

IDENTIFICATION

PRODUCT CODE: MAINDEC-08-DJEXC-B-D
PRODUCT NAME: 4K TO 32K PDP-8A PROCESSOR EXERCISER
PRODUCT DATE: JULY, 1977
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: BRUCE HANSEN

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

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

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

COPYRIGHT (C) 1975, 1977

BY DIGITAL EQUIPMENT CORPORATION

TABLE OF CONTENTS

1.0	ABSTRACT
2.0	REQUIREMENTS
2.1	HARDWARE
2.2	STORAGE
2.3	PREREQUISITE SOFTWARE
3.0	RESTRICTIONS
3.1	HARDWARE RESTRICTIONS
3.2	SOFTWARE RESTRICTIONS
4.0	STANDARD TEST PROCEDURE
4.1	PDP-8A OPTION BOARD #1 (M8316) HARDWARE SETUP
4.2	LOADING THE PROGRAM
4.3	RUN 4K TO 32K PDP-8A PROCESSOR EXERCISER
4.3.1	RUNNING PROGRAM ON NON ACTIVE CONSOLE PACKAGE
4.3.2	RUNNING PROGRAM ON A ACTIVE CONSOLE PACKAGE
4.4	CONSOLE PACKAGE CONTROL CHARACTERS
5.0	ERRORS
5.1	CATASTROPHIC ERRORS
5.2	ERROR HALTS/ERROR PRINTOUTS
5.3	MEMORY REFERENCE INSTRUCTION ERRORS
5.4	MEMORY REFERENCE TEST INSTRUCTION SETUP
5.5	OPERATE INSTRUCTION ERRORS
5.6	OPERATE TEST INSTRUCTION SETUP
5.7	ILLEGAL INTERRUPT ERRORS
5.8	INACTIVE DEVICE ERRORS
5.9	NO INTERRUPT ERRORS
6.0	SWITCH REGISTER SETTINGS
6.1	NORMAL OPERATING SWITCHES
6.2	ERROR RELATED SWITCHES
7.0	REVISIONS
8.0	PROGRAM DESCRIPTION
9.0	FLOW CHARTS
10.0	LISTING

ABSTRACT

MODIFIED TO RUN ON VT78 SYSTEM - APRIL 1977

THE 4K TO 32K PDP-8A PROCESSOR EXERCISER WILL TEST THE EXECUTION OF MEMORY REFERENCE AND OPERATE INSTRUCTIONS IN A 4K TO 32K PDP-8A. ALL TEST INSTRUCTIONS, ADDRESSES, MEMORY, AC, MQ AND LINK DATA ARE GENERATED FROM A RANDOM NUMBER GENERATOR. THE SERIAL LINE UNIT TRANSMITTER AND REAL TIME CLOCK WILL BE TESTED IN INTERRUPT MODE IF THE SYSTEM UNDER TEST IS A PDP-8/A WITH OPTION BOARD #1 (M8316) INSTALLED, OR A VT78.

THE PROGRAM RELOCATES ITSELF A PAGE AT A TIME, UP AND DOWN, WITHIN ANY 4K MEMORY FIELD. IF THE COMPUTER CONTAINS MORE THAN 4K OF MEMORY, THE PROGRAM WILL RELOCATE UP AND DOWN BETWEEN MEMORY FIELDS. AT LEAST 3K OF MEMORY IS REQUIRED IN THE LAST EXTENDED MEMORY FIELD FOR THE PROGRAM TO TEST AND TO RELOCATED INTO IT.

THE PROGRAM IS CAPABLE OF RUNNING ON THE PDP-8A AND VT78 APT TEST LINES.

A CONSOLE PACKAGE HAS BEEN INCLUDED IN 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 MODE MUST BE SELECTED WHEN TESTING A VT78) - THIS ALLOWS THE OPERATOR CONTROL OF THE DIAGNOSTIC THROUGH THE TERMINAL. THE DIAGNOSTIC WILL ASK FOR THE VALUE OF SWITCH REGISTER, BEFORE CONTINUING WITH THE EXECUTION OF THE DIAGNOSTIC. ALL ERRORS, EXCEPT FOR RELOCATION ERRORS WHICH WILL RESULT IN A HALT, WILL BE PRINTED ON THE TERMINAL. THE NUMBER OF PASSES WILL ALSO BE PRINTED ON THE TERMINAL AFTER 4096 PROCESSOR TEST INSTRUCTIONS HAVE BEEN EXECUTED.
2. CONSOLE PACKAGE NOT ACTIVE - THIS WILL RESULT IN THE USE OF HALTS FOR ERRORS, HALT AT END AT PASS IF SELECTED, USE OF HARDWARE OR PSEUDO SWITCH REGISTER, NOT ASKING SWITCH REGISTER QUESTION.

2.0 REQUIREMENTS

2.1 HARDWARE

THE FOLLOWING HARDWARE IS REQUIRED FOR EXECUTION OF THIS PROGRAM:

PROCESSOR: PDP-8A, VT78

MEMORY: 4K TO 32K OF MEMORY IN 4K INCREMENTS

OPTIONS: IF MEMORY SIZE IS GREATER THAN 4K A PDP-8A OPTION BOARD #2 IS REQUIRED. (DOES NOT APPLY TO VT78)

IF THE SERIAL LINE UNIT AND REAL TIME CLOCK ARE TO BE TESTED, A PDP-8A OPTION BOARD #1 AND A TERMINAL ARE REQUIRED. (DOES NOT APPLY TO VT78)

2.2 STORAGE -----

THE PROGRAM INITIALLY OCCUPIES LOCATIONS 0000 TO 0130 AND 0200 TO 5314. THE PROGRAM USES LOCATIONS 0000 TO 0130, AND 5200 TO 5314 FOR PROGRAM INITIALIZATION. ONCE THE PROGRAM HAS BEEN STARTED, THESE LOCATIONS WILL BE DESTROYED. ALL LOCATIONS OUTSIDE THE PROGRAM AREA IN THE PROGRAM FIELD AND ANY OTHER FIELD, IF SELECTED ARE USED AS A TEST AREA. THE TEST AREA IS INITIALLY FILLED WITH HALTS AND REFILLED AFTER EVERY 4096 TEST INSTRUCTIONS HAVE BEEN EXECUTED. IF THE PROGRAM IS ALLOWED TO RELOCATE, IT WILL RELOCATE UP AND BACK A PAGE AT A TIME WITHIN A 4K MEMORY FIELD AND UP AND DOWN BETWEEN MEMORY FIELDS IF MORE THAN 4K. THE PROGRAM INITIALLY MUST BE LOADED INTO FIELD 0.

2.3 PREREQUISITE SOFTWARE -----

FOR PDP-8/A SYSTEM;

PDP-8A CPU TEST

PDP-8A MEMORY TEST

KMB-A OPTION TEST #2 - IF PDP-8A OPTION BOARD #2 IN SYSTEM

DKC8-AA OPTION TEST #1 - IF PDP-8A OPTION BOARD #1 IN SYSTEM

FOR VT78 SYSTEM:

VI78 CPU DIAGNOSTIC

VT78 MOS MEMORY DIAGNOSTIC

3.0 RESTRICTIONS -----

3.1 HARDWARE RESTRICTIONS -----

- A. MINIMUM OF 4K OF MEMORY TO A MAXIMUM OF 32K OF MEMORY IN 4K CONSECUTIVE INCREMENTS.
- B. IF THE PDP-8A OPTION BOARD #1 IS TO BE TESTED BY THE PROGRAM, A TERMINAL MUST BE CONNECTED TO THE SERIAL LINE UNIT, BAUD RATES AND STOP BITS MUST BE SETUP, TTY FILTER MUST BE SET IF ASR/KSR TELETYPE, AND THE REAL TIME CLOCK MUST BE ENABLED.

SOFTWARE RESTRICTIONS

////////////////////////////////////
/WARNING:

/ALL SOFTWARE RESTRICTIONS LISTED BELOW PLUS FIELD AND MEMORY
/SIZE MUST BE ADHERED TO, OTHERWISE, THERE IS NO GUARANTY
/WHAT WILL HAPPEN TO THE PROGRAM.

////////////////////////////////////

- A. ONCE THE PROGRAM HAS BEEN STARTED, THE PROGRAM LOADERS WILL BE DESTROYED.
- B. THE PROGRAM CANNOT BE INITIALIZED TO RUN WITH THE CONSOLE PACKAGE ACTIVE AND TO BE INITIALIZED TO RUN ON THE PDP-8A OR VT78 APT TEST LINE.
- C. BEFORE EACH PROGRAM START, LOCATIONS 0020 AND 0021 IN THE PROGRAM FIELD MUST BE INITIALIZED FOR THE SWITCH REGISTER VALUE AND THE AMOUNT OF MEMORY TO BE TESTED RESPECTIVELY. LOCATION 0020 INITIALLY IS PRESET TO ALL 0'S AND LOCATION 0021 INITIALLY IS PRESET TO 0007 (8K OF MEMORY AND NO FRONT PANEL SWITCH REGISTER).
- D. ONCE THE PROGRAM HAS RELOCATED INTO ANOTHER MEMORY AREA, AND IT IS DESIRED TO CHANGE MEMORY SIZE, MEMORY SIZE CANNOT BE DECREASED BELOW THE 1K SEGMENT THAT THE PROGRAM IS LOCATED IN.
- E. IF THE FRONT PANEL SWITCH REGISTER WAS SELECTED OR IF THE CONSOLE PACKAGE WAS SET TO ACTIVE, ALWAYS STOP THE PROGRAM BY SETTING THE SWITCH REGISTER TO 0400. THIS IS DONE TO INSURE THAT THE PROGRAM IS NOT IN THE PROCESS OF RELOCATING. FOR THOSE SYSTEMS WITHOUT A FRONT PANEL SWITCH REGISTER OR A NON ACTIVE CONSOLE PACKAGE, IT WOULD BE ADVANTAGEOUS TO RELOAD THE PROGRAM.
- F. ONCE RUNNING THE CONSOLE PACKAGE NON-ACTIVE AND NOW DESIRING TO RUN IT ACTIVE, ONE MUST RELOAD THE DIAGNOSTIC AND INITIALIZE FOR A ACTIVE CONSOLE PACKAGE.
- G. RUNNING OF THIS PROGRAM WILL NOT CHECK BOOTSTRAPS AND AUTO RESTARTS ON THE PDP-8A OPTION BOARD #2.
- H. RUNNING OF THIS PROGRAM WILL NOT CHECK THE 12 BIT PARALLEL I/O AND THE SERIAL LINE UNIT RECIEVERS. HOWEVER, THE SERIAL LINE UNIT WILL BE USED FOR INTERAGATION IF CONSOLE PACKAGE IS ACTIVE.

4.0 STANDARD TEST PROCEDURE

IF THE SYSTEM UNDER TEST IS A VT78 GOTO PARAGRAPH 4.2.

IF THE PDP-8A OPTION BOARD #1 IS TO BE TESTED OR IF THE CONSOLE PACKAGE IS TO BE USED, DO PARAGRAPH 4.1, PDP-8A OPTION BOARD #1 HARDWARE SETUP. IF NEITHER OF THE ABOVE IS TO BE USED GO TO PARAGRAPH 4.2, LOADING THE PROGRAM.

4.1 PDP-8A OPTION BOARD #1 (M8316) HARDWARE SETUP

THIS SECTION IS TO BE EXECUTED IF THE M8316 HAS NOT BEEN SETUP FOR THE TERMINALS BAUD RATE, STOP BITS, TTY FILTER, AND IF THE REAL TIME CLOCK HAS NOT BEEN ENABLED. SET THE FOLLOWING SWITCHES ON THE M8316 TO THE DESIRED BAUD RATES AND STOP BITS FROM THE TABLE BELOW AND ALSO SET THE SWITCHES TO ENABLE THE REAL TIME CLOCK.

BAUD ----	RATE ----	S1-1 ----	S1-2 ----	S1-3 ----
*110	BAUD	OFF	OFF	OFF
150	BAUD	OFF	OFF	ON
300	BAUD	OFF	ON	OFF
600	BAUD	OFF	ON	ON
1200	BAUD	ON	OFF	OFF
2400	BAUD	ON	OFF	ON
4800	BAUD	ON	ON	OFF
9600	BAUD	ON	ON	ON

STOP BITS -----	S1-7 ----
1 STOP BIT	ON
2 STOP BITS	OFF

1 STOP BIT	ON
2 STOP BITS	OFF

REAL TIME CLOCK -----	S1-5 ----	S1-6 ----
ENABLED	ON	ON
DISABLED	ON	OFF

ENABLED	ON	ON
DISABLED	ON	OFF

*NOTE: IF THE TERMINAL IS A ASR33/KSR33, SET S1-8 TO ON (TTY FILTER), OTHERWISE SET S1-8 TO OFF.

4.2

LOADING THE PROGRAM

SYSTEM UNDER TEST - PDP-8/A;
LOAD THE PROGRAM INTO FIELD 0 USING THE STANDARD BINARY
LOADER TECHNIQUE.

SYSTEM UNDER TEST - VT78;
FOR VT78 SYSTEMS THIS DIAGNOSTIC IS PROVIDED ALONG WITH A SYSTEM
MONITOR ON FLOPPY MEDIA. TO LOAD THE SYSTEM MONITOR SIMPLY
INSERT THE FLOPPY DISKETTE CONTAINING THE DIAGNOSTIC INTO EITHER
DRIVE RXA0 OR RXA1 AND PRESS THE VT78 START BUTTON. THE SYSTEM WILL
RESPOND ON THE VIDEO DISPLAY WITH A START MESSAGE FOLLOWED BY A
PROMPT CHARACTER, TO CALL AND START THE DIAGNOSTIC
TYPE 'R DJEXCB' FOLLOWED BY THE RETURN KEY. THE SYSTEM WILL RESPOND
WITH A DISPLAY OF THE PROGRAM NAME AND CURRENT PSEUDO
SWITCH REGISTER SETTING AND WAIT FOR USER ACTION.
CONTINUE WITH PARAGRAPH 4.3.2 B - OPTIONS AVAILABLE TO USER.

NOTE: THIS DIAGNOSTIC ON THE FLOPPY DISKETTE HAS BEEN PRE-
INITIALIZED TO RUN ON A VT78 (LOC 22 BIT 2=1) WITH CONSOLE
PACKAGE ACTIVE (LOC 22 BIT 3=1) AND TO EXERCISE SERIAL LINE UNIT #1
AND REAL TIME CLOCK (LOC 21 BITS 0 & 1 =1). THE DIAGNOSTIC
CAN ALSO BE RUN ON A VT78 WITHOUT EXERCISING THE SERIAL LINE
AND THE REAL TIME CLOCK. TO EXECUTE IN THIS MODE THE USER MUST
USE 'ODT' TO CHANGE LOCATION 0021 FROM 30XX (XX IS THE MEMORY SIZE
INDICATOR) TO 00XX. NORMALLY XX IS SET TO 17 (16K OF MEMORY)
FOR VT78 TESTING BUT CAN ALSO BE CHANGED USING ODT TO A
SMALLER NUMBER (REF NOTE IN PARAG 4.3 B)
TO EXERCISE ONLY A PORTION OF THE VT78 MEMORY.
FOR TESTING ON THE VT78 APT TEST LINE LOCATION 0021 MUST BE
SET TO 3013 (FIELD 3 CONTAINS THE APT LOADER/MONITOR)
AND LOCATION 0022 SET TO 5000.

4.3

RUN 4K TO 32K PDP-8A PROCESSOR EXERCISER

THE PROGRAM MUST "ALWAYS" BE INITIALIZED AT ANY PROGRAM
RESTART. ONCE THE PROGRAM IS INITIALIZED TO RUN WITH OR
WITHOUT THE CONSOLE PACKAGE OR TO RUN ON THE APT TEST
LINE, IT CANNOT BE RE-INITIALIZED TO CHANGE THE ABOVE CONDITIONS
UNLESS THE PROGRAM IS RELOADED.

IN ORDER TO FIND, INITIALIZE, AND START THE PROGRAM DO THE
FOLLOWING STEPS.

- A. IF THE PROGRAM WAS JUST LOADED, THE PROGRAM WILL RESIDE
IN THE FIRST 3K OF FIELD 0 AND THE STARTING ADDRESS

WILL BE 0200. IF THIS WAS THE CASE GO TO STEP B TO
INITIALIZE THE PROGRAM. TO FIND THE PROGRAM AND
STARTING ADDRESS OF THE PROGRAM, ONCE THE PROGRAM HAS
BEEN STARTED, DO THE FOLLOWING STEPS.

1. LOAD EXTENDED ADDRESS TO FIELD 0.
2. LOAD ADDRESS TO ADDRESS 0005 AND EXAMINE THAT LOCATION
3. THE CONTENTS OF ADDRESS 0005 WILL CONTAIN THE STARTING ADDRESS OF THE PROGRAM AND THE FIELD THAT THE PROGRAM IS LOCATED IN. THE CONTENTS OF ADDRESS 0005 WILL BE IN A FORM OF SAFO. SA EQUALS THE MOST SIGNIFICANT SIX BITS OF THE STARTING ADDRESS. F EQUALS THE FIELD THAT THE PROGRAM IS LOCATED IN. SA00 WILL BE THE NEW STARTING ADDRESS OF THE PROGRAM.
4. LOAD THE INSTRUCTION AND DATA FIELD TO THE FIELD THAT THE PROGRAM IS LOCATED IN OBTAINED FROM STEP 3 ABOVE.
5. GO TO STEP B.

- B. THE PROGRAM WHEN FIRST LOADED IS INITIALIZED TO RUN WITH A NON ACTIVE CONSOLE PACKAGE, NO HARDWARE SWITCH REGISTER, NO PDP-8A OPTION BOARD #1 TESTING, NOT ON PDP-8A APT TEST LINE, AND MEMORY SIZE OF 8K. TO CHANGE THE INITIAL CONFIGURATION OR IF THE PROGRAM IS TO BE RESTARTED LOAD ADDRESS TO 0021 IN THE PROGRAM FIELD. NOW DO ONE OF THE FOLLOWING STEPS FOR THE TYPE OF INITIALIZATION REQUIRED.

NOTE: XX IN THE FOLLOWING STEPS INDICATE MEMORY SIZE. XX=03 INDICATES A MEMORY SIZE OF 4K. ADDING A ONE TO THE NUMBER IN XX WILL INCREASE MEMORY SIZE BY 1K. XX=07 EQUALS 8K, XX=13 EQUALS 12K, ETC..

1. NON ACTIVE CONSOLE PACKAGE

-
- A. IF THE PSEUDO SWITCH REGISTER IS TO BE USED GO TO SECTION B OF THIS STEP. IF THE HARDWARE SWITCH REGISTER IS TO BE USED, GO TO SECTION C IN THIS STEP.
 - B. DEPOSIT INTO LOCATION 0021, 00XX IF NO PDP-8A OPTION BOARD #1 IS TO BE TESTED, OTHERWISE DEPOSIT 30XX TO TEST THE PDP-8A OPTION BOARD #1. NOW, LOAD ADDRESS TO 0020 AND DEPOSIT INTO THIS LOCATION THE SWITCH REGISTER SETTINGS DESIRED (NORMALLY ALL ZEROS). LOAD ADDRESS TO 0022 AND DEPOSIT ALL ZEROS TO INDICATE A NON ACTIVE CONSOLE PACKAGE. GO TO PARAGRAPH 4.3.1 - RUNNING PROGRAM ON NON ACTIVE CONSOLE PACKAGE.
 - C. DEPOSIT INTO LOCATION 0021, 40XX IF NO PDP-8A OPTION BOARD #1 IS TO BE TESTED, OTHERWISE, DEPOSIT 70XX TO TEST THE PDP-8A OPTION BOARD #1 NOW LOAD ADDRESS TO 0022 AND DEPOSIT ALL ZEROS TO INDICATE A NON ACTIVE CONSOLE PACKAGE. GO TO PARAGRAPH 4.3.1 - RUNNING PROGRAM ON NON ACTIVE CONSOLE PACKAGE.

2. ACTIVE CONSOLE PACKAGE

- A. DEPOSIT INTO LOCATION 0021, 00XX IF NO PDP-8A OPTION BOARD #1 IS TO BE TESTED, OTHERWISE, DEPOSIT 30XX INTO THIS LOCATION TO TEST IT.
- B. LOAD ADDRESS TO 0020 AND DEPOSIT INTO THIS LOCATION THE SWITCH REGISTER SETTINGS DESIRED (NORMALLY ALL ZEROES).
- C. LOAD ADDRESS TO 0022, AND DEPOSIT INTO THIS LOCATION 0400, TO INDICATE A ACTIVE CONSOLE PACKAGE. GO TO PARAGRAPH 4.3.2 - RUNNING PROGRAM ON A ACTIVE CONSOLE PACKAGE.

3. PDP-8A APT TEST LINE

- A. DEPOSIT INTO LOCATION 0021, 00XX IF NO PDP-8A OPTION BOARD #1 IS TO BE TESTED, OTHERWISE, DEPOSIT 30XX INTO THIS LOCATION TO TEST IT.
- B. LOAD ADDRESS TO 0020 AND DEPOSIT ALL ZEROES.
- C. LOAD ADDRESS TO 0022 AND DEPOSIT 4000, TO INDICATE TO THE PROGRAM THAT ITS ON THE APT TEST LINE.
- D. START PROGRAM USING APT SCRIPTS.

4.3.1 RUNNING PROGRAM ON NON ACTIVE CONSOLE PACKAGE

- A. LOAD ADDRESS TO THE STARTING ADDRESS OBTAINED FROM PARAGRAPH 4.3 PART A. PRESS "INIT" AND THEN "RUN" THE PROGRAM SHOULD NOW RUN.
- B. IF THE PROGRAM WAS INITIALIZED TO TEST THE PDP-8A OPTION BOARD #1, A BINARY COUNT PATTERN SHOULD BE TYPING OUT ON THE CONSOLE TERMINAL. THE PROGRAM WILL ALSO BE TESTING FOR REAL TIME CLOCK INTERRUPTS.
- C. TO STOP THE PROGRAM, "ALWAYS" SET THE SWITCH REGISTER OR PSEUDO SWITCH REGISTER WHICHEVER SELECTED TO 0400. FAILURE TO DO THIS MAY DESTROY THE PROGRAM WHEN IT IS RESTARTED. THE PROGRAM WILL HALT AFTER 4096 TEST INSTRUCTIONS HAVE BEEN EXECUTED IF THE SWITCH REGISTER WAS SET TO 0400. RESETTNG THE SWITCH REGISTER AND PRESSING "INIT" AND THEN "RUN" WILL CAUSE THE PROGRAM TO CONTINUE TESTING.
- D. REFER TO SECTION ON ERRORS FOR ANY HALT OTHER THEN
END OF PASS HALT.
- E. RUN THIS PROGRAM FOR 30 MINUTES. A PROGRAM PASS WILL RANGE BETWEEN 5 SECONDS TO 30 SECONDS DEPENDING ON MEMORY SIZE AND CYCLE TIME.
- F. TO RESTART THE PROGRAM, GO TO PARAGRAPH 4.3 - RUN 4K TO 32K PDP-8A PROCESSOR EXERCISER.

4.3.2 RUNNING PROGRAM ON A ACTIVE CONSOLE PACKAGE

-
- A. LOAD ADDRESS TO THE STARTING ADDRESS OBTAINED FROM PARAGRAPH 4.3 PART A. PRESS "INIT" AND THEN "RUN".
 - B. THE PROGRAM WILL NOW TYPE OUT THE PSEUDO SWITCH REGISTER, LOCATION 0020, ON THE CONSOLE TERMINAL AS "SR=0000" AND THEN WAIT FOR THE OPERATOR TO RESPOND. TYPING IN "CP" WILL START RUNNING THE PROGRAM USING THE NUMBER TYPED OUT AS THE SWITCH REGISTER SETTING. TYPING IN 4 OCTAL DIGITS WILL CHANGE THE PSEUDO SWITCH REGISTER AND START RUNNING THE PROGRAM USING THE NUMBER TYPED AS THE NEW SWITCH REGISTER SETTING.
 - C. IF THE PROGRAM WAS INITIALIZED TO TEST THE PDP-8A OPTION BOARD #1 OR IF THE SYSTEM UNDER TEST IS A VT78, A BINARY COUNT PATTERN SHOULD BE TYPING OUT ON THE CONSOLE TERMINAL. THE PROGRAM WILL ALSO BE TESTING THE REAL TIME CLOCK TO INTERRUPT.
 - D. THE PROGRAM WILL NOW RUN UNTIL STOPPED BY THE OPERATOR OR A ERROR IS ENCOUNTERED. A END OF PASS MESSAGE WILL BE TYPED OUT IN APPROXIMATELY ONE MINUTE (2 MIN FOR A VT78). THE NUMBER TYPED OUT WILL BE UPDATED ABOUT EVERY MINUTE (2 MINUTES FOR A VT78) THE END OF PASS TYPEOUT WILL LOOK LIKE THIS "DJEXCB PASS 0001".
 - E. SETTING OF THE PSEUDO SWITCH REGISTER TO 0400 EITHER AT PROGRAM START OR VIA THE CONTROL G CHARACTER WILL CAUSE THE PROGRAM TO TYPEOUT THE END OF PASS MESSAGE FOLLOWED BY THE PSEUDO SWITCH REGISTER. A CARRIAGE RETURN IS NEEDED TO CONTINUE FROM THIS TYPEOUT.
 - F. REFER TO PARAGRAPH 4.4, CONSOLE PACKAGE CONTROL CHARACTERS, FOR THE CONTROL CHARACTERS AND THEIR FUNCTIONS.
 - G. REFER TO THE SECTION ON ERRORS FOR ANY ERROR TYPEOUTS OR HALTS WHILE RUNNING THE PROGRAM.
 - H. RUN THIS PROGRAM FOR APPROXIMATELY 30 MINUTES (1 HOUR ON A VT78).
 - I. TO RESTART THE PROGRAM OR TO RE-INITIALIZE IT, GO TO PARAGRAPH 4.3, RUN 4K TO 32K PDP-8A PROCESSOR EXERCISER OR PARAGRAPH 4.2 FOR A VT78 SYSTEM..

4.4 CONSOLE PACKAGE CONTROL CHARACTERS

CONTROL CHARACTERS ARE USED TO GIVE THE OPERATOR THE ABILITY TO PERFORM THE FOLLOWING FUNCTIONS.

NOTE: THE PROGRAM WILL RESPOND TO THE CONTROL CHARACTERS IN FIVE SECONDS OR LESS.

CONTROL C

THIS WILL RESTORE THE FLOPPY BOOTSTRAP AND THEN REBOOT TO THE FLOPPY MONITOR.

CONTROL L

THIS WILL SWITCH THE TERMINAL MESSAGES FROM THE CONSOLE TO A LINE PRINTER (DEV CODE 66). TO RESTORE MESSAGES ON THE TERMINAL CONTROL L MUST BE TYPED AGAIN. IF NO PRINTER IS AVAILABLE AND CONTROL L IS TYPED THE RESULT WILL BE THAT THE CONSOLE PACKAGE WILL WAIT FOR A CONTROL CHAR. THE CONTROL L WILL OUTPUT TO THE TERMINAL AND THE PROGRAM WILL ATTEMPT TO CONTINUE.

CONTROL O

THIS WILL STOP THE PRINTING OF ERROR MESSAGES, ALL OTHER MESSAGES WILL BE DISPLAYED. TO START PRINTING THE ERROR MESSAGE TYPE CONTROL O AGAIN. BY TYPING CONTROL O, THE ERROR MESSAGE PRINTOUT WILL BE EFFECTED AND THE PROGRAM WILL ATTEMPT TO CONTINUE.

CONTROL S

THIS WILL STOP PROGRAM EXECUTION AND WAIT IN A LOOP FOR A CONTINUE. THE ONLY WAY TO CONTINUE WILL BE TO TYPE A CONTROL Q, G OR C. THIS IS A NON PRINTING CHARACTER.

CONTROL Q

THIS IS TO CONTINUE A PROGRAM AFTER A CONTROL S IS TYPED. THIS IS A NON-PRINTING CHARACTER.

CONTROL G

THIS WILL ALLOW THE PSEUDO SWITCH REGISTER TO BE CHANGED AT ANY TIME THE DIAGNOSTIC IS RUNNING. THIS WILL ALLOW A CHANGE TO THE SWITCH REGISTER TO BE MADE AFTER THE INITIAL SETTING WAS MADE.

TERMINATING CHARACTERS:

CARRIAGE RETURN -- THIS WILL RESTORE THE PSEUDO SWITCH REGISTER WITH A NEW VALUE IF ONE WAS ENTERED OR KEEP THE OLD VALUE IF NO NUMBERS WERE TYPED IN. THE PROGRAM

WILL THEN RETURN TO THE POINT AT WHICH IT WAS INTERRUPTED AND RESUME OPERATION.

LINE FEED -- A LINE FEED WILL RESTORE THE PSEUDO SWITCH SWITCH REGISTER WITH THE NEW VALUE TYPED IN OR IF NO NUMBERS WERE ENTERED RESTORE THE OLD VALUE. THE PROGRAM WILL THEN RETURN TO THE BEGINNING OF THE PROGRAM.

SWITCH REGISTER MESSAGE

THIS MESSAGE IS USED TO SETUP THE PSEUDO SWITCH REGISTER BEFORE PROGRAM EXECUTION TAKES PLACE. THE SWITCH REGISTER IS SETUP WHEN THE FOURTH CHARACTER IS ENTERED OR A CARRIAGE RETURN IS TYPED.

END OF PASS

A INDICATION WILL BE GIVEN WHEN THE DIAGNOSTIC HAS MADE A SUCCESSFUL PASS. THE PRINTOUT WILL INDICATE THE DIAGNOSTIC MAINDEC NUMBER, THE WORD PASS, AND A FOUR DIGIT PASS NUMBER.

5.0

ERRORS

ALL ERRORS DETECTED, EXCEPT FOR CATASTROPHIC ERRORS, WILL RESULT IN A ERROR HALT FOR A NON ACTIVE CONSOLE PACKAGE OR A ERROR PRINTOUT FOR A ACTIVE CONSOLE PACKAGE. A ERROR PRINTOUT FOR A ACTIVE CONSOLE PACKAGE WILL LOOK LIKE THE FOLLOWING:

DJEXCB FAILED PC;AAAA AC;BBBB MQ;CCCC FL;DDDD

DJEXCB	MAINDEC NAME
PC;AAAA	ADDRESS WHERE PROGRAM DETECTED A ERROR
AC;BBBB	ERROR INFORMATION IN THE AC
MQ;CCCC	NOT APPLICABLE TO PROGRAM
FL;DDDD	NOT APPLICABLE TO PROGRAM

THE ABOVE TYPEOUT WILL BE FOLLOWED BY THE SWITCH REG QUESTION IF BIT 0 IN THE PSEUDO SWITCH REGISTER IS A 0. IF BIT 0 WAS SET THE PROGRAM WILL CONTINUE. RESPONDING WITH A CARRIAGE RETURN TO THE PSR QUESTION WILL CONTINUE DIAGNOSTIC FROM ERROR.

CATASTROPHIC ERRORS WILL RESULT IN A ERROR HALT FOR BOTH ACTIVE AND INACTIVE CONSOLE PACKAGE. USING THE ERROR HALT ADDRESS OR THE ERROR PC PRINTOUT ADDRESS ON A ACTIVE CONSOLE PACKAGE, REFER TO THE ADDRESSES IN THE TABLE IN PARAGRAPH 5.1, CATASTROPHIC ERROR HALTS, AND IN PARAGRAPH 5.2, ERROR HALTS/ERROR PRINTOUTS, TO DETERMINE WHAT TYPE OF ERROR WAS DETECTED BY THE PROGRAM. ANY ERRORS HALTS WHICH DO NOT CORRESPOND TO AN ADDRESS LISTED IN THE TABLES IN PARAGRAPH 5.1 AND 5.2 ARE ALSO CATASTROPHIC ERRORS. THESE ERRORS ARE PROBABLY DUE TO EXECUTION OF A INSTRUCTION TO THE WRONG ADDRESS OR FIELD.

5.1

CATASTROPHIC ERROR HALTS

ALL ADDRESSES WHICH CORRESPOND TO A ADDRESS IN THE TABLE BELOW OR A ADDRESS WHICH DOES NOT CORRESPOND TO A ADDRESS

IN THE TABLE IN PARAGRAPH 5.2 ARE CATASTROPHIC ERRORS. THESE ERRORS CANNOT BE RECOVERED FROM AND THE PROGRAM MUST BE RELOADED. THE HEADERS FOR THE ERROR HALTS LISTED BELOW ARE DEFINED AS FOLLOWS:

BAT EMP - BATTERY BECAME EMPTY WHILE RUNNING EXERCISER
 ROL UP - RELOCATION ERROR WHILE RELOCATING UP WITHIN A FIELD
 ROL DWN - RELOCATION ERROR WHILE RELOCATING DOWN WITHIN A FIELD.
 SWP FLD - RELOCATION ERROR WHILE RELOCATING TO ANOTHER FIELD.

BAT EMP	ROL UP	ROL DWN	SWP FLD
-----	-----	-----	-----
3253	0233	5165	0466
3453	0433	5365	0666
3653	0633	5565	1066
4053	1033	5765	1266
4253	1233	6165	1466
4453	1433	6365	1666
4653	1633	6565	2066
5053	2033	6765	2266
5253	2233	7165	2466
5453	2433	7365	2666
5653	2633	7565	3066
6053	3033	7765	3266

5.2

ERROR HALTS/ERROR PRINTOUTS

THE ADDRESSES LISTED BELOW ARE THE ERROR HALT ADDRESSES ON A NON ACTIVE CONSOLE PACKAGE OR THE PC ERROR ADDRESS IN A ERROR PRINTOUT FOR A ACTIVE CONSOLE PACKAGE. REFERENCE THE ADDRESSES IN THE TABLE BELOW TO THE ADDRESS OBTAINED FROM THE ERROR HALT OR ERROR PRINTOUT, AND GO TO THE PARAGRAPH DESCRIBING THE ERROR AND FOR THE ERROR RECOVERY. ANY ERROR HALTS WHICH OCCUR AND DO NOT CORRESPOND TO ANY OF THE ADDRESSES LISTED BELOW OR TO THE TABLE IN PARAGRAPH 5.1 ARE CATASTROPHIC ERRORS. THE HEADERS FOR THE ERROR ADDRESSES LISTED BELOW ARE DEFINED AS FOLLOWS:

MRI ERR - MEMORY REFERENCE INSTRUCTION ERROR (AND-TAD-ISZ-DCA-JMS-JMP)
 OPR ERR - OPERATE INSTRUCTION ERROR
 ILL INT - ILLEGAL INTERRUPT
 INA DEV - INACTIVE DEVICE ERROR
 NO INT - NO INTERRUPTS FROM PDP-8A OPTION BOARD #1

SAFO - SA IS THE STARTING ADDRESS OF THE PROGRAM (SA00) AND F IS DETERMINED BY THE OPERATOR FOR THE FIELD WHICH THE PROGRAM HALTED OR PRINTED OUT IN. SAFO

SHOULD AGREE WITH ADDRESS 0005 IN FIELD 0.

