

CTP

COMMUNICATIONS TEST
MD-11-DZQCA-G

EP-DZQCA-G-DL-A

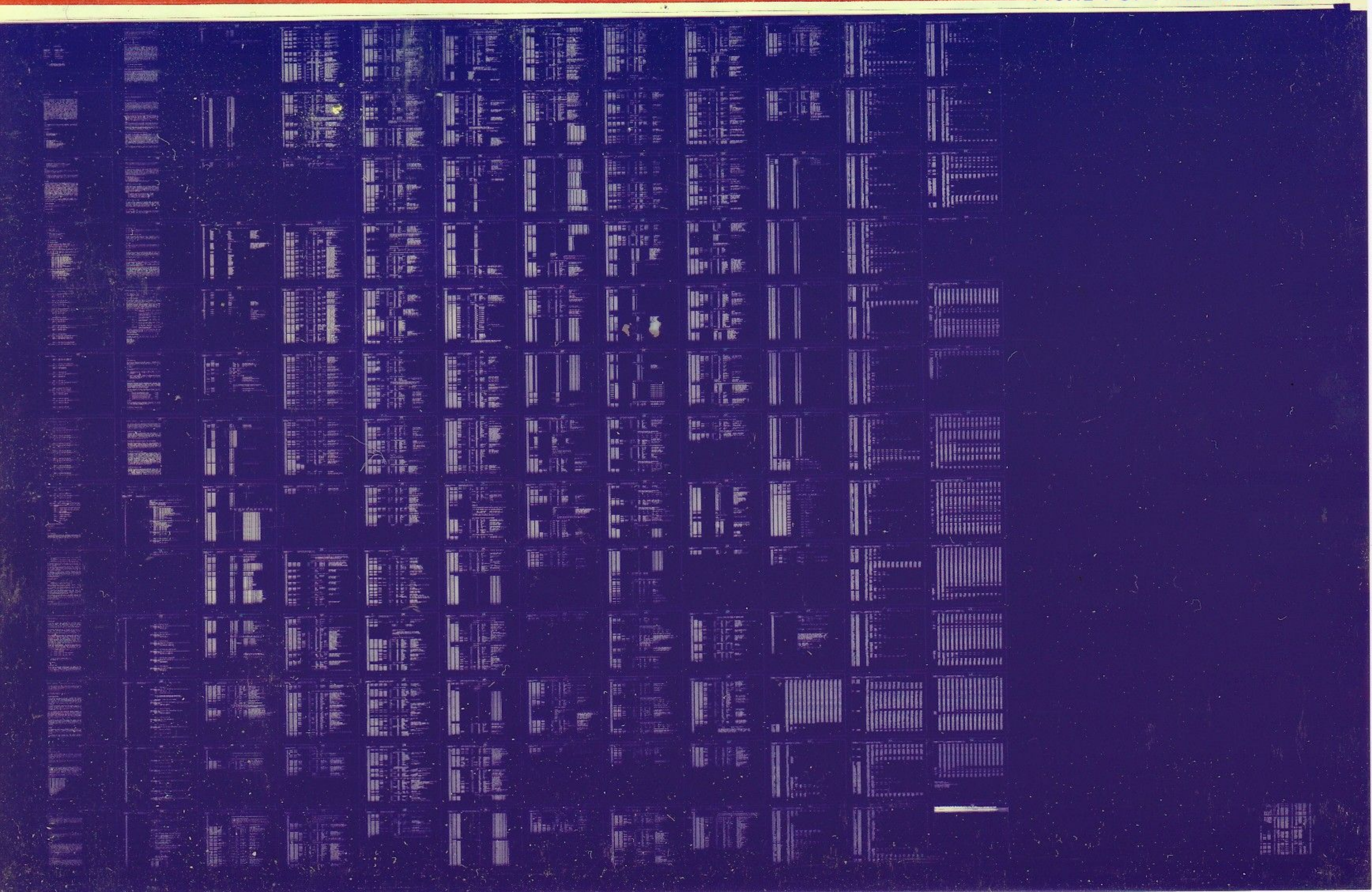
NOV 1976

COPYRIGHT © 1976

digital

FICHE 1 OF 1

MADE IN USA



IDENTIFICATION

PGODUCT CODE: MAINDEC-11-DZQCA-F
SUPERSEDES: MAINDEC-11-D9D-
PRODUCT NAME: COMMUNICATION TEST PROGRAM (CTP)
DATE: APRIL 1973
MAINTATINER: DIAGNOSTIC GROUP
AUTHOR: WALT ARMSTRONG

COPYRIGHT (C) 1972, 1973
DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS 01754

1.0 ABSTRACT

The function of this test is to detect malfunctions caused by the interaction of multiple communication devices. The test is designed to run 32 DC11 asynchronous line interfaces, 16 KL11 user teletype interfaces, 32 DP11 synchronous line interfaces, 16 DM11 asynchronous multiplexers, 16 DN11 auto calling units (64 lines), one DX11 (IBM 360 or 370 interface), 31 DL11C,D or E asynchronous interfaces, 8 DJ11 asynchronous interfaces one LP11 line printer, one TC11 DECTape, one RF11 disk, one RK11 disk, one RP11 disk, and one KL11 console teletype interface. All devices are fully interrupt driven. This allows a background program to monitor the comm devices (DC's, KL's, DP's, DM's, DL's, DH's, DJ's, DN's) and continuously check NPR data (RF, RK, RP, TC). On the first pass of this program the user must generate the hardware configuration to be tested. This is accomplished by leaving the start address in the console switches when pressing START. A systems configuration of devices currently on the system will be output on the CTY. The program then halts, permitting the user to select the device to be tested. After the initial configuring has been accomplished the restart procedure is to "LOAD ADDRESS", zero switches, "START".

1.1 NOTE:

This program is run on the original hardware configuration (for communication systems) specified by the customer key sheet.

2.0 REQUIREMENTS

2.1 Equipment

- A. PDP-11/20 System (BK core)
- B. High speed reader
- C. Console typewriter

2.1.1 Options

- A. LP11 line printer
- B. RF11, RK11 and/or RP11 disks
- C. TC11 DECTape
- D. DC11 asynchronous line (32.MAX)
- E. KL11 asynchronous interface (16.MAX)
- F. DP11 synchronous line (32.MAX)
- G. DM11 async mux (16.MAX)
- H. DN11 auto calling unit (16.MAX)
- I. DX11 IBM Front end

- J. DM11BB modem control mux (16.MAX)
- K. KG11A cycle redundancy check option (8.MAX)
- L. DL11C,D,E asynchronous interface (31.MAX)
- M. DM11 (to be added)
- N. DJ11 asynchronous interface (8.MAX)

NOTE:

PERIPHERAL OUTPUT, such as VT05 and VT06 display and LA30 hard copy is "not" meaningful for other than the line printer or local console TTY.

NOTE:

32 DP11's at 3KHZ (maintenance mode) requires in excess of 100% of the processor time. To run 32 simultaneously the maintenance clocks must be slowed by adding a capacitor to the maint. clk. cap.

2.2 Storage

The core requirements of this program is a function of the hardware under test. It consists of a fixed core requirement (location 000000 thru LINKER and a variable core requirement linker and up. This variable segment is a linker area to point unique-vectorized multiple devices to a common interrupt service routine (ISR). Therefore the size of the linker area is a function of the number of devices under test. The linker area requires 3 words per vector followed by 400 bytes per DM11. The 400 byte DM11 buffers must start on a modulo 400 boundary. The DX11 requires 2000(8) bytes of table storage. This storage area must begin on a modulo (2000) boundary. If the DX11 is run online an additional (10)(420) octal bytes are required for tumble table trace tables and control functions. After the DX11 buffers, the rest of unused memory is filled with a special memory exercising routine.

Storage requirements = location 000000 to LINKER plus 3 words per vector plus 400 bytes per DM, plus 2000 or more bytes per DX11. see 7.1 restrictions DX11 online testing.

3.0 LOADING PROCEDURE

This program is assembled in absolute format. The ABS loader is used to load the program.

4.0 USER PROCEDURE**4.1 Initialize Peripheral Devices**

A. Line printer:

- 1) Power "ON"
- 2) Ready light on
- 3) "ON LINE" light on

B. DECTape:

- 1) Mount spare certified DECTape
- 2) Place drive in REMOTE
- 3) Select desired unit number
(when in doubt use unit 0)
(ACT11 select unit 1)

4.2

Operator Interaction

A. Load address 000200

- B. Leave start address in WSR, depress START
A systems configuration map will be output on the CTY.
To suppress further systems configuration map output on
restarts set switch register bit #13.

- C. Program will HALT and display a "1" one in the data
lights indicating switch register selection #1.

- D. Set desired switch register option. Switch in up
position is a "1" and will cause the corresponding
device to be tested.

DATA LIGHTS = 1, GENERAL TEST CONFIGURATION *****

SW00 = 1, TEST NON-COMM DEVICES
 SW01 = 1, TEST DC11 ASYNC LINE UNIT
 SW02 = 1, TEST KL11 MULTI-USER TTY'S
 SW03 = 1, TEST DP11 SYNC LINE UNITS
 SW04 = 1, TEST DM11A ASYNC MUX
 SW05 = 1, TEST DN11 AUTO CALLING UNITS
 SW06 = 1, TEST DM11BB MODEM CONTROL MULTIPLEXERS
 SW07 = 1, TEST KG11A CYLIC REDUNDANCY OPTION
 SW08 = 1, TEST DX11
 SW09 = 1, TEST DL11C,D,E
 SW10 = 1, TEST DH11 (TO BE ADDED)
 SW11 = 1, TEST DJ11

DATA LIGHTS = 2, NON-COMM TEST CONFIGURATION *****

NOTE: SW00 = 1, TEST CTY CONSOL TELETYPE
 DYNAMIC SWITCH SETTING
 SW13 MUST BE SELECTED IE = 1
 TO TEST CTY (CONSOLE TTY) READER
 SW01 = 1, TEST LP11 LINE PRINTER
 SW02 = 1, TEST TC11 DECTAPE (SEE "NOTE:" AT
 9.8.1)
 SW03 = 1, TEST RF11 DISK
 SW04 = 1, TEST RK11 DISK
 SW05 = 1, TEST RP11 DISK

DATA LIGHTS = 3, DC11 TEST CONFIGURATION *****

SW00 = 1, TEST DC LINE 0
SW01 = 1, TEST DC LINE 1

:

SW15 = 1, TEST DC LINE 15

DATA LIGHTS = 4, DC11 TEST CONFIGURATION (CONTINUED)

SW00 = 1, TEST DC LINE 16
SW01 = 1, TEST DC LINE 17

:

SW15 = 1, TEST DC LINE 31

DATA LIGHTS = 5, KL11 TEST CONFIGURATION *****

SW00 = 1, TEST KL LINE 0
SW01 = 1, TEST KL LINE 1

:

SW15 = 1, TEST KL LINE 15

DATA LIGHTS = 6, DP11 TEST CONFIGURATION *****

SW00 = 1, TEST DP LINE 0
SW01 = 1, TEST DP LINE 1

:

SW15 = 1, TEST DP LINE 15

DATA LIGHTS = 7, DP11 TEST CONFIGURATION (CONTINUED)

SW00 = 1, TEST DP LINE 16
SW01 = 1, TEST DP LINE 17

:

SW15 = 1, TEST DP LINE 31

DATA LIGHTS = 10, DM11A TEST CONFIGURATION *****

SW00 = 1, TEST DM11A #0 (LINES 0-15)
SW01 = 1, TEST DM11A #1 (LINES 16-31)

:

SW15 = 1, TEST DM11A #15 (LINES 240-255)

DATA LIGHTS = 11, DN11 TEST CONFIGURATION *****

SW00 = 1, TEST DN11 #0 (LINES 0-3)
SW01 = 1, TEST DN11 #1 (LINES 4-7)

.

SW15 = 1, TEST DN11 #15 (LINES 60-63)

DATA LIGHTS = 12, DM11BB TEST CONFIGURATION *****

SW00 = 1, TEST DM11BB #0
SW01 = 1, TEST DM11BB #1

.

SW15 = 1, TEST DM11B #15

DATA LIGHTS = 13, KG11A TEST CONFIGURATION *****

SW00 = 1, TEST KG11 #0
SW01 = 1, TEST KG11 #1

.

SW07 = 1, TEST KG11 #7

DATA LIGHTS = 14, DX11 TEST CONFIGURATION *****

SW00 = 1, TEST DX11 #0

.

SW08 = 1, ONLINE TEST OF DX11 #0

DATA LIGHTS = 177400, DX11 #0 CU ADDRESS

DATA LIGHTS = 15, DL11C,D,E TEST CONFIGURATION *****

SW00 = 1, TEST DL11 LINE #0
SW01 = 1, TEST DL11 LINE #1

.

SW15 = 1, TEST DL11 LINE #15

DATA LIGHTS = 16, DL11C,D,E TEST CONFIGURATION(CONTINUED)

SW00 = 1, TEST DL11 LINE #16
SW01 = 1, TEST DL11 LINE #17

.

SW14 = 1, TEST DL11 LINE #30

DATA LIGHTS = 200, DJ11 TEST CONFIGURATION


```
SW00 = 1, TEST DJ11 #0
:
SW07 = 1, TEST DJ11 #7
DATA LIGHTS = 20, DJ11 #0 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #0 LINE #0
:
SW15 = 1, TEST DJ11 #0 LINE #15
DATA LIGHTS = 21, DJ11 #1 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #1 LINE #0
:
SW15 = 1, TEST DJ11 #1 LINE #15
DATA LIGHTS = 22, DJ11 #2 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #2 LINE #0
:
SW15 = 1, TEST DJ11 #2 LINE #15
DATA LIGHTS = 23, DJ11 #3 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #3 LINE #0
:
SW15 = 1, TEST DJ11 #3 LINE #15
DATA LIGHTS = 24, DJ11 #4 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #4 LINE #0
:
SW15 = 1, TEST DJ11 #4 LINE #15
DATA LIGHTS = 25, DJ11 #5 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #5 LINE #0
:
SW15 = 1, TEST DJ11 #5 LINE #15
DATA LIGHTS = 26, DJ11 #6 TEST CONFIGURATION *****
```



```
SW00 = 1, TEST DJ11 #6 LINE #0
:
SW15 = 1, TEST DJ11 #6 LINE #15
DATA LIGHTS = 27, DJ11 #7 TEST CONFIGURATION *****
SW00 = 1, TEST DJ11 #7 LINE #0.
:
SW15 = 1, TEST DJ11 #7 LINE #15
DATA LIGHTS = 177777, DYNAMIC SWITCH SETTING *****
```

NOTE: SW09 SELECTION MAY RESULT IN DEVICE NOT RUNNING REPORTS

```
SW08 = 1, INHIBIT MEMORY TEST WORST CASE NOISE
SW09 = 1, INHIBIT MEMORY TEST FAST READ/WRITE
SW10 = 1, BELL ON ERROR
      0, BELL ON PASS COMPLETED
SW11 = 1, INHIBIT ITERATIONS
SW13 = 1, INHIBIT ERROR TYPEOUT
SW14 = 1, SCOPE LOOP
SW15 = 1, HALT ON ERROR !
```

5.0 PROGRAM AND/OR OPERATOR ACTION

5.1 Normal HALTS

See Section 4.2 operator interaction

6.0 ERRORS

6.1 Error Reporting

The error reporting format was developed specifically for communication devices. The format assumes that R1 contains the line number (e.g. DC11 #1, DM11 #3 etc.) and that R2 contains the device address. The printout is as follows:

```
PC      PS      SP
nnnnn  mmm     iiiii
```


Where:

PC "nnnnnn" is the address (within the program) of the error.
PS "mmmmm" is the contents of the processor status register.
SP "iiiiii" is the contents of the stack pointer.

The numeric typeout is not (with exception of the error PC) necessarily significant for non-communication devices, the DX11 or the DN11.

For the DX11 offline exerciser R1 contains the address of the DX input data buffer and R2 contains the input byte count. Error reporting in the online mode is done at the 360 console using FRIEND.

In addition to hard errors (errors subject to the HALT-ON-ERROR switch) CTP also has a "soft error" reporting routine that indicates that a device has not been serviced within the 8 sec to infinity window of the background monitor. The printout does not inhibit interrupts and is in the following format:

DEVICE NOT RUNNING:
"DEVICE""(n-m)"

Where:

DEVICE will be any of the following:

"ANCILLARIES", "DC11", "KL11", "DP11", "DM11A", "DN11",
"DM11B", "KG11", "DX11", "DL11", "DJ11".

And "(n-m)" will have a numeric value of from 0 thru 31 decimal.

This output is to the console terminal and will cause the terminal test routine to error. Therefore if system overload problems are anticipated the operator should not run the console terminal test routine.

When execution is under control of the ACT11 Monitor all devices reported by the test configuration processor are tested and DEVICES that do not respond due to systems overload are reported as DEVICE NOT RUNNING and those devices are shut down.

6.2

Error HALT

If console switch 15 is on the program will HALT at location HALTER following the error type out. If switch 15 is down the program will restart on error.

7.0 RESTRICTIONS

If the line printer was selected for test the restart address is 1000. If 1000 is left in the consol switches when start is pressed the program will assume a new test configuration is desired. In this way the line printer may be deselected and restart address is 200 on the next pass.

7.1 DX11 ONLINE TESTING requires a minimum of 12 Kwords of memory. For testing in 8K word memory systems, the number of devices on line parameter located at symbolic location "NONL" will be modified to equal 1, (by the program).

7.2 PDP-11 SYSTEMS using 8k words of memory and testing low speed double buffered communications devices under monitor control (i.e., ACT-11). Depending on the cpu (i.e., mainframe) or type of memory used, it is possible that background processing can be completed in less than a character transfer time of a device being tested. This is due to the fact that monitor mode execution incorporates a QUICK PASS (i.e., suppress iterations) on pass #1, for quick verification of systems. This can result in

"DEVICE NOT RUNNING REPORTS". Should this occur under test, reload CTP using dump mode on the ACT-11 daughter station. Then clear location 42 octal and initialize for test starting at 4.0 user procedure. On 8k word memory systems device not running reports can occur with the setting (i.e.,=1) of dynamic word switch register switches:

11=1 INHIBIT ITERATIONS

9=1 INHIBIT MEMORY TEST FAST READ/WRITE

8=1 INHIBIT MEMORY TEST WORST CASE NOISE

NOTE:

DEVICE NOT RUNNING REPORTS SHOULD NOT BE IGNORED BUT RESOLVED AS STATED ABOVE.

8.0 MISCELLANIOUS

8.1 Program Run Time

The run time is dependent on the amount of memory on the system and the number of devices selected to run. The minimum run time, with 8K of memory and no devices selected, is about 8 seconds. Add approximately 8 seconds for every 4K of additional memory. Each additional device takes up to about 10% additional time, until 100% of the processor is

used servicing devices, i.e. some large systems will not be able to select all available devices at the same time. A quick Verification pass may be selected by setting switch register switches #8 and #9 inhibit memory tests and switch #11 inhibit iterations.

8.2 For hardware specifications, refer to the PDP-11 Peripheral and interface handbook.

9.C PROGRAM DESCRIPTION

Following the load address and start sequence the program examines the console switches to see if the start address bit is still up. If so the program assumes that this is the first pass or a new test configuration is desired. If the start address is not in the console switch the program assumes that "no" hardware has been added or removed from the system.

For mechanical devices, such as the TTY reader, there is no automatic re-synchronization if its tape becomes out of phase with the data. It will become necessary to stop the exerciser and manually resynchronize the tape and restart the exerciser.

There is a primer area that checks the switch register to see what devices are to be initiated. The primer area sets the interrupt enable bit in the device status register, initializes the data pattern and initiates an operation to raise data flags on devices that can not initiate them themselves. Then the primer jumps to the processor test where the individual devices are serviced at the interrupt rate.

The function of this test is to drive the unibus to the highest possible level of throughput in order to detect malfunctions which might arise from device interaction. It was for this reason that the device selection options are so extensive. This program is "NOT" a PROCESSOR or DEVICE DIAGNOSTIC.

9.1 CLREVEC

This subroutine loads location 000300 through 776 with .+2 and HALT. This is a series of instruction starting at location 0, designed to detect, and isolate unexpected traps and interrupts to the trap and interrupt vector area of memory.

The principle of this routine is: The vector entrance address points to the next sequential word which contains a HALT (0000). (This location is also the status for that

vector entrance, but this has no effect on it also being the next instruction).

If a HALT occurs in the trap or interrupt vector area, register six should be examined to determine the location the program was at, when the interrupt or trap occurred. (Memory as specified by R6 contains the PC of the instruction following this instruction where the trap occurred).

Following the .+2 initialization the emulator vector and start address is initialized. Emulator vector and start address in initialized.

9.2 SYSMAP

The function of this routine is to create a map of all the devices of the system capable of being tested by this program. The program assumes that all devices have their standard address and that all communication devices follow the address and vector assignment scheme specifies by the communication memo "relative vector assignments for communication devices in PDP-11 production systems".

SYSMAP generates ten (10) maps, a one (1) indicates that that device or line exits, e.g. bit 0 up indicates line zero exits.

```

SY.DC Map of DC11's lines 0-15.
SY.DC Map of DC11's lines 16-31.
SY.KL Map of KL11 lines 0-15.
SY.DP Map of DP11 lines 0-15.
SY.DP Map of DP11 lines 16-31.
SY.DM Map of DM11 units 0-15.
SY.DN Map of DN11 units 0-15.
SY.DL Map of DL11 units 0-15.
SY.DL Map of DL11 units 16-30.
SY.DH Map of DH11 units 0-15.
SY.DJ Map of DJ11 units 0-7.
SY.ANCMAP Map of ancillary devices.
  BIT0=KL11 Console TTY
  BIT1=LP11 Line Printer
  BIT2=TC11 DECTape
  BIT3=RF11 Disk
  BIT4=RK11 Disk
  BIT5=RP11 Disk

```

9.3 SWITCH

This routine enables the operator to select the system test configuration as per 4.2

9.4 SYSGEN

This routine verifies that all devices selected for test exist. Following this validation, the routine initializes the device vectors to point to the linker area and generates the linker code. This section of code connects multiple vectors to the device interrupt service routine (ISR). The linker area also contains the line number corresponding to the vector it handles.

9.5 REPRT

This routine was intended to report the hardware and test configuration on the console TTY.

9.6 DEVICE

This routine is the entry point of the program when the switch register is zero. The function of this routine is to initialize the status registers and data files of those devices selected for test.

9.7 BCKGND

9.7.1 Monitor

When the test configuration has been selected and as corresponding devices initialized the processor priority is lowered to zero and the devices are serviced. If the ISR's do not require 100% of the processor time (as is the case with 32.DPII) the spare time is used to run a background routine. This routine monitors NPR data files (RF11, TC11) for data errors and silent devices (DC11's, DP11's) to verify that they are in fact running. The switch register is scanned once per pass in background and if SW9=0 (i.e., is reset) CORTST per 9.7.4 is invoked.

In the case of NPR data check the disk and DECTape input files are compared with a master output file.

9.7.2 CORTST (Memory Expansion)

This memory expansion is accomplished by repeating a unique test in unused memory. In the straight line code a test is made to see if extra memory is to be used. If so the memory is filled with the unique test, then the processor test jumps to the start of the expanded code. This code is executed in these memories until the end of the code is reached, then it will return to the normal processor test.

The amount of memory is determined by doing a TST (0)+ until a time out trap occurs. The value in the register at this time is considered to be high memory. 1000(8) bytes are subtracted from this value to leave a buffer at the top of memory, for ACT11 use.

A copy of the rotate byte instruction test is used as the unique test to fill memory. This test is written in position independent code.

9.8 Interrupt Service Routines

9.8.1 FENDX (TC11 Forward End Zone)

FENDZ is the first address in the DEC tape interrupt vector (214). This routine will read, in reverse block numbers until the reverse end zone is found. At this point the interrupt vector and command register are modified to read all block number in the forward direction. Each block number read is compared with the expected block number count and miscomparisons reported. When each block is found (with the exception of block 0) a block two hundred fifty six word of test data is written onto tape. After all block numbers have been read the tape is driven into the forward end zone. Here the direction is reversed and 11 block numbers are read in reverse. Starting with block 1100 through block 1 the data is read from tape and stored in a storage buffer for comparison. While the I/O routines are inactive a background program compares TC11 read data with TC11 write data and errors are reported. At this point the bell is rung and test restarted.

NOTE:

TC11 DECTAPE DEVICE ADDRESS:

ACT11 = UNIT #1 otherwise UNIT #0 is tested.

9.8.2 DSKVEC (RF11 Disk Service Routine)

This routine is entered from the RF11 interrupt vector, the first section of this routine determines which disk function (write, write check, or read) to perform. Each function is executed in blocks of two hundred and fifty six words. When function has been executed through out the entire disk area the next function is selected and executed in the same manner two data buffers are involved in the disk exerciser, The first (BUFF*) is a two hundred and fifty six word block and serves as output data for RF11 disk and TC11 DECTape. The data file is a symmetrical pattern of N,-N where N=1 through 128, followed by N-1,N where n=128 through 1. The second buffer (INBERF:) is a storage buffer for the disk read function. While the I/O interrupt routines are

inactive a background program continuously compares the input buffer with the output buffer.

9.8.3 RKSTART (RK-11 Disk)

This routine performs a write and a write check of the disk. The data that is written on the disk is part of test program code that is never modified. This segment of memory is written in contiguous block thru the disk memory. After the total disk(s) has been written, a write check is used to verify that the data has been written correctly on the disk. Note that no "DATO" are used in exercising the disk (data is not transferred into memory). There is a location in the program that if modified will allow exercising up to four disks.

9.8.4 LPVECT (Line Printer Service Routine)

The LP11 line printer service routine is entered through an interrupt vector at location 200. A wedge is formed by printing a continuous set of sixty four characters (40 through 137). The first line prints a full row of eighty columns. Then each successive line prints one less column until zero is reached and the wedge restarts.

NOTE:

That on successive lines of printer output that the previous lines first character is discarded. This causes the printed text (output) to rotate to the left one character per line.

Because the wedge does not contain a multiple of sixty four characters, sixty four different wedges are printed. If the line printer has a capacity of ninety six characters, location LPSIZE should be changed accordingly.

9.8.5 DC.XMT DC11 Transmitter Service Routine

Each line of the DC11 has separate vectors for the transmitter and receiver. These vectors point to a unique instruction in the linker area. This instruction in a JSR register 5 to the ISR and is followed by the line number. Upon entry to the ISR registers R1 and 2 are saved and the line number is fetched. The line number scaled to form an index to the data for that line and scaled again to form an index to the device status register. The data generated by the transmitter ISR is a binary count pattern 0-377. Since the transmitter is operating in the 5 bits/character mode this pattern appears to the receiver as four patterns of 0-37.

The 5 bit mode was selected to give the highest possible rate of interrupts. This scheme also verifies that the 3

MSB get stripped and that there is no cross talk between bits. The ISR transmits one character per entry and return to the mainline by incrementing R6 and executing an RTI.

9.8.6 DC.RCV DC11 Receiver Service Routine

The entry and exit scheme for the receiver ISR is identical to that of the transmitter ISR. In this routine the done bit is tested to verify that it is a legitimate interrupt. Bit 8 of the receiver status is set indicating to the background monitor that the line has been serviced and a comparison is made between the contents of the receiver data buffer and the expected data (0-37).

Because the DC11 is:

1. Operating in maintenance mode
2. Not double buffered
3. An asynchronous device

Data overrun should never occur as a result of processor time consumed by higher priority devices.

9.8.7 KL.XMT KL11 Transmitter ISR

KL.XMT services all user KL11's up to sixteen. Because the device and vector addresses of the console KL11 are not contiguous with the user KL11's a separate routine handles the CTY. KL.XMT is entered through the linker area in order to present a line number to the ISR if interrupt enable and done are set the ISR transmits a character in a binary count pattern (0-147), and exits.

NOTE:

The receiver routine is set up to run in the maintenance mode. This means that is an ASR33 is to be used there must not be any tape in the reader. If the reader is to be tested location KLXSET must be patched to 100.

9.8.8 KL.RCV KL11 Receiver ISR

The entry and exit scheme of the receiver ISR is identical to that of the transmitter. Receiver done and interrupt enable are tested prior to the data check to detect false interrupts. The data test first checks for leader. If the data is not leader it must either be a bell (207) or the correct data in the binary pattern (0-147)

9.8.9 Console KL11 ISR's

The console IRS are entered directly from the console vectors (60-66), and the exit is by RTI data handling is identical to that of the KL11 users ISR.

9.8.10 DP11 Receiver ISR

The DP11 receiver ISR goes active on 4 sync's and receives and checks a binary count pattern 0-377 then kills active and waits to be resynced.

9.8.11 DP.XMT DP11 Transmitter ISR

The DP transmitter ISR transmits 4 sync characters and a binary count pattern 0-377. This process continues indefinitely unless the system overhead gets too heavy. If this happens RCV O'RUN will set and the transmitter will stop and resync.

9.8.12 DN11 ISR

The DN11 routines first transmits an all ones character to each DM receiver to determine the character length of each DM. Then a binary count is transmitted in proportion to the character length.

9.8.13 DN11 ISR

The DN routine utilizes the maintenance mode to test the interface logic. After the interrupt has been served, a software flag is set which signals the background program to transmit the next data when it is ready. This routine only assumes one line per DN11. If an additional line is there, but doesn't work, it may never be detected.

9.8.14 DX11 Offline ISR

The function of the offline DX11 code is to create NPR's at a rate similar to that which would occur while operating online. To accomplish this end a fast Service-In/Service-Out sequence is enabled by setting the SOSIEN (SRVO-SRVI ENABLE) flop.

Following all the table initialization the 360 simulator out lines are loaded with a test data pattern (125) and the byte count and destination address are set. The DX function bits are set to do a 360 WRITE (from the 360 into PDP memory) and the SOSIEN is then set. This causes a fast SERVICE-IN/SERVICE-OUT sequence to take place until the byte count goes to zero. At that time the DONE light is set and the DX11 interrupts to the offline interrupt service routine DX.ISR. In this routine the received data buffer is checked for correct data and then is cleared and the sequence restarted.

Some errors that have been typical are Data Lates on the

RK03 disk and "holes" in either DX or RK data buffers. These "holes" were caused by noise on the M7B21's which caused BUS BUSY to drop.

9.8.15 DX11 Online Responder

The online DX11 module of CTP is a responder to a 360 program that allows the PDP-11 system to be exercised in conjunction with the 360 system.

The 360 channel to the DX11 is usually driven by an IBM diagnostic program called FRIEND, which generates an arbitrary CCW string for a particular device (see DX11-B Maintenance Manual for FRIEND operator procedures). The DX11 CTP module recognizes eight 360 addresses, and all 256 commands, and responds to them as follows:

Addresses X0 - X3: Selections accepted (see command responses).

X4 - X7: Selections rejected with unit-check, except for sense, which receives Intervention Required.

Commands Read : Clears Sense byte and transfers 256 byte buffer to 360.

Write : Clears Sense byte and transfers 256 bytes from 360.

Sense : Returns one byte Sense byte to 360 (Sense byte not cleared).

NOP : No operation - Sense byte not altered.

Others : Rejected via Unit-Check. Sense byte set to Command-Reject.

The effect of these operations is that addresses X0 - X3 appear "online" to the 360, and X4 - X7 appear "offline". Also, for the online addresses, Read, Write, Sense, and NOP are legal commands.

FRIEND COMMAND STRING

In order to exercise the DX11 online to the 360, the following command string should be typed to FRIEND:

```
DEV = XXX  
WRITE FROM SA  
DATA = 256 X 11  
READ INTO SB  
DL = 256  
COMPARE SA, SB
```


GO

ERROR DETECTION

No errors are detected by the CTP module. FRIEND detects data errors caused by the DX11 and also the failure of the DX11 to respond (which probably means that the DX11 failed to interrupt).

CTP SETUP

Several program constants may require alteration before CTP is started.

DX11 interrupt priority - default is 200 for BR4. In DX11 interrupts at BR5 or BR6, this constant must be changed.

DX11 control unit address - default is hexadecimal 10 (octal 20). This value should be changed to reflect the jumpered 360 address of the DX11 (see DX11-B Maintenance Manual).

9.8.16 DL11A,B,C,D,E

Since each DL11 unit has four registers each requires four addresses. Address space assignments for the DL11A and B are the same as for the KL11; that is, unit 0 occupies addresses 777560-777566, and units 1-15 occupy from 776500 through 776676. For the DL11C,D, and E unit 0 will have address 775610 thru unit 31 at 776170.

The four registers and their addresses are listed for DL11 unit 0, where xx is 756 for DL11A and B, and 561 for DL11C,D,E.

- | | |
|--|--------|
| 1. RECEIVER STATUS REGISTER (RCSR) | 77XXX0 |
| 2. RECEIVER DATA BUFFER REGISTER (RBUF) | 77XXX2 |
| 3. TRANSMITTER STATUS REGISTER (XCSR) | 77XXX4 |
| 4. TRANSMITTER DATA BUFFER REGISTER (XBUF) | 77XXX6 |

DL11A,B

Are KL compatible devices with 8 bit character, no parity and two stop bit. If one is used for the console teletype it is assigned VECTOR ADDRESS 60 and 64.

9.8.17 DL.XMT DL11 Transmitter ISR

DL.XMT services all user DL11's up to thirty one. Because

the device and vector addresses of the console DL11A and B are not contiguous with DL11C,D,E's the DL11A and B's are treated as KL11's and a separate routine handles them. DL.XMT is entered through the linker area. Transmitter ready is verified on each entry and a binary preprocessed pattern of (0-37) is transmitted.

9.8.18 DL.RCV DL11 Receiver ISR

The DL11 is tested in maintenance mode and entry in the receiver ISR is identical to that of the transmitter. The receiver done is tested prior to the data check to detect false interrupts. Each entry into the receiver ISR the DL DEVICE FLAG bit is set in the systems RUN MAP to indicate that the device has been serviced.

9.8.19 DJ.XMT/DJ.RCV Transmitter/Receiver Service Routines

Each DJ11 has a separate Vector for the transmitter and receiver. These vectors point to a UNIQUE instruction in the LINKER AREA. This instruction is a JSR register 5 to the associated ISR and is followed by the device number. On entry into the ISR the device number is fetched and scaled to form a device base address used by the ISR to service that device.

9.8.20 DJ.XMT DJ11 Transmitter ISR

On initial entry bit 15 XMIT READY is tested to detect false interrupts. The line number which requires service is extracted from the transmitter buffer register and used as an index to retrieve its associated character for transmission. The associated character entry is transmitted and then incremented to the next value. A 5 level binary preprocessed data pattern of 0-37 is transmitted. Expanded character level and error report capability has been included, the method of implementation can be found in the form of annotation within the DEVICE ISR program listing. The device CSR register is tested and further line activity is processed as required. A running tally of transmitted characters is recorded in location DJ.XTLY.

9.8.21 DJ.RCV DJ11 Receiver ISR

The DJ11 is tested in maintenance mode and entry into the receiver ISR is identical to that of the transmitter. The RECEIVER DONE bit 7 is tested to detect false interrupts. The received character word is verified at a 16 bit level. i.e., OVERRUN, FRAMING and PARITY errors will result in error display processing. Each entry into the receiver ISR the DJ DEVICE FLAG bit is set in the system RUN MAP to

indicate that the device has been serviced.

10.0 FLOW CHART

11.0 LISTING
MAINDEC-11-DZQCA-G
DZQCAG.P11

COMMUNICATION TEST PROGRAM (CTP)
SWITCH SETTINGS

MACY11 27(732) 17-SEP-76 15:12 PAGE 2

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

```

.TITLE MAINDEC-11-DZQCA-G COMMUNICATION TEST PROGRAM (CTP)
.ABS
.ENABLE TIM
;COPYRIGHT 1972, 1973 BY DIGITAL EQUIPMENT CORPORATION
;MAYNARD, MASSACHUSETTS 01754
;PROGRAMMER: WALT ARMSTRONG
;DATA LIGHTS = 1, GENERAL TEST CONFIGURATION *****
;HALT ADDRESS 3114
;SW00 = 1, TEST NON-COMM DEVICES
;SW01 = 1, TEST DC11 ASYNC LINE UNIT
;SW02 = 1, TEST KL11 MULTI-USER TTY'S
;SW03 = 1, TEST DP11 SYNC LINE UNITS
;SW04 = 1, TEST DM11A ASYNC MUX
;SW05 = 1, TEST DN11 AUTO CALLING UNITS
;SW06 = 1, TEST DM11BB MODEM CONTROL MULTIPLEXERS
;SW07 = 1, TEST KG11A CYCLIC REDUNDANCY CHECK OPTIONS
;SW08 = 1, TEST DX11
;SW09 = 1, TEST DL11C, D, E
;SW10 = 1, TEST DH11 (TO BE ADDED)
;SW11 = 1, TEST DJ11
;DATA LIGHTS = 2, NON-COMM TEST CONFIGURATION *****
;HALT ADDRESS 3140
;SW00 = 1, TEST CTY CONSOL TELETYPE
;NOTE: DYNAMIC SWITCH SETTING
;SW13 MUST BE SELECTED IE = 1
;TO TEST CTY (CONSOLE TTY) READER
;SW01 = 1, TEST LP11 LINE PRINTER
;SW02 = 1, TEST TC11 DECTAPE
;SW03 = 1, TEST RF11 DISK
;SW04 = 1, TEST RK11 DISK
;SW05 = 1, TEST RP11 DISK
;DATA LIGHTS = 3, DC11 TEST CONFIGURATION *****
;HALT ADDRESS 3164
;SW00 = 1, TEST DC LINE 0
;SW01 = 1, TEST DC LINE 1
;
;
;SW15 = 1, TEST DC LINE 15
;DATA LIGHTS = 4, DC11 TEST CONFIGURATION (CONTINUED)
;HALT ADDRESS 3206
;SW00 = 1, TEST DC LINE 16
;SW01 = 1, TEST DC LINE 17
;
;

```

51
52
53
54
55
56

J02

: SWIS = 1, TEST DC LINE 31
: DATA LIGHTS = 5, KL11 TEST CONFIGURATION *****
: HALT ADDRESS 3232
: SW00 = 1, TEST KL LINE 0

57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112

```

:      SW01 = 1, TEST KL LINE 1
:      .
:
:      SW15 = 1, TEST KL LINE 15
:
: DATA LIGHTS = 6, DP11 TEST CONFIGURATION *****
: HALT ADDRESS 3256
:      SW00 = 1, TEST DP LINE 0
:      SW01 = 1, TEST DP LINE 1
:      .
:
:      SW15 = 1, TEST DP LINE 15
:
: DATA LIGHTS = 7, DP11 TEST CONFIGURATION (CONTINUED)
: HALT ADDRESS 3300
:      SW00 = 1, TEST DP LINE 16
:      SW01 = 1, TEST DP LINE 17
:      .
:
:      SW15 = 1, TEST DP LINE 31
:
: DATA LIGHTS = 10, DM11A TEST CONFIGURATION *****
: HALT ADDRESS 3324
:      SW00 = 1, TEST DM11A #0 (LINES 0-15)
:      SW01 = 1, TEST DM11A #1 (LINES 16-31)
:      .
:
:      SW15 = 1, TEST DM11A #15 (LINES 240-255)
:
: DATA LIGHTS = 11, DN11 TEST CONFIGURATION *****
: HALT ADDRESS 3350
:      SW00 = 1, TEST DN11 #0 (LINES 0-3)
:      SW01 = 1, TEST DN11 #1 (LINES 4-7)
:      .
:
:      SW15 = 1, TEST DN11 #15 (LINES 60-63)
:
: DATA LIGHTS = 12, DM11BB TEST CONFIGURATION *****
: HALT ADDRESS 3374
:      SW00 = 1, TEST DM11BB #0
:      SW01 = 1, TEST DM11BB #1
:      .
:
:      SW15 = 1, TEST DM11BB #15
:
: DATA LIGHTS = 13, KG11A TEST CONFIGURATION *****
: HALT ADDRESS 3420
:      SW00 = 1, TEST KG11 #0
:      SW01 = 1, TEST KG11 #1
:      .

