PDP11

REMOTE DIAGNOSTIC EXERCISER

MD-11-DZKCI-A

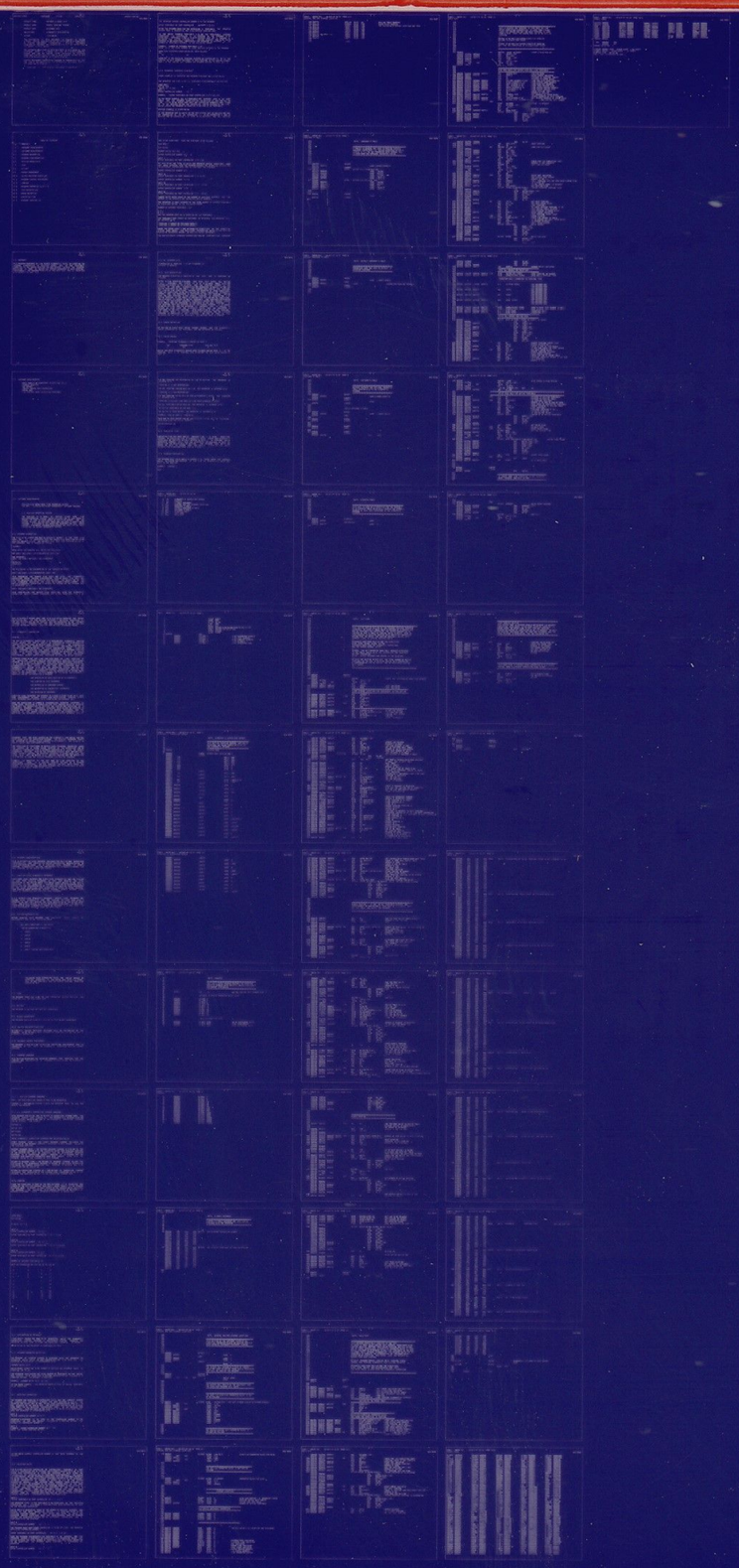
EP-DZKCI-A-DL-A

FICHE 1 OF 1

DEC 1977

digital

MADE IN USA



PRODUCT CODE MAINDEC-11-DZKCI-A-D

PRODUCT NAME REMOTE TERMINAL TESTER

PRODUCT DATE DECEMBER 1977

MAINTAINER DIAGNOSTIC ENGINEERING

AUTHOR CLEM WALSH

SEQ 0002

TABLE OF CONTENTS

----- -- --------

SEQ 0003

1 0 ABSTRACT

THIS PROGRAM PERFORMS ON LINE REMOTE DIAGNOSTIC TESTS, BY STARTING THE INTERNAL DIAGNOSTICS IN SELECTED RT801, 803, OR 805 TERMINALS, CONNECTED TO THE SERIAL BUS FOR THE DPM SYSTEM WITHIN THE PROGRAM, PROVISION IS MADE TO REPORT ALL ERRORS GENERATED BY EACH TERMINAL STRUCTURALLY, THE PROGRAM CONSISTS OF ONE MAIN TEST, AND IS ORGANIZED WITH RESPECT TO THAT TEST

SEQ 0004

2 0 HARDWARE REQUIREMENTS

PDP11 FAMILY OF COMPUTERS, EXCEPT THE LSI-11
CONSOLE TELETYPE
16K MEMORY
ISB11A (SERIAL BUS CONTROLLER)
SERIAL BUS
1-63 FACTORY DATA COLLECTION TERMINALS

3 0 SOFTWARE REQUIREMENTS

RSX-11M DISK BASED REAL TIME OPERATING SYSTEM
THE DPM - "DISTRIBUTED PLANT MANAGEMENT" SOFTWARE PACKAGE

3 1 RSX-11M OPERATING SYSTEM

THE RSX-11M IS A SMALL TO MEDIUM SIZED REAL TIME MULTIPROGRAMMING SYSTEM WHICH UTILIZES 16K TO 124K WORDS OF MEMORY, OR 128K TO 1024K WORDS OF PDP-11/70 MEMORY. THE 16K SYSTEM ALLOWS 8K FOR USER TASKS AND INCLUDES A SUBSET MACRO PACKAGE A MINIMUM OF 24K WORDS ARE REQUIRED FOR FULL MACRO SUPPORT, CONCURRENT PROGRAM DEVELOPMENT AND APPLICATION TASKS EXECUTION, OR MEMORY MANAGEMENT SUPPORT

4 0 PROGRAM GENERATION

THE FILE TO BE LOADED AND RUN IN SYSTEM'S MEMORY IS THE TASK FILE DZKC1 TSK IT IS GENERATED FROM THE SOURCE FILE DZKC1 MAC BY USING RSX-11M INDIRECT FILES AN INDIRECT FILE DZKC1 CMD CONTAINS A LIST OF MCR COMMANDS, AND IS INVOKED BY TYPING

"@LDZKC1"

UPON WHICH THE CONSOLE WILL PRINT THE FOLLOWING

```
MAC DZKC1.OBJ,DZKC1 LST=DIAGSUPER/ML,DZKC1 MAC

TKB @TKBDZKC1
DZKC1.TSK,DZKC1 MAP=DZKC1 OBJ,DIAGSUPER
/
UNITS=75
STACK=512
//
```

THE FOLLOWING IS AN EXPLANATION OF THE CONSOLE PRINTOUT

DZKC1 OBJ,DZKC1 LST=DIAGSUPER/ML,DZKC1 MAC

THIS ASSEMBLES THE SOURCE FILE DZKC1 MAC WITH THE DIAGNOSTIC SUPERVISOR MACRO PACKAGE DIAGSUPER/ML, INTO AN OBJECT FILE DZKC1 OBJ FOR TASK BUILDING BY THE RSX11-M TASK BUILDER THE SWITCH /ML ON THE FILE DIAGSUPER SPECIFIES THE FILE AS A USER'S MACRO LIBRARY THE PDP-11 DIAGNOSTIC SUPERVISOR IS NEEDED FOR ASSEMBLING THE PROGRAM, AND IS DESCRIBED IN 4.1

DZKC1.TSK,DZKC1.MAP=DZKC1.OBJ,DIAGSUPER

THIS TASK-BUILDS THE OBJECT FILE DZKC1 OBJ WITH THE DIAGNOSTIC SUPERVISOR OBJECT LIBRARY DIAGSUPER, INTO A TASK IMAGE DZKC1 TSK TO BE

RUN IN SYSTEM'S MEMORY DZKC1 MAP IS A FILE WHICH CONTAINS THE MEMORY ALLOCATION MAP THE TASK BUILDER SWITCHES /, UNITS=75, STACK=512, //, ARE NOT PRINTED OUT ON THE CONSOLE, BUT ARE CONTAINED IN THE INDIRECT COMMAND FILE "TKBDZKC1 CMD" AND ARE DESCRIBED IN THE RSX-11M TASK BUILDER REFERENCE MANUAL RSX-11M TASK BUILDER REFERENCE MANUAL.

4 1 DIAGNOSTIC SUPERVISOR

GENERAL--------

THE PDP-11 DIAGNOSTIC SUPERVISOR IS A MODULARIZED DIAGNOSTIC MONITOR WHICH SERVES THE PDP-11 FAMILY OF COMPUTERS BY PROVIDING RUN-TIME SUPPORT FOR FUNCTIONAL OR REPAIR LEVEL DIAGNOSTICS. THE DIAGNOSTICS ARE DESIGNED FOR SUPERVISOR COMPATIBILITY, AND CREATED FOR DESIGN ENGINEERING, MANUFACTURING, OR FIELD ENGINEERING USAGE FUNCTION LEVEL PROGRAMS PROVIDE FOR ERROR DIAGNOSIS AT A HARDWARE FUNCTION LEVEL, WHILE REPAIR LEVEL DIAGNOSTICS PROVIDE FOR REPAIR AT A MODULE LEVEL, AS SUCH, THEY EXERCISE, DIAGNOSE, OR TEST INDIVIDUAL PERIPHERALS, MEMORY SUBSYSTEMS, PROCESSOR OPTIONS, OR ENTIRE SYSTEMS

SEPARATE STAND ALONE (OFF LINE) AND USER MODE (ON LINE) VERSIONS OF THE SUPERVISOR COMMONLY PROVIDE NON-DIAGNOSTIC SERVICES FOR A SINGLE PROGRAM, OR A SCRIPT OF PROGRAMS, THAT HAVE BEEN PREVIOUSLY ASSEMBLED IN A STAND ALONE ENVIRONMENT THE SUPERVISOR SHARES RESIDENCY WITH THE DIAGNOSTIC(S) ONLY, WHILE IN USER MODE THE SUPERVISOR AND THE DIAGNOSTIC(S) RESIDE WITH BOTH AN OPERATING SYSTEM AND THE USER PROGRAM(S) HOWEVER, IN EITHER ENVIRONMENT, SUPERVISOR SERVICES ARE ELICITED BY A RESIDENT DIAGNOSTIC VIA THE GENERATION OF UNIQUE MACRO CALLS TO THE SUPERVISOR IN THIS MANNER THE FOLLOWING NON-DIAGNOSTIC SERVICES CAN BE INITIATED FOR A PROGRAM

THE EXECUTION OF EACH SECTION OF A DIAGNOSTIC

THE LOOPING OF TEST PROGRAMS

THE REPORTING OF HARDWARE ERRORS

THE REPORTING OF UNEXPECTED INTERRUPTS

THE PRINTING OF MESSAGES.

FUNCTION LEVEL PROGRAMS CAN OPERATE IN EITHER A STAND ALONE OR A USER MODE ENVIRONMENT WITHOUT SOURCE CODE MODIFICATION HOWEVER, REPAIR LEVEL PROGRAMS MUST OPERATE IN A STAND ALONE ENVIRONMENT ONLY

ONCE THE SUPERVISOR IS LOADED A STANDARDIZED OPERATOR INTERFACE IS ESTABLISHED, PROVIDING A COMMUNICATIONS PATH THROUGH WHICH AN OPERATOR CAN DIRECT THE SUPERVISOR TO INITIATE THE LOADING AND EXECUTION OF A DIAGNOSTIC PROGRAM INTERFACE DIALOGUE ALSO ALLOWS AN OPERATOR TO EXAMINE AND/OR MODIFY THE CONTENT OF THE GENERAL REGISTERS, AND THE CONTENTS OF THE MEMORY LOCATIONS UTILIZED BY THE DIAGNOSTIC(S) IN ADDITION, THE INTERFACE ALLOWS AN OPERATOR TO ACCESS SEPARATE UTILITY

PROGRAMS THAT ARE ALSO LOCATED ON THE DIAGNOSTIC LOAD MEDIA, AND ARE ASSOCIATED WITH THE SUPERVISOR PACKAGE THE UTILITY PROGRAMS PERMIT AN OPERATOR TO INSTALL FIELD CHANGES TO A DIAGNOSTIC, AND TO BUILD OR MODIFY SCRIPT AND CONFIGURATION FILES.

THE SERVICING OF FUNCTIONAL OR REPAIR LEVEL PROGRAMS OPERATING UNDER THE CONTROL OF A STAND ALONE VERSION OF THE SUPERVISOR, CAN BE INDIRECTLY CONTROLLED FROM A SEPARATE COMPUTER BY A LOCAL OR REMOTE DIAGNOSTIC MONITOR, SUCH AS THE AUTOMATED PRODUCT TEST (APT/APT-RD) UNDER INDIRECT CONTROL, AN OPERATOR INTERFACE IS ESTABLISHED WITH THE LOCAL (E G APT) OR REMOTE (E G APT-RD) MONITOR AND THE SUPERVISOR IS DIRECTED TO LOAD AND EXECUTE PROGRAMS VIA MONITOR COMMANDS

FINALLY, IN ADDITION TO THE OFF AND ON LINE VERSIONS OF THE SUPERVISOR, THERE IS A SPECIAL VERSION DEFINED AS A PROGRAM DEVELOPMENT SYSTEM(PDS) THE PDS VERSION INCLUDES BOTH DEBUG AND UPDATE UTILITIES AS CORE RESIDENT FEATURES AND IS ESSENTIALLY A SUPER SUBSET OF THE BASIC STAND ALONE VERSION

5 0 PROGRAM CONSIDERATIONS

THIS IS A FUNCTION LEVEL PROGRAM, DESIGNED FOR RSX-11M AND SUPERVISOR COMPATIBILITY, AND CREATED FOR DESIGN ENGINEERING, MANUFACTURING, OR FIELD ENGINEERING USAGE AS A FUNCTION LEVEL PROGRAM, IT STARTS TERMINAL DIAGNOSTICS ON LINE, AND REPORTS ALL DIAGNOSTIC ERRORS, FOR EACH TERMINAL ON THE SERIAL BUS SYSTEM

5 1 FUNCTION LEVEL DIAGNOSTIC PROGRAMS

FUNCTIONAL LEVEL PROGRAMS ARE FIRST LEVEL DIAGNOSTICS THAT ARE SIMILAR TO USER MODE LEVEL-2 PROGRAMS IN THAT THEY REQUIRE THE TOTAL DEDICATION OF THE DEVICE UNDER TEST WHILE DIRECT ACCESS TO THE DEVICE REGISTERS IS NOT PERMITTED HOWEVER, DATA CAN BE EXCHANGED WITH A DEVICE VIA THE IMPLEMENTATION OF I/O ROUTINES SUCH AS THOSE ENGENDERED BY THE Q10$ DIRECTIVES USED IN RSX-11M IT SHOULD BE NOTED, HOWEVER, THAT RUNNING UNDER OPERATING SYSTEM CONTROL, STABLE PROGRAM LOOPS CANNOT BE GAURANTEED FOR FUNCTION LEVEL PROGRAMS