MPI	ERR	OPR	EPR	ILL	INT	INA	DEV	NO	INT	SAFO
1336	1742	3130	3263	3315	02F0					
1536	2142	3330	3453	3515	04F0					
1736	2342	3530	3663	3715	06F0					
2136	2542	3730	4063	4115	10F0					
2336	2742	4130	4263	4315	12F0					
2536	3142	4330	4463	4515	14F0					
2736	3342	4530	4663	4715	16F0					
3136	3542	4730	5063	5115	20F0					
3336	3742	5130	5263	5315	22F0					
3536	4142	5330	5463	5515	24F0					
3736	4342	5530	5663	5715	26F0					
4136	4542	5730	6063	6115	30F0					

5.3 MEMORY REFERENCE INSTRUCTION ERRORS

THE PROGRAM WILL HALT FOR A NON ACTIVE CONSOLE PACKAGE AT ADDRESS XX36 OR THE ERROR PC PRINTOUT ON A ACTIVE CONSOLE PACKAGE WILL BE XX36. THIS ADDRESS WILL BE FOR MEMORY REFERENCE (AND-TAD-ISZ-DCA-JMS-JMP) ERRORS. XX36 WILL BE A COMMON HALT OR A COMMON ERROR PC ADDRESS FOR ALL PRINTOUTS. RECORD THE CONTENTS OF THE AC OF THE HALT OR ERROR PRINTOUT INTO THE FIRST ITEM IN THE TABLE. IF ON A NON ACTIVE CONSOLE PACKAGE PRESS "RUN" TO OBTAIN THE NEXT BIT OF ERROR INFORMATION. IF ON A ACTIVE CONSOLE PACKAGE, TYPE CARRIAGE RETURN TO GET THE NEXT BIT OF INFORMATION FROM THE ERROR PRINTOUT. CONTINUE WITH THE ABOVE SEQUENCE UNTIL EACH ITEM IN THE TABLE BELOW IS FILLED.

MEMORY REFERENCE INSTRUCTION INFORMATION TABLE

HALT #	ADDRESS	CONTENTS OF AC	DESCRIPTION
HALT #1	XX36		FIELD THAT PROGRAM PUT INSTRUCTION IN
HALT #2	XX36		INSTRUCTION RETURNED FROM THIS FIELD AFTER EXECUTION OF INSTRUCTION
HALT #3	XX36		EXPECTED PC RETURN FROM INSTRUCTION
HALT #4	XX36		ACTUAL PC RETURN FROM INSTRUCTION
HALT #5	XX36		ADDRESS WHERE INSTRUCTION WAS PLACED
HALT #6	XX36		TEST INSTRUCTION - THE INSTRUCTION WHICH WAS EXECUTED
HALT #7	XX36		REFERENCE ADDRESS - ADDRESS WHICH THE INSTRUCTION WILL REFERENCE, OR IF THE INSTRUCTION IS INDIRECT, THIS ADDRESS WILL CONTAIN THE INDIRECT ADDRESS.

HALT #8 XX36

INDIRECT ADDRESS - THIS IS THE
INDIRECT ADDRESS WHICH THE TEST
INSTRUCTION WILL REFERENCE. N/A
FOR DIRECT ADDRESSING INSTRUCTIONS.

HALT #9 XX36

INITIAL MEMORY DATE- MEMORY
DATA WHICH IS PUT INTO
REFERENCE ADDRESS OR INDIRECT
ADDRESS IF INSTRUCTION IS
DIRECT OR INDIRECT, N/A FOR
JMP OR JMS INSTRUCTIONS.

HALT#10 XX36

FINAL MEMORY DATA- CONTENTS OF
REFERENCE ADDRESS OR INDIRECT
ADDRESS AFTER EXECUTION OF
INSTRUCTION, FOR A JMP
INSTRUCTION, THIS NUMBER SHOULD
BE EQUAL TO A CIF X, FOR
A JMS INSTRUCTION, THIS NUMBER
SHOULD EQUAL THE INSTRUCTION
ADDRESS (HALT #5) PLUS 1.

HALT#11 XX36

THE CONTENTS OF THE AC
BEFORE THE EXECUTION OF THE INSTRUCTION

HALT#12 XX36

THE CONTENTS OF THE AC AFTER
THE EXECUTION OF THE TEST INSTRUCTION

HALT#13 XX36

THE STATE OF THE LINK, BEFORE
THE EXECUTION OF THE INSTRUCTION

HALT#14 XX36

THE STATE OF THE LINK, AFTER THE
EXECUTION OF THE TEST INSTRUCTION.

HALT#15 XX36

THE CONTENTS OF THE MQ BEFORE
THE TEST INSTRUCTION IS EXECUTED

HALT#16 XX36

THE CONTENT OF THE MQ AFTER
THE EXECUTION OF THE TEST INSTRUCTION.

THIS IS THE END OF THE MEMORY REFERENCE INSTRUCTION ERROR INFORMATION.
REFER TO MEMORY REFERENCE TEST INSTRUCTION SETUP SECTION,
PARAGRAPH 5.4, TO DETERMINE THE TYPE OF ERROR.

TO LOOP ON THIS ERROR ON A NON ACTIVE CONSOLE PACKAGE,
SET THE SWITCH REGISTER TO 7000 AND PRESS ONLY "RUN". THE
PROGRAM IS NOW IN A SCOPE LOOP, LOOPING OF THE FAILING CONDITIONS

TO LOOP ON THIS ERROR ON A ACTIVE CONSOLE PACKAGE, TYPE
CONTROL G AND THEN TYPE IN 7000 FOR THE SWITCH REGISTER VALUE.
THE PROGRAM IS NOW IN A SCOPE LOOP, TYPE CONTROL O TO INHIBIT ERROR
PRINTOUTS.

MEMORY REFERENCE TEST INSTRUCTION SETUP

TO DETERMINE THE TYPE OF ERROR, THE OPERATOR MUST UNDERSTAND THE TEST INSTRUCTION SETUP. THE TEST INSTRUCTION SETUPS ARE BROKEN UP INTO GROUPS WHICH ARE LISTED AND DESCRIBED BELOW.

A. AND'S THROUGH DCA'S DIRECT ADDRESSING MODE

1. INSTRUCTION SETUP IS PUT IN SOME RANDOM FIELD
2. LOCATION 4 OF THIS RANDOM FIELD CONTAINS THE RETURN POINTER TO THE PROGRAM.
3. THE CONTENTS OF THE LINK, AC AND MQ CONTAINS SOME RANDOM NUMBER
4. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS -1
5. INSTRUCTION ADDRESS -1 = CIF TO PROGRAM FIELD.
6. INSTRUCTION ADDRESS = THE TEST INSTRUCTION
7. INSTRUCTION ADDRESS +1 = JMS I 4 - RETURN TO PROGRAM
8. INSTRUCTION ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM
9. REFERENCE ADDRESS = INITIAL MEMORY DATA - LOCATION THE INSTRUCTION WILL EXECUTE.

B. AND'S THROUGH DCA'S INDIRECT ADDRESS MODE

1. INSTRUCTION SETUP IS PUT IN SOME RANDOM FIELD
2. LOCATION 4 OF THIS FIELD CONTAINS THE RETURN POINTER TO THE PROGRAM
3. THE CONTENTS OF THE LINK, AC, AND MQ CONTAINS SOME RANDOM NUMBER
4. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS -1
5. INSTRUCTION ADDRESS -1 = CIF TO PROGRAM FIELD
6. INSTRUCTION ADDRESS = THE TEST INDIRECT INSTRUCTION
7. INSTRUCTION ADDRESS +1 = JMS I 4 - RETURN TO PROGRAM
8. INSTRUCTION ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM
9. REFERENCE ADDRESS = INDIRECT ADDRESS - THE ADDRESS THE INSTRUCTION WILL REFERENCE
10. INDIRECT ADDRESS = INITIAL MEMORY DATA - THE LOCATIONS THE INSTRUCTIONS WILL EXECUTE

C. JMP'S - DIRECT ADDRESSING MODE

1. INSTRUCTION SETUP IS PUT IN SOME RANDOM FIELD
2. LOCATION 4 OF THIS RANDOM FIELD CONTAINS THE RETURN POINTER TO THE PROGRAM.
3. THE CONTENTS OF THE LINK, AC, AND MQ CONTAIN SOME RANDOM NUMBER
4. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS
5. INSTRUCTION ADDRESS = THE TEST JUMP INSTRUCTION
6. REFERENCE ADDRESS = CIF TO PROGRAM FIELD. TEST INSTRUCTION JUMPS TO HERE
7. REFERENCE ADDRESS +1 = JMS I 4 - RETURN TO PROGRAM
8. REFERENCE ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM

D. JMP'S - INDIRECT ADDRESS MODE

1. INSTRUCTION SETUP IS PUT IN SOME RANDOM FIELD
2. LOCATIONS 4 OF THIS RANDOM FIELD CONTAINS THE RETURN POINTER TO THE PROGRAM.
3. THE CONTENTS OF THE LINK, AC, AND MQ CONTAINS SOME RANDOM NUMBER.
4. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS
5. INSTRUCTION ADDRESS = THE TEST JMP INDIRECT INSTRUCTION

6. REFERENCE ADDRESS = CONTAINS THE INDIRECT ADDRESS
7. INDIRECT ADDRESS = CIP TO PROGRAM FIELD
8. INDIRECT ADDRESS +1 = JMS I 4 - RETURN TO PROGRAM
9. INDIRECT ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM

E. JMS'S - DIRECT ADDRESS MODE

1. INSTRUCTION SETUP IS PUT IN SOME RANDOM FIELD
2. LOCATION 4 OF THIS FIELD CONTAINS THE RETURN POINTER TO THE PROGRAM.
3. THE CONTENTS OF THE LINK, AC, AND MQ CONTAINS SOME RANDOM NUMBER.
4. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS
5. INSTRUCTION ADDRESS = THE TEST JMS INSTRUCTION
6. REFERENCE ADDRESS = SHOULD CONTAIN INSTRUCTION ADDRESS +1 AFTER EXECUTION OF TEST INSTRUCTION
7. REFERENCE ADDRESS +1 = CIP TO PROGRAM FIELD
8. REFERENCE ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM
9. REFERENCE ADDRESS +3 = JMS I 4 - RETURN TO PROGRAM

F. JMS'S - INDIRECT ADDRESS MODE

1. INSTRUCTION SETUP IS PUT IN SOME RANDOM FIELD
2. LOCATION 4 OF THIS FIELD CONTAINS THE RETURN POINTER TO PROGRAM.
3. THE CONTENTS OF THE LINK, AC, AND MQ CONTAINS SOME RANDOM NUMBER.
4. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS
5. INSTRUCTION ADDRESS = THE TEST JMS INDIRECT INSTRUCTION
6. REFERENCE ADDRESS = INDIRECT ADDRESS
7. INDIRECT ADDRESS = SHOULD CONTAIN THE INSTRUCTION ADDRESS +1 AFTER EXECUTION OF INSTRUCTION.
8. INDIRECT ADDRESS +1 = CIP TO PROGRAM FIELD
9. INDIRECT ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM
10. INDIRECT ADDRESS +3 = JMS I 4 - RETURN TO PROGRAM

5.5 OPEPATE INSTRUCTION ERRORS

THE PROGRAM WILL HALT FOR A NON ACTIVE CONSOLE PACKAGE AT ADDRESS XX42 OR THE ERPOR PC PRINTOUT ON A ACTIVE CONSOLE PACKAGE WILL BE XX42. THIS ADDRESS WILL BE FOR ALL OPERATE INSTRUCTION ERRORS. XX42 WILL BE A COMMON HALT OR ERROR PC PRINTOUT FOR ALL IDEMS LISTED BELOW IN THE TABLE. RECORD THE CONTENTS OF THE AC OF THE HALT OR ERPOR PRINTOUT INTO THE FIRST IDEM IN THE TABLE. IF ON A NON ACTIVE CONSOLE PACKAGE PRESS "RUN" TO OBTAIN THE NEXT BIT OF ERROR INFORMATION. IF ON A ACTIVE CONSOLE PACKAGE, TYPE CARRIAGE RETURN TO GET THE NFXT BIT OF INFORMATION FROM THE ERROR PRINTOUT. CONTINUE THE SEQUENCE UNTIL EACH IDEM IN THE TABLE BELOW IS FILLED.

OPEPATE INSTRUCTION INFORMATION TABLE

HALT #	ADDRESS	CONTENTS OF AC	DESCRIPTION
-----	-----	-----	-----
HALT #1	XX42		FIELD THAT PROGRAM PUT INSTRUCTION IN
HALT #2	XX42		INSTRUCTION RETURNED FROM THIS FIELD AFTER EXECUTION OF INSTRUCTION
HALT #3	XX42		EXPECTED PC RETURN FROM INSTRUCTION

HALT #4 XX42	ACTUAL PC RETURN FROM INSTRUCTION
HALT #5 XX42	ADDRESS WHERE INSTRUCTION WAS PLACED
HALT #6 XX42	TEST INSTRUCTION - THE INSTRUCTION WHICH WAS EXECUTED
HALT #7 XX42	THE CONTENTS OF THE AC BEFORE THE INSTRUCTION WAS EXECUTED
HALT #8 XX42	THE SIMULATED RESULTS OF THE AC, AS CALCULATED BY THE PROGRAM, OF WHAT THE AC SHOULD BE AFTER THE EXECUTION OF THE TEST OPERATE INSTRUCTION
HALT #9 XX42	THE CONTENTS OF THE AC AFTER THE EXECUTION OF THE TEST INSTRUCTION
HALT#10 XX42	THE CONTENTS OF THE LINK BEFORE THE TEST OPERATE INSTRUCTION WAS EXECUTED.
HALT#11 XX42	THE SIMULATED RESULTS OF THE LINK AFTER THE TEST INSTRUCTION WAS EXECUTED AS CALCULATED BY THE PROGRAM
HALT#12 XX42	THE CONTENT OF THE LINK AFTER THE EXECUTION OF THE TEST INSTRUCTION
HALT#13 XX42	THE CONTENTS OF THE MQ BEFORE THE EXECUTION OF THE TEST INSTRUCTION
HALT#14 XX42	THE SIMULATED RESULTS OF THE MQ, AFTER EXECUTION OF THE TEST OPERATE INSTRUCTION AS CALCULATED BY THE PROGRAM.
HALT#15 XX42	THE CONTENTS OF THE MQ AFTER THE EXECUTION OF THE TEST INSTRUCTION.

THIS IS THE END OF THE OPERATE INSTRUCTION ERROR INFORMATION. ERRORS ENCOUNTERED UNDER THIS SECTION MAY BE DUE TO THE EXECUTION OF THE TEST OPERATE INSTRUCTION OR THE SIMULATION OF TEST INSTRUCTION DONE BY THE PROGRAM. REFER TO PARAGRAPH 5.6 FOR OPERATE TEST INSTRUCTION SETUP.

TO LOOP ON THIS ERROR ON A NON ACTIVE CONSOLE PACKAGE, SET THE SWITCH REGISTER TO 7000 AND PRESS ONLY "RUN". THE PROGRAM IS NOW IN A SCOPE LOOP, LOOPING ON THE FAILING CONDITIONS.

TO LOOP ON THIS ERROR ON A ACTIVE CONSOLE PACKAGE, TYPE CONTROL G AND THEN TYPE IN 7000 FOR THE SWITCH REGISTER VALUE. THE PROGRAM IS NOW IN A SCOPE LOOP. TYPE CONTROL O TO INHIBIT ERROR PRINTOUTS.

5.6

OPERATE TEST INSTRUCTION SETUP

TO DETERMINE THE TYPE OF ERROR, THE OPERATOR MUST UNDERSTAND THE TEST INSTRUCTION SETUP. THE OPERATE TEST INSTRUCTION SETUP IS LISTED BELOW.

- A. BEFORE THE EXECUTION OF THE TEST OPERATE INSTRUCTION, THE PROGRAM SIMULATES THE RESULTS OF THE EXECUTION OF THE TEST OPERATE INSTRUCTION UPON THE LINK, AC, AND MQ
- B. THE INSTRUCTION SETUP IS PLACED IN SOME RANDOM FIELD.
- C. LOCATION 4 OF THIS RANDOM FIELD CONTAINS THE RETURN POINTER TO THE PROGRAM.
- D. THE CONTENTS OF THE LINK, AC AND MQ CONTAINS SOME RANDOM NUMBER.
- E. THE PROGRAM JUMPS TO THE INSTRUCTION ADDRESS -1.
- F. INSTRUCTION ADDRESS -1 = CIF TO PROGRAM FIELD
- G. INSTRUCTION ADDRESS = THE TEST OPERATE INSTRUCTION
- H. INSTRUCTION ADDRESS +1 = JMS I 4 - RETURN TO PROGRAM.
- I. INSTRUCTION ADDRESS +2 = JMS I 4 - RETURN TO PROGRAM.

5.7

ILLEGAL INTERRUPT ERRORS

THE PROGRAM WILL HALT AT ADDRESS XX30 ON A INACTIVE CONSOLE PACKAGE OR XX30 WILL BE THE ERROR PC PRINTOUT ON AN ACTIVE CONSOLE PACKAGE. THIS ERROR ADDRESS SIGNIFIES THAT A INTERRUPT OCCURRED FROM A DEVICE NOT BEING TESTED BY THE EXERCISER. THIS ERROR ALSO MAY BE CAUSED BY A FLAG GETTING CLEARED ON A INTERRUPT, OR A SKIP NOT FAILING TO SKIP ON A FLAG. TO RECOVER FROM THIS ERROR, RESTART THE PROGRAM (PARAGRAPH 4.3) IF THE ERROR STILL EXISTS, USE A SCOPE TO SEE WHAT OTHER FLAG IS SET BESIDES THE FOLLOWING EXPECTED FLAGS, SERIAL LINE UNIT TRANSMIT FLAG OR RECEIVE FLAG AND THE REAL TIME CLOCK FLAG. INTERRUPTS ARE ONLY EXPECTED IF THE OPTION BOARD #1 IS TO BE TESTED OR IF THE CONSOLE PACKAGE IS ACTIVE. AC LOW FLAG IS ALSO EXPECTED ON A POWER FAILURE.

5.8 INACTIVE DEVICE ERROR

THIS ERROR WILL ONLY OCCUR IF THE PROGRAM WAS INITIALIZED TO TEST THE PDP-8A OPTION BOARD #1 OR IF THE SYSTEM UNDER TEST IS A VT78. THE PROGRAM WILL HALT AT ADDRESS XX63 ON A INACTIVE CONSOLE PACKAGE, OR XX63 WILL BE THE ERROR PC PRINTOUT ON A ACTIVE CONSOLE PACKAGE. THIS ERROR WILL OCCUR IF THE SERIAL LINE UNIT'S TRANSMIT FLAG OR THE REAL TIME CLOCK FLAG HAS NOT INTERRUPTED IN A GIVEN AMOUNT OF TIME. WHEN THE PROGRAM IS RUNNING, A BINARY COUNT PATTERN SHOULD BE OUTPUTTED TO THE CONSOLE TERMINAL. IF IT ISN'T THEN THE SERIAL LINE UNIT IS INACTIVE, IF THE BINARY COUNT PATTERN IS OUTPUTTED TO THE CONSOLE, THEN THE REAL TIME CLOCK IS INACTIVE. USE A SCOPE TO TROUBLE SHOOT THIS PROBLEM. TO RECOVER FROM THIS ERROR, RESTART THE PROGRAM .

5.9 NO INTERRUPT ERRORS

THIS ERROR WILL ONLY OCCUR IF THE PROGRAM WAS INITIALIZED TO TEST THE PDP-8A OPTION BOARD #1 OR THE SYSTEM UNDER TEST IS A VT78. THE PROGRAM WILL HALT AT ADDRESS XX15 ON A INACTIVE CONSOLE PACKAGE, OR XX15 WILL BE THE ERROR PC PRINTOUT ON A ACTIVE CONSOLE PACKAGE. THIS ERROR INDICATES THAT NO INTERRUPTS FROM THE SERIAL LINE UNIT'S TRANSMIT FLAG OR REAL TIME CLOCK FLAG HAVE OCCURRED OR THAT THE LAST FLAG EXPECTED FROM EACH DEVICE NEVER INTERRUPTED. TO RECOVER FROM THIS ERROR PRESS "RUN" ON A INACTIVE CONSOLE PACKAGE OR TYPE CARRIAGE RETURN ON A ACTIVE CONSOLE PACKAGE. USE A SCOPE AND LOOK AT THE SERIAL LINE UNIT'S TRANSMIT FLAG AND THE REAL TIME CLOCK.

6.0 SWITCH REGISTER SETTINGS

6.1 NORMAL OPERATING SWITCHES

SR2=1	(1000)	-	INHIBIT PROGRAM RELOCATION
SR3=1	(0400)	-	STOP THE PROGRAM AT THE COMPLETION OF A PROGRAM PASS

6.2 ERROR RELATED SWITCHES

SP0=1	(4000)	-	INHIBIT ERROR HALTS EXCEPT FOR RELOCATION ERRORS
SP1=1	(2000)	-	LOOP ON TEST CONDITIONS FOR MEMORY REFERENCE OR OPERATE INSTRUCTIONS
SP2=1	(1000)	-	INHIBIT PROGRAM RELOCATION

7.0 REVISIONS

REVISION B - ADDITIONS TO ALLOW USE ON VT78 SYSTEMS.

8.0 PROGRAM DESCRIPTION

THE 4K TO 32K PDP-8A PROCESSOR EXERCISER CHECKS THE EXECUTION OF ALL MEMORY REFERENCE AND OPERATE INSTRUCTIONS IN ALL SELECTED FIELDS AND ADDRESSES. ALL INSTRUCTIONS, FIELDS, AND DATA ARE SELECTED FROM A RANDOM NUMBER GENERATOR. THE PROGRAM FILLS MEMORY WITH HALTS AFTER EVERY 4096 TEST INSTRUCTIONS HAVE BEEN EXECUTED. IF THE PROGRAM WAS INITIALIZED TO TEST THE PDP-8A OPTION BOARD #1 OR IF THE SYSTEM UNDER TEST IS A VT78, IT WILL BE TESTED IN INTERRUPT MODE. A BINARY COUNT WILL BE TRANSMITTED ON THE SERIAL LINE UNIT. THE PROGRAM WILL EXPECT INTERRUPTS WHILE RUNNING THE MAIN LINE PROGRAM FROM THE SERIAL LINE UNIT AND REAL TIME CLOCK. AFTER EVERY 4096 TEST INSTRUCTIONS HAVE BEEN EXECUTED, THE PROGRAM RELOCATES ITSELF, A PAGE AT A TIME, "UP AND DOWN" WITHIN A MEMORY FIELD. ONCE THE PROGRAM HAS RELOCATED "UP AND DOWN" WITHIN A SPECIFIC MEMORY FIELD, IT WILL RELOCATE UP INTO THE NEXT FIELD IF MORE THAN 4K OF MEMORY EXISTS AND THE NEXT FIELD CONTAINS AT LEAST 3K OF MEMORY. THIS PROCEDURE WILL CONTINUE UNTIL THE LAST MEMORY FIELD IS ENTERED. THEN THE PROGRAM WILL RELOCATE ITSELF DOWN A FIELD AT A TIME UNTIL FIELD ZERO IS REACHED. THEN THE ENTIRE SEQUENCE IS REPEATED. REFER TO THE PARAGRAPHS BELOW FOR MORE DETAILED INFORMATION.

REFER TO PARAGRAPH 5.4 FOR MEMORY REFERENCE TEST INSTRUCTION SETUP. THE PROGRAM VERIFIES THE EXECUTION OF ALL MEMORY REFERENCE INSTRUCTIONS (AND-TAD-ISZ-DCA-JMS-JMP) FOR THE FOLLOWING:

- A. THE INSTRUCTIONS RETURNED TO THE PROGRAM FROM THE CORRECT FIELD
- B. THE INSTRUCTION RETURNED TO THE PROGRAM FROM THE CORRECT ADDRESS
- C. CORRECT ADDRESSING MODES:
 - 1. DIRECT AND INDIRECT ADDRESSING
 - 2. SAME PAGE AND PAGE 0 ADDRESSING
 - 3. AUTO INDEX ADDRESSING
- D. THE CORRECT MEMORY AND AC DATA AFTER THE EXECUTION OF THE TEST INSTRUCTION.
- E. THE LINK DOESN'T CHANGE FOR THE FOLLOWING INSTRUCTIONS AND, ISZ, DCA, JMS AND JMP
- F. THE MO DOESN'T CHANGE.

REFER TO PARAGRAPH 5.6 FOR OPERATE TEST INSTRUCTION SETUP. THE PROGRAM SIMULATES THE EXECUTION OF THE TEST "OPERATE" INSTRUCTION AND VERIFIES THE HARDWARE EXECUTION OF THAT SAME OPERATE INSTRUCTION FOR THE FOLLOWING:

- A. THE INSTRUCTION RETURNED TO THE PROGRAM FROM THE CORRECT FIELD
- B. THE INSTRUCTION RETURNED TO THE PROGRAM FROM THE CORRECT ADDRESS
- C. AC DATA RETURNED EQUALS THE SIMULATED AC DATA
- D. THE LINK DATA RETURNED EQUALS THE SIMULATED LINK DATA
- E. THE MQ DATA RETURNED EQUALS THE SIMULATED MQ DATA

THE PDP-8A OPTION BOARD #1 IS EXERCISED IN INTERRUPT MODE IF SELECTED. THE PROGRAM WHEN FIRST STARTED AND AFTER EACH PROGRAM RELOCATION SETS UP FOR A BINARY COUNT PATTERN ON THE SERIAL LINE UNIT. THE INTERRUPT ENABLE FLIP-FLOPS ARE THEN SET FOR THE SERIAL LINE UNIT AND REAL TIME CLOCK. A WORD IS THEN TRANSMITTED ON THE SERIAL LINE UNIT. THE PROGRAM NOW TURNS THE INTERRUPT ON AND JUMPS TO THE MAIN PART OF THE PROGRAM TO GENERATE AND TEST MEMORY REFERENCE AND OPERATE INSTRUCTIONS. WHEN A INTERRUPT OCCURS, THE PROGRAM DOES THE FOLLOWING:

- A. SAVE THE AC, LINK AND THE INTERRUPTED PC
- B. THE PROGRAM DOES ONE OF THE FOLLOWING DEPENDING ON THE FLAG SET.
 - 1. SLU XMIT FLAG - CLEARS XMIT FLAG - UPDATE WORD- TRANSMIT NEW WORD. GO TO STEP C.
 - 2. REAL TIME CLOCK FLAG - CLEAR REAL TIME CLOCK FLAG. GO TO STEP C.
- C. THE PROGRAM CHECKS FOR THE SERIAL LINE UNIT AND REAL TIME CLOCK TO BE ACTIVE. IT THEN RESTORES THE LINK, AND AC, ISSUES A RMF INSTRUCTION AND RETURNS TO THE PROGRAM WHERE IT WAS INTERRUPTED FROM.
- D. WHEN THE PROGRAM IS READY TO BE RELOCATED, THE PROGRAM WAITS FOR THE FLAGS AND THEN TURNS THE INTERRUPT OFF.

9.0 FLOWCHARTS

NONE

10.0 LISTING

ATTACHED


```

1      /4K TO 32K PDP-8A PROCESSOR EXERCISER
2      /
3      /MAL-DEC-08-DJEXC-A-L
4      /
5      /COPYRIGHT 1975, DIGITAL EQUIPMENT CORPORATION
6      /
7      /PROGRAMMER;   BRUCE HANSEN
8
9
10     7421   MQL=7421
11     7701   ACL=7701
12     7604   LAS=7604
13     7402   HLT=7402
14     6001   ION=6001
15     6002   IOF=6002
16     6160   SIMCLR=6160   /CLEAR SIMULATOR LOGIC
17     0244   RMF=6244
18     6035   KIE=6035
19     6007   CAF=6007   /CLEAR ALL FLAGS
20     6101   SBE=6101   /SKIP ON BATTERY EMPTY
21     6102   SPL=6102   /SKIP ON AC LOW
22     6103   CAL=6103   /CLEAR AC LOW F/F
23     6135   CLLE=6135   /SET INT. ENA ON REAL TIME CLOCK IF DATA BIT 11 ON A 1
24     6136   CLCL=6136   /CLEAR REAL TIME CLOCK FLAG
25     6137   CLSK=6137   /SKIP ON REAL TIME CLOCK FLAG
26
27     0000   *0
28
29
30     0000   0302           302           /REVISION B
31     0001   6202           CIF 00/XX
32     0002   5403           JMP I INT
33     0003   3103           INT,     INTERS
34     0004   0000           RETPNT, 0   /MRI AND OPR RETURN POINTER
35     0005   0200           STRFLD, BGN /STARTING ADDRESS AND FIELD PROGRAM IS LOCATED IN
36
37
38     0010   *10
39     0010   0000           AUTO10, 0
40     AUTO11,
41
42
43     0020   *20
44     0020   0000           SWITCH, 0
45     0021   0007           OP1SEL, 0007
46     0022   0000           OP2SEL, 0000
47
48
49     /SWITCH REGISTER SETTINGS
50
51     /SRO=1 INHIBIT ERROR HALT
52     /SR1=1 LOOP ON ERROR OR TEST CONDITIONS
53     /SR2=1 INHIBIT PROGRAM RELOCATION
54     /SR3=1 HALT AFTER EXECUTION OF A PROGRAM PASS(4096 TEST INSTRUCTIONS)
55

```

```

56
57     /LOCATIONS 0005 TO 0177 WILL BE OVERLAYED ONCE THE PROGRAM HAS BEEN STARTED.
58     /IF THE PROGRAM HAS BEEN SETUP TO RUN WITH OR WITHOUT THE FRONT PANEL
59     /SWITCH REGISTER, IT CANNOT BE REINITIALIZED AGAIN. THE ONLY WAY TO
60     /CHANGE THE FRONT PANEL STATUS IS TO RELOAD THE PROGRAM AND REINITIALIZE IT.
61
62     /THE FOLLOWING ROUTINE WILL CHANGE "TAD (I) SAVSWR" TO LAS, IF THE
63     /OPERATOR SET BIT 0 OF LOCATION 21 TO A ONE.
64
65     0023   0000   PATCH, 0
66     0024   1124   TAD     K5771
67     0025   3526   DCA I   LGC200
68     0026   1125   TAD     K5772
69     0027   3527   DCA I   LGC201
70     0030   7340   CLA CLL  CMA
71     0031   1023   TAD     PATCH
72     0032   3023   DCA     PATCH
73     0033   6160   SIMCLR
74     0034   1022   TAD     OP2SEL
75     0035   3471   DCA I   SAVOP2
76     0036   1022   TAD OP2SEL
77     0037   7006   RTL     /VT78/ CHECK IF RUNNING ON A VT78 SYSTEM
78     0040   7710   SPA CLA /VT78/ BIT 2--HARDWARE CONFIG, WORD 2
79     0041   5472   JMP I XPTCH3 /VT78/ RUNNING ON A VT78!!!
80     0042   1022   PATCH1, TAD OP2SEL /CHECK FOR THE ACT LINE BIT
81     0043   7700   SMA     CLA /IS IT SET ?
82     0044   5423   JMP I PATCH /NO RETURN TO THE PROGRAM
83     0045   1115   TAD     OVRLAY
84     0046   3010   DCA     AUTO10
85     0047   1116   TAD     MROVR
86     0050   3011   DCA     AUTO11
87     0051   4073   JMS     MOVOPR /GO OVERLAY FIRST 5 LOCATIONS OF ERROR
88     0052   1117   TAD     OVRLY1
89     0053   3010   DCA     AUTO10
90     0054   1120   TAD     OPROVR
91     0055   3011   DCA     AUTO11
92     0056   4073   JMS     MOVOPR /GO OVERLAY FIRST 5 LOCATIONS OF ERROPR
93     0057   1021   TAD     OP1SEL /GET THE HARDWARE CONFIGURATION
94     0060   0122   AND     CON37 /MASK OFF MEMORY SIZE
95     0061   1123   TAD     MIN37 /CHECK TO SEE IF 32K SELECTED
96     0062   7640   SZA     CLA /IS THERE 32K SELECTED?
97     0063   5470   JMP I PATCHC /NO, GO TO NEXT BUFFER TO GET NEXT OVERLAY
98     0064   7240   CLA     CMA /SUBTRACT 1K FROM 32K
99     0065   1021   TAD     OP1SEL
100    0066   3021   DCA     OP1SEL /SAVE MEMORY SIZE AS 31K
101    0067   5470   JMP I PATCHC /CONTINUE THE OVERLAY FOR ACT LINE
102    0070   5200   PATCHC, PATCH2
103    0071   3456   SAVOP2, SELOP2
104    0072   5307   XPTCH3, PATCH3 /VT78/
105
106    0073   0000   MOVOPR, 0
107    0074   1121   TAD     M5
108    0075   3130   DCA     PATMOV
109    0076   1410   TAD I   AUTO10
110    0077   3411   DCA I   AUTO11