```

113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168

```

:
:
:      SW07 = 1, TEST KG11 #7
:
: DATA LIGHTS = 14, DX11 TEST CONFIGURATION *****
: HALT ADDRESS 3444
:      SW00 = 1, TEST DX11 #0
:
:      SW08 = 1, ONLINE TEST OF DX11 #0
:
: NOTE:
:      FOR DX11 ONLINE TEST SELECTION SEE RESTRICTIONS
:      7.1 LISTING ANNOTATION AT BEGINING OF THIS MAINDEC
:
:
:      DATA LIGHTS = 177400, DX11 #0 CU ADDRESS
: HALT ADDRESS 3512
:
: DATA LIGHTS = 15, DL11C,D,E TEST CONFIGURATION *****
: HALT ADDRESS 3550
:      SW00 = 1, TEST DL11 LINE #0
:      SW01 = 1, TEST DL11 LINE #1
:
:
:
:      SW15 = 1, TEST DL11 LINE #15
:
: DATA LIGHTS = 16, DL11C,D,E TEST CONFIGURATION (CONTINUED)
: HALT ADDRESS 3572
:      SW00 = 1, TEST DL11 LINE #16
:      SW01 = 1, TEST DL11 LINE #17
:
:
:
:      SW14 = 1, TEST DL11 LINE #30
:
: DATA LIGHTS = 200, DJ11 TEST CONFIGURATION
: HALT ADDRESS 3620
:      SW00 = 1, TEST DJ11 #0
:
:
:
:      SW07 = 1, TEST DJ11 #7
:
: DATA LIGHTS = 20, DJ11 #0 LINE TEST CONFIGURATION SELECTION *****
: HALT ADDRESS 3644
:
:      SW00 = 1, TEST DJ11 #0 LINE #0
:
:
:
:      SW15 = 1, TEST DJ11 #0 LINE #15
:
:
: DATA LIGHTS = 21, DJ11 #1 LINE TEST CONFIGURATION SELECTION *****
```



```
169 ;HALT ADDRESS 3670
170 ;
171 ; SW00 = 1,TEST DJ11 #1 LINE #0
172 ;
173 ;
174 ;
175 ; SW15 = 1,TEST DJ11 #1 LINE #15
176 ;
177 ;
178 ;
179 ;DATA LIGHTS = 22, DJ11 #2 LINE TEST CONFIGURATION SELECTION *****
180 ;HALT ADDRESS 3714
181 ;
182 ; SW00 = 1,TEST DJ11 #2 LINE #0
183 ;
184 ;
185 ;
186 ; SW15 = 1,TEST DJ11 #2 LINE #15
187 ;
188 ;
189 ;
190 ;DATA LIGHTS = 23, DJ11 #3 LINE TEST CONFIGURATION SELECTION *****
191 ;HALT ADDRESS 3740
192 ;
193 ; SW00 = 1,TEST DJ11 #3 LINE #0
194 ;
195 ;
196 ;
197 ; SW15 = 1,TEST DJ11 #3 LINE #15
198 ;
199 ;
200 ;
201 ;DATA LIGHTS = 24, DJ11 #4 LINE TEST CONFIGURATION SELECTION *****
202 ;HALT ADDRESS 3764
203 ;
204 ; SW00 = 1,TEST DJ11 #4 LINE #0
205 ;
206 ;
207 ;
208 ; SW15 = 1,TEST DJ11 #4 LINE #15
209 ;
210 ;
211 ;
212 ;DATA LIGHTS = 25, DJ11 #5 LINE TEST CONFIGURATION SELECTION *****
213 ;HALT ADDRESS 4010
214 ;
215 ; SW00 = 1,TEST DJ11 #5 LINE #0
216 ;
217 ;
218 ;
219 ; SW15 = 1,TEST DJ11 #5 LINE #15
220 ;DATA LIGHTS = 26, DJ11 #6 LINE TEST CONFIGURATION SELECTION *****
221 ;HALT ADDRESS 4034
222 ; SW00 = 1,TEST DJ11 #6 LINE #0
223 ;
224 ;
```

225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280

```

; SW15 = 1,TEST DJ11 #6 LINE #15

;DATA LIGHTS = 27, DJ11 #7 LINE TEST CONFIGURATION SELECTION *****
;HALT ADDRESS 4060

; SW00 = 1,TEST DJ11 #7 LINE #0
;
;
; SW15 = 1,TEST DJ11 #7 LINE #15

;DATA LIGHTS = 177777, DYNAMIC SWITCH SETTING *****
;HALT ADDRESS 4176

;NOTE:
; SW08 = 1 INHIBIT MEMORY TEST WORST CASE NOISE
;
; SW09 = 1, INHIBIT MEMORY TEST FAST READ WRITE
; SW10 = 1, BELL ON ERROR
;       0, BELL ON PASS COMPLETED
;NOTE: SW11 SELECTION MAY RESULT IN DEVICE NOT RUNNING REPORT
;
; SW11 = 1, INHIBIT ITERATIONS
;
;NOTE: SW13 = 1, INHIBIT ERROR TYPEOUT
;       SW13 MUST BE SELECTED IE = 1
;       TO TEST CTY (CONSOLE TTY)
;
;ACT11: SW13 = 1 INHIBIT SHUTDOWN OF DEVICE(S) NOT RUNNING
;
; THE FOLLOWING CTY (CONSOLE TTY) REPORT WILL
; CONSTITUTE A PASS WHEN RUN UNDER THE ACT11
; MONITOR:
;
; DZQCA-G TESTED WITH
; SYSTEM CONFIGURATION
; ETC.

; SW14 = 1, SCOPE LOOP
; SW15 = 1, HALT ON ERROR !

;DEVICE FIRST #VECTOR VECTOR
;TYPE ADDRESS WORDS OFFSET
;DC11 174000 4 10
;KL11 176500 4 10
;DP11 174770 4 10
;DM11 175000 4 10
;DN11 175200 2 10
;DMB11 170500 2 4
;KG11 170700 0 0
```


295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350

000001
000002
000004
000010
000020
000040
000100
000200
000400
001000
002000
004000
010000
020000
040000
100000
000001
000002
000004
000010
000020
000040
000100
000200
000400
001000
002000
004000
010000
020000
040000
100000
000400
000200
000240
000300
000340
000000
000001
000002
000003
000004
000005
000006
000007
000005
000006
000007
177774
177776
177570
177570
104400
000000

:EQUALITIES
BIT0 = 000001
BIT1 = 000002
BIT2 = 000004
BIT3 = 000010
BIT4 = 000020
BIT5 = 000040
BIT6 = 000100
BIT7 = 000200
BIT8 = 000400
BIT9 = 001000
BIT10 = 002000
BIT11 = 004000
BIT12 = 010000
BIT13 = 020000
BIT14 = 040000
BIT15 = 100000
SW00 = 000001
SW01 = 000002
SW02 = 000004
SW03 = 000010
SW04 = 000020
SW05 = 000040
SW06 = 000100
SW07 = 000200
SW08 = 000400
SW09 = 001000
SW10 = 002000
SW11 = 004000
SW12 = 010000
SW13 = 020000
SW14 = 040000
SW15 = 100000
SW8 = SW08
LEVEL4 = 200
LEVEL5 = 240
LEVEL6 = 300
LEVEL7 = 340
R0 = %0
R1 = %1
R2 = %2
R3 = %3
R4 = %4
R5 = %5
R6 = %6
R7 = %7
TTY = %5
SP = %6
PC = %7
SLR = 177774
PS = 177776
SWR = 177570
DISPLAY = SWR
SCOPE = TRAP
HERE = 0

MAINDEC-11-DZQCA-G
DZQCA.G.P11

COMMUNICATION TEST PROGRAM (CTP)
DEFINITIONS

351 000240
352 000000
353 104000
354 000004
355 166000
356 000207

NUP = 240
OPEN = 0
HLT = EMT
TYPE = 10T
.BIT = 166000
BELL = 207

;ERROR REPORT ROUTINE

```

357
358
359
360
361      000000      REQ=0
362      000002      LREQ=2
363      000004      DEVSTA=4
364      000006      ONLF=6
365      000010      SNS=10
366      000012      BUFA=12
367      000014      BUFC=14
368      000016      STS=16
369      000020      DXBF=20
370      000420      DCTZ=20+256.
371
372      ;DEVICE CONTROL TABLE ENTRY DEFNS (DCT)
373
374
375
376
377
378
379
380
381
382
383      000002
384      000003
385
386      000001
387      001000
388      020000
389      000100
390      000200
391      000005
392      000003
393      000007
394      004000
395      010000
396      000400
397      000010
398      000004
399
400      010000
401      000002
402      000100
403      001000
404      002000
405      000200
406      004000
407      000020
408      000040
409
410      000002
411      000010
412      000004

;REGISTER DEFNS FOR DX11 CODE
DCT=%0
TTP=%1
T1=%2
T2=%3
T3=%4
W=%5
STS=16
;DX DEFNS
CUART=2
CUCRT=3
GO=1
ONLINA=1000
ENDEN=20000
IE=100
DONE=200
FCTNW=5
FCTNR=3
FCTNS=7
BSYEN=4000
SMEN=10000
CUBSY=400
ISSREJ=10
CMDCHN=4
SYSRST=10000
STKSTB=2
ESEND=100
CHENDS=1000
UCHKS=2000
CHIS=200
INFDC=4000
CUDEND=20
CHDEND=40
UC=2
CE=10
DE=4

;CURRENT REQUEST
;LAST REQUEST
;DEVICE STATE
;ONLINE FLAG
;SENSE BYTE
;BUFFER ADDRESS
;BUFFER BYTE COUNT
;STATUS BYTE
;DATA BUFFER AREA

;CURRENT DCT BASE (BACKGROUND TASK)
;CURRENT TUMBLE TABLE POINTER (INTERRUPT
;WORK REGISTER (BACKGROUND)
;WORK REGISTER (BACKGROUND)
;WORK REGISTER (BACKGROUND)
;DCT BASE (INTERRUPT)
;STATUS BYTE

```


413	000100	SM=100	
414			
415	000040	CUE=40	
416	000004	SENSE=4	
417			
418	000200	CMDREJ=200	
419	000100	INTREQ=100	
420			
421		;REQUEST CODES	
422			
423	000000	RNULL=0	;NO REQUEST
424	000001	RINP=1	;REQUEST INPUT
425	000002	ROUTP=2	;REQUEST OUTPUT
426	000003	RSTS=3	;REQUEST STATUS
427	000004	RCUE=4	;REQUEST CUE
428			
429		;DEVICE STATE CODES	
430			
431	000000	IDLE=0	;IDLE
432			
433	000001	DATA=1	;DATA TRANSFER OR ES VALID
434			
435	000002	STA=2	;ES REQUIRED
436			
437	000003	CHAIN=3	;COMMAND CHAINING INDICATED
438			
439			;INTERRUPT VECTOR ADDRESS
440			
441			;INT ROUTINE
442			;OPERATING STATUS

```

443
444
445
446          000000
447
448          000020
449 000020 023450 000000
450 000024 024074 000340
451 000030 023210 000340
452 000034 023750 000340
453
454
455
456 000200 000167 001476
457
458
459 001000 000167 000676
460
461
462
463
464 001020 000000
465
466 001022 000000
467
468
469
470
471
472
473
474
475 001024 000000
476
477
478 001026 000300
479
480
481
482

```

```

;TRAP INITIALIZATION
. = 0 ;TRAP CATCHER FROM 0-776
1.8 . = 20
      IOTS, 0 ;IOT TRAP VECTOR (20)
      PDOWNS, 340 ;POWER FAIL INTERRUPT VECTOR (24)
      HLT$, 340 ;EMULATOR TRAP VECTOR (30)
      TRAP$, 340 ;TRAP VECTOR (34)

;START ADDRESS INITIALIZATION
. = 200
4.9 START: JMP BEGIN

;RESTART ADDRESS IF LINE PRINTER IS TESTED
. = 1000
4.9 RSTART: JMP BEGIN

;CTP OVERLAY LINKAGE ENTRIES
. = 1020 ;BACKGROUND OVERLAY
OVINIT: 0 ;INITIALIZATION LINKAGE POINTER
. = 1022 ;BACKGROUND OVERLAY
OVPRIM: 0 ;DEVICE PRIMER LINKAGE POINTER

;NOTE:
; CTP WILL ACCOMADATE FOREGROUND OVERLAY'S FOR DEVICES
; WHICH ARE NOT NORMALLY TESTED WITHIN CTP ITSELF.

. = 1024
CLINK: 0 ;MONITOR CONTROL ENTRY
. = 1026 ;NEXT AVAILABLE FLOATING VECTOR ADDRESS GLOBAL
FVECT: 300 ;DEFAULT = BASE

```


483
484
485 001200 001200
486 001202 000000
487 001204 000000 000000
488
489 000017
490
491 000001
492 000002
493 000004
494 000010
495 000020
496 000040
497 000100
498 000200
499 000400
500 001000
501
502 004000
503 000001
504 000002
505 000004
506 000010
507 000020
508 000040
509
510 001210 000000
511 001212 000000
512 001214 000000
513 001216 000000
514 001220 000000
515 001222 000000
516 001224 000000
517 001226 000000
518 001230 000000
519 001232 000000
520 001234 000000
521 001236 000000
522 001240 000000
523 001242 000000
524 001244 000000
525
526 001246 000000
527 001250 000000
528 001252 000000
529 001254 000000
530 001256 000000
531 001260 000000
532 001262 000000
533 001264 000000
534 001266 000000
535 001270 000000
536 001272 000000
537 001274 000000
538 001276 000000

=1200
ICNT: 0
ERRORS: 0
PASSES: 0,0
MAPSIZ = 15.
FL.ANC= 000001
FL.DC = 000002
FL.KL = 000004
FL.DP = 000010
FL.DMA= 000020
FL.DN = 000040
FL.DMB= 000100
FL.KG = 000200
FL.DX = 000400
FL.DLC= 001000
FL.DH= 002000
FL.DJ = 004000
FL.CTY= 000001
FL.LP = 000002
FL.TC = 000004
FL.RF = 000010
FL.RK = 000020
FL.RP = 000040

; ITERATION COUNT
; NUMBER OF ERRORS
; NUMBER OF PASSES

; CONSTANT TO KEEP TRACK OF MAP SIZE

SY.MAP: 0
SY.ANC: 0
SY.DC1: 0
SY.DC2: 0
SY.KL: 0
SY.DP1: 0
SY.DP2: 0
SY.DMA: 0
SY.DN: 0
SY.DMB: 0
SY.KG: 0
SY.DX: 0
SY.DL1: 0
SY.DL2: 0
SY.DJ: 0

; SYSTEM CONFIGURATION MAP

SW.MAP: 0
SW.ANC: 0
SW.DC1: 0
SW.DC2: 0
SW.KL: 0
SW.DP1: 0
SW.DP2: 0
SW.DMA: 0
SW.DN: 0
SW.DMB: 0
SW.KG: 0
SW.DX: 0
SW.DL1: 0

; DEVICES SELECTED FOR TEST MAP

539	001300	000000
540	001302	000000
541		
542		
543	001304	000000
544	001306	000000
545	001310	000000
546	001312	000000
547	001314	000000
548	001316	000000
549	001320	000000
550	001322	000000
551	001324	000000
552	001326	000000
553	001330	000000
554	001332	000000
555	001334	000000
556	001336	000000
557	001340	000000
558		
559	001342	
560		
561		
562		
563		
564	001342	000000
565	001344	000000
566	001346	000000
567	001350	000000
568	001352	000000
569	001354	000000
570	001356	000000
571	001360	000000
572	001362	000000
573	001364	000000
574	001366	000000
575	001370	000000
576	001372	000000
577	001374	000000
578	001376	000000
579	001400	000000
580	001402	000000
581	001404	000000
582	001406	000000
583	001410	000000
584	001412	000000
585	001414	000000
586	001416	000000
587	001420	000000
588	001422	000000
589	001424	000000
590	001426	000000
591	001430	000000
592	001432	000000
593	001434	000000
594	001436	000000

SW.DL2: 0
SW.DJ: 0

RU.MAP: 0
RU.ANC: 0
RU.DC1: 0
RU.DC2: 0
RU.KL: 0
RU.DP1: 0
RU.DP2: 0
RU.DMA: 0
RU.DN: 0
RU.DMB: 0
RU.KG: 0
RU.DX: 0
RU.DL1: 0
RU.DL2: 0
RU.DJ: 0

;DEVICES RUNNING MAP

MAP:
:BIT11 BIT10 BIT9 BIT8 BIT7 BIT6 BIT5 BIT4 BIT3 BIT2 BIT1
:DJ11 DH11 DL11C DX11 KG11 DM11B DN11 DM11A DP11 KL11 DC11
: : : DL11D : : : : : : : : : :
: : : DL11E : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :
: : : : : : : : : : : : : : : : : :

;MONITOR MAP FOR DEVICE UNIT #0
;MONITOR MAP FOR DEVICE UNIT #1
;MONITOR MAP FOR DEVICE UNIT #2
;MONITOR MAP FOR DEVICE UNIT #3
;MONITOR MAP FOR DEVICE UNIT #4
;MONITOR MAP FOR DEVICE UNIT #5
;MONITOR MAP FOR DEVICE UNIT #6
;MONITOR MAP FOR DEVICE UNIT #7
;MONITOR MAP FOR DEVICE UNIT #10
;MONITOR MAP FOR DEVICE UNIT #11
;MONITOR MAP FOR DEVICE UNIT #12
;MONITOR MAP FOR DEVICE UNIT #13
;MONITOR MAP FOR DEVICE UNIT #14
;MONITOR MAP FOR DEVICE UNIT #15
;MONITOR MAP FOR DEVICE UNIT #16
;MONITOR MAP FOR DEVICE UNIT #17
;MONITOR MAP FOR DEVICE UNIT #20
;MONITOR MAP FOR DEVICE UNIT #21
;MONITOR MAP FOR DEVICE UNIT #22
;MONITOR MAP FOR DEVICE UNIT #23
;MONITOR MAP FOR DEVICE UNIT #24
;MONITOR MAP FOR DEVICE UNIT #25
;MONITOR MAP FOR DEVICE UNIT #26
;MONITOR MAP FOR DEVICE UNIT #27
;MONITOR MAP FOR DEVICE UNIT #30
;MONITOR MAP FOR DEVICE UNIT #31
;MONITOR MAP FOR DEVICE UNIT #32
;MONITOR MAP FOR DEVICE UNIT #33
;MONITOR MAP FOR DEVICE UNIT #34
;MONITOR MAP FOR DEVICE UNIT #35
;MONITOR MAP FOR DEVICE UNIT #36


```

595 001440 000000          0          ;MONITOR MAP FOR DEVICE UNIT #37
596
597          ;SYSTEM MAPS OF NON-COMM FLOATING VECTOR DEVICES
598 001442 000000          SY.DR1: 0          ;MAP OF DR11A AND DR11C
599 001444 000000          SY.DR2: 0
600 001446 000000          SY.PAR: C          ;MAP OF PA611 READERS
601 001450 000000          SY.PAP: 0          ;MAP OF PA611 PUNCHES
602 001452 000000          SY.DT: 0          ;MAP OF DT11
603 001454 000000          SY.NXT: 0          ;MAP NEXT
604
605 001456 177777          SW.DJ0: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #0
606 001460 177777          SW.DJ1: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #1
607 001462 177777          SW.DJ2: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #2
608 001464 177777          SW.DJ3: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #3
609 001466 177777          SW.DJ4: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #4
610 001470 177777          SW.DJ5: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #5
611 001472 177777          SW.DJ6: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #6
612 001474 177777          SW.DJ7: 177777         ;LINE ACTIVITY SWITCHES FOR DJ11 #7
613
614 001476 000000          ENDVEC: 0          ;END OF VECTOR AREA
615 001500 031424          ENDLNK: LINKER     ;END OF LINKER AREA
616 001502 000000          ENDTAB: 0          ;END OF DM11 TABLES
617 001504 000000          ENDCOR: 0          ;END OF USEABLE CORE
618
619 001506 000000          NO.DC: 0           ;NUMBER OF DC11'S
620 001510 000000          NO.KL: 0           ;NUMBER OF KL11'S
621 001512 000000          NO.DP: 0           ;NUMBER OF DP11'S
622 001514 000000          NO.DMA: 0          ;NUMBER OF DM11A'S
623 001516 000000          NO.DN: 0           ;NUMBER OF DN11'S
624 001520 000000          NO.DMB: 0          ;NUMBER OF DM11BB'S
625 001522 000000          NO.KG: 0           ;NUMBER OF KG11'S
626 001524 000000          NO.DR: 0           ;NUMBER OF DR11'S
627 001526 000000          NO.PAR: 0          ;NUMBER OF PA611 READERS
628 001530 000000          NO.PAP: 0          ;NUMBER OF PA611 PUNCHES
629 001532 000000          NO.DT: 0           ;NUMBER OF DT11'S
630 001534 000000          NO.DX: 0           ;NUMBER OF DX11'S
631 001536 000000          NO.DLC: 0          ;NUMBER OF DL11C'S
632 001540 000000          NO.DJ: 0           ;NUMBER OF DJ11'S
633
634 001542 000040          MX.DC: 32          ;MAXIMUM NUMBER OF DC11'S ALLOWABLE
635 001544 000020          MX.KL: 16          ;MAXIMUM NUMBER OF KL11'S ALLOWABLE
636 001546 000040          MX.DP: 32          ;MAXIMUM NUMBER OF DP11'S ALLOWABLE
637 001550 000020          MX.DMA: 16         ;MAXIMUM NUMBER OF DM11A'S ALLOWABLE
638 001552 000020          MX.DN: 16         ;MAXIMUM NUMBER OF DN11'S ALLOWABLE
639 001554 000020          MX.DMB: 16         ;MAXIMUM NUMBER OF DM11BB'S ALLOWABLE
640 001556 000020          MX.KG: 16         ;MAXIMUM NUMBER OF KG11'S ALLOWABLE
641 001560 000040          MX.DR: 32         ;MAXIMUM NUMBER OF DR11'S ALLOWABLE
642 001562 000020          MX.PAR: 16        ;MAXIMUM NUMBER OF PA611 READERS ALLOWABLE
643 001564 000020          MX.PAP: 16        ;MAXIMUM NUMBER OF PA611 PUNCHES ALLOWABLE
644 001566 000010          MX.DT: 8          ;MAXIMUM NUMBER OF DT11'S ALLOWABLE
645 001570 000004          MX.DX: 4          ;MAXIMUM NUMBER OF DX11'S ALLOWABLE
646 001572 000037          MX.DLC: 31        ;MAXIMUM NUMBER OF DL11'S ALLOWABLE
647 001574 000010          MX.DJ: 8          ;MAXIMUM NUMBER OF DJ11'S ALLOWABLE
648 001576 000000          MX.NEXT: 0        ;ADDITIONAL ENTRIES

```

649
650 001600 174000
651 001602 176500
652 001604 174770
653 001606 175000
654 001610 175200
655 001612 170500
656 001614 170700
657 001616 167770
658 001620 172600
659 001622 172700
660 001624 177420
661 001626 176200
662 001630 175610
663 001632 160010
664
665
666
667
668
669 001634 000240
670 001636 000200
671 001640 000240
672 001642 000240
673 001644 000200
674 001646 000200
675 001650 000200
676 001652 000200
677 001654 000240
678 001656 000200
679 001660 000200
680 001662 000300
681 001664 000240
682 001666 000240
683 001670 000240
684
685 001672 000300
686 001674 000104
687
688 001676 176504
689 001700 175614

AD.DC: 174000
AD.KL: 176500
AD.DP: 174770
AD.DMA: 175000
AD.DN: 175200
AD.DMB: 170500
AD.KG: 170700
AD.DR: 167770
AD.PAR: 172600
AD.PAP: 172700
AD.DT: 177420
AD.DX: 176200
AD.DLC: 175610
AD.DJ: 160010

;ADR OF FIRST DC11
;ADR OF FIRST KL11, DL11A, DL11B
;ADR OF FIRST DP11
;ADR OF FIRST DM11A
;ADR OF FIRST DN11
;ADR OF FIRST DM11BB (NO VECTOR REQUIRED)
;ADR OF FIRST KG11 (NO VECTOR REQUIRED)
;ADR OF FIRST DR11A, DR11C
;ADR OF FIRST PA611 READER
;ADR OF FIRST PA611 PUNCH
;ADR OF FIRST DT11
;ADR OF FIRST DX11
;ADR OF FIRST DL11C, DL11D, DL11E
;ADR OF FIRST DJ11

*** NOTE:
;MODIFY (AD.???) TO 160000 TO REMOVE THAT TYPE DEVICE FROM THE CONFIGURATION.
;MODIFY (AD.???) TO 174000 TO ALLOW UNCUT M7821 TO BE USED

BR.DC: 240
BR.KL: 200
BR.DP: 240
BR.DMA: 240
BR.DN: 200
BR.DMB: 200
BR.DX: 200
BR.DLC: 200
BR.DJ: 240
BR.CTY: 200
BR.LP: 200
BR.TC: 300
BR.RF: 240
BR.RK: 240
BR.RP: 240

;PRIORITY = 5 FOR DC11
;PRIORITY = 4 FOR KL11, DL11A, DL11B
;PRIORITY = 5 FOR DP11
;PRIORITY = 5 FOR DM11A
;PRIORITY = 4 FOR DN11
;PRIORITY = 4 FOR DM11BB
;PRIORITY = 4 FOR DX11
;PRIORITY = 4 FOR DL11C,D,E
;PRIORITY = 5 FOR DJ11

FIRSTV: 300
KLXSET: 104

;ADDRESS OF FIRST FLOATING VECTOR
;PATCH TO 100 TO TEST THE ASR33 READER

AD.KLX: 176504
AD.DLX: 175614

;ADDRESS OF FIRST KL XMIT REGISTER
;ADDRESS OF FIRST DL11C,D,E XMIT REGISTER


```

690
691 001702 012767 000340 176066 6.4 BEGIN: MOV #340, PS ;PRIORITY =7
692 001710 012706 001200 3.8 MOV #ICNT, SP ;STACK POINTER = ICNT
693 001714 005037 000006 3.7 CLR @#6 ;TRAP CATCHER
694 001720 012737 001734 000004 5.2 MOV #BEGINX, @#4 ;SET
695 001726 012737 000400 177774 5.2 MOV #400, @#SLR ;STACK LIMIT REGISTER 11/40 OR 11/45
696 001734 012737 000006 000004 5.2 BEGINX: MOV #6, @#4 ;+2 STACK OVERRUN
697 001742 005067 177234 4.9 CLR ERRORS ;CLR ERROR COUNT
698 001746 005067 177232 4.9 CLR PASSES ;CLR PASS COUNT
699 001752 005067 000756 4.9 CLR RPTFLG ;CLR REPORT CONFIGURATION CONTROL FLAG
700 001756 022737 023450 000020 4.7 CMP #IOTS, @#20 ;TEST FOR OVERLAY
701 001764 001406 2.6 BEQ 1$ ;BRANCH IF NONE
702 001766 013767 000020 007750 6.4 MOV @#20, IOTSAV ;SAVE OVERLAY ADDRESS
703 001774 012737 023450 000020 5.2 MOV #IOTS, @#20 ;RESTORE TYPE ROUTINE ADDRESS
704 002002 005037 000022 3.7 1$: CLR @#22 ;MAKE SURE TYPE PRIORITY IS LOW
705
706 002006 005737 000042 3.2 TST @#42 ;CHECK ACT11 MONITOR
707 002012 001004 2.6 BNE 2$ ;YES BRANCHES
708
709 002014 032767 001200 175546 6.5 BIT #1200, SWR ;NEW SYSTEM CONFIGURATION
710
711 ;THE START OR RESTART ADDRESS SHOULD BE LEFT IN THE SWITCHES IF
712 ;THIS IS THE FIRST PASS SINCE THE PROGRAM HAS BEEN LOADED OR
713 ;IF THERE HAS BEEN ANY HARDWARE ADDED OR REMOVED.
714
715 002022 001413 2.6 BEQ DVICE ;NO, INITIALIZE DEVICES AS PER
716 ;LAST SWITCH SETTINGS
717 002024 013767 000042 176772 6.4 2$: MOV @#42, CLINK ;SET MONITOR QUICK PASS CONTROL
718 002032 004767 000030 7.0 JSR PC, CLRVEC ;LOAD VECTOR AREA WITH +2, IOT
719 002036 004767 000066 7.0 JSR PC, SYSMAP ;DETERMINE SYSTEM CONFIGURATION
720 002042 004767 000774 7.0 JSR PC, SWITCH ;DETERMINE TEST CONFIGURATION FROM SWITCHES
721 002046 004767 002132 7.0 JSR PC, SYSGEN ;INITIALIZE VECTOR AREA AS PER TEST CONF.
722 002052 004767 003036 7.0 DVICE: JSR PC, DVICE ;INITIALIZE DEVICES AS PER TEST CONFIGURATION
723 002056 004767 006670 7.0 JSR PC, BCKGND ;RUN BACKGROUND PROGRAM TO
724 ;MONITOR SILENT DEVICES AND NPR
725 ;DATA FILES
726 002062 000773 2.6 BR DVICE ;LOOP
727
728 002064 000000 1.8 HALT ;HALT HERE SIGNIFIES RTS FAILURE
729

```

730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752

```
*****CLRVEC*****  
;CLRVEC, ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH  
; .+2  
; HALT  
*****CLRVEC*****  
CLRVEC: MOV #300,R2 ;R2 COMM VECTOR AREA ADRS  
MOV #302,R1 ;INIT R1 WITH ADRS OF HALT  
CV1: MOV R1,(R2)+ ;MOVE .+2 INTO VECTOR  
CLR (R2)+ ;MOVE HALT INTO VECTOR + 2  
CMP (R1)+,(R1)+ ;INC TO NEXT VECTOR AREA  
CMP #1000,R1 ;END OF VECTOR AREA  
BGT CV1 ;NO  
MOV #137,200 ;REINIT START ADRS  
MOV #BEGIN,202 ;  
RTS PC ;RETURN  
*****CLRVEC*****
```

002066	012702	000300		
002072	012701	000302		
002076	010122			
002100	005022			
002102	022121			
002104	022701	001000		
002110	003372			
002112	012767	000137	176060	
002120	012767	001702	176054	
002126	000207			


```

753
754
755
756
757
758
759
760
761
762
763
764
765 002130 004567 020712      7.0  SYSMAP: JSR      %5,CLEAR      ;CLEAR MONOTOR MAPS
766 002134 001210              SY.MAP      ;START HERE
767 002136 000122              MAPSIZ*3+40+5 ;START ADDRESS FIVE TABLES
768 002140 012700 001600      3.8  MOV      #AD.DC, R0      ;GET ADDRESS TABLE POINTER
769 002144 012702 001214      3.8  MOV      #SY.DC1, R2     ;GET SYSTEM MAP POINTER
770 002150 012703 001506      3.8  MOV      #NO.DC, R3     ;SET R3 = FIRST DEVICE COUNT
771 002154 012704 001542      3.8  MOV      #MX.DC, R4
772 002160 012767 003020 175616 6.4  MOV      #NEXT, 4
773 002166 012767 000340 175612 6.4  MOV      #340, 6      ;SET UP TIME OUT VECTOR
774                                     ;TIME OUT STATUS
775 002174 004567 000536      7.0  ANYDC: JSR      %5,MAPIT      ;MAP DC11'S
776 002200 000002              FL.DC      ;DC11 FLAG
777
778 002202 012702 001220      3.8  ANYKL: MOV      #SY.KL, R2     ;RESET SYSMAP TABLE POINTER
779 002206 004567 000524      7.0  JSR      %5,MAPIT      ;MAP KL11'S
780 002212 000004              FL.KL      ;KL11 FLAG
781
782 002214 052767 100000 000546 6.4  ANYDP: BIS      #BIT15, MITS ;CHANCE ADD TO SUB
783 002222 004567 000510      7.0  JSR      %5,MAPIT      ;MAP DP11'S
784 002226 000010              FL.DP      ;DP11 FLAG
785 002230 042767 100000 000532 7.0  BIC      #BIT15, MITS ;CHANCE SUB TO ADD
786
787 002236 012702 001226      3.8  ANYDMA: MOV     #SY.DMA, R2    ;RESET SYSMAP TABLE POINTER
788 002242 004567 000470      7.0  JSR      %5,MAPIT      ;MAP DM11A'S
789 002246 000020              FL.DMA     ;DM11A FLAG
790
791 002250 004567 000462      7.0  ANYDN: JSR      %5,MAPIT      ;MAP DN11'S
792 002254 000040              FL.DN      ;DN11 FLAG
793
794 002256 004567 000454      7.0  ANYDMB: JSR     %5,MAPIT      ;MAP DM11BB'S
795 002262 000100              FL.DMB     ;DM11BB FLAG
796
797 002264 004567 000446      7.0  ANYKG: JSR      %5,MAPIT      ;MAP KG11'S
798 002270 000200              FL.KG      ;KG11 FLAG
799
800 002272 012702 001442      3.8  ANYDR: MOV     #SY.DR1, R2    ;RESET SYSTEM MAP POINTER
801 002276 052767 100000 000464 6.4  BIS      #BIT15, MITS ;CHANCE ADD TO SUB
802 002304 004567 000426      7.0  JSR      %5,MAPIT      ;MAP DR11A'S AND DR11C'S
803 002310 000000              0          ;NO FLAG
804 002312 042767 100000 000450 7.0  BIC      #BIT15, MITS ;CHANCE SUB TO ADD
805
806 002320 006267 000446      4.9  ANYPAR: ASR     MITS+2      ;CHANGE ADR INCREMENT TO 4
807 002324 004567 000406      7.0  JSR      %5,MAPIT      ;MAP PA611 READERS
808 002330 000000              0          ;NO FLAG

```

```

809
810 002332 004567 000400      7.0 ANYPAP: JSR      %5,MAPIT      ;MAP PA611 PUNCHES
811 002336 000000                0                          ;NO FLAG
812
813 002340 006267 000426      4.9 ANYDT:  ASR      MITS+2      ;CHANGE ADR INCREMENT TO 2
814 002344 004567 000366      7.0                JSR      %5,MAPIT      ;MAP DT11'S
815 002350 000000                0                          ;NO FLAG
816
817 002352 012702 001236      3.8 ANYDX:  MOV      #SY.DX, R2    ;RESET SYSTEM MAP POINTER
818 002356 012767 000040 000406  6.4                MOV      #40, MITS+2    ;CHANGE ADR INCREMENT TO 40
819 002364 004567 000346      7.0                JSR      %5,MAPIT      ;MAP DX11'S
820 002370 000400                .WORD  FL.DX            ;DX11 FLAG
821
822 002372 012767 000010 000372  6.4 ANYDLC: MOV      #10, MITS+2    ;CHANGE ADR INCREMENT BACK TO 10
823 002400 004567 000332      7.0                JSR      %5,MAPIT      ;MAP DL11C,D,E'S
824 002404 001000                FL.DLC                    ;DL11 FLAG
825
826 002406 012702 001244      3.8 ANYDJ:  MOV      #SY.DJ, R2    ;RESET SYSTEMS MAP POINTER
827 002412 004567 000320      7.0                JSR      %5,MAPIT      ;MAP DJ11'S
828 002416 004000                FL.DJ                    ;DJ11 FLAG
829
830
831 ;ANCIL, CODE TO DETERMINE IF RF, TC, LP EXIST
832
833 002420 012767 003034 175356  6.4 ANCIL:  MOV      #ANEXT,4      ;SET UP TIME OUT POINTER
834 002426 012767 002466 000404  6.4                MOV      #ANC1, NT2    ;TIME OUT TRAPS TO NEXT TST
835 002434 005777 020352      5.6                TST      @CTKS        ;DOES CTY EXIST
836 002440 052767 000001 176544  6.4                BIS      @BIT0, SY.ANC ;YES, SET CTY BIT
837 ;***** NO SACK TIMEOUT TEST *****
838
839 002446 105777 020344      5.6 NOSAC:  TSTB     @CTPS        ;IS PUNCH READY
840 002452 100375                BPL     NOSAC          ;WAIT FOR READY
841 002454 052777 000100 020334  7.6                BIS      #100,@CTPS   ;SET INTERRUPT ENABLE
842 002462 005077 020330      6.1                CLR     @CTPS        ;CLEAR INTERRUPT ENABLE
843 ;PROCESSOR WILL HANG HERE IF NO-SACK-TIMEOUT FAILS
844 ;*****
845 002466 012767 002506 000344  6.4 ANC1:  MOV      #ANC2, NT2    ;TIME OUT TO NEXT DEVICE
846 002474 005777 020326      5.6                TST     @LP11        ;DOES LINE PRINTER EXIST
847 002500 052767 000002 176504  6.4                BIS     @BIT1, SY.ANC ;YES, SET LP11 BIT
848 002506 012767 002526 000324  6.4 ANC2:  MOV      #ANC3, NT2    ;TIME OUT TO NEXT DEVICE
849 002514 005777 020302      5.6                TST     @TC11        ;DOES DECTAPE EXIST
850 002520 052767 000004 176464  6.4                BIS     @BIT2, SY.ANC ;SET TC11 BIT
851 002526 012767 002546 000304  6.4 ANC3:  MOV      #ANC4, NT2    ;TIME OUT TO NEXT DEVICE
852 002534 005777 020264      5.6                TST     @RF11        ;DOES RF11 DISK EXIST
853 002540 052767 000010 176444  6.4                BIS     @BIT3, SY.ANC ;YES, SET RF11 BIT
854 002546 012767 002566 000264  6.4 ANC4:  MOV      #ANC5, NT2    ;TIME OUT TO NEXT DEVICE
855 002554 005777 017570      5.6                TST     @RKCSA       ;TEST FOR RK11 DISK
856 002560 052767 000020 176424  6.4                BIS     @BIT4, SY.ANC ;SET RK FLAG
857 002566 012767 002606 000244  6.4 ANC5:  MOV      #ANC6, NT2    ;TIMEOUT TO NEXT DEVICE
858 002574 005777 017732      5.6                TST     @RPSA        ;DOES RP EXIST
859 002600 052767 000040 176404  6.4                BIS     @BIT5, SY.ANC ;YES SET RP BIT
860 002606 012767 000006 175170  6.4 ANC6:  MOV      #6,4          ;REINSTATE TRAP TO HALT
861 002614 012767 000000 175164  6.4                MOV     #HALT,6
862 002622 005737 000042      3.2                TST     @#42
863 002626 001417                BEQ     SYSNO
864 002630 012701 001210      3.8                MOV     #SY.MAP,R1    ;GET SYSTEM MAP ADR

```



```

865 002634 012702 001246 3.8 MOV #SW.MAP,R2 ;GET SWITCH SETTING MAP ADR
866 002640 012703 000017 3.8 MOV #MAPSIZ,R3 ;GET MAP SIZE
867 002644 012122 5.2 SY.SW: MOV (R1)+,(R2)+ ;MOVE SY.MAP INTO SW.MAP
868 002646 005303 2.3 DEC R3 ;CHECK FOR END
869 002650 003375 2.6 BGT SY.SW ;BRANCH IF NOT DONE
870 002652 062716 000004 5.2 ADD #4,(6) ;SKIP "JSR PC SWITCH"
871 002656 005767 000052 4.4 TST RPTFLG ;TEST CONFIGURATION REPORT FLAG
872 002662 001401 2.6 BEQ SYSMD ; FIRST PASS BRANCHES
873 002664 000207 3.5 RTS PC
874
875 002666 005767 176320 4.4 SYSMD: TST SY.ANC ;CHECK FOR ANY ANCILLARIES
876 002672 001403 2.6 BEQ IS ;BRANCH IF NONE
877 002674 052767 000001 176306 6.4 BIS #BIT0,SY.MAP ;SET ANCIL BIT OF SYSTEM MAP
878 002702 032737 020000 177570 5.3 IS: BIT #BIT13,#SWR ;CHECK INHIBIT CONFIGURATION REPORT
879 002710 001010 2.6 BNE SYM1 ;BRANCH IF SET
880 002712 000004 027714 TYPE, MSG1 ;TYPE "SYSTEM CONFIGURATION"
881 002716 012701 001212 3.8 MOV #SY.ANC,R1 ;POINT TO SYSTEM MAP
882 002722 010167 000006 4.9 MOV R1,RPTFLG ;SET PASS CONTROL FLAG
883 002726 004767 020140 7.0 JSR PC,REPORT ;REPORT CONFIGURATION
884 002732 000207 3.5 SYM1: RTS PC ;RETURN TO MAIN LINE
885
886 002734 000000 RPTFLG: 0 ;CONFIGURATION REPORT CONTROL FLAG
887
888
889 ;MAPIT, THIS SUBROUTINE PLACES A FLAG IN THE MAP OF EACH DEVICE TYPE.
890 ; EACH BIT IN THE DEVICE MAP INDICATES A UNIT. FOR EXAMPLE,
891 ; A LOGIC ONE IN BIT LOCATIONS ZERO AND ONE OF THE DP11 MAP
892 ; INDICATES TWO DP11'S EXIST.
893
894 002736 012001 3.8 MAPIT: MOV (R0)+,R1 ;FETCH DEVICE ADDRESS
895 002740 012467 020020 6.4 MOV (R4)+,MAX ;FETCH MAX NUMBER OF DEVICE
896 002744 005067 020074 4.9 CLR DEVCNT ;CLEAR DEVICE COUNT
897 002750 012767 000001 020010 6.4 MIT1: MOV #1,MARK ;INC "DEVICE EXISTS" MARKER
898 002756 005711 3.2 MIT2: TST #R1 ;TIME OUT TO NEXT DEVICE IF NON EXISTANT
899
900 ;NOTE: IF FIRST DEVICE DOES NOT EXIST THIS
901 ; ROUTINE WILL STOP LOOKING FOR ANY MORE
902
903 002760 051567 176224 6.4 BIS (R5),SY.MAP ;SET BIT IN MAIN MAP
904 002764 005267 020054 4.9 INC DEVCNT ;INC DEVICE COUNT
905 002770 062701 000010 3.8 MIT5: ADD #10,R1 ;INC TO NEXT ADRS
906 002774 056712 017766 6.4 BIS MARK,R2 ;SET MAP BIT
907 003000 005367 017760 4.9 DEC MAX ;HAS THE MAX # BEEN TESTED
908 003004 001406 2.6 BEQ MIT4 ;YES
909 003006 006167 017754 4.9 ROL MARK ;NO, SHIFT MARKER
910 003012 103361 2.6 BCC MIT2 ;MAP WORD FULL
911 003014 005722 3.2 TST (R2)+ ;INC TO NEXT MAP
912 003016 000754 2.6 BR MIT1
913
914 ;NEXT, TIME OUT ROUTINE TO ADJUST STACK AND JUMP TO
915 ;NEXT DEVICE TEST
916
917 003020 022626 4.7 NEXT: CMP (SP)+,(SP)+ ;ADJUST STACK
918 003022 005722 3.2 MIT4: TST (R2)+ ;UPDATE SYSTEM MAP POINTER
919 003024 016723 020014 6.4 MOV DEVCNT,(R3)+ ;SAVE DEVICE COUNT
920 003030 005725 3.2 TST (R5)+ ;INC RETURN ADR.

```

```
921 003032 000205          3.5          RTS          %5          ;RETURN
922
923
924 003034 022626          4.7          ;TIMEOUT TRAP FOR ANCILLARIES
925 003036 000137          3.7          ANEXT:  CMP          (SP)+,(SP)+          ;ADJUST STACK
926 003040 000000          NT1:  JMP          @PC+          ;TEST NEXT DEVICE
927
928
;***** SYSMAP *****
```



```

929
930
931
932
933
934
935 003042 004567 020000 7.0 SWITCH: JSR RS, CLEAR ;CLEAR TABLE SUBROUTINE
936 003046 001246 SW.MAP ;START HERE
937 003050 000017 MAPSIZ ;THIS MANY WORDS
938 003052 005067 003366 4.9 CLR DX.OLF ;RESET DX ACTIVITY
939
940 003056 005737 000042 3.2 TST J#42 ;CHECK ACT11 MONITOR
941 003062 001412 2.6 BEQ IS ;NO BYPASS
942
943 003064 012700 000017 3.8 MOV #MAPSIZ,RO ;SCHEDULE
944 003070 012701 001210 3.8 MOV #SY.MAP,R1 ;ALL
945 003074 012702 001246 3.8 MOV #SW.MAP,R2 ;DEVICES
946 003100 012122 5.2 25: MOV (R1)+,(R2)+ ;ON SYSTEM
947 003102 005300 2.3 DEC RO ;FOR
948 003104 001375 2.6 BNE 25 ;TEST
949 003106 000207 3.5 RTS PC ;RETURN TO MAINLINE
950
951 003110 012700 000001 3.8 IS: MOV #1, RO ;SET RO TO DISPLAY SETTING 1
952 003114 000000 1.8 HLT01: HALT ;SELECT DEVICES
953 ;PRESS CONTINUE
954 003116 016767 174446 176122 7.6 MOV SWR, SW.MAP ;SAVE SWITCH REG
955 003124 032767 000001 176114 6.5 BIT #BIT0, SW.MAP ;CHECK FOR NON-COMM DEVICES
956 003132 001406 2.6 BEQ STCH0 ;BRANCH IF NONE
957 003134 012700 000002 3.8 MOV #2, RO ;DISPLAY SETTING #2
958 003140 000000 1.8 HLT02: HALT ;SELECT NON-COMM TEST CONFIGURATION
959 ;PRESS CONTINUE
960 003142 016767 174422 176100 7.6 STCH0: MOV SWR, SW.ANC ;SAVE SWITCH REG
961 003150 032767 000002 176070 6.5 BIT #BIT1, SW.MAP ;CHECK FOR ANY DC11'S
962 003156 001417 2.6 BEQ STCH1 ;BRANCH IF NONE
963 003160 012700 000003 3.8 MOV #3, RO ;DISPLAY SETTING #3
964 003164 000000 1.8 HLT03: HALT ;FIRST DC11 SETTING
965 ;PRESS CONTINUE
966 003166 016767 174376 176056 7.6 MOV SWR, SW.DC1 ;SAVE SWITCH REG
967 003174 005767 176016 4.4 TST SY.DC2 ;ARE THERE DC11 LINES 16-31
968 003200 001406 2.6 BEQ STCH1 ;BRANCH IF NOT
969 003202 012700 000004 3.8 MOV #4, RO ;DISPLAY SETTING #4
970 003206 000000 1.8 HLT04: HALT ;SECOND DC11 SETTING
971 ;PRESS CONTINUE
972 003210 016767 174354 176036 7.6 STCH1: MOV SWR, SW.DC2 ;SAVE SETTINGS
973 003216 032767 000004 176022 6.5 BIT #BIT2, SW.MAP ;NEW KL11 SETTINGS
974 003224 001406 2.6 BEQ STCH2 ;NO
975 003226 012700 000005 3.8 MOV #5, RO ;YES, DISPLAY SETTING 5=TTY
976 003232 000000 1.8 HLT05: HALT ;SET SWR TO TEST PATTERS
977 ;PRESS CONTINUE
978 003234 016767 174330 176014 7.6 STCH2: MOV SWR, SW.KL ;SAVE SETTINGS
979 003242 032767 000010 175776 6.5 BIT #BIT3, SW.MAP ;NEW DP11 TEST CONFIGURATION
980 003250 001417 2.6 BEQ STCH3 ;NO
981 003252 012700 000006 3.8 MOV #6, RO ;DISPLAY SETTING 6=DP11
982 003256 000000 1.8 HLT06: HALT ;SELECT DP11 TEST CONFIGURATION
983 ;PRESS CONTINUE
984 003260 016767 174304 175772 7.6 MOV SWR, SW.DP1 ;SAVE DP LINES 0-15

```

985	003266	005767	175732		4.4	TST	SY.DP2		:ARE THERE DP11 LINES 16-31
986	003272	001406			2.6	BEQ	STCH3		:BRANCH IF NOT
987	003274	012700	000007		3.8	MOV	#7,	RO	:DISPLAY SETTING 7
988	003300	000000			1.8	HLT07:	HALT		:SELECT DP LINES 16-31
989									:PRESS CONTINUE
990	003302	016767	174262	175752	7.6	MOV	SWR,	SW.DP2	:SAVE SETTINGS
991	003310	032767	000020	175730	6.5	STCH3:	#BIT4,	SW.MAP	:NEW DM11 TEST CONFIGURATION
992	003316	001406			2.6	BEQ	STCH4'		
993	003320	012700	000010		3.8	MOV	#10,	RO	:DISPLAY SETTING #10
994	003324	000000			1.8	HLT10:	HALT		:SELECT DM11'S
995									:PRESS CONTINUE
996	003326	016767	174236	175730	7.6	MOV	SWR,	SW.DMA	:SAVE SETTINGS
997	003334	032767	000040	175704	6.5	STCH4:	#BIT5,	SW.MAP	:NEW DN11 TEST CONFIGURATION
998	003342	001406			2.6	BEQ	STCH5'		
999	003344	012700	000011		3.8	MOV	#11,	RO	:DISPLAY SETTING #11
1000	003350	000000			1.8	HLT11:	HALT		:SELECT DN11'S
1001									:PRESS CONTINUE
1002	003352	016767	174212	175706	7.6	MOV	SWR,	SW.DN	:SAVE SETTINGS
1003	003360	032767	000100	175660	6.5	STCH5:	#BIT6,	SW.MAP	:CHECK FOR DM11BB'S
1004	003366	001406			2.6	BEQ	STCH6'		:BRANCH IF NONE
1005	003370	012700	000012		3.8	MOV	#12,	RO	:DISPLAY SETTING #12
1006	003374	000000			1.8	HLT12:	HALT		:SELECT DM11BB'S
1007									:PRESS CONTINUE
1008	003376	016767	174166	175664	7.6	MOV	SWR,	SW.DMB	:SAVE SETTINGS
1009	003404	032767	000200	175634	6.5	STCH6:	#BIT7,	SW.MAP	:CHECK FOR KG11'S
1010	003412	001406			2.6	BEQ	STCH7'		:BRANCH IF NONE
1011	003414	012700	000013		3.8	MOV	#13,	RO	:DISPLAY SETTING #13
1012	003420	000000			1.8	HLT13:	HALT		:SELECT KG11'S
1013									:PRESS CONTINUE
1014	003422	016767	174142	175642	7.6	MOV	SWR,	SW.KG	:SAVE SETTINGS
1015	003430	032767	000400	175610	6.5	STCH7:	#BIT8,	SW.MAP	:CHECK FOR DX11'S
1016	003436	001436			2.6	BEQ	STCH8'		:BRANCH IF NONE
1017	003440	012700	000014		3.8	MOV	#14,	RO	:DISPLAY SETTING #14
1018	003444	000000			1.8	HLT14:	HALT		:SELECT DX11'S
1019									:PRESS CONTINUE
1020	003446	113767	177570	175620	6.4	MOVB	@SWR,	SW.DX	:SAVE SETTINGS
1021	003454	113737	177571	006444	5.2	MOVB	@SWR+1,	@DX.OLF	:SAVE ON-LINE SELECTION
1022	003462	053767	006444	175604	6.4	BIS	@DX.OLF,	SW.DX	:MAKE SURE DX SELECTED FOR ON-LINE SELECTED.
1023	003470	012701	000001		3.8	MOV	#1,	R1	:SET UP MARKER
1024	003474	012700	177400		3.8	MOV	#177400,	RO	:SET UP DATA LIGHTS
1025	003500	012702	017430		3.8	MOV	@DXLEGA,	R2	:ADDRESS POINTER IN R2
1026	003504	030137	001274		3.8	STCH70:	R1,	@SW.DX	:CHECK FOR DX11 SELECTED
1027	003510	001403			2.6	BEQ	STCH80		:BRANCH IF NOT
1028	003512	000000			1.8	HLT70:	HALT		:SELECT DX11 LEGAL ADDRESS
1029									:PRESS CONTINUE
1030	003514	013712	177570		5.2	MOV	@SWR,	(R2)	:SAVE IT
1031	003520	005722			3.2	STCH80:	(R2)+		:UPDATE POINTER
1032	003522	006301			2.3	ASL	R1		:UPDATE MARKER
1033	003524	005200			2.3	INC	RO		:UPDATE DATA LIGHTS
1034	003526	120067	176002		4.4	CMPB	RO	NO.DX	:CHECK FOR ALL DONE
1035	003532	001364			2.6	BNE	STCH70		:BRANCH IF NOT
1036	003534	032767	001000	175504	6.5	STCH8:	#BIT9,	SW.MAP	:CHECK FOR DL11C,D,E'S
1037	003542	001420			2.6	BEQ	STCH9'		:BRANCH IF NONE
1038	003544	012700	000015		3.8	MOV	#15,	RO	:DISPLAY SETTING #15
1039	003550	000000			1.8	HLT15:	HALT		:SELECT DL11C,D,E'S
1040									:PRESS CONTINUE