5 1 1 REPAIR LEVEL DIAGNOSTIC PROGRAMS - REPAIR LEVEL PROGRAMS ARE SECOND LEVEL DIAGNOSTICS THAT ARE SIMILAR TO USER MODE LEVEL-3 PROGRAMS IN THAT THEY REQUIRE THE TOTAL DEDICATION OF THE DEVICE UNDER TEST, AND DIRECT ACCESS TO DEVICE REGISTERS IS ALLOWED IN ADDITION, SINCE REPAIR LEVEL DIAGNOSTICS CANNOT BE RESIGNED TO A USER MODE (ON LINE) ENVIRONMENT, PROGRAM LOOPS ARE APPLICABLE

6 0 TESTING PREREQUISITES

BEFORE RUNNING THIS PROGRAM, THE FOLLOWING TESTS SHOULD BE SUCCESSFULLY RUN IN THE FOLLOWING ORDER

1 ALL APPLICABLE PDP-11 CPU TESTS

2 ISB11A STANDALONE DIAGNOSTICS

- 1 DZKCC
- 2 DZKCD
- 3 DZKCA
- 4 DZKCE
- 5 DZKCF
- 6 DZKCI (SERIAL BUS EXERCISER)

7 FOLLOWING THESE STANDALONE TESTS, THE ISB11A SHOULD BE REMAINED CONNECTED TO THE SERIAL BUS, WITH 1-63 RT801, 803, OR 805 TERMINALS, AND THE REMOTE TERMINAL EXERCISOR SHOULD BE RUN

7 0 XXDP

THE PROGRAM RUNS ON LINE TO THE OPERATING SYSTEM RSX-11M, AND THEREFORE WILL NOT RUN LNDER XXDP

8 0 ACT/APT

THE PROGRAM IS NEITHER ACT NOR APT COMPATIBLE

9 0 MEMORY MANAGEMENT

THE PROGRAM DOES NOT DIRECTLY UTILIZE OR TEST MEMORY MANAGEMENT

10 0 SWITCH REGISTER FUNCTIONS

NO EXPLICIT SWITCH REGISTER SETTINGS WILL BE RECOGNIZED BY THE PROGRAM, SO AS TO FACILITATE INTEGRAT ON UNDER RSX-11M AND THE PDP-11 DIAGNOSTIC SUPERVISOR

11 0 PROGRAM CONTROL PROCEDURES

THE PROGRAM IS RUN ON LINE IN RSX-11M OPERATING ENVIRONMENT AND IS CONTROLLED BY RSX (MCR) OR DIAGNOSTIC SUPERVISOR MONITOR COMMAND LANGUAGE

11 1 COMMAND LANGUAGE

THIS SECTION DESCRIBES THE OPERATOR COMMANDS, THAT PROVIDES FOR THE LOADING, AND MANIPULATION OF PROGRAMS BY RSX-11M AND THE DIAGNOSTIC SUPERVISOR

11 1 1 RSX-11M COMMAND LANGUAGE -

RUN - THE RUN DIRECTIVE CAUSES A TASK TO BE REQUESTED.

CONTROL C - ACTIVATING CONTROL C GETS THE OPERATOR BACK TO MCR AND ABORTS THE PROGRAM

11 1 1 1 DIAGNOSTIC SUPERVISOR COMMAND LANGUAGE -

UPON PROGRAM EXECUTION, AND TO RETURN TO SUPERVISOR COMMAND MODE, THE OPERATOR SHOULD TYPE "CONTROL A" THE PROGRAM WILL RESPOND WITH THE PROMPT DCP-A>, WHEREIN ANY ONE OF THE DIAGNOSTIC SUPERVISOR COMMANDS CAN BE ENTERED. FOR EXAMPLE

CONTROL A

DCP-A> STA

DCP-A>RES

DCP-A>CON

THESE DIAGNOSTIC SUPERVISOR COMMANDS ARE DESCRIBED BELOW

START PROGRAM (STA) - THE START PROGRAM COMMAND INITIATES THE EXECUTION OF THE PROGRAM CURRENTLY CONTAINED IN MAIN MEMORY, INCLUDING THE DIALOGUE PORTIONS

RESTART PROGRAM (RES) - THE RESTART PROGRAM COMMAND IS SIMILAR TO THE START PROGRAM COMMAND WITH THE EXCEPTION BEING THAT DIALOGUE PORTIONS OF THE PROGRAM CAN ONLY BE EXECUTED VIA AN APPROPRIATE OPERATOR RESPONSE TO A SUPERVISOR QUERY. HOWEVER IT IS ASSUMED THAT THE REQUIRED CONFIGURATION PARAMETERS HAVE BEEN LOADED, ALONG WITH THE PROGRAM, PRIOR TO THE ISSUANCE OF THIS COMMAND.

RETURN TO PROGRAM (CON) - THE RETURN TO PROGRAM COMMAND ALLOWS THE EXECUTION OF THE DIAGNOSTIC PROGRAM TO RESUME AT THE FIRST INSTRUCTION FOLLOWING THE CURRENT SUPERVISOR CALL. HOWEVER, IF DESIRED, NEW FLAG CONDITIONS MAY BE SPECIFIED.

RETURN TO SUPERVISOR (CONTROL A) - THE RETURN TO SUPERVISOR COMMAND INITIATES THE EXECUTION OF THE CLEANUP CODE, CONTAINED IN THE ACTIVE PROGRAM, AND PROVIDES AN EXIT TO THE SUPERVISOR COMMAND MODE

12.0 LOADING

SINCE THE EXERCISER IS PART OF THE DPM SOFTWARE, IT IS REQUESTED AND LOADED BY THE RSX-11M COMMAND "RUN DZKC1" WHERE "DZKC1" IS THE PROGRAM RELEASED NAME. A PROMPT IS THEN ISSUED TO THE CONSOLE WHICH INITIATED THE PROGRAM. THE FOLLOWING IS A CONSOLE PRINTOUT OF AN EXAMPLE OF PROGRAM DIAGLOGUE (NOTE OPERATOR'S RESPONSE IS UNDERLINED)

SEQ 0011

```
>RUN DZKC1
-----------
DCP-A>STA
      -----

# UNITS (D) ? 8
            --

UNIT 1
ENTER CONTROLLER NUMBER : (O) 0 ?
                           --
ENTER TERMINALS ON THAT CONTROLLER : (D) 1 ? 1-4
                                      ----

UNIT 5
ENTER CONTROLLER NUMBER . (O) 0 ? 1
                          ----
ENTER TERMINALS ON THAT CONTROLLER · (D) 4 ? 1,2,3,4
                                     ---------

UNIT 9
ENTER CONTROLLER NUMBER  (O) 2 ?
                         ---
ENTER TERMINALS ON THAT CONTROLLER· (D) 8 ? 1,2,3-4
                                    ---------

NUMBER OF ENTERED TERMINALS= 12

UNIT ** CONTROLLER ** LOGICAL SB ** LUN **
  1              0             1        9
  2              0             2       10
  3              0             3       11
  4              0             4       12
  1              1             6       13
  2              1             7       14
  3              1             8       15
  4              1             9       16
  1              2            11       17
  2              2            12       18
  3              2            13       19
  4              2            14       20
```

12 1 EXPLANATION OF PRINTOUT

>"RUN DZKC1" CAUSES THE TASK TO BE REQUESTED, WHILE THE DIAGNOSTIC SUPERVISOR CONTROL PROGRAM -- REVISION-A (DCP-A), REQUESTS THE OPERATOR TO TYPE "STA" TO START THE EXERCISER EXAMPLE: DCP-A>STA

DESCRIPTION OF THE PRINTOUT IS CONTINUED IN 13 0

13 0 PROGRAM PARAMETER SELECTION

THE PROGRAM AT STARTUP ENTERS A DIALOGUE WITH THE OPERATOR TO DETERMINE WHICH UNITS ON WHAT CONTROLLERS ARE TO BE EXERCISED. THE OPERATOR AT THIS POINT IS INTERROGATED WITH.

"NUMBER UNITS (D)?"

WHICH MEANS, ENTER THE TOTAL NUMBER OF DEVICES THE OPERATOR WANTS TO EXERCISE (UP TO 63)

THE OPERATOR THEN ENTERS THE TOTAL NUMBER OF TERMINALS (ON THE SERIAL BUS SYSTEM) TO BE EXERCISED IN DECIMAL REPRESENTATION, FOLLOWED BY A CARRIAGE RETURN NO DEFAULT IS PROVIDED.

EXAMPLE. NUMBER UNITS (D)? 12 <CR>

IN THE ABOVE EXAMPLE , THE OPERATOR WANTS A TOTAL OF TWELVE TERMINALS TO BE EXERCISED

13 1 SELECTED CONTROLLER

THE PROGRAM CAN EXERCISE 8 CONTROLLERS (0-7), BUT THE LARGEST NUMBER OF CONTROLLERS PRESENTLY PLANNED FOR DPM SYSTEMS IS 4(0-3) THE OPERATOR CAN SELECT ANY OF ONE, OR ALL FOUR CONTROLLERS(0 THRU 3), IN ANY SEQUENCE ON THE SERIAL BUS THE EXERCISER THEN BUILDS UP TO A 63 WORD TABLE FOR ALL SELECTED CONTROLLERS (MAX OF 63 DEVICES PER CONTROLLER), PLACING EACH IN THE HIGH BYTE OF AN UP TO 63 WORD TABLE HERE THE OPERATOR IS INTERROGATED WITH.

UNIT 1
"ENTER CONTROLLER NUMBER (0) 0 ?"

OPERATOR'S RESPONSE IS TO TYPE IN THE CONTROLLER NUMBER TO BE EXERCISED, OR USE THE DEFAULT FOLLOWED BY A CARRIAGE RETURN THE DEFAULT IS CONTROLLER NUMBER 0

UNIT 1
EXAMPLE: ENTER CONTROLLER NUMBER (0)? <CR>
WHERE <CR> MEANS CARRIAGE RETURN.

SEQ 0013

IN THE ABOVE EXAMPLE CONTROLLER NUMBER 0 HAS BEEN ENTERED TO THE PROGRAM

13 2 SELECTED UNITS

FOR A GIVEN SELECTED CONTROLLER, OR CONTROLLERS, TERMINALS ON THAT SPECIFIC CONTROLLER CAN BE ENTERED INDIVIDUALLY, SEQUENTIALLY, OR IN ANY ORDER, EACH SEPERATED BY A COMMA THE OPERATOR IS ALSO GIVEN THE FLEXIBILITY OF ENTERING ALL 63 TERMINALS SIMULTANEOUSLY. THE SYSTEM IS THEN MAPPED, ONLINE TERMINALS EXERCISED, OFFLINE TERMINALS REPORTED AS BEING OFFLINE, AND NON-EXISTENT TERMINALS REPORTED AS NOT SYSGENED. (A BRIEF DESCRIPTION OF SYSGEN IS GIVEN IN 1.1 OF THE SYSTEM GENERATION MANUAL). THE PROGRAM THEN EQUATES THOSE TERMINALS TO LOGICAL DEVICES, AS SYSGENED ON THE SYSTEM, STORES THEM IN THE SAME TABLE AS THE CONTROLLERS, WITH THE TERMINAL NUMBER IN THE LO-BYTE OF THE WORD ONE TASK CAN EXERCISE 63 TERMINALS AT ONE TIME. IF A SYSTEM HAS MORE THAN 63 TERMINALS SPREAD OUT OVER MULTIPLE CONTROLLERS, MULTIPLE TASKS MUST BE RUN TO EXERCISE THOSE TERMINALS GREATER THAN 63 HOWEVER EACH TASK MUST BE INITIATED FROM DIFFERENT CONSOLE TERMINALS. INTERROGATION HERE CONTINUES WITH·

UNIT 1
"ENTER TERMINALS ON THAT CONTROLLER (D) 1 ? 1-4 "

THE OPERATOR TYPES IN THE TERMINALS TO BE EXERCISED, ON THE SELECTED CONTROLLER (0) 1-4 MEANS 4 TERMINALS HAVE BEEN ENTERED ALL AT ONCE ON THE SELECTED CONTROLLER

UNITS KEEPS A SEQUENTIAL COUNT OF THE NUMBER OF DEVICES ENTERED, AND POINTS TO THE NEXT SEQUENTIAL DEVICE IT DOES NOT POINT TO THE PHYSICAL TERMINAL NUMBER. AT THIS POINT THE TOTAL 12 UNITS THE OPERATOR WANTED EXERCISED HAVE NOT BEEN ENTERED, THUS THE PROGRAM ASSUMES THE REMAINING DEVICES APE ON DIFFERENT CONTROLLERS, AND SO CONTINUES WITH THE DIALOGUE

UNIT 5
ENTER CONTROLLER NUMBER (0)) ? 1

THE PROGRAM ASKS WHAT OTHER CONTROLLER IS TO BE UTILIZED· THE OPERATOR SPECIFIES CONTROLLER NUMBER 1.

ENTER TERMINALS ON THAT CONTROLLER (D) 4 ? 1,2,3,4

HERE THE PROGRAM INTERROGATES FOR TERMINALS TO BE ENTERED, AND THE OPERATOR ENTERS 4 TERMINALS ON CONTPOLLER 1 IN SEQUENTIAL ORDER. NOT HAVING ENTERED THE TOTAL NUMBER OF TERMINALS (12) PREVIOUSLY ASKED FOR TO BE EXERCISED, THE PROGRAM AGAIN CONTINUES TO INTERROGATE THE OPERATOR.

UNIT 9
ENTER CONTROLLER NUMBER (0) 1 ? 2

THE OPERATOR ENTERS CONTROLLER NUMBER 2 TO THE PROGRAM

ENTER TERMINALS ON THAT CONTROLLER : (D) 8 ? 1,2,3-4

AFTER THE PROGRAM ASKS FOR THE REMAINING 4 TERMINALS, THE OPERATOR ENTERS THEM IN SEQUENTIAL AND SIMULTANEOUS COMBINATIONS.

IN THE ABOVE EXAMPLE TERMINALS 1,2 ON THE PREVIOUSLY ENTERED CONTROLLER 1 E (CONTROLLER 2) IS ENTERED TO THE PROGRAM TO BE EXERCISED TERMINALS 3 AND 4 IS ALSO ENTERED, BUT AS 3 THROUGH 4, ALLOWING FOR TYPING FLEXIBILITY HAVING COUNTED THE TOTAL UNITS ENTERED TO BE EXERCISED, IMMEDIATELY THE OPERATOR IS INFORMED OF THE NUMBER OF TERMINALS ENTERED, IN THE FOLLOWING FORMAT.

