

IDENTIFICATION

PRODUCT CODE MAINDEC-15-DAUCA-B-D
PRODUCT NAME UNICHANNEL-15 DIAGNOSTIC
DATE CREATED JANUARY 7, 1974
MAINTAINER PDP-15 DIAGNOSTICS
AUTHOR R. CHRISTOPHER

27

"The material in this document is for information purposes only and is subject to change without notice. Digital Equipment Corporation assumes no responsibility for the use of software on equipment which is not supplied by it. Digital Equipment Corporation assumes no responsibility for any errors which may appear in the document."

COPYRIGHT © 1973, 1974
DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT

THIS DIAGNOSTIC TESTS THE UNICHANNEL15 HARDWARE WHICH CONSISTS OF THE PDP-15/PDP-11 COMMON MEMORY ACCESSED THROUGH THE MX15-B, THE DR15, AND THE DR11-C'S (2).

THE DIAGNOSTIC CONSISTS OF TWO SEPARATE PROGRAMS, A PDP-15 PROGRAM AND A PDP-11 PROGRAM. LISTINGS OF BOTH PROGRAMS MAY BE FOUND IN THE BACK OF THIS DOCUMENT, AND SHOULD BE CONSULTED TO DETERMINE WHAT EACH PROCESSOR WAS DOING AT THE TIME A FAILURE OCCURRED.

2. EQUIPMENT REQUIREMENTS

A. PDP-15 (WITH AT LEAST 12K OF MEMORY, PAPER TAPE READER AND CONSOLE TTY)

B. UNICHANNEL15 HARDWARE CONSISTING OF:

4K OR 8K PDP-11/05

MX15-B

DR15

2 DR11-C'S

3. STORAGE

THE PDP-11 PROGRAM OCCUPIES 2K OF CORE

THE PDP-15 PROGRAM OCCUPIES 4K OF CORE

4. PRELIMINARY PROGRAMS

ALL APPLICABLE PDP-15 AND PDP-11 MAINDEC DIAGNOSTICS MUST RUN SUCCESSFULLY BEFORE ATTEMPTING TO RUN THIS DIAGNOSTIC. APPLICABLE MEANING DIAGNOSTICS WHICH CHECK ALL HARDWARE PRESENT MINUS THE UNICHANNEL HARDWARE.

5. LOADING PROCEDURES

PLACE THE BANK MODE SWITCH ON THE PDP-15 ON A 1 AND LOAD THE DIAGNOSTIC IN THE FOLLOWING ORDER.

A. PDP-11

1. PLACE THE ABSL11 LOADER IN THE PDP-15 READER
2. SET THE PDP-15 ADDRESS SWITCHES TO 17700
3. PRESS RESET AND THEN READ IN
4. SET THE PDP-11 ADDRESS SWITCHES TO 60000 FOR 4K OF LOCAL PDP-11 MEMORY OR 100000 FOR 8K
5. PRESS LOAD ADDRESS AND THEN START ON THE PDP-11
6. PLACE THE PDP-11 BINARY IN THE PDP-15 READER
7. PRESS CONTINUE ON THE PDP-15
8. THE PDP-11 BINARY IS NOW LOADED

B. PDP-15

1. PLACE THE PDP-15 BINARY IN THE PDP-15 READER
2. SET THE PDP-15 ADDRESS SWITCHES TO 17700
3. PRESS RESET AND THEN READ IN

6. SWITCH OPTIONS

A. PDP-11 (NONE)

B. PDP-15

AC SWITCH	DESCRIPTION																		
0=0	HALT ON ERROR																		
0=1	LOOP ON ERROR																		
1=1	INDICATE ERROR AND GO ON TO NEXT TEST (HAS PRIORITY OVER AC SWITCH 0)																		
2=0	RUN MULTIPLE PASSES OF EACH TEST																		
2=1	RUN EACH TEST ONLY ONCE PER PASS																		
3=1	DELETE TTY OUTPUT																		
4=1	RING BELL ON ERROR																		
5=1	TYPE # OF PASSES (IN OCTAL) COMPLETED SINCE PROGRAM WAS LAST STARTED. MESSAGE WILL BE TYPED AT END OF CURRENT PASS AND DOUBLES AS END OF PASS INDICATOR.																		
6-8	INDICATES AMOUNT OF COMMON MEMORY AVAILABLE. (MEMORY THAT CAN BE ACCESSED BY BOTH PROCESSORS)																		
	<table border="0"> <thead> <tr> <th>AC SW 6-8</th> <th>COMMON MEM</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>8K</td> </tr> <tr> <td>1</td> <td>12K</td> </tr> <tr> <td>2</td> <td>16K</td> </tr> <tr> <td>3</td> <td>20K</td> </tr> <tr> <td>4</td> <td>24K</td> </tr> <tr> <td>5</td> <td>"</td> </tr> <tr> <td>6</td> <td>"</td> </tr> <tr> <td>7</td> <td>"</td> </tr> </tbody> </table>	AC SW 6-8	COMMON MEM	0	8K	1	12K	2	16K	3	20K	4	24K	5	"	6	"	7	"
AC SW 6-8	COMMON MEM																		
0	8K																		
1	12K																		
2	16K																		
3	20K																		
4	24K																		
5	"																		
6	"																		
7	"																		
	24K OF COMMON MEMORY IS A LEGAL SELECTION ONLY WHEN THE 11/05 HAS ONLY 4K OF LOCAL MEMORY.																		
9=1	HALT AT THE END OF EACH TEST																		
10=1	RANDOMLY SELECT THE ORDER IN WHICH TESTS ARE RUN, AND IF AN ERROR OCCURS, REPORT THE 3 TESTS WHICH WERE RUN PREVIOUS TO IT.																		
11-17	LOOP ON TEST SELECTED (WHEN SET TO 177, THE POWER FAIL TEST WILL BE SELECTED)																		
NOTE:	SWITCH OPTIONS ABOVE CAN BE SET WHEN PROGRAM IS RUNNING.																		

7. STARTING AND RESTARTING ADDRESSES

-
- A. PDP-11
 - START=200
 - RESTART=1100
 - B. PDP-15
 - START=200
 - RESTART=200

8. STARTING PROCEDURE

EITHER PROGRAM MAY BE STARTED FIRST, THEY WILL AUTOMATICALLY SYNC UP.

- A. PDP-11
 - A. LOAD THE STARTING (OR RESTARTING) ADDRESS INTO THE SWITCHES
 - B. PRESS START
- B. PDP-15
 - A. LOAD THE STARTING ADDRESS INTO THE ADDRESS SWITCHES
 - B. SELECT AC SWITCH OPTIONS
 - C. PRESS RESET
 - D. PRESS STRART

9. OPERATING PROCEDURE

-
- A MAXIMUM OF AC SWITCH OPTIONS HAS BEEN PROVIDED. WITH NONE OF THE AC SWITCHES SET THE PROGRAM WILL PERFORM THE FOLLOWING:
- A. ALL TESTS WILL BE RUN (WITH THE EXCEPTION OF POWER FAIL, WHICH MUST BE SELECTED MANUALLY)
 - B. TESTS WILL BE RUN FROM FIRST TO LAST, IN ORDER
 - C. THE PROGRAM WILL HALT ONLY ON ERROR
 - D. EACH TEST WILL BE RUN A PRESELECTED NUMBER OF TIMES BEFORE THE NEXT SEQUENTIAL TEST IS RUN. EACH TEST IS NOT EXECUTED THE SAME NUMBER OF TIMES, BUT INSTEAD IS REPEATED AS REQUIRED FOR THAT PARTICULAR TEST
 - E. ALL ERRORS WILL BE INDICATED BY AN ERROR MESSAGE OUTPUT ON THE PDP-15 TTY, AND THEN ONLY THE PDP-15 WILL HALT. PRESSING CONTINUE WILL REPEAT THE FAILING TEST IN BOTH PROCESSORS.
 - F. A FAST SERIES OF THREE TTY BELLS WILL BE OUTPUT AT THE END OF A PASS
 - G. THE PROGRAM WILL ASSUME ONLY THE MINIMUM OF 8K OF COMMON MEMORY IS AVAILABLE UNLESS OTHERWISE INDICATED IN AC SWITCHES 6-8

10. ERRORS

ALL ERRORS WILL BE INDICATED BY A PDP-15 TTY MESSAGE, A SAMPLE WOULD BE:

TST- GOOD - BAD
AA BBBB BB CCCCC

WHICH WOULD INDICATE THAT TEST AA FAILED, THAT IT EXPECTED TO FIND DATA EQUAL TO BBBB BB BUT INSTEAD FOUND CCCCC. ANOTHER SAMPLE:

TST- ADDR - GOOD - BAD
AA BBBB BB CCCCC DDDDD

WHICH WOULD INDICATE THAT TEST AA FAILED, THAT IT EXPECTED TO FIND DATA CCCCC AT ADDRESS BBBB BB AND INSTEAD FOUND DDDDD. ANOTHER:

TST-ERROR DESCRIPTION

WILL INDICATE THE TEST THAT FAILED AND A BRIEF DESCRIPTION OF WHAT HAPPENED. IN ALL CASES THE PROGRAM LISTINGS SHOULD BE CONSULTED TO DETERMINE EXACTLY WHAT EACH PROCESSOR WAS TRYING TO DO. WHEREVER POSSIBLE IDENTICAL OR VERY SIMILAR PROGRAM TAGS WERE USED IN BOTH THE PDP-15 AND PDP-11 PROGRAMS TO HELP IN CORRELATING THE TWO.

11. PROGRAM DESCRIPTION

THERE ARE TWO COMMON MEMORY LOCATIONS WHICH ARE USED TO KEEP THE TWO PROGRAMS IN SYNC. BOTH PROGRAMS CONSTANTLY READ AND WRITE THESE LOCATIONS. IF FOR ANY REASON (I.E. BAD INFORMATION, PROCESSOR NOT ADDRESSING CORRECT LOC, ETC.) THE PROCESSORS ARE NOT RECEIVING WHAT THEY EXPECT, THE DIAGNOSTIC WILL NOT RUN. THE PDP-15 WILL USUALLY DETECT THIS FACT AND TYPE THE MESSAGE "WAITING FOR PDP-11 TO INDICATE READY TO TEST". THESE TWO LOCATIONS ARE TAGGED "IDNE15" AND "IDNE11". IT IS STILL POSSIBLE THAT THE PROGRAMS MAY GET OUT OF SYNC FOR THE ABOVE REASONS AND SIMPLY HANG. THE FACT THAT THE PROGRAM IS HUNG MAY BE DETERMINED BY MONITORING THE LIMIT REGISTER ON THE 15'S CONSOLE. IT ALWAYS CONTAINS THE TEST NUMBER CURRENTLY BEING RUN BY THE PDP-15. THE PDP-11 PROGRAM NEVER HALTS UNLESS AN UNEXPECTED TRAP OCCURS. WHENEVER ONE PROGRAM IS STARTED OR RESTARTED THE OTHER PROGRAM MUST ALSO BE RESTARTED. TTY ERROR MESSAGES WILL ALWAYS INDICATE PDP-15 ADDRESSES. THE DATA WORDS OUTPUT WILL EITHER BE THE ADDRESS OF THAT LOCATION OR ALL 1'S DEPENDING ON THE TEST EXECUTED.

12.

PROGRAM LISTINGS (PDP-15 AND PDP-11)

```

        .TITLE * UNICHANNEL15 * MAINDEC-15-DAUCA-B * MAY 25, 1973 *
        .ABS
/COPYRIGHT 1973, DIGITAL EQUIPMENT CORP., MAYNARD, MASS, 01754
/
/AC SWITCH OPTIONS
/0      =0 MALT ON ERROR
/1      =1 LOOP ON ERROR (FOREVER)
/2      =1 INDICATE ERROR & GO ON TO NEXT TEST (HAS PRIORITY OVER AC 0)
/3      =0 RUN MULTIPLE PASSES OF EACH TEST
/4      =1 RUN EACH TEST ONCE
/5      =1 DELETE TTY OUTPUT
/6      =1 RING BELL ON ERROR
/7      =1 TYPE # OF COMPLETE PASSES AT THE END OF CURRENT PASS (IN OCTAL)
/8      (ALSO MAY BE USED TO INDICATE END OF EACH PASS)
/9-10   = INDICATES AMOUNT OF COMMON MEMORY AVAILABLE
/11     0=8K
/12     1=12K
/13     2=16K
/14     3=20K
/15     4=24K          (ILLEGAL IF PDP-11 HAS 8K LOCAL MEMORY)
/16     5= "
/17     6= "
/18     7= "
/19     =1 MALT AT END OF EACH TEST
/20     =1 RANDOMLY SELECT THE ORDER IN WHICH TESTS ARE RUN, AND IF
/21     AN ERROR OCCURS, REPORT THE 3 TESTS WHICH WERE RUN PREVIOUSLY.
/22-23  = LOOP ON TEST SELECTED
/24-25  = 177 SELECT POWER FAIL TEST
/
700401   TSF=700401
700402   TCF=700402
700406   TLF=700406
700301   KSF=700301
700312   KRF=700312
703201   PFSF=703201
722000   PAL=722000
705512   RPL=705512
701741   MPSNE=701741
/
/*****NOTE: TEST BEING EXECUTED IS INDICATED IN LIMIT REGISTER,
        .EJECT
    
```

```

00200    .LOC    200
/
00200    600215  JMP    ISTART
/
/COMMON MEM INTER-COMM LOC'S (ADDRESSES MUST NOT BE CHANGED)
00201    .LOC    201
00201    000000  TEST   0      /INDICATES # OF TEST THAT FAILED
00202    .LOC    202
00202    000000  ADR    0
00203    .LOC    203
00203    000000  GOOD   0      /AFTER ERROR CONTAINS ADDRESS AND DATA EXPECTED
/ (SAME). THE ADDRESS POINTING TO AN UPPER OR
/ LOWER 2K SEGMENT OF A 4K PAGE, IN COMBINATION
/ WITH THE TEST#, ALLOWS OPERATOR TO DETERMINE
/ IF ERROR WAS DETECTED WHILE 15 OR 11 WAS
/ READING. IF 4TH DIGIT FROM RIGHT IN THE GOOD
/ WORD IS 3 OR LESS, THE ERROR OCCURRED IN THE
/ LOWER 2K SEGMENT. IF NOT IT WAS THE UPPER 2K.
00204    .LOC    204
00204    000000  BAD    0      /AFTER ERROR CONTAINS DATA READ
00205    .LOC    205
00205    000000  ERRFLG 0      /-1 OR 17777 INDICATES ERROR
00206    .LOC    206
00206    000000  RWSA15 0      /INDICATES 15'S STARTING ADDRESS FOR READ OR
/ WRITE
00207    .LOC    207
00207    000000  RWSA11 0      /INDICATES 11'S STARTING ADDRESS FOR READ OR
/ WRITE
00210    .LOC    210
00210    000000  ERRIN) 0      /SET BY ERROR, ONLY CLEARED AT START OF PROGRAM
00211    .LOC    211
00211    000000  BR5TV  0      /CONTAINS PDP-11 BR5 TRAP VECTOR FOR DR11-C
00212    .LOC    212
00212    000000  BR7TV  0      /CONTAINS PDP-11 BR7 TRAP VECTOR FOR DR11-C
00213    .LOC    213
00213    000000  IDNE15 0      /-1 INDICATES 15 DONE FUNCTION
00214    .LOC    214
00214    000000  IDNE11 0      /177777 INDICATES 11 DONE FUNCTION
        .EJECT
    
```



```

00215 140213 / ISTART DZM IDNE15 /CLEAR PDP-15 DONE FUNCTION INDICATOR
00216 140204 DZM BAD
00217 143424 DZM PASCNT
00220 206403 LAC (HLT
00221 040021 DAC 21
00222 206444 LAC (TEST1
00223 043455 DAC ITST
00224 203432 LAC INIT
00225 740200 SZA /FIRST TIME THROUGH?
00226 600273 JMP START /NO
00227 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00230 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00231 103051 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00232 005141 MES36 /TEST API?
00233 143415 DZM IAPI
00234 700301 KSF
00235 600234 JMP *-1
00236 700312 KRB
00237 546405 SAD (331 /Y?
00240 741000 SKP /YES
00241 600244 JMP *+3
00242 443415 ISZ IAPI
00243 600251 JMP ISTA.1
00244 546406 SAD (310 /N?
00245 600251 JMP ISTA.1 /NO
00246 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00247 004156 MES2
00250 600215 JMP ISTART
00251 103643 ISTA.1 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00252 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00253 004160 MES3
00254 103574 JMS READ
00255 600251 JMP ISTA.1 /BAD INPUT, TRY AGAIN
00256 741200 SNA
00257 206407 LAC (300
00260 040211 ISTA.2 DAC BR5TV /STORE BR5 TRAP VECTOR
00261 103643 ISTA.2 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00262 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00263 004175 MES4
00264 103574 JMS READ
00265 600261 JMP ISTA.2
00266 741200 SNA
00267 206410 LAC (310
00270 040212 ISTA.2 DAC BR7TV /STORE BR7 TRAP VECTOR
00271 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00272 143423 DZM TMPCNT
.EJECT
    
```

```

00273 200203 / START LAC GOOD
00274 546411 SAD (125252 /LOAD TIME?
00275 741000 SKP /YES
00276 600302 JMP STAR.1
00277 200214 LAC IDNE11 /GET PDP-11 DONE FUNCTION INDICATOR
00300 546412 SAD (177777
00301 600311 JMP STAR.2 /11 PROGRAM INDICATES READY TO TEST
00302 443423 STAR.1 ISZ TMPCNT
00303 600273 JMP START
00304 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00305 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00306 005101 MES37
00307 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00310 600273 STAR.2 JMP START
00311 140203 STAR.2 DZM GOOD
00312 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00313 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00314 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00315 005125 MES35
00316 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00317 206413 LAC (52525
00320 040204 DAC BAD /INDICATE LOAD TIME TO 11
.EJECT
    
```

```

/DETERMINE FROM AC SMS 15-17 AMOUNT OF COMMON MEN AND SET UP TABLE
/OF FIRST ADDRESS OF EACH 4K PAGE.
00321 104120 STAR.3 JMS NXMCK /CHECK FOR NEXM FLAG
00322 140201 DZM TEST
00323 140213 DZM IDNE15 /CLEAR PDP-15 DONE FUNCTION INDICATOR
00324 143432 DZM INIT
00325 443432 ISZ INIT
00326 140210 DZM ERRIND
00327 103227 JMS M11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
00330 206414 STAR.4 LAC (ADRTBL-1
00331 040017 DAC 17 /INIT PAGE POINTER
00332 220017 LAC* 17
00333 043420 DAC ADRPNT /SET TO TEST FIRST 4K PAGE
00334 750004 LAS /GET CONTENTS OF AC SWITCHES
00335 103712 JMS RTMR /ROTATE AC RIGHT 3X
00336 103712 JMS RTMR /ROTATE AC RIGHT 3X
00337 103712 JMS RTMR /ROTATE AC RIGHT 3X
00340 506415 AND (?)
00341 546416 SAD (5
00342 506417 AND (4
00343 546420 SAD (6
00344 506417 AND (4
00345 546415 SAD (7
00346 506417 AND (4 /5, 6 & 7 ARE ILLEGAL!
00347 740001 CHA
00350 043423 DAC THPCNT /INITIALIZE COUNT
00351 206421 LAC (ADRTBL
00352 040010 DAC 10
00353 443423 ISZ THPCNT /8K?
00354 741000 SKP /NO
00355 600377 JMP STAR.5 /YES
00356 206422 LAC (20000
00357 060010 DAC* 10
00360 443423 ISZ THPCNT /12K?
00361 741000 SKP
00362 600377 JMP STAR.5
00363 206423 LAC (30000
00364 060010 DAC* 10
00365 443423 ISZ THPCNT /16K?
00366 741000 SKP
00367 600377 JMP STAR.5
00370 206424 LAC (40000
00371 060010 DAC* 10
00372 443423 ISZ THPCNT
00373 741000 SKP
00374 600377 JMP STAR.5
00375 206425 LAC (50000
00376 060010 DAC* 10 /MAX 24K COMMON MEN!
00377 777777 LAM -1
00400 060010 DAC* 10 /IND END OF TABLE
.EJECT

```

```

00401 206414 LAC (ADRTBL-1
00402 040017 DAC 17
00403 103054 JMS TSTSEL /SET UP TO DO NXT TST
00404 103027 JMS PAS
00405 200201 LAC TEST /GET TEST NUMBER
00406 343405 TAD TSTTBL
00407 043421 DAC TSTPNT
00410 223421 LAC* TSTPNT
00411 506420 AND (7777
00412 546426 SAD (7777 /DONE LAST TEST?
00413 600433 JMP STAR.7 /YES
00414 043421 DAC TSTPNT
00415 140205 STAR.6 DZM ERRFLG /CLEAR ERROR FLAG
00416 104120 JMS NXMCK /CHECK FOR NEXM FLAG
00417 123421 JMS* TSTPNT /EXECUTE SELECTED TEST
00420 200205 LAC ERRFLG /GET ERROR INDICATOR
00421 506412 AND (177777
00422 546412 SAD (177777
00423 600430 JMP ,+5
00424 750004 LAS /GET CONTENTS OF AC SWITCHES
00425 506427 AND (400
00426 740200 SZA /HLT AT END OF TEST?
00427 740000 MLT
00430 443437 ISZ PASIND
00431 600415 JMP STAR.6
00432 600330 JMP STAR.4
00433 140201 STAR.7 DZM TEST
00434 443424 ISZ PASCNT
00435 740000 NOP
00436 760207 LAM 207
00437 103742 JMS SPTYF /TYPE CHAR REGARDLESS OF AC SM 3
00440 760207 LAM 207 /TYPE CHAR REGARDLESS OF AC SM 3
00441 103742 JMS SPTYF /TYPE CHAR REGARDLESS OF AC SM 3
00442 760207 LAM 207 /TYPE CHAR REGARDLESS OF AC SM 3
00443 103742 JMS SPTYF /GET CONTENTS OF AC SWITCHES
00444 750004 LAS (10000
00445 506430 AND
00446 741200 SNA
00447 600330 JMP STAR.4
00450 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00451 203424 LAC PASCNT
00452 104013 JMS OCT
00453 103716 JMS TYP
00454 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
00455 005622 MESS4
00456 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
00457 600330 JMP STAR.4
.EJECT

```

```

/*****
/TST1. 15 WRITES IN LOWER 2K, 11 READS & CHECKS IT.
TST1 0
00460 000000 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00461 203420 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00462 040206 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00463 040207 LAM -4000
00464 774000 JMS IBUFF /INITIALIZE BUFFER
00465 104106 LAM -1
00466 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
00467 040213 LAC RNSA15
00470 200206 TAD (-1
00471 346431 DAC 10 /INIT ADDR POINTER FOR 15
00472 040010 JMS WRT2K /WRITE ADDR INTO LOWER 2K OF PAGE
00473 103362 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
00474 103222 JMS ERRCHK /ERROR?
00475 102625 JMP TST1.A /NO
00476 600500 JMP E1R0U /TYPE TST #, ADDRESS, GOOD & BAD DATA
00477 602712 TST1.A JMS PAGSEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00500 103173 JMP TST1+1
00501 600461 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
00502 602674
/*****
/TST2. 15 WRITES IN UPPER 2K, 11 READS & CHECKS IT.
TST2 0
00503 000000 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00504 203420 TAD (4000 /MAKE FIRST ADDR OF UPPER 2K
00505 346432 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00506 040206 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00507 040207 LAM -4000
00510 774000 JMS IBUFF /INITIALIZE BUFFER
00511 104106 LAM -1
00512 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
00513 040213 LAC RNSA15
00514 200206 TAD (-1
00515 346431 DAC 10 /INIT ADDR POINTER FOR 15
00516 040010 JMS WRT2K /WRITE ADDR INTO UPPER 2K OF PAGE
00517 103362 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
00520 103222 JMS ERRCHK /ERROR?
00521 102625 JMP TST2.A /NO
00522 600524 JMP E1R0U /TYPE TST #, ADDRESS, GOOD & BAD DATA
00523 602712 TST2.A JMS PAGSEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00524 103173 JMP TST2+1
00525 600504 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
00526 602674
/*****
.EJECT

```

```

/*****
/TST3. 11 WRITES IN LOWER 2K, 15 READS & CHECKS IT.
TST3 0
00527 000000 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00530 203420 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00531 040207 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00532 040206 LAM -4000
00533 774000 JMS IBUFF /INITIALIZE BUFFER
00534 104106 LAM -1
00535 200206 LAC RNSA15
00536 346431 TAD (-1
00537 040010 DAC 10 /INIT ADDR POINTER FOR 15
00540 103222 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
00541 103373 JMS READ2K /READ & CHECK 2K
00542 102625 JMS ERNCHK /ERROR?
00543 600545 JMP TST3.A /NO
00544 602712 JMP E1R0U /TYPE TST #, ADDRESS, GOOD & BAD DATA
00545 103173 TST3.A JMS PAGSEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00546 600530 JMP TST3+1
00547 602674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
/TST4. 11 WRITES IN UPPER 2K, 15 READS & CHECKS IT.
TST4 0
00550 000000 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00551 203420 TAD (4000 /MAKE FIRST ADDR OF UPPER 2K
00552 346432 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00553 040207 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00554 040206 LAM -4000
00555 774000 JMS IBUFF /INITIALIZE BUFFER
00556 104106 LAM -1
00557 200206 LAC RNSA15
00560 346431 TAD (-1
00561 040010 DAC 10 /INIT ADDR POINTER FOR 15
00562 103222 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
00563 103373 JMS READ2K /READ & CHECK 2K
00564 102625 JMS ERRCHK /ERROR?
00565 600567 JMP TST4.A /NO
00566 602712 JMP E1R0U /TYPE TST #, ADDRESS, GOOD & BAD DATA
00567 103173 TST4.A JMS PAGSEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00570 600551 JMP TST4+1
00571 602674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/*****
/TST5. 15 WRITES IN LOWER 2K WHILE 11 IS WRITING IN UPPER 2K, 11
/HEADS & CHECKS LOWER 2K WHILE 15 IS READING & CHECKING UPPER 2K.
TST5 0
00572 000000 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00573 203420 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00574 040206 TAD (4000 /MAKE FIRST ADDR OF UPPER 2K
00575 346432 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00576 040207 LAC (-10000 /INITIALIZE BUFFER
00577 206433 JMS Ibuff
00600 104100 LAM -1
00601 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
00602 040213 LAC RNSA15
00603 200206 TAD (-1
00604 346431 DAC 10 /INIT ADDR POINTER FOR 15
00605 040010 JMS WRT2K /WRT ADDR INTO LOWER 2K OF PAGE
00606 103302 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
00607 103227 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00610 203420 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00611 040207 TAD (4000
00612 346432 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00613 040206 LAM -1
00614 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
00615 040213 LAC RNSA15
00616 200206 TAD (-1
00617 346431 LAC 10 /INIT ADDR POINTER FOR 15
00620 040010 JMS READ2K /READ & CHECK UPPER 2K
00621 103373 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
00622 103227 JMS ERRCHK /ERROR?
00623 102625 JMP TST5,A /NO
00624 000626 JMP E1ROU /TYPE TST #, ADDRESS, GOOD & BAD DATA
00625 002712 TST5,A JMS PAGESEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00626 103173 JMP TST5+1
00627 000573 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
00630 002674
/*****
.EJECT