```

```

111 0100 2130      ISZ  PATMOV
112 0101 5076      JMP  ,-J
113 0102 5473      JMP  I  MUVOVP
114
115 0103 6002      AEROV1, IOF
116 0104 6272      CIF      70
117 0105 1767      1767
118 0106 5717      5717
119 0107 6520      6520
120
121 0110 6002      AERUV2, IOF
122 0111 6272      CIF      70
123 0112 1745      1745
124 0113 5712      5712
125 0114 6520      6520
126
127 0115 0102      OVRLAY, AEROV1=1
128 0116 1312      MRIOVR, ERROR=1
129 0117 0107      OVRLY1, AEROV2=1
130 0120 2305      OPROVR, ERROPR=1
131
132 0121 7773      M5,      =5
133 0122 0037      CON37,  37
134 0123 7741      MIN37,  -37
135
136
137 0124 5771      K5771,  5771
138 0125 5772      K5772,  5772
139 0126 0200      LOC200, BGN
140 0127 0201      LOC201, BGN+1
141 0130 0000      PATMOV,  0
142
143
144          0200      *200
145          /
146 0700 0000      BGN,      0/JMS PATCH/JMP I XBGRAN
147 0201 0000      0/JMS PATCH/JMP I XBGCON
148
149 0702 0000      CHANGE,  0
150 0703 1602      TAD I CHANGE      /GET THE WORD TO MODIFY
151 0704 7450      SNA              /IS IT EQUAL TO ZERO
152 0705 5602      JMP I CHANGE      /YES ALL DONE MODIFYING
153 0706 1212      TAD SUBADD        /SUBTRACT OR ADD 200
154 0707 3602      DCA I CHANGE      /RESTORE THE MODIFIED WORD
155 0710 2202      ISZ CHANGE
156 0711 5203      JMP ,-6           /GET THE NEXT WORD TO MODIFY
157          /
158 0712 0000      SUBADD,  0
159 0713 0000      DIRFLG,  0
160 0714 0200      L0WLIM,  200
161 0715 7400      M400,    -400
162          /
163 0716 1245      SWAP1,  TAD SZPRG      /ROUTINE TO SWAP PROGRAM UP
164 0717 3202      DCA CHANGE        /SAVE PROGRAM SIZE
165 0720 1375      TAD XENDPR        /MODIFIED END OF PROGRAM
    
```

```

166 0721 3246      DCA CNTR2
167 0722 1251      TAD M200
168 0723 1375      TAD XENDPR
169 0724 3247      DCA CNTR3      /GET ACTUAL END OF PROGRAM
170 0725 1647      MOVUP,  TAD I CNTR3
171 0726 3646      DCA I CNTR2
172 0727 1647      TAD I CNTR3      /COMPARE THE WORD THAT WAS RELOCATED
173 0730 7041      CIA
174 0731 1646      TAD I CNTR2
175 0732 7640      SZA CLA
176 0733 7402      HLT              /COMPARE ERROR DURING RELOCATION
177 0734 7040      CMA
178 0735 1247      TAD CNTR3
179 0736 3247      DCA CNTR3
180 0737 7040      CMA
181 0740 1246      TAD CNTR2
182 0741 3246      DCA CNTR2
183 0742 2202      ISZ CHANGE
184 0743 5225      JMP MOVUP
185 0744 5776      JMP I RSCNT
186          /
187 0745 3001      SZPRG,  BGN=PRGEND-1
188 0746 0000      CNTR2,  0
189 0747 0000      CNTR3,  0
190 0750 0400      K400,  400
191 0751 7600      M200,    -200
192          /
193          /
194 0752 4756      LPCNT,  JMS I GETSWR      /LOOP ON INSTRUCTION IF SR1 =1
195 0753 7004      RAL
196 0754 7700      SNA CLA
197 0755 5271      JMP XCNT
198 0756 1762      RESETT, TAD I XINSTR      /EXIT, AND BUMP COUNTERS
199 0757 0250      AND K400          /IS INSTRUCTION INDIRECT
200 0760 7650      SNA CLA
201 0761 5763      JMP I XNTIND
202 0762 1762      TAD I XINSTR      /NO, RESET DATA IN REFERENCE ADDRESS
203 0763 0214      AND  L0WLIM        /YES, REGENERATE REFERENCE ADDRESS
204 0764 7640      SZA CLA          /MASK OUT PAGE BIT
205 0765 1764      TAD I XASAVA
206 0766 1765      TAD I XASAVB
207 0767 3766      DCA I XREFAD
208 0770 5767      JMP I XLOOP
209 0771 2247      XCNT,  ISZ CNTR3      /BUMP PASS COUNTER
210 0772 5770      JMP I RSCNTX
211 0773 4760      JMS I WAIT
212 0774 4755      JMS I CBPASS      /IF OPTION 1 SELECTED WAIT FOR FLAGS
213 0775 7402      HLT              /END OF A PROGRAM PASS
214 0776 4756      JMS I GETSWR      /END OF A PROGRAM PASS HALT IF SR=400
215 0777 7006      RIL              /CHECK FOR RELOCATION
216 0780 7710      SPA CLA          /DO NOT RELOCATE IF SR2=1
217 0781 5776      JMP I RSCNT
218 0782 6224      ARRANG, RIF
219 0783 7041      CIA              /GET HOME FIELD
220 0784 1754      TAD I XFLD        /IS IT EQUAL TO LAST FIELD
    
```

```

221 0305 7640 SZA CLA
222 0306 7240 CIA CMA /NO,PROGRAM IS IN A 4K FIELD
223 0307 7450 SNA /YES, IN LAST FIELD GET UPPER LIMITS
224 0310 1773 IAD I XUPERL
225 0311 3774 DCA I HIGHLM /SAVE UPPER LIMIT
226 0312 1213 TAD DIRFLG /IS THE PROGRAM ROLLING UP OR ROLLING BACK
227 0313 7640 SZA CLA
228 0314 5761 JMP I XROLBK /THE PROGRAM IS ROLLING BACK
229 0315 1375 ROLLUP, TAD XENDPR /GET END OF PROGRAM AND COMPARE IT
230 0316 7040 CMA /WITH HIGH LIMITS
231 0317 1774 TAD I HIGHLM
232 0320 7650 SNA CLA
233 0321 5761 JMP I XROLBK /THIS IS NEEDED FOR A 1K FIELD OTHER THAN 0
234 0322 5324 JMP SETFLG
235 0323 7240 CLA CMA /NO,SET REVERSE FLAG
236 0324 3213 SETFLG, DCA DIRFLG /-1 IF GOING REVERSE; 0 1G FORWARD
237 0325 1213 TAD DIRFLG /ROLLING UP OR ROLLING BACK?
238 0326 7640 SZA CLA
239 0327 1215 TAD M400 /ROLLING BACK IF DIRECTIONN FLAG = -1
240 0330 1214 TAD LOWLM /ROLLING UP IF FLAG = 0
241 0331 3212 DCA SUBADD /SAVE 200 OR -200
242 0332 5353 JMP ACHNG
243
244
245 0333 0000 SAVSWR, 0
246 /
247
248 0334 0000 F0INIT, 0
249 0335 6201 CDF 00 /CHANGE DATA FIELD TO FIELD 0
250 0336 6224 RIF /READ THE INSTRUCTION FIELD
251 0337 1335 TAD F0INIT+1 /GET THE CDF INSTRUCTION
252 0340 7001 IAC /MAKE IT A CIF TO PROGRAM FIELD
253 0341 3742 DCA I CIFFDO /PUT IT IN LOCATION 1 OF FIELD 0
254 0342 0001 CIFFDO, INT-2
255 0343 7240 CLA CMA /SET THE AC TO ALL ONE'S
256 0344 1742 TAD I CIFFDO /CHANGE CIF BACK TO CDF PROGRAM FIELD
257 0345 3347 DCA .+2 /PUT IT IN NEXT LOCATION
258 0346 4757 JMS I SETINT
259 0347 7402 HLT/CDF TO PROGRAM FIELD
260 0350 5734 JMP I F0INIT /RETURN TO PROGRAM
261
262 0353 *353
263 /
264 0353 4202 ACHNG, JMS CHANGE
265 /
266 0354 1144 XFLD, FLDLIM
267 0355 3401 CBPASS, XCBPAS

768 0356 3466 GEISWR, XCBSW
769 0357 2137 SETINT, INTSET
770 0360 3300 WAIT, WAITEN
771 0361 0401 XROLBK, ROLBAK
772 0362 0746 XINSTR, INSTR
773 0363 0625 XNTIND, NOTIND
774 0364 1146 XASAVA, ASAVA
775 0365 1147 XASAVB, ASAVB
    
```

```

276 0366 0747 XREFAD, REFAD
277 0367 0602 XLOOP, LOOPID+1
278 0370 1001 RSCNTX, GENFLD
279 0371 3026 XBGRAN, BGNCON+1
280 0372 3025 XBGC0N, BGNCON
281 0373 1450 XUPERL, UPRLIM
282 0374 1145 HIGHLM, HGHLIM
283 0375 5176 XENDPR, PRGEND
284 0376 5025 RSCNT, STARTP
285 0377 0900 0
286
287
288 0400 *400
289 0400 5351 JMP AACHNG
290 /
291 0401 1367 ROLBAK, TAD BEGIN /GET BEGINNING OF PROGRAM AND COMPARE IT
292 0402 7041 CIA /WITH THE LOW LIMIT
293 0403 1770 TAD I XLWLM /
294 0404 7640 SZA CLA /IS IT EQUAL
295 0405 5771 JMP I RTFLGR /NO,ROLL THE PROGRAM BACK
296 0406 3772 DCA I RTFLG /SET DIRECTION FLAG TO FORWARD
297 0407 1773 TAD I MAXFLD /IS THE PROGRAM LIMIT ONLY 2K-4K
298 0410 7650 SNA CLA
299 0411 5774 JMP I RTFLGF /YES, DO NOT SWAP BUT ROLL THE PROGRAM UP
300 0412 1300 TAD FLDFLG /SWAP THE PROGRAM UP OR DOWN
301 0413 7640 SZA CLA
302 0414 5222 JMP SWAPDN /SWAP THE PROGRAM DOWN
303 0415 6224 SWAPUP, RIF /GET PROGRAM FIELD
304 0416 1301 TAD K10 /ADD 1 FIELD TO IT
305 0417 7041 CIA
306 0420 1773 TAD I MAXFLD
307 0421 5753 JMP I CSWPUP
308 0422 6224 SWAPDN, RIF /GET HOME FIELD
309 0423 7450 SNA /IS IT EQUAL TO FIELD 0
310 0424 5215 JMP SWAPUP /YES,SWAP THE PROGRAM UP
311 0425 1303 TAD M10 /SUBTRACT 1 FIELD
312 0426 7640 SZA CLA /IS IT EQUAL TO FIELD 0?
313 0427 5232 JMP SFDFG-1 /NO,SET FLAG TO REVERSE AND SWAP DOWN
314 0430 3300 DCA FLDFLG /YES,BUT SWAP DOWN AND SET FLAG TO FORWARD
315 0431 5236 JMP .+5 /GO SWAP IT
316 0432 7240 CLA CMA
317 0433 3300 SFDFG, DCA FLDFLG /FIELD FLAG=0 SWAP UP;-1 SWAP DOWN
318 0434 1300 TAD FLDFLG /SWAPING UP OR DOWN
319 0435 7640 SZA CLA
320 0436 1302 TAD M20 /SWAPPING DOWN
321 0437 1301 TAD K10 /SWAPING UP
322 0440 3276 DCA NEWDFA+1 /SAVE 10 OR -10
323 0441 6224 RIF /GET HOME FIELD
324 0442 1276 TAD NEWDFA+1 /ADD OR SUBTRACT A FIELD
325 0443 1326 TAD B6201
326 0444 3257 DCA NEWDTF /PUT 62X1 IN THE SWAP ROUTINE
327 0445 6224 RIF /GET HOME FIELD
328 0446 1326 TAD B6201
329 0447 3263 DCA SWPFLD /TO RETURN BACK TO HOME FIELD
330 0450 1257 TAD NEWDTF
    
```



```

441 0607 1343          1AD A7
442 0610 7630          SZL CLA          /WAS IT WITHIN AUTO BOUNDARY
443 0611 7610          NOTAUT, SKP CLA          /NO, NOT AUTO-INDEX
444          5767          VI7RO=JMP I XVI78G          /VI78/
445 0612 7340          OWLVI, CLA CLL CMA/JMP I XVI78G /VI78//AUTO INDEX,SUBTRACT 1 FROM INDIRECT ADDRESS
446          /VI78/NOTE:ON THE VI78 PROCESSOR THE PAGE
447          /VI78/ BIT MUST BE ZERO FOR AUTO INDEXING TO
448          /VI78/ WORK--EVEN IF THE INST IS ON PAGE 0.
449          /VI78/ THEREFORE ZERO BIT WHENEVER
450          /VI78/ AUTO INDEXING FROM PAGE 0.
451 0613 1350          RFTV, TAD INDD          /SAVE INDIRECT ADDRESS
452 0614 3310          DCA SETRET
453 0615 1354          IAD RANFLD
454 0616 1356          TAD K6201          /CHANGE TO A RANDOM DATA FIELD
455 0617 3220          DCA ,+1
456 0620 7402          HLT/CFD
457 0621 1310          TAD SETRET          /GET INDIRECT ADDRESS
458 0622 3747          DCA I REFAD          /PUT INDIRECT ADDRESS INTO REF ADD
459 0623 1350          TAD INDD
460 0624 3347          DCA REFAD          /MAKE REFAD=INDD
461 0625 7330          NOTIND, CLA CLL CML RAR
462 0626 1346          TAD INSTR
463 0627 7630          SZL CLA          /WHAT TYPE OF INSTR
464 0630 5265          JMP JMRJMS          /IT WAS A JMP OR JMS
465 0631 1354          TAD RANFLD
466 0632 1356          TAD K6201
467 0633 3234          DCA ,+1
468 0634 7402          HLT/CFD          /CHANGE TO A RANDOM DATA FIELD
469 0635 1351          TAD DATATH          /GET INITIAL MEMORY DATA AND PUT IT IN
470 0636 3747          DCA I REFAD          /REF ADD OR INDIRECT ADD FOR AND THROUGH DCA
471 0637 7240          OPRINT, CLA CMA          /SUBTRACT 1 FROM INSTRUCTION ADDRESS
472 0640 1345          TAD ADDR          /AND SAVE IT
473 0641 3344          DCA HOMCIF
474 0642 6224          RIF
475 0643 1357          TAD K6202          /SET UP HOME INSTRUCTION FIELD
476 0644 3744          DCA I HOMCIF          /IN INSTRUCTION ADDRESS=1 FOR AND=DCA
477 0645 7301          CLA CLL IAC
478 0646 1345          TAD ADDR
479 0647 4310          JMS SETRET          /SETUP RETURN,INSTR ADD+1,+2=4400 FOR AND=DCA
480
481 0650 1346          NOTJJ, TAD INSTR          /LOCATION 0 CONTAINS RETURN POINTER
482 0651 3745          DCA I ADDR          /PUT INSTRUCTION IN INSTRUCTION ADDRESS
483 0652 1355          TAD SAVLNK
484 0653 7104          CLL RAL
485 0654 1353          TAD MQDATA          /GET THE RANDOM MQ DATA
486 0655 7421          MQL          /AND LOAD IT INTO THE MQ
487 0656 7200          CLA          /SAFETY CLEAR THE AC IN CASE MQL DOESN'T
488 0657 1354          TAD RANFLD          /MAKE UP A CIF TO A RANDOM FIELD
489 0660 1357          TAD K6202
490 0661 3263          DCA ,+2
491 0662 1352          TAD DATAH          /GET THE AC DATA INTO THE AC
492 0663 7402          HLT/CFD          /D.F. HAS BEEN CHANGED NOW CHANGE I,F.
493 0664 5744          JMP I HOMCIF          /GO EXECUTE INSTRUCTION IN RANDOM FIELD
494
495

```

////////////////////////////////////

```

496          //FOR AND'S THROUGH DCA'S DIRECTS THE INSTRUCTION SETUP IS AS FOLLOWS:
497          /
498          /SOME RANDOM FIELD
499          /LOCATION 4 OF THIS FIELD EQUALS RETURN POINTER TO PROGRAM FIELD
500          /THE AC EQUALS SOME RANDOM NUMBER
501          /INST ADD=1= CIF TO PROGRAM FIELD
502          /INST ADD = TEST INSTRUCTION
503          /INST ADD+1= JMS I 4
504          /INST ADD+2= JMS I 4
505          /
506          /REF ADD = INITIAL MEMORY DATA, THIS IS THE LOC THE INST WILL REFERENCE
507          //////////////////////////////////////
508
509          //////////////////////////////////////
510          //FOR AND'S THROUGH DCA'S INDIRECTS THE INST SETUP IS AS FOLLOWS
511          /
512          /SOME RANDOM FIELD
513          /LOCATION 4 OF THIS FIELD EQUALS RETURN POINTER TO PROGRAM FIELD
514          /THE AC EQUALS SOME RANDOM NUMBER
515          /INSTR ADD=1= CIF TO HOME FIELD
516          /INST ADD = TEST INSTRUCTION
517          /INST ADD+1= JMS I 4
518          /INST ADD+2= JMS I 4
519          /
520          /REF ADD = INDIRECT ADDRESS
521          /
522          /IND ADD = INITIAL MEMORY DAA
523          //////////////////////////////////////
524 0665 1346          JMPJMS, TAD INSTR          /GET THE INSTRUCTION
525 0666 7066          RTL          /IS IT A JMP OR JMS?
526 0667 7700          SMA CLA
527 0670 7001          IAC          /JMS ADD 1 TO REFERENCE ADDRESS FOR CIF INST
528 0671 1347          TAD REFAD          /GET REFERENCE ADDRESS
529 0672 3310          DCA SETRET          /AND SAVE IT FOR THE CIF INSTRUCTION
530 0673 1354          TAD RANFLD          /MAKE CDF INST TO THE RANDOM FIELD
531 0674 1356          TAD K6201
532 0675 3276          DCA ,+1
533 0676 7402          HLT/CFD          /CHANGE TO RANDOM DATA FIELD
534 0677 6224          RIF
535 0700 1357          TAD K6202          /MAKE A CIF INSTRUCTION TO HOME FIELD
536 0701 3710          DCA I SETRET          /PUT IT IN REFERENCE ADD OR INDIRECT ADD
537 0702 7001          IAC
538 0703 1310          TAD SETRET
539 0704 4310          JMS SETRET          /SETUP LOC 4 AND JMS I 4 IN APPROPRIATE PLACES
540 0705 1345          TAD ADDR          /GET INSTRUCTION ADDRESS
541 0706 3344          DCA HOMCIF          /SAVE IT
542 0707 5250          JMP NOTJJ          /GO GET INSTRUCTION AND SETUP
543          //////////////////////////////////////
544          //THE INSTRUCTION SETUP FOR JMP DIRECTS IS AS FOLLOWS:
545          /
546          /SOME RANDOM FIELD
547          /LOC 4 OF THIS FIELD EQUALS RETURN POINTER TO PROGRAM FIELD
548          /THE AC EQUALS SOME RANDOM NUMBER
549          /INST ADD =JMP INSTRUCTION
550          /

```

```

551 /REF ADD =CIF TO PROGRAM FIELD
552 /REF ADD+1 =JMS I 4
553 /REF ADD+2 =JMS I 4
554 ///////////////////////////////////////////////////
555 ///////////////////////////////////////////////////
556 //THE INSTRUCTION SETUP FOR JMP INDIRECTS IS AS FOLLOWS:
557 /
558 /SOME RANDOM FIELD
559 /LOC 4 OF THIS FIELD EQUALS RETURN POINTER TO PROGRAM FIELD
560 /THE AC EQUALS SOME RANDOM NUMBER
561 /INST ADD =JMP INDIRECT INSTRUCTION
562 /
563 /REF ADD =INDIRECT ADDRESS
564 /
565 /IND ADD =CIF TO PROGRAM FIELD
566 /IND ADD+1 =JMS I 4
567 /IND ADD+2 =JMS I 4
568 ///////////////////////////////////////////////////
569 ///////////////////////////////////////////////////
570 //THE INSTRUCTION SETUP FOR JMS DIRECTS IS AS FOLLOWS
571 /
572 /SOME RANDOM FIELD
573 /LOC 4 OF THIS FIELD EQUALS RETURN POINTER TO PROGRAM FIELD
574 /THE AC EQUALS SOME RANDOM NUMBER
575 /INST ADD =JMS DIRECT INSTRUCTION
576 /
577 /REF ADD =SOME UNKNOWN NUMBER
578 /REF ADD+1 =CIF TO PROGRAM FIELD
579 /REF ADD+2 =JMS I 4
580 /REF ADD+3 =JMS I 4
581 ///////////////////////////////////////////////////
582 //THE INSTRUCTION SETUP FOR JMS INDIRECTS IS AS FOLLOWS
583 /
584 /SOME RANDOM FIELD
585 /LOC 4 OF THIS FIELD EQUALS RETURN POINTER TO PROGRAM FIELD
586 /THE AC EQUALS SOME RANDOM NUMBER
587 /INST ADD =JMS INDIRECT INSTRUCTION
588 /
589 /REF ADD =INDIRECT ADDRESS
590 /
591 /IND ADD =SOME UNKNOWN NUMBER
592 /IND ADD+1=CIF TO PROGRAM FIELD
593 /IND ADD+2=JMS I 4
594 ///////////////////////////////////////////////////
595 /IND ADD+3=JMS I 4
596 ///////////////////////////////////////////////////
597
598
599
600
601 /THIS ROUTINE SETS UP LOC 0 IN SOME FIELD FOR RETURN POINTER
602 /TO THE PROGRAM AND ALSO SETS UP THE JMS I 0'S AFTER THE EXECUTION OF THE
603 /INSTRUCTION.
604
605 0710 0000 SETRET, 0
    
```

```

606 0711 3362 DCA JMSLOC
607 0712 7301 CLA CLL IAC
608 0713 1362 TAD JMSLOC
609 0714 3363 DCA JMSLOC
610 0715 1364 TAD KJMS
611 0716 3762 DCA I JMSLOC
612 0717 1364 TAD KJMS
613 0720 3763 DCA I JMSLOC
614 0721 1376 TAD JMSRET
615 0722 3761 DCA I FLDRET
616 0723 5710 JMP I SETRET
617
618 0724 4773 ERRORP, JMS I ZGETWD
619 0725 4770 JMS I YHALT
620 0726 1352 TAD DATAH
621 0727 4770 JMS I YHALT
622 0730 1774 TAD I ZFIND
623 0731 4770 JMS I YHALT
624 0732 1355 TAD SAVLNK
625 0733 4770 JMS I YHALT
626 0734 1771 TAD I FLINK
627 0735 4770 JMS I YHALT
628 0736 1353 TAD MQDATA
629 0737 4770 JMS I YHALT
630 0740 1772 TAD I FMQUAT
631 0741 4770 JMS I YHALT
632 0742 5775 JMP I ZCNT
633
634 0743 0007 /
635 0744 0000 A7, 7
636 0745 0000 HOMCIF, 0
637 0746 0000 ADDR0, 0
638 0747 0000 INSTR, 0
639 0750 0000 REFAD, 0
640 0751 0000 INAD, 0
641 0752 0000 DATATH, 0
642 0753 0000 DATAH, 0
643 0754 0000 MODATA, 0
644 0755 0000 RANFLD, 0
645 0756 6201 SAVLNK, 0
646 0757 6202 K6201, 6201
647 0760 7770 K6202, 6202
648 0761 0004 K7770, 7770
649 0762 0000 FLDRET, 4
650 0763 0000 JMSLOC, 0
651 0764 4401 JMSLOC, 0
652 /
653 0765 *765
654 /
655 0765 4766 ACHG, JMS I ARERNG
656 /
657 0766 0202 ARERNG, CHANGE
658 0767 4700 XVT78G, VT78G
659 0770 1335 YHALT, HALT
660 0771 2746 FLINK, LINKDN
    
```

```

661 0772 2750 F1QDAT, MDDONE
662 0773 2112 ZGETWD, GLIWD
663 0774 0527 ZFLWD, FLALL
664 0775 0252 ZCNT, LPCNT
665 0776 0504 JMSHET, RETHN
666 0777 0000 U
667
668
669          1000 *1000
670 1000 5350 JMP A1CHG
671
672
673 1001 4763 /
674 1002 0325 GENFLD, JMS I ARANDY /GET A RANDOM FIELD
675 1003 0327 AND K70 /MASK WORD FOR FIELD BITS
676 1004 1330 AND FLDSK /MASK WORD FOR FIELD
677 1005 0325 TAD CONFLD /CONSTRAINT WORD FOR FIELD
678 1006 3756 AND K70
679 1007 1756 DCA I FLDRAN /COMPARE RANDOM FIELD WITH UPPER LIMITS
680 1010 7041 TAD I FLDRAN
681 1011 1344 CIA
682 1012 7510 TAD FLDLIM
683 1013 5201 SPA /WITHIN LIMITS ?
684 1014 7640 JMP GENFLD /NO REGENERATE A NEW FIELD
685 1015 7240 SZA CLA /WAS IT THE LAST MEMORY FIELD
686 1016 7450 CMA CMA /NO SET UPPER BOUNDARY = TO 7777
687 1017 1757 SNA /
688 1020 3345 TAD I XUPLIM /GET THE UPPER LIMIT OF LAST FIELD
689 1021 1345 DCA HGLIM /SAVE THE UPPER BOUNDARY
690 1022 7041 TAD HGLIM /SETUP A NUMBER FOR BOUNDARY COMPARE
691 1023 1326 CIA
692 1024 3760 TAD ADD11
693 DCA I XBNDCN /SAVE THE NUMBER FOR CHECKING BOUNDRIES
694 1025 4763 MEMDAT, JMS I ARANDY /GENERATE RANDOM MEMORY DATA FOR AND>DCA
695 1026 0335 AND MDMSK /MASK WORD FOR MEMORY DATA
696 1027 1336 TAD CONMDT /CONSTRAINT WORD
697 1030 3771 DCA I ADATAT /SAVE IT
698 1031 4763 ACDATA, JMS I ARANDY /GENERATE RANDOM AC DATA
699 1032 0337 AND ACDMSK /MASK WORD
700 1033 1340 TAD CONACD /CONSTRAINT WORD
701 1034 3772 DCA I ADATAH /SAVE THE AC DATA WORD
702 1035 7010 RAR /MOVE THE LINK INTO AC BIT 0
703 1036 3774 DCA I LNKSAB /SAVE THE LINK
704 1037 4763 GENMOD, JMS I ARANDY /GENERATE RANDOM MQ DATA
705 1040 0341 AND MQDMSK /MASK WORD FOR MQ DATA
706 1041 1342 TAD CONMQD /CONSTRAINT WORD FOR MQ DATA
707 1042 3773 DCA I AMQDAT /SAVE THE MQ DATA WORD
708 1043 4763 GENADD, JMS I ARANDY /GENERATE RANDOM ADDRESS FOR INSTRUCTION
709 1044 0345 AND HGLIM /MASK OFF ADDRESS BITS FOR THIS FIELD
710 1045 0331 AND ADRMSK /MASK WORD FOR INSTRUCTION ADDRESS
711 1046 1332 TAD CONADR /CONSTRAINT WORD
712 1047 4764 JMS I ABNRY1 /IS IT WITHIN LIMITS
713 1050 5243 JMP GENADD /NO, TRY AGAIN
714 1051 3766 DCA I ADDR8 /THIS IS THE INSTRUCTIONS ADDRESS
715 1052 1766 TAD I ADDR8

```

```

716 1053 0305 AND CONST1
717 1054 3346 DCA ASAVA /SAVE PAGE BITS FOR FORMING REFERENCE ADDRESS
718 1055 4755 GENINS, JMS I XGENT1 /GENERATE RANDOM INSTRUCTION
719 1056 4764 JMS I ABNRY1 /IS IT WITHIN LIMITS
720 1057 5305 JMP CONST1 /NO, TRY AGAIN
721 1060 4761 JMS I ASAME1 /COMPARE TO ADDR8
722 1061 5305 JMP CONST1 /THERE EQUAL OR TO CLOSE TRY AGAIN
723 1062 3770 DCA I AREFAD /STORE REFERENCE ADDRESS
724 1063 6214 RDF
725 1064 1323 TAD C6201 /PUT CDF HOME FIELD INTO INSTRUCTION RETURN
726 1065 3765 DCA I XRETHR
727 1066 1767 TAD I AINSTR /INSTR = INSTRUCTION TO TEST
728 1067 0324 AND A400
729 1070 7650 SNA CLA /WAS INSTR INDIRECT
730 1071 5775 JMP I ANTIND /NO, NOT INDIRECT GO SETUP TEST CONDITIONS
731 1072 4763 GENIND, JMS I ARANDY /GENERATE RANDOM INDIRECT ADDRESS
732 1073 0345 AND HGLIM /MASK OFF ADDRESS BITS FOR THIS FIELD
733 1074 0333 AND INDMSK /MASK WORD FOR INDIRECT ADDRESS
734 1075 1334 TAD CONIND /CONSTRAINT WORD FOR INDIRECT
735 1076 4764 JMS I ABNRY1 /IS IT WITHIN BOUNDARIES
736 1077 5312 JMP CONST2 /NO, TRY AGAIN
737 1080 4761 JMS I ASAME1 /COMPARE TO ADDR8
738 1081 5312 JMP CONST2 /TRY AGAIN
739 1082 4762 JMS I ASAME2 /COMPARE TO REFAD
740 1083 5312 JMP CONST2 /TRY AGAIN
741 1084 5776 JMP I ALOPID /GO SETUP TEST CONDITIONS
742
743
744 1105 7600 /
745 1106 1343 CONST1, 7600
746 1107 7640 TAD CONFLG
747 1110 5243 SZA CLA
748 1111 5255 JMP GENADD
749 JMP GENINS
750
751 1112 1343 /
752 1113 7710 CONST2, TAD CONFLG
753 1114 5243 SPA CLA
754 1115 5272 JMP GENADD
755 JMP GENIND
756
757 1116 7510 DECSWP, SPA /IS IT WITHIN FIELD LIMITS
758 1117 5752 JMP I FLDFGR /NO, SET DIRECTION OF SWAP TO REVERSE
759 1120 7650 SNA CLA /WAS IT THE LAST FIELD?
760 1121 5754 JMP I CHK1KF /GO CHECK TO SEE IF NEXT FIELD IS 3K
761 1122 5753 JMP I FLDFGF /NO, SET DIRECTION OF SWAP TO FORWARD
762
763 1123 6201 /
764 1124 0400 C6201, 6201
765 1125 0070 A400, 400
766 1126 0011 K70, 70
767 1127 7777 ADD11, 11
768 1130 0000 FLDSK, 7777
769 1131 7777 CONFLD, 0
770 1132 0000 ADRMSK, 7777
771 1133 7777 CONADR, 0
772 INDSK, 7777

```

```

771 1134 0000 CONIND, 0
772 1135 7777 MDTMSK, 7777
773 1136 0000 CONMDT, 0
774 1137 7777 ACDMSK, 7777
775 1140 0000 CONACD, 0
776 1141 7777 MQDMSK, 7777
777 1142 0000 CONMOD, 0000
778 1143 0000 CONFLG, 0
779 1144 0000 FLDLIM, 0
780 1145 0000 MGHLM, 0
781 1146 0000 ASAVA, 0
782 1147 0000 ASAVB, 0
783
784 1150 *1150
785 /
786 1150 4751 A1CHG, JMS I AIRRNG
787 /
788 1151 0202 AIRRNG, CHANGE
789 1152 0432 FLDFGR, SFLDGF-1
790 1153 0433 FLOFGF, SFLDGF
791 1154 2555 CHK1KF, FLDCCHK
792 1155 1601 XGENI, INSGEN
793 1156 0754 FLDRAN, RANFLD
794 1157 1550 XUPLIM, UPRLIM
795 1160 1551 XBNDCN, BNDCON
796 1161 1463 ASAME1, SAME1
797 1162 1473 ASAME2, SAME2
798 1163 1401 ARANDY, RANDY
799 1164 1435 ABNRY1, BNDRY1
800 1165 0510 XRETHR, RETHR+4
801 1166 0745 AADDRS, ADDRS
802 1167 0746 AINSTR, INSTR
803 1170 0747 AREFAD, REFAD
804 1171 0751 ADATAT, DATATH
805 1172 0752 ADATAH, DATAH
806 1173 0753 AMQDAT, MQDATA
807 1174 0755 LNKSAV, SAVLNK
808 1175 0625 ANTIND, NOTIND
809 1176 0601 ALOPID, LOOPID
810 1177 0000 0
811
812 /
813 1200 *1200
814 /
815 1700 5347 JMP BCHNG /GO ULTER
816 /
817 1701 4755 ANDTST, JMS I TSTPC /CHECK PC FROM RETURN
818 1702 1775 TAD I BDATTH
819 1703 0776 AND I BDATHR
820 1704 7041 CIA
821 1705 1772 TAD I DATFN
822 1706 7640 COMPAR, SZA CLA /DID AND WORK
823 1707 5313 JMP ERROR /RANDOM AND FAILED
824 1710 1760 TAD I LINKSV /CHECK TO SEE IF THE LINK CHANGED
825 1711 7041 CIA
    
```

```

826 1212 1761 TAD I LINKRT /
827 1213 7640 SZA CLA /
828 1214 5313 JMP ERROR /ERROR, THE INSTRUCTION CHANGED THE LINK
829 1215 1763 TAD I MQDAT /CHECK TO SEE IF THE INSTR CHANGED THE MQ
830 1216 7041 CIA /
831 1217 1762 TAD I DONEMQ /
832 1220 7640 SZA CLA /
833 1221 5313 JMP ERROR /THE INSTRUCTION CHANGED THE MQ
834 1222 5765 JMP I BLPNT /
835 /
836 1223 4755 TADTST, JMS I TSTPC /CHECK PC FROM RETURN
837 1224 7340 CLA CLL CMA
838 1225 0775 AND I BDATTH
839 1226 1776 TAD I BDATHR
840 1227 7041 CIA
841 1230 1772 TAD I DATFN
842 1231 7640 SZA CLA
843 1232 5313 JMP ERROR
844 1233 5765 JMP I BLPNT
845 /
846 1234 7301 ISZTST, CLA CLL IAC
847 1235 1775 TAD I BDATTH
848 1236 7650 SNA CLA /SHOULD THE ISZ SKIP
849 1237 7001 IAC /YES
850 1240 4755 JMS I TSTPC /CHECK FOR CORRECT PC
851 1241 1776 TAD I BDATHR
852 1242 7041 CIA
853 1243 1772 TAD I DATFN
854 1244 7640 SZA CLA /DID AC CHANGE ON ISZ
855 1245 5313 JMP ERROR /AC FAILED ON ISZ
856 1246 1774 TAD I BREFAD /GET INCREMENTED DATA WORD
857 1247 3756 DCA I XBSAVA
858 1250 4757 JMS I XGETWD
859 1251 7041 CIA
860 1252 7001 IAC
861 1253 1775 TAD I BDATTH
862 1254 5206 JMP COMPAN /DID ISZ WORK
863 /
864 1255 4755 DCATST, JMS I TSTPC /CHECK PC FROM RETURN
865 1256 1774 TAD I BREFAD
866 1257 3756 DCA I XBSAVA
867 1260 4757 JMS I XGETWD
868 1261 7041 CIA
869 1262 1776 TAD I BDATHR
870 1263 7640 SZA CLA /DID DCA WORK
871 1264 5313 JMP ERROR /DCA FAILED
872 1265 1772 TAD I DATFN /DID AC CLEAR ON DCA
873 1266 5206 JMP COMPAN /??
874 /
875 1267 4755 JNSTST, JMS I TSTPC /CHECK PC FROM RETURN
876 1270 1776 TAD I BDATHR
877 1271 7041 CIA
878 1272 1772 TAD I DATFN
879 1273 7640 SZA CLA /DID JMS CHANGE AC
880 1274 5313 JMP ERROR /JMS CHANGED AC
    
```



```

881 1275 1771 TAD I BREFAD
882 1276 1766 OCA I ASAVA
883 1277 4757 JMS I AGETWD
884 1200 7041 CIA
885 1201 7001 IAC
886 1202 1766 TAD I BADUPS
887 1203 5206 JMP COMPAR /DID JMS WORK
888
889 1204 4755 JMPTST, JMS I ISTPC /CHECK PC FROM RETURN
890 1205 1776 TAD I BDATHR
891 1206 7041 CIA
892 1207 1772 TAD I DATFN
893 1210 5206 JMP COMPAR /DID JMP AFFECT THE AC
894
895
896 1211 0000 PCSAVE, 0
897 1212 0200 C200, 200
898
899 1213 1764 ERROR, TAD I XRNFLO
900 1214 4335 JMS HALT /FIELD THAT INSTRUCTION WAS PUT IN
901 1215 1773 TAD I XRETFI
902 1216 4335 JMS HALT /PROGRAM RETURNED FROM THIS FIELD
903 1217 1211 TAD PCSAVE
904 1220 4335 JMS HALT /EXPECTED PC RETURN
905 1221 1754 TAD I RETUPN
906 1222 4335 JMS HALT /ACTUAL PC RETURN
907 1223 1766 TAD I BADUPS
908 1224 4335 JMS HALT /INSTRUCTION ADDRESS
909 1225 1767 TAD I FINSTR
910 1226 4335 JMS HALT /INSTRUCTION
911 1227 1767 TAD I FINSIR
912 1230 0312 AND C200
913 1231 7640 SZA CLA
914 1232 1770 ERRPSR, TAD I ZASAVA
915 1233 1771 TAD I ZASAVB
916 1234 5753 JMP I XERROR /GET REST OF ERROR INFORMATION
917
918 1235 0000 HALT, 0
919 1236 4751 JMS I CBERK/HLT /ERROR INFORMATION IN AC
920 1237 7200 CLA
921 1240 5735 JMP I HALT
922
923
924 1247 1347 *1347
925
926 1247 4750 BCHNG, JMS I BRERNG
927
928 1250 0202 BRERNG, CHANGE
929 1251 4401 CBERK, XCBERR
930 1252 0252 ERRRET, LPCNT
931 1253 1415 XERROR, ERROK1
932 1254 0504 RETURN, REIHM
933 1255 2073 TSTPC, PCTSI
934 1256 2145 XBSAVA, BSAVA
935 1257 7112 XGETWD, GFTWD

```

```

936 1260 0755 LINKSV, SAVLNK
937 1261 2746 LINKRT, LINKDN
938 1262 2750 DUNED0, MGDONE
939 1263 0753 MGDAT, MGDAL1
940 1264 0754 XRNFLO, RANFLO
941 1265 0752 MLCPCNT, LPCNT
942 1266 0745 HADDNS, ADDR6
943 1267 0746 FINSTR, INSTR
944 1270 1146 ZASAVA, ASAVA
945 1271 1147 ZASAVB, ASAVB
946 1272 0527 DATFN, FILALL
947 1273 0525 XRETFI, RETFLO
948 1274 0747 BREFAD, RFAD
949 1275 0751 BDATHR, DATAH
950 1276 0752 BDATHR, DATAH
951 1277 0000 0
952
953 1280 1400 *1400
954
955 1280 5362 JMP CCHNG
956
957 1201 0000 RANDY, 0
958 1202 7301 CIA CLL IAC
959 1203 1343 TAD RAN1
960 1204 1344 TAD RAN2
961 1205 7106 CLL RTL
962 1206 3343 OCA RAN1
963 1207 1344 TAD RAN2
964 1210 7012 RTK
965 1211 1343 TAD RAN1
966 1212 3344 OCA RAN2
967 1213 1344 RANDY1, TAD RAN2
968 1214 5601 JMP I RANDY
969
970
971
972 1215 3774 ERROR1, OCA I CREFAD
973 1216 1774 TAD I CREFAD
974 1217 4765 JMS I XHALT /REFERENCE ADDRESS
975 1220 1771 TAD I ZINDAD /INDIRECT ADDRESS IF ANY
976 1221 4765 JMS I XHALT /INITIAL MEMORY DATA
977 1222 1767 TAD I COATAT
978 1223 4765 JMS I XHALT
979 1224 1766 TAD I ZINSTR
980 1225 0347 AND C400
981 1226 7650 SNA CLA
982 1227 5232 JMP +3
983 1230 1771 TAD I ZINDAD
984 1231 3774 OCA I CREFAD
985 1232 1774 TAD I CREFAD
986 1233 3770 OCA I ZBSAVA
987 1234 5772 JMP I XERR2 /GO GET REST OF INFORMATION
988
989
990

```

/THIS SECTION OF THE SURROUTINE CHECKS FOR ILLEGAL ADDRESSES WHICH /ARE AS FOLLOWS:0000 = 0006 AND UPPER TEST AREA LIMIT,-1 AND -2.

