -1
2054  3010            DCA     10

                APT001,

2055  1410            TAD I   10
2056  3267            DCA     MSRGHT
2057  1267            TAD     MSRGHT
2060  7112            CLL RTR
2061  7012            RTR
2062  7012            RTR                     /POSITION FIRST CHARACTER
2063  4270            JMS     TYPECH          /TYPEOUT FIRST CHARACTER
2064  1267            TAD     MSRGHT
2065  4270            JMS     TYPECH          /TYPEOUT SECOND CHARACTER
              /*APT*/  JMP     MESAGE+4        /CONTINUE TYPING

2066  5255            JMP     APT001          /APT/CONTINUE TYPING,

2067  0000    MSRGHT, 0
2070  0000    TYPECH, 0
2071  0131            AND     K77
2072  7450            SNA
2073  5410            JMP I   10              /IS IT END OF MESSAGE?
                                              /RETURN TO PROGRAM
```

```
2074  1120            TAD     M34             /SUBTRACT 34
2075  7440            SZA
2076  5301            JMP     .+3
2077  1132            TAD     K207            /CODE IS BELL
2100  5321            JMP     MTP
2101  1117            TAD     M4              /SUBTRACT 4
2102  7500            SMA                     /CODE LESS THAN 40?
2103  5306            JMP     .+3             /NO
2104  1324            TAD     K340            /YES, ADD 300, CODE IS ALPHA
2105  5321            JMP     MTP
2106  1116            TAD     M3              /SUBTRACT 3
2107  7440            SZA
2110  5313            JMP     .+3
2111  1133            TAD     K212            /CODE IS LINE FEED
2112  5321            JMP     MTP
2113  1115            TAD     M2              /SUBTRACT 2
2114  7440            SZA
2115  5320            JMP     .+3
2116  1134            TAD     K215            /CODE IS CR
2117  7410            SKP
2120  1323            TAD     K245            /ADD 200 TO OTHER CODES >40
2121  4436    MTP,    TYPE                    /TYPEOUT CHARACTER IN AC
2122  5670            JMP I   TYPECH

2123  0245    K245,   245
2124  0340    K340,   340
2125  0707    K707,   707
2126  6060    K6060,  6060


                /
                /TYPEOUT "PROGRAM IS IN SELECTED FIELD"
                /
2127  1024    PINF,   TAD     NOTTY           /GET THE TERMINAL FLAG
2130  7640            SZA     CLA             /IS THERE A TERMINAL ON THE SYSTEM
2131  5775'           JMP     CHEXA           /NO-GO SETUP THE SR AGAIN
2132  4540            JMS  I  XMESAG
2133  4543            TEXT                    "%#PROGRAM IN SELECTED FIELD"
2134  2022
2135  1707
2136  2201
2137  1540
2140  1116
2141  4023
2142  0514
2143  0503
2144  2405
2145  0440
2146  0611
2147  0514
2150  0400
2151  5775'           JMP     CHEXA           /SETUP SWITCHES AGAIN

                /
                /TYPEOUT "NONE" FOR NO LEGAL STACK SELECTION
```



```
                /
2152  1024    NOSTK,  TAD     NOTTY           /GET THE TERMINAL FLAG
2153  7640            SZA     CLA             /IS THERE A TERMINAL ON THE SYSTEM
2154  5775'           JMP     CHEXA           /NO-ABORT MESSAGE AND GO SETUP SR AGAIN
2155  4540            JMS  I  XMESAG
2156  1617            TEXT                    "NONE"
2157  1605
2160  0000
2161  5775'           JMP     CHEXA           /SETUP SWITCHES  AGAIN

2175  0214
2176  4027
2177  4026
      2200            PAGE

                /
                /ERROR ROUTINE (BELL ON ERROR HAS PRIORITY)
                /
2200  0000    RETURN, 0                       /PROGRAM RETURN ADDRESS
2201  6002    CODERR, IOF
2202  4440            LAS                     /CHECK FOR BELL ON ERROR
2203  0044            AND     SW2
2204  7650            SNA CLA
2205  5215            JMP     NOBELL
2206  1024    RBELL,  TAD     NOTTY           /GET TERMINAL FLAG
2207  7640            SZA     CLA             /IS THERE A TERMINAL ON THE SYSTEM
2210  5213            JMP     .+3             /NO-DO NOT RING BELL BUT RETURN TO PRGRAM
2211  1132            TAD     K207            /BELL CODE
2212  4436            TYPE                    /RING BELL
2213  4404            JMS  I  KSFCHK          /CHECK FOR CONSOLE RECEIVE FLAG
2214  5600            JMP I   RETURN
2215  4440    NOBELL, LAS                     /CHECK FOR INHIBIT TYPEOUT
2216  0043            AND     SW1
2217  7640            SZA CLA
2220  5251            JMP     STOP            /INHIBIT TYPEOUT
2221  6224            RIF                     /READ INST FIELD
2222  7012            RTR
2223  7010            RAR
2224  0121            AND     K7
2225  1136            TAD     K4060
2226  3242            DCA     ERROR0
2227  1200            TAD     RETURN
2230  1114            TAD     M1
2231  3113            DCA     ERRLOC
2232  4541            JMS I   XSIXTY
2233  0113            ERRLOC
2234  2243            ERROR1
2235  1024            TAD     NOTTY           /GET THE TERMINAL FLAG
2236  7640            SZA     CLA             /IS THERE A TERMINAL ON SYSTEM
2237  5650            JMP  I  ADDER           /NO-GO TO APPROPRIATE ERROR ROUTINE
2240  4540            JMS I   XMESAG          /TYPEOUT ERROR LOCATION
2241  4543            4543
2242  0000    ERROR0, 0                       /FIELD
2243  0000    ERROR1, 0
2244  0000            0                       /PROGRAM LOCATION OF ERROR JMS
```

```
2245  4040            4040
2246  0000            0000
2247  5650            JMP I   .+1             /TYPEOUT ERROR
2250  0000    ADDER,  0                       /ADDRESS OF ERROR TYPEOUT

2251  4440    STOP,   LAS                     /HALT AFTER ERROR
2252  0042            AND     SW0
2253  7650            SNA CLA
2254  5263            JMP     LIMIT           /INHIBIT ERROR HALT
2255  4441            CHKCON                  /CHECK TO SEE IF CONSOLE ACTIVE
2256  7610            SKP     CLA             /CONSOLE INACTIVE-HALT WITH AC=ERR LOC
2257  5777'           JMP     CONHLT          /CONSOLE ACTIVE-GO TYPE ERR LOCATION
2260  7240            CLA     CMA             /SUBTRACT 1 FROM JMS ERROR
2261  1200            TAD     RETURN          /GET THE ERR ADDRESS +1
2262  7402            HLT                     /HALT AFTER ERROR-AC=ERROR LOCATION
2263  7200    LIMIT,  CLA
2264  4404            JMS  I  KSFCHK          /CHECK FOR CONSOLE RECEIVE FLAG
2265  4440            LAS
2266  0046            AND     SW4             /CHANGE STACK LIMITS?
2267  7640            SZA CLA
2270  5776'           JMP     CHEXA           /YES
2271  5600            JMP I   RETURN          /NO


                      /
                      /ADDRESS ERROR TEST 1
                      /
2272  0000    ERR1,   0
2273  2112            ISZ     COUNT           /ADDRESS ERROR OCCURRED
2274  7410            SKP
2275  5273            JMP     .-2
2276  7200            CLA
2277  1272            TAD     ERR1
2300  3543            DCA I   XRETUR          /STORE RETURN ADDRESS
2301  1304            TAD     .+3
2302  3545            DCA I   XADDER          /STORE ERROR TYPEOUT ADDRESS
2303  7410            SKP
2304  3400            PERR1
2305  1375            TAD     (6100
2306  3774'           DCA     Z24             /TEST 1
2307  4440    ERR1A,  LAS
2310  0044            AND     SW2             /BELL ON ERROR?
2311  7640            SZA CLA
2312  5206            JMP     RBELL           /YES
2313  4440            LAS
2314  0043            AND     SW1             /INHIBIT ERROR TYPEOUT?
2315  7640            SZA CLA
2316  5544            JMP I   XSTOP           /YES
2317  2073            ISZ     HEAD1
2320  7410            SKP
2321  4773'           JMS     HEAD12          /TYPEOUT ERROR HEADING
2322  5542            JMP I   XCODER          /GO TO ERROR ROUTINE
                      /
                      /ADDRESS ERROR TEST 2
```



```
                      /
2323  0000    ERR2,   0
2324  2112            ISZ     COUNT           /ADDRESS ERROR OCCURRED
2325  7410            SKP
2326  5324            JMP     .-2
2327  7200            CLA
2330  1323            TAD     ERR2
2331  3543            DCA I   XRETUR          /STORE RETURN ADDRESS
2332  1335            TAD     .+3
2333  3545            DCA I   XADDER          /STORE ERROR TYPEOUT ADDRESS
2334  7410            SKP
2335  3400            PERR1
2336  1372            TAD     (6200
2337  3774'           DCA     Z24             /TEST 2
2340  5307            JMP     ERR1A

2372  6200
2373  2510
2374  3434
2375  6100
2376  0214
2377  2400
      2400            PAGE

                      /HALT AFTER ERROR MESSAGE-REPORTED IF SR0=1

2400  4426    CONHLT, PRNTMS                  /GO REPORT ERROR MESSAGE
2401  2407            ERRMES                  /POINTER TO ERROR MESSAGE
2402  7240            CLA     CMA             /PICK UP ERROR LOCATION -1
2403  1777'           TAD     RETURN          /
2404  4434            PRNT4                   /PRINT THE 4 OCTAL DIGITS
2405  4776'           JMS     PSEUDO          /FORCE PROGRAM TO SR QUESTION
2406  5775'           JMP     LIMIT           /GO CHECK SR 4 FOR STACK CHANGES