```

```

/*****
/TST6. 11 WRITES IN LOWER 2K WHILE 15 IS WRITING IN UPPER 2K, 15
/HEADS & CHECKS LOWER 2K WHILE 11 IS READING & CHECKING UPPER 2K.
TST6 0
00631 000000 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00632 203420 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00633 040207 TAD (4000 /MAKE FIRST ADDR OF UPPER 2K
00634 346432 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00635 040206 LAC (-10000 /INITIALIZE BUFFER
00636 206433 JMS Ibuff
00637 104100 LAM -1
00640 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
00641 040213 LAC RNSA15
00642 200206 TAD (-1
00643 346431 DAC 10 /INIT ADDR POINTER FOR 15
00644 040010 JMS WRT2K /WRT ADDR INTO UPPER 2K OF PAGE
00645 103362 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
00646 103227 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
00647 203420 DAC RNSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
00650 040206 TAD (4000
00651 346432 DAC RNSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
00652 040207 LAM -1
00653 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
00654 040213 LAC RNSA15
00655 200206 TAD (-1
00656 346431 DAC 10 /INIT ADDR POINTER FOR 15
00657 040010 JMS READ2K /READ & CHECK LOWER 2K
00660 103373 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
00661 103227 JMS ERRCHK /ERROR?
00662 102625 JMP TST6,A /NO
00663 000665 JMP E1ROU /TYPE TST #, ADDRESS, GOOD & BAD DATA
00664 002712 TST6,A JMS PAGESEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00665 103173 JMP TST6+1
00666 000632 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
00667 002674
/*****
.EJECT

```

```

/*****
/TST7. 15 WRITES IN LOWER 2K, 11 READS & CHECKS IT WHILE 15 IS
/WRITING IN UPPER 2K.
TST7  0
00670 000000      LAC      ADRPNT      /GET FIRST ADDRESS OF CURMENT PAGE
00671 203420      DAC      RNSA15     /STORE FIRST OPERATION ADDRESS FOR PDP-15
00672 040206      DAC      RNSA11     /STORE FIRST OPERATION ADDRESS FOR PDP-11
00673 040207      LAC      (-10000    /INITIALIZE BUFFER
00674 206433      JMS      Ibuff
00675 104106      LAM      -1
00676 777777      DAC      IDNE15     /INDICATE PDP-15 DONE FUNCTION
00677 040213      LAC      RNSA15
00700 200206      TAD      (-1
00701 346431      DAC      10
00702 040010      JMS      WRT2K      /INIT ADDR POINTER FOR 15
00703 103362      LAC      RNSA15     /WRT ADDR INTO LOWER 2K OF PAGE
00704 200206      TAD      (4000
00705 346432      DAC      RNSA15     /STORE FIRST OPERATIWN ADDRESS FOR PDP-15
00706 040206      TAD      (-1
00707 346431      DAC      10
00710 040010      LAM      -1
00711 777777      DAC      IDNE15     /INDICATE PDP-15 DONE FUNCTION
00712 040213      JMS      WRT2K      /WRT UPPER 2K WHILE 11 IS READING
00713 103362      JMS      M11
00714 103227      JMS      ERRCHK
00715 102625      JMP      TST7,A     /NO
00716 000720      JMP      E1R0U     /TYPE TST #, ADDRESS, GOOD & BAD DATA
00717 002712      JMS      PAGESL    /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00720 103173      JMP      TST7*1
00721 000071      JMP      PERCHK    /CHECK FOR A PREVIOUS ERROR CONDITION
00722 002074
/*****
.EJECT

```

```

/*****
/TST10. 15 WRITES IN UPPER 2K, 11 READS & CHECKS IT WHILE 15 IS
/WRITING IN LOWER 2K.
TST10 0
00723 000000      LAC      ADRPNT      /GET FIRST ADDRESS OF CURRENT PAGE
00724 203420      TAD      (4000
00725 346432      DAC      RNSA15     /STORE FIRST OPERATION ADDRESS FOR PDP-15
00726 040206      DAC      RNSA11     /STORE FIRST OPERATION ADDRESS FOR PDP-11
00727 040207      LAC      (-10000    /INITIALIZE BUFFER
00730 206433      JMS      Ibuff
00731 104106      LAM      -1
00732 777777      DAC      IDNE15     /INDICATE PDP-15 DONE FUNCTION
00733 040213      LAC      RNSA15
00734 200206      TAD      (-1
00735 346431      DAC      10
00736 040010      JMS      WRT2K      /INIT ADDR POINTER FOR 15
00737 103362      LAC      RNSA15     /WRT ADDR INTO UPPER 2K OF PAGE
00740 203420      LAC      ADRPNT      /GET FIRST ADDRESS OF CURRENT PAGE
00741 040206      DAC      RNSA11     /STORE FIRST OPERATION ADDRESS FOR PDP-15
00742 346431      TAD      (-1
00743 040010      DAC      10
00744 777777      LAM      -1
00745 040213      DAC      IDNE15     /INDICATE PDP-15 DONE FUNCTION
00746 103362      JMS      WRT2K      /WRT LOWER 2K WHILE 11 IS READING
00747 103227      JMS      M11
00748 102625      JMS      ERRCHK
00751 000753      JMP      TST10A    /NO
00752 002712      JMP      E1R0U     /TYPE TST #, ADDRESS, GOOD & BAD DATA
00753 103173      JMS      PAGESL    /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
00754 000724      JMP      TST10*1
00755 002074      JMP      PERCHK    /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

*****
/TST11. 11 WRITES IN LOWER 2K, 15 READS & CHECKS IT WHILE 11 IS
/WRITING IN UPPER 2K.
TST11  0
00756  000000          LAC  ADRPNT          /GET FIRST ADDRESS OF CURRENT PAGE
00757  203420          DAC  RWSA11          /STORE FIRST OPERATION ADDRESS FOR PDP-11
00760  040207          DAC  RWSA11          /STORE FIRST OPERATION ADDRESS FOR PDP-15
00761  040206          DAC  RWSA15          /STORE FIRST OPERATION ADDRESS FOR PDP-15
00762  206433          LAC  (-10000          /INITIALIZE BUFFER
00763  104100          JMS  IBUFF
00764  777777          LAM  -1
00765  040213          DAC  IDNE15          /INDICATE PDP-15 DONE FUNCTION
00766  200200          LAC  RWSA15
00767  346431          TAD  (-1
00770  040010          DAC  10
00771  103227          JMS  W11            /INIT ADDR POINTER FOR 15
00772  203420          LAC  ADRPNT          /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
00773  346432          TAD  (4000          /GET FIRST ADDRESS OF CURRENT PAGE
00774  040207          DAC  RWSA11          /STORE FIRST OPERATION ADDRESS FOR PDP-11
00775  777777          LAM  -1
00776  040213          DAC  IDNE15          /INDICATE PDP-15 DONE FUNCTION
00777  103375          JMS  READ2K         /READ LOWER 2K WHILE 11 IS WRITING UPPER 2K
01000  103227          JMS  W11            /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01001  102625          JMS  ERRCHK         /ERROR?
01002  001004          JMP  TST11A         /NO
01003  002712          JMP  E1R0U          /TYPE TST #, ADDRESS, GOOD & BAD DATA
01004  103173          TST11A JMS  PAGESEL        /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
01005  000757          JMP  TST11+1
01006  002674          JMP  PERCHK         /CHECK FOR A PREVIOUS ERROR CONDITION
*****
.EJECT

```

```

*****
/TST12. 11 WRITES IN UPPER 2K, 15 READS & CHECKS IT WHILE 11 IS
/WRITING IN LOWER 2K.
TST12  0
01007  000000          LAC  ADRPNT          /GET FIRST ADDRESS OF CURRENT PAGE
01010  203420          DAC  RWSA11          /STORE FIRST OPERATION ADDRESS FOR PDP-11
01011  346432          TAD  (4000          /STORE FIRST OPERATION ADDRESS FOR PDP-15
01012  040207          DAC  RWSA15          /STORE FIRST OPERATION ADDRESS FOR PDP-15
01013  040206          DAC  RWSA15          /STORE FIRST OPERATION ADDRESS FOR PDP-15
01014  206433          LAC  (-10000          /INITIALIZE BUFFER
01015  104100          JMS  IBUFF
01016  777777          LAM  -1
01017  040213          DAC  IDNE15          /INDICATE PDP-15 DONE FUNCTION
01020  200200          LAC  RWSA15
01021  346431          TAD  (-1
01022  040010          DAC  10
01023  103227          JMS  W11            /INIT ADDR POINTER FOR 15
01024  203420          LAC  ADRPNT          /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01025  040207          DAC  RWSA11          /GET FIRST ADDRESS OF CURRENT PAGE
01026  777777          LAM  -1
01027  040213          DAC  IDNE15          /STORE FIRST OPERATION ADDRESS FOR PDP-11
01028  103373          JMS  READ2K         /INDICATE PDP-15 DONE FUNCTION
01031  103227          JMS  W11            /READ UPPER 2K WHILE 11 IS WRITING LOWER 2K
01032  102625          JMS  ERRCHK         /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01033  001005          JMP  TST12A         /ERROR?
01034  002712          JMP  E1R0U          /NO
01035  103173          TST12A JMS  PAGESEL        /TYPE TST #, ADDRESS, GOOD & BAD DATA
01036  001010          JMP  TST12+1        /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
01037  002674          JMP  PERCHK         /CHECK FOR A PREVIOUS ERROR CONDITION
*****
.EJECT

```

```

/*****
/TST13, 11 WRITES ALL ONES IN LOW ORDER BYTES OF LOWER 2K WHILE 15
/IS WRITING -1 IN UPPER 2K, 15 READS & CHECKS LOWER 2K WHILE 11 IS
/READING & CHECKING LOW ORDER BYTES IN UPPER 2K.
TST13 0
01040 000000 LAM -10000
01041 770000 DAC CNT3 /INITIALIZE COUNT
01042 043451 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
01043 203420 DAC RWSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
01044 040207 TAD (4000
01045 346432 DAC RWSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
01046 040206 LAC RWSA11
01047 200207 TAD (-1
01050 346431 DAC 10
01051 040010 DZM+ 10 /INIT BUFF FOR 11
01052 160010 ISZ CNT3
01053 443451 JMP -2
01054 601052 LAM -1
01055 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
01056 040213 LAC RWSA15
01057 200206 TAD (-1
01060 346431 DAC 10
01061 040010 LAM -4000 /INITIALIZE COUNT
01062 774000 DAC WCNT
01063 043417 LAM -1
01064 777777 DAC+ 10
01065 060010 ISZ WCN1
01066 443417 JMP -3
01067 601064 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01070 103227 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
01071 203420 DAC RWSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
01072 040206 TAD (4000
01073 346432 DAC RWSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
01074 040207 LAM -1
01075 777777 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
01076 040213 LAC RWSA15
01077 200206 TAD (-1
01100 346431 DAC 10
01101 040010 LAM -4000
01102 774000 DAC RCNT /INITIALIZE COUNT
01103 043416 TST13A LAC+ 10
01104 220010 DAC TBAD
01105 043422 SAD (377 /DATA CORRECT?
01106 546434 JMP TST13B /YES
01107 601116 LAC (377
01110 206434 DAC TGOOD
01111 043433 LAM -1
01112 777777 DAC ERRIND /SET PREVIOUS ERROR INDICATOR
01113 040210 JMS EFCHK /CHECK ERRFLG TO DETERMINE WHETHER OR NOT TO STORE DATA
01114 103011 JMP TST13C
01115 601120 TST13B ISZ RCNT /DONE?
01116 443410 JMP TST13A /NO
01117 601104 TST13C JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01120 103227 JMS ERRCHK /ERROR?
01121 102620 JMP TST13D /NO
01122 601124

```

```

01123 602712 JMP EIRU /TYPE TST #, ADDRESS, GOOD & BAD DATA
01124 103173 TST13D JMS PAGSEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
01125 601041 JMP TST13+1
01126 602674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/*****
/TST14. 11 WRITES ALL ONES IN HIGH ORDER BYTES OF LOWER 2K WHILE
/15 IS WRITING -1 IN UPPER 2K, 15 HEADS & CHECKS LOWER 2K WHILE 11 IS
/READING & CHECKING HIGH ORDER BYTES IN UPPER 2K.
TST14 0
01127 000000 LAM -10000
01130 770000 DAC CNT3 /INITIALIZE COUNT
01131 043451 DAC CNT3 /GET FIRST ADDRESS OF CURRENT PAGE
01132 203420 LAC ADDRPT /STORE FIRST OPERATION ADDRESS FOR PDP-11
01133 000207 DAC RWSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-15
01134 346432 TAD (4000
01135 000206 DAC RWSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
01136 200207 LAC RWSA11
01137 346431 TAD (-1
01140 000010 DAC 10
01141 100010 DZM* 10 /INIT BUFF FOR 11
01142 443451 ISZ CNT3
01143 001141 JMP ,-2
01144 777777 LAM -1
01145 000213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
01146 200206 LAC RWSA15
01147 346431 TAD (-1
01150 000010 DAC 10
01151 774000 LAM =4000
01152 043417 DAC RCNT /INITIALIZE COUNT
01153 777777 LAM -1
01154 000010 DAC* 10
01155 443417 ISZ RCNT
01156 001153 JMP ,-3
01157 103227 JMS M11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01160 203420 LAC ADDRPT /GET FIRST ADDRESS OF CURRENT PAGE
01161 000206 DAC RWSA15 /STORE FIRST OPERATION ADDRESS FOR PDP-15
01162 346432 TAD (4000
01163 000207 DAC RWSA11 /STORE FIRST OPERATION ADDRESS FOR PDP-11
01164 777777 LAM -1
01165 000213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
01166 200206 LAC RWSA15
01167 346431 TAD (-1
01170 000010 DAC 10
01171 774000 LAM =4000
01172 043416 DAC RCNT /INITIALIZE COUNT
01173 220010 TST14A LAC* 10
01174 043422 DAC T8AD
01175 546435 SAD (177400 /DATA CORRECT?
01176 001205 JMP TST14B /YES
01177 206435 LAC (177400
01200 043433 DAC T6000
01201 777777 LAM -1
01202 000210 DAC ERRIND /SET PREVIOUS ERROR INDICATOR
01203 103011 JMS EFCHK /CHECK ERRFLG TO DETERMINE WHETHER OR NOT TO STORE DATA
01204 001207 JMP TST14C
01205 443416 TST14B ISZ RCNT /DONE?
01206 001173 JMP TST14A /NO
01207 103227 TST14C JMS M11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
01210 102025 JMS ERRCHK /ERROR?
01211 001213 JMP TST14D

```

```

01212 002712 JMP E1R0U /TYPE TST #, ADDRESS, GOOD & BAD DATA
01213 103173 TST14D JMS PAGSEL /SEL NXT MEM PAGE FOR TESTING (IF ANY LEFT)
01214 001130 JMP TST14+1
01215 002674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```



```

/*****
/TST15. TEST FOR TCBP FLAG TO BE SET BY LIOR.
TST15  0
01216  000000  XCT  LIOR          /LOAD TCBP, SET TCBP FLG IN PDP-11
01217  406370  JMS  I15W11       /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01220  103222  JMS  ERRCHK       /ERROR?
01221  102625  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01222  602674  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01223  102736  MES6
01224  004222

/*****
/TST16. TEST FOR DATI FROM PDP-11 LOC 167764 TO CLEAR TCBP FLG.
TST16  0
01225  000000  XCT  LIOR          /LOAD TCBP, SET TCBP FLG IN PDP-11
01226  406370  JMS  I15W11       /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01227  103222  JMS  ERRCHK       /ERROR?
01230  102625  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01231  602674  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01232  102736  MES7
01233  004236

/*****
/TST17. TEST FOR DATI FROM PDP-11 LOC 167774 TO NOT CLEAR TCBP FLG.
TST17  0
01234  000000  XCT  LIOR          /LOAD TCBP, SET TCBP FLG IN PDP-11
01235  406370  JMS  I15W11       /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01236  103222  JMS  ERRCHK       /ERROR?
01237  102625  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01240  602674  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01241  102736  MES8
01242  005203

/*****
/TST20. TEST FOR TCBP ACCEPTED WITH SIOA IOT.
TST20  0
01243  000000  XCT  SIOA          /SKIP?
01244  406366  SKP  /NO
01245  741000  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01246  602674  JMS  IERR         /INDICATE AN ERROR!
01247  102620  JMS  ERRCHK       /ERROR?
01250  102625  NOP
01251  740000  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01252  102736  MES8
01253  004260

/*****
/TST21. CLEAR TCBP ACCEPTED FLG WITH CIOD IOT & TEST SIOA FOR NO SKIP.
TST21  0
01254  000000  XCT  CIOD          /CLEAR TCBP ACCEPTED FLG
01255  406367  XCT  SIOA          /NO SKIP?
01256  406366  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01257  602674  JMS  IERR         /INDICATE AN ERROR!
01260  102620  JMS  ERRCHK       /ERROR?
01261  102625  NOP
01262  740000  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01263  102736  MES9
01264  004306

*****
.EJECT

```

```

/*****
/TST22. TEST FOR TCBP ACCEPTED TO BE SET BY DOING DATI TO LOC 167764
/IN PDP-11.
TST22  0
01265  000000  XCT  CIOD          /CLEAR TCBP ACCEPTED FLG
01266  406367  XCT  LIOR          /LOAD TCBP, SET TCBP FLG IN PDP-11
01267  406370  JMS  I15W11       /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01270  103222  XCT  SIOA          /SKIP?
01271  406366  SKP  /NO
01272  741000  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01273  602674  JMS  IERR         /INDICATE AN ERROR!
01274  102620  JMS  ERRCHK       /ERROR?
01275  102625  NOP
01276  740000  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01277  102736  MES6
01300  006130

/*****
/TST23. TEST FOR TCBP ACCEPTED NOT TO BE SET BY DOING DATI TO
/LOC 167774 IN PDP-11.
TST23  0
01301  000000  XCT  CIOD          /CLEAR TCBP ACCEPTED FLG
01302  406367  XCT  LIOR          /LOAD TCBP, SET TCBP FLG IN PDP-11
01303  406370  JMS  I15W11       /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01304  103222  XCT  SIOA          /NO SKIP?
01305  406366  JMP  PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01306  602674  JMS  IERR         /INDICATE AN ERROR!
01307  102620  JMS  ERRCHK       /ERROR?
01310  102625  NOP
01311  740000  JMS  E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
01312  102736  MES6
01313  006165

*****
.EJECT

```

```

/*****
/TST24. TESTS FOR CORRECT TRANSMISSION OF TCBP FROM 15 TO 11.
TST24  #
01314  #0000#          DZM      TG000      /SOFT TCBP
01315  143433        LAM      -1
01316  777777        DAC      IDNE15     /INDICATE PDP-15 DONE FUNCTION
01317  #00213        XCT      CI00      /CLEAR TCBP ACCEPTED FLG
01320  486367        LAC      TG000
01321  2#3433        XCT      LIOR      /LOAD TCBP, SET TCBP FLG IN PDP-11
01322  48637#        CLLIRAL  /MAKE AN 11 ADDRESS
01323  744#1#        DAC      GOOD
01324  #002#3        LAC      IDNE11     /GET PDP-11 DONE FUNCTION INDICATOR
01325  2#0214        SAD      (177777
01326  546412        JMP      TST24C
01327  6#1364        XCT      SIDA      /TCBP ACCEPTED?
01330  486366        JMP      #4        /NO
01331  8#1325        LAC      RWSA15     /GET BITS 0 & 1 (NOW BITS 16 & 17)
01332  2#02#6        CLLIRAR  /MOVE INTO CORRECT POSITION
01333  744#2#        RTR      /GET BITS 2-17
01334  742#2#        TAD      RWSA11     /CORRECT?
01335  34#2#7        SAD      GOOD
01336  54#2#3        JMP      TST24B     /YES
01337  6#1355        DAC      BAD        /NO
0134#  #02#4        JMS      IERR      /INDICATE AN ERROR!
01341  1#2#2#        JMS      ERRCHK     /ERROR?
01342  1#2#25        NUP
01343  74#0#0#        JMS      HDR2      /TYPE "TST- GOOD - BAD" HEADER
01344  1#277#        LAC      GOOD
01345  2#02#3        JMS      OCT
01346  1#4#13        JMS      TYP
01347  1#371#        LAC      BAD
0135#  2#02#4        JMS      OCT
01351  1#4#13        JMS      CRLF      /TYPE CARRIAGE RETURN & LINE FEED
01352  1#3#43        JMS      PNETST
01353  1#3753        JMP      BP
01354  6#2#43        TST24B  ISZ      TG000     /INC SOFT TCBP, DONE?
01355  434333        LAC      TG000
01356  2#3433        SAD      (4#0#0#
01357  546436        SKP
0136#  741#0#0#        JMP      TST24A     /DONE TEST?
01361  6#132#        LAM      -1        /NO
01362  777777        DAC      IDNE15     /INDICATE PDP-15 DONE FUNCTION
01363  #00213        TST24C  JMP      PERCHK     /CHECK FOR A PREVIOUS ERROR CONDITION
01364  6#2#674
/*****
.EJECT

```

```

/*****
/TST25. TEST FOR API 0 DONE FLG TO BE SET.
TST25  #
01365  #0000#          JMS      I15#11     /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01366  1#3222        JMS      ERRCHK     /ERROR?
0137#  1#2#25        JMP      PERCHK     /CHECK FOR A PREVIOUS ERROR CONDITION
0137#  6#2#74        JMS      E2ROU     /TYPE TST # AND AN ERROR DESCRIPTION
01372  1#2736        MES11
/*****
/TST26. TEST FOR API 1 DONE FLG TO BE SET.
TST26  #
01373  #0000#          JMS      I15#11     /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01374  1#3222        JMS      ERRCHK     /ERROR?
01375  1#2#25        JMP      PERCHK     /CHECK FOR A PREVIOUS ERROR CONDITION
01376  6#2#74        JMS      E2ROU     /TYPE TST # AND AN ERROR DESCRIPTION
01377  1#2736        MES12
014#  #0435#
/*****
/TST27. TEST FOR API 2 DONE FLAG TO BE SET.
TST27  #
014#  #0000#          JMS      I15#11     /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
014#  1#3222        JMS      ERRCHK     /ERROR?
014#  1#2#25        JMP      PERCHK     /CHECK FOR A PREVIOUS ERROR CONDITION
014#  6#2#74        JMS      E2ROU     /TYPE TST # AND AN ERROR DESCRIPTION
014#  1#2736        MES13
014#  #0437#
/*****
/TST30. TEST FOR API 3 DONE FLAG TO BE SET.
TST3#  #
014#  #0000#          JMS      I15#11     /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
0141#  1#3222        JMS      ERRCHK     /ERROR?
01411  1#2#25        JMP      PERCHK     /CHECK FOR A PREVIOUS ERROR CONDITION
01412  6#2#74        JMS      E2ROU     /TYPE TST # AND AN ERROR DESCRIPTION
01413  1#2736        MES14
01414  #044#6
/*****
/TST31. TEST FOR API DONE FLG TO BE SET.
TST31  #
01415  #0000#          JMS      I15#11     /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01416  1#3222        JMS      ERRCHK     /ERROR?
01417  1#2#25        JMP      PERCHK     /CHECK FOR A PREVIOUS ERROR CONDITION
0142#  6#2#74        JMS      E2ROU     /TYPE TST # AND AN ERROR DESCRIPTION
01421  1#2736        MES15
01422  #04422
/*****
.EJECT

```

```

/*****
/TST32. TEST FOR SAPI0 TO SKIP.
TST32 0
01423 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01424 103222 XCT SAPI0 /SKIP ON API LEVEL 0 FLG SET
01425 406373 SKP /CHECK FOR A PREVIOUS ERROR CONDITION
01426 741000 JMP PERCHK /INDICATE AN ERROR!
01427 602674 JMS IERR /ERROR?
01430 102620 JMS ERRCHK /ERROR?
01431 102620 NOP
01432 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01433 102736 MES29
01434 005020
/*****
/TST33. TEST FOR SAPI1 TO SKIP.
TST33 0
01435 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01436 103222 XCT SAPI1 /SKIP ON API LEVEL 1 FLG SET
01437 406374 SKP /CHECK FOR A PREVIOUS ERROR CONDITION
01440 741000 JMP PERCHK /INDICATE AN ERROR!
01441 602674 JMS IERR /ERROR?
01442 102620 JMS ERRCHK /ERROR?
01443 102620 NOP
01444 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01445 102736 MES30
01446 005030
/*****
/TST34. TEST FOR SAPI2 TO SKIP.
TST34 0
01447 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01450 103222 XCT SAPI2 /SKIP ON API LEVEL 2 FLG SET
01451 406375 SKP /CHECK FOR A PREVIOUS ERROR CONDITION
01452 741000 JMP PERCHK /INDICATE AN ERROR!
01453 602674 JMS IERR /ERROR?
01454 102620 JMS ERRCHK /ERROR?
01455 102620 NOP
01456 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01457 102736 MES31
01460 005040
/*****
/TST35. TEST FOR SAPI3 TO SKIP.
TST35 0
01461 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01462 103222 XCT SAPI3 /SKIP ON API LEVEL 3 FLG SET
01463 406376 SKP /CHECK FOR A PREVIOUS ERROR CONDITION
01464 741000 JMP PERCHK /INDICATE AN ERROR!
01465 602674 JMS IERR /ERROR?
01466 102620 JMS ERRCHK /ERROR?
01467 102620 NOP
01470 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01471 102736 MES32
01472 005050
.EJECT

```