EXAMPLE NUMBER OF ENTERED TERMINALS = 12
WHERE 12 IS THE TOTAL NUMBER OF LEGAL DEVICES ENTERED TO THE PROGRAM

```
**UNITS** **CONTROLLER** **LOGICAL SB** **LUN**
**U
   XX        YY              XX              %
```

WHERE XX IS THE PHYSICAL TERMINAL ENTERED ON CONTROLLER YY, ZZ IS THE LOGICAL SB UNIT NUMBER ASSIGNED TO EACH TERMINAL BY RSX DURING SYSGEN, AND % IS THE LOGICAL UNIT NUMBER THE PROGRAM ASSIGNS TO EACH UNIT XX

13 3 OPERATOR INTERFACE DIALOGUE

OTHER EXAMPLES OF OPERATOR AND PROGRAM DIALOGUE ARE LISTED BELOW

THE OPERATOR CAN TYPE IN ALL 63 TERMINALS SIMULTANEOUSLY AS FOLLOWS

```
>RUN DZKC1
DCP-A> STA
 UNITS (D) ?  63
UNIT 1
ENTER CONTROLLER NUMBER     (O) ?
```

EXAMPLE "ENTER TERMINALS ON THAT CONTROLLER (1)?"1-63 <CR>

IN THE ABOVE EXAMPLE, ALL 63 TERMINALS ARE ENTERED INTO A 63 WORD TABLE, AND THE PROGRAM EXERCISES ALL 63 DEVICES IF THEY ARE ALL ON LINE IT IS TO BE NOTED THAT ONLY ONE CONTROLLER NUMBER(0 THRU 7)AND UP TO 63 DEVICES ON ANY ONE CONTROLLER, CAN BE ENTERED AT ANY TIME, WHICH LIMITS THE PROGRAM TO EXERCISE A MAXIMUM OF 63 TERMINALS

ANOTHER EXAMPLE IS GIVEN BELOW.

IF THE OPERATOR HAS 63 UNITS ON THE DPM SYSTEM, BUT 15 DEVICES ARE ARE ON CONTROLLER 0, 15 ON CONTROLLER 1, 15 ON CONTROLLER 2, AND 18 DEVICES ARE ON CONTROLLER 3 (A TOTAL OF 63 DEVICES), AND ALL DEVICES

ARE TO BE EXERCISED THEN THE DIALOGUE IS AS FOLLOWS:

```
RUN DZKC1

DCP-A>STA

NUMBER UNITS (D)? 63

ENTER CONTROLLER NUMBER (0)?  0

UNIT 1
ENTER TERMINALS ON THAT CONTROLLER (1)? 1-15
```

AT THIS POINT THE TOTAL 63 UNITS THE OPERATOR WANTED EXERCISED, HAVE NOT BEEN ENTERED, THUS THE PROGRAM ASSUMES THE REMAINING DEVICES ARE ON DIFFERENT CONTROLLERS, AND SO CONTINUES THE DIALOGUE.

```
ENTER CONTROLLER NUMBER (0)?  1

UNIT 16
ENTER TERMINALS ON THAT CONTROLLER (1)? 16-30

ENTER CONTROLLER NUMBER (1)? 2

UNIT 31
ENTER TERMINALS ON THAT CONTROLLER (1)?  31-45

ENTER CONTROLLER NUMBER (2)?  3

UNIT 46
ENTER TERMINALS ON THAT CONTROLLER (1)?  46-63
```

(WHERE UNITS KEEPS COUNT OF THE NUMBER OF DEVICES ENTERED, AND THE NUMBER IN THE BRACKETS ARE DEFAULTED DEVICE NUMBERS)

THE OPERATOR IS THEN INFORMED OF THE TOTAL NUMBER OF ENTERED TERMINALS AS EXPLAINED BEFORE IN THE FOLLOWING FORMAT

```
NUMBER OF ENTERED TERMINALS = 63

E T C
E T C
```

AND THE PROGRAM GOES ON TO EXERCISE ALL 63 TERMINALS

FOR TERMINALS THAT CANNOT BE ASSIGNED, OR ATTACHED, THE OPERATOR WILL BE INFORMED WITH:

```
"TERMINAL X CANNOT BE ASSIGNED-$DSW=Y"
"TERMINAL X CANNOT BE ATTACHED-$DSW=Y"
```

WHERE THE ERROR CODES Y ARE RETURNED BY DIRECTIVES IN THE DIRECTIVE STATUS WORD ($DSW) FOR ADDITIONAL INFORMATION, REFER TO THE RSX-11M EXECUTIVE REFERENCE MANUAL, OR RSX-11M POCKET REFERENCE

FOR NON-EXISTENCE SYSGENED CONTROLLERS AND/OR TERMINALS, THE OPERATOR

WILL BE INFORMED WITH

"CONTROLLER X, TERMINAL Y IS NOT SYSGENED IN"
PROGRAM IS ABORTED

14 0 TEST DESCRIPTION

THE PROGRAM ESSENTIALLY CONSISTS OF ONE TEST, AND IS EXECUTED AS FOLLOWS

AFTER INITIAL OPERATOR DIALOGUE, AND UPON TEST ENTRY, ALL ACTIVE F D C TERMINALS ARE PLACED IN A UP TO 63 WORD TABLE, WITH THE CONTROLLER NUMBER IN THE HI-BYTE, AND THE TERMINAL NUMBER IN THE LO-BYTE OF THE WORD DIAGNOSTICS ARE REQUESTED TO BEGIN. IN THE 1ST ACTIVE TERMINALS ON THE TABLE THE PROGRAM THEN CHECKS FOR TERMINAL OFF LINE/ON LINE CONDITIONS, INFORMS THE OPERATOR OF THOSE CONDITIONS, LOGS THE CONDITIONS IN AN OFFLINE/ONLINE TABLE, CHECKS FOR DIAGNOSTIC ERRORS, AND THE PROCEDURE IS CONTINUED TO THE LAST TERMINAL. HAVING DONE THE LAST TERMINAL, A TWO MINUTE INTERVAL IS STARTED,THE 1'ST ACTIVE TERMINAL IN THE TABLE IS CHECKED FOR OFFLINE/ONLINE CONDITIONS, AND A REQUEST IS ISSUED TO ONLINE TERMINALS. FOR TERMINALS RESPONDING WITHIN THE TWO MINUTE INTERVAL, DIAGNOSTIC ERRORS ARE CHECKED, LOGGED IN A TABLE, AND THE PROCEDURE IS CONTINUED TO THE LAST TERMINAL. IF THE TERMINAL DOES NOT RESPOND IN THE TWO MINUTE INTERVAL, IT IS LOGGED AS BEING OFFLINE, AND THE OPERATOR IS INFORMED THAT IT FAILED TO RETURN ON LINE FROM THE DIANOSTIC TESTS, AND THE PROCEDURE IS CONTINUED TO THE LAST TERMINAL UPON COMPLETION OF THE LAST TERMINAL, THE ERROR TABLE IS CHECKED, ERRORS ARE REPORTED, AND THE 2 MINUTE INTERVAL IS CANCELLED IF NOT FINISHED AN END OF PASS IS THEN RECORDED FOR ONLY ON-LINE TERMINALS, AND ANOTHER PASS IS STARTED

15 0 ERROR REPORTING

AT THE END OF EVERY PASS, MAJOR ERRORS SENSED, ARE THE DIAGNOSTIC ERRORS GENERATED BY THE TERMINALS, AND ARE REPORTED AS FOLLOWS

15 1 MAJOR ERRORS

EXAMPLE "TERMINAL DIAGNOSTIC ERRORS ON PASS 1"

"SB"	TERMINAL-TYPE	FAILING-TEST
1	801	21

WHICH INDICATES DIAGNOSTIC ERRORS HAVE OCCURED DURING PASS 1, IN SB TERMINAL UNIT 1, OF TERMINAL-TYPE RT-801AND THE TEST WHICH FAILED IS TEST 21

FOR ANY TERMINAL NOT RESPONDING ON LINE AT ANYTIME, THE OPERATOR IS INFORMED WITH

"TERMINAL X IS NOT RESPONDING"

FOR ANY TERMINAL COMING BACK ON LINE, THE OPERATOR IS INFORMED WITH:

"TERMINAL X IS NOW RESPONDING"

FOR ANY TERMINAL GOING OFFLINE DURING DIAGNOSTIC TESTS, THE OPERATOR IS INFORMED WITH

"TERMINAL X DID NOT COME BACK ON LINE FROM DIAGNOSTIC TESTS"

FOR ALL TERMINALS GOING OFFLINE, THE OPERATOR IS INFORMED WITH

"NO ACTIVE TERMINALS ON THE BUS" !!!!

FOR AN END OF PASS REPORT, THE OPERATOR IS INFORMED WITH

EXAMPLE END OF PASS 1, TIME=8 15

THIS END OF PASS REPORT CAN BE INHIBITED BY TYPING THE THE FOLLOWING, UPON RECEIVING THE DCP-A> PROMPT

DCP-A>STA/FLA IXE

16 0 EXECUTION TIME

EXECUTION-TIME PER PASS VARIES MAXIMUM TIME = 2 MINS PASS-TIME DEPENDS ON EXECUTION TIME OF DIAGNOSTIC TESTS ERROR FREE DIAGNOSTIC TESTS WILL BE EXECUTED QUICKLY (APPROX 15 SECS) A TERMINAL WHICH GENERATES ERRORS, WILL RUN THE DIAGNOSTIC (AT MINIMUM) 90 SECS, AT WHICH TIME THE DEVICE WILL GO OFFLINE THE PROGRAM THEN WAITS AN ADDED 30 SECS TO ENSURE THAT THE DEVICE IS BACK ON LINE, WHICH ACCOUNTS FOR A MAXIMUM PASS TIME OF 2 MINS

17 0 PROGRAM TERMINATION

THE PROGRAM WILL CYCLE UNTIL A CONTROL C IS TYPED UPON THE CONSOLE WHICH INITIATED THE PROGRAM AT THAT TIME THE PROGRAM WILL ABORT AND EXIT TO THE MONITOR

