```
:                       .TITLE  ZUDEDO  PDP 11 UDA DRV FMTR
        .REM
```

IDENTIFICATION

PRODUCT CODE: AC-S836D-MC

PRODUCT NAME: CZUDEDO PDP 11 UDA DRV FMTR

PRODUCT DATE: 24 MAY-83

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: DALE KECK

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

This program will format any disk drive connected to a UDA50 disk controller. At the time of this writing, there are three such drives in existance - the RA60, RA80 and RA81. No changes to this program will be needed to format new disk drives as they become available.

There are three ways to format a disk with this program:

1. Reformat - Format the disk with the bad sector information that was written onto the disk at the factory. This is the normal way to format a disk.

2. Reconstruct - Format the disk without using any bad sector information. This should be used only when the bad sector information has been destroyed or for some reason can no longer be read from the disk. This method may also be specified in the disk drive's maintenance manual for special cases (eg. changing an RM/RA80 spare HDA from RM80 format to RA80 format.

3. Restore - Format the disk using bad sector information obtained from a disk file on the XXDP+ system load device. This method is provided for use by manufacturing. No files are provided, nor any method of obtaining the files, at this time.

The format operation is performed by a Diagnostic Utilities and Protocol (DUP) program loaded into the UDA50 disk controller. The host program simply downline loads the DUP program into the UDA50 and monitors its execution. The DUP program obtains parameters from the host program (eg. drive number and format mode) and requests the