```

/*****
/TST36. TEST SAPI0 FOR NO SKIP.
TST36 0
01473 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01474 103222 XCT SAPI0 /SKIP ON API LEVEL 0 FLG SET
01475 406373 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01476 602674 JMS IERR /INDICATE AN ERROR!
01477 102620 JMS ERRCHK /ERROR?
01500 102620 NOP
01501 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01502 102736 MES25
01503 004700
/*****
/TST37. TEST SAPI1 FOR NO SKIP.
TST37 0
01504 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01505 103222 XCT SAPI1 /SKIP ON API LEVEL 1 FLG SET
01506 406374 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01507 602674 JMS IERR /INDICATE AN ERROR!
01510 102620 JMS ERRCHK /ERROR?
01511 102620 NOP
01512 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01513 102736 MES26
01514 004724
/*****
/TST40. TEST SAPI2 FOR NO SKIP.
TST40 0
01515 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01516 103222 XCT SAPI2 /SKIP ON API LEVEL 2 FLG SET
01517 406375 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01520 602674 JMS IERR /INDICATE AN ERROR!
01521 102620 JMS ERRCHK /ERROR?
01522 102620 NOP
01523 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01524 102736 MES27
01525 004750
/*****
/TST41. TEST SAPI3 FOR NO SKIP.
TST41 0
01526 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01527 103222 XCT SAPI3 /SKIP ON API LEVEL 3 FLG SET
01530 406376 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01531 602674 JMS IERR /INDICATE AN ERROR!
01532 102620 JMS ERRCHK /ERROR?
01533 102620 NOP
01534 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01535 102736 MES28
01536 004774
.EJECT

```

```

/*****
/TST42. TEST CAPI0 TO CLEAR API0 FLG.
TST42  0
01537  000000
01540  103222      JMS      I15W11      /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01541  406377      XCT      CAPI0        /CLR FLG
01542  103354      JMS      TAPI         /TEST API?
01543  601506      JMP      TST42A       /NO
01544  705512      RPL                     /HEAD API STATUS
01545  040204      DAC      BAD
01546  546437      SAD      (70000      /0 AND ONLY 0 CLEARED?
01547  602674      JMP      PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01500  102620      JMS      IERR        /INDICATE AN ERROR!
01551  102625      JMS      ERRCHK      /ERROR?
01552  740000      NOP
01553  206437      LAC      (70000
01554  040203      DAC      GOOD
01555  602747      JMP      E3ROU       /TYPE TST #, GOOD AND BAD DATA
01556  406373      TST42A XCT      SAPI0        /0 CLRED?
01557  602674      JMP      PERCHK      /YES
01500  102620      JMS      IERR
01561  102625      JMS      ERRCHK
01562  740000      NOP
01563  102736      JMS      E2ROU
01564  004552      MES20
/*****
/*****
/TST43. TEST CAPI1 TO CLEAR API1 FLG.
TST43  0
01565  000000
01566  103222      JMS      I15W11      /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01567  406400      XCT      CAPI1        /CLR FLG
01570  103354      JMS      TAPI         /TEST API?
01571  601604      JMP      TST43A       /NO
01572  705512      RPL                     /HEAD API STATUS
01573  040204      DAC      BAD
01574  546440      SAD      (130000     /1 AND ONLY 1 CLEARED?
01575  602674      JMP      PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01576  102620      JMS      IERR        /INDICATE AN ERROR!
01577  102625      JMS      ERRCHK      /ERROR?
01600  740000      NOP
01601  206440      LAC      (130000
01602  040203      DAC      GOOD
01603  602747      JMP      E3ROU       /TYPE TST #, GOOD AND BAD DATA
01604  406374      TST43A XCT      SAPI1        /1 CLRED?
01605  602674      JMP      PERCHK      /YES
01606  102620      JMS      IERR
01607  102625      JMS      ERRCHK
01610  740000      NOP
01611  102736      JMS      E2ROU
01612  004612      MES22
/*****
.EJECT

```

```

/*****
/TST44. TEST CAPI2 TO CLEAR API2 FLG.
TST44  0
01613  000000
01614  103222      JMS      I15W11      /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01615  406401      XCT      CAPI2        /CLR FLG
01616  103354      JMS      TAPI         /TEST API?
01617  601632      JMP      TST44A       /NO
01620  705512      RPL                     /HEAD API STATUS
01621  040204      DAC      BAD
01622  546441      SAD      (150000     /2 AND ONLY 2 CLEARED?
01623  602674      JMP      PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
01624  102620      JMS      IERR        /INDICATE AN ERROR!
01625  102625      JMS      ERRCHK      /ERROR?
01626  740000      NOP
01627  206441      LAC      (150000
01630  040203      DAC      GOOD
01631  602747      JMP      E3ROU       /TYPE TST #, GOOD AND BAD DATA
01632  406375      TST44A XCT      SAPI2        /2 CLRED?
01633  602674      JMP      PERCHK      /YES
01634  102620      JMS      IERR
01635  102625      JMS      ERRCHK
01636  740000      NOP
01637  102736      JMS      E2ROU
01640  005506      MES49
/*****
.EJECT

```

```

/*****
/TST45. TEST CAPI3 TO CLEAR API3 FLG.
TST45 0
01641 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01642 103222 XCT CAPI3 /CLR FLG
01643 405442 JMS TAPI /TEST API?
01644 103354 JMP TST45A /NO
01645 601660 RPL TST45A /HEAD API STATUS
01646 705512 DAC BAD
01647 040204 SAD (160000 /3 AND ONLY 3 CLEARED?
01650 546442 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01651 602674 JMS IERR /INDICATE AN ERROR!
01652 102620 JMS ERRCHK /ERROR?
01653 102625 NOP
01654 740000 LAC (160000
01655 206442 DAC GOOD
01656 040203 JMP E3ROU /TYPE TST #, GOOD AND BAD DATA
01657 602747 TST45A XCT SAPI3 /3 CLR'D?
01660 406376 JMP PERCHK /YES
01661 602674 JMS IERR
01662 102620 JMS ERRCHK
01663 102625 NOP
01664 740000 JMS E2ROU
01665 102736 MES68
01666 006153
/*****
/TST46. TEST FOR DR15 INT ENABLE SET USING RDRS IOT.
TST46 0
01667 000000 XCT RDRS
01670 406371 AND (1
01671 506443 SZA /INT ENABLE SET?
01672 740200 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01673 602674 JMS IERR /INDICATE AN ERROR!
01674 102620 JMS ERRCHK /ERROR?
01675 102625 NOP
01676 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01677 102736 MES33
01700 005060
/*****
.EJECT

```

```

/*****
/TST47. TEST FOR DR15 INT ENABLE BEING CLEARED BY LDRS IOT.
TST47 0
01701 000000 CLA
01702 750000 XCT LDRS /CLR INT ENABLE
01703 406372 XCT RDRS /READ DR STAT
01704 406371 SNA /INT ENABLE CLR?
01705 741200 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01706 602674 JMS IERR /INDICATE AN ERROR!
01707 102620 JMS ERRCHK /ERROR?
01710 102625 NOP
01711 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01712 102736 MES34
01713 005105
/*****
/TST50. TEST FOR DR15 INT ENABLE BEING SET BY LDRS IOT.
TST50 0
01714 000000 LAC (1
01715 206443 XCT LDRS /SET INT ENABLE
01716 406372 XCT RDRS /READ DR STAT
01717 406371 SAD (1 /INT ENABLE SET?
01720 546443 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01721 602674 JMS IERR /INDICATE AN ERROR!
01722 102620 JMS ERRCHK /ERROR?
01723 102625 NOP
01724 740000 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01725 102736 MES16
01726 004435
/*****
/TST51. TEST FOR CORRECT API LEVEL REQUEST WHEN 11 INDICATES LEVEL 0.
TST51 0
01727 000000 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
01730 103354 JMP TST51
01731 621727 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01732 103222 RPL TST51 /HEAD API STATUS
01733 705512 DAC BAD
01734 040204 SAD (100000 /0 REQUEST, ONE AND ONLY?
01735 546444 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01736 602674 JMS IERR /INDICATE AN ERROR!
01737 102620 JMS ERRCHK /ERROR?
01740 102625 NOP
01741 740000 LAC (100000
01742 206444 DAC GOOD
01743 040203 JMP E3ROU /TYPE TST #, GOOD AND BAD DATA
01744 602747
/*****
/TST52. TEST FOR API LEVEL 0 DONE FLG (ONLY) TO CLR.
TST52 0
01745 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01746 103222 JMS ERRCHK /ERROR?
01747 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01750 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
01751 102736 MES17
01752 004453
/*****
.EJECT

```

```

/*****
/TST53. TEST FOR CORRECT API LEVEL REQUEST WHEN 11 INDICATES LEVEL 1.
TST53 0
01753 000000 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
01754 103354 JMP+ TST53
01755 621753 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01756 103222 RPL /READ API STATUS
01757 705512 DAC BAD
01760 040204 SAD (40000 /1 REQUEST, ONE AND ONLY?
01761 546424 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01762 602674 JMS IERR /INDICATE AN ERROR!
01763 102020 JMS ERRCHK /ERROR?
01764 102625 NOP
01765 740000 LAC (40000
01766 206424 DAC GOOD
01767 040203 JMP E3R0U /TYPE TST #, GOOD AND BAD DATA
01770 602747

/*****
/TST54. TEST FOR API LEVEL 1 DONE FLG (ONLY) TO CLR.
TST54 0
01771 000000 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
01772 103222 JMS ERRCHK /ERROR?
01773 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
01774 602674 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
01775 102736 MES10
01776 004524

/*****
/TST55. TEST FOR CORRECT API LEVEL REQUEST WHEN 11 INDICATES LEVEL 2.
TST55 0
02000 000000 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
02001 103354 JMP+ TST55
02002 621777 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02003 103222 RPL /READ API STATUS
02004 705512 DAC BAD
02005 040204 SAD (20000 /2 REQUEST, ONE AND ONLY?
02006 546422 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02007 602674 JMS IERR /INDICATE AN ERROR!
02008 102020 JMS ERRCHK /ERROR?
02009 102625 NOP
02010 740000 LAC (20000
02011 546422 DAC GOOD
02012 206422 JMP E3R0U /TYPE TST #, GOOD AND BAD DATA
02013 040203

/*****
/TST56. TEST FOR API LEVEL 2 DONE FLG (ONLY) TO CLR.
TST56 0
02015 000000 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02016 103222 JMS ERRCHK /ERROR?
02017 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02018 602674 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02019 102736 MES21
02022 004564

.EJECT

```

```

/*****
/TST57. TEST FOR CORRECT API LEVEL REQUEST WHEN 11 INDICATES LEVEL 3.
TST57 0
02023 000000 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
02024 103354 JMP+ TST57
02025 622023 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02026 103222 RPL /READ API STATUS
02027 705512 DAC BAD
02030 040204 SAD (10000 /3 REQUEST, ONE AND ONLY?
02031 546430 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02032 602674 JMS IERR /INDICATE AN ERROR!
02033 102020 JMS ERRCHK /ERROR?
02034 102625 NOP
02035 740000 LAC (10000
02036 206430 DAC GOOD
02037 040203 JMP E3R0U /TYPE TST #, GOOD AND BAD DATA
02040 602747

/*****
/TST60. TEST FOR API LEVEL 3 DONE FLG (ONLY) TO CLR.
TST60 0
02041 000000 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02042 103222 JMS ERRCHK /ERROR?
02043 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02044 602674 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02045 102736 MES23
02046 004024

/*****
/TST61. TEST FOR API DONE FLAG TO BE CLR WHEN API LEVEL 0
/MAS A REQUEST PENDING.
TST61 0
02047 000000 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02048 103222 JMS ERRCHK /ERROR?
02049 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02050 602674 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02051 102736 MES24
02054 004652

/*****
/TST62. TEST FOR API DONE FLAG TO BE CLR WHEN API LEVEL 1
/MAS A REQUEST PENDING.
TST62 0
02055 000000 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02056 103222 JMS ERRCHK /ERROR?
02057 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02058 602674 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02059 102736 MES45
02062 005375

/*****
/TST63. TEST FOR API DONE FLAG TO BE CLR WHEN API LEVEL 2
/MAS A REQUEST PENDING.
TST63 0
02063 000000 JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02064 103222 JMS ERRCHK /ERROR?
02065 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02066 602674 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02067 102736 MES46
02070 005423

.EJECT

```

```

/*****
/TS164. TEST FOR API DONE FLAG TO BE CLR WHEN API LEVEL 3
/MAS A REQUEST PENDING.
TST64 0
02071 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02072 103222 JMS ENRCHK /ERROR?
02073 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02074 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02075 102736 MES47
02076 005451
/*****
/TS165. TEST API DONE INT ENABLE TO SET.
TST65 0
02107 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02108 103222 JMS ENRCHK /ERROR?
02109 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02110 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02111 102736 MES39
02112 005222
/*****
/TS166. TEST API DONE INT ENABLE TO CLEAR.
TST66 0
02105 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02106 103222 JMS ENRCHK /ERROR?
02107 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02108 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02109 102736 MES40
02110 005240
/*****
/TS167. CLEAR API DONE ENABLE AND TEST FOR NO PDP-11 INTERRUPT
/FRM API DONE FLAG.
TST67 0
02113 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02114 103222 JMS ENRCHK /ERROR?
02115 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02116 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02117 102736 MES44
02118 005347
/*****
/TS170. CLEAR TCBP ENABLE AND TEST FOR NO PDP-11 INTERRUPT
/FRM TCBP FLAG.
TST70 0
02121 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02122 103222 XLT LIOR /LOAD TCBP, SET TCBP FLG IN PDP-11
02123 406370 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02124 103222 JMS ENRCHK /ERROR?
02125 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02126 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02127 102736 MES43
02128 005321
/*****
.EJECT

```

```

/*****
/TS171. TEST FOR API DONE TO CAUSE A PDP-11 INT TO CORRECT VECTOR
/ADDRESS.
TST71 0
02131 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02132 103222 JMS ENRCHK /ERROR?
02133 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02134 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02135 102736 MES41
02136 005256
/*****
/TS172. TEST FOR TCBP TO CAUSE A PDP-11 INT TO CORRECT VECTOR
/ADDRESS.
TST72 0
02137 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02138 103222 XLT LIOR /LOAD TCBP, SET TCBP FLG IN PDP-11
02139 406370 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02140 103222 JMS ENRCHK /ERROR?
02141 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02142 602674 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02143 102736 MES42
02144 005300
/*****
/TS173. TEST FOR API DONE TO CAUSE A PDP-11 INT AT CORRECT BR
/LEVEL.
TST73 0
02147 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02148 103222 JMS ENRCHK /ERROR?
02149 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02150 602155 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02151 102736 MES56
02152 005660
02153 103222 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02154 102625 JMS ENRCHK /ERROR?
02155 602674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02156 102736 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02157 005700 MES57
02158 005742
/*****
/TS174. TEST FOR TCBP TO CAUSE A PDP-11 INT AT CORRECT BR LEVEL.
TST74 0
02162 000000 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02163 103222 XLT LIOR /LOAD TCBP, SET TCBP FLG IN PDP-11
02164 406370 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02165 103222 JMS ENRCHK /ERROR?
02166 102625 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02167 602172 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02168 102736 MES58
02169 005723
02170 103222 JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02171 102625 JMS ENRCHK /ERROR?
02172 602674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02173 102736 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02174 005742 MES59
02175 005742
/*****
.EJECT

```

```

/*****
/IST75. CLEAR DR15 INT ENABLE AND TEST FOR NO PDP-15 INT FROM DR11
/API0 FLG.
02177 000000 TST75 0
02200 777777 LAM -100
02201 043452 DAC CNT4 /INITIALIZE COUNT
02202 750000 CLA /TURN PROGRAM INT ON
02203 406372 XCT LDRE /CLR INT ENABLE
02204 206445 LAC (JMP TST75A
02205 040001 DAC 1 /INIT RETURN
02206 700042 ION /TURN PROGRAM INT ON
02207 777777 LAM -1
02210 040213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
02211 443452 ISZ CNT4
02212 002211 JMP --1
02213 700002 IOF /TURN PROGRAM INT OFF
02214 103227 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02215 002674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
02216 102620 TST75A JMS IERR /INDICATE AN ERROR!
02217 103227 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02220 102625 JMS ERRCHK /ERROR?
02221 740000 NOP
02222 102736 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02223 005704 MES68
/*****
/ST76. TEST FOR DR11 API0 FLG TO CAUSE PDP-15 INT.
02224 000000 TST76 0
02225 206440 LAC (JMP TST76B
02226 040001 DAC 1 /INIT RETURN
02227 143452 DZM CNT4 /INITIALIZE COUNT
02230 700042 ION /TURN PROGRAM INT ON
02231 777777 LAM -1
02232 040213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
02233 443452 ISZ CNT4
02234 002233 JMP --1
02235 700002 IOF /TURN PROGRAM INT OFF
02236 103227 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02237 102620 TST76A JMS IERR /INDICATE AN ERROR!
02240 102625 JMS ERRCHK /ERROR?
02241 740000 NOP
02242 102736 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02243 006010 MES61
02244 103227 TST76B JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02245 406373 XCT SAPI0 /SKIP ON API LEVEL 0 FLG SET
02246 002237 JMP TST76A /NO
02247 002674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/*****
/ST77. TEST FOR DR11 API1 FLG TO CAUSE PDP-15 INT.
02250 000000 TST77 0
02251 206447 LAC (JMP TST77B
02252 040001 DAC 1 /INIT RETURN
02253 143452 DZM CNT4 /INITIALIZE COUNT
02254 700042 ION /TURN PROGRAM INT ON
02255 777777 LAM -1
02256 040213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
02257 443452 ISZ CNT4
02260 002257 JMP --1
02261 700002 IOF /TURN PROGRAM INT OFF
02262 103227 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02263 102620 TST77A JMS IERR /INDICATE AN ERROR!
02264 102625 JMS ERRCHK /ERROR?
02265 740000 NOP
02266 102736 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02267 006027 MES62
02270 103227 TST77B JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02271 406374 XCT SAPI1 /SKIP ON API LEVEL 1 FLG SET
02272 002263 JMP TST77A /NO
02273 002674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
/ST100. TEST FOR DR11 API2 FLG TO CAUSE PDP-15 INT.
02274 000000 ST100 0
02275 206450 LAC (JMP ST100B
02276 040001 DAC 1
02277 143452 DZM CNT4 /INITIALIZE COUNT
02300 700042 ION /TURN PROGRAM INT ON
02301 777777 LAM -1
02302 040213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
02303 443452 ISZ CNT4
02304 002303 JMP --1
02305 700002 IOF /TURN PROGRAM INT OFF
02306 103227 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02307 102620 ST100A JMS IERR /INDICATE AN ERROR!
02310 102625 JMS ERRCHK /ERROR?
02311 740000 NOP
02312 102736 JMS E2R0U /TYPE TST # AND AN ERROR DESCRIPTION
02313 006046 MES63
02314 103227 ST100B JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
02315 406373 XCT SAPI2 /SKIP ON API LEVEL 2 FLG SET
02316 002307 JMP ST100A /NO
02317 002674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```



```

/*****
/ST101. TEST FOR DR11 API3 FLG TO CAUSE PDP-15 INT.
ST101  H
      LAC      (JMP ST101B
      DAC      1
      DZM      CNT4          /INITIALIZE COUNT
      ION      /TURN PROGRAM INT ON
      LAW      -1
      DAC      IONE15        /INDICATE PDP-15 DONE FUNCTION
      ISZ      CNT4
      JMP      .-1
      IOF      /TURN PROGRAM INT OFF
      JMS      W11           /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
ST101A JMS      IERR        /INDICATE AN ERROR!
      JMS      ERRCHK       /ERROR?
      NDP
      JMS      E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
      MES44
ST101B JMS      W11           /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
      XCT      SAPI3        /SKIP ON API LEVEL 3 FLG SET
      JMP      ST101A       /NO
      JMP      PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/*****
/ST102. TEST FOR API BREAKS TO CORRECT PORTS FROM 0 TO 177 FROM
/DR11 APIL0
ST102  H
      JMS      TAPI         /DETERMINE WHETHER OR NOT TO TEST API
      JMP      ST102
      JMS      IAPORT       /INIT API PORT ADDRESSES
      LAW      -200
      DAC      TPCNT        /INITIALIZE COUNT
      DZM      PORT
ST102A JMS      I15W11      /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
      DZM      CNT5        /INITIALIZE COUNT
      LAC      (400000
      ISA
      /INITIATE SELECTED ACTIVITY (API)
      SPA
      JMP      ST102B
      ISZ      CNT5
      JMP      .-3
      JMS      NXMCK        /CHECK FOR NEXM FLAG
      JMS      IERR        /INDICATE AN ERROR!
      JMS      ERRCHK       /ERROR?
      NDP
      JMS      E2ROU        /TYPE TST # AND AN ERROR DESCRIPTION
      MES48
ST102B ANU      (777
      SAD      PORT        /BREAK TO CORRECT PORT?
      JMP      ST102C       /YES
      DAC      TEMP2
      JMS      IERR        /INDICATE AN ERROR!
      JMS      ERRCHK       /ERROR?
      NDP
      JMS      HDR2         /TYPE "TST= GOOD = BAD" HEADER
      LAC      PORT
      JMS      OCT
      JMS      TYP
      LAC      TEMP2
      JMS      OCT
      JMS      CRLF        /TYPE CARRIAGE RETURN & LINE FEED
      JMS      PRETST
      JMP      BP
ST102C JMS      NXMCK        /CHECK FOR NEXM FLAG
      ISZ      PORT
      ISZ      TPCNT        /DONE?
      JMP      ST102A       /NO
      JMP      PERCHK       /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/*****
/ST103. TEST FOR API BREAKS TO CORRECT PORTS FROM 8 TO 177 FROM
/DR11 APIL1.
ST103  B
02417 000000 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
02420 183354 JHP* ST103
02421 022417 JMS IAPORT
02422 183207 JMS IAPORT
02423 777000 LAM -200
02424 043423 DAC THPCNT /INITIALIZE COUNT
02425 143453 DZM PORT
02426 183222 ST103A JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02427 143454 DZM CNTS /INITIALIZE COUNT
02430 206436 LAC (400000
02431 705004 ISA /INITIATE SELECTED ACTIVITY (API)
02432 750000 CLA
02433 741100 SPA
02434 002445 JHP ST103B
02435 443454 ISZ CNTS
02436 002433 JHP *-3
02437 104120 JMS NXMCK /CHECK FOR NEXM FLAG
02440 102620 JMS IERR /INDICATE AN ERROR!
02441 102625 JMS ERRCHK /ERROR?
02442 740000 NOP
02443 102736 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02444 005477 MES40
02445 506452 ST103B AND (777
02446 543453 SAL PORT /BREAK TO CORRECT PORT?
02447 002465 JHP ST103C /YES
02450 043427 DAC TEMP2
02451 102620 JMS IERR /INDICATE AN ERROR!
02452 102625 JMS ERRCHK /ERROR?
02453 740000 NOP
02454 102776 JMS HDR2 /TYPE "TST= GOOD = BAD" HEADER
02455 203453 LAC PORT
02456 104013 JMS OCT
02457 103716 JMS TYP
02460 203427 LAC TEMP2
02461 104013 JMS OCT
02462 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02463 103753 JMS PRETST
02464 002643 JHP BP
02465 104120 ST103C JMS NXMCK /CHECK FOR NEXM FLAG
02466 443453 ISZ PORT
02467 443423 ISZ THPCNT /DONE?
02470 002426 JHP ST103A /NO
02471 002674 JHP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

42

```

/*****
/ST104. TEST FOR API BREAKS TO CORRECT PORTS FROM 8 TO 177 FROM
/DR11 APIL2.
ST104  B
02472 000000 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
02473 183354 JHP* ST104
02474 022472 JMS IAPORT
02475 183207 JMS IAPORT
02476 777000 LAM -200
02477 043423 DAC THPCNT /INITIALIZE COUNT
02500 143453 DZM PORT
02501 183222 ST104A JMS I15W11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02502 143454 DZM CNTS /INITIALIZE COUNT
02503 206436 LAC (400000
02504 705004 ISA /INITIATE SELECTED ACTIVITY (API)
02505 750000 CLA
02506 741100 SPA
02507 002520 JHP ST104B
02510 443454 ISZ CNTS
02511 002506 JHP *-3
02512 104120 JMS NXMCK /CHECK FOR NEXM FLAG
02513 102620 JMS IERR /INDICATE AN ERROR!
02514 102625 JMS ERRCHK /ERROR?
02515 740000 NOP
02516 102736 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02517 005477 MES40
02520 506452 ST104B AND (777
02521 543453 SAD PORT /BREAK TO CORRECT PORT?
02522 002540 JHP ST104C /YES
02523 043427 DAC TEMP2
02524 102620 JMS IERR /INDICATE AN ERROR!
02525 102625 JMS ERRCHK /ERROR?
02526 740000 NOP
02527 102776 JMS HDR2 /TYPE "TST= GOOD = BAD" HEADER
02530 203453 LAC PORT
02531 104013 JMS OCT
02532 103716 JMS TYP
02533 203427 LAC TEMP2
02534 104013 JMS OCT
02535 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02536 103753 JMS PRETST
02537 002643 JHP BP
02540 104120 ST104C JMS NXMCK /CHECK FOR NEXM FLAG
02541 443453 ISZ PORT
02542 443423 ISZ THPCNT /DONE?
02543 002501 JHP ST104A /NO
02544 002674 JHP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/*****
/ST105. TEST FOR API BREAKS TO CORRECT PORTS FROM 0 TO 177 FROM
/DR11 APIL3.
02545 000000 ST105 0
02546 103354 JMS TAPI /DETERMINE WHETHER OR NOT TO TEST API
02547 622545 JMP* ST105
02550 103207 JMS IAPORT
02551 777600 LAW =200
02552 443423 DAC THPCNT /INITIALIZE COUNT
02553 143453 DZM PORT
02554 103222 ST105A JMS I15M11 /INDICATE PDP-15 DONE FUNCTION & WAIT FOR PDP-11
02555 143454 DZM CNT5 /INITIALIZE COUNT
02556 206436 LAC (400000
02557 705504 ISA /INITIATE SELECTED ACTIVITY (API)
02560 750000 CLA
02561 741100 SPA
02562 602573 JMP ST105B
02563 443454 ISZ CNT5
02564 602561 JMP =-3
02565 104120 JMS NXMCK /CHECK FOR NEXM FLAG
02566 102620 JMS IERR /INDICATE AN ERROR!
02567 102625 JMS ERRCHK /ERROR?
02570 740000 NOP
02571 102736 JMS E2ROU /TYPE TST # AND AN ERROR DESCRIPTION
02572 005477 MES4B
02573 506452 ST105B AND (777
02574 543453 SAD PORT /BREAK TO CORRECT PORT?
02575 602613 JMP ST105C /YES
02576 043427 DAC TEMP2
02577 102620 JMS IERR /INDICATE AN ERROR!
02580 102625 JMS ERRCHK /ERROR?
02601 740000 NOP
02602 102776 JMS HDR2 /TYPE "TST= GOOD = BAD" HEADER
02603 203453 LAC PORT
02604 104013 JMS OCT
02605 103715 JMS TYP
02606 203427 LAC TEMP2
02607 104013 JMS OCT
02610 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02611 103753 JMS PRETST
02612 002643 JMP BP
02613 104120 ST105C JMS NXMCK /CHECK FOR NEXM FLAG
02614 443453 ISZ PORT
02615 443423 ISZ THPCNT /DONE?
02616 602554 JMP ST105A /NO
02617 602674 JMP PERCHK /CHECK FOR A PREVIOUS ERROR CONDITION
/*****
.EJECT

```

