KT11-C

EXERCISER

MD-11-DCKTG-E

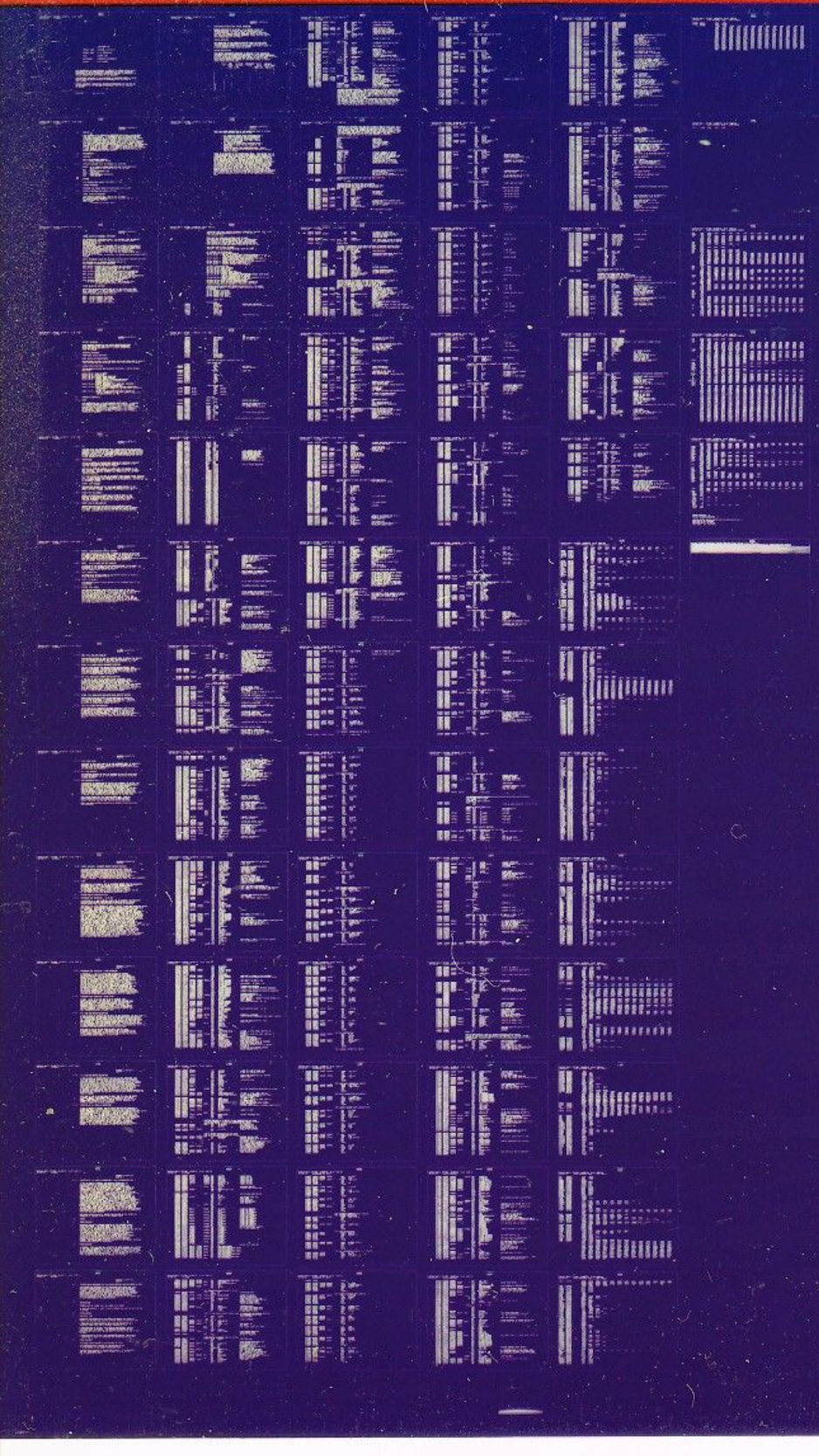
EP DCKTG E DL A

OCT 1976

FICHE 1 OF 1

digital

Made in U.S.A.



```
.REM    *
```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DCKTG-E-D
PRODUCT NAME:	KT11-C EXERCISER
DATE:	JANUARY 1976
MAINTAINER:	DIAGNOSTIC PROGRAMMING
AUTHOR:	RICK FADDEN

1.0 ABSTRACT

THIS PROGRAM IS AN INTERACTIVE EXERCISER FOR A PDP-11/45 EQUIPPED WITH THE KT11-C OPTION. IT PERFORMS A TEST OF INSTRUCTIONS AND CONCURRENT OPERATIONS OF I/O EQUIPMENT WHILE RELOCATING THRU MEMORY. IT PROVIDES NUMEROUS MODES OF TESTING, FROM 4K EXECUTION WITH THE KT11-C TURNED OFF AND ONLY KERNEL MODE IN USE, TO 128K EXECUTION WITH EACH USER PAGE MAPPED SEQUENTIALLY TO EVERY 4K BANK OF MEMORY, TC11 AND RF11 BUFFER AND CODE RELOCATION THRU ALL MEMORY, AND SUPERVISOR MAPPING OF OTHER I/O DEVICES. THIS PROGRAM IS NOT TO BE CONSIDERED A TOTAL CHECK OF THE SYSTEM. IF AN ERROR IS DETECTED IN AN I/O DEVICE, IT WILL PROBABLY BE NECESSARY TO CORRECT THE MALFUNCTION WITH THE RESPECTIVE DIAGNOSTIC FOR THAT DEVICE.

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP-11/45 STANDARD COMPUTER
KT11-C MEMORY MANAGEMENT OPTION

2.1.1 OPTIONAL HARDWARE THAT THE PROGRAM WILL EXERCISE

MEMORY	UP TO 128KW OF MEMORY-DOES NOT HAVE TO BE CONTIGUOUS, BUT BLOCKS OF LESS THAN 4KW WILL NOT BE USED
RF11	DISK
TC11	DECTAPE-TRANSPORT ZERO
KW11-L	LINE CLOCK
KL11	ASR33 OR ASR35 TELEPRINTER
LP11	LINE PRINTER

2.2 STORAGE

THIS PROGRAM USES MEMORY FROM 00000 TO 17760.

3.0 LOADING PROCEDURE

PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.

4.0 STARTING PROCEDURE AND SWITCH SETTINGS

4.1 NORMAL STARTING PROCEDURE

LOAD STARTING ADDRESS 200. (300 IF LP11 IS USED).
SET DESIRED MEMORY MANAGEMENT SELECTION SWITCHES (SEE 4.2)- ALL DOWN FOR WORST CASE TESTING.
PRESS START.

(CONTINUED ON THE NEXT PAGE)

4.1 NORMAL STARTING PROCEDURE (CONTINUED)

THE PROGRAM WILL IMMEDIATELY HALT. AT THE HALT, SET THE DESIRED DEVICE SELECTION SWITCHES (SEE 4.3) AND THE DESIRED DYNAMIC SWITCHES (SEE 5.1.2).
PRESS CONTINUE.
THE PROGRAM WILL PRINT A $ (UNLESS THE TTY OUTPUT IS SELECTED) AT THE END OF EACH BANK. IF SWITCHES 0,1 AND 2 WERE ALL DOWN WHEN START WAS PRESSED (SELECTING THE USE OF 4K PHYSICAL ADDRESS SPACE AS 32K VIRTUAL ADDRESS SPACE-SEE 5.3.1) AN ASTERISK WILL BE TYPED AT THE END OF A FULL PASS THRU ALL MEMORY (UNLESS THE TTY OUTPUT IS SELECTED).
NOTE THAT IF TTY OUTPUT IS SELECTED, THE DOLLAR SIGN AND ASTERISK ARE NOT PRINTED.

4.2 MEMORY MANAGEMENT SELECTION SWITCHES (INITIAL SWITCH REGISTER SETTINGS).

THE SWITCHES SET AT STARTUP DETERMINE THE WAY IN WHICH MEMORY IS MAPPED AND EXERCISED:

SW0=1 OR UP---INHIBIT THE KT11-C (SR0<0> WILL NOT BE SET AT ALL)
SW1=1 OR UP---INHIBIT USE OF SUPERVISOR AND USER MODES
(ALSO INHIBITS 4K AS 32K)
SW2=1 OR UP---INHIBIT 4K AS 32 K (ALSO INHIBITED IF EITHER SW0 OR SW1 IS SET)-SEE SECTION 5.3.1 FOR EXPLANATION
SW3=1 OR UP---INHIBIT RELOCATION OF RF11 AND TC11 CODE AND BUFFERS-SEE SECTIONS 5.3.4 AND 5.3.5 FOR AN EXPLANATION
SW4=1 OR UP---INHIBIT CYCLING ACCESS KEY OF SUPERVISOR PAGE 7
-SEE SECTION 5.3.7 FOR AN EXPLANATION
SW5=1 OR UP---INHIBIT VARIABLE CORE EXPANSION
=0 OR DOWN-CORE EXPAND UNLESS SW0, 1 AND 2 ARE ALL DOWN
(IN WHICH CASE 4K AS 32K IS RUN INSTEAD)

4.3 DEVICE SELECTION SWITCHES

THE DEVICE SELECTION SWITCHES ARE SET AT THE FIRST (AND ONLY) HALT. EACH SWITCH, IF SET, INHIBITS A SINGLE I/O DEVICE FROM BEING EXERCISED. IF A DEVICE DOES NOT EXIST, THE CORRESPONDING INHIBIT SWITCH DOES NOT HAVE TO BE SET.

SW0=1 OR UP---INHIBIT TTY OUTPUT
SW4=1 OR UP---INHIBIT LINE CLOCK
SW5=1 OR UP---INHIBIT LINE PRINTER
SW6=1 OR UP---INHIBIT RF11 DISK
SW7=1 OR UP---INHIBIT TC11 DECTAPE

4.4 RESTART PROCEDURE

USING RESTART ADDRESS 310 THE SWITCH REGISTER SETTINGS GIVEN PREVIOUSLY ARE USED (FOR BOTH MEMORY MANAGEMENT SELECTION AND DEVICE SELECTION). NO HALT OCCURS AFTER START IS PRESSED.

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

5.1.1 BASIC SWITCH SETTINGS-STARTUP

SEE SECTIONS 4.2 AND 4.3 FOR THE BASIC SWITCH SETTINGS USED AT STARTUP. THOSE SWITCHES ARE NOT RECHECKED AFTER THEY ARE INITIALLY STORED.

5.1.2 DYNAMIC SWITCH SETTINGS

THE FOLLOWING SWITCHES ARE RECHECKED PERIODICALLY DURING PROGRAM EXECUTION:

SW15=1 OR UP---HALT ON ERROR
SW14=1 OR UP---SCOPE LOOP
SW13=1 OR UP---INHIBIT ERROR PRINT OUT
SW12=1 OR UP---INHIBIT TRACE TRAPPING
SW11=1 OR UP---INHIBIT SUBTEST ITERATION AND INHIBIT TESTS WHICH USE ALL COMBINATIONS OF NUMBERS (SEE 5.2.1)
SW10=1 OR UP---INHIBIT PROCESSOR TEST (ONCE SET, PROCESSOR TEST IS PERMANENTLY INHIBITED)
SW09=1 OR UP---INHIBIT CYCLING SUPERVISOR MAPPING (SUPERVISOR PAGE CURRENTLY MAPPED WILL BE USED UNTIL THE SWITCH IS SET =0)
SW08=1 OR UP---INHIBIT CYCLING SUPERVISOR PAGE 7 ACCESS KEY (THE KEY CURRENTLY IN USE WILL REMAIN IN USE UNTIL THE SWITCH IS SET =0)

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

THIS SUBROUTINE CALL IS PLACED BETWEEN EACH SUBTEST. IT RECORDS THE STARTING ADDRESS OF EACH SUBTEST AS IT IS BEING ENTERED. IF A SCOPE LOOP IS REQUESTED, IT WILL JUMP TO THE START OF THE SUBTEST THAT THE SCOPE LOOP IS REQUESTED FOR. IF A SCOPE LOOP IS NOT REQUESTED, THERE WILL BE 256 ITERATIONS ON THAT SUBTEST BEFORE THE NEXT SUBTEST IS ENTERED. SWITCH 11 ON A 1 INHIBITS ITERATION OF SUBTESTS.

5.2.2 HLT

THIS EMT CALLS THE SUBROUTINE PRINT, WHICH PRINTS OUT THE LOCATION COUNTER AT THE TIME OF FAILURE, THE CONTENTS OF THE PROCESSOR STATUS REGISTER, THE CONTENTS OF THE CURRENT BANK COUNTER, AND THE STARTING ADDRESS (VIRTUAL) OF THE BACKGROUND PROCESSOR SUBTEST BEING EXECUTED AT THE TIME OF FAILURE. NOTE THAT THE LOCATION COUNTER WILL BE THE VIRTUAL ADDRESS OF THE HLT PLUS TWO.

5.2.3 TRAPCATCHER

THIS IS A SERIES OF INSTRUCTIONS STARTING AT LOCATION 0 DESIGNED TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS TO THE TRAP AND INTERRUPT VECTOR AREA OF MEMORY.

EACH VECTOR ENTRANCE ADDRESS IS LOADED WITH THE ADDRESS OF THE NEXT LOCATION. THE NEXT LOCATION IS LOADED WITH A HALT (000000). THUS AN ILLEGAL TRAP OR INTERRUPT WILL CAUSE A HALT AT THE TRAP LOCATION PLUS TWO.

IF A HALT OCCURS IN THE TRAP OR INTERRUPT AREA EXAMINE KERNEL REGISTER SIX. IT WILL CONTAIN THE CURRENT KERNEL STACK ADDRESS. THE CONTENTS OF THE CURRENT KERNEL STACK ADDRESS IS THE VIRTUAL PC AT THE TIME THE TRAP OR INTERRUPT OCCURRED.

5.2.4 EMTSRV (EMT HANDLER)

THIS ROUTINE DECODES THE EMT CALLS AND PASSES CONTROL TO THE CORRECT SERVICE ROUTINE. THE ROUTINES HANDLED BY EMT CALLS ARE PRINT (HLT CALL), EOBSRV (EOB CALL), AND RTIKEX (RTIK CALL).

5.2.5 RTIKEX (RTI VIA KERNEL)

SINCE I/O EXECUTION IN SUPERVISOR IS DONE BY MANY OF THE ROUTINES IN THIS EXERCISER, A MEANS IS REQUIRED TO RETURN FROM THE INTERRUPTS TO THE MODE THAT WAS IN EFFECT WHEN THE INTERRUPT OCCURRED. THE CALL RTIK IS USED TO DO THIS.

5.2.6 EOBSRV (END OF BANK SERVICE)

THE VARIOUS EXECUTION OPTIONS FOR THIS EXERCISER REQUIRE SPECIAL HANDLING WHEN THE END OF THE PROCESSOR TESTS IS REACHED IN A BANK. THIS SERVICE ROUTINE PERFORMS THE VARIOUS MAPPING FUNCTIONS, DEPENDING UPON THE INITIAL SWITCH REGISTER SETTINGS.

5.2.7 BEGINX (CORE EXPANSION SPECIAL HANDLER)

WHEN CORE EXPANSION IS UTILIZED, A NUMBER OF SPECIAL ACTIONS MUST BE TAKEN AT THE BEGINNING OF EACH BANK. THE SCOPE ROUTINE VECTOR IS LOADED TO POINT TO THE NEW BANK, AND IF TC11 AND RF11 CODE AND BUFFER RELOCATION IS ALLOWED, THEIR POINTERS AND BUFFER ADDRESSES MUST BE CHANGED. THIS ROUTINE PERFORMS THESE OPERATIONS.

5.2.8 ABSRV (KT11-C ABORT AND TRAP HANDLER)

IF CYCLING OF THE ACCESS KEY FOR SUPERVISOR PAGE 7 IS ALLOWED, I/O REFERENCES TO THE EXTERNAL BANK WILL SOMETIMES ABORT OR TRAP. THIS ROUTINE HANDLES THESE ABORTS AND TRAPS.

5.2.9 PFAIL (POWER FAIL)

THE POWER FAIL ROUTINE JUMPS TO THE RESTART ROUTINE ON POWER UP, AFTER PRINTING AN ERROR PRINTOUT.

5.2.10 TYOUT (TTY OUTPUT)

THIS ROUTINE OUTPUTS A COUNT PATTERN IN THE INTERRUPT MODE TO THE TELEPRINTER.

5.2.11 RFSTART (RF11 DISK)

THIS ROUTINE PERFORMS A WRITE AND A WRITE CHECK OF THE DISK. THE DATA THAT IS WRITTEN ON THE DISK IS A PART OF THE TEST PROGRAM CODE THAT IS NEVER MODIFIED. THIS SEGMENT OF CORE IS WRITTEN IN CONTIGUOUS BLOCKS ON THE DISK. AFTER THE TOTAL DISK HAS BEEN WRITTEN, A WRITE CHECK IS USED TO VERIFY THAT THE DATA HAS BEEN WRITTEN CORRECTLY ON THE DISK. NOTE THAT NO "DATI" ARE USED IN EXERCISING THE DISK (DATA IS NOT TRANSFERRED INTO MEMORY). THERE IS A LOCATION IN THE PROGRAM THAT IF MODIFIED WILL ALLOW EXERCISING UP TO EIGHT DISKS.

5.2.12 ENDZ (TC11 END ZONE HANDLER)

THIS ROUTINE IS PART OF THE TC11 SERVICE CODE. IT DRIVES THE DECTAPE INTO THE FORWARD OR REVERSE END ZONE, THEN REVERSES IT. IT ALSO DOES THE NECESSARY SETUP TO BEGIN READING OR WRITING THE TAPE.

5.2.13 REGEN (TC11 WRITE BUFFER REGENERATE ROUTINE)

THE TC11 CODE WRITES THE ENTIRE DECTAPE GOING FORWARD, THEN READS IT IN REVERSE. THE BUFFER IS REGENERATED BEFORE WRITING THE TAPE, AND IS CLEARED OUT ONCE THE ENTIRE TAPE HAS BEEN WRITTEN. THIS ROUTINE REGENERATES THE WRITE BUFFER. IT IS CALLED BY A PRIORITY INTERRUPT REQUEST ON LEVEL 2.

5.2.14 RBN (TC11 READ BLOCK NUMBER SERVICE ROUTINE)

AT THE END OF EACH "BLOCK NUMBER FOUND" INTERRUPT, THIS ROUTINE IS ENTERED (UNLESS END ZONE IS BEING SEARCHED FOR). IT CHECKS FOR THE CORRECT SEQUENCE OF BLOCK NUMBERS, THEN SETS UP THE TC11 TO WRITE A BLOCK IF THE TAPE IS TRAVELLING FORWARD. IF IT IS GOING IN REVERSE, THE ROUTINE CHECKS TO SEE IF DATA IS STILL BEING CHECKED FROM A PREVIOUS READ. IF IT'S NOT, THE ROUTINE SETS UP TO READ A BLOCK. IF DATA IS STILL BEING CHECKED FROM BEFORE, IT SIMPLY DOES ANOTHER READ BLOCK NUMBER.

5.2.15 NXTBLK (TC11 READ BLOCK AND WRITE BLOCK SERVICE ROUTINE)

WHEN A READ BLOCK OR A WRITE BLOCK OPERATION IS COMPLETED, THIS ROUTINE IS ENTERED. IT CHECKS THE ERROR BIT, THEN SETS UP A CALL TO CHECK DATA IF DATA WAS JUST READ IN. THE ROUTINE ALSO SETS UP A READ BLOCK NUMBER OPERATION.

5.2.16 TCCK (TC11 CHECK DATA ROUTINE)

WHEN A READ BLOCK OPERATION HAS BEEN COMPLETED, THIS ROUTINE IS CALLED VIA A PRIORITY INTERRUPT REQUEST AT LEVEL 3. THE ENTIRE BUFFER IS CHECKED, AND THE CONTENTS OF THE BUFFER IS ALTERED AS THE CHECK PROGRESSES. THUS, IF A READ BLOCK OPERATION DOES NOT ACTUALLY READ IN ANY DATA, THE DATA CHECK ROUTINE WILL FIND BAD DATA INSTEAD OF SEEING GOOD DATA FROM AN EARLIER READ.

5.2.17 LCLK (LINE CLOCK)

THIS TEST OF THE LINE CLOCK IS IN THE INTERRUPT MODE. IF OPERATING CORRECTLY THE ROUTINE WILL OUTPUT A MINUTES ELAPSED COUNT TO THE HIGH BYTE OF THE DISPLAY REGISTER. THE MINUTE COUNT IS BASED ON A 60 CYCLE LINE FREQUENCY.

5.2.18 LP1 (LINE PRINTER)

THIS ROUTINE OUTPUTS TO THE LINE PRINTER IN THE FLAG MODE WHILE FILLING THE BUFFER, AND IN THE INTERRUPT MODE WHILE THE BUFFER IS BEING PRINTED.

5.2.19 CORE EXPANSION (DET1)

THIS ROUTINE IS CONTROLLED BY SWITCH 5 AT STARTUP. IF CALLED, THE PROCESSOR BACKGROUND CODE WILL BE EXPANDED THRU AVAILABLE MEMORY (UP TO 28K). THE ROUTINE DETERMINES THE MAXIMUM MEMORY SIZE BY DOING A "DATO" TO A LOCATION IN EACH BANK (IF THE BANK DOES NOT EXIST, A TIMEOUT WILL OCCUR). AN IMAGE OF BANK 0 IS THEN TRANSFERRED TO EACH EXISTING BANK. THE CODE IN EACH BANK EXCEPT THE LAST IS MODIFIED TO CHANGE THE END OF BANK CALL TO A JUMP TO BEGINX (CORE EXPANSION SPECIAL HANDLER) IN THE NEXT BANK.

THE LISTING SHOWS ONLY THE CODE FOR BANK ZERO. WHEN AN ERROR OCCURS THAT IS NOT IN BANK ZERO, IGNORE THE BANK BITS OF THE PRINT OUT AND USE THE LISTING FOR BANK ZERO.

SEE 5.3.2 FOR A FURTHER EXPLANATION OF MODES OF EXECUTION USING CORE EXPANSION.

5.2.20 PIRSRV (PRIORITY INTERRUPT REQUEST SERVICE ROUTINE)

TC11 DATA BUFFER GENERATION AND CHECKING LENDS ITSELF TO HANDLING AT A LEVEL LOWER THAN THAT OF THE TC11 INTERRUPT. THE PRIORITY INTERRUPT REQUEST PROVIDES THE ABILITY TO DO THIS. THUS, SOFTWARE REQUESTS ARE USED IN THE DECTAPE CODE TO CALL THESE ROUTINES. THE DATA CHECK IS CALLED BY A LEVEL 3 REQUEST, AND THE BUFFER GENERATION ROUTINE IS CALLED BY A LEVEL 2 REQUEST. THE PIRSRV ROUTINE HANDLES THESE TWO PIRQ INTERRUPTS.

5.2.21 TRTRP

THE TRACE BIT IS NORMALLY SET ON ALTERNATE PASSES THRU THE BACKGROUND PROCESSOR TESTS (IT IS SET DURING THE EVEN NUMBERED PASSES). IF THE TRACE BIT IS SET, THIS ROUTINE'S ADDRESS IS LOADED IN THE TRACE TRAP VECTOR. IT SIMPLY DOES AN RTT BACK TO THE PROCESSOR TESTS. SWITCH 12 SET TO A ONE WILL INHIBIT THE USE OF THE TRACE BIT WITH THE BACKGROUND PROCESSOR TESTS.

5.3 PROGRAM AND/OR OPERATOR ACTION

5.3.1 PROCESSOR TEST EXECUTION - 4K AS 32K

IF SWITCHES 0, 1, AND 2 ARE ALL DOWN (=0) AT STARTUP, THE PROCESSOR TEST WILL BE EXECUTED TREATING EACH 4K BANK AS 32K OF VIRTUAL ADDRESS SPACE. THE FOLLOWING DETAILS THIS MODE OF OPERATION.

USER PAGE 0 IS FIRST MAPPED RW, BANK 0, AND ALL OTHER USER PAGES ARE MAPPED NON-RESIDENT. THE PROCESSOR TESTS ARE EXECUTED IN USER THRU USER PAGE 0. WHEN DONE, USER PAGE 0 IS CHANGED TO NON-RESIDENT, AND USER PAGE 1 IS MAPPED RW, BANK 0. THE PC IS CHANGED TO ADDRESS THE START OF THE PROCESSOR TESTS THRU PAGE 1, AND ANOTHER PASS THRU THE PROCESSOR TESTS IS EXECUTED. AT THE END OF THIS PASS, USER PAGE 2 IS MAPPED RW, BANK 0, AND USER PAGE 1 IS MADE NON-RESIDENT. THE PC IS AGAIN CHANGED. THIS TIME TO ACCESS USER PAGE 2, AND THE PROCESSOR TESTS ARE EXECUTED THRU USER PAGE 2. THIS CYCLE IS REPEATED FOR THE REMAINING USER PAGES, MAPPING EACH IN TURN TO BANK 0 AND CHANGING THE PC TO EXECUTE THRU THE ONE CURRENTLY MAPPED. WHEN THE PASS USING USER PAGE 7 IS COMPLETED, A SEARCH IS MADE FOR THE NEXT 4K BANK OF MEMORY. WHEN A BANK IS FOUND, THE PROGRAM IS COPIED INTO THAT BANK FROM BANK 0. USER PAGE 0 IS MAPPED TO THE NEW BANK, AND THE PC IS CHANGED TO EXECUTE THRU USER PAGE 0. THE PREVIOUS CYCLE IS REPEATED, BUT THIS TIME EACH USER PAGE IS MAPPED IN TURN TO THE NEW BANK. ONCE EXECUTION THRU USER PAGE 7 IS COMPLETED, A SEARCH IS MADE FOR THE NEXT BANK. THE PREVIOUS BANK IS CLEARED (EXCEPT FOR THE LOADER), AND THE PROGRAM IS COPIED FROM BANK 0 INTO THE CURRENT BANK. THE CYCLE REPEATS UNTIL THE EXTERNAL BANK IS REACHED, AT WHICH POINT USER 0 IS MAPPED BACK TO BANK 0 AND THE PROCESS STARTS AGAIN.

5.3.2 PROCESSOR TEST EXECUTION - CORE EXPANSION

IF SWITCH 0, 1, OR 2 IS UP AND SW5 IS DOWN AT STARTUP, THE PROCESSOR TESTS WILL BE CORE EXPANDED THRU AVAILABLE MEMORY UP TO 28K. THE ROUTINE DETI DOES THIS CORE EXPANSION, COPYING BANK 0 INTO EACH OF THE OTHER BANKS. THE EMT CALL AT THE END OF EACH BANK (E08) WHICH CALLS THE END OF BANK SERVICE ROUTINE IS CHANGED TO A JUMP TO BEGINX IN THE NEXT BANK. THE E08 CALL IN THE LAST BANK IS LEFT ALONE. IF SWITCHES 0 AND 1 WERE BOTH DOWN AT STARTUP, USER PAGES 0 THRU 6 ARE MAPPED SO THAT THE PHYSICAL AND VIRTUAL ADDRESSES CORRESPOND, AND THE PROCESSOR TESTS ARE THEN RUN IN USER. IF SW0 WAS DOWN BUT SW1 WAS SET, KERNEL PAGES 0-6 ARE MAPPED SO THAT THE PHYSICAL AND VIRTUAL ADDRESSES ARE THE SAME, AND THE PROCESSOR TESTS ARE THEN RUN IN KERNEL MODE. IF SW0 WAS SET, ORDINARY CORE EXPANSION IS RUN WITH NO SPECIAL MAPPING REQUIRED (KT11-C IS TURNED OFF).