1041	003552	016767	174012	175516	7.6		MOV	SWR,	SW.DL1	;SAVE SETTINGS
1042	003560	005767	175456		4.4		TST	SY.DL2		;ARE THERE DL11 LINES 16-31
1043	003564	001407			2.6		BEG	STCH9		;BRANCH IF NOT
1044	003566	012700	000016		3.8		MOV	#16,	RO	;DISPLAY SETTING #16
1045	003572	000000			1.8	HLT16:	HALT			;SELECT DL11C,D,E'S
1046										;PRESS CONTINUE
1047	003574	016767	173770	175476	7.6		MOV	SWR,	SW.DL2	;SAVE SETTINGS
1048										
1049	003602	000400			2.6		BR	STCH9		;*****
1050										
1051	003604	032767	004000	175434	6.5	STCH9:	BIT	#BIT11,	SW.MAP	;CHECK FOR DJ11'S
1052	003612	001526			2.6		BEG	STCH30		;BRANCH IF NONE
1053										
1054	003614	012700	000200		3.8		MOV	#200,	RO	;DISPLAY SETTING #200
1055	003620	000000			1.8	HLT200:	HALT			;SELECT DJ11 DEVICE #0-#7
1056	003622	016767	173742	175452	7.6		MOV	SWR,	SW.DJ	;RECORD SETTING
1057										
1058										
1059	003630	032767	000001	175444	6.5		BIT	#BIT0,	SW.DJ	;DJ11 DEVICE #0 TO BE TESTED?
1060	003636	001406			2.6		BEG	STCH10		;BRANCH IF NOT
1061										
1062	003640	012700	000020		3.8		MOV	#20,	RO	;DISPLAY SETTING #20
1063	003644	000000			1.8	HLT20:	HALT			;SELECT DJ11 DEVICE #0 LINES 0 THRU 15 FOR TEST
1064										
1065	003646	016767	173716	175602	7.6		MOV	SWR,	SW.DJ0	;SAVE SETTING
1066										
1067										
1068	003654	032767	000002	175420	6.5	STCH10:	BIT	#BIT1,	SW.DJ	;DJ11 DEVICE #1 TO BE TESTED?
1069	003662	001406			2.6		BEG	STCH11		;BRANCH IF NOT
1070										
1071	003664	012700	000021		3.8		MOV	#21,	RO	;DISPLAY SETTING #21
1072	003670	000000			1.8	HLT21:	HALT,			;SELECT DJ11 DEVICE #1 LINES 0 THRU 15 FOR TEST
1073										
1074	003672	016767	173672	175560	7.6		MOV	SWR,	SW.DJ1	;SAVE SETTING
1075										
1076										
1077	003700	032767	000004	175374	6.5	STCH11:	BIT	#BIT2,	SW.DJ	;DJ11 DEVICE #2 TO BE TESTED?
1078	003706	001406			2.6		BEG	STCH12		;BRANCH IF NOT
1079	003710	012700	000022		3.8		MOV	#22,	RO	;DISPLAY SETTING #22
1080	003714	000000			1.8	HLT22:	HALT			;SELECT DJ11 DEVICE #2 LINES 0 THRU 15 FOR TEST
1081										
1082	003716	016767	173646	175536	7.6		MOV	SWR,	SW.DJ2	;SAVE SETTING
1083										
1084										
1085										
1086	003724	032767	000010	175350	6.5	STCH12:	BIT	#BIT3,	SW.DJ	;DJ11 DEVICE #3 TO BE TESTED?
1087	003732	001406			2.6		BEG	STCH13		;BRANCH IF NOT
1088										
1089	003734	012700	000023		3.8		MOV	#23,	RO	;DISPLAY SETTING #23
1090	003740	000000			1.8	HLT23:	HALT			;SELECT DJ11 DEVICE #3 LINES 0 THRU 15 FOR TEST
1091										
1092	003742	016767	173622	175514	7.6		MOV	SWR,	SW.DJ3	;SAVE SETTING
1093										
1094										
1095										
1096	003750	032767	000020	175324	6.5	STCH13:	BIT	#BIT4,	SW.DJ	;DJ11 DEVICE #4 TO BE TESTED?

```

1097 003756 001406          2.6          BEQ      STCH14          ;BRANCH IF NOT
1098
1099 003760 012700 000024          3.8          MOV      #24,   R0          ;DISPLAY SETTING #24
1100 003764 000000          1.8 HLT24:  HALT          ;SELECT DJ11 DEVICE #4 LINES 0 THRU 15 FOR TEST
1101
1102 003766 016767 173576 175472          7.6          MOV      SWR,   SW.DJ4   ;SAVE SETTING
1103
1104
1105
1106 003774 032767 000040 175300          6.5 STCH14:  BIT      #BIT5, SW.DJ   ;DJ11 DEVICE #5 TO BE TESTED?
1107 004002 001406          2.5          BEQ      STCH15          ;BRANCH IF NOT
1108
1109 004004 012700 000025          3.8          MOV      #25,   R0          ;DISPLAY SETTING #25
1110 004010 000000          1.8 HLT25:  HALT          ;SELECT DJ11 DEVICE #5 LINES 0 THRU 15 FOR TEST
1111
1112 004012 016767 173552 175450          7.6          MOV      SWR,   SW.DJ5   ;SAVE SETTING
1113
1114
1115
1116 004020 032767 000100 175254          6.5 STCH15:  BIT      #BIT6, SW.DJ   ;DJ11 DEVICE #6 TO BE TESTED?
1117 004026 001406          2.5          BEQ      STCH16          ;BRANCH IF NOT
1118
1119 004030 012700 000026          3.8          MOV      #26,   R0          ;DISPLAY SETTING #26
1120 004034 000000          1.8 HLT26:  HALT          ;SELECT DJ11 DEVICE #6 LINES 0 THRU 15 FOR TEST
1121
1122 004036 016767 173526 175426          7.6          MOV      SWR,   SW.DJ6   ;SAVE SETTING
1123
1124
1125
1126 004044 032767 000200 175230          6.5 STCH16:  BIT      #BIT7, SW.DJ   ;DJ11 DEVICE #7 TO BE TESTED?
1127 004052 001406          2.6          BEQ      STCH30          ;BRANCH IF NOT
1128
1129 004054 012700 000027          3.8          MOV      #27,   R0          ;DISPLAY SETTING #27
1130 004060 000000          1.8 HLT27:  HALT          ;SELECT DJ11 DEVICE #7 LINES 0 THRU 15 FOR TEST
1131
1132 004062 016767 173502 175404          7.6          MOV      SWR,   SW.DJ7   ;SAVE SETTING
1133 004070 012701 001210          3.8 STCH30:  MOV      #SY.MAP,R1   ;GET SYSTEM MAP ADR
1134 004074 012702 001246          3.8          MOV      #SW.MAP,R2   ;GET SWITCH SETTING MAP ADR
1135
1136 004100 012703 000017          3.8          MOV      #MAPSIZ,R3   ;GET MAP SIZE
1137 004104 005111          3.7 SWCHK:  COM      (R1)
1138 004106 031122          5.3          BIT      (R1), (R2)+   ;CHECK IF SELECTED DEV. EXIST
1139 004110 001425          2.6          BEQ      SWCHK2        ;BRANCH IF THEY DO
1140 004112 005121          3.7          COM      (R1)+        ;RESTORE SY.MAP
1141 004114 000004 004120          TYPE      +2
1142 004120 005015 042523 042514          .ASCIZ  <15><12>"SELECTED NON-EXISTANT DEVICE"
1143 004126 052103 042105 047040
1144 004134 047117 042455 044530
1145 004142 052123 047101 020124
1146 004150 042504 044526 042503
1147 004156 000
1148
1149 004160 000167 176656          4.9          JMP      .EVEN        ;TRY AGAIN!
1150
1151 004164 005121          3.7 SWCHK2:  COM      (R1)+
1152 004166 005303          2.3          DEC      R3           ;RESTORE SY.MAP AND INCREMENT
                           ;CHECK FOR END OF MAPS

```


MAINDEC-11-DZQCA-G
DZQCA.P11

COMMUNICATION TEST PROGRAM (CTP)
SWITCH (CONSOLE SWITCH SAVE ROUTINE)

I04

MACY11 27(732) 17-SEP-76 15:12 PAGE 27

1153	004170	003345		2.6	BGT	SWCHK			;BRANCH IF NOT DONE
1154	004172	012700	177777	3.8	MOV	#-1.	RO		;DISPLAY SETTING #-1
1155	004176	000000		1.8	HLT77: HALT				;SELECT DYNAMIC SWITCH SETTINGS
1156									
1157	004200	005000		2.3	CLR	RO			
1158	004202	000207		3.5	RTS	PC			;RETURN TO MAIN LINE

```

1159
1160
1161
1162
1163
1164
1165
1166
1167
1168 004204 016700 175462 5.0 SYSGEN: MOV FIRSTV, R0 ;LOAD R0 WITH FIRST COMM VECTOR
1169 004210 012701 031424 3.8 MOV #LINKER, R1 ;LOAD R1 WITH ADRS OF LINKER AREA
1170 004214 012702 001634 3.8 MOV #BR.DC, R2 ;SET UP BR TABLE POINTER
1171 004220 012703 001506 3.8 MOV #NO.DC, R3 ;SET UP DEVICE COUNT TABLE POINTER
1172 004224 012704 001252 3.8 MOV #SW.DC1, R4 ;SET UP DEVICE SELECTED MAP POINTER
1173
1174 ;LINK DC'S WITH THEIR ISR
1175
1176 004230 004537 004572 5.8 JSR %5, @#LINK ;GENERATE CODE TO LINK DC'S WITH ISR
1177 004234 012776 DC.RCV ;RCV ISR ADRS
1178 004236 012730 DC.XMT ;XMIT ISR ADRS
1179
1180 ;LINK KL VECTORS WITH THEIR ISR'S
1181
1182 004240 012704 001256 3.8 MOV #SW.KL, R4
1183 004244 004537 004572 5.8 JSR %5, @#LINK ;LINK KL'S WITH ISR'S
1184 004250 013302 KL.RCV ;RCV ISR ADRS
1185 004252 013204 KL.XMT ;XMIT ISR ADRS
1186
1187 ;LINK DP'S WITH ISR'S
1188
1189 004254 004537 004572 5.8 JSR %5, @#LINK ;LINK DP'S WITH ISR'S
1190 004260 013532 DP.RCV ;RCV ISR ADRS
1191 004262 013704 DP.XMT ;XMIT ISR ADRS
1192
1193 ;LINK DM'S WITH ISR'S
1194
1195 004264 012704 001264 3.8 MOV #SW.DMA, R4
1196 004270 004537 004572 5.8 JSR %5, @#LINK ;LINK DM'S WITH ISR'S
1197 004274 014342 DM.RCV ;RCV ISR ADRS
1198 004276 014222 DM.XMT ;XMIT ISR ADRS
1199
1200 ;LINK DN'S WITH ISR'S
1201
1202 004300 004537 004676 5.8 JSR %5, @#LINK1 ;LINK DN'S WITH ISR
1203 004304 015054 DN.ISR ;ISR ADR
1204
1205 ;LINK DM11BB'S WITH ISR
1206
1207 004306 004537 004676 5.8 JSR %5, @#LINK1
1208 004312 015370 .WORD DMBISR
1209
1210 004314 005723 3.2 TST (R3)+ ;SKIP KG DEVICE COUNT
1211 004316 005724 3.2 TST (R4)+ ;SKIP KG SWITCH MAP
1212
1213 004320 004567 000452 7.0 JSR %5, SKIPVA ;DR11
1214 004324 000010 .WORD 10

```



```

1215 004326 004567 000444      7.0      JSR      %5,SKIPVA      ;PA611 R
1216 004332 000004                .WORD    4
1217 004334 004567 000436      7.0      JSR      %5,SKIPVA      ;PA611 P
1218 004340 000004                .WORD    4
1219 004342 004567 000430      7.0      JSR      %5,SKIPVA      ;DT11
1220 004346 000010                .WORD    10
1221
1222                                     ;LINK DX11'S WITH ISR
1223
1224 004350 004767 000450      7.0      JSR      PC,      MOD10      ;MAKE SURE VECTOR ADR POINTER IS ON MOD10
1225 004354 004537 004572      5.8      JSR      %5,%LINK
1226 004360 015442                .WORD    DX.ISR      ;DEFAULT TO OFF LINE
1227 004362 015436                .WORD    DMY.ISR
1228 004364 005767 002054      4.4      TST      DX.OLF      ;CHECK FOR ANY DX ON-LINE FLAGS
1229 004370 001451                BEQ      6$      ;BRANCH IF NONE
1230 004372 012767 000001 016366      6.4      MOV      #1,      MARK      ;SET UP MARKER
1231 004400 016767 175130 016356      7.6      MOV      NO.DX,    MAX      ;SET UP MAX COUNTER
1232 004406 036767 016354 174660      7.7 1$:    BIT      MARK,    SW.DX      ;CHECK FOR DX SELECTED
1233 004414 001402                BEQ      2$      ;BRANCH IF NOT
1234 004416 162701 000014      3.8      SUB      #14,    R1      ;DECREMENT LINKER POINTER
1235 004422 006367 016340      4.9 2$:    ASL      MARK
1236 004426 005367 016332      4.9      DEC      MAX      ;UPDATE MARKER
1237 004432 001365                BNE      1$      ;CHECK FOR DONE
1238 004434 012767 000001 016324      6.4      MOV      #1,      MARK      ;BRANCH IF NOT
1239 004442 016767 175066 016314      7.6      MOV      NO.DX,    MAX      ;SET UP MARKER
1240 004450 036767 016312 174616      7.7 3$:    BIT      MARK,    SW.DX      ;SET UP COUNTER
1241 004456 001411                BEQ      5$      ;CHECK FOR DX SELECTED
1242 004460 036767 016302 001756      7.7      BIT      MARK,    DX.OLF      ;BRANCH IF NOT
1243 004466 001403                BEQ      4$      ;CHECK FOR ON-LINE
1244 004470 012761 015636 000002      6.4      MOV      #DXOL.ISR,2(R1) ;CHANGE THE ISR ADDRESS TO ON LINE
1245 004476 062701 000014      3.8 4$:    ADD      #14,    R1      ;RESTORE R1
1246 004502 006367 016260      4.9 5$:    ASL      MARK      ;UPDATE MARKER
1247 004506 005367 016252      4.9      DEC      MAX      ;CHECK FOR DONE
1248 004512 001356                BNE      3$      ;BRANCH IF NOT
1249
1250                                     ;LINK DL11C,D,E WITH THEIR ISR'S
1251
1252 004514 004767 000304      7.0 6$:    JSR      PC,      MOD10      ;MAKE SURE VECTOR ADR POINTER IS ON MOD10
1253 004520 004537 004572      5.8      JSR      %5,%LINK      ;LINK DL'S WITH ISR'S
1254 004524 017742                .WORD    DL.RCV      ;RECEIVER ISR ADR
1255 004526 017652                .WORD    DL.XMT      ;TRANSMITTER ISR ADR
1256
1257                                     ;LINK DJ11'S WITH THEIR ISR'S
1258
1259
1260 004530 004767 000270      7.0      JSR      PC,      MOD10      ;MAKE SURE VECTOR ADR POINTER IS ON MOD10
1261 004534 012704 001302      3.8      MOV      #SW.DJ,R4      ;DEVICE SELECTED MAP POINTER MOD10
1262 004540 004537 004572      5.8      JSR      %5,%LINK      ;LINK DJ'S WITH ISR'S
1263 004544 020344                .WORD    DJ.RCV      ;RECEIVER ISR ADR
1264 004546 020224                .WORD    DJ.XMT      ;TRANSMITTER ISR ADR
1265
1266                                     ;*****
1267                                     ;      MOV      #SW.NEXT,R4
1268                                     ;      JSR      PC,      %LINK
1269                                     ;*****
1270 004550 004767 000250      7.0      JSR      PC,      MOD10      ;MAKE SURE VECTOR ADD POINTER IS ON MOD10

```

```

1271
1272 004554 010067 174716 4.9 MOV RO,ENDVEC ;STORE END OF VECTOR AREA
1273 004560 010037 001026 3.7 FVECTX: MOV RO,#FVECT ;SET NXT AVAILABLE FLOATING VECTOR ADDRESS GLOBA
1274 004564 010167 174710 4.9 MOV RI,ENDLNK ;STORE END OF LINKER AREA
1275 004570 000207 3.5 RTS PC ;RETURN TO MAIN LINE
1276
1277
1278 ;LINK, ROUTINES TO LINK DEVICE VECTORS WITH THEIR ISR'S
1279
1280 004572 005067 016172 4.9 LINK: CLR LINE ;SET LINE EQUAL TO ZERO
1281 004576 012567 000304 6.4 MOV (R5)+, RCVISR ;FETCH ADRS OF RCV ISR ADRS
1282 004602 012567 000302 6.4 MOV (R5)+, XMTISR ;FETCH ADRS OF XMIT ISR ADRS
1283 004606 012267 000300 6.4 MOV (R2)+, PRTLVL ;GET PRIORITY LEVEL
1284 004612 012367 016146 6.4 MOV (R3)+, MAX ;FETCH NUMBER OF DEVICES ON THE SYSTEM
1285 004616 001465 2.6 BEQ LNK15 ;BRANCH IF NONE
1286 004620 012767 000001 016140 6.4 LNK1: MOV #1,MARK ;SET UP LINE POINTER
1287 004626 036714 016134 6.5 LNK2: BIT MARK, (R4) ;WAS THIS LINE SELECTED
1288 004632 001003 2.6 BNE LNK3 ;YES
1289 004634 062700 000010 3.8 ADD #10,RO ;NO, INC TO NEXT VECTOR ADRS
1290 004640 000404 2.6 BR LNK4 ;CONTINUE
1291 004642 004767 000170 7.0 LNK3: JSR %7,CODRCV ;LOAD LINKING CODE
1292 004646 004767 000210 7.0 JSR %7,CODXMT
1293 004652 005267 016112 4.9 LNK4: INC LINE ;INC TO NEXT LINE NUMBER
1294 004656 005367 016102 4.9 DEC MAX ;HAS MAXIMUM NUMBER BEEN CHECKED
1295 004662 003443 2.6 BLE LNK15 ;BRANCH TO EXIT IF FINISHED
1296 004664 006367 016076 4.9 ASL MARK ;SHIFT LINE POINTER
1297 004670 103356 2.6 BCC LNK2
1298 004672 005724 3.2 TST (R4)+ ;INC TO MAP 2
1299 004674 000751 2.6 BR LNK1 ;REINIT LINE POINTER
1300
1301 004676 005067 016066 4.9 LINK1: CLR LINE ;SET LINE EQUAL TO ZERO
1302 004702 012567 000200 6.4 MOV (R5)+, RCVISR ;FETCH ADRS OF RCV ISR ADRS
1303 004706 012267 000200 6.4 MOV (R2)+, PRTLVL ;GET PRIORITY LEVEL
1304 004712 012367 016046 6.4 MOV (R3)+, MAX ;FETCH NUMBER OF DEVICES ON THE SYSTEM
1305 004716 001425 2.6 BEQ LNK15 ;BRANCH IF NONE
1306 004720 012767 000001 016040 6.4 LNK11: MOV #1,MARK ;SET UP LINE POINTER
1307 004726 036714 016034 6.5 LNK12: BIT MARK, (R4) ;WAS THIS LINE SELECTED
1308 004732 001003 2.6 BNE LNK13 ;YES
1309 004734 062700 000004 3.8 ADD #4,RO ;NO, INC TO NEXT VECTOR ADRS
1310 004740 000402 2.6 BR LNK14 ;CONTINUE
1311 004742 004767 000070 7.0 LNK13: JSR %7,CODRCV ;LOAD LINKING CODE
1312 004746 005267 016016 4.9 LNK14: INC LINE ;INC TO NEXT LINE NUMBER
1313 004752 005367 016006 4.9 DEC MAX ;HAS MAXIMUM NUMBER BEEN CHECKED
1314 004756 003405 2.6 BLE LNK15 ;BRANCH TO EXIT IF FINISHED
1315 004760 006367 016002 4.9 ASL MARK ;SHIFT LINE POINTER
1316 004764 103360 2.6 BCC LNK12
1317 004766 005724 3.2 TST (R4)+ ;INC TO MAP 2
1318 004770 000753 2.6 BR LNK11 ;REINIT LINE POINTER
1319 004772 005724 3.2 LNK15: TST (R4)+
1320 004774 000205 3.5 RTS %5 ;RETURN
1321
1322 004776 004767 000022 7.0 SKIPVA: JSR PC, MOD10 ;MAKE SURE VECTOR ADR POINTER IS ON MOD10
1323 005002 012367 015756 6.4 MOV (R3)+,MAX
1324 005006 005367 015752 4.9 SKIPV1: DEC MAX
1325 005012 100402 2.6 BMI SKIPV2
1326 005014 061500 3.8 ADD (5),RO

```



```

1327 005016 000773      2.6      BR      SKIPV1
1328 005020 005725      3.2      SKIPV2: TST    (R5)+
1329 005022 000205      3.5      RTS      R5
1330
1331      ;SUBROUTINE TO UPDATE VECTOR ADDRESS POINTER (RO) TO A MOD10(8) BOUNDARY
1332 005024 062700 000004 3.8      MOD10: ADD     #4,    RO      ;INC BY 4
1333 005030 042700 000004 4.4      BIC     #4,    RO      ;MAKE MOD10(8)
1334 005034 000207      3.5      RTS      PC
1335
1336      ;CODE, ROUTINES TO FILL IN CODE FOR VECTOR AND LINKER AREA
1337
1338 005036 010120      3.7      CODRCV: MOV    R1,(RO)+  ;POINT RCV VECTOR TO LINKER
1339 005040 016720 000046 6.4      MOV    PRTLVL,(RO)+ ;SET UP DEVICE PRIORITY
1340 005044 012721 004537 5.2      MOV    #4537,(R1)+  ;LOAD LINKER WITH JSR
1341 005050 016721 000032 6.4      MOV    RCVISR,(R1)+ ;LOAD LINKER WITH DESTINATION
1342 005054 016721 015710 6.4      MOV    LINE,(R1)+  ;LOAD LINKER WITH LINE #
1343 005060 000207      3.5      RTS      %7
1344
1345 005062 010120      3.7      CODXMT: MOV    R1,(RO)+  ;POINT XMT VECTOR TO LINKER
1346 005064 016720 000022 6.4      MOV    PRTLVL,(RO)+ ;SET UP DEVICE PRIORITY
1347 005070 012721 004537 5.2      MOV    #4537,(R1)+  ;LOAD LINKER WITH JSR
1348 005074 016721 000010 6.4      MOV    XMTISR,(R1)+ ;LOAD LINKER WITH DESTINATION
1349 005100 016721 015664 6.4      MOV    LINE,(R1)+  ;LOAD LINKER WITH LINE #
1350 005104 000207      3.5      RTS      %7
1351
1352 005106 000000      RCVISR: 0      ;TEMP RCV ISR ADR
1353 005110 000000      XMTISR: 0      ;TEMP XMT ISR ADR
1354 005112 000000      PRTLVL: 0      ;TEMP PRIORITY LEVEL
1355
1356      ;***** SYSGEN *****

```

```

1357
1358
1359
1360
1361
1362
1363
1364
1365
1366 005114 000240 1.5 DEVICE: NOP
1367 005116 012737 005140 000004 5.2 MOV #XMISR, R4 ;TRAP TO XMISR ON TIMEOUT
1368 005124 016700 174350 5.0 MOV ENDLNK, R0 ;FETCH BEGINNING OF FREE MEMORY
1369 005130 010067 174346 4.9 MOV R0, ENDTAB ;SET UP END OF TABLE POINTER
1370 005134 005720 3.2 TST (R0)+ ;TIMEOUT AT END OF MEMORY
1371 005136 000776 2.6 BR -2
1372
1373 ;ENTER HERE ON TIMEOUT
1374
1375 005140 022626 4.7 XMISR: CMP (SP)+, (SP)+ ;ADJUST STACK
1376 005142 012737 000006 000004 5.2 MOV #6, R4 ;REINSTATE TIME OUT TRAP
1377 005150 162700 001000 3.8 SUB #1000, R0 ;SAVE ROOM TO BOOT.
1378 005154 005737 000042 3.2 TST R4 ;CHECK FOR MONITOR
1379 005160 001410 2.6 BEQ XMHOOK ;BRANCH IF NONE
1380 005162 023767 000042 005310 5.9 CMP R4, PROEND
1381 005170 001404 2.6 BEQ XMHOOK
1382 005172 013700 000042 3.8 MOV R4, R0 ;42 HAS TOP OF CORE
1383 005176 162700 000200 3.8 SUB #200, R0 ;SAVE ROOM FOR LAST CORE TEST
1384 005202 010067 174276 4.9 XMHOOK: MOV R0, ENDCOR ;SAVE END OF USEABLE CORE
1385
1386 ;INITIALIZE DC11'S
1387
1388 005206 032767 000002 174032 6.5 DCINIT: BIT #FL.DC, SW.MAP ;CHECK FOR DC11'S
1389 005214 001433 2.6 BEQ KLINIT ;BRANCH IF NONE
1390 005216 016700 174264 5.0 MOV NO.DC, R0 ;NUMBER OF DC11'S ON SYSTEM
1391 005222 012701 001252 3.8 MOV #SW.DC1, R1 ;SELECTED MAP ADR.
1392 005226 016702 174346 5.0 MOV AD.DC, R2 ;DEVICE ADR
1393 005232 012703 013104 3.8 MOV #DCDATA, R3 ;DATA TABLE ADR.
1394 005236 012704 000001 3.8 1$: MOV #1, R4 ;MARKER
1395 005242 012723 000400 5.2 2$: MOV #400, (R3)+ ;INIT DATA TABLE
1396 005246 030411 2.6 BIT R4, (R1) ;CHECK FOR LINE SELECTED
1397 005250 001405 2.6 BEQ 3$ ;BRANCH IF NOT
1398 005252 012712 003130 5.2 MOV #3130, (R2) ;5-BIT, HIGH SPEED, INTENB RCV
1399 005256 012762 000534 000004 6.4 3$: MOV #534, 4(R2) ;1 STOP CODE, INT ENB XMT, HIGH SPEED MAIN
1400 005264 062702 000010 3.8 3$: ADD #10, R2 ;INC TO NEXT DEV ADR.
1401 005270 005300 2.6 DEC R0 ;CHECK FOR DONE
1402 005272 001404 2.6 BEQ KLINIT ;BRANCH IF DONE
1403 005274 006304 2.6 ASL R4 ;INC TO NEXT UNIT
1404 005276 103361 2.6 BCC 2$ ;BRANCH IF SAME MAP
1405 005300 005721 3.2 TST (R1)+ ;INC TO NEXT MAP
1406 005302 000755 2.6 BR 1$ ;BRANCH TO RESET MARKER
1407
1408 ;TEST KL11'S
1409
1410 005304 005767 173746 4.4 KLINIT: TST SW.KL ;CHECK FOR KL11'S SELECTED
1411 005310 001443 2.6 BEQ DPINIT ;BRANCH IF NONE
1412 005312 012701 013432 3.8 MOV #KLDATA, R1 ;KL DATA STORAGE

```



```

1413 005316 012767 000020 015440 6.4      MOV      #16.,MAX      ;8 WORDS TIMES 2 BYTES = 16
1414 005324 012721 000400          5.2      KLDAT:  MOV      #400,(R1)+    ;TWO BYTES "1" DATA
1415 005330 005367 015430          4.9      DEC      MAX        ;HAVE ALL DATA BUFFERS BEEN INITIALIZED
1416 005334 001373          2.6      BNE     KLDAT      ;NO
1417
1418 005336 016701 015436          5.0      MOV      KLRSD,R1    ;KL11 RCV STATUS, LINE 0
1419 005342 016702 015426          5.0      MOV      KLXSD,R2    ;KL11 XMT STATUS, LINE 0
1420 005346 012767 000001 015412 6.4      MOV      #1,MARK    ;SET UP MARKER
1421 005354 036767 015406 173674 7.7      KLCSR:  BIT      MARK,SW.KL ;ACTIVATE THIS TTY
1422 005362 001406          2.6      BEQ     KLOFF      ;NO
1423 005364 052711 000101          5.2      BIS     #101,DR1    ;INT ENB RDR, RDR ENB
1424 005370 056712 174300          6.4      BIS     KLXSET,DR2  ;INT ENB PUNCH MAINTENANCE MODE
1425 005374 005761 000002          4.4      TST     2(R1)      ;DBL-BUFFERED FLUSH DL' TYPE BRK/CHARS
1426 005400 062701 000010          3.8      KLOFF:  ADD     #10,R1  ;INC TO NEXT ADRS
1427 005404 062702 000010          3.8      ADD     #10,R2    ;INC TO NEXT ADRS
1428 005410 000241          1.5      CLC
1429 005412 006367 015350          4.9      ASL     MARK      ;HAVE ALL BEEN CHECKED
1430 005416 103356          2.6      BCC     KLCSR     ;NO
1431
1432          ;ACTIVATE DP11'S
1433
1434 005420 032767 000010 173620 6.5      DPINIT: BIT      #FL.DP, SW.MAP ;CHECK FOR DP11'S SELECTED
1435 005426 001461          2.6      BEQ     DMANIT    ;BRANCH IF NONE
1436 005430 012767 001260 000044 6.4      MOV      #SW.DP1,DP1A+4
1437 005436 012704 014122          3.8      MOV      #DPXMTDATA,R4 ;DP TRANSMITTER DATA
1438 005442 012703 014022          3.8      MOV      #DPRCVDATA,R3 ;DP RCV DATA FILE
1439 005446 012701 174770          3.8      MOV      #174770,R1   ;R1=DP RECEIVER CSR ADRS
1440 005452 012702 174774          3.8      MOV      #174774,R2   ;R2=DP TRANSMITTER CSR ADRS
1441 005456 052767 100000 015274 6.4      BIS     #BIT15,TMPDAT ;SET UP CONTROL FLAG
1442 005464 005167 015270          4.9      DP1B:  COM     TMPDAT
1443 005470 012767 000001 015270 6.4      MOV      #1,MARK
1444 005476 036737 015264 001260 6.5      DP1A:  BIT      MARK,#SW.DP1 ;TEST THIS DP LINE
1445 005504 001411          2.6      BEQ     DPIC      ;NO
1446 005506 052711 000105          5.2      BIS     #105,DR1   ;RCV INT ENB, MAINT MODE STRIP SYNC
1447 005512 116761 007334 000003 7.6      MOV     B,TSYNC,3(R1) ;LOAD SYNC BUFFER
1448 005520 052712 000312          5.2      BIS     #312,DR2   ;INIT TRANSMITTER STATUS
1449
1450          ;BIT7=DONE
1451          ;BIT6=INTERRUPT ENABLE
1452          ;BIT3=TRANSMIT SYNC ON INT
1453          ;BIT1=IDLE SYNC
1453 005524 105062 000003          4.9      DPIC:  CLRB     3(R2)  ;CLR SYNC EXT
1454 005530 062701 177770          3.8      ADD     #10,R1     ;INDEX TO NEXT RCV CSR
1455 005534 062702 177770          3.8      ADD     #10,R2     ;
1456 005540 005023          3.7      CLR     (R3)+     ;XMIT
1457 005542 012724 002000          5.2      MOV     #2000,(R4)+ ;CLEAR DP RCV DATA FILE
1458 005546 000241          1.5      CLC
1459 005550 006367 015212          4.9      ASL     MARK      ;INIT SYNC COUNT (4 IN HI-BYTE)
1460 005554 103350          2.6      BCC     DP1A     ;CLEAR CARRY
1461 005556 062767 000002 177716 6.4      ADD     #2,DP1A+4  ;HAS A BANK OF 16 BEEN INT'D
1462 005564 005767 015170          4.4      TST     TMPDAT    ;NO
1463 005570 100735          2.6      BMI     DP1B     ;YES, ENTER SECOND BANK
1464
1465          ;DM INITIALIZATION
1466
1467 005572 005767 173466          4.4      DMANIT: TST     SW.DMA  ;CHECK FOR DM11A'S SELECTED FOR TEST
1468 005576 001002          2.6      BNE     IS        ;BRANCH IF ANY

```

```

1469 005600 000167 000534      4.9      JMP      DNINIT      ;OTHERWISE SKIP
1470
1471 005604 112767 177777 021422  6.4 15:  MOVB    #-1,    BINCNT ;SET FIRST DATA BYTE TO ALL 1'S
1472 005612 004567 015230      7.0      JSR     %5,    CLEAR  ;CLEAR TABLE
1473 005616 014612      .WORD   DM.LDAT ;STARTING AT THIS ADDRESS
1474 005620 000120      .WORD   80.     ;THIS MANY WORDS
1475
1476
1477      ;AT THIS POINT THE FOLLOWING TABLES HAVE BEEN CLEARED
1478      ;DM.LDAT      ;DM DTAT LIMITS FOR EACH CHANNEL
1479      ;DM.CAT      ;ADDRESSES OF CURRENT ADDRESS TABLES
1480      ;DM.WCT      ;ADDRESSES OF WORD COUNT TABLES
1481      ;DM.TT      ;TUMBLE TABLE POINTERS
1482      ;DM.RCVDAT   ;RECEIVE DATA FILES
1483 005622 105067 173654      4.9      CLRB    ENDTAB    ;FORM MOD400(8) ADRS
1484 005626 105267 173651      4.9      INCB    ENDTAB+1
1485 005632 016700 173644      5.0      MOV     ENDTAB, R0 ;RO = ADRS OF FIRST CAT
1486 005636 016701 015144      5.0      MOV     DMAADR,R1 ;(R1)= ADRS OF DM CSR ADRS
1487
1488 005642 004537 006264      5.8      JSR     %5,2#BMOV ;BLOCK MOVE
1489 005646 006250      DMA.11 ;FROM THIS ADRS
1490 005650 014422      DM.R6   ;TO THIS ADRS
1491 005652 000003      3      ;THIS MANY WORD
1492
1493 005654 012702 014652      3.8      MOV     #DM.CAT,R2 ; (R2)=ADRS OF CAT ADRS'S
1494 005660 012703 014712      3.8      MOV     #DM.WCT,R3 ; (R3)=ADRS OF WCT ADRS'S
1495 005664 012704 014752      3.8      MOV     #DM.TT,R4 ; (R4)=TT POINTERS
1496
1497      ;ACTIVATE LINE 0 OF EACH CHANNEL AND TRANSMIT A
1498      ;CHARACTER OF ALL "1" S TO DETERMINE CHARACTER LENGTH
1499      ;OF EACH CHANNEL
1500
1501      ;NOTE:
1502      ;IF DEVICE FAILS TO RESPOND PROGRAM WILL HANG UP
1503
1504
1505 005670 012767 000001 015070  6.4      MOV     #1,MARK    ;SETUP CHANNEL ACTIVE POINTER
1506 005676 036737 015064 001264  6.5 DMA.0: BIT     MARK,2#SW.DMA ;TEST CHANNEL 0 FOR ACTIVATION
1507 005704 001437      2.6      BEQ     DMA.1     ;BRANCH IF CH 0 NOT SELECTED
1508
1509 005706 012710 027234      5.2      MOV     #BINCNT,2R0 ;LOAD CAT WITH ADRS OF BINARY COUNT
1510 005712 010012      3.7      MOV     R0,2R2     ;SAVE CAT ADRS IN CAT ADRS TABLE
1511 005714 062700 000040      3.8      ADD     #40,R0     ;(R0)=ADRS OF WORD COUNT TABLE (WCT)
1512 005720 010023      3.7      MOV     R0,(R3)+   ;SAVE WCT ADRS IN WCT ADRS TABLE
1513 005722 012710 177777      3.2      MOV     #-1,2R0   ;SET LINE 0 WORD COUNT = ONE
1514 005726 062700 000140      3.8      ADD     #140,R0   ;(R0)=TUMBLE TABLE ADRS
1515 005732 010024      3.7      MOV     R0,(R4)+  ;SAVE TT ADRS IN TT ADRS TABLE
1516 005734 012767 000100 014754  6.4 DMA.00: MOV     #100,CNT ;TT WORD COUNT
1517 005742 005020      3.7      CLR     (R0)+     ;CLEAR TT ENTRY
1518 005744 005367 014746      4.9      DEC     CNT      ;DEC TT WORD COUNT
1519 005750 001374      2.6      BNE    DMA.00    ;BRANCH IF TT NOT CLEARED
1520 005752 042721 000001      3.8      BIC     #BIT0,(R1)+ ;ZERO TT POINTER ,INC TO BAR
1521 005756 005721      3.2      TST    (R1)+     ;INC TO BCR
1522 005760 042721 177777      3.8      BIC     #177777,(R1)+ ;CLEAR ALL BREAK BITS
1523 005764 012221      3.2      MOV     (R2)+,(R1)+ ;LOAD BASE ADRS REG WITH CAT ADRS
1524 005766 052761 000105 177770  6.4      BIS     #105,-10(R1) ;SET RCV INTERRUPT ENABLE, MAINT,GO

```


1525	005774	052761	000001	177772	6.4		BIS	#BIT0, -6(R1)	;ACTIVATE LINE 0, BAR
1526	006002	000405			2.6		BR	DMA.2	
1527									
1528	006004	062701	000010		3.8	DMA.1:	ADD	#10,R1	;INC R1 TO NEXT CSR ADRS
1529	006010	005722			3.2		TST	(R2)+	;INC CAT ADRS TABLE POINTER
1530	006012	005723			3.2		TST	(R3)+	;INC WCT ADRS TABLE POINTER
1531	006014	005724			3.2		TST	(R4)+	;INC TT POINTER TABLE POINTER
1532									
1533	006016	006367	014744		4.9	DMA.2:	ASL	MARK	;SHIFT CH POINTER TO NEXT CH
1534	006022	103325			2.6		BCC	DMA.0	;CONTINUE UNTIL ALL 16 CH CHECKED
1535	006024	012701	014612		3.8		MOV	#DM.LDAT,R1	
1536	006030	012767	000001	014730	6.4		MOV	#1, MARK	
1537	006036	033767	022766	173220	6.5	DMA.3:	BIT	#MARK, SW.DMA	
1538	006044	001402			2.6		BEQ	DMA.13	
1539	006046	005711			3.2		TST	(R1)	
1540	006050	001776			2.6		BEQ	.-2	
1541	006052	005721			3.2	DMA.13:	TST	(R1)+	
1542	006054	006367	014706		4.9		ASL	MARK	
1543	006060	103366			2.6		BCC	DMA.3	
1544	006062	004567	000176		7.0		JSR	%5,BMOV	;REINSTATE RCV ISR
1545	006066	006256					DMA.12		
1546	006070	014422					DM.R6		
1547	006072	000003					3		
1548									
1549									
1550									
1551	006074	112767	000000	021132	6.4		MOVB	#0,BINCNT	;RESTORE DATA FILE
1552	006102	016701	014700		5.0		MOV	DMADRS,R1	;FIRST DM CSR ADRS
1553	006106	012702	014652		3.8		MOV	#DM.CAT,R2	;TABLE OF CURRENT ADRS TABLE
1554	006112	012703	014712		3.8		MOV	#DM.WCT,R3	;TABLE OF WORD COUNT TABLE
1555	006116	012705	014752		3.8		MOV	#DM.TT,R5	;TABLE OF TUMBLE TABLE
1556	006122	012704	014612		3.8		MOV	#DM.LDAT,R4	;WORD COUNT
1557	006126	012767	000001	014632	6.4		MOV	#1,MARK	
1558	006134	036767	014626	173122	7.7	DMA.10:	BIT	MARK,SW.DMA	;ACTIVATE THIS LINE?
1559	006142	001427			2.6		BEQ	DMA.8	
1560	006144	012267	000020		6.4		MOV	(R2)+,DMA.4	;CAT ADRS
1561	006150	012367	000026		6.4		MOV	(R3)+,DMA.5	;WCT ADRS
1562	006154	005114			3.7		COM	#R4	
1563	006156	012467	000016		6.4		MOV	(R4)+,DMA.6	;WCT
1564									
1565	006162	004567	000124		7.0		JSR	%5,LOAD	;LOAD
1566	006166	027234					BINCNT		;BINARY COUNT ADRS
1567	006170	000000				DMA.4:	OPEN		;INTO CAT
1568	006172	000020					16.		
1569									
1570	006174	004567	000112		7.0		JSR	%5,LOAD	;LOAD
1571	006200	000000				DMA.6:	OPEN		;DATA LIMIT INTO
1572	006202	000000				DMA.5:	OPEN		;WORD COUNT TABLE (WCT)
1573	006204	000020					16.		;WORT COUNT TABLE (WCT)
1574									
1575	006206	012721	010105		5.2		MOV	#010105,(R1)+	;RCV+XMT INT ENB,MAINT,GO
1576	006212	052721	177777		5.2		BIS	#177777,(R1)+	;ACTIVATE ALL LINES
1577	006216	022121			4.7		CMP	(R1)+,(R1)+	;INC TO NEXT CSR
1578	006220	000405			2.6		BR	DMA.9	
1579									
1580	006222	005722			3.2	DMA.8:	TST	(R2)+	;INC TO NEXT CAT ADRS

;DM11 NPR CORE INITIALIZATION

```

1581 006224 005723          3.2          TST      (R3)+          ;INC TO NEXT WCT ADRS
1582 006226 005724          3.2          TST      (R4)+          ;INC TO NEXT WCT
1583 006230 062701 000010  3.8          ADD      #10,R1         ;INC TO NEXT CSR
1584 006234 006367 014526  4.9 DMA.9:  ASL      MARK          ;SHIFT CH POINTER TO NEXT CH
1585 006240 103335          2.6          BCC     DMA.10         ;BRANCH IF NOT END CH
1586 006242 010067 173234  4.9          MOV     RO, ENDTAB     ;SAVE END OF CODE
1587 006246 000434          2.6          BR      DNINIT        ;GO SET UP NON-COMM DEVICES
1588
1589
1590
1591
1592
1593
1594 006250 111561 014612  6.4 DMA.11: MOVB   @R5,DM.LDAT(R1) ;LOAD RECEIVED DATA INTO
1595 006254 000421          2.6          BR      .+4           ;DATA LIMIT
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606 006256 121561 015012  5.9 DMA.12: CMPB   @R5,DM.RDAT(R1) ;NORMAL RUNNING CODE FOR RCV.ISR
1607 006262 001401          2.6          BEQ     .+4
1608
1609
1610
1611
1612
1613 006264 004767 014654  7.0 BMOV:  JSR     PC, SAVRG   ;SAVE REGISTERS
1614 006270 012501          3.8          MOV     (R5)+,R1       ;FETCH SOURCE ADRS
1615 006272 012502          3.8          MOV     (R5)+,R2       ;FETCH DESTINATION ADRS
1616 006274 012503          3.8          MOV     (R5)+,R3       ;FETCH WORD COUNT
1617 006276 012122          5.2 T.B1:  MOV     (R1)+,(R2)+ ;MOVE
1618 006300 005303          2.3          DEC     R3             ;DEC WORD COUNT
1619 006302 001375          2.6          BNE    T.B1           ;BRANCH IF BLOCK NOT MOVED
1620 006304 004767 014656  7.0          JSR     PC, RSTRG     ;RESTORE R0-R4
1621 006310 000205          3.5          RTS     %5            ;RETURN
1622
1623 006312 004767 014626  7.0 LOAD:  JSR     PC, SAVRG   ;SAVE REGISTERS
1624 006316 012501          3.8          MOV     (R5)+,R1       ;FETCH DATA
1625 006320 012502          3.8          MOV     (R5)+,R2       ;FETCH DESTINATION ADRS
1626 006322 012503          3.8          MOV     (R5)+,R3       ;FETCH WORD COUNT
1627 006324 010122          3.7 T.LOD1: MOV     R1,(R2)+    ;LOAD DATA
1628 006326 005303          2.3          DEC     R3             ;DEC WORD COUNT
1629 006330 001375          2.6          BNE    T.LOD1        ;BRANCH IF LOAD NOT COMPLETE
1630 006332 004767 014630  7.0          JSR     PC, RSTRG     ;RESTORE REG
1631 006336 000205          3.5          RTS     %5            ;RETURN
1632
1633
1634 006340 005767 172722  4.4 DNINIT: TST     SW.DN      ;ACTIVATE DN11'S
1635 006344 001446          2.6          BEQ     DXINIT        ;CHECK FOR ANY DN11'S SELECTED
1636 006346 012701 015242  3.8          MOV     #DNDATA,R1