EXAMPLE CONTROL C

MCR>

DZKC1 MACRO M11 10-OCT-77 10 33
TABLE OF CONTENTS

```
DZKC1   MACRO M11    10-OCT-77 10 33  PAGE 1                                                    SEQ 0019


                                        TITLE  DZKC1
                                        ENABL  AMA
                                      .NLIST   MD,ME
                                      .LIST    MEB
                                      .MCALL   SVC
                                      .MCALL   ASTX$S,Q10$,ALUN$,DIR$,GLUN$,EXIT$S
                                       MCALL   MRKT$S,CMKT$S,GTIM$S
                                       MCALL   WTSE$S
                                       GLOBL   STADR
    10 000000                     SVC
    11         000001             SVCINS= 1                       ;LIST GENERATED SYMBOLS
    12         000001             SVCTST= 1                       ;LIST TEST NUMBERS
    13         000001             SVCSUB= 1                       ;LIST SUBTEST NUMBERS
    14         000001             SVCGBL= 1                       ;LIST BLOBALS
    15         000001             SVCTAG= 1                       ;LIST GENERATED TAGS
    22
```

```
                        SBTTL  DIAGNOSTIC SUPERVISOR HEADER
                ;
                ;       ******************************************
                ;       THIS SECTION CONTAINS GENERAL INFO WHICH
                ;       DESCRIBES THE MAJOR CHARACTERISTICS OF
                ;       THE DIAGNOSTIC PROGRAM
                ;       ******************************************
000000                  POINTER NONE

000000          STADR   HEADER DZKC1,A,0,63,300,1
000000          LSNAME
000000  104             .ASCII  @D@
000001  132             .ASCII  @Z@
000002  113              ASCII  @K@
000003  103              ASCII  @C@
000004  111              ASCII  @I@
000005  000              BYTE   0
000006  000              BYTE   0
000007  000              BYTE   0
000010          LSREV
000010  101              ASCII  @A@
000011          LSDEPO
000011  060              ASCII  @0@
000012          LSMREV
000012  001              BYTE   CSREVISION
000013  004              BYTE   CSEDIT
000014          LSUNIT
000014  000000           WORD   0
000016          LSTIM1
000016  000063           WORD   63
000020          LSTIMU
000020  000300           WORD   300
000022          LSTIML
000022  000001           WORD   1
000024          LSEF
000024  000000           WORD   0
000026  000000           WORD   0
000030          LSSPC
000030  000000           WORD   0
000032          LSEXP1
000032  000000           WORD   0
000034          LSEXP2
000034  000000           WORD   0
000036          LSEXP3
000036  000000           WORD   0
000040          LSDTP
000040  005032'          WORD   LSDISPATCH
000042          LSICP
000042  005034'          WORD   LSINIT
000044          LSCCP
000044  007676'          WORD   LSCLEAN
000046          LSHPCP
000046  004772'          WORD   LSHARD
000050          LSSPCP
000050  000000           WORD   0
000052          LSDEVP
000052  000210'          WORD   LSDVTYP
```

```
000054                          L$REPP
000054  000000                  .WORD   0
000056                          L$HPTP
000056  005020'                 .WORD   L$HW
000060                          L$SPTP
000060  000000                  .WORD   0
000062                          L$DRCT
000062  000202'                  WORD   L$DR
000064                          L$DRS
000064  000206'                  WORD   L$DRST
000066                          L$STA
000066  000000                  WORD   0
000070                          L$AUT
000070  000000                  WORD   0
000072                          L$DUT
000072  000000                  WORD   0
000074                          L$PWRU
000074  000000                  WORD   0
000076                          L$LADP
000076  011166'                  WORD   L$LAST
```

```
                                        SBTTL  EQUATES
                                ;
                                ;       **************************************************
                                ;       THIS FACILITATES PROGRAM ASSEMBLY BY EQUATING
                                ;       SPECIFIC SYMBOLS WITH SPECIFIC VALUES VIA DIRECT
                                ;       ASSIGNMENT STATEMENTS.
                                ;       **************************************************

000011                          LUN     =11                     ,DEFINE LOGICAL UNIT NUMBER FROM 11

                                ,OFFSETS IN DEVICE PARAMETER BLOCK (DCB)

000002                          IO.FUN= 2
000004                          IO.LUN= 4
000006                          IO.EFN= 6
000010                          IO.IST= 10
000012                          IO.AST= 12
000014                          IO.A1=  14
000016                          IO.A2=  16
000020                          IO.A3=  20
000022                          IO.A4=  22
000024                          IO.A5=  24

                                , RSX I/O FUNCTION CODES

002400                          IO.WPC= 2400                    ,WRITE PERIPHERAL BLOCK
003400                          IO.RXP= 3400                    ,READ TRANSPARENT
003000                          IO.RPC= 3000                    ,READ PERIPHERAL BLOCK
```

```
      72          000001                  BIT00=  1
      73          000002                  BIT01=  2
      74          000004                  BIT02=  4
      75          000010                  BIT03=  10
      76          000020                  BIT04=  20
      77          000040                  BIT05=  40
      78          000100                  BIT06=  100
      79          000200                  BIT07=  200
      80          000400                  BIT08=  400
      81          001000                  BIT09=  1000
      82          002000                  BIT10=  2000
      83          004000                  BIT11=  4000
      84          010000                  BIT12=  10000
      85          020000                  BIT13=  20000
      86          040000                  BIT14=  40000
      87          100000                  BIT15=  100000
```

```
     89                                      .SBTTL  P-TABLE MESSAGES
     90                                ;
     91                                ;       **********************************************
     92                                ;       ++NOTE++ P-TABLE MESSAGES ARE USED DURING
     93                                ;       DIALOGUE WITH THE OPERATOR.
     94                                ;       **********************************************
     95                                ;
     96                                ;
     97                                ;
     98 000100      105     115     124  GETCNT. .ASCIZ/ENTER CONTROLLER NUMBER  /
        000103      105     122     040
        000106      103     117     116
        000111      124     122     117
        000114      114     114     105
        000117      122     040     116
        000122      125     115     102
        000125      105     122     040
        000130      072     000
     99 000132      105     116     124  GETERM·  ASCIZ/ENTER TERMINALS ON THAT CONTROLLER  /
        000135      105     122     040
        000140      124     105     122
        000143      115     111     116
        000146      101     114     123
        000151      040     117     116
        000154      040     124     110
        000157      101     124     040
        000162      103     117     116
        000165      124     122     117
        000170      114     114     105
        000173      122     040     072
        000176      000
    100
    101                                                 EVEN
```

```
                                        .SBTTL  GENERAL AND DPB STORAGE LOCATIONS
                                ;
                                ;       ************************************************
                                ;       THIS SECTION OF THE SOFTWARE CONTAINS ALL
                                ;       DIRECTIVE PARAMETER BLOCKS AS USED BY RSX-11M.
                                ;       ALSO CONTAINS PERMANENT/TEMPORARY STORAGE.
                                ;       ************************************************
                                ;
                                ;

000200                          DEVREG  1,1
000200  000001                                 .WORD   1
000202                          L$DR .
000202  000001                                 .WORD   1
000204  000001                                 .WORD   1
000206                          L$DRST
                                           .BLKW   1
                                ;       ************************************************
                                ;       THE DEVREG CALL ASSEMBLES A PAIR OF TABLES
                                ;       WHICH ARE USED TO DEFINE THOSE DEVICE REGISTERS
                                ;       THAT WILL BE USED BY THE PROGRAM.
                                ;       ************************************************
000210                          DEVTYPE SB
000210                          L$DVTYP
000210   123   102   000                    .ASCIZ  @SB@
                                           .EVEN
                                ;
                                ;       ************************************************
                                ;       THE DEVTYP CALL SPECIFIES THE EQUIPMENT THAT
                                ;       THE EXERCISER PROGRAM WILL SERVE
                                ;       ************************************************
                                ;
                                ;
                                ;       ************************************************
                                ;       THE QUED I/O DIRECTIVE PARAMETER BLOCK IS SET
                                ;       UP AS FOLLOWS ... ..
                                ;       ************************************************

000214                          Q1ODPB  Q1O$    IO.FUN,LUN,,,IOST,AST,<SCBADR,SCBCNT,40,PCBADR,PCBCNT>
000214     001   014                    .BYTE   1,12
000216  000002                          WORD    IO FUN
000220  000011                          WORD    LUN
000222     000   000                    .BYTE   ,0
000224  000744'                         WORD    IOST
000226  007634'                         WORD    AST
000230  000700'                         .WORD   SCBADR
000232  000702'                         .WORD   SCBCNT
000234  000040                          WORD    40
000236  000704'                         .WORD   PCBADR
000240  000706'                         .WORD   PCBCNT
000242  000000                          .WORD   0

                                ;
                                ;       ************************************************
                                ;       THE ASSISN LOGICAL UNIT PARAMETER BLOCK IS
                                ;       DEFINED AS FOLLOWS
                                ;       ************************************************
```

```
    139 000244                          ALUDPB: ALUN$   LUN,SB,0                 ;DIRECTIVE PARAMETER BLOCK FOR ALUN
        000244      007     004                 .BYTE   7,4
        000246  000011                          .WORD   LUN
        000250      123     102                 .ASCII  /SB/
        000252  000000                          .WORD   0
    140                                 ;
    141                                 ;
    142                                 ,       ***********************************************
    143                                 ,       THE GET LUN PARAMETER BLOCK IS DEFINED AS
    144                                 ,       FOLLOWS.....
    145                                 ,       ***********************************************
    146                                 ;
    147                                 ,
    148                                 ,
    149 000254                          GLUDPB  GLUN$   11,ERRBRT               ;PARAMETER BLOCK FOR GLUN
        000254      005     003                 .BYTE   5,3
        000256  000011                          .WORD   11
        000260  000262'                         .WORD   ERRBRT
    150                                 ,
    151                                 ,
    152                                 ,       ***********************************************
    153                                 ,               STORAGE LOCATIONS
    154                                 ,       ***********************************************
    155                                 ,
    156                                 ,
    157 000262                          ERRBRT  .BLKW   6                       ;GLUN INFO BUFFER I E. BROADCAST ERROR
    158 000276                          UNTBL:  .BLKW   64.                     ,RUNTIME DEVICE UNIT TABLES
    159 000476                          DEVTBL: .BLKW   64.                     ;ACTIVE LUN,,DEV TBLLE
    160 000676  000000                  UNITS:  .WORD   0                       ,NUMBER OF ACTIVE TERMINALS
    161
    162                                 ;*******************************************
    163                                 ,Q10 DEVICE DEPENDENT PARAMETERS.
    164                                 ;*******************************************
    165
    166 000700  000000                  SCBADR: .WORD   0
    167 000702  000000                  SCBCNT.   WORD   0
    168 000704  000000                  PCBADR. .WORD   0
    169 000706  000000                  PCBCNT: .WORD   0
    170
    171                                 ;*******************************************
    172
    173
    174 000710  000000                  TEMP    .WORD   0               ,THE FOLLOWING FIVE LOCATIONS ARE TEMPORARY
    175 000712  000000                  TEMP1.  .WORD   0
    176 000714  000000                  TEMP2:  .WORD   0
    177 000716  000000                  TEMP4:  .WORD   0
    178 000720  000000                  TEMP5:  .WORD   0
    179 000722  000000                  TEMP6   .WORD   0
    180 000724  000000                  TEMP7   .WORD   0
    181 000726  000001                  PASFG   .WORD   1               ,PROGRAM PASS INDICATOR
    182 000730  000000                  MINFLG  .WORD   0               ,2 MINUTE FLAG INDICATOR
    183 000732  000000                  ASTFLG: .WORD   0               ,AST INDICATOR
    184 000734  000000                  ERRFLG  .WORD   0               ,ERROR FLAG INDICATOR
    185 000736  000000                  PASS      WORD  0               ,PASS COUNT INDICATOR
    186 000740                          BUFF      BLKW  2               ,OUTPUT BUFFER ADDRESS
    187 000744                          IOST      BLKW  2               ,I/O STATUS ADDRESS
    188 000750                          BUF       BLKW  10              ,TIME PARAMETER BUFFER
```

```
    189 000770                          OFFLIN  .BLKW   64.          ,OFFLINE TABLE ADDRESS
    190 001170                          ERRBLK  .BLKW   448.         ;ERROR BLOCK ADDRESS
    191 002770                          IBUF0.  .BLKB   128          ;THE FOLLOWING 8 BUFFER LOCATIONS ARE TEMP
    192 003170                          IBUF1   .BLKB   128.
    193 003370                          IBUF2   .BLKB   128
    194 003570                          IBUF3.  .BLKB   128
    195 003770                          IBUF4.  .BLKB   128.
    196 004170                          IBUF5   .BLKB   128
    197 004370                          IBUF6   .BLKB   128
    198 004570                          IBUF7    BLKB   128
    199
    200                                 EVEN
```

```
                                        SBTTL  HARDWARE P-TABLE
                                ;
                                ;
                                ;       ************************************************************
                                ;       P-TABLES IN GENERAL ARE USED TO PROVIDE CERTAIN
                                ;       HARDWARE PARAMETERS I.E. VECTOR ADDRESSES,PRIORITY
                                ;       LEVELS, E.T.C. THE GET PARAMETER DATA (GPRMD) CALL
                                ;       IS USED HERE FOR HARDWARE PARAMETER CODING....
                                ;       ************************************************************
                                ;
                                ;
                                ;
                                BGNHRD
004770                                                  .WORD L10000-L$HARD/2
004770  000012
004772                          L$HARD .
                                GPRMD   GETCNT,0,0,177400,0,7,YES
004772                                                   WORD   T$CODE
004772  000032
004774  000100'                                         .WORD   GETCNT
004776  177400                                           WORD   177400
005000  000000                                          .WORD   T$LOLIM
005002  000007                                          .WORD   T$HILIM
                                GPRMD   GETERM,0,D,377,1,63 ,YES
005004                                                  .WORD   T$CODE
005004  000052
005006  000132'                                         .WORD   GETERM
005010  000377                                           WORD   377
005012  000001                                           WORD   T$LOLIM
005014  000077                                           WORD   T$HILIM
                                                WORD    T$HILIM

                                ENDHRD
                                                EVEN
005016                          L10000
```

```
DZKC1   MACRO M11    10-OCT-77 10:33  PAGE 8                                                    SEQ 0029
DEFAULT HARDWARE P-TABLE

                                        SBTTL  DEFAULT HARDWARE P-TABLE
                                ;
                                ;
                                ;       ****************************************
                                ;       PARAMETERS HERE ARE USED AS DEFAULT IN
                                ;       OPERATOR DIALOGUE . . . . . .
                                ;       ****************************************
                                ;
                                ;

005016                          BGNHW
005016  000001                                  WORD   L10001-L$HW/2
005020                          L$HW
005020  000001                          WORD   1                        ,CONTROLLER/TERMINAL DEFAULT