```

/
02620 000000 IERR 0
02621 777777 LAW =1
02622 040210 DAC ERRIND /SET PREVIOUS ERROR INDICATOR
02623 040205 DAC ERRFLG
02624 622620 JMP* IERR
02625 000000 ERRCHK 0
02626 200205 LAC ERRFLG /GET ERROR INDICATOR
02627 506412 AND (177777
02630 546412 SAD (177777 /ERROR?
02631 602633 JMP ERRC.1 /YES
02632 022625 JMP* ERRCHK /NO
02633 442625 ERRC.1 ISZ ERRCHK
02634 750004 LAS /GET CONTENTS OF AC SWITCHES
02635 506422 AND (20000
02636 741200 SNA
02637 622625 JMP* ERRCHK
02640 206453 LAC (207
02641 103742 JMS SPTYP /TYPE CHAR REGARDLESS OF AC SW 3
02642 622625 JMP* ERRCHK
/IF AC SW 1=1, RUN ALL TESTS WHETHER THEY FAIL OR NOT.
02643 750004 BP LAS /GET CONTENTS OF AC SWITCHES
02644 506454 AND (600000
02645 741200 SNA
02646 740040 HLT /HLT ON ERR?
02647 777777 LAW =1
02650 043437 DAC PASIND
02651 750004 LAS /GET CONTENTS OF AC SWITCHES
02652 506455 AND (200000
02653 741200 SNA /DO NXT TST?
02654 602561 JMP BP.2 /NO
02655 206456 HPA LAC (JMP*
02656 343421 TAD TSTPNT
02657 042660 DAC BP.1
02660 000000 BP.1 0
02661 750004 BP.2 LAS /GET CONTENTS OF AC SWITCHES
02662 506457 AND (200
02663 740200 SZA
02664 602655 JMP BPA
02665 206454 LAC (JMP
02666 343421 TAD TSTPNT
02667 042673 DAC BP.3
02670 140205 DZM ERKFLG
02671 442673 ISZ BP.3
02672 104120 JMS NXMCK /CHECK FOR NEXM FLAG
02673 000000 BP.3 0
.EJECT

```

```

/
02674 200210 PERCHK LAC ENRIND
02675 506412 AND (177777)
02676 546412 SAD (177777) /PREVIOUS ERROR?
02677 602714 JMP PERC.2 /YES
02700 206456 PERC.1 LAC (JMP*)
02701 343421 TAD TSTPNT
02702 042703 DAC .+1
02703 000000 B
02704 750004 PERC.2 LAS /GET CONTENTS OF AC SWITCHES
02705 506455 AND (200000)
02706 741200 SNA /NXT TST?
02707 602661 JMP BP.2 /NO
02710 140210 DZM ERRIND
02711 602700 JMP PERC.1

/
02712 103043 E1ROU JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02713 103051 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
02714 004146 MES1 /TST- ADDR = GOOD = BAD
02715 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02716 206460 LAC (3
02717 043430 DAC DIG
02720 200201 LAC TEST /GET TEST NUMBER
02721 140113 JMS OCT /TST
02722 103716 JMS TYP
02723 200202 LAC ADR
02724 104013 JMS OCT /ADDR WHERE ERR WAS DETECTED
02725 103716 JMS TYP
02726 200203 LAC GOOD
02727 104013 JMS OCT /GOOD DATA
02730 103716 JMS TYP
02731 200204 LAC BAD
02732 104013 JMS OCT /BAD DATA
02733 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02734 103753 JMS PRETST
02735 602643 JMP BP

/
02736 000000 E2ROU 0
02737 222730 LAC* E2ROU
02740 042743 DAC E2RO.1
02741 102703 JMS HDR1 /TYPE "TST-ERROR DESCRIPTION" HEADER
02742 103051 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
02743 000000 E2RO.1 0
02744 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02745 103753 JMS PRETST
02746 602643 JMP BP
.EJECT
    
```

```

/
02747 102776 E3ROU JMS HDR2 /TYPE "TST- GOOD - BAD" HEADER
02750 200203 LAC GOOD
02751 104013 JMS OCT
02752 103716 JMS TYP
02753 200204 LAC BAD
02754 104013 JMS OCT
02755 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02756 103051 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
02757 004501 MES10
02760 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02761 103753 JMS PRETST
02762 602643 JMP BP

/
02763 000000 HDR1 0
02764 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02765 103051 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
02766 004212 MES5 /TST-ERROR DESCRIPTION
02767 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
02770 206460 LAC (3
02771 043430 DAC DIG
02772 200201 LAC TEST /GET TEST NUMBER
02773 104013 JMS OCT
02774 103716 JMS TYP
02775 622763 JMP* HDR1

/
02776 000000 HDR2 0
02777 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03000 103051 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03001 004334 MES10 /TST- GOOD - BAD
03002 103043 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03003 206460 LAC (3
03004 043430 DAC DIG
03005 200201 LAC TEST /GET TEST NUMBER
03006 104013 JMS OCT
03007 103716 JMS TYP
03010 622776 JMP* HDR2

/
03011 000000 EFCHK 0
03012 200205 LAC ERRFLG /GET ERROR INDICATOR
03013 506412 AND (177777)
03014 546412 SAD (177777)
03015 623011 JMP* EFCHK
03016 777777 LAC -1
03017 040205 DAC ERRFLG
03020 203422 LAC TBAD
03021 340214 DAC BAD
03022 203433 LAC TGOOD
03023 040203 DAC GOOD
03024 200210 LAC 10
03025 040202 DAC ADR
03026 623011 JMP* EFCHK
.EJECT
    
```

```

/
03027 000000 PAS 0
03030 750004 LAS /GET CONTENTS OF AC SWITCHES
03031 506444 AND (100000
03032 741200 SNA
03033 603037 JMP PAS.A
03034 777777 LAW -1
03035 043437 DAC PASIND
03036 623027 JMP* PAS
03037 200201 PAS.A LAC TEST /GET TEST NUMBER
03040 343465 TAD TSTIBL
03041 043437 DAC PASIND
03042 223437 LAC* PASIND
03043 103712 JMS RTHR /ROTATE AC RIGHT 3X
03044 103712 JMS RTHR /ROTATE AC RIGHT 3X
03045 103712 JMS RTHR /ROTATE AC RIGHT 3X
03046 103712 JMS RTHR /ROTATE AC RIGHT 3X
03047 506461 AND (77
03050 740001 CMA
03051 346443 TAD (1
03052 043437 DAC PASIND
03053 623027 JMP* PAS
/SUBROUTINE TO SELECT NEXT TEST
TSTSEL 0
03054 000000 LAS /GET CONTENTS OF AC SWITCHES
03055 750004 AND (177
03056 506462 SNA /LOOP ON TEST?
03057 741200 JMP TSTS.2 /NO
03060 603100 DAC TEST /YES
03061 402001 PAL
03062 722000 SAO (177
03063 546462 JMP TSTS.1
03064 603076 TAD (-106
03065 346463 SPA /LEGAL TEST?
03066 741100 JMP* TSTSEL /YES
03067 623054 JMS HDR1 /TYPE "TST=ERROR DESCRIPTION" HEADER
03070 102763 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03071 103651 MESS65
03072 000104 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03073 103643 DZM TEST
03074 140201 JMP TSTSEL+1
03075 603055 .EJECT

```

```

03076 103252 TSTS.1 JMS PWRFL
03077 603055 JMP TSTSEL+1
03100 750004 TSTS.2 LAS /GET CONTENTS OF AC SWITCHES
03101 506457 AND (200
03102 741200 SNA /RANDOM TEST
03103 603107 JMP TSTS.4 /NO
03104 750004 LAS /GET CONTENTS OF AC SWITCHES
03105 506455 AND (200000
03106 740200 SZL /OO NXT TST?
03107 603130 JMP TSTS2A /YES
03110 200210 LAC ERRIND
03111 506412 AND (177777
03112 546412 SAO (177777
03113 741000 SKP
03114 603130 JMP TSTS2A
03115 223455 LAC* ITST
03116 402001 DAC TEST
03117 722000 PAL
03120 443455 ISZ ITST
03121 203455 LAC ITST
03122 546464 SAO (TEST4+1
03123 741000 SKP
03124 623054 JMP* TSTSEL
03125 206404 LAC (TEST1
03126 043455 DAC ITST
03127 623054 JMP* TSTSEL
.EJECT

```

```

03130 223441 TSTS2A LAC* RANPNT
03131 303443 TAD RAN
03132 506452 AND (177
03133 043442 DAC RANSAV
03134 443441 ISZ RANPNT
03135 740000 NOP
03136 443443 ISZ RAN
03137 740000 NOP
03140 203441 LAC RANPNT
03141 543450 SAD SIZ
03142 741000 SKP
03143 603146 JMP ,+3
03144 206457 LAC (200
03145 043441 DAC RANPNT
03146 203442 LAC RANSAV
03147 741200 SNA
03150 443442 ISZ RANSAV
03151 346463 TAD (-106
03152 740100 SMA
03153 603100 JMP TSTS.2
03154 200201 TSTS.3 LAC TEST
03155 043444 DAC TEST1
03156 200201 LAC TEST
03157 043445 DAC TEST2
03160 200201 LAC TEST
03161 043446 DAC TEST3
03162 203442 LAC RANSAV
03163 043447 DAC TEST4
03164 040201 DAC TEST
03165 722000 PAL
03166 623054 JMP* TSTSEL
03167 440201 TSTS.4 ISZ TEST
03170 200201 LAC TEST
03171 722000 PAL
03172 623054 JMP* TSTSEL
.EJECT

```

```

/LEGAL TEST?
/NO, TRY AGAIN
/GET TEST NUMBER2
/GET TEST NUMBER3
/GET TEST NUMBER4
/GET TEST NUMBER

```

```

03173 000000 /SUBROUTINE TO SELECT NXT MEM PAGE
03174 220017 PAGSEL 0
03175 546431 LAC* 17
03176 603201 SAD (-1
03177 043420 JMP PAGES.1
03200 623173 DAC ADPNT
03201 206414 JMP* PAGSEL
03202 040017 LAC (ADRTBL-1
03203 220017 DAC 17
03204 043420 LAC* 17
03205 443173 DAC ADPNT
03206 623173 ISZ PAGSEL
03207 000000 JMP* PAGSEL
03210 143453 /SUBROUTINE TO INIT API PORT ADDRESSES 0 TO 177.
03211 777000 IAPORT 0
03212 043423 DZH PORT
03213 700000 LAN -200
03214 343453 IAPD.A DAC TNPCNT
03215 063453 LAN
03216 443453 TAD PORT
03217 443423 GAC* PORT
03220 603213 ISZ PORT
03221 623207 ISZ TNPCNT
03222 000000 JMP IAPD.A
03223 777777 /IND 15 READY & WAIT FOR 11.
03224 040213 I15*11 0
03225 103227 LAN -1
03226 623222 DAC IDNE15
03227 000000 JMS W11
03230 143413 /WAIT FOR 11 TO RESPOND.
03231 777776 W11 0
03232 043414 DZH CNT1
03233 200214 LAN -2
03234 546412 DAC CNT2
03235 741000 W11.A LAC IDNE11
03236 603241 SAD (177777
03237 140214 SKP
03240 623227 W11.B JMP W11.B
03241 443413 DZH IDNE11
03242 603233 JMP* W11
03243 443414 ISZ CNT1
03244 603233 JMP W11.A
03245 103643 ISZ CNT2
03246 103651 JMS CRLF
03247 005101 JMS MESS
03250 103643 MESS37
03251 603230 JMS CRLF
.EJECT

```

```

/END OF TABLE?
/YES
/NO. SET TO NXT PAGE.
/INITIALIZE COUNT
/INITIALIZE COUNT
/GET PDP-11 DONE FUNCTION INDICATOR
/11 READY?
/TYPE CARRIAGE RETURN & LINE FEED
/TYPE A MESSAGE VIA THE FOLLOWING POINTER
/TYPE CARRIAGE RETURN & LINE FEED

```

```

/*****
/*****
/POWER FAIL TEST. BOTH PDP-15 & 11 ARE TESTED TO RECOVER FROM
/A POWER FAILURE. BOTH PRUGRAMS MUST BE RESTARTED AFTER RUNNING THIS TEST.
03252 020000 PWRFL 0
03253 206465 LAC (JMP PWR.F,B
03254 040001 DAC 1 /INIT RETURN
03255 777777 LAW =1
03256 040213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
03257 103227 JMS W11 /WAIT FOR PDP-11 DONE FUNCTION INDICATOR
03260 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03261 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03262 005542 MES51
03263 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03264 700042 PWR.F,A ION /TURN PROGRAM INT ON
03265 200214 LAC IDNE11 /GET PDP-11 DONE FUNCTION INDICATOR
03266 540213 SAD IDNE15 /REF COMMON MEM WHILE WAITING
03267 740000 NOP
03270 603264 JMP PWR.F,A
03271 703201 PWR.F,B PFSF /POWER FAILURE?
03272 741000 SKP /NO
03273 603276 JMP PWR.F,C
03274 104120 JMS NXMCK /CHECK FOR NEXM FLAG
03275 603264 JMP PWR.F,A
03276 206466 PWR.F,C LAC (JMP PWR.F,D
03277 040000 DAC 0
03300 206467 LAC (JMP PWR.F,G
03301 040021 DAC 21 /IN CASE OF CAL DURING PWR UP
03302 740040 HLT
03303 603302 JMP =-1
03304 777776 PWR.F,D LAW =2
03305 043414 DAC CNT2 /INITIALIZE COUNT
03306 143413 DZM CNT1 /INITIALIZE COUNT
03307 777777 LAW =1
03310 040213 DAC IDNE15 /INDICATE PDP-15 DONE FUNCTION
03311 200214 PWR.F,E LAC IDNE11 /GET PDP-11 DONE FUNCTION INDICATOR
03312 546412 SAD (177777
03313 603325 JMP PWR.F,F
03314 443413 ISZ CNT1
03315 603311 JMP PWR.F,E
03316 443414 ISZ CNT2
03317 603311 JMP PWR.F,E
03320 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03321 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03322 005561 MES52
03323 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03324 603253 JMP PWR.FL+1
03325 140214 PWR.F,F DZM IDNE11
03326 200203 LAC GOOD
03327 546412 SAD (177777 /IS PWR FAIL FIRST?
03330 603336 JMP PWR.F,1 /YES
03331 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03332 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03333 005560 MES53
03334 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED

```

```

03335 603253 JMP PWR.FL+1 /START OVER
03336 103643 PWR.F,1 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03337 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03340 005520 MES50
03341 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03342 603253 JMP PWR.FL+1
03343 103643 PWR.F,G JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03344 200020 LAC 20
03345 506470 AND (177777
03346 104013 JMS OCT
03347 103716 JMS TYP
03350 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03351 005637 MES55
03352 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03353 603253 JMP PWR.FL+1
/*****
/*****
.EJECT

```

```

/
03354 000000 TAPI 0
03355 203415 LAC IAPI
03356 741200 SNA /TEST API?
03357 623354 JMP* TAPI /NO
03358 443354 ISZ TAPI
03361 623354 JMP* TAPI

/SUBROUTINE TO WRITE PDP-15 ADDRESSES INTO 2K OF COMMON MEMORY STARTING
/AT ADDRESS+1 WHERE LOC 10 CONTAINS ADDRESS. A RANDOM FASHION DELAY IS
/EXECUTED BEFORE EACH WORD IS WRITTEN.
03362 000000 WRT2K 0
03363 774000 LAM -4000
03364 043417 DAC WCNT /INITIALIZE COUNT
03365 200010 LAC 10
03366 346443 TAD (1
03367 060010 DAC* 10 /WRITE AN ADDRESS INTO A LOC
03370 443417 ISZ WCNT /DONE?
03371 603365 JMP -4 /NO
03372 023362 JMP* WRT2K

/SUBROUTINE TO READ & CHECK PDP-15 ADDRESSES WRITTEN BY 11. EACH LOC
/SHOULD CONTAIN ITS PDP-15 ADDRESS. IF AN ERROR IS ENCOUNTERED, LOC
/ERRFLG IS SET TO -1, THE GOOD WORD AND ADDRESS (SAME) IS DEPOSITED IN
/LOC GOOD, AND THE INCORRECT DATA WHICH WAS READ IS DEPOSITED IN LOC
/BAU. TESTING BEGINS AT ADDRESS+1 WHERE LOC 10 CONTAINS ADDRESS.
/DELAY BEFORE EACH WORD IS CHECKED.
03373 000000 READ2K 0
03374 774000 LAM -4000
03375 043416 DAC RCNT /INITIALIZE COUNT
03376 220010 READ,1 LAC* 10
03377 043422 DAC TBAU /TEMP STORAGE
03400 540010 SAD 10 /DATA READ, SAME AS ADDRESS?
03401 603410 JMP READ,2 /YES
03402 200010 LAC 10
03403 043433 DAC TGOOD
03404 777777 LAM -1 /NO
03405 040210 DAC ERRIND /SET PREVIOUS ERROR INDICATOR
03406 103011 JMS EFCMK /CHECK ERRFLG TO DETERMINE WHETHER OR NOT TO STORE DATA
03407 623373 JMP* READ2K
03410 443416 READ,2 ISZ RCNT /DONE?
03411 603376 JMP READ,1 /NO
03412 623373 JMP* READ2K
.EJECT
    
```

```

/CONSTANTS & VARIABLES
03413 000000 CNT1 0
03414 200000 CNT2 0
03415 000000 IAPI 0
03416 000000 RCNT 0
03417 000000 WCNT 0
03420 000000 ADHPNT 0
03421 000000 TSTPNT 0
03422 000000 TBAU 0
03423 000000 TRPCNT 0
03424 000000 PASCNT 0
03425 000000 TEMP 0
03426 000000 TEMP1 0
03427 000000 TEMP2 0
03430 000000 MSAVE 0
03431 000000 SXBT 0
03432 000000 INIT 0
03433 000000 TGOOD 0
03434 000000 SAVE 0
03435 000000 LZEX 0
03436 000000 DIG 0
03437 000000 PASIND 0
03440 000000 TYPTMP 0
03441 000200 RANPNT 200
03442 000000 RANSAV 0
03443 000000 RAM 0
03444 000000 TEST1 0
03445 000000 TEST2 0
03446 000000 TEST3 0
03447 000000 TEST4 0
03450 006503 SIZ .SIZE
03451 000000 CNT3 0
03452 000000 CNT4 0
03453 000000 PORT 0
03454 000000 CNT5 0
03455 000000 ITST 0
03456 000000 IBCNT 0

/
03457 010000 AURTBL 10000 /IND 0K MIN OF COMMON MEM
03460 000000 0 /IND 12K IF NOT -1
03461 000000 0 /IND 16K IF NOT -1
03462 000000 0 /IND 20K IF NOT -1
03463 000000 0 /IND 24K IF NOT -1
03464 777777 -1 /IND END OF TABLE
.EJECT
    
```


	/	
03465	003465	TSTBL .
03466	020460	TST1+20000
03467	020503	TST2+20000
03470	020527	TST3+20000
03471	020550	TST4+20000
03472	020572	TST5+20000
03473	020631	TST6+20000
03474	020670	TST7+20000
03475	020723	TST10+20000
03476	020756	TST11+20000
03477	021007	TST12+20000
03500	021040	TST13+20000
03501	021127	TST14+20000
03502	771216	TST15+770000
03503	771225	TST16+770000
03504	771234	TST17+770000
03505	771243	TST20+770000
03506	771254	TST21+770000
03507	771265	TST22+770000
03510	771301	TST23+770000
03511	021314	TST24+20000
03512	771365	TST25+770000
03513	771373	TST26+770000
03514	771401	TST27+770000
03515	771407	TST30+770000
03516	771415	TST31+770000
03517	771423	TST32+770000
03520	771435	TST33+770000
03521	771447	TST34+770000
03522	771461	TST35+770000
03523	771473	TST36+770000
03524	771504	TST37+770000
03525	771515	TST40+770000
03526	771526	TST41+770000
03527	771537	TST42+770000
03530	771565	TST43+770000
03531	771613	TST44+770000
03532	771641	TST45+770000
03533	771667	TST46+770000
03534	771701	TST47+770000
03535	771714	TST50+770000
03536	771727	TST51+770000
03537	771745	TST52+770000
03540	771753	TST53+770000
03541	771771	TST54+770000
03542	771777	TST55+770000
03543	772015	TST56+770000
03544	772023	TST57+770000
03545	772041	TST60+770000
03546	772047	TST61+770000
03547	772055	TST62+770000
03550	772063	TST63+770000
03551	772071	TST64+770000
03552	772077	TST65+770000

03553	772105	TST66+770000
03554	502113	TST67+500000
03555	502121	TST70+500000
03556	502131	TST71+500000
03557	502137	TST72+500000
03560	502147	TST73+500000
03561	502162	TST74+500000
03562	502177	TST75+500000
03563	502224	TST76+500000
03564	502250	TST77+500000
03565	502274	ST100+500000
03566	502320	ST101+500000
03567	102344	ST102+100000
03570	102417	ST103+100000
03571	102472	ST104+100000
03572	102545	ST105+100000
03573	777777	=1

EJECT

/SUBROUTINE TO READ 6 OR LESS OCTAL DIGITS FROM KEYBOARD. INCORRECT
 /CHARACTERS CAUSE RETURN TO NORMAL FLOW, CORRECT CAUSE RETURN TO NORMAL
 /FLOW+. RESULT IN TEMP1 & AC.