2407  4323    ERRMES, TEXT                    "#SW0=1-HALT AFTER ERROR-ERROR DETECTED AT LOCATION "
2410  2760
2411  7561
2412  5510
2413  0114
2414  2440
2415  0106
2416  2405
2417  2240
2420  0522
2421  2217
2422  2255
2423  0522
2424  2217
2425  2240
2426  0405
2427  2405
2430  0324
2431  0504
2432  4001
```

```
2433  2440
2434  1417
2435  0301
2436  2411
2437  1716
2440  4000

                /
                /ADDRESS ERROR TEST 3
                /
2441  0000    ERR3,   0
2442  2112            ISZ     COUNT           /ADDRESS ERROR OCCURRED
2443  7410            SKP
2444  5242            JMP     .-2
2445  7200            CLA
2446  1241            TAD     ERR3
2447  3543            DCA I   XRETUR          /STORE RETURN ADDRESS
2450  1253            TAD     .+3
2451  3545            DCA I   XADDER          /STORE ERROR TYPEOUT ADDRESS
2452  7410            SKP
2453  3400            PERR1
2454  1374            TAD     (6300
2455  3773'           DCA     Z24             /TEST 3
2456  5772'           JMP     ERR1A

                /
                /ADDRESS ERROR TEST 4
                /
2457  0000    ERR4,   0
2460  2112            ISZ     COUNT           /ADDRESS ERROR OCCURRED
2461  7410            SKP
2462  5260            JMP     .-2
2463  7200            CLA
2464  1257            TAD     ERR4
2465  3543            DCA I   XRETUR          /STORE RETURN ADDRESS
2466  1271            TAD     .+3
2467  3545            DCA I   XADDER          /STORE ERROR TYPEOUT ADDRESS
2470  7410            SKP
2471  3400            PERR1
2472  1371            TAD     (6400
2473  3773'           DCA     Z24             /TEST 4
2474  5772'           JMP     ERR1A

                /
                /RELOCATION MOVE ERROR OCCURRED
                /
2475  0000    ERRM,   0
2476  2112            ISZ     COUNT           /RELO ERROR OCCURRED
2477  7410            SKP
2500  5276            JMP     .-2
2501  7200            CLA
2502  1275            TAD     ERRM
2503  3543            DCA I   XRETUR          /STORE RETURN ADDRESS
2504  1307            TAD     .+3
2505  3545            DCA I   XADDER          /STORE ERROR TYPEOUT ADDRESS
```



```
2506  5542            JMP I   XCODER
2507  3436            PERRM

                /
                /TYPEOUT TEST 1 OR 2 ERROR HEADING
                /
2510  0000    HEAD12, 0
2511  1024            TAD     NOTTY           /GET THE TERMINAL FLAG
2512  7640            SZA     CLA             /IS THERE A TERMINAL ON THE SYSTEM
2513  5710            JMP  I  HEAD12          /NO-RETURN WITHOUT TELETYPE OUTPUT
2514  4540            JMS I   XMESAG
2515  4543            TEXT    "%#PR LOC   ADDR   GOOD  BAD  TEST"
2516  2022
2517  4014
2520  1703
2521  4040
2522  4001
2523  0404
2524  2240
2525  4040
2526  0717
2527  1704
2530  4040
2531  0201
2532  0440
2533  4024
2534  0523
2535  2400
2536  5710            JMP I   HEAD12

2571  6400
2572  2307
2573  3434
2574  6300
2575  2263
2576  4247
2577  2200
      2600            PAGE

                /
                /TYPEOUT PROGRAM TITLE
                /
2600  0000    TITLE,  0
2601  1024            TAD     NOTTY           /CHECK TO SEE IF CONSOLE AVAILABLE
2602  7640            SZA     CLA             /IS THERE A CONSOLE TERMINAL?
2603  5600            JMP  I  TITLE           /NO-DO NOT TRY TO TYPE MESSAGE
2604  4540            JMS I   XMESAG
2605  4543            TEXT            "%##EA8-E EXT MEM ADDR TEST#"
2606  4305
2607  0170
2610  5585
2611  4005
2612  3024
2613  4015
2614  0515
```

```
2615  4001
2616  0404
2617  2240
2620  2405
2621  2324
2622  4300
2623  5600            JMP I   TITLE

                      /
                      /TYPEOUT TO SET SWITCHES
                      /
2624  0000    SETSW,  0
2625  1024            TAD     NOTTY           /GET THE CONSOLE FLAG
2626  7640            SZA     CLA             /IS THERE A CONSOLE TERMINAL
2627  5244            JMP     SETSW1          /NO-HALT TO ALLOW SR SETTINGS
2630  4540            JMS I   XMESAG
2631  4543            TEXT            "%#SETUP SR & CONTX#"
2632  2305
2633  2425
2634  2040
2635  2322
2636  4046
2637  4003
2640  1716
2641  2445
2642  4300
2643  4441            CHKCON                  /CHECK TO SEE IF CONSOLE ACTIVE
2644  7402    SETSW1, HLT                     /CONSOLE INACTIVE-SET SR AND CONT
2645  4777'           JMS     PSEUDO          /ASK SR QUESTION IF ON ACTIVE CONSOLE
2646  5624            JMP I   SETSW
                      /
                      /TYPEOUT 'NO PROGRAM RELOCATION WILL OCCUR'
                      /
2647  0000    PNOREL, 0
2650  1024            TAD     NOTTY           /GET THE CONSOL FLAG
2651  7640            SZA     CLA             /IS THERE A  TERMINAL
2652  5303            JMP     Z8+1            /NO ABORT ROUTINE
2653  4540            JMS I   XMESAG
2654  4543            TEXT            "%#NO RELOCATION, PROG IN STACK "
2655  1617
2656  4022
2657  0514
2660  1703
2661  0124
2662  1117
2663  1654
2664  4020
2665  2217
2666  0740
2667  1116
2670  4023
2671  2401
2672  0313
2673  4000
2674  6224            RIF
```



```
2675  7106            CLL RTL
2676  7004            RAL
2677  1376            TAD     (6000
2700  3302            DCA     Z8
2701  4540            JMS I   XMESAG
2702  0000    Z8,     0
2703  7240            STA
2704  3073            DCA     HEAD1           /RESET ERROR HEADING
2705  5647            JMP I   PNOREL
                      /
                      /PROGRAM RELOCATION WILL OCCUR
                      /
2706  0000    PREL,   0
2707  1024            TAD     NOTTY           /GET THE CONSOLE TERMINAL FLAG
2710  7640            SZA     CLA             /IS THERE A TERMINAL ON SYSTEM
2711  5326            JMP     PREL1           /NO-ABORT TYPEOUT THEN RETURN
2712  4540            JMS I   XMESAG
2713  4543            TEXT            "%#PROG WILL RELOCATE"
2714  2022
2715  1707
2716  4027
2717  1114
2720  1440
2721  2205
2722  1417
2723  0301
2724  2405
2725  0000
2726  7240    PREL1,  STA
2727  3073            DCA     HEAD1           /RESET ERROR HEADING
2730  5706            JMP I   PREL
                      /


                      /
                      /FIND HIGHEST STACK NUMBER IN THIS SYSTEM
                      /
2731  0000    HIGHST, 0
2732  7300            CLA CLL
2733  3100            DCA     KBINT           /CLEAR HIGH STACK COUNTER
2734  6211            CDF1
2735  4775'           JMS     CSS             /CHECK FOR FIELD 1
2736  6221            CDF2
2737  4775'           JMS     CSS             /CHECK FOR FIELD 2
2740  6231            CDF3
2741  4775'           JMS     CSS             /CHECK FOR FIELD 3
2742  6241            CDF4
2743  4775'           JMS     CSS             /CHECK FOR FIELD 4
2744  6251            CDF5
2745  4775'           JMS     CSS             /CHECK FOR FIELD 5
2746  6261            CDF6
2747  4775'           JMS     CSS             /CHECK FOR FIELD 6
2750  6271            CDF7
2751  4775'           JMS     CSS             /CHECK FOR FIELD 7
```

```
2752  5731    KHIGH,  JMP I   HIGHST

2775  3000
2776  6000
2777  4247
      3000            PAGE
                      /
                      /CHECK IF SELECTED STACK IS IN SYSTEM
                      /
3000  0000    CSS,    0
3001  7300            CLA CLL
3002  6224            RIF
3003  1137            TAD     K6201
3004  3210            DCA     CSSB
3005  1114            TAD     M1
3006  3615            DCA I   CHECK
3007  1615            TAD I   CHECK
3010  6201    CSSB,   CDF     00              /PROGRAM DATA FIELD
3011  7650            SNA CLA                 /SKIP IF STACK IS IN SYSTEM
3012  5777'           JMP     KHIGH
3013  2100            ISZ     KBINT           /INCREMENT STACK COUNTER
3014  5600            JMP I   CSS

3015  3016    CHECK,  CHECK0
3016  0000    CHECK0, 0


                      /
                      /TYPEOUT NUMBER OF STACKS IN SYSTEM
                      /
3017  0000    TSTSYS, 0
3020  1024            TAD     NOTTY           /GET TERMINAL FLAG
3021  7640            SZA     CLA             /IS THERE A TERMINAL ON THE SYSTEM
3022  5617            JMP  I  TSTSYS          /NO-ABORT TELETYPE MESSAGE
3023  4540            JMS I XMESAG
3024  4543            4543
3025  0000            0000
3026  1100            TAD     KBINT
3027  7001            IAC
3030  4436            TYPE                    /TYPEOUT NUMBER
3031  4540            JMS I   XMESAG
3032  4023            TEXT            " STACKS IN THIS SYSTEM"
3033  2401
3034  0313
3035  2340
3036  1116
3037  4024
3040  1011
3041  2340
3042  2331
3043  2324
3044  0515
3045  0000
3046  5617            JMP I   TSTSYS
```