5.3.3 PROCESSOR TEST EXECUTION - BANK 0 ONLY

IF SW0, 1 OR 2 IS UP AND SW5 IS UP AT STARTUP, ONLY BANK 0 IS UTILIZED. IN THIS CASE, IF SW0 AND SW1 WERE DOWN THE PROCESSOR TESTS ARE EXECUTED IN USER, WITH USER PAGE 0 MAPPED TO BANK 0. IF SW0 WAS DOWN AND SW1 WAS UP, THE PROCESSOR TESTS ARE EXECUTED IN KERNEL, WITH KERNEL PAGE 0 MAPPED TO BANK 0. IF SW0 WAS UP, THE KT11-C IS TURNED OFF AND THE PROCESSOR TESTS ARE EXECUTED IN KERNEL MODE OR USER MODE (DEPENDING ON SW1) IN BANK 0 ONLY.

5.3.4 RF11 CODE AND BUFFER RELOCATION

IF SW3 IS DOWN AT STARTUP AND THE RF11 IS NOT INHIBITED, THE RF11 CODE WILL BE EXECUTED FROM THE SAME BANK THAT THE PROCESSOR TESTS ARE RUNNING IN. THE PHYSICAL MEMORY ADDRESS USED BY THE RF11 FOR WRITE AND WRITE CHECK OPERATIONS WILL ALSO BE IN THE SAME BANK AS THE PROCESSOR TESTS ARE RUNNING IN, THUS THE EXTENDED MEMORY ADDRESS BITS WILL GET SET IF THE PROCESSOR TESTS ARE RUNNING ABOVE 32K. THIS RELOCATION OCCURS IF CORE EXPANSION IS SELECTED (EVEN IF THE KT11-C IS INHIBITED) OR IF THE PROCESSOR TESTS ARE BEING RUN IN THE 4K AS 32K MODE.

NOTE THAT IF THE 4K AS 32K MODE IS USED, THE RF11 WILL BE MAPPED TO KERNEL PAGE 1, AND KERNEL PAGE 1 IS MAPPED TO THE CURRENT BANK. OTHERWISE, RELOCATION IS DONE BY MAPPING THE RF11 TO THE CURRENT BANK DIRECTLY (IN WHICH CASE VIRTUAL AND PHYSICAL ADDRESSES ARE MAPPED EQUAL IF KT11-C IS ON).

5.3.5 TC11 CODE AND BUFFER RELOCATION

IF SW3 IS DOWN AT STARTUP AND IF THE TC11 IS NOT INHIBITED, THE TC11 CODE WILL BE EXECUTED FROM THE SAME BANK THAT THE PROCESSOR TESTS ARE RUNNING IN. THE READ AND WRITE BUFFERS FOR THE TC11 WILL ALSO BE LOCATED IN THE BANK IN WHICH THE PROCESSOR TESTS ARE RUNNING. DUE TO THE COMPLEXITY OF RELOCATING THE TC11 CODE AND BUFFERS, RELOCATION IS DONE ONLY WHEN THE DECTAPE IS AT THE FORWARD END ZONE. THUS, BEFORE THE BANK IN WHICH THE PROCESSOR TESTS ARE RUNNING IS CHANGED, THE CODE WAITS FOR THE DECTAPE TO FINISH A PASS OF WRITING AND READING THE TAPE. A PASS THRU ALL OF MEMORY RUNNING TC11 RELOCATION WILL REQUIRE THE DECTAPE TO MAKE AS MANY PASSES AS THERE ARE BANKS OF MEMORY.

NOTE THAT IF THE 4K AS 32K MODE IS USED, THE TC11 WILL BE MAPPED TO KERNEL PAGE 1, AND KERNEL PAGE 1 IS MAPPED TO THE CURRENT BANK. OTHERWISE, RELOCATION IS DONE BY MAPPING THE TC11 TO THE CURRENT BANK DIRECTLY (IF KT11-C IS ON, VIRTUAL ADDRESSES ARE MAPPED TO EQUAL THE PHYSICAL ADDRESSES).

5.3.6 EXECUTION OF I/O CODE IN SUPERVISOR

IF SW1 IS DOWN AT STARTUP, THE FOLLOWING DEVICES ARE SERVICED IN SUPERVISOR MODE: TELETYPE OUTPUT, LINE CLOCK, AND LINE PRINTER. IF MEMORY MANAGEMENT IS NOT INHIBITED, THESE DEVICES' SERVICE ROUTINES EXECUTE THRU SUPERVISOR PAGE 0, THEN PAGE 1, AND SO ON, SEQUENTIALLY UP TO SUPERVISOR PAGE 6. THE PAGE BEING ACCESSED IS CHANGED EACH TIME THE END OF BANK CODE (EOBSRV ROUTINE) IS EXECUTED, BY CHANGING THE ADDRESSES STORED IN THE DEVICES' INTERRUPT VECTORS. WHEN SUPERVISOR PAGE 6 WAS THE ONE JUST USED, THE CYCLE STARTS OVER WITH PAGE 0. PAGE 7 IS USED FOR ACCESSING THE EXTERNAL BANK. A DYNAMIC SWITCH (SW09) IS PROVIDED TO INHIBIT CYCLING THE SUPERVISOR MAPPING. IF THIS SWITCH IS SET, THE SUPERVISOR CODE WILL REMAIN MAPPED TO THE PAGE IT WAS MAPPED TO WHEN THE SWITCH WAS SET.

5.3.7 EXECUTION OF I/O CODE IN SUPERVISOR - USE OF SUPERVISOR PAGE 7

IF SW1 IS DOWN AT STARTUP, THE DEVICES LISTED IN 5.3.6 WILL BE SERVICED IN SUPERVISOR MODE, AS EXPLAINED PREVIOUSLY. IF SW4 IS ALSO DOWN AT STARTUP, SUPERVISOR PAGE 7 (MAPPED TO THE EXTERNAL BANK) WILL HAVE ITS ACCESS KEY CHANGED EACH TIME THE END OF BANK CODE (EOBSRV) IS EXECUTED. THE ACCESS KEY WILL BE SET FIRST TO 0, THEN 1, AND SO ON UP THRU 7. AFTER BEING SET TO 7, IT WILL BE RECYCLED TO 0 AND THE CYCLE WILL BE REPEATED. DEPENDING ON THE KEY IN EFFECT AT THE TIME, THE SUPERVISOR I/O MAY TRAP, ABORT, OR EXECUTE WHEN REFERENCING THE EXTERNAL PAGE. IF A TRAP OR ABORT OCCURS THE KERNEL ROUTINE ABSRV IS ENTERED, THIS ROUTINE PERFORMS CERTAIN CHECKS ON THE TRAP OR ABORT, AND IF IT WAS AN ABORT, IT REEXECUTES THE INSTRUCTION IN KERNEL MODE (MAPPED R/W). TO REGAIN CONTROL AFTER THE INSTRUCTION IS COMPLETED, THE ROUTINE SETS THE T-BIT AND CHANGES THE TRACE TRAP RETURN TO POINT TO A SECTION OF THE ABORT SERVICE CODE. AFTER THE INSTRUCTION IS COMPLETED, A TRACE TRAP TO THE KERNEL ABORT SERVICE CODE OCCURS. THE CODE RESTORES THE PREVIOUS TRACE-TRAP VECTOR, RESTORES THE T-BIT TO ITS PREVIOUS STATE, AND RETURNS TO THE SUPERVISOR DEVICE SERVICE CODE AT THE INSTRUCTION AFTER THE ONE WHICH ABORTED.

NOTE THAT SWITCH 8 UP WILL INHIBIT THE CYCLING OF THE ACCESS KEY DYNAMICALLY, LEAVING IT SET TO THE CURRENT VALUE.

6.0 ERRORS

6.1 ERROR PRINTOUT

PRINTOUTS ARE IN AN EXTENDED VERSION OF THE STANDARD FORMAT, USING FOUR WORDS. THE FIRST WORD IS THE OCTAL VALUE OF THE VIRTUAL PC+2 OF THE DETECTED ERROR. THE SECOND WORD IS THE CONTENTS OF THE PROCESSOR STATUS REGISTER WHEN THE ERROR WAS DETECTED. THE THIRD IS THE TOP 12 BITS OF THE 18 BIT ADDRESS OF THE BANK BEING CURRENTLY USED FOR EXECUTION OF THE PROCESSOR TESTS (AND RF11 AND TC11 CODE IF IT IS BEING RELOCATED). TO GET THE STARTING ADDRESS OF THE CURRENT BANK SIMPLY APPEND TWO ZEROS TO THE END OF THE OCTAL VALUE PRINTED OUT (I.E. 007400 INDICATES THE BANK BEGINNING AT PHYSICAL ADDRESS 740000). THE FOURTH WORD IS THE PC (VIRTUAL) OF THE CURRENT PROCESSOR SUBTEST BEING EXECUTED IN BACKGROUND.

6.2 ERROR RECOVERY

IN GENERAL, TEST FAILURES WILL PRINTOUT AN ERROR MESSAGE AND CONTINUE. IF THE "HALT ON ERROR" SWITCH IS SET, HITTING CONTINUE WILL RECOVER. IF THE PROGRAM HANGS UP IN A LOOP, THE ERROR IS LIKELY TO BE A SIGNAL WHICH WAS NEVER RECEIVED. IF A HALT OCCURS IN THE TRAP AND VECTOR AREA THE PROGRAM MUST BE RESTARTED. IF THE PROGRAM HALTS IN THE MAIN FLOW, CONSULT THE LISTING IF NO MESSAGE IS TYPED OUT.

6.3 FINDING WHICH PROCESSOR TEST WAS BEING EXECUTED WHEN AN ERROR OCCURRED

SOME ERRORS ARE DEPENDENT ON THE PROCESSOR TEST BEING RUN (SUCH AS LATENCY ERRORS WHICH ONLY SHOW UP IN WORST-CASE PROCESSOR TIMING). THE SCOPE ROUTINE CONTAINS A LOCATION CALLED "RETURN" WHICH STORES THE STARTING ADDRESS OF THE PROCESSOR TEST CURRENTLY BEING EXECUTED. NOTE THAT THE SCOPE ROUTINE IS EXECUTED IN USER MODE IF SW1 IS DOWN AT STARTUP, AND IS THEREFORE RELOCATED WITH THE PROCESSOR TESTS. THE CONTENTS OF RETURN IN THE CURRENT BANK IS THE FOURTH WORD OF THE ERROR PRINTOUT.

7.0 RESTRICTIONS

PROGRAM MUST BE LOADED INTO THE LOWER 4K OF MEMORY.

IF THE LINE PRINTER IS USED, STARTING ADDRESS 300 MUST BE USED INSTEAD OF SA 200.

8.0 MISCELLANEOUS

8.1 EXECUTION TIME

EXECUTION TIME VARIES WITH THE AMOUNT OF MEMORY, THE TYPES OF MEMORY, THE DEVICES RUN, AND THE OPTIONAL MODES OF EXECUTION USED.

A PASS RUN WITH CORE EXPANSION AND 4K AS 32K RELOCATION BOTH INHIBITED TAKES LESS THAN 10 SECONDS (RUNNING NO I/O).

A PASS RUN WITH 4K AS 32K, THRU 128K OF CORE MEMORY, WITH NO I/O TAKES ABOUT 15 MINUTES. (AN ASTERISK IS PRINTED AT THE END OF A FULL PASS, AND A DOLLAR SIGN IS PRINTED AT THE END OF EACH 4K BANK).

A PASS RUN WITH 4K AS 32K, WITH DECTAPE RELOCATION, TAKES ABOUT 1 MINUTE PER 4K BANK, IF ITERATIONS ARE INHIBITED (SW11 IS UP).

8.2 STACK POINTERS

THE KERNEL STACK POINTER IS INITIALIZED TO 17760.

THE SUPERVISOR STACK POINTER IS INITIALIZED TO 600.
IT IS RELOCATED TO THE CURRENT SUPERVISOR PAGE.

THE USER STACK POINTER IS INITIALIZED TO 400. IT IS RELOCATED THRU ALL USER PAGES AND TO EVERY 4K BANK IF THE 4K AS 32K MODE OF EXECUTION IS RUN.

8.3 MONITORING PHYSICAL AND VIRTUAL ADDRESSES

DURING EXECUTION OF 4K AS 32K, IT IS HELPFUL TO SET THE ADDRESS SELECTOR TO PHYSICAL AND THE DISPLAY SELECTOR TO DATA REGISTER. IF THIS IS DONE, THE ADDRESS LIGHTS WILL INDICATE THE CURRENT PHYSICAL ADDRESSES WHILE THE DATA LIGHTS WILL SHOW THE CURRENT VIRTUAL ADDRESSES (SINCE THEY ARE USED AS DATA A GREAT DEAL OF THE TIME).

8.4 DISPLAY REGISTER

THE LOWER BYTE OF THE DISPLAY REGISTER SHOWS THE PASS COUNT. THE UPPER BYTE IS USED WHEN THE KW11-L LINE CLOCK IS RUN TO INDICATE MINUTES ELAPSED SINCE THE PROGRAM WAS STARTED (BASED ON A LINE FREQUENCY OF 60 CYCLES PER SECOND).

9.0 PROGRAM DESCRIPTION

THIS MEMORY MANAGEMENT EXERCISER IS DESIGNED TO RUN BACKGROUND PROCESSOR TESTS AND FOREGROUND CONCURRENT I/O WITH MEMORY MANAGEMENT UTILIZED IN ANY OF SEVERAL DIFFERENT MODES. THE VARIOUS MODES AVAILABLE FOR UTILIZING MEMORY MANAGEMENT ARE INCLUDED TO AID IN FAULT ISOLATION BY PROVIDING A SERIES OF STEPS FROM SIMPLE TO COMPLEX. THESE LEVELS ARE AS FOLLOWS:

	KT11-C ON	CORE EXPAN (SEE 5.3.2)	4K AS 32 (TO 128K) (SEE 5.3.1)	USER/SUPER KERNEL	RF/TC RELOC (SEE 5.3.4 AND 5.3.5)	VARIABLE USE OF SUPER (SEE 5.3.6 AND 5.3.7)
1				X		
2		X				
3		X		X		
4		X			X	
5				X	X	
6	X					
7	X			X		X
10	X	X				
11	X	X		X		X
12	X	X			X	
13	X			X	X	X
14	X		X	X		X
	X		X	X	X	X

NOTE THAT WHERE VARIABLE USE OF THE SUPERVISOR SPACE IS ALLOWED, THERE ARE FOUR POSSIBLE LEVELS FOR ITS USE:

1. ALL EXECUTION OCCURS THRU SUPERVISOR PAGE 0, WITH PAGE 7 MAPPED RW FOR I/O REFERENCES.
2. EXECUTION OCCURS THRU SUPERVISOR PAGES 1 THRU 6, ONE AT A TIME, WITH PAGE 7 MAPPED RW FOR I/O REFERENCES.
3. ALL EXECUTION OCCURS THRU SUPERVISOR PAGE 0 WITH PAGE 7 CYCLED THRU ALL VALUES OF THE ACCESS KEY
4. EXECUTION OCCURS THRU SUPERVISOR PAGES 1 THRU 6, ONE AT A TIME, WITH THE ACCESS KEY FOR PAGE 7 CYCLED THRU ALL VALUES.