```

03574 000000 READ 0
03575 777772 LAM -0
03576 043423 DAC TPCNT /INITIALIZE COUNT
03577 143426 DZM TEMP1
03600 700301 REA.1 KSF
03601 603600 JMP -1
03602 700312 KR0
03603 043425 DAC TEMP
03604 346471 TAD (-215
03605 740200 SZA /CRT
03606 003612 JMP +4 /NO
03607 203426 LAC TEMP1
03610 443574 ISZ READ /YES
03611 023574 JMP+ READ
03612 203425 LAC TEMP
03613 346472 TAD (-200
03614 750100 SPAICLA />200?
03615 603622 JMP +5 /YES
03616 103651 REA.2 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03617 004156 MES2
03620 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03621 023574 JMP+ READ
03622 203425 LAC TEMP
03623 346473 TAD (-270
03624 750100 SPAICLA /<270?
03625 603610 JMP REA.2 /NO
03626 203425 LAC TEMP /YES
03627 506415 AND (7
03630 043425 DAC TEMP
03631 744000 CLL
03632 203426 LAC TEMP1
03633 740010 RAL
03634 742010 RTL
03635 343425 TAD TEMP
03636 043426 DAC TEMP1
03637 443423 ISZ TPCNT
03640 603600 JMP REA.1
03641 443574 ISZ READ
03642 023574 JMP+ READ

/
03643 000000 CRLF 0
03644 700215 LAM 215
03645 103716 JMS TYP
03646 700212 LAM 212
03647 103716 JMS TYP
03650 623643 JMP+ CRLF
.EJECT
  
```

```

/
03651 000000 MESS 0
03652 223651 LAC+ MESS
03653 506470 AND (17777
03654 043430 DAC MSAVE
03655 443651 ISZ MESS
03656 103733 JMS CTTY /TYPE?
03657 023651 JMP+ MESS
03660 223430 MES.1 LAC+ MSAVE
03661 043431 DAC SXBT
03662 103712 JMS RTHR /ROTATE AC RIGHT 3X
03663 103712 JMS RTHR /ROTATE AC RIGHT 3X
03664 103712 JMS RTHR /ROTATE AC RIGHT 3X
03665 103712 JMS RTHR /ROTATE AC RIGHT 3X
03666 103677 JMS MESSA
03667 203431 LAC SXBT
03670 103712 JMS RTHR /ROTATE AC RIGHT 3X
03671 103712 JMS RTHR /ROTATE AC RIGHT 3X
03672 103677 JMS MESSA
03673 203431 LAC SXBT
03674 103677 JMS MESSA
03675 443430 ISZ MSAVE
03676 003660 JMP MES.1

/
03677 000000 MESSA 0
03700 506461 AND (77
03701 741200 SNA
03702 623651 JMP+ MESS
03703 043425 DAC TEMP
03704 346474 TAD (-40
03705 751100 SPAICLA
03706 206475 LAC (100
03707 343425 TAD TEMP
03710 103716 JMS TYP
03711 623677 JMP+ MESSA

/
03712 000000 RTHR 0
03713 740020 RAR
03714 742020 RTR
03715 623712 JMP+ RTHR
.EJECT
  
```

```

03716 000000 /
03717 043440 TYP 0
03720 103733 DAC TYPTMP
03721 623716 JMS CTTY /TYP?
03722 203440 JMP* TYP
03723 741200 LAC TYPTMP
03724 760240 SNA
03725 700406 LAW 240
03726 700401 TLS
03727 603726 TSF
03730 700402 JHP .-1
03731 750000 TCF
03732 623716 CLA
JMP* TYP

03733 000000 /
03734 750004 CTTY 0
03735 506424 LAS /GET CONTENTS OF AC SWITCHES
03736 740200 AND (40000
03737 623733 SZA /TYP?
03740 443733 JMP* CTTY /NO
03741 623733 ISZ CTTY
JMP* CTTY

03742 000000 /
03743 741200 SPTYP 0
03744 760240 SNA
03745 700406 LAW 240
03746 700401 TLS
03747 603746 TSF
03750 700402 JHP .-1
03751 750000 TCF
03752 623742 CLA
JMP* SPTYP
.EJECT
    
```

```

03753 000000 /
03754 750004 PRKTST 0
03755 506457 LAS /GET CONTENTS OF AC SWITCHES
03756 741200 AND (200
03757 623753 SNA /RANDOM TEST?
03760 103651 JMP* PRETST /NO
03761 006204 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03762 103643 MES70
03763 103651 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03764 006251 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03765 103643 MES75
03766 103651 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03767 006250 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03770 200201 LAC TEST /GET TEST NUMBER1
03771 104013 JMS OCT
03772 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
03773 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
03774 006254 MES72
03775 200201 LAC TEST /GET TEST NUMBER2
03776 104013 JMS OCT
03777 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
04000 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
04001 006240 MES73
04002 200201 LAC TEST /GET TEST NUMBERS
04003 104013 JMS OCT
04004 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
04005 103651 JMS MESS /TYPE A MESSAGE VIA THE FOLLOWING POINTER
04006 006244 MES74
04007 200201 LAC TEST /GET TEST NUMBER4
04010 104013 JMS OCT
04011 103643 JMS CRLF /TYPE CARRIAGE RETURN & LINE FEED
04012 623753 JMP* PRETST
.EJECT
    
```

```

/
04013 000000 OCT 0
04014 043434 DAC SAVE
04015 103733 JMS CTTY /IYP?
04016 624013 JMP* OCT
04017 143435 DZM LZER
04020 203430 LAC DIG
04021 741200 SNA /5 ?
04022 604031 JMP OCT.1
04023 546417 SAD (4
04024 604046 JMP OCT.2
04025 546400 SAD (3
04026 604053 JMP OCT.3
04027 546443 SAD (1
04030 604062 JMP OCT.4
04031 203434 OCT.1 LAC SAVE
04032 103712 JMS RTHR /ROTATE AC RIGHT 3X
04033 103712 JMS RTHR /ROTATE AC RIGHT 3X
04034 103712 JMS RTHR /ROTATE AC RIGHT 3X
04035 103712 JMS RTHR /ROTATE AC RIGHT 3X
04036 103712 JMS RTHR /ROTATE AC RIGHT 3X
04037 104070 JMS OCTY1
04040 203434 LAC SAVE
04041 103712 JMS RTHR /ROTATE AC RIGHT 3X
04042 103712 JMS RTHR /ROTATE AC RIGHT 3X
04043 103712 JMS RTHR /ROTATE AC RIGHT 3X
04044 103712 JMS RTHR /ROTATE AC RIGHT 3X
04045 104070 JMS OCTY1
04046 203434 OCT.2 LAC SAVE
04047 103712 JMS RTHR /ROTATE AC RIGHT 3X
04050 103712 JMS RTHR /ROTATE AC RIGHT 3X
04051 103712 JMS RTHR /ROTATE AC RIGHT 3X
04052 104070 JMS OCTY1
04053 203434 OCT.3 LAC SAVE
04054 103712 JMS RTHR /ROTATE AC RIGHT 3X
04055 103712 JMS RTHR /ROTATE AC RIGHT 3X
04056 104070 JMS OCTY1
04057 203434 LAC SAVE
04060 103712 JMS RTHR /ROTATE AC RIGHT 3X
04061 104070 JMS OCTY1
04062 203434 OCT.4 LAC SAVE
04063 506415 AND (7
04064 346476 TAD (200
04065 103716 JMS TYP
04066 143436 DZM DIG
04067 624013 JMP* OCT
.EJECT
    
```

```

/
04070 000000 OCTY1 0
04071 506415 AND (7
04072 741200 SNA
04073 604100 JMP OCTY.2
04074 443435 ISZ LZER
04075 346476 OCTY.1 TAD (200
04076 103716 JMS TYP
04077 624070 JMP* OCTY1
04100 203435 OCTY.2 LAC LZER
04101 750200 SZALCLA
04102 604075 JMP OCTY.1
04103 760240 LAW 240
04104 103716 JMS TYP
04105 624070 JMP* OCTY1
/
04106 000000 Ibuff 0
04107 043456 DAC IBCNT
04110 203420 LAC ADRPNT /GET FIRST ADDRESS OF CURRENT PAGE
04111 346431 TAD (-1
04112 040010 DAC 10
04113 777777 LAW -1
04114 060010 DAC* 10
04115 443456 ISZ IBCNT
04116 604114 JMP .-2
04117 624106 JMP* Ibuff
.EJECT
    
```

```

04120 000000 /
04121 701741 NXMCK 0
04122 504144 JMP MPSNE /NEXM SET?
04123 103643 JMS NXMCK1 /NOPE
04124 103643 JMS CRLF /YES..
04125 103651 JMS CRLF / ..
04126 006275 JMS MESS / ..GO TELL THE MAN!
04127 204120 MES76
04130 506477 LAC NXMCK
04131 104013 AND [077777
04132 103643 JMS OCT /PRINT THE PC
04133 103651 JMS CRLF
04134 006307 JMS MESS /MAKE SOME SUGGESTIONS
04135 103643 MES76A
04136 103651 JMS CRLF
04137 006325 JMS MESS
04140 103643 MES76B
04141 103651 JMS CRLF
04142 006351 JMS MESS
04143 103643 MES76C
04144 703302 JMS CRLF
04145 624120 NXMCK1 CAF
JMP+ NXMCK
.EJECT
    
```

```

04146 242324 /MESSAGES
04147 554001 MES1 .SIXBT 'TST= ADDR = GOOD = BAD#'
04150 040422
04151 405540
04152 071717
04153 044055
04154 400201
04155 040000
04156 777777 MES2 .SIXBT '???'
04157 000000
04160 042261 MES3 .SIXBT 'DR11-C BR5 TV LOC (TYPE CR FOR 300) = #'
04161 615503
04162 400222
04163 654024
04164 264014
04165 170340
04166 502431
04167 200540
04170 032240
04171 061722
04172 406360
04173 605140
04174 754000
04175 042261 MES4 .SIXBT 'DR11-C BR7 TV LOC (TYPE CR FOR 310) = #'
04176 615503
04177 400222
04200 674024
04201 264014
04202 170340
04203 502431
04204 200540
04205 032240
04206 061722
04207 406361
04210 605140
04211 754000
04212 242324 MES5 .SIXBT 'TST=ERROR DESCRIPTION#'
04213 550522
04214 221722
04215 400405
04216 230322
04217 112024
04220 111710
04221 000000
04222 141117 MES6 .SIXBT 'LIDR FAILED TO SET DR11 TC8P FLG, #'
04223 224006
04224 011114
04225 050440
04226 241740
04227 230524
04230 400422
04231 616140
04232 240302
04233 204000
    
```

04234 140756
04235 000000
04236 040124
04237 114024
04240 174024
04241 042055
04242 610140
04243 141703
04244 006166
04245 676766
04246 644006
04247 011114
04250 050440
04251 241740
04252 031405
04253 012240
04254 240302
04255 204006
04256 140756
04257 000000
04260 030106
04261 400601
04262 111405
04263 044024
04264 174023
04265 052454
04266 401722
04267 002311
04270 170140
04271 060111
04272 140504
04273 002417
04274 002313
04275 112040
04276 171654
04277 002403
04300 022040
04301 010303
04302 052024
04303 050440
04304 061407
04305 560000
04306 031117
04307 044006
04310 011114
04311 050440
04312 241740
04313 031422
04314 002403
04315 022040
04316 010303
04317 052024
04320 050440
04321 061407
04322 401722

MSG7 .SIXBT 'DATI TO PDP-11 LOC 167764 FAILED TO CLEAR TCBP FLG.0'

MSG8 .SIXBT 'CAF FAILED TO SET, OR SIOA FAILED TO SKIP ON, TCBP ACCEPTED FLG.0'

MSG9 .SIXBT 'CIOD FAILED TO CLR TCBP ACCEPTED FLG OR SIOA SKIPPED ON NO FLG.0'

04323 002311
04324 170140
04325 231311
04326 202005
04327 044017
04330 164016
04331 174006
04332 140756
04333 000000
04334 242324
04335 554007
04336 171704
04337 005540
04340 020164
04341 000000
04342 042261
04343 614001
04344 201140
04345 004004
04346 171605
04347 000614
04350 074006
04351 011114
04352 050440
04353 241740
04354 230524
04355 560000
04356 042261
04357 614001
04360 201140
04361 614004
04362 171605
04363 000614
04364 074006
04365 011114
04366 050440
04367 241740
04370 230524
04371 560000
04372 042261
04373 614001
04374 201140
04375 024004
04376 171605
04377 000614
04400 074006
04401 011114
04402 050440
04403 241740
04404 230524
04405 560000
04406 042261
04407 614001
04410 201140
04411 634004

MSG10 .SIXBT 'TST- GOOD - BAD0'

MSG11 .SIXBT 'DR11 API 0 DONE FLG FAILED TO SET.0'

MSG12 .SIXBT 'DR11 API 1 DONE FLG FAILED TO SET.0'

MSG13 .SIXBT 'DR11 API 2 DONE FLG FAILED TO SET.0'

MSG14 .SIXBT 'DR11 API 3 DONE FLG FAILED TO SET.0'

04412	171605	
04413	400614	
04414	074006	
04415	011114	
04416	050440	
04417	241740	
04420	230524	
04421	560000	
04422	042201	MES15 .SIXBT 'DR11 API DONE FLG FAILED TO SET,0'
04423	614001	
04424	201140	
04425	041716	
04426	054006	
04427	140740	
04430	060111	
04431	140504	
04432	402417	
04433	402305	
04434	245000	
04435	042201	MES16 .SIXBT 'DR15 INT ENABLE FAILED TO BE SET BY LORS,0'
04436	654011	
04437	102440	
04440	051501	
04441	021405	
04442	400501	
04443	111405	
04444	044024	
04445	174002	
04446	054023	
04447	052440	
04450	023140	
04451	140422	
04452	235000	
04453	042201	MES17 .SIXBT 'DR11 API LEVEL 0 DONE FLG FAILED TO CLR OR WRONG DONE FLG CLRED,0'
04454	614001	
04455	201140	
04456	140526	
04457	051440	
04460	004004	
04461	171605	
04462	400614	
04463	074006	
04464	011114	
04465	050440	
04466	241740	
04467	031422	
04470	401722	
04471	402722	
04472	171607	
04473	400417	
04474	100540	
04475	061407	
04476	400314	
04477	220504	
04500	560000	

04501	021124	MES18 .SIXBT 'BIT 2=API LEVEL 0, 3=LEVEL 1, 4=LEVEL 2 AND 5=LEVEL 3,0'
04502	406275	
04503	012011	
04504	401405	
04505	260514	
04506	406054	
04507	406375	
04510	140526	
04511	051440	
04512	615440	
04513	647514	
04514	052005	
04515	144002	
04516	400116	
04517	044005	
04520	751405	
04521	260514	
04522	406356	
04523	000000	
04524	042201	MES19 .SIXBT 'DR11 API LEVEL 1 DONE FLG FAILED TO CLR OR WRONG DONE FLG CLRED,0'
04525	614001	
04526	201140	
04527	140526	
04530	051440	
04531	614004	
04532	171605	
04533	400614	
04534	074006	
04535	011114	
04536	050440	
04537	241740	
04540	031422	
04541	401722	
04542	402722	
04543	171607	
04544	400417	
04545	100540	
04546	061407	
04547	400314	
04550	220504	
04551	560000	
04552	030120	MES20 .SIXBT 'CAPI0 FAILED TO CLR API0 FLG,0'
04553	116040	
04554	060111	
04555	140504	
04556	402417	
04557	400314	
04560	224001	
04561	201160	
04562	400614	
04563	075600	
04564	042201	MES21 .SIXBT 'DR11 API LEVEL 2 DONE FLG FAILED TO CLR OR WRONG DONE FLG CLRED,0'
04565	614001	
04566	201140	
04567	140526	

04570 051440
 04571 624004
 04572 171605
 04573 400614
 04574 074006
 04575 011114
 04576 050440
 04577 241740
 04600 031422
 04601 401722
 04602 402722
 04603 171607
 04604 400417
 04605 160540
 04606 061407
 04607 400314
 04610 220504
 04611 560000
 04612 030120
 04613 116140
 04614 060111
 04615 140504
 04616 402417
 04617 400314
 04620 224001
 04621 201161
 04622 400614
 04623 075600
 04624 042261
 04625 614001
 04626 201140
 04627 140526
 04630 051440
 04631 634004
 04632 171605
 04633 400614
 04634 074006
 04635 011114
 04636 050440
 04637 241740
 04640 031422
 04641 401722
 04642 402722
 04643 171607
 04644 400417
 04645 160540
 04646 061407
 04647 400314
 04650 220504
 04651 560000
 04652 012011
 04653 400417
 04654 160540
 04655 061407
 04656 400601

MES22 ,SIXBT 'CAPI1 FAILED TO CLR API1 FLG,0'

MES23 ,SIXBT 'DR11 API LEVEL 3 DONE FLG FAILED TO CLR OR WRONG DONE FLG CLRED,0'

MES24 ,SIXBT 'API DONE FLG FAILED TO CLEAR WHEN LEVEL 0 HAD A REQUEST PENDING,0'

04657 111405
 04660 044024
 04661 174003
 04662 140501
 04663 224027
 04664 100516
 04665 401405
 04666 260514
 04667 406040
 04670 100104
 04671 400140
 04672 220521
 04673 250523
 04674 244020
 04675 031604
 04676 111607
 04677 560000
 04700 230120
 04701 116040
 04702 231311
 04703 202005
 04704 044027
 04705 112410
 04706 401617
 04707 401405
 04710 260514
 04711 400040
 04712 061407
 04713 401722
 04714 400301
 04715 064006
 04716 011114
 04717 050440
 04720 241740
 04721 031422
 04722 400614
 04723 075600
 04724 230120
 04725 116140
 04726 231311
 04727 202005
 04730 044027
 04731 112410
 04732 401617
 04733 401405
 04734 260514
 04735 400140
 04736 061407
 04737 401722
 04740 400301
 04741 064006
 04742 011114
 04743 050440
 04744 241740
 04745 031422

MES25 ,SIXBT 'SAPI0 SKIPPED WITH NO LEVEL 0 FLG OR CAF FAILED TO CLR FLG,0'

MES26 ,SIXBT 'SAPI1 SKIPPED WITH NO LEVEL 1 FLG OR CAF FAILED TO CLR FLG,0'

04746	400614	
04747	075600	
04750	230120	MES27 .SIXBT 'SAPI2 SKIPPED WITH NO LEVEL 2 FLG OR CAF FAILED TO CLR FLG.#'
04751	116240	
04752	231311	
04753	202005	
04754	044027	
04755	112410	
04756	401617	
04757	401405	
04760	260514	
04761	406240	
04762	061407	
04763	401722	
04764	400301	
04765	064006	
04766	011114	
04767	050440	
04770	241740	
04771	031422	
04772	400614	
04773	075600	
04774	230120	MES28 .SIXBT 'SAPI3 SKIPPED WITH NO LEVEL 3 FLG OR CAF FAILED TO CLR FLG.#'
04775	116340	
04776	231311	
04777	202005	
05000	044027	
05001	112410	
05002	401617	
05003	401405	
05004	260514	
05005	406340	
05006	061407	
05007	401722	
05010	400301	
05011	064006	
05012	011114	
05013	050440	
05014	241740	
05015	031422	
05016	400614	
05017	075600	
05020	230120	MES29 .SIXBT 'SAPI0 FAILED TO SKIP.#'
05021	116040	
05022	060111	
05023	140504	
05024	402417	
05025	402313	
05026	112056	
05027	000000	
05030	230120	MES30 .SIXBT 'SAPI1 FAILED TO SKIP.#'
05031	116140	
05032	060111	
05033	140504	
05034	402417	

05035	402313	
05036	112056	
05037	000000	
05040	230120	MES31 .SIXBT 'SAPI2 FAILED TO SKIP.#'
05041	116240	
05042	060111	
05043	140504	
05044	402417	
05045	402313	
05046	112056	
05047	000000	
05050	230120	MES32 .SIXBT 'SAPI3 FAILED TO SKIP.#'
05051	116340	
05052	060111	
05053	140504	
05054	402417	
05055	402313	
05056	112056	
05057	000000	
05060	030106	MES33 .SIXBT 'CAF FAILED TO SET DR15 INT ENABLE OR RDRS FAILED TO READ IT.#'
05061	400601	
05062	111405	
05063	044024	
05064	174023	
05065	052440	
05066	042261	
05067	654011	
05070	162440	
05071	051601	
05072	021405	
05073	401722	
05074	402204	
05075	222340	
05076	060111	
05077	140504	
05100	402417	
05101	402205	
05102	010440	
05103	112456	
05104	000000	
05105	042261	MES34 .SIXBT 'DR15 INT ENABLE FAILED TO BE CLEARED BY LDRS.#'
05106	654011	
05107	162440	
05110	051601	
05111	021405	
05112	400601	
05113	111405	
05114	044024	
05115	174023	
05116	054003	
05117	140501	
05120	220504	
05121	400231	
05122	401404	
05123	222356	

#5124	#00000	
#5125	251611	MES35 .SIXBT 'UNICHANNEL15 DIAGNOSTIC IS RUNNING.0'
#5126	#31001	
#5127	161005	
#5130	146165	
#5131	400411	
#5132	#10710	
#5133	172324	
#5134	110340	
#5135	112340	
#5136	222516	
#5137	161116	
#5140	#75000	
#5141	112340	MES36 .SIXBT 'IS API AVAILABLE & TO BE TESTED? (TYPE Y OR N) 0'
#5142	#12011	
#5143	400120	
#5144	#11114	
#5145	#10214	
#5146	#54046	
#5147	402417	
#5150	400205	
#5151	402405	
#5152	232405	
#5153	#47740	
#5154	502431	
#5155	200540	
#5156	314017	
#5157	224016	
#5160	514000	
#5161	270111	MES37 .SIXBT 'WAITING FOR PDP-11 PROGRAM TO INDICATE READY TO TEST!0'
#5162	241116	
#5163	#74006	
#5164	172240	
#5165	200420	
#5166	556161	
#5167	402022	
#5170	170722	
#5171	011540	
#5172	241740	
#5173	111604	
#5174	110301	
#5175	240540	
#5176	220501	
#5177	#43140	
#5200	241740	
#5201	240523	
#5202	244100	
#5203	#40124	MES38 .SIXBT 'DAT1 TO PDP-11 LOC 167774 CLEARED TCBP FLG.0'
#5204	114024	
#5205	174020	
#5206	#42055	
#5207	016140	
#5210	141703	
#5211	406100	
#5212	676767	

#5213	644003	
#5214	140501	
#5215	220504	
#5216	402403	
#5217	#22046	
#5220	#61407	
#5221	500000	
#5222	#42261	MES39 .SIXBT 'DR11 API DONE INT ENABLE FAILED TO SET.0'
#5223	614001	
#5224	201140	
#5225	#41716	
#5226	#54011	
#5227	102440	
#5230	#51601	
#5231	#21405	
#5232	400601	
#5233	111405	
#5234	#44024	
#5235	174023	
#5236	052456	
#5237	000000	
#5240	#42261	MES40 .SIXBT 'DR11 API DONE INT ENABLE FAILED TO CLEAR.0'
#5241	614001	
#5242	201140	
#5243	#41716	
#5244	#54011	
#5245	102440	
#5246	#51601	
#5247	021405	
#5250	400601	
#5251	111405	
#5252	#44024	
#5253	174003	
#5254	140501	
#5255	225000	
#5256	#42261	MES41 .SIXBT 'DR11 API DONE FLG FAILED TO CAUSE A PDP-11 INTERRUPT.0'
#5257	614001	
#5260	201140	
#5261	#41716	
#5262	#54006	
#5263	140740	
#5264	#60111	
#5265	140504	
#5266	402417	
#5267	400301	
#5270	252305	
#5271	400140	
#5272	200420	
#5273	556161	
#5274	401116	
#5275	240522	
#5276	222520	
#5277	245000	
#5300	#42261	MES42 .SIXBT 'DR11 TCBP FLG FAILED TO CAUSE A PDP-11 INTERRUPT.0'
#5301	614024	

05302 030220
 05303 400614
 05304 074003
 05305 011114
 05306 050440
 05307 241740
 05310 030125
 05311 230540
 05312 014020
 05313 042055
 05314 616140
 05315 111624
 05316 052222
 05317 252024
 05320 560000
 05321 042261
 05322 614024
 05323 030220
 05324 400614
 05325 074003
 05326 012523
 05327 050440
 05330 014020
 05331 042055
 05332 616140
 05333 111624
 05334 402710
 05335 111405
 05336 402403
 05337 022040
 05340 051601
 05341 021405
 05342 402701
 05343 234003
 05344 140501
 05345 220504
 05346 560000
 05347 042261
 05350 614001
 05351 201140
 05352 041665
 05353 400614
 05354 074003
 05355 012523
 05356 050440
 05357 011640
 05360 111624
 05361 402710
 05362 111405
 05363 400120
 05364 114004
 05365 160540
 05366 051601
 05367 021405
 05370 402701

MES43 .SIXBT 'DR11 TCBP FLG CAUSED A PDP-11 INT WHILE TCBP ENABLE WAS CLEARED,0'

MES44 .SIXBT 'DR11 API ONE FLG CAUSED AN INT WHILE API ONE ENABLE WAS CLEARED,0'

05371 234003
 05372 140501
 05373 220504
 05374 560000
 05375 012011
 05376 400417
 05377 160540
 05400 061407
 05401 400601
 05402 111405
 05403 044024
 05404 174003
 05405 140501
 05406 224027
 05407 100516
 05410 401405
 05411 260514
 05412 406140
 05413 100104
 05414 400140
 05415 220521
 05416 250523
 05417 244020
 05420 051604
 05421 111607
 05422 560000
 05423 012011
 05424 400417
 05425 160540
 05426 061407
 05427 400601
 05430 111405
 05431 044024
 05432 174003
 05433 140501
 05434 224027
 05435 100516
 05436 401405
 05437 260514
 05440 406240
 05441 100104
 05442 400140
 05443 220521
 05444 250523
 05445 244020
 05446 051604
 05447 111607
 05450 560000
 05451 012011
 05452 400417
 05453 160540
 05454 061407
 05455 400601
 05456 111405
 05457 044024

MES45 .SIXBT 'API DONE FLG FAILED TO CLEAR WHEN LEVEL 1 HAD A REQUEST PENDING,0'

MES46 .SIXBT 'API DONE FLG FAILED TO CLEAR WHEN LEVEL 2 HAD A REQUEST PENDING,0'

MES47 .SIXBT 'API DONE FLG FAILED TO CLEAR WHEN LEVEL 3 HAD A REQUEST PENDING,0'

05460	174003	
05461	140501	
05462	224027	
05463	100516	
05464	401405	
05465	200514	
05466	400340	
05467	100104	
05470	400140	
05471	220521	
05472	250523	
05473	240020	
05474	051004	
05475	111007	
05476	500000	
05477	012011	MES48 .SIXBT 'API FAILED TO OCCUR,0'
05500	400001	
05501	111405	
05502	044024	
05503	174017	
05504	030325	
05505	225000	
05506	030120	MES49 .SIXBT 'CAPI2 FAILED TO CLR API2 FLG,0'
05507	110240	
05510	000111	
05511	140504	
05512	402417	
05513	400314	
05514	224001	
05515	201102	
05516	400014	
05517	075000	
05520	200420	MES50 .SIXBT 'PDP-15 PWR FAILED FIRST, TST SUCCESSFULLY COMPLETED,0'
05521	550105	
05522	402027	
05523	224006	
05524	011114	
05525	050440	
05526	001122	
05527	232450	
05530	402423	
05531	244023	
05532	250303	
05533	052323	
05534	002514	
05535	143140	
05536	031715	
05537	201405	
05540	240504	
05541	500000	
05542	141703	MES51 .SIXBT 'LOCK 15'S CONSOLE, REMOVE POWER FROM SYSTEM,00'
05543	134001	
05544	054723	
05545	400317	
05546	102317	

05547	140554	
05550	402205	
05551	151720	
05552	054020	
05553	172705	
05554	224006	
05555	221715	
05556	402331	
05557	232405	
05560	155000	
05561	200420	MES52 .SIXBT 'PDP-11 FAILED TO RECOVER FROM POWER FAILURE,0'
05562	550101	
05563	400001	
05564	111405	
05565	044024	
05566	174022	
05567	050317	
05570	200522	
05571	400022	
05572	171540	
05573	201727	
05574	052240	
05575	100111	
05576	142522	
05577	055000	
05600	200420	MES53 .SIXBT 'PDP-11 PWR FAILED FIRST, TST SUCCESSFULLY COMPLETED,0'
05601	550101	
05602	402027	
05603	224006	
05604	011114	
05605	050440	
05606	001122	
05607	232450	
05610	402423	
05611	244023	
05612	250303	
05613	052323	
05614	002514	
05615	143140	
05616	031715	
05617	201405	
05620	240504	
05621	500000	
05622	754043	MES54 .SIXBT ' = # OF OCTAL PASSES SINCE STARTED,00000'
05623	401700	
05624	401703	
05625	240114	
05626	402001	
05627	232305	
05630	234023	
05631	111003	
05632	054023	
05633	240122	
05634	240504	
05635	505252	

05636 525200
 05637 754014 MES55 .SIXBT 'L= LOCATION AT WHICH CAL OCCURRED DURING POWER UP.#'
 05640 170301
 05641 241117
 05642 164001
 05643 244027
 05644 101103
 05645 104003
 05646 011440
 05647 170303
 05650 252222
 05651 050440
 05652 042522
 05653 111607
 05654 402017
 05655 270522
 05656 402520
 05657 560000
 05660 012011 MES56 .SIXBT 'API DONE FLG CAUSED A PDP-11 INT AT PS LEVEL 5.#'
 05661 400417
 05662 160540
 05663 061407
 05664 400301
 05665 252305
 05666 044001
 05667 402004
 05670 205501
 05671 614011
 05672 162440
 05673 012440
 05674 202340
 05675 140526
 05676 051440
 05677 655600
 05700 012011 MES57 .SIXBT 'API DONE FLG FAILED TO CAUSE A PDP-11 INT AT PS LEVEL 4.#'
 05701 400417
 05702 160540
 05703 061407
 05704 400601
 05705 111405
 05706 044024
 05707 174003
 05710 012523
 05711 054001
 05712 402004
 05713 205501
 05714 614011
 05715 162440
 05716 012440
 05717 202340
 05720 140526
 05721 051440
 05722 645000
 05723 240302 MES58 .SIXBT 'TCBP FLG CAUSED A PDP-11 INT AT PS LEVEL 7.#'
 05724 204006

05725 140740
 05726 030125
 05727 230504
 05730 400140
 05731 200420
 05732 556101
 05733 401116
 05734 244001
 05735 244020
 05736 234014
 05737 052605
 05740 144067
 05741 560000
 05742 240302 MES59 .SIXBT 'TCBP FLG FAILED TO CAUSE A PDP-11 INT AT PS LEVEL 6.#'
 05743 204006
 05744 140740
 05745 060111
 05746 140504
 05747 402417
 05750 400301
 05751 252305
 05752 400140
 05753 200420
 05754 556101
 05755 401116
 05756 244001
 05757 244020
 05760 234014
 05761 052605
 05762 144066
 05763 560000
 05764 042201 MES60 .SIXBT 'DR11 API0 FLG CAUSED A PDP-15 INT WITH DR15 INT ENABLE =0.#'
 05765 614001
 05766 201100
 05767 400614
 05770 074003
 05771 012523
 05772 050440
 05773 014020
 05774 042055
 05775 616540
 05776 111624
 05777 402711
 06000 241040
 06001 042201
 06002 654011
 06003 162440
 06004 051601
 06005 021405
 06006 407500
 06007 560000
 06010 042201 MES61 .SIXBT 'DR11 API0 FLG FAILED TO CAUSE A PDP-15 INT.#'
 06011 614001
 06012 201100
 06013 400614