```
                      /
                      /TYPEOUT CHARACTER IN THE AC AND A SPACE
                      /
3047  0000    TYPESP, 0
3050  4436            TYPE                    /TYPEOUT CHAR IN AC
3051  1135            TAD     K240
3052  4436            TYPE                    /TYPE A SPACE
3053  5647            JMP I   TYPESP
                      /
                      /RESTORE STACKS FOR RELOCATION
                      /
3054  0000    RESTK,  0
3055  7200            CLA
3056  1050            TAD     STACK0
3057  3060            DCA     STK0
3060  1051            TAD     STACK1
3061  3061            DCA     STK1
3062  1052            TAD     STACK2
3063  3062            DCA     STK2
3064  1053            TAD     STACK3
3065  3063            DCA     STK3
3066  1054            TAD     STACK4
3067  3064            DCA     STK4
3070  1055            TAD     STACK5
3071  3065            DCA     STK5
3072  1056            TAD     STACK6
3073  3066            DCA     STK6
3074  1057            TAD     STACK7
3075  3067            DCA     STK7
3076  5654            JMP I   RESTK


                      /
                      /TYPEOUT STACKS SELECTED FOR TESTING
                      /
3077  0000    TOSEL,  0
3100  1024            TAD     NOTTY           /GET TERMINAL FLAG
3101  7640            SZA     CLA             /IS THERE A TERMINAL ON SYSTEM
3102  5677            JMP  I  TOSEL           /NO-ABORT TELETYPE MESSAGE
3103  4540            JMS I   XMESAG
3104  4543            TEXT            "%#STACKS SEL'D ARE "
3105  2324
3106  0103
3107  1323
3110  4023
3111  0514
3112  4704
3113  4001
3114  2205
3115  4000
3116  1057            TAD     STACK7
3117  7640            SZA CLA
3120  5323            JMP     .+3
3121  1376            TAD     (267
```

```
3122  4247            JMS     TYPESP          /STACK 7 IS SELECTED
3123  1056            TAD     STACK6
3124  7640            SZA CLA
3125  5330            JMP     .+3
3126  1375            TAD     (266
3127  4247            JMS     TYPESP          /STACK 6 IS SELECTED
3130  1055            TAD     STACK5
3131  7640            SZA CLA
3132  5335            JMP     .+3
3133  1374            TAD     (265
3134  4247            JMS     TYPESP          /STACK 5 IS SELECTED
3135  1054            TAD     STACK4
3136  7640            SZA CLA
3137  5342            JMP     .+3
3140  1373            TAD     (264
3141  4247            JMS     TYPESP          /STACK 4 IS SELECTED
3142  1053            TAD     STACK3
3143  7640            SZA CLA
3144  5347            JMP     .+3
3145  1372            TAD     (263
3146  4247            JMS     TYPESP          /STACK 3 IS SELECTED
3147  1052            TAD     STACK2
3150  7640            SZA CLA
3151  5354            JMP     .+3
3152  1371            TAD     (262
3153  4247            JMS     TYPESP          /STACK 2 IS SELECTED
3154  1051            TAD     STACK1
3155  7640            SZA CLA
3156  5361            JMP     .+3
3157  1370            TAD     (261
3160  4247            JMS     TYPESP          /STACK 1 IS SELECTED
3161  1050            TAD     STACK0
3162  7640            SZA CLA
3163  5366            JMP     .+3
3164  1367            TAD     (260
3165  4247            JMS     TYPESP          /STACK 0 IS SELECTED
3166  5677            JMP I   TOSEL

3167  0260
3170  0261
3171  0262
3172  0263
3173  0264
3174  0265
3175  0266
3176  0267
3177  2752
      3200            PAGE


                      /
                      /TWO SPECIAL SCOPE LOOPS
                      /

3200  4440  LOOP1,    LAS                     /SWITCH ADDRESS
```



```
3201  3206            DCA     SWAD
3202  1206            TAD     SWAD
3203  3606            DCA I   SWAD
3204  1606            TAD I   SWAD
3205  5200            JMP     LOOP1

3206  0000  SWAD,     0

3207  4440  LOOP2,    LAS                     /READ LOWER LIMIT
3210  3233            DCA     FIRST
3211  7402            HLT                     /SET SR FOR UPPER LIMIT
3212  4440            LAS                     /READ UPPER LIMIT
3213  3234            DCA     LAST
3214  1233  LOOP2A,   TAD     FIRST
3215  3235            DCA     SWAD0
3216  1235  LOOP2B,   TAD     SWAD0
3217  3635            DCA I   SWAD0
3220  1635            TAD I   SWAD0
3221  7200            CLA
3222  1235            TAD     SWAD0
3223  7041            CIA
3224  1234            TAD     LAST
3225  7650            SNA CLA
3226  5214            JMP     LOOP2A
3227  2235            ISZ     SWAD0
3230  5216            JMP     LOOP2B
3231  7402            HLT                     /HALT RESULTED IN ILLEGAL LIMITS
3232  5207            JMP     LOOP2

3233  0000  FIRST,    0
3234  0000  LAST,     0
3235  0000  SWAD0,    0


      3400            PAGE


3400  1104  PERR1,    TAD     STKTST
3401  7112            CLL RTR
3402  7010            RAR
3403  1136            TAD     K4060
3404  3222            DCA     Z20             /FIELD OF ERROR
3405  4541            JMS I   XSIXTY
3406  0077            TESTAD
3407  3423            Z21                     /FAILING ADDRESS
3410  4541            JMS I   XSIXTY
3411  0106            GDATA
3412  3426            Z22                     /GOOD
3413  4541            JMS I   XSIXTY
3414  0105            BDATA
3415  3431            Z23                     /BAD
3416  1024            TAD     NOTTY           /GET THE TERMINAL FLAG
3417  7640            SZA     CLA             /IS THERE A TERMINAL ON SYSTEM
3420  5271            JMP     ADDERR          /NO-HALT WITH ERROR INFO IN AC
3421  4540            JMS I   XMESAG
```

```
3422  0000    Z20,    0
3423  0000    Z21,    0
3424  0000            0                       /FAILING ADDRESS
3425  4040            4040
3426  0000    Z22,    0
3427  0000            0                       /GOOD
3430  4040            4040
3431  0000    Z23,    0
3432  0000            0                       /BAD
3433  4040            4040
3434  0000    Z24,    0                       /TEST
3435  5544            JMP I   XSTOP


3436  1104    PERRM,  TAD     STKTST
3437  7112            CLL RTR
3440  7010            RAR
3441  1136            TAD     K4060
3442  3262            DCA     Z10
3443  4541            JMS I   XSIXTY
3444  0107            MOVE
3445  3463            Z11
3446  1024            TAD     NOTTY           /GET THE TERMINAL FLAG
3447  7640            SZA     CLA             /IS THERE A TERMINAL ON THE SYSTEM
3450  5325            JMP     RELERR          /NO-GO HALT WITH ERROR INFO IN AC
3451  4540            JMS I   XMESAG
3452  2205            TEXT                    "RELO ERR AT "
3453  1417
3454  4005
3455  2222
3456  4001
3457  2440
3460  0000
3461  4540            JMS I   XMESAG
3462  0000    Z10,    0
3463  0000    Z11,    0
3464  0000            0
3465  0000            0
3466  7240            STA
3467  3073            DCA     HEAD1
3470  5544            JMP I   XSTOP

                /ERROR ROUTINE FOR ADDRESS ERRORS ON A SYSTEM WITH NO CONSOLE TERMINAL.
                /ERRORS WILL BE REPORTED BY HALTS WITH ERROR INFO IN THE AC. REFER
                /TO THE COMMENTS AT EACH HALT FOR THE ERROR INFO SUPPLIED.

3471  1777'   ADDERR, TAD     ERROR0          /GET THE PROGRAM FIELD
3472  0121            AND     K7              /MASK TO THE FIELD BITS
3473  7402            HLT                     /AC=FIELD PROGRAM IS LOCATED IN
3474  7200            CLA
3475  1113            TAD     ERRLOC
3476  7402            HLT                     /AC=PROGRAM ADDRESS WHERE ERROR JMS OCCURED
3477  7200            CLA
3500  1222            TAD     Z20             /PICK UP THE TEST FIELD
```



```
3501  0121            AND     K7              /MASK TO THE FIELD BITS
3502  7402            HLT                     /AC=FIELD CONTAINING THE ERROR
3503  7200            CLA
3504  1077            TAD     TESTAD
3505  7402            HLT                     /AC=ADDRESS OF LOCATION IN ERROR
3506  7200            CLA
3507  1106            TAD     GDATA
3510  7402            HLT                     /AC= WHAT THE DATA SHOULD BE
3511  7200            CLA
3512  1105            TAD     BDATA           /
3513  7402            HLT                     /AC=WHAT THE DATA WAS
3514  7200            CLA
3515  1234            TAD     Z24             /GET THE TEST NUMBER THAT FAILED
3516  7112            CLL     RTR             /PUT TEST # INTO BITS 9-11
3517  7012            RTR
3520  7012            RTR
3521  0121            AND     K7              /MASK TO FIELD BITS
3522  7402            HLT                     /AC=TEST # (1-4) OF THE TEST RUNNING
                                              /WHEN FAILURE OCCURED
3523  7200            CLA
3524  5544            JMP  I  XSTOP           /CONTINUE TEST-GO CHECK SR0

                /ERROR ROUTINE FOR RELOCATION ERRORS. ERROR INFO WILL BE PROVIDED BY
                /ERROR HALTS WITH THE ERROR INFORMATION IN THE AC. REFER TO THE
                /COMMENTS AT EACH HALT FOR ERROR INFORMATION SUPPLIED.