991			BNDRY1, 0	
992	1435	0000		
993	1436	3354	DCA CSAVB	
994	1437	1354	IAD CSAVB	/GET THE NUMBER
995	1440	1345	IAD MM7	/SUBTRACT 7 FROM IT
996	1441	7100	CLL	/CLEAR OUT THE LINK

997	1442	1351	TAD BNDCON	/ADD IN BOUNTRY CONSTANT=6012,4012,2012,0012
998	1443	7630	SZL CLA	
999	1444	5635	JMP I BNDRY1	/ILLEGAL ADDRESS, RETURN TO RANDOM NUMBER GENERATOR
1000				
1001				/THIS SECTION OF SUBROUTINE CHECKS FOR ILLEGAL ADDRESS WHICH ARE
1002				/THE PROGRAM AREA-3 TO PROGRAM END +1
1003				
1004	1445	7346	BNDOK1, CLA CLL CMA RTL	
1005	1446	1376	TAD PRGBG	
1006	1447	7041	CIA	
1007	1450	1354	TAD CSAVB	
1008	1451	7420	SNL	
1009	1452	5257	JMP BNDOK2	
1010	1453	7161	CIA STL	
1011	1454	1352	TAD PRGSIZ	

```

1012 1455 7620 SNL CLA
1013 1456 5635 JMP I BNDRY1
1014 1457 2234 BNDUK2, ISZ BNDRY1
1015 1460 7340 CLA CLL CMA
1016 1461 0354 AND CSAVB
1017 1462 5635 JMP I BNDRY1
1018 /
1019 1463 0000 SAME1, 0
1020 1464 3355 DCA CSAVC
1021 1465 1775 TAD I CADDRS
1022 1466 3353 DCA CSAVA
1023 1467 4303 JMS TSAME
1024 1470 2263 ISZ SAME1
1025 1471 1355 TAD CSAVC
1026 1472 5663 JMP I SAME1
1027 /
1028 1473 0000 SAME2, 0
1029 1474 3355 DCA CSAVC
1030 1475 1774 TAD I CREFAD
1031 1476 3353 DCA CSAVA
1032 1477 4303 JMS TSAME
1033 1480 2273 ISZ SAME2
1034 1481 1355 TAD CSAVC
1035 1482 5673 JMP I SAME2
1036 /
1037 1483 0000 TSAME, 0
1038 1484 7344 CLA CLL CMA RAL
1039 1485 1355 TAD CSAVC
1040 1486 7041 CIA
1041 1487 1353 TAD CSAVA
1042 1488 7510 SPA
1043 1489 5320 JMP INSOX
1044 1490 7161 CIA STL
1045 1491 1356 TAD C5
1046 1492 7620 SNL CLA
1047 1493 2303 ISZ TSAME
1048 1494 7420 SNL
1049 1495 2303 ISZ TSAME
1050 1496 7300 INSOX, CLA CLL
1051 1497 5703 JMP I TSAME
1052 /
1053 /
1054 1422 0000 LIMITS, 0
1055 1423 1021 TAD OPISEL /GET MEMORY SIZE FROM HARDWARE CONFIGURATION
1056 1424 0346 AND K37 /MASK OFF MEMORY BITS
1057 1425 7104 CLL RAL
1058 1426 3350 DCA UPRLIM
1059 1427 1350 TAD UPRLIM
1060 1428 0360 AND C70
1061 1429 3773 DCA I XFLDLM
1062 1430 1350 TAD UPRLIM
1063 1431 0357 AND C7
1064 1432 7112 CLL RTR
1065 1433 7012 RTR
1066 1434 1361 TAD C1777

```

```

1067 1537 3350 DCA UPRLIM
1068 1540 1020 TAD SWITCH
1069 1541 3764 DCA I SAVESW
1070 1542 5722 JMP I LIMITS
1071 /
1072 /
1073 1443 1234 RAN1, 1234
1074 1444 5670 RAN2, 5670
1075 1445 7771 MM7, -7
1076 1446 0037 K37, 37
1077 1547 0400 C400, 400
1078 1550 0000 UPRLIM, 0
1079 1551 0000 BNDCON, 0
1080 1552 5002 PRGSIZ, PRGEND+4-BGN
1081 1553 0000 CSAVA, 0
1082 1554 0000 CSAVB, 0
1083 1555 0000 CSAVC, 0
1084 1556 0005 C5, 0005
1085 1557 0007 C7, 7
1086 1560 0070 C70, 70
1087 1561 1777 C1777, 1777
1088 /
1089 1562 *1562
1090 1562 4763 CCHNG, JMS I CRERNG /ROUTINE TO ULTER
1091 /
1092 1563 0202 CRERNG, CHANGE
1093 1564 0333 SAVESW, SAVSWR
1094 1565 1335 XHALT, HALT
1095 1566 0746 ZINSTR, INSTR
1096 1567 0751 CDATAT, DATAH
1097 1570 2145 ZBSAVA, BSAVA
1098 1571 0750 ZINDAD, INDAD
1099 1572 0724 XERR2, ERROR2
1100 1573 1144 XFLDLM, FLDLIM
1101 1574 0747 CREFAD, REFAD
1102 1575 0745 CADDR6, ADDRS
1103 1576 0200 PRGBG, BGN
1104 1577 0000 J
1105 /
1106 /
1107 /
1108 /RANDOM OPERATES=GROUP 1 - GROUP2 - AND HQ OPERATES
1109 /
1110 1600 *1600
1111 /
1112 1600 5346 JMP FCHNG
1113 /
1114 /
1115 /
1116 1601 0000 INSGEN, 0 /ROUTINE TO GENERATE A RANDOM INSTRUCTION
1117 1602 4755 JMS I BRANDY /GO GENERATE A RANDOM NUMBER
1118 1603 0242 AND INMSK /MASK WORD FOR INSTRUCTION
1119 1604 1243 TAD CONINS /CONSTRAINT WORD FOR INSTRUCTION
1120 1605 3754 DCA I EINSTR /SAVE THE INSTRUCTION
1121 1606 6201 CDF 00 /CHANGE DATA FIELD TO FIELD 0

```

```

1122 1407 6224 RIF /READ THE INSTRUCTION FIELD
1123 1610 1356 TAD STARI /GET THE STARTING ADDRESS
1124 1411 3612 DCA I ADDRESS /PUT FIELD AND STARTING ADDRESS INTO LOC 5
1125 1612 0005 ADDRESS, STRFLD /ADDRESS 5 OF FIELD 0 = STARTING ADDRESS AND PRG FIELD
1126 1613 6224 RIF /READ THE INSTRUCTION FIELD
1127 1614 1206 TAD ADDRESS-4 /GET THE CDF INSTRUCTION
1128 1615 3216 DCA +1 /PUT CDF TO PROGRAM FIELD IN NEXT LOCATION
1129 1616 7402 HLT/CDF /CHANGE OF BACK TO PROGRAM FIELD
1130 1617 1754 TAD I EINSTR /CHECK TO SEE IF IT WAS A IOT
1131 1620 0244 AND K7000
1132 1621 1245 TAD M6000
1133 1622 7450 SNA
1134 1623 5202 JMP INSGEN+1 /IT WAS A IOT REGENERATE A NEW INSTRUCTION
1135 1624 1244 TAD K7000 /IS IT AN OPERATE INSTRUCTION
1136 1625 7650 SNA CLA
1137 1626 5256 JMP OPRBGN /YES IT WAS AN OPERATE
1138 1627 1351 TAD MRIPNT /GET THE RETURN POINTER FOR MRI INSTRUCTIONS
1139 1630 3753 DCA I ZJMSRT /SAVE IT
1140 1631 1754 TAD I EINSTR /NOT A IOT OR OPERATE
1141 1632 0246 AND K177 /CREATE A REFERENCE ADDRESS
1142 1633 3761 DCA I AASAVB
1143 1634 1754 TAD I EINSTR /GET THE INSTR
1144 1635 0247 AND A200 /PAGE ZERO OR SAME PAGE
1145 1636 7640 SZA CLA
1146 1637 1760 TAD I AABAVA
1147 1640 1761 TAD I AASAVB
1148 1641 5601 JMP I INSGEN /RETURN AND CHECK IT
1149
1150 1642 7777 INSMK, 7777
1151 1643 0000 COMINS, 0
1152 1644 7000 K7000, 7000
1153 1645 2000 M6000, -6000
1154 1646 0177 K177, 177
1155 1647 0200 A200, 200
1156 1650 0400 B400, 400
1157 1651 0014 A14, 14
1158 1652 7764 NEG14, -14
1159 1653 0001 A1, 1
1160 1654 0006 BP6, 6
1161 1655 7721 K7721, 7721
1162
1163 1656 1352 OPRBGN, TAD OPRPNT /GET THE RETURN POINTER FOR OPR INSTRUCTIONS
1164 1657 3753 DCA I ZJMSRT /SAVE IT
1165 1660 1754 TAD I EINSTR
1166 1661 0250 AND B400
1167 1662 7640 SZA CLA
1168 1663 5272 JMP ILLQP2

1169 1664 1754 ILLQP1, TAD I EINSTR /OP1-CHECK BITS 8 AND 9 TO BE ON A ONE
1170 1665 0251 AND A14
1171 1666 1252 TAD NEG14
1172 1667 7650 SNA CLA
1173 1670 5202 JMP INSGEN+1 /ILLEGAL-REGENERATE A NEW INSTRUCTION
1174 1671 5306 JMP ILLMQ+3 /GO SETUP RANDOM AC AND MQ DATA
1175
1176 1672 1754 ILLQP2, TAD I EINSTR /IS THE INSTR A MQ OR OP2 INSTR

```

```

1177 1673 0253 AND A1
1178 1674 7640 SZA CLA
1179 1675 5303 JMP ILLMQ /INSTR IS A MQ INSTR CHECK FOR ILLEGAL INSTR
1180 1676 1754 TAD I EINSTR /IS THE INSTR A OBR OR HLT
1181 1677 0254 AND BP6
1182 1700 7440 SZA
1183 1701 5202 JMP INSGEN+1 /INSTR IS A OBR OR HLT REGENERATE
1184 1702 5306 JMP ILLMQ+3 /GO SET UP SIMULATED AC DATA AND MQ
1185
1186 1703 1754 ILLMQ, TAD I EINSTR /GET THE INSTRUCTION
1187 1704 0255 AND K7721 /MASK OUT FOR LEGAL MQ INSTRUCTIONS
1188 1705 3754 DCA I EINSTR /AND SAVE IT
1189
1190 1706 1762 TAD I XDATAH
1191 1707 3763 DCA I XSIMAC /PUT INITIAL WORD IN SIMULATED AC
1192 1710 1764 TAD I XSVLNK
1193 1711 3765 DCA I XSMLNK /PUT INITIAL LINK IN SIMULATED LINK
1194 1712 1757 TAD I INTMQD /GET THE RANDOM MQ DATA
1195 1713 3766 DCA I XSIMMQ /PUT INITIAL MQ DATA IN SIMULATED MQ
1196 1714 7326 CLA CLL CML RTL /SET UP INSTRUCTION RETURN POINTER
1197 1715 1773 TAD I OADDRS /GET THE INSTRUCTION ADDRESS AND ADD 2
1198 1716 3767 DCA I XEXPRT /SET UP EXPECTED RETURN UNLESS A SKIP
1199 1717 6214 RDF /READ THE DATA FIELD
1200 1720 1333 TAD 06201 /ADD IN THE CDF INSTRUCTION
1201 1721 3774 DCA I XRTOPF /SET UP A LOC TO RETURN TO OWN DATA FIELD
1202 1722 1754 TAD I EINSTR /IS THE INSTRUCTION A OP1 OR OP2
1203 1723 0250 AND B400
1204 1724 7650 SNA CLA
1205 1725 5770 JMP I XSMOP1 /OP1 GO SIMULATE THE INSTRUCTION
1206 1726 1754 TAD I EINSTR /IS THE INSTR A MQ INSTR
1207 1727 0253 AND A1
1208 1730 7650 SNA CLA
1209 1731 5771 JMP I XSMOP2 /OP2- GO SIMULATE THE INSTRUCTION
1210 1732 5772 JMP I XSMQI /MQ- GO SIMULATE THE MQ INSTR
1211
1212 1733 6201 06201, 6201
1213
1214
1215 1734 1766 OPERR1, TAD I XSIMMQ /GET THE SIMULATED MQ
1216 1735 4341 JMS HLTOPR
1217 1736 1775 TAD I GMQDON /GET THE FINAL MQ
1218 1737 4341 JMS HLTOPR
1219 1740 5776 JMP I GLPSWO /GO LOOK AT SRO TO LOOP ON INSTR
1220
1221
1222 1741 0000 HLTOPR, 0
1223 1742 4750 JMS I C8EROR/HLT
1224 1743 7300 CLA CLL
1225 1744 5741 JMP I HLTOPR
1226
1227
1228
1229 1746 *1746
1230
1231 1746 4747 FCHNG, JMS I FRERNG

```

```

1232 /
1233 1747 0202 FREFWC, CHANGE
1234 1750 4401 CBEROR, XBERR
1235 1751 0504 MRIPNI, RETHR
1236 1752 2675 OPRPNT, OPRRIT
1237 1753 0776 ZJMSRT, JMSRRT
1238 1754 0746 EINSTP, INSTR
1239 1755 1401 BPANDY, RANDY
1240 1756 0200 STAKT, BGN
1241 1757 0753 INTMQD, MQDATA
1242 1760 1146 AASAVA, ASAVA
1243 1761 1147 AASAVB, ASAVB
1244 1762 0752 XDATAH, DATAHR
1245 1763 2752 XSIMAC, SIMAC
1246 1764 0755 XSVLNK, SAVLNK
1247 1765 2753 XSMLNK, SIMLNK
1248 1766 2754 XSIMMQ, SIMMQ
1249 1767 2751 XEXPRT, EXPRET
1250 1770 2001 XSMOP1, SIMOP1
1251 1771 2201 XSMOP2, SIMOP2
1252 1772 2252 XSMQI, SIMMQI
1253 1773 0745 QADDRS, ADDRS
1254 1774 2705 XRTOPF, RETTOP
1255 1775 2750 GMQDON, MQDON
1256 1776 2737 GLPSWO, LPSWO
1257 1777 0000 0
1258 /
1259
1260
1261 2000 *2000
1262 /
1263 2000 5347 JMP GCHNG
1264 /
1265
1266 /BEGINNING OF OPERATE GROUP ONE SIMULATION
1267
1268 2001 1762 SIMOP1, TAD I CINSTR /GET THE INSTRUCTION
1269 2002 0271 AND PUS200 /IS BIT 4 SET TO CLEAR THE AC
1270 2003 7640 SZA CLA
1271 2004 3773 DCA I OSIMAC /YES,CLEAR OUT THE SIMULATED AC
1272 2005 1762 TAD I CINSTR /GET THE INSTRUCTION
1273 2006 0267 AND K100 /IS BIT 5 SET TO CLEAR THE LINK
1274 2007 7640 SZA CLA
1275 2010 3774 DCA I OSMLNK /YES,CLEAR THE SIMULATED LINK
1276 2011 1762 TAD I CINSTR /GET THE INSTRUCTION
1277 2012 0266 AND K40 /IS BIT 6 SET TO COMPLEMENT THE AC
1278 2013 7640 SZA CLA
1279 2014 4763 JMS I XSMCMA /YES GO SIMULATE A CMA
1280 2015 1762 TAD I CINSTR /GET THE INSTR
1281 2016 0265 AND K20 /IS BIT 7 SET TO COMPLEMENT THE LINK
1282 2017 7640 SZA CLA
1283 2020 4764 JMS I XSMCML /YES, GO SIMULATE A CML
1284 2021 1762 TAD I CINSTR /GET THE INSTRUCTION
1285 2022 0261 AND K1 /IS BIT 11 SET TO INCREMENT THE AC
1286 2023 7640 SZA CLA
    
```

```

1287 2024 4765 JMS I XSMIAC /YES GO SIMULATE IAC
1288 2025 1762 TAD I CINSTR /GET THE INSTRUCTION
1289 2026 0262 AND K2 /IS BIT 10 SET TO RTR OR RTL
1290 2027 7640 SZA CLA
1291 2030 5242 JMP SIMTWC /YES GO CHECK TO SEE WHICH ONE
1292 2031 1762 TAD I CINSTR /GET THE INSTRUCTION
1293 2032 0264 AND K14 /IS IT A ROTATE LEFT OR RIGHT
1294 2033 1272 TAD NEG10 /RART
1295 2034 7450 SNA
1296 2035 4766 JMS I XSMRAR /YES GO SIMULATE A ROTATE RIGHT
1297 2036 1263 TAD K4 /NO,RAL?
1298 2037 7650 SNA CLA
1299 2040 4767 JMS I XSMRAL /YES,GO SIMULATE A ROTATE LEFT
1300 2041 5254 JMP OPRSET /GO TEST THE INSTRUCTION
1301
1302 2042 1762 SIMTWC, TAD I CINSTR /GET THE INSTRUCTION
1303 2043 0264 AND K14 /BIT 8 AND 9 = 0
1304 2044 7450 SNA
1305 2045 4770 JMS I XSMBSW /YES,GO SIMULATE A BYTE SWAP
1306 2046 1272 TAD NEG10 /RTR?
1307 2047 7450 SNA
1308 2050 4771 JMS I XSMRTR /YES, GO SIMULATE A ROTATE TWICE RIGHT
1309 2051 1263 TAD K4 /RTL?
1310 2052 7650 SNA CLA
1311 2053 4772 JMS I XSMRTL /YES,GO SIMULATE A ROTATE TWICE LEFT
1312
1313 2054 1776 OPRSET, TAD I OFIELD
1314 2055 1270 TAD D6201
1315 2056 3257 DCA +1
1316 2057 7402 HLT/GDF /CHANGE TO THE RANDOM DATA FIELD
1317 2060 5775 JMP I INTOPR /GO SETUP THE OPERATE INSTRUCTION
1318
1319 2061 0001 K1, 1
1320 2062 0002 K2, 2
1321 2063 0004 K4, 4
1322 2064 0014 K14, 14
1323 2065 0020 K20, 20
1324 2066 0040 K40, 40
1325 2067 0100 K100, 100
1326 2070 6201 D6201, 6201
1327 2071 0200 POS200, 200
1328 2072 7770 NEG10, -10
1329
1330 /
1331 2073 0000 PCTST, 0
1332 2074 7001 IAC
1333 2075 1754 TAD I XJMSLC
1334 2076 3755 DCA I XPCSAV
1335 2077 1755 TAD I XPCSAV
1336 2100 7041 CIA
1337 2101 1756 TAD I XRETPC
1338 2102 7640 SZA CLA
1339 2103 5761 JMP I MRIERR
1340 2104 1757 TAD I FLOXRN
1341 2105 7041 CIA
    
```

```

1342 2106 1760 TAD I FLXRET
1343 2107 7640 SZA CLA
1344 2110 5761 JMP I MHIERR
1345 2111 5673 JMP I PCTST
1346
1347 2112 0000 /
GETWD, 0
1348 2113 1757 TAD I FLDXRN
1349 2114 1344 TAD A6201
1350 2115 3316 DCA .+1
1351 2116 7402 HLT/CFD
1352 2117 1745 TAD I BSAVA
1353 2120 3345 DCA BSAVA
1354 2121 6224 RTF
1355 2122 1344 TAD A6201
1356 2123 3324 DCA .+1
1357 2124 7402 HLT/CFD
1358 2125 1345 TAD BSAVA
1359 2126 5712 JMP I GETWD
1360
1361
1362 2127 0000 /
RANCON, 0
1363 2130 1752 TAD I ZCNFLG
1364 2131 7650 SNA CLA
1365 2132 5727 JMP I RANCON
1366 2133 1753 TAD I XWDMOV
1367 2134 7402 HLT
1368 2135 7604 LAS
1369 2136 5727 JMP I RANCON
1370
1371
1372 2137 0000 INTSET, 0
1373 2140 1351 TAD XINT
1374 2141 3743 DCA I TINT
1375 2142 5737 JMP I INTSET
1376 2143 0003 TINT, INT
1377 /
1378
1379 2144 6201 A6201, 6201
1380 2145 0000 BSAVA, 0
1381 /
1382
1383 /
1384 2147 *2147
1385 /
1386 2147 4750 GCHNG, JMS I GRERNG
1387 /
1388 2150 0202 GRERNG, CHANGE
1389
1389 2151 3103 XINT, INTERS
1390 2152 1143 ZCNFLG, CONFLG
1391 2153 5173 XWDMOV, MOVWDX
1392 2154 0762 XJMSLC, JMSLOC
1393 2155 1311 XPCSAV, PCSAVE
1394 2156 0504 XRETPC, RETHR
1395 2157 0754 FLDXRN, RANFLD
1396 2160 0525 FLXRET, RETFLO
    
```

```

1397 2161 1313 MHIERR, ERROR
1398 2162 0746 CINSTR, INSTR
1399 2163 2401 XSMCMA, SIMCMA
1400 2164 2426 XSMCML, SIMCML
1401 2165 2435 XSMIAC, SIMIAC
1402 2166 2442 XSMRAR, SIMRAR
1403 2167 2461 XSMRAL, SIMRAL
1404 2170 2477 XSMBSW, SIMBSW
1405 2171 2520 XSMRTP, SIMRTR
1406 2172 2537 XSMRTL, SIMRTL
1407 2173 2752 OSIMAC, SIMAC
1408 2174 2753 OSMLNK, SIMLNK
1409 2175 0637 INTOPR, OPRINT
1410 2176 0754 OFIELD, RANPLD
1411 2177 0000 0
1412
1413 2200 *2200
1414 /
1415 2200 5343 JMP HCHNG
1416 /
1417
1418
1419 /BEGINNING OF OPERATE GROUP 2 SIMULATION
1420
1421 2201 3251 SIMOP2, DCA SKPFLG /CLEAR THE SKIP FLAG
1422 2202 1745 SMACHK, TAD I DINSTR
1423 2203 0243 AND Z100
1424 2204 7650 SNA CLA
1425 2205 5211 JMP SZACHK
1426 2206 4746 JMS I XSMSMA
1427 2207 5211 JMP SZACHK
1428 2210 5226 JMP SETSKP
1429 2211 1745 SZACHK, TAD I DINSTR
1430 2212 0244 AND Z40
1431 2213 7650 SNA CLA
1432 2214 5220 JMP SNLCHK
1433 2215 4747 JMS I XSMSZA
1434 2216 5220 JMP SNLCHK
1435 2217 5226 JMP SETSKP
1436 2220 1745 SNLCHK, TAD I DINSTR
1437 2221 0245 AND Z20
1438 2222 7650 SNA CLA
1439 2223 5227 JMP COMCHK
1440 2224 4750 JMS I XSMSNL
1441 2225 5227 JMP COMCHK
1442 2226 2251 SETSKP, ISZ SKPFLG
1443 2227 1745 COMCHK, TAD I DINSTR
1444 2230 0250 AND POS10
1445 2231 7640 SZA CLA
1446 2232 7240 CLA CMA
1447 2233 1251 TAD SKPFLG
1448 2234 7640 SZA CLA
1449 2235 2751 ISZ I ZEXPRT
1450 2236 1745 TAD I DINSTR
1451 2237 0246 AND Z200
    
```

```

1452 2740 7640          52A  CLA
1453 2741 3757          DCA  I  XACSIM
1454 2742 5753          JMP  I  ZSETUP          /GO SETUP AND TEST INSTR
1455
1456 2243 0100          Z100, 100
1457 2244 0040          Z40, 40
1458 2245 0020          Z20, 20
1459 2246 0200          Z200, 200
1460 2247 0320          Z320, 320
1461 2250 0010          POS10, 10
1462 2251 0000          SKPFLG, 0
1463
1464          /BEGINNING OF OPERATE GROUP 2 MQ INSTRUCTION SIMULATION
1465
1466 2252 1745          SIMMQL, TAD  I  DINSTR  /GET THE INSTRUCTION
1467 2253 0247          AND  ZJ20  /MASK OUT FOR LEGAL BITS 4,5 & 7
1468 2254 7450          SNA
1469 2255 5753          JMP  I  ZSETOP  /INSTRUCTION IS A NOP
1470 2256 1304          TAD  NEG20  /SUBTRACT 20
1471 2257 7450          SNA
1472 2260 5754          JMP  I  XSMQL  /GO SIMULATE A MQL
1473 2261 1305          TAD  M60
1474 2262 7450          SNA
1475 2263 5755          JMP  I  XSMQA  /GO SIMULATE A MQA
1476 2264 1304          TAD  NEG20
1477 2265 7450          SNA
1478 2266 5756          JMP  I  XSMSWP /GO SIMULATE A SWP
1479 2267 1305          TAD  M60
1480 2270 7450          SNA
1481 2271 5757          JMP  I  XSMCLA /GO SIMULATE A CLA
1482 2272 1304          TAD  NEG20
1483 2273 7450          SNA
1484 2274 5760          JMP  I  XSMCAM /GO SIMULATE A CAM
1485 2275 1305          TAD  M60
1486 2276 7450          SNA
1487 2277 5761          JMP  I  XSMACL /GO SIMULATE A ACL
1488 2300 1304          TAD  NEG20
1489 2301 7650          SNA  CLA
1490 2302 5762          JMP  I  XCLSWP /GO SIMULATE A SWP,CLA
1491 2303 7402          HLT
1492 2304 7760          /NONE OF THE ABOVE
1493 2305 7720          NEG20, -20
1494          M60, -60
1495
1496          /
1496 2306 1763          ERROPR, TAD  I  GRANFL /GET THE RANDOM DATA FIELD
1497 2307 4764          JMS  I  OPRHLT
1498 2310 1765          TAD  I  GOPRET  /GET THE ACTUAL RETURN FIELD
1499
1499 2311 4764          JMS  I  OPRHLT
1500 2312 1751          TAD  I  ZEXPRT /GET THE EXPECTED RETURN PC
1501 2313 4764          JMS  I  OPRHLT
1502 2314 1766          TAD  I  GACTRT  /GET THE ACTUAL RETURN PC
1503 2315 4764          JMS  I  OPRHLT
1504 2316 1767          TAD  I  GADDRS /GET THE INSTRUCTION ADDRESS
1505 2317 4764          JMS  I  OPRHLT
1506 2320 1745          TAD  I  DINSTR  /GET THE INSTRUCTION

```

```

1507 2321 4764          JMS  I  OPRHLT
1508 2322 1770          TAD  I  GDATAH /GET THE INITIAL AC DATA
1509 2323 4764          JMS  I  OPRHLT
1510 2324 1752          TAD  I  XACSIM  /GET THE SIMULATED AC
1511 2325 4764          JMS  I  OPRHLT
1512 2326 1771          TAD  I  GDATAD /GET THE FINAL AC
1513 2327 4764          JMS  I  OPRHLT
1514 2330 1772          TAD  I  GSVLNK /GET THE INITIAL LINK
1515 2331 4764          JMS  I  OPRHLT
1516 2332 1773          TAD  I  GSMLNK /GET THE SIMULATED LINK
1517 2333 4764          JMS  I  OPRHLT
1518 2334 1774          TAD  I  GLNKDN /GET THE FINAL LINK
1519 2335 4764          JMS  I  OPRHLT
1520 2336 1775          TAD  I  GMQDAT /GET THE INITIAL MQ DATA
1521 2337 4764          JMS  I  OPRHLT
1522 2340 5776          JMP  I  OPRER1
1523
1524
1525
1526          2343          *2343
1527
1528 2343 4744          HCHNG, JMS  I  HRERNG
1529
1530 2344 0202          HRERNG, CHANGE
1531 2345 0746          DINSTR, INSTR
1532 2346 2611          XBSMA, SIMSMA
1533 2347 2601          XBSZA, SIMSZA
1534 2350 2626          XBSNL, SIMSNL
1535 2351 2751          ZEXPRT, EXPRET
1536 2352 2752          XACSIM, SIMAC
1537 2353 2054          ZSETOP, OPRSET
1538 2354 2640          XSMQL, SIMMQL
1539 2355 2644          XSMQA, SIMMQA
1540 2356 2652          XSMSWP, SIMSWP
1541 2357 2661          XSMCLA, SIMCLA
1542 2360 2663          XSMCAM, SIMCAM
1543 2361 2666          XSMACL, SIMACL
1544 2362 2671          XCLSWP, CLASWP
1545 2363 0754          GRANFL, RANFLD
1546 2364 1741          OPRHLT, HLTOPR
1547 2365 2747          GOPRET, OPRETF
1548 2366 2675          GACTRT, OPRETF
1549 2367 0745          GADDRS, ADDR5
1550 2370 0752          GDATAH, DATAHR
1551 2371 2745          GDATAD, DATADN
1552 2372 0755          GSVLNK, SAVLNK
1553 2373 2753          GSMLNK, SIMLNK
1554 2374 2746          GLNKDN, LINKDN
1555 2375 0753          GMQDAT, MQDATA
1556 2376 1734          OPRER1, OPERR1
1557 2377 0000          0
1558
1559          2400          *2400
1560
1561 2400 5367          JMP  ICHNG

```

```

1562 /
1563
1564 /ROUTINE TO SIMULATE A COMPLEMENT
1565
1566 2401 0000 SIMCHA, 0
1567 2402 1224 TAD M14
1568 2403 3225 DCA CNT
1569 2404 3226 DCA SIMCML
1570 2405 1775 TAD I BSIMAC
1571 2406 7104 CLL RAL
1572 2407 3235 DCA SIMIAC
1573 2410 7420 SNL
1574 2411 2226 ISZ SIMCML
1575 2412 1226 TAD SIMCML
1576 2413 2225 ISZ CNT
1577 2414 5216 JMP ,+2
1578 2415 5222 JMP ENDCNA
1579 2416 7104 CLL RAL
1580 2417 3226 DCA SIMCML
1581 2420 1235 TAD SIMIAC
1582 2421 5206 JMP SIMCHA+5
1583 2422 3775 ENDCNA, DCA I BSIMAC
1584 2423 5601 JMP I SIMCHA
1585
1586 2424 7764 M14, -14
1587 2425 0000 CNT, 0
1588
1589 /ROUTINE TO SIMULATE A CML
1590
1591 2426 0000 SIMCML, 0
1592 2427 1776 TAD I BSMLNK /GET THE SIMULATED LINK
1593 2430 7650 SNA CLA /IS IT A 0
1594 2431 1234 TAD K4000 /YES, MAKE IT A ONE
1595 2432 3776 DCA I BSMLNK /SAVE IT
1596 2433 5626 JMP I SIMCML
1597
1598 2434 4000 K4000, 4000
1599
1600 /ROUTINE TO SIMULATE A IAC
1601
1602 2435 0000 SIMIAC, 0
1603 2436 2775 ISZ I BSIMAC /BUMP THE SIMULATED AC
1604 2437 5241 JMP ,+2
1605 2440 4226 JMS SIMCML
1606 2441 5635 JMP I SIMIAC
1607
1608 /ROUTINE TO SIMULATE A RAR
1609
1610 2442 0000 SIMRAR, 0
1611 2443 7300 CLA CLL /CLEAR OUT A LINK AND THE AC
1612 2444 1776 TAD I BSMLNK /GET THE SIMULATED LINK
1613 2445 7004 RAL RAL /PUT IT IN THE LINK
1614 2446 1260 TAD NEG6
1615 2447 3225 DCA CNT
1616 2450 1775 TAD I BSIMAC /GET THE SIMULATED AC