THERE IS NO MONITOR IN THE CONVENTIONAL SENSE. EACH DEVICE THAT IS TO BE EXERCISED HAS ITS OWN STAND ALONE ROUTINE THAT OPERATES IN THE INTERRUPT MODE. THESE ROUTINES NEED NO SUPERVISION OR MONITORING AFTER THEY ARE INITIATED. THERE IS A PRIMER AREA THAT CHECKS THE SWITCH REGISTER TO SEE WHAT DEVICES ARE TO BE INITIATED. IT SETS THE INTERRUPT ENABLE BIT IN THE DEVICE STATUS REGISTER, INITIALIZES THE DATA PATTERN, AND INITIATES AN OPERATION TO RAISE DATA FLAGS ON DEVICES THAT CAN NOT INITIATE THEM THEMSELVES. THE PRIMER CODE THEN ENTERS THE KT11-C SETUP CODE. THE RF11 AND TC11 PRIMER CODE IS IN WITH THE KT11-C SETUP CODE SINCE THESE DEVICES, AS CODED REQUIRE CERTAIN PARTS OF THE KT11-C CODE TO BE RUN FIRST. AFTER MEMORY MANAGEMENT IS TURNED ON, EXECUTION OF THE BACKGROUND PROCESSOR TESTS BEGINS, AND THE I/O DEVICES ARE SERVICED WHEN THEY INTERRUPT. ALL CYCLING FEATURES OCCUR AT THE END OF THE PROCESSOR TESTS IN EACH BANK.

*

```
                        ;COPYRIGHT 1972,1973, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
                        ;PDP-11/45 SYSTEM EXERCISER WITH KT11-C --- TTY, KW11-L
                        ;LP11,RF11,AND TC11

                        ;RF11 AND TC11 CODE RUN IN KERNEL MODE IN SAME
                        ;PHYSICAL BANK AS BACKGROUND TESTS (IF ALL SWITCHES DOWN)
                        ;ALL OTHER I/O IS RUN IN SUPERVISOR MODE, WITH EXTERNAL
                        ;BANK ACCESS KEY CYCLED THRU ALL VALUES (SUPERVISOR ONLY)
                        ;CPU TESTS RUN IN USER MODE UNLESS INHIBITED BY SR SETTINGS

                        ;SA - 200 (300 IF LP11 IS USED)
                        ;RESTART - 310 (SR SETTINGS PREVIOUSLY MADE ARE USED)

                        ;AT STARTUP, SR SETTINGS ARE:
                        ;SR 0=1 OR UP --- RUN WITHOUT KT11-C
                        ;SR 1=1 OR UP --- RUN ALL IN KERNEL MODE (INHIBITS RUNNING 4K AS 32K
                                          ;AND INHIBITS RUNNING I/O IN SUPERVISOR)
                        ;SR 2=1 OR UP --- INHIBIT RUNNING 32K USER RELOCATION FROM EVERY 4K
                                          ;BANK (ALLOW NORMAL CORE EXPANSION)
                        ;SR 3=1 OR UP --- INHIBIT RF AND TC RELOCATION OF CODE AND BUFFERS
                        ;SR 4=1 OR UP --- INHIBIT CYCLING ACCESS KEYS IN S7
                        ;SR 5=1 OR UP --- INHIBIT VARIABLE CORE EXPANSION

                        ;AT HALT, SR SETTINGS ARE:
                        ;SR 15=1 OR UP---HALT ON ERROR
                        ;SR 14=1 OR UP---SCOPE LOOP
                        ;SR 13=1 OR UP---INHIBIT PRINT OUT
                        ;SR 12=1 OR UP---INHIBIT TRACE TRAPPING
                        ;SR 11=1 OR UP---INHIBIT SUB-PROGRAM ITERATION AND INHIBIT TESTS WHICH
                                         ;USE ALL COMBINATIONS OF NUMBERS
                        ;SR 10=1 OR UP---INHIBIT PROCESSOR TEST (ONCE SET, CLEARING IT WILL NOT
                                         ;CAUSE PROCESSOR TEST TO CONTINUE)
                        ;SR 09=1 OR UP---STOP CYCLING SUPERVISOR PAGES
                        ;SR 08=1 OR UP---STOP CYCLING SUPERVISOR PAGE 7 ACCESS KEY
                        ;THESE SWITCHES ARE ALSO CHECKED AT THE END OF EACH PASS

                        ;SPECIAL DELETE SWITCHES-SET RESPECTIVE SWITCH TO A 1 TO INHIBIT
                        ;INITIATION OF DEVICE - IF SWITCH IS DOWN, DEVICE WILL BE STARTED
                        ;IF PRESENT
                        ;SW 0=1 INHIBIT TTY OUTPUT
                        ;SW 4=1 INHIBIT LINE CLOCK
                        ;SW 5=1 INHIBIT LINE PRINTER
                        ;SW 6=1 INHIBIT RF11 DISK
                        ;SW 7=1 INHIBIT TC11 DECTAPE

                                .SBTTL DATA

                        ;DEFINITIONS
000240                          NOP=240                         ;SYSTEM NULL OPERATION
104400                          SCOPE=TRAP                      ;TRAP USED SCOPE LOOP AND ITERATION
177776                          CC=177776
177776                          PSR=CC
000616                          TCSR=TTCSR
000620                          TDBR=TTDBR
177570                          SR=177570
104006                          HLT=104006                      ;ERROR PRINTOUT CALL
```

```
                104010                  RTIK=104010                     ;RTI VIA KERNEL CALL
                104012                  EOB=104012                      ;END OF BANK CALL
                000000                  R0=%0
                000001                  R1=%1
                000002                  R2=%2
                000003                  R3=%3
                000004                  R4=%4
                000005                  R5=%5
                000006                  SP=%6
                000006                  R6=SP
                000007                  PC=%7
                177776                  STATUS=PSR
                001024                  SEGREG=SREG1
                001026                  REG1=SREG2

                                ;LOAD TRAP CATCHER IN LOCATIONS 0 THRU 577

                                ;LOAD VECTOR AREA
                000030                  .=30
000030  013536                          EMTSRV                          ;EMT CALLS
000032  000340                          340                             ;HIGHEST PRIORITY
                000034                  .=34
000034  013262                          $COPEC                          ;USER TRAP
000036  000000                          0
                000046                  .=46
000046  015014                          LOGICAL
                000052                  .=52
000052  040000                          40000
                000240                  .=240
000240  013470                          PIRSRV
000242  000340                          340

                                ;LOAD STARTING AREA
                000200                  .=200
000200  000137  001264                  JMP     @#START
                000300                  .=300
000300  000137  001264                  JMP     @#START
                000310                  .=310
000310  000137  001250                  JMP     @#RSTRT

                                ;DATA AREA
                000400                  .=400
000400  000000                  UBUFF:  0                               ;BUFFER FOR USER SP
                000600                  .=600
000600  000000                  SBUFF:  0                               ;BUFFER FOR SUPERVISOR STACK
                000606                  .=.+4                           ;FOR STACK OVERRUN
000606  177560                  TRCSR:  177560                          ;TTY READER STATUS REGISTER
000610  177562                  TRDR:   177562
000612  000060                  TTRVC:  60
000614  000062                  TTRST:  62
000616  177564                  TTCSR:  177564                          ;TTY PUNCH STATUS REGISTER
000620  177566                  TTDBR:  177566
000622  000064                  TTPVC:  64
000624  000066                  TTPST:  66
000626  000000                  TTSAV:  0                       ;STORE CONTENTS OF TTY PRINTER CSR HERE
000630  177546                  LKCSR:  177546
```

```
000632  000100                  KWLVC:  100
000634  000102                  KWLST:  102
000636  177514                  LPCSR:  177514
000640  177516                  LPDBR:  177516
000642  000200                  LPVC:   200
000644  000202                  LPST:   202
000646  177470                  RFDAE:  177470          ;DISK ADDRESS AND ERROR
000650  177472                  RFDBR:  177472          ;DATA BUFFER REGISTER
000652  177466                  RFDAR:  177466          ;DISK ADDRESS REGISTER
000654  177462                  RFWC:   177462          ;WORD COUNT REGISTER
000656  177464                  RFCAR:  177464          ;CURRENT ADDRESS REGISTER
000660  177460                  RFCSR:  177460          ;STATUS REGISTER
000662  177461                  RFCSRH: 177461          ;HIGH BYTE ADDRESS OR CSR
000664  000204                  RFVC:   204
000666  000206                  RFST:   206
000670  177572                  SR0:    177572          ;KT11-C STATUS REGISTERS
000672  177574                  SR1:    177574
000674  177576                  SR2:    177576
000676  172516                  SR3:    172516
000700  177600                  UIPDR0: 177600
000702  177602                  UIPDR1: 177602
000704  177616                  UIPDR7: 177616
000706  177620                  UDPDR0: 177620
000710  177622                  UDPDR1: 177622
000712  177636                  UDPDR7: 177636
000714  177640                  UIPAR0: 177640
000716  177642                  UIPAR1: 177642
000720  177656                  UIPAR7: 177656
000722  177660                  UDPAR0: 177660
000724  177662                  UDPAR1: 177662
000726  177676                  UDPAR7: 177676
000730  172200                  SIPDR0: 172200
000732  172220                  SDPDR0: 172220
000734  172214                  SIPDR6: 172214
000736  172256                  SIPAR7: 172256
000740  172276                  SDPAR7: 172276
000742  172216                  SIPDR7: 172216
000744  172236                  SDPDR7: 172236
000746  172300                  KIPDR0: 172300
000750  172302                  KIPDR1: 172302
000752  172304                  KIPDR2: 172304
000754  172306                  KIPDR3: 172306
000756  172316                  KIPDR7: 172316
000760  172320                  KDPDR0: 172320
000762  172322                  KDPDR1: 172322
000764  172324                  KDPDR2: 172324
000766  172326                  KDPDR3: 172326
000770  172336                  KDPDR7: 172336
000772  172340                  KIPAR0: 172340
000774  172342                  KIPAR1: 172342
000776  172344                  KIPAR2: 172344
001000  172346                  KIPAR3: 172346
001002  172356                  KIPAR7: 172356
001004  172360                  KDPAR0: 172360
001006  172362                  KDPAR1: 172362
001010  172364                  KDPAR2: 172364
```

```
001012  172366                  KDPAR3: 172366
001014  172376                  KDPAR7: 172376

001016  177600                  IPORTAB:177600
001020  172200                          172200
001022  172300                  IPOREND:172300
001024  000000                  SREG1:  0               ;STORES KT11-C SWITCH REGISTER SETTINGS
001026  000000                  SREG2:  0               ;STORES SR SETTINGS
001030  000000                  SEGCNT: 0               ;USED TO LOAD KT11-C REGISTERS
001032  177342                  TCCM:   177342          ;CONTROL AND FUNCTION
001034  177340                  TCST:   177340          ;GENERAL STATUS
001036  177350                  TCDT:   177350          ;DATA
001040  177344                  TCWC:   177344          ;WORD COUNT
001042  177346                  TCBA:   177346          ;BUS ADDRESS
001044  000214                  TCIV:   214             ;DECTAPE INTERRUPT VECTOR
001046  000216                  TCSTA:  216

001050  000000                  CURBNK: 0               ;SAF TO POINT TO CURRENT BANK
001052  000000                  OLDBNK: 0               ;SAVES ADDRESS OF PREVIOUS BANK
001054  000000                  CURIPAR: 0              ;ADDRESS OF CURRENT IPAR
001056  000000                  CUROPAR: 0
001060  000000                  CURIPDR:0
001062  000000                  CUROPDR: 0
001064  000000                  CSIPDR: 0
001066  000000                  BNKSTR: 0               ;PC TO POINT TO BEGIN THRU CURRENT PAGE
001070  000000                  TRPB:   0
001072  000000                  PSSAV:  0
001074  177770                  UBRK:   177770          ;MICROBREAK REGISTER ADDRESS
001076  177772                  PIR:    177772
001100  000240                  PIRQIV: 240
001102  000242                  PIRQST: 242
001104  000000                  DISPLY: 0               ;VALUE LOADED INTO DISPLAY REGISTER
                                                        ;LOW BYTE SHOWS PASS COUNT
                                                        ;HIGH BYTE SHOWS MINUTES ELAPSED IF RUNNING KW11

                                :PARITY TRAP SERVICE ROUTINE
001106  005077  177666          PARSRV: CLR     @KIPAR3
001112  005077  177674                  CLR     @KDPAR3
001116  012777  077406  177630          MOV     #77406,@KIPDR3
001124  012777  077406  177634          MOV     #77406,@KDPDR3
001132  012737  001234  000114          MOV     #PARFD,@#114    ;CHANGE PARITY TRAP RETURN
001140  012737  001230  000004          MOV     #PARTO,@#4      ;SETUP TIMEOUT RETURN
001146  005037  000006                  CLR     @#6

001152  012700  060000          PAR1:   MOV     #60000,R0       ;INITIALIZE R0 TO TEST BANK THRU KERNEL 3
001156  012702  010000                  MOV     #10000,R2       ;R2 IS USED AS A COUNTER
001162  011001                  PAR2:   MOV     @R0,R1          ;READ LOCATION AND SAVE CONTENTS
001164  005720                          TST     (R0)+           ;MOVE POINTER
001166  077203                          SOB     R2,PAR2         ;LOOP UNTIL 4K HAS BEEN CHECKED
001170  062777  000200  177602  PAR3:   ADD     #200,@KIPAR3
001176  062777  000200  177606          ADD     #200,@KDPAR3
001204  022777  007600  177566          CMP     #7600,@KIPAR3
001212  003357                          BGT     PAR1
001214  011600                          MOV     @SP,R0          ;LOAD PC AT TIME OF ABORT INTO R0
001216  104006                          HLT                     ;PARITY TRAP OCCURRED BUT MEMORY
                                                                ;SCAN DID NOT DETECT A PARITY ERROR-
                                                                ;R0 CONTAINS PC AT TIME OF ABORT
```

```
001220  000000                          HALT                    ;HALT AFTER PARITY ERROR
001222  000005                          RESET                   ;CLEAR PARITY REGISTERS AND DEVICES
001224  000167  000020                  JMP     RSTRT           ;RESTART TEST
001230  022626                  PART0:  CMP     (SP)+,(SP)+     ;TIMEOUT OCCURRED- BANK NOT PRESENT
001232  000756                          BR      PAR3            ;GO CHECK FOR ANOTHER BANK
001234  022626                  PARFD:  CMP     (SP)+,(SP)+     ;BAD PARITY FOUND
001236  104006                          HLT                     ;PARITY ERROR OCCURRED AND MEMORY
                                                                ;SCAN FOUND A MEMORY LOCATION
                                                                ;CONTAINING BAD PARITY. R0 CONTAINS
                                                                ;THE ADDRESS OF THE LOCATION
                                                                ;HAVING BAD PARITY
                                                                ;(VIRTUAL MAPPED THRU KERNEL PAGE 3)
001240  000000                          HALT                    ;HALT AFTER PARITY ERROR
001242  000005                          RESET                   ;IF CONTINUED, CLEAR PARITY AND DEVICES
001244  000167  000000                  JMP     RSTRT           ;RESTART PROGRAM

                                ;RESTART ADDRESS USING INITIAL SR SETTINGS
001250  012706  017760          RSTRT:  MOV     #KSTACK,R6
001254  113737  177571  001027          MOVB    @#SR+1,@#SREG2+1
001262  000437                          BR      START1

                                        .SBTTL I/O CODE
                                ;START UP FOR MINI MONITOR

001264  012706  017760          START:  MOV     #KSTACK,R6      ;SET UP KERNEL STACK
001270  005737  000042                  TST     @#42            ;RUNNING UNDER A MONITOR?
001274  001417                          BEQ     STARTX          ;NO- BRANCH
001276  012737  014774  000004          MOV     #LOGIC,@#4      ;YES- CHECK TO SEE IF THIS IS AN 11/45
001304  005037  000006                  CLR     @#6
001310  005777  177562                  TST     @PIR            ;IF NO TRAP THIS IS AN 11/45
001314  012737  000006  000004          MOV     #6,@#4
001322  005037  001024                  CLR     @#SREG1         ;RUN WITH ALL SWITCHES DOWN IF UNDER A MONITOR
001326  005037  001026                  CLR     @#SREG2
001332  000410                          BR      START0
001334  013737  177570  001024  STARTX: MOV     @#SR,@#SREG1    ;STORE KT11-C SWITCHES
001342  000000                          HALT
001344  013737  177570  001026          MOV     @#SR,@#SREG2    ;STORE DEVICE SELECTION SWITCHES AND
                                                                ;DYNAMIC SWITCHES
001352  000403                          BR      START1
001354  012737  000361  001026  START0: MOV     #361,@#SREG2
001362  012737  000001  013354  START1: MOV     #1,@#ICOUNT     ;INITIALIZE ITERATION COUNT
001370  005037  001104                  CLR     @#DISPLY        ;INITIALIZE PASS COUNT
001374  004767  015400                  JSR     %7,CRLF         ;OUTPUT CARRIAGE RETURN, LINE FEED
001400  012737  013262  000034          MOV     #SCOPEC,@#34    ;INITIALIZE SCOPE TO KERNEL PAGE 0
001406  005037  000036                  CLR     @#36
001412  012737  006700  013360          MOV     #BEGIN,@#RETURN
001420  005037  013356                  CLR     @#SCOPEF
001424  012737  013536  000030          MOV     #EMTSRV,@#30    ;INITIALIZE EMT CALL TO KERNEL PAGE 0
001432  012737  000340  000032          MOV     #340,@#32
001440  012737  006172  004276          MOV     #NXTBLK,@#LLIMIT
                                ;IN-LINE ROUTINE TO SET ACTION ENABLE IN ALL MA11 OR MF11 PARITY
                                ;REGISTERS FOUND
001446  012737  001106  000114  PARSET: MOV     #PARSRV,@#114   ;SET UP PARITY TRAP VECTOR
001454  012737  000340  000116          MOV     #340,@#116      ;CAUSE LOCK UP AFTER PARITY TRAP
001462  012701  172000                  MOV     #172000,R1      ;INITIALIZE R1 TO THE FIRST POSSIBLE
                                                                ;PARITY REGISTER ADDRESS
```

```
001466  012737  000006  000004          MOV     #6,@#4
001474  012737  000002  000006          MOV     #RTI,@#6
001502  012721  000001          1$:     MOV     #1,(R1)+        ;IF NO TIMEOUT, PARITY PRESENT-
                                                                ;SET PARITY ACTION ENABLE TO CAUSE
                                                                ;TRAP TO 114 ON PARITY ERROR,
                                                                ;TO INHIBIT SETUP OF PARITY MEMORY
                                                                ;CHANGE THE #1 TO A ZERO
001506  020127  172140                  CMP     R1,#172140      ;DONE WITH PARITY ADDRESSES?
001512  103773                          BLO     1$              ;NO- BRANCH
001514  005037  000006          PARDN:  CLR     @#6             ;YES- RESTORE TRAPCATCHER
001520  012737  000340  177776          MOV     #340,@#PSR      ;LOCK OUT INTERRUPTS
001526  005037  016546                  CLR     @#PRTON         ;INITIALIZE PRINT ROUTINE BUSY FLAG
001532  012737  000000  001050          MOV     #0,@#CURBNK     ;INITIALIZE BANK POINTER
001540  032737  000002  001024          BIT     #2,@#SREG1      ;INHIBIT USER/SUPERVISOR/KERNEL?
001546  001017                          BNE     STO             ;YES - SKIP OVER
001550  052737  140000  000036          BIS     #140000,@#36    ;SET USER BIT IN SCOPE STATUS
001556  012746  000600                  MOV     #SBUFF,-(R6)    ;SETUP SUPERVISOR STACK
001562  052737  010000  177776          BIS     #10000,@#PSR
001570  106606                          MTPD    SP
001572  012746  000400                  MOV     #UBUFF,-(R6)    ;SETUP USER STACK
001576  052737  030000  177776          BIS     #30000,@#PSR
001604  106606                          MTPD    SP
001606  012737  003220  000004  STO:    MOV     #NODEV,@#4
001614  005037  000006                  CLR     @#6
001620  012700  000001                  MOV     #1,R0
001624  005037  177776                  CLR     @#PSR
001630  033700  001026                  BIT     @#SREG2,R0      ;INHIBIT TTY OUTPUT?
001634  001024                          BNE     ST1             ;YES, GO CHECK NEXT
001636  005777  176754                  TST     @TTCSR          ;CHECK FOR TTY PRESENT
001642  012777  000200  176754          MOV     #200,@TTPST
001650  005037  003656                  CLR     @#DATA2         ;INITIALIZE BASE DATA
001654  012777  003670  176740          MOV     #TYOUTR,@TTPVC  ;NO, SETUP INTERRUPT VECTOR
001662  032737  000002  001024          BIT     #2,@#SREG1      ;RUN KERNEL ONLY?
001670  001003                          BNE     .+10            ;YES - SKIP NEXT
001672  052777  040000  176724          BIS     #40000,@TTPST   ;NO - RUN DEVICE IN SUPERVISOR
001700  052777  000100  176710          BIS     #100,@TTCSR     ;START TTY OUTPUT
001706  006300                  ST1:    ASL     R0
001710  006300                  ST2:    ASL     R0
001712  006300                  ST3:    ASL     R0
001714  006300                  ST4:    ASL     R0
001716  033700  001026                  BIT     @#SREG2,R0      ;INHIBIT LINE CLOCK?
001722  001026                          BNE     ST5             ;YES, GO CK NEXT
001724  005777  176700                  TST     @LKCSR
001730  105037  001105                  CLRB    @#DISPLY+1      ;INITIALIZE MINUTE COUNTER
001734  012777  003726  176670          MOV     #LK3,@KWLVC
001742  012777  000240  176664          MOV     #240,@KWLST     ;SERVICE KW11-L AT LEVEL
                                                                ;5 TO ALLOW TC11 IN
001750  032737  000002  001024          BIT     #2,@#SREG1      ;RUN KERNEL ONLY?
001756  001003                          BNE     .+10            ;YES- BRANCH
001760  052777  040000  176646          BIS     #40000,@KWLST   ;NO- RUN DEVICE IN SUPERVISOR
001766  005067  001770                  CLR     TIME            ;INITIALIZE COUNT
001772  052777  000100  176630          BIS     #100,@LKCSR     ;START LINE CLOCK
002000  006300                  ST5:    ASL     R0
002002  033700  001026                  BIT     @#SREG2,R0      ;INHIBIT LINE PRINTER?
002006  001036                          BNE     ST6             ;YES, GO CK NEXT
002010  005777  176622                  TST     @LPCSR
```

```
002014  012767  000137  002076          MOV     #137,SOLPAT             ;RESET FOR START OF LINE PATTERN
002022  012767  000117  001772          MOV     #79.,CLINCT             ;LINE COUNT
002030  012767  000137  002064          MOV     #137,CURPAT
002036  012777  000014  176574          MOV     #14,@LPDBR              ;LINE FEED TO POSITION BUFFER
002044  012777  004000  176570          MOV     #LPINTR,@LPVC           ;INTERRUPT ENABLE
002052  012777  000200  176564          MOV     #200,@LPST              ;PROCESSOR LEVEL 4
002060  032737  000002  001024          BIT     #2,@#SREG1              ;RUN KERNEL ONLY?
002066  001003                          BNE     .+10                    ;YES- BRANCH
002070  052777  040000  176546          BIS     #40000,@LPST
002076  012777  000100  176532          MOV     #100,@LPCSR             ;INTERRUPT ENABLE
002104  005037  001070          ST6:    CLR     @#TRPB                  ;NO "T" BIT FIRST PASS
002110  012737  000006  000004          MOV     #6,@#4                  ;RESTORE TRAP CATCHER
002116  004737  017074                  JSR     %7,@#USER               ;FOR I/O PROGRAM INSERTION
002122  004767  001112                  JSR     %7,DETI                 ;CHECK FOR CORE EXPANSION
002126  032767  000001  176670  PAGE:   BIT     #1,SREG1                ;INHIBIT KT11-C?
002134  001402                          BEQ     .+6
002136  000167  000522                  JMP     STRF                    ;YES - GO CHECK FOR RUNNING RF11
002142  004767  011260                  JSR     %7,NRALL                ;NO - MAKE ALL PAGES INITIALLY NON-RESIDENT
002146  012777  077406  176602          MOV     #77406,@KIPDR7          ;MAP KERNEL 7 TO THE EXTERNAL BANK, RW
002154  012777  077406  176606          MOV     #77406,@KDPDR7
002162  012777  007600  176612          MOV     #7600,@KIPAR7
002170  012777  007600  176616          MOV     #7600,@KDPAR7
002176  032737  000002  001024          BIT     #2,@#SREG1              ;INHIBIT USER/SUPERVISOR/KERNEL?
002204  001423                          BEQ     SEGM1                   ;NO - BRANCH
002206  012701  000007                  MOV     #7,R1                   ;YES - MAP KERNEL PAGES 0-6 TO PA
002212  016702  176554                  MOV     KIPAR0,R2
002216  005003                          CLR     R3
002220  010312          SETEX:  MOV     R3,@R2
002222  010362  000020                  MOV     R3,20(R2)
002226  012762  077406  177740          MOV     #77406,-40(R2)
002234  012762  077406  177760          MOV     #77406,-20(R2)
002242  005722                          TST     (R2)+
002244  062703  000200                  ADD     #200,R3
002250  077115                          SOB     R1,SETEX
002252  000576                          BR      SETSEG
002254  012777  077406  176464  SEGM1:  MOV     #77406,@KIPDR0          ;MAP KERNEL 0 TO BANK 0, RW
002262  012777  077406  176470          MOV     #77406,@KDPDR0
002270  012777  077406  176432          MOV     #77406,@SIPDR0          ;MAP SUPERVISOR 0
002276  012777  077406  176426          MOV     #77406,@SDPDR0          ;TO BANK 0, RW
002304  016767  176420  176552          MOV     SIPDR0,CSIPDR           ;SET UP POINTER TO TRACK CURRENT SUPERVISOR PAGE
002312  032767  000004  176504          BIT     #4,SREG1                ;INHIBIT RUNNING 4K AS 32K?
002320  001453                          BEQ     USEALL                  ;NO, SETUP FOR RUNNING 4K AS 32K
002322  012701  000007                  MOV     #7,R1                   ;YES, MAP USER 0-6 TO PA
002326  016702  176362                  MOV     UIPAR0,R2
002332  005003                          CLR     R3
002334  010312          SETUSE: MOV     R3,(R2)
002336  010362  000020                  MOV     R3,20(R2)
002342  062703  000200                  ADD     #200,R3
002346  012762  077406  177740          MOV     #77406,-40(R2)
002354  012762  077406  177760          MOV     #77406,-20(R2)
002362  005722                          TST     (R2)+
002364  077115                          SOB     R1,SETUSE
002366  012701  000007                  MOV     #7,R1                   ;MAP KERNEL 0 THRU 6 TO PHYSICAL ADDRESS
002372  016702  176374                  MOV     KIPAR0,R2               ;FOR I/O RELOCATION
002376  005003                          CLR     R3
002400  010312          SETKER: MOV     R3,(R2)
```

```
002402  010362  000020                  MOV     R3,20(R2)
002406  062703  000200                  ADD     #200,R3
002412  012762  077406  177740          MOV     #77406,-40(R2)
002420  012762  077406  177760          MOV     #77406,-20(R2)
002426  005722                          TST     (R2)+
002430  077115                          SOB     R1,SETKER
002432  012777  007600  176266          MOV     #7600,@UDPAR7           ;MAP USER 7 D-SPACE TO THE EXTERNAL
002440  012777  077406  176244          MOV     #77406,@UDPDR7          ;BANK TO ALLOW ACCESSING PROCESSOR STATUS
002446  000436                          BR      SETSUP
002450  012777  077406  176222  USEALL: MOV     #77406,@UIPDR0          ;MAP USER 0 TO BANK 0, RW
002456  012777  077406  176222          MOV     #77406,@UDPDR0
002464  012777  077406  176256          MOV     #77406,@KIPDR1          ;MAP KERNEL 1 TO BANK 0, RW
002472  012777  077406  176262          MOV     #77406,@KDPDR1
002500  012767  000000  176342          MOV     #0,CURBNK               ;CURRENT PAR CONTENTS
002506  016767  176202  176340          MOV     UIPAR0,CURIPAR          ;CURRENT PAGE ADDRESS REGISTER ADDRESSES
002514  016767  176202  176334          MOV     UDPAR0,CURDPAR
002522  016767  176152  176330          MOV     UIPDR0,CURIPDR          ;CURRENT PAGE DESCRIPTOR REGISTER ADDRESSES
002530  016767  176152  176324          MOV     UDPDR0,CURDPDR
002536  012767  006700  176322          MOV     #BEGIN,BNKSTR           ;CURRENT STARTING PC
002544  012777  007600  176164  SETSUP: MOV     #7600,@SIPAR7           ;MAP SUPERVISOR 7 TO THE
002552  012777  007600  176160          MOV     #7600,@SDPAR7           ;EXTERNAL BANK
002560  032767  000020  176236          BIT     #20,SREG1               ;CHECK SWITCH 4 - INHIBIT CYCLING ACCESS KEY?
002566  001015                          BNE     SETSP1                  ;YES - BRANCH
002570  012777  077400  176144          MOV     #77400,@SIPDR7          ;MAP SUPERVISOR PAGE 7 4K, NR
002576  012777  077400  176140          MOV     #77400,@SDPDR7
002604  012737  015654  000250          MOV     #ABSRV,@#250
002612  012737  000340  000252          MOV     #340,@#252
002620  000413                          BR      SETSEG
002622  012777  077406  176112  SETSP1: MOV     #77406,@SIPDR7          ;MAP SUPERVISOR PAGE 7 4K, RW
002630  012777  077406  176106          MOV     #77406,@SDPDR7
002636  012737  000252  000250          MOV     #252,@#250
002644  005037  000252                  CLR     @#252
002650  012777  000007  176020  SETSEG: MOV     #7,@SR3                 ;ENABLE ALL D-SPACES
002656  012777  001001  176004          MOV     #1001,@SR0              ;TURN ON MEMORY MANAGEMENT AND ENABLE TRAPPING (
002664  012737  003220  000004  STRF:   MOV     #NODEV,@#4
002672  012700  000100                  MOV     #100,R0
002676  036700  176124                  BIT     SREG2,R0                ;TEST FOR INHIBITING RF11 DISK
002702  001040                          BNE     STTC                    ;SKIP IF SET
002704  005737  000660                  TST     @#RFCSR
002710  016701  176110                  MOV     SREG1,R1
002714  042701  177770                  BIC     #177770,R1
002720  005701                          TST     R1
002722  001004                          BNE     STRF1
002724  012777  024142  175732          MOV     #IRF+20000,@RFVC        ;IF SW0, 1, 2 ALL ZERO, MAP RF11 TO
002732  000403                          BR      .+10                    ;KERNEL 1
002734  012777  004142  175722  STRF1:  MOV     #IRF,@RFVC              ;IF SW0, 1, 2 NOT ALL ZERO, MAP RF11 TO
002742  012777  000240  175716          MOV     #240,@RFST              ;KERNEL 0
002750  012767  043503  001316          MOV     #43503,FUNCTION         ;WRITE CHECK/WRITE
002756  105277  175700                  INCB    @RFCSR+1                ;INITIALIZE DISK-DAR,DAE
002762  016777  001312  175664          MOV     WORDCT,@RFWC            ;LENGTH OF TRANSFER
002770  016777  001302  175660          MOV     LLIMIT,@RFCAR           ;CORE ADDRESS OF START OF TRANSFER
002776  116777  001272  175654          MOVB    FUNCTION,@RFCSR         ;START RF11 READ OR WRITE
003004  012737  004336  004302  STTC:   MOV     #TCBUF,@#TCBUFP         ;INITIALIZE PA AND VA OF TC11 BUFFER
003012  012737  004336  004304          MOV     #TCBUF,@#TCBUFV         ;TO BANK 0
003020  012737  000001  004330          MOV     #1,@#TCEMA              ;INITIALIZE TC11 BUFFER EXTENDED
                                                                        ;MEMORY ADDRESS BITS TO 0
```

```
003026  005037  004332                  CLR     @#TCEOP                 ;CLEAR TC11 END OF PASS FLAG
003032  005037  004334                  CLR     @#TCREL                 ;CLEAR TC11 RELOCATION FLAG
003036  006300                          ASL     R0
003040  036700  175762                  BIT     SREG2,R0                ;CHECK FOR INHIBITING TC11 DECTAPE
003044  001041                          BNE     MODE1                   ;SKIP IF SET
003046  005777  175762                  TST     @TCST
003052  005067  001250                  CLR     DATACK                  ;INITIALIZE FLAG
003056  013701  001024                  MOV     @#SREG1,R1
003062  032701  000050                  BIT     #50,R1
003066  001403                          BEQ     SETREL
003070  032701  000017                  BIT     #17,R1
003074  001002                          BNE     .+6
003076  005237  004334          SETREL: INC     @#TCREL                 ;SET FLAG INDICATING RELOCATION IS ALLOWED
003102  032701  000007                  BIT     #7,R1
003106  001007                          BNE     .+20
003110  012777  025336  175726          MOV     #ENDZ+20000,@TCIV       ;IF SW0, 1, 2 ALL ZERO, MAP TC11
003116  012737  024336  004304          MOV     #TCBUF+20000,@#TCBUFV   ;POINT VIRTUAL ADDRESS OF TC11 DATA
                                                                        ;BUFFER TO KERNEL PAGE 1
003124  000403                          BR      .+10                    ;TO KERNEL PAGE 1
003126  012777  005336  175710          MOV     #ENDZ,@TCIV             ;IF SW0, 1, 2 NOT ALL ZERO, MAP TC11
003134  012777  000300  175704          MOV     #300,@TCSTA             ;TO KERNEL PAGE 0
003142  012777  004503  175662          MOV     #R+IE+RB+DO+UN1,@TCCM   ;START REVERSE READ BLOCK NUMBER
003150  005037  177776          MODE1:  CLR     @#PSR                   ;PRIORITY LEVEL 0
003154  012737  000006  000004          MOV     #6,@#4                  ;RESTORE TRAP CATCHER TO ADDRESS ERROR VECTOR
003162  032767  000002  175634          BIT     #2,SREG1
003170  001005                          BNE     MAIN+2
003172  012737  140000  177776          MOV     #140000,@#PSR           ;CHANGE TO USER
003200  000401                          BR      .+4
003202  000001          MAIN:   WAIT
003204  036727  175616  002000          BIT     SREG2,#2000             ;INHIBIT PROCESSOR TEST
003212  001373                          BNE     MAIN
003214  000167  003460                  JMP     BEGIN

                                ;ROUTINE EXECUTED IF A DEVICE IS NOT INHIBITED BUT IS FOUND
                                ;TO NOT BE PRESENT. IT SETS THE CORRESPONDING INHIBIT IN SREG2
                                ;AND ALTERS THE PC TO SKIP THE STARTUP FOR THE DEVICE
003220  050037  001026  NODEV:  BIS     R0,@#SREG2              ;SET INHIBIT BIT
003224  162716  000006          SUB     #6,(SP)                 ;ALTER RETURN PC TO
                                                                ;SKIP DEVICE STARTUP
003230  042766  000017  000002  BIC     #17,2(SP)               ;CLEAR Z BIT ON STACK
003236  000002                  RTI

                                ;PDP-11 MEMORY DETERMINATION AND SETUP
                                ;USE WITH VARIABLE CORE QUANTITY SYSTEMS
003240  012767  104012  007752  DET1:  MOV     #EOB,DONE               ;RESTORE INITIAL CODE
003246  032767  000007  175550          BIT     #7,SREG1                ;INHIBIT RUNNING 4K AS 32K USER?
                                                                        ;OR INHIBIT KT11-C?
                                                                        ;OR INHIBIT USER & SUPERVISOR MODES?
003254  001001                          BNE     .+4                     ;YES - ALLOW CORE EXPANSION
003256  000207                          RTS     %7                      ;NO - INHIBIT CORE EXPANSION
003260  032737  000040  001024          BIT     #40,@#SREG1             ;CHECK VARIABLE CORE SWITCH
003266  001401                          BEQ     DET4                    ;USE VARIABLE CORE ROUTINE
003270  000207                          RTS     %7                      ;4K ONLY (SWITCH SET)
003272  012737  003342  000004  DET4:  MOV     #DET2,@#4               ;TRAP VECTOR SETUP
003300  012737  000340  000006          MOV     #340,@#6                ;TRAP STATUS SETUP
003306  000241                          CLC
```

```
003310  005537  037770          EIGHT:  ADC     @#37770                 ;CHECK FOR 8K
003314  005537  057770          TWELVE: ADC     @#57770                 ;CHECK FOR 12K
003320  005537  077770          SXTEEN: ADC     @#077770                ;CHECK FOR 16K
003324  005537  117770          TWENTY: ADC     @#117770                ;CHECK FOR 20K
003330  005537  137770          TWOFOR: ADC     @#137770                ;CHECK FOR 24K
003334  005537  157770          TWOEIG: ADC     @#157770                ;CHECK FOR 28K
003340  000437                          BR      STRT28
003342  012602                  DET2:   MOV     (6)+,%2                 ;RETRIEVE TRAP PC
003344  005726                          TST     (6)+                    ;DISCARD TRAP STATUS WORD
003346  062702  000060                  ADD     #STRT4-EIGHT-4,R2
003352  000112                          JMP     @R2

003354  005000                  MOVE:   CLR     %0                      ;SET UP MAIN CORE POINTER
003356  010102                          MOV     %1,%2
003360  062702  013426                  ADD     #D+2,%2                 ;SET UP MAX CORE MOVE
003364  012021                          MOV     (0)+,(1)+               ;MOVE WORD
003366  020201                          CMP     %2,%1                   ;MOVE COMPLETE?
003370  001375                          BNE     .-4                     ;MOVE ANOTHER WORD
003372  000207                          RTS     %7                      ;MOVE COMPLETE
003374  000521                  STRT4:  BR      DET3
003376  000240                          NOP
003400  000240                          NOP
003402  004767  000110          STRT8:  JSR     %7,XFER8                ;START 8K TRANSFER
003406  000506                          BR      MOD4                    ;START 4K MODIFY
003410  004767  000072          STRT12: JSR     %7,XFER12               ;START 12K TRANSFER
003414  000475                          BR      MOD8                    ;START 8K MODIFY
003416  004767  000054          STRT16: JSR     %7,XFER16               ;START 16K TRANSFER
003422  000464                          BR      MOD12                   ;START 12K MODIFY
003424  004767  000036          STRT20: JSR     %7,XFER20               ;START 20K TRANSFER
003430  000453                          BR      MOD16                   ;START 16K MODIFY
003432  004767  000020          STRT24: JSR     %7,XFER24               ;START 24K TRANSFER
003436  000442                          BR      MOD20                   ;START 20K MODIFY
003440  004767  000002          STRT28: JSR     %7,XFER28               ;START 28K TRANSFER
003444  000431                          BR      MOD24                   ;START 24K MODIFY
003446  012701  140000          XFER28: MOV     #140000,%1              ;SET UP MOVE START LOCATION
003452  004767  177676                  JSR     %7,MOVE                 ;GO TO MOVE SUBROUTINE
003456  012701  120000          XFER24: MOV     #120000,%1
003462  004767  177666                  JSR     %7,MOVE
003466  012701  100000          XFER20: MOV     #100000,%1
003472  004767  177656                  JSR     %7,MOVE
003476  012701  060000          XFER16: MOV     #60000,%1
003502  004767  177646                  JSR     %7,MOVE
003506  012701  040000          XFER12: MOV     #40000,%1
003512  004767  177636                  JSR     %7,MOVE
003516  012701  020000          XFER8:  MOV     #20000,%1
003522  004767  177626                  JSR     %7,MOVE
003526  000207                          RTS     %7                      ;RETURN FROM TRANSFERS
003530  012767  000137  127462  MOD24:  MOV     #137,DONE+120000
003536  012767  146466  127456          MOV     #BEGINX+140000,DONE+120002
003544  012767  000137  107446  MOD20:  MOV     #137,DONE+100000
003552  012767  126466  107442          MOV     #BEGINX+120000,DONE+100002
003560  012767  000137  067432  MOD16:  MOV     #137,DONE+60000
003566  012767  106466  067426          MOV     #BEGINX+100000,DONE+60002
003574  012767  000137  047416  MOD12:  MOV     #137,DONE+40000
003602  012767  066466  047412          MOV     #BEGINX+60000,DONE+40002
003610  012767  000137  027402  MOD8:   MOV     #137,DONE+20000
```

```
003616  012767  046466  027376          MOV     #BEGINX+40000,DONE+20002
003624  012767  000137  007366  MOD4:   MOV     #137,DONE
003632  012767  026466  007362          MOV     #BEGINX+20000,DONE+2
003640  005037  000006          DET3:   CLR     @#6
003644  012737  000006  000004          MOV     #6,@#4
003652  000207                          RTS     %7

                                ;TTY TRANSMITTER PRINT VALUES 0 TO 377
003654  005027  000000          TYOUT:  CLR     #0                      ;INITIALIZE DATA
        003656                  DATA2=.-2
003660  016777  177772  174732 TYOUT1: MOV     DATA2,@TTDBR            ;OUTPUT TO DEVICE
003666  104010                  RTIK                                    ;RETURN TO MAINLINE
003670  017767  174722  174730 TYOUTR: MOV     @TTCSR,TTSAV            ;SAVE CONTENTS OF TTY CSR
003676  105767  174724          TSTB    TTSAV                           ;CHECK TO SEE IF READY WAS SET
003702  100401                  BMI     .+4                             ;BRANCH IF FLAG FOUND
003704  104006                  HLT                                     ;FALSE INTERRUPT RETURN
003706  000240                  NOP
003710  005267  177742          INC     DATA2                           ;INCREMENT DATA
003714  022767  000400  177734  CMP     #400,DATA2                      ;TEST DATA FOR UPPER LIMIT