```



```

1637 006352 012702 175200          3.8      MOV      #175200,R2
1638 006356 012767 000001 014402      6.4      MOV      #1,MARK
1639 006364 036737 014376 001266      6.5      DNPR1:  BIT      MARK,2#SW.DN
1640 006372 001415          2.6      BEQ      DNPR2
1641 006374 012704 000004          3.8      MOV      #4,R4 ;SET UP COUNTER
1642 006400 105021          3.7      DNPR4:  CLRB   (R1)+ ;CLEAR PONTER TABLE
1643 006402 012712 000111          5.2      MOV      #111,(R2) ;SET INTENB, MAINT, FCRQ
1644 006406 056722 006730          6.4      BIS      DNXMTD,(R2)+ ;MOVE DATA BITS INTO CSR
1645 006412 005304          2.3      DEC      R4 ;COUNT LINES
1646 006414 001371          2.6      BNE      DNPR4 ;BRANCH IF MORE
1647 006416 052762 000004 177770      6.4      BIS      #BIT2,-10(R2) ;SET MASTER INTERUPT ENABLE
1648 006424 000403          2.6      BR       DNPR3
1649 006426 022121          4.7      DNPR2:  CMP      (R1)+,(R1)+ ;ADD 4 TO R1
1650 006430 062702 000010          3.8      ADD      #10,%2
1651 006434 006367 014326          4.9      DNPR3:  ASL      MARK
1652 006440 103351          2.6      BCC      DNPR1
1653 006442 000407          2.6      BR       DXINIT
1654                                     ;DX11 INITIALIZATION
1655
1656 006444 000000          DX.OLF: 0 ;DX11 ON-LINE FLAGS
1657 006446 000000          SPW:    0 ;ADRS OF STATUS POINTER WORD
1658 006450 000000          TT:     0 ;ADRS OF TUMBLE TABLE
1659 006452 000000          TTTT:   0 ;TT TRACE TABLE
1660 006454 000000          DST:    0 ;ADRS OF DEVICE STATUS TABLE
1661 006456 000000          DXDAT:  0 ;ADRS OF DX NPR DATA FILE
1662 006460 000000          DCTA:   0 ;ADRS OF DEVICE CONTROL TABLE AREA
1663
1664 006462 005767 172606          4.4      DXINIT:  TST      SW.DX ;CHECK FOR ANY DX11'S
1665 006466 001002          2.6      BNE
1666 006470 000167 000534          4.9      JMP      DLINIT
1667
1668 006474 052767 001777 173000      6.4      DXI.0:  BIS      #1777,ENDTAB
1669 006502 005267 172774          4.9      INC      ENDTAB
1670 006506 016767 172770 177732      7.6      MOV      ENDTAB,SPW
1671 006514 062767 001000 172760      6.4      ADD      #1000,ENDTAB
1672 006522 016767 172754 177720      7.6      MOV      ENDTAB,TT
1673 006530 062767 001000 172744      6.4      ADD      #1000,ENDTAB
1674 006536 016767 172740 177706      7.6      MOV      ENDTAB,TTTT
1675 006544 062767 001000 172730      6.4      ADD      #1000,ENDTAB
1676 006552 016767 172724 177674      7.6      MOV      ENDTAB,DST
1677 006560 062767 000400 172714      6.4      ADD      #256,ENDTAB
1678 006566 016767 172710 177662      7.6      MOV      ENDTAB,DXDAT
1679 006574 062767 000400 172700      6.4      ADD      #256,ENDTAB
1680 006602 016701 177642          5.0      MOV      TT,R1 ;BOTTOM OF TT
1681 006606 005021          3.7      CL1:    CLR      (R1)+ ;CLEAR TT
1682 006610 020167 177640          4.4      CMP      R1,DST ;TEST FOR END OF TT TRACE TABLE
1683 006614 001374          2.6      BNE      CL1 ;BRANCH IF NOT END
1684 006616 016777 177624 010560      8.8      MOV      SPW,DXOS ;LOAD DX OFFSET REG
1685
1686                                     ;DST DEVICE STATUS TABLE SETUP
1687
1688                                     ;SET DST TO CU FIRST
1689
1690 006624 012701 000400          3.8      MOV      #256,R1 ;FILL DST WITH UC
1691 006630 016702 177620          5.0      MOV      DST,R2
1692 006634 112722 000002          5.2      DS.4:  MOVB   #UC,(R2)+

```

```

1693 006640 005301          2.3      DEC      R1
1694 006642 001374          2.6      BNE      DS.4
1695
1696                      ;SET SPW TO ALL UC ALSO
1697
1698 006644 016701 177576          5.0      MOV      SPW,R1 ;SPW IMMEDIATE UNIT CHECK
1699 006650 012702 000400          3.8      MOV      #256,R2
1700 006654 012721 000002          5.2      DS.5:  MOV      #UC,(R1)+
1701 006660 005302          2.3      DEC      R2
1702 006662 001374          2.6      BNE      DS.5
1703
1704 006664 016701 177564          5.0      DS.0:  MOV      DST,R1
1705 006670 012727 000020          5.2      MOV      #16.,(PC)+
1706 006674 000000          DS.1:  0
1707 006676 012702 017572          3.8      MOV      #CMD.STAT,R2
1708 006702 112221          5.2      DS.2:  MOVB    (R2)+,(R1)+
1709 006704 005367 177764          4.9      DEC      DS.1
1710 006710 001374          2.6      BNE      DS.2
1711
1712
1713
1714                      ;DX11 ON LINE INITIALIZATION
1715
1716 006712 005767 177526          4.4      DXOLI:  TST      DX.OLF ;TEST FOR OFF/ON LINE TEST
1717 006716 001527          2.6      BEQ      DXOFLL ;BRANCH IF OFFLINE
1718 006720 012767 177770 000022          6.4      MOV      #-DEVNUM,NUMDEV
1719
1720                      ;THE FOLLOWING CODE CHECKS MEMORY SIZE FOR 8K SYSTEMS
1721                      ;IT REDUCES THE NUMBER OF ONLINE DEVICES FOR TEST TO 1.
1722                      ;I.E. IF ONLINE AND 8K OF MEMORY SET NONL = 1
1723
1724 006726 032737 140000 001504          5.3      BIT      #140000,#ENDCOR ;TEST MORE THAN 8K OF MEMORY
1725 006734 001006          2.6      BNE      CX15 ;(YES) BRANCHES
1726
1727 006736 012737 000001 007174          5.2      MOV      #1,#NONL ;(NO) REDUCE NUMBER OF DEVICES ON LINE TO 1.
1728 006744 012727 177776          5.2      MOV      #177776,(PC)+
1729 006750 177770          NUMDEV: 177770
1730
1731 006752 005067 007232          4.9      CX15:  CLR      DEV ;START POLLING DEV 0
1732 006756 005067 007250          4.9      CLR      WAIT
1733 006762 016701 177462          5.0      MOV      TT,TTP ;INIT TT POINTER
1734 006766 010167 007234          4.9      MOV      TTP,SAVTTP ;SAVE TT POINTER
1735 006772 016767 172504 177460          7.6      MOV      ENDTAB,DCTA ;START OF DEV CNTL TBL ADRS
1736
1737                      ;INITIALIZE SOFTWARE TABLES
1738 007000 013702 007174          3.8      MOV      #NONL,T1 ;GET # OF DEVICES
1739 007004 016700 177450          5.0      MOV      DCTA,DCT ;START OF AREA
1740 007010 012703 017632          3.8      MOV      #DCTP,T2 ;START OF DCTP TABLE
1741
1742 007014 010023          3.7      INIT2:  MOV      DCT,(T2)+ ;STORE ADDRESS OF DCT
1743 007016 012704 000420          3.8      MOV      #DCTZ,T3 ;CLEAR OUT THIS DCT
1744 007022 105020          3.7      INIT1:  CLRB    (DCT)+
1745 007024 005304          2.3      DEC      T3
1746 007026 001375          2.6      BNE      INIT1
1747 007030 005302          2.3      DEC      T1 ;GO FOR NEXT DCT
1748 007032 001370          2.6      BNE      INIT2 ;TILL ALL DONE

```



```

1749 007034 010067 172442      4.9      MOV      DCT,ENDTAB
1750
1751
1752
1753                                ;PUT FIRST NONL DEVICES ONLINE
1754
1755 007040 013702 007174      3.8      MOV      @#NONL, T1      ;GET # OF DEVS
1756 007044 012703 017632      3.8      MOV      @DCTP, T2     ;START WITH DEV #0
1757 007050 017704 007154      6.2      MOV      @DCU, T3      ;GET 1ST SPW ENTRY
1758 007054 006304      2.3      ASL      T3
1759 007056 066704 177364      5.0      ADD      SPW, T3       ;OFFSET SPW
1760
1761 007062 012300      3.8      INIT3: MOV      (T2)+, DCT      ;GET DCT
1762 007064 012760 177777 000006      6.4      MOV      #-1, ONLF(DCT) ;SET FLAG
1763 007072 016724 177356      6.4      MOV      DST, (T3)+    ;SET UP SPW WITH DST
1764
1765 007076 005302      2.3      DEC      T1            ;DO NEXT TILL DONE
1766 007100 001370      2.6      BNE      INIT3
1767
1768                                ;SET DXCS ACCORDING TO MODE
1769 007102 012777 000001 010272      7.6      MOV      #1, @DXCS     ;DXRESET
1770 007110 112777 000100 177336      7.6      MOV      @SM, @DST     ;RESPONSE TO TIO IS SM FOR 2703
1771 007116 005077 010260      6.1      CLR      @DXCS         ;CLEAR OUT DXCS BITS
1772 007122 052777 020000 010252      7.6      BIS      @ENDEN, @DXCS
1773 007130 005767 007070      4.4      TST      MODE          ;2703?
1774 007134 001410      2.6      BEQ      INIT4         ;YES - JUST PUT ONLINE
1775 007136 052777 004000 010236      7.6      BIS      @BSYEN, @DXCS ;CUBSY ENABLED FOR 2848
1776 007144 042777 020000 010230      8.2      BIC      @ENDEN, @DXCS
1777 007152 105077 177276      6.1      CLRB    @DST          ;RESPONSE TO TIO IS 0 FOR 2848
1778
1779 007156 052777 001000 010216      7.6      INIT4: BIS      @ONLINA, @DXCS ;PUT DX ONLINE
1780 007164 052777 000100 010210      7.6      BIS      #100, @DXCS  ;INTEN
1781 007172 000416      2.6      BR      DLINIT
1782
1783
1784 007174 000004      NONL:   4              ;NUMBER OF DEVICES TO PUT ONLINE
1785
1786                                ;OFFLINE INITIALIZATION
1787
1788
1789 007176 012777 000001 010176      7.6      DXOFI: MOV      #1, @DXCS
1790 007204 004767 006324      7.0      JSR      PC, CLRDXD    ;CLEAR DX DATA FILE
1791 007210 012777 100525 010174      7.6      MOV      #100525, @DXMO ;LOAD BUS0 WITH DATA PATTERN
1792 007216 052777 000100 010156      7.6      BIS      #100, @DXCS
1793 007224 004767 006324      7.0      JSR      PC, DXGO
1794                                ;INITIALIZE DL11C, DL11D, DL11E
1795
1796 007230 032767 001000 172010      6.5      DLINIT: BIT      @FL.DLC, SW.MAP ;CHECK FOR DL11'S
1797 007236 001500      2.6      BEQ      DJINIT       ;BRANCH IF NONE
1798 007240 016700 172272      5.0      MOV      NO.DLC, R0    ;NUMBER OF DL11'S ON SYSTEM
1799 007244 012701 001276      3.8      MOV      @SW.DL1, R1   ;SELECTED MAP ADR
1800 007250 016702 172354      5.0      MOV      AD.DLC, R2    ;DEVICE ADR
1801 007254 012703 020114      3.8      MOV      @DLDATA, R3   ;DATA TABLE ADR
1802 007260 012704 000001      3.8      1$:  MOV      #1, R4    ;MARKER
1803 007264 012723 000400      5.2      2$:  MOV      #400, (R3)+ ;INIT DATA TABLE
1804 007270 030411      3.8      BIT      R4, (R1)     ;CHECK FOR LINE SELECTED

```



```

1846                                     .SBTTL          DJ11 INITIALIZATION
1847
1848
1849                                     ;INITIALIZE DJ11
1850
1851 007440 032767 004000 171600      6.5 DJINIT: BIT   #FL.DJ, SW.MAP ;CHECK FOR DJ11'S
1852 007446 001500                    2.6 BEQ   ANCNIT  ;BRANCH IF NONE
1853
1854 007450 004567 013372              7.0 JSR   %5,   CLEAR ;CLEAR DJ TABLES
1855 007454 030420                    .WORD DJ.XTBL ;START ADDRESS
1856 007456 000256                    .WORD 256 ;THIS MANY WORDS
1857
1858 007460 016767 172054 000156      7.6 MOV   NO.DJ, DJ.CNT ;NO. OF DJ'S
1859 007466 005000                    CLR   R0 ;INITIALIZE INDEX >R0
1860 007470 012701 001302              3.8 MOV   #SW.DJ, R1 ;SELECT MAP ADDRESS >R1
1861 007474 016702 172132              5.0 MOV   AD.DJ,  R2 ;DEVICE BASE ADDRESS >R2
1862 007500 012703 031022              3.8 MOV   #DJ.EXP,R3 ;EXPECTED DATA TABLE >R3
1863 007504 012704 000001              3.8 MOV   #1,    R4 ;DEVICE CURSOR >R4
1864
1865 007510 005067 011012              4.9 CLR   DJ.XTLY ;CLR XMIT TALLY COUNTER
1866 007514 005067 011010              4.9 CLR   DJ.RTLY ;CLR RECV TALLY COUNTER
1867
1868 007520 030411                    3.8 1$: BIT   R4,    (R1) ;CHECK CURRENT DEVICE SELECTED
1869 007522 001444                    2.6 BEQ   5$ ;NO BRANCHES
1870
1871
1872 007524 005067 000112              4.9 CLR   DJLINE
1873 007530 016062 001456 000004      7.6 MOV   SW.DJO(R0),4(R2) ;SET DJTCR LINE SCAN CONTROL (XMIT)
1874
1875 007536 000240                    1.5 NOP ;*****
1876
1877 007540 012767 000001 000100      6.4 MOV   #1,DJ.BNT ;LINE CURSOR
1878
1879 007546 005013                    3.7 2$: CLR   (R3) ;INITIALIZE
1880 007550 036760 000072 001456      7.7 BIT   DJ.BNT, SW.DJO(R0) ;CHECK CURRENT LINE SELECTED
1881 007556 001405                    2.6 BEQ   3$ ;BRANCH IF NOT
1882
1883 007560 016713 000056              6.4 MOV   DJLINE, (R3) ;SET LINE#
1884 007564 000313                    3.7 SWAB (R3) ;TO LINE FIELD 11 THRU 8
1885 007566 052713 100000              5.2 BIS   #BIT15, (R3) ;SET VALID DATA BIT
1886
1887 007572 005267 000044              4.9 3$: INC   DJLINE ;ADVANCE LINE INDEX
1888 007576 005723                    3.2 TST  (R3)+ ;ADVANCE TABLE INDEX
1889
1890 007600 006367 000042              4.9 ASL   DJ.BNT ;NEXT POSITION
1891 007604 103360                    2.6 BCC  2$ ;TEST LINE GROUP COMPLETED
1892
1893 007606 012712 150515              5.2 ; SET DJCSR REG BITS
1894 ; MOV   #150515,(R2) ;XMT XMT STATUS SCN RCV CLR MAIN RCV
1895 ; ;RDY INT ENABLE ENB INT MOS ENB ENB
1896 007612 062700 000002              3.8 4$: ADD   #2,    R0 ;ADVANCE INDEX
1897 007616 006304                    2.3 ASL   R4 ;DEVICE CURSOR
1898 007620 062702 000010              3.8 ADD   #10,   R2 ;ADVANCE TO NEXT DEVICE ADDRESS
1899 007624 005367 000014              4.9 DEC   DJ.CNT ;CHECK SCAN COMPLETED
1900
1901 007630 001333                    2.6 BNE  1$ ;NO BRANCHES

```

```

1902
1903 007632 000406          2.6          BR          ANCNIT          ;YES INITIALIZE ANCILLARIES
1904
1905 007634 062703 000040  3.8  SS:          ADD          #40,          R3          ;ADVANCE TABLE INDEX
1906 007640 000764          2.6          BR          4$          ;CONTINUE
1907
1908 007642 000000          DJLINE: 0          ;LINE INDEX
1909 007644 000000          DJ.CNT: 0          ;DEVICE COUNTER
1910 007646 000000          DJ.BNT: 0          ;LINE CURSOR
1911
1912
1913
1914          ;INITIALIZE ANCILLARIES
1915          ;NOTE:
1916          ; UNCONDITIONAL EXECUTION OF THE FOLLOWING
1917          ; CODE ASSURES THAT ALL DEVICES WILL HAVE RESPONDED
1918          ; DURING QUICK VERIFICATION SELECTION. (I.E. SWITCHES
1919          ; #11, #9, #8.)
1920
1921
1922          ;INITIALIZE DISK AND DECTAPE DATA FILES
1923
1924 007650 005067 013030          4.9  ANCNIT: CLR          FSEG          ;CLEAR SEGMENT COUNT
1925 007654 012767 024234 013026  6.4          MOV          #BUFF,DFILE ;INIT DATA FILE ADDRESS
1926 007662 012767 000001 013022  6.4  DATINT: MOV          #1,ONE ;MAKE ONE = 1
1927 007670 012767 000001 013010  6.4          MOV          #1,FDATA ;INIT FILE DATA
1928 007676 012767 000101 012776  6.4          MOV          #101,LIMIT ;INIT DATA LIMIT
1929 007704 016777 012776 012776  8.8  DI1:  MOV          FDATA,DFILE ;STORE DATA
1930 007712 062767 000002 012770  6.4          ADD          #2,DFILE ;INC DATA ADDRESS TO NEXT WORD
1931 007720 005167 012762          4.9          COM          FDATA ;1'S COMPLEMENT
1932 007724 016777 012756 012756  8.8          MOV          FDATA,DFILE ;STORE DATA
1933 007732 062767 000002 012750  6.4          ADD          #2,DFILE ;INC DATA ADDRESS TO NEXT WORD
1934
1935 007740 005167 012742          4.9          COM          FDATA ;1'S COMPLEMENT DATA
1936 007744 066767 012742 012734  7.6          ADD          ONE,FDATA ;ADD ONE (OR -1) TO DATA
1937 007752 026767 012730 012722  7.1          CMP          FDATA,LIMIT ;FIRST HALF OF DATA COMPLETE
1938 007760 001351          2.6          BNE          DI1 ;NO
1939 007762 005267 012716          4.9          INC          FSEG
1940 007766 022767 000006 012710  5.9          CMP          #6,FSEG ;ALL DATA FILES INITIALIZED
1941 007774 001414          2.6          BEQ          CTYNIT ;BRANCH IF DONE
1942 007776 032767 000001 012700  6.5          BIT          #1,FSEG ;DECREMENT DATA
1943 010004 001726          2.6          BEQ          DATINT
1944 010006 012767 177777 012666  6.4          MOV          #-1,LIMIT ;INIT LOWER LIMIT (-0) OF DATA
1945 010014 005367 012666          4.9          DEC          FDATA ;DEC TO 100
1946 010020 005167 012662          4.9          COM          FDATA ;FDATA=1'S COM OF 100
1947 010024 000727          2.6          BR          DI1
1948
1949          ;TEST CTY INITIALIZE
1950
1951 010026 032767 000001 171214  6.5  CTYNIT: BIT          #BIT0,SW.ANC ;TEST CTY
1952 010034 001430          2.6          BEQ          LPINIT ;NO
1953 010036 012737 020614 000060  5.2          MOV          #CTYR,#60
1954 010044 016737 171606 000062  6.4          MOV          BR.CTY,#62
1955 010052 012737 020534 000064  5.2          MOV          #CTYP,#64
1956 010060 016737 171572 000066  6.4          MOV          BR.CTY,#66
1957 010066 012767 000001 010626  6.4          MOV          #1,CTKDAT ;INIT KEYBOARD DATA

```


Year	Address 1	Address 2	Address 3	Address 4	Address 5	Address 6	Instruction	Comments
1958	010074	012767	000000	010510	6.4		MOV #0,CTPDAT	;INIT PUNCH DATA
1959	010102	052777	000100	012706	7.6		BIS #BIT6,ACTPS	;INT ENB PUNCH
1960	010110	052777	000101	012674	7.6		BIS #101,ACTKS	;INT ENBRDR, RDRENB
1961								
1962								
1963								;INITIALIZE LINE PRINTER
1964	010116	032767	000002	171124	6.5	LPINIT:	BIT #BIT1,SW.ANC	;TEST LINE PRINTER
1965	010124	001422			2.6		BEQ TCINIT	;NO
1966	010126	012737	020742	000200	5.2		MOV #LPVECT, @#200	;INITIALIZE INT VECTOR
1967	010134	016737	171520	000202	6.4		MOV BR.LP, @#202	;PRIORITY=4
1968	010142	016767	010564	010566	7.6		MOV LPSIZE, CHRLIN	;FIRST LINE CHARACTER COUNT
1969	010150	016767	010556	010556	7.6		MOV LPSIZE, LINGTH	;LINE LENGHT
1970	010156	012767	000040	010554	6.4		MOV #40,CARGEN	;FIRST CHARACTER
1971	010164	052777	000100	010534	7.6		BIS #100,@LPS	;INT ENB
1972								
1973								;INITIALIZE TC11 DECTAPE
1974								
1975	010172	032767	000004	171050	6.5	TCINIT:	BIT #BIT2,SW.ANC	;ACTIVATE DECTAPE
1976	010200	001431			2.6		BEQ RFINIT	
1977	010202	012777	021142	010672	7.6		MOV #FENDZ,@TCIV	
1978	010210	016777	171446	010666	8.8		MOV BR.TC,@TCPRT	
1979	010216	012701	021124		3.8		MOV #SAT,R1	;R1 FIRST TC COMMAND CONTROL WORD
1980	010222	042721	003400		5.8	USEL:	BIC #3400,(R1)+	;CLEAR UNIT SELECT NUMBER
1981	010226	022701	021140		3.8		CMP #REVRN,R1	;ALL CONTROL WORDS INITIALIZED
1982	010232	001373			2.6		BNE USEL	;NO
1983	010234	005737	000042		3.2		TST @#42	;CHECK ACT11 MONITOR
1984	010240	001403			2.6		BEQ NACT	;NO BRANCHES
1985								
1986	010242	052767	000400	010670	6.4		BIS #400, REVRN	;YES SELECT DEVICE #1
1987								
1988	010250	016777	010664	010612	8.8	NACT:	MOV REVRN,@TCCM	;REVERSE
1989								;READ BLOCK NUMBER
1990								;INTERRUPT ENABLE
1991								;DO
1992	010256	012767	000000	010626	6.4		MOV #0,TCFIRST	;FIRST BLOCK #
1993								;RF11 DISK?
1994								
1995	010264	032767	000010	170756	6.5	RFINIT:	BIT #BIT3,SW.ANC	;ACTIVATE RF12
1996	010272	001417			2.6		BEQ RKINIT	;NO
1997	010274	012737	021720	000204	5.2		MOV #RFISR,@#204	
1998	010302	016737	171356	000206	6.4		MOV BR.RF,@#206	
1999	010310	005067	011374		4.9		CLR SERV	;ZERO SERVICE QUEUE
2000	010314	005067	011372		4.9		CLR LOWADR	;CLEAR LSB'S OF DISK ADDRESS (RF11)
2001	010320	005067	011370		4.9		CLR UPADRS	;CLEAR MSB'S OF DISK ADDRESS (RF11)
2002	010324	052777	000100	011336	7.6		BIS #BIT6,@DCS	;DISK INTERRUPT ENABLE
2003								;ACTIVATE RK11 DISK
2004	010332	032767	000020	170710	6.5	RKINIT:	BIT #BIT4,SW.ANC	;TEST RK11
2005	010340	001432			2.6		BEQ RPINIT	
2006	010342	032777	000040	011762	7.7		BIT #40,@RKDS	;TEST WRITE PROTECT SWITCH
2007	010350	001401			2.6		BEQ .+4	;BYPASS
2008	010352	104000					HLT	;HALT WRITE PROTECTED
2009	010354	012737	022424	000220	5.2		MOV #RKISR,@#220	
2010	010362	016737	171300	000222	6.4		MOV BR.RK,@#222	
2011	010370	012767	043503	012112	6.4		MOV #43503,RKFUNCTION	;WRITE AND WRITE CHECK
2012	010376	016777	012112	011742	8.8		MOV LLIMIT,@RKBAR	
2013	010404	016777	012102	011732	8.8		MOV RKWORDCT,@RKWC	

2014	010412	012777	000000	011716	7.6	MOV	#0,ARKDAE	
2015	010420	012777	000103	011722	7.6	MOV	#103,ARKCSR	
2016								
2017								
2018	010426	032767	000040	170614	6.5	RPINIT: BIT	#BITS, SW.ANC	;CHECK FOR RP11 SELECTED
2019	010434	001435			2.6	BEQ	OVERLY	;BRANCH FOR NONE
2020	010436	012737	022536	000254	5.2	MOV	#RPISR, @#254	
2021	010444	016737	171220	000256	6.4	MOV	BR.RP, @#256	
2022	010452	012767	043503	012220	6.4	MOV	#43503, RPFUNC	;WRITE/WRITE CHECK DISK
2023	010460	016777	012030	012042	8.8	RPD: MOV	LLIMIT, @RPBAR	
2024	010466	016777	012204	012032	8.8	MOV	RPWORDCT, @RPWC	
2025	010474	012777	000015	012030	7.6	MOV	#00015, @RPCSR	;HOME
2026	010502	105777	012024		5.6	TSTB	@RPCSR	
2027	010506	100375			2.6	BPL	.-4	;CONTROL READY
2028	010510	005777	012006		5.6	TST	@RPDSR	
2029	010514	100375			2.6	BPL	.-4	;UNIT DRIVE READY
2030	010516	005077	012000		6.1	CLR	@RPDSR	;CLR ATTENTION
2031	010522	012777	000103	012002	7.6	MOV	#103, @RPCSR	
2032								
2033								
2034								
2035	010530	005737	001020		3.2	OVERLY: TST	@OVINIT	;TEST INITIALIZATION REQUIRED
2036	010534	001405			2.6	BEQ	PRIME	;NO BRANCHES
2037	010536	013737	001020	010546	5.2	MOV	@OVINIT, @#LINKOI+2	;YES SET LINKAGE JSR ENTRY
2038	010544	004737	001020		5.8	LINKOI: JSR	PC, @OVINIT	;LINKOI+2 = (OVINIT)
2039								
2040	010550	005737	001022		3.2	PRIME: TST	@OVPRIM	;TEST DEVICE PRIME REQUIRED
2041	010554	001405			2.6	BEQ	CORNIT	;NO BRANCHES
2042	010556	013737	001022	010566	5.2	MOV	@OVPRIM, @#LINKOP+2	;YES SET LINKAGE JSR ENTRY
2043	010564	004737	001022		5.8	LINKOP: JSR	PC, @OVPRIM	;LINKOP+2 = (OVPRIM)
2044								
2045								
2046								
2047	010570	005767	001150		4.4	CORNIT: TST	IOTSAV	;CHECK FOR OVERLAY
2048	010574	001403			2.6	BEQ	IS	;BRANCH IF NONE
2049	010576	012767	057400	170676	6.4	MOV	#57400, ENDTAB	;START OF MEMORY TEST ROUTINE NOW STARTS AFTER 0
2050	010604	013700	001504		3.8	IS: MOV	@ENDCOR, R0	;GET HIGHEST USEABLE CORE ADDRESS
2051	010610	013701	001502		3.8	MOV	@ENDTAB, R1	;GET END OF CORE USED BY PROGRAM
2052	010614	020001			2.3	CMP	R0, R1	;IS THERE ENOUGH ROOM?
2053	010616	103420			2.6	BLO	XMRTS	;IF NOT BRANCH
2054	010620	012702	010700		3.8	XMLOP1: MOV	#MEMTST, R2	;MOVE THE CODE BETWEEN
2055	010624	012221			5.2	XMLOP2: MOV	(R2)+, (R1)+	;MEMTST AND MEMEND 'TILL
2056	010626	026162	177776	177776	7.1	CMP	-2(R1), -2(R2)	;VALIDATE CORRECT VALUE STORED
2057	010634	001403			2.6	BEQ	XMLOP3	;OK BRANCHES
2058								
2059	010636	014103			5.0	MOV	-(R1), R3	;WAS TO R3
2060	010640	014204			5.0	MOV	-(R2), R4	;SHOULD BE TO R4
2061								
2062	010642	104004						
2063								
2064	010644	022702	010752		3.8	XMLOP3: CMP	#MEMEND, R2	;CORE IS FULL
2065	010650	001365			2.6	BNE	XMLOP2	
2066	010652	000240			1.5	NOP		;PATCH FOR TRACE
2067	010654	020100			2.3	CMP	R1, R0	;HOW ARE WE DOING ON MEMORY
2068	010656	101760			2.6	BLOS	XMLOP1	
2069	010660	012721	000137		5.2	XMRTS: MOV	#137, (R1)+	;SET UP JMP @# TO RETURN TO BCKGND


```

2095
2096
2097
2098
2099
2100
2101
2102
2103 010752 012767 000140 167015 6.4 BCKGND: MOV #140,PS ;SET PRIORITY TO LEVEL 3
2104 010760 012701 024234 3.8 ;MOV #BUFF,R1 ;COMMON OUTPUT BUFFER
2105 010764 012702 026234 3.8 ;MOV #INBFC,R2 ;TC11 INPUT BUFFER
2106 010770 012703 025234 3.8 ;MOV #INBFR,R3 ;RF11, INPUT BUFFER
2107 010774 021122 4.7 15: CMP (R1),(R2)+ ;TC11 DATA OK
2108 010776 001401 2.6 BEQ .+4 ;YES
2109 011000 104000 4.7 HLT ;NO, REPORT ERROR
2110 011002 022123 4.7 CMP (R1)+,(R3)+ ;RF11 DATA OK
2111 011004 001401 2.6 BEQ .+4 ;YES
2112 011006 104000 4.7 HLT ;NO, REPORT ERROR
2113 011010 022701 025234 3.8 CMP #BUFF+1000,R1 ;TESTED ENTIRE BUFFER
2114 011014 001367 2.6 BNE 15 ;NO
2115 011016 005067 170156 4.9 25: CLR ICNT ;SCOPE
2116 011022 005067 013042 4.9 CLR LAD ;DN11 BACKGROUND
2117 ;DNACT: TST SW.DN ;CHECK FOR DN11'S
2118 011026 005767 170234 4.4 ;BEQ DM11BB ;BRANCH IF NONE
2119 011032 001422 2.6 SCOPE
2120 011034 104400 3.8 MOV #DNDATA,R1 ;START AT END OF DN DATA TABLE
2121 011036 012701 015242 5.0 MOV AD.DN,R2
2122 011042 016702 170542 5.0 MOV NO.DN,R3
2123 011046 016703 170444 5.0 ASL R3 ;X2
2124 011052 006303 2.3 ASL R3 ;X4
2125 011054 006303 3.2 DNBK1: TSTB (R1)+ ;CHECK FOR ACTIVE FLAG
2126 011056 105721 2.6 BPL DNBK2 ;BRANCH IF NOT
2127 011060 100004 4.9 COMB -(R1) ;RESTORE DATA POINTER
2128 011062 105141 3.8 MOVB (R1)+,R4 ;RETRIEVE IT
2129 011064 112104 6.4 DNBK2: BIS DNMNTD(4),(2) ;TRANSMIT, INTERRUPT ENABLED
2130 011066 056412 015342 3.2 TST (R2)+
2131 011072 005722 2.3 DEC R3
2132 011074 005303 2.6 BGT DNBK1
2133 011076 003367
2134
2135
2136
2137 ;DN11-BB MODEM CONTROL MULTIPLEXER
2138 ;SCANNER LOGIC TEST
2139 ;INPUT 1'S INTO ALL SCANNER MEMORY LOCATIONS
2140 ;VERIFY THAT AN INTERRUPT OCCURS FOR EACH LINE
2141 004000 CLRSCN=4000
2142 002000 CLRMUX=2000
2143 001000 MAINT=1000
2144 000400 STEP=400
2145 000100 INTENA=100
2146 000040 SCNENA=40
2147 000200 DONE=200
2148 000020 BUSY=20
2149 011100 005767 170164 4.4 DM11BB: TST SW.DMB ;CHECK FOR DM11BB'S
2150 011104 001470 2.6 BEQ KGSTRT ;BRANCH IF NONE

```


2151	011106	104400				SCOPE		
2152	011110	005001			2.3	CLR	R1	
2153	011112	016702	170474		5.0	MOV	AD.DMB,R2	
2154	011116	012767	000001	011642	6.4	MOV	#1,MARK	
2155	011124	036767	011636	170136	7.7	DMBTOP: BIT	MARK,SW.DMB	
2156	011132	001447			2.6	BEQ	DMBBOT	
2157	011134	012712	006000		5.2	SCNT1: MOV	#CLRMUX+CLRSCN,(R2)	
2158	011140	032712	000020		5.3	BIT	#BUSY,(R2)	;CLEAR ALL MULTIPLEXER FLIPFLOPS
2159	011144	001374			2.6	BNE	.-6	
2160	011146	012700	000020		3.8	MOV	#16.,R0	;SET UP TO WRITE 1'S INTO
2161	011152	005012			3.7	CLR	(R2)	;ALL LINE ENABLE FLIPFLOPS
2162	011154	012762	000001	000002	6.4	SCNT1A: MOV	#1 2(R2)	;SET FLIPFLOP
2163	011162	052712	000400		5.2	BIS	#STEP,(R2)	;GO TO NEXT LINE
2164	011166	005300			2.3	DEC	R0	
2165	011170	001371			2.6	BNE	SCNT1A	
2166	011172	012705	171340		3.8	MOV	#171340,R5	
2167	011176	012700	000020		3.8	MOV	#16.,R0	
2168	011202	012712	001117		5.2	MOV	#MAINT+INTENA+17,(R2)	;SET INTERRUPT ENABLE AND MAINTENANCE MO
2169	011206	052767	000200	166562	6.4	SCNT1B: BIS	#200,PS	;INHIBIT INTERRUPTS
2170	011214	052712	000040		5.2	BIS	#SCNENA,(R2)	;START SCANNER
2171	011220	042767	000200	166550	7.0	BIC	#200,PS	
2172	011226	105712			3.2	TSTB	(R2)	
2173	011230	100375			2.6	BPL	.-4	
2174	011232	020512			3.2	CMP	R5,(R2)	
2175	011234	001401			2.6	BEQ	SCNT1D	
2176	011236	104000				HLT		;SCANNER LOGIC ERROR
2177	011240	042712	000240		5.8	SCNT1D: BIC	#SCNENA+DONE,(R2)	;CLEAR SCAN ENABLE AND DONE
2178	011244	005205			2.3	INC	R5	;UPDATE EXPECTED RESULT
2179	011246	005300			2.3	DEC	R0	;CONTINUE IF NOT DONE
2180	011250	001356			2.6	BNE	SCNT1B	
2181	011252	005721			3.2	DMBBOT: TST	(R1)+	
2182	011254	062702	000010		3.8	ADD	#10, R2	;UPDATE DM11BB ADDRESS POINTER
2183	011260	006367	011502		4.9	ASL	MARK	
2184	011264	103317			2.6	BCC	DMBTOP	
2185								
2186								
2187								
2188								
2189								
2190	011266	005767	170000		4.4	KGSTRT: TST	SW.KG	;CHECK FOR ANY KG11'S
2191	011272	001567			2.6	BEQ	BCKCOR	;SKIP IF NONE
2192	011274	104400				SCOPE		
2193	011276	005001			2.3	CLR	R1	
2194	011300	016702	170310		5.0	MOV	AD.KG,R2	
2195	011304	012767	000001	011454	6.4	MOV	#1,MARK	
2196	011312	036767	011450	167752	7.7	KGTOP: BIT	MARK,SW.KG	
2197	011320	001440			2.6	BEQ	KGEND	
2198	011322	012700	011444		3.8	KGTEST: MOV	#KGIP,R0	;INSRTUCTION TO BE EXECUTED
2199	011326	012703	011452		3.8	MOV	#KGD,R3	;TABLE OF DATA
2200	011332	012704	011512		3.8	MOV	#KGC,R4	;TABLE OF RESULTS
2201	011336	012712	000020		5.2	KGTA: MOV	#20,(R2)	;CLEAR BCC REGISTER
2202	011342	011012			5.2	MOV	(R0),(R2)	;SET UP CSR FROM INSTRUCTION TABLE
2203	011344	011362	000004		6.4	MOV	(R3),4(R2)	;SEND DATA TO DEVICE
2204								;START BCC COMPUTATION
2205	011350	105712			3.2	TSTB	(R2)	;TEST FOR DONE
2206	011352	100023			2.6	BPL	KGEND	; (NO) CONTINUE WITH NEXT

```

2207
2208 011354 016267 000002 000060 7.6 MOV 2(R2), KGTEMP ;DEVICE WILL BE REPORTED AS NOT RUNNING
2209 011362 021467 000054 5.9 CMP (R4), KGTEMP ;READ RESULTS OF BCC
2210 011366 001401 2.6 BEQ .+4 ;COMPARE EXPECTED AND RECEIVED RESULTS
2211 011370 104000 HLT ;BCC INCORRECT, ERROR
2212
2213 011372 022324 4.7 CMP (R3)+, (R4)+ ;ADVANCE DATA AND RESULT POINTERS
2214 011374 020327 011512 CMP R3, #KGCP ;END OF DATA
2215 011400 001356 2.6 BNE KGTA ;NO CONTINUE
2216 011402 012703 011452 MOV #KGDP, R3 ;RESET DATA POINTER
2217 011406 022067 000036 CMP (R0)+, KGC12 ;ADVANCE INSTRUCTION POINTER
2218 011412 001351 BNE KGTA
2219 011414 052761 000200 001342 BIS #FL.KG, MAP(R1) ;SET RUN MAP FLAG
2220 011422 005721 3.2 KGEND: TST (R1)+
2221 011424 062702 000010 ADD #10, R2 ;UPDATE KG11 ADDRESS POINTER
2222 011430 006367 011332 ASL MARK
2223 011434 103326 2.6 BCC KGTOP
2224 011436 000167 000210 JMP BCKCOR
2225
2226 011442 000000 KGTEMP: 0
2227 011444 000111 KGIP: 000111 ;CRC16 INSTRUCTION WORD
2228 011446 000115 KGCIT: 000115 ;CCITT INSTRUCTION WORD
2229 011450 000100 KGC12: 000100 ;CRC12 INSTRUCTION WORD
2230
2231 ;DATA TABLE FOR CRC TESTS
2232
2233 011452 000401 KGDP: 000401
2234 011454 177376 177376
2235 011456 001002 001002
2236 011460 176775 176775
2237 011462 002004 002004
2238 011464 175773 175773
2239 011466 004010 004010
2240 011470 173767 173767
2241 011472 010020 010020
2242 011474 167757 167757
2243 011476 020040 020040
2244 011500 157737 157737
2245 011502 040100 040100
2246 011504 137677 137677
2247 011506 100200 100200
2248 011510 077577 077577
2249
2250 ;RESULT TABLE FOR CRC16
2251
2252 011512 050300 KGCP: 050300
2253 011514 160301 160301
2254 011516 120600 120600
2255 011520 010601 010601
2256 011522 001403 001403
2257 011524 131402 131402
2258 011526 003006 003006
2259 011530 133007 133007
2260 011532 006014 006014
2261 011534 136015 136015
2262 011536 014030 014030

```


2263	011540	124031	124031
2264	011542	030060	030060
2265	011544	100061	100061
2266	011546	060140	060140
2267	011550	150141	150141

;RESULT TABLE FOR CCITT

2271	011552	004121	004121
2272	011554	174351	174351
2273	011556	010242	010242
2274	011560	160032	160032
2275	011562	020504	020504
2276	011564	150774	150774
2277	011566	041210	041210
2278	011570	131060	131060
2279	011572	102420	102420
2280	011574	072650	072650
2281	011576	001061	001061
2282	011600	171211	171211
2283	011602	002142	002142
2284	011604	172332	172332
2285	011606	004304	004304
2286	011610	174174	174174

;RESULT TABLE FOR CRC12

2289	011612	024051	024051
2290	011614	030061	030061
2291	011616	024421	024421
2292	011620	030411	030411
2293	011622	025041	025041
2294	011624	031071	031071
2295	011626	026401	026401
2296	011630	032431	032431
2297	011632	021001	021001
2298	011634	035031	035031
2299	011636	036001	036001
2300	011640	022031	022031
2301	011642	000000	000000
2302	011644	014030	014030
2303	011646	000000	000000
2304	011650	014030	014030

2306	011652	032737	001000	177570	5.3	BCKCOR:	BIT	#BIT9,	Q#SWR	;CHECK MEMORY TEST INHIBIT
2307	011660	001007			2.6		BNE	EXPND3		;SW<9> SET BYPASS MEMORY TEST
2308	011662	104400					SCOPE			
2309	011664	005000			2.3		CLR	RD		;START AT 0
2310	011666	005520			3.7	COREX:	ADC	(RD)+		;FAST READ/WRITE ON ALL MEMORY
2311	011670	001376			2.6		BNE	COREX		;CHECK FOR A HALT
2312	011672	026700	167606		5.0		CMP	ENDCOR, RD		;CHECK FOR END
2313	011676	101373			2.6		BHI	COREX		;BRANCH IF NOT
2314	011700	032737	000400	177570	5.3	EXPND3:	BIT	#BIT8,	Q#SWR	;CHECK MEMORY TEST INHIBIT
2315	011706	001003			2.6		BNE	OVERL		;SW<8> SET BYPASS MEMORY TEST
2316										
2317	011710	104400					SCOPE			
2318	011712	000177	167564		6.1		JMP	QENDTAB		;DO WORST NOISE TEST ON EXCESS MEMORY

;MEMORY TEST

```

2319
2320
2321
2322
2323
2324 011716 005767 000022 4.4
2325 011722 001411 2.6
2326 011724 013746 177776 6.4
2327 011730 012746 011746 6.4
2328 011734 012767 011742 012126 6.4
2329 011742 000137 3.7
2330
2331 011744 000000
2332
2333 011746 004567 011074 7.0
2334 011752 001310
2335 011754 000015
2336 011756 012700 001310 3.8
2337 011762 012702 001542 3.8
2338 011766 004567 000536 7.0
2339 011772 000002
2340 011774 004567 000530 7.0
2341 012000 000004
2342 012002 004567 000522 7.0
2343 012006 000010
2344 012010 004567 000514 7.0
2345 012014 000020
2346 012016 004567 000506 7.0
2347 012022 000040
2348 012024 004567 000500 7.0
2349 012030 000100
2350 012032 004567 000472 7.0
2351 012036 000200
2352 012040 012702 001570 3.8
2353 012044 004567 000460 7.0
2354 012050 000400
2355 012052 005767 174366 4.4
2356 012056 001403 2.6
2357 012060 052767 000001 167244 6.4
2358 012066 004567 000436 7.0
2359 012072 001000
2360 012074 004567 000430 7.0
2361 012100 004000
2362 012102 004567 010740 7.0
2363 012106 001342
2364 012110 000040
2365 012112 012701 001250 3.8
2366 012116 012702 001306 3.8
2367 012122 012703 000016 3.8
2368 012126 012767 030052 000070 6.4
2369 012134 022122 4.7
2370 012136 001457 2.6

```

; IOT TRAPS TO OVERLAY BACKGROUND
; IF ONE EXISTS

```

OVERL: TST IOTSAV ; TEST FOR BACKGROUND OVERLAY
        BEQ TABTRN ; NOT OVERLAY BRANCH
        MOV @#PS, -(R6) ; YES PUT CURRENT STATUS ON STACK
        MOV #TABTRN, -(R6) ; PUT RETURN ON STACK FOR RTI
        MOV #Z0, LAD ; INITIALIZE SCOPE LINKAGE
        JMP @ (R7)+ ; LINK TO BACKGROUND PSEUDO IOT

```

IOTSAV: 0 ; BACKGROUND OVERLAY START ADDRESS

```

TABTRN: JSR %5, CLEAR ; START HERE
        RU.DC1 ; THIS MANY WORDS
        MAPSIZ-2

```

```

        MOV #RU.DC1, R0
        MOV #MX.DC, R2
        JSR %5, TRANSP
        FL.DC
        JSR %5, TRANSP
        FL.KL
        JSR %5, TRANSP
        FL.DP
        JSR %5, TRANSP
        FL.DMA
        JSR %5, TRANSP
        FL.DN
        JSR %5, TRANSP
        FL.DMB
        JSR %5, TRANSP
        FL.KG
        MOV #MX.DX, R2
        JSR %5, TRANSP

```

FL.DX ; CHECK FOR DX ON LINE

TST DX.OLF ; FORCE SET DX RUN FLAG

```

        BEQ Z1
        BIS #1, RU.DX
        JSR %5, TRANSP
        FL.DLC
        JSR %5, TRANSP
        FL.DJ
        JSR %5, CLEAR

```

```

        MAP
        32.
        MOV #SW.ANC, R1
        MOV #RU.ANC, R2
        MOV #MAPSIZ-1, R3
        MOV #MSGMAP, TYDEV
        MONIT: CMP (R1)+, (R2)+
        BEQ MONIT1

```

; THE FOLLOWING CODE ALLOWS ADDITIONAL TIME FOR LOW SPEED
; DOUBLE BUFFERED DEVICES TO RESPOND ON FIRST PASS UNDER
; ACT11 (MONITOR MODE). I.E. QUICK VERIFICATION.