3525  1777'   RELERR, TAD     ERROR0          /GET THE PROGRAM FIELD
3526  0121            AND     K7              /MASK TO THE FIELD BITS
3527  7402            HLT                     /AC=FIELD PROGRAM IS LOCATED IN
3530  7200            CLA
3531  1113            TAD     ERRLOC
3532  7402            HLT                     /AC=PROGRAM ADDRESS WHERE ERROR JMS OCCURED
3533  7200            CLA
3534  1262            TAD     Z10
3535  0121            AND     K7
3536  7402            HLT                     /AC=FIELD PROGRAM RELOCATING TO
3537  7200            CLA
3540  1107            TAD     MOVE
3541  7402            HLT                     /AC=ADDRESS OF LOCATION IN ERROR
3542  7200            CLA
3543  5544            JMP  I  XSTOP           /CONTINUE PROGRAM

3577  2242
      3600    PAGE                            /APT/

                /APT/   ROUTINE TO INITIALIZE FOR RUNNING UNDER APT CONTROL
                /C8/    OR UNDER CLASSIC 8 CONTROL.

3600  6002    APTIZ,  IOF                     /APT/
3601  6441            CMKCON                  /WAS CONSOLE ACTIVE
3602  5214            JMP     APTIZ1          /NO = CHECK FOR FIELD LIMITS
3603  3024            DCA     NOTTY           /CONSOLE AVAILABLE-CLEAR FLAG IF SET TO FOOL
3604  7240            CLA     CMA             /SET PASS COUNTER TO ZERO ON 1ST PASS
3605  3777'           DCA     PASCNT          /SAVE PASS COUNTER
3606  1376            TAD     (JMS  I  KSFCHK /SETUP TO LOOK FOR CONSOLE RECEIVE FLAG
```

```
3607  3775'           DCA     APTOK0          /C8/

                /C8/    THE NEXT LOC WILL = NOP AFTER BEING USED ONCE.

3610  4774'           JMS     C8SM            /C8/GO SAVE PG 37 OF FLD 1.
3611  1373            TAD     (7000           /C8/MODIFY ABOVE LOC TO: NOP.
3612  3210            DCA     .-2             /C8/
3613  5302            JMP     APTIZ0          /C8/

3614  1024    APTIZ1, TAD     NOTTY           /GET CONSOLE FLAG
3615  7650            SNA     CLA             /IS THERE A CONSOLE ON THE SYSTEM
3616  5223            JMP     .+5             /YES-GO GET THE FIELD LIMITS
3617  1020            TAD     PSR             /NO-SETUP LIMITS TO 07
3620  0372            AND     (7700           /MASK TO SR BITS
3621  1121            TAD     K7              /ADD FIELD LIMITS
3622  3020            DCA     PSR             /SAVE SR AS XX07
3623  4771'           JMS     APTFL           /APT/GO GET FIELD LIMITS.
3624  7200            CLA                     /APT/
3625  1022            TAD     HCW2            /APT/RUN UNDER APT CONTROL?
3626  7700            SMA CLA                 /APT/SKP IF YES.
3627  5302            JMP     APTIZ0          /APT/
3630  1370            TAD     (JMS I IAPTOK   /SETUP FOR APT CONTROL
3631  3775'           DCA     APTOK0          /APT/
3632  1373            TAD     (7000           /APT/MODIFY SOME LOCS TO: NOP.
3633  3767'           DCA     APTN00          /APT/
3634  1373            TAD     (7000           /APT/
3635  3766'           DCA     APTN01          /APT/
3636  1373            TAD     (7000           /APT/
3637  3765'           DCA     APTN02          /APT/
3640  1373            TAD     (7000           /APT/
3641  3764'           DCA     APTN03          /APT/
3642  1373            TAD     (7000           /APT/
3643  3763'           DCA     APTN04          /APT/
3644  1373            TAD     (7000           /APT/
3645  3762'           DCA     APTN05          /APT/
3646  1373            TAD     (7000           /APT/
3647  3761'           DCA     APTN06          /APT/
3650  1360            TAD     (APTJ50         /APT/MODIFY SOME LOCS TO: JMP .+N.
3651  0357            AND     (177            /APT/
3652  1356            TAD     (5200           /APT/
3653  3755'           DCA     APTJ00          /APT/
3654  1354            TAD     (APTJ51         /APT/
3655  0357            AND     (177            /APT/
3656  1356            TAD     (5200           /APT/
3657  3753'           DCA     APTJ01          /APT/
3660  1352            TAD     (APTJ52         /APT/
3661  0357            AND     (177            /APT/
3662  1356            TAD     (5200           /APT/
3663  3751'           DCA     APTJ02          /APT/
3664  1350            TAD     (APTJ53         /APT/
3665  0357            AND     (177            /APT/
3666  1356            TAD     (5200           /APT/
3667  3747'           DCA     APTJ03          /APT/
3670  1346            TAD     (JMS I IAPTER   /APT/MODIFY SOME LOCS TO: JMS I IAPTER.
3671  3745'           DCA     APTE00          /APT/
```



```
3672  1745'           TAD     APTE00          /APT/
3673  3744'           DCA     APTE01          /APT/
3674  1745'           TAD     APTE00          /APT/
3675  3743'           DCA     APTE02          /APT/
3676  1745'           TAD     APTE00          /APT/
3677  3742'           DCA     APTE03          /APT/
3700  1745'           TAD     APTE00          /APT/
3701  3741'           DCA     APTE04          /APT/

3702  1024    APTIZ0, TAD     NOTTY           /IS THERE A CONSOLE TERMINAL
3703  7650            SNA     CLA             /
3704  5740'           JMP     RUN0            /YES- START PROGRAM
3705  1737'           TAD     MIN50           /NO-SETUP TIME COUNTER
3706  3146            DCA     MIN5            /SAVE THE TIME COUNTER
3707  1336            TAD     (6003           /SETUP TO RUN ALL TESTS
3710  3076            DCA     RUNTST          /SAVE IT
3711  6002            IOF
3712  6224            RIF
3713  1335            TAD     (6201
3714  3315            DCA     .+1
3715  6201            CDF     0
3716  5734'           JMP     CHEXA+1         /GO RUN THE TEST


3734  0215
3735  6201
3736  6003
3737  1631
3740  1600
3741  1537
3742  1465
3743  1414
3744  1306
3745  1705
3746  4405
3747  1236
3750  1243
3751  1144
3752  1157
3753  0534
3754  0547
3755  0516
3756  5200
3757  0177
3760  0531
3761  0514
3762  0400
3763  1000
3764  0532
3765  0323
3766  0214
3767  0213
3770  4406
3771  4041
3772  7700
```

```
3773  7000
3774  4127
3775  1201
3776  4004
3777  1261
      4000    PAGE                        /APT/

              /APT/   ROUTINE TO 'NOTIFY' APT THAT THE PROGRAM IS RUNNING OK.

4000  0000    APTOK,  0                   /APT/
4001  6002            IOF                 /APT/
4002  7200            CLA                 /APT/
4003  1222            TAD     APTIMX      /APT/DELAY 100MS.
4004  3224            DCA     APTCTX      /APT/
4005  1223            TAD     APTIMY      /APT/
4006  3225            DCA     APTCTY      /APT/
4007  2225            ISZ     APTCTY      /APT/
4010  5207            JMP     .-1         /APT/
4011  2224            ISZ     APTCTX      /APT/
4012  5205            JMP     .-5         /APT/
4013  6224            RIF                 /APT/AC=IF.
4014  1377            TAD     (6201       /APT/CREATE A CDF INST.
4015  3216            DCA     .+1         /APT/MODIFY NEXT CDF INST.
4016  6201            CDF                 /APT/(MODIFIED CDF) DF=IF.
4017  6272            CIF     70          /APT/IF=FIELD 7.
4020  4776'           JMS     6500        /APT/CALL APT - 'PROG OK'.
4021  5600            JMP I   APTOK       /APT/RTN FROM APT - RTN TO CALL+1.

4022  7771    APTIMX, -7                  /APT/
4023  0000    APTIMY, 0                   /APT/
4024  0000    APTCTX, 0                   /APT/
4025  0000    APTCTY, 0                   /APT/


              /APT/   ROUTINE TO HANDLE ERRORS UNDER APT CONTROL.

4026  0000    APTER,  0                   /APT/
4027  6002            IOF                 /APT/
4030  7200            CLA                 /APT/
4031  6224            RIF                 /APT/AC=IF.
4032  1377            TAD     (6201       /APT/CREATE A CDF INST.
4033  3236            DCA     .+3         /APT/MODIFY NEXT CDF INST.
4034  7240            CLA CMA             /APT/
4035  1226            TAD     APTER       /APT/AC=ERROR PC.
4036  6201            CDF                 /APT/(MODIFIED CDF) DF=IF.
4037  6272            CIF     70          /APT/IF=FIELD 7.
4040  5775'           JMP     6520        /APT/CALL APT - 'ERROR'.