```

```

1617 2451 7006 RTL /ROTATE 12 PLACES TO THE LEFT
1618 2452 2225 ISZ CNT
1619 2453 5251 JMP ,+2
1620 2454 3775 DCA I BSIMAC /SAVE THE SIMULATED ROTATE
1621 2455 7010 RAR RAR /GET THE LINK
1622 2456 3776 DCA I BSMLNK /SAVE THE LINK
1623 2457 5642 JMP I SIMRAR /RETURN
1624
1625 2460 7772 NEG6, -6
1626
1627 /ROUTINE TO SIMULATE A RAL
1628
1629 2461 0000 SIMRAL, 0
1630 2462 7300 CLA CLL
1631 2463 1776 TAD I BSMLNK /GET THE SIMULATED LINK
1632 2464 7004 RAL RAL /PUT IT IN THE LINK
1633 2465 1260 TAD NEG6
1634 2466 3225 DCA CNT
1635 2467 1775 TAD I BSIMAC /GET THE SIMULATED AC
1636 2470 7012 RTR RTR /ROTATE IT RIGHT 12 TIMES
1637 2471 2225 ISZ CNT
1638 2472 5270 JMP ,+2
1639 2473 3775 DCA I BSIMAC /SAVE THE SIMULATED ROTATE
1640 2474 7010 RAR RAR
1641 2475 3775 DCA I BSMLNK /SAVE THE SIMULATED LINK
1642 2476 5661 JMP I SIMRAL /RETURN
1643
1644 /ROUTINE TO SIMULATE A BYTE SWAP
1645
1646 2477 0000 SIMBSW, 0
1648 2400 7300 CLA CLL
1649 2401 1776 TAD I BSMLNK /GET THE SIMULATED LINK
1650 2402 7010 RAR RAR
1651 2403 7012 RTR RTR
1652 2404 7012 RTR RTR
1653 2405 1775 TAD I BSIMAC /GET THE SIMULATED AC
1654 2406 0317 AND K7700
1655 2407 1775 TAD I BSIMAC /GET IT AGAIN
1656 2410 7006 RTL
1657 2411 7006 RTL
1658 2412 7006 RTL
1659 2413 3775 DCA I BSIMAC /SAVE THE SIMULATED BYTE SWAP
1660 2414 7010 RAR RAR
1661 2415 3776 DCA I BSMLNK /SAVE THE LINK
1662 2416 5677 JMP I SIMBSW /RETURN
1663 2417 7700 K7700, 7700
1664
1665 /ROUTINE TO SIMULATE RTR
1666
1667 2420 0000 SIMRTR, 0
1668 2421 7300 CLA CLL
1669 2422 1776 TAD I BSMLNK /GET THE SIMULATED LINK
1670 2423 7004 RAL RAL /PUT IT IN THE LINK
1671 2424 1336 TAD M13

```



```

1672 2525 3225 UCA CNT /SETUP A COUNTER FOR 11 RAL'S
1673 2526 1775 TAD I BSIMAC /GET THE SIMULATED AC
1674 2527 7004 RAL
1675 2530 2225 ISZ CNT
1676 2531 5327 JMP ,+2
1677 2532 3775 GCA I BSIMAC /SAVE THE SIMULATED ROTATED AC
1678 2533 7010 RAR
1679 2534 3776 DCA I BSIMAC /SAVE THE SIMULATED LINK
1680 2535 5720 JMP I SIMRTR /RETURN
1681
1682 2536 7765 M13, -13
1683
1684 /ROUTINE TO SIMULATE RTL
1685
1686 2537 0000 SIMRTL, 0
1687 2540 7300 CLA CLL
1688 2541 1776 TAD I BSIMLNK /GET THE SIMULATED LINK
1689 2542 7004 RAL /AND PUT IT IN THE LINK
1690 2543 1336 TAD M13
1691 2544 3225 DCA CNT /SET UP A COUNTER TO DO 11 RAR'S
1692 2545 1775 TAD I BSIMAC /GET THE SIMULATED AC
1693 2546 7010 RAR
1694 2547 2225 ISZ CNT
1695 2550 5346 JMP ,+2
1696 2551 3775 DCA I BSIMAC /SAVE THE SIMULATED ROTATED AC
1697 2552 7010 RAR
1698 2553 3776 DCA I BSIMLNK /SAVE THE SIMULATED ROTATED LINK
1699 2554 5737 JMP I SIMRTL
1700
1701 /
1702 2555 1771 /FLDCHK, TAD I UPPERL /IS THE LAST FIELD = 2K
1703 2556 1365 TAD M3777
1704 2557 7640 SZA CLA
1705 2560 5772 JMP I TSFLDF /NO GO SWAP IT UP
1706 2561 6224 RIF /READ THE INSTRUCTION FIELD
1707 2562 7640 SZA CLA /IS IT EQUAL TO FIELD 0
1708 2563 5773 JMP I TSWPDN /NO, GO SWAP THE PROGRAM DOWN
1709 2564 5774 JMP I ROLFLG /YES, DO NOT SWAP FIELDS BUT ROLL UP
1710
1711 2565 4001 M3777, -3777
1712
1713 /
1714 2567 *2567
1715 /
1716 2567 4770 ICHNG, JMS I IRERNG
1717 /
1718 2570 0202 IRERNG, CHANGE
1719
1720 2571 1550 UPPERL, UPRLIM
1721 2572 0433 TSFLDF, SFLOFG
1722 2573 0422 TSWPDN, SWAPDN
1723 2574 0324 ROLFLG, SETFLG
1724 2575 2752 BSIMAC, SINAC
1725 2576 2753 BSIMLNK, SIMLNK
1726 2577 0000 0
    
```

```

1727
1728 2600 *2600
1729
1730 /
1731 2600 5364 JMP JCHNG
1732 /
1733
1734 /ROUTINE TO SIMULATE A SZA
1735
1736 2601 0000 SIMSZA, 0
1737 2602 7240 CLA CMA
1738 2603 1770 TAD I CSIMAC
1739 2604 3223 DCA ACUTST
1740 2605 2223 ISZ ACUTST
1741 2606 5601 JMP I SIMSZA
1742 2607 2201 ISZ SIMSZA
1743 2610 5601 JMP I SIMSZA
1744
1745 /ROUTINE TO SIMULATE A SMA
1746
1747 2611 0000 SIMSMA, 0
1748 2612 1770 TAD I CSIMAC
1749 2613 0225 AND D4000
1750 2614 1224 TAD K7777
1751 2615 3223 DCA ACUTST
1752 2616 2223 ISZ ACUTST
1753 2617 5221 JMP ,+2
1754 2620 5611 JMP I SIMSMA
1755 2621 2211 ISZ SIMSMA
1756 2622 5611 JMP I SIMSMA
1757
1758 2623 0000 ACUTST, 0
1759 2624 7777 K7777, -1
1760 2625 4000 D4000, 4000
1761
1762 /ROUTINE TO SIMULATE A SNL
1763
1764 2626 0000 SIMSNL, 0
1765 2627 1771 TAD I CSMLNK
1766 2630 0225 AND D4000
1767 2631 1224 TAD K7777
1768 2632 3223 DCA ACUTST
1769 2633 2223 ISZ ACUTST
1770 2634 5236 JMP ,+2
1771 2635 5626 JMP I SIMSNL
1772 2636 2226 ISZ SIMSNL
1773 2637 5626 JMP I SIMSNL
1774
1775 /ROUTINE TO SIMULATE A MQL
1776
1777 2440 1770 SIMMQL, TAD I CSIMAC /GET THE SIMULATED AC
1778 2441 3772 DCA I CSIMMQL /PUT IT IN THE SIMULATED MQL
1779 2442 3770 DCA I CSIMAC /CLEAR OUT THE SIMULATED AC
1780 2443 5773 JMP I COPRST /GO EXECUTE THE INSTRUCTION
1781
    
```

```

1782 /ROUTINE TO SIMULATE MCA
1783
1784 2644 1772 SIMMGA, TAD I CSIMMQ /GET THE SIMULATED MQ
1785 2645 7040 CMA /COMPLEMENT THE RESULTS
1786 2646 0770 AND I CSIMAC /MASK RESULTS WITH SIMULATED AC
1787 2647 1772 TAD I CSIMMQ /INCLUSIVE OR THE SIMULATED MQ
1788 2650 3770 DCA I CSIMAC /THE SIMULATED AC = INCLUSIVE OR OF MQ & AC
1789 2651 5773 JMP I COPRST /GO EXECUTE THE INSTR.
1790
1791 /ROUTINE TO SIMULATE A SWP
1792
1793 2652 1770 SIMSWP, TAD I CSINAC /GET THE SIMULATED AC
1794 2653 3223 DCA ACUTST /AND SAVE IT
1795 2654 1772 TAD I CSIMMQ /GET THE SIMULATED MQ
1796 2655 3770 DCA I CSINAC /AND PUT IT IN THE SIMULATED AC
1797 2656 1223 TAD ACUTST /GET THE SIMULATED AC
1798 2657 3772 DCA I CSIMMQ /AND PUT IT IN THE SIMULATED MQ
1799 2660 5773 JMP I COPRST /GO EXECUTE THE INSTRUCTION
1800
1801 /ROUTINE TO SIMULATE A CLA
1802
1803 2661 3770 SIMCLA, DCA I CSINAC /CLEAR THE SIMULATED AC
1804 2662 5773 JMP I COPRST /GO EXECUTE THE INSTRUCTION
1805
1806 /ROUTINE TO SIMULATE A CAM
1807
1808 2663 3770 SIMCAM, DCA I CSINAC /CLEAR THE SIMULATED AC
1809 2664 3772 DCA I CSIMMQ /CLEAR THE SIMULATED MQ
1810 2665 5773 JMP I COPRST /GO EXECUTE THE INSTRUCTION
1811
1812 /ROUTINE TO SIMULATE A ACL
1813
1814 2666 1772 SIMACL, TAD I CSIMMQ /GET THE SIMULATED MQ
1815 2667 3770 DCA I CSIMAC /PUT IT IN THE SIMULATED AC
1816 2670 5773 JMP I COPRST /GO EXECUTE THE INSTR
1817
1818 /ROUTINE TO SIMULATE A CLA,SWP
1819
1820 2671 1772 CLASWP, TAD I CSIMMQ /GET THE SIMULATED MQ
1821 2672 3770 DCA I CSINAC /PUT IT IN THE SIMULATED AC
1822 2673 3772 DCA I CSIMMQ /CLEAR THE SIMULATED MQ
1823 2674 5773 JMP I COPRST /GO EXECUTE THE INSTRUCTION
1824
1825 /
1826 2675 0000 OPRRET, 0 /RETURN HERE FROM OPERATE INSTRUCTION
1827 2676 3345 DCA DATADN /SAVE THE FINAL AC
1828 2677 7010 RAR
1829
1830 2700 3346 DCA LINKDN /SAVE THE FINAL LINK
1831 2701 6214 RDF /GET THE RANDOM DATA FIELD
1832 2702 3347 DCA OPRETF /SAVE THE DF FROM OPERATE RETURN
1833 2703 7701 ACL /GET THE FINAL MQ DATA
1834 2704 3350 DCA MQDONE /SAVE IT
1835 2705 7402 RETIDF, HLT/CDP /CHANGE DF BACK TO PROGRAM FIELD
1836 2706 1775 OPRCOM, TAD I XFIELD /GET THE EXPECTED INSTRUCTION FIELD
1837 2707 7041 CIA
    
```

```

1837 2710 1347 TAD OPRETF /GET THE FIELD INSTRUCTION RETURNED FROM
1838 2711 7640 SZA CLA
1839 2712 5344 JMP OPRETT /PROGRAM RETURNED FROM THE WRONG FIELD
1840 2713 1351 TAD EXPRET /GET THE EXPECTED RETURN PC
1841 2714 7041 CIA
1842 2715 1275 TAD OPRRET /GET THE ACTUAL RETURN PC
1843 2716 7640 SZA CLA
1844 2717 5344 JMP OPRETT /EXPECTED PC DOES NOT AGREE WITH ACTUAL
1845 2720 1354 TAD SIMMQ /GET THE SIMULATED MQ
1846 2721 7041 CIA
1847 2722 1350 TAD MQDONE /GET THE ACTUAL MQ
1848 2723 7640 SZA CLA
1849 2724 5344 JMP OPRETT /ERROR, ACTUAL MQ DOES NOT EQUAL SIMULATED MQ
1850 2725 1353 TAD SIMLNK /GET THE SIMULATED LINK
1851 2726 7041 CIA
1852 2727 1346 TAD LINKDN /GET THE ACTUAL LINK
1853 2730 7640 SZA CLA
1854 2731 5344 JMP OPRETT /ERROR SIMULATED AND ACTUAL LINK ARE NOT EQUAL
1855 2732 1352 TAD SIMAC /GET THE SIMULATED AC
1856 2733 7041 CIA
1857 2734 1345 TAD DATADN /GET THE ACTUAL AC RETURNED
1858 2735 7640 SZA CLA
1859 2736 5344 JMP OPRETT /SIMULATED AND ACTUAL AC DO NOT AGREE
1860 2737 4766 LPSWO, JMS I CBSWR /IS SRI SET TO LOOP ON THE INSTRUCTION
1861 2740 7004 RAL
1862 2741 7710 SPA CLA
1863 2742 5774 JMP I XINSGN /YES GO LOOP ON THE INSTRUCTION
1864 2743 5776 JMP I XXCNT /GO BUMP INSTRUCTION COUNTER
1865
1866 2744 5767 OPRERR, JMP I OERROR /NO, GO HALT WITH ERROR INFORMATION IN AC
1867
1868 2745 0000 DATADN, 0
1869 2746 0000 LINKDN, 0
1870 2747 0000 OPRETF, 0
1871 2750 0000 MQDONE, 0
1872 2751 0000 EXPRET, 0
1873 2752 0000 SIMAC, 0
1874 2753 0000 SIMLNK, 0
1875 2754 0000 SIMMQ, 0
1876 /
1877 2764 /*2764
1878 /
1879 2764 4765 JCHNG, JMS I JRRNG
1880 /
1881 2765 0202 JRRNG, CHANGE
1882 2766 3466 CBSWR, XCSWR
1883 2767 2306 OERROR, ERROPR
1884 2770 2752 CSIMAC, SIMAC
1885 2771 2753 CSMLNK, SIMLNK
1886 2772 2754 CSIMMQ, SIMMQ
1887 2773 2054 COPRST, OPRSET
1888 2774 1606 XINSGN, INSGEN+5
1889 2775 0754 XFIELD, RANFLD
1890 2776 0271 XXCNT, XCNT
1891 2777 0000 /
    
```

```

1892
1893
1894          3000      *3000
1895
1896      3000  5347      /
1897                      JMP KCHNG
1898
1899      /ROUTINE TO FILL MEMORY WITH HALTS AROUND THE PROGRAM
1900
1901      3001  1766      FILRND, TAD I CONTLM /GET THE UPPER LIMIT COUNTER
1902      3002  3224      DCA TEMP /SAVE IT
1903      3003  1767      TAD I FLD CNT /CHECK TO SEE IF IT IS FIELD 0
1904      3004  7650      SNA CLA /IS IT FIELD 0?
1905      3005  1223      TAD MM4 /YES, SUBTRACT 4 FROM THE BEGINNING ADDRESS
1906      3006  1370      TAD ABGN
1907      3007  7041      CIA /NEGATE THE NUMBER FOR A COUNTER
1908      3010  3766      DCA I CONTLM /SAVE IT
1909      3011  1767      TAD I FLD CNT /CHECK TO SEE IF IT IS FIELD 0
1910      3012  7650      SNA CLA /IS IT FIELD 0?
1911      3013  7307      CLA CLL IAC RTL /YES, START FILLING FIELD 0 AT ADDRESS 4
1912      3014  4771      JMS I ZFILL /FILL THE FIRST HALF OF PROGRAM FIELD
1913      3015  1224      TAD TEMP /GET THE UPPER LIMIT COUNTER
1914      3016  1372      TAD ENDOFP /ADD END OF PROGRAM TO IT
1915      3017  3766      DCA I CONTLM /SAVE THIS NUMBER AS THE COUNTER
1916      3020  1372      TAD ENDOFP /GET THE ADDRESS TO START FILLING MEMORY
1917      3021  4771      JMS I ZFILL /WITH HALTS
1918      3022  5773      JMP I XADD1 /RETURN FOR NEXT FIELD
1919
1920      3023  7774      MM4, -4
1921      3024  0000      TEMP, 0
1922
1923      3025  7240      BGNCON, CLA CMA /CONSTRAINT STARTING ADDRESS
1924      3026  3753      DCA I XCNFLG /RANDOM STARTING ADDRESS
1925      3027  4754      JMS I XLIMIT /SETUP MEMORY LIMITS
1926      3030  1021      TAD OPISEL
1927      3031  3235      DCA SELOP1
1928      3032  4776      CL8BGN, JMS I CBSMS /CHECK FOR CLASIC OR TO CHANGE SR
1929      3033  3751      DCA I SETDOX /SET DOSET TO 0
1930      3034  5762      JMP I CONSET /RETURN TO PROGRAM TO SETUP MASK AND CONSTRAINT WORDS
1931
1932      3035  0000      SELOP1, 0
1933
1934      3036  0000      SETOP1, 0
1935      3037  1235      TAD SELOP1
1936      3040  7001      PAL /CHECK TO SEE IF OPTION 1 WAS SELECTED
1937      3041  7700      SMA CLA /WAS IT 1?
1938      3042  5636      JMP I SETOP1 /NO, JUST RUN MRI AND OPR
1939
1940      3043  3301      DCA KILL
1941      3044  7240      CLA CMA /SET ALL PROGRAM FLAGS TO INACTIVE STATE
1942      3045  3757      DCA I FLGXMT /SLU XMIT FLAG
1943      3046  7240      CLA CMA
1944      3047  3760      DCA I FLGRTC /REAL TIME CLOCK FLAG
1945      3050  1300      TAD MM55
1946      3051  3761      DCA I DVINAC /SETUP A DEVICE INACTIVE COUNTER
1947      3052  7301      CLA CLL IAC /SET DATA 11 TO A ONE
    
```

```

1947      3053  6035      KIE /SET SLU INTERRUPT ENABLE
1948      3054  6135      CLLE /SET RTC INTERRUPT ENABLE
1949      3055  7300      CLA CLL
1950      3056  1302      TAD FIRST /GET FIRST TIME IN FLAG
1951      3057  7650      SNA CLA /WAS IT SET
1952      3060  2302      ISZ FIRST /YES SET IT
1953      3061  1751      TAD I SETDOX /GET END OF PASS COUNTER
1954      3062  7650      SNA CLA /PRINTED END OF PASS?
1955      3063  5270      JMP .+5 /YES RESET TRANSMIT WORD AND COUNTERS
1956      3064  2756      ISZ I SLUXMT /INCREMENT TRANSMIT WORD
1957      3065  2755      ISZ I XTYCNT /INCREMENT TRANSMIT COUNTER
1958      3066  5275      JMP .+7 /GO TRANSMIT THE CHAR
1959      3067  4752      JMS I STRNWL /PRINT A CR LF
1960      3070  1344      TAD K237
1961      3071  3756      DCA I SLUXMT /SAVE THE XMIT WORD
1962      3072  1345      TAD M101
1963      3073  3755      DCA I XTYCNT /SAVE THE COUNTER
1964      3074  5264      JMP .-10 /GO INCREMENT AND PRINT
1965      3075  1756      TAD I SLUXMT /GET THE WORD TO BE TRANSMITTED BY SLU
1966      3076  6046      TDS /CLEAR XMIT FLAG AND TRANSMIT WORD
1967      3077  5636      JMP I SETOP1 /RETURN TO PROGRAM
1968
1969      3100  7723      MM55, -55
1970      3101  0000      KILL, 0
1971      3102  0000      FIRST, 0
1972
1973      /INTERRUPT SERVICE ROUTINE
1974
1975      3103  3337      INTERS, DCA INTAC /SAVE THE AC
1976      3104  7010      RAR /GET THE LINK INTO BIT 0
1977      3105  3340      DCA INTLNK /SAVE THE LINK
1978      3106  1741      TAD I ADDR0 /GET THE INTERRUPT PC
1979      3107  3342      DCA INTRET /SAVE IT
1980      3110  6224      RIF /READ THE INSTRUCTION FIELD
1981      3111  1343      TAD KKCDF /ADD CDF INSTRUCTION TO BITS 6-8
1982      3112  3313      DCA .+1 /PUT CDF TO PROGRAM FIELD IN NEXT LOCATION
1983      3113  7402      HLT/CFD /TO PROGRAM FIELD
1984      3114  6041      TSF /SKIP ON SLU XMIT FLAG
1985      3115  7410      SKP
1986      3116  5764      JMP I SERXMT /GO SERVICE SLU XMIT FLAG
1987      3117  4774      JMS I CBCHKR /CHECK FOR CLASIC CONTROL CHARACTER
1988      3120  7610      SKP CLA /NOT CLASIC OR RECEIVE FLAG NOT SET
1989      3121  5764      JMP I SERXMT /CHECK TO SEE IF OPTION 1 WAS SELECTED
1990      3122  6137      CLSK /SKIP ON REAL TIME CLOCK FLAG
1991      3123  7410      SKP
1992      3124  5765      JMP I SERRTC /GO SERVICE REAL TIME CLOCK FLAG
1993      3125  6102      SPL /SKIP ON AC LOW F/F
1994      3126  7410      SKP
1995      3127  5763      JMP I POWERF /POWER FAILURE GO CLEAR AC LOW AND RETURN
1996      3130  4775      JMS I CBERRR /ILLEGAL INTERRUPT
1997      3131  1340      RETPRG, TAD INTLNK /GET THE LINK
1998      3132  7104      CLL RAL /RESTORE IT
1999      3133  1337      TAD INTAC /RESTORE THE AC
2000      3134  6244      RMP /RESTORE MEMORY FIELDS
2001      3135  6001      ION /TURN THE INTERRUPT ON
    
```

```

2002 3136 5742          JMP I INTRET /RETURN TO PROGRAM
2003
2004 3137 0000 INTIAC, 0
2005 3140 0000 INTLNK, 0
2006 3141 0000 ADDR60, 0
2007 3142 0000 INTRET, 0
2008 3143 6201 KKCDF, CDF      00
2009 3144 0237 K237, 237
2010 3145 7677 M101, -101
    
```

```

2011
2012          /
2013          3147 *3147
2014          /
2015 3147 4750 KCHNG, JMS I KRERNG
2016          /
2017 3150 0202 KRERNG, CHANGE
2018 3151 3454 SETDOX, DOSET
2019 3152 4224 STRNWL, XCBCRL
2020 3153 1143 XCNFLG, CONFLG
2021 3154 1522 XLIMIT, LIMITS
2022 3155 3275 XTYCNT, TTYCNT
2023 3156 3230 SLUXMT, XMTSLU
2024 3157 3226 FLGRIC, RTCPFLG
2025 3160 3227 FLGRIC, RTCPFLG
2026 3161 3273 DVINAC, INACDV
2027 3162 5001 CONSET, SETCON
2028 3163 3246 POWERF, POWFAL
2029 3164 3201 SERAMT, XMTSER
2030 3165 3232 SERRTC, RTCSER
2031 3166 1145 CONTLN, HGHLIM
2032 3167 0247 FLD CNT, CNTN3
2033 3170 0200 ABGN, BGN
2034 3171 0527 ZFILL, FILALL
2035 3172 5176 ENDUFP, PRGEND
2036 3173 5036 XADD1, ADUONE
2037 3174 4250 C8CHKR, XC8CKP
2038 3175 4401 C8ERRR, XC8ERR
2039 3176 4722 C8SMS, XSTMS
2040 3177 0000 0
2041
2042          /
2043          3200 *3200
2044          /
2045 3200 5361          JMP LCHNG
2046          /
2047          /SERIAL LINE UNIT TRANSMIT SERVICE ROUTINE
2048
2049 3201 1770 XMTSER, TAD I OP1 /CHECK TO SEE IF OPTION 1 WAS SELECTED
2050 3202 7004 RAL /MOVE THE HARDWARE BIT INTO LINK
2051 3203 7700 SMA CLA /WAS OPTION 1 SELECTED ?
2052 3204 5773 JMP I PRGRET /NO RETURN TO THE PROGRAM
2053 3205 3226 DCA XMTFLG /SET SLU XMIT FLAG ACTIVE
2054 3206 6042 TCF /CLEAR TRANSMIT FLAG
2055 3207 1766 TAD I KILLIT /GET THE KILL FLAG
2056 3210 7640 SZA CLA /INTERRUPTS STILL EXPECTED ?
2057 3211 5241 JMP OUT /NO, GO WAIT FOR LAST INTERRUPT FROM CLOCK
2058 3212 2230 ISZ XMTSLU /ADD 1 TO THE CHARACTER TO BE PRINTED
2059 3213 2275 ISZ TTYCNT /DONE A LINE
2060 3214 5222 JMP GTCHAR /NO GO TRANSMIT NEXT CHARACTER
2061 3215 1276 TAD K240 /GET THE CODE FOR A SPACE
2062 3216 3230 DCA XMTSLU /SAVE IT
2063 3217 1277 TAD NEG100 /SET UP THE LINE COUNTER
2064 3220 3275 DCA TTYCNT /SAVE THE LINE COUNTER
2065 3221 4765 JMS I CORSTL /ISSUE A CARRIAGE RETURN LINE FEED
    
```

```

2066 3222 1230 GTCHAP, TAD XMISLU /GET THE CHARACTER TO BE PRINTED
2067 3223 604A ILS /TRANSMIT IT
2068 3224 7300 CLA CLL
2069 3225 5255 JMP CHKACT /GO CHECK DEVICE TO BE ACTIVE
2070
2071 3226 0000 XMIFLG, 0
2072 3227 0000 RTCFLG, 0
2073 3230 0000 XMISLU, 0
2074 3231 0377 CC377, 377
2075
2076 /REAL TIME CLOCK INTERRUPT SERVICE ROUTINE
2077
2078 3232 3227 RTCSER, DCA RTCFLG /SET REAL TIME CLOCK FLAG TO ACTIVE
2079 3233 6136 CLCL /CLEAR CLOCK FLAG
2080 3234 7000 NOP/JMS I ACTLIN/ THIS LOCATION USED IF ACT LINE AND OPTION 1 SELECTED
2081 3235 1766 TAD I KILLIT /GET THE KILL FLAG
2082 3236 7650 SNA CLA /WAS IT SET
2083 3237 5255 JMP CHKACT /CHECK DEVICE TO BE ACTIVE
2084 3240 6135 CLLE /YES - CLEAR RTC INT ENA
2085
2086 3241 2245 OUT, ISZ CNTEND
2087 3242 5773 JMP I PRGRET /WAIT FOR NEXT INTERRUPT
2088 3243 6002 IOF /TURN THE INTERRUPT OFF
2089 3244 5767 JMP I RELGO /RETURN TO PROGRAM FOR RELOCATION OR RUN
2090
2091 3245 0000 CNTEND, 0
2092
2093 /POWER FAIL INTERRUPT SERVICE ROUTINE
2094
2095
2096 3246 6103 POWFAL, CAL /CLEAR AC LOW F/F
2097 3247 6102 SPL /SKIP ON AC LOW AS A LEVEL
2098 3250 7410 SKP
2099 3251 6101 SBE
2100 3252 7410 SKP
2101 3253 7402 HLT /BATTERY EMPTY - ITS ALL OVER
2102 3254 5773 JMP I PRGRET /RETURN TO THE PROGRAM
2103
2104 3255 1226 CHKACT, TAD XMIFLG /CHECK ALL DEVICES TO BE INTERRUPTING
2105 3256 1227 TAD RTCFLG
2106 3257 7650 SNA CLA /ARE THEY ?
2107 3260 5264 JMP RESET /YES, RESET ALL FLAGS TO INACTIVE
2108 3261 2273 ISZ INACDV /BUMP INACTIVE COUNTER
2109 3262 5773 JMP I PRGRET /RETURN TO THE PROGRAM
2110 3263 4763 JMS I ERRCS /ONE OR MORE DEVICES ARE INACTIVE
2111 3264 7340 RESET, CLA CLL CMA /SET ALL DEVICES TO INACTIVE
2112 3265 3226 DCA XMIFLG
2113
2114 3266 7240 CLA CMA
2115 3267 3227 DCA RTCFLG
2116 3270 1274 TAD MM55
2117 3271 3273 DCA INACDV /RESET INACTIVE COUNTER
2118 3272 5773 JMP I PRGRET /RETURN TO THE PROGRAM
2119
2120 3273 0000 INACDV, 0
2121 3274 7723 MM55, -55
    
```

```

2121 3275 7700 TTYCNT, -100
2122 3276 0240 K240, 240
2123 3277 7700 NEG100, -100
2124
2125 3300 0000 WAITEN, 0
2126 3301 1770 TAD I DP1 /WAS OPTION 1 SELECTED
2127 3302 7004 RAL
2128 3303 7700 SMA CLA
2129 3304 5316 JMP ,+12 /NO, RETURN TO RELOCATION
2130 3305 2766 ISZ I KILLIT
2131 3306 3331 DCA KILCN1 /CLEAR KILL COUNTER 1
2132 3307 1333 TAD M24
2133 3310 3332 DCA KILCN2 /SET UP FOR ABOUT A 300MS DELAY
2134 3311 2331 ISZ KILCN1
2135 3312 5311 JMP , -1
2136 3313 2332 ISZ KILCN2
2137 3314 5311 JMP , +3
2138 3315 4763 JMS I ERRCS /OPTION 1 SLU OR RTC FAILED TO INTERRUPT
2139 3316 6002 IOF /TURN THE INTERRUPT OFF
2140 3317 5700 JMP I WAITEN /RETURN TO PROGRAM
2141
2142
2143 3320 0000 CBGET, 0
2144 3321 7200 CLA
2145 3322 1774 TAD I MGSAY8
2146 3323 7421 MQL /RESTORE THE MG
2147 3324 1775 TAD I FLBAY8
2148 3325 7004 RAL /RESTORE THE LINK
2149 3326 7200 CLA
2150 3327 1776 TAD I ACSAY8 /RESTORE THE AC
2151 3330 5720 JMP I CBGET /GET THE REGISTERS
2152
2153
2154 3331 0000 KILCN1, 0
2155 3332 0000 KILCN2, 0
2156 3333 7754 M24, -24
2157
2158
2159 /ROUTINE TO RESTORE THE MONITOR ON A CONTROL C
2160
2161 3334 6002 RESSBOT, IOF /TURN THE INTERRUPT OFF
2162 3335 1364 TAD BUOTAD
2163 3336 3010 DCA AUTO10
2164 3337 1354 TAD K23
2165 3340 3011 DCA AUTO11
2166 3341 6224 RIF
2167 3342 1344 TAD CDFINS
2168 3343 3344 DCA ,+1
2169 3344 6201 CDFINS, CDF 00
2170 3345 1410 TAD I AUTO10
2171 3346 6201 CDF 00
2172 3347 3411 DCA I AUTO11
2173 3350 2355 ISZ M36
2174 3351 5344 JMP , -5
2175 3352 6202 CIF 00
    
```

```

2176 3153 5033 JMP 33
2177
2178 3154 0023 K23, 23
2179 3155 7747 M36, -36
2180
2181 3361 *3361
2182 /
2183 3161 4762 LCHNG, JMS I LDRNG
2184 /
2185 3162 0202 LDRNG, CHANGE
2186 3163 4401 ERRCS, XCERR
2187 3164 4625 BOOTAD, BOOTSV=1
2188 3165 4600 CBRSTL, TYPELT
2189 3166 3101 KILLIT, KILL
2190 3167 0274 RELGD, XCNT+3
2191 3170 3035 OPI, SELOP1
2192 3171 1333 PSRERR, ERRPSR+1
2193 3172 1320 ACTLIN, ERROR+5
2194 3173 3131 PRGRET, RETPRG
2195 3174 4525 HQSAV8, HQSAVE
2196 3175 4526 FLSAV8, FLSAVE
2197 3176 4524 ACSAV8, ACSAVE
2198 3177 0000 0
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222

```

/CONSOL SRC-V1R4= CONSOL PACKAGE

/SET UP A LAS TO BE EQUAL TO THE CALL C8CKSW

/PROGRAM SHOULD CHECK FOR A CONTROL CHARACTER FROM THE CONSOL
/EVERY FIVE SECONDS OR LESS

/SETUP CNTVAL FOR A RANGE OF 1 TO 4 MINUTES FOR C8PASS TO PRINT PASS
/SETUP OF CNTVAL WILL BE FOUND IN C8PASS
/THIS VALUE SHOULD BE A POSITIVE NUMBER,

/SET UP XDOSW AS THE VALUE NEEDED FOR A RETURN FOR CONTROL R
/RETURN TO ASK THE SWITCH REGISTER QUESTION,

/CHANGE 1 AND 2 APRIL 16 1975

/CHANGE 3 APRIL 18,1975

/CHANGE 4 APRIL 22 1975

/CHANGE 5 APRIL 23 1975

```

2223
2224
2225
2226
2227
2228
2229
2230

```

```

2231 /CHANGE 6 APRIL 24,1975
2232 /CHANGE 7 APRIL 1975
2233
2234
2235
2236
2237
2238
2239
2240 6661 PSKF= 6661
2241 6662 PCLF= 6662
2242 6663 PSKE= 6663
2243 6664 PSTB= 6664
2244 6665 PSIE= 6665
2245 6004 GTF= 6004
2246 7701 ACL= 7701
2247 6007 CAF= 6007
2248 7421 MGL= 7421
2249 /#6
2250 7501 MGA= 7501
2251 /#6
2252 3400 PAGE
2253
2254
2255 3400 5362 JMP C8CHG1 /GO CHANGE THE LINKS FOR RELOCATION
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271 3401 0000 XC8PAS, 0
2272 /#1
2273 3402 7200 CLA
2274 /#1
2275 3403 1256 TAD SELOP2 /CHECK IF A CLASSIC
2276 3404 0343 AND C8400 /MASK FOR CLASSIC BIT
2277 3405 7640 SZA CLA /SKIP IF NOT CLASSIC
2278 3406 5214 JMP DOPACK /IS CLASSIC
2279 3407 4767 JMS I C8CKSW /CHECK BR SETTING
2280 3410 0343 AND C8400 /FOR HALT ON END OF C8PASS
2281 3411 7640 SZA CLA /I= HALT 0 CONTINUE
2282 3412 5601 JMP I XC8PAS /GO TO HALT
2283 3413 5233 JMP C8BY1 /CONTINUE ON RUNNING PROGRAM
2284 3414 4235 DOPACK, JMS CKCOUT /CLASS CHECK C8PASS COUNT
2285 3415 5233 JMP C8BY1 /C8PASS COUNT NOT DONE REED PROGRAM

```

/CHANGE 6 APRIL 24,1975
/CHANGE 7 APRIL 1975

/THE CALL TABLE IS A CONDITIONAL ASSEMBLY.
/ TO ASSEMBLE THE CALL REMOVE THE / BEFORE CONSOL =0.
/IN COMBINING THE CONSOL PACKAGE TO A DIAGNOSTIC,
/ THE CALL TABLE IS TO BE AT THE BEGINNING OF A PROGRAM,
/CONSOL=0

PSKF= 6661
PCLF= 6662
PSKE= 6663
PSTB= 6664
PSIE= 6665
GTF= 6004
ACL= 7701
CAF= 6007
MGL= 7421

/#6
MGA= 7501

/#6
PAGE

JMP C8CHG1 /GO CHANGE THE LINKS FOR RELOCATION

/C8PASS
/THIS IS CALLED AT THE END OF EACH PROGRAM COMPLETION
/THE VALUE OF** CNTVAL** WILL BE DETERMINED BY THE TIME IT TAKES
/THE PROGRAM TO COMPLETE THIS MANY C8PASS TO BE IN THE 1 TO 4 MINUTE
/RANGE

/ C8PASS=JMS XC8PAS
/EX, C8PASS
/ HLT /HALT IF NON CONSOL PACKAGE
/ JMP START1 /CONTINUE RUNNING THIS PROGRAM
/RETURN TO LOCATION CALL PLUS ONE WITH THE AC=0 IF NON CONSOL PACKAGE AND HLT
/IF CONTINUE TO RUN THEN RETURN TO CALL PLUS2 AC=0