2371
2372
2373
2374


```

2375
2376 012140 005737 001204 3.2 TST @#PASSES ;TEST FIRST PASS
2377 012144 001014 2.6 BNE MONITX ;(NO) BRANCHES
2378
2379 012146 005727 3.2 TST (PC)+ ;MONITOR MODE
2380 012150 000000 MONITX: 0 ;STALL FLAG
2381 012152 001011 2.6 BNE MONITX ;(INACTIVE) BRANCHES
2382 012154 024142 7.1 CMP -(R1),-(R2) ;RESTORE
2383 012156 062737 000001 012150 5.2 13: ADD #1,@#MONITZ ;DELAY
2384
2385 012164 022737 177777 012150 4.7 CMP #177777,@#MONITZ ;HERE
2386 012172 001371 2.6 BNE 1$
2387
2388 012174 000757 2.6 BR MONIT ;TRY ONCE MORE
2389
2390 012176 005267 167000 4.9 MONITX: INC ERRORS
2391 012202 032737 020000 177570 5.3 BIT #BIT13, @#SWR ;CHECK FOR INHIBIT TYPEOUT
2392 012210 001032 2.6 BNE MONIT1
2393 012212 004767 000364 7.0 JSR PC, CTYDWN
2394 012216 000004 030022 TYPE, MSGDNR
2395 012222 000004 TYPE,
2396 012224 000000 TYDEV: 0
2397 ;
2398 012226 014105 5.0 MOV -(R1),TTY ;DEVICE NOT RUNNING
2399 012230 004767 011322 7.0 JSR PC,BITYP$ ;PUT -(R1) INTO TTY
2400 012234 000004 027642 TYPE, SPACE+3 ;TYPE -(R1) IN BITS
2401 012240 014205 5.0 MOV -(R2),TTY ;PUT -(R2) INTO TTY
2402 012242 004767 011310 7.0 JSR PC,BITYP$ ;TYPE -(R2) IN BITS
2403
2404
2405 012246 005737 000042 3.2 TST @#42 ;CHECK ACT11 MONITOR
2406 012252 001406 2.6 BEQ 2$ ;NO BRANCHES
2407
2408 012254 005112 3.7 COM (R2) ;EXTRACT DEVICE(S) NOT RUNNING MASK*
2409 012256 041211 5.8 BIC (R2),(R1) ;SHUT DOWN DEVICE(S) NOT RUNNING
2410 012260 005112 3.7 COM (R2) ;RESTORE ACTIVITY
2411
2412 012262 012737 177777 001024 5.2 MOV #177777,@#CLINK ;RESET MONITOR PASS CONTROL
2413
2414 012270 022122 4.7 2$: CMP (R1)+,(R2)+
2415 012272 004767 000336 7.0 JSR PC, CTYUP
2416 012276 062767 000020 177720 6.4 MONIT1: ADD #20, TYDEV ;UPDATE POINTER TO NEXT MESSAGE
2417 012304 005303 2.3 DEC R3
2418 012306 003312 2.6 BGT MONIT
2419 012310 005067 166772 4.9 CLR RU.ANC ;CLEAR ANC RUN FLAG
2420 012314 005267 166664 4.9 INC PASSES ;UPDATE PASS COUNTER
2421 012320 032737 020000 177570 5.3 BIT #BIT13, @#SWR ;CHECK FOR INHIBIT TYPEOUT
2422 012326 001024 2.6 BNE BELLI ;BRANCH IF SET
2423 012330 004767 000246 7.0 JSR PC, CTYDWN
2424 012334 000004 027634 PASEND: TYPE, RETURN ;TYPE CR-LF
2425 012340 016705 166640 5.0 MOV PASSES,TTY ;TYPE PASSES IN OCTAL
2426 012344 004767 011226 7.0 JSR PC,PRINTS ;AND SUPPRESS LEADING ZERO'S
2427 012350 000004 027743 TYPE, MSGPAS ;TYPE " PASSES, "
2428 012354 016705 166622 5.0 MOV ERRORS,TTY ;TYPE ERRORS IN OCTAL
2429 012360 004767 011212 7.0 JSR PC,PRINTS ;AND SUPPRESS LEADING ZERO'S
2430 012364 000004 027760 TYPE, MSGERR ;TYPE " ERRORS."

```

2431	012370	000004	027634			TYPE,	RETURN		;TYPE CR-LF
2432	012374	004767	000234		7.0	JSR	PC	CTYUP	
2433	012400	032767	002000	165162	6.5	BELL1: BIT	#BIT10,SWR		;CHECK FOR INHIBIT BELL
2434	012406	001006			2.6	BNE	BELL2		
2435	012410	004767	000166		7.0	JSR	PC	CTYDWN	
2436	012414	000004	000207			TYPE	BELL		;TYPE THE CHARACTER BELL
2437	012420	004767	000210		7.0	JSR	PC	CTYUP	
2438	012424	005737	000042		3.2	BELL2: TST	@#42		;TEST CONTENTS OF 42 (SOFT VECTOR)
2439	012430	001435			2.6	BEQ	HOOK		;BRANCH IF NO MONITOR
2440	012432	005737	001024		3.2	TST	@#CLINK		;TEST MONITOR CONTROL
2441	012436	001007			2.6	BNE	BELL3		;QUICK PASS BYPASS
2442	012440	005737	012722		3.2	TST	@#XLINK		;TEST REPORT FLAG
2443	012444	001404			2.6	BEQ	BELL3		;CLEARED BYPASS
2444	012446	004767	000216		7.0	JSR	PC	SYSRPT	;REPORT CONFIGURATION TESTED
2445	012452	005037	012722		3.7	CLR	@#XLINK		;CLEAR REPORT FLAG
2446	012456	005037	001024		3.7	BELL3: CLR	@#CLINK		;CLEAR MONITOR CONTROL
2447	012462	022737	040000	000042	4.7	CMP	#40000,@#42		;TEST PSEUDO MONITOR
2448	012470	001401			2.6	BEQ	IS		; (YES) OVERLAY BRANCHES
2449	012472	000005			1.5	RESET			; (NO) ACT11/DDT RESET
2450	012474	013700	000042		3.9	IS: MOV	@#42, %0		;INITIALIZE FOR
2451	012500	004710			5.8	PROEND: JSR	%7, (0)		;RETURN TO MONITOR
2452	012502	000240			1.5	NOP			;ROOM FOR OVERLAY
2453	012504	000240			1.5	NOP			
2454	012506	000240			1.5	NOP			
2455	012510	022737	040000	000042	4.7	CMP	#40000,@#42		;TEST PSEUDO MONITOR
2456	012516	001402			2.6	BEQ	HOOK		; (YES) BYPASS
2457	012520	004767	172370		7.0	JSR	PC,DEVICE		;RESTART DEVICES MONITOR RETURN
2458	012524	000167	176222		4.9	HOOK: JMP	BCKGND		;REPEAT
2459									
2460									
2461	012530	012503			3.8	TRANSP: MOV	(R5)+,R3		;GET FLAG
2462	012532	012267	010226		6.4	MOV	(R2)+,MAX		;GET MAX NO DEVICE
2463	012536	012701	001342		3.8	MOV	#MAP,R1		;GET MAP POINTER
2464	012542	012704	000001		3.8	TR1: MOV	#1,R4		;GET RUN MAP MARKER
2465	012546	030321			3.8	TR2: BIT	R3,(R1)+		;CHECK FOR FLAG
2466	012550	001403			2.6	BEQ	TR3		
2467	012552	050410			3.7	BIS	R4,(R0)		
2468	012554	050367	166524		4.9	BIS	R3,RU.MAP		
2469	012560	005367	010200		4.9	TR3: DEC	MAX		
2470	012564	001404			2.6	BEQ	TR4		
2471	012566	006304			2.3	ASL	R4		
2472	012570	103366			2.6	BCC	TR2		
2473	012572	005720			3.2	TST	(R0)+		
2474	012574	000762			2.6	BR	TR1		
2475	012576	005720			3.2	TR4: TST	(R0)+		
2476	012600	000205			3.5	RTS	R5		
2477									
2478	012602	032767	000001	166440	6.5	CTYDWN: BIT	#BIT0, SW.ANC		;CHECK FOR CTY
2479	012610	001410			2.6	BEQ	IS		;NO BRANCHES
2480	012612	042777	000100	010176	8.2	BIC	#BIT6, @CTPS		;CLEAR PUNCH INT ENB
2481	012620	004767	000036		7.0	JSR	PC, WAITER		;WAIT
2482	012624	042777	000100	010160	8.2	BIC	#BIT6, @CTKS		;CLEAR READER INT ENB
2483	012632	000207			3.5	IS: RTS	PC		
2484									
2485	012634	032767	000001	166406	6.5	CTYUP: BIT	#BIT0, SW.ANC		;CHECK FOR CTY
2486	012642	001406			2.6	BEQ	2\$;NO BRANCHES

2487	012644	052777	000100	010144	7.6		BIS	#BIT6,	@CTPS	
2488	012652	052777	000100	010132	7.6		BIS	#BIT6,	@CTKS	
2489	012660	000207			3.5	2\$:	RTS	PC		
2490										
2491	012662	000004	030412				WAITER:	TYPE,	PADDERS	;INSERT Padder CHARACTERS (377)
2492	012666	000207			3.5		RTS	PC		
2493										
2494	012670	004767	177706		7.0	SYSRPT:	JSR	PC,	CTYDWN	
2495	012674	000004	027665				TYPE,	MSGX		;OUTPUT DZQCA-F TESTED
2496	012700	000004	027714				TYPE,	MSG1		;SYSTEMS CONFIGURATION
2497	012704	012701	001250		3.8		MOV	#SW.ANC,R1		;CURRENT SELECTION TABLE
2498	012710	004767	010156		7.0		JSR	PC,	REPORT	;OUTPUT
2499	012714	004767	177714		7.0		JSR	PC,	CTYUP	
2500	012720	000207			3.5		RTS	PC		;RETURN
2501										
2502	012722	177777					XLINK:	177777		;ONE PASS CONTROL FLAG (ACTIVE)
2503										
2504										
2505	012724	000000					COUNT1:	0		
2506	012726	000000					COUNT2:	0		
2507										
2508										
2509										
2510										
2511										
2512										
2513										
2514										
2515										
2516										
2517										
2518										
2519										
2520										
2521	012730	010146			4.9	DC.XMT:	MOV	R1,-(SP)		;SAVE REGISTER 1 ON STACK
2522	012732	010246			4.9		MOV	R2,-(SP)		;SAVE REGISTER 2 ON STACK
2523	012734	011501			3.8		MOV	(R5),R1		
2524	012736	006301			2.3		ASL	R1		;FORM MOD(2) INDEX
2525	012740	010102			2.3		MOV	R1,R2		;DUPLICATE LINE NUMBER
2526	012742	006302			2.3		ASL	R2		;FORM MOD10(8) INDEX
2527	012744	006302			2.3		ASL	R2		
2528	012746	062702	174004		3.8		ADD	#174004,R2		;FORM DEVICE ADDRESS
2529	012752	105712			3.2		TSTB	@R2		;TEST FOR TRANS DONE INC TO BUFF
2530	012754	100401			2.6		BMI	.+4		;YES
2531	012756	104000					HLT			;NO. REPORT ERROR, PC, PS,
2532	012760	105261	013104		4.9	DCOUT:	INCB	DCDATA(R1)		;INCREMENT SAVE DATA
2533	012764	005722			3.2		TST	(R2)+		;INC TO XMIT BUFF
2534	012766	116112	013104		6.4		MOVB	DCDATA(R1),(R2)		;TRANSMIT DATA
2535	012772	000167	005216		4.9		JMP	CLEANUP		
2536										
2537										
2538										
2539	012776	010146			4.9	DC.RCV:	MOV	R1,-(SP)		;SAVE REGISTER 1 ON STACK
2540	013000	010246			4.9		MOV	R2,-(SP)		;SAVE REGISTER 2 ON STACK
2541	013002	011501			3.8		MOV	(R5),R1		
2542	013004	006301			2.3		ASL	R1		;FORM MOD(2) INDEX

2543	013006	010102			2.3	MOV	R1,R2	;DUPLICATE LINE NUMBER
2544	013010	006302			2.3	ASL	R2	;FORM MOD10(8) INDEX
2545	013012	006302			2.3	ASL	R2	
2546	013014	062702	174000		3.8	ADD	#174000,R2	;DEVICE-CSR ADDRESS
2547	013020	005712			3.2	TST	(R2)	;ERROR BIT SET?
2548	013022	100001			2.6	BPL	+.4	;NO
2549	013024	104000				HLT		
2550	013026	105712			3.2	TSTB	(R2)	;DONE
2551	013030	100401			2.6	BMI	+.4	;YES
2552	013032	104000				HLT		;NO, FALSE INTERRUPT
2553	013034	005722			3.2	TST	(R2)+	;INC TO DATA BUFF
2554	013036	121261	013105		5.9	CMPB	(R2),DCDATA+1(R1)	;TEST DATA
2555	013042	001401			2.6	BEG	+.4	
2556	013044	104000				HLT		;FALSE DATA
2557	013046	122761	000037	013105	5.9	CMPB	#37,DCDATA+1(R1)	;UPPER DATA LIMIT FOR 5 BIT ASCII
2558	013054	001403			2.6	BEG	DCRX1	;YEY
2559	013056	105261	013105		4.9	INCB	DCDATA+1(R1)	;NO INCREMENT DATA
2560	013062	000403			2.6	BR	DCRX2	
2561	013064	112761	000000	013105	6.4	DCRX1: MOVB	#0,DCDATA+1(R1)	;REINITIALIZE DATA
2562	013072	052761	000002	001342	6.4	DCRX2: BIS	#FL.DC,MAP(R1)	;SET FLAG TO INDICATE LINE SERVICED
2563	013100	000167	005110		4.9	JMP	CLEANUP	

;DC11 CSR AND DATA STORAGE

DCDATA:

2567	013104							;DC11 DATA, LINE 0
2568	013104	000000				0		;DC11 DATA, LINE 1
2569	013106	000000				0		;DC11 DATA, LINE 2
2570	013110	000000				0		;DC11 DATA, LINE 3
2571	013112	000000				0		;DC11 DATA, LINE 4
2572	013114	000000				0		;DC11 DATA, LINE 5
2573	013116	000000				0		;DC11 DATA, LINE 6
2574	013120	000000				0		;DC11 DATA, LINE 7
2575	013122	000000				0		;DC11 DATA, LINE 10
2576	013124	000000				0		;DC11 DATA, LINE 11
2577	013126	000000				0		;DC11 DATA, LINE 12
2578	013130	000000				0		;DC11 DATA, LINE 13
2579	013132	000000				0		;DC11 DATA, LINE 14
2580	013134	000000				0		;DC11 DATA, LINE 15
2581	013136	000000				0		;DC11 DATA, LINE 16
2582	013140	000000				0		;DC11 DATA, LINE 17
2583	013142	000000				0		;DC11 DATA, LINE 20
2584	013144	000000				0		;DC11 DATA, LINE 21
2585	013146	000000				0		;DC11 DATA, LINE 22
2586	013150	000000				0		;DC11 DATA, LINE 23
2587	013152	000000				0		;DC11 DATA, LINE 24
2588	013154	000000				0		;DC11 DATA, LINE 25
2589	013156	000000				0		;DC11 DATA, LINE 26
2590	013160	000000				0		;DC11 DATA, LINE 27
2591	013162	000000				0		;DC11 DATA, LINE 30
2592	013164	000000				0		;DC11 DATA, LINE 31
2593	013166	000000				0		;DC11 DATA, LINE 32
2594	013170	000000				0		;DC11 DATA, LINE 33
2595	013172	000000				0		;DC11 DATA, LINE 34
2596	013174	000000				0		;DC11 DATA, LINE 35
2597	013176	000000				0		;DC11 DATA, LINE 36
2598	013200	000000				0		


```

2599 013202 000000          0          ;DC11 DATA, LINE 37
2600
2601          ;KL11 TRANSMITTER (PUNCH) SERVICE ROUTINE
2602
2603          176504          KLXA=176504          ;KL11 TRANSMITTER ADDRESS
2604
2605 013204 010146          4.9  KL.XMT:  MOV    R1,-(SP)          ;SAVE REGISTER 1 ON STACK
2606 013206 010246          4.9          MOV    R2,-(SP)          ;SAVE REGISTER 2 ON STACK
2607 013210 011501          3.8          MOV    (R5),R1
2608 013212 006301          2.3          ASL   R1          ;FORM MOD(2) INDEX
2609 013214 010102          2.3          MOV   R1,R2          ;DUPLICATE LINE NUMBER
2610 013216 006302          2.3          ASL   R2          ;FORM MOD10(8) INDEX
2611 013220 006302          2.3          ASL   R2
2612 013222 105762 176504  4.4          TSTB  KLXA(R2)          ;"READY" SET
2613 013226 100401          2.6          BMI  .+4          ;YES
2614 013230 104000          2.6          HLT
2615 013232 032762 000100 176504  6.5          BIT   #100,KLXA(R2)    ;INT ENB, INC TO DATA BUFFER
2616 013240 001001          2.6          BNE  .+4          ;YES
2617 013242 104000          2.6          HLT
2618 013244 005722          3.2          TST  (R2)+          ;FALSE INT, POSSIBLE CROSS TALK
2619 013246 105261 013432  4.9          INCB  KLDATA(R1)       ;INC INDX TO DATA BUFFER
2620 013252 122761 000147 013432  5.9          CMPB  #147,KLDATA(R1)  ;INC DATA
2621 013260 001003          2.6          BNE  KLXT          ;UPPER LIMIT OF DATA
2622 013262 112761 000000 013432  6.4          MOVB  #0,KLDATA(R1)   ;NO
2623 013270 116162 013432 176504  7.6  KLXT:  MOVB  #0,KLDATA(R1) ;REINITIALIZE DATA
2624 013276 000167 004712  4.9          MOVB  KLDATA(R1),KLXA(R2) ;TRANSMIT DATA
2625          JMP   CLEANUP
2626          ;KL11, RECEIVER SERVICE ROUTINE
2627
2628          176502          KLRA=176502          ;KL11 RECEIVER BUFFER ADDRESS
2629
2630 013302 010146          4.9  KL.RCV:  MOV    R1,-(SP)          ;SAVE REGISTER 1 ON STACK
2631 013304 010246          4.9          MOV    R2,-(SP)          ;SAVE REGISTER 2 ON STACK
2632 013306 011501          3.8          MOV    (R5),R1
2633 013310 006301          2.3          ASL   R1          ;FORM MOD(2) INDEX
2634 013312 010102          2.3          MOV   R1,R2          ;DUPLICATE LINE NUMBER
2635 013314 006302          2.3          ASL   R2          ;FORM MOD10(8) INDEX
2636 013316 006302          2.3          ASL   R2
2637 013320 105762 176500  4.4          TSTB  KLRA-2(R2)       ;"DONE" SET
2638 013324 100401          2.6          BMI  .+4          ;YES
2639 013326 104000          2.6          HLT
2640 013330 032762 000100 176500  6.5          BIT   #100,KLRA-2(R2)  ;INT ENB SET
2641 013336 001001          2.6          BNE  .+4          ;YES
2642 013340 104000          2.6          HLT
2643 013342 105762 176502  4.4          TSTB  KLRA(R2)        ;FALSE INTERRUPT
2644 013346 001422          2.6          BEQ  KLR2          ;TEST FOR LEADER
2645 013350 126261 176502 013433  7.1          CMPB  KLRA(R2),KLDATA+1(R1) ;BRANCH IF LEADER
2646 013356 001405          2.6          BEQ  KLR3          ;NOT LEADER TEST FOR DATA
2647 013360 122762 000207 176502  5.9          CMPB  #207,KLRA(R2)   ;CORRECT DATA
2648 013366 001403          2.6          BEQ  KLR4          ;BELL
2649 013370 104000          2.6          HLT
2650 013372 105261 013433  4.9  KLR3:  INCB  KLDATA+1(R1)  ;YES
2651 013376 122761 000147 013433  5.9  KLR4:  CMPB  #147,KLDATA+1(R1) ;NO, REPORT DATA ERROR
2652 013404 001003          2.6          BNE  KLR2          ;INCREMENT DATA
2653 013406 112761 000001 013433  6.4  KLR1:  MOVB  #1,KLDATA+1(R1) ;TEST FOR UPPER LIMIT
2654 013414 005262 176500  4.9  KLR2:  INC   KLRA-2(R2)    ;BASE DATA
          ;START READER

```

```

2655 013420 052761 000004 001342 6.4 BIS #FL.KL,MAP(R1) ;SET RUN FLAG
2656 013426 000167 004562 4.9 JMP CLEANUP
2657
2658 ;KL11 TRANSMITTER DATA (BYTES)
2659
2660 013432 KLXB=KLDAPA
2661 013433 KLRB=KLDAPA+1
2662
2663 013432 KLDAPA:
2664 013432 000000 ;KL11 DATA, LINE 0
2665 013434 000000 ;KL11 DATA, LINE 1
2666 013436 000000 ;KL11 DATA, LINE 2
2667 013440 000000 ;KL11 DATA, LINE 3
2668 013442 000000 ;KL11 DATA, LINE 4
2669 013444 000000 ;KL11 DATA, LINE 5
2670 013446 000000 ;KL11 DATA, LINE 6
2671 013450 000000 ;KL11 DATA, LINE 7
2672 013452 000000 ;KL11 DATA, LINE 10
2673 013454 000000 ;KL11 DATA, LINE 11
2674 013456 000000 ;KL11 DATA, LINE 12
2675 013460 000000 ;KL11 DATA, LINE 13
2676 013462 000000 ;KL11 DATA, LINE 14
2677 013464 000000 ;KL11 DATA, LINE 15
2678 013466 000000 ;KL11 DATA, LINE 16
2679 013470 000000 ;KL11 DATA, LINE 17
2680 013472 000000 ;KL11 DATA, LINE 20
2681 013474 000000 ;KL11 DATA, LINE 21
2682 013476 000000 ;KL11 DATA, LINE 22
2683 013500 000000 ;KL11 DATA, LINE 23
2684 013502 000000 ;KL11 DATA, LINE 24
2685 013504 000000 ;KL11 DATA, LINE 25
2686 013506 000000 ;KL11 DATA, LINE 26
2687 013510 000000 ;KL11 DATA, LINE 27
2688 013512 000000 ;KL11 DATA, LINE 30
2689 013514 000000 ;KL11 DATA, LINE 31
2690 013516 000000 ;KL11 DATA, LINE 32
2691 013520 000000 ;KL11 DATA, LINE 33
2692 013522 000000 ;KL11 DATA, LINE 34
2693 013524 000000 ;KL11 DATA, LINE 35
2694 013526 000000 ;KL11 DATA, LINE 36
2695 013530 000000 ;KL11 DATA, LINE 37
2696
2697
2698 013532 010146 4.9 DP.RCV: MOV R1,-(SP) ;SAVE REGISTER 1 ON STACK
2699 013534 010246 4.9 MOV R2,-(SP) ;SAVE REGISTER 2 ON STACK
2700 013536 011501 3.8 MOV (R5),R1
2701 013540 006301 3.8 ASL R1 ;FORM MOD(2) INDEX
2702 013542 010102 3.8 MOV R1,R2 ;DUPLICATE LINE NUMBER
2703 013544 006302 3.8 ASL R2 ;FORM MOD10(8) INDEX
2704 013546 006302 3.8 ASL R2 ;
2705 013550 005402 3.8 NEG R2 ;
2706 013552 062702 174770 ADD #174770,R2 ;R2=RCV STATUS REG
2707 013556 032712 000200 BIT #BIT7,R2 ;DONE SET
2708 013562 001001 2.6 BNE .+4 ;
2709 013564 104000 HLT ;REPORT FALSE INTERRUPT
2710

```


2711	013566	116261	000002	014023	7.6	MOVB	2(R2),DPRCVDATA+1(R1)	:SAVE RCV DATA
2712	013574	126161	014023	014022	7.1	CMPB	DPRCVDATA+1(R1),DPRCVDATA(R1)	:CHECK DATA
2713	013602	001426			2.6	BEQ	DP.R1	:BRANCH IF DATA OK
2714	013604	126167	014023	001241	7.1	CMPB	DPRCVDATA+1(R1),	:SYNC+1 :DATA=SYNC PLUS ONE
2715	013612	001420			2.6	BEQ	DP.R2	:BRANCH IF SYNC
2716	013614	032762	040000	000004	6.5	BIT	#RCVORUN,4(R2)	:TEST XMT STATUS FOR RCV O'RUN
2717	013622	001001			2.6	BNE	+.4	:DATA CHECK INVALID IF O'RUN
2718								:THEREFORE, REMOVE ACTIVE, EXIT
2719	013624	104000				HLT		:REPORT DATA ERROR
2720	013626	105061	014022		4.9	CLRB	DPRCVDATA(R1)	:ZERO EXPECTED DATA
2721	013632	105061	014122		4.9	CLRB	DPXMTDATA(R1)	:CLEAR XMIT DATA
2722	013636	042762	160000	000004	7.0	BIC	#160000,4(R2)	:CLR O'RUN+ERRORS
2723	013644	052762	000010	000004	6.4	BIS	#RESYNC,4(R2)	:SET RESYNC FLAG
2724	013652	000405			2.6	BR	DP.R5	:EXIT
2725	013654	105261	014022		4.9	DP.R2: INCB	DPRCVDATA(R1)	:MAKE UP FOR SYNC STRIP
2726	013660	105261	014022		4.9	DP.R1: INCB	DPRCVDATA(R1)	:INC EXPECTED DATA
2727	013664	001002			2.6	BNE	DP.R4	:EXIT
2728	013666	042712	004000		5.8	DP.R5: BIC	#ACTIVE,DP.R2	:CLEAR ACTIVE
2729	013672	052761	000010	001342	6.4	DP.R4: BIS	#FL.DP,MAP(R1)	:SET DP RUN FLAG
2730	013700	000167	004310		4.9	JMP	CLEANUP	
2731								
2732								
2733								;DP TRANSMITTER ISR
2734	013704	010146			4.9	DP.XMT: MOV	R1,-(SP)	:SAVE REGISTER 1 ON STACK
2735	013706	010246			4.9	MOV	R2,-(SP)	:SAVE REGISTER 2 ON STACK
2736	013710	011501			3.8	MOV	(R5),R1	
2737	013712	006301			2.3	ASL	R1	:FORM MOD(2) INDEX
2738	013714	010102			2.3	MOV	R1,R2	:DUPLICATE LINE NUMBER
2739	013716	006302			2.3	ASL	R2	:FORM MOD10(8) INDEX
2740	013720	006302			2.3	ASL	R2	
2741	013722	005402			2.3	NEG	R2	:DP STATUS REGS ARE DESENDING
2742	013724	062702	174774		3.8	ADD	#174774,R2	:DP XMT STATUS
2743	013730	105712			3.2	TSTB	DP.R2	:TEST DONE BIT
2744	013732	100401			2.6	BMI	+.4	:BRANCH IF DONE
2745	013734	104000				HLT		:REPORT ERROR, FALSE INTERRUPT
2746	013736	032712	000010		5.3	BIT	#RESYNC,(R2)	:TEST FOR RESYNC
2747	013742	001012			2.6	BNE	DP.X2	:BRANCH IF IN SYNC
2748	013744	116162	014122	000002	7.6	MOVB	DPXMTDATA(R1),2(R2)	:TRANSMIT DATA
2749	013752	105261	014122		4.9	INCB	DPXMTDATA(R1)	:INC TRANSMIT DATA
2750	013756	001017			2.6	BNE	DP.X3	:BRANCH IF NOT DONE
2751	013760	052712	000010		5.2	BIS	#RESYNC,(R2)	:SET RE-SYNC BITS
2752	013764	000167	004224		4.9	JMP	CLEANUP	:EXIT
2753	013770	116262	177777	000002	7.6	DP.X2: MOVB	-1(R2),2(R2)	:XMIT SYNC CHAR
2754	013776	105361	014123		4.9	DECB	DPXMTDATA+1(R1)	:DEC SYNC COUNT
2755	014002	001005			2.6	BNE	DP.X3	:EXIT IF SYNC COUNT NOT ZERO
2756	014004	112761	000004	014123	6.4	MOVB	#4,DPXMTDATA+1(R1)	:REINIT SYNC COUNT
2757	014012	042712	000010		5.8	BIC	#RESYNC,(R2)	:CLEAR RESYNC FLAG
2758	014016	000167	004172		4.9	DP.X3: JMP	CLEANUP	
2759								
2760	014022							DPRCVDATA:
2761								:LOW BYTE FOR EXPECTED DATA, HI BYTE FOR RECIEVED DATA
2762	014022	000000					0	:DP11 RECEIVER DATA LINE 0
2763	014024	000000					0	:DP11 RECEIVER DATA LINE 1
2764	014026	000000					0	:DP11 RECEIVER DATA LINE 2
2765	014030	000000					0	:DP11 RECEIVER DATA LINE 3
2766	014032	000000					0	:DP11 RECEIVER DATA LINE 4


```

2823 014210 000000          0          :DP11 TRANSMITTER DATA LINE 33
2824 014212 000000          0          :DP11 TRANSMITTER DATA LINE 34
2825 014214 000000          0          :DP11 TRANSMITTER DATA LINE 35
2826 014216 000000          0          :DP11 TRANSMITTER DATA LINE 36
2827 014220 000000          0          :DP11 TRANSMITTER DATA LINE 37
2828
2829
2830          :DM11 TRANSMITTER ISR
2831          :ONE DM11 REPRESENTS ONE CHANNEL, 16 LINES/CHANNEL
2832
2833 014222 010146          4.9 DM.XMT: MOV R1,-(SP)          :SAVE REGISTER 1 ON STACK
2834 014224 010246          4.9          MOV R2,-(SP)          :SAVE REGISTER 2 ON STACK
2835 014226 011501          3.8          MOV (R5),R1
2836 014230 006301          2.3          ASL R1          :FORM MOD(2) INDEX
2837 014232 010102          2.3          MOV R1,R2          :DUPLICATE LINE NUMBER
2838 014234 006302          2.3          ASL R2          :FORM MOD10(8) INDEX
2839 014236 006302          2.3          ASL R2
2840 014240 062702 175000 3.8          ADD #175000,R2          :FORM DEVICE ADRS
2841 014244 032712 060000 5.3          BIT #60000,R2          :TEST FOR O'RUN OR TIMEOUT ERROR
2842 014250 001401          2.6          BEQ DM.X1          :BRANCH IF NO ERRORS
2843 014252 104000          2.6          HLT          :REPORT ERRORS
2844 014254 005722          3.2 DM.X1: TST (R2)+          :TEST "TRANSMITTER READY"
2845          :INC TO "BUFFER ACTIVE" REGISTER
2846 014256 100401          2.6          BMI DM.X2          :BRANCH IF READY
2847 014260 104000          2.6          HLT          :REPORT FALSE INTERRUPT
2848 014262 016205 000004 5.0 DM.X2: MOV 4(R2),R5          :FETCH BASE ADRS
2849 014266 012767 000001 000252 6.4          MOV #1,DMARK          :SET UP LINE POINTER
2850 014274 036712 000246 6.5 DM.X3: BIT DMARK,R2          :TEST LINE FOR REACTIVATION
2851 014300 001007          5.2          BNE DM.X4          :BRANCH IF CHANNEL STILL ACTIVE
2852 014302 012715 027234 5.2          MOV #BINCNT,R5          :REINIT BASE ADRS
2853 014306 016165 014612 000040 7.6          MOV DM.LDAT(R1),40(R5) :REINIT WCT
2854 014314 056712 000226 6.4          BIS DMARK,R2          :ACTIVE LINE
2855
2856 014320 005725          3.2 DM.X4: TST (R5)+          :INCREMENT DM CORE INDEX TO NEXT LINE
2857 014322 006367 000220 4.9          ASL DMARK          :SHIFT LINE POINTER TO NEXT LINE
2858 014326 103362          2.6          BCC DM.X3          :BRANCH UNLESS 16 LINES TESTED
2859 014330 042762 100000 177776 7.0          BIC #BIT15,-2(R2)      :CLEAR XMT DONE
2860 014336 000167 003652 4.9          JMP CLEANUP
2861
2862          :DM11 RECEIVER ISR
2863          (LINE0=17000)          :TUMBLE TABLE MASK TO DETERMINE LINE#
2864
2865 014342 010146          4.9 DM.RCV: MOV R1,-(SP)          :SAVE REGISTER 1 ON STACK
2866 014344 010246          4.9          MOV R2,-(SP)          :SAVE REGISTER 2 ON STACK
2867 014346 011501          3.8          MOV (R5),R1
2868 014350 006301          2.3          ASL R1          :FORM MOD(2) INDEX
2869 014352 010102          2.3          MOV R1,R2          :DUPLICATE LINE NUMBER
2870 014354 006302          2.3          ASL R2          :FORM MOD10(8) INDEX
2871 014356 006302          2.3          ASL R2
2872 014360 062702 175000 3.8          ADD #175000,R2          :FORM DEVICE ADDRESS
2873 014364 105712          3.2          TSTB R2          :TEST FOR RCV DONE
2874 014366 100401          2.6          BMI DM.R1          :BRANCH IF DONE
2875 014370 104000          2.6          HLT          :REPORT FALSE INTERRUPT
2876 014372 016105 014752 5.0 DM.R1: MOV DM.TT(R1),R5          :FETCH TUMBLE TABLE POINTER
2877 014376 005715          3.2 DM.R2: TST R5          :TEST FOR VALID ENTRY
2878 014400 100401          2.6          BMI DM.R3          :BRANCH IF VALID ENTRY

```

Line No.	Address	Count	Label	Instruction	Comment
2879	014402	104000		HLT	:REPORT INVALID ENTRY
2880	014404	011561	014552	DM.R3: MOV	:TEST FOR MORE ENTRIES
2881					:SAVE TT DATA
2882	014410	100045		BPL	:EXIT IF NO MORE ENTRIES
2883	014412	142761	000341 014553	BICB	:MASK NON LINE # BITS
2884	014420	001024		BNE	:BRANCH IF NOT LINE 0
2885					:*****
2886	014422	121561	015012	DM.R6: CMPB	:COMPARE RCV DATA WITH EXPECTED
2887	014426	001401		BEQ	:BRANCH IF DATA OK
2888					:*****
2889	014430	104000		HLT	:REPORT DATA ERROR LINE 0
2890	014432	005161	014612	COM	
2891	014436	026161	014612 015012	CMP	:TEST FOR DATA LIMIT
2892	014444	001006		BNE	:BRANCH IF NOT END OF DATA
2893	014446	012761	000000 015012	MOV	:REINITILIZE DATA
2894	014454	005161	014612	COM	
2895	014460	000404		BR	
2896	014462	005161	014612	DM.R7: COM	
2897	014466	005261	015012	INC	:INCREMENT EXPECTED DATA
2898	014472	005025		DM.R4: CLR	:CLEAR TT ENTRY
2899	014474	016267	000006 000046	MOV	:SAVE BASE ADRS
2900	014502	062767	000400 000040	ADD	:END OF TT
2901	014510	020567	000034	CMP	:TEST FOR END OF TT
2902	014514	001333		BNE	:BRANCH IF NOT END
2903	014516	162705	000200	SUB	:POINT R5 TO BOTTOM OF TT
2904	014522	000730		BR	:CONTINUE TO EXAMINE TT
2905	014524	010561	014752	DM.R5: MOV	:STORE TT POINTER
2906	014530	052761	000020 001342	BIS	:SET RUN FLAG
2907	014536	042712	000200	BIC	:CLEAR DONE
2908	014542	000167	003446	JMP	
2909					
2910					
2911	014546	000000		DMARK:	:SCRATCH PAD FOR LINE #
2912	014550	000000		DM.R8:	:SCRATCH PAD
2913	014552			TTDAT:	:SAVED TT DATA
2914	014552	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 0
2915	014554	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 1
2916	014556	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 2
2917	014560	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 3
2918	014562	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 4
2919	014564	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 5
2920	014566	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 6
2921	014570	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 7
2922	014572	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 10
2923	014574	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 11
2924	014576	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 12
2925	014600	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 13
2926	014602	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 14
2927	014604	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 15
2928	014606	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 16
2929	014610	000000			:DM11 TUMBLE TABLE DATA, CHANNEL 17
2930					
2931	014612			DM.LDAT:	:DATA LIMITS FOR EACH CHANNEL
2932	014612	000000			:DM11 DATA LIMIT, CHANNEL 0
2933	014614	000000			:DM11 DATA LIMIT, CHANNEL 1
2934	014616	000000			:DM11 DATA LIMIT, CHANNEL 2

3047	015130	010304		2.3		MOV	R3	R4	:SET UP " SHOULD BE"
3048	015132	142704	000200	4.4		BICB	#BIT7,	R4	:CLEAR "DONE"
3049	015136	006201		2.3		ASR	R1		:FROM
3050	015140	006201		2.3		ASR	R1		:UNIT #
3051	015142	104004				HLT+4			:DN11 "DONE" SET WHEN IT SHOULDN'T HAVE
3052									
3053	015144	026512	015354	5.9	1\$:	CMP	DNRCVD(5),	(2)	:CHECK DATA
3054	015150	001401		2.6		BEQ	2\$:BRANCH IF OK
3055	015152	104000				HLT			:DN11 ERROR (ADRS OF LINE CSR IN R2)
3056									
3057	015154	005012		3.7	2\$:	CLR	(2)		:CLR DN11 STATUS REGISTER
3058	015156	005725		3.2		TST	(5)+		:UPDATE POINTER OFFSET
3059	015160	022705	000010	3.8		CMP	#10,	%5	:CHECK FOR END OF DATA TABLE
3060	015164	002001		2.6		BGE	.+4		
3061	015166	005005		2.3		CLR	%5		
3062	015170	005105		2.3		COM	%5		
3063	015172	110561	015242	4.9		MOVB	%5	DNDATA(1)	
3064	015176	012712	000011	5.2		MOV	#11,	(2)	:SET MAINT. FCRO
3065	015202	005267	000160	4.9		INC	DNIFLG		: "SET" INTERRUPT SERVICED FLAG
3066	015206	005722		3.2	DN. I2:	TST	(2)+		
3067	015210	005201		2.3		INC	%1		
3068	015212	032702	000007	4.4		BIT	#7,	%2	:CHECK FOR LAST LINE
3069	015216	001336		2.6		BNE	DN. I1		
3070	015220	005767	000142	4.4		TST	DNIFLG		:CHECK FOR INTERRUPT SERVICED
3071	015224	001001		2.6		BNE	.+4		:BRANCH IF IT WAS
3072	015226	104000				HLT			:DN11 FALSE INTERRUPT
3073									
3074	015230	052762	000004 177770	6.4		BIS	#BIT2,	-10(2)	:MASTER INTERRUPT ENABLE
3075	015236	000167	002752	4.9		JMP	CLEANUP		
3076									

3077	015242					DNDATA:			:DATA TABLE POINTER TABLE
3078	015242	000				.BYTE	0		:DN11 DATA, LINE 0
3079	015243	000				.BYTE	0		:DN11 DATA, LINE 1
3080	015244	000				.BYTE	0		:DN11 DATA, LINE 2
3081	015245	000				.BYTE	0		:DN11 DATA, LINE 3
3082	015246	000				.BYTE	0		:DN11 DATA, LINE 4
3083	015247	000				.BYTE	0		:DN11 DATA, LINE 5
3084	015250	000				.BYTE	0		:DN11 DATA, LINE 6
3085	015251	000				.BYTE	0		:DN11 DATA, LINE 7
3086	015252	000				.BYTE	0		:DN11 DATA, LINE 10
3087	015253	000				.BYTE	0		:DN11 DATA, LINE 11
3088	015254	000				.BYTE	0		:DN11 DATA, LINE 12
3089	015255	000				.BYTE	0		:DN11 DATA, LINE 13
3090	015256	000				.BYTE	0		:DN11 DATA, LINE 14
3091	015257	000				.BYTE	0		:DN11 DATA, LINE 15
3092	015260	000				.BYTE	0		:DN11 DATA, LINE 16
3093	015261	000				.BYTE	0		:DN11 DATA, LINE 17
3094	015262	000				.BYTE	0		:DN11 DATA, LINE 20
3095	015263	000				.BYTE	0		:DN11 DATA, LINE 21
3096	015264	000				.BYTE	0		:DN11 DATA, LINE 22
3097	015265	000				.BYTE	0		:DN11 DATA, LINE 23
3098	015266	000				.BYTE	0		:DN11 DATA, LINE 24
3099	015267	000				.BYTE	0		:DN11 DATA, LINE 25
3100	015270	000				.BYTE	0		:DN11 DATA, LINE 26
3101	015271	000				.BYTE	0		:DN11 DATA, LINE 27
3102	015272	000				.BYTE	0		:DN11 DATA, LINE 30


```

3159 015370 010146 4.9 DMBISR: MOV R1, -(SP) ;SAVE R1
3160 015372 010246 4.9 MOV R2, -(SP) ;SAVE R2
3161 015374 011501 3.8 MOV (R5), R1 ;GET UNIT NUMBER
3162 015376 006301 2.3 ASL R1 ;*2
3163 015400 010102 2.3 MOV R1, R2 ;ALSO IN R2
3164 015402 006302 2.3 ASL R2 ;*4
3165 015404 006302 2.3 ASL R2 ;*10
3166 015406 063702 001612 3.8 ADD @#AD.DMB,R2 ;SET UP DM11BB CSR ADDRESS
3167 015412 105712 3.2 TSTB (R2) ;IS DONE BIT SET
3168 015414 100401 2.6 BMI .+4
3169 015416 104000 HLT ;INTERRUPT WITH DONE NOT SET
3170 015420 052761 000100 001342 6.4 BIS #FL.DMB,MAP(R1)
3171 015426 000167 002562 4.9 JMP CLEANUP

; DX11 DEFINITIONS
CE=10
DE=4
TIOC=400 ;TEST I/O
WRITEC=001 ;WRITE
READC=002 ;READ
NOPC=403 ;NOP
SENSEC=4 ;SENSE
ILLC=405 ;ILLEGAL COMMAND
UC=2

DEVNUM=8.
MAX.DEV.CU: DEVNUM

ERRDST: UC

;OFFLINE INTERRUPT SERVICE ROUTINE
;DUMMY DX INTERRUPT SERVICE ROUTINE

3195 015436 104000 DMY.ISR: HLT ;FALSE DX INTERRUPT CHECK M7821
3196 015440 000002 4.8 RTI

3199 015442 010146 4.9 DX.ISR: MOV %1, -(6) ;SAVE REGISTERS
3200 015444 010246 4.9 MOV %2, -(6)
3201 015446 016701 171004 5.0 MOV DXDAT, %1
3202 015452 012702 000200 3.8 MOV #128, %2
3203 015456 022711 052525 4.7 DXDCK: CMP #52525, (1) ;CHECK DATA (TWO BYTES)
3204 015462 001404 2.6 BEQ DXDTE ;BRANCH IF OK
3205 015464 122741 000125 5.9 CMPB #125, -(1) ;RESTORE DISPLAY VALUE
3206 015470 104002 HLT+2
3207 015472 000416 2.6 BR DXOUT
3208 015474 005021 3.7 DXDTE: CLR (1)+ ;CLEAR CURRENT AND INCREMENT TO NEXT (TW
3209 015476 005302 2.3 DEC %2
3210 015500 001366 2.6 BNE DXDCK
3211 015502 042777 000004 001712 8.2 BIC #4,DXES ;CLEAR SOSIEN
3212 015510 042777 000200 001664 8.2 BIC #200,DXCS ;CLEAR DONE
3213 015516 004767 000032 7.0 JSR %7,DXGO ;START NEXT NPR
3214 015522 052767 000400 163612 6.4 BIS #FL.DX,MAP

```

```

3215 015530 000137 020214      3.7 DXOUT: JMP      @#CLEANUP
3216
3217
3218 015534 016701 170716      5.0 CLRDXD: MOV      DXDAT,R1
3219 015540 012702 000400      3.8      MOV      #256.,R2
3220 015544 105021      3.7 DS.3:  CLRB     (R1)+
3221 015546 005302      2.3      DEC      R2
3222 015550 001375      2.6      BNE     DS.3
3223 015552 000207      3.5      RTS
3224
3225 015554 016777 170676 001624      8.8 DXGO:  MOV      DXDAT,@DXBA      ;SET UP BASE ADRS REG
3226 015562 012777 177400 001620      7.6      MOV      #-256.,@DXBC      ;SET DXBC
3227 015570 052777 000003 001604      7.6      BIS      #3,@DXCS      ;SET GO
3228 015576 052777 060000 001606      7.6      BIS      #60000,@DXMO      ;CUI - SELO, HLDO
3229 015604 042777 060000 001600      8.2      BIC      #60000,@DXMO
3230 015612 052777 002000 001572      7.6      BIS      #2000,@DXMO      ;CMDO
3231 015620 042777 002000 001564      8.2      BIC      #2000,@DXMO      ;
3232
3233 015626 052777 000004 001566      7.6      BIS      #4,@DXES      ;SOSIEN FOR FAST NPR
3234 015634 000207      3.5      RTS      %7
3235
3236
3237 ;INTRRUP SERVICE ROUTINE FOR ONLINE OPERATION
3238 .REM *
3239 DCT=%0 ;CURRENT DCT BASE (BACKGROUND TASK)
3240 TTP=%1 ;CURRENT TUMBLE TABLE POINTER (INTERRUPT)
3241 T1=%2 ;WORK REGISTER (BACKGROUND)
3242 T2=%3 ;WORK REGISTER (BACKGROUND)
3243 T3=%4 ;WORK REGISTER (BACKGROUND)
3244 W=%5 ;DCT BASE (INTERRUPT)
3245 *
3246 ;NOTE:
3247 ; NO ERRORS ARE DETECTED BY CTP FOR ONLINE OPERATION
3248 ; AND THE DX IS NOT MONITORED FOR DEVICE ACTIVITY (RUNNING).
3249 ;
3250
3251 015636 052767 000400 163476      6.4 DXOL.ISR: BIS     #FL.DX,MAP
3252 015644 010527      3.7      MOV     %5,(PC)+
3253 015646 000000      3.7 R55:    O
3254 015650 010427      3.7      MOV     %4,(PC)+
3255 015652 000000      3.7 R45:    O
3256 015654 010327      3.7      MOV     %3,(PC)+
3257 015656 000000      3.7 R35:    O
3258 015660 010227      3.7      MOV     %2,(PC)+
3259 015662 000000      3.7 R25:    O
3260 015664 010127      3.7      MOV     %1,(PC)+
3261 015666 000000      3.7 R15:    O
3262 015670 010027      3.7      MOV     %0,(PC)+
3263 015672 000000      3.7 R05:    O
3264 015674 016701 000326      5.0      MOV     SAVTTP,TTP      ;RESTORE TTP
3265 015700 042777 000200 001474      8.2 RETRY: BIC     #DONE,@DXCS      ;CLEAR LOCK
3266 015706 011161 001000      6.4      MOV     (TTP),1000(TTP) ;TRACE T/T
3267 015712 016161 000002 001002      7.6      MOV     2(TTP),1002(TTP)
3268 015720 032711 010000      5.3 BIT     #SYSRST,@TTP      ;SYSTEM RESET?
3269 015724 001404      2.6 BEQ     INT1              ;NO
3270 015726 042777 000400 001446      8.2 BIC     #CUBSY,@DXCS