003722  001754                  BEQ     TYOUT                           ;AT UPPER LIMIT START OVER
003724  000755                  BR      TYOUT1                          ;FINISH REST OF DATA

                                ;TEST OF LINE CLOCK -OUTPUT MINUTE COUNT TO UPPER BYTE OF DISPLAY REGISTER.
                                                                        ;ON INTERRUPTS ENTER HERE
003726  105777  174676  LK3:    TSTB    @LKCSR                          ;TEST FOR DONE
003732  100401                  BMI     .+4
003734  104006                  HLT                                     ;FALSE INTERRUPT
003736  042777  000200  174664  BIC     #200,@LKCSR
003744  005267  000012  LK4:    INC     TIME
003750  022767  007020  000004  CMP     #3600.,TIME                     ;ONE MINUTE UP?
003756  003007                  BGT     LK2                             ;NO- EXIT
003760  005027  000000  LK1:    CLR     #0                              ;YES- CLEAR LINE CLOCK TIMER
        003762          TIME=   .-2
003764  105267  175115          INCB    DISPLY+1
003770  016737  175110  177570  MOV     DISPLY,@#SR                     ;LOAD DISPLAY REGISTER
                                                                        ;WITH MINUTE COUNT
003776  104010          LK2:    RTIK                                    ;RETURN TO MAINLINE

                                ;LINE PRINTER INTERRUPT SERVICE ROUTINE
004000  105777  174632  LPINTR: TSTB    @LPCSR                          ;TEST FOR FLAG
004004  100401                  BMI     .+4
004006  104006                  HLT                                     ;FALSE RETURN FROM MAIN LINE
004010  026727  000006  000117  LP6:    CMP     CLINCT,#79.             ;TEST FOR END OF LINE
004016  001415                  BEQ     LP4                             ;GO GENERATE CR/LF
004020  005227  000000          INC     #0                              ;INCREMENT LINE POSITION COUNT
        004022                  CLINCT=.-2                              ;POSITION OF LINE
004024  026727  000072  000137  CMP     CURPAT,#137                     ;TEST FOR MAXIMUM PATTERN
004032  001403                  BEQ     LP3                             ;YES - GO TO LP3 AND RESET
004034  005267  000062          INC     CURPAT                          ;NO - INCREMENT TO NEXT PATTERN
004040  000431                  BR      LP2                             ;GO SEND IT TO LINE PRINTER
004042  012767  000040  000052  LP3:    MOV     #40,CURPAT              ;RESET PATTERN AND SEND TO PRINTER
004050  000425                  BR      LP2                             ;SENT TO LINE PRINTER
004052  005067  177744  LP4:    CLR     CLINCT                          ;RESET LINE COUNT
004056  012777  000012  174554  MOV     #12,@LPDBR                      ;LINE FEED
004064  105777  174546          TSTB    @LPCSR
004070  100375                  BPL     .-4
```

```
004072  026727  000022  000137          CMP     SOLPAT,#137             ;START OF LINE PATTERN
004100  001403                          BEQ     LP5
004102  005267  000012                  INC     SOLPAT                  ;INCREMENT START OF LINE
004106  000403                          BR      LP1
004110  012767  000040  000002  LP5:    MOV     #40,SOLPAT              ;RESET START OF LINE
004116  012727  000000  000000  LP1:    MOV     #0,#0                   ;START OF LINE TO CURRENT
        004122                          CURPAT=.-2                      ;CHARACTER BEING PRINTED
        004120                          SOLPAT=.-4                      ;START OF LINE CHARACTER
004124  016777  177772  174506  LP2:    MOV     CURPAT,@LPDBR           ;CURRENT PATTERN TO LINE PRINTER
004132  105777  174500                  TSTB    @LPCSR
004136  100724                          BMI     LP6
004140  104010                          RTIK                            ;RETURN TO MAIN LINE

                                ;RF11 DISK
004142  105777  174512          IRF:    TSTB    @RFCSR                  ;INTERRUPT VECTOR POINTS HERE
004146  100402                          BMI     .+6
004150  104006                          HLT                             ;RF11 READY NOT UP
004152  000434                          BR      RFSTART
004154  005777  174500                  TST     @RFCSR                  ;ERROR SET?
004160  100012                          BPL     ERROK                   ;BRANCH IF NOT
004162  032777  020000  174470          BIT     #20000,@RFCSR           ;YES-WRITE CHECK ERROR?
004170  001404                          BEQ     ERRSET                  ;NO-BRANCH
004172  104006                          HLT                             ;YES-RF11 WRITE CHECK ERROR
004174  000337  004274                  SWAB    @#FUNCTION              ;CHANGE COMMAND TO DO A WRITE
004200  000421                          BR      RFSTART
004202  104006          ERRSET: HLT                                     ;RF11 ERROR SET-NOT WRITE CHECK ERROR
004204  000417                          BR      RFSTART
004206  005777  174442          ERROK:  TST     @RFWC
004212  100002                          BPL     .+6
004214  104006                          HLT                             ;RF-11 WORD COUNT NOT ZERO
004216  000412                          BR      RFSTART
004220  122777  000003  174420          CMPB    #3,@RFDAE               ;DISK AT UPPER LIMIT? 7=2, 17=4, 37=8
004226  001010                          BNE     RF1                     ;NO
004230  027727  174416  174000          CMP     @RFDAR,#174000          ;AS FAR ON DISK AS WE CAN GO?
004236  101404                          BLOS    RF1                     ;NO
004240  000337  004274                  SWAB    @#FUNCTION              ;CHANGE COMMAND
004244  105277  174412          RFSTART:        INCB    @RFCSRH         ;INITIALIZE DISK - DAR-DAE
004250  013777  004276  174400  RF1:    MOV     @#LLIMIT,@RFCAR         ;CORE BASE
004256  013777  004300  174370          MOV     @#WORDCT,@RFWC          ;LENGTH OF TRANSFER
004264  113777  004274  174366          MOVB    @#FUNCTION,@RFCSR       ;WRITE OR WRITE CHECK TO DISK
004272  104010                          RTIK                            ;RETURN TO MAINLINE CODE
004274  000000          FUNCTION:       0                               ;DISK COMMAND
004276  006172          LLIMIT: NXTBLK                                  ;FIRST CORE ADDRESS OF TRANSFER
004300  176000          WORDCT: -2000                                   ;LENGTH OF TRANSFER

;DECTAPE DIAGNOSTIC ROUTINE.  THE TAPE ON DRIVE 1 IS FIRST DRIVEN TO THE
;FORWARD END ZONE. THE DESIRED DATA IS THEN GENERATED IN THE DECTAPE BUFFER
;AREA AND DATA IS WRITTEN ONTO ALL BLOCKS FROM THE BLOCK NUMBER IN TCFRST
;THRU THE BLOCK NUMBER IN TCLAST.  BLOCK NUMBERS ARE ALSO CHECKED FOR
;BEING IN ORDER. AFTER THE BLOCK NUMBER IN TCLAST IS WRITTEN, TAPE IS
;DRIVEN INTO THE REVERSE END ZONE.
;THE TAPE IS THEN STARTED IN REVERSE, AND WHEN THE CLOSEST BLOCK THAT
;WAS WRITTEN (TCLAST) IS FOUND, IT IS READ INTO THE DECTAPE BUFFER AREA.
;THE PROGRAM INTERRUPT REQUEST FACILITY IS THEN USED TO BOOK A REQUEST
;FOR CHECKING THE DATA AT LEVEL 3, AND NO FURTHER DATA IS READ IN
;UNTIL THAT DATA HAS BEEN CHECKED.  AFTER IT IS CHECKED, THE DATA IS
```

```
                                ;SCRAMBLED TO GUARANTEE THAT NEW DATA IS REALLY READ IN NEXT TIME. WHILE
                                ;THIS IS GOING ON, BLOCK NUMBERS ARE CHECKED FOR BEING IN ORDER AS THE
                                ;TAPE TRAVELS TOWARD THE FORWARD END ZONE.ONCE THE DATA IS FULLY CHECKED
                                ;THE NEXT BLOCK THAT COMES UP IS READ IN AND THE PROCESS REPEATED.  ONCE
                                ;THE BLOCK WHOSE NUMBER IS IN TCFRST HAS BEEN READ, THE TAPE IS DRIVEN
                                ;INTO THE FORWARD END ZONE AND THE WHOLE SEQUENCE IS REPEATED.

                                ;FUNCTION VALUES IN CSR
            000004              RD=4                        ;READ DATA
            000014              WD=14                       ;WRITE DATA
            000002              RB=2                        ;READ BLOCK NUMBER
            000100              IE=100                      ;INTERRUPT ENABLE
            000001              DO=1                        ;DO - THE FUNCTION
            004000              R=4000                      ;REVERSE
            000400              UN1=400                     ;UNIT 1

                                ;DECTAPE DATA
004302      004336              TCBUFP: TCBUF               ;PHYSICAL ADDRESS OF BUFFER
004304      004336              TCBUFV: TCBUF               ;VIRTUAL ADDRESS OF BUFFER
004306      000000              TCFRST: 0                   ;FIRST BLOCK TO BE SEARCHED FOR
004310      001101              TCLAST: 577.                ;LAST BLOCK TO BE SEARCHED FOR
004312      000000              TCBLK:  0                   ;CURRENT BLOCK FOUND
004314      000000              TCEXP:  0                   ;THE BLOCK THAT IS EXPECTED
004316      000000              TCSTOR: 0                   ;SAVES CONTENTS OF TC STATUS
                                                            ;REGISTER AT TIME OF INTERRUPT
004320      000000              FBLK:   0
004322      000000              STOR1:  0
004324      000000              STOR2:  0
004326      000000              DATACK: 0
004330      000001              TCEMA:  1                   ;EXTENDED MEMORY ADDRESS BITS AND DO BIT
004332      000000              TCEOP:  0                   ;END OF PASS FLAG
004334      000000              TCREL:  0                   ;SET IF TC CODE IS TO BE
                                                            ;RELOCATED TO THE SAME BANK AS USER

                                ;DECTAPE BUFFER USED FOR BOTH WRITE AND READ FUNCTIONS
004336      000000              TCBUF:  0
            005336                      .=.+776

                                ;DECTAPE ROUTINE TO DRIVE TAPE INTO ENDZONE
                                ;REGENERATES DECTAPE WRITE BUFFER BEFORE WRITING TAPE
                                ;MUST BE ENTERED WITH FUNCTION SET TO READ BLOCK #
005336  042777  017777  173500  ENDZ:   BIC     #17777,@TCIV
005344  052777  005336  173472          BIS     #ENDZ,@TCIV
005352  005777  173456                  TST     @TCST               ;END ZONE SET?
005356  100403                          BMI     ENDZ1               ;YES-BRANCH
005360  105277  173446                  INCB    @TCCM               ;NO-GO AGAIN
005364  000002                          RTI
005366  042777  017777  173450  ENDZ1:  BIC     #17777,@TCIV
005374  052777  005570  173442          BIS     #RBN,@TCIV
005402  005037  004320                  CLR     @#FBLK              ;LOOKING FOR FIRST DESIRED BLOCK
005406  032777  004000  173416          BIT     #4000,@TCCM         ;REVERSE SET?
005414  001007                          BNE     ENDZ2               ;YES, NOW AT FORWARD END ZONE, BRANCH
005416  052777  004001  173406          BIS     #4001,@TCCM         ;NO, SET IT AND DO
005424  013737  004310  004314          MOV     @#TCLAST,@#TCEXP
005432  000002                          RTI
005434  005737  004334          ENDZ2:  TST     @#TCREL             ;TC RELOCATION ALLOWED?
```

```
005440  001403                          BEQ     BKREQ                   ;NO,BRANCH TO BOOK PIRQ
005442  005737  004332                  TST     @#TCEOP                 ;YES-DON'T STOP TC IF THIS IS
                                                                        ;THE INITIAL SEARCH FOR THE
                                                                        ;FORWARD END ZONE
005446  001003                          BNE     STOPTC
005450  052777  002000  173420  BKREQ:  BIS     #2000,@PIR              ;YES, BOOK LEVEL 2 PIRQ TO RESTART
                                                                        ;TC11 AND REGENERATE WRITE BUFFER
005456  104010                  STOPTC: RTIK

                                ;ROUTINE TO REGENERATE WRITE BUFFER AND START UP DECTAPE WRITING BLOCKS
                                ;FORWARD.  MUST RUN AT LOWER LEVEL THAN READ BUFFER DATA CHECK ROUTINE
                                ;BOTH ARE RUN UNDER PIRQ
005460  010137  004322          REGEN:  MOV     R1,@#STOR1              ;SAVE REGISTERS
005464  010237  004324                  MOV     R2,@#STOR2
005470  005037  004320                  CLR     @#FBLK                  ;INITIALIZE FIRST BLOCK FOUND FLAG
005474  012701  000001                  MOV     #1,R1                   ;REGENERATE WRITE BUFFER
005500  013702  004304                  MOV     @#TCBUFV,R2
005504  010122                  REGENL: MOV     R1,(R2)+
005506  010112                          MOV     R1,@R2
005510  005122                          COM     (R2)+
005512  005201                          INC     R1
005514  022701  000201                  CMP     #201,R1
005520  001371                          BNE     REGENL
005522  013701  004322                  MOV     @#STOR1,R1              ;RESTORE REGISTERS
005526  013702  004324                  MOV     @#STOR2,R2
005532  042777  017777  173304          BIC     #17777,@TCIV            ;CHANGE INTERRUPT RETURN
005540  052777  005570  173276          BIS     #RBN,@TCIV              ;TO READ BLOCK NUMBER ROUTINE
005546  012777  000503  173256          MOV     #RB+IE+DO+UN1,@TCCM     ;FORWARD READ BLOCK #
005554  013737  004306  004314          MOV     @#TCFRST,@#TCEXP
005562  005037  004332                  CLR     @#TCEOP
005566  000207                          RTS     X7

                                ;DECTAPE ROUTINE TO PROCESS INTERRUPT FROM READING A BLOCK NUMBER
                                ;SETS UP FOR WRITING A BLOCK IF TAPE IS GOING FORWARD
                                ;SETS UP FOR READING A BLOCK IF TAPE IS GOING IN REVERSE
                                ;DATACK IS NON ZERO WHENEVER A PREVIOUS READ HAS NOT
                                ;BEEN FULLY CHECKED-IF NONZERO, NO DATA IS READ IN
005570  017727  173240  004316  RBN:    MOV     @TCST,@#TCSTOR          ;SAVE STATUS IN CASE OF ERROR
005576  032777  100200  173226          BIT     #100200,@TCCM           ;CHECK ERROR AND READY
005604  100006                          BPL     RBN2                    ;BRANCH IF ERROR NOT SET
005606  104006                          HLT                             ;ERROR SET-CHECK CONTENTS OF TCSTOR
005610  012777  004502  173214  RBN1:   MOV     #RB+IE+R+UN1,@TCCM      ;REVERSE READ BLOCK NUMBER
005616  000167  177514                  JMP     ENDZ                    ;RESTART ON ERROR
005622  001002                  RBN2:   BNE     RBN3                    ;BRANCH IF READY SET
005624  104006                          HLT                             ;READY NOT SET
005626  000770                          BR      RBN1                    ;RESTART
005630  005737  004320          RBN3:   TST     @#FBLK                  ;FIRST EXPECTED BLOCK FOUND YET?
005634  100526                          BMI     RBN10                   ;YES-BRANCH
005636  027737  173174  004314          CMP     @TCDT,@#TCEXP           ;BLOCK NUMBER EQUAL TO FIRST ONE DESIRED?
005644  001067                          BNE     RBN7                    ;NO, BRANCH
005646  005137  004320                  COM     @#FBLK                  ;YES, SET FIRST BLOCK FOUND FLAG (NEGATIVE INDIC
005652  042777  017777  173164  RBN4:   BIC     #17777,@TCIV
005660  052777  006172  173156          BIS     #NXTBLK,@TCIV
005666  012777  177400  173144          MOV     #-400,@TCWC
005674  013777  004302  173140          MOV     @#TCBUFP,@TCBA
005702  032777  004000  173122          BIT     #4000,@TCCM             ;REVERSE SET?
```

```
005710  001007                          BNE     RBN5                    ;YES-BRANCH TO READ BLOCK
005712  012777  000514  173112          MOV     #WD+IE+UN1,@TCCM        ;NO-WRITE BLOCK
005720  053777  004330  173104          BIS     @#TCEMA,@TCCM
005726  104010                          RTIK
005730  005737  004326          RBN5:   TST     @#DATACK                ;IF NOT ZERO INDICATES READ BUFFER
                                                                        ;NOT YET FULLY CHECKED
005734  001011                          BNE     RBN6                    ;SO DON'T READ BLOCK
005736  005137  004326                  COM     @#DATACK                ;WAS ZERO-SO SET IT NOW
005742  012777  004504  173062          MOV     #RD+R+IE+UN1,@TCCM      ;AND READ A BLOCK
005750  053777  004330  173054          BIS     @#TCEMA,@TCCM
005756  104010                          RTIK
005760  027737  173052  004306  RBN6:   CMP     @TCDT,@#TCFRST
005766  003004                          BGT     .+12
005770  005237  004332                  INC     @#TCEOP
005774  000167  177336                  JMP     ENDZ
006000  042777  017777  173036          BIC     #17777,@TCIV            ;READ BLOCK NUMBER
006006  052777  005570  173030          BIS     #RBN,@TCIV
006014  012777  004503  173010          MOV     #RB+R+DO+IE+UN1,@TCCM
006022  104010                          RTIK
006024  032777  004000  173000  RBN7:   BIT     #4000,@TCCM             ;CHECK REVERSE BIT
006032  001415                          BEQ     RBN9                    ;BRANCH IF FORWARD
006034  027737  172776  004314          CMP     @TCDT,@#TCEXP
006042  003006                          BGT     RBN8                    ;CONTINUE IF NOT THERE YET
006044  104006                          HLT                             ;PAST DESIRED POINT
006046  012777  000502  172756          MOV     #RB+IE+UN1,@TCCM        ;READ BLOCK NUMBER, CHANGE DIRECTION
006054  000167  177256                  JMP     ENDZ
006060  105277  172746          RBN8:   INCB    @TCCM
006064  104010                          RTIK
006066  023777  004314  172742  RBN9:   CMP     @#TCEXP,@TCDT
006074  003371                          BGT     RBN8                    ;BRANCH IF NOT THERE YET
006076  104006                          HLT                             ;PAST DESIRED POINT
006100  012777  004502  172724          MOV     #RB+R+IE+UN1,@TCCM      ;READ BLOCK NUMBER, CHANGE DIRECTION
006106  000167  177224                  JMP     ENDZ
006112  032777  004000  172712  RBN10:  BIT     #4000,@TCCM             ;REVERSE SET?
006120  001011                          BNE     RBN11                   ;YES-BRANCH
006122  023777  004310  172706          CMP     @#TCLAST,@TCDT          ;NO, CHECK FOR PAST LAST DESIRED BLOCK
006130  002002                          BGE     .+6
006132  000167  177200                  JMP     ENDZ                    ;IF PAST GO TO ENDZ ROUTINE
006136  005237  004314                  INC     @#TCEXP                 ;UP BLOCK NUMBER EXPECTED
006142  000402                          BR      .+6
006144  005337  004314          RBN11:  DEC     @#TCEXP
006150  023777  004314  172660          CMP     @#TCEXP,@TCDT           ;SEQUENTIAL BLOCK?
006156  001002                          BNE     .+6                     ;NO
006160  000167  177466                  JMP     RBN4                    ;YES, READ OR WRITE IT
006164  104006                          HLT                             ;BLOCK NUMBER NOT SEQUENTIAL
006166  000167  177416                  JMP     RBN1                    ;RESTART

                        ;DECTAPE ROUTINE INTERRUPTED TO AFTER READING OR WRITING A BLOCK
006172  017737  172636  004316  NXTBLK: MOV     @TCST,@#TCSTOR          ;SAVE DTA STATUS IN CASE OF ERROR
006200  032777  100200  172624          BIT     #100200,@TCCM           ;CHECK ERROR AND READY
006206  100003                          BPL     NXTB1                   ;BRANCH IF ERROR NOT SET
006210  104006                          HLT                             ;ERROR SET-CK CONTENTS OF TCSTOR
006212  000167  177372                  JMP     RBN1                    ;RESTART
006216  001003                  NXTB1:  BNE     NXTB2                   ;BRANCH IF READY SET
006220  104006                          HLT                             ;READY NOT SET
006222  000167  177362                  JMP     RBN1
```

```
006226  105737  004326          NXTB2:  TSTB    @#DATACK                ;CLEARED INDICATES REQUEST ALREADY MADE
                                                                        ;OR NO DATA WAS READ
006232  001405                          BEQ     NXTB3
006234  105037  004326                  CLRB    @#DATACK                ;CLEAR AND BOOK
006240  052777  004000  172630          BIS     #4000,@PIR              ;BOOK A PIRQ AT LEVEL 3
006246  032777  004000  172556  NXTB3:  BIT     #R,@TCCM                ;REVERSE SET?
006254  001022                          BNE     NXTB4                   ;YES,SKIP NEXT
006256  042777  017777  172560          BIC     #17777,@TCIV
006264  112777  000103  172540          MOVB    #RB+IE+DO,@TCCM
006272  023737  004310  004314          CMP     @#TCLAST,@#TCEXP
006300  003004                          BGT     .+12
006302  052777  005336  172534          BIS     #ENDZ,@TCIV
006310  104010                          RTIK
006312  052777  005570  172524          BIS     #RBN,@TCIV
006320  104010                          RTIK
006322  042777  017777  172514  NXTB4:  BIC     #17777,@TCIV
006330  112777  000103  172474          MOVB    #RB+IE+DO,@TCCM
006336  023737  004306  004314          CMP     @#TCFRST,@#TCEXP
006344  002407                          BLT     .+20
006346  052777  005336  172470          BIS     #ENDZ,@TCIV
006354  012737  000001  004332          MOV     #1,@#TCEOP
006362  104010                          RTIK
006364  052777  005570  172452          BIS     #RBN,@TCIV
006372  104010                          RTIK

                                ;DECTAPE ROUTINE TO CHECK DATA READ IN DURING READ BLOCK
006374  000240                  TCCK:   NOP                             ;FOR DEBUGGING IF NEEDED
006376  000240                          NOP
006400  000240                          NOP
006402  010137  004322                  MOV     R1,@#STOR1
006406  010237  004324                  MOV     R2,@#STOR2
006412  012701  000200                  MOV     #200,R1
006416  013702  004304                  MOV     @#TCBUFV,R2
006422  005112                  TCCK1:  COM     @R2
006424  020122                          CMP     R1,(R2)+
006426  001402                          BEQ     .+6
006430  104006                          HLT                             ;TC11 DATA ERROR
006432  000406                          BR      TCCK2
006434  020112                          CMP     R1,@R2
006436  001402                          BEQ     .+6
006440  104006                          HLT                             ;TC11 DATA ERROR
006442  000402                          BR      TCCK2
006444  005422                          NEG     (R2)+                   ;SCRAMBLE
006446  077113                          SOB     R1,TCCK1
006450  013701  004322          TCCK2:  MOV     @#STOR1,R1
006454  013702  004324                  MOV     @#STOR2,R2
006460  005037  004326                  CLR     @#DATACK
006464  000207                          RTS     %7

006466  010701                  BEGINX: MOV     PC,R1                   ;SETUP R1 TO CONTAIN TOP BITS OF
006470  042701  017777                  BIC     #17777,R1               ;ADDRESS (SELECTS BANK)
006474  072127  177772                  ASH     #-6,R1                  ;RIGHT SHIFT SIX
006500  010137  001050                  MOV     R1,@#CURBNK
006504  072127  000006                  ASH     #6,R1                   ;LEFT SHIFT SIX
006510  042737  160000  000034          BIC     #160000,@#34            ;SETUP SCOPE VECTOR RETURN TO
006516  050137  000034                  BIS     R1,@#34                 ;REFERENCE CURRENT BANK
```

```
006522  032737  000010  001024          BIT     #10,@#SREG1             ;INHIBIT I/O RELOCATION?
006530  001063                          BNE     BEGIN                   ;YES - SKIP REST
006532  052737  000340  177776          BIS     #340,@#PSR              ;RAISE PRIORITY TO LEVEL 7
006540  042777  160000  172116          BIC     #160000,@RFVC           ;CHANGE RF VECTOR RETURN TO ADDRESS
006546  050177  172112                  BIS     R1,@RFVC                ;NEW BANK
006552  042737  160000  004276          BIC     #160000,@#LLIMIT        ;CHANGE RF BUFFER ADDRESS TO
006560  050137  004276                  BIS     R1,@#LLIMIT             ;REFERENCE NEW BANK
006564  042737  000340  177776          BIC     #340,@#PSR              ;DROP PRIORITY TO LEVEL 0
006572  005737  004334                  TST     @#TCREL                 ;TC11 RELOCATING?
006576  001440                          BEQ     BEGIN                   ;NO-SKIP
006600  005737  004332                  TST     @#TCEOP                 ;WAIT FOR TC11 TO FINISH A FULL
006604  001775                          BEQ     .-4                     ;PASS
006606  042777  000100  172216          BIC     #100,@TCCM              ;CLEAR TC11 IE
006614  005777  172212                  TST     @TCCM                   ;WAIT FOR ENDZONE
006620  100375                          BPL     .-4
006622  005737  004326                  TST     @#DATACK                ;WAIT FOR DATA TO BE CHECKED
006626  100775                          BMI     .-4
006630  012737  004336  004302          MOV     #TCBUF,@#TCBUFP         ;SETUP TC11 PHYSICAL BUFFER ADDRESS
006636  050137  004302                  BIS     R1,@#TCBUFP             ;TO REFERENCE NEW BANK
006642  012737  004336  004304          MOV     #TCBUF,@#TCBUFV         ;SETUP TC11 VIRTUAL BUFFER ADDRESS
006650  050137  004304                  BIS     R1,@#TCBUFV             ;TO REFERENCE NEW BANK
006654  012777  005336  172162          MOV     #ENDZ,@TCIV             ;SETUP TC11 VECTOR RETURN ADDRESS
006662  050177  172156                  BIS     R1,@TCIV                ;TO REFERENCE NEW BANK
006666  005037  004332                  CLR     @#TCEOP                 ;REINITIALIZE TC END OF PASS FLAG
006672  052777  002000  172176          BIS     #2000,@PIR              ;BOOK LEVEL 2 REQUEST TO REGENERATE
                                                                        ;WRITE BUFFER AND RUN ANOTHER PASS
                                                                        ;OF DECTAPE CODE

                                .SBTTL BACKGROUND CPU TESTS
                        ;BINARY INSTRUCTIONS
                        ;INDEX, AND INDIRECT TEST OF PDP-11
006700  010767  004454          BEGIN:  MOV     PC,RETURN               ;FOR SCOPING - SETUP ADDRESS OF BEGIN2 IN
006704  062767  000142  004446          ADD     #BEGIN2-BEGIN-2,RETURN  ;THIS BANK THRU CURRENT ASR IN RETURN
006712  010705                          MOV     PC,R5                   ;SETUP R5 WITH VIRTUAL ADDRESS
006714  042705  017777                  BIC     #17777,R5               ;OFFSET
006720  012767  013362  004436          MOV     #B,B+2                  ;SETUP DATA FOR BACKGROUND CPU TESTS
006726  060567  004432                  ADD     R5,B+2
006732  012767  013376  004434          MOV     #A+4,A+2
006740  060567  004430                  ADD     R5,A+2
006744  012767  013402  004426          MOV     #A+10,A+6
006752  060567  004422                  ADD     R5,A+6
006756  012767  013404  004422          MOV     #C,C+2
006764  060567  004416                  ADD     R5,C+2
006770  012767  013414  004420          MOV     #TEMP,TEMP+2
006776  060567  004414                  ADD     R5,TEMP+2
007002  012767  013424  004412          MOV     #TEMP+10,TEMP+6
007010  060567  004406                  ADD     R5,TEMP+6
007014  012767  000400  004332          MOV     #400,ICOUNT             ;ITERATION COUNT
                        ;TEST COMPARE INSTRUCTION INDEXED
007022  012700  177770          BEGIN1: MOV     J-10,R0                 ;R0,R1,AND R2 ARE RESERVED FOR SPECIFIC VALUES
007026  060500                          ADD     R5,R0
007030  012701  177774                  MOV     #-4,R1
007034  060501                          ADD     R5,R1
007036  012702  000010                  MOV     #+10,R2
007042  060502                          ADD     R5,R2
```

```
007044  026027  013372  125252  BEGIN2: CMP     A(0),#125252            ;(A INDEX BY MINUS 10) TO #125252
007052  001401                          BEQ     .+4
007054  104006                          HLT                             ;COMPARE WITH INDEX FAILED
007056  104400                          SCOPE

007060  012703  000004                  MOV     #4,R3
007064  060503                          ADD     R5,R3
007066  026063  013372  013372          CMP     A(0),A(3)
007074  001401                          BEQ     .+4
007076  104006                          HLT
007100  026360  013372  013372          CMP     A(3),A(0)
007106  001401                          BEQ     .+4
007110  104006                          HLT
007112  104400                          SCOPE

007114  026162  013372  013372          CMP     A(1),A(2)
007122  001401                          BEQ     .+4
007124  104006                          HLT
007126  026261  013372  013372          CMP     A(2),A(1)
007134  001401                          BEQ     .+4
007136  104006                          HLT
007140  104400                          SCOPE

                                ;TEST MOVE INSTRUCTION FOR INDEXING

007142  016067  013372  004244          MOV     A(0),TEMP
007150  026727  004240  125252          CMP     TEMP,#125252
007156  001401                          BEQ     .+4
007160  104006                          HLT
007162  104400                          SCOPE

007164  012762  052525  013414          MOV     #052525,TEMP(2)
007172  026727  004226  052525          CMP     TEMP+10,#052525
007200  001401                          BEQ     .+4
007202  104006                          HLT
007204  104400                          SCOPE

                                ;TEST BIC INSTRUCTION FOR INDEXING

007206  012767  177777  004200          MOV     #-1,TEMP
007214  046267  013372  004172          BIC     A(2),TEMP
007222  026727  004166  125252          CMP     TEMP,#125252
007230  001401                          BEQ     .+4
007232  104006                          HLT
007234  104400                          SCOPE

007236  012767  177777  004140          MOV     #-1,TEMP-10
007244  042767  052525  004132          BIC     #052525,TEMP-10
007252  026727  004126  125252          CMP     TEMP-10,#125252
007260  001401                          BEQ     .+4
007262  104006                          HLT
007264  104400                          SCOPE

                                ;TEST SUBTRACT INSTRUCTION FOR INDEXING

007266  012767  125252  004120          MOV     #125252,TEMP
```

```
007274  166760  004062  013424          SUB     B,TEMP+10(0)
007302  001401                          BEQ     .+4
007304  104006                          HLT
007306  104400                          SCOPE

007310  012767  052525  004076          MOV     #052525,TEMP
007316  166267  013372  004070          SUB     A(2),TEMP
007324  001401                          BEQ     .+4
007326  104006                          HLT
007330  104400                          SCOPE

                                ;TEST UNARYS INDEXED
007332  012767  177777  004054          MOV     #-1,TEMP
007340  005060  013424                  CLR     D(0)
007344  005767  004044                  TST     TEMP
007350  001401                          BEQ     .+4
007352  104006                          HLT
007354  104400                          SCOPE

007356  012767  177777  004030          MOV     #-1,TEMP
007364  005162  013404                  COM     C(2)
007370  005767  004020                  TST     TEMP
007374  001401                          BEQ     .+4
007376  104006                          HLT
007400  104400                          SCOPE

007402  012767  177777  004004          MOV     #-1,TEMP
007410  005262  013404                  INC     C(2)
007414  005767  003774                  TST     TEMP
007420  001401                          BEQ     .+4
007422  104006                          HLT
007424  104400                          SCOPE

007426  012767  000001  003760          MOV     #1,TEMP
007434  005360  013424                  DEC     D(0)
007440  005767  003750                  TST     TEMP
007444  001401                          BEQ     .+4
007446  104006                          HLT
007450  104400                          SCOPE

007452  012767  000001  003734          MOV     #1,TEMP
007460  005460  013424                  NEG     D(0)
007464  022767  177777  003722          CMP     #-1,TEMP
007472  001401                          BEQ     .+4
007474  104006                          HLT
007476  104400                          SCOPE

007500  012767  177777  003706          MOV     #-1,TEMP
007506  000261                          SEC
007510  005562  013404                  ADC     C(2)
007514  005767  003674                  TST     TEMP
007520  001401                          BEQ     .+4
007522  104006                          HLT
007524  104400                          SCOPE

007526  012767  000001  003660          MOV     #1,TEMP
```

```
007534  000261                          SEC
007536  005660  013424                  SBC     D(0)
007542  005767  003646                  TST     TEMP
007546  001401                          BEQ     .+4
007550  104006                          HLT
007552  104400                          SCOPE

                                ;TEST JMP INDIRECT

007554  010704                          MOV     %7,%4
007556  062704  000010                  ADD     #10,%4
007562  000114                          JMP     @%4
007564  104006                          HLT
007566  000240                          NOP
007570  104400                          SCOPE


                                ;TEST INDIRECT ADDRESSING WITH INDEXING
                                ;TEST COMPARE INSTRUCTION
007572  027727  003566  125252          CMP     @8+2,#125252
007600  001401                          BEQ     .+4
007602  104006                          HLT
007604  104400                          SCOPE

                                ;TEST MOVE INSTRUCTIONS
007606  012777  125252  003602          MOV     #125252,@TEMP+2
007614  026767  003542  003572          CMP     B,TEMP
007622  001401                          BEQ     .+4
007624  104006                          HLT
007626  104400                          SCOPE

                                ;TEST BIC INSTRUCTION INDIRECT WITH INDEXING
007630  012767  177777  003556          MOV     #-1,TEMP
007636  042777  125252  003552          BIC     #125252,@TEMP+2
007644  022767  052525  003542          CMP     #52525,TEMP
007652  001401                          BEQ     .+4
007654  104006                          HLT
007656  104400                          SCOPE

                                ;TEST SUBTRACT INSTRUCTION INDIRECT WITH INDEXING
007660  012767  125252  003526          MOV     #125252,TEMP
007666  167777  003472  003522          SUB     @8+2,@TEMP+2
007674  005767  003514                  TST     TEMP
007700  001401                          BEQ     .+4
007702  104006                          HLT
007704  104400                          SCOPE

                                ;TEST ADD INDIRECT WITH INDEXING
007706  012767  125252  003500          MOV     #125252,TEMP
007714  067777  003460  003474          ADD     @A+6,@TEMP+2
007722  026727  003466  177777          CMP     TEMP,#-1
007730  001401                          BEQ     .+4
007732  104006                          HLT
007734  104400                          SCOPE

                                ;TEST UNARYS INDIRECT WITH INDEXING
```

```
007736  012767  177777  003450          MOV     #-1,TEMP
007744  005077  003446                  CLR     @TEMP+2
007750  005767  003440                  TST     TEMP
007754  001401                          BEQ     .+4
007756  104006                          HLT
007760  104400                          SCOPE

007762  012767  125252  003424          MOV     #125252,TEMP
007770  005177  003422                  COM     @TEMP+2
007774  022767  052525  003412          CMP     #52525,TEMP
010002  001401                          BEQ     .+4
010004  104006                          HLT
010006  104400                          SCOPE

010010  005067  003400                  CLR     TEMP
010014  005377  003376                  DEC     @TEMP+2
010020  026727  003370  177777          CMP     TEMP,#-1
010026  001401                          BEQ     .+4
010030  104006                          HLT
010032  104400                          SCOPE

010034  012767  000001  003352          MOV     #1,TEMP
010042  005477  003350                  NEG     @TEMP+2
010046  022767  177777  003340          CMP     #-1,TEMP
010054  001401                          BEQ     .+4
010056  104006                          HLT
010060  104400                          SCOPE

010062  012767  177777  003324          MOV     #-1,TEMP
010070  000261                          SEC
010072  005577  003320                  ADC     @TEMP+2
010076  005767  003312                  TST     TEMP
010102  001401                          BEQ     .+4
010104  104006                          HLT
010106  104400                          SCOPE

010110  012767  000001  003276          MOV     #1,TEMP
010116  000261                          SEC
010120  005677  003272                  SBC     @TEMP+2
010124  005767  003264                  TST     TEMP
010130  001401                          BEQ     .+4
010132  104006                          HLT
010134  104400                          SCOPE

                        ;TEST BIC INSTRUCTION
010136  012767  177777  003240          MOV     #-1,C
010144  012703  177772                  MOV     #-6,%3
010150  060503                          ADD     R5,R3
010152  010304                          MOV     %3,%4
010154  047374  013372  013414          BIC     @A(3),@TEMP(4)
010162  022767  052525  003214          CMP     #52525,C
010170  001401                          BEQ     .+4
010172  104006                          HLT
010174  104400                          SCOPE

                        ;TEST COMPARE INSTRUCTION INDEXED
```

```
010176  126162  013372  013372          CMPB    A(1),A(2)
010204  001401                          BEQ     .+4
010206  104006                          HLT
010210  126261  013372  013372          CMPB    A(2),A(1)
010216  001401                          BEQ     .+4
010220  104006                          HLT
010222  104400                          SCOPE

                                ;TEST MOVE BYTE INSTRUCTION INDEXED
010224  116267  013372  003162          MOVB    A(2),TEMP
010232  126727  003156  000125          CMPB    TEMP,#000125
010240  001401                          BEQ     .+4
010242  104006                          HLT
010244  104400                          SCOPE

                                ;TEST BICB INSTRUCTION FOR INDEXING

010246  012767  177777  003150          MOV     #-1,TEMP+10
010254  142762  125252  013414          BICB    #125252,TEMP(2)
010262  126727  003136  002525          CMPB    TEMP+10,#2525
010270  001401                          BEQ     .+4
010272  104006                          HLT
010274  104400                          SCOPE

                                ;TEST UNARYS INDEXED (BYTE INSTRUCTIONS)

010276  012767  177777  003110          MOV     #-1,TEMP
010304  012703  177771                  MOV     #-7,%3
010310  060503                          ADD     R5,R3
010312  105063  013424                  CLRB    D(3)
010316  026727  003072  000377          CMP     TEMP,#000377
010324  001401                          BEQ     .+4
010326  104006                          HLT
010330  104400                          SCOPE

010332  012767  177777  003054          MOV     #-1,TEMP
010340  105160  013424                  COMB    D(0)
010344  105767  003044                  TSTB    TEMP
010350  001401                          BEQ     .+4
010352  104006                          HLT
010354  104400                          SCOPE

010356  012767  177777  003030          MOV     #-1,TEMP
010364  105260  013424                  INCB    D(0)
010370  026727  003020  177400          CMP     TEMP,#177400
010376  001401                          BEQ     .+4
010400  104006                          HLT
010402  104400                          SCOPE

010404  012767  000001  003002          MOV     #1,TEMP
010412  105362  013404                  DECB    C(2)
010416  105767  002772                  TSTB    TEMP
010422  001401                          BEQ     .+4
010424  104006                          HLT
010426  104400                          SCOPE
```

```
010430  012767  000001  002756          MOV     #1,TEMP
010436  105460  013424                  NEGB    D(0)
010442  026727  002746  000377          CMP     TEMP,#377
010450  001401                          BEQ     .+4
010452  104006                          HLT
010454  104400                          SCOPE

010456  012767  177777  002730          MOV     #-1,TEMP
010464  000261                          SEC
010466  105562  013404                  ADCB    C(2)
010472  026727  002716  177400          CMP     TEMP,#177400