XC8PAS, 0

/#1

CLA

/#1

TAD SELOP2 /CHECK IF A CLASSIC
AND C8400 /MASK FOR CLASSIC BIT
SZA CLA /SKIP IF NOT CLASSIC
JMP DOPACK /IS CLASSIC
JMS I C8CKSW /CHECK BR SETTING
AND C8400 /FOR HALT ON END OF C8PASS
SZA CLA /I= HALT 0 CONTINUE
JMP I XC8PAS /GO TO HALT
JMP C8BY1 /CONTINUE ON RUNNING PROGRAM
DOPACK, JMS CKCOUT /CLASS CHECK C8PASS COUNT
JMP C8BY1 /C8PASS COUNT NOT DONE REED PROGRAM

```

2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285

```

```

2286 3416 2253      ISZ  PASCNT      /C8PASS COUNT DONE SET C8PASS COUNT
2287
2288 3417 4770      /#2      JMS  I  C8CRLF
2289
2290 3420 1366      /#2      TAD   PASMES      /GET THE ADDRESS OF THE MESSAGE
2291 3421 3223      DCA   .+2          /SAVE IT IN LOCATION FOLLOWING PRINT CALL
2292 3422 4771      JMS  I  C8PRNT     /C8PRNT BUFFER
2293 3423 3457      MESPAS
2294 3424 1253      TAD   PASCNT     /GET NUMBER
2295 3425 4772      JMS  I  C8OCT     /CONVERT IT TO ASCII
2296 3426 4770      JMS  I  C8CHLF    /DO A CARRIAGE RETURN
2297 3427 4767      JMS  I  C8CKSW    /CHECK A HALT AT END OF C8PASS
2298 3430 0343      AND   C8400      /MASK BIT
2299 3431 7640      SZA  CLA         /HALT #1 NO SKIP CONTINUE #0
2300 3432 4773      JMS  I  C8INQU    /STOP PROGRAM EXECUTION-LOOK FOR INPUT
2301 3433 2201      C8BY1, ISZ  XC8PAS /BUMP RETURN
2302 3434 5601      JMP  I  XC8PAS
2303 3435 0000      CKCOUT, 0
2304 3436 1254      TAD   DOSET      /CHECK IF SET UP NEEDED
2305 3437 7640      SZA  CLA         /0=SET UP C8PASS COUNT VALUE
2306                                     /1=C8PASS COUNT VALUE OK
2307 3440 5245      JMP  NOSET       /C8PASS COUNT VALUE ON
2308 3441 1255      TAD   CNTVAL     /GET COUNT VALUE FOR THIS PROG
2309 3442 7040      CMA
2310 3443 3252      DCA  DUCNT       /SET TO NEGATIVE
2311 3444 2254      ISZ  DUCNT       /STORE IN HERE
2312                                     /INDICATE VALUE SET UP
2313 3445 2252      /#2      NOSET, ISZ  DUCNT /COUNT THE NUMBER OF PASSES
2314
2315 3446 5233      /#2      JMP  C8BY1     /EXIT FOR ANOTHER PASS
2316 3447 3254      DCA  DUCNT       /SET TO C8PRNT C8PASS
2317 3450 2235      ISZ  CKCOUT      /BUMP RETURN FOR
2318 3451 5635      JMP  I  CKCOUT   /C8PASS C8TYPE OUT
2319 3452 0000      DOCNT, 0
2320 3453 0000      PASCNT, 0
2321 3454 0000      DOSET, 0
2322 3455 0004      CNTVAL, 4
2323 3456 0000      SELOP2, 0
2324 3457 0412      MESPAS, TEXT  "DJEXCB PASS "
3460 0530
3461 0302
3462 4040
3463 2001
3464 2323
3465 4000

```

2325
2326

```

2327 /*****
2328
2329 /C8CKSW
2330 /ROUTINE THAT WILL CHECK WHERE TO READ THE
2331 /C8 SWITCHES FROM IE, FROM PANEL OR PSEUDO C8SWIT REGISTER
2332 / C8CKSW= JMS XC8SW
2333 /EX  C8CKSW      /READ THE C8SWIT REGISTER
2334 /RETURN WITH THE CONTENTS OF SWITCH REGISTER

```

```

2335
2336 /RETURN TO NEXT LOCATION FOLLOWING CALL WITH THE AC= TO VALUE OF C8SWIT SETTING
2337
2338
2339 3466 0000      XC8SW, 0
2340
2341 3467 7200      /#1      CLA          /CLEAR AC
2342
2343 3470 1764      /#1      TAD  I  SELO1     /GET WD FOR INDICATOR
2344 3471 7710      SPA  CLA         /CHECK IF FRONT PANEL 4000
2345 3472 7614      7614
2346 3473 1765      TAD  I  SAVS1    /DO LAB AND SKIPGET FROM PANEL WITH LAB
2347 3474 5666      JMP  I  XC8SW    /PSEUDO SW
2348                                     /EXIT WITH STATUS BIT IN AC.
2349
2350 /*****
2351
2352 /C8TTYI
2353 /THIS ROUTINE WILL LOOK FOR A INPUT FROM THE CONSOL
2354 / C8TTYI= JMS XC8TTY
2355 /EX.  C8TTYI     /READ CHAR FROM THE CONSOL DEVICE
2356 /RETURN TO CALL PLUS ONE AC CONTAINS THE CHAR
2357 /
2358 /
2359 /
2360 3475 0000      XC8TTY, 0
2361 3476 6031      KSF          /LOOK FOR KEYBOARD FLAG
2362 3477 5276      JMP  .-1
2363 3500 6036      KRB          /GET CHAR
2364 3501 0344      AND   C8177    /MASK FOR 7 BITS
2365 3502 1345      TAD   C8200    /ADD THE EIGHTH BIT
2366 3503 3776      DCA  I  C8CHAR  /STORE IT
2367 3504 1776      TAD  I  C8CHAR
2368 3505 5675      JMP  I  XC8TTY  /EXIT
2369

```

```

2370
2371 /*****
2372
2373 /C8PRNT
2374
2375 /THIS ROUTINE WILL TYPE THE CONTENTS OF THE C8 PRINT BUFFER, THE LOCATION
2376 /OF THE BUFFER WILL BE IN THE ADDR8 FOLLOWING THE CALL, C8 PRINTING OF THE BUFFER
2377 /WILL STOP WHEN A 00 CHAR IS DETECTED, CHARACTERS ARE PACKED 2 PER WORD.
2378
2379 / C8PRNT= JMS XCRPNT
2380
2381
2382 /EX. C8PRNT /C8PRNT THE CONTENTS OF THE FOLLOWING BUFFER
2383 / MESS77 /LOCATION OF C8PRNT BUFFER
2384 /C8PRNT WILL USE THE LOCATION FOLLOWING THE CALL AS THE POINTER FOR THE
2385 /C8PRNT ROUTINE, RETURN TO CALL PLUS TWO WITH AC= 0
2386
2387
2388
2389
2390 3506 0000 XC8PNT, 0
2391 3407 7300 CLA CLL
2392 3510 1706 TAD I XC8PNT /GET C8PRNT BUFFERS STARTING LOCATION
2393 3411 3341 DCA PTSTOR /STORE IN PTSTOR
2394 3412 2306 ISZ XC8PNT /BUMP RETURN
2395 3513 1741 C8DD1, TAD I PTSTOR /GET DATA WORD
2396 3414 0346 AND C87700 /MASK FOR LEFT BYTE
2397 3515 7450 SNA /CHECK IF 00 TERMINATE
2398 3416 5706 JMP I XC8PNT /EXIT
2399 3517 7500 SMA /IS AC MINUS
2400 3420 7020 CHL /MAKE CHAR A 300 AFTER ROTATE
2401 3421 7001 IAC /MAKE CHAR A 200 AFTER ROTATE
2402 3422 7012 RTR
2403 3423 7012 RTR
2404 3424 7012 RTR /PUT CHAR IN BITS 4-11 MAKE IT 8 BIT ASCII
2405 3425 4775 JMS I C8TYPE /C8PRNT IT ON CONSOLE
2406 3426 1741 TAD I PTSTOR /GET DATA WORD
2407 3427 0347 AND C80077 /MASK FOR RIGHT BYTE
2408 3430 7450 SNA /CHECK IF 00 TERMINATOR
2409 3431 5706 JMP I XC8PNT //EXIT
2410 3432 1350 TAD C83740 /ADD FUDGE FACTOR TO DETERMINE IF 200
2411 3433 7500 SMA /OR 300 IS TO BE ADD TO CHAR
2412 3434 1351 TAD C80100 /ADD 100
2413 3435 1352 TAD C8240 /ADD 200
2414 3436 4775 JMS I C8TYPE /C8TYPE ONLY BITS 4-11
2415 3437 2341 ISZ PTSTOR /BUMP POINTER FOR NEXT WORD
2416 3440 5313 JMP C8DD1 /DO AGAIN

2417 3441 0000 PTSTOR, 0 /STOR FOR C8PRNT BUFFER
2418 3442 0000 STOPNT, 0 /0000 C8PRNT 7777=DU NOT C8PRNT
2419
2420 3443 0400 C8400, 400
2421 3444 0177 C8177, 177
2422 3445 0200 C8200, 200
2423 3446 7700 C87700, 7700
2424 3447 0077 C80077, 0077
    
```

```

2425 3450 3740 C83740, 3740
2426 3451 0100 C80100, 100
2427 3452 0240 C8240, 240
2428
2429 3562 *3562
2430 /
2431 3562 4763 C8CHG1, JMS I C8RER1
2432 /
2433 3563 0202 C8RER1, CHANGE
2434 3564 3035 SELD1, SELDPI
2435 3565 0333 SAVS1, SAVSWR
2436 3566 3457 PASHES, HESPAS
2437 3567 3466 C8CKSW, XC8SW
2438 3570 4224 C8CRLF, XC8CRL
2439 3571 3506 C8PRNT, XC8PNT
2440 3572 4201 C8OCT, XC8OCT
2441 3573 3726 C8INGU, XC8ING
2442 3574 4250 C8CKPA, XC8CKP
2443 3575 4306 C8TYPE, XC8TYP
2444 3576 4304 C8CHAR, CHAR
2445 3577 0000 0
2446
2447
2448
2449 3600 PAGE
2450
2451
2452 3600 5346 JMP C8CHG2
2453
2454 /*****
2455
2456 /C8CNTR
2457 /THIS ROUTINE WILL CHECK FOR THE PRESENCE OF CONTROL CHARACTERS
2458 /IT WILL CHECK FOR THE FOLLOWING CHAR C-G-Q-O-L-S
2459 / C8CNTR= JMS XC8CNT
2460
2461 /EX. C8CNTR /CHECK FOR CONTROL CHARACTER
2462 / JMP ANYTHING /LOC FOLLOWING CALL IS FOR CONTINUING THE PROGRAM
2463 / JMP ANYTHING /LOC. IS FOR RETURN IF INMODE SET AND NOT CNTRL CHAR
2464 /
2465
2466 /RETURN IS TO CALL PLUS ONE IFCONTINUE
2467 /RETURN IS TO CALL PLUS TWO IF INMODE SET AND NOT CONTROL CHAR
2468 /
2469 /RETURN IS TO CALL PLUS TWO IF INMODE IS NOT SET AND NO
2470 /CONTROL CHAR ..THIS WILL PRINT THE CHARACTER AND A ?
2471 /CLEAR THE AC AND RETURN CALL+2.
2472
2473 3601 0000 XC8CNT, 0
2474 3602 3765 DCA I C8ACSV /SAVE THE AC
2475 3603 1762 TAD I SELD2C
2476 3604 0266 AND C80400 /CHECK IF ON CONSOLE ACTIVE
2477 3605 7640 SZA CLA
2478 3606 5211 JMP ,+3 /ON ACTIVE CONSOLE
2479 3607 1765 TAD I C8ACSV /GET AC FOR RETURN
    
```



```

2480 3610 5601 JMP I XC8CNT /EXIT NOT ON ACTIVE CONSOLE
2481 3611 6004 GIF
2482 3612 3766 DCA I C8FLSV
2483
2484 3613 7501 MQA
2485
2486 3614 3767 /#6
2487 3615 3255 DCA I C8MQSV /SAVE THE MQ
2488 3616 1350 DCA INDEXA /SET DISPLACEMENT INTO TABLE B
2489 3617 3256 TAD XTABLA /GET ADDR8 OF TABLE A
2490 3620 1656 DCA GETDAT /CONTAINS POINTER TO CONTROL CHAR
2491 3621 7450 REDDA, TAD I GETDAT /GET CONTROL CHAR FROM TABLE
2492 3622 5231 SNA /CHECK FOR A 0 END OF TABLE
2493 3623 1771 JMP DONEA /END OF TABLE NO CONTROL CHAR
2494 3624 7650 TAD I CCHAR8 /COMPARE CHAR TO CONTROL CHAR
2495 3625 5246 SNA CLA /0 IF MATCH
2496 3626 2255 JMP GOITA /MATCH
2497 3627 2256 ISZ INDEXA /NO MATCH NOT END OF TABLE REDD
2498 3630 5220 JMP REDDA /BUMP GETDAT FOR COMPARE OF NEXT CNTRL CHAR.
2499 3631 1772 DONEA, TAD I C8INMD /CHECK IF PROGRAM EXPECTS CHAR
2500 3632 7640 SZA CLA /1=CHAR EXPECTED 0= NO CHAR EXPECTED
2501 3633 5243 JMP EXITA /CHAR EXPECTED
2502 3634 1771 TAD I CCHAR8 /GET CHAR = NOT CONTROL + NOT EXPECTED
2503 3635 4760 JMS I C8TYP /C8PRNT CHAR
2504 3636 1267 TAD C8277 /GET CODE FOR "?"
2505 3637 4760 JMS I C8TYP
2506 3640 4761 JMS I C8CRL
2507
2508 3641 2201 /#1 1SZ XC8CNT /BUMP RETURN
2509
2510 3642 5601 /#2 JMP I XC8CNT /EXIT CALL+2
2511
2512 /#2
2513 3643 2201 EXITA, ISZ XC8CNT /BUMP RETURN FOR MAIN PROGRAM CHECK OF CHAR
2514 3644 1771 TAD I CCHAR8 /PUT CHAR IN AC.
2515 3645 5601 JMP I XC8CNT /EXIT
2516 3646 1351 GOITA, TAD XTABLB /GET START OF TABLE B
2517 3647 1255 TAD INDEXA /GET NOW FAR INTO TABLE
2518 3650 3254 DCA GOTDA /STORE IT
2519 3651 1654 TAD I GOTDA /GET THE ROUTINE STARTING ADDRESS
2520 3652 3254 DCA GOTDA /STORE IT IN HERE
2521 3653 5654 JMP I GOTDA /GOTO CONTROL CHAR ROUTINE
2522 3654 0000 GOTDA, 0000 /ADD OF CNTRL ROUTINE TO EXECUTE
2523 3655 0000 INDEXA, 0000 /DISPLACEMENT INTO CNTRL TABLE
2524 3656 0000 GETDAT, 0000 /LOCATION OF ADDR8 OF CONTROL CHAR.
2525 3657 7575 TABLA, 7575 /CNTRL C BACK TO MONITOR 203
2526 3660 7564 7564 /CNTRL L SWITCH ERROR PRINTING DEVICE 214
2527 3661 7561 7561 /CNTRL O STOP OUTPUTTING DATA 217
2528 3662 7557 7557 /CNTRL Q START DISPLAYING CHAR, AGAIN 221
2529 3663 7555 7555 /CNTRL S STOP SENDING CHAR TO DISPLAY WAIT FOR CNTRL Q 223
2530 3664 7571 7571 /CONTROL G CHANGE SWITCH REGISTER ON FLY
2531 3665 0000 0000
2532
2533 3666 0400 C80400, 400
2534 3667 0277 C8277, 277

```

```

2535 3670 0100 C8100, 100
2536
2537 /
2538 /START SENDING CHAR, TO THE DISPLAY
2539 /THIS WILL RETURN CONTROL TO CALL THAT WAS SET BY
2540 /THE CALL FOR CONTROL S.
2541 /
2542 /#7
2543 /#7
2544 3671 3312 CNTRL0, DCA C8SETS /CLEAR SOFT FLAG FOR CNTRL S
2545
2546 /#7
2547 3672 3772 /#2 DCA I C8INMD /CLEAR THE INMODE FLAG
2548
2549 3673 4775 /#2 JMS I C88GET /GET THE REGISTERS
2550 3674 5713 JMP I C8RETR /GO TO CALL SAVED BY CNTRL S
2551
2552 /
2553 /
2554 /
2555 /
2556 /STOP SENDING CHAR. TO DISPLAY UNTIL A "Q" IS RECEIVED
2557 /
2558 /
2559 /#7
2560 3675 1312 CNTRLS, TAD C8SETS /IF1 DO NOT STORE IN C8RETR
2561 /#7
2562 3676 7640 SZA CLA
2563 3677 5303 JMP C8D07 /DONT SET UP C8RETR
2564
2565 3700 7001 /#5 IAC /MAKE RETURN CALL PLUS 2
2566
2567 3701 1201 TAD XC8CNT /GET RETURN FOR THIS CALL
2568 3702 3313 DCA C8RETR /STORE IT HERE FOR USE BE CNTRL Q
2569
2570 3703 2312 C8D07, ISZ C8SETS /GET FLAG TO SAVE CALL
2571 /#4
2572 3704 4763 /#5 JMS I C8TTYI /LOOK FOR THE INPUT
2573
2574 3705 4775 /#5 JMS I C88GET /GET REGISTERS
2575
2576 3706 4764 /#5 JMS I C8CNTR /CHECK FOR THE CONTROL CHAR
2577 3707 7000 NOP
2578
2579 3710 7200 /#4 CLA
2580
2581 3711 5275 /#7 JMP CNTRLS /IF NOT A CNTRL Q C OR G(FOLLOWED BY LF) RE = ASK
2582 /#7
2583 /#2
2584 3712 0000 C8SETS, 0
2585 3713 0000 C8RETR, 0
2586 /
2587 /SWITCH OUTPUT FROM ONE OUTPUT DEVICE TO ANOTHER - THE TWO OUTPUTS ARE THE
2588 /CONSOLE AND THE PRINTER WITH DEVICE CODE 66.
2589 /

```

```

2590 /
2591 3714 1773 CNTRL, TAD I C8TTYL /GET PRESENT C8SWIT INDICATOR
2592 3715 7040 CMA /COMPLEMENT IT
2593 3716 3773 DCA I C8TTYL /STOR NEW C8SWIT
2594 3717 4774 JMS I C8UPAR /C8PRNT * AND CHAR ON NEW DEVICE
2595 /#3
2596 /#3
2597 /#3
2598 /#3
2599 3720 5601 JMP I XC8CNT /EXIT
2600 /
2601 /STOP C8PRNTING C8ERR MESSAGES - TO CONTINUE C8PRNTING C8TYPE *0
2602 /
2603 /
2604 3721 4774 CNTRLO, JMS I C8UPAR
2605 3722 1776 TAD I C8STOP /GET STOP OR START C8PRNT INDICATOR
2606 3723 7040 CMA
2607 3724 3776 DCA I C8STOP /STORE OPPOSITE STATE
2608 3725 5601 JMP I XC8CNT /EXIT
2609 /
2610 /*****
2611 /
2612 /C8INQU
2613 /C8INQU ROUTINE WILL ASK SWITCH REGISTER QUESTION IF CONSOLE IS ACTIVE,
2614 //
2615 /
2616 C8INQU= JMS XC8INQ
2617 /
2618 /EX C8INQU /C8 WILL ASK SWITCH REG QUESTION
2619 / DO ANYTHING /RETURN IS CALL PLUS ONE AC #0 CONTINUE
2620 /
2621 3726 0000 XC8INQ, 0
2622 3727 7300 CLA CLL
2623 3730 1762 TAD I SELO2C /GET THE WORD
2624 3731 0336 AND AC4008 /CHECK FOR CONSOLE ACTIVE
2625 3732 7650 SNA CLA
2626 3733 5726 JMP I XC8INQ /NOT CONSOLE LEAVE
2627 3734 4770 JMS I XPSW /ASK SWITCH REG QUESTION
2628 3735 5726 JMP I XC8INQ
2629 /
2630 3736 0400 AC4008,400
2631 /
2632 /
2633 3746 *3746
2634 3746 4747 C8CHG2, JMS I C8RER2
2635 /
2636 3747 0202 C8RER2, CHANGE
2637 3750 3657 XTABLA, TABLA
2638 3751 3752 XTABLB, TABLB
2639 3752 4345 TABLC, CNTRLC
2640 3753 3714 CNTRLL
2641 3754 3721 CNTRLO
2642 3755 3671 CNTRLQ
2643 3756 3675 CNTRLS
2644 3757 4001 CNTRLG
    
```

```

2645 3760 4306 C8TYP, XC8TYP
2646 3761 4224 C8CRL, XC8CRL
2647 3762 3456 SELO2C, SELOP2
2648 3763 3475 C8TTYI, XC8TTYI
2649 3764 3601 C8CNT, XC8CNT
2650 3765 4524 C8ACSV, AC8SAVE
2651 3766 4526 C8FLSV, FLSAVE
2652 3767 4525 C8MQSV, MQSAVE
2653 3770 4042 XPSW, XC8PSW
2654 3771 4304 CCHAR8, CHAR
2655 3772 4305 C8INMD, INMODE
2656 3773 4332 C8TTYL, TTYLPT
2657 3774 4016 C8UPAR, UPAROW
2658 3775 3320 C88GET, C8GET
2659 3776 3542 C88TOP, STOPNT
2660 3777 0000 0
2661 /
2662 4000 PAGE
2663 /
2664 /
2665 4000 5354 JMP C8CHG3
2666 /
2667 /CONTROL G
2668 /CHANGE THE SWITCH REGISTER ANYTIME CNTRL G AND RETURN TO
2669 /THE PROGRAM RUNNING.
2670 /
2671 /
2672 4001 4771 CNTRLG, JMS I C8CR /PRINT A CR & LF
2673 4002 1214 TAD C88ETD /CHECK IF THE RETURN ADDR IS SAFE
2674 4003 7640 SZA CLA
2675 4004 5210 JMP C8DO11 /DO NOT CHANGE THE RETURN ADDR
2676 4005 1772 TAD I XC8CN /GET THE RETURN ADDR AND SAVE IT
2677 4006 3215 DCA C8RETD /SAVE THE RETURN HERE
2678 4007 2214 ISZ C88ETD /INDICATE RETURN SAVED DONT DESTROY
2679 4010 4764 C8DO11, JMS I C88WIT /GO CHANGE THE SWITCH REGISTER
2680 4011 3214 DCA C88ETD /CLEAR THE FLAG
2681 /#3
2682 4012 4774 JMS I C8GET8 /RESTORE THE AC HQ LINK ETC
2683 /#3
2684 4013 5615 JMP I C8RETD /RETURN TO THE PROGRAM
2685 /
2686 4014 0000 C88ETD, 0
2687 4015 0000 C8RETD, 0
2688 /
2689 /
2690 /
2691 /
2692 4016 0000 UPAROW, 0 /C8PRNT THE "" AND THE CHAR C8TYPED IN
2693 4017 1225 TAD C8336 /CODE FOR *
2694 4020 4765 JMS I C8TY
2695 4021 1773 TAD I CHAR8 /C8TYPE THE CHAR
2696 4022 1241 TAD C100 /ADD 100 TO FORM GOOD ASCII CHARACTER
2697 4023 4765 JMS I C8TY
2698 4024 5616 JMP I UPAROW /EXIT
2699 /
    
```

2700
 2701 4025 0336 C8336, 336
 2702 4026 0400 C4008, 400
 2703 4027 0040 C840, 40
 2704 4030 7566 C8M212, -212
 2705 4031 7776 D42, -2
 2706 4032 7737 DM41, -41
 2707 4033 0900 SXPSW, 0
 2708 4034 7510 CRN270, -270
 2709 4035 0907 C87, 7
 2710 4036 0277 CRK277, 277
 2711 4037 7775 CRN3, -3
 2712 4040 7777 DM1, -1
 2713 4041 0100 C100, 100
 2714
 2715
 2716
 2717
 2718
 2719
 2720

```

2721 /*****
2722
2723
2724 /C8SWIT
2725 /ROUTINE WILL CHECK IF CONSOL IS ACTIVE IF IT IS ACTIVE DISPLAY
2726 /SM QUESTION, IN NOT ACTIVE IT WILL NOT PRINT THE SM QUESTIONOUT
2727 /RETURN TO CALL PLUS ONE AC=0.
2728 /C8SWIT WILL SET UP THE PSEUDO C8SWIT
2729 /REGISTER WITH THE NEW DATA ENTERED
2730 /
2731 / C8SWIT= JMS XC8PSW
2732 /
2733 /EX. CBDOR, C8SWIT /SET UP PSEUDO C8SWIT REGISTER IF
2734 /ON THE CONSOL PACKAGE, RETURN IS CALL PLUS ONE AC = 0
2735
2736
2737
2738
2739 4042 0000 XC8PSW, 0
2740 /#1
2741 4043 7200 CLA
2742 /#1
2743 4044 1762 TAD I SEL02 /GET THE HARD WARE CONFIG WORD
2744 4045 0226 AND C4008 /MASK FOR CONSOL BIT
2745 4046 7650 SNA CLA /SKIP IF CONSOL PACKAGE IS ACTIVE
2746 4047 5642 JMP I XC8PSW /RETURN WITHOUT ASKING PSEUDO SWITCH
2747 /#3
2748 4050 1346 TAD C8SNST /IS THE SECOND ENTRY FLAG SET?
2749 4051 7640 SZA CIA /SKIP IF FIRST ENTRY
2750 4052 5255 JMP PTSR /SECOND ENTRY WITH OUT A EXIT GO TO SM QUESTION
2751 4053 2346 ISZ C8SNST /FIRSTAR INY SET FLAG
2752 4054 1242 TAD XC8PSW /SAVE THE RETURN ADDRESS
2753 4055 3233 DCA SXPSW
2754 /#3
2755 4056 1357 PTSR, TAD AMES
2756 4057 3261 DCA ,+2
2757 4060 4766 JMS I C8PRN /C8PRNT SR=XXX
2758 4061 4150 MERA
2759 4062 1763 TAD I SAV2 /GET CONTENTS OF SM
2760 4063 4767 JMS I C8OCTA /CONVERT IT TO ASCII
2761 4064 1227 TAD C840 /GET SPACE
2762 4065 4765 JMS I C8TY
2763 4066 2776 ISZ I C8INH /SET FLAG FOR CHAR EXECTED
2764 4067 4770 JMS I C8ECHO /LOOK FOR INPUT
2765 4070 4307 JMS T8TCHA /NOT CONTROL TEST IT IS LEGAL
2766 4071 1773 TAD I CHAR8 /STORE NEW CHNED M SM REG
2767 4072 3763 DCA I SAV2
2768
2769 4073 1237 TAD C8N3 /GET A MINUS 3
2770 4074 3347 DCA T8PCNT /STORE IN TEMP COUNT
2771 4075 4770 GETCH1, JMS I C8ECHO /GET NEXT CHAR
2772 4076 4307 JMS T8TCHA /CHECK IF CR + GOOD CHAR
2773 4077 1763 TAD I SAV2 /GET C8SWIT REGISTER
2774 4100 7106 RTL CLL /ROTATE IT LEFT 3 PLACES
2775 4101 7004 RAL
    
```

```

2776 4102 1773 TAD I CHAR8 /GET CHAR + ADD IT TO PREVIOUS CONTENTS
2777 4103 3763 DCA I SAV2 /SAVE NEW CONTENTS
2778 4104 2347 ISZ TPCNT /BUMP COUNT
2779 4105 5275 JMP GLICM1 /JMP BACK + GET NEXT CHAR
2780 4106 5343 JMP ENDIT /END 4 CHAR CBTYPED IN
2781 4107 0000 TSTCHA, 0
2782 4110 1230 TAD CBN212 /IS IT A LF (212) ?
2783 4111 7450 SNA /SKIP IF NOT
2784 4112 5756 JMP I CBLF /YES
2785 4113 1231 TAD DM2 /IS IT A CNTR L (214) ?
2786 4114 7450 SNA
2787 4115 5341 JMP LERR1 /YES
2788 4116 1240 TAD DM1 /IS IT A CR (215) ?
2789 4117 7450 SNA
2790 4120 5343 JMP ENDIT /YES
2791 4121 1231 TAD DM2 /IS IT A CNTR O (217) ?
2792 4122 7450 SNA
2793 4123 5341 JMP LERR1 /YES
2794 4124 1232 TAD DM41 /CHECK IF IT IS IN THE OCTAL 4 RANGE (260-267)
2795 4125 7710 SPA CLA /IF NOT POSITIVE CBERR CHAR SMALLER THEN 260
2796 4126 5337 JMP ERR1 /CBERR = CHAR TOO SMALL
2797 4127 1773 TAD I CHAR8 /GET CHAR
2798 4130 1234 TAD CBN270 /GET A =270 + CHECK IF IT IS LARGER THEN 7
2799 4131 7700 SNA CLA /SKIP IF LESS THEN 7
2800 4132 5337 JMP ERR1 /CBERR ON CHAR NOT IN RANGE
2801 4133 1773 TAD I CHAR8 /GET CHAR
2802 4134 0235 AND C87 /MASK FOR RIGHT BYTE
2803 4135 3773 DCA I CHAR8 /STORE IN CHAR
2804 JMP I TSTCHA /GET CHAR IN AC
2805 4136 5707 /EXIT
2806 4137 1236 ERR1, TAD C8K277 /CBPRNT
2807 4140 4765 JMS I C8TY /
2808 4141 4771 LERR1, JMS I C8CR /
2809 4142 5243 JMP XC8PSW+1 /EXIT + ASK AGAIN
2810 4143 4771 ENDIT, JMS I C8CR /DO A CR LF
2811 /#3
2812 4144 3346 DCA C8SWST /CLEAR THE PSW ENTRY FLAG
2813 4145 5633 JMP I SXP8W /EXIT ROUTINE
2814 4146 0000 C8SWST, 0
2815 /#3
2816
2817 4147 0000 TPCNT, 0
2818 4150 2322 MESA, TEXT "SR= "
4151 7540
4152 0000
2819
2820
2821 /
2822 4154 *4154
2823 /
2824 4154 4755 C8CHG3, JMS I C8RER3
2825 /
2826 4155 0202 C8RER3, CHANGE
2827 4156 4531 CBLF, XCBLF
2828 4157 4150 AMES, MESA
    
```

```

2829 4160 4345 XCTLC, CNTRLC
2830 4161 4001 XCTLG, CNTRLG
2831 4162 3456 SEL02, SELOP2
2832 4163 0333 SAV2, SAVSWR
2833 4164 4042 C8SWIT, XC8PSW
2834 4165 4306 C8TY, XC8TYP
2835 4166 3506 C8PRN, XC8PNT
2836 4167 4201 C8OCTA, XC8OCT
2837 4170 4273 C8ECHO, XC8ECH
2838 4171 4224 C8CR, XC8CRL
2839 4172 3601 XC8CN, XC8CNT
2840 4173 4304 CHAR8, CHAR
2841 4174 3320 C8GETA, C8GET
2842 4175 4536 C8BY4, C8BY4
2843 4176 4305 C8INM, INMODE
2844 4177 0000 0
2845
2846 4200 PAGE
2847
2848
2849
2850 4200 5361 JMP C8CHG4
2851 /*****
2852 /C8OCTA
2853 /OCTAL TO ASCII CONVERSION
2854 /THIS ROUTINE WILL TAKE THE OCTAL NUMBER IN THE AC AND CONVERT IT TO ASCII
2855 /THE RESULT WILL BE PRINTED ON THE CONSOL DISPLAY
2856 / C8OCTA= JMS XC8OCT
2857 /
2858 /EX. C8OCTA /AC CONTAINS NUMBER TO BE CHANGE
2859 /
2860
2861 4201 0000 XC8OCT, 0
2862 4202 7106 CLL RTL
2863 4203 7006 RTL
2864 4204 3222 DCA C8TMP1 /POSITION THE FIRST CHAR FOR PRINTING
2865 4205 1242 TAD C8N4 /SAVE CORRECT POSITIONED WORD HERE
2866 4206 3223 DCA C8CKP /STORE COUNTER IN HERE
2867 4207 1222 TAD C8TMP1 /GET FIRST NUMBER
2868 4210 0243 AND C8P7 /MASK
2869 4211 1244 TAD C8P260 /ADD THE PRINT CONSTANT
2870 4212 4767 JMS I C8TPE /TYPE THE NUMBER
2871 4213 1222 TAD C8TMP1 /
2872 4214 7006 RTL
2873 4215 7004 RAL /PUT NEXT NUMBER IN POSITION
2874 4216 3222 DCA C8TMP1 /STORE IT
2875 4217 2223 ISZ C8CKP /DONE YET WITH FOUR NUMBERS
2876 4220 5207 JMP C8DD4 /NOT YET DO MORE
2877 4221 5601 JMP I XC8OCT /DONE WITH FOUR
2878 4222 0000 C8TMP1, 0
2879 4223 0000 C8CKP, 0
    
```

```

2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891 4224 0000  XC8CRL, 0
2892 4225 7300      CLA CLL
2893 4226 1245      TAD   C8P215      /GET CODE FOR CR
2894 4227 4767      JMS   I  C8TPE
2895 4230 1240      TAD   FILLER
2896 4231 7040      CWA
2897 4232 3241      DCA   FILCNT      /STORE FILLER IN HERE
2898 4233 1246      TAD   C8P212      /GET CODE FOR LF
2899 4234 4767      C8D02, JMS   I  C8TPE
2900 4235 2241      ISZ   FILCNT      /CHECK ON FILLER CHAR
2901 4236 5234      JMP   C8D02      /TYPE A NON PRINTING CHAR
2902 4237 5624      JMP   I  XC8CRL      /EXIT
2903
2904 4240 0004      /#1/#2      FILLER, 0004      /FILLER SET FOR 4 CHAR
2905
2906
2907 4242 7774      /#1/#2      FILCNT, 0      /COUNTER FOR FILL
2908 4243 0007      C8N4,   =4
2909 4244 0260      C8P7,   7
2910 4245 0215      C8P260, 260
2911 4246 0212      C8P215, 215
2912 4247 0400      C8P212, 212
2913
2914
2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938 4250 0000      XC8CKP, 0
2939 4251 3773      DCA   I  AC8C8      /SAVE THE AC
2940 4252 6604      GTF
2941 4253 3774      DCA   I  FL8C8      /SAVE THE FLAGS
2942
2943
2944
2945 4254 7501      /#2      #6      MOA      /PUT MQ IN AC
2946
2947 4255 3775      /#6      DCA   I  MQ8C8      /SAVE THE MQ
2948
2949 4256 6031      /#2      KSF      /CHECK THE KEYBOARD FLAG
2950 4257 5271      JMP   C8BY3      /EXIT TO CALL PLUS 1
2951 4260 1766      TAD   I  SELO2A      /IS CONSOLE ACTIVE
2952 4261 0247      AND   CP400
2953 4262 7650      SNA   CLA
2954 4263 5271      JMP   C8BY3      /EXIT TO CALL PLUS 1
2955 4264 4770      JMS   I  C8TYI      /GET THE CHAR
2956
2957 4265 4776      /#2      JMS   I  GETC8      /GET THE FLAGS
2958 4266 4771      JMS   I  C8NTR      /CHECK IF CONTROL CHAR,
2959
2960 4267 7000      /#2      NOP      /RETURN IF A CONTINUE CHAR,
2961 4270 2250      ISZ   XC8CKP      /BUMP RETURN FOR CALL PLUS 2
2962 4271 4776      C8BY3, JMS   I  GETC8      /GET REGISTERS
2963 4272 5650      JMP   I  XC8CKP      /SAY GOOD BY
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975 4273 0000      /#1      XC8ECH, 0
2976 4274 4770      JMS   I  C8TYI      /WAIT FOR CHAR FROM KEYBOARD
2977
2978
2979 4275 2305      /#1      ISZ   INMODE      /SET INMODE IDENTIFYING THIS AS A EXPECTED CHAR
2980 4276 4771      JMS   I  C8NTR      /GO CHECK IF IT IS A CONTROL CHAR
2981 4277 5302      JMP   EXECH      /WAS A CONTROL CHAR - CONTINUE RUNNING
2982 4280 4767      JMS   I  C8TPE      /NOT A CONTROL CHAR C8PRNT IT
2983 4281 3305      DCA   INMODE      /CLEAR FLAG THAT CHAR EXPECTED
2984 4282 1304      EXECH, TAD   CHAR      /GET CHAR IN AC
2985 4283 5673      JMP   I  XC8ECH      /EXIT
2986 4284 0000      CHAR, 0
2987 4285 0000      INMODE, 0
2988
2989