```


3271	015734	000466		2.6		BR	INTOUT		
3272									
3273	015736	147761	000266	000002	9.4	INT1:	BICB	DCU, CUART(TTP)	;CLEAR CU OUT OF DXCA
3274	015744	116105	000002		5.0		MOVB	CUART(TTP),W	
3275	015750	006305			2.3		ASL	W	
3276	015752	016505	017632		5.0		MOV	DCTP(W),W	
3277									
3278	015756	005767	000250		4.4		TST	WAIT	;PREVIOUS REQUEST?
3279	015762	001412			2.6		BEQ	INT2	;NO
3280									
3281	015764	010546			4.9		MOV	W, -(SP)	;NO - REGUEUE IT
3282	015766	116705	000216		5.0		MOVB	DEV, W	
3283	015772	006305			2.3		ASL	W	
3284	015774	016505	017632		5.0		MOV	DCTP(W),W	
3285	016000	016565	000002	000000	7.6		MOV	LREQ(W),REQ(W)	
3286	016006	012605			3.8		MOV	(SP)+, W	;BACK TO CURRENT DEV
3287									
3288	016010	005067	000216		4.9	INT2:	CLR	WAIT	;CLEAR WAIT FLAG

K07

MAINDEC-11-DZQCA-G
DZQCAG.P11

COMMUNICATION TEST PROGRAM (CTP)
DX11 INTERRUPT SERVICE ROUTINE

MACY11 27(732) 17-SEP-76 15:12 PAGE 68

3289	016014	005711	3.2	TST	(TTP)	:ANY ENTRY?	
3290	016016	001447	2.6	BEQ	CLEAN	:NO - CLEAN UP	
3291							
3292	016020	032711	004000	5.3	BIT	#INFDSC,(TTP)	:CHECK FOR HIO


```

3293 016024 001404          2.6      BEQ      DS1
3294 016026 004767 000772    7.0      JSR      PC,      DXDSC
3295 016032 000167 000054    4.9      JMP      INTOUT
3296 016036 032711 000102    5.3      DS1:    BIT      #ESEND!STKSTB,(TTP) ;CHECK FOR ES END
3297 016042 001404          2.6      BEQ      DS2
3298 016044 004767 000630    7.0      JSR      PC,      ESDONE
3299 016050 000167 000036    4.9      JMP      INTOUT
3300 016054 032711 000200    5.3      DS2:    BIT      #CHIS, (TTP) ;CHECK FOR ISS
3301 016060 001404          2.6      BEQ      DS3
3302 016062 004767 001044    7.0      JSR      PC,      DXISS
3303 016066 000167 000020    4.9      JMP      INTOUT
3304 016072 032711 000060    5.3      DS3:    BIT      #CUDEND!CHDEND,(TTP) ;CHECK FOR DT END
3305 016076 001404          2.6      BEQ      DS4
3306 016100 004767 000530    7.0      JSR      PC,      DTDONE
3307 016104 000167 000002    4.9      JMP      INTOUT
3308
3309 016110 000240          1.5      DS4:    NOP ;UNRECOGNIZED T/T ENTRY
3310
3311          INTOUT:
3312 016112 005011          3.7      CLR      @TTP ;CLEAR THIS ENTRY
3313 016114 062701 000004    3.8      ADD      #4,      TTP ;SET FOR NEXT
3314 016120 020167 170326    4.4      CMP      TTP,TTT ;END OF TT
3315 016124 002665          2.6      BLT      RETRY ;NO - OK
3316 016126 016701 170316    5.0      MOV      TT,TTP ;YES - START OVER
3317 016132 000167 177542    4.9      JMP      RETRY
3318
3319          ;EXIT FROM INTERRUPT
3320
3321          .SBTTL      DX11 BACKGROUND TASK
3322          ;DX BACKGROUND TASK
3323          ;POLL REQUEST LOOP
3324          .REM      *
3325          DCT=%0 ;CURRENT DCT BASE (BACKGROUND TASK)
3326          TTP=%1 ;CURRENT TUMBLE TABLE POINTER (INTERRUPT)
3327          T1=%2 ;WORK REGISTER (BACKGROUND)
3328          T2=%3 ;WORK REGISTER (BACKGROUND)
3329          T3=%4 ;WORK REGISTER (BACKGROUND)
3330          W=%5 ;DCT BASE (INTERRUPT)
3331
3332          *
3333 016136 005067 000046    4.9      CLEAN: CLR      DEV ;DO ALL POSSIBLE DEVICES
3334 016142 016702 000042    5.0      LOOP:  MOV      DEV, T1 ;GET DEVICE TO TEST
3335 016146 006302          2.3      ASL      T1 ;WORD INDEX
3336 016150 016200 017632    5.0      MOV      DCTP(T1),DCT ;SET UP DCT
3337 016154 016002 000000    5.0      MOV      REQ(DCT),T1 ;GET REQUEST CODE
3338 016160 006302          2.3      ASL      T1 ;WORD INDEX
3339 016162 004772 016212    8.2      JSR      PC,      @REQTBL(T1) ;DISPATCH TO HANDLER
3340
3341 016166 005267 000016    4.9      INC      DEV ;TO NEXT DEVICE
3342 016172 046767 170552 000010    8.2      BIC      NUMDEV,DEV ;MASK TO DESIRED LEVEL
3343 016200 005767 000004          4.4      TST      DEV ;ALL DONE?
3344 016204 001356          2.6      BNE      LOOP ;NO
3345 016206 000572          2.6      DXLP:  BR      SS4 ;EXIT
3346 016210 000000          2.6      DEV:   O ;CURRENT POLL
3347
3348 016212 016234          REQTBL: SNUL

```

MAINDEC-11-DZQCA-G
DZQCA.G.P11

COMMUNICATION TEST PROGRAM (CTP)
DX11 BACKGROUND TASK

M07
MACY11 27(732) 17-SEP-76 15:12 PAGE 70

3349	016214	016236
3350	016216	016336
3351	016220	016436
3352	016222	016522
3353		
3354	016224	177777
3355	016226	000000

SINPUT
SOUTPT
SSTAT
SSCUE
;MODE CONTROL
MODE: -1
SAVTP: 0

;0 FOR 2703, -1 FOR 2848
;SAVED TT POINTER

3378	016320	016077	000014	001062	8.8		MOV	BUFC(DCT), @DXBC	
3379	016326	052777	000003	001046	7.6		BIS	#FCTNR, @DXCS	;SET FUNCTION
3380	016334	000207			3.5	ST1:	RTS	PC	;DONE
3381									
3382	016336	005060	000000		4.9	SOUTPT:	CLR	REQ(DCT)	
3383	016342	026027	000004	000001	5.9		CMP	DEVSTA(DCT), #DATA	
3384	016350	001031			2.6		BNE	S01	
3385	016352	012767	000002	177652	6.4		MOV	#CLDT, WAIT	
3386	016360	116777	177624	001012	8.8		MOVB	DEV, @DXCA	;SET DEV
3387	016366	157777	177636	001004	10.0		BISB	@CU, @DXCA	
3388	016374	016077	000012	001004	8.8		MOV	BUFA(DCT), @DXBA	
3389	016402	012777	177774	001000	7.6		MOV	#-4, @DXBC	
3390	016410	026027	000014	177774	5.9		CMP	BUFC(DCT), #-4	
3391	016416	003403			2.6		BLE	+10	
3392	016420	016077	000014	000762	8.8		MOV	BUFC(DCT), @DXBC	
3393	016426	052777	000005	000746	7.6		BIS	#FCTNR, @DXCS	
3394	016434	000207			3.5	S01:	RTS	PC	
3395									
3396	016436	005060	000000		4.9	SSTAT:	CLR	REQ(DCT)	
3397	016442	026027	000004	000001	5.9		CMP	DEVSTA(DCT), #DATA	
3398	016450	001404			2.6		BEQ	SS1	
3399	016452	026027	000004	000002	5.9		CMP	DEVSTA(DCT), #STA	
3400	016460	001017			2.6		BNE	SS2	
3401	016462	012767	000001	177542	6.4	SS1:	MOV	#CLUES, WAIT	
3402	016470	116777	177514	000702	8.8		MOVB	DEV, @DXCA	;SET DEV
3403	016476	157777	177526	000674	10.0		BISB	@CU, @DXCA	
3404	016504	116077	000016	000672	8.8		MOVB	STS(DCT), @DXOS	
3405	016512	052777	000007	000662	7.6		BIS	#FCTNS, @DXCS	
3406	016520	000207			3.5	SS2:	RTS	PC	
3407									
3408									
3409	016522	005060	000000		4.9	SSCUE:	CLR	REQ(DCT)	
3410	016526	005760	000004		4.4		TST	DEVSTA(DCT)	
3411	016532	001017			2.6		BNE	SS3	
3412	016534	012767	000001	177470	6.4		MOV	#CLUES, WAIT	
3413	016542	116777	177442	000630	8.8		MOVB	DEV, @DXCA	
3414	016550	157777	177454	000622	10.0		BISB	@CU, @DXCA	
3415	016556	116077	000016	000620	8.8		MOVB	STS(DCT), @DXOS	
3416	016564	052777	000007	000610	7.6		BIS	#FCTNS, @DXCS	
3417	016572	000207			3.5	SS3:	RTS	PC	
3418	016574	010167	177426		4.9	SS4:	MOV	TTP, SAVTTP	
3419	016600	016705	177042		5.0		MOV	R55, %5	
3420	016604	016704	177042		5.0		MOV	R45, %4	
3421	016610	016703	177042		5.0		MOV	R35, %3	
3422	016614	016702	177042		5.0		MOV	R25, %2	
3423	016620	016701	177042		5.0		MOV	R15, %1	
3424	016624	016700	177042		5.0		MOV	R05, %0	
3425	016630	000137	020220		3.7		JMP	@#CLEUP1	
3426									
3427									
3428									
3429									
3430									
3431	016634								
3432	016634	032711	004040		5.3	DTDONE:	BIT	#CHDEND! INFDC, @TTP	;CH END?
3433	016640	001045			2.6		BNE	REGES	;YES - REQUEST STATUS

;DT END EXPECTED

3434	016642	032711	000020		5.3	BIT	#CUDEND, @TTP		;CU DATA END?
3435	016646	001413			2.6	BEQ	DTERR		;NO - ERROR
3436	016650	062765	000004	000014	6.4	ADD	#4, BUFC(W)		;DECREMENT RESID COUNT
3437	016656	002036			2.6	BGE	REGES		;IF OVERFLOW - DO ES
3438	016660	062765	000004	000012	6.4	ADD	#4, BUFA(W)		;BUMP BUFFER ADDRESS
3439	016666	016565	000002	000000	7.6	MOV	LREQ(W), REQ(W)		;REQUEST NEXT DT
3440	016674	000207			3.5	RTS	PC		;ALL DONE
3441									
3442	016676	000000			1.8	DTERR:	HALT		;ERROR ON DT
3443									
3444									;ES END EXPECTED
3445									
3446	016700					ESDONE:			
3447	016700	032711	000002		5.3	BIT	#STKSTB, @TTP		;STATUS STACKED?
3448	016704	001040			2.6	BNE	REQUX		;YES - REQUEUE
3449	016706	005065	000002		4.9	CLR	LREQ(W)		
3450	016712	005065	000000		4.9	CLR	REQ(W)		
3451	016716	005065	000004		4.9	CLR	DEVSTA(W)		;SET TO IDLE STATE
3452	016722	032711	000004		5.3	BIT	#CMDCHN, (TTP)		
3453	016726	001403			2.6	BEQ	ESI		
3454	016730	012765	000003	000004	6.4	MOV	#CHAIN, DEVSTA(W)		
3455	016736	004767	000110		7.0	JSR	PC, TSTCUE		
3456	016742	000207			3.5	RTS	PC		
3457									
3458									;REQUEUE REQUEST
3459									
3460	016744	016565	000002	000000	7.6	REQU:	MOV LREQ(W), REQ(W)		;REQUEUE
3461	016752	000207			3.5	RTS	PC		;BACK TO DISPATCH
3462									
3463									;REQUEST ENDING STATUS
3464									
3465	016754	012765	000014	000016	6.4	REGES:	MOV #CE!DE, STS(W)		;PUT IN DESIRED
3466	016762	012765	000003	000000	6.4	REQST:	MOV #RSTS, REQ(W)		;REQUEST
3467	016770	012765	000003	000002	6.4		MOV #RSTS, LREQ(W)		;STATUS
3468	016776	012765	000002	000004	6.4		MOV #STA, DEVSTA(W)		;SET STATE
3469	017004	000207			3.5	RTS	PC		;GET OUT
3470									
3471									;STACK STATUS
3472									
3473	017006	005767	177212		4.4	REQUX:	TST MODE		;2703 MODE?
3474	017012	001754			2.6		BEQ REQU		;YES - REQUEUE STATUS
3475	017014	012767	000001	177210	6.4		MOV #CUES, WAIT		;SET WAIT FLAG AGAIN
3476	017022	000207			3.5	RTS	PC		
3477									
3478									;HIO RECEIVED
3479									
3480	017024	026527	000004	000001	5.9	DXDSC:	CMP DEVSTA(W), #DATA		;COMMAND ACTIVE?
3481	017032	001750			2.6		BEQ REGES		;YES - DO ES
3482									
3483	017034	026527	000004	000002	5.9		CMP DEVSTA(W), #STA		;STATUS REQUIRED?
3484	017042	001744			2.6		BEQ REGES		;YES - PRESENT IT
3485									
3486	017044	004767	000002		7.0	JSR	PC, TSTCUE		;CHECK FOR CUE
3487	017050	000207			3.5	RTS	PC		;AND RETURN
3488									
3489									;TEST FOR CUE REQUIRED

```

3490
3491 017052                                TSTCUE:
3492
3493 017052 005767 177146                    4.4    TST    MODE                ;2703
3494 017056 001424                    2.6    BEQ    TSTX                ;YES - NO CUE
3495
3496 017060 005765 000004                    4.4    TST    DEVSTA(W)          ;DEVICE IDLE?
3497 017064 001021                    2.6    BNE    TSTX
3498
3499 017066 032777 000400 000306              7.7    BIT    #CUBSY, @DXCS      ;CU BUSY SET?
3500 017074 001415                    2.6    BEQ    TSTX
3501 017076 032777 000010 000272              7.7    BIT    #ISSREJ, @DXDS    ;ISS REJECTED?
3502 017104 001411                    2.6    BEQ    TSTX
3503
3504 017106 012765 000040 000016              6.4    MOV    #CUE, STS(W)      ;SET STATUS
3505 017114 012765 000004 000000              6.4    MOV    #RCUE, REQ(W)    ;AND REQUEST
3506 017122 012765 000004 000002              6.4    MOV    #RCUE, LREQ(W)
3507
3508 017130 000207                    3.5    TSTX:  RTS    PC          ;RETURN
3509
3510
3511                                ;ISS HANDLER
3512
3513 017132                                DXISS:
3514
3515                                ;SET UP STATUS BYTE FROM INFO IN T/T
3516
3517 017132 005065 000016                    4.9    CLR    STS(W)            ;RESET TO START
3518 017136 032711 002000                    5.3    BIT    #UCHKS, @TTP     ;UNIT CHECK?
3519 017142 001415                    2.6    BEQ    ISSH1
3520 017144 052765 000002 000016              6.4    BIS    #UC, STS(W)
3521 017152 005765 000006                    4.4    TST    ONLF(W)          ;DEVICE ONLINE?
3522 017156 001404                    2.6    BEQ    ISSH1A           ;NO - SET INT REQ
3523 017160 052765 000200 000010              6.4    BIS    #CMDREJ, SNS(W) ;SET COMMAND REJECT IN SENSE
3524 017166 000403                    2.6    BR    ISSH1             ;GO ON
3525
3526 017170 052765 000100 000010              6.4    ISSH1A: BIS    #INTREQ, SNS(W) ;SET INTERVENTION IN SENSE
3527
3528 017176 032711 001000                    5.3    ISSH1: BIT    #CHENDS, @TTP ;CHANNEL END?
3529 017202 001403                    2.6    BEQ    ISSH2
3530 017204 052765 000014 000016              6.4    BIS    #CE!DE, STS(W)
3531
3532                                ;IF STATUS NON-ZERO, HANDLE STACKED CASE
3533
3534 017212 005765 000016                    4.4    ISSH2: TST    STS(W)      ;STATUS ZERO?
3535 017216 001402                    2.6    BEQ    ISSH3            ;YES - GO ON
3536 017220 000167 177454                    4.9    JMP    ESDONE           ;NO - DO ES
3537
3538                                ;COMMAND ACCEPTED - HANDLE TIO, SENSE, READ, WRITE
3539
3540 017224 105761 000003                    4.4    ISSH3: TSTB   CUCRT(TTP)  ;TIO?
3541 017230 001425                    2.6    BEQ    ISSHX
3542 017232 122761 000004 000003              5.9    CMPB   #SENSE, CUCRT(TTP) ;SENSE?
3543 017240 001022                    2.6    BNE    ISSH4            ;NO
3544
3545                                ;HANDLE SENSE HERE

```



```

3546
3547 017242 012765 000001 000004 6.4      MOV      #DATA, DEVSTA(W)      ;SET STATE
3548 017250 012765 000010 000012 6.4      MOV      #SNS,  BUFA(W)      ;SET BA
3549 017256 060565 000012 000012 4.9      ADD      W,      BUFA(W)      ;
3550 017262 012765 177777 000014 5.4      MOV      #-1,   BUFC(W)      ;SET BC
3551 017270 012765 000002 000000 6.4      MOV      #ROUTP, REQ(W)      ;REQUEST OUTPUT
3552 017276 012765 000002 000002 6.4      MOV      #ROUTP, LREQ(W)     ;
3553 017304 000207 000002 000002 3.5      ISSHX:  RTS      PC          ;ALL DONE
3554
3555      ;READ, WRITE
3556
3557      ZFUB=256.
3558 017306 005065 000010 000004 4.9      ISSH4:  CLR      SNS(W)      ;CLEAR SENSE
3559 017312 012765 000001 000004 6.4      MOV      #DATA, DEVSTA(W)   ;SET STATE
3560 017320 012765 000020 000012 6.4      MOV      #DXBF, BUFA(W)     ;SET BA
3561 017326 060565 000012 000012 4.9      ADD      W,      BUFA(W)     ;
3562 017332 012765 177400 000014 6.4      MOV      #-ZFUB, BUFC(W)    ;SET BC
3563 017340 132761 000001 000003 6.5      BITB    #1,     CUCRT(TTP)  ;WRITE?
3564 017346 001404 000001 000000 2.6      BEQ     ISSH5              ;NO
3565 017350 012765 000001 000000 6.4      MOV      #RINP,  REQ(W)     ;YES - REQUEST INPUT
3566 017356 000403 000001 000000 2.6      BR      ISSH6              ;
3567 017360 012765 000002 000000 6.4      ISSH5:  MOV      #ROUTP, REQ(W) ;REQUEST OUTPUT
3568 017366 016565 000000 000002 7.6      ISSH6:  MOV      REQ(W), LREQ(W) ;
3569 017374 000207 000000 000002 3.5      RTS      PC          ;ALL DONE
3570
3571
3572 017376 176200 000000 000000 3.5      DXDS:   176200              ;DEVICE STATUS ->TT
3573 017400 176202 000000 000000 3.5      DXCA:   176202              ;COMMAND AND ADDRESS ->TT
3574 017402 176204 000000 000000 3.5      DXCS:   176204              ;CONTROL UNIT STATUS
3575 017404 176206 000000 000000 3.5      DXOS:   176206              ;OFFSET AND STATUS
3576 017406 176210 000000 000000 3.5      DXBA:   176210              ;BUS ADDRESS FOR NPR'S
3577 017410 176212 000000 000000 3.5      DXBC:   176212              ;BYTE COUNT
3578 017412 176214 000000 000000 3.5      DXMO:   176214              ;MAINTENANCE OUT
3579 017414 176216 000000 000000 3.5      DXMI:   176216              ;MAINTENANCE IN
3580 017416 176220 000000 000000 3.5      DXCB:   176220              ;CONTROL BITS
3581 017420 176222 000000 000000 3.5      DXND:   176222              ;NPR DATA
3582 017422 176224 000000 000000 3.5      DXES:   176224              ;EXTRA SIGNALS
3583 017424 176226 000000 000000 3.5      DXMOB:  176226              ;MAINTENANCE OUT BUFFERED
3584 017426 176230 000000 000000 3.5      DXES1:  176230              ;EXTRA EXTRA SIGNALS
3585
3586      ;LEGAL ADDRESS LIST
3587
3588 017430 000020 000000 000000 3.5      DXLEGA: .WORD 20              ;PATCH PATCH CH ADRS HERE
3589 017430 177777 000000 000000 3.5      .WORD -1
3590 017432 177777 000000 000000 3.5      .WORD -1
3591 017434 177777 000000 000000 3.5      .WORD -1
3592 017436 177777 000000 000000 3.5      .WORD -1
3593 017440 177777 000000 000000 3.5      .WORD -1
3594 017442 177777 000000 000000 3.5      .WORD -1
3595 017444 177777 000000 000000 3.5      .WORD -1
3596 017446 177777 000000 000000 3.5      .WORD -1
3597 017450 177777 000000 000000 3.5      .WORD -1
3598 017452 177777 000000 000000 3.5      .WORD -1
3599 017454 177777 000000 000000 3.5      .WORD -1
3600 017456 177777 000000 000000 3.5      .WORD -1
3601 017460 177777 000000 000000 3.5      .WORD -1

```

3602 017462 177777
3603 017464 177777
3604 017466 177777
3605 017470 177777
3606
3607 017472
3608
3609 017472 000000
3610 017474 000000
3611 017476 000000
3612 017500 000000
3613 017502 000000
3614 017504 000000
3615 017506 000000
3616 017510 000000
3617 017512 000000
3618 017514 000000
3619 017516 000000
3620 017520 000000
3621 017522 000000
3622 017524 000000
3623 017526 000000
3624 017530 000000
3625
3626
3627
3628
3629 017532
3630
3631 017532 000400
3632 017534 000001
3633 017536 000002
3634 017540 000403
3635 017542 000004
3636 017544 000405
3637 017546 177777
3638 017550 177777
3639 017552 177777
3640 017554 177777
3641 017556 177777
3642 017560 177777
3643 017562 177777
3644 017564 177777
3645 017566 177777
3646 017570 177777
3647
3648
3649 017572
3650 017572 000
3651 017573 000
3652 017574 000
3653 017575 014
3654 017576 000
3655 017577 002
3656 017600 002
3657 017601 002

.WORD -1
.WORD -1
.WORD -1
.WORD -1

SCALD.ADRS:

.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0

;LIST OF DEFAULT COMMANDS

DFLT.CMD:

TIOC ;TEST I/O COMMAND
WRITEC ;WRITE COMMAND
READC ;READ COMMAND
NOPC ;NOP COMMAND
SENSEC ;SENSE COMMAND
ILLC ;ILLAGAL COMMAND
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR
-1 ;LIST TERMINATOR

CMD.STAT:

.BYTE 0
.BYTE 0
.BYTE 0
.BYTE CE!DE
.BYTE 0
.BYTE UC
.BYTE UC
.BYTE UC


```

3658 017602 002
3659 017603 002
3660 017604 002
3661 017605 002
3662 017606 002
3663 017607 002
3664 017610 002
3665 017611 002
3666
3667
3668 017612
3669 017612 000
3670 017613 000
3671 017614 000
3672 017615 014
3673 017616 000
3674 017617 002
3675 017620 002
3676 017621 002
3677 017622 002
3678 017623 002
3679 017624 002
3680 017625 002
3681 017626 002
3682 017627 002
3683 017630 002
3684 017631 002
3685
3686
3687
3688 000000
3689 017632
3690 017632 000000
3691 000001
3692 017634 000000
3693 000002
3694 017636 000000
3695 000003
3696 017640 000000
3697 000004
3698 017642 000000
3699 000005
3700 017644 000000
3701 000006
3702 017646 000000
3703 000007
3704 017650 000000
3705 000010
3706
3707
3708
3709
3710 017652 010146
3711 017654 010246
3712 017656 011501
3713 017660 006301

```

```

.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
;DEFAULT STATUS LIST
DFLT.STAT:
.BYTE 0 ;TIO ACCEPT
.BYTE 0 ;WRITE ACCEPT
.BYTE 0 ;READ ACCEPT
.BYTE CE!DE ;NOP GETS IMMEDIATE END
.BYTE 0 ;SENSE ACCEPT
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
.BYTE UC
;DEVICE CONTROL TABLE (DCT) POINTER
N=0
DCTP:
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
.WORD 0 ;POINTER TO DCT DEVICE 'N
N=N+1
;DL11 C,D,E (ASYNCHRONOUS R/T) TRANSMITTER SERVICE ROUTINE
4.9 DL.XMT: MOV R1,-(SP) ;SAVE REGISTER 1 ON STACK
4.9 MOV R2,-(SP) ;SAVE REGISTER 2 ON STACK
3.8 MOV (R5),R1
2.3 ASL R1 ;FORM MOD(2) INDEX

```

3714	017662	010102			2.3	MOV	R1,R2	;DUPLICATE LINE NUMBER
3715	017664	006302			2.3	ASL	R2	;FORM MOD10(8) INDEX
3716	017666	006302			2.3	ASL	R2	
3717	017670	066702	162004		2.0	ADD	AD,DLX,R2	;FORM DEVICE ADDRESS
3718	017674	105712			2.2	TSTB	R2	;TEST FOR TRANS DONE INC TO BUFF
3719	017676	100407			2.6	BMI	1\$;YES
3720	017700	011203			2.3	MOV	R2,R3	;IS DISPLAY
3721	017702	010304			2.8	MOV	R3,R4	;SHOULD BE
3722	017704	052704	000200		2.8	BIS	#BIT7,R4	;XMIT READY
3723	017710	042704	174000		4.4	BIC	#174000,R4	

NOTE:

MODIFY HLT TO HLT+4 IE (104000=104004)
FOR EXTENDED ERROR DISPLAY
BIT 07:(R4) ONLY SIGNIFICANT BIT (SHOULD BE)

PC	PS	SP	UNIT	DEVADR	WAS	SHOULD BE
(R7)	(PSW)	(R6)	(R1)	(R2)	(R3)	(R4)

3734	017714	104000				HLT		;NO REPORT ERROR
3735	017716	105261	020114		4.9	1\$: INCB	DLDATA(R1)	;INCREMENT SAVE DATA
3736	017722	142761	000340	020114	7.0	BICB	#340,DLDATA(R1)	
3737	017730	005722			3.2	DLOUT: TST	(R2)+	;INC TO XMIT BUFF
3738	017732	116112	020114		6.4	MOV	DLDATA(R1),(R2)	;TRANSMIT DATA
3739	017736	000167	000252		4.9	JMP	CLEANUP	

;DL11 C,D,E (ASCHRONOUS R/T) RECEIVER SERVICE ROUTINE

3743	017742	010146			4.9	DL.RCV: MOV	R1,-(SP)	;SAVE REGISTER 1 ON STACK
3744	017744	010246			4.9	MOV	R2,-(SP)	;SAVE REGISTER 2 ON STACK
3745	017746	011501			3.8	MOV	(R5),R1	
3746	017750	006301			2.3	ASL	R1	;FORM MOD(2) INDEX
3747	017752	010102			2.3	MOV	R1,R2	;DUPLICATE LINE NUMBER
3748	017754	006302			2.3	ASL	R2	;FORM MOD10(8) INDEX
3749	017756	006302			2.3	ASL	R2	
3750	017760	066702	161644		5.0	ADD	AD,DLC,R2	;SET UP DEVICE CSR ADDRESS
3751	017764	105712			2.2	TSTB	(R2)	;DONE
3752	017766	100405			2.6	BMI	2\$;YES
3753	017770	011203			2.8	MOV	R2,R3	;IS DISPLAY
3754	017772	010304			2.3	MOV	R3,R4	;SHOULD BE
3755	017774	052704	000200		3.8	BIS	#BIT7,R4	;RCV DONE

NOTE:

MODIFY HLT TO HLT+4 IE (104000=104004)
FOR EXTENDED ERROR DISPLAY
BIT 07:(R4) ONLY SIGNIFICANT BIT (SHOULD BE)

PC	PS	SP	UNIT	DEVADR	WAS	SHOULD BE
(R7)	(PSW)	(R6)	(R1)	(R2)	(R3)	(R4)

3766	020000	104000				HLT		;NO, FALSE INTERRUPT
3767	020002	005722			3.2	2\$: TST	(R2)+	
3768	020004	005712			3.2	TST	(R2)	;ERROR BIT SET?
3769	020006	100005			2.6	BPL	3\$;NO

3770	020010	011203			3.8
3771	020012	010304			2.3
3772	020014	042704	174000		4.4
3773					
3774					
3775					
3776					
3777					
3778					
3779					
3780					
3781					
3782					
3783	020020	104000			
3784					
3785	020022	011267	000064		6.4
3786	020026	126761	000060	020115	7.1
3787	020034	001407			2.6
3788	020036	116104	020115		5.0
3789	020042	016703	000044		5.0
3790	020046	042704	174000		4.4
3791					
3792					
3793					
3794					
3795					
3796					
3797					
3798					
3799					
3800					
3801	020052	104000			
3802	020054	122761	000037	020115	5.9
3803	020062	001403			2.6
3804	020064	105261	020115		4.9
3805	020070	000403			2.6
3806	020072	112761	000000	020115	6.4
3807	020100	052761	001000	001342	6.4
3808	020106	000167	000102		4.9
3809					
3810	020112	000000			
3811					
3812					
3813					
3814					
3815	020114				
3816	020114	000000			
3817	020116	000000			
3818	020120	000000			
3819	020122	000000			
3820	020124	000000			
3821	020126	000000			
3822	020130	000000			
3823	020132	000000			
3824	020134	000000			
3825	020136	000000			

NOTE:

MODIFY HLT TO HLT+4 IE (104000=104004)
FOR EXTENDED ERROR DISPLAY
BIT 15:(R4) ONLY SIGNIFICANT BIT SHOULD BE

PC	PS	SP	UNIT	DEVADR	WAS	SHOULD BE
(R7)	(PSW)	(R6)	(R1)	(R2)	(R3)	(R4)

HLT

3\$:

```

MOV (R2), DLSAVE ;SAVE THE DATA
CMPB DLSAVE, DLDATA+1(R1) ;TEST THE DATA
BEQ 4$ ;BRANCH IF OK
MOVB DLDATA+1(R1),R4 ;SHOULD BE
MOV DLSAVE, R3 ;IS DISPLAY
BIC #174000,R4

```

NOTE:

MODIFY HLT TO HLT+4 IE (104000=104004)
FOR EXTENDED ERROR DISPLAY
BITS 07:00:(R4) ONLY SIGNIFICANT BITS (SHOULD BE)

PC	PS	SP	UNIT	DEVADR	WAS	SHOULD BE
(R7)	(PSW)	(R6)	(R1)	(R2)	(R3)	(R4)

HLT

4\$:

```

;FALSE DATA
CMPB #37,DLDATA+1(R1) ;UPPER DATA LIMIT FOR 5 BIT ASCII
BEQ DLRX1 ;YAY
INCB DLDATA+1(R1) ;NO INCREMENT DATA
BR DLRX2
MOVB #0,DLDATA+1(R1) ;REINITIALIZE DATA
BIS #FL.DLC,MAP(R1) ;SET FLAG TO INDICATE LINE SERVICED
JMP CLEANUP

```

DLSAVE: 0

;DATA SAVE

;DL11 CSR AND DATA STORAGE

DLDATA:

0
0
0
0
0
0
0
0
0
0
0
0
0

;DL11 DATA, LINE 0
;DL11 DATA, LINE 1
;DL11 DATA, LINE 2
;DL11 DATA, LINE 3
;DL11 DATA, LINE 4
;DL11 DATA, LINE 5
;DL11 DATA, LINE 6
;DL11 DATA, LINE 7
;DL11 DATA, LINE 10
;DL11 DATA, LINE 11


```

3882
3883
3884 020262 011203      3.8      ;      MOV      *****
3885 020264 010304      2.3      ;      MOV      (R2), R3      ;IS DISPLAY
3886 020266 042704 074000 4.4      ;      BIC      R3, R4      ;SHOULD BE
3887
3888 020272 052704 100000 3.8      ;      BIC      #74000,R4
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3900 020276 104000
3901
3902 020300 116204 000007 5.0      1$:      MOV      7(R2), R4      ;FETCH XMIT BUFFER LINE REQUEST FIELD
3903 020304 060304      2.3      ;      ADD      R3, R4      ;COMBINE OFFSET AND LINE #
3904 020306 006304      2.3      ;      ASL      R4      ;SCALE MOD20(16) XMIT CHARACTER OFFSET
3905 020310 016462 030420 000006 7.6      ;      MOV      DJ.XTBL(R4),6(R2) ;TRANSMIT DATA CHARACTER ON CURRENT LINE
3906 020316 105264 030420      4.9      ;      INCB   DJ.XTBL(R4)      ;ADVANCE TO NEXT CHARACTER
3907 020322 042764 000340 030420 7.0      ;      BIC      #340,DJ.XTBL(R4);MASK TO LEVEL 5
3908
3909
3910
3911
3912
3913
3914
3915
3916 020330 005267 000172 4.9      ;      BIC      #340,DJ.EXP(R4) IN DJ.RCV HAS TO BE MODIFIED ALSO.
3917 020334 005712      3.2      ;      INC      DJ.XTLY      ;ADVANCE DJ XMIT CHARACTER TALLY
3918 020336 100760      2.6      ;      TST      (R2)      ;TEST ADDITIONAL ACTIVITY
3919
3920 020340 000167 000146 4.9      ;      BMI      1$      ;YES SERVICE ADDITIONAL LINES
3921
3922
3923
3924
3925
3926
3927 020344 010146 4.9      DJ.RCV: MOV      R1, -(SP)      ;SAVE
3928 020346 010246 4.9      ;      MOV      R2, -(SP)      ;REGISTERS
3929 020350 010346 4.9      ;      MOV      R3, -(SP)      ;ON
3930 020352 010446 4.9      ;      MOV      R4, -(SP)      ;STACK
3931
3932 020354 011501 3.8      ;      MOV      (R5), R1      ;FETCH DEVICE # >R1
3933
3934 020356 006301 2.3      ;      ASL      R1
3935 020360 010102 2.3      ;      MOV      R1,R2
3936 020362 006302 2.3      ;      ASL      R2      ;MOD10 (8)
3937 020364 006302 2.3      ;      ASL      R2      ;OFFSET >R2

```

NOTE:

MODIFY HLT TO HLT+4 IE (104000=104004)
FOR EXTENDED ERROR DISPLAY
BIT 15:(R4) ONLY SIGNIFICANT BIT SHOULD BE

PC	PS	SP	UNIT	DEVADR	WAS	SHOULD BE
(R7)	(PSW)	(R6)	(R1)	(R2)	(R3)	(R4)

HLT ;ERROR XMIT READY BIT 15 FAILURE

NOTE

FOR HIGHER DATA LEVEL VERIFICATION
MODIFY #340 ENTRY TO THE FOLLOWING:
#300 = 6 LEVEL CODE
#200 = 7 LEVEL CODE
#400 = 8 LEVEL CODE

BIC #340,DJ.EXP(R4) IN DJ.RCV HAS TO BE MODIFIED ALSO.
INC DJ.XTLY ;ADVANCE DJ XMIT CHARACTER TALLY
TST (R2) ;TEST ADDITIONAL ACTIVITY
BMI 1\$;YES SERVICE ADDITIONAL LINES
JMP CLENDJ ;NO RESTORE REGISTERS AND EXIT

;DJ11 (ASYNCHRONOUS) RECEIVER SERVICE ROUTINE

DJ.RCV: MOV R1, -(SP) ;SAVE
MOV R2, -(SP) ;REGISTERS
MOV R3, -(SP) ;ON
MOV R4, -(SP) ;STACK
MOV (R5), R1 ;FETCH DEVICE # >R1
ASL R1
MOV R1,R2
ASL R2 ;MOD10 (8)
ASL R2 ;OFFSET >R2


```

4013
4014
4015
4016 020534 032777 000200 002254 7.7 CTYP: BIT #BIT7,ACTPS ;DONE?
4017 020542 001001 2.6 BNE .+4 ;YES
4018 020544 104000 HLT ;NO,REPORT FALSE INTERRUPT
4019 020546 105267 000040 4.9 INCB CTPDAT ;INC TRANSMIT DATA
4020 020552 122767 000147 000032 5.9 CMPB #147,CTPDAT ;TEST FOR UPPER LIMIT
4021 020560 001002 2.6 BNE CTP1
4022 020562 105067 000024 4.9 CLRB CTPDAT ;REINITIALIZE XMIT DATA
4023 020566 116777 000020 002224 8.8 CTP1: MOVB CTPDAT,ACTPB ;TRANSMIT DATA
4024 020574 052767 000001 160502 6.4 BIS #FL.ANC,RU.MAP
4025 020602 052767 000001 160476 6.4 BIS #FL.CTY,RU.ANC
4026 020610 000002 4.8 RTI ;RETURN
4027
4028 020612 000000 CTPDAT: 0 ;TRANSMIT DATA
4029
4030
4031
4032
4033 020614 032777 000200 002170 7.7 CTYR: BIT #BIT7,ACTKS ;DONE?
4034 020622 001001 2.6 BNE .+4 ;YES
4035 020624 104000 HLT ;NO,REPORT FALSE INTERRUPT
4036 020626 105777 002162 5.6 TSTB ACTKB ;TEST FOR LEADER
4037 020632 001421 2.6 BEQ CTYR4 ;BRANCH IF LEADER
4038 020634 127767 002154 000060 8.3 CMPB ACTKB,CTKDAT ;NOT LEADER TEST FOR DATA
4039 020642 001413 2.6 BEQ CTYR3 ;BRANCH IF DATA
4040 020644 017767 002144 000052 8.8 MOV ACTKB,CTKRCV ;RECORD CHARACTER
4041 020652 042767 000200 000044 7.0 BIC #200,CTKRCV ;MASK OFF 8TH BIT
4042 020660 122777 000007 002126 7.1 CMPB #7,ACTKB ;NOT DATA, TEST FOR BELL
4043 020666 001412 2.6 BEQ CTYR2 ;BRANCH IF BELL
4044 020670 104000 HLT ;NONE OF ABOVE,REPORT ERROR
4045 020672 105267 000024 4.9 CTYR3: INCB CTKDAT ;INC DATA
4046 020676 122767 000147 000016 5.9 CTYR4: CMPB #147,CTKDAT ;TEST FOR UPPER LIMIT
4047 020704 001003 2.6 BNE CTYR2
4048 020706 112767 000001 000006 6.4 MOVB #1,CTKDAT ;REINIT RCV DATA
4049 020714 005277 002072 6.1 CTYR2: INC ACTKS ;START READER
4050 020720 000002 4.8 RTI
4051
4052 020722 000001 CTKDAT: 1 ;RCV DATA
4053
4054 020724 000000 CTKRCV: 0 ;RCV DATA RECORDED
4055 ;*****
4056
4057 ;LINE PRINTER TEST SECTION
4058 ;HARDWARE REGISTERS
4059
4060 020726 177514 LPS: 177514 ;LINE PRINTER STATUS
4061 ;BIT15=ERROR
4062 ;BIT7=READY
4063 ;BIT6=INTERRUPT ENABLE
4064 020730 177516 LPB: 177516 ;DATA BUFFER REGISTER
4065 ;BITS 0-6=7 BIT ASCII CHARACTER
4066 ;MEMORY LOCATIONS USED AS FLAGS AND COUNTERS
4067 020732 000120 LPSIZE: 120 ;MAXIMUM LINE LENGTH IN CHARACTERS
4068 020734 000120 LINGTH: 120 ;NUMBER OF CHARACTERS PER LINE

```



```

4069 020736 000120          CHRLIN: 120          ;LINE LENGTH IN CHARACTERS
4070 020740 000040          CARGEN: 40          ;CHARACTER GENERATOR
4071
4072          ;LPVEC. LINE PRINTER INTERRUPT SERVICE ROUTINE
4073
4074 020742 005777 177760    5.6 LPVECT: TST      @LPS          ;ERROR
4075 020746 100001          2.6          BPL      .+4          ;NO
4076 020750 104000          HLT
4077 020752 105777 177750    5.6          TSTB     @LPS          ;YES, REPORT ERROR
4078 020756 100401          2.6          BMI      .+4          ;READY
4079 020760 104000          HLT          ;YES
4080 020762 005267 177752    4.9 LPD:   INC      CARGEN          ;NO
4081 020766 022767 000140 177744    5.9          CMP      @140, CARGEN ;GENERATE NEXT CHARACTER
4082 020774 001003          2.6          BNE     LP1          ;64 CHARACTER LINE PRINTER
4083 020776 012767 000040 177734    6.4          MOV      @40, CARGEN ;RE-INITIALIZE CHARACTER GENERATOR
4084 021004 005367 177726    4.9 LP1:  DEC     CHRLIN          ;COUNT CHARACTERS
4085 021010 001015          2.6          BNE     LP2          ;BRANCH IF NOT AT LINE'S END
4086 021012 012777 000012 177710    7.6          MOV      @12, @LPB    ;YES, LINE FEED
4087 021020 005367 177710    4.9          DEC     LENGTH        ;END OF WEDGE
4088 021024 001003          2.6          BNE     LP3          ;NO
4089 021026 016767 177700 177700    7.6 LP3:  MOV     LPSIZE, LENGTH ;YES, REINITIALIZE LINE LENGTH
4090 021034 016767 177674 177674    7.6          MOV     LENGTH, CHRLIN ;DECREASE # OF CHAR PER LINE
4091 021042 000002          4.8          RTI
4092
4093 021044 016777 177670 177656    8.8 LP2:  MOV     CARGEN, @LPB ;TRANSMIT CHARACTER
4094 021052 052767 000001 160224    6.4          BIS     @FL.ANC, RU.MAP
4095 021060 052767 000002 160220    6.4          BIS     @FL.LP, RU.ANC
4096 021066 000002          4.8          RTI
4097          ;TC11 DECTAPE EXERCISER
4098
4099          ;CONTROL REGISTER ADDRESSES
4100
4101 021070 177342          TCCM: 177342          ;TC COMMAND REGISTER
4102 021072 177340          TCST: 177340          ;TC STATUS REGISTER
4103 021074 177350          TCDT: 177350          ;TC DATA REGISTER
4104 021076 177344          TCWC: 177344          ;TC WORD COUNT
4105 021100 177346          TCBA: 177346          ;TC BUS ADDRESS
4106          ;CONSTANTS FOR FUNCTION CODE
4107 021102 000214          TCIV: 214            ;TC INTERRUPT VECTOR
4108 021104 000216          TCPRT: 216           ;UNIT SELECT CODE
4109 021106 000000          UNIT: 0
4110
4111 021110 000000          TCMSK: 0
4112 021112 000000          TCFIRST: 0          ;FIRST BLOCK TO BE SEARCHED
4113 021114 001101          TCLAST: 1101        ;LASTED BLOCK SEARCHED FOR
4114 021116 000000          TCBLK: 0            ;CURRENT BLOCK FOUND
4115 021120 000000          TCEXPT: 0           ;EXPECTED BLOCK
4116 021122 000000          SEGCNT: 0
4117 021124 000101          SAT: 101            ;DO, STOP ALL TRANSPORTS
4118 021126 000103          RNUM: 103           ;DO, READ BLOCK NUMBER
4119 021130 000105          RDAT: 105           ;DO, READ DATA
4120 021132 000107          RALL: 107           ;DO, READ ALL
4121 021134 000113          WRTM: 113           ;DO, WRITE TIMING AND MARK TRACK
4122 021136 000115          WDATA: 115          ;DO, WRITE DATA
4123 021140 004103          REVRN: 4103
4124