005022                          ENDHW
005022                          L10001
```

```
                                        SBTTL  SOFTWARE P-TABLE
                                ;
                                ;
                                ;       ****************************************
                                ;       ANOTHER SEPERATE AND OPTIONAL PARAMETER
                                ;       TABLE ASSEMBLED WITH THE DIAGNOSTIC
                                ;       PROGRAM. . . ...
                                ;       ****************************************
005022                          BGNSFT
005022  000000                                  WORD L10002-L$SOFT/2
005024                          L$SOFT
005024                          ENDSFT
                                        EVEN
005024                          L10002

                                ;BUILD SOFTWARE P-TABLE

005024                          BGNSW
005024  000000                          WORD   L10003-L$SW/2
005026                          L$SW

005026                          ENDSW
005026                          L10003

005026                          BGNRPT
005026                          L$RPT

005026                          ENDRPT
005026                          L10004
005026  104025                          EMT     C$RPT
```

```
                        SBTTL  DISPATCH TABLE
                ;
                ;
                ;       ************************************************
                ;       THIS CALL IS USED AS A DIRECTIVE FOR THE ASSEMBLY
                ;       OF A DISPATCH TABLE WHICH WILL CONTAIN THE SYMBOLIC
                ;       ADDRESS OF EACH TEST CONTAINED IN THE DIAGNOSTIC
                ;       PROGRAM . . . . . .
                ;       ************************************************
                ;
                ;
005030                  DISPATCH        1
005030  000001                                  .WORD   1
005032                  L$D SPATCH
005032  006274'                                 WORD    T1
```

```
                                SBTTL  INIT CODE
                        ;
                        ;
                        ;       ********************************************************************
                        ;       THIS INITIALIZATION SECTION OF THE PROGRAM PROVIDES OPERATOR
                        ;       INTERFACE.  AT ENTRY A TABLE HOLDS A MAXIMUM OF 3 CONTROLLERS
                        ;       AND 63 TERMINALS PER CONTROLLER.  THE SYSTEM GENERATION IS
                        ;       CHECKED SO THAT A TABLE OF LOGICAL UNITS IS BUILT  THIS IS
                        ;       THEN EQUATED TO PHYSICAL DEVICES ON A SPECIFIED CONTROLLER
                        ;       HAVING DONE THE ABOVE, THE ROUTINE ASSIGNES LOGICAL UNIT
                        ;       NUMBERS (LUNS) TO LOGICAL DEVICES (SB'S)....
                        ;       --
                        ;       RESOURCES USED FROM RSX-11M ARE THE DIRECTIVES,
                        ;       QIO$, DIR$, ALUN$, GLUN$, EXIT$S
                        ;       RESOURCES USED FROM THE DIAGNOSTIC SUPERVISER,
                        ;       ARE THE MACROS GPHARD, PRINT
                        ;       --
                        ;       GPHARD :  THE GET PARAMETER HARD CALL, REQUEST A POINTER
                        ;       TO THE TABLE ,WHICH HOLDS THE CONTROLLERS AND TERMINALS
                        ;       ON THAT CONTROLLER
                        ;       PRINT :  OUTPUT MESSAGES AND ERRORS TO THE TELETYPE
                        ;       --
                        ;       AT EXIT THE ROUTINE PRINTS OUT, THE TOTAL NUMBER OF ACTIVE
                        ;       TERMINALS ON THE BUS, A PHYSICAL DEVICE ON A CONTROLLER,
                        ;       A CORRESPONDING LOGICAL DEVICE, AND AN ASSOCIATED LOGICAL
                        ;       UNIT NUMBER
                        ;       ++
                        ;       ********************************************************************
                        ;;
                        ;
                        ;
005034                  BGNINIT
005034                  LSINIT
                                CMKT$S                          ;CANCEL ANY OUTSTANDING MARK TIME REQUEST
005034  012746                  MOV     (PC)+,-(SP)
005036     033     001          BYTE    27.,1
005040  104377                  EMT      0(377)
005042  005037  000726'         CLR     PASFG                  ;INIT INDICATOR
005046  005037  000736'         CLR     PASS                   ;INIT INDICATOR
                        ;       ************************************************************
                        ;       CLEAR STORAGE LOCATIONS FOR SUBSEQUENT INITIALIZATION
                        ;       ************************************************************
005052  012701  000262'         MOV     #ERRBRT,R1
005056  005021          139$    CLR     (R1)+
005060  020127  002770'         CMP     R1,#IBUF0
005064  001374                  BNE     139$
005066  012701  002770'         MOV     #IBUF0,R1
005072  105021          50$     CLRB    (R1)+
005074  020127  004570'         CMP     R1,#IBUF7
005100  001374                  BNE     50$
                        ;       ************************************************************
005102  005037  000226'         CLR     Q10DPB+10.AST          ;NO AST FOR DETACH
005106  012737  000000G 000216' MOV     #IO.DET,Q10DPB+10.FUN  ;I/O FUNCTION CODE
005114  012705  000011          MOV     #LUN,R5                ;LOGICAL UNIT NUMBER
005120  010537  000220'  30$     MOV     R5,Q10DPB+10.LUN       ;LUN DPB
005124                          DIR$    #Q10DPB                ;ISSUE I/O REQUEST
```

```
        005124  012746  000214'                MOV     #Q1ODPB,-(SP)
        005130  104377                          EMT      0(377)
    330 005132  005205                          INC     R5                      ,UPDATE LUN POINTER
    331 005134  020527  000113                  CMP     R5,#75.                 ,REQUEST FINISHED?
    332 005140  003767                          BLE     30$                     ;CONTINUE REQUEST IF NOT
    333 005142  012705  000276'                MOV     #UNTBL,R5               ,GET 64 ONE-WORD UNIT TABLE
    334 005146  013703  000014'                MOV     L$UNIT,R3               ,NUMBER OF UNITS FROM HEADER
    335 005152  010337  000676'                MOV     R3,UNITS                ,SAFE KEEP NUMBER OF UNITS
    336 005156  005004                          CLR     R4                      ,INIT POINTER TO P-TABLE
    337 005160                          1$      GPHARD  R4,R0                   ,GET POINTER TO HARDWARE P-TABLE
        005160  010400                                  MOV     R4,R0
        005162  104055                                  EMT     C$GPHRD
    338 005164  011025                          MOV     (R0),(R5)+              ,GET TERM & CONTROLLER # FROM HARDWARE
    339                                                                         ,P-TABLE ,PUT IN UNIT TABLE
    340 005166  005204                          INC     R4                      ,NEXT TABLE
    341 005170  005303                          DEC     R3                      ,LAST P-TABLE?
    342 005172  001372                          BNE     1$                      ,BRANCH IF NO---
    343 005174  012715  177777                  MOV     #-1,(R5)                ,ELSE INSERT END OF TABLE FLAG
    344 005200  005000                          CLR     R0                      ,OFFSETS TO TEMPORARY TABLES IN IBUF0
    345 005202  005003                          CLR     R3                      ,SB # INDICATOR
    346 005204  012702  177777                  MOV     #-1,R2                  ,CONTROLLER # INDICATOR FIRST TIME THRU
    347 005210  005004                          CLR     R4                      ,TERMINAL # INDICATER
    348 005212  012737  000011  000246' 2$      MOV     #LUN,ALUDPB+2           ,SET THE LUN =11
    349 005220  010337  000252'                 MOV     R3,ALUDPB+6             ,SET THE DEVICES
    350 005224                                  DIR$    #ALUDPB                 ,ISSUE I/O REQUEST
        005224  012746  000244'                MOV     #ALUDPB,-(SP)
        005230  104377                          EMT      0(377)
    351 005232  103444                          BCS     4$                      ,CHECK IF REQUEST IS REJECTED
    352 005234  012737  000011  000256'        MOV     #LUN,GLUDPB+2           ,ELSE SET THE LUN FOR THE G- DIRECTIVE
    353 005242  012737  000262' 000260'        MOV     #ERRBRT,GLUDPB+4        ,GET THE LUN INFO BUFFER ADDRESS
    354 005250                                  DIR$    #GLUDPB                 ,ISSUE I/O DIRECTIVE
        005250  012746  000254'                MOV     #GLUDPB,-(SP)
        005254  104377                          EMT      0(377)
    355 005256  005737  000266'                TST     ERRBRT+4                ,CHECK FOR BROADCAST CHANNEL
    356 005262  001413                          BEQ     3$                      ,BRANCH IF BROADCAST CHANNEL
    357 005264  110460  002770'                MOVB    R4,IBUF0+0(R0)          ,STORE TERMINAL # IN TABLE
    358 005270  110260  002771'                MOVB    R2,IBUF0+1(R0)          ,STORE CONTROLLER #
    359 005274  010360  003770'                MOV     R3,IBUF4(R0)            ,STORE SB #
    360 005300  005204                          INC     R4                      ,UPDATE PHYSICAL TERMINAL #
    361 005302  062700  000002                 ADD     #2,R0                   ,UPDATE POINTER
    362 005306  005203                          INC     R3                      ,UPDATE SB #
    363 005310  000740                          BR      2$                      ,CONTINUE I/O REQUEST TO NEXT TERMINAL
    364 005312  005202                  3$      INC     R2                      ,INCREMENT CONTROLLER # I E  LOG NEXT CONTROLLER
    365 005314  005004                          CLR     R4                      ;CLEAR TERMINAL #  I E TERMINALS ON NEXT CONTROLLER
    366 005316  110460  002770'                MOVB    R4,IBUF0+0(R0)          ,STORE TERMINAL #
    367 005322  110260  002771'                MOVB    R2,IBUF0+1(R0)          ,STORE CONTROLLER #
    368 005326  010360  003770'                MOV     R3,IBUF4(R0)            ,STORE SB
    369 005332  005204                          INC     R4                      ,UPDATE TERMINAL #
    370 005334  005203                          INC     R3                      ,UPDATE SB #
    371 005336  062700  000002                 ADD     #2,R0                   ,UPDATE POINTER
    372 005342  000723                          BR      2$                      ,ASSIGN NEXT TERMINAL
    373 005344  012760  177777  002770'  4$     MOV     #-1,IBUF0(R0)           ,END OF SB TABLE
    374 005352  013703  000676'                MOV     UNITS,R3                ,GET RUNTIME UNIT #
    375 005356  005004                          CLR     R4                      ,POINTER TO UNTBL CONTAINS
    376                                                                         ,PHYSICAL # IN LO-BYTE
    377                                                                         ,& CONTROLLER # IN HI-BYTE
    378 005360  005001                  5$      CLR     R1                      ,POINTER TO IBUF0 TABLE
```

```
                                                MOV     UNTBL(R4),R2            ;GET CONTROLLER # TERM # FROM UNIT TABLE
                                        6$      CMP     R2,IBUFO(R1)            ;CHECK IF SAME FROM MAPPED TABLE
                                                BEQ     7$                      ;BRANCH IF SAME
                                                TST     IBUFO(R1)               ,CHECK IF END OF TABLE
                                                BMI     8$                      ;BRANCH IF END
                                                ADD     #2,R1                   ,GET NEXT ITEM
                                                BR      6$                      ;CONTINUE CHECKING
                                        7$      MOVB    IBUF4(R1),DEVTBL+0(R4)  ;STORE LOGICAL SB NUMBER
                                                ADD     #2,R4                   ;GET NEXT UNTBL ENTRY
                                                DEC     R3                      ,LAST PHYSICAL TERMINAL TO BE STORED
                                                BNE     5$                      ,BRANCH IF NO--
                                                BR      11$                     ,ELSE GET NEXT DEVICE
                                        8$      MOV     UNTBL(R4),TEMP          ;TEMPORARY STORAGE FOR PRINT
                                                PRINTF  #MSG0,<B,TEMP+1>,<B,TEMP+0> ,PRINT THE NON-SYSGEN MESSAGE
005436  005046                                                    CLR     -(SP)
005440  153716  000710'                                          BISB    TEMP+0,(SP)
005444  005046                                                    CLR     -(SP)
005446  153716  000711'                                          BISB    TEMP+1,(SP)
005452  012746  007700'                                          MOV     #MSG0,-(SP)
005456  012746  000003                                          MOV     #3,-(SP)
005462  010600                                                    MOV     SP,R0
005464  104017                                                    EMT     C$PNTF
005466  062706  000010                                          ADD     #10,SP
                                                ;
                                                ;
                                                ;       ********************************************************************
                                                ;       BUBBLE ROUTINE    IF ONE, OR ALL TERMINALS ENTERED, ARE
                                                ;       UNASSIGNABLE OR UNATTACHABLE, THIS ROUTINE ELIMINATES THEM
                                                ;       OFF THE TABLE, AND ABORTS THE PROGRAM
                                                ;       --
                                                ;       ********************************************************************
                                                ;
                                                ;
                                                ;
                                                DEC     R3                      ,ONE TOO MANY UNITS? UNASSIGNABLE?
                                                BEQ     10$                     ,BRANCH IF ONE TOO MANY
                                                PUSH    <R3,R4>                 ,SAVE
005476  010346
005500  010446
005502  016464  000300' 000276'  9$            MOV     UNTBL+2(R4),UNTBL(R4)   ,BUBBLE UP FROM BELOW  TO THIS LOCATION
005510  062704  000002                          ADD     #2,R4                   ,NEXT LOCATION
005514  005303                                  DEC     R3                      ,DECREASE LOCATION
005516  001371                                  BNE     9$                      ,BRANCH IF NOT DONE
005520                                          POP     <R4,R3>                 ,RESTORE
005520  012604
005522  012603
005524  005337  000676'                        DEC     UNITS                   ,ELIMINATE ONE UINIT OFF TABLE
005530  005737  000676'                        TST     UNITS                   ,ARE THERE ANY MORE DEVICES
005534  001311                                  BNE     5$                      ,YES--BRANCH
005536                                          PRINTF  #MABORT                 ,NO-PRINT ABORT MESSAGE
005536  012746  010342'                                        MOV     #MABORT,-(SP)
005542  012746  000001                                        MOV     #1,-(SP)
005546  010600                                                  MOV     SP,R0
005550  104017                                                  EMT     C$PNTF
005552  062706  000004                                        ADD     #4,SP
005556                                          EXIT$S                          ,AND EXIT TEST
005556  012746                                  MOV     (PC)+,-(SP)
```

```
        005560     063     001                      BYTE    51,1
        005562  104377                              EMT      0<377>
    417 005564  005337  000676'          10$       DEC     UNITS                   ;ONE TOO MANY UNITS
    418 005570  005737  000676'                    TST     UNITS                   ,LAST DEVICE?
    419 005574  001013                              BNE     11$                     ;BRANCH IF NO
    420 005576                                      PRINTF  #MABORT                 ;ELSE PRINT ABORT MESSAGE
        005576  012746  010342'                              MOV     #MABORT,-(SP)
        005602  012746  000001                              MOV     #1,-(SP)
        005606  010600                                      MOV     SP,R0
        005610  104017                                      EMT     C$PNTF
        005612  062706  000004                              ADD     #4,SP
    421 005616                                      EXIT$S                          ;AND EXIT
        005616  012746                              MOV     (PC)+,-(SP)
        005620     063     001                      BYTE    51,1
        005622  104377                              EMT      0<377>
    422 005624  005001                  11$         CLR     R1                      ,DEVICE TABLE POINTER
    423 005626  012705  000011                      MOV     #LUN,R5                 ,SET THE LUN
    424 005632  013703  000676'                      MOV     UNITS,R3                ,KEEP COUNT
    425 005636  012737  0000006 000216'             MOV     #10.ATT,Q10DPB+10.FUN   ,SET FOR ATTACHES
    426 005644  110537  000246'          12$       MOVB    R5,ALUDPB+2             ,SET LUN FOR DIR$
    427 005650  116102  000476'                    MOVB    DEVTBL+0(R1),R2         ,GET THE SB DEVICE
    428 005654  110237  000252'                    MOVB    R2,ALUDPB+6             ,SET THE DEVICE NUMBER
    429 005660                                      DIR$    #ALUDPB                 ,ISSUE THE I/O REQUEST
        005660  012746  000244'                    MOV     #ALUDPB,-(SP)
        005664  104377                              EMT      0<377>
    430 005666  103017                              BCC     13$                     ,BRANCH IF REQUEST IS ACCEPTED
    431 005670  013704  0000006                     MOV     $DSW,R4                 ,IF NOT GET DSW ERROR CODE
    432 005674                                      PRINTF  #MSG1,<B,R2>,<B,R4>     ,PRINT ERROR MESSAGE