              /APT/   ROUTINE TO GET THE SPECIFICATIONS OF THE FIELDS TO BE TESTED
              /APT/   (FIELD LIMITS).  FIRST ATTEMPT IS FROM HCW1.  IF THERE IS
              /APT/   NO SPEC THERE THEN SPECS WILL BE TAKEN FROM THE SWITCH REGISTER
              /APT/   (PSR OR HARD SR).  IN EITHER CASE THE RESULT IS PRESERVED
              /APT/   IN THE PSR.  IF FIELD 7 IS SPECIFIED THEN FIELD 6 IS FORCED.
```



```
4041  0000    APTFL,  0                   /APT/
4042  7200            CLA                 /APT/
4043  1021            TAD     HCW1        /APT/GET MEM SIZE FROM HCW1.
4044  0374            AND     (37         /APT/
4045  7012            RTR                 /APT/CONVERT TO HIGHEST FIELD LIMIT.
4046  0121            AND     K7          /APT/
4047  7450            SNA                 /APT/SKP IF VALID MEM SIZE WAS IN HCW1.
4050  5270            JMP     APTFL0      /APT/GO TRY SW REG FOR FIELD LIMITS.
4051  3224            DCA     APTCTX      /APT/FIELD 7 SPEC'D?
4052  1022            TAD     HCW2        /GET HARDWARE CONTROL WORD 2
4053  7700            SMA     CLA         /WAS APT SELECTED?
4054  5261            JMP     .+5         /NO=DO NOT CHECK FOR FIELD 7
4055  1224            TAD     APTCTX      /APT/
4056  1373            TAD     (7771       /APT/
4057  7650            SNA CLA             /APT/SKP IF NO.
4060  7040            CMA                 /APT/FORCE FLD 6 AS HIGHEST FLD LIMIT.
4061  1224            TAD     APTCTX      /APT/
4062  3224            DCA     APTCTX      /APT/
4063  1020            TAD     PSR         /APT/
4064  0372            AND     (7700       /APT/
4065  1224            TAD     APTCTX      /APT/
4066  3020            DCA     PSR         /APT/PSR=FIELD LIMITS.
4067  5641            JMP I   APTFL       /APT/RTN TO CALL+1.

4070  7200    APTFL0, CLA                 /APT/
4071  4440            GETSR               /GET FIELD LIMITS FROM SWITCH REGISTER
4072  0371            AND     (77         /APT/
4073  3224            DCA     APTCTX      /APT/SEPARATE & TEMP STORE LO & HI LIMITS.
4074  1224            TAD     APTCTX      /APT/
4075  0370            AND     (7          /APT/
4076  3225            DCA     APTCTY      /APT/TEMP STORE HI LIMIT.
4077  1224            TAD     APTCTX      /APT/
4100  0367            AND     (70         /APT/
4101  3224            DCA     APTCTX      /APT/TEMP STORE LO LIMIT.
4102  1022            TAD     HCW2        /GET HARDWARE WORD 2
4103  7700            SMA     CLA         /WAS APT SELECTED?
4104  5321            JMP     NOTAPT      /NO=DO NOT CHECK FOR FIELD 7
4105  1224            TAD     APTCTX      /APT/FLD 7 SPEC'D AS LO LIMIT?
4106  1366            TAD     (7710       /APT/
4107  7640            SZA CLA             /APT/SKP IF YES.
4110  5313            JMP     .+3         /APT/
4111  1365            TAD     (60         /APT/FORCE FLD 6 AS LO LIMIT.
4112  3224            DCA     APTCTX      /APT/
4113  1225            TAD     APTCTY      /APT/FLD 7 SPEC'D AS HI LIMIT?
4114  1373            TAD     (7771       /APT/
4115  7640            SZA CLA             /APT/SKP IF YES.
4116  5321            JMP     .+3         /APT/
4117  1364            TAD     (6          /APT/FORCE FLD 6 AS HI LIMIT.
4120  3225            DCA     APTCTY      /APT/
4121  1020    NOTAPT, TAD     PSR         /APT/
4122  0372            AND     (7700       /APT/
4123  1224            TAD     APTCTX      /APT/
4124  1225            TAD     APTCTY      /APT/
4125  3020            DCA     PSR         /APT/PSR=FIELD LIMITS.
```

```
4126  5641          JMP I   APTFL            /APT/RTN TO CALL+1.

                    /C8/    ROUTINE  TO SAVE PAGE 37 OF FIELD 1

4127  0000    C8SM,   0
4130  7200            CLA
4131  6224            RIF                     /READ THE INSTRUCTION FIELD
4132  1377            TAD     (6201           /ADD CDF 0 TO IT
4133  3342            DCA     C8SM0           /MODIFY THE CDF INSTR AT LOC C8SM0
4134  1363            TAD     (7577           /SET UP PAGE 37 POINTER -1
4135  3010            DCA     10              /SAVE IN AUTO INDEX 10
4136  1362            TAD     (C8SA-1         /GET ADDRESS -1 OF STORAGE AREA
4137  3011            DCA     11              /SAVE IN AUTO INDEX 11
4140  6211    C8SM1,  CDF     10              /CHANGE DATA FIELD TO 1
4141  1410            TAD  I  10              /GET THE WORD
4142  6201    C8SM0,  CDF                     /CHANGE DATA FIELD TO PRG FIELD
4143  3411            DCA  I  11              /SAVE IN STORE AREA
4144  1010            TAD     10              /CHECK TO SEE IF PAGE DONE
4145  7040            CMA
4146  7640            SZA     CLA             /DONE SAVING PAGE
4147  5340            JMP     C8SM1           /NO=DO NEXT WORD
4150  5727            JMP  I  C8SM            /YES=RETURN TO CALL+1

4151  4323    SRMESG, TEXT                    "#SR="
4152  2275
4153  0000
4154  7743    QESTMK, TEXT                    "?#"
4155  0000
4156  3603    UPARRC, TEXT                    "^C#"
4157  4300
4160  3607    UPARRG, TEXT                    "^G#"
4161  4300

4162  4777
4163  7577
4164  0006
4165  0060
4166  7710
4167  0070
4170  0007
4171  0077
4172  7700
4173  7771
4174  0037
4175  6520
4176  6500
4177  6201
      4200    PAGE                            /APT/

                    /C8/    ROUTINE TO RESTORE PAGES 37 OF FIELD 0 AND 1

4200  7200    C8RM,   CLA
4201  4426            PRNTMS                  /GO PRINT UPARROW C
4202  4156            UPARRC                  /POINTER TO MESSAGE
```



```
4203  6224            RIF                     /GET THE PRESENT DATA  FIELD
4204  1377            TAD     (6201           /GET THE CDF INSTRUCTION
4205  3216            DCA     C8RM0           /SAVE THE NEW CDF INSTRUCTION
4206  1376            TAD     (7577           /SET UP AUTO INDEX FOR RESTORE OF 0
4207  3010            DCA     10              /SAVE IN AUTO INDEX 10
4210  1375            TAD     (C8SA-1         /SETUP STORAGE  POINTER
4211  3011            DCA     11              /SAVE IN AUTO INDEX 11
4212  1376            TAD     (7577           /SEUP AUTO INDEX OF RESTORE OF FILD 1
4213  3012            DCA     12              /SAVE IN AUTO INDEX 12
4214  1376            TAD     (7577           /SETUP NEXT POINTER
4215  3013            DCA     13              /SAVE IN AUTO INDEX 13
4216  6201    C8RM0,  CDF                     /MODIFIED CDF INSTRUCTION TO PRG FIELD
4217  1010            TAD     10              /RESTORATION DONE
4220  7040            CMA
4221  7450            SNA                     /SKIP IF NO
4222  5235            JMP     C8RM1           /DONE-GO TO MONITOR AT 7600
4223  7621            7621                    /CLEAR AC AND MQ
4224  1410            TAD  I  10              /GET DATA FROM PROGRAM FIELD
4225  7421            7421                    /PUT IT IN THE MQ
4226  1411            TAD  I  11              /GET DATA TO BE PUT IN FIELD 1
4227  6211            CDF     10              /CHANGE DATA FIELD TO 1
4230  3413            DCA  I  13              /PUT IT IN FIELD 1
4231  7521            7521                    /SWAP AC AND MQ
4232  6201            CDF     00              /CHANGE DATA FIELD TO 0
4233  3412            DCA  I  12              /RESTORE FIELD 0 PAGE 37
4234  5216            JMP     C8RM0           /GO DO NEXT WORD
4235  6203    C8RM1,  CDF     CIF             /CHANGE DATA AND INSTR FIELD TO 0
4236  5637            JMP  I  .+1             /GO TO 7600 OF THAT FIELD
4237  7600            7600                    /MONITOR STARTING ADDRESS

                    /ROUTINE TO GET VALUE OF SWITCH REGISTER USED

4240  0000    SRGET,  0
4241  7300            CLA     CLL
4242  1021            TAD     HCW1            /GET HARDWARE CONTROL WORD 1
4243  7710            SPA     CLA             /PSEUDO OR HARDWARE SWITCH REGISTER
4244  7614            7614                    /USE HARDWARE
4245  1020            TAD     PSR             /USE PSEUDO SWITCH REGISTER
4246  5640            JMP  I  SRGET           /RETURN WITH SR VALUE IN AC

                    /ROUTINE USED FOR CONSOLE SWITCH REGISTER CHANGES

4247  0000    PSEUDO, 0
4250  7200            CLA
4251  4441            CHKCON                  /WAS CONSOLE ACTIVE
4252  5647            JMP  I  PSEUDO          /NO=RETURN BACK TO PROGRAM
4253  4426    SRQEST, PRNTMS                  /PRINT SR QUESTION
4254  4151            SRMESG                  /POINTER TO MESSAGE
4255  4440            GETSR                   /GET THE VALUE OF THE SWITCH REGISTER
4256  4434            PRNT4                   /PRINT THE 4 DIGITS
4257  7346            CLA CLL CMA RTL         /SETUP A COUNTER TO ACCEPT 4 DIGITS
4260  3333            DCA     TTYCNT          /SAVE THE COUNTER
4261  1374            TAD     (CHARR0         /GET POINTER FOR FIRST CHARACTER
4262  3265            DCA     CHGCHR          /SAVE THE POINTER FOR DIGITS
4263  4425            LISN                    /WAIT FOR KEYBOARD INPUT
```