```

4125	021142	012777	021142	177732	7.6	FENDZ:	MOV	#FENDZ, @TCIV	; VECTOR=END ZONE
4126	021150	005777	177716		5.6		TST	@TCST	; TEST FOR END ZONE
4127	021154	100404			2.6		BMI	FEND1	
4128	021156	005277	177706		6.1		INC	@TCCM	; SET DO
4129	021162	000167	000464		4.9		JMP	TCRTI	; NO, WAIT
4130									
4131	021166	012777	021220	177706	7.6	FEND1:	MOV	#TCF1, @TCIV	; INT VECTOR=READ ALL BLOCK #
4132	021174	042777	104000	177666	8.2		BIC	#104000, @TCCM	; CLEAR ERROR AND REVERSE BIT
4133	021202	016767	177704	177710	7.6		MOV	TCFIRST, TCEXPT	; INIT EXPECTED BLOCK
4134	021210	105277	177654		6.1	TCF1A:	INCB	@TCCM	; DO
4135	021214	000167	000432		4.9		JMP	TCRTI	
4136									
4137	021220	032777	100200	177642	7.7	TCF1:	BIT	#100200, @TCCM	; ANY ERROR ON READ
4138	021226	100001			2.6		BPL	.+4	
4139	021230	104000					HLT		; PRINT ERROR
4140	021232	001001			2.6		BNE	.+4	; READ?
4141	021234	104000					HLT		; PRINT ERROR
4142	021236	027767	177632	177654	8.3		CMP	@TCDT, TCEXPT	; TEST FOR CORRECT BLOCK #
4143	021244	002761			2.6		BLT	TCF1A	; IF BLK # LOWER THAN EXPECTED
4144	021246	001401			2.6		BEQ	TCF2	; NO, IS BLK # = EXPECTED
4145	021250	104000					HLT		; NO, WE MISSED BLK #
4146									
4147	021252	012777	021270	177622	7.6	TCF2:	MOV	#TCF3, @TCIV	; VECTOR FOR SEQUENTIAL READS
4148	021260	105277	177604		6.1		INCB	@TCCM	; DO
4149	021264	000167	000362		4.9		JMP	TCRTI	
4150									
4151	021270	032777	100200	177572	7.7	TCF3:	BIT	#100200, @TCCM	; TEST ERROR AND READY
4152	021276	100001			2.6		BPL	.+4	
4153	021300	104000					HLT		; PRINT FORWARD READ ERROR
4154	021302	001001			2.6		BNE	.+4	
4155	021304	104000					HLT		; FALSE READ
4156	021306	027767	177552	177600	8.3		CMP	@TCDT, TCLAST	; HAVE ALL BLOCK #'S BEEN READ
4157	021314	001410			2.6		BEQ	RENDZ	; YES, DRIVE UNIT IN END ZONE
4158	021316	005267	177576		4.9		INC	TCEXPT	; NO, INC BLOCK #
4159	021322	027767	177546	177570	8.3		CMP	@TCDT, TCEXPT	; IS CURRENT BLOCK CORRECT
4160	021330	001401			2.6		BEQ	.+4	
4161	021332	104000					HLT		
4162	021334	000423			2.6		BR	TCWBK	; TCWBK, WRITES 1 BLOCK
4163									
4164									
4165	021336	012777	021336	177536	7.6	RENDZ:	MOV	#RENDZ, @TCIV	; VECTOR=REVERSE END ZONE
4166	021344	016767	177544	177546	7.6		MOV	TCLAST, TCEXPT	; INIT FOR REVERSE SEARCH
4167	021352	005777	177514		5.6		TST	@TCST	
4168	021356	100403			2.6		BMI	REND1	; YES, CHANGE TAPE DIRECTION
4169	021360	005277	177504		6.1		INC	@TCCM	; DO
4170	021364	000532			2.6		BR	TCRTI	
4171	021366	016777	177546	177474	8.8	REND1:	MOV	REVRN, @TCCM	; REVERSE AND READ BLOCK #, DO
4172	021374	012777	021454	177500	7.6		MOV	#TCR1, @TCIV	; SET UP NEW INTERRUPT VECTOR
4173	021402	000523			2.6		BR	TCRTI	
4174									
4175									
4176									
4177	021404	012777	021436	177470	7.6	TCWBK:	MOV	#TCWB1, @TCIV	; INTERRUPT VECTOR POINTS TO WRITE
4178	021412	012777	177400	177456	7.6		MOV	#-400, @TCWC	; WORD COUNT=1BLOCK OF DATA
4179	021420	012777	024234	177452	7.6		MOV	#BUFF, @TCBA	; INITIALIZE BUS ADDRESS
4180	021426	116777	177504	177434	8.8		MOVB	WDATA, @TCCM	; WRITE DATA

;WRITE FORWARD ALL BLOCKS EXCEPT 0


```

4181 021434 000506          2.6          BR          TCRTI
4182 021436 012777 021270 177436 7.6 TCWB1:  MOV      #TCF3, @TCIV      ;SEARCH FOR NEXT BLOCK #
4183 021444 116777 177456 177416 8.8          MOVB     RNUM, @TCCM      ;READ BLOCK #
4184 021452 000477          2.6          BR          TCRTI
4185
4186          ;READ REVERSE ALL BLOCKS
4187
4188 021454 032777 100200 177406 7.7 TCR1:  BIT      #100200, @TCCM      ;TEST ERROR, READY
4189 021462 100001          2.6          BPL      .+4
4190 021464 104000          HLT
4191 021466 001001          2.6          BNE      .+4          ;ERROR ON READ BLOCK REVERSE
4192 021470 104000          HLT          ;FALSE INTERRUPT
4193 021472 027767 177376 177420 8.3          CMP      @TCDT, TCEXPT      ;IS BLOCK # CORRECT
4194 021500 001406          2.6          BEQ      TCR2          ;YES
4195 021502 002002          2.6          BGE      TCR1A         ;NO, HAVE WE PASSED THE BLOCK
4196 021504 104000          HLT          ;YES
4197 021506 000713          2.6          BR          RENDZ        ;FIND END ZONE AND RESTART
4198 021510 005277 177354          6.1 TCR1A:  INC      @TCCM
4199 021514 000456          2.6          BR          TCRTI
4200 021516 012777 021532 177356 7.6 TCR2:  MOV      #TCR3, @TCIV      ;TEST BLOCK # ON INTERRUPT
4201 021524 105277 177340          6.1          INCB     @TCCM
4202 021530 000450          2.6          BR          TCRTI
4203
4204          ;FIND SEQUENTIAL BLOCKS IN REVERSE DIRECTION
4205
4206 021532 032777 100200 177330 7.7 TCR3:  BIT      #100200, @TCCM      ;TEST ERROR, READY
4207 021540 100001          2.6          BPL      .+4
4208 021542 104000          HLT          ;ERROR, REVERSE READ
4209
4210          BNE      .+4
4211 021546 104000          HLT          ;FALSE INTERRUPT
4212 021550 026777 177336 177316 8.3          CMP      TCFIRST, @TCDT      ;HAVE ALL BLOCK #'S BEEN READ
4213 021556 001002          2.6          BNE      TCR4
4214 021560 000167 177356          4.9          JMP      FENDZ          ;YES, ENTER FORWARD END ZONE, RESTART
4215 021564 005367 177330          4.9 TCR4:  DEC      TCEXPT          ;NO, DEC BLOCK #
4216 021570 027767 177300 177322 8.3          CMP      @TCDT, TCEXPT
4217 021576 001401          2.6          BEQ      .+4
4218 021600 104000          HLT
4219 021602 000400          2.6          BR          TCRBK        ;READ A BLOCK
4220
4221          ;READ DATA REVERSE ALL BLOCKS EXCEPT FIRST AND LAST
4222 021604 012777 021636 177270 7.6 TCRBK:  MOV      #TCRB1, @TCIV      ;READ ON INTERRUPT
4223 021612 012777 177400 177256 7.6          MOV      #-400, @TCWC      ;1 BLOCK OF WORDS
4224 021620 012777 026234 177252 7.6          MOV      #INBFC, @TCBA      ;INIT READ BUFFER
4225 021626 116777 177276 177234 8.8          MOVB     RDAT, @TCCM      ;READ DATA
4226 021634 000406          2.6          BR          TCRTI
4227 021636 012777 021532 177236 7.6 TCRB1:  MOV      #TCR3, @TCIV      ;BLOCK SEARCH
4228 021644 116777 177256 177216 8.8          MOVB     RNUM, @TCCM      ;READ NUMBER
4229 021652 052767 000001 157424 6.4 TCRTI:  BIS      #FL.ANC, RU.MAP
4230 021660 052767 000004 157420 6.4          BIS      #FL.TC, RU.ANC
4231 021666 000002          4.8          RTI
4232
4233          ;RF11 DISK EXERCISER SECTION
4234
4235 021670 177460          DCS:    177460          ;DISK CONTROL REGISTER
4236 021672 177462          WC:    177462          ;WORD COUNT REGISTER

```

```

4237 021674 177464
4238 021676 177466
4239 021700 177470
4240 021702 177472
4241 021704 177474
4242 021706 177476
4243 021710 000000
4244 021712 000000
4245 021714 000000
4246 021716 000003
4247
4248
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4260
4261
4262
4263 021720 005767 177764
4264 021724 001412
4265 021726 032767 000001 177754
4266 021734 001054
4267 021736 032767 000002 177744
4268 021744 001116
4269 021746 104000
4270 021750 000000
4271
4272
4273
4274
4275 021752 005777 177712
4276 021756 100001
4277 021760 104000
4278 021762 052777 000400 177700
4279 021770 012777 024234 177676
4280 021776 012777 177400 177666
4281 022004 016777 177702 177664
4282 022012 016777 177676 177660
4283 022020 052777 000103 177642
4284 022026 062767 000430 177656
4285 022034 005567 177654
4286 022040 026767 177652 177646
4287 022046 001122
4288 022050 005067 177636
4289 022054 005267 177630
4290 022060 005067 177630
4291 022064 000513
4292

```

```

CMA: 177464
DAR: 177466
DAE: 177470
DBR: 177472
MAR: 177474
ADS: 177476
SERV: 0
LOWADR: 0
UPADRS: 0
RFLIM: 3

```

```

;CURRENT MEMORY ADDRESS REGISTER
;DSK ADRS REG (LOW 16 BITS)
;DSK ADRS EXT AND ERK REGISTER
;DATA BUFFER REGISTER
;MAINTENANCE REGISTER
;ADRS OF DSK SEGMENT REG
;SERVICE QUEUE
;LSB'S OF RF11 DISK ADDRESS
;MSB'S OF RF11 DISK ADDRESS
;UPPER LIMIT, 256K FOR ONE DISK

```

```

;DISK ADDRESS EXTENTION BITS
;BIT0=TA05
;BIT1=TA06
;BIT2=DA00
;BIT3=DA01
;BIT4=DA02
;BIT5=DA03

```

```

;TRACK ADRS BITS
;DISK ADRS BIT00

```

```

;RF11 INTERRUPT ENTRY POINT
;DETERMINE FUNCTION TO PERFORM.
;WRITE, WRITE CHEC, READ

```

```

RFISR: TST      SERV
        BEQ      WRITE
        BIT      #1,SERV
        BNE      WRCHK
        BIT      #2,SERV
        BNE      READ
        HLT
        HALT

```

```

;DISK EMULATOR

```

```

;WRITE 256 WORDS ON DISK

```

```

WRITE: TST      JDCS
        BPL      .+4
        HLT
        BIS      #400,JDCS
        MOV      #BUFF,DCMA
        MOV      #-400,ANC
        MOV      LOWADR,ADAR
        MOV      UPADRS,DAE
        BIS      #103,JDCS
        ADD      #430,LOWADR
        ADC      UPADRS
        CMP      RFLIM,UPADRS
        BNE      RFRTI
        CLR      LOWADR
        INC      SERV
        CLR      UPADRS
        BR       RFRTI

```

```

;ERROR
;NO, SERVICE INTERRUPT
;YES, REPORT ERROR
;BIT8, DISK CLEAR
;INITIALIZE CURRENT ADDRESS
;WORD COUNT = 400(B)
;INIT DSK ADRS (16 LOW BITS)
;INIT DSK ADRS (EXT)
;INT ENB, WRITE, GO
;INC DISK ADRS BY 430
;ADD CARRY TO MSB'S
;TEST FOR UPPER LIMIT, 256K
;NO, RTI
;YES, CLEAR ADDRESS
;INC SERVICE COUNT TO NEXT FUNCTION
;YES.

```



```

4293                                     ;WRITE CHECK 256 WORDS ON DISK
4294                                     ;
4295
4296 022066 005777 177576      5.6 WRTCHK: TST      @DCS      ;ERROR
4297 022072 100001            2.6          SPL      .+4      ;NO
4298 022074 104000            2.6          HLT                      ;YES, REPORT ERROR
4299 022076 052777 000400 177564 7.6          BIS      @400,@DCS ;DISK CLEAR
4300 022104 012777 024234 177562 7.6          MOV      @BUFF,@CMA ;INITIALIZE CURRENT ADDRESS
4301 022112 012777 177400 177552 7.6          MOV      #-400,@C  ;WORD COUNT = 400(8)
4302 022120 016777 177566 177550 8.8          MOV      LOWADR,@DAR
4303 022126 016777 177562 177544 8.8          MOV      UPADR,@DAE
4304 022134 052777 000107 177526 7.6          BIS      @107,@DCS ;INT ENB, WRITE CHECK, GO
4305 022142 062767 000430 177542 6.4          ADD      @430,LOWADR ;INC DISK ADRS BY 430
4306 022150 005567 177540      4.9          ADC      UPADR      ;ADD CARRY TO MSB'S
4307 022154 026767 177536 177532 7.1          CMP      RFLIM,UPADR ;TEST FOR UPPER LIMIT, 256K
4308 022162 001054            2.6          BNE      RFRTI      ;NO, RTI
4309 022164 005067 177522      4.9          CLR      LOWADR     ;YES, CLEAR ADDRESS
4310 022170 005267 177514      4.9          INC      SERV      ;INC SERVICE COUNT TO NEXT FUNCTION
4311 022174 005067 177514      4.9          CLR      UPADR     ;YES,
4312 022200 000445            2.6          BR       RFRTI
4313
4314                                     ;READ 256 WORDS FROM DISK
4315                                     ;
4316
4317 022202 005777 177462      5.6 READ:  TST      @DCS      ;ERROR
4318 022206 100001            2.6          BPL      .+4      ;NO
4319 022210 104000            2.6          HLT                      ;YES, REPORT ERROR
4320 022212 052777 000400 177450 7.6          BIS      @400,@DCS ;DISK CLEAR
4321 022220 012777 025234 177446 7.6          MOV      @INBRF,@CMA ;INITIALIZE MEMORY ADDRESS
4322 022226 012777 177400 177436 7.6          MOV      #-400,@C
4323 022234 016777 177452 177434 8.8          MOV      LOWADR,@DAR
4324 022242 016777 177446 177430 8.8          MOV      UPADR,@DAE
4325 022250 052777 000105 177412 7.6          BIS      @105,@DCS ;INT ENB, WRITE
4326 022256 062767 000430 177426 6.4          ADD      @430,LOWADR ;INC DISK ADRS BY 430
4327 022264 005567 177424      4.9          ADC      UPADR      ;ADD CARRY TO MSB'S
4328 022270 026767 177422 177416 7.1          CMP      RFLIM,UPADR ;TEST FOR UPPER LIMIT, 256K
4329 022276 001006            2.6          BNE      RFRTI      ;NO, RTI
4330 022300 005067 177404      4.9          CLR      SERV      ;FUNCTION = WRITE
4331 022304 005067 177402      4.9          CLR      LOWADR     ;YES, CLEAR ADDRESS
4332 022310 005067 177400      4.9          CLR      UPADR     ;YES
4333 022314 052767 000001 156762 6.4 RFRTI: BIS      @FL.ANC,RU.MAP
4334 022322 052767 000010 156756 6.4          BIS      @FL.RF,RU.ANC
4335 022330 000002            4.8          RTI
4336
4337                                     ;***** RK11*****
4338
4339                                     ;177400
4340 022332 177400            ;177402
4341 022334 177413            RKDS: 177400 ;DRIVE STATUS
4342 022336 177412            RKDAH: 177413 ;MAINTENANCE REG
4343 022340 177416            RKDAE: 177412 ;DRIVE STATUS REGISTER
4344 022342 177412            RKDBR: 177416 ;HIGH BYTE DISK ADRS
4345 022344 177406            RKDAR: 177412 ;DISK ADRS
4346 022346 177410            RKDC: 177406 ;DATA BUFFER REG
4347 022350 177404            RKDBR: 177412 ;DISK ADRS REG
4348 022352 177405            RKDC: 177406 ;WORD COUNT REG
                                     RKBAR: 177410 ;CURRENT ADRS REG
                                     RKCSR: 177404 ;STATUS REG
                                     RKCSRH: 177405 ;HIGH BYTE ADRS OF CSR

```

```

4349
4350 022354 000240      1.5  RKSTART:  NOP
4351 022356 012777 000000 177752  7.6      MOV    #0, @RKDAE      ;INIT DISK DAR-DAE
4352 022364 016777 000124 177754  8.8      RK1:   MOV    LLIMIT, @RKBAR   ;DATA BASE
4353 022372 016777 000114 177744  8.8      MOV    RKWORDCT, @RKWC ;LENGTH OF TRANSFER
4354 022400 116777 000104 177742  8.8      MOV    RKFUNCTION, @RKCSR ;WRITE OR WRITE CHECK TO DISK
4355 022406 052767 000001 156670  6.4      BIS    #FL.ANC, RU.MAP
4356 022414 052767 000020 156664  6.4      BIS    #FL.RK, RU.ANC
4357 022422 000002
4358 022424 105777 177720  5.6      RKISR: TSTB   @RKCSR      ;READY?
4359 022430 100402      2.6      BMI    .+6
4360 022432 104000
4361 022434 000747      2.6      BR     RKSTART
4362 022436 005777 177702  5.6      TST   @RKWC      ;IS WORD COUNT ZERO
4363 022442 001402      2.6      BEQ   .+6
4364 022444 104000
4365 022446 000742      2.6      BR     RKSTART
4366 022450 005777 177674  5.6      TST   @RKCSR
4367 022454 100002      2.6      BPL   .+6
4368 022456 104000
4369 022460 000735      2.6      BR     RKSTART
4370 022462 032777 000037 177646  7.7      BIT   #37, @RKDAE   ;DISK AT UPPER LIMIT
4371 022470 001335      2.6      BNE   RK1         ;NO
4372 022472 122777 000031 177634  7.1      CMPB  #31, @RKDAH   ;
4373 022500 001331      2.6      BNE   RK1
4374 022502 000367 000002  4.9      SWAB  RKFUNCTION   ;CHANG COMMAND
4375 022506 000722      2.6      BR     RKSTART
4376 022510 000000      RKFUNCTION: 0 ;DISK COMMAND
4377 022512 177400      RKWORDCT:  -400 ;WORD COUNT
4378 022514 024234      LLIMIT:  BUFF ;BEGINING OF DATA FILE
4379
4380
4381 ;*****
4382 ;RP11 DISK SERVICE ROUTINE
4383 RPCA: 176722 ;RP11 CYLINDER ADDRESS DISK
4384 RPDAR: 176725 ;RP11 HIGH BYTE OF DISK ADDRESS
4385 RPDAR: 176710 ;RP11 DRIVE STATUS REGISTER
4386 RPDAR: 176724 ;RP11 DISK ADDRESS REGISTER
4387 RPMC: 176716 ;RP11 WORD COUNT REGISTER
4388 RPBAR: 176720 ;RP11 CURRENT ADDRESS REGISTER
4389 RPCR: 176714 ;RP11 STATUS REGISTER
4390 RPCR: 176715 ;RP11 HIGH BYTE ADDRESS OF CSR
4391 RPISR: TSTB   @RPCSR   ;INTERRUPT VECTOR POINTS HERE
4392 RPISR: BMI    .+6
4393 RPISR: HLT
4394 RPISR: BR     RPSTART ;RP11 READY NOT UP
4395 RPISR: TST   @RPCSR
4396 RPISR: BPL   RPX
4397 022556 016700 177740  5.0      MOV    RPDSR, R0 ;DEVICE STATUS REG ADDRESS > R0
4398 022562 017701 177734  6.2      MOV    @RPDSR, R1 ;CONTENTS OF STATUS REG > R1
4399 022566 016702 177736  5.0      MOV    RPBAR, R2 ;DEVICE CURRENT ADD REG > R2
4400 022572 017703 177732  6.2      MOV    @RPBAR, R3 ;CONTENTS OF ADD REG > R3
4401 022576 017704 177730  6.2      MOV    @RPCSR, R4 ;CONTENTS OF STATUS REG > R4
4402
4403 ;NOTE:
4404 ; CHANGE HLT TO HLT+4 (104000 = 104004)
; TO DISPLAY CONTENTS OF REGISTERS SPECIFIED

```



```
4428  
4429  
4430 ;SCRATCH PAD AND CONSTANTS  
4431 022702 000101 LIMIT: 101 ;HIGH LIMIT OF DISK DATA  
4432 022704 000000 FSEG: 0 ;SEGMENT COUNT (1/2 OF A FILE)  
4433 022706 000001 FDATA: 1 ;FILE DATA  
4434 022710 000000 DFILE: 0 ;DATA FILE  
4435 022712 000001 ONE: 1  
4436 022714 000000 DACNT: 0 ;BASE DATA INITIALIZATION COUNT  
4437 022716 000000 CNT: 0 ;TTY INITIALIZATION VECTOR  
4438 022720 000000 PUNCNT: 0 ;PUNCH INTERROGATION COUNT  
4439 022722 000000 RDRCNT: 0 ;READER INTERROGATION COUNT  
4440 022724 000000 ACTVOC: 0 ;ACTIVE DATA SETS  
4441 022726 000000 SAVPC: 0 ;SAVE PC  
4442 022730 000000 SAVCC: 0 ;SAVE PROCESSOR STATUS  
4443 022732 000000 SAVR2: 0 ;SAVE R2 FOR ERROR PRINT  
4444 022734 000000 SAVR3: 0 ;SAVE R3 FOR ERROR PRINT  
4445 022736 000000 SAVR4: 0 ;SAVE R4 FOR ERROR PRINT  
4446 022740 000000 EMASK: 0 ;EMULATOR MASK  
4447 022742 000000 STOR0: 0 ;STORE R0  
4448 022744 000000 STOR1: 0 ;STORE R1  
4449 022746 000000 STOR2: 0 ;STORE R2  
4450 022750 000000 STOR3: 0  
4451 022752 000000 STOR4: 0  
4452 022754 000000 STOR5: 0  
4453 022756 000000 STOR6: 0 ;STORE R6  
4454 022760 000000 TMPDAT: 0  
4455 022762 000000 STOHLT: 0 ;STORE EMULATOR NUMBER  
4456  
4457 022764 000000 MAX: 0 ;MAX # OF DEVICE  
4458 022766 000000 MARK: 0 ;DEVICE MARKER;  
4459  
4460 ;DEVICE PRIORITY TABLE  
4461  
4462 022770 000000 LINE: 0 ;STORAGE FOR LINE NUMBER  
4463 022772 000000 TERTMP: 0  
4464  
4465 ;DEVICE ADDRESSES  
4466  
4467 022774 176504 KLX50: 176504 ;FIRST MULTI-USER KL11 ADDRESS  
4468 022776 176500 KLADR5: 176500  
4469 023000 176500 KLR50: 176500  
4470 023002 174000 DCADR5: 174000 ;ADDRESS ASSIGNED TO FIRST DC11  
4471 023004 174770 DPADR5: 174770 ;ADDRESS ASSIGNED TO FIRST DP11  
4472 023006 175000 DMADR5: 175000 ;FIRST DM11 ADRS  
4473 023010 175200 DNADR5: 175200 ;FIRST DN11 ADRS  
4474 023012 177560 CTKS: 177560  
4475 023014 177562 CTKB: 177562  
4476 023016 177564 CTPS: 177564  
4477 023020 177566 CTPB: 177566  
4478 023022 177342 TC11: 177342 ;TC11 ADDRESS  
4479 023024 177460 RF11: 177460 ;RF11 ADDRESS  
4480 023026 177514 LP11: 177514 ;LP11 ADDRESS  
4481  
4482 023030 000000 NDCS: 0 ;NUMBER OF DC11'S  
4483 023032 000000 NKLS: 0 ;NUMBER OF KL11'S
```


MAINDEC-11-DZQCA-G
DZQCA.P11

COMMUNICATION TEST PROGRAM (CTP)
SCRATCH PAD AND CONSTANTS

4484	023034	000000
4485	023036	000000
4486	023040	000000
4487	023042	000000
4488	023044	000000

NUPS:	0
NDMS:	00
NDNS:	00
ANCMAP:	00
DEVCNT:	0

:	NUMBER OF DP11'S
:	NUMBER OF DM'S
:	NUMBER OF DN'S
:	MAP OF NON-COM ANCILLARIES
:	DEVICE COUNT

```

4489
4490
4491
4492
4493 023046
4494 023046 004767 000072 7.0
4495 023052 012501 3.8
4496 023054 012502 3.8
4497 023056 005021 3.7
4498 023060 005302 2.3
4499 023062 001375 2.6
4500 023064 004767 000076 7.0
4501 023070 000205 3.5
4502
4503 023072 012700 000016 3.8
4504 023076 012767 030052 000012 6.4
4505 023104 005711 3.2
4506 023106 001407 2.6
4507 023110 000004 027634
4508 023114 000004
4509 023116 000000
4510 023120 011105 3.8
4511 023122 004767 000430 7.0
4512 023126 005721 3.2
4513 023130 062767 000020 177760 6.4
4514 023136 005300 2.3
4515 023140 001361 2.6
4516 023142 000207 3.5
4517
4518
4519 023144 012667 000014 6.4
4520 023150 010446 4.9
4521 023152 010346 4.9
4522 023154 010246 4.9
4523 023156 010146 4.9
4524 023160 010046 4.9
4525 023162 000137 3.7
4526 023164 000000
4527
4528
4529 023166 012667 000014 6.4
4530 023172 012600 3.8
4531 023174 012601 3.8
4532 023176 012602 3.8
4533 023200 012603 3.8
4534 023202 012604 3.8
4535
4536 023204 000137 3.7
4537 023206 000000

```

; THIS CALL CLEARS CORE BETWEEN THE ADDRESS LISTED
 ; AFTER THE CALL AND THE NUMBER OF WORDS SPECIFIED
 CLEAR: JSR PC SAVRG ; SAVE R0-R4
 MOV (R5)+,R1 ; FETCH ADRS OF FIRST WORD TO BE CLEARED
 MOV (R5)+,R2 ; FETCH WORD COUNT
 CLR1: CLR (R1)+ ; CLEAR MEMORY
 DEC R2 ; DEC WORD COUNT
 BNE CLR1 ; BRANCH IF AREA NOT CLEARED
 JSR PC, RSTRG ; RESTORE R0-R4
 RTS %5 ; RETURN
 REPORT: MOV #MAPSIZ-1,R0
 MOV #MSGMAP,2\$; SET UP MESSAGE POINTER
 1\$: TST (R1) ; CHECK FOR ANY DEVICES
 BEQ 3\$; BRANCH IF NONE
 TYPE, RETURN
 TYPE
 2\$: 0
 MOV (R1),TTY ; PUT (R1) INTO TTY
 JSR PC,BITYPS ; TYPE (R1) IN BITS
 3\$: TST (R1)+ ; UPDATE POINTER
 ADD #20, 2\$; UPDATE MESSAGE POINTER
 DEC R0 ; DONE?
 BNE 1\$; BRANCH IF NOT
 RTS PC
 ; SAVE REGS 0 TO 4 SUBROUTINE.
 SAVRG: MOV (6)+, SVRPC ; SAVE PC
 MOV %4, -(6) ; SAVE REGS 0 - 4
 MOV %3, -(6) ; IN STACK.
 MOV %2, -(6)
 MOV %1, -(6)
 MOV %0, -(6)
 JMP 2(7)+ ; SIMULATE RTS PC
 SVRPC: 0
 ; RESTORE REGS 0 TO 4 SUBROUTINE.
 RSTRG: MOV (6)+, RSTPC ; SAVE PC
 MOV (6)+, %0 ; RESTORE REGS 0 - 4
 MOV (6)+, %1 ; FROM STACK.
 MOV (6)+, %2
 MOV (6)+, %3
 MOV (6)+, %4
 RSTPC: JMP 2(7)+ ; SIMULATE RTS PC
 0


```

;          SOCTAL          OCTAL TYPEOUT ROUTINE
; THIS ROUTINE IS USED TO TYPE AN OCTAL NUMBER ON THE TTY. IT WILL TYPE
; ALL 6 CHARACTERS, SUPPRESS LEADING ZEROES, TYPE AN 18 BIT ADDRESS, OR TYPE
; THE 16 BITS. IT IS CALLED VIA THE DUMP, SDUMP, DUMP18, OR BITYPE MACRO'S.
4617
4618
4619
4620
4621
4622
4623 023556 012767 170101 000140 6.4 BITYPE: MOV #170101,.PR ;SET BIT FLAG ANS 16. CHARACTER COUNT
4624 023564 000411 2.6 BR .PTIT ;NOW TYPE IT IN BIT FORM
4625 023566 112767 000001 000130 6.4 PRINTR: MOVB #1,.PR ;SET ZERO FILL SWITCH
4626 023574 000402 2.6 BR .+6 ;SKIP
4627 023576 005067 000122 4.9 PRINTS: CLR .PR ;SUPPRESS LEADING ZERO'S
4628 023602 112767 177772 000115 6.4 MOVB #-6,.PR+1 ;SET COUNT
4629 023610 010446 4.9 .PTIT: MOV R4,-(6) ;SAVE R4
4630 023612 012704 023726 3.8 MOV #.PR+2,R4 ;SET POINTER TO FIRST ASCII CHAR.
4631 023616 105014 3.7 CLRB (4) ;CLEAR FIRST BYTE
4632 023620 000411 2.6 BR .PRF ;ROTATE FIRST BIT
4633 023622 105014 3.7 .PRL: CLRB (4) ;CLEAR BYTE OF CHARACTER
4634 023624 032767 000100 000072 6.5 BIT #100,.PR ;BIT TYPING MODE?
4635 023632 001004 2.6 BNE .PRF ;YES - SKIP 2 ROTATES
4636 023634 006105 2.3 ROL TTY ;ROTATE BIT INTO C
4637 023636 106114 3.7 ROLB (4) ;PACK IT
4638 023640 006105 2.3 ROL TTY ;ROTATE BIT INTO C
4639 023642 106114 3.7 ROLB (4) ;PACK IT
4640 023644 006105 2.3 .PRF: ROL TTY ;ROTATE BIT INTO C
4641 023646 106114 3.7 ROLB (4) ;PACK IT
4642 023650 105714 3.2 TSTB (4) ;IS IT ZERO?
4643 023652 001402 2.6 BEQ .+6 ;SKIP INC
4644 023654 105267 000044 4.9 INCB .PR ;SET FILL SWITCH
4645 023660 105767 000040 4.4 TSTB .PR ;CHECK FILL SWITCH
4646 023664 001402 2.6 BEQ .+6 ;SKIP BITSET
4647 023666 152724 000060 5.2 BISB #'0,(4)+ ;MAKE INTO ASCII CHAR
4648 023672 105267 000027 4.9 INCB .PR+1 ;INC COUNT
4649 023676 001351 2.6 BNE .PRL ;REPEAT
4650 023700 022704 023726 3.8 CMP #.PR+2,R4 ;EMPTY BUFFER?
4651 023704 001002 2.6 BNE .+6 ;SKIP IF NOT
4652 023706 112724 000060 5.2 MOVB #'0,(4)+ ;LOAD 1 ZERO
4653 023712 105014 3.7 CLRB (4) ;NULL TERMINATOR
4654 023714 000004 023726 TYPE .PR+2 ;TYPE IT
4655 023720 012604 3.8 MOV (6)+,R4 ;RESTORE R4
4656 023722 000207 3.5 RTS PC ;RETURN
4657 023724 000012 .PR: .BLKW 12 ;COUNT, SWITCH, AND OUTPUT BUFFER
4658

```

```

4659                                     :      $SCOPE          SCOPE LOOP HANDLER
4660
4661                                     : THIS ROUTINE HANDLES THE ITERATIONS, LOOPING, ERROR
4662                                     : LOOPING, AND THE DISPLAYING OF THE TEST NUMBER.
4663
4664                                     : "SCOPE" IS PLACED BETWEEN EACH SUBTEST IN THE TEST AND
4665                                     : RECORDS THE STARTING ADDRESS OF THE SUBTEST IN "LAD:"
4666
4667 023750 005737 001024          3.2 TRAPS:  TST      @#1024          ; CHECK MONITOR CONTROL
4668 023754 001022                2.6          BNE      SVLADS          ; (ACTIVE) QUICK ITERATIONS
4669 023756 032737 040000 177570  5.3          BIT      @SW14,@#SWR    ; LOOP ON TEST?
4670 023764 001026                2.6          BNE      KITS           ; LOOP ON TEST IS SET
4671 023766 032737 004000 177570  5.3          BIT      @SW11,@#SWR    ; KILL ITERATIONS
4672 023774 001012                2.6          BNE      SVLADS$        ; YES - KILL ITERATIONS
4673 023776 105767 155177          4.4          TSTB     ICNT+1        ; FIRST ONE?
4674 024002 001404                2.6          BEQ      2$            ; BRANCH IF FIRST
4675 024004 126767 000062 155167  7.1          CMPB     TIMES,ICNT+1  ; DONE?
4676 024012 001013                2.6          BNE      KITS           ; BRANCH IF NOT
4677 024014 112767 000001 155157  6.4 2$:  MOVB     @1,ICNT+1    ; FIRST ITERATION
4678 024022 105267 155152          4.9 SVLADS: INCB     ICNT      ; COUNT TEST NUMBERS
4679 024026 011667 000036          6.4          MOV      (6),LAD       ; SAVE LOOP ADDRESS
4680 024032 016737 155142 177570  6.4          MOV      ICNT,@#DISPLAY ; DISPLAY TEST NO. AND ITERATION COUNT
4681 024040 000002                4.8          RTI                    ; RETURN
4682
4683 024042 105267 155133          4.9 KITS:  INCB     ICNT+1    ; INC THE ITERATION COUNT
4684 024046 016737 155126 177570  6.4 OVERS: MOV      ICNT,@#DISPLAY ; SET UP DISPLAY
4685 024054 005767 000010          4.4          TST      LAD           ; FIRST ONE?
4686 024060 001760                2.6          BEQ      SVLADS$        ; YES
4687 024062 016716 000002          6.4          MOV      LAD,(6)       ; FUDGE RETURN ADDRESS
4688 024066 000002                4.8          RTI                    ; FIXES P5
4689
4690 024070 000000                LAD:    0              ; LOOP ADDRESS
4691 024072 000377                TIMES:  377           ; RUN 377 TIMES

```


4692	024074	012777	024222	000126	7.6	PDOWN:	MOV	#ILLUP, @PUVECS	:SET FOR FAST UP
4693	024102	012777	000340	000122	7.6		MOV	#340, @PUVECS+2	:PRIO:7
4694	024110	010046			4.9		MOV	R0, -(6)	:PUSH R0 ON STACK
4695	024112	010146			4.9		MOV	R1, -(6)	:PUSH R1 ON STACK
4696	024114	010246			4.9		MOV	R2, -(6)	:PUSH R2 ON STACK
4697	024116	010346			4.9		MOV	R3, -(6)	:PUSH R3 ON STACK
4698	024120	010446			4.9		MOV	R4, -(6)	:PUSH R4 ON STACK
4699	024122	010546			4.9		MOV	R5, -(6)	:PUSH R5 ON STACK
4700	024124	010667	000076		4.9		MOV	SP, .SAVR6	:SAVE SP
4701	024130	012777	024140	000072	7.6		MOV	#PUPS, @PUVECS	:SET UP VECTOR
4702	024136	000000			1.8		HALT		:WAIT FOR PF
4703									
4704	024140	016706	000062		5.0	PUPS:	MOV	.SAVR6, SP	:GET SP
4705	024144	005001			2.3		CLR	R1	:WAIT LOOP FOR THE TTY
4706	024146	005201			2.3	IS:	INC	R1	:WAIT FOR THE INC
4707	024150	001376			2.6		BNE	IS	:OF WORD
4708	024152	012605			3.8		MOV	(6)+, R5	:POP STACK INTO R5
4709	024154	012604			3.8		MOV	(6)+, R4	:POP STACK INTO R4
4710	024156	012603			3.8		MOV	(6)+, R3	:POP STACK INTO R3
4711	024160	012602			3.8		MOV	(6)+, R2	:POP STACK INTO R2
4712	024162	012601			3.8		MOV	(6)+, R1	:POP STACK INTO R1
4713	024164	012600			3.8		MOV	(6)+, R0	:POP STACK INTO R0
4714	024166	012737	024074	000024	5.2		MOV	#PDOWN, @#24	:SET UP THE POWER DOWN VECTOR
4715	024174	012737	000340	000026	5.2		MOV	#340, @#26	:PRIO:7
4716	024202	000004	024206				TYPE	+2	:ASCIZ <15><12>"POWER"
4717	024216	000167	155630		4.9		JMP	DEVICE	:JMP TO USER ADDRESS
4718									
4719	024222	000000			1.8	ILLUP:	HALT		:THE POWER UP SEQUENCE WAS STARTED
4720	024224	000776			2.6		BR	.-2	:BEFORE THE POWER DOWN WAS COMPLETE
4721									
4722	024226	000000					.SAVR6: 0		:PUT THE SP HERE
4723	024230	000024	000026				PUVECS: 24,26		:POWER UP VECTOR

4724		
4725		
4726	024234	000001
4727		025234
4728	025234	000001
4729		026234
4730	026234	000001
4731		027234
4732	027234	
4733	027234	000
4734	027235	001
4735	027236	002
4736	027237	003
4737	027240	004
4738	027241	005
4739	027242	006
4740	027243	007
4741	027244	010
4742	027245	011
4743	027246	012
4744	027247	013
4745	027250	014
4746	027251	015
4747	027252	016
4748	027253	017
4749	027254	020
4750	027255	021
4751	027256	022
4752	027257	023
4753	027260	024
4754	027261	025
4755	027262	026
4756	027263	027
4757	027264	030
4758	027265	031
4759	027266	032
4760	027267	033
4761	027270	034
4762	027271	035
4763	027272	036
4764	027273	037
4765	027274	040
4766	027275	041
4767	027276	042
4768	027277	043
4769	027300	044
4770	027301	045
4771	027302	046
4772	027303	047
4773	027304	050
4774	027305	051
4775	027306	052
4776	027307	053
4777	027310	054
4778	027311	055
4779	027312	056

.OUTPUT DATA FILE FOR DISK AND DECTAPE
BUFF: 1 ;COMMON OUTPUT FILE
.=BUFF+1000
INBFRF: 1 ;RF11 INPUT BUFFER
.=INBFRF+1000
INBFTC: 1 ;TC11 INPUT BUFFER
.=INBFTC+1000
BINCNT: ;BEGINNING OF BINARY COUNT TABLE
.BYTE 0
.BYTE 1
.BYTE 2
.BYTE 3
.BYTE 4
.BYTE 5
.BYTE 6
.BYTE 7
.BYTE 10
.BYTE 11
.BYTE 12
.BYTE 13
.BYTE 14
.BYTE 15
.BYTE 16
.BYTE 17
.BYTE 20
.BYTE 21
.BYTE 22
.BYTE 23
.BYTE 24
.BYTE 25
.BYTE 26
.BYTE 27
.BYTE 30
.BYTE 31
.BYTE 32
.BYTE 33
.BYTE 34
.BYTE 35
.BYTE 36
.BYTE 37
.BYTE 40
.BYTE 41
.BYTE 42
.BYTE 43
.BYTE 44
.BYTE 45
.BYTE 46
.BYTE 47
.BYTE 50
.BYTE 51
.BYTE 52
.BYTE 53
.BYTE 54
.BYTE 55
.BYTE 56

4780	027313	057	.BYTE	57
4781	027314	060	.BYTE	60
4782	027315	061	.BYTE	61
4783	027316	062	.BYTE	62
4784	027317	063	.BYTE	63
4785	027320	064	.BYTE	64
4786	027321	065	.BYTE	65
4787	027322	066	.BYTE	66
4788	027323	067	.BYTE	67
4789	027324	070	.BYTE	70
4790	027325	071	.BYTE	71
4791	027326	072	.BYTE	72
4792	027327	073	.BYTE	73
4793	027330	074	.BYTE	74
4794	027331	075	.BYTE	75
4795	027332	076	.BYTE	76
4796	027333	077	.BYTE	77
4797	027334	100	.BYTE	100
4798	027335	101	.BYTE	101
4799	027336	102	.BYTE	102
4800	027337	103	.BYTE	103
4801	027340	104	.BYTE	104
4802	027341	105	.BYTE	105
4803	027342	106	.BYTE	106
4804	027343	107	.BYTE	107
4805	027344	110	.BYTE	110
4806	027345	111	.BYTE	111
4807	027346	112	.BYTE	112
4808	027347	113	.BYTE	113
4809	027350	114	.BYTE	114
4810	027351	115	.BYTE	115
4811	027352	116	.BYTE	116
4812	027353	117	.BYTE	117
4813	027354	120	.BYTE	120
4814	027355	121	.BYTE	121
4815	027356	122	.BYTE	122
4816	027357	123	.BYTE	123
4817	027360	124	.BYTE	124
4818	027361	125	.BYTE	125
4819	027362	126	.BYTE	126
4820	027363	127	.BYTE	127
4821	027364	130	.BYTE	130
4822	027365	131	.BYTE	131
4823	027366	132	.BYTE	132
4824	027367	133	.BYTE	133
4825	027370	134	.BYTE	134
4826	027371	135	.BYTE	135
4827	027372	136	.BYTE	136
4828	027373	137	.BYTE	137
4829	027374	140	.BYTE	140
4830	027375	141	.BYTE	141
4831	027376	142	.BYTE	142
4832	027377	143	.BYTE	143
4833	027400	144	.BYTE	144
4834	027401	145	.BYTE	145
4835	027402	146	.BYTE	146

4836	027403	147	.BYTE	147
4837	027404	150	.BYTE	150
4838	027405	151	.BYTE	151
4839	027406	152	.BYTE	152
4840	027407	153	.BYTE	153
4841	027410	154	.BYTE	154
4842	027411	155	.BYTE	155
4843	027412	156	.BYTE	156
4844	027413	157	.BYTE	157
4845	027414	160	.BYTE	160
4846	027415	161	.BYTE	161
4847	027416	162	.BYTE	162
4848	027417	163	.BYTE	163
4849	027420	164	.BYTE	164
4850	027421	165	.BYTE	165
4851	027422	166	.BYTE	166
4852	027423	167	.BYTE	167
4853	027424	170	.BYTE	170
4854	027425	171	.BYTE	171
4855	027426	172	.BYTE	172
4856	027427	173	.BYTE	173
4857	027430	174	.BYTE	174
4858	027431	175	.BYTE	175
4859	027432	176	.BYTE	176
4860	027433	177	.BYTE	177
4861	027434	200	.BYTE	200
4862	027435	201	.BYTE	201
4863	027436	202	.BYTE	202
4864	027437	203	.BYTE	203
4865	027440	204	.BYTE	204
4866	027441	205	.BYTE	205
4867	027442	206	.BYTE	206
4868	027443	207	.BYTE	207
4869	027444	210	.BYTE	210
4870	027445	211	.BYTE	211
4871	027446	212	.BYTE	212
4872	027447	213	.BYTE	213
4873	027450	214	.BYTE	214
4874	027451	215	.BYTE	215
4875	027452	216	.BYTE	216
4876	027453	217	.BYTE	217
4877	027454	220	.BYTE	220
4878	027455	221	.BYTE	221
4879	027456	222	.BYTE	222
4880	027457	223	.BYTE	223
4881	027460	224	.BYTE	224
4882	027461	225	.BYTE	225
4883	027462	226	.BYTE	226
4884	027463	227	.BYTE	227
4885	027464	230	.BYTE	230
4886	027465	231	.BYTE	231
4887	027466	232	.BYTE	232
4888	027467	233	.BYTE	233
4889	027470	234	.BYTE	234
4890	027471	235	.BYTE	235
4891	027472	236	.BYTE	236

4892	027473	237	.BYTE	237
4893	027474	240	.BYTE	240
4894	027475	241	.BYTE	241
4895	027476	242	.BYTE	242
4896	027477	243	.BYTE	243
4897	027500	244	.BYTE	244
4898	027501	245	.BYTE	245
4899	027502	246	.BYTE	246
4900	027503	247	.BYTE	247
4901	027504	250	.BYTE	250
4902	027505	251	.BYTE	251
4903	027506	252	.BYTE	252
4904	027507	253	.BYTE	253
4905	027510	254	.BYTE	254
4906	027511	255	.BYTE	255
4907	027512	256	.BYTE	256
4908	027513	257	.BYTE	257
4909	027514	260	.BYTE	260
4910	027515	261	.BYTE	261
4911	027516	262	.BYTE	262
4912	027517	263	.BYTE	263
4913	027520	264	.BYTE	264
4914	027521	265	.BYTE	265
4915	027522	266	.BYTE	266
4916	027523	267	.BYTE	267
4917	027524	270	.BYTE	270
4918	027525	271	.BYTE	271
4919	027526	272	.BYTE	272
4920	027527	273	.BYTE	273
4921	027530	274	.BYTE	274
4922	027531	275	.BYTE	275
4923	027532	276	.BYTE	276
4924	027533	277	.BYTE	277
4925	027534	300	.BYTE	300
4926	027535	301	.BYTE	301
4927	027536	302	.BYTE	302
4928	027537	303	.BYTE	303
4929	027540	304	.BYTE	304
4930	027541	305	.BYTE	305
4931	027542	306	.BYTE	306
4932	027543	307	.BYTE	307
4933	027544	310	.BYTE	310
4934	027545	311	.BYTE	311
4935	027546	312	.BYTE	312
4936	027547	313	.BYTE	313
4937	027550	314	.BYTE	314
4938	027551	315	.BYTE	315
4939	027552	316	.BYTE	316
4940	027553	317	.BYTE	317
4941	027554	320	.BYTE	320
4942	027555	321	.BYTE	321
4943	027556	322	.BYTE	322
4944	027557	323	.BYTE	323
4945	027560	324	.BYTE	324
4946	027561	325	.BYTE	325
4947	027562	326	.BYTE	326