        005674  005046                                      CLR     -(SP)
        005676  150416                                      BISB    R4,(SP)
        005700  005046                                      CLR     -(SP)
        005702  150216                                      BISB    R2,(SP)
        005704  012746  007773'                             MOV     #MSG1,-(SP)
        005710  012746  000003                              MOV     #3,-(SP)
        005714  010600                                      MOV     SP,R0
        005716  104017                                      EMT     C$PNTF
        005720  062706  000010                              ADD     #10,SP
    433                                                                         ,UNASSIGNABLE TERMINAL
    434 005724  000437                              BR      17$                 ,DO NOT TRY TO ATTACH--
    435                                                                         ,BUT GO GET RID OF UNIT
    436                                                                         ,FROM THE DEVICE TABLE
    437 005726  110561  000477'          13$       MOVB    R5,DEVTBL+1(R1)         ,PUT THE LUN IN THE DEVICE TABLE
    438 005732  010537  000220'                    MOV     R5,Q10DPB+10 LUN        ,SET THE LUN FOR AN ATTACH
    439 005736                                      DIR$    #Q10DPB                 ,ISSUE I/O REQUEST
        005736  012746  000214'                    MOV     #Q10DPB,-(SP)
        005742  104377                              EMT      0<377>
    440 005744  103411                              BCS     16$                     ,ERROR INDICATOR
    441 005746  062701  000002          14$         ADD     #2,R1                   ,GET NEXT SB (LOGICAL)
    442 005752  005205                              INC     R5                      ;NEXT LUN
    443 005754  005303                              DEC     R3                      ,CHECK FOR LAST SB
    444 005756  001332                              BNE     12$                     ,BRANCH IF NOT--GO DO NEXT ASSIGN
    445 005760                          15$
    446 005760  012761  177777  000476'             MOV     #-1,DEVTBL(R1)          ,INSERT END OF FLAG IN DEVICE TABLE
    447 005766  000457                              BR      22$                     ,AND PRINT ACTIVE DEVICE MAP
    448 005770  013704  0000006         16$         MOV     $DSW,R4                 ,PRINT ERROR MESSAGE  I E UNATTACHABLE DEVICE
    449 005774                                      PRINTF  #MSG,<B,R2>,<B,R4>
        005774  005046                                      CLR     -(SP)
```

```
        005776  150416                                          BISB    R4,(SP)
        006000  005046                                          CLR     -(SP)
        006002  150216                                          BISB    R2,(SP)
        006004  012746  010261'                                MOV     #MSG,-(SP)
        006010  012746  000003                                  MOV     #3,-(SP)
        006014  010600                                          MOV     SP,R0
        006016  104017                                          EMT     C$PNTF
        006020  062706  000010                                  ADD     #10,SP
    450                                 ;
    451                                 ;
    452                                 ;       ***********************************************
    453                                 ;       BUBBLE ROUTINE...
    454                                 ;       ***********************************************
    455                                 ;
    456                                 ;
    457 006024  005303                  17$     DEC     R3                      ;ONE TOO MANY UNITS IN DEVICE TABLE
    458                                                                         ;BECAUSE DEVICE IS UNASSIGNABLE
    459                                                                         ;OR CANNOT BE ATTACHED
    460 006026  001417                          BEQ     19$                     ;BRANCH IF ONE TOO MANY
    461 006030                                  PUSH    <R3,R1>
        006030  010346
        006032  010146
    462 006034  016161  000300' 000276' 18$     MOV     UNTBL+2(R1),UNTBL(R1)   ;BUBBLE UP FROM BELOW TO THIS LOCATION
    463 006042  016161  000500' 000476'         MOV     DEVTBL+2(R1),DEVTBL(R1) ;"      "   "    "   "  "      "
    464 006050  062701  000002                  ADD     #2,R1                   ;NEXT LOCATION
    465 006054  005303                          DEC     R3                      ;DECREASE UNITS LOCATION
    466 006056  001366                          BNE     18$                     ;BRANCH IF NOT FINISHED
    467 006060                                  POP     <R1,R3>
        006060  012601
        006062  012603
    468 006064  000667                          BR      12$                     ;GO DO NEXT ASSIGN I E ALUN$
    469 006066  005337  000676'         19$     DEC     UNITS                   ;ELIMINATE ONE UNIT OFF TABLE
    470                                                                         ;THIS INDICATES ONE UNIT LESS TO TEST
    471 006072  005737  000676'                 TST     UNITS                   ;CHECK FOR ONE DEVICE AND UNASSIGNABLE
    472 006076  001330                          BNE     15$                     ;BRANCH IF NOT------
    473 006100                                  PRINTF  #MABORT                 ;PRINT ABORT MESSAGE
        006100  012746  010342'                                MOV     #MABORT,-(SP)
        006104  012746  000001                                  MOV     #1,-(SP)
        006110  010600                                          MOV     SP,R0
        006112  104017                                          EMT     C$PNTF
        006114  062706  000004                                  ADD     #4,SP
    474 006120                                  EXIT$S                          ;AND EXIT
        006120  012746                          MOV     (PC)+,-(SP)
        006122     063     001                   BYTE    51.,1
        006124  104377                          EMT      0<377>
    475 006126                          22$
    476 006126  013703  000676'                 MOV     UNITS,R3                ;GET NUMBER OF ACTIVE DEVICES
    477 006132                                  PRINTF  #MSG2,<B,R3>            ;PRINT NUMBER OF ACTIVE DEVICES
        006132  005046                                  CLR     -(SP)
        006134  150316                                  BISB    R3,(SP)
        006136  012746  010052'                        MOV     #MSG2,-(SP)
        006142  012746  000002                          MOV     #2,-(SP)
        006146  010600                                  MOV     SP,R0
        006150  104017                                  EMT     C$PNTF
        006152  062706  000006                          ADD     #6,SP
    478                                                                         ;AND HEADER FOR ACTIVE DEVICE MAP
    479 006156  005000                          CLR     R0                      ;POINTER TO TABLES
```

```
    480 006160  116001  000477'      20$     MOVB    DEVTBL+1(R0),R1         ,GET LUN TO BE PRINTED
    481 006164  116002  000476'              MOVB    DEVTBL+0(R0),R2         ;GET LOGICAL SB NUMBER
    482 006170  116004  000277'              MOVB    UNTBL+1(R0),R4          ;GET CONTROLLER NUMBER
    483 006174  116005  000276'              MOVB    UNTBL+0(R0),R5          ,GET PHYSICPL TERM#
    484 006200                               PUSH    R0                      ;SAVE R0**SUPERVISOR USES IT
        006200  010046
    485 006202                               PRINTF  #MSG3,<B,R5>,<B,R4>,<B,R2>,<B,R1>  ,
        006202  005046                                       CLR     -(SP)
        006204  150116                                       BISB    R1,(SP)
        006206  005046                                       CLR     -(SP)
        006210  150216                                       BISB    R2,(SP)
        006212  005046                                       CLR     -(SP)
        006214  150416                                       BISB    R4,(SP)
        006216  005046                                       CLR     -(SP)
        006220  150516                                       BISB    R5,(SP)
        006222  012746  010176'                              MOV     #MSG3,-(SP)
        006226  012746  000005                               MOV     #5,-(SP)
        006232  010600                                       MOV     SP,R0
        006234  104017                                       EMT     C$PNTF
        006236  062706  000014                               ADD     #14,SP
    486 006242                               POP     R0                      ,RESTORE R0
        006242  012600
    487                                                                      ,PRINT ACTIVE DEVICE MAP
    488 006244  062700  000002               ADD     #2,R0
    489 006250  005303                       DEC     R3
    490 006252  001342                       BNE     20$
    491 006254  005237  000736'      21$.    INC     PASS                    ,INIT PASS LOCATION
    492 006260  013702  000676'              MOV     UNITS,R2                ,GET NUMBER OF TERMINALS
    493 006264  006302                       ASL     R2                      ,POINTER TO END OF TERMINALS
    494 006266  005037  000722'              CLR     TEMP6                   ,USED AS PRINT FLAG
    495
    496
    497 006272                       ENDINIT
        006272                       L10005
        006272  104011                                       EMT     CSINIT
    498
```

```
                                        .SBTTL  MAIN TEST
                                ;
                                ,
                                ,       ************************************************************************
                                ,       THIS PROGRAM ESSENTIALLY CONSISTS OF ONE TEST.
                                ,       AT ENTRY A TABLE HOLDS LOGICAL UNIT NUMBERS, ASSOCIATED WITH
                                ;       LOGICAL DEVICES. ANOTHER TABLE HOLDS PHYSICAL DEVICES ON ANY
                                ,       ONE OF THREE CONTROLLERS. A DIRECTIVE PARAMETER BLOCK IS THEN
                                ,       BUILT, FOLLOWED BY A REQUEST TO ALL LIVE TERMINALS ON THE BUS,
                                ,       TO START THEIR INTERNAL DIAGNOSTICS. THE PROGRAM WAITS FOR ALL
                                ,       TERMINALS TO COMPLETE THE DIAGNOSTICS, BY IMPLEMENTING A --
                                ,       SYSTEM DELAY DIRECTIVE, THEN READS, LOGS, AND REPORTS ALL
                                ,       ERRORS FROM EACH TERMINAL.
                                ,       --
                                ,       AT EXIT, PROGRAM PASSES, LOGICAL SB'S, TERMINAL TYPES,
                                ,       ERRORS, AND TERMINAL DIAGNOSTIC TEST #'S ARE REPORTED
                                ,       --
                                ,       RESOURCES USED FROM RSX-11M ARE THE DIRECTIVES,
                                ,       QIO$, DIR$, MRKT$S, AND ASYNCHRONOUS TRAP (AST)
                                ,       THE PRINT MACRO IS USED FROM THE DIAGNOSTIC SUPERVISOR
                                ,       --
                                ,       ************************************************************************
                                ,
                                ,

006274                          BGNTST
006274                          T1

006274                          START
006274  005037  000734'                CLR     ERRFLG          ,INIT ERROR INDICATOR FOR EACH PASS
006300  005037  000730'                CLR     MINFLG          ,INIT 2 MIN FLAG INDICATOR
006304  005001                        CLR     R1              ,INIT POINTER FOR DEVICE TABLE
006306  005037  000732'                CLR     ASTFLG          ;INIT AST INTERRUPT INDICATOR
006312  012737  040500  000740'        MOV     #40500,BUFF     ;LOAD PCB
006320  012737  040100  000742'        MOV     #40100,BUFF+2   ;LOAD TERMINAL DIAGNOSTICS COMMANDS
                                ,       ************************************************
                                ,               INITIALIZE THE ERROR TABLE
                                ,       ************************************************
006326  005061  001170'        20$     CLR     ERRBLK(R1)
006332  005721                        TST     (R1)+
006334  022701  000700                CMP     #448.,R1
006340  001372                        BNE     20$
006342  005001                        CLR     R1
                                ,       ************************************************
                                ,       BUILD QIO DIRECTIVE PARAMETER BLOCK .
                                ,       AND ISSUE THE DIRECTIVE TO TERMINAL
                                ,       ************************************************
006344  012737  002400  000216'        MOV     #IO.WPC,QIODPB+10.FUN   ;START DPB WITH FUNCTION CODE
006352  012737  000744' 000224'        MOV     #IOST,QIODPB+10.IST     ,LOAD STATUS WORD ADDRESS
006360  012737  007634' 000226'        MOV     #AST,QIODPB+10.AST      ;LOAD TERMINAL WRITE AST
006366  005037  000236'                CLR     QIODPB+10.A4            ;CLEAR INPUT BUFFER ADDRESS
006372  005037  000240'                CLR     QIODPB+10.A5            ,LOAD INPUT BUFFER SIZE
006376  012737  000740' 000230'        MOV     #BUFF,QIODPB+10 A1      ,LOAD OUTPUT BUFFER ADDRESS
006404  012737  000004  000232'        MOV     #4,QIODPB+10 A2         ,LOAD OUTPUT BUFFER SIZE
006412  116137  000477' 000220' 1$      MOVB    DEVTBL+1(R1),QIODPB+10 LUN      ,LOAD DPB WITH LUN
006420                                  DIR$    #QIODPB                 ,ISSUE I/O REQUEST WPC
```

```
        006420  012746  000214'                MOV     #Q10DPB,-(SP)
        006424  104377                         EMT      0<377>
556     006426  005737  000732'         2$     TST     ASTFLG                  ;CHECK FOR I/O COMPLETION (AST)
557     006432  001775                         BEQ     2$                      ;CONTINUE CHECK IF NOT COMPLETE
558     006434  005037  000732'                CLR     ASTFLG                  ;RESET AST INDICATOR
559     006440  005737  000734'                TST     ERRFLG                  ;CHECK FOR ERRORS FROM AST
560     006444  001423                         BEQ     4$                      ;BRANCH IF NONE
561     006446  005761  000770'                TST     OFFLIN(R1)              ;WAS THE TERMINAL OFF LINE?
562     006452  001015                         BNE     3$                      ;BRANCH IF YES-ONT
563     006454  005261  000770'                INC     OFFLIN(R1)              ;SET TERMINAL OFFLINE FLAG
564     006460                                 PRINTF  #MSG4,<B,DEVTBL+0(R1)>  ;INFORM OPERATOR OF NO RESPONSE
        006460  005046                                 CLR     -(SP)
        006462  156116  000476'                       BISB    DEVTBL+0(R1),(SP)
        006466  012746  010373'                       MOV     #MSG4,-(SP)
        006472  012746  000002                        MOV     #2,-(SP)
        006476  010600                                MOV     SP,R0
        006500  104017                                EMT     C$PNTF
        006502  062706  000006                        ADD     #6,SP
565     006506  005037  000734'         3$     CLR     ERRFLG                  ;RESET ERROR INDICATOR
566     006512  000420                         BR      5$                      ;GET NEXT TERMINAL
567     006514  005761  000770'         4$     TST     OFFLIN(R1)              ;CHECK IF TERMINAL WAS PREVIOUSLY OFFLINE
568     006520  001415                         BEQ     5$                      ;BRANCH IF NO
569     006522  005061  000770'                CLR     OFFLIN(R1)              ;ELSE MAKE HER ON LINE
570     006526                                 PRINTF  #MSG5,<B,DEVTBL+0(R1)>  ;INFORM OPERATOR TERMINAL IS RESPONDING
        006526  005046                                 CLR     -(SP)
        006530  156116  000476'                       BISB    DEVTBL+0(R1),(SP)
        006534  012746  010443'                       MOV     #MSG5,-(SP)
        006540  012746  000002                        MOV     #2,-(SP)
        006544  010600                                MOV     SP,R0
        006546  104017                                EMT     C$PNTF
        006550  062706  000006                        ADD     #6,SP
571     006554  005721                 5$     TST     (R1)+                   ;UPDATE POINTER
572     006556  020102                         CMP     R1,R2                   ;IS IT LAST TERMINAL
573     006560  001314                         BNE     1$                      ;NO  GET NEXT TERMINAL
```

```
                                        ,       ************************************************************
                                        ,       ,WAIT 2 MINUTES FOR TERMINAL TO COME BACK ON LINE
                                        ,       ,AFTER COMPLETING DIAGNOSTICS.  IF LESS THAN 2 MINS
                                        ,       ;CHECK FOR ERRORS AND GET THE NEXT TERMINAL. IF MORE
                                        ,       ,THAN 2 MINS, CHECK FOR ERRORS AND LOG AS AN OFFLINE
                                        ,       ;TERMINAL.
                                        ,       --
                                        ,       MRKT$S IS AN RSX-11M SYSTEM DIRECTIVE MEANING,
                                        ,       DECLARE A SIGNIFICANT EVENT...
                                        ,       --
                                        ,       WTSE$S IS AN RSX-11M SYSTEM DIRECTIVE MEANING,
                                        ;       WAIT FOR THE COMPLETION OF A SIGNIFICANT EVENT  .
                                        ,       --
                                        ;       ************************************************************

006562                                  6$:     MRKT$S  ,#120.,#2,#MAST         ;START 2 MINS PERIOD
006562  012746  007662'                          MOV     #MAST,-(SP)
006566  012746  000002                          MOV     #2,-(SP)
006572  012746  000170                          MOV     #120.,-(SP)
006576  005046                                  CLR     -(SP)
006600  012746                                  MOV     (PC)+,-(SP)
006602     027     005                          .BYTE   23.,5
006604  104377                                  EMT      0<377>

                                        ,       ************************************************************