```

```

2935
2936
2937
2938 4250 0000      XC8CKP, 0
2939 4251 3773      DCA   I  AC8C8      /SAVE THE AC
2940 4252 6604      GTF
2941 4253 3774      DCA   I  FL8C8      /SAVE THE FLAGS
2942
2943
2944
2945 4254 7501      /#2      #6      MOA      /PUT MQ IN AC
2946
2947 4255 3775      /#6      DCA   I  MQ8C8      /SAVE THE MQ
2948
2949 4256 6031      /#2      KSF      /CHECK THE KEYBOARD FLAG
2950 4257 5271      JMP   C8BY3      /EXIT TO CALL PLUS 1
2951 4260 1766      TAD   I  SELO2A      /IS CONSOLE ACTIVE
2952 4261 0247      AND   CP400
2953 4262 7650      SNA   CLA
2954 4263 5271      JMP   C8BY3      /EXIT TO CALL PLUS 1
2955 4264 4770      JMS   I  C8TYI      /GET THE CHAR
2956
2957 4265 4776      /#2      JMS   I  GETC8      /GET THE FLAGS
2958 4266 4771      JMS   I  C8NTR      /CHECK IF CONTROL CHAR,
2959
2960 4267 7000      /#2      NOP      /RETURN IF A CONTINUE CHAR,
2961 4270 2250      ISZ   XC8CKP      /BUMP RETURN FOR CALL PLUS 2
2962 4271 4776      C8BY3, JMS   I  GETC8      /GET REGISTERS
2963 4272 5650      JMP   I  XC8CKP      /SAY GOOD BY
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975 4273 0000      /#1      XC8ECH, 0
2976 4274 4770      JMS   I  C8TYI      /WAIT FOR CHAR FROM KEYBOARD
2977
2978
2979 4275 2305      /#1      ISZ   INMODE      /SET INMODE IDENTIFYING THIS AS A EXPECTED CHAR
2980 4276 4771      JMS   I  C8NTR      /GO CHECK IF IT IS A CONTROL CHAR
2981 4277 5302      JMP   EXECH      /WAS A CONTROL CHAR - CONTINUE RUNNING
2982 4280 4767      JMS   I  C8TPE      /NOT A CONTROL CHAR C8PRNT IT
2983 4281 3305      DCA   INMODE      /CLEAR FLAG THAT CHAR EXPECTED
2984 4282 1304      EXECH, TAD   CHAR      /GET CHAR IN AC
2985 4283 5673      JMP   I  XC8ECH      /EXIT
2986 4284 0000      CHAR, 0
2987 4285 0000      INMODE, 0
2988
2989

```

```

2990 /CBTYPE
2991 /THIS ROUTINE WILL C&PRNT ON THE CONSOLE OR THE LPT WITH DEVICE CODE 66.
2992 /
2993 /
2994 /      CRTYPE= JMS XC&TYP
2995
2996 /EX.  CBTYPE      /C&PRNT THE CHAR IN THE AC.
2997 /      /RETURN CALL PLUS ONE AC #0000
2998 /      /DO NOT CLEAR THE LINK IN THIS ROUTINE NEEDED BYC&OCT
2999
3000
3001 4906 0000 XC&TYP, 0
3002 4907 3331 DCA  PNT&UF      /STORE CHAR
3003 4910 1332 TAD  TTYLPT      /CHECK 0=TTY 7777=LPT
3004 4911 7640 SZA  CLA
3005 4912 5321 JMP  XD&LPT      /DO OUT PUT ON LPT
3006 4913 1331 TAD  PNT&UF
3007 4914 6046     ILS
3008 4915 6041     TSF
3009 4916 5315 JMP  ,=-1
3010 4917 6042     TCF
3011 4920 5327 JMP  C&BY5
3012 4921 1331 XD&LPT, TAD  PNT&UF      /GET CHAR
3013 4922 6666     PST&  PCLF      /C&PRNT IT
3014
3015 4923 4333 /#6      JMS  C&HANG      /CHECK KEYBOARD IF HUNG
3016
3017 4924 6661 /#6
3018 4925 5323     PSKF
3019 4926 6662     JMP  ,=-2      /WAIT UNTIL DONE
3020 4927 7200 C&BY5, CLA
3021 4930 5706     JMP  I  XC&TYP      /CLEAR THE AC
3022 4931 0900 PNT&UF, 0
3023 4932 0000 TTYLPT, 0
3024
3025
3026 4933 0000 /#6      C&HANG, 0      /WILL CHECK KEYBOARD FOR CNTRL CHAR
3027 /WILL NEED IF LPT HANGS TO GET OUT
3028
3029 4934 7300     CLA  CLL
3030 4935 1250     TAD  XC&CKP      /SAVE C&CKPA RETURN LINKAGE
3031 4936 3344     DCA  LC&KPA
3032 4937 4772     JMS  I  C&KPA      /SEE IF KEYBOARD INPUT
3033 4940 7000     NOP
3034 4941 1344     TAD  LC&KPA      /RESTORE C&CKPA RETURN LINKAGE
3035 4942 3250     DCA  XC&CKP
3036 4943 5733     JMP  I  C&HANG      /IF HUNG IN LPT SKIP FLAG NOT SET
3037 4944 0000 LC&KPA, 0
3038
3039
3040
3041 /RETURN TO MONITOR
3042 /#7
3043 4945 7200 CNTRL, CLA
3044 4946 3764     DCA  I  TTYLC&      /CLEAR SOFTWARE FLAG FOR TERMINAL PRINTER
    
```

```

3045 4947 4765     JMS  I  UPARC&      /C&PRNT A~ AND A CHAR
3046 4950 5763     JMP  I  BOTRES      /GO RESTORE MONITOR AND GO TO IT
3047
3048
3049
3050 4361 /+4361
3051 /
3052 4961 4762 C&CHG4, JMS  I  C&RER4
3053 /
3054 4962 0202 C&RER4, CHANGE
3055 4963 3334 BOTRES, RE&BOT
3056 4964 4332 TTYLC&, TTYLPT
3057 4965 4016 UPARC&, UPAR&W
3058 4966 3456 SEL&2A, SEL&2P
3059 4967 4306 C&TPE, XC&TYP
3060 4970 3475 C&TYI, XC&TYI
3061 4971 3601 C&NTR, XC&CNT
3062 4972 4250 C&KPA, XC&CKP
3063 4973 4524 AC&SC&, AC&SAVE
3064 4974 4526 FL&SC&, FL&SAVE
3065 4975 4525 MQ&SC&, MQ&SAVE
3066 4976 3320 GETC&, C&GET
3067 4977 0000 0
3068
3069 4400 PAGE
3070
3071
3072 4100 5344     JMP  C&CHG5
3073
3074 /*****
3075
3076 /C&ERR
3077 /THIS ROUTINE WILL DETERMINE WHAT TO DO WHEN A C&ERR IS ENCOUNTERED
3078 /WILL CHECK IF CLASSIC SYSTEM, WILL CHECK C&SWIT REGISTERS.
3079 /
3080 /C&ERR= JMS  XC&ERR
3081 /EX.  C&ERR      /GO TO C&ERR CALL IF NOT CONS&L /0/
3082 /      /RETURN IS CALL PLUS ONE AC #0000
3083 /
3084
3085 4401 0000 XC&ERR, 0
3086 4402 6002 IOF
3087 4403 3324 DCA  ACS&VE      /SAVE AC
3088 4404 6004 GTF
3089 4405 3326 DCA  FL&SAVE      /SAVE THE FLAGS
3090 4406 7501 MQA
3091 4407 3325 DCA  MQ&SAVE      /SAVE THE MQ
3092 4410 7340 CLA  CLL  CMA      /SUBTRACT A 1 FOR TRUE LOCATION
3093 4411 1201 TAD  XC&ERR      /GET RETURN LOCATION
3094 4412 3323 DCA  PC&AV      /SAVE ADD OF C&ERR CALL
3095 4413 1764 TAD  I  SEL&2B      /GET LOCATION 22
3096 4414 0330 AND  C&8400      /MASK FOR CLASSIC SYSTEM
3097 4415 7650 SNA  CLA      /SKIP IF CLASSIC BIT IN LOC 22 SET
3098 4416 5265 JMP
3099 4417 1773 TAD  I  NT&STOP      /NO ERROR PRINTING
    
```

```

3100 4420 7640 SZA CLA
3101 4421 5260 JMP C8DU10 /DO NOT PRINT
3102 /#?
3103 4422 4756 JMS I CNLFPC8
3104 /#2
3105 4423 1350 TAD MESERR
3106 4424 3226 DCA ,+2
3107 4425 4757 JMS I PRNTC8
3108 4426 4477 ERRMES /PRINT THE ERROR MESSAGE
3109 4427 1351 TAD PCMES
3110 4430 3232 DCA ,+2
3111 4431 4757 JMS I PRNTC8
3112 4432 4507 MESPC /PRINT THE PC STATEMENT
3113 4433 1323 TAD PCSAV
3114 4434 4760 JMS I OCTAC8 /CONVERT 4 DIGIT PC TO ASCII
3115 4435 1352 TAD ACMES
3116 4436 3240 DCA ,+2
3117 4437 4757 JMS I PRNTC8
3118 4440 4512 MESAC /PRINT THE AC MESS
3119 4441 1324 TAD ACSAVE
3120 4442 4760 JMS I OCTAC8
3121 4443 1353 TAD MQMES
3122 4444 3246 DCA ,+2
3123 4445 4757 JMS I PRNTC8
3124 4446 4515 MESMQ /PRINT MQ
3125 4447 1325 TAD MQSAVE
3126 4450 4760 JMS I OCTAC8
3127 4451 1354 TAD FLMES
3128 4452 3254 DCA ,+2
3129 4453 4757 JMS I PRNTC8
3130 4454 4520 MESFL /PRINT FL
3131 4455 1326 TAD FLSAVE
3132 4456 4760 JMS I OCTAC8
3133 4457 4756 JMS I CNLFPC8
3134 4460 4761 C8DU10, JMS I CKSWC8 /CHECK SWITCH REGISTER
3135 /#1
3136 4461 7710 SPA CLA /SKIP IF BIT 0 NOT SET
3137 /#1
3138 4462 5274 JMP C8BY2 /LEAVE
3139 4463 4762 JMS I INQUC8 /GO TO THE INQUIRE ROUTINE
3140 4464 5274 JMP C8BY2 /LEAVE
3141 4465 4761 NTCLAS, JMS I CKSWC8 /CHECK PSEUDO SWITCH REGISTER
3142 /CHECK THE CSWIT REGISTER
3143 /#1
3144 4466 7710 SPA CLA /SKIP IF HALT
3145 /#1
3146 4467 5601 JMP I XC8ERR /NO HALT CONTINUE
3147 4470 1327 TAD C8HLT
3148 4471 3723 DCA I PCSAV /CODE FOR HLT
3149 /#5
3150 4472 4763 JMS I C8GETS /PUT IT IN CALL LOC.
3151 /#5
3152 4473 5723 JMP I PCSAV /EXIT TO CALL AND HALT
3153 4474 7000 C8BY2, NOP
3154 4475 4763 JMS I C8GETS /GET THE REGISTERS
    
```

```

3155 /#3
3156 /#3
3157 4476 5601 JMP I XC8ERR
3158
3159 4477 0412 ERRMES, TEXT "DJEXCB FAILED "
4400 0530
4401 0302
4402 4040
4403 0601
4404 1114
4405 0504
4406 4000
3160 4407 4040 MESPC, TEXT " PC:"
4410 2003
4411 7200
3161 4412 4040 MESAC, TEXT " AC:"
4413 0103
4414 7200
3162 4415 4040 MESMQ, TEXT " MQ:"
4416 1521
4417 7200
3163 4420 4040 MESFL, TEXT " FL:"
4421 0614
4422 7200
3164 4423 7777 PCSAV, 7777
3165 4424 7777 ACSAVE, 7777
3166 4425 7777 MQSAVE, 7777
3167 4426 7777 FLSAVE, 7777
3168 4427 7402 C8HLT, 7402
3169 4430 0400 C88400, 400
3170
3171 /
3172 /GO TO THE QUESTION C8SWIT
3173 /
3174 4431 7200 XC8LF, CLA
3175 4432 3755 DCA I XPASSCT /ZERO PASS COUNTER
3176 4433 3766 DCA I C8STS
3177 4434 3767 DCA I C8IND
3178 4435 4770 JMS I C8LPCR
3179 4436 3771 C8BY4, DCA I C888SW /CLEAR FLAG FOR CONTROL G
3180 4437 5772 JMP I X088W /GO TO ADDRESS FOR C8SWIT
3181
3182 /
3183 4544 *4544
3184 /
3185 4544 1374 C8CHGS, TAD PAGJMP
3186 4445 3377 DCA C8QSPG
3187 4446 4747 JMS I C8RERS
3188 /
3189 4447 0202 C8RERS, CHANGE
3190 4450 4477 MESERR, ERRMES
3191 4451 4507 PCMES, MESPC
3192 4452 4512 ACMES, MESAC
3193 4453 4515 MQMES, MESMQ
3194 4454 4520 FLMES, MESFL
    
```

```

3195 4555 3453 XPASSCT,PASCNT
3196 4556 4224 CRLFCB, XCBCLL
3197 4557 3506 PRNTCB, XCBPNT
3198 4560 4201 OCTACB, XCMUCT
3199 4561 3465 CKSWCB, XCBSSW
3200 4562 3726 INQUCB, XCHINQ
3201 4563 3320 CBGETS, CBGET
3202 4564 3456 SELU2B, SELOP2
3203 4565 4016 CBUPA, UPAROW
3204 4566 3712 CBSTS, CBSETS
3205 4567 4305 CBIND, INMODE
3206 4570 4224 CBLFCR, XCBCLL
3207 4571 4146 CBSSSW, CBSSWT
3208 4572 3032 XDOSW, CL88GN
3209 4573 3542 NISTOP, STUPNT
3210 4574 4765 PAGJMP, ZCHANGE-1 /VT78/
3211 4575 0000 0
3212 4576 5777 JMP I CROSPG
3213 4577 4765 CROSPG, ZCHANGE-1 /VT78/
3214
3215 4600 *4600
3216
3217 4600 0000 TYPEIT, 0
3218 4601 1213 TAD CP215
3219 4602 4217 JMS TYPE
3220 4603 1214 TAD CP212
3221 4604 4217 JMS TYPE
3222 4605 1215 TAD M4
3223 4606 3216 DCA TYPECT
3224 4607 4217 JMS TYPE
3225 4610 2216 ISZ TYPECT
3226 4611 5207 JMP ,=2
3227 4612 5600 JMP I TYPEIT
3228
3229 4613 0215 CP215, 215
3230 4614 0212 CP212, 212
3231 4615 7774 M4, =4
3232 4616 0000 TYPECT, 0
3233
3234 4617 0000 TYPE, 0
3235 4620 6046 TLS
3236 4621 6041 TSF
3237 4622 5221 JMP ,=1
3238 4623 6042 TCF
3239 4624 7200 CLA
3240 4625 5617 JMP I TYPE
3241
3242 4626 7126 BOOTS, 7126
3243 4627 1060 1060
3244 4630 6751 6751
3245 4631 7201 7201
3246 4632 4053 4053
3247 4633 4053 4053
3248 4634 7104 7104
3249 4635 6755 6755
    
```

```

3250 4636 5054 5054
3251 4637 6754 6754
3252 4640 7450 7450
3253 4641 7610 7610
3254 4642 5046 5046
3255 4643 1060 1060
3256 4644 7041 7041
3257 4645 1061 1061
3258 4646 3061 3060
3259 4647 5024 5024
3260 4650 6751 6751
3261 4651 4053 4053
3262 4652 3002 3002
3263 4653 2050 2050
3264 4654 5047 5047
3265 4655 0000 0000
3266 4656 6753 6753
3267 4657 5033 5033
3268 4660 6752 6752
3269 4661 5453 5453
3270 4662 7024 7024
3271 4663 6030 6030
3272 4664 0000 0000
3273
3274
3275 /ROUTINE TO HANDLE HACK IN VT78 CPU--PAGE BIT MUST BE 0 FOR AUTO INDEXING
3276 / TO WORK--EVEN IF THE INSTRUCTION IS ON PAGE 0.
3277 /ROUTINE USED ONLY WHEN TESTING ON VT78 PROCESSOR.
3278
3279 4700 *4700
3280 4700 1771 VT78GL, TAD I ZADDRS /VT78/CHECK ADDRESS TO SEE IF IT IS IN THE
3281 /VT78/ VULNERABLE RANGE (0000-0177).
3282 4701 0316 AND K7600 /VT78/
3283 4702 7640 SZ A CLA /VT78/
3284 4703 5314 JMP RETN /VT78/ OUTSIDE VULNERABLE RANGE
3285 4704 1770 TAD I VINSTR /VT78/CHECK INST, TO SEE IF IT IS THE VULNERABLE
3286 /VT78/ TYPE--FORM X6X0
3287 4705 0317 AND K0700 /VT78/
3288 4706 1320 TAD M0600 /VT78/
3289 4707 7640 SZ A CLA /VT78/
3290 4710 5314 JMP RETN /VT78/ INST NOT VULNERABLE TYPE
3291 4711 1770 TAD I VINSTR /VT78/ZERO PAGE BIT SO THAT INST WILL WORK PROPERLY
3292 4712 0321 AND K7577 /VT78/
3293 4713 3770 DCA I VINSTR /VT78/
3294 4714 7340 RETN, CLA CLL CMA /VT78/
3295 4715 5767 JMP I BACK /VT78/
3296
3297 4716 7600 K7600, 7600 /VT78/
3298 4717 0700 K0700, 0700 /VT78/
3299 4720 7200 M0600, =600 /VT78/
3300 4721 7577 K7577, 7577 /VT78/
3301
3302 4722 0000 XSTMS, 0
3303 4723 7200 CLA
3304 4724 1772 TAD I SEL02X /GET HARDWARE CONFIG WORD
    
```



```

3305 4725 0337 AND C400X /MASK FOR CONSOL BIT
3306 4726 7650 SNA CLA /SKIP IF CONSOL PACKAGE IS ACTIVE
3307 4727 5727 JMP I XSTMS /RETURN WITHOUT PRINTING HEADING & ASKING PSEUDO SWITCH
3308 4730 1373 TAD MESHD
3309 4731 3333 DCA +2
3310 4732 4774 JMS I CBPX /PRINT PROGRAM HEADER MESSAGE
3311 4733 4740 HDMES
3312 4734 4775 JMS I XLF /PRINT A CR & LF
3313 4735 4776 JMS I XPBR /ASK SWITCH REG QUESTION
3314 4736 5722 JMP I XSTMS
3315 4737 0400 C400X, 400
3316 4740 1501 HDMES, TEXT *MAINDEC-08-DJEXC-B*
4741 1110
4742 0405
4743 0355
4744 6070
4745 5504
4746 1205
4747 3003
4750 5502
4751 0000

3317
3318
3319 4765 *4765
3320 4765 4766 JMS I ZCHANGE /VT78/
3321 4766 0202 ZCHANGE,CHANGE /VT78/
3322 4767 0613 BACK, RFVT /VT78/
3323 4770 0746 VINSTR, INSTR /VT78/
3324 4771 0745 ZADDHS, ADDRS /VT78/
3325 4772 3456 SELD2X, SELOP2
3326 4773 4740 MESHD, HDMES
3327 4774 3506 CBPX, XCSPNT
3328 4775 4224 XLF, XCSPRL
3329 4776 4042 XPBR, XCSPSW
3330 4777 0000 0 /VT78/
3331
3332
3333 5000 *5000
3334 /
3335 5000 5274 JMP DCHNG
3336 /
3337 5001 1271 SETCUN, TAD M7 /ROUTINE TO SET UP CONSTRAINT WORDS
3338 5002 3347 DCA UPDWN
3339 5003 1325 TAD TABLE
3340 5004 3373 DCA MOVWDX
3341 5005 3704 DCA I XFIRST
3342 5006 1773 CONRAN, TAD I MOVWDX

3343 5007 3374 DCA MVWDPG
3344 5010 4710 JMS I XRANCN
3345 5011 7040 CMA
3346 5012 3774 DCA I MVWDPG
3347 5013 2373 ISZ MOVWDX
3348 5014 1773 TAD I MOVWDX
3349 5015 3374 DCA MVWDPG
3350 5016 4710 JMS I XRANCN
    
```

```

3351 5017 3774 DCA I MVWDPG
3352 5020 2373 ISZ MOVWDX
3353 5021 2347 ISZ UPDWN
3354 5022 5206 JMP CONRAN
3355 5023 3720 DCA I XDRFLG
3356 5024 3721 DCA I XFLOFG
3357 5025 4705 STARTP, JMS I OP1SET
3358 5026 7344 CLA CLL CMA RAL
3359 5027 3706 DCA I ENDCNT
3360 5030 4707 JMS I INITFO
3361 5031 6001 ION /TURN THE INTERRUPT ON
3362 5032 5240 JMP FILL
3363 5033 7300 CLA CLL
3364 5034 3722 DCA I XCNT3
3365 5035 5724 RESCNT, JMP I XGNFLD /GO, BABY GO !!!
3366 /
3367 5036 1273 ADDONE, TAD C10
3368 5037 1722 TAD I XCNT3
3369 5040 3722 FILL, DCA I XCNT3 /START WITH FIELD 0
3370 5041 1722 TAD I XCNT3 /IS THIS FIELD =TO LAST FIELD OR OVER
3371 5042 7041 CIA
3372 5043 1711 TAD I ZFLDLH
3373 5044 7510 SPA
3374 5045 5233 JMP RESCNT-2
3375 5046 7650 SNA CLA
3376 5047 1712 TAD I ZUPLIM /YES, SET UPPER LIMITS(-1777- -7777)
3377 5050 7041 CIA
3378 5051 3713 DCA I ZHIGH /IF NOT LAST FIELD UPPER LIMITS=0
3379 5052 6224 RIF /READ THE INSTRUCTION FIELD
3380 5053 7041 CIA /NEGATE THE FIELD
3381 5054 1722 TAD I XCNT3 /GET THE FIELD TO FILL
3382 5055 7650 SNA CLA /IS IT EQUAL TO THE PROGRAM FIELD
3383 5056 5715 JMP I XFLRND /YES FILL AROUND THE PROGRAM
3384 5057 1722 TAD I XCNT3 /GET THE FIELD TO FILL
3385 5060 7650 SNA CLA /IS IT EQUAL TO FIELD ZERO
3386 5061 7307 CLA CLL IAC RTL /YES ADD FOUR TO FILL COUNTER
3387 5062 1713 TAD I ZHIGH /GET COUNTER
3388 5063 3713 DCA I ZHIGH /RESTORE IT
3389 5064 1722 TAD I XCNT3 /GET THE FIELD TO FILL
3390 5065 7650 SNA CLA
3391 5066 7307 CLA CLL IAC RTL /START FILLING AT ADDRESS 4 FOR FIELD 0
3392 5067 4714 JMS I XFILL /FILL THE WHOLE FIELD
3393 5070 5236 JMP ADDONE
3394 /
3395
3396 5071 7771 M7, -7
3397 5072 0200 K200, 200
3398 5073 0010 C10, 10
3399 /
3400 5074 *5074
3401 /
3402
3403 5074 1720 DCHNG, TAD I XDRFLG
3404 5075 7640 SZA CLA
3405 5076 1317 TAD XSWAP2
    
```

```

3406 5077 7450          SNA
3407 5100 1316          TAD XSWAP1
3408 5101 3347          DCA UPDWN
3409 5102 4703          JMS I DRERNG
3410 5103 0202          DPERNG, CHANGE
3411 5104 3107          XFIRST, FIRST
3412 5105 3036          OPISET, SETOPI
3413 5106 3245          EMDCNT, CNTEND
3414 5107 0334          INITFO, FOINIT
3415 5110 2127          XRAMCN, RANCUN
3416 5111 1144          ZFLDLM, FLDLIM
3417 5112 1550          ZUPLIM, UPRLIM
3418 5113 1145          ZHIGH, HGHLIM
3419 5114 0527          XFILL, FILALL
3420 5115 3001          XFLRND, FLRND
3421 5116 0216          XSWAP1, SWAP1
3422 5117 5150          XSWAP2, SWAP2
3423 5120 0213          XDRFLG, DIRFLG
3424 5121 0500          XFLOFG, FLDFLG
3425 5122 0247          XCNTF3, CNTF3
3426 5123 0200          PRGBGN, BGN
3427 5124 1001          XGNFLD, GENFLO
3428 5125 5126          TABLE, XFDMSK
3429 5126 1127          XFDMSK, FLDMSK
3430 5127 1130          XCNFLD, CONFLD
3431 5130 1131          XADRS, ADMRSK
3432 5131 1132          XCNADR, CONADR
3433 5132 1642          XINSMS, INSMK
3434 5133 1643          XCNINS, CONINS
3435 5134 1133          XINDMS, INDMK
3436 5135 1134          XCNIND, CONIND
3437 5136 1135          XMDTMS, MDTMSK
3438 5137 1136          XCNMDT, CONMDT
3439 5140 1137          XACDMS, ACDMSK
3440 5141 1140          XCNACD, CONACO
3441 5142 1141          XMQDMS, MQDMSK
3442 5143 1142          XCNMQD, CONMQD
3443 5144 5025          XRSCNT, STARTP
3444 5145 0000          0
3445 5146 5747          JMP I ,+1
3446 5147 7402          UPDWN, HLT                /ADDRESS OF SWAP ROUTINE
3447 5150 1375          SWAP2, TAD SIZPRG
3448 5151 3347          DCA UPDWN
3449 5152 1323          TAD PRGBGN
3450 5153 3373          DCA MOVWDX
3451 5154 1323          TAD PRGBGN
3452 5155 1272          TAD K200

3453 5156 3374          DCA MVWDPG
3454 5157 1774          MOVWDX, TAD I MVWDPG
3455 5160 3773          DCA I MOVWDX
3456 5161 1774          TAD I MVWDPG                /COMPARE THE WORDS BEING RELOCATED
3457 5162 7041          CIA
3458 5163 1773          TAD I MOVWDX
3459 5164 7640          SZA CLA
3460 5165 7402          HLT                /COMPARE ERROR DURING RELOCATION
    
```

```

3461 5166 2373          ISZ MOVWDX
3462 5167 2374          ISZ MVWDPG
3463 5170 2347          ISZ UPDWN
3464 5171 5357          JMP MOVWDX
3465 5172 5744          JMP I XRSCNT
3466 5173 0000          MOVWDX, 0
3467 5174 0000          MVWDPG, 0
3468 5175 3001          SIZPRG, BGN-PRGEND-1
3469 5176 5176          PRGEND, .

3470
3471
3472          0200          *200
3473 0900 4023          JMS PATCH
3474 0901 4023          JMS PATCH
3475
3476
3477          5200          *5200
3478
3479
3480 5900 1021          PATCH2, TAD          OPISEL /GET THE HARDWARE CONFIGURATION
3481 5901 7004          RAL                /PUT OPTION1 BIT IN BIT 0
3482 5902 7700          SMA          CLA          /IS OPTION 1 SELECTED?
3483 5903 5223          JMP          SLOWRN /NO, OVERLAY SECTIONS OF RANDY AND ERROR1
3484 5904 1263          TAD          K4772 /YES, SET UP A TALK LOOP TO PROM
3485 5905 3664          DCA I          OPIHND /PUT JMS I ACTLIN IN RTC8ER+2
3486 5906 1265          TAD          OPIOVR
3487 5907 3010          DCA          AUTO10
3488 5910 1266          TAD          ERRORS
3489 5911 3011          DCA          AUTO11
3490 5912 1240          TAD          MM20
3491 5913 3130          DCA          PATMOV
3492 5914 1410          TAD I          AUTO10
3493 5915 3411          DCA I          AUTO11
3494 5916 2130          ISZ          PATMOV
3495 5917 5214          JMP          ,=3
3496 5920 1262          TAD          R5771
3497 5921 3641          DCA I          OVR4
3498 5922 5423          JMP I          PATCH
3499
3500          /THIS SECTION OF CODE WILL OVERLAY LAST 2 LOCATIONS OF RANDY ALL OF ERROR1 EXCEPT LAST 2 LOC,
3501 5923 1236          SLOWRN, TAD          ACTOVR
3502 5924 3010          DCA          AUTO10
3503 5925 1237          TAD          STRRND
3504 5926 3011          DCA          AUTO11
3505 5927 1240          TAD          MM20
3506 5930 3130          DCA          PATMOV
3507 5931 1410          TAD I          AUTO10
3508 5932 3411          DCA I          AUTO11
3509 5933 2130          ISZ          PATMOV
3510 5934 5231          JMP          ,=J
3511 5935 5423          JMP I          PATCH
3512
3513 5936 5241          ACTOVR, ACGDOV-1
3514 5937 1412          STRRND, RANDY1-1
3515 5940 7760          MM20,          -20
    
```

3516	5741	3763	DVR4,	RESFT=1	
3517			/		
3518	5742	2230	ACGDV,	2230	
3519	5743	5226		5226	
3520	5744	2231		2231	
3521	5745	5226		5226	
3522	5746	6002		IOF	
3523	5747	7240	CLA		CMA
3524	5750	3231		3231	
3525	5751	3230		3230	
3526	5752	6272	CTF		70
3527	5753	4632		4632	
3528	5754	6001	IGN		
3529	5755	1344	CONTRD,	1344	
3530	5756	5601		5601	
3531	5757	0000		0	
3532	5760	7777		=1	
3533	5761	6500		6500	
3534			/		
3535			/		
3536	5762	5771	R5771,	5771	
3537	5763	4772	K4772,	4772	
3538	5764	3234	OP1HND,	RTCSER+2	
3539	5765	5766	OP1OVR,	OVRDOP1=1	
3540	5766	1317	ERRORS,	ERROR+4	
3541			/		
3542	5767	0000	OVRDOP1,	0	
3543	5770	2331		2331	
3544	5771	5720		5720	
3545	5772	1332		1332	
3546	5773	3331		3331	
3547	5774	6272	CTF		70
3548	5775	4730		4730	
3549	5776	5720		5720	
3550	5777	6500		6500	
3551	5300	7634		-144	
3552	5301	7634		-144	
3553	5302	6002	IOF		
3554	5303	1333		1333	
3555	5304	6272	CTF		70
3556	5305	5737		5737	
3557	5306	6520		6520	
3558					
3559					
3560	5307	1312	PATCH3,	TAD OWVT78	/VT78/ OVERWRITE TO RUN ON VT78 SYSTEM
3561	5310	3713	DCA I	XOWLVT	/VT78/
3562	5311	5714	JMP I	RFVTP	/VT78/
3563	5312	5767	OWVT78,	VT780W	/VT78/ OVERWRITE DATA
3564	5313	0612	XOWLVT,	OWLVT	/VT78/ OVERWRITE LOCATION
3565	5314	0042	RFVTP,	PATCH1	/VT78/
3566					
3567					
3568					
3569					

\$\$\$

0000	11111100	10000000	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0100	11111111	11111111	11111111	10000000	00000000	00000000	00000000	00000000	00000000
0200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0300	11111111	11111111	11111111	11111111	11111111	10011111	11111111	11111111	11111111
0400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0500	11111111	11111111	11111111	11111111	11111111	01111111	11111111	11111111	11111111
0600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0700	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1300	11111111	11111111	11111111	11111111	10000001	11111111	11111111	11111111	11111111
1400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1500	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1700	11111111	11111111	11111111	11111111	11111011	11111111	11111111	11111111	11111111
2000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
2100	11111111	11111111	11111111	11111111	11111101	11111111	11111111	11111111	11111111
2200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
2300	11111111	11111111	11111111	11111111	10011111	11111111	11111111	11111111	11111111
2400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
2500	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111101	11111111
2600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
2700	11111111	11111111	11111111	11111111	11111111	11111000	00001111	11111111	11111111
3000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
3100	11111111	11111111	11111111	11111111	11111101	11111111	11111111	11111111	11111111
3200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
3300	11111111	11111111	11111111	11111111	11111111	11111100	01111111	11111111	11111111
3400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
3500	11111111	11111111	11111111	11111111	11111111	11100000	00111111	11111111	11111111
3600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
3700	11111111	11111111	11111111	11111110	00000011	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 10000000 01111111 11111111

4400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4500 11111111 11111111 11111111 11111111 00001111 11111111 11111111 11111111

4600 11111111 11111111 11111111 11111111 11111111 11111111 11111000 00000000
4700 11111111 11111111 11111111 11111111 11111111 11000000 00000111 11111111

5000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111110

5200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5300 11111111 11111000 00000000 00000000 00000000 00000000 00000000 00000000