010500  001401                          BEQ     .+4
010502  104006                          HLT
010504  104400                          SCOPE

010506  012767  000401  002700          MOV     #401,TEMP
010514  012704  177771                  MOV     #-7,%4
010520  060504                          ADD     R5,R4
010522  000261                          SEC
010524  105664  013424                  SBCB    D(4)
010530  022767  000001  002656          CMP     #1,TEMP
010536  001401                          BEQ     .+4
010540  104006                          HLT
010542  104400                          SCOPE

                                ;TEST INDIRECT ADDRESSING WITH INDEXING
                                ;TEST COMPARE INSTRUCTION
010544  127727  002614  125252          CMPB    @B+2,#125252
010552  001401                          BEQ     .+4
010554  104006                          HLT
010556  104400                          SCOPE

010560  122777  125252  002576          CMPB    #125252,@B+2
010566  001401                          BEQ     .+4
010570  104006                          HLT
010572  104400                          SCOPE

                                ;TEST MOVE INSTRUCTIONS
010574  117703  002564                  MOVB    @B+2,%3
010600  122703  125252                  CMPB    #125252,%3
010604  001401                          BEQ     .+4
010606  104006                          HLT
010610  104400                          SCOPE

010612  112777  125252  002576          MOVB    #125252,@TEMP+2
010620  126767  002536  002566          CMPB    B,TEMP
010626  001401                          BEQ     .+4
010630  104006                          HLT
010632  104400                          SCOPE

                                ;TEST BIC INSTRUCTION INDIRECT WITH INDEXING
010634  012704  177777                  MOV     #-1,%4
010640  147704  002520                  BICB    @B+2,%4
010644  120427  052525                  CMPB    %4,#52525
010650  001401                          BEQ     .+4
010652  104006                          HLT
```

```
010654  104400                          SCOPE

010656  012767  177777  002530          MOV     #-1,TEMP
010664  142777  125252  002524          BICB    #125252,@TEMP+2
010672  122767  052525  002514          CMPB    #52525,TEMP
010700  001401                          BEQ     .+4
010702  104006                          HLT
010704  104400                          SCOPE

                                ;TEST UNARYS INDIRECT WITH INDEXING
010706  012767  177777  002500          MOV     #-1,TEMP
010714  105077  002476                  CLRB    @TEMP+2
010720  105767  002470                  TSTB    TEMP
010724  001401                          BEQ     .+4
010726  104006                          HLT
010730  104400                          SCOPE

010732  012767  125252  002454          MOV     #125252,TEMP
010740  105177  002452                  COMB    @TEMP+2
010744  122767  052525  002442          CMPB    #052525,TEMP
010752  001401                          BEQ     .+4
010754  104006                          HLT
010756  104400                          SCOPE

010760  005067  002430                  CLR     TEMP
010764  105277  002426                  INCB    @TEMP+2
010770  122767  000001  002416          CMPB    #1,TEMP
010776  001401                          BEQ     .+4
011000  104006                          HLT
011002  104400                          SCOPE

011004  005067  002404                  CLR     TEMP
011010  105377  002402                  DECB    @TEMP+2
011014  126727  002374  177777          CMPB    TEMP,#-1
011022  001401                          BEQ     .+4
011024  104006                          HLT
011026  104400                          SCOPE

011030  012767  000001  002356          MOV     #1,TEMP
011036  105477  002354                  NEGB    @TEMP+2
011042  122767  177777  002344          CMPB    #-1,TEMP
011050  001401                          BEQ     .+4
011052  104006                          HLT
011054  104400                          SCOPE

011056  012767  177777  002330          MOV     #-1,TEMP
011064  000261                          SEC
011066  105577  002324                  ADCB    @TEMP+2
011072  022767  177400  002314          CMP     #177400,TEMP
011100  001401                          BEQ     .+4
011102  104006                          HLT
011104  105767  002304                  TSTB    TEMP
011110  001401                          BEQ     .+4
011112  104006                          HLT
011114  104400                          SCOPE
```

```
011116  012767  000001  002270          MOV     #1,TEMP
011124  000261                          SEC
011126  105377  002264                  DECB    @TEMP+2
011132  005767  002256                  TST     TEMP
011136  001401                          BEQ     .+4
011140  104006                          HLT
011142  104400                          SCOPE

                                ;TEST OF COMBINED INDEXING AND INDIRECT

011144  012703  177772                  MOV     #-6,%3
011150  012704  000002                  MOV     #+2,%4
011154  060503                          ADD     R5,R3
011156  060504                          ADD     R5,R4
011160  127374  013372  013372          CMPB    @A(3),@A(4)
011166  001401                          BEQ     .+4
011170  104006                          HLT
011172  104400                          SCOPE

                                ;TEST BIC INSTRUCTION
011174  012703  000006                  MOV     #+6,%3
011200  012767  177777  002206          MOV     #-1,TEMP
011206  060503                          ADD     R5,R3
011210  147367  013372  002176          BICB    @A(3),TEMP
011216  122767  125252  002170          CMPB    #125252,TEMP
011224  001401                          BEQ     .+4
011226  104006                          HLT
011230  104400                          SCOPE

011232  012704  177772                  MOV     #-6,%4
011236  012767  177777  002140          MOV     #-1,C
011244  060504                          ADD     R5,R4
011246  142774  125252  013414          BICB    #125252,@TEMP(4)
011254  126727  002124  000125          CMPB    C,#000125
011262  001401                          BEQ     .+4
011264  104006                          HLT
011266  104400                          SCOPE

011270  012703  013364                  MOV     #B+2,%3         ;ADDRESS OF ADDRESS OF B
011274  060503                          ADD     R5,R3
011276  023367  002060                  CMP     @(3)+,B
011302  001401                          BEQ     .+4
011304  104006                          HLT
011306  104400                          SCOPE

011310  012704  013366                  MOV     #B+4,%4
011314  060504                          ADD     R5,R4
011316  025467  002040                  CMP     @-(4),B
011322  001401                          BEQ     .+4
011324  104006                          HLT
011326  104400                          SCOPE

011330  012703  013410                  MOV     #C+4,%3
011334  060503                          ADD     R5,R3
011336  012767  177777  002040          MOV     #-1,C
011344  105053                          CLRB    @-(3)
```

```
011346  026727  002032  177400          CMP     C,#177400
011354  001401                          BEQ     .+4
011356  104006                          HLT
011360  104400                          SCOPE

011362  012767  177777  002014          MOV     #-1,C
011370  012703  177772                  MOV     #-6,%3
011374  060503                          ADD     R5,R3
011376  010304                          MOV     R3,R4
011400  147374  013372  013414          BICB    @A(3),@TEMP(4)
011406  022767  177525  001770          CMP     #177525,C
011414  001401                          BEQ     .+4
011416  104006                          HLT
011420  104400                          SCOPE

                                ;TEST JSR INSTRUCTION
011422  012703  011434                  MOV     #TJSR1,R3
011426  060503                          ADD     R5,R3
011430  004767  000002                  JSR     %7,TJSR2        ;PLACE PC ON STACK
011434  000404                TJSR1:    BR      TJSR3           ;RETURN HERE ON RTS %7
011436  021603                TJSR2:    CMP     @%6,R3          ;CHECK FOR CORRECT PC ON STACK
011440  001401                          BEQ     .+4
011442  104006                          HLT                     ;INCORRECT PC ON STACK
011444  000207                          RTS     %7              ;RETURN TO INST AFTER JSR
011446  104400                TJSR3:    SCOPE

011450  000257                          CCC
011452  012704  011462                  MOV     #TJSR4,R4
011456  060504                          ADD     R5,R4
011460  004717                          JSR     %7,@%7          ;INSTRUCTION UNDER TEST
011462  021604                TJSR4:    CMP     @%6,R4          ;TEST THE STACK
011464  001401                          BEQ     .+4
011466  104006                          HLT                     ;PC OF JSR DID NOT GO TO STACK
011470  005726                          TST     (6)+            ;REPOSITION THE STACK
011472  104400                          SCOPE

                                ;TEST NESTED SUBROUTINES
011474  000257                          CCC                     ;CLEAR CONDITION CODES
011476  004767  001552                  JSR     %7,SUBR6
011502  100401                          BMI     .+4
011504  104006                          HLT                     ;JSR OR RTS FAILED
011506  001401                          BEQ     .+4
011510  104006                          HLT                     ;JSR OR RTS FAILED
011512  102401                          BVS     .+4
011514  104006                          HLT                     ;JSR OR RTS FAILED
011516  103401                          BCS     .+4
011520  104006                          HLT                     ;JSR OR RTS FAILED
011522  104400                          SCOPE

                                ;TEST ROTATE ODD BYTE
011524  104400                          SCOPE
011526  000257                          CCC                     ;CLEAR "C"
011530  012767  123456  001656          MOV     #123456,TEMP
011536  106067  001653                  RORB    TEMP+1          ;ROTATE ODD BYTE
011542  103401                          BCS     .+4
011544  104006                          HLT                     ;C NOT SET
```

```
011546  102401                  BVS     .+4
011550  104006                  HLT
011552  022767  051456  001634  CMP     #051456,TEMP            ;V NOT SET
011560  001401                  BEQ     .+4
011562  104006                  HLT                             ;ROTATE FAILED
011564  104400                  SCOPE

011566  000277                  SCC                             ;SET C
011570  012767  123456  001616  MOV     #123456,TEMP
011576  106067  001613          RORB    TEMP+1
011602  103401                  BCS     .+4
011604  104006                  HLT                             ;C NOT SET
011606  102001                  BVC     .+4
011610  104006                  HLT                             ;V NOT CLEARED
011612  022767  151456  001574  CMP     #151456,TEMP
011620  001401                  BEQ     .+4
011622  104006                  HLT                             ;ROTATE FAILED
011624  104400                  SCOPE

011626  000257                  CCC
011630  012767  123456  001556  MOV     #123456,TEMP
011636  106167  001553          ROLB    TEMP+1
011642  103401                  BCS     .+4
011644  104006                  HLT                             ;C NOT SET
011646  102401                  BVS     .+4
011650  104006                  HLT                             ;V NOT SET
011652  022767  047056  001534  CMP     #047056,TEMP
011660  001401                  BEQ     .+4
011662  104006                  HLT                             ;ROTATE BYTE FAILED
011664  104400                  SCOPE

011666  000277                  SCC                             ;SET C
011670  012767  123456  001516  MOV     #123456,TEMP
011676  106167  001513          ROLB    TEMP+1
011702  103401                  BCS     .+4
011704  104006                  HLT                             ;C NOT SET
011706  102401                  BVS     .+4
011710  104006                  HLT                             ;V NOT SET
011712  022767  047456  001474  CMP     #047456,TEMP
011720  001401                  BEQ     .+4
011722  104006                  HLT                             ;ROTATE ODD BYTE FAILED
011724  104400                  SCOPE

011726  000257                  CCC                             ;CLEAR C
011730  012767  177777  001456  MOV     #-1,TEMP
011736  106267  001453          ASRB    TEMP+1
011742  103401                  BCS     .+4
011744  104006                  HLT                             ;C NOT SET
011746  102001                  BVC     .+4
011750  104006                  HLT                             ;V NOT CLEARED
011752  026727  001436  177777  CMP     TEMP,#-1
011760  001401                  BEQ     .+4
011762  104006                  HLT                             ;SHIFT FAILED
011764  104400                  SCOPE

011766  000277                  SCC
```

```
011770  012767  177777  001416          MOV     #-1,TEMP
011776  106367  001413                  ASLB    TEMP+1
012002  103401                          BCS     .+4
012004  104006                          HLT                     ;C NOT SET
012006  102001                          BVC     .+4
012010  104006                          HLT                     ;V NOT CLEARED
012012  026727  001376  177377          CMP     TEMP,#177377
012020  001401                          BEQ     .+4
012022  104006                          HLT                     ;SHIFT BYTE FAILED
012024  104400                          SCOPE

                                ;END OF RESTRICTION ON R0-R2
012026  005067  001322                  CLR     ICOUNT          ;NO ITERATION
012032  032767  004000  166766          BIT     #4000,SREG2
012040  001076                          BNE     TSRT2A

                                ;TEST ALL COMBINATIONS OF NUMBERS WITH COMPARE INSTRUCTION
012042  005000                  COMPAR: CLR     %0              ;INIT %0
012044  005001                          CLR     %1              ;INIT %1
012046  020001                  CMP1:   CMP     %0,%1           ;ARE THEY EQUAL?
012050  001401                          BEQ     .+4
012052  104006                          HLT                     ;R0 AND R1 DID NOT COMPARE
012054  020027  177777                  CMP     %0,#-1          ;AT UPPER LIMIT
012060  001403                          BEQ     CMP2            ;YES EXIT
012062  005200                          INC     %0              ;INCREMENT TO NEXT NUMBER
012064  005201                          INC     %1
012066  000767                          BR      CMP1
012070  104400                  CMP2:   SCOPE

                                ;TEST ROTATING ALL NUMBERS
012072  012767  177777  000132          MOV     #-1,REFF        ;INITIALIZE BASE NUMBER
012100  005267  000126          TSROT:  INC     REFF            ;INCREMENT NUMBER
012104  004767  000014                  JSR     %7,ROTALL       ;GO TO COMPARE ROUTINE
012110  026727  000116  177777          CMP     REFF,#-1        ;TEST ALL VALUES
012116  001370                          BNE     TSROT           ;NO TEST THEM ALL
012120  104400                          SCOPE
012122  000445                          BR      TSRT2A          ;WE ARE DONE

012124  016767  000102  000102  ROTALL: MOV     REFF,TEST
012132  006067  000076                  ROR     TEST
012136  006067  000072                  ROR     TEST
012142  006067  000066                  ROR     TEST
012146  006167  000062                  ROL     TEST
012152  006167  000056                  ROL     TEST
012156  006167  000052                  ROL     TEST
                                ;TEST COMBINATION OF N, C AND V
012162  100004                          BPL     .+12
012164  103007                          BCC     .+20            ;Z=1
012166  102013                          BVC     .+30            ;Z=1, C=1
012170  104006                          HLT                     ;Z=C, BUT V=1
012172  000411                          BR      .+24
012174  103006                          BCC     .+16            ;Z=0
012176  102407                          BVS     .+20            ;Z=0, C=1
012200  104006                          HLT                     ;Z NOT EQUAL C, V=1
012202  000405                          BR      .+14
```

```
012204  102404                          BVS     .+12                    ;Z=1, C=0
012206  104006                          HLT                             ;Z NOT EQUAL C, V=1
012210  000402                          BR      .+6
012212  102001                          BVC     .+4                     ;Z=0, C=0
012214  104006                          HLT                             ;Z=C, BUT V=1
012216  026767  000012  000006          CMP     TEST,REFF
012224  001401                          BEQ     .+4
012226  104006                          HLT                             ;INITIAL NOT EQUAL TO FINAL
012230  000207                          RTS     %7
012232  000000                  REFF:   0
012234  000000                  TEST:   0
        012232                          REF=REFF

                                ;TEST ROTATING BYTE EVEN/ODD, ALL NUMBERS
012236  012767  177777  177766  TSRT2A: MOV     #-1,REFF
012244  005267  177762          TSROT2: INC     REFF
012250  004767  000016                  JSR     %7,ROTBE
012254  004767  000120                  JSR     %7,ROTBO
012260  022767  177777  177744          CMP     #-1,REFF
012266  001366                          BNE     TSROT2
012270  000503                          BR      ROTEN1
012272  016767  177734  177734  ROTBE:  MOV     REFF,TEST
012300  106067  177730                  RORB    TEST                    ;ROTATE BYTE EVEN
012304  106067  177724                  RORB    TEST
012310  106067  177720                  RORB    TEST
012314  106167  177714                  ROLB    TEST
012320  106167  177710                  ROLB    TEST
012324  106167  177704                  ROLB    TEST
                                ;TEST COMBINATION OF N, C AND V
012330  100004                          BPL     .+12
012332  103007                          BCC     .+20                    ;Z=1
012334  102013                          BVC     .+30                    ;Z=1, C=1
012336  104006                          HLT                             ;Z=C, BUT V=1
012340  000411                          BR      .+24
012342  103006                          BCC     .+16                    ;Z=0
012344  102407                          BVS     .+20                    ;Z=0, C=1
012346  104006                          HLT                             ;Z NOT EQUAL C, V=1
012350  000405                          BR      .+14
012352  102404                          BVS     .+12                    ;Z=1, C=0
012354  104006                          HLT                             ;Z NOT EQUAL C, V=1
012356  000402                          BR      .+6
012360  102001                          BVC     .+4                     ;Z=0, C=0
012362  104006                          HLT                             ;Z=C, BUT V=1
012364  026767  177644  177640          CMP     TEST,REFF
012372  001401                          BEQ     .+4
012374  104006                          HLT
012376  000207                          RTS     %7
012400  106067  177631          ROTBO:  RORB    TEST+1                  ;ROTATE BYTE ODD
012404  106067  177625                  RORB    TEST+1
012410  106067  177621                  RORB    TEST+1
012414  106167  177615                  ROLB    TEST+1
012420  106167  177611                  ROLB    TEST+1
012424  106167  177605                  ROLB    TEST+1
                                ;TEST COMBINATION OF N, C AND V
012430  100004                          BPL     .+12
012432  103007                          BCC     .+20                    ;Z=1
```

```
012434  102013                          BVC     .+30                    ;Z=1, C=1
012436  104006                          HLT                             ;Z=C, BUT V=1
012440  000411                          BR      .+24
012442  103006                          BCC     .+16                    ;Z=0
012444  102407                          BVS     .+20                    ;Z=0, C=1
012446  104006                          HLT                             ;Z NOT EQUAL C, V=1
012450  000405                          BR      .+14
012452  102404                          BVS     .+12                    ;Z=1, C=0
012454  104006                          HLT                             ;Z NOT EQUAL C, V=1
012456  000402                          BR      .+6
012460  102001                          BVC     .+4                     ;Z=0, C=0
012462  104006                          HLT                             ;Z=C, BUT V=1
012464  026767  177544  177540        CMP     TEST,REFF
012472  001401                          BEQ     .+4
012474  104006                          HLT
012476  000207                          RTS     %7
012500  104400                  ROTEN1: SCOPE

                                ;INHIBIT SOME TESTS WHEN SW11 IS SET
012502  032767  004000  166316          BIT     #4000,SREG2
012510  001402                          BEQ     TSTART
012512  000167  000502                  JMP     DONE

                                ;ADD AND SUBTRACT ALL NUMBERS AGAINST FIXED NUMBERS
                                ;A+B=C, C-A=B, BF SHOULD EQUAL BI
012516  011667  000066          TSTART: MOV     %6,NUMA
012522  012767  000001  177502          MOV     #1,REF
012530  005267  177476          ARITST: INC     REF
012534  004767  000012                  JSR     %7,ADSUB
012540  022767  177777  177464          CMP     #-1,REFF
012546  001370                          BNE     ARITST
012550  000420                          BR      ARIEND
012552  016767  177454  177454  ADSUB:  MOV     REF,TEST
012560  066767  000024  177446          ADD     NUMA,TEST
012566  166767  000016  177440          SUB     NUMA,TEST
012574  026767  177432  177432          CMP     REF,TEST
012602  001401                          BEQ     .+4
012604  104006                          HLT
012606  000207                          RTS     %7
012610  000000                  NUMA:   0
012612  104400                  ARIEND: SCOPE

                                ;TEST COMPLIMENTING ALL NUMBERS
012614  005067  000574                  CLR     TEMP                    ;BASE DATA
012620  005067  000574                  CLR     TEMP+4                  ;BASE REFERENCE
012624  005167  000564          TCOM:   COM     TEMP                    ;COMPLIMENT DATA
012630  005367  000564                  DEC     TEMP+4                  ;DECREMENT REFERENCE
012634  026767  000554  000556        CMP     TEMP,TEMP+4             ;COMPARE
012642  001401                          BEQ     .+4                     ;TEST
012644  104006                          HLT                             ;COMPLIMENT OR DECREMENT FAILED
012646  005167  000542                  COM     TEMP
012652  005267  000536                  INC     TEMP                    ;INCREMENT AND TEST FOR DONE
012656  001362                          BNE     TCOM                    ;NOT FINISHED GO LOOP
012660  104400                          SCOPE

                                ;TEST COMB (EVEN BYTE)
```

```
012662  005067  000526                  CLR     TEMP                    ;BASE DATA
012666  005067  000526                  CLR     TEMP+4                  ;REFERENCE DATA
012672  105167  000516          TCOM2:  COMB    TEMP
012676  005367  000516                  DEC     TEMP+4
012702  126767  000506  000510          CMPB    TEMP,TEMP+4             ;COMPARE
012710  001401                          BEQ     .+4
012712  104006                          HLT                             ;COMPLIMENT OR INCREMENT BYTE FAILED
012714  105167  000474                  COMB    TEMP
012720  105267  000470                  INCB    TEMP
012724  001362                          BNE     TCOM2
012726  104400                          SCOPE

                                ;TEST COMB (ODD BYTE)
012730  005067  000460                  CLR     TEMP                    ;BASE DATA
012734  005067  000460                  CLR     TEMP+4                  ;REFERENCE DATA
012740  105167  000451          TCOM3:  COMB    TEMP+1                  ;ODD BYTE
012744  005367  000450                  DEC     TEMP+4
012750  126767  000441  000442          CMPB    TEMP+1,TEMP+4
012756  001401                          BEQ     .+4
012760  104006                          HLT                             ;COMPLIMENT BYTE FAILED
012762  105167  000427                  COMB    TEMP+1
012766  105267  000423                  INCB    TEMP+1
012772  001362                          BNE     TCOM3
012774  104400                          SCOPE

                                ;TEST COMPARE ALL VALUE EVEN BYTE WITH ODD
012776  005067  000412                  CLR     TEMP                    ;BASE VALUE
013002  126767  000406  000405  TSCOMB: CMPB    TEMP,TEMP+1             ;COMPARE
013010  001401                          BEQ     .+4
013012  104006                          HLT                             ;COMPARE FAILED
013014  002001                          BGE     .+4
013016  104006                          HLT                             ;V IS NOT = TO N
013020  003401                          BLE     .+4
013022  104006                          HLT                             ;V IS SET
013024  062767  000401  000362          ADD     #401,TEMP
013032  022767  177777  000354          CMP     #-1,TEMP
013040  001360                          BNE     TSCOMB
013042  104400                          SCOPE

                                ;TEST TO SEE IF I/O DEVICES WERE SELECTED
013044  016767  165756  000026          MOV     SREG2,CKWAIT            ;GET DEVICE INHIBIT FLAGS
013052  005167  000022                  COM     CKWAIT                  ;COMPLEMENT SO DEVICES SELECTED
                                                                        ;CORRESPOND TO ONES
013056  032767  000161  000014          BIT     #161,CKWAIT             ;ANY DEVICES BESIDES TC11 SELECTED?
013064  001406                          BEQ     WAIT4                   ;BRANCH IF NO
013066  000001                          WAIT                            ;INTERRUPTS WILL OCCUR IF DEVICES
013070  000001                          WAIT                            ;BESIDES TC11 ARE SELECTED
013072  000001                          WAIT
013074  000001                          WAIT
013076  000401                          BR      .+4
013100  000000          CKWAIT: 0
013102  104400          WAIT4:  SCOPE

                                ;TEST SWAB
013104  012767  000200  177122          MOV     #0200,TEST
013112  000367  177116                  SWAB    TEST
```

```
013116  100001                          BPL     .+4
013120  104006                          HLT
013122  001401                          BEQ     .+4
013124  104006                          HLT
013126  000367  177102                  SWAB    TEST
013132  100401                          BMI     .+4
013134  104006                          HLT
013136  001001                          BNE     .+4
013140  104006                          HLT
013142  104400                          SCOPE

                                ;TEST ALL COMBINATIONS OF SWAB
013144  005067  177064                  CLR     TEST                    ;NUMBER UNDER TEST
013150  005067  177056                  CLR     REF                     ;REFERENCE NUMBER
013154  000367  177054          SWABA:  SWAB    TEST                    ;OPERATION UNDER TEST
013160  026767  177050  177044          CMP     TEST,REF                ;TEST SWAB INSTRUCTION
013166  001401                          BEQ     .+4
013170  104006                          HLT                             ;SWAB FAILED
013172  000367  177036                  SWAB    TEST
013176  005267  177030                  INC     REF                     ;INCREMENT REFERENCE NUMBER
013202  105267  177027                  INCB    TEST+1                  ;INC TEST NUMBER
013206  001362                          BNE     SWABA                   ;LOOP TILL DONE
013210  104400                          SCOPE
013212  012767  000400  000134          MOV     #400,ICOUNT

                                ;*****************************************************************************
                                ;END OF USER CODE IN BANK
                                ;CALL KERNEL
                                ;ALTERRED IN CORE EXPANSION
013220  104012                  DONE:   EOB
013222  000240                          NOP                             ;TO ALLOW CORE EXPANSION TO PATCH IN JMP

                                ;GROUP OF NESTED SUBROUTINES
013224  000207                  SUBR1:  RTS     %7                      ;ONE INSTRUCTION
013226  000277                  SUBR2:  SCC                             ;ONE DEEP
013230  000207                          RTS     %7
013232  004767  177770          SUBR3:  JSR     %7,SUBR2                ;TWO DEEP
013236  000207                          RTS     %7
013240  004767  177766          SUBR4:  JSR     %7,SUBR3                ;THREE DEEP
013244  000207                          RTS     %7
013246  004767  177766          SUBR5:  JSR     %7,SUBR4                ;FOUR DEEP
013252  000207                          RTS     %7
013254  004767  177766          SUBR6:  JSR     %7,SUBR5                ;FIVE DEEP
013260  000207                          RTS     %7

                                .SBTTL UTILITY ROUTINES

                                ;SCOPE AND/OR ITERATION LOOP FOR EACH TEST
013262  032767  002000  165536  SCOPEC: BIT     #2000,SREG2             ;INHIBIT PROCESSOR TESTS?
013270  001403                          BEQ     .+10                    ;NO
013272  022626                          CMP     (SP)+,(SP)+
013274  000167  167702                  JMP     MAIN                    ;YES
013300  032767  040000  165520          BIT     #40000,SREG2            ;TEST SR FOR SCOPE
013306  001012                          BNE     SCOPEB                  ;YES, SCOPE
013310  032767  004000  165510          BIT     #4000,SREG2             ;NO-TEST FOR ITERATION
```

```
013316  001011                          BNE     SCOPEG                  ;INHIBIT ITERATION
013320  026767  000032  000026          CMP     SCOPEF,ICOUNT           ;COMPARE CURRENT COUNT TO MAX NUMBER
013326  100005                          BPL     SCOPEG                  ;EXIT-DONE
013330  005267  000022                  INC     SCOPEF                  ;INCREMENT COUNT
013334  016716  000020          SCOPEB: MOV     RETURN,@SP
013340  000002                          RTI
013342  005067  000010          SCOPEG: CLR     SCOPEF                  ;CLEAR COUNT
013346  011667  000006                  MOV     @%6,RETURN              ;SAVE SCOPE RETURN POINTER
013352  000002                          RTI                             ;RETURN INLINE-NEXT TEST
013354  000400                  ICOUNT: 400                             ;ITERATION COUNT
013356  000000                  SCOPEF: 0                               ;COUNT LOCATION FOR ITERATION LOOP
013360  000000                  RETURN: 0                               ;ADDRESS OF LAST TEST

                                ;FIXED VALUES FOR USE IN TEST
013362  125252                  B:      125252
013364  013362                          B                               ;ADDRESS OF B
013366  052525                          052525
        013372                          .=B+10
013372  177777                  A:      -1
013374  013376                          A+4
        013376                          .=A+4
013376  125252                          125252
013400  013402                          A+10                            ;ADDRESS OF A+10
013402  052525                          052525

                                ;FOR STORAGE
013404  000000                  C:      0
013406  013404                          C                               ;ADDRESS OF C
        013414                          .=C+10
013414  000000                  TEMP:   0
013416  013414                          TEMP                            ;ADDRESS OF TEMP
        013422                          .=TEMP+6
013422  013424                          TEMP+10                         ;ADDRESS OF TEMP+10 OR "D"
013424  000000                  D:      0

                                ;SUBROUTINE TO INITIALIZE ALL PAGES TO NR, BANK 0, 1 PAGE, UP
013426  010146                  NRALL:  MOV     R1,-(R6)                ;SAVE REGISTERS
013430  010246                          MOV     R2,-(R6)
013432  010346                          MOV     R3,-(R6)
013434  012701  001016                  MOV     #IPDRTAB,R1             ;R1 HOLDS ADDRESS OF CURRENT POSITION
                                                                        ;IN TABLE OF ADDRESSES
013440  012703  000040          NRLOOP: MOV     #32.,R3                 ;R3 USED AS COUNTER
013444  012102                          MOV     (R1)+,R2                ;R2 CONTAINS ADDRESS OF PDR OR
                                                                        ;PAR TO BE CLEARED
013446  005022                          CLR     (R2)+                   ;CLEAR ALL ASR'S FOR THIS MODE
013450  077302                          SOB     R3,.-2
013452  020127  001022                  CMP     R1,#IPDREND             ;CHECK FOR DONE
013456  003770                          BLE     NRLOOP                  ;CLEAR ALL IN NEXT MODE IF NOT DONE
013460  012603                          MOV     (R6)+,R3
013462  012602                          MOV     (R6)+,R2
013464  012601                          MOV     (R6)+,R1
013466  000207                          RTS     %7

                                ;PROGRAMMED INTERRUPT REQUEST SERVICE ROUTINE
013470  117737  165402  177776  PIRSRV: MOVB    @PIR,@#PSR              ;SET PRIORITY
013476  032777  000002  165372          BIT     #2,@PIR                 ;CHECK FOR LEVEL 2 VS LEVEL 3
```

```
013504  001006                          BNE     PIR3                    ;IF SET, IS LEVEL 3
013506  042777  002000  165362          BIC     #2000,@PIR
013514  004767  171740                  JSR     %7,REGEN                ;LEVEL 2-REGENERATE WRITE BUFFER
013520  000002                          RTI
013522  042777  004000  165346  PIR3:   BIC     #4000,@PIR
013530  004767  172640                  JSR     %7,TCCK                 ;LEVEL 3-CHECK DATA READ FROM DTA
013534  000002                          RTI

                                ;EMT HANDLER
                                ;FIRST 3 CALLS LEFT OPEN IN TABLE FOR EASY PATCHES
013536  162716  000002          EMTSRV: SUB     #2,@SP                  ;GET CALL
013542  006576  000000                  MFPI    @(SP)
013546  012667  000022                  MOV     (SP)+,EPC
013552  062716  000002                  ADD     #2,@SP
013556  105067  000013                  CLRB    EPC+1                   ;SAVE OFFSET ONLY
013562  062767  013576  000004          ADD     #EMTAB,EPC              ;POINT TO TABLE OF ADDRESSES
013570  017707  000000                  MOV     @EPC,PC                 ;JUMP TO DESIRED ROUTINE
013574  000000                  EPC:    0
        000000                          PATCH1=0
        000000                          PATCH2=0
        000000                          PATCH3=0
013576  000000                  EMTAB:  PATCH1                          ;PATCH IN ADDRESS OF ROUTINE
013600  000000                          PATCH2
013602  000000                          PATCH3
013604  016262                          PRINT                           ;ERROR PRINTOUT
013606  013612                          RTIKEX                          ;RTI VIA KERNEL
013610  013700                          EOBSRV                          ;END OF BANK

013612  022626                  RTIKEX: CMP     (SP)+,(SP)+             ;FIX KERNEL STACK POINTER
013614  032737  030000  177776          BIT     #30000,@#PSR            ;PREVIOUS KERNEL?
013622  001001                          BNE     .+4                     ;BRANCH IF NOT
013624  000002                          RTI                             ;RETURN IF FROM KERNEL
013626  106506                          MFPD    SP                      ;GET PREVIOUS STACK POINTER
013630  012667  000042                  MOV     (SP)+,SPSAV             ;RESTORE KERNEL STACK
013634  062767  000002  000034          ADD     #2,SPSAV
013642  106577  000030                  MFPD    @SPSAV                  ;PUSH PS ON KERNEL STACK
013646  162767  000002  000022          SUB     #2,SPSAV
013654  106577  000016                  MFPD    @SPSAV                  ;PUSH PC ON KERNEL STACK
013660  062767  000004  000010          ADD     #4,SPSAV
013666  016746  000004                  MOV     SPSAV,-(SP)
013672  106606                          MTPD    SP                      ;RESTORE CORRECTED PREVIOUS STACK
                                                                        ;POINTER
013674  000002                          RTI                             ;RETURN
013676  000000                  SPSAV:  0

                                ;END OF BANK SERVICE- DOES ALL MAPPING FUNCTIONS AFTER STARTUP
                                ;CALLED AT THE END OF EACH BACKGROUND CPU PASS THRU 4K (VIRTUAL) UNLESS
                                ;CORE EXPANDED (IN WHICH CASE IT IS CALLED AFTER EACH PASS THRU ALL MEMORY
                                ;TO 28K)
013700  113737  177571  001027  EOBSRV: MOVB    @#SR+1,@#SREG2+1        ;READ SWITCHES AGAIN
013706  032737  000001  001024          BIT     #1,@#SREG1              ;KT11-C INHIBITED?
013714  001503                          BEQ     EOB2                    ;NO - CONTINUE
013716  004767  003026          EOB1:   JSR     %7,BELL                 ;YES-SIGNAL END OF PASS
013722  042766  000020  000002          BIC     #20,2(SP)               ;CLEAR TRACE BIT OF STATUS ON STACK
013730  012737  000016  000014          MOV     #16,@#14                ;SETUP TRACE RETURN TO CAUSE HALT
013736  005037  000016                  CLR     @#16                    ;IF A TRACE TRAP OCCURS
```

```
013742  032737  010000  001026          BIT     #10000,@#SREG2          ;INHIBIT TRACE TRAPPING?
013750  001011                          BNE     E0B1A                   ;YES - BRANCH
013752  005167  165112                  COM     IRPB                    ;SWITCH TRACE FLAG
013756  100006                          BPL     E0B1A                   ;IF NOT SET, LEAVE TRACE OFF
013760  052766  000020  000002          BIS     #20,2(SP)               ;IF SET, SET TRACE BIT OF STATUS ON STACK
013766  012737  016260  000014          MOV     #TRTRP,@#14
013774  105267  165104          E0B1A:  INCB    DISPLY
014000  016737  165100  177570          MOV     DISPLY,@#SR
014006  032737  000040  001024          BIT     #40,@#SREG1             ;CORE EXPANSION INHIBITED?
014014  001040                          BNE     E0B1C                   ;YES, BRANCH OVER
014016  013746  000006                  MOV     @#6,-(SP)               ;NO, COPY NEW DYNAMIC SWITCHES TO
014022  013746  000004                  MOV     @#4,-(SP)               ;ALL BANKS UP TO 28K
014026  012737  014076  000004          MOV     #E0B1B,@#4
014034  005037  000006                  CLR     @#6
014040  013701  001026                  MOV     @#SREG2,R1
014044  010137  021026                  MOV     R1,@#SREG2+20000
014050  010137  041026                  MOV     R1,@#SREG2+40000
014054  010137  061026                  MOV     R1,@#SREG2+60000
014060  010137  101026                  MOV     R1,@#SREG2+100000
014064  010137  121026                  MOV     R1,@#SREG2+120000
014070  010137  141026                  MOV     R1,@#SREG2+140000
014074  000401                          BR      .+4
014076  022626          E0B1B:  CMP     (SP)+,(SP)+
014100  012637  000004                  MOV     (SP)+,@#4
014104  012637  000006                  MOV     (SP)+,@#6
014110  012716  006466                  MOV     #BEGINX,(SP)            ;SETUP PC TO RETURN TO BEGINX
014114  000002                          RTI
014116  012716  006700          E0B1C:  MOV     #BEGIN,(SP)             ;IF NO CORE EXPANSION, RETURN TO
014122  000002                          RTI                             ;BEGIN
014124  032737  000002  001024  E0B2:   BIT     #2,@#SREG1              ;USER/SUPERVISOR/KERNEL INHIBITED?
014132  001271                          BNE     E0B1                    ;YES - SET PC AND RETURN
014134  032737  001000  001026          BIT     #1000,@#SREG2           ;STOP CYCLING SUPERVISOR PAGES?
014142  001132                          BNE     SUPD1                   ;YES- BRANCH
014144  023737  001064  000734          CMP     @#CSIPDR,@#SIPDR6       ;NO-LAST SUPERVISOR PAGE DONE (PAGE 6)?