                                        ,       READ AND CHECK FOR ERRORS FROM ALL LIVE TERMINALS
                                        ,       ************************************************************
006606  012737  040102  000740'                MOV     #40102,BUFF             ;PCB FOR READ
006614  005037  000734'                        CLR     ERRFLG                  ;RESET ERROR INDICATOR
006620  005001                                  CLR     R1                      ;POINTER TO TERMINAL TABLES
006622  005003                                  CLR     R3                      ;POINTER TO ERROR BLOCK
006624  012737  003000  000216'                MOV     #IO.RPC,Q10DPB+10.FUN   ;LOAD DPB WITH READ FUNCTION
006632  012737  000744' 000224'                MOV     #IOST,Q10DPB+10.IST     ;LOAD STATUS WORD ADDRESS
006640  012737  007634' 000226'                MOV     #AST,Q10DPB+10.AST      ;LOAD READ AST ADDRESS
006646  012737  000016  000232'                MOV     #14.,Q10DPB+10.A2       ;LOAD INPUT BUFFER SIZE
006654  012737  000740' 000236'                MOV     #BUFF,Q10DPB+10.A4      ;LOAD OUTPUT BUFFER ADDRESS
006662  012737  000004  000240'                MOV     #4,Q10DPB+10.A5         ,LOAD OUTPUT BUFFER SIZE
006670  005761  000770'          7$    TST     OFFLIN(R1)              ;CHECK OFFLINE TABLE
006674  001111                                BNE     9$                      ,BRANCH IF OFFLINE
006676  012737  001170' 000710'                MOV     #ERRBLK,TEMP            ;GET ADDRESS OF ERROR BLOCK
006704  060337  000710'                        ADD     R3,TEMP                 ;UPDATE ADDRESS OF EACH TERMINAL
006710  013737  000710' 000230'                MOV     TEMP,Q10DPB+10.A1       ,GET UPDATED ADDRESS OF ERROR BLOCK
006716  116137  000477' 000220'                MOVB    DEVTBL+1(R1),Q10DPB+10 LUN      ,LOAD LOGICAL UNIT NUMBER
006724                                  25$
006724                                  DIR$    #Q10DPB                 ;ISSUE I/O REQUEST
006724  012746  000214'                        MOV     #Q10DPB,-(SP)
006730  104377                                  EMT      0<377>
006732  103020                                  BCC     8$                      ,REQUEST ACCEPTED
006734  013737  000000G 000724'                MOV     $DSW,TEMP7              ,GET DSW ERROR
006742                                  PRINTF  #MSG10,TEMP7            ,Q10 NOT ACCEPTED
006742  013746  000724'                                MOV     TEMP7,-(SP)
006746  012746  010751'                                MOV     #MSG10,-(SP)
006752  012746  000002                                MOV     #2,-(SP)
006756  010600                                        MOV     SP,R0
006760  104017                                        EMT     C$PNTF
```

```
                     006762  062706  000006                          ADD     #6,SP
                618  006766  005261  000770'                 INC     OFFLIN(R1)              ;DROP TERMINAL
                619  006772  000452                         BR      9$
                620  006774                          8$:     MRKT$S  #1,#5.,#1               ,MARK FOR SIG EVENT
                     006774  005046                         CLR     -(SP)
                     006776  012746  000001                 MOV     #1,-(SP)
                     007002  012746  000005                 MOV     #5.,-(SP)
                     007006  012746  000001                 MOV     #1.,-(SP)
                     007012  012746                         MOV     (PC)+,-(SP)
                     007014     027     005                 .BYTE   23,5
                     007016  104377                         EMT      0<377>
                621  007020  005737  000732'                TST     ASTFLG                  ;CHECK FOR I/O COMPLETION (AST)
                622  007024  001006                         BNE     50$                     ,BRANCH IF AST COMPLETE
                623  007026                                 WTSE$S  #1                      ,WAIT FOR SIG EVENT
                     007026  012746  000001                 MOV     #1,-(SP)
                     007032  012746                         MOV     (PC)+,-(SP)
                     007034     051     002                  BYTE   41.,2
                     007036  104377                         EMT      0<377>
                624  007040  000755                         BR      8$
                625  007042  005037  000732'         50$    CLR     ASTFLG                  ,RESET AST INDICATOR
                626  007046  005737  000734'                TST     ERRFLG                  ,ANY STATUS CODE ERROR?
                627  007052  001422                         BEQ     9$                      ,BRANCH IF NO ---
                628  007054  005037  000734'                CLR     ERRFLG                  ;YES - TERMINAL OFFLINE,THEN RESET ERROR FLAG
                629  007060  005737  000730'                TST     MINFLG                  ;IS 2 MINS UP?
                630  007064  001717                         BEQ     25$                     ;IF NO- AGAIN ISSUE I/O REQUEST
                631  007066  005261  000770'                INC     OFFLIN(R1)              ;SET OFFLINE FLAG
                632  007072                                 PRINTF  #MSG6,<B,DEVTBL+0(R1)>  ;INFORM OPERATOR TERMINAL DIED
                     007072  005046                                 CLR     -(SP)
                     007074  156116  000476'                         BISB    DEVTBL+0(R1),(SP)
                     007100  012746  010513'                         MOV     #MSG6,-(SP)
                     007104  012746  000002                         MOV     #2,-(SP)
                     007110  010600                                 MOV     SP,R0
                     007112  104017                                 EMT     C$PNTF
                     007114  062706  000006                         ADD     #6,SP
                633  007120  005721                 9$     TST     (R1)+                           ,GET NEXT TERMINAL
                634  007122  062703  000016                 ADD     #14.,R3                         ,GET NEXT ERROR BLOCK
                635  007126  020102                         CMP     R1,R2                           ;LAST TERMINAL?
                635  007130  001257                         BNE     7$                              ,GO CHECK PRESENT TERMINAL IF NO
                637  007132  005001                         CLR     R1                              ,INIT DEVICE TABLE POINTER
                638  007134  005003                         CLR     R3                              ;INIT POINTER TO ERROR BLOCK
                639  007136  005037  000734'                CLR     ERRFLG                          ,RESET ERROR INDICATOR
                640  007142  005005                         CLR     R5                              ,INIT TERMINAL ERROR COUNT
                641  007144  010304                 12$    MOV     R3,R4                           ,ERROR POINTER
                642  007146  105764  001176'         10$    TSTB    ERRBLK+6(R4)                    ,CHECK FOR ERRORS IN TABLE
                643  007152  001513                         BEQ     13$                             ,IF NO ERRORS-GET NEXT ERROR BLOCK
                644  007154  005737  000734'                TST     ERRFLG                          ,IS HEADER PRINTED?
                645  007160  001024                         BNE     11$                             ,YES - PRINT ERRORS
                646  007162                                 PRINTF  #MSG7,PASS                      ,PRINT HEADER
                     007162  013746  000736'                         MOV     PASS,-(SP)
                     007166  012746  010621'                         MOV     #MSG7,-(SP)
                     007172  012746  000002                         MOV     #2,-(SP)
                     007176  010600                                 MOV     SP,R0
                     007200  104017                                 EMT     C$PNTF
                     007202  062706  000006                         ADD     #6,SP
                647  007206                                 PRINTF  #MSG8                           ,PRINT SUB-HEAD
                     007206  012746  010676'                         MOV     #MSG8,-(SP)
                     007212  012746  000001                         MOV     #1,-(SP)
```

```
                                                        MOV     SP,R0
                                                        EMT     C$PNTF
                                                        ADD     #4,SP
   648 007226  005237  000734'                  INC     ERRFLG                  ;HEADER PRINT INDICATOR
   649                                  ;       ****************************************
   650                                          ;SAVE ALL ERRORS FOR PRINT-OUT....
   651                                  ;       ****************************************
   652 007232  116137  000476' 000712' 11$:     MOVB    DEVTBL(R1),TEMP1        ;SAVE TERMINAL SB NUMBER
   653 007240  116337  001174' 000714'          MOVB    ERRBLK+4(R3),TEMP2      ;SAVE ASCII-TERMINAL TYPE
   654                                  ;       ----------------------------------------
   655                                          ;TRANSFORM ASCII CHARACTER TO TERMINAL TYPE
   656                                  ;       ----------------------------------------
   657                                  ;
   658 007246  042737  177700  000714'          BIC     #177700,TEMP2           ;101 BECOMES 001
   659                                                                          ;102 BECOMES 002
   660                                                                          ;103 BECOMES 003
   661 007254  006337  000714'                  ASL     TEMP2                   ;001 BECOMES 002
   662                                                                          ;002 BECOMES 004
   663                                                                          ;003 BECOMES 006
   664 007260  005337  000714'                  DEC     TEMP2                   ;002 BECOMES 001
   665                                                                          ;004 BECOMES 003
   666                                                                          ;006 BECOMES 005
   667 007264  062737  001440  000714'          ADD     #800.,TEMP2             ;001 BECOMES 801
   668                                                                          ;003 BECOMES 803
   669                                                                          ;004 BECOMES 805
   670
   671 007272  116437  001176' 000716'          MOVB    ERRBLK+6(R4),TEMP4      ;SAVE HI-BYTE TEST NUMBER IN ASCII
   672 007300  116437  001177' 000720'          MOVB    ERRBLK+7(R4),TEMP5      ;SAVE LO-BYTE TEST NUMBER IN ASCII
   673 007306  042737  000300  000716'          BIC     #300,TEMP4              ;TRANSFORM ASCII
   674 007314  042737  000300  000720'          BIC     #300,TEMP5              ;TRANSFORM ASCII
   675                                  ,       ****************************************
   676                                          ;PRINT ALL ERRORS AND END PASS
   677                                  ,       ****************************************
   678 007322                                   PRINTF  #FMT,<B,TEMP1>,TEMP2,<B,TEMP4>,<B,TEMP5>
       007322  005046                                   CLR     -(SP)
       007324  153716  000720'                         BISB    TEMP5,(SP)
       007330  005046                                   CLR     -(SP)
       007332  153716  000716'                         BISB    TEMP4,(SP)
       007336  013746  000714'                         MOV     TEMP2,-(SP)
       007342  005046                                   CLR     -(SP)
       007344  153716  000712'                         BISB    TEMP1,(SP)
       007350  012746  011011'                         MOV     #FMT,-(SP)
       007354  012746  000005                         MOV     #5,-(SP)
       007360  010600                                   MOV     SP,R0
       007362  104017                                   EMT     C$PNTF
       007364  062706  000014                         ADD     #14,SP
   679 007370  005205                           INC     R5                      ;UPDATE TERMINAL ERROR COUNT
   680 007372  005724                           TST     (R4)+                   ;UPDATE POINTER BY 2
   681 007374  020527  000004                   CMP     R5,#4                   ;CHECK IF END OF THIS BLOCK
   682 007400  001262                           BNE     10$                     ;IF NOT END,CHECK FOR MORE ERRORS
   683 007402  062703  000016          13$      ADD     #14 ,R3                 ;GET NEXT ERROR BLOCK
   684 007406  005721                           TST     (R1)+                   ;UPDATE TERMINAL POINTER
   685 007410  005005                           CLR     R5                      ;RESET ERROR COUNTER
   686 007412  020102                           CMP     R1,R2                   ;CHECK FOR END OF ERROR TABLE
   687 007414  001253                           BNE     12$                     ;CONTINUE CHECKING ERRORS IF NOT END
   688 007416  005737  000730'                  TST     MINFLG                  ;IS 2 MINS UP?
   689 007422  001006                           BNE     27$                     ;BRANCH IF YES
```

```
690 007424                              CMKT$S  ,,ERR                   ;ELSE CANCEL 2 MINS PERIOD
    007424  012746                      MOV     (PC)+,-(SP)
    007426     033      001             .BYTE   27.,1
    007430  104377                      EMT      0<377>
    007432  103002                      .IIF EQ  0<$$$T1-37>,   BCC     .+6
    007434  004737  007606'              JSR     PC,ERR
691                                     ;*************************************************************************
692                                     ;PRINT PASS ONLY FOR ACTIVE TERMINALS
693                                     ;*************************************************************************
694 007440  005004                  27$ CLR     R4                      ;POINTER TO OFFLINE TERMINALS
695 007442  005001                      CLR     R1                      ;POINTER TO OFFLINE TABLE
696 007444  005761  000770'          30$ TST     OFFLIN(R1)              ;IS TERMINAL OFFLINE?
697 007450  001423                      BEQ     26$                     ;BRANCH IF NO AND DO END OF PASS
698 007452  005204                      INC     R4                      ;KEEP TRACK OF NON-ACTIVE TERMINALS
699 007454  005721                      TST     (R1)+                   ;UPDATE POINTER TO OFFLINE TABLE
700 007456  023704  000676'              CMP     UNITS,R4                ;LAST TERMINAL?
701 007462  001370                      BNE     30$                     ;CONTINUE CHECKING IF NO
702 007464  005737  000722'              TST     TEMP6                   ;IS PRINT FLAG SET
703 007470  001012                      BNE     24$                     ;IF YES DO NOT PRINT
704 007472                              PRINTF  #NACT                   ;INFORM OF NO ACTIVE TERMINALS
    007472  012746  011113'                          MOV     #NACT,-(SP)
    007476  012746  000001                           MOV     #1,-(SP)
    007502  010600                                   MOV     SP,R0
    007504  104017                                   EMT     C$PNTF
    007506  062706  000004                           ADD     #4,SP
705 007512  005237  000722'              INC     TEMP6                   ;SET PRINT FLAG
706 007516  000427                  24$ BR      46$                     ;DO NEXT PASS
707 007520  005037  000722'          26$ CLR     TEMP6                   ;RESET PRINT FLAG
708 007524                              GTIM$S  #BUF                    ;GET THE SYSTEM TIME
    007524  012746  000750'                MOV     #BUF,-(SP)
    007530  012746                          MOV     (PC)+,-(SP)
    007532     075      002                 BYTE    61 ,2
    007534  104377                          EMT      0<377>
709 007536                              PRINTX  #EOP,PASS,BUF+G TIHR,BUF+G TIMI         ;REPORT PASS COUNT
    007536  013746  000760'                        MOV     BUF+G TIMI,-(SP)
    007542  013746  000756'                        MOV     BUF+G.TIHR,-(SP)
    007546  013746  000736'                        MOV     PASS,-(SP)
    007552  012746  011042'                        MOV     #EOP,-(SP)
    007556  012746  000004                         MOV     #4,-(SP)
    007562  010600                                 MOV     SP,R0
    007564  104015                                 EMT     C$PNTX
    007566  062706  000012                         ADD     #12,SP
710 007572  005237  000736'              INC     PASS                    ;UPDATE PASS COUNT
711 007576                          46$ BREAK
    007576  104022                          EMT     C$BRK
712 007600  000137  006274'              JMP     START                   ;GO DO NEXT PASS
713
714
715
716 007604                          ENDTST
    007604                          L10006
    007604  104001                          EMT     C$ETST
717                                     ;
718                                     ;       ***************************************************
719                                     ;IF CANCEL MARK TIME DIRECTIVE IS NOT ACCEPTED
720                                     ;ABORT PROGRAM, AND EXIT TO MCR
721                                     ;       ***************************************************
```

```
722                                             ,
723 007606                                      ERR
724 007606                                              PRINTF  #MABORT                 ;PRINT ABORT MESSAGE
    007606  012746  010342'                                        MOV     #MABORT,-(SP)
    007612  012746  000001                                          MOV     #1,-(SP)
    007616  010600                                                  MOV     SP,R0
    007620  104017                                                  EMT     C$PNTF
    007622  062706  000004                                          ADD     #4,SP
725 007626                                              EXIT$S                          ;AND EXIT
    007626  012746                                      MOV     (PC)+,-(SP)
    007630     063      001                             .BYTE   51 ,1
    007632  104377                                      EMT      0<377>
```

```
                                ;*****************************************************************