4948	027563	327	.BYTE	327
4949	027564	330	.BYTE	330
4950	027565	331	.BYTE	331
4951	027566	332	.BYTE	332
4952	027567	333	.BYTE	333
4953	027570	334	.BYTE	334
4954	027571	335	.BYTE	335
4955	027572	336	.BYTE	336
4956	027573	337	.BYTE	337
4957	027574	340	.BYTE	340
4958	027575	341	.BYTE	341
4959	027576	342	.BYTE	342
4960	027577	343	.BYTE	343
4961	027600	344	.BYTE	344
4962	027601	345	.BYTE	345
4963	027602	346	.BYTE	346
4964	027603	347	.BYTE	347
4965	027604	350	.BYTE	350
4966	027605	351	.BYTE	351
4967	027606	352	.BYTE	352
4968	027607	353	.BYTE	353
4969	027610	354	.BYTE	354
4970	027611	355	.BYTE	355
4971	027612	356	.BYTE	356
4972	027613	357	.BYTE	357
4973	027614	360	.BYTE	360
4974	027615	361	.BYTE	361
4975	027616	362	.BYTE	362
4976	027617	363	.BYTE	363
4977	027620	364	.BYTE	364
4978	027621	365	.BYTE	365
4979	027622	366	.BYTE	366
4980	027623	367	.BYTE	367
4981	027624	370	.BYTE	370
4982	027625	371	.BYTE	371
4983	027626	372	.BYTE	372
4984	027627	373	.BYTE	373
4985	027630	374	.BYTE	374
4986	027631	375	.BYTE	375
4987	027632	376	.BYTE	376
4988	027633	377	.BYTE	377
4989			.EVEN	

4990				
4991	027634	005015	000	
4992	027637	015	020012	020040
4993	027644	000		
4994	027645	015	042012	053105
4995	027652	041511	020105	020040
4996	027660	046440	050101	000
4997	027665	015	042012	050532
4998	027672	040503	043455	052040
4999	027700	051505	042524	020104
5000	027706	044527	044124	000040
5001	027714	005015	054523	052123
5002	027722	046505	041440	047117
5003	027730	044506	052507	040522

RETURN: .ASCIZ <15><12>
SPACE: .ASCIZ <15><12>" "
MSG0: .ASCIZ <15><12>"DEVICE MAP"
MSGX: .ASCIZ <15><12>"DZQCA-G TESTED WITH "
MSG1: .ASCIZ <15><12>"SYSTEM CONFIGURATION"

5004	027736	044524	047117	000
5005	027743	040	040520	051523
5006	027750	042450	024523	020054
5007	027756	000040		
5008	027760	042440	051122	051117
5009	027766	051450	027051	005015
5010	027774	000		
5011	027775	377	005015	020377
5012	030002	050040	020103	020040
5013	030010	020040	051520	020040
5014	030016	051440	000120	
5015				
5016	030022	005015	042504	044526
5017	030030	042503	047040	052117
5018	030036	051040	047125	044516
5019	030044	043516	006472	000012
5020	030052	047101	044503	046114
5021	030060	051101	042511	035123
5022	030066	020040	000040	
5023	030072	041504	030461	024040
5024	030100	026460	032461	035051
5025	030106	020040	000040	
5026	030112	041504	030461	024040
5027	030120	033061	031455	024461
5028	030126	020072	000040	
5029	030132	046113	030461	020054
5030	030140	046104	030461	026101
5031	030146	035102	000040	
5032	030152	050104	030461	024040
5033	030160	026460	032461	035051
5034	030166	020040	000040	
5035	030172	050104	030461	024040
5036	030200	033061	031455	024461
5037	030206	020072	000040	
5038	030212	046504	030461	035101
5039	030220	020040	020040	020040
5040	030226	020040	000040	
5041	030232	047104	030461	020072
5042	030240	020040	020040	020040
5043	030246	020040	000040	
5044	030252	046504	030461	041102
5045	030260	020072	020040	020040
5046	030266	020040	000040	
5047	030272	043513	030461	020072
5048	030300	020040	020040	020040
5049	030306	020040	000040	
5050	030312	054104	030461	020072
5051	030320	020040	020040	020040
5052	030326	020040	000040	
5053	030332	046104	030461	042103
5054	030340	020105	030050	030455
5055	030346	024465	000072	
5056	030352	046104	030461	042103
5057	030360	024105	033061	031455
5058	030366	024461	000072	
5059	030372	045104	030461	024040

MSGPAS: .ASCIZ " PASS(ES), "

MSGERR: .ASCIZ " ERROR(S)."<15><12>

MSGHED: .ASCIZ <377><15><12><377>" PC PS SP"

UNIT DEVADR IS SHOULD BE"
MSGDNR: .ASCIZ <15><12>"DEVICE NOT RUNNING:"<15><12>

MSGMAP: .ASCIZ "ANCILLARIES: "

.ASCIZ "DC11 (0-15): "

.ASCIZ "DC11 (16-31): "

.ASCIZ "KL11, DL11A,B: "

.ASCIZ "DP11 (0-15): "

.ASCIZ "DP11 (16-31): "

.ASCIZ "DM11A: "

.ASCIZ "DN11: "

.ASCIZ "DM11BB: "

.ASCIZ "KG11: "

.ASCIZ "DX11: "

.ASCIZ "DL11CDE (0-15):"

.ASCIZ "DL11CDE(16-31):"

.ASCIZ "DJ11 (0-7): "

```

5060 030400 026460 024467 020072
5061 030406 020040 000040
5062
5063 030412 177777 177777 000 PADDERS: .ASCIZ <377><377><377><377>
5064
5065
5066 030420 .EVEN
5067
5068 030420 000000 DJ.XTBL: 0 ;DJ11 XMIT TABLE BUFFERS
5069 031016 031016 .=DJ.XTBL+376
5070 031016 000000 0,0
5071
5072 031022 000000 DJ.EXP: 0 ;DJ11 RECV TABLE BUFFERS
5073 031420 031420 .=DJ.EXP+376
5074 031420 000000 0,0
5075
5076 .EVEN

```


5077
5078 031424 000000
5079
5080
5081
5082
5083
5084
5085
5086
5087
5088
5089
5090
5091
5092
5093
5094
5095
5096
5097
5098
5099
5100
5101
5102
5103
5104
5105
5106
5107
5108
5109
5110
5111
5112
5113
5114
5115
5116
5117
5118

000001

```

LINKER: 0 ;BEGINNING OF LINKER AREA
:LINKER AREA LOOKS LIKE THIS
:VEC300: JSR %5,2#RCV.ISR FOR COMM DEVICE 0
          DEV.# =0
:VEC304: JSR %5,2#XMT.ISR
          DEV.# =0
          "
          "
          SAME DEVICE (IF MORE THAN ONE) NEXT LINE NUMBER
          "
          "
          .
:VECXXX: JSR %5,2#RCV.ISR FOR NEXT DEVICE TYPE
          DEV.# =0
          "
          =0
:THE VECTOR FOR THE FIRST COMM DEVICE SHOULD BE 300
:THE CONTENTS OF 300 SHOULD BE # LINKER AND
:LINKER SHOULD POINT TO THE INTERRUPT SERVICE ROUTINE
:FOR THAT DEVICE AND SHOULD INTICATE LINE ZERO
:NOTE: SPECIAL PATCHES FOR PRODUCTION CHECKOUT WILL MAKE
:EXCEPTIONS TO THIS RULE-OK
:FOLLOWING THE LINKER AREA IS THE DM CORE.
:CATO: =CURRENT ADDRESS TABLE FOR CHANNEL 0
"+40: =WORD COUNT
"+100: =BIT ASSEMBLY TABLE
"+140: =UNUSED
"+200: =TUMBLE TABLE
:CAT1: =CURRENT ADDRESS TABLE FOR CHANNEL 1
:ETC.
          .END

```

A = 000400

564#	565#	566#	567#	568#	569#	570#	571#	572#	573#	574#	575#	576#
577#	578#	579#	580#	581#	582#	583#	584#	585#	586#	587#	588#	589#
590#	591#	592#	593#	594#	595#	596#	2568#	2569#	2570#	2571#	2572#	2573#
2574#	2575#	2576#	2577#	2578#	2579#	2580#	2581#	2582#	2583#	2584#	2585#	2586#
2587#	2588#	2589#	2590#	2591#	2592#	2593#	2594#	2595#	2596#	2597#	2598#	2599#
2600#	2664#	2665#	2666#	2667#	2668#	2669#	2670#	2671#	2672#	2673#	2674#	2675#
2676#	2677#	2678#	2679#	2680#	2681#	2682#	2683#	2684#	2685#	2686#	2687#	2688#
2689#	2690#	2691#	2692#	2693#	2694#	2695#	2696#	2762#	2763#	2764#	2765#	2766#
2767#	2768#	2769#	2770#	2771#	2772#	2773#	2774#	2775#	2776#	2777#	2778#	2779#
2780#	2781#	2782#	2783#	2784#	2785#	2786#	2787#	2788#	2789#	2790#	2791#	2792#
2793#	2794#	2796#	2797#	2798#	2799#	2800#	2801#	2802#	2803#	2804#	2805#	2806#
2807#	2808#	2809#	2810#	2811#	2812#	2813#	2814#	2815#	2816#	2817#	2818#	2819#
2820#	2821#	2822#	2823#	2824#	2825#	2826#	2827#	2828#	2914#	2915#	2916#	2917#
2918#	2919#	2920#	2921#	2922#	2923#	2924#	2925#	2926#	2927#	2928#	2929#	2930#
2932#	2933#	2934#	2935#	2936#	2937#	2938#	2939#	2940#	2941#	2942#	2943#	2944#
2945#	2946#	2947#	2948#	2950#	2951#	2952#	2953#	2954#	2955#	2956#	2957#	2958#
2959#	2960#	2961#	2962#	2963#	2964#	2965#	2966#	2968#	2969#	2970#	2971#	2972#
2973#	2974#	2975#	2976#	2977#	2978#	2979#	2980#	2981#	2982#	2983#	2984#	2986#
2987#	2988#	2989#	2990#	2991#	2992#	2993#	2994#	2995#	2996#	2997#	2998#	2999#
3000#	3001#	3002#	3078#	3079#	3080#	3081#	3082#	3083#	3084#	3085#	3086#	3087#
3088#	3089#	3090#	3091#	3092#	3093#	3094#	3095#	3096#	3097#	3098#	3099#	3100#
3101#	3102#	3103#	3104#	3105#	3106#	3107#	3108#	3109#	3110#	3111#	3112#	3113#
3114#	3115#	3116#	3117#	3118#	3119#	3120#	3121#	3122#	3123#	3124#	3125#	3126#
3127#	3128#	3129#	3130#	3131#	3132#	3133#	3134#	3135#	3136#	3137#	3138#	3139#
3140#	3141#	3142#	3816#	3817#	3818#	3819#	3820#	3821#	3822#	3823#	3824#	3825#
3826#	3827#	3828#	3829#	3830#	3831#	3832#	3833#	3834#	3835#	3836#	3837#	3838#
3839#	3840#	3841#	3842#	3843#	3844#	3845#	3846#	3847#	3848#	4733#	4734#	4735#
4736#	4737#	4738#	4739#	4740#	4741#	4742#	4743#	4744#	4745#	4746#	4747#	4748#
4749#	4750#	4751#	4752#	4753#	4754#	4755#	4756#	4757#	4758#	4759#	4760#	4761#
4762#	4763#	4764#	4765#	4766#	4767#	4768#	4769#	4770#	4771#	4772#	4773#	4774#
4775#	4776#	4777#	4778#	4779#	4780#	4781#	4782#	4783#	4784#	4785#	4786#	4787#
4788#	4789#	4790#	4791#	4792#	4793#	4794#	4795#	4796#	4797#	4798#	4799#	4800#
4801#	4802#	4803#	4804#	4805#	4806#	4807#	4808#	4809#	4810#	4811#	4812#	4813#
4814#	4815#	4816#	4817#	4818#	4819#	4820#	4821#	4822#	4823#	4824#	4825#	4826#
4827#	4828#	4829#	4830#	4831#	4832#	4833#	4834#	4835#	4836#	4837#	4838#	4839#
4840#	4841#	4842#	4843#	4844#	4845#	4846#	4847#	4848#	4849#	4850#	4851#	4852#
4853#	4854#	4855#	4856#	4857#	4858#	4859#	4860#	4861#	4862#	4863#	4864#	4865#
4866#	4867#	4868#	4869#	4870#	4871#	4872#	4873#	4874#	4875#	4876#	4877#	4878#
4879#	4880#	4881#	4882#	4883#	4884#	4885#	4886#	4887#	4888#	4889#	4890#	4891#
4892#	4893#	4894#	4895#	4896#	4897#	4898#	4899#	4900#	4901#	4902#	4903#	4904#
4905#	4906#	4907#	4908#	4909#	4910#	4911#	4912#	4913#	4914#	4915#	4916#	4917#
4918#	4919#	4920#	4921#	4922#	4923#	4924#	4925#	4926#	4927#	4928#	4929#	4930#
4931#	4932#	4933#	4934#	4935#	4936#	4937#	4938#	4939#	4940#	4941#	4942#	4943#
4944#	4945#	4946#	4947#	4948#	4949#	4950#	4951#	4952#	4953#	4954#	4955#	4956#
4957#	4958#	4959#	4960#	4961#	4962#	4963#	4964#	4965#	4966#	4967#	4968#	4969#
4970#	4971#	4972#	4973#	4974#	4975#	4976#	4977#	4978#	4979#	4980#	4981#	4982#
4983#	4984#	4985#	4986#	4987#	4988#	4989#						

ACTIVE= 004000
ACTVOC 022724
ADS 021706
AD.DC 001600
AD.DJ 001632
AD.DLC 001630
AD.DLX 001700
AD.DMA 001606
AD.DMB 001612

2728
4440#
4242#
650# 768 1392
663# 1861 3876 3942
662# 1800 3750
689# 3717
653#
655# 2153 3166

DM.TT	014752	1495	1555	2876	2905*	2985#				
DM.WCT	014712	1494	1554	2967#						
DM.XMT	014222	1198	2833#							
DM.X1	014254	2842	2844#							
DM.X2	014262	2846	2848#							
DM.X3	014274	2850#	2858							
DM.X4	014320	2851	2856#							
DM1188	011100	2119	2149#							
DNACT	011026	2118#								
DNADRS	023010	3039	4473#							
DNBK1	011056	2126#	2133							
DNBK2	011072	2127	2131#							
DNDATA	015242	1636	2121	3044	3063*	3077#				
DNIFLG	015366	3040*	3065*	3070	3155#					
DNINIT	006340	1469	1587	1634#						
DNPR1	006364	1639#	1652							
DNPR2	006426	1640	1649#							
DNPR3	006434	1648	1651#							
DNPR4	006400	1642#	1646							
DNRCVD	015354	3053	3149#							
DNXMTD	015342	1644	2130	3143#						
DN.ISR	015054	1203	3031#							
DN.I1	015114	3042#	3069							
DN.I2	015206	3043	3066#							
DONE =	000200	390#	2147#	2177	3265					
DPADRS	023004	4471#								
DP1A	005476	1436#	1444#	1460	1461*					
DP1B	005464	1442#	1463							
DP1C	005530	1445	1454#							
DPINIT	005420	1411	1434#							
DPRCVD	014022	1438	2711#	2712	2714	2720*	2725*	2726*	2760#	
DPXMTD	014122	1437	2721#	2748	2749*	2754*	2756*	2795#		
DP.RCV	013532	1190	2698#							
DP.R1	013660	2713	2726#							
DP.R2	013654	2715	2725#							
DP.R4	013672	2727	2729#							
DP.R5	013666	2724	2728#							
DP.XMT	013704	1191	2734#							
DP.X2	013770	2747	2753#							
DP.X3	014016	2750	2755	2758#						
DST	006454	1660#	1676#	1682	1691	1704	1763	1770*	1777*	
DS.0	006664	1704#								
DS.1	006674	1706#	1709#							
DS.2	006702	1708#	1710							
DS.3	015544	3220#	3222							
DS.4	006634	1692#	1694							
DS.5	006654	1700#	1702							
DS1	016036	3293	3296#							
DS2	016054	3297	3300#							
DS3	016072	3301	3304#							
DS4	016110	3305	3309#							
DTDONE	016634	3306	3431#							
DTERR	016676	3435	3442#							
DVICE	002052	715	722#	726	4583	4717				
DXBA	017406	3225*	3374*	3388*	3576#					
DXBC	017410	3226*	3375*	3378*	3389*	3392*	3577#			

		1801*	1803*	1862*	1879*	1883*	1884*	1885*	1888	1905*	2059*	2106*	2110	2123*
		2124*	2125*	2132*	2199*	2203	2213	2214	2216*	2367*	2417*	2461*	2465	2468
		3046*	3047	3720*	3721	3753*	3754	3770*	3771	3789*	3860	3871*	3873*	3884*
		3885	3903	3929	3944*	3945	3949	3965*	3967	3973	4000*	4400*	4697	4710*
R3S	015656	3257#	3421											
R4	=%000004	338#	771*	895	1172*	1182*	1195*	1211	1261*	1287	1298	1307	1317	1319
		1394*	1396	1403*	1437*	1457*	1495*	1515*	1531	1556*	1562*	1563	1582	1641*
		1645*	1802*	1804	1839*	1863*	1868	1897*	2060*	2129*	2200*	2209	2213	2464*
		2467	2471*	3047*	3048*	3721*	3722*	3723*	3754*	3755*	3771*	3772*	3788*	3790*
		3861	3885*	3886*	3888*	3902*	3903*	3904*	3905	3906*	3907*	3930	3949*	3950*
		3967*	3968*	3969*	3971*	3972*	3973	3976*	3990*	3991*	3999*	4401*	4629	4630*
		4650	4655*	4698	4709*									
R4S	015652	3255#	3420											
R5	=%000005	339#	903	920	935*	1281	1282	1302	1328	1329*	1555*	1594	1606	1614
		1615	1616	1624	1625	1626	2166*	2174	2178*	2461	2476*	2523	2541	2607
		2632	2700	2736	2835	2848*	2852*	2853*	2856	2867	2876*	2877	2880	2886
		2898*	2901	2903*	2905	3033	3161	3712	3745	3851*	3863	3932	4003*	4495
		4496	4699	4708*										
R5S	015646	3253#	3419											
R6	=%000006	340#	2326*	2327*	4560									
R7	=%000007	341#	2329											
SAT	021124	1979	4117#											
SAVCC	022730	4442#												
SAVE	007436	1826*	1827	1829	1845#									
SAVPC	022726	4441#												
SAVRG	023144	1613	1623	4494	4519#									
SAVR2	022732	4443#												
SAVR3	022734	4444#												
SAVR4	022736	4445#												
SAVTTP	016226	1734*	3264	3355#	3418*									
SCALD.	017472	3607#												
SCNENA=	000040	2146#	2170	2177										
SCNT1	011134	2157#												
SCNT1A	011154	2162#	2165											
SCNT1B	011206	2169#	2180											
SCNT1D	011240	2175	2177#											
SCOPE =	104400	349#	2091	2120	2151	2192	2308	2317						
SEGCNT	021122	4116#												
SENSE =	000004	416#	3542											
SENSEC=	000004	3182#	3635											
SERV	021710	1999*	4243#	4263	4265	4267	4289*	4310*	4330*					
SINPUT	016236	3349	3368#											
SKIPVA	004776	1213	1215	1217	1219	1322#								
SKIPV1	005006	1324#	1327											
SKIPV2	005020	1325	1328#											
SLR =	177774	345#	695*											
SM =	000100	413#	1770											
SMEN =	010000	395#												
SNS =	000010	365#	3523*	3526*	3548	3558*								
SNULL	016234	3348	3366#											
SOUTPT	016336	3350	3382#											
S01	016434	3384	3394#											
SP =	=%000006	343#	692*	917	924	1375	2521*	2522*	2539*	2540*	2605*	2606*	2630*	2631*
		2698*	2699*	2734*	2735*	2833*	2834*	2865*	2866*	3031*	3032*	3159*	3160*	3281*
		3286	3710*	3711*	3743*	3744*	3849	3850	3851	3858*	3859*	3860*	3861*	3927*
		3928*	3929*	3930*	3999	4000	4001	4002	4003	4582*	4700	4704*		

MAINDEC-11-DZQCA-G COMMUNICATION TEST PROGRAM (CTP)
 DZQACG.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

SPACE	027637	2400	4556	4559	4565	4992#												
SPW	006446	1657#	1670*	1684	1698	1759												
SSCUE	016522	3352	3409#															
SSTAT	016436	3351	3396#															
SS1	016462	3398	3401#															
SS2	016520	3400	3406#															
SS3	016572	3411	3417#															
SS4	016574	3345	3418#															
STA	= 000002	435#	3399	3468	3483													
STALL	= 104400	3021#																
START	000200	456#																
STCHO	003150	956	961#															
STCH1	003216	962	968	973#														
STCH10	003654	1060	1068#															
STCH11	003700	1069	1077#															
STCH12	003724	1078	1086#															
STCH13	003750	1087	1096#															
STCH14	003774	1097	1106#															
STCH15	004020	1107	1116#															
STCH16	004044	1117	1126#															
STCH2	003242	974	979#															
STCH3	003310	980	986	991#														
STCH30	004070	1052	1127	1133#														
STCH4	003334	992	997#															
STCH5	003360	998	1003#															
STCH6	003404	1004	1009#															
STCH7	003430	1010	1015#															
STCH70	003504	1026#	1035															
STCH8	003534	1016	1036#															
STCH80	003520	1027	1031#															
STCH9	003604	1037	1043	1049	1051#													
STEP	= 000400	2144#	2163															
STKSTB	= 000002	401#	3296	3447														
STOHLT	022762	4455#																
STORD	022742	4447#																
STOR1	022744	4448#																
STOR2	022746	4449#																
STOR3	022750	4450#																
STOR4	022752	4451#																
STOR5	022754	4452#																
STOR6	022756	4453#																
STS	= 000016	368#	380#	3404	3415	3465*	3504*	3517*	3520*	3530*	3534							
ST1	016334	3370	3380#															
SVLADS	024022	4668	4672	4678#	4686													
SVRPC	023164	4519#	4526#															
SWCHK	004104	1137#	1153															
SWCHK2	004164	1139	1151#															
SWITCH	003042	720	935#	1149														
SWR	= 177570	347#	348	709	878	954	960	966	972	978	984	990	996	1002				
		1008	1014	1020	1021	1030	1041	1047	1056	1065	1074	1082	1092	1102				
		1112	1122	1132	2306	2314	2391	2421	2433	4543	4548	4571	4575	4669				
		4671																
SW.ANC	001250	527#	960*	1951	1964	1975	1995	2004	2018	2365	2478	2485	2497					
SW.DC1	001252	528#	966*	1172	1391													
SW.DC2	001254	529#	972*															
SW.DJ	001302	540#	1056*	1059	1068	1077	1086	1096	1106	1116	1126	1261	1860					

MAINDEC-11-DZQCA-G COMMUNICATION TEST PROGRAM (CTP)
DZQCA.G.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

E12
MACY11 27(732) 17-SEP-76 15:12 PAGE 128

.TYPE 023554 4599* 4600 4616*

MAINDEC-11-DZQCA-G
DZQCA.G.P:1

COMMUNICATION TEST PROGRAM (CTP)
CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ADC	2310	4285	4306	4327											
ADD	870	905	1245	1289	1309	1326	1332	1400	1426	1427	1454	1455	1461	1511	1514
	1528	1583	1650	1671	1673	1675	1677	1679	1759	1836	1896	1898	1905	1930	1933
	1936	2182	2221	2383	2416	2528	2546	2706	2742	2840	2872	2900	3039	3166	3313
	3436	3438	3549	3561	3717	3750	3876	3903	3942	3971	4284	4305	4326	4513	4566
	4608	4611													
ASL	1032	1235	1246	1296	1315	1403	1429	1459	1533	1542	1584	1651	1758	1839	1890
	1897	2124	2125	2183	2222	2471	2524	2526	2527	2542	2544	2545	2608	2610	2611
	2633	2635	2636	2701	2703	2704	2737	2739	2740	2836	2838	2839	2857	2868	2870
	2871	3034	3036	3038	3162	3164	3165	3275	3283	3335	3338	3713	3715	3716	3746
	3748	3749	3967	3868	3869	3873	3904	3934	3936	3937	3941	3972			
ASR	806	813	3049	3050	3951	3977									
BCC	910	1297	1316	1404	1430	1460	1534	1543	1585	1652	1840	1891	2184	2223	2472
	2858														
BCS	2081														
BEG	701	715	863	872	876	908	941	956	962	968	974	980	986	992	998
	1004	1010	1016	1027	1037	1043	1052	1060	1069	1078	1087	1097	1107	1117	1127
	1139	1229	1233	1241	1243	1285	1305	1379	1381	1389	1397	1402	1411	1422	1435
	1445	1507	1538	1540	1559	1607	1635	1640	1717	1774	1797	1805	1828	1830	1838
	1852	1869	1891	1941	1943	1952	1965	1976	1984	1996	2005	2007	2019	2036	2041
	2048	2057	2086	2089	2108	2111	2119	2150	2156	2175	2191	2197	2210	2325	2356
	2370	2406	2439	2443	2448	2456	2466	2470	2479	2486	2555	2558	2644	2646	2648
	2713	2715	2842	2887	3054	3204	3269	3279	3290	3293	3297	3301	3305	3398	3435
	3453	3474	3481	3484	3494	3500	3502	3519	3522	3529	3535	3541	3564	3787	3803
	3974	4037	4039	4043	4144	4157	4160	4194	4217	4264	4363	4506	4544	4602	4643
	4646	4674	4686												
BGE	3060	3437	4195												
BGT	747	869	1153	2133	2418										
BHI	2313														
BIC	785	804	1333	1520	1522	1776	1980	2171	2177	2409	2480	2482	2722	2728	2757
	2859	2907	3041	3211	3212	3229	3231	3265	3270	3342	3723	3772	3790	3886	3907
	3969	4041	4132	4562	4612										
BICB	2883	3048	3273	3736	3991										
BIS	782	801	836	841	847	850	853	856	859	877	903	906	1022	1423	1424
	1441	1446	1448	1524	1525	1576	1644	1647	1668	1772	1775	1779	1780	1792	1834
	1885	1959	1960	1971	1986	2002	2130	2163	2169	2170	2219	2357	2467	2468	2487
	2488	2562	2655	2723	2729	2751	2854	2906	3035	3074	3170	3214	3227	3228	3230
	3233	3251	3379	3393	3405	3416	3520	3523	3526	3530	3722	3755	3807	3888	3950
	3997	4024	4025	4094	4095	4229	4230	4278	4283	4299	4304	4320	4325	4333	4334
	4355	4356	4423	4424											
BISB	3373	3387	3403	3414	4647										
BIT	709	878	955	961	973	979	991	997	1003	1009	1015	1026	1036	1051	1059
	1068	1077	1086	1096	1106	1116	1126	1138	1232	1240	1242	1287	1307	1388	1396
	1421	1434	1444	1506	1537	1558	1639	1724	1796	1804	1851	1868	1880	1942	1951
	1964	1975	1995	2004	2006	2018	2155	2158	2196	2306	2314	2391	2421	2433	2465
	2478	2485	2615	2640	2707	2716	2746	2841	2850	3068	3268	3292	3296	3300	3304
	3432	3434	3447	3452	3499	3501	3518	3528	4016	4033	4137	4151	4188	4206	4265
	4267	4370	4543	4548	4575	4597	4634	4669	4671						
BITB	3563														
BLE	1295	1314	3377	3391											
BLO	2053														
BLOS	2068														
BLT	3315	4143													
BMI	1325	1463	1809	1818	2530	2551	2613	2638	2744	2846	2874	2878	3168	3719	3752
	3881	3918	3947	3995	4078	4127	4168	4359	4391	4564					
BNE	707	879	948	1035	1237	1248	1288	1308	1416	1468	1519	1619	1629	1646	1665

MAINDEC-11-DZQCA-G COMMUNICATION TEST PROGRAM (CTP)
 DZQCA.G.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

	1683	1694	1702	1710	1725	1746	1748	1766	1811	1820	1901	1938	1982	2065	2114
	2159	2165	2180	2215	2218	2307	2311	2315	2377	2381	2386	2392	2422	2434	2441
	2616	2621	2641	2652	2708	2717	2727	2747	2750	2755	2851	2884	2892	2902	3069
	3071	3210	3222	3344	3370	3384	3400	3411	3433	3448	3497	3543	4017	4021	4034
	4047	4082	4085	4088	4140	4154	4191	4210	4213	4266	4268	4287	4308	4329	4371
	4373	4410	4499	4515	4549	4551	4576	4598	4610	4635	4649	4651	4668	4670	4672
BPL	4676	4707													
	840	2027	2029	2127	2173	2206	2548	2882	3043	3045	3769	4075	4138	4152	4189
BR	4207	4276	4297	4318	4367	4395	4415	4572	4605						
	726	912	1049	1290	1299	1310	1318	1327	1371	1406	1526	1578	1587	1595	1648
	1653	1781	1842	1903	1906	1947	2388	2474	2560	2724	2895	2904	3207	3271	3345
	3524	3566	3805	4162	4170	4173	4181	4184	4197	4199	4202	4219	4226	4291	4312
	4361	4365	4369	4375	4393	4408	4570	4606	4624	4626	4632	4720			
BVC	2083														
CLC	1428	1458													
CLR	693	697	698	699	704	744	842	896	938	1157	1280	1301	1456	1517	1681
	1731	1732	1771	1807	1816	1859	1865	1866	1872	1879	1924	1999	2000	2001	2030
	2071	2072	2115	2116	2152	2161	2193	2309	2419	2445	2446	2898	3040	3057	3061
	3208	3288	3312	3333	3368	3382	3396	3409	3449	3450	3451	3517	3558	4288	4290
CLRB	4309	4311	4330	4331	4332	4497	4581	4627	4705						
CMP	1453	1483	1642	1744	1777	2720	2721	3220	4022	4631	4633	4653			
	700	745	746	917	924	1375	1380	1577	1649	1682	1829	1937	1940	1981	2052
	2056	2064	2067	2085	2088	2107	2110	2113	2174	2209	2213	2214	2217	2312	2369
	2382	2385	2414	2447	2455	2891	2901	3053	3059	3203	3314	3369	3376	3383	3390
	3397	3399	3480	3483	3973	4081	4142	4156	4159	4193	4212	4216	4286	4307	4328
	4609	4650													
CMPB	1034	1606	1827	2554	2557	2620	2645	2647	2651	2712	2714	2886	3205	3542	3786
	3802	4020	4038	4042	4046	4372	4409	4675							
COM	1137	1140	1151	1442	1562	1931	1935	1946	2408	2410	2890	2894	2896	3062	
COMB	2128														
DEC	868	907	947	1152	1236	1247	1294	1313	1324	1401	1415	1518	1618	1628	1645
	1693	1701	1709	1745	1747	1765	1837	1899	1945	2132	2164	2179	2417	2469	3209
	3221	4084	4087	4215	4498	4514									
DECB	2754	4563													
EMT	353														
HALT	447	728	861	952	958	964	970	976	982	988	994	1000	1006	1012	1018
	1028	1039	1045	1055	1063	1072	1080	1090	1100	1110	1120	1130	1155	3442	4270
	4573	4702	4719												
INC	904	1033	1293	1312	1669	1810	1819	1887	1939	2178	2390	2420	2654	2897	3065
	3067	3341	3916	3992	4049	4080	4128	4158	4169	4198	4289	4310	4574	4706	
INCB	1484	2532	2559	2619	2650	2725	2726	2749	3735	3804	3906	3990	4019	4045	4134
	4148	4201	4644	4648	4678	4683									
IOT	354														
JMP	456	460	925	1149	1469	1666	2224	2318	2329	2458	2535	2563	2624	2656	2730
	2752	2758	2860	2908	3075	3171	3215	3295	3299	3303	3307	3317	3425	3536	3739
	3808	3920	4129	4135	4149	4214	4525	4536	4583	4717					
JSR	719	719	720	721	722	723	765	775	779	783	788	791	794	797	802
	807	810	814	819	823	827	883	935	1176	1183	1189	1196	1202	1207	1213
	1215	1217	1219	1224	1225	1252	1253	1260	1262	1270	1291	1292	1311	1322	1472
	1488	1544	1565	1570	1613	1620	1623	1630	1790	1793	1854	2038	2043	2333	2338
	2340	2342	2344	2346	2348	2350	2353	2358	2360	2362	2393	2399	2402	2415	2423
	2426	2429	2432	2435	2437	2444	2451	2457	2481	2494	2498	2499	3213	3294	3298
	3302	3306	3339	3455	3486	4494	4500	4511	4545	4547	4555	4558	4561	4569	
MOV	691	692	694	695	696	702	703	717	741	742	743	748	749	768	769
	770	771	772	773	778	787	800	817	818	822	826	833	834	845	848
	851	854	857	860	861	864	865	866	867	881	882	894	895	897	919

TRAP	349	3021													
TST	706	835	846	849	852	855	858	862	871	875	898	911	918	920	940
	967	985	1031	1042	1210	1211	1228	1298	1317	1319	1328	1370	1378	1405	1410
	1425	1462	1467	1521	1529	1530	1531	1539	1541	1580	1581	1582	1634	1664	1716
	1773	1841	1888	1983	2028	2035	2040	2047	2118	2131	2149	2181	2190	2220	2324
	2355	2376	2379	2405	2438	2440	2442	2473	2475	2533	2547	2553	2618	2844	2856
	2877	3058	3066	3070	3278	3289	3343	3410	3473	3493	3496	3521	3534	3737	3767
	3768	3879	3917	4074	4126	4167	4263	4275	4296	4317	4362	4366	4394	4505	4512
	4550	4571	4667	4685											
TSTB	839	1808	1817	2026	2126	2172	2205	2529	2550	2612	2637	2643	2743	2873	3042
	3167	3540	3718	3751	3945	3993	4036	4077	4358	4390	4414	4601	4604	4642	4645
	4673														
.ABS	3														
.ASCIZ	1142	4578	4717	4991	4992	4994	4997	5001	5005	5008	5011	5016	5020	5023	5026
	5029	5032	5035	5038	5041	5044	5047	5050	5053	5056	5059	5063			
.BLKW	4657														
.BYTE	3078	3079	3080	3081	3082	3083	3084	3085	3086	3087	3088	3089	3090	3091	3092
	3093	3094	3095	3096	3097	3098	3099	3100	3101	3102	3103	3104	3105	3106	3107
	3108	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3119	3120	3121	3122
	3123	3124	3125	3126	3127	3128	3129	3130	3131	3132	3133	3134	3135	3136	3137
	3138	3139	3140	3141	3650	3651	3652	3653	3654	3655	3656	3657	3658	3659	3660
	3661	3662	3663	3664	3665	3669	3670	3671	3672	3673	3674	3675	3676	3677	3678
	3679	3680	3681	3682	3683	3684	4733	4734	4735	4736	4737	4738	4739	4740	4741
	4742	4743	4744	4745	4746	4747	4748	4749	4750	4751	4752	4753	4754	4755	4756
	4757	4758	4759	4760	4761	4762	4763	4764	4765	4766	4767	4768	4769	4770	4771
	4772	4773	4774	4775	4776	4777	4778	4779	4780	4781	4782	4783	4784	4785	4786
	4787	4788	4789	4790	4791	4792	4793	4794	4795	4796	4797	4798	4799	4800	4801
	4802	4803	4804	4805	4806	4807	4808	4809	4810	4811	4812	4813	4814	4815	4816
	4817	4818	4819	4820	4821	4822	4823	4824	4825	4826	4827	4828	4829	4830	4831
	4832	4833	4834	4835	4836	4837	4838	4839	4840	4841	4842	4843	4844	4845	4846
	4847	4848	4849	4850	4851	4852	4853	4854	4855	4856	4857	4858	4859	4860	4861
	4862	4863	4864	4865	4866	4867	4868	4869	4870	4871	4872	4873	4874	4875	4876
	4877	4878	4879	4880	4881	4882	4883	4884	4885	4886	4887	4888	4889	4890	4891
	4892	4893	4894	4895	4896	4897	4898	4899	4900	4901	4902	4903	4904	4905	4906
	4907	4908	4909	4910	4911	4912	4913	4914	4915	4916	4917	4918	4919	4920	4921
	4922	4923	4924	4925	4926	4927	4928	4929	4930	4931	4932	4933	4934	4935	4936
	4937	4938	4939	4940	4941	4942	4943	4944	4945	4946	4947	4948	4949	4950	4951
	4952	4953	4954	4955	4956	4957	4958	4959	4960	4961	4962	4963	4964	4965	4966
	4967	4968	4969	4970	4971	4972	4973	4974	4975	4976	4977	4978	4979	4980	4981
	4982	4983	4984	4985	4986	4987	4988								
.ENABL	1	4													
.END	5118														
.ENDC	4601	4616	4617	4633	4658	4670	4689	4700	4708	4717	4718				
.EVEN	1148	4717	4989	5066	5076										
.IF	4597	4607	4617	4633	4658	4667	4689	4700	4708	4716	4717				
.IFF	4616	4617	4667	4717											
.IIF	4681	4685	4690	4691	4692										
.IRP	4694	4708													
.LIST	1	295	357	443	447	564	565	566	567	568	569	570	571	572	573
	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588
	589	590	591	592	593	594	595	596	690	732	753	929	1159	1357	1653
	1794	1913	2095	2507	2568	2569	2570	2571	2572	2573	2574	2575	2576	2577	2578
	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591	2592	2593
	2594	2595	2596	2597	2598	2599	2600	2664	2665	2666	2667	2668	2669	2670	2671
	2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686
	2687	2688	2689	2690	2691	2692	2693	2694	2695	2696	2762	2763	2764	2765	2766

2767	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781
2782	2783	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2796	2797
2798	2799	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812
2813	2814	2815	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827
2828	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927
2928	2929	2930	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943
2944	2945	2946	2947	2948	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959
2960	2961	2962	2963	2964	2965	2966	2968	2969	2970	2971	2972	2973	2974	2975
2976	2977	2978	2979	2980	2981	2982	2983	2984	2986	2987	2988	2989	2990	2991
2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3078	3079	3080	3081
3082	3083	3084	3085	3086	3087	3088	3089	3090	3091	3092	3093	3094	3095	3096
3097	3098	3099	3100	3101	3102	3103	3104	3105	3106	3107	3108	3109	3110	3111
3112	3113	3114	3115	3116	3117	3118	3119	3120	3121	3122	3123	3124	3125	3126
3127	3128	3129	3130	3131	3132	3133	3134	3135	3136	3137	3138	3139	3140	3141
3142	3173	3708	3816	3817	3818	3819	3820	3821	3822	3823	3824	3825	3826	3827
3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3839	3840	3841	3842
3843	3844	3845	3846	3847	3848	3855	4428	4489	4538	4587	4617	4659	4692	4717
4724	4733	4734	4735	4736	4737	4738	4739	4740	4741	4742	4743	4744	4745	4746
4747	4748	4749	4750	4751	4752	4753	4754	4755	4756	4757	4758	4759	4760	4761
4762	4763	4764	4765	4766	4767	4768	4769	4770	4771	4772	4773	4774	4775	4776
4777	4778	4779	4780	4781	4782	4783	4784	4785	4786	4787	4788	4789	4790	4791
4792	4793	4794	4795	4796	4797	4798	4799	4800	4801	4802	4803	4804	4805	4806
4807	4808	4809	4810	4811	4812	4813	4814	4815	4816	4817	4818	4819	4820	4821
4822	4823	4824	4825	4826	4827	4828	4829	4830	4831	4832	4833	4834	4835	4836
4837	4838	4839	4840	4841	4842	4843	4844	4845	4846	4847	4848	4849	4850	4851
4852	4853	4854	4855	4856	4857	4858	4859	4860	4861	4862	4863	4864	4865	4866
4867	4868	4869	4870	4871	4872	4873	4874	4875	4876	4877	4878	4879	4880	4881
4882	4883	4884	4885	4886	4887	4888	4889	4890	4891	4892	4893	4894	4895	4896
4897	4898	4899	4900	4901	4902	4903	4904	4905	4906	4907	4908	4909	4910	4911
4912	4913	4914	4915	4916	4917	4918	4919	4920	4921	4922	4923	4924	4925	4926
4927	4928	4929	4930	4931	4932	4933	4934	4935	4936	4937	4938	4939	4940	4941
4942	4943	4944	4945	4946	4947	4948	4949	4950	4951	4952	4953	4954	4955	4956
4957	4958	4959	4960	4961	4962	4963	4964	4965	4966	4967	4968	4969	4970	4971
4972	4973	4974	4975	4976	4977	4978	4979	4980	4981	4982	4983	4984	4985	4986
4987	4988	4989	5077											
564	2568	2664	2762	2796	2914	2932	2950	2968	2986	3078	3816	4732		
	8													
	295	357	443	447	564	565	566	567	568	569	570	571	572	573
	574	575	576	577	578	579	580	581	582	583	584	585	586	587
	589	590	591	592	593	594	595	596	690	732	753	929	1159	1357
1794	1913	2095	2507	2568	2569	2570	2571	2572	2573	2574	2575	2576	2577	2578
2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591	2592	2593
2594	2595	2596	2597	2598	2599	2600	2664	2665	2666	2667	2668	2669	2670	2671
2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686
2687	2688	2689	2690	2691	2692	2693	2694	2695	2696	2762	2763	2764	2765	2766
2767	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781
2782	2783	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2796	2797
2798	2799	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812
2813	2814	2815	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827
2828	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927
2928	2929	2930	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943
2944	2945	2946	2947	2948	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959
2960	2961	2962	2963	2964	2965	2966	2968	2969	2970	2971	2972	2973	2974	2975
2976	2977	2978	2979	2980	2981	2982	2983	2984	2986	2987	2988	2989	2990	2991
2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3078	3079	3080	3081

.MACR
.MACRO
.MCALL
.NLIST

	3082	3083	3084	3085	3086	3087	3088	3089	3090	3091	3092	3093	3094	3095	3096
	3097	3098	3099	3100	3101	3102	3103	3104	3105	3106	3107	3108	3109	3110	3111
	3112	3113	3114	3115	3116	3117	3118	3119	3120	3121	3122	3123	3124	3125	3126
	3127	3128	3129	3130	3131	3132	3133	3134	3135	3136	3137	3138	3139	3140	3141
	3142	3173	3708	3816	3817	3818	3819	3820	3821	3822	3823	3824	3825	3826	3827
	3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3639	3840	3841	3842
	3843	3844	3845	3846	3847	3848	3855	4428	4489	4538	4587	4617	4659	4692	4717
	4724	4733	4734	4735	4736	4737	4738	4739	4740	4741	4742	4743	4744	4745	4746
	4747	4748	4749	4750	4751	4752	4753	4754	4755	4756	4757	4758	4759	4760	4761
	4762	4763	4764	4765	4766	4767	4768	4769	4770	4771	4772	4773	4774	4775	4776
	4777	4778	4779	4780	4781	4782	4783	4784	4785	4786	4787	4788	4789	4790	4791
	4792	4793	4794	4795	4796	4797	4798	4799	4800	4801	4802	4803	4804	4805	4806
	4807	4808	4809	4810	4811	4812	4813	4814	4815	4816	4817	4818	4819	4820	4821
	4822	4823	4824	4825	4826	4827	4828	4829	4830	4831	4832	4833	4834	4835	4836
	4837	4838	4839	4840	4841	4842	4843	4844	4845	4846	4847	4848	4849	4850	4851
	4852	4853	4854	4855	4856	4857	4858	4859	4860	4861	4862	4863	4864	4865	4866
	4867	4868	4869	4870	4871	4872	4873	4874	4875	4876	4877	4878	4879	4880	4881
	4882	4883	4884	4885	4886	4887	4888	4889	4890	4891	4892	4893	4894	4895	4896
	4897	4898	4899	4900	4901	4902	4903	4904	4905	4906	4907	4908	4909	4910	4911
	4912	4913	4914	4915	4916	4917	4918	4919	4920	4921	4922	4923	4924	4925	4926
	4927	4928	4929	4930	4931	4932	4933	4934	4935	4936	4937	4938	4939	4940	4941
	4942	4943	4944	4945	4946	4947	4948	4949	4950	4951	4952	4953	4954	4955	4956
	4957	4958	4959	4960	4961	4962	4963	4964	4965	4966	4967	4968	4969	4970	4971
	4972	4973	4974	4975	4976	4977	4978	4979	4980	4981	4982	4983	4984	4985	4986
	4987	4988	4989	5077											
.PAGE	1846	3708	4538	4617	4659	4692									
.REM	1	3238	3325												
.REPT	447	564	2568	2664	2762	2796	2914	2932	2950	2968	2986	3078	3609	3637	3690
.SBTTL	3816	4733													
1	295	357	443	690	732	753	929	1159	1357	1653	1794	1846	1913	2095	
2	2507	3173	3322	3708	3855	4428	4489	4538	4587	4617	4659	4692	4724	5077	
.TITLE	2														
.WORD	820	1208	1214	1216	1218	1220	1226	1227	1254	1255	1263	1264	1473	1474	1855
	1856	3589	3590	3591	3592	3593	3594	3595	3596	3597	3598	3599	3600	3601	3602
	3603	3604	3605	3609	3610	3611	3612	3613	3614	3615	3616	3617	3618	3619	3620
	3621	3622	3623	3624	3690	3692	3694	3696	3698	3700	3702	3704			

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

* DZQCAG/SOL/CRF/PAGNUM=DZQCAG.SML, DZQCAG
RUN-TIME: 22 40 11 SECONDS
RUN-TIME RATIO: 215/74=2.8
CORE USED: 18K (35 PAGES)