```

06350 510000
06351 404040 MES76C .SIXBT ' 2. INSTALL ECU 8B15-19. GOOD LUCK!#1
06352 625640
06353 111623
06354 240114
06355 144005
06356 031740
06357 020261
06360 655501
06361 715640
06362 400717
06363 170440
06364 142503
06365 134100

06366 706001 / SIOA 706001 /SKIP ON I/O DATA ACCEPTED, SKIPS ON I/O DATA
/ACCEPTED FLAG WHICH IS SET WHEN 11 READS TC8P.
06367 706002 CIOD 706002 /CLEAR I/O DATA ACCEPTED FLAG
06370 706004 LIOR 706004 /LOAD I/O REG. LOADS AC INTO I/O REG (NEW TC8P)
/FLAG, LOADS AC INTO I/O REG (BECOMES NEW TC8P)
06371 706112 RDRS 706112 /READ DR STATUS REG (BIT 17=INT ENABLE)
06372 706122 LORS 706122 /LOAD DR STATUS REG (BIT 17=INT ENABLE)
06373 706101 SAPI0 706101 /SKIP ON DR API LEVEL 0 FLAG
06374 706121 SAPI1 706121 /SKIP ON DR API LEVEL 1 FLAG
06375 706141 SAPI2 706141 /SKIP ON DR API LEVEL 2 FLAG
06376 706161 SAPI3 706161 /SKIP ON DR API LEVEL 3 FLAG
06377 706104 CAPI0 706104 /CLEAR DR API LEVEL 0 FLAG
06400 706124 CAPI1 706124 /CLEAR DR API LEVEL 1 FLAG
06401 706144 CAPI2 706144 /CLEAR DR API LEVEL 2 FLAG
06402 706164 CAPI3 706164 /CLEAR DR API LEVEL 3 FLAG
000000 .END
06403 740040 +L
06404 003444 +L
06405 000331 +L
06406 000310 +L
06407 000300 +L
06410 000310 +L
06411 125232 +L
06412 177777 +L
06413 052525 +L
06414 003456 +L
06415 000007 +L
06416 000005 +L
06417 000004 +L
06420 000006 +L
06421 003457 +L
06422 020000 +L
06423 030000 +L
06424 040000 +L
06425 050000 +L
06426 007777 +L
06427 000400 +L
06430 010000 +L
06431 777777 +L
06432 004000 +L

```

```

06433 770000 +L
06434 000377 +L
06435 177400 +L
06436 400000 +L
06437 070000 +L
06440 130000 +L
06441 150000 +L
06442 160000 +L
06443 000001 +L
06444 100000 +L
06445 602216 +L
06446 002244 +L
06447 602270 +L
06450 602314 +L
06451 602340 +L
06452 000777 +L
06453 000207 +L
06454 000000 +L
06455 200000 +L
06456 620000 +L
06457 000200 +L
06460 000003 +L
06461 000077 +L
06462 000177 +L
06463 777672 +L
06464 003450 +L
06465 603271 +L
06466 603304 +L
06467 603343 +L
06470 017777 +L
06471 777503 +L
06472 777520 +L
06473 777510 +L
06474 777740 +L
06475 000100 +L
06476 000200 +L
06477 077777 +L

```

SIZE=06503 NO ERROR LINES

```

1          .TITLE UNICHANNEL 15 MAINDEC-15-DAUCA-B
2          .ENABL ABS
3          ;
4          ;COPYRIGHT 1972, 1973 DIGITAL EQUIPMENT CORP., MAYNARD, MASS, 01
5          ;
6          ;STARTING ADDRESS = 200
7          ;RESTART ADDRESS = 1100
8          000000 R0=X0
9          000001 R1=X1
10         000002 R2=X2
11         000003 R3=X3
12         000004 R4=X4
13         000005 R5=X5
14         000006 SP=X6
15         000007 PC=X7
16         177570 DS=177570
17         177770 PSW=177770
18         167772 APIL00=167772 ;LOADING BYTE CAUSES API LEVEL 0 TO 15
19         167773 APIL18=167773 ;LOADING BYTE CAUSES API LEVEL 1 TO 15
20         167702 APIL28=167702 ;LOADING BYTE CAUSES API LEVEL 2 TO 15
21         167703 APIL38=167703 ;LOADING BYTE CAUSES API LEVEL 3 TO 15
22         167770 DRS1=167770 ;BIT 6 #1 ENABLES AN INT ON BR8 WHEN
23         ; API DONE (BIT 7) IS SET.
24         ;BIT 7 #1 API DONE (NONE OF THE 4 API
25         ; CHANNELS HAS A REQUEST PENDING
26         167774 DRS2=167774 ;BIT 0 #BIT 1 OF TCBP
27         ;BIT 1 #BIT 0 OF TCBP
28         ;BIT 6 #1 API 2 DONE FLAG (NO REQUEST
29         ; PENDING ON API LEVEL 2)
30         ;BIT 7 #1 API 0 DONE FLAG (NO REQUEST
31         ; PENDING ON API LEVEL 0)
32         ;BITS 8&9 #01 INDICATES 4K LOCAL MEMORY
33         ; #10 INDICATES 8K LOCAL MEMORY
34         ;BIT 14 #1 API 3 DONE FLAG (NO REQUEST
35         ; PENDING ON API LEVEL 3)
36         ;BIT 15 #1 API 1 DONE FLAG (NO REQUEST
37         ; PENDING ON API LEVEL 1)
38         167700 DRS3=167700 ;BIT 0 #1 ENABLE TCBP INT, CAUSES INT ON
39         ; BR7 WHEN TCBP IS RECEIVED FROM 15.
40         ;BIT 7 #1 NEW TCBP FLAG (SET WHEN TCBP IS
41         ; RECEIVED FROM 15. CLEARED WHEN 11
42         ; DOES DATI TO LOC 167764)
43         167704 DRS4=167704 ;CONTAINS BITS 2-17 OF TCBP
44         ;
45         000200          .=200
46         000200 000137      JMP 0#ISTART
47         001100          .=1100
48         001100          .=1100
49         ;
50         01100 012700      ISTART: MUV #.,SP
51         001100          001100
52         01104 004537      JSR R5,#RIVA
53         005714          005714
54         01110 022737      CMP #1,#INIT

```

```

000001
000410

```

00014 074000
 00015 011114
 00016 050440
 00017 241740
 00020 030125
 00021 230540
 00022 014020
 00023 042055
 00024 016540
 00025 111024
 00026 500000
 00027 042261
 00030 014001
 00031 201161
 00032 400014
 00033 074000
 00034 011114
 00035 050440
 00036 241740
 00037 030125
 00040 230540
 00041 014020
 00042 042055
 00043 016540
 00044 111024
 00045 500000
 00046 042261
 00047 014001
 00050 201162
 00051 400014
 00052 074000
 00053 011114
 00054 050440
 00055 241740
 00056 030125
 00057 230540
 00060 014020
 00061 042055
 00062 016540
 00063 111024
 00064 500000
 00065 042261
 00066 014001
 00067 201163
 00070 400014
 00071 074000
 00072 011114
 00073 050440
 00074 241740
 00075 030125
 00076 230540
 00077 014020
 00100 042055
 00101 016540
 00102 111024

MES02 .SIXBT 'DR11 API1 FLG FAILED TO CAUSE A PDP-15 INT,0'

MES03 .SIXBT 'DR11 API2 FLG FAILED TO CAUSE A PDP-15 INT,0'

MES04 .SIXBT 'DR11 API3 FLG FAILED TO CAUSE A PDP-15 INT,0'

00103 500000
 00104 311720
 00105 401001
 00106 260540
 00107 230514
 00110 050324
 00111 050440
 00112 014010
 00113 171655
 00114 053011
 00115 232401
 00116 162440
 00117 240523
 00120 244016
 00121 251502
 00122 052250
 00123 402701
 00124 112411
 00125 160756
 00126 565656
 00127 500000
 00130 040124
 00131 114024
 00132 174020
 00133 042055
 00134 016140
 00135 141703
 00136 406166
 00137 076766
 00140 044000
 00141 011114
 00142 050440
 00143 241740
 00144 230524
 00145 402403
 00146 022040
 00147 010303
 00150 052024
 00151 050450
 00152 000000
 00153 030120
 00154 116340
 00155 060111
 00156 140504
 00157 402417
 00160 400314
 00161 224001
 00162 201103
 00163 400014
 00164 075600
 00165 040124
 00166 114024
 00167 174020
 00170 042055
 00171 016140

MES05 .SIXBT 'YOU HAVE SELECTED A NON-EXISTANT TEST NUMBER, WAITING.....0'

MES06 .SIXBT 'DATI TO PDP-11 LOC 167764 FAILED TO SET TCBP ACCEPTED,0'

MES08 .SIXBT 'CAPI3 FAILED TO CLR API3 FLG,0'

MES07 .SIXBT 'DATI TO PDP-11 LOC 167774 SET TCBP ACCEPTED,0'

00172	141703	
00173	400166	
00174	076767	
00175	644023	
00176	052440	
00177	240302	
00200	204001	
00201	030305	
00202	202405	
00203	005000	
00204	525252	MES70 ,SIXBT '*****THE FOLLOWING SEQUENCE OF TESTS CAUSED THE LAST TEST#'
00205	525224	
00206	100540	
00207	001714	
00210	141727	
00211	111607	
00212	402305	
00213	212505	
00214	100305	
00215	401706	
00216	402405	
00217	232423	
00220	400301	
00221	252305	
00222	044024	
00223	100540	
00224	140123	
00225	244024	
00226	052324	
00227	000000	
00230	404040	MES71 ,SIXBT ' 1ST-#'
00231	404061	
00232	232455	
00233	000000	
00234	404040	MES72 ,SIXBT ' 2ND-#'
00235	404062	
00236	100455	
00237	000000	
00240	404040	MES73 ,SIXBT ' 3RD-#'
00241	404063	
00242	220455	
00243	000000	
00244	404040	MES74 ,SIXBT ' FAILED-#'
00245	404066	
00246	011114	
00247	050455	
00250	000000	
00251	111004	MES75 ,SIXBT 'INDICATED TO FAIL ON THE FIRST PASS THROUGH THE SEQUENCE.#'
00252	110301	
00253	240504	
00254	402417	
00255	400601	
00256	111440	
00257	171040	
00260	241005	

00261	400011	
00262	222324	
00263	402001	
00264	232340	
00265	241022	
00266	172507	
00267	104024	
00270	100540	
00271	230521	
00272	250510	
00273	030550	
00274	000000	
00275	525252	MES76 ,SIXBT '***NEXM FLAG IS SET, PC = #'
00276	100530	
00277	154006	
00300	140107	
00301	401123	
00302	402305	
00303	245640	
00304	402003	
00305	407540	
00306	000000	
00307	404040	MES76A ,SIXBT ' 1. CHECK JUMPER CARD G720, A006, IN 0015.#'
00310	615640	
00311	031005	
00312	031340	
00313	122515	
00314	200522	
00315	400301	
00316	220440	
00317	076762	
00320	705440	
00321	010260	
00322	665440	
00323	111640	
00324	020201	
00325	655000	
00326	404040	MES76B ,SIXBT ' JUMPER BU1 TO BU2 SHOULD BE REMOVED (ECO 0015-12)#'
00327	404040	
00330	122515	
00331	200522	
00332	400225	
00333	614024	
00334	174002	
00335	256240	
00336	231017	
00337	251404	
00340	400205	
00341	402265	
00342	151720	
00343	050440	
00344	500503	
00345	174002	
00346	026165	
00347	556162	

```

53 01110 001400      BEQ ISTA.1          ;FIRST TIME THROUGH
54                      ;INIT I1 ADDRESS FOR COMMON MEM INTER-COMM. I1 ADDRESS FOR COMM
55                      ;LUC (WHICH CORRESPONDS TO LOC ADDRESSED BY I5)
56                      ;I5 ADDRESS X 2 + 20000 + 20000 IF 8K LOCAL MEM
57 01120 000037      CLR #NDS13
      000304
58 01124 113737      MOVW #167775,#000365
      167775
      000365
59 01132 052737      BIS #170000,#0813
      170000
      000364
60 01140 106337 GC:   ASLB #000365
      000365
61 01144 103001      BCC IS,A
62 01146 000774      BR GC
63 01150 013701 IS,A1 MOV #NDS13,R1
      000364
64 01154 002701      ADD #402,R1
      000402
65 01160 012702      MOV #2,R2
      000002
66 01164 010137      MOV R1,#NTEST
      000400
67 01170 000201      ADD R2,R1
68 01172 010137      MOV R1,#ADR
      000400
69 01176 000201      ADD R2,R1
70 01200 010137      MOV R1,#NGOOD
      000374
71 01204 000201      ADD R2,R1
72 01206 010137      MOV R1,#BAD
      000376
73 01212 000201      ADD R2,R1
74 01214 010137      MOV R1,#ERRFLG
      000372
75 01220 000201      ADD R2,R1
76 01222 010137      MOV R1,#RWSA15
      000366
77 01226 000201      ADD R2,R1
78 01230 010137      MOV R1,#RWSA11
      000370
79 01234 000201      ADD R2,R1
80 01236 010137      MOV R1,#ERRIND
      000410
81 01242 000201      ADD R2,R1
82 01244 010137      MOV R1,#BRSTV
      000412
83 01250 000201      ADD R2,R1
84 01252 010137      MOV R1,#BR7TV
      000414
85 01256 000201      ADD R2,R1
86 01260 010137      MOV R1,#IDNE15
      000402
87 01264 000201      ADD R2,R1

88 01266 010137      MOV R1,#IDNE11
      000404

```

```

89
90 01272 012777 I STA,1: MOV #125252,#GOOD      ;INDICATE LOAD TIME
      125252
      005074
91 01300 012777      MOV #=1,#IDNE11        ;IND 11 READY
      177777
      005075
92 01300 022777 I STA,2: CMP #52525,#BAD
      052525
      005002
93 01314 001374      BNE I STA,2          ;LOAD TIME?
94 01310 005077      CLR #BAD              ;YES
      005054
95 01322 012777 I STA,3: MOV #=1,#IDNE11
      177777
      005054
96 01330 005037      CLR #INIT
      006410
97 01334 005237      INC #INIT
      006410
98 01340 012737 START: MOV #340,#PSW
      000340
      177770
99 01340 022777 STAR,1: CMP #=1,#IDNE15
      177777
      005020
100 1354 001374      BNE STAR,1          ;IS READY?
101 1350 005077      CLR #IDNE15         ;YES, CLEAR 15 READY
      005020
102 1302 022777      CMP #177,#TEST
      000177
      005010
103 1370 001414      BEQ STAR,2          ;POWER FAIL TEST?
104 1372 017701      MOV #TEST,R1         ;NO
      005002
105 1370 005301      ASL R1
106 1400 063701      ADD #NTSTBL,R1
      006434
107 1404 011137      MOV (R1),#NTSTPNT
      006400
108 1410 000005      RESET
109 1412 004577      JSR RS,#NTSTPNT      ;EXECUTE SELECTED TEST
      004770
110 1410 000137      JMP #NSTART        ;NXT TEST
      001340
111 1422 000005 STAR,2: RESET
112 1424 004537      JSR RS,#NPRFL
      005752
113 1430 000137      JMP #NSTART
      001340
114
115 ;*****
116 ;TST1. 15 WRITES IN LOWER 2K, 11 READS & CHECKS IT.
117 1434 017701 I ST1: MOV #RWSA11,R1
      004730

118 1440 010102      MOV R1,R2
119 1442 004537      JSR RS,#NCVRT        ;CONVERT 15 ADDR TO 11

```

```

005700
120 1446 022777 TST1.A: CMP #-1,@IDNE15
177777
004726
121 1454 011374 BNE TST1.A ;15 DONE WRITING?
122 1456 005077 CLR @IDNE15 ;YES. CLEAR 15 DONE
004726
123 1462 004537 JSR R5,@READ2K ;READ & CHECK 2K
005002
124 1466 012777 MOV #-1,@IDNE11 ;IND 11 DONE READING
177777
004710
125 1474 000205 RTS R5
126 ;*****
127 ;TST2. 15 WRITES IN UPPER 2K, 11 READS & CHECKS IT.
128 1476 017701 TST2: MOV @RNSA11,R1
004006
129 1482 010102 MOV R1,R2
130 1484 004537 JSR R5,@CONVRT ;CONVERT 15 ADDR TO 11
005700
131 1510 022777 TST2.A: CMP #-1,@IDNE15
177777
004064
132 1516 011374 BNE TST2.A ;15 DONE WRITING?
133 1520 005077 CLR @IDNE15 ;YES. CLEAR 15 DONE
004656
134 1524 004537 JSR R5,@READ2K ;READ & CHECK 2K
005002
135 1530 012777 MOV #-1,@IDNE11 ;IND 11 DONE READING
177777
004064
136 1536 000205 RTS R5
137 ;*****
138 ;TST3. 11 WRITES IN LOWER 2K, 15 READS & CHECKS IT.
139 1540 017701 TST3: MOV @RNSA11,R1
004624
140 1544 010102 MOV R1,R2
141 1546 004537 JSR R5,@CONVRT
006264
142 1552 012777 MOV #-1,@IDNE11 ;IND 11 DONE WRITING
177777
004624
143 1560 000205 RTS R5
144 ;*****
145 ;TST4. 11 WRITES IN UPPER 2K, 15 READS & CHECKS IT
146 1562 017701 TST4: MOV @RNSA11,R1
004602
147 1566 010102 MOV R1,R2
148 1570 004537 JSR R5,@CONVRT
006204
149 1574 012777 MOV #-1,@IDNE11 ;IND 11 DONE WRITING
177777
004602
150 1582 000205 RTS R5
151 ;*****

152 ;TST5. 15 WRITES IN LOWER 2K WHILE 11 IS WRITING IN UPPER 2K, 11
153 ;READS & CHECKS LOWER 2K WHILE 15 IS READING & CHECKING UPPER 2K

```

```

154 1004 017701 TST5:  MOV #RWSA11,R1
      004500
155 1010 010102      MOV R1,R2
156 1012 004537      JSR R5,##CONVRT      ;CONVERT 15 ADDR TO 11
      006204
157 1016 004537      JSR R5,##DTST
      005530
158 1022 017701      MOV #RWSA11,R1
      004542
159 1026 010102      MOV R1,R2
160 1030 004537      JSR R5,##CNVRT      ;CONVERT 15 ADDR TO 11
      005700
161 1034 004537      JSR R5,##READ2K     ;READ & CHECK LOWER 2K
      005602
162 1040 012777      MOV #-1,#IDNE11    ;IND 11 DONE READING
      177777
      004536
163 1046 000205      RTS R5
164      ;*****
165 *****;*****
166 ;TST6. 11 WRITES IN LOWER 2K WHILE 15 IS WRITING IN UPPER 2K,15
167 ;HEADS & CHECKS LOWER 2K WHILE 11 IS READING & CHECKING UPPER 2K
168 1050 017701 TST6:  MOV #RWSA11,R1
      004514
169 1054 010102      MOV R1,R2
170 1056 004537      JSR R5,##CONVRT
      006204
171 1062 004537      JSR R5,##DTST
      005530
172 1066 017701      MOV #RWSA11,R1
      004476
173 1072 010102      MOV R1,R2
174 1074 004537      JSR R5,##CNVRT      ;CONVERT 15 ADDRESS TO 11
      005700
175 1700 004537      JSR R5,##READ2K     ;READ & CHECK UPPER 2K
      005602
176 1704 012777      MOV #-1,#IDNE11    ;IND 11 DONE READING
      177777
      004472
177 1712 000205      RTS R5
178 *****;*****
179 ;TST7. 15 WRITES IN LOWER 2K, 11 READS & CHECKS IT WHILE 15 IS
180 ;WRITING IN UPPER 2K.
181 1714 017701 TST7:  MOV #RWSA11,R1
      004450
182 1720 010102      MOV R1,R2
183 1722 004537      JSR R5,##CNVRT      ;CONVERT 15 ADDR TO 11
      005700
184 1726 022777 TST7.A: CMP #-1,#IDNE15
      177777
      004446
185 1734 001374      BNE TST7.A         ;15 DONE WRITING?
186 1736 005077      CLR #IDNE15       ;YES. CLR 15 DONE
      004440
187 1742 004537      JSR R5,##READ2K     ;READ & CHECK LOWER 2K WHILE 15
      005002
188 1746 012777      MOV #-1,#IDNE11    ;IND 11 DONE

```

```

177777
004430
189 1754 000205      RTS R5
190      ;*****
191      ;TST10. 15 WRITES IN UPPER 2K, 11 READS & CHECKS IT WHILE 15 IS
192      ;WRITING IN LOWER 2K.
193 1756 017701      TST10:  MOV @RWSA11,R1
      004404
194 1762 010102      MOV R1,R2
195 1764 004537      JSR R5,@CONVRT      ;CONVERT 15 ADDR TO 11
      005700
196 1770 022777      TST10AI CMP #-1,@IDNE15
      177777
197 1770 001374      BNE TST10A          ;15 DONE WRITING?
198 2000 005077      CLR @IDNE15        ;YES. CLR 15 DONE
      004376
199 2004 004537      JSR R5,@WHEAD2K   ;RD & CHK UPPER 2K WHILE 15 WRTS
      005062
200 2010 012777      MOV #-1,@IDNE11   ;IND 11 DONE
      177777
      004300
201 2016 000205      RTS R5
202      ;*****
203      ;TST11. 11 WRITES IN LOWER 2K, 15 READS & CHECKS IT WHILE 11 IS
204      ;WRITING IN UPPER 2K.
205 2020 017701      TST11:  MOV @RWSA11,R1
      004344
206 2024 010102      MOV R1,R2
207 2026 004537      JSR R5,@CONVRT
      006204
208 2032 004537      JSR R5,@WDTST
      005530
209 2030 017701      MOV @RWSA11,R1
      004326
210 2042 010102      MOV R1,R2
211 2044 004537      JSR R5,@CONVRT
      006204
212 2050 012777      MOV #-1,@IDNE11   ;IND 11 DONE
      177777
      004326
213 2050 000205      RTS R5
214      ;*****
215      ;*****
216      ;TST12. 11 WRITES IN UPPER 2K, 15 READS & CHECKS IT WHILE 11 IS
217      ;WRITING IN LOWER 2K.
218 2060 017701      TST12:  MOV @RWSA11,R1
      004304
219 2064 010102      MOV R1,R2
220 2066 004537      JSR R5,@CONVRT
      006204
221 2072 004537      JSR R5,@WDTST
      005530
222 2070 017701      MOV @RWSA11,R1
      004266

223 2102 010102      MOV R1,R2
224 2104 004537      JSR R5,@CONVRT

```

```

006204
225 2110 012777      MOV #-1,#IDNE11      ;IND 11 DONE
      177777
      004206
226 2110 000205      RTS R5
227      ;*****
228      ;TST13, 11 WRITES ALL ONES IN LOW ORDER BYTES OF LOWER 2K WHILE
229      ;IS WRITING -1 IN UPPER 2K, 15 READS & CHECKS LOWER 2K WHILE 11
230      ;READING & CHECKING LOW ORDER BYTES IN UPPER 2K.
231 2120 017701 TST13: MOV #RWSA11,R1
      004244
232 2124 004537      JSR R5,##CNVRT
      005700
233 2130 012737      MOV #4000,##MCNT
      004000
      006420
234 2130 004537 TST13A: JSR R5,##DLY
      006276
235 2142 112721      MOVB #-1,(R1)+      ;WRT A LOC
      177777
236 2146 005201      INC R1
237 2150 005337      DEC #MCNT
      006420
238 2154 001370      BNE TST13A          ;DONE?
239 2150 004537      JSR R5,##DSTST
      005530
240 2162 017701      MOV #RWSA11,R1
      004202
241 2160 010102      MOV R1,R2
242 2170 004537      JSR R5,##CNVRT
      005700
243 2174 012737      MOV #4000,##MCNT
      004000
      006422
244 2202 004537 TST13B: JSR R5,##DLY
      006276
245 2200 122721      CMPE #377,(R1)+
      000377
246 2212 001007      BNE TST13C          ;DATA CORRECT?
247 2214 005201      INC R1              ;YES
248 2216 005202      INC R2
249 2220 005337      DEC #MCNT
      006422
250 2224 001300      BNE TST13B          ;DONE?
251 2220 000137      JMP #TST13D         ;YES
      002276
252 2232 012777 TST13C: MOV #-1,#ERRIND
      177777
      004150
253 2240 022777      CMP #-1,#ERRFLG
      177777
      004124
254 2246 001413      BEQ TST13D
255 2250 012777      MOV #-1,#ERRFLG
      177777