                                ;++AST SERVICE ROUTINE...
                                ;THIS SYSTEM TRAP IS ENTERED UPON A WRITE/READ I/O REQUEST
                                ;COMPLETION. UPON ENTRY, A TWO WORD LOCATION CONTAINS I/O
                                ;ERROR CODES. AN ERROR FLAG IS SET, FOR ANY TERMINAL
                                ;GENERATING ERRORS AT COMPLETION OF AN I/O REQUEST. AT EXIT,
                                ;THE ROUTINE INDICATES THAT THE TRAP HAS OCCURRED, AND THE
                                ;STACK POINTER IS READJUSTED, FOR THE FOLLOWING AST
                                ;*****************************************************************
                                ;
                                ;

007634                          AST                             ;TERMINAL WRITE-READ AST
007634  105737  000744'          TSTB    IOST                    ;CHECK FOR ERROR CODE
007640  100002                  BPL     1$                      ;BRANCH IF NO ERRORS
007642  005237  000734'          INC     ERRFLG                  ;SET ERROR INDICATOR
007646  005237  000732'  1$      INC     ASTFLG                  ;SET AST INDICATION
007652  005726                  TST     (SP)+                   ;AS PER RSX-ASTX$S
007654                          ASTX$S                          ;RETURN
007654  012746                  MOV     (PC)+,-(SP)
007656    163     001            BYTE    115.,1
007660  104377                  EMT      0<377>

                                ;*****************************************************************
                                ;THIS MARK-TIME IS ENTERED UPON COMPLETION OF A 2 MINUTE
                                ;PERIOD  HERE A 2 MINUTE COMPLETION FLAG IS SET
                                ;*****************************************************************
                                ;
007662                          MAST
007662  005237  000730'          INC     MINFLG                  ;SET 2 MINUTE FLAG
007666  005726                  TST     (SP)+                   ;INC AS PER RSX-ASTX$S
007670                          ASTX$S                          ;RETURN
007670  012746                  MOV     (PC)+,-(SP)
007672    163     001            BYTE    115.,1
007674  104377                  EMT      0<377>
```

```
    759
    760 007676                          BGNCLN
        007676                          L$CLEAN
    761
    762 007676                          ENDCLN
        007676                          L10007
        007676  104012                              EMT     C$CLEAN
    763
```

```
    765 007700  045  116  045  MSG0:   ASCIZ  "%N%ACONTROLLER %D1%A, TERMINAL %D3%A IS NOT SYSGENED IN %N"
        007703  101  103  117
        007706  116  124  122
        007711  117  114  114
        007714  105  122  040
        007717  045  104  061
        007722  045  101  054
        007725  040  124  105
        007730  122  115  111
        007733  116  101  114
        007736  040  045  104
        007741  063  045  101
        007744  040  111  123
        007747  040  116  117
        007752  124  040  123
        007755  131  123  107
        007760  105  116  105
        007763  104  040  111
        007766  116  056  045
        007771  116  000
    766 007773  045  116  045  MSG1:   .ASCIZ  "%N%ATERMINAL %D2%A CANNOT BE ASSIGNED-$DSW=%06"
        007776  101  124  105
        010001  122  115  111
        010004  116  101  114
        010007  040  045  104
        010012  062  045  101
        010015  040  103  101
        010020  116  116  117
        010023  124  040  102
        010026  105  040  101
        010031  123  123  111
        010034  107  116  105
        010037  104  055  044
        010042  104  123  127
        010045  075  045  117
        010050  066  000
    767 010052  045  116  045  MSG2    ASCII  "%N%ANUMBER OF ENTERED TERMINALS=%D2%N"
        010055  101  116  125
        010060  115  102  105
        010063  122  040  117
        010066  106  040  105
        010071  116  124  105
        010074  122  105  104
        010077  040  124  105
        010102  122  115  111
        010105  116  101  114
        010110  123  075  045
        010113  104  062  045
        010116  116
    768 010117  045  116  045        ASCIZ  "%N%AUNIT ** CONTROLLER ** LOGICAL SB ** LUN **"
        010122  101  125  116
        010125  111  124  040
        010130  052  052  040
        010133  103  117  116
        010136  124  122  117
        010141  114  114  105
        010144  122  040  052
```

```
        010147  052  040  114
        010152  117  107  111
        010155  103  101  114
        010160  040  123  102
        010163  040  052  052
        010166  040  114  125
        010171  116  040  052
        010174  052  000
   769  010176  045  116  045  MSG3    .ASCIZ"%N%D3%A          %D3%A          %D3%A       %D3%N"
        010201  104  063  045
        010204  101  040  040
        010207  040  040  040
        010212  040  040  040
        010215  040  040  040
        010220  040  045  104
        010223  063  045  101
        010226  040  040  040
        010231  040  040  040
        010234  040  040  040
        010237  040  045  104
        010242  063  045  101
        010245  040  040  040
        010250  040  040  040
        010253  045  104  063
        010256  045  116  000
   770  010261  045  116  045  MSG      ASCIZ  "%N%ATERMINAL %D2%A CANNOT BE ATTACHED-$DSW=%03%N"
        010264  101  124  105
        010267  122  115  111
        010272  116  101  114
        010275  040  045  104
        010300  062  045  101
        010303  040  103  101
        010306  116  116  117
        010311  124  040  102
        010314  105  040  101
        010317  124  124  101
        010322  103  110  105
        010325  104  055  044
        010330  104  123  127
        010333  075  045  117
        010336  063  045  116
        010341  000
   771  010342  045  116  045  MABORT   ASCIZ  "%N%APROGRAM IS ABORTED%N"
        010345  101  120  122
        010350  117  107  122
        010353  101  115  040
        010356  111  123  040
        010361  101  102  117
        010364  122  124  105
        010367  104  045  116
        010372  000
   772  010373  045  116  045  MSG4:   .ASCIZ  "%N%ATERMINAL   %D3%A IS NOT RESPONDING %N"
        010376  101  124  105
        010401  122  115  111
        010404  116  101  114
        010407  011  045  104
        010412  063  045  101
```

```
        010415      040      111      123
        010420      040      116      117
        010423      124      040      122
        010426      105      123      120
        010431      117      116      104
        010434      111      116      107
        010437      040      045      116
        010442      000
    773 010443      045      116      045  MSG5:    ASCIZ  "%N%ATERMINAL   %D3%A IS NOW RESPONDING %N"
        010446      101      124      105
        010 :       122      115      111
        010454      116      101      114
        010457      011      045      104
        010462      063      045      101
        010465      040      111      123
        010470      040      116      117
        010473      127      040      122
        010476      105      123      120
        010501      117      116      104
        010504      111      116      107
        010507      040      045      116
        010512      000
    774 010513      045      116      045  MSG6     ASCII  "%N%ATERMINAL   %D3%A DID NOT COME BACK ON LINE "
        010516      101      124      105
        010521      122      115      111
        010524      116      101      114
        010527      011      045      104
        010532      063      045      101
        010535      040      104      111
        010540      104      040      116
        010543      117      124      040
        010546      103      117      115
        010551      105      040      102
        010554      101      103      113
        010557      040      117      116
        010562      040      114      111
        010565      116      105      040
    775 010570      106      122      117           ASCIZ  "FROM DIAGNOSTIC TESTS %N"
        010573      115      040      104
        010576      111      101      107
        010601      116      117      123
        010604      124      111      103
        010607      040      124      105
        010612      123      124      123
        010615      040      045      116
        010620      000
    776 010621      045      116      045  MSG7.    ASCIZ  "%N%ATERMINAL DIAGNOSTIC ERRORS ON PASS %D5%N"
        010624      101      124      105
        010627      122      115      111
        010632      116      101      114
        010635      040      104      111
        010640      101      107      116
        010643      117      123      124
        010646      111      103      040
        010651      105      122      122
        010654      117      122      123
        010657      040      117      116
```

```
        010662  040  120  101
        010665  123  123  040
        010670  045  104  065
        010673  045  116  000
    777 010676  045  116  045  MSG8:   .ASCIZ  "%N%ASB #          TERMINAL-TYPE                FAILING TEST %N"
        010701  101  123  102
        010704  040  043  011
        010707  040  040  040
        010712  124  105  122
        010715  115  111  116
        010720  101  114  055
        010723  124  131  120
        010726  105  011  011
        010731  106  101  111
        010734  114  111  116
        010737  107  040  124
        010742  105  123  124
        010745  072  045  116
        010750  000
    778 010751  045  116  045  MSG10:  ASCIZ  "%N%AQ10 IS REJECTED--$DSW=%06%N"
        010754  101  121  111
        010757  117  040  111
        010762  123  040  122
        010765  105  112  105
        010770  103  124  105
        010773  104  055  055
        010776  044  104  123
        011001  127  075  045
        011004  117  066  045
        011007  116  000
    779 011011  045  116  045  FMT      .ASCIZ  "%N%D3%S12%D3%S28%01%01%N"
        011014  104  063  045
        011017  123  061  062
        011022  045  104  063
        011025  045  123  062
        011030  070  045  117
        011033  061  045  117
        011036  061  045  116
        011041  000
    780 011042  045  116  045  EOP.     ASCII  "%N%AEND OF PASS %D6"
        011045  101  105  116
        011050  104  040  117
        011053  106  040  120
        011056  101  123  123
        011061  040  045  104
        011064  066
    781 011065  045  101  054          ASCIZ  "%A,  TIME=%D2%A %Z2%N"
        011070  040  040  124
        011073  111  115  105
        011076  075  045  104
        011101  062  045  101
        011104  072  045  132
        011107  062  045  116
        011112  000
    782 011113  045  116  045  NACT:   ASCIZ  "%N%ANO ACTIVE TERMINALS ON THE BUS!!!!!!%N"
        011116  101  116  117
        011121  040  101  103
```

```
        011124      124      111      126
        011127      105      040      124
        011132      105      122      115
        011135      111      116      101
        011140      114      123      040
        011143      117      116      040
        011146      124      110      105
        011151      040      102      125
        011154      123      041      041
        011157      041      041      041
        011162      045      116      000
    783                                  EVEN
    784
    785
    786 011166                           LASTAD                  ,GENERATES 1'ST WORD OF FREE MEMORY
                                                                  EVEN
        011166                           L$LAST
    787                                                BLKW    2024
    788
    789
    790         000001                    END
```

```
ALUDPB  000244R      C$INLP= 000020      GETCNT  000100R      IO.RPC= 003000      L$TIM1  000016RG
AST     007634R      C$KWOF= 000042      GETERM  000132R      IO.RXP= 003400      L$UNIT  000014RG
ASTFLG  000732R      C$KWON= 000041      GLUDPB  000254R      IO.WPC= 002400      L10000  005016R
A.LULU= 000002       C$LGOF= 000027      G$EXCP= 000400       I$AU  = 000041      L10001  005022R
A.LUNA= 000004       C$LGON= 000026      G$HILI= 000002       I$CLN = 000041      L10002  005024R
A.LUNU= 000006       C$LOOP= 000100      G$LOLI= 000001       I$DU  = 000041      L10003  005026R
BIT00 = 000001       C$MANI= 000065      G$NO  = 000000       I$HRD = 000041      L10004  005026R
BIT01 = 000002       C$MPME= 000051      G$OFFS= 000400       I$INIT= 000041      L10005  006272R
BIT02 = 000004       C$MSG = 000023      G$OFSI= 000376       I$MOD = 000041      L10006  007604R
BIT03 = 000010       C$PNTB= 000014      G$PRMA= 000001       I$MSG = 000041      L10007  007676R
BIT04 = 000020       C$PNTF= 000017      G$PRMD= 000002       I$PWR = 000041      MABORT  010342R
BIT05 = 000040       C$PNTN= 000013      G$PRML= 000000       I$RPT = 000041      MAST    007662R
BIT06 = 000100       C$PNTS= 000016      G$RADA= 000140       I$SEG = 000041      MINFLG  000730R
BIT07 = 000200       C$PNTX= 000015      G$RADB= 000000       I$SFT = 000041      MSG     010261R
BIT08 = 000400       C$POIN= 000040      G$RADD= 000040       I$SRV = 000041      MSG0    007700R
BIT09 = 001000       C$PWR = 000073      G$RADF= 000200       I$SUB = 000041      MSG1    007773R
BIT10 = 002000       C$Q10 = 000377      G$RADL= 000120       I$TST = 000041      MSG10   010751R
BIT11 = 004000       C$REFG= 000064      G$RADO= 000020       LUN   = 000011      MSG2    010052R
BIT12 = 010000       C$RELA= 000052      G$RADT= 000100       L$AUT   000070RG    MSG3    010176R
BIT13 = 020000       C$RELM= 000053      G$XFER= 000004       L$CCP   000044RG    MSG4    010373R
BIT14 = 040000       C$RELO= 000037      G$YES = 000010       L$CLEA  007676RG    MSG5    010443R
BIT15 = 100000       C$REQT= 000060      G.LUBA= 000004       L$DEPO  000011RG    MSG6    010513R
BUF     000750R      C$RESE= 000040      G.LUBL= 000006       L$DEVP  000052RG    MSG7    010621R
BUFF    000740R      C$REVI= 000001      G.LUCW= 000004       L$DISP  005032RG    MSG8    010676R
C$ABRT= 000021       C$RPT = 000025      G.LUFB= 000003       L$DR    000202RG    NACT    011113R
C$ADR = 000020       C$SEFG= 000062      G.LULU= 000002       L$DRCT  000062RG    NR    = 000001
C$AU  = 000071       C$SPRI= 000046      G.LUNA= 000000       L$DRS   000064RG    OFFLIN  000770R
C$BRK = 000022       C$SVEC= 000044      G.LUNU= 000002       L$DRST  000206RG    O$APTS= 000000
C$BSEG= 000004       C$UNBU= 000035      G.TICP= 000016       L$DTP   000040RG    O$AU  = 000000
C$BSUB= 000002       C$WEFG= 000063      G.TICT= 000014       L$DUT   000072RG    O$BGNR= 000000
C$BUFF= 000034       C$WTM = 000030      G.TIDA= 000004       L$DVTY  000210RG    O$BGNS= 000000
C$CEFG= 000061       C$WTU = 000031      G.TIHR= 000006       L$EF    000024RG    O$DU  = 000000
C$CLEA= 000012       DEVTBL  000476R     G.TIMI= 000010       L$EXP1  000032RG    O$GNSW= 000000
C$CLP1= 000006       EOP     011042R     G.TIMO= 000002       L$EXP2  000034RG    O$POIN= 000001
C$CVEC= 000043       ERR     007606R     G.TISC= 000012       L$EXP3  000036RG    O$PWR = 000000
C$DCLN= 000057       ERRBLK  001170R     G.TIYR= 000000       L$HARD  004772RG    PASFG   000726R
C$DODU= 000070       ERRBRT  000262R     IBUF0   002770R      L$HPCP  000046RG    PASS    000736R
C$DRPT= 000024       ERRFLG  000734R     IBUF1   003170R      L$HPTP  000056RG    PCBADR  000704R
C$DSTI= 000033       FMT     011011R     IBUF2   003370R      L$HW    005020RG    PCBCNT  000706R
C$DU  = 000072       F$AU  = 000015      IBUF3   003570R      L$ICP   000042RG    Q10DPB  000214R
C$EDIT= 000004       F$BGN = 000040      IBUF4   003770R      L$INIT  005034RG    Q.IOAE= 000012
C$ENTI= 000032       F$CLEA= 000007      IBUF5   004170R      L$LADP  000076RG    Q IOEF= 000006
C$ERDF= 000002       F$DU  = 000016      IBUF6   004370R      L$LAST  011166RG    Q IOFN= 000002
C$ERHR= 000003       F$END = 000041      IBUF7   004570R      L$MREV  000012RG    Q IOLU= 000004
C$ERSF= 000001       F$HARD= 000004      IOST    000744R      L$NAME  000000RG    Q IOPL= 000014
C$ERSO= 000004       F$HW  = 000013      IO.AST= 000012       L$PWRU  000074RG    Q IOPR= 000007
C$ESCA= 000010       F$INIT= 000006      IO.ATT= ****** GX    L$REPP  000054RG    Q.IOSB= 000010
C$ESEG= 000005       F$MOD = 000000      IO.A1 = 000014       L$REV   000010RG    SCBADR  000700R
C$ESUB= 000003       F$MSG = 000011      IO.A2 = 000016       L$RPT   005026RG    SCBCNT  000702R
C$ETST= 000001       F$PWR = 000017      IO.A3 = 000020       L$SOFT  005024RG    STADR   000000RG
C$EXIT= 000036       F$RPT = 000012      IO.A4 = 000022       L$SPC   000030RG    START   006274R
C$GMAN= 000056       F$SEG = 000003      IO.A5 = 000024       L$SPCP  000050RG    SVCCNT= 177777
C$GMEM= 000050       F$SOFT= 000005      IO.DET= ****** GX    L$SPTP  000060RG    SVCGBL= 000001
C$GPHR= 000055       F$SRV = 000010      IO.EFN= 000006       L$STA   000066RG    SVCINS= 000001
C$GPRI= 000045       F$SUB = 000002      IO.FUN= 000002       L$SW    005026RG    SVCSTK= 177777
C$GTIM= 000066       F$SW  = 000014      IO.IST= 000010       L$TIML  000022RG    SVCSUB= 000001
C$INIT= 000011       F$TEST= 000001      IO.LUN= 000004       L$TIMU  000020RG    SVCTAG= 000001
```

```
DZKC1   MACRO M11    10-OCT-77 10.33  PAGE 16-6                                  SEQ 0053
SYMBOL TABLE

S$CTST= 000001          T$CODE= 000052          T$SAVL= 177777          T$$HW = 010001          X$ALWA= 000000
S$LSYM= 010000          T$ERRN= 000000          T$SEGL= 177777          T$$INI= 010005          X$FALS= 000040
TEMP    000710R         T$EXCP= 000000          T$SUBN= 000000          T$$RPT= 010004          X$OFFS= 000400
TEMP1   000712R         T$HILI= 000077          T$TAGL= 177777          T$$SOF= 010002          X$TRUE= 000020
TEMP2   000714R         T$LOLI= 000001          T$TAGN= 010010          T$$SW = 010003          $DSW  = ****** GX
TEMP4   000716R         T$LSYM= 010000          T$TEMP= 000007          T$$TES= 010006          $$$ARG= 000004
TEMP5   000720R         T$MCAL= 000000          T$TEST= 000001          T1      006274RG        $$$OST= 000020
TEMP6   000722R         T$NEST= 177777          T$$CLE= 010007          UNITS   000676R         $$$T1 = 000037
TEMP7   000724R         T$NSKO= 000007          T$$HAR= 010000          UNTBL   000276R         $$$T2 = 000250R
T$ARGC= 000001

  ABS   000000    000
        015236    001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:   21948 WORDS  ( 86 PAGES)
DYNAMIC MEMORY   21140 WORDS  ( 81 PAGES)
ELAPSED TIME   00 02 26
DZKC1,DZKC1=DIAGSUPER/ML,DZKC1
```