```
4264  0001            1                       /CHECK FOR A OCTAL DIGIT
4265  4303    CHGCHR, CHARR0                  /THIS LOCATION WILL GET MODIFIED
4266  7566            -212                    /CHECK FOR LINE FEED
4267  0200            START                   /LINE FEED TYPED- RETURN TO START
4270  7563            -215                    /CHECK FOR CARRAIGE RETURN
4271  4317            RETYPE                  /RETYPE SR AND CONT IF DIGITS TYPED
4272  7575            -203                    /CHECK FOR A CONTROL C
4273  4200            C8RM                    /CONTROL C TYPED -RETURN TO MONITOR
4274  7555            -223                    /CHECK FOR A CONTROL S
4275  4364            CNTRS                   /WAS CONTROL S WAIT FOR ^Q OR ^C
4276  0000            0                       /NONE OF ABOVE CHARACTERS-ILLEGAL CHAR
4277  4300            .+1                     /GO TO NEXT ADDRESS TO PRINT ?
4300  4426            PRNTMS                  /GO PRINT ?
4301  4154            QESTMK                  /POINTER TO ? MESSAGE
4302  5253            JMP     SRQEST          /RETURN AND ASK QUESTION AGAIN
4303  3020    CHARR0, DCA     PSR             /SAVE THE LEAST SIGNIFICANT BIT
4304  1373            TAD     (CHARR1         /UPDATA POINTER FOR CHARACTERS 2 3 4
4305  3265            DCA     CHGCHR          /SAVE THE POINTER ADDRESS
4306  5263            JMP     CHGCHR-2        /RETURN FOR NEXT CHARACTER INPUT
4307  3332    CHARR1, DCA     SAVCHR          /SAVE THE CHARACTER TYPED
4310  1020            TAD     PSR             /GET THE VALUE OF SR
4311  7106            CLL     RTL             /MOVE IT INTO NEXT POSITION
4312  7004            RAL
4313  1332            TAD     SAVCHR          /ADD NEW CHARACTER TO IT
4314  3020            DCA     PSR             /SAVE THE NEW VALUE
4315  2333            ISZ     TTYCNT          /DONE ALL 4 CHARACTERS
4316  5263            JMP     CHGCHR-2        /NO GET NEXT INPUT FROM KEYBOARD
4317  1374    RETYPE, TAD     (CHARR0         /GET POINTER TO SEE IF SR ECHOED
4320  7041            CIA                     /NEGATE THE POINTER
4321  1265            TAD     CHGCHR          /GET THE POINTER STORED
4322  7650            SNA     CLA             /ECHO VALUE OF SR?
4323  5647            JMP  I  PSEUDO          /NO-ONLY CR WAS TYPED-USE ORIGINAL VALUE
4324  4426            PRNTMS                  /RE-ECHO VALUE TYPED
4325  4151            SRMESG                  /POINTER TO SR MESSAGE
4326  4440            GETSR                   /GET VALUE OF SR
4327  4434            PRNT4                   /PRINT THE 4 OCTAL DIGITS
4330  4437            CRLF                    /ISSUE A CR AND LF
4331  5647            JMP  I  PSEUDO          /RETURN TO PROGRAM

4332  0000    SAVCHR, 0
4333  0000    TTYCNT, 0


                /ROUTINE TO CHECK FOR CONSOLE RECEIVE FLAG

4334  0000    CHKKSF, 0
4335  6031            KSF                     /SKIP  ON CONSOLE RECEIVE FLAG
4336  5734            JMP  I  CHKKSF          /RECEIVE FLAG NOT SET RETURN TO PROGRAM
4337  4441            CHKCON                  /CHECK TO SEE IF CONSOLE WAS ACTIVE
4340  7410            SKP                     /NOT ACTIVE-CLEAR FLAG AND RETURN
4341  5344            JMP     .+3             /ACTIVE CONSOLE-CHECK FOR ^C OR ^G
4342  6032            KCC                     /CLEAR CONSOLE RECEIVE FLAG
4343  5734            JMP  I  CHKKSF          /RETURN TO PROGRAM
4344  4425            LISN                    /CHECK THE KEYBOARD CHARACTER
4345  7575            -203                    /CODE FOR ^C
```



```
4346  4200            C8RM                    /WAS A CONTROL C-EXIT TO MONITOR
4347  7571            -207                    /CODE FOR ^G
4350  4360            CNTRLG                  /WAS  ^G ECHO CHAR-ENTER SR QUESTION
4351  7555            -223                    /CHECK FOR A CONTROL S
4352  4366            CNTRS1                  /WAS A CONTROL S WAIT FOR ^Q OR ^C
4353  0000            0                       /CHAR WAS NOT ^C OR ^ G
4354  4355            .+1                     /ECHO CHAR AND QUESTION MARK
4355  4426            PRNTMS                  /PRINT ? AND CR  LF
4356  4154            QESTMK                  /POINTER TO MESSAGE
4357  5734            JMP  I  CHKKSF          /RETURN TO PROGRAM

4360  4426    CNTRLG, PRNTMS                  /PRINT ^G AND CR LF
4361  4160            UPARRG                  /POINTER  TO MESSAGE
4362  4247            JMS     PSEUDO          /GO ASK THE SR QUESTION
4363  5734            JMP  I  CHKKSF          /RETURN TO THE PROGRAM

4364  4772'   CNTRS,  JMS     WAITQC          /GO WAIT FOR A CONTROL Q OR C
4365  5263            JMP     CHGCHR-2        /GO WAIT FOR NEXT CHAR

4366  4772'   CNTRS1, JMS     WAITQC          /WAIT FOR A CONTROL Q OR C
4367  5734            JMP  I  CHKKSF          /RETURN TO PROGRAM


4372  4710
4373  4307
4374  4303
4375  4777
4376  7577
4377  6201
      4400            PAGE


4400  0000    FILLER, 0                       /SET TO NUMBER OF FILLERS REQUIRED

                /INPUT ONE OCTAL NUMBER TO AC 9 THRU 11
                /GOOD RETURN IS JMS+2

4401  0000    ONEOCK, 0                       /CALL BY "ONEOCT"
4402  4425            LISN
4403  0001            1
4404  4407            .+3
4405  0000            0
4406  4410            .+2
4407  2201            ISZ     ONEOCK
4410  5601            JMP I   ONEOCK

                /INPUT TWO OCTAL NUMBERS TO AC 6 THRU 11
                /GOOD RETURN IS JMS+2

4411  0000    TWOOCK, 0                       /CALL BY "TWOOCT"
4412  4201            JMS     ONEOCK
4413  5611            JMP I   TWOOCK
4414  7104            CLL RAL
4415  7006            RTL
```

```
4416  3224            DCA     XPRNT2
4417  4201            JMS     ONEOCK
4420  5611            JMP I   TWOOCK
4421  1224            TAD     XPRNT2
4422  2211            ISZ     TWOOCK
4423  5611            JMP I   TWOOCK

                /PRINT THE TWO OCTAL NUMBERS IN THE AC 6 THRU 11

4424  0000    XPRNT2, 0                       /CALL BY "PRNT2"
4425  3211            DCA     TWOOCK
4426  1211            TAD     TWOOCK
4427  7012            RTR
4430  7010            RAR
4431  4432            PRNT1
4432  1211            TAD     TWOOCK
4433  4432            PRNT1
4434  5624            JMP I   XPRNT2

                /TYPE THE ASCII CHARACTER IN THE AC

4435  0000    XTYPE,  0                       /CALL BY "TYPE"
4436  3251            DCA     CHAR            /SAVE THE CHARACTER
4437  4441            CHKCON                  /CHECK FOR A ACTIVE CONSOLE
4440  7610            SKP     CLA             /CONSOLE INACTIVE-TYPE THE CHARACTER
4441  4777'           JMS     CNTRLS          /CONSOLE ACTIVE-CHECK FOR CONTROL S
4442  1251            TAD     CHAR            /GET THE CHARACTER SAVED AND PRINT
4443  6046            TLS
4444  7200            CLA
4445  6041            TSF
4446  5245            JMP     .-1
4447  6042            TCF
4450  5635            JMP I   XTYPE

4451  0000    CHAR,   0


                /TYPE A CR AND LF WITH NUMBER OF FILLERS
                /AS DETERMINED BY LOCATION "FILLER"

4452  0000    XCRLF,  0                       /CALL BY "CRLF"
4453  7200            CLA
4454  1134            TAD     K215
4455  4436            TYPE
4456  1200            TAD     FILLER
4457  7040            CMA
4460  3266            DCA     XORS
4461  1133            TAD     K212
4462  4436            TYPE
4463  2266            ISZ     XORS
4464  5262            JMP     .+2
4465  5652            JMP I   XCRLF

4466  0000    XORS,   0
```



```
                /PRINT 2 SPACES

4467  0000    SPACX2, 0       /CALL BY "SPACE2"
4470  4426            PRNTMS
4471  4473            .+2
4472  5667            JMP I SPACX2
4473  4040            4040
4474  0010            0010    /USED BY LISN


                /COMPARE INPUT TO LIST FOLLOWING CALL
                /INPUT ONE CHARACTER IF AC=0
                /USE LAST INPUT IF AC NON ZERO