004114
256 2256 010277      MOV R2,#ADR

```

```

#04140
257 2262 #12777      MOV #377,#G0DD
#00377
#04104
258 2470 #05301      DEC R1
259 2272 114177      MOVB -(R1),#BAD
#04100
260 2270 #12777 TST130: MOV #-1,#IDNE11
177777
#04100
261 2364 #00205      RTS R5
262 ;*****
263 ;*****
264 ;TST14. I1 WRITES ALL ONES IN HIGH ORDER BYTES OF LOWER 2K WHILE
265 ;I5 IS WRITING -1 IN UPPER 2K, I5 READS & CHECKS LOWER 2K WHILE
266 ;READING & CHECKING HIGH ORDER BYTES IN UPPER 2K,
267 2306 #17701 TST14: MOV #RNSA11,R1
#04050
268 2312 #04537      JSR R5,#MCNVRT
#05700
269 2310 #12737      MOV #4000,#NRCNT
#04000
270 2324 #04537 TST14A: JSR R5,#NDLY
#06270
271 2330 #05201      INC R1
272 2332 112721      MOVB #-1,(R1)+      JNRT LOC
177777
273 2330 #05337      DEC #NRCNT
#06420
274 2342 #01370      BNE TST14A          JDONE?
275 2344 #04537      JSR R5,#MDTBT
#05530
276 2350 #17701      MOV #RNSA11,R1
#04014
277 2354 #10102      MOV R1,R2
278 2350 #04537      JSR R5,#MCNVRT
#05700
279 2362 #12737      MOV #4000,#NRCNT
#04000
280 2370 #04537 TST14B: JSR R5,#NDLY
#06270
281 2374 #05201      INC R1
282 2370 122721      CMPE #377,(R1)+
#00377
283 2402 #01000      BNE TST14C          JDATA CORRECT?
284 2404 #05202      INC R2
285 2400 #05337      DEC #NRCNT
#06422
286 2412 #01300      BNE TST14B          JDONE?
287 2414 #00137      JMP #TST14D         JYES
#02470
288 2420 #12777 TST14C: MOV #-1,#ERRIND
177777

#03702
289 2420 #22777      CMP #-1,#ERRFLG

```



```

177777
003736
290 2434 001415      BEQ TST140
291 2436 012777      MOV #-1,0ERRFLG
177777
003726
292 2444 010277      MOV R2,#ADM
003760
293 2450 012777      MOV #177400,#GOOD
177400
003716
294 2450 014177      MOV =(R1),#BAD
003714
295 2462 042777      BIC #377,#BAD
000377
003706
296 2470 012777      TST140: MOV #-1,#IDNE11
177777
003706
297 2470 000205      RTS R5
298
299
300
301 2500 105737      TST15: TSTB #NDMS3
167760
302 2504 100402      BMI TST15A      /TCBP FLG SET?
303 2506 004537      JSR R5,#METST
005512
304 2512 012777      TST15A: MOV #-1,#IDNE11      /IND 11 DONE
177777
003664
305 2520 000205      RTS R5
306
307
308 2522 013737      TST16: MOV #NDRS4,#NDRS4      /DATI
167764
167764
309 2530 105737      TSTB #NDMS3
167760
310 2534 100002      BPL TST16A      /TCBP FLG CLR?
311 2536 004537      JSR R5,#METST
005512
312 2542 012777      TST16A: MOV #-1,#IDNE11      /IND 11 DONE
177777
003634
313 2550 000205      RTS R5
314
315
316 2552 013737      TST17: MOV #NDRS2,#NDRS2      /DATI
167774
167774
317 2560 105737      TSTB #NDRS3
167760
318 2564 100402      BMI TST17A      /TCBP FLG SET?
319 2566 004537      JSR R5,#METST

005512
320 2572 012777      TST17A: MOV #-1,#IDNE11

```

```

177777
003604
J21 2000 000205      RTS R5
J22                ;*****
J23                ;TST20. TEST FOR TCBP ACCEPTED WITH SIDA,
J24 2002 000205      TST20:  RTS R5
J25                ;*****
J26                ;TST21. CLEAR TCBP ACCEPTED FLG WITH CIOD & TEST SIDA FOR NO SKY
J27 2004 000205      TST21:  RTS R5
J28                ;*****
J29                ;TST22. TEST FOR TCBP ACCEPTED TO BE SET BY DOING A DATI TO LOC
J30                ;167764 IN PDP-11.
J31 2006 013737      TST22:  MOV #NDRS4,#NDRS4      ;DATI
                167764
                167764
J32 2014 012777      MOV #=1,#IDNE11
                177777
                003562
J33 2022 000205      RTS R5
J34                ;*****
J35                ;TST23. TEST FOR TCBP ACCEPTED TO NOT BE SET BY DOING A DATI TO
J36                ;167774 IN PDP-11.
J37 2024 013737      TST23:  MOV #NDRS2,#NDRS2      ;DATI
                167774
                167774
J38 2032 012777      MOV #=1,#IDNE11
                177777
                003544
J39 2040 000205      RTS R5
J40                ;*****
J41                ;*****
J42                ;TST24. TESTS FOR CORRECT TRANSMISSION OF TCBP FROM 15 TO 11.
J43 2042 005037      TST24:  CLR #NTMPCNT
                006000
J44 2040 105737      TST24A: TSTB #NDK33
                167760
J45 2052 100410      BMI TST24C      ;NEW TCBP?
J46 2054 022777      CMP #=1,#ERRFLG
                177777
                003510
J47 2062 001411      BEQ TST24B
J48 2064 005237      INC #NTMPCNT
                006360
J49 2070 001422      BEQ TST24D
J50 2072 022777      CMP #=1,#IDNE15      ;NO
                177777
                003502
J51 2700 001302      BNE TST24A      ;15 DONE?
J52 2702 005077      CLR #IDNE15      ;CLR 15 DONE
                003474
J53 2700 000205      TST24B: RTS R5
J54 2710 013777      TST24C: MOV #NDRS2,#RWSA15      ;BITS 0 & 1 OF TCBP
                167774
                003450
J55 2710 042777      BIC #177774,#RWSA15      ;MASK OFF OTHERS

```

177774
003442

```

J56 2724 013777      MOV #NDRS4,PRKSA11      ;BITS 2-17 OF TCBP
      167764
      003430
J57 2732 000137      JMP #TST24
      002642
J58 2736 012777      TST240: MOV #-1,#IDNE11
      177777
      003440
J59 2744 000200      RTS R5
J60      ;*****
J61      ;TST25. TEST FOR API 0 DONE FLG TO BE SET.
J62 2746 105737      TST25: TSTB #NDRS2
      167774
J63 2752 100402      BMI TST25A              ;API 0 DONE SET?
J64 2754 004537      JSR R5,#NETST          ;NO
      005512
J65 2760 012777      TST25A: MOV #-1,#IDNE11      ;IND 11 DONE
      177777
      003410
J66 2766 000200      RTS R5
J67      ;*****
J68      ;TST26. TEST FOR API 1 DONE FLG TO BE SET.
J69 2770 005737      TST26: TST #NDRS2
      167774
J70 2774 100402      BMI TST26A              ;API 1 DONE SET?
J71 2776 004537      JSR R5,#NETST          ;NO
      005512
J72 3002 012777      TST26A: MOV #-1,#IDNE11
      177777
      003374
J73 3010 000200      RTS R5
J74      ;*****
J75      ;TST27. TEST FOR API 2 DONE FLG TO BE SET.
J76 3012 032737      TST27: BIT #100,#NDRS2
      000100
      167774
J77 3020 001002      BNE TST27A              ;API 2 DONE SET?
J78 3022 004537      JSR R5,#NETST          ;NO
      005512
J79 3026 012777      TST27A: MOV #-1,#IDNE11
      177777
      003350
J80 3034 000200      RTS R5
J81      ;*****
J82      ;*****
J83      ;TST30. TEST FOR API 3 DONE FLG TO BE SET.
J84 3036 032737      TST30: BIT #40000,#NDRS2
      040000
      167774
J85 3044 001002      BNE TST30A              ;API 3 DONE SET?
J86 3046 004537      JSR R5,#NETST          ;NO
      005512
J87 3052 012777      TST30A: MOV #-1,#IDNE11
      177777
      003324

J88 3060 000200      RTS R5
J89      ;*****

```

```

390          ;TST31. TEST FOR API DONE FLG TO BE SET.
391 3002 105737 TST31: TSTB #WDRS1
          107770
392 3060 100402      BHI TST31A          ;API DONE SET?
393 3070 004537      JSR R5,#METST      ;NO
          005512
394 3074 012777 TST31A: MOV #-1,#IDNE11      ;IND 11 DONE
          177777
          003302
395 3102 000200      RTS R5
396          ;*****
397          ;TST32. TEST FOR SAPI0 TO SKIP.
398 3104 110037 TST32: MOVB R0,#APIL0B      ;SET API L0
          167772
399 3110 012777      MOV #-1,#IDNE11
          177777
          003206
400 3110 000200      RTS R5
401          ;*****
402          ;TST33. TEST FOR SAPI1 TO SKIP.
403 3120 110037 TST33: MOVB R0,#APIL1B      ;SET API L1
          167773
404 3124 012777      MOV #-1,#IDNE11
          177777
          003202
405 3132 000200      RTS R5
406          ;*****
407          ;TST34. TEST FOR SAPI2 TO SKIP.
408 3134 110037 TST34: MOVB R0,#APIL2B      ;SET API L2
          167762
409 3140 012777      MOV #-1,#IDNE11
          177777
          003230
410 3140 000200      RTS R5
411          ;*****
412          ;TST35. TEST FOR SAPI3 TO SKIP.
413 3150 110037 TST35: MOVB R0,#APIL3B      ;SET API L3
          167763
414 3154 012777      MOV #-1,#IDNE11
          177777
          003222
415 3162 000200      RTS R5
416          ;*****
417          ;TST36. TEST SAPI0 FOR NO SKIP.
418 3164 110037 TST36: MOVB R0,#APIL1B      ;SET API L1
          167773
419 3170 110037      MOVB R0,#APIL2B      ;SET API L2
          167762
420 3174 110037      MOVB R0,#APIL3B      ;SET API L3
          167763
421 3200 012777      MOV #-1,#IDNE11
          177777
          003176
422 3200 000200      RTS R5
423          ;*****

424          ;*****
425          ;TST37. TEST SAPI1 FOR NO SKIP.

```

```

426 3210 110037 TST371 MOVB R0,#NAPIL0B      ISET API L0
      167772
427 3214 110037      MOVB R0,#NAPIL2B      ISET API L2
      167762
428 3220 110037      MOVB R0,#NAPIL3B      ISET API L3
      167763
429 3224 012777      MOV  #=1,#IDNE11
      177777
      003152
430 3232 000205      RTS R5
431      ;*****
432      ITEST40. TEST SAPI2 FOR NO SKIP.
433 3234 110037 TST401 MOVB R0,#NAPIL0B      ISET API L0
      167772
434 3240 110037      MOVB R0,#NAPIL1B      ISET API L1
      167773
435 3244 110037      MOVB R0,#NAPIL3B      ISET API L3
      167763
436 3250 012777      MOV  #=1,#IDNE11
      177777
      003126
437 3256 000205      RTS R5
438      ;*****
439      ITEST41. TEST SAPI3 FOR NO SKIP.
440 3260 110037 TST411 MOVB R0,#NAPIL0B      ISET API L0
      167772
441 3264 110037      MOVB R0,#NAPIL1B      ISET API L1
      167773
442 3270 110037      MOVB R0,#NAPIL2B      ISET API L2
      167762
443 3274 012777      MOV  #=1,#IDNE11
      177777
      003102
444 3302 000205      RTS R5
445      ;*****
446      ITEST42. TEST CAPI0 TO CLEAR API0 FLG.
447 3304 110037 TST421 MOVB R0,#NAPIL0B      ISET API L0
      167772
448 3310 110037      MOVB R0,#NAPIL1B      ISET API L1
      167773
449 3314 110037      MOVB R0,#NAPIL2B      ISET API L2
      167762
450 3320 110037      MOVB R0,#NAPIL3B      ISET API L3
      167763
451 3324 012777      MOV  #=1,#IDNE11
      177777
      003052
452 3332 000205      RTS R5
453      ;*****
454      ITEST43. TEST CAPI1 TO CLEAR API1 FLG.
455 3334 110037 TST431 MOVB R0,#NAPIL0B
      167772
456 3340 110037      MOVB R0,#NAPIL1B
      167773
457 3344 110037      MOVB R0,#NAPIL2B

      167762
458 3350 110037      MOVB R0,#NAPIL3B

```

```

167763
459 3354 012777      MOV #=1,0IDNE11
177777
003022
460 3362 000205      RTS R5
461          ;*****
462          ;TST44. TEST CAPI2 TO CLEAR API2 FLG.
463 3364 110037      TST44:  MOVB R0,#NAPIL0B
167772
464 3370 110037      MOVB R0,#NAPIL1B
167773
465 3374 110037      MOVB R0,#NAPIL2B
167762
466 3400 110037      MOVB R0,#NAPIL3B
167763
467 3404 012777      MOV #=1,0IDNE11
177777
002772
468 3412 000205      RTS R5
469          ;*****
470          ;TST45. TEST CAPI3 TO CLEAR API3 FLG.
471 3414 110037      TST45:  MOVB R0,#NAPIL0B
167772
472 3420 110037      MOVB R0,#NAPIL1B
167773
473 3424 110037      MOVB R0,#NAPIL2B
167762
474 3430 110037      MOVB R0,#NAPIL3B
167763
475 3434 012777      MOV #=1,0IDNE11
177772
002742
476 3442 000205      RTS R5
477          ;*****
478          ;TST46. TEST FOR DR15 INT ENABLE SET USING RDRB IOT.
479          ;TST46:  TEST FOR DR15 INT ENABLE SET USING RDRB IOT.
480 3444 000205      TST46:  RTS R5
481          ;*****
482          ;TST47. TEST FOR DR15 INT ENABLE BEING CLEARED BY LDRB IOT.
483 3446 000205      TST47:  RTS R5
484          ;*****
485          ;TST50. TEST FOR DR15 INT ENABLE BEING SET BY LDRB IOT.
486 3450 000205      TST50:  RTS R5
487          ;*****
488          ;TST51. TEST FOR CORRECT API LEVEL REQ WHEN 11 INDICATES LEVEL 0
489 3452 110037      TST51:  MOVB R0,#NAPIL0B      ;SET API LB
167772
490 3456 012777      MOV #=1,0IDNE11
177777
002720
491 3464 000205      RTS R5
492          ;*****
493          ;TST52. TEST FOR API LEVEL 0 DONE FLG (ONLY) TO CLR.
494 3466 110037      TST52:  MOVB R0,#NAPIL0B      ;SET API LB
167772

495 3472 013737      MOV #NDRS2,#NTEMP
167774

```

```

000432
496 3500 042737      BIC #37477,#TEMP
      037477
000432
497 3500 022737      CMP #140100,#TEMP
      140100
000432
498 3514 001402      BEQ TST52A          ;ONLY L0 DONE CLR?
499 3516 004537      JSR R5,#NETST      ;NO
      005512
500 3022 012777 TST52A: MOV #-1,#IDNE11
      177777
      002654
501 3530 000205      RTS R5
502      ;*****
503      ;
504      ;*****
505      ;TST53. TEST FOR CORRECT API LEVEL REQ WHEN 11 INDICATES LEVEL 1
506 3032 110037 TST53:  MOVB R0,#APIL1B      ;SET API L1
      167773
507 3530 012777      MOV #-1,#IDNE11
      177777
      002640
508 3544 000205      RTS R5
509      ;*****
510      ;TST54. TEST FOR API LEVEL 1 DONE FLG (ONLY) TO CLR.
511 3040 110037 TST54:  MOVB R0,#APIL1B      ;SET API L1
      167773
512 3502 013737      MOV #NDHS2,#TEMP
      167774
      000432
513 3500 042737      BIC #37477,#TEMP
      037477
      000432
514 3506 022737      CMP #40300,#TEMP
      040300
      000432
515 3574 001402      BEQ TST54A          ;ONLY L1 CLR?
516 3070 004537      JSR R5,#NETST      ;NO
      005512
517 3002 012777 TST54A: MOV #-1,#IDNE11
      177777
      002574
518 3010 000205      RTS R5
519      ;*****
520      ;TST55. TEST FOR CORRECT API LEVEL REQ WHEN 11 INDICATES LEVEL 2
521 3012 110037 TST55:  MOVB R0,#APIL2B      ;SET API L2
      167762
522 3010 012777      MOV #-1,#IDNE11
      177777
      002560
523 3024 000205      RTS R5
524      ;*****
525      ;*****
526      ;TST56. TEST FOR API LEVEL 2 DONE FLG (ONLY) TO CLR.

527 3020 110037 TST56:  MOVB R0,#APIL2B      ;SET API L2
      167762

```

```

528 3032 013737      MOV #NDRS2,#TEMP
      167774
      006432
529 3040 042737      BIC #37477,#TEMP
      037477
      006432
530 3040 022737      CMP #140200,#TEMP
      140200
      006432
531 3054 001402      BEQ TST56A          ;ONLY L2 DONE CLR?
532 3056 004537      JSR R5,#NETST      ;NO
      005512
533 3062 012777 TST56A: MOV #-1,#IDNE11
      177777
      002514
534 3070 000205      RTS R5
535      ;*****
536      ;TST57. TEST FOR CORRECT API LEVEL REQ WHEN 11 INDICATES LEVEL 3
537 3072 110037 TST57: MOVB R0,#APIL3B      ;SET API L3
      167703
538 3076 012777      MOV #-1,#IDNE11
      177777
      002500
539 3704 000205      RTS R5
540      ;*****
541      ;TST60. TEST FOR API LEVEL 3 DONE FLG (ONLY) TO CLR.
542 3706 110037 TST60: MOVB R0,#APIL3B      ;SET API L3
      167703
543 3712 013737      MOV #NDRS2,#TEMP
      167774
      006432
544 3720 042737      BIC #37477,#TEMP
      037477
      006432
545 3720 022737      CMP #100300,#TEMP
      100300
      006432
546 3734 001402      BEQ TST60A          ;ONLY L3 DONE CLR?
547 3736 004537      JSR R5,#NETST      ;NO
      005512
548 3742 012777 TST60A: MOV #-1,#IDNE11
      177777
      002434
549 3750 000205      RTS R5
550      ;*****
551      ;TST61. TEST FOR API DONE FLG TO BE CLR WHEN API LEVEL 0
552      ;HAS A REQUEST PENDING.
553 3752 110037 TST61: MOVB R0,#APIL0B      ;SET API L0
      167772
554 3756 032737      BIT #200,#NDRS1
      000200
      167770
555 3764 001402      BEQ TST61A          ;API CLR?
556 3766 004537      JSR R5,#NETST      ;NO
      005512

557 3772 012777 TST61A: MOV #-1,#IDNE11
      177777

```



```

002404
556 4000 000205      RTS R5
559                ;*****
560                ;TST62. TEST FOR API DONE FLG TO BE CLR WHEN API LEVEL 1
561                ;HAS A REQUEST PENDING.
562 4002 110037      TST62:  MOVB R0,#API1L1B      ;SET API L1
                    167773
563 4006 032737      BIT #200,#DRS1
                    000200
                    167770
564 4014 001402      BEQ TST62A      ;API CLR?
565 4016 004537      JSR R5,#METST
                    005512
566 4022 012777      TST62A:  MOV #-1,#IDNE11
                    177777
                    002354
567 4030 000205      RTS R5
568                ;*****
569                ;TST63. TEST FOR API DONE FLG TO BE CLR WHEN API LEVEL 2
570                ;HAS A REQUEST PENDING.
571 4032 110037      TST63:  MOVB R0,#API1L2B      ;SET API L2
                    167762
572 4036 032737      BIT #200,#DRS1
                    000200
                    167770
573 4044 001402      BEQ TST63A      ;API CLR?
574 4046 004537      JSR R5,#METST
                    005512
575 4052 012777      TST63A:  MOV #-1,#IDNE11
                    177777
                    002324
576 4060 000205      RTS R5
577                ;*****
578                ;TST64. TEST FOR API DONE FLG TO BE CLR WHEN API LEVEL 3
579                ;HAS A REQUEST PENDING.
580                ;TST64:  MOVB R0,#API1L3B      ;SET API L3
581 4062 110037      TST64:  MOVB R0,#API1L3B      ;SET API L3
                    167763
582 4066 032737      BIT #200,#DRS1
                    000200
                    167770
583 4074 001402      BEQ TST64A      ;API CLR?
584 4076 004537      JSR R5,#METST
                    005512
585 4102 012777      TST64A:  MOV #-1,#IDNE11
                    177777
                    002274
586 4110 000205      RTS R5
587                ;*****
588                ;TST65. TEST API DONE INT ENABLE TO SET.
589 4112 012737      TST65:  MOV #100,#DRS1      ;SET INT ENABLE
                    000100
                    167770
590 4120 032737      BIT #100,#DRS1
                    000100

                    167770
591 4126 001002      BNE TST65A      ;INT ENABLE SET?

```

```

592 4130 004537 JSR R5,#NETST /NO
005512
593 4134 012777 TST65A: MOV #-1,#IDNE11
177777
002242
594 4142 000205 RTS R5
;*****
595 ;TST66. TEST API DONE INT ENABLE TO CLEAR.
596 ;TST66: MOV #100,#DRS1 /SET ENABLE
597 4144 012737 TST66: MOV #100,#DRS1 /SET ENABLE
000100
167770
598 4152 042737 BIC #100,#DRS1 /CLEAR IT
000100
167770
599 4160 032737 BIT #100,#DRS1
000100
167770
600 4160 001402 BEQ TST66A /INT ENABLE CLEAR?
001 4170 004537 JSR R5,#NETST /NO
005512
602 4174 012777 TST66A: MOV #-1,#IDNE11
177777
002202
603 4202 000205 RTS R5
;*****
604 ;TST67. CLEAR API DONE ENABLE AND TEST FOR NO PDP-11 INTERRUPT
605 ;FROM API DONE.
606 ;TST67: MOV #BR5TV,R0
607 4204 017700 TST67: MOV #BR5TV,R0
002202
608 4210 012720 MOV #TST67B,(R0)+
004200
609 4214 012710 MOV #340,(R0)
000340
610 4220 012737 MOV #-100,#THPCNT
177700
006300
611 4220 042737 BIC #100,#DRS1 /CLR ENABLE
000100
167770
612 4234 005037 CLR #PPSW
177770
613 4240 005237 TST67A: INC #THPCNT
006300
614 4244 001375 BNE TST67A
615 4240 012737 MOV #340,#PPSW
000340
177770
616 4254 000137 JMP #TST67C
004200
617 4260 004537 TST67B: JSR R5,#NETST
005512
618 4264 022020 CMP (SP)+,(SP)+
619 4260 004537 TST67C: JSR R5,#IWA
005714
620 4272 012777 MOV #-1,#IDNE11

```

177777
002104

```

021 4300 000205      RTS R5
022                ;*****
023                ;*****
024                ;TST70, CLEAR TCBP ENABLE AND TEST FOR NO PDP=11 INTERRUPT
025                ;FROM TCBP FLG.
026 4302 017700      TST70: MOV #BR7TV,R0
                002100
027 4306 012720      MOV #TST70B,(R0)+
                004302
028 4312 012710      MOV #340,(R0)
                000340
029 4316 012737      MOV #-100,#NTHPCNT
                177700
                006300
030 4324 042737      BIC #100,#NDRS3
                000100
                107700
031 4332 004537      JSR R5,#NDTST
                005530
032 4336 005037      CLR #NPSW
                17777b
033 4342 005237      TST70A: INC #NTHPCNT
                006300
034 4346 00137b      BNE TST70A
035 4350 012737      MOV #340,#NPSW
                000340
                17777b
036 4356 000137      JMP #NTST70C
                004370
037 4362 004537      TST70B: JSR R5,#NETST
                005b12
038 4366 02262b      CMP (SP)+,(SP)+
039 4370 013737      TST70C: MOV #NDRS4,#NDRS4
                107704
                107704
040 4376 004537      JSR R5,#NIVA
                005714
041 4402 012777      MOV #-1,#IDNE11
                177777
                001774
042 4410 000205      RTS R5
043                ;*****
044                ;TST71, TEST FOR API DONE TO CAUSE A PDP=11 INT TO CORRECT VECTO
045                ;ADDRESS.
046 4412 017700      TST71: MOV #BR5TV,R0
                001774
047 4416 012720      MOV #TST71C,(R0)+      ;INIT TRAP
                004506
048 4422 012710      MOV #340,(R0)
                000340
049 4426 005037      CLR #NTHPCNT
                006300
050 4432 005037      CLR #NPSW
                17777b
051 4436 012737      MOV #100,#NDRS1      ;SET ENABLE

```

000100
107770

```

052 4444 005237 TST71A: INC @THPCNT
      006300
053 4450 001375      BNE TST71A      JWAIT FOR INT
054 4452 012737      MOV #340,@NPSN
      000340
      177776
055 4460 004537      JSR R5,@NETST
      005512
056 4464 042737 TST71B: BIC #100,@NDRS1
      000100
      167770
057 4472 004537      JSR R5,@NIVA
      005714
058 4476 012777      MOV #-1,@IDNE11
      177777
      001700
059 4504 000205      RTS R5
060 4506 022620 TST71C: CMP (SP)+,(SP)+
001 4510 000137      JMP @TST71B
      004404