014152  001455                          BEQ     SUPCYC                  ;YES, BRANCH TO RECYCLE
014154  016701  164704                  MOV     CSIPDR,R1
014160  005721                          TST     (R1)+                   ;ADDRESS NEXT PDR
014162  012711  077406                  MOV     #77406,@R1
014166  012761  077406  000020          MOV     #77406,20(R1)
014174  062777  020000  164420          ADD     #20000,@TTPVC           ;UPDATE SUPERVISOR I/O VECTORS
                                                                        ;TO REFERENCE NEXT PAGE
014202  062777  020000  164402          ADD     #20000,@TTRVC
014210  062777  020000  164414          ADD     #20000,@KWLVC
014216  032737  000040  001026          BIT     #40,@#SREG2             ;UPDATE LP11 VECTOR ONLY IF USED
014224  001003                          BNE     .+10
014226  062777  020000  164406          ADD     #20000,@LPVC
014234  013737  177776  001072          MOV     @#PSR,@#PSSAV           ;UPDATE SUPERVISOR STACK POINTER
014242  042737  030000  177776          BIC     #30000,@#PSR
014250  052737  010000  177776          BIS     #10000,@#PSR
014256  106506                          MFPD    SP
014260  062716  020000                  ADD     #20000,(SP)
014264  106606                          MTPD    SP
014266  013737  001072  177776          MOV     @#PSSAV,@#PSR
014274  005061  177776                  CLR     -2(R1)                  ;MAKE PREVIOUS PAGE NON-RESIDENT
014300  005061  000016                  CLR     16(R1)
014304  000447                          BR      SUPDON
```

```
014306  016701  164416          SUPCYC: MOV     SIPDR0,R1               ;MAP TO FIRST SUPERVISOR PAGE
014312  012711  077406                  MOV     #77406,@R1
014316  012761  077406  000020          MOV     #77406,20(R1)
014324  042777  160000  164270          BIC     #160000,@TTPVC
014332  042777  160000  164252          BIC     #160000,@TTRVC
014340  042777  160000  164264          BIC     #160000,@KWLVC
014346  042777  160000  164266          BIC     #160000,@LPVC
014354  013737  177776  001072          MOV     @#PSR,@#PSSAV           ;MAP SUPERVISOR STACK TO PAGE 0
014362  042737  030000  177776          BIC     #30000,@#PSR
014370  052737  010000  177776          BIS     #10000,@#PSR
014376  106506                        MFPD    SP
014400  042716  160000                BIC     #160000,(SP)
014404  106606                        MTPD    SP
014406  013737  001072  177776          MOV     @#PSSAV,@#PSR
014414  005061  000014                CLR     14(R1)                  ;MAP SUPERVISOR PAGE 6 NON-RESIDENT
014420  005061  000034                CLR     34(R1)
014424  010137  001064          SUPDON: MOV     R1,@#CSIPDR
014430  042737  000340  177776  SUPD1:  BIC     #340,@#PSR
014436  032737  000420  001024          BIT     #420,@#SREG1            ;INHIBIT CYCLING ACCESS KEY OF S7?
014444  001010                  BNE     SUPD2                   ;YES, SKIP
014446  005277  164272                INC     @SDPDR7                 ;NO, SETUP NEXT KEY IN SUPERVISOR
014452  042777  000010  164264          BIC     #10,@SDPDR7             ;PAGE 7
014460  017777  164260  164254          MOV     @SDPDR7,@SIPDR7
014466  032737  000004  001024  SUPD2:  BIT     #4,@#SREG1              ;INHIBIT 4K AS 32K?
014474  001402                  BEQ     .+6
014476  000167  177214                JMP     E081                    ;YES - SET PC AND RETURN
014502  023737  001054  000720          CMP     @#CURIPAR,@#UIPAR7      ;LAST USER ASR DONE?
014510  001460                  BEQ     NXTBNK                  ;YES - GO FIND NEXT BANK
014512  062737  020000  000034          ADD     #20000,@#34             ;UPDATE SCOPE VECTOR ADDRESS IN BANK 0
014520  062737  020000  001066          ADD     #20000,@#BNKSTR         ;UPDATE BANK START TO REFERENCE CURRENT ASR
014526  013716  001066                MOV     @#BNKSTR,(SP)
014532  005077  164316                CLR     @CURIPAR                ;SET PREVIOUS PAGE NR, BANK 0
014536  005077  164314                CLR     @CURDPAR
014542  005077  164312                CLR     @CURIPDR
014546  005077  164310                CLR     @CURDPDR
014552  062767  000002  164274  NXTSEG: ADD     #2,CURIPAR              ;UPDATE POINTERS TO NEXT PAGE
014560  062767  000002  164270          ADD     #2,CURDPAR
014566  062767  000002  164264          ADD     #2,CURIPDR
014574  062767  000002  164260          ADD     #2,CURDPDR
014602  012777  077406  164250          MOV     #77406,@CURIPDR         ;SET NEXT PAGE RW, 4K
014610  012777  077406  164244          MOV     #77406,@CURDPDR
014616  016777  164226  164230          MOV     CURBNK,@CURIPAR         ;MAP NEXT PAGE TO CURRENT BANK
014624  016777  164220  164224          MOV     CURBNK,@CURDPAR
014632  052737  030000  177776          BIS     #30000,@#PSR            ;SET PREVIOUS MODE TO USER
014640  106506                        MFPD    R6                      ;PICK UP USER STACK POINTER
014642  062716  020000                ADD     #20000,@R6              ;MAP IT TO NEXT ASR
014646  106606                        MTPD    R6                      ;PUT IT BACK
014650  000002                        RTI                             ;GO BACK TO MAINLINE
014652  005327  000000          NXTBNK: DEC     #0                      ;STALL SO DOUBLE BELL WILL BE
014656  001375                  BNE     .-4                     ;NOTED
014660  004767  002064                JSR     %7,BELL
014664  012746  000400          NXBNK1: MOV     #UBUFF,-(SP)            ;MAP USER STACK TO PAGE 0
014670  052737  030000  177776          BIS     #30000,@#PSR
014676  106606                        MTPD    R6
014700  042737  160000  000034          BIC     #160000,@#34            ;INITIALIZE SCOPE VECTOR ADDRESS
014706  013746  000004                MOV     @#4,-(SP)               ;STORE ADDRESS ERROR RETURN
```

```
014712  013746  000006                  MOV     @#6,-(SP)
014716  012737  015646  000004          MOV     #CORCK,@#4              ;SETUP TRAP RETURN
014724  005037  000006                  CLR     @#6
014730  013737  001050  001052          MOV     @#CURBNK,@#OLDBNK       ;SAVE ADDRESS OF PREVIOUS BANK
014736  062737  000200  001050  BNKTST: ADD     #200,@#CURBNK
014744  022737  007600  001050          CMP     #7600,@#CURBNK          ;CHECK FOR EXTERNAL BANK
014752  001067                          BNE     E0B3                    ;IF NOT, TEST FOR ITS PRESENCE
014754  012737  000000  001050          MOV     #0,@#CURBNK             ;START OVER, TESTING BANK 0
014762  105237  001104                  INCB    @#DISPLY
014766  013737  001104  177570          MOV     @#DISPLY,@#SR
014774  013701  000042          LOGIC:  MOV     @#42,R1
015000  001413                          BEQ     BNKT
015002  000005                          RESET
015004  005046                          CLR     -(SP)                   ;CLEAR T-BIT VIA RTI
015006  012746  015014                  MOV     #LOGICAL,-(SP)
015012  000002                          RTI
015014  004711                  LOGICAL:        JSR     %7,@R1
015016  000240                          NOP
015020  000240                          NOP
015022  000240                          NOP
015024  000137  001250                  JMP     @#RSTRT
015030  032737  000001  001026  BNKT:   BIT     #1,@#SREG2              ;TTY OUTPUT SELECTED?
015036  001410                          BEQ     BNKT1                   ;YES, SKIP OUTPUTTING ASTERISK
015040  004767  001734                  JSR     %7,CRLF
015044  105777  163546                  TSTB    @TCSR                   ;WAIT FOR TELETYPE
015050  100375                          BPL     .-4
015052  012777  000252  163540          MOV     #252,@TDBR              ;OUTPUT ASTERISK TO SIGNAL END OF PASS
015060  042766  000020  000006  BNKT1:  BIC     #20,6(SP)               ;CLEAR TRACE BIT OF STATUS ON STACK
015066  012737  000016  000014          MOV     #16,@#14
015074  005037  000016                  CLR     @#16
015100  032737  010000  177570          BIT     #10000,@#SR
015106  001011                          BNE     E0B3
015110  005167  163754                  COM     TRPB
015114  100006                          BPL     E0B3
015116  052766  000020  000006          BIS     #20,6(SP)
015124  012737  016260  000014          MOV     #TRTRP,@#14
015132  016777  163712  163636  E0B3:   MOV     CURBNK,@KIPAR2          ;MAP KERNEL PAGE 2 TO BANK BEING LOOKED FOR
015140  016777  163704  163642          MOV     CURBNK,@KDPAR2
015146  012777  077406  163576          MOV     #77406,@KIPDR2
015154  012777  077406  163602          MOV     #77406,@KDPDR2
015162  013737  041000  041000          MOV     @#41000,@#41000
015170  013737  045000  045000          MOV     @#45000,@#45000
015176  013737  051000  051000          MOV     @#51000,@#51000
015204  013737  055000  055000          MOV     @#55000,@#55000
015212  012637  000006                  MOV     (SP)+,@#6               ;RESTORE ADDRESS ERROR RETURN
015216  012637  000004                  MOV     (SP)+,@#4
015222  005001                          CLR     R1                      ;R1 ADDRESSES BANK 0 THRU KERNEL ASR0
015224  012702  040000                  MOV     #40000,R2               ;R2 ADDRESSES NEW BANK THRU KERNEL ASR2
015230  012703  013424                  MOV     #D,R3                   ;R3 IS USED TO CHECK FOR END OF TRANSFER
015234  006203                          ASR     R3
015236  012122          CORMOV: MOV     (R1)+,(R2)+             ;LOAD NEW BANK
015240  077302                  SOB     R3,CORMOV
015242  016767  163446  163604  E0B4:   MOV     UIPAR0,CURIPAR          ;FIRST ASR CHECKED IS USER ASR0
015250  016767  163446  163600          MOV     UDPAR0,CURDPAR
015256  016767  163416  163574          MOV     UIPDR0,CURIPDR
015264  016767  163416  163570          MOV     UDPDR0,CURDPDR
```

```
015272  016777  163552  163554          MOV     CURBNK,@CURIPAR
015300  016777  163544  163550          MOV     CURBNK,@CURDPAR
015306  012777  077406  163544          MOV     #77406,@CURIPDR
015314  012777  077406  163540          MOV     #77406,@CURDPDR
015322  005077  163372                  CLR     @UIPAR7
015326  005077  163374                  CLR     @UDPAR7
015332  005077  163346                  CLR     @UIPDR7
015336  005077  163350                  CLR     @UDPDR7
015342  032737  000010  001024          BIT     #10,@#SREG1     ;CHECK FOR INHIBITING I/O RELOCATION
015350  001111                          BNE     E085            ;SKIP IF SET
015352  005737  004334                  TST     @#TCREL         ;TC11 RELOCATION ALLOWED?
015356  001414                          BEQ     E084A           ;NO- BRANCH
015360  005737  004332                  TST     @#TCEOP         ;YES- WAIT FOR TC11 TO SET
015364  001775                          BEQ     .-4             ;END OF PASS FLAG
015366  042777  000100  163436          BIC     #100,@TCCM      ;CLEAR TC11 IE
015374  005777  163432                  TST     @TCCM           ;WAIT FOR END ZONE
015400  100375                          BPL     .-4
015402  005737  004326                  TST     @#DATACK        ;WAIT FOR DATA CHECK TO FINISH
015406  100775                          BMI     .-4
015410  052737  000340  177776  E084A:  BIS     #340,@#PSR      ;SET PRIORITY TO LEVEL 7
015416  016777  163426  163350          MOV     CURBNK,@KIPAR1  ;MAP KERNEL PAGE 1 TO NEW BANK
015424  016777  163420  163354          MOV     CURBNK,@KDPAR1
015432  016700  163412                  MOV     CURBNK,R0       ;CALCULATE NEW PHYSICAL ADDRESS
015436  010001                          MOV     R0,R1
015440  072027  000006                  ASH     #6,R0           ;LEFT SHIFT 6 PLACES - LEAVES LOWER
015444  042700  017777                  BIC     #17777,R0       ;16 BITS OF THE BANK ADDRESS
015450  072127  177772                  ASH     #-6,R1          ;RIGHT SHIFT 6 PLACES - SETS UP TOP 2
015454  042701  177717                  BIC     #177717,R1      ;BITS OF ADDRESS IN BITS 4 AND 5
015460  042737  160000  004276          BIC     #160000,@#LLIMIT ;CLEAR BITS 13-15 OF RF11 BUFFER ADDRESS
015466  050037  004276                  BIS     R0,@#LLIMIT     ;MAP RF11 BUFFER ADDRESS TO
                                                                ;CURRENT BANK
015472  042737  030060  004274          BIC     #30060,@#FUNCTION ;SET UP EXTENDED ADDRESS BITS
015500  150137  004274                  BISB    R1,@#FUNCTION
015504  150137  004275                  BISB    R1,@#FUNCTION+1
015510  032737  000100  001026          BIT     #100,@#SREG2    ;RF11 INHIBITED?
015516  001003                          BNE     .+10            ;YES, SKIP NEXT
015520  105777  163134                  TSTB    @RFCSR          ;WAIT FOR RF11 READY
015524  100375                          BPL     .-4
015526  042737  000340  177776          BIC     #340,@#PSR      ;DROP PRIORITY TO LEVEL 0
015534  005737  004334                  TST     @#TCREL         ;TC11 RELOCATION ALLOWED?
015540  001415                          BEQ     E085            ;NO- SKIP TC RELOCATION
015542  042737  160000  004302          BIC     #160000,@#TCBUFP ;YES - RELOCATE TC BUFFER PHYSICAL
015550  050037  004302                  BIS     R0,@#TCBUFP     ;ADDRESS
015554  042737  000060  004330          BIC     #60,@#TCEMA     ;SET UP EXTENDED MEMORY BITS
015562  050137  004330                  BIS     R1,@#TCEMA
015566  052777  002000  163302          BIS     #2000,@PIR      ;BOOK PIRQ LEVEL 2 TO REGENERATE
                                                                ;WRITE BUFFER AND STARTUP TC11
015574  026727  163252  000000  E085:   CMP     OLDBNK,#0       ;WAS PREVIOUS BANK 0?
015602  001414                          BEQ     E086            ;YES - DON'T CLEAR IT
015604  016777  163242  163176          MOV     OLDBNK,@KDPAR2  ;NO - CLEAR IT TO CATCH RELOCATION
015612  012777  077406  163144          MOV     #77406,@KDPDR2  ;ERRORS
015620  012701  040000                  MOV     #40000,R1
015624  012703  007630                  MOV     #7630,R3
015630  005021                  BNKLP:  CLR     (R1)+
015632  077302                          SOB     R3,BNKLP
015634  012716  006700          E086:   MOV     #BEGIN,(SP)     ;SETUP USER PC ON THIS STACK
```

```
015640  011667  163222                  MOV     (SP),BNKSTR             ;STORE IT ALSO
015644  000002                          RTI                             ;RETURN TO USER
015646  022626                  CORCK:  CMP     (SP)+,(SP)+             ;RESTORE STACK POINTER
015650  000167  177062                  JMP     BNKTST                  ;TRY AGAIN

                                ;KT11-C TRAP AND ABORT SERVICE ROUTINE
015654  000240                  ABSRV:  NOP
015656  032777  010000  163004          BIT     #10000,@SR0             ;MEMORY MANAGEMENT TRAP?
015664  001404                          BEQ     .+12                    ;NO, CONTINUE
015666  042777  010000  162774          BIC     #10000,@SR0             ;YES, CLEAR TRAP FLAG AND RETURN
015674  000002                          RTI
015676  122777  000077  162764          CMPB    #77,@SR0                ;ONLY AN ABORT FROM A SUPERVISOR D-SPACE
015704  001404                          BEQ     .+12                    ;PAGE 7 REFERENCE IS ALLOWED
015706  000000                          HALT                            ;ABORT FROM A PAGE OTHER THAN
015710  000005                          RESET                           ;SUPERVISOR D-SPACE 7 - FATAL ERROR
015712  000167  163332                  JMP     RSTRT                   ;RESTART AFTER FATAL ERROR
015716  017716  162750                  MOV     @SR1,(SP)               ;GET SR1
015722  001454                          BEQ     REEX                    ;GO FIX PC IF NO REGISTERS WERE AFFECTED
015724  012767  000002  000262          MOV     #2,ALLREG
015732  042716  177770          LSTSET: BIC     #177770,(SP)            ;WAS PC AFFECTED?
015736  022716  000007                  CMP     #7,(SP)
015742  001436                          BEQ     NXTSET                  ;YES - BRANCH
015744  022716  000006                  CMP     #6,(SP)                 ;NO - WAS IT THE STACK POINTER?
015750  001016                          BNE     GENR                    ;NO - GO CHANGE REGISTERS
015752  032737  030000  177776          BIT     #30000,@#PSR            ;YES - WAS IT THE KERNEL STACK?
015760  001002                          BNE     .+6
015762  104006                          HLT                             ;CANNOT RESTORE KERNEL STACK POINTER
015764  000433                          BR      REEX                    ;EXIT
015766  000240                          NOP
015770  106506                          MFPD    SP                      ;GET STACK POINTER
015772  004767  000220                  JSR     %7,SHIFT
015776  166616  177774                  SUB     -4(SP),(SP)             ;RESTORE SP TO ORIGINAL VALUE
016002  106606                          MTPD    SP
016004  000415                          BR      NXTSET                  ;GO CHECK NEXT SET
016006  000240                  GENR:   NOP
016010  142767  000007  000016          BICB    #7,RSREG
016016  051667  000012                  BIS     (SP),RSREG
016022  004767  000170                  JSR     %7,SHIFT
016026  016667  177774  000002          MOV     -4(SP),RSREG+2
016034  162700                  RSREG:  SUB     (7)+,%0                 ;THIS INSTRUCTION RESTORES THE REGISTER
016036  000000                          .WORD   0
016040  017716  162626          NXTSET: MOV     @SR1,(SP)
016044  000316                          SWAB    (SP)
016046  005367  000142                  DEC     ALLREG
016052  001327                          BNE     LSTSET
016054  017716  162614          REEX:   MOV     @SR2,(SP)               ;SET PC ON STACK TO VA OF INSTRUCTION
016060  013746  000014                  MOV     @#14,-(SP)              ;PUSH CONTENTS OF TBIT VECTOR
016064  013746  000016                  MOV     @#16,-(SP)
016070  016646  000006                  MOV     6(SP),-(SP)             ;PUSH COPY OF ABORT STATUS
016074  016646  000006                  MOV     6(SP),-(SP)             ;PUSH COPY OF VA
016100  042716  160000                  BIC     #160000,(SP)            ;CHANGE TO BANK 0 PC ON STACK
016104  042766  140000  000002          BIC     #140000,2(SP)           ;KERNEL
016112  052766  000020  000002          BIS     #20,2(SP)               ;SET T BIT
016120  012737  016144  000014          MOV     #TSEG,@#14              ;CHANGE T-BIT VECTOR ADDRESS
016126  012737  000340  000016          MOV     #340,@#16
016134  042777  176000  162526          BIC     #176000,@SR0            ;CLEAR ERROR BITS IN SR0
```

```
016142  000006                          RTT
016144  042766  007737  000012  TSEG:   BIC     #7737,12(SP)            ;RESTORE STATUS
016152  042766  170020  000002          BIC     #170020,2(SP)
016160  056666  000002  000012          BIS     2(SP),12(SP)
016166  042766  017777  000010          BIC     #17777,10(SP)           ;FIX RETURN PC
016174  051666  000010                  BIS     (SP),10(SP)
016200  022626                          CMP     (SP)+,(SP)+
016202  012637  000016                  MOV     (SP)+,@#16
016206  012637  000014                  MOV     (SP)+,@#14
016212  000002                          RTI                             ;RETURN
016214  000000          ALLREG: 0
016216  000240          SHIFT:  NOP
016220  017746  162446                  MOV     @SR1,-(SP)
016224  022767  000001  177762          CMP     #1,ALLREG
016232  001001                          BNE     .+4
016234  000316                          SWAB    (SP)
016236  105066  000001                  CLRB    1(SP)
016242  006716                          SXT     (SP)
016244  042716  000007                  BIC     #7,(SP)
016250  006216                          ASR     (SP)
016252  006216                          ASR     (SP)
016254  006226                          ASR     (SP)+
016256  000207                          RTS     %7

                        ;RTT EXECUTED WHEN TRACE IS ON
016260  000006          TRTRP:  RTT

                        ;ENTERED WITH SYSTEM TRAP CALL (HLT)
                        ;PRINT OUT THE ERROR PC+2, STATUS REGISTER,
                        ;LOCATION IN BACKGROUND (TOP 12 BITS OF 18 BIT ADDRESS), AND
                        ;CONTENTS OF RETURN IN CURRENT BANK
016262  005767  000260  PRINT:  TST     PRTON                   ;CHECK PRINT ON FLAG
016266  001401                  BEQ     .+4
016270  000002                  RTI                             ;IF ANOTHER HALT IS BEING PRINTED, SKIP THIS ONE
016272  005267  000250          INC     PRTON
016276  012767  000340  161472  MOV     #340,PSR                ;SET PRIORITY TO 7
016304  036727  161260  020000  BIT     SR,#20000               ;TEST FOR INHIBIT PRINT OUT
016312  001401                  BEQ     .+4                     ;BRANCH TO PRINT
016314  000500                  BR      CK                      ;INHIBIT, CHECK FOR HALT
016316  012667  000220          MOV     (6)+,SAVPC              ;PC OF FAILING ROUTINE
016322  012667  000216          MOV     (6)+,SAVPSR             ;PSR OF ERROR CONDITION
016326  024646                  CMP     -(6),-(6)               ;RESTORE STACK
016330  012767  000200  161440  MOV     #200,PSR
016336  004767  000436          JSR     %7,CRLF                 ;OUTPUT CARRIAGE RETURN AND LINE FEED
016342  016767  000174  000356  MOV     SAVPC,PTEMP1            ;LOAD WITH FAILING PC+2
016350  004767  000174          JSR     %7,PROCT                ;PRINT FAILING PC+2
016354  004767  000352          JSR     %7,SPACE
016360  016767  000160  000340  MOV     SAVPSR,PTEMP1           ;LOAD PROCESSOR STATUS
016366  004767  000156          JSR     %7,PROCT                ;PRINT PROCESSOR STATUS
016372  004767  000334          JSR     %7,SPACE
016376  016767  162446  000322  MOV     CURBNK,PTEMP1
016404  004767  000140          JSR     %7,PROCT
016410  004767  000316          JSR     %7,SPACE
016414  012737  000340  177776  MOV     #340,@#PSR
016422  017746  162360          MOV     @KDPAR1,-(SP)
016426  017746  162330          MOV     @KDPDR1,-(SP)
```

```
016432 016777 162412 162346          MOV     CURBNK,@KDPAR1
016440 012777 077406 162314          MOV     #77406,@KDPDR1
016446 013767 033360 000252          MOV     @#RETURN+20000,PTEMP1   ;GET CONTENTS OF CURRENT RETURN
016454 012677 162302                 MOV     (SP)+,@KDPDR1
016460 012677 162322                 MOV     (SP)+,@KDPAR1
016464 012737 000200 177776          MOV     #200,@#PSR
016472 004767 000052                 JSR     %7,PROCT
016476 004767 000230                 JSR     %7,SPACE
016502 032737 000001 001026          BIT     #1,@#SREG2              ;TTY OUTPUT ALLOWED?
016510 001002                        BNE     .+6                     ;NO, SKIP
016512 004767 000262                 JSR     %7,CRLF                 ;YES- OUTPUT CR,LF
016516 005767 161046          CK:    TST     SR                      ;CHECK SR FOR HALT SWITCH
016522 100001                        BPL     .+4                     ;BRANCH IF NOT SET
016524 000000                        HALT                            ;HALT ON ERROR UP
016526 005067 000014                 CLR     PRTON                   ;ROUTINE DONE - CLEAR FLAG
016532 000002                        RTI                             ;RETURN TO MAIN LINE
016534 000000          SAVR2:  0
016536 000000          SAVR3:  0
016540 000000          SAVR4:  0
016542 000000          SAVPC:  0
016544 000000          SAVPSR: 0
016546 000000          PRTON:  0

016550 012727 000006 016554  PROCT:  MOV     #6,#PTEMP3              ;CLEAR R4 FOR COUNTING CHARACTERS OUTPUT
       016554                        PTEMP3=.-2
016556 005067 000142                 CLR     PRFLG                   ;INITIALIZE CARRY FLAG FOR ROTATES
016562 012767 000260 000140          MOV     #260,PTEMP2             ;SETUP R3
016570 005767 000132                 TST     PTEMP1                  ;CHECK BIT 15 OF NUMBER
016574 100002                        BPL     .+6                     ;BRANCH IF ZERO
016576 005267 000126                 INC     PTEMP2                  ;INCREMENT R3 IF ONE
016602 006167 000120                 ROL     PTEMP1                  ;ROTATE LEFT MOST OCTAL TO RIGHT END
016606 006167 000114                 ROL     PTEMP1
016612 005567 000106                 ADC     PRFLG                   ;STORE CARRY
016616 105777 161774          P.WAIT: TSTB    @TCSR                   ;WAIT FOR TTY READY
016622 100375                        BPL     P.WAIT
016624 016777 000100 161766          MOV     PTEMP2,@TDBR            ;OUTPUT NEXT CHARACTER
016632 005367 177716          P.CONT: DEC     PTEMP3                  ;COUNT
016636 001001                        BNE     P.CNT1                  ;BRANCH IF NOT DONE
016640 000207                        RTS     %7                      ;BRANCH IF NOT DONE
016642 000241                 P.CNT1: CLC                             ;CLEAR CARRY
016644 005767 000054                 TST     PRFLG                   ;CHECK FOR PREVIOUS CARRY
016650 001403                        BEQ     .+10                    ;BRANCH IF PREVIOUSLY ZERO
016652 005067 000046                 CLR     PRFLG                   ;INITIALIZE FLAG
016656 000261                        SEC                             ;SET CARRY
016660 006167 000042                 ROL     PTEMP1                  ;ROTATE NEXT CHARACTER INTO RIGHT END OF REGISTE
016664 006167 000036                 ROL     PTEMP1
016670 006167 000032                 ROL     PTEMP1
016674 005567 000024                 ADC     PRFLG                   ;STORE CARRY
016700 016767 000022 000022          MOV     PTEMP1,PTEMP2           ;LOAD DATA INTO R3
016706 042767 177770 000014          BIC     #177770,PTEMP2          ;CLEAR ALL BUT LOWEST OCTAL DIGIT
016714 052767 000260 000006          BIS     #260,PTEMP2             ;SET TO ASCII EQUIVALENT
016722 000735                        BR      P.WAIT                  ;LOOP
016724 000000          PRFLG:  0
016726 000000          PTEMP1: 0                                     ;CONTAINS VALUE TO BE OUTPUT
016730 000000          PTEMP2: 0                                     ;SCRATCH
```

```
                                        ;ISSUE SPACE
016732  105777  161660        SPACE:  TSTB    @TCSR           ;WAIT FOR TTY READY
016736  100375                BPL     .-4
016740  012777  000240  161652        MOV     #240,@TDBR      ;OUTPUT A SPACE
016746  000207                RTS     %7

                                        ;BELL ON PASS COMPLETE
016750  032737  000001  001026  BELL:   BIT     #1,@#SREG2      ;TTY OUTPUT SELECTED?
016756  001001                BNE     .+4                     ;NO, RING BELL
016760  000207                RTS     %7                      ;YES, SKIP
016762  105777  161630        TSTB    @TCSR
016766  100375                BPL     .-4
016770  012777  000244  161622        MOV     #244,@TDBR
016776  000207                RTS     %7

                                        ;SUBROUTINE TO OUTPUT CARRIAGE RETURN AND LINEFEED
017000  105777  161612        CRLF:   TSTB    @TCSR           ;WAIT FOR TTY READY
017004  100375                BPL     .-4
017006  012777  000215  161604        MOV     #215,@TDBR      ;OUTPUT CARRIAGE RETURN
017014  105777  161576        TSTB    @TCSR                   ;WAIT FOR TTY READY
017020  100375                BPL     .-4
017022  012777  000212  161570        MOV     #212,@TDBR      ;OUTPUT LINEFEED
017030  000207                RTS     %7                      ;RETURN

                                        ;ENTER HERE ON POWER FAIL
017032  013746  000024        PFAIL:  MOV     @#24,-(6)       ;SAVE CONTENTS OF POWER FAIL VECTOR
017036  010667  000030                MOV     %6,SAVR6        ;STORE STACK POSITION
017042  012737  017052  000024        MOV     #RESTRT,@#24    ;LOAD VECTOR WITH POWER UP RETURN ADDRESS
017050  000000                HALT                            ;HALT ON POWER DOWN
017052  016706  000014  RESTRT: MOV     SAVR6,%6        ;RESTORE STACK WHEN POWERING UP
017056  012637  000024                MOV     (6)+,@#24       ;RESTORE ORIGINAL CONTENTS OF POWER FAIL VECTOR
017062  022626                CMP     (SP)+,(SP)+             ;RESTORE THE STACK
017064  104006                HLT                             ;POWER FAIL OCCURRED
017066  000167  162156        JMP     RSTRT
017072  000000        SAVR6:  0

017074  000207        USER:   RTS     %7              ;OVERLAY USER ROUTINE HERE
        017760                .=17760
017760  000000        KSTACK: 0
        000001                .END
```

```
ENDZ2   005434          1420    1424#
E08   = 104012           754#   1189    2463
E08SRV  013700          2571    2594#
E081    013716          2597#   2631    2682
E081A   013774          2602    2604    2607#
E081B   014076          2613    2623#
E081C   014116          2610    2628#
E082    014124          2596    2630#
E083    015132          2719    2744    2746    2749#
E084    015242          2765#
E084A   015410          2780    2788#
E085    015574          2778    2809    2816#
E086    015634          2817    2824#
EPC     013574          2557*   2559*   2560*   2561    2562#
ERROK   004206          1332    1340#
ERRSET  004202          1334    1338#
FBLK    004320          1394#   1418*   1439*   1471    1475*
FUNCTI  004274          1136*   1140    1336*   1348*   1352    1354#   2800*   2801*   2802*
GENR    016006          2848    2859#
HLT   = 104006           752#    918     927    1272    1283    1298    1329    1335    1338    1342    1465    1469    1504
                        1511    1525    1532    1535    1573    1577    1651    1658    1661    1666    1669    1677    1683
                        1692    1699    1707    1713    1721    1728    1735    1742    1749    1757    1765    1773    1782
                        1789    1797    1805    1813    1821    1828    1835    1842    1850    1858    1869    1875    1878
                        1885    1894    1905    1912    1919    1926    1933    1941    1951    1958    1963    1970    1976
                        1984    1991    1999    2006    2013    2020    2027    2035    2038    2046    2057    2067    2076
                        2083    2090    2099    2109    2119    2129    2137    2139    2141    2143    2152    2154    2157
                        2164    2166    2169    2176    2178    2181    2188    2190    2193    2200    2202    2205    2212
                        2214    2217    2231    2259    2263    2266    2269    2272    2297    2301    2304    2307    2310
                        2322    2326    2329    2332    2335    2358    2370    2383    2396    2406    2408    2410    2434
                        2436    2439    2441    2450    2851    3025
ICOUNT  013354           961*   1640*   2222*   2456*   2490    2498#
IE    = 000100          1166    1380#   1452    1466    1482    1489    1498    1505    1512    1545    1553
IPDREN  001022           870#   2535
IPDRTA  001016           868#   2528
IRF     004142          1132    1134    1327#
KDPAR0  001004           862#
KDPAR1  001006           863#   2790*   2935    2937*   2941*
KDPAR2  001010           864#   2750*   2818*
KDPAR3  001012           865#    902*    914*
KDPAR7  001014           866#   1054*
KDPDR0  000760           852#   1069*
KDPDR1  000762           853#   1101*   2936    2938*   2940*
KDPDR2  000764           854#   2752*   2819*
KDPDR3  000766           855#    904*
KDPDR7  000770           856#   1052*
KIPAR0  000772           857#   1058    1086
KIPAR1  000774           858#   2789*
KIPAR2  000776           859#   2749*
KIPAR3  001000           860#    901*    913*    915
KIPAR7  001002           861#   1053*
KIPDR0  000746           847#   1028*
KIPDR1  000750           848#   1100*
KIPDR2  000752           849#   2751*
KIPDR3  000754           850#    903*
KIPDR7  000756           851#   1051*
KSTACK  017760           938     945    3031#
```

```
KWLST   000634            810#  1022*  1026#
KWLVC   000632            809#  1021*  2643#  2662#
LKCSR   000630            808#  1019   1028*  1281   1284*
LK1     003760           1288#
LK2     003776           1287   1293#
LK3     003726           1021   1281#
LK4     003744           1285#
LLIMIT  004276            970*  1139   1350   1355#  1598*  1599*  2797*  2798*
LOGIC   014774            948   2723#
LOGICA  015014            778   2727   2729#
LPCSR   000636            811#  1032   1042*  1296   1311   1322
LPDBR   000640            812#  1036*  1310*  1321*
LPINTR  004000           1037   1296#
LPST    000644            814#  1038*  1041*
LPVC    000642            813#  1037*  2646*  2663*
LP1     004116           1316   1318#
LP2     004124           1306   1308   1321#
LP3     004042           1304   1307#
LP4     004052           1300   1309#
LP5     004110           1314   1317#
LP6     004010           1299#  1323
LSTSET  015732           2844#  2869
MAIN    003202           1170   1173#  1175   2485
MODE1   003150           1149   1167#
MOD12   003574           1228   1254#
MOD16   003560           1230   1252#
MOD20   003544           1232   1250#
MOD24   003530           1234   1248#
MOD4    003624           1224   1258#
MOD8    003610           1226   1256#
MOVE    003354           1213#  1236   1238   1240   1242   1244   1246
NODEV   003220            999   1123   1181#
NOP     =  000240         745#
NRALL   013426           1050   2525#
NRLOOP  013440           2530#  2536
NUMA    012610           2346*  2354   2355   2360#
NXBNK1  014664           2708#
NXTBLK  006172            970   1355   1477   1529#
NXTBNK  014652           2684   2705#
NXTB1   006216           1531   1534#
NXTB2   006226           1534   1537#
NXTB3   006246           1539   1542#
NXTB4   006322           1543   1552#
NXTSEG  014552           2692#
NXTSET  016040           2846   2858   2866#
OLDBNK  001052            882#  2716*  2816   2818
PAGE    002126           1047#
PARDN   001514            986#
PARFD   001234            905    926#
PARSET  001446            973#
PARSRV  001106            901#   973
PARTO   001230            906    924#
PAR1    001152            908#   916
PAR2    001162            910#   912
PAR3    001170            913#   925
PATCH1= 000000           2563#  2566
```

```
PATCH2=  000000          2564#  2567
PATCH3=  000000          2565#  2568
PC     =x000007           763#  1586   1624   1626   2561*
PFAIL    017032          3018#
PIR      001076           892#   950   1430*  1541*  1617*  2543   2544   2546*  2549*  2814*
PIRQIV   001100           893#
PIRQST   001102           894#
PIRSRV   013470           782   2543#
PIR3     013522          2545   2549#
PRFLG    016724          2962*  2969*  2977   2979*  2984*  2989#
PRINT    016262          2569   2912#
PROCT    016550          2926   2929   2932   2943   2960#
PRTON    016546           988*  2912   2915*  2951*  2958#
PSR    = 177776           748#   764    987*   994*   997*  1002*  1167*  1171*  1595*  1600*  2543*  2574   2647
                         2648*  2649*  2653*  2664   2665*  2666*  2670*  2674*  2700*  2709*  2788*  2807*  2849
                         2916*  2923*  2934*  2942*
PSSAV    001072           890#  2647*  2653   2664*  2670
PTEMP1   016726          2925*  2928*  2931*  2939*  2964   2967*  2968*  2981*  2982*  2983*  2985   2990#
PTEMP2   016730          2963*  2966*  2972   2985*  2986*  2987*  2991#
PTEMP3=  016554          2960*  2961#  2973*
P.CNT1   016642          2974   2976#
P.CONT   016632          2973#
P.WAIT   016616          2970#  2971   2988
R      = 004000          1166   1382#  1466   1489   1498   1512   1542
R8     = 000002          1166   1379#  1452   1466   1498   1505   1512   1545   1553
RBN      005570          1417   1451   1462#  1497   1550   1559
RBN1     005610          1466#  1470   1526   1533   1536
RBN10    006112          1472   1514#
RBN11    006144          1515   1521#
RBN2     005622          1464   1468#
RBN3     005630          1468   1471#
RBN4     005652          1476#  1524
RBN5     005730          1481   1485#
RBN6     005760          1487   1492#
RBN7     006024          1474   1500#
RBN8     006060          1503   1507#  1510
RBN9     006066          1501   1509#
R0     = 000004          1377#  1489
REEX     016054          2842   2852   2870#
REF    = 012232          2276#  2347*  2348*  2353   2356   2446*  2448   2452*
REFF     012232          2240*  2241*  2243   2248   2270   2274#  2276   2279*  2280*  2283   2286   2308   2333
                         2350
REGEN    005460          1437#  2547
REGENL   005504          1442#  1447
REG1   = 001026           766#
RESTRT   017052          3020   3022#
RETURN   013360           966*  1624*  1625*  2493   2496*  2500#  2939
RFCAR    000656           819#  1139*  1350*
RFCSR    000660           820#  1127   1140*  1327   1331   1333   1352*  2805
RFCSRH   000662           821#  1137*  1349*
RFDAE    000646           815#  1344
RFDAR    000652           817#  1346
RFDBR    000650           816#
RFST     000666           823#  1135*
RFSTAR   004244          1330   1337   1339   1343   1349#
RFVC     000664           822#  1132*  1134*  1596*  1597*
```

```
RFWC    000654          818#  1138*  1340   1351*
RF1     004250         1345   1347   1350#
ROTALL  012124         2242   2248#
ROTBE   012272         2281   2286#
ROTBO   012400         2282   2312#
ROTEN1  012500         2285   2337#
RSREG   016034         2860*  2861*  2863*  2864#
RSTRT   001250          791    923    935    938#  2733   2840   3026
RTIK  = 104010          753#  1268   1293   1324   1353   1432   1484   1491   1499   1508   1549   1551   1558
                       1560
RTIKEX  013612         2570   2573#
R0    =%000000          755#   908*   910    911    917*   1001*  1003   1013*  1014*  1015*  1016*  1017   1029*
                       1030   1124*  1125   1147*  1148   1181   1642*  1643*  2791*  2792   2793*  2794*  2798
                       2811
R1    =%000001          756#   910*   975*   979*   984    1057*  1066*  1075*  1084*  1085*  1094*  1128*  1129*
                       1130   1152*  1153   1155   1158   1437   1440*  1442   1443   1445*  1446   1448*  1566
                       1568*  1571   1575   1580*  1581*  1586*  1587*  1588*  1589   1590*  1592   1597   1599
                       1611   1613   1615   1644*  1645*  2525   2528*  2531   2535   2539*  2615*  2616   2617
                       2618   2619   2620   2621   2636*  2637   2638*  2639*  2654*  2655*  2657*  2658*  2659*
                       2671*  2672*  2673   2723*  2729   2759*  2763   2792*  2795*  2796*  2801   2802   2813
                       2820*  2822*
R2    =%000002          757#   909*   912*   1058*  1060*  1061*  1062*  1063*  1064   1076*  1078*  1079*  1081*
                       1082*  1083   1086*  1088*  1089*  1091*  1092*  1093   1210*  1211   1438   1441*  1442*
                       1443*  1444*  1449*  1567   1569*  1570*  1571   1575   1579*  1582*  1646*  1647*  2526
                       2531*  2533*  2538*  2760*  2763*
R3    =%000003          758#   1059*  1060   1061   1065*  1077*  1078   1079   1080*  1087*  1088   1089   1090*
                       1654*  1655*  1864*  1901*  2053*  2063*  2080*  2094*  2104*  2105   2113*  2114*  2117
                       2527   2530*  2534*  2537*  2761*  2762*  2764*  2821*  2823*
R4    =%000004          759#   1946*  2054*  2072*  2087*  2105*  2124*  2125*  2127
R5    =%000005          760#   1626*  1627*  1629   1631   1633   1635   1637   1639   1643   1645   1647   1655
                       1864   1901   1946   2053   2054   2063   2072   2080   2087   2094   2104   2114   2125
R6    =%000006          762#    938*   945*   993*   996*  2525*  2526*  2527*  2537   2538   2539   2701   2702*
                       2703*  2710*
SAVPC   016542         2920*  2925   2956#
SAVPSR  016544         2921*  2928   2957#
SAVR2   016534         2953#
SAVR3   016536         2954#
SAVR4   016540         2955#
SAVR6   017072         3019*  3022   3027#
SBUFF   000600          797#   993
SCOPE = 104400          746#  1652   1662   1670   1678   1684   1693   1700   1708   1714   1722   1729   1736
                       1743   1750   1758   1766   1775   1783   1790   1798   1806   1814   1822   1829   1836
                       1843   1851   1859   1870   1879   1886   1895   1906   1913   1920   1927   1934   1942
                       1952   1959   1964   1971   1977   1985   1992   2000   2007   2014   2021   2028   2039
                       2047   2058   2068   2077   2084   2091   2100   2110   2121   2131   2144   2147   2158
                       2170   2182   2194   2206   2218   2237   2245   2337   2361   2374   2387   2400   2414
                       2428   2442   2455
SCOPEB  013334         2487   2493#
SCOPEC  013262          775    964   2482#
SCOPEF  013356          967*  2490   2492*  2495*  2499#
SCOPEG  013342         2489   2491   2495#
SDPAR7  000740          844#  1109*
SDPDR0  000732          841#  1071*
SDPDR7  000744          846#  1113*  1118*  2677*  2678*  2679
SEGCNT  001030          873#
SEGM1   002254         1056   1068#
```