4475  0000    XLISN,  0                       /CALL BY "LISN"
4476  7640            SZA CLA
4477  5325            JMP     LISN1           /USE LAST INPUT SINCE AC NOT ZERO
4500  6031            KSF
4501  5300            JMP     .-1
4502  6036            KRB
4503  0357            AND     K177
4504  1360            TAD     K200
4505  3267            DCA     SPACX2
4506  1267            TAD     SPACX2
4507  1361            TAD     M212
4510  7450            SNA                     /IS IT A LF?
4511  5315            JMP     .+4             /YES
4512  1116            TAD     M3
4513  7640            SZA CLA                 /IS IT A CR?
4514  5317            JMP     .+3             /NO
4515  4437            CRLF
4516  5325            JMP     LISN1
4517  1267            TAD     SPACX2          /GET THE CHAR
4520  1376            TAD     (-223           /CHECK FOR A CONTROL S
4521  7650            SNA     CLA             /WAS IT A CONTROL S
4522  5325            JMP     LISN1           /YES-DO NOT ECHO CHARACTER
4523  1267            TAD     SPACX2
4524  4436            TYPE                    /PRINT THE CHARACTER
4525  1675    LISN1,  TAD I   XLISN           /GET COMPARE VALUE
4526  2275            ISZ     XLISN
4527  7450            SNA                     /EXIT?
4530  5336            JMP     LISN3           /YES
4531  7500            SMA
4532  5346            JMP     LISNUM          /LOOK FOR OCTAL NUMBER
4533  1267            TAD     SPACX2          /COMPARE
4534  7640            SZA CLA                 /EQUAL?
4535  5343            JMP     LISN2           /NO
4536  3266    LISN3,  DCA     XORS
4537  1675            TAD I   XLISN
4540  3275            DCA     XLISN
4541  1266            TAD     XORS
4542  5675            JMP I   XLISN           /AC IS ZERO UNLESS OCTAL NUMBER
4543  7200    LISN2,  CLA
4544  2275            ISZ     XLISN
```

```
4545  5325            JMP     LISN1
4546  7200  LISNUM,  CLA                     /LOOK FOR OCTAL NUMBER
4547  1267            TAD     SPACX2
4550  1354            TAD     M270
4551  7500            SMA                     /IS IT LESS THAN 8?
4552  5343            JMP     LISN2           /NO, SO NOT AN OCTAL NUMBER
4553  1122            TAD     K10
4554  7510  M270,     SPA                     /IS IT GREATER THAN ZERO?
4555  5343            JMP     LISN2           /NO, SO NOT A NUMBER
4556  5336            JMP     LISN3
4557  0177  K177,     0177
4560  0200  K200,     0200
4561  7566  M212,     7566


                /ROUTINE TO CHECK TO SEE IF CONSOLE ACTIVE
                /RETURN CALL+1 IF CONSOLE INACTIVE
                /RETURN TO CALL+2 IF CONSOLE ACTIVE

4562  0000  CONCHK, 0
4563  1022            TAD     HCW2            /GET HARDWARE WORD 2
4564  0375            AND     (400            /MASK TO CONSOLE BIT
4565  7650            SNA     CLA             /WAS CONSOLE ACTIVE
4566  5762            JMP  I  CONCHK          /NO RETURN TO CALL PLUS 1
4567  2362            ISZ     CONCHK          /CONSOLE ACTIVE BUMP RETURN
4570  5762            JMP  I  CONCHK          /RETURN TO CALL PLUS 2


4575  0400
4576  7555
4577  4675
      4600  PAGE


                /PRINT PACKED ASCII TEXT TERMINATED BY
                /SIX-BIT 00

4600  0000  MESAGX, 0                       /CALL BY "MESAGE"
4601  7200            CLA
4602  1600            TAD I   MESAGX
4603  3240            DCA     FOROCK
4604  2200            ISZ     MESAGX          /SET UP RETURN
4605  1640            TAD I FOROCK
4606  7012            RTR
4607  7012            RTR
4610  7012            RTR
4611  4216            JMS     MESAGF
4612  1640            TAD I FOROCK
4613  4216            JMS     MESAGF
4614  2240            ISZ     FOROCK
4615  5205            JMP     .-10
4616  0000  MESAGF, 0
4617  0131            AND     K77
4620  7450            SNA                     /TERMINATOR (00)?
```



```
4621  5600            JMP I   MESAGX          /YES
4622  1235            TAD     M43
4623  7450            SNA                     /CRLF?
4624  5233            JMP     .+7             /YES
4625  1236            TAD     K3
4626  7510            SPA                     /200 OR 300
4627  1237            TAD     K100            /300
4630  1135            TAD     K240            /200
4631  4436            TYPE
4632  5616            JMP I MESAGF
4633  4437            CRLF
4634  5616            JMP I   MESAGF
4635  7735  M43,      7735
4636  0003  K3,       0003
4637  0100  K100,     0100

                /INPUT 4 OCTAL NUMBERS TO AC
                /GOOD RETURN IS CALL+2

4640  0000  FOROCT, 0                       /CALL BY "FOROCT"
4641  4430            TWOOCT
4642  5640            JMP I   FOROCK
4643  7106            CLL RTL
4644  7006            RTL
4645  7006            RTL
4646  3254            DCA     XPRNT4
4647  4430            TWOOCT
4650  5640            JMP I   FOROCK
4651  1254            TAD     XPRNT4
4652  2240            ISZ     FOROCK
4653  5640            JMP I   FOROCK

                /PRINT FOUR OCTAL NUMBERS IN AC 11 THRU 0 FOLLOWED
                /BY TWO SPACES

4654  0000  XPRNT4, 0                       /CALL BY "PRNT4"
4655  3240            DCA     FOROCK
4656  1240            TAD     FOROCK
4657  7012            RTR
4660  7012            RTR
4661  7012            RTR
4662  4433            PRNT2
4663  1240            TAD     FOROCK
4664  4433            PRNT2
4665  4435            SPACE2
4666  5654            JMP I   XPRNT4
                /
                /PRINT THE OCTAL NUMBER IN AC 9 THRU 11
4667  0000  XPRNT1, 0                       /CALL BY "PRNT1"
4670  0121            AND     K7
4671  1274            TAD     K260
4672  4436            TYPE
4673  5667            JMP I   XPRNT1

4674  0260  K260,     260
```

```
                        /ROUTINE TO CHECK FOR A CONTROL S WHILE TYPING OUT MESSAGES
                        /TO EXIT ROUTINE IF A CONTROL S WAS TYPED-A CONTROL Q OR C MUST BE
                        /INPUTTED ON THE KEYBOARD

4675  0000  CNTRLS, 0
4676  6031        KSF                     /SKIP ON CONSOLE KEYBOARD FLAG
4677  5675        JMP  I  CNTRLS          /RETURN TO TYPE ROUTINE-FLAG NOT SET
4700  6034        KRS                     /READ THE CHARACTER STATICALLY
4701  0377        AND     (177            /MASK TO 7 BIT ASCII
4702  1376        TAD     (-23            /CHECK FOR A CONTROL S
4703  7640        SZA     CLA             /WAS IT A CONTROL S
4704  5675        JMP  I  CNTRLS          /NO-RETURN WITH KEYBOARD FLAG STILL SET
4705  6032        KCC                     /CLEAR KEYBOARD FLAG FROM ^S
4706  4310        JMS     WAITQC          /WAIT FOR CONTROL Q OR C
4707  5675        JMP  I  CNTRLS          /RETURN TO PRINT MESSAGE BEING TYPED

4710  0000  WAITQC, 0                     /ROUTINE TO WAIT FOR CONTROL Q OR C
4711  6031        KSF                     /WAIT FOR A CONTROL Q OR C TO EXIT
4712  5311        JMP     .-1             /
4713  6036        KRB                     /READ THE CHARACTER TYPED
4714  0377        AND     (177            /MASK TO 7 BIT ASCII
4715  1375        TAD     (-3             /CHECK FOR A CONTROL C
4716  7450        SNA                     /WAS IT A CONTROL C?
4717  5774'       JMP     C8RM            /YES-RESTORE MONITOR AND RETURN
4720  1373        TAD     (-7             /CHECK FOR A LINE FEED CHARACTER
4721  7450        SNA                     /WAS IT A LINE FEED
4722  5772'       JMP     START           /YES GO RESTART THE PROGRAM
4723  1373        TAD     (-7             /CHECK FOR A CONTROL Q ^Q
4724  7640        SZA     CLA             /WAS IT A CONTROL Q
4725  5311        JMP     WAITQC+1        /NO-WAIT FOR APPROPRIATE KEY
4726  5710        JMP  I  WAITQC          /RETURN TO WHENCE IT CAME


4772  0200
4773  7771
4774  4200
4775  7775
4776  7755
4777  0177
      5000  PAGE

5000  0000  C8SA,  0                     /THIS PAGE USED TO SAVE PG 37 OF FIELD 1

                    $
```



```
0000  11111110  00000000  11101111  11111111  11111111  11111111  11111111  11111111
0100  11111111  11111111  11111111  11111111  11111110  00000000  00000000  00000000

0200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0300  11111111  11111111  11111111  11100000  00000000  00011111  11111111  11111111

0400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0500  11111111  11111111  11111111  11111111  11111111  11000000  00001111  11111111

0600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0700  11111111  11111111  11111111  11111111  11111111  11111111  11100000  00000011


1000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1100  11111111  11111111  11111111  11111111  11111111  11111111  11000001  11111111

1200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1300  11111111  11111111  11111111  11110000  00000000  00000000  00000000  11111111

1400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1500  11111111  11111111  11111111  11111111  11111111  11111111  11111110  00000111