062      ;*****
063      ;*****
064      ;TST72. TEST FOR TCBP TO CAUSE A PDP-11 INT TO CORRECT VECTOR
065      ;ADDRESS.
066 4514 017700 TST72: MOV @B^TV,R0
      001674
067 4520 012720      MOV #TST72C,(R0)+      JINIT TRAP
      004614
068 4524 012710      MOV #340,(R0)
      000340
069 4530 005037      CLR @THPCNT
      006300
070 4534 004537      JSR R5,@NDTST
      005530
071 4540 005037      CLR @NPSN
      177776
072 4544 012737      MOV #100,@NDRS3      JSET ENABLE
      000100
      167760
073 4552 005237 TST72A: INC @THPCNT
      006300
074 4556 001375      BNE TST72A      JWAIT
075 4560 012737      MOV #340,@NPSN
      000340
      177776
076 4566 004537      JSR R5,@NETST
      005512
077 4572 042737 TST72B: BIC #100,@NDRS3
      000100
      167760
078 4080 004537      JSR R5,@NIVA
      005714
079 4084 012777      MOV #-1,@IDNE11
      177777
      001572

080 4612 000205      RTS R5
081 4614 022620 TST72C: CMP (SP)+,(SP)+

```

```

002 4010 J00137      JMP #TST72B
      004572
003      ;*****
004      ;TST73. TEST FOR API DONE TO CAUSE A PDP-11 INT AT CORRECT BR
005      ;LEVEL.
006 4022 017700      TST73: MOV #BR5TV,R0
      001504
007 4026 012720      MOV #TST73E,(R0)+      ;INIT TRAP
      004740
008 4032 012710      MOV #J40,(R0)
      000340
009 4036 012737      MOV #=-100,#NTMPCNT
      177700
      006360
010 4044 012737      MOV #240,#PSW      ;SET FOR BR5
      000240
      177776
011 4052 012737      MOV #100,#DRS1     ;SET ENABLE
      000100
      167770
012 4060 005237      TST73A: INC #NTMPCNT
      006360
013 4064 001375      BNE TST73A      ;WAIT
014 4066 004537      JSR R5,#METST
      005530
015 4072 012740      MOV #TST73F,-(R0)      ;INIT TRAP
      004754
016 4076 012737      MOV #200,#PSW      ;SET FOR BR4
      000200
      177776
017 4704 005237      TST73B: INC #NTMPCNT
      006360
018 4710 001375      BNE TST73B
019 4712 004537      TST73C: JSR R5,#METST
      005512
020 4716 012737      TST73D: MOV #340,#PSW
      000340
      177776
021 4724 042737      BIC #100,#DRS1
      000100
      167770
022 4732 004537      JSR R5,#M1VA
      005714
023 4736 012777      MOV #=-1,#IDNE11
      177777
      001440
024 4744 000205      MTS M5
025 4746 022626      TST73E: CMP (SP)+,(SP)+
026 4750 000137      JMP #TST73C
      004712
027 4754 022626      TST73F: CMP (SP)+,(SP)+
028 4756 000137      JMP #TST73D
      004716
029      ;*****
030      ;*****

711      ;TST74. TEST FOR TCBP TO CAUSE A PDP-11 INT AT CORRECT BR LEVEL.
712 4762 017700      TST74: MOV #BR7TV,R0

```

```

001420
713 4760 012720      MOV #TST74E,(R0)+      /INIT TRAP
005112
714 4772 012710      MOV #340,(R0)
004340
715 4776 012737      MOV #-100,#NTHPCNT
177700
006360
716 5004 004537      JSR R5,#NDTST
005530
717 5010 012737      MOV #340,#NPSM        /SET FOR BR7
000340
177776
718 5010 012737      MOV #100,#NDRS3       /SET ENABLE
000100
167760
719 5024 005237 TST74A: INC #NTHPCNT
006360
720 5030 001375      BNE TST74A            /WAIT
721 5032 004537      JSR R5,#NDTST
005530
722 5036 012740      MOV #TST74F,-(R0)    /INIT TRAP
005120
723 5042 012737      MOV #300,#NPSM        /SET FOR BR6
000300
177776
724 5050 005237 TST74B: INC #NTHPCNT
006360
725 5054 001375      BNE TST74B
726 5056 004537 TST74C: JSR R5,#NETST
005512
727 5062 012737 TST74D: MOV #340,#NPSM
000340
177776
728 5070 042737      BIC #100,#NDRS3
000100
167760
729 5076 004537      JSR R5,#N1VA
005714
730 5102 012777      MOV #-1,#IUNE11
177777
001274
731 5110 000205      RTS R5
732 5112 022020 TST74E: CMP (SP)+,(SP)+
733 5114 000137      JMP #NTHPCNT
005800
734 5120 022020 TST74F: CMP (SP)+,(SP)+
735 5122 000137      JMP #NTHPCNT
005002
736 ;*****
737 ;TST75. CLEAR DR15 INT ENABLE AND TEST FOR NO PDP-15 INT FROM DR
738 ;API0 FLG.
739 5120 110037 TST75: MOV0 R0,#API000      /SET API LB
167772
740 5132 012777      MOV #-1,#IUNE11

```

177777
001244

```

741 5140 000205      RTS R5
742                ;*****
743                ;TST76. TEST FOR DR11 APIL0 FLG TO CAUSE PDP-15 INT.
744 5142 110037      TST76:  MOVB R0,#APIL0B      ;SET API L0
                167772
745 5140 012777      MOV #=1,#IDNE11
                177777
                001230
746 5154 000205      RTS R5
747                ;*****
748                ;TST77. TEST FOR DR11 APIL1 FLG TO CAUSE PDP-15 INT.
749 5150 110037      TST77:  MOVB R0,#APIL1B      ;SET API L1
                167773
750 5162 012777      MOV #=1,#IDNE11
                177777
                001214
751 5170 000205      RTS R5
752                ;*****
753                ;ST100. TEST FOR DR11 APIL2 FLG TO CAUSE PDP-15 INT.
754 5172 110037      ST100:  MOVB R0,#APIL2B      ;SET API L2
                167762
755 5170 012777      MOV #=1,#IDNE11
                177777
                001200
756 5204 000205      RTS R5
757                ;*****
758                ;ST101. TEST FOR DR11 APIL3 FLG TO CAUSE PDP-15 INT.
759 5200 110037      ST101:  MOVB R0,#APIL3B      ;SET API L3
                167763
760 5212 012777      MOV #=1,#IDNE11
                177777
                001164
761 5220 000205      RTS R5
762                ;*****
763                ;*****
764                ;ST102. TEST FOR API BREAKS TO CORRECT PORTS FROM 0 TO 177 FROM
765                ;DR11 APIL0.
766 5222 005003      ST102:  CLR R3
767 5224 110337      ST102A: MOVB R3,#APIL0B      ;SET APIL0 WITH PORT ADDRESS
                167772
768 5230 012777      MOV #=1,#IDNE11
                177777
                001140
769 5230 020327      CMP R3,#177
                000177
770 5242 001415      BEQ ST102C      ;DONE?
771 5244 005203      INC R3
772 5246 022777      ST102B: LMP #=1,#ERRFLG
                177777
                001110
773 5254 001410      BEQ ST102C
774 5250 022777      CMP #=1,#IDNE15
                177777
                001110
775 5264 001370      BNE ST102B

776 5260 005077      CLR #IDNE15
                001110

```

```

777 5272 000137      JMP #ST102A
      005224
778 5276 000205 ST102C: RTS R5
779          ;*****
780          ;ST103. TEST FOR API BREAKS TO CORRECT PORTS FROM 0 TO 1100 FROM
781          ;DN11 APIL1.
782 5300 005003 ST103: CLR R3
783 5302 110337 ST103A: MOV# R3,#APIL1B      ;SET APIL1 WITH PORT ADDRESS
      167773
784 5300 012777      MOV #=1,#IDNE11
      177777
      001070
785 5314 020327      CMP R3,#177
      000177
786 5320 001415      BEQ ST103C
787 5322 005203      INC R3
788 5324 022777 ST103B: CMP #=1,#ERRFLG
      177777
      001040
789 5332 001410      BEQ ST103C
790 5334 022777      CMP #=1,#IDNE15
      177777
      001040
791 5342 001370      BNE ST103B
792 5344 005077      CLR #IDNE15
      001032
793 5350 000137      JMP #ST103A
      005302
794 5354 000205 ST103C: RTS R5
795          ;*****
796          ;ST104. TEST FOR API BREAKS TO CORRECT PORTS FROM 0 TO 177 FROM
797          ;DR11 APIL2.
798 5350 005003 ST104: CLR R3
799 5360 110337 ST104A: MOV# R3,#APIL2B      ;SET APIL2 WITH PORT ADDRESS
      167762
800 5364 012777      MOV #=1,#IDNE11
      177777
      001012
801 5372 020327      CMP R3,#177
      000177
802 5370 001415      BEQ ST104C
803 5400 005203      INC R3
804 5402 022777 ST104B: CMP #=1,#ERRFLG
      177777
      000762
805 5410 001410      BEQ ST104C
806 5412 022777      CMP #=1,#IDNE15
      177777
      000762
807 5420 001370      BNE ST104B
808 5422 005077      CLR #IDNE15
      000754
809 5420 000137      JMP #ST104A
      005360
810 5432 000205 ST104C: RTS R5

```

```

811          ;*****
812          ;*****

```



```

813          JST105, TEST FOR API BREAKS TO CORRECT PORTS FROM 0 TO 177 FROM
814          JUM11 APIL3.
815 5434 005003 ST1051 CLR R3
816 5436 110317 ST105A1 MOV R3,0#APIL3B          ISET APIL3 WITH PORT ADDRESS
          167763
817 5442 012777      MOV #=1,0IDNE11
          177777
          000704
818 5456 020327      CMP R3,#177
          000177
819 5454 001415      BEQ ST105C
820 5456 005203      INC R3
821 5460 022777 ST105B: CMP #=1,0ERRFLG
          177777
          000704
822 5466 001410      BEQ ST105C
823 5470 022777      CMP #=1,0IDNE15
          177777
          000704
824 5476 001370      BNE ST105B
825 5500 005077      CLR 0IDNE15
          000676
826 5504 000137      JMP #ST105A
          005436
827 5510 000205 ST105C: RTS R5
828          ;
829          ;
830          ;
831 5512 012777 ETST: MOV #=1,0ERRIND
          177777
          000670
832 5520 012777      MOV #=1,0ERRFLG
          177777
          000644
833 5526 000205      RTS R5
834          ;
835 5530 012777 DTST: MOV #=1,0IDNE11
          177777
          000646
836 5536 022777 DTST.1: CMP #=1,0IDNE15
          177777
          000636
837 5544 001374      BNE DTST.1
838 5546 005077      CLR 0IDNE15
          000630
839 5552 000205      RTS R5
840          ;SUBROUTINE WRITES POP=15 ADDRESSES CONTAINED IN R2 INTO 2K OF C
841          ;MEM STARTING AT ADDRESS INDICATED IN R1, A RANDOM FASHION DELAY
842          ;EXECUTED BEFORE EACH WORD IS WRITTEN.
843 5554 012737 WRT2K: MOV #4000,0#WCNT
          004000
          006420
844 5562 004537 WRT2.1: JSR R5,0NDLY
          006276
845 5566 010221      MOV R2,(R1)+          IWRITE A LOC

846 5570 005202      INC R2          ;BUMP 15 ADDRESS
847 5572 005337      DEC 0#WCNT

```

```

006420
848 5570 001371      BNE RRT2.1          ;DONE?
849 5000 000205      RTS R5              ;YES
850
851      ;
852      ;SUBROUTINE READS & CHECKS DATA (PDP-15 ADDRESSES) WRITTEN BY 15
853      ;IF AN ERROR IS ENCOUNTERED, PDP-15 LOC ERRFLG IS SET TO -1, THE
854      ;GOOD WORD AND PDP-15 ADDRESS (SAME) IS DEPOSITED IN PDP-15 LOC
855      ;AND THE BAD WORD IS DEPOSITED IN PDP-15 LOC BAD. TESTING BEGINS
856      ;ADDRESS INDICATED IN R1. DELAY BEFORE EACH WORD IS CHECKED.
856 5002 012737      READ2K: MOV #4000,0#RCNT
      004000
      006422
857 5010 004537      READ.1: JSR R5,0#DLY
      006270
858 5014 010277      MOV R2,0#ADR
      000610
859 5020 010277      MOV R2,0#GOOD      ;STORE DATA EXPECTED
      000550
860 5024 012177      MOV (R1)+,0#BAD    ;STORE DATA READ
      000546
861 5030 027777      CMP 0#GOOD,0#BAD   ;DATA CORRECT?
      000540
862 5036 001005      BNE READ.2          ;DATA CORRECT?
863 5040 003202      INC R2              ;YES
864 5042 005337      DEC 0#RCNT
      006422
865 5040 001300      BNE READ.1          ;DONE?
866 5050 000205      RTS R5              ;YES
867 5052 012777      READ.2: MOV #-1,0#ERRIND
      177777
      000530
868 5060 022777      CMP #-1,0#ERRFLG
      177777
      000504
869 5060 001400      BEQ READ.3          ;ERRFLG ALREADY SET?
870 5070 012777      MOV #-1,0#ERRFLG
      177777
      000474
871 5070 000205      HEAD.3: RTS R5
872
873      ;
874      ;SUBROUTINE TO CONVERT PDP-15 ADDRESS INTO PDP-11 ADDRESS AND LE
875      ;11 ADDRESS IN R1. 15 ADDRESS MUST BE IN R1 UPON ENTERING ROUTIN
875 5700 000301      CNVRT: ASL R1      ;15 ADDR X 2
876 5702 000240      NOP
877 5704 000240      NOP
878 5706 003701      ADD 0#DS13,R1      ;15 ADDR X 2 + 20000 + DS BIT 13
      000364
879 5712 000205      RTS R5
880      ;SUBROUTINE TO INIT VECTOR ADDRESSES FROM 0 TO 774. VECTOR ADDRE
881      ;WILL CONTAIN ,+2, AND VECTOR ADDRESSES+2 WILL CONTAIN HALTS.
882 5714 012737      IVA:  MOV #-200,0#THPCNT
      177000
      000300
883 5722 012700      MOV #2,R0
      000002
884 5720 005001      CLR R1

```

```

005 5730 010021 IVA,A:  MOV R0,(R1)+
000 5732 012721      MOV #HALT,(R1)+
000000
007 5730 062700      ADD #4,R0
000004
008 5742 005237      INC #HTMPCNT
006300
009 5746 001370      BNE IVA,A
050 5750 000200      RTS R5
001
092
093
094
095 5752 004537 PWRFLI JSR R5,#IVA
005714
096 5756 012737      MOV #PWRFLG,#04
006244
000004
097 5764 012737      MOV #340,#06
000340
000000
098 5772 005037      CLR #TMP
006356
099 5776 012737      MOV #PWRFB,#024
006032
000024
900 6004 012737      MOV #340,#026
000340
000026
901 6012 012777      MOV #-1,#IDNE11
177777
000304
902 6020 027777 PWRFLA1 CMP #IDNE11,#IDNE15  IREF COMMON MEM WHILE WAITING
000300
000354
903 6026 000137      JMP #PWRFA,A
000020
904 6032 012737 PWRFLB1 MOV #PWRFC,#024
000102
000024
905 6040 010037      MOV R0,#TR0
000340
906 6044 010137      MOV R1,#TR1
000342
907 6050 010237      MOV R2,#TR2
000344
908 6054 010337      MOV R3,#TR3
000340
909 6060 010437      MOV R4,#TR4
000350
910 6064 010537      MOV R5,#TR5
000352
911 6070 010637      MOV SP,#TSP
000354
912 6074 000000      HALT

913 6076 000137      JMP #PWRFB,B
000032

```

```

914 6102 005037 PWR,F,C: CLR #NTPCNT
      006300
915 6100 012737      MOV #3,#NTPCNT1
      000003
      006302
916 6114 005237 PWR,F,D: INC #NTPCNT
      006300
917 6120 001375      BNE PWR,F,D
918 6122 005337      DEC #NTPCNT1
      006302
919 6120 001372      BNE PWR,F,D
920 6130 022777 PWR,F,E: CMP #-1,#IDNE15
      177777
      000244
921 6130 001404      BEQ PWR,F,F
922 6140 005237      INC #NTPCNT
      006300
923 6144 001371      BNE PWR,F,E
924 6140 000000      HALT
925
926 6150 005077 PWR,F,I: CLR #IDNE15
      000220
927 6154 005077      CLR #GOOD
      000214
928 6160 032737      BLT #1,#NTP
      000001
      006350
929 6160 001403      BEQ PWR,I
930 6170 012777      MOV #-1,#GOOD
      177777
      000170
931 6170 012777 PWR,I:  MOV #-1,#IDNE11
      177777
      000200
932 6204 013700      MOV #TR0,R0
      006340
933 6210 013701      MOV #TR1,R1
      000342
934 6214 013702      MOV #TR2,R2
      006344
935 6220 013703      MOV #TR3,R3
      006340
936 6224 013704      MOV #TR4,R4
      006350
937 6230 013705      MOV #TR5,R5
      006352
938 6234 013706      MOV #TSP,SP
      006354
939 6240 022626      CMP (SP)+,(SP)+
940 6242 000205      RTS R5
941 6244 012737 PWR,F,G:  MOV #6,#N4
      000000
      000004
      000006
942 6252 005037      CLR #N6
      000006

943 6250 005237      INC #NTP
      006356

```

JPDP-15 FAILED TO SET -1 IN IDNE
FOR PDP-11 REFERENCED WRONG LOCA

```

944 0262 000777      BR      ;WAIT FOR POWER FAIL INT AFTER B
945                    ;TIME OUT.....
946                    ;*****
947                    ;*****
948                    ;CONVERT 15 ADDRESS TO 11 & WRITE 2K.
949 0264 004537      CONWRT: JSR R5,#CNVRT
                        005700
950 0270 004537      JSR R5,#WRT2K
                        005554
951 0274 000205      RTS R5
952                    ;
953 0270 013704      DLY:   MOV #DLYPNT,R4
                        006424
954 0302 012437      MOV (R4)+,#DLYCNT
                        006420
955 0306 052737      BIS #177700,#DLYCNT
                        177700
                        006420
956 0314 005237      DLY,A: INC #DLYCNT
                        006420
957 0320 001370      BNE DLY,A
958 0322 020437      CMP R4,#NSIZE
                        006050
959 0320 001001      BNE DLY,B
960 0330 005004      CLR R4
961 0332 010437      DLY,B: MOV R4,#DLYPNT
                        006424
962 0336 000205      RTS R5
963                    ;CONSTANTS & VARIABLES
964 0340 000000      TR0:   0
965 0342 000000      TR1:   0
966 0344 000000      TR2:   0
967 0346 000000      TR3:   0
968 0350 000000      TR4:   0
969 0352 000000      TR5:   0
970 0354 000000      TSP:   0
971 0356 000000      TMP:   0
972 0360 000000      TPCNT:  0
973 0362 000000      TPCN1:  0
974 0364 000000      DS13:   0
975 0366 000000      KWSA15:  0
976 0370 000000      KWSA11:  0
977 0372 000000      ERKFLG:  0
978 0374 000000      BUDD:   0
979 0376 000000      BAU:    0
980 0400 000000      TEST:   0
981 0402 000000      IDNE15:  0
982 0404 000000      IDNE11:  0
983 0406 000000      TSTPNT:  0
984 0410 000000      ERKIND:  0
985 0412 000000      BR5TV:   0
986 0414 000000      BR7TV:   0
987 0416 000000      INIT:   0
988 0420 000000      WCNT:   0
989 0422 000000      KCNT:   0

990 0424 000000      DLYPNT:  0
991 0426 000000      DLYCNT:  0

```

992	0430	000000	ADMI	0
993	0432	000000	TEMP:	0
994				
995	0434	006434	TSTBL:	.
996	0436	001434		TST1
997	0440	001470		TST2
998	0442	001540		TST3
999	0444	001562		TST4
1000	440	001604		TST5
1001	450	001650		TST6
1002	452	001714		TST7
1003	454	001750		TST10
1004	450	002020		TST11
1005	460	002060		TST12
1006	462	002120		TST13
1007	464	002300		TST14
1008	460	002500		TST15
1009	470	002522		TST16
1010	472	002552		TST17
1011	474	002602		TST20
1012	470	002604		TST21
1013	500	002606		TST22
1014	502	002624		TST23
1015	504	002642		TST24
1016	506	002740		TST25
1017	510	002770		TST26
1018	512	003012		TST27
1019	514	003030		TST30
1020	516	003062		TST31
1021	520	003104		TST32
1022	522	003120		TST33
1023	524	003134		TST34
1024	520	003150		TST35
1025	530	003164		TST36
1026	532	003210		TST37
1027	534	003234		TST40
1028	530	003260		TST41
1029	540	003304		TST42
1030	542	003334		TST43
1031	544	003364		TST44
1032	546	003414		TST45
1033	550	003444		TST46
1034	552	003446		TST47
1035	554	003450		TST50
1036	556	003452		TST51
1037	560	003460		TST52
1038	562	003532		TST53
1039	564	003540		TST54
1040	560	003612		TST55
1041	570	003620		TST56
1042	572	003672		TST57
1043	574	003700		TST60
1044	576	003752		TST61
1045	000	004002		TST62
1046	002	004032		TST63

1047	004	004002		TST64
1048	006	004112		TST65

```

1049 010 004144 TST06
1050 012 004204 TST07
1051 014 004302 TST70
1052 016 004412 TST71
1053 020 004514 TST72
1054 022 004622 TST73
1055 024 004702 TST74
1056 026 005126 TST75
1057 030 005142 TST76
1058 032 005156 TST77
1059 034 005172 ST100
1060 036 005206 ST101
1061 040 005222 ST102
1062 042 005300 ST103
1063 044 005350 ST104
1064 046 005434 ST105
1065
1066 050 006650 SIZE:
1067
1068 000001 .END
    
```

```

AUK 006430 APIL0B= 167772 APIL1B= 167773
APIL2B= 167762 APIL3B= 167763 BAD 006376
BRSTV 006412 BR7TV 006414 CNVRT 005700
CUNWR1 006204 DLY 006276 DLYCNT 006426
DLYPN1 006424 DLY,A 006314 DLY,B 006332
UKS1 = 167774 DRS2 = 167774 DRS3 = 167760
UKS4 = 167764 DS = 177570 DS13 006364
JTST 005530 DTST.1 005536 ERRFLG 006372
ERRIND 006410 ETST 005512 GC 001140
GUUU 006374 IDNE11 006404 IDNE15 006402
IN1T 006410 ISTART 001100 ISTA.1 001272
ISTA.2 001300 ISTA.3 001322 IS,A 001150
IVA 005714 IVA,A 005730 PC =%000007
PSW = 177776 PWRFL 005752 PWRF,A 006020
PWRF,b 006032 PWRF,C 006102 PWRF,D 006114
PWRF,E 006130 PWRF,F 006150 PWRF,G 006244
PKK.1 006170 KCNT 006422 READ.1 005610
READ.2 005652 READ.3 005676 READ2K 005602
KWSA11 006370 KWSA15 006366 R0 =%000000
R1 =%000001 R2 =%000002 R3 =%000003
R4 =%000004 R5 =%000005 SIZE 006650
SP =%000006 STANT 001340 STAR.1 001346
STAN.2 001422 ST100 005172 ST101 005206
ST102 005222 ST102A 005224 ST102B 005246
ST102C 005276 ST103 005300 ST103A 005302
ST103B 005324 ST103C 005354 ST104 005356
ST104A 005360 ST104B 005402 ST104C 005432
ST105 005434 ST105A 005436 ST105B 005460
ST105C 005510 TEMP 006432 TEST 006400
TMP 006356 THPCNT 006360 THPCN1 006362
TK0 006340 TR1 006342 TR2 006344
TK3 006346 TR4 006350 TR5 006352
TSP 006354 TSTPNT 006406 TSTTBL 006434
T0T1 001434 TST1,A 001446 TST10 001756
TST10A 001770 TST11 002020 TST12 002060
TST13 002120 TST13A 002136 TST13B 002202
TST13C 002232 TST13D 002276 TST14 002306
TST14A 002324 TST14B 002370 TST14C 002420
TST14D 002470 TST15 002500 TST15A 002512
TST16 002522 TST16A 002542 TST17 002552
TST17A 002572 TST2 001476 TST2,A 001510
TST20 002602 TST21 002604 TST22 002606
TST23 002624 TST24 002642 TST24A 002646
TST24B 002706 TST24C 002710 TST24D 002736
TST25 002746 TST25A 002760 TST26 002770
TST26A 003002 TST27 003012 TST27A 003026
TST3 001540 TST30 003036 TST30A 003052
TST31 003062 TST31A 003074 TST32 003104
TST33 003120 TST34 003134 TST35 003150
TST36 003164 TST37 003210 TST4 001562
TST40 003234 TST41 003260 TST42 003304
TST43 003334 TST44 003364 TST45 003414
TST46 003444 TST47 003446 TST5 001604
TST0 003450 TST51 003452 TST52 003466
TST02A 003522 TST03 003532 TST54 003546
    
```

TST54A 003002
TST56A 003602

TST55 003012
TST57 003672

TST50 003620
TST6 001650

JNICHANNEL 15 MAINDEC-15-DAUCA MACRO-11 VR05A16 PAGE 1+
SYMBOL TABLE

TST00	003700	TST60A	003742	TST61	003752
TST01A	003772	TST62	004002	TST62A	004022
TST03	004032	TST63A	004052	TST64	004062
TST04A	004102	TST65	004112	TST65A	004134
TST06	004144	TST66A	004174	TST67	004204
TST07A	004240	TST67B	004260	TST67C	004266
TST7	0041714	TST7.A	001726	TST7M	004302
TST70A	004342	TST70B	004362	TST70C	004370
TST71	004412	TST71A	004444	TST71B	004464
TST71C	004500	TST72	004514	TST72A	004532
TST720	004572	TST72C	004614	TST73	004622
TST73A	004060	TST73B	004704	TST73C	004712
TST73D	004710	TST73E	004746	TST73F	004754
TST74	004762	TST74A	005024	TST74B	005050
TST74C	005056	TST74D	005062	TST74E	005112
TST74F	005120	TST75	005120	TST76	005142
TST77	005156	MCNT	006420	WRT2K	005554
4RT2.1	005502				