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700
    
```

```

A1      1653      AUTO10 0010      C8400 3543      C8N270 4034
A14     1651      AUTO11 0011      C87      4035      C8N3      4037
A1CHG   1150      B400    1650      C87700   3546      C8N4     4242
A1RPNNG 1151      B6201   0526      C88400   4530      C8NTR    4371
A200    1647      B7       0543      C88888W  4571      C8OCT    3572
A900    1124      BACK    4767      C88GET   3775      C8OCTA   4167
A9201   2144      BADDRS  1366      C8ACSV   3765      C8P212   4246
A7       0743      BCHNG   1347      C8BY1    3433      C8P215   4245
AACHNG  0551      BDATHR  1376      C8BY2    4474      C8P260   4244
AADPRS  1166      BDATHR  1375      C8BY3    4271      C8P7     4243
ASAVB   1760      BEGIN   0507      C8BY4    4536      C8PASS   0355
ASAVB   1761      BGN      0200      C8BY5    4327      C8PRN    4166
ABGN    3170      BGNCON  3025      C8CHAR   3576      C8PRNT   3571
ARNRY1  1164      BGTST   0524      C8CHG1   3562      C8PX     4774
AC4008  3736      BINSTR  0560      C8CHG2   3746      C8RER1   3563
ACDATA  1031      BLCNT   1485      C8CHG3   4154      C8RER2   3747
ACDMSK  1137      BNDCON  1551      C8CHG4   4361      C8RER3   4155
ACGD0V  5242      HNDOK1  1445      C8CHG5   4544      C8RER4   4362
ACHC    0765      BNDUK2  1457      C8CHKR   3174      C8RER5   4547
ACHNG   0353      BNDRY1  1435      C8CKP    4223      C8RETD   4015
ACL      7701      BNDTAD  3364      C8CKPA   3574      C8RETR   3713
ACMPS   4552      BNDTSV  4626      C8CKSW   3567      C8RTL    3365
ACSAVB  3376      BOTRES  4363      C8CNTR   3764      C8SETD   4014
ACSAVF  4524      BP6     1654      C8CR     4171      C8SETS   3712
ACSCB   4373      BRANDY  1755      C8CRL    3761      C8SM8    3176
ACTLIN  3372      BREFAD  1374      C8CKLF   3570      C8STOP   3776
ACTOVR  5236      BREANG  1350      C8DD1    3513      C8STS    4566
ACUTST  2623      BSAYA   2145      C8DD10   4460      C8SWIT   4164
ADATAH  1172      BSIAC   2575      C8DD11   4010      C8SWR    2766
ADATAT  1171      BSMENK  2576      C8DD2    4234      C8SWST   4146
ADD11   1126      C10     5073      C8DD4    4207      C8TMP1   4222
ADDNHE  5036      C100    4041      C8DD7    3703      C8TPE    4367
ADDRS   0745      C1777   1561      C8ECHO   4170      C8TTYI   3763
ADDRS0  3141      C200    1312      C8EROR   1750      C8TYL    3773
ADDRS5  1612      C400    1547      C8ERR    1351      C8TY     4165
ADRMSK  1131      C4008   4026      C8ERRR   3175      C8TYI    4370
AERNV1  0103      C400X   4737      C8FLSV   3766      C8TYP    3760
AERNV2  0110      C5       1556      C8GET    3320      C8TYPE   3575
ATNSTR  1167      C6201   1123      C8GETB   4174      C8UPA    4569
ALOPID  1176      C7       1567      C8GETS   4563      C8UPAR   3774
AMES    4157      C70     1560      C8HANG   4333      CADDR8   1575
AMQMAT  1173      C80077  3547      C8HLT    4527      CAF       6007
ANDTST  1701      C80100  3551      C8IND    4567      CAL       6103
ANTYND  1175      C80400  3666      C8INM    4176      CC377    3231
ARANDY  1163      C8100   3670      C8INMD   3772      CC8BY4   4175
APEPAD  1170      C8177   3544      C8INQU   3573      CCHAR8   3771
AREONG  0766      C8200   3545      C8K277   4036      CCHNG    1562
APRANG  0302      C8240   3552      C8KPA    4372      CDATAT   1567
ASAME1  1161      C8277   3667      C8LF     4156      CDFINS   3344
ASAME2  1162      C8336   4025      C8LPCR   4570      CDHLT1   0540
ASAVA   1146      C83740  3550      C8M212  4030      CDHLT2   0544
ASAVB   1147      C840    4027      C8M8VB   3767      CHANGE   0202
    
```

CHAP	4304	CSHMAR	2770	FLDCHK	2555	HIGHLM	0374
CHARB	4173	CSIMMO	2772	FLDCNT	3167	HLT	7402
CHK1KF	1154	CSMLNK	2771	FLDFGF	1153	HLTFIL	0466
CHRACT	3255	CSWUP	0553	FLDFGR	1152	HLTOPR	1741
CIFPDD	0342	D4000	2825	FLDFLG	0500	HUMCIF	0744
CINSTR	2162	D6201	2070	FLDLIM	1144	HRERNG	2344
CKCOUT	3435	DATADN	2745	FLDMSK	1127	ICHNG	2567
CKSWC8	4561	DATAHR	0752	FLDRAN	1156	ILLMQ	1703
CLBRGN	3032	DATATH	0751	FLDRET	0761	ILLOP1	1664
CLASWP	2671	DATFN	1372	FLDXKN	2157	ILLOP2	1672
CLCT	6136	DCATST	1255	FLGRTC	3160	INACDV	3273
CLLF	6135	DCHNG	5074	FLGXMT	3157	INDAD	0750
CLSW	6137	DECSWP	1116	FLINK	0771	INDEXA	3655
CNT	2425	DINSTR	2345	FLMES	4554	INDMSK	1133
CNTFND	3245	DIRFLG	0213	FLSAV8	3375	INITFO	5107
CNTR2	0246	DM1	4040	FLSAVE	4526	INMODE	4305
CNTR3	0247	DM2	4031	FLSC8	4374	INQUC8	4562
CNTRLC	4345	DM41	4032	FLXRET	2160	INSGEN	1601
CNTRLG	5001	DOCNT	3452	FMQDAT	0772	INSMK	1642
CNTRLL	3714	DONEA	3631	FRERNG	1747	INSOK	1520
CNTRLO	3721	DONEMQ	1362	GACTRT	2366	INSTR	0746
CNTRLO	3671	DOPACK	3414	GADDRS	2367	INT	0003
CNTRLS	3675	DOSET	3454	GCHNG	2147	INTAC	3137
CNTVAL	3455	DRERNG	5103	GDATA0	2371	INTERS	3103
CDMCHK	2227	DVINAC	3161	GDATAH	2370	INTLND	3140
CDMPAR	1206	EINSTR	1754	GENADD	1043	INTMQD	1757
CDN7	0122	ENDCHA	2422	GENFLD	1001	INTOPR	2175
CONACD	1140	ENDCNT	5106	GENIND	1072	INTRET	3142
CONADR	1132	ENDIT	4143	GENINS	1055	INTSET	2137
CONFLD	1130	ENDOFF	3172	GENMQD	1037	IOF	6002
CONFLG	1143	ERR1	4137	GETC8	4376	ION	6001
CONYND	1134	ERRC8	3363	GETCH1	4075	IRERNG	2570
CUNYNS	1643	ERRMES	4477	GETDAT	3656	ISZTST	1234
CUNVOT	1136	ERROPR	2306	GETSWR	0356	JCHNG	2764
CUNMQD	1142	ERROR	1313	GETWD	2112	JMPJM8	0665
CONRAN	5006	ERROR1	1415	GLNKDN	2374	JMPST	1304
CONSET	3162	ERROR2	0724	GLPSWO	1776	JMBLOC	0762
CONST1	1105	ERROR5	5266	GMQDAT	2375	JMBLOC	0763
CONRT2	1112	ERRPSR	1332	GMQDON	1775	JMBRET	0776
CONRLM	3166	ERRRET	1352	GOITA	3646	JMSTST	1267
CONTRD	5255	EXECH	4302	GOPRET	2365	JRERNG	2765
COPRST	2773	EXITA	3643	GOTOA	3684	K0700	4717
CP212	4614	EXPRET	2751	GRANFL	2363	K1	2061
CP215	4613	FOINIT	0334	GRERNG	2150	K10	0501
CP400	4247	FCHNG	1746	GSMLNK	2373	K100	2067
CREPAD	1574	FILALL	0527	GSMLNK	2372	K14	2064
CREANG	1563	FILCNT	4241	GTCHAR	3222	K177	1646
CRLPC8	4556	FILL	5040	GTF	6004	K2	2062
CROPCG	4577	FILLER	4240	HALT	1335	K20	2065
CSAVA	1553	FILRND	3001	HCHNG	2343	K200	5072
CSAVB	1554	FINSTR	1367	HDNES	4740	K23	3354
CSAVC	1555	FIRST	3102	HGLIM	1145	K237	3144

K24n	3276	M36	3355	NOTIND	0625	POS200	2071
K37	1546	M3777	2565	NOTJJ	0650	POWERF	3163
K4	2063	M4	4615	NTCLAS	4465	PONFAL	3246
K40	2066	M400	0215	NTSTOP	4573	PRBG	1576
K400	0250	M5	0121	O6201	1733	PRBGN	5123
K4000	2434	M60	2305	OADDRS	1773	PRGEND	5176
K4772	5263	M6000	1645	OCTAC8	4560	PRGRET	3376
K5771	0124	M7	5071	OERRDR	2767	PRGSIZ	1552
K5772	0125	MAXFLD	0573	OFIELD	2176	PRNTC8	4557
K6201	0756	MDTMSK	1135	OP1	3370	PSIE	6665
K6202	0757	MEMDAT	1025	OP1HND	5264	PSKE	6663
K70	1125	MESA	4150	OP1OVR	5265	PSKF	6661
K7000	1644	MESAC	4812	OP1SEL	0021	PSRERR	3371
K7577	4721	MESERR	4550	OP1SET	5105	PSTB	6664
K7600	4716	MESFL	4520	OP28EL	0022	PTSR	4056
K7700	2517	MESH	4773	OPERR1	1734	PTSTOR	3541
K7721	1655	MESHQ	4515	OPRBN	1656	R5771	5262
K7770	0760	MESPAS	3457	OPRCOM	2706	RAN1	1543
K7777	2624	MESPC	4907	OPRER1	2376	RAN2	1544
KCHNG	3147	MIN37	0123	OPRERR	2744	RANCON	2127
KTE	6035	MH20	5240	OPRETF	2747	RANDY	1401
KLLCN1	3331	MH4	3023	OPRHLT	2364	RANDY1	1413
KLLCN2	3332	MH5	3100	OPRINT	0637	RANFLD	0754
KTL	3101	MH7	1545	OPROVR	0120	REDOA	3620
KILLIT	3366	MHM55	3274	OPRPNT	1752	REFAD	0747
KJMS	0764	MOVDMN	3157	OPRRET	2675	RELGO	3367
KKCDF	3143	MOVQVR	0073	OPRSET	2054	RESBOT	3334
KPERNG	3150	MOVUP	0225	OSINAC	2173	RESCNT	5035
LAS	7604	MOVNDX	5173	OSMLNK	2174	RESET	3264
LCBKA	4344	MQA	7501	OUT	3241	RESEIT	0256
LCNG	3361	MQDAT	1363	OVR4	5241	RETFLD	0525
LFR1	4141	MQDATA	0753	OVRLAY	0115	RETHR	0504
LMTS	1522	MQDMSK	1141	OVRLY1	0117	RETN	4714
LINKN	2746	MQDONE	2750	OVRP1	5267	RETPNT	0004
LINRT	1361	ML	7421	OWLVT	0612	RETPRG	3131
LINRSV	1360	MMES	4553	OWVT8	5312	RETTDF	2705
LNKAV	1174	MOSAV8	3374	PAGJMP	4574	RETURN	1354
LOC900	0126	MOSAVE	4525	PASCNT	3453	RFVT	0613
LOC901	0127	MOSC8	4375	PASMES	3566	RFVTP	5314
LODID	0601	MRIERR	2161	PATCH	0023	RWF	6244
LOWLIM	0214	MRIQVR	0116	PATCH1	0042	ROLBK	0401
LPCNT	0252	MRIPTN	1751	PATCH2	5200	ROLFLG	2574
LPSND	2737	MVWDPG	5174	PATCH3	5307	ROLLUP	0315
LPERNG	3362	NEG10	2072	PATCCH	0070	RSCNT	0376
M0600	4720	NEG100	3277	PATMOV	0130	RSCNTX	0370
M10	0503	NEG14	1652	PCLF	6662	RTCFLG	3227
M101	3145	NEG20	2304	PCMES	4551	RTCSE	3232
M13	2536	NEG6	2460	PCSAV	4523	RTPLG	0572
M14	2424	NEWDA	0475	PCSAVE	1311	RTPLGF	0574
M20	0502	NEWDF	0457	PCST	2073	RTFLGR	0571
M20n	0251	NOSET	3445	PNTBUF	4331	RTLINK	0557
M24	3333	NOTAUT	0611	POS10	2250	RTMQD	0556

SAMP1	1463	SI2PRG	5175	UPRLIM	1550	XXXPRT	1767
SAMP2	1473	SKPFLG	2451	VINSTR	4770	XFDMSK	5126
SAV7	4163	SLOWRN	5223	VT7BGL	4700	XFIELD	2775
SAVFSW	1564	SLUXMT	3156	VT7BOW	5767	XFILL	5114
SAVINK	0755	SMACHK	2202	WAIT	0360	XFIRST	5104
SAVCP2	0071	SNLCHK	2220	WAITEN	3300	XFLD	0354
SAVSI	3565	SPL	6102	XACDMS	5140	XFLDFG	5121
SAVSWR	0333	START	1756	XACBIS	2352	XFLDLM	1573
SBE	6101	STARTP	5025	XADD1	3173	XFLRND	5115
SEL01	3564	STUPNT	3542	XAD RMS	5130	XGENTI	1155
SEL02	4162	STRFLD	0005	XASAVA	0364	XGETWD	1357
SEL02A	4366	STRNWL	3152	XASAVB	0365	XGNFLD	5124
SEL02B	4561	STRRNO	5237	XRGCON	0372	XGO	0576
SEL02C	3762	SUBADD	0212	XBGRAN	0371	XHALT	1565
SEL02X	4772	SWAP1	0216	XBNDCN	1160	XINDMS	5134
SEL0P1	3035	SWAP2	5150	XBSAVA	1356	XINSGN	2774
SELOP2	3456	SWAPDN	0422	XCBCPK	4250	XINBMS	5132
SERRTC	3165	SWAPUP	0415	XCBCN	4172	XINSTR	0362
SERXMT	3164	SWITCH	0020	XCBCNT	3601	XINT	2151
SETCON	5001	SWPFLD	0463	XCBCRL	4224	XJMSLC	2154
SETDUX	3151	SWPUP	0452	XCBCSH	4273	XLF	4775
SETPLG	0324	XXPSN	4033	XCBCRR	4401	XLIMIT	3154
SETTNT	0357	SZACHK	2211	XCBCINQ	3726	XLOOP	0367
SETOP1	3036	SZPRG	0245	XCBLF	4531	XLWLM	0570
SETRET	0710	TABLA	3657	XCBCDT	4201	XMDTMS	5136
SETSRK	2226	TABLB	3752	XCBCPAS	3401	XMQDMS	5142
SFLNFG	0433	TABLC	5125	XCBCPNT	3506	XMTFLG	3226
SIMAC	2752	TADTST	1223	XCBCPSW	4042	XMTSER	3201
SIMACL	2666	TEMP	3024	XCBSW	3466	XMTSLU	3230
SIMBSW	2477	TINT	2143	XCCTTY	3475	XNTIND	0363
SIMCAM	2663	TMPCNT	4147	XCCTYP	4306	XDWLVT	5313
SIMCLA	2661	TSAME	1503	XCHNGE	0552	XPABSC	4555
SIMCLR	6160	TSFLDF	2572	XCLSWP	2362	XPCSAV	2155
SIMCMA	2401	TSTCHA	4107	XCNACD	5141	XPSR	4776
SIMCML	2426	TSTIN1	0562	XCNADH	5131	XPSW	3770
SIMTAC	2435	TSTIN2	0563	XCNFLD	5127	XPTCH3	0072
SIMLNK	2753	TSTIN3	0564	XCNFLG	3153	XRANCN	5110
SIMMQ	2754	TSTIN4	0565	XCNIND	5135	XREFAD	0366
SIMMQA	2644	TSTIN5	0566	XCNINS	5133	XRETFI	1373
SIMMQI	2252	TSTIN6	0561	XCNMDT	5137	XRETHR	1165
SIMMQL	2640	TSTPC	1355	XCNMQD	5143	XRETPC	2156
SIMOP1	2001	TSWPDN	2573	XCNT	0271	XRNFLD	1364
SIMOP2	2201	TYCNT	3275	XCNTR3	5122	XROLBK	0361
SIMRAL	2461	TTYLC0	4364	XCTLC	4160	XRSNT	5144
SIMRAR	2442	TTYLPT	4332	XCTG	4161	XRTOPF	1774
SIMRTL	2537	TYPE	4617	XDATAH	1762	XSIMAC	1763
SIMPTR	2520	TYPECT	4616	XDOLPT	4321	XSIMMQ	1766
SIMMA	2611	TYPEIT	4600	XDOBW	4572	XSIZE	0575
SIMMNL	2626	UPARCB	4365	XDRFLG	5120	XSMACL	2361
SIMMWP	2652	UPAROW	4016	XENDPR	0375	XSMBSW	2170
SIMQZA	2601	UPDNW	5147	XERR2	1572	XSMCAM	2360
SIMTWC	2042	UPPERL	2571	XERROR	1353	XSMCLA	2357

XSMMA	2163	ZLIMIT	0555				
XSMNL	2164	ZSETOP	2353				
XSMTAC	2165	ZUPLIM	5112				
XSMVINK	1765						
XSMWQA	2355						
XSMWQI	1772						
XSMWQL	2354						
XSMWP1	1770						
XSMWP2	1771						
XSMWAL	2167						
XSMWAR	2166						
XSMRTL	2172						
XSMRTR	2171						
XSMMA	2346						
XSMNL	2350						
XSMWQP	2356						
XSMQZA	2347						
XSTPLD	0554						
XSTM6	4722						
XSVINK	1764						
XSWAP1	5116						
XSWAP2	5117						
XTABLA	3750						
XTABLB	3751						
XTYANT	3155						
XUPRL	0373						
XUPTM	1157						
XVT7BG	0767						
XNDMOW	2153						
XXCNT	2776						
YHALT	0770						
Z100	2243						
Z20	2245						
Z200	2246						
Z320	2247						
Z40	2244						
ZADPRS	4771						
ZASAVA	1370						
ZASAVB	1371						
ZBSAVA	1570						
ZCHANG	4766						
ZCNFLG	2152						
ZCNT	0775						
ZEXPRT	2351						
ZFILL	3171						
ZFIWD	0774						
ZFLNLM	5111						
ZGETWD	0773						
ZHIGH	5113						
ZINAD	1571						
ZINSTR	1566						
ZJMKRT	1753						

ERRORS DETECTED: 0
 LINKS GENERATED: 0
 RUN-TIME: 7 SECONDS
 4K CORE USED

A1	1159#	1177	1207																	
A14	1157#	1170																		
A1CHG	670	786#																		
A1RRNG	786	788#																		
A200	1144	1155#																		
A400	726	763#																		
A6201	1149	1355	1379#																	
A7	441	634#																		
AACHNG	289	405#																		
AADDRS	714	715	801#																	
AASAVA	1146	1242#																		
AASAVB	1142	1147	1243#																	
ABGN	1905	2033#																		
ABNRY1	712	719	735	799#																
AC4008	2624	2630#																		
ACDATA	698#																			
ACDMSK	699	774#	3439																	
ACGDDV	3513	3518#																		
ACHG	433	655#																		
ACHNG	242	264#																		
ACL	11#	367	1832	2246#																
ACMES	3115	3192#																		
ACSAVE	2150	2197#																		
ACSAVE	2197	2650	3063	3087	3119	3165#														
ACSCN	2938	3063#																		
ACTLIN	2193#																			
ACTOVR	3901	3513#																		
ACUTST	1739	1740	1751	1752	1758#	1760	1769	1794	1797											
ADATAH	701	805#																		
ADATAT	697	804#																		
ADD11	691	765#																		
ADDONE	2036	3367#	3393																	
ADDRS	472	478	482	540	636#	801	942	1102	1283	1549	3324									
ADDRS0	1978	2006#																		
ADDRS5	1124	1125#	1127																	
ADRMSK	710	768#	3431																	
AEROV1	115#	127																		
AEROV2	121#	129																		
AINSTR	727	802#																		
ALOPID	741	809#																		
AMES	2755	2828#																		
AMQDAT	707	806#																		
ANDTST	414	817#																		
ANTIND	730	808#																		
ARANDY	673	694	698	704	708	731	798#													
AREFAD	723	803#																		
AREPNG	655	657#																		
ARRANG	218#																			
ASAME1	721	737	796#																	
ASAME2	739	797#																		
ASAVA	274	717	781#	944	1242															
ASAVB	275	782#	945	1243																
AUTO10	39#	84	89	109	2163	2170	3487	3492	3502	3507										

SEQ 0100

AUTO11	40#	86	91	110	2165	2172	3489	3493	3504	3508							
B400	1156#	1166	1203														
B6201	125	328	361#	387	390												
B7	174	396#															
BACK	3295	3322#															
BADDS	486	407	942#														
BCHNG	415	926#															
BDATHR	419	439	451	869	876	890	950#										
BDATH	418	438	847	881	949#												
BEGIN	791	420#															
BGN	35	139	140	146#	187	420	1080	1103	1240	2033	3426	3468					
BGNCON	779	280	1923#														
BGDST	175	379#															
BINSTP	171	413#															
BGPCHT	434	844	941#														
BNDCON	795	997	1079#														
BNDDR1	1004#																
BNDDR2	1009	1014#															
BNDRY1	799	992#	999	1013	1014	1017											
BOOTAD	2162	2187#															
ROOTSV	2187	3242#															
BOTRES	3046	3055#															
BP6	1160#	1181															
BRANDY	1117	1239#															
BREFAD	456	865	881	948#													
BREFNG	426	928#															
BBAVA	434	1097	1352	1353	1358	1380#											
BIMAC	1470	1583	1603	1616	1620	1635	1639	1653	1655	1659	1673	1677	1692	1696			
	1723#																
BSMLNK	1592	1595	1612	1622	1631	1641	1649	1661	1669	1679	1688	1698	1724#				
C10	3467	3398#															
C100	2696	2713#															
C1777	1066	1087#															
C200	497#	912															
C400	480	1077#															
C4008	2702#	2744															
C400X	3305	3315#															
C5	1045	1084#															
C6201	725	762#															
C7	1063	1085#															
C70	1060	1086#															
C80077	2407	2424#															
C80100	2412	2426#															
C80400	2476	2533#															
C8100	2435#																
C8177	2364	2421#															
C8200	2365	2422#															
C8240	2413	2427#															
C8277	2404	2534#															
C8336	2493	2701#															
C83740	2410	2425#															
C840	2703#	2761															
C8400	2776	2280	2298	2420#													

SEQ 0101

C87	2709#	2802															
C87700	2196	2423#															
C88400	3096	3169#															
C8885#	3179	3207#															
C88GFT	2449	2574	2658#														
C8ACSV	2474	2479	2650#														
C8BY1	2283	2285	2301#	2315													
C8BY2	3138	3140	3153#														
C8BY3	2448	2952	2960#														
C8BY4	2442	3179#															
C8BY5	3011	3020#															
C8CHAR	2366	2367	2444#														
C8CHG1	2255	2431#															
C8CHG2	2452	2634#															
C8CHG3	2665	2824#															
C8CHG4	2449	3052#															
C8CHG5	3072	3189#															
C8CHKP	1987	2037#															
C8CKP	2466	2475	2879#														
C8CKPA	2442#																
C8CKSW	2779	2297	2437#														
C8CNTR	2476	2649#															
C8CR	2672	2808	2810	2838#													
C8CRL	2406	2646#															
C8CRLF	2288	2296	2438#														
C8DO1	2195#	2416															
C8DO10	3101	3134#															
C8DO11	2675	2679#															
C8DO2	2499#	2901															
C8DO4	2467#	2876															
C8DO7	2563	2570#															
C8ECHO	2764	2771	2837#														
C8EROR	1223	1234#															
C8ERR	419	929#															
C8ERRP	1496	2038#															
C8FLSV	2482	2651#															
C8GET	2143#	2151	2658	2841	3066	3201											
C8GETB	2482	2841#															
C8GETS	3150	3154	3201#														
C8HANG	3015	3027#	3036														
C8HLT	3147	3168#															
C8IND	3177	3205#															
C8INM	2763	2843#															
C8INMD	2499	2547	2655#														
C8INQU	2300	2441#															
C8K277	2710#	2806															
C8KPA	3032	3062#															
C8LFL	2784	2827#															
C8LFCR	3178	3206#															
C8M212	2704#	2782															
C8MOSV	2486	2652#															
C8N270	2708#	2798															
C8N3	2711#	2769															

SEQ 0102

HOMCLIF	473	476	493	541	635*													
HPRRNG	1428	1530*																
ICHNG	1461	1716*																
ILLMO	1174	1179	1184	1186*														
ILLOP1	1169*																	
ILLOP2	1168	1176*																
INACDV	2026	2108	2116	2119*														
INDAD	435	451	459	639*	109R													
INDEXA	2487	2496	2517	2523*														
INDMSK	733	770*	3435															
INITFO	3460	3414*																
INMODE	2655	2843	2979	2983	2987*	3205												
INQUC8	3139	3200*																
INSGEN	792	1116*	1134	1148	1173	1183	1888											
INSMSK	1118	1150*	3433															
INSOK	1043	1050*																
INSTR	272	413	462	481	524	637*	802	943	1095	1238	1398	1531	3323					
INT	32	33*	254	1376														
INTAC	1975	1999	2004*															
INTERS	33	1389	1975*															
INTLNK	1977	1997	2005*															
INTMQD	1194	1241*																
INTOPR	1117	1409*																
INTRET	1979	2002	2007*															
INTSET	269	1372*	1375															
IOF	15*	115	121	2088	2139	2161	3086	3522	3553									
IUN	14*	2001	3361	3528														
IPRRNG	1716	1718*																
ISZTST	416	846*																
JCHNG	1731	1879*																
JMPJMS	464	524*																
JMPTST	419	889*																
JMSLOC	606	608	611	649*	1392													
JMSLOD	609	613	650*															
JMSRET	614	665*	1237															
JMSTST	418	875*																
JRRRNG	1879	1881*																
K0700	3287	3298*																
K1	1285	1319*																
K10	304	321	357*															
K100	1273	1325*																
K14	1293	1303	1322*															
K177	1141	1154*																
K2	1289	1320*																
K20	1281	1323*																
K200	3197*	3452																
K23	2164	2176*																
K237	1860	2009*																
K240	2061	2122*																
K37	1056	1076*																
K4	1297	1309	1321*															
K40	1277	1324*																
K400	190*	199																

SEQ 0107

K4000	1494	1598*																
K4772	3484	3937*																
K5771	66	137*																
K5772	68	138*																
K6201	454	466	531	645*														
K6202	475	489	535	646*														
K70	674	677	764*															
K7000	1131	1135	1152*															
K7577	3292	3300*																
K7600	3282	3297*																
K7700	1654	1663*																
K7721	1161*	1187																
K7770	436	647*																
K7777	1750	1759*	1767															
KCHNG	1496	2015*																
KIE	18*	1947																
KILCN1	2131	2134	2154*															
KILCN2	2133	2136	2155*															
KILL	1939	1970*	2189															
KILLIT	2055	2081	2130	2189*														
KJMS	610	612	651*															
KKCDF	1981	2008*																
KRPRNG	2015	2017*																
LAS	12*	136*																
LCAKPA	3031	3034	3037*															
LCHNG	2045	2183*																
LERP1	2787	2793	2808*															
LIMITS	1054*	1070	2021															
LINKDN	412	660	937	1554	1829	1852	1869*											
LINKRT	826	937*																
LINKSV	824	936*																
LNKSAV	703	807*																
LOC200	67	139*																
LOC201	69	140*																
LOOPIID	277	435*	809															
LOWLIM	160*	203	240	421														
LPCNT	194*	664	930	941														
LPSWO	1256	1860*																
LPRRNG	2183	2185*																
M0600	3288	3299*																
M10	111	359*																
M101	1962	2010*																
M13	1471	1682*	1690															
M14	1467	1586*																
M20	320	358*																
M200	167	191*																
M24	2132	2156*																
M36	2173	2179*																
M3777	1703	1711*																
M4	3222	3231*																
M400	161*	239																
M5	107	132*																
M60	1473	1479	1485	1493*														

SEQ 0108

STOPAI	2418#	2659	3209						
STRFLD	35#	1125							
STPNWL	1059	2019#							
STRPHD	3503	3514#							
SUBAUD	153	158#	241						
SWAP1	163#	3421							
SWAP2	3472	3447#							
SWAPDN	302	308#	1721						
SWAPUP	303#	310							
SWITCH	44#	106#							
SWPFLD	329	341#							
SWPUP	332#								
SXPSW	2707#	2753	2813						
SZACHK	1425	1427	1429#						
SZPRG	163	187#	426						
TABLA	2525#	2637							
TABLB	2638	2639#							
TABLE	3339	3428#							
TADTST	415	836#							
TEMP	1901	1912	1920#						
TINT	1374	1376#							
TMPCNT	2770	2778	2817#						
TSAME	1023	1032	1037#	1047	1049	1051			
TSFLDP	1705	1720#							
TSCHA	2765	2772	2781#	2805					
TSTIN1	415#								
TSTIN2	416#								
TSTIN3	417#								
TSTIN4	418#								
TSTIN5	419#								
TSTINS	379	414#							
TSTPC	417	836	850	864	875	889	933#		
TSWPDN	1708	1721#							
TTYCNT	2022	2059	2064	2121#					
TYLCO	3044	3056#							
TYLPT	2656	3003	3023#	3056					
TYPE	3219	3221	3224	3234#	3240				
TYPECT	3723	3225	3232#						
TYPEIT	2188	3217#	3227						
UPARC8	3045	3057#							
UPAROW	2657	2692#	2698	3057	3203				
UPDOWN	3338	3353	3408	3446#	3448	3463			
UPPERL	1702	1719#							
UPRIM	781	794	1058	1059	1062	1067	1078#	1719	3417
VINSTR	3285	3291	3293	3323#					
VT78GL	658	3280#							
VT78OW	444#	3563							
WAIT	211	270#							
WAITEN	270	2125#	2140						
XACDMS	3439#								
XACSIM	1453	1510	1536#						
XADD1	1917	2036#							
XADPMS	3431#								

XASAVA	205	274#										
XASAVB	206	275#										
XBGCON	280#											
XBGPAN	279#											
XBNDON	692	795#										
XBSAVA	857	866	882	934#								
XCCKP	2037	2442	2937#	2959	2961	3030	3035	3062				
XCRCN	2676	2839#										
XCRCNT	2473#	2480	2508	2510	2513	2515	2567	2599	2608	2649	2839	3061
XCRCRL	2019	2438	2646	2838	2891#	2902	3196	3206	3328			
XCBECH	2837	2975#	2985									
XCBFRR	929	1234	2038	2186	3085#	3093	3146	3157				
XCBIHQ	2441	2621#	2626	2628	3200							
XCBLF	2827	3174#										
XCBOCT	2440	2836	2861#	2877	3198							
XCBPAS	767	2271#	2282	2301	2302							
XCBPNT	2190#	2392	2394	2398	2409	2439	2835	3197	3327			
XCRP6W	2653	2739#	2746	2752	2809	2833	3329					
XCR6W	268	1882	2339#	2347	2437	3199						
XC8TTP	2160#	2368	2648	3060								
XC8TYP	2143	2645	2834	3001#	3021	3059						
XCHNGE	405	407#										
XCLSWP	1490	1544#										
XCNACD	3440#											
XCNADR	3432#											
XCNFLD	3430#											
XCNFLG	1924	2020#										
XCNIND	3436#											
XCNINS	3434#											
XCNMOT	3438#											
XCNMOD	3442#											
XCNT	197	209#	1890	2190								
XCNTR3	3364	3368	3369	3370	3381	3384	3389	3425#				
XCTLG	2829#											
XCTLG	2830#											
XDATAH	1190	1244#										
XDOLPT	3005	3012#										
XDOSW	3180	3208#										
XDRFLG	3355	3403	3423#									
XENDPR	165	168	229	283#								
XERR2	987	1099#										
XERROR	916	931#										
XEXPRT	1198	1249#										
XFDMSK	3428	3429#										
XFIELD	1835	1889#										
XFILL	3492	3419#										
XFIRST	3341	3411#										
XFLD	720	266#										
XFLDFG	3356	3424#										
XFLDLM	1061	1100#										
XFLRND	3483	3420#										
XGENTI	718	792#										
XGETWD	858	867	883	935#								

XGNFLD	3765	3427#							
XGO	353	327#							
XHALT	974	976	978	1094#					
XINDMS	3435#								
XINSGN	1863	1888#							
XINSMS	3133#								
XINSTR	198	202	272#						
XINT	1773	1389#							
XJMSJC	1333	1392#							
XLFL	3112	3328#							
XLIMIT	1025	2021#							
XLOOP	708	277#							
XLWLM	293	334	421#						
XMDTMS	3437#								
XMQDMS	3441#								
XMTFLG	2024#	2053	2071#	2104	2112				
XMTSER	2029	2049#							
XMTSLU	2023	2058	2062	2066	2073#				
XNTIND	701	273#							
XOWLVT	3461	3564#							
XPASSC	3175	3195#							
XPCSAV	1334	1335	1393#						
XPSR	3313	3329#							
XPSW	2627	2653#							
XPTCH3	79	104#							
XRANCN	3344	3350	3415#						
XREFAD	707	276#							
XRETFI	901	947#							
XRETHR	726	800#							
XRETFC	1337	1394#							
XRNFLD	999	940#							
XPOLBF	228	233	271#						
XRSCNI	3443#	3465							
XRTOPF	1701	1254#							
XSIMAC	1191	1245#							
XSIMMQ	1195	1215	1248#						
XSIZE	332	426#							
XSMACL	1487	1543#							
XSMBSW	1305	1404#							
XSMCAM	1484	1542#							
XSMCLA	1481	1541#							
XSMCMA	1279	1399#							
XSMCMI	1283	1400#							
XSMIAC	1287	1401#							
XSMLNK	1193	1247#							
XSMMOA	1475	1539#							
XSMMOI	1210	1252#							
XSMMLL	1472	1538#							
XSMOPI	1205	1250#							
XSMOP2	1209	1251#							
XSMRAL	1299	1403#							
XSMRAR	1296	1402#							
XSMPTL	1311	1406#							

SEQ 0115

XSMRTR	1308	1405#							
XSMMA	1426	1532#							
XSMNL	1440	1534#							
XSMWP	1478	1540#							
XMSZA	1433	1533#							
XSTFLD	386	409#							
XSTMS	2039	3302#	3307	3314					
XSVLNK	1192	1246#							
XSWAP1	3407	3421#							
XSWAP2	3405	3422#							
XTABLA	2488	2637#							
XTABLB	2516	2638#							
XTYCNI	1957	1963	2022#						
XUPERL	224	281#							
XUPLJM	487	794#							
XVT78G	444	658#							
XWDMOV	1366	1391#							
XXCNT	1864	1890#							
XHALT	619	621	623	625	627	629	631	659#	
Z100	1423	1456#							
Z20	1437	1458#							
Z200	1451	1459#							
Z320	1460#	1467							
Z40	1430	1457#							
ZADDRS	3280	3324#							
ZASAVA	914	944#							
ZASAVB	915	945#							
ZBSAVA	986	1097#							
ZCHANG	3210	3213	3320	3321#					
ZCNFLG	1363	1390#							
ZCNT	432	664#							
ZEXPRT	1449	1500	1535#						
ZFILL	1911	1916	2034#						
ZFIND	622	663#							
ZFLDLM	3372	3416#							
ZGETWD	618	662#							
ZHIGH	3378	3387	3388	3418#					
ZINDAO	975	983	1098#						
ZINSTR	979	1095#							
ZJMSRT	1139	1164	1237#						
ZLIMIT	398	410#							
ZSETOP	1454	1469	1537#						

SEQ 0116