1600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1700  11111111  11111111  11111111  11111111  11111111  11111110  00000000  00000011


2000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2100  11111111  11111111  11111111  11111111  11111111  11111111  11000000  00000111

2200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2300  11111111  11111111  11111111  11111111  10000000  00000000  00000000  00111111

2400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2500  11111111  11111111  11111111  11111110  00000000  00000000  00000000  01111111

2600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2700  11111111  11111111  11111111  11111111  11111111  11100000  00000000  00000111


3000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
3100  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111

3200  11111111  11111111  11111111  11111100  00000000  00000000  00000000  00000000
3300  00000000  00000000  00000000  00000000  00000000  00000000  00000000  00000000

3400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
3500  11111111  11111111  11111111  11111111  11110000  00000000  00000000  00000001

3600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
3700  11111111  11111110  00000000  00001111  11111111  11111111  11111111  11111111
```

```
4000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
4100  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111

4200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
4300  11111111  11111111  11111111  11111111  11111111  11111111  11111111  00111111

4400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
4500  11111111  11111111  11111111  11111111  11111111  11111111  11111111  1000#111

4600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
4700  11111111  11111111  11111110  00000000  00000000  00000000  00000000  00111111


5000  10000000  00000000  00000000  00000000  00000000  00000000  00000000  00000000
5100  00000000  00000000  00000000  00000000  00000000  00000000  00000000  00000000

5200
5300

5400
5500

5600
5700


6000
6100

6200
6300

6400
6500

6600
6700


7000
7100

7200
7300

7400
7500

7600
7700
```



```
ABOVE   1715        C8SM0   4142        CHEXO5  1120        GETSR   4440
ADDER   2250        C8SM1   4140        CHEXO6  1134        GETSRX  0040
ADDER1  1316        CBF0    6203        CHEXO7  1144        GTF     6004
ADDER2  1424        CBF1    6213        CHEXOA  1006        HCW1    0021
ADDER3  1476        CBF2    6223        CHGCHR  4265        HCW2    0022
ADDER4  1550        CBF3    6233        CHKCAC  0041        HEAD1   0073
ADDERR  3471        CBF4    6243        CHKCON  4441        HEAD12  2510
ADDRT1  1307        CBF5    6253        CHKKSF  4334        HIGHST  2731
ADDRT2  1415        CBF6    6263        CHKSW3  1647        IAPTER  0005
ADDRT3  1466        CBF7    6273        CIF0    6202        IAPTOK  0006
ADDRT4  1540        CDF0    6201        CIF1    6212        INSAME  0074
APT000  2052        CDF1    6211        CIF2    6222        K10     0122
APT001  2055        CDF2    6221        CIF3    6232        K100    4637
APTCTX  4024        CDF3    6231        CIF4    6242        K177    4557
APTCTY  4025        CDF4    6241        CIF5    6252        K20     0123
APTE00  1705        CDF5    6251        CIF6    6262        K200    4560
APTE01  1306        CDF6    6261        CIF7    6272        K207    0132
APTE02  1414        CDF7    6271        CINT    6204        K212    0133
APTE03  1465        CHAR    4451        CNTRLG  4360        K215    0134
APTE04  1537        CHARR0  4303        CNTRLS  4675        K240    0135
APTER   4026        CHARR1  4307        CNTRS   4364        K245    2123
APTFL   4041        CHECK   3015        CNTRS1  4366        K260    4674
APTFL0  4070        CHECK0  3016        CNV     2024        K3      4636
APTIMX  4022        CHEXA   0214        CODERR  2201        K30     0124
APTIMY  4023        CHEXB   0253        CONCHK  4562        K340    2124
APTIZ   3600        CHEXC   0262        CONHLT  2400        K40     0125
APTIZ0  3702        CHEXC1  0274        COUNT   0112        K4060   0136
APTII1  3614        CHEXD   0302        CRLF    4437        K50     0126
APTJ00  0516        CHEXD1  0310        CRLFP   0037        K60     0127
APTJ01  0534        CHEXE   0317        CSS     3000        K6003   1636
APTJ02  1144        CHEXE2  0400        CSSB    3010        K6060   2126
APTJ03  1236        CHEXM   0476        CUF     6264        K6201   0137
APTJ50  0531        CHEXM1  0503        DOWN    1745        K6203   1637
APTJ51  0547        CHEXM2  0510        ERR1    2272        K7      0121
APTJ52  1157        CHEXM3  0515        ERR1A   2307        K70     0130
APTJ53  1243        CHEXN   0532        ERR2    2323        K707    2125
APTN00  0213        CHEXN0  0533        ERR3    2441        K77     0131
APTN01  0214        CHEXN1  0600        ERR4    2457        KABOVE  0071
APTN02  0323        CHEXN2  0622        ERRLOC  0113        KBELOW  0072
APTN03  0532        CHEXN3  0640        ERRM    2475        KBINT   0100
APTN04  1000        CHEXN4  0656        ERRMES  2407        KDOWN   0110
APTN05  0400        CHEXN5  0674        ERROR0  2242        KHIGH   2752
APTN06  0514        CHEXN6  0712        ERROR1  2243        KSFCHK  0004
APTOK   4000        CHEXN7  0730        ESL     0102        LAS     4440
APTOK0  1201        CHEXN8  0746        EXTAD0  0205        LAST    3234
BDATA   0105        CHEXN9  0761        FILLER  4400        LEGAL   0410
BELOW   1734        CHEXO   1000        FIRST   3233        LEGAL0  0075
C8RM    4200        CHEXO0  1024        FIVE    1262        LEGALA  0463
C8RM0   4216        CHEXO1  1040        FOROCK  4640        LIMIT   2263
C8RM1   4235        CHEXO2  1054        FOROCP  0031        LI8N    4425
C8SA    5000        CHEXO3  1070        FOROCT  4431        LISN1   4525
C8SM    4127        CHEXO4  1104        GDATA   0106        LISN2   4543
```

```
LI8N3   4536
LI8NUM  4546
LOOP1   3288
LOOP2   3287
LOOP2A  3214
LOOP2B  3216
M1      0114
M2      0115
M212    4561
M270    4554
M3      0116
M34     0120
M4      0117
M43     4635
ME8AGE  2042
ME8AGF  4616
ME8AGP  0026
ME8AGX  4600
MIN5    0146
MIN50   1631
MIN51   1632
MIN52   1633
MIN53   1634
MIN54   1635
MOVE    0107
MSRGHT  2067
MS8L    1751
MTP     2121
NOBELL  2215
NORELO  0070
NO8TK   2152
NOTAPT  4121
NOTTY   0024
ONEOCK  4401
ONEOCP  0027
ONEOCT  4427
PASCNT  1261
PASME8  1263
PERR1   3400
PERRM   3436
PINF    2127
PNOREL  2647
PREL    2706
PREL1   2726
PRNT1   4432
PRNT2   4433
PRNT4   4434
PRNTM8  4426
PSEUDO  4247
PSR     0020
QESTMK  4154
R8ELL   2206
RDF     6214
RELERR  3525
RELO    1655
RELO2   1674
RELO3   1676
RELO4   1701
RELO5   1712
RE8TK   3054
RETURN  2200
RETYPE  4317
RIB     6234
RIF     6224
RMF     6244
RTF     6005
RUN0    1600
RUN1    1605
RUN2    1612
RUN3    1617
RUN4    1624
RUNTST  0076
SAME    1640
SAVCHR  4332
SETSW   2624
SETSW1  2644
SINT    6254
SIXTY   2000
SIXTY0  2037
SIXTY1  2040
SIXTY2  2041
SPACE2  4435
SPACX2  4467
SPCX2P  0035
SRGET   4240
SRME8G  4151
SRQE8T  4253
SSL     0101
STACK0  0050
STACK1  0051
STACK2  0052
STACK3  0053
STACK4  0054
STACK5  0055
STACK6  0056
STACK7  0057
START   0200
STK0    0060
STK1    0061
STK2    0062
STK3    0063
STK4    0064
STK5    0065
STK6    0066
STK7    0067
STKPIN  0103
STKTST  0184
STOP    2251
SUF     6274
SW8     0042
SW1     0043
SW2     0044
SW3     0045
SW4     0046
SW5     0047
SW68    0130
SW911   0121
SWAD    3206
SWAD8   3235
TDF1    1275
TDF2    1482
TDF3    1445
TDF4    1516
TEMP    0111
TEST    1200
TEST1   1273
TEST1A  1276
TEST1B  1302
TEST2   1400
TEST2A  1403
TEST2B  1410
TEST3   1443
TEST3A  1446
TEST3B  1456
TEST4   1514
TEST4A  1517
TEST4B  1530
TESTAD  0077
TITLE   2600
TOSEL   3077
TSTSYS  3017
TTYCNT  4333
TWOOCK  4411
TWOOCP  0030
TWOOCT  4430
TYPE    4436
TYPECH  2070
TYPEP   0036
TYPESP  3047
UPARRC  4156
UPARRG  4160
WAITQC  4710
XADDER  0145
XCODER  0142
XCRLF   4452
XLI8N   4475
XLI8P   0025
XME8AG  0140
XOR8    4466
XPRN1P  0032
XPRN2P  0033
XPRN4P  0034
XPRNT1  4667
XPRNT2  4424
XPRNT4  4654
XRETUR  0143
X8IXTY  0141
XSTOP   0144
XTYPE   4435
Z10     3462
Z11     3463
Z20     3422
Z21     3423
Z22     3426
Z23     3431
Z24     3434
Z8      2702
```



```
ERRORS DETECTED: 0

LINKS GENERATED: 185

RUN-TIME:  13 SECONDS

3K CORE USED
```