L05


```
SEGREG= 001024           765#
SETEX   002220          1060#  1066
SETKER  002400          1088#  1094
SETREL  003076          1154   1157#
SETSEG  002650          1067   1116   1121#
SETSP1  002622          1111   1117#
SETSUP  002544          1097   1108#
SETUSE  002334          1078#  1084
SHIFT   016216          2855   2862   2892#
SIPAR7  000736           843#  1108*
SIPDR0  000730           840#  1070*  1072   2657
SIPDR6  000734           842#  2634
SIPDR7  000742           845#  1112*  1117*  2679*
SOLPAT= 004120          1033*  1313   1315*  1317*  1320#
SP     =%000006          761#   762    917    924    926    935*   938*  1182*  1184*  2484  2493*  2555*  2556
                        2557   2558*  2573   2577   2578   2584*  2585*  2598*  2605*  2611*  2612*  2623   2624
                        2625   2626*  2628*  2650   2651*  2652*  2667   2668*  2669*  2687*  2708*  2712*  2713*
                        2726*  2727*  2740*  2747*  2757   2758   2824*  2825   2827   2841*  2844*  2845   2847
                        2854   2856*  2857*  2861   2863   2866*  2867*  2870*  2871*  2872*  2873*  2874*  2875*
                        2876*  2877*  2882*  2883*  2884*  2885*  2886*  2887   2888   2889   2893*  2896*  2897*
                        2898*  2899*  2900*  2901*  2902*  2935*  2936*  2940   2941   3024
SPACE   016732          2927   2930   2933   2944   2994#
SPSAV   013676          2578*  2579*  2580   2581*  2582   2583*  2584   2588#
SR     = 177570           751#   939    955    957   1291*  2594   2608*  2722*  2743  2917  2948
SREG1   001024           765    871#   952*   955*   990   1009   1024   1039   1047  1055  1073  1110  1128
                        1152   1169   1190   1195   1593   2595   2609   2630   2675  2680  2777
SREG2   001026           766    872#   939*   953*   957*   960*  1003   1017   1030  1125  1148  1174  1181*
                        2223   2340   2417   2482   2486   2488   2594*  2601   2615  2616* 2617* 2618* 2619*
                        2620*  2621*  2632   2644   2734   2803   2945   3000
SR0     000670           824#  1122*  2832   2834*  2836   2880*
SR1     000672           825#  2841   2866   2893
SR2     000674           826#  2870
SR3     000676           827#  1121*
START   001264           787    789    945#
STARTX  001334           947    955#
STARTO  001354           954    960#
START1  001362           940    959    961#
STATUS= 177776           764#
STOPTC  005456          1429   1432#
STOR1   004322          1395#  1437*  1448   1566*  1581
STOR2   004324          1396#  1438*  1449   1567*  1582
STRF    002664          1049   1123#
STRF1   002734          1131   1134#
STRT12  003410          1225#
STRT16  003416          1227#
STRT20  003424          1229#
STRT24  003432          1231#
STRT28  003440          1207   1233#
STRT4   003374          1210   1220#
STRT8   003402          1223#
STTC    003004          1126   1141#
ST0     001606           991    999#
ST1     001706          1004   1013#
ST2     001710          1014#
ST3     001712          1015#
ST4     001714          1016#
```

```
ST5     002000            1018    1029#
ST6     002104            1031    1043#
SUBR1   013224            2467#
SUBR2   013226            2468#   2470
SUBR3   013232            2470#   2472
SUBR4   013240            2472#   2474
SUBR5   013246            2474#   2476
SUBR6   013254            2135    2476#
SUPCYC  014306            2635    2657#
SUPDON  014424            2656    2673#
SUPD1   014430            2633    2674#
SUPD2   014466            2676    2680#
SWABA   013154            2447#   2454
SXTEEN  003320            1203#
TCBA    001042             878#   1479*
TCBLK   004312            1390#
TCBUF   004336            1141    1142    1161    1386    1387    1404#   1610    1612
TCBUFP  004302            1141*   1386#   1479    1610*   1611*   2810*   2811*
TCBUFV  004304            1142*   1161*   1387#   1441    1569    1612*   1613*
TCCK    006374            1563#   2550
TCCK1   006422            1570#   1580
TCCK2   006450            1574    1578    1581#
TCCM    001032             874#   1166*   1414*   1419    1421*   1452*   1463    1466*   1480    1482*   1483*   1489*   1490*
                          1498*   1500    1505*   1507*   1512*   1514    1530    1542    1545*   1553*   1605*   1606    2783*
                          2784
TCDT    001036             876#   1473    1492    1502    1509    1516    1522
TCEMA   004330            1143*   1398#   1483    1490    2812*   2813*
TCEOP   004332            1145*   1399#   1426    1454*   1494*   1557*   1603    1616*   2781
TCEXP   004314            1391#   1422*   1453*   1473    1502    1509    1519*   1521*   1522    1546    1554
TCFRST  004306            1388#   1453    1492    1554
TCIV    001044             879#   1160*   1164*   1410*   1411*   1416*   1417*   1450*   1451*   1476*   1477*   1496*   1497*
                          1544*   1548*   1550*   1552*   1556*   1559*   1614*   1615*
TCLAST  004310            1389#   1422    1516    1546
TCOM    012624            2366#   2373
TCOM2   012672            2379#   2386
TCOM3   012740            2392#   2399
TCREL   004334            1146*   1157*   1400#   1424    1601    2779    2808
TCSR  = 000616             749#   2737    2970    2994    3003    3009    3012
TCST    001034             875#   1150    1412    1462    1529
TCSTA   001046             880#   1165*
TCSTOR  004316            1392#   1462*   1529*
TCWC    001040             877#   1478*
TDBR  = 000620             750#   2739*   2972*   2996*   3005*   3011*   3014*
TEMP    013414            1636*   1637*   1638*   1639*   1674*   1675    1680*   1681    1688*   1689*   1690    1695*   1696*
                          1697    1704*   1705*   1710*   1711*   1717*   1719    1724*   1726    1731*   1733    1738*   1740
                          1745*   1747    1752*   1755    1760*   1763    1786*   1787    1793*   1794*   1795    1801*   1802*
                          1803    1809*   1810*   1811    1817*   1818*   1819    1824*   1825*   1826    1831*   1832*   1833
                          1838*   1839*   1840    1845*   1847*   1848    1853*   1855*   1856    1866*   1882*   1883    1890*
                          1891*   1892    1899*   1903    1908*   1910    1915*   1917    1922*   1924    1929*   1931    1936*
                          1939    1944*   1949    1973*   1974    1987*   1988*   1989    1995*   1996*   1997    2002*   2003*
                          2004    2009*   2010*   2011    2016*   2017*   2018    2023*   2024*   2025    2030*   2032*   2033
                          2036    2041*   2043*   2044    2062*   2064*   2065    2073*   2106*   2149*   2150*   2155    2161*
                          2162*   2167    2173*   2174*   2179    2185*   2186*   2191    2197*   2198*   2203    2209*   2210*
                          2215    2364*   2365*   2366*   2367*   2368    2371*   2372*   2377*   2378*   2379*   2380*   2381
                          2384*   2385*   2390*   2391*   2392*   2393*   2394    2397*   2398*   2403*   2404    2411*   2412
                          2518#   2519    2520    2521
```

```
TEST    012234          2248*  2249*  2250*  2251*  2252*  2253*  2254*  2270   2275#  2286*  2287*  2288*  2289*
                        2290*  2291*  2292*  2308   2312*  2313*  2314*  2315*  2316*  2317*  2333   2353*  2354*
                        2355*  2356   2431*  2432*  2437*  2445*  2447*  2448   2451*  2453*
TIME  = 003762          1027*  1285*  1286   1289#
TJSR1   011434          2113   2116#
TJSR2   011436          2115   2117#
TJSR3   011446          2116   2121#
TJSR4   011462          2124   2127#
TRCSR   000606           799#
TRDR    000610           800#
TRPB    001070           889#  1043*  2603*  2745*
TRTRP   016260          2606   2748   2906#
TSCOMB  013002          2404#  2413
TSEG    016144          2878   2882#
TSROT   012100          2241#  2244
TSROT2  012244          2280#  2284
TSRT2A  012236          2224   2246   2279#
TSTARI  012516          2341   2346#
TTCSR   000616           749    803#  1005   1012*  1269
TTDBR   000620           750    804#  1267*
TTPST   000624           806#  1006*  1011*
TTPVC   000622           805#  1008*  2640*  2660*
TTRST   000614           802#
TTRVC   000612           801#  2642*  2661*
TTSAV   000626           807#  1269*  1270
TWELVE  003314          1202#
TWENTY  003324          1204#
TWOEIG  003334          1206#
TWOFOR  003330          1205#
TYOUT   003654          1265#  1276
TYOUTR  003670          1008   1269#
TYOUT1  003660          1267#  1277
UBRK    001074           891#
UBUFF   000400           795#   996   2708
UDPAR0  000722           837#  1104   2766
UDPAR1  000724           838#
UDPAR7  000726           839#  1095*  2774*
UDPDR0  000706           831#  1099*  1106   2768
UDPDR1  000710           832#
UDPDR7  000712           833#  1096*  2776*
UIPAR0  000714           834#  1076   1103   2765
UIPAR1  000716           835#
UIPAR7  000720           836#  2683   2773*
UIPDR0  000700           828#  1098*  1105   2767
UIPDR1  000702           829#
UIPDR7  000704           830#  2775*
UN1   = 000400          1166   1383#  1452   1466   1482   1489   1498   1505   1512
USEALL  002450          1074   1098#
USER    017074          1045   3029#
WAIT4   013102          2421   2428#
WD    = 000014          1378#  1482
WORDCT  004300          1138   1351   1356#
XFER12  003506          1225   1243#
XFER16  003476          1227   1241#
XFER20  003466          1229   1239#
XFER24  003456          1231   1237#
```

```
XFER2B  003446          1233    1235#
XFER8   003516          1223    1245#
.     = 017762           769#    771#    774#    777#    779#    781#    786#    788#    790#    794#    796#    798#   1010
                        1025    1040    1048    1133    1156    1159    1163    1172    1193    1218    1266    1271    1282
                        1289    1297    1302    1312    1319    1320    1328    1341    1405#   1493    1517    1520    1523
                        1547    1555    1572    1576    1604    1607    1609    1650    1657    1660    1665    1668    1676
                        1682    1691    1698    1706    1712    1720    1727    1734    1741    1748    1756    1764    1781
                        1788    1796    1804    1812    1820    1827    1834    1841    1849    1857    1868    1874    1877
                        1884    1893    1904    1911    1918    1925    1932    1940    1950    1957    1962    1969    1975
                        1983    1990    1998    2005    2012    2019    2026    2034    2037    2045    2056    2066    2075
                        2082    2089    2098    2108    2118    2128    2136    2138    2140    2142    2151    2153    2156
                        2163    2165    2168    2175    2177    2180    2187    2189    2192    2199    2201    2204    2211
                        2213    2216    2230    2256    2257    2258    2260    2261    2262    2264    2265    2267    2268
                        2271    2294    2295    2296    2298    2299    2300    2302    2303    2305    2306    2309    2319
                        2320    2321    2323    2324    2325    2327    2328    2330    2331    2334    2357    2369    2382
                        2395    2405    2407    2409    2426    2433    2435    2438    2440    2449    2483    2506#   2509#
                        2517#   2520#   2534    2575    2622    2645    2681    2706    2738    2782    2785    2787    2804
                        2806    2833    2837    2850    2895    2913    2918    2946    2949    2961    2965    2978    2995
                        3001    3004    3010    3013    3030#
```

TNCV 2220# 2255 2293 2318

E06

F06

DCKTGE.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
        2893  2916  2920  2921  2923  2925  2928  2931  2934  2935  2936  2937  2938  2939  2940
        2941  2942  2960  2963  2972  2985  2996  3005  3011  3014  3018  3019  3020  3022  3023
MOVB     939  1140  1362  1545  1553  1882  1967  1973  2543  2594
MTPD     995   998  2585  2652  2669  2703  2710  2857
NEG     1579  1746  1839
NEGB    1930  2024
NOP     1221  1222  1273  1563  1564  1565  1774  2464  2730  2731  2732  2831  2853  2859  2892
RESET    922   934  2725  2839
ROL     2252  2253  2254  2967  2968  2981  2982  2983
ROLB    2174  2186  2290  2291  2292  2315  2316  2317
ROR     2249  2250  2251
RORB    2150  2162  2287  2288  2289  2312  2313  2314
RTI      978  1185  1415  1423  2494  2497  2548  2551  2576  2587  2627  2629  2704  2728  2826
        2835  2890  2914  2952
RTS     1194  1197  1219  1247  1262  1455  1584  2120  2273  2311  2336  2359  2467  2469  2471
        2473  2475  2477  2540  2903  2975  2997  3002  3006  3015  3029
RTT     2881  2906
SBC     1762  1855
SBCB    1948
SCC     2160  2184  2208  2468
SEC     1753  1761  1846  1854  1937  1947  2031  2042  2980
SOB      912  1066  1084  1094  1580  2534  2764  2823
SUB     1182  1705  1711  1802  2355  2555  2581  2856  2864
SWAB    1336  1348  2432  2437  2447  2451  2867  2896
SXT     2898
TRAP     746
TST      911   946   950  1005  1019  1032  1064  1083  1093  1127  1130  1150  1209  1331  1340
        1412  1424  1426  1471  1485  1601  1603  1606  1608  1719  1726  1733  1740  1755  1763
        1803  1819  1848  1856  2044  2130  2637  2779  2781  2784  2786  2808  2912  2948  2964
        2977
TSTB    1270  1281  1296  1311  1322  1327  1537  1910  1924  1997  2036  2737  2805  2970  2994
        3003  3009  3012
WAIT    1173  2422  2423  2424  2425
.ABS     697
.DSABL   697
.END    3032
.LIST      1   697   769
.MACR   2220
.NLIST     1   697   769
.REM       1
.REPT    769
.SBTTL     1   697   742   942  1621  2479
.TITLE   697
.WORD   2865
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

#DCKTGE,DCKTGE.SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DCKTGE.P11
RUN-TIME: 9 19 4 SECONDS
RUN-TIME RATIO: 72/34=2.0
CORE USED: 10K (20 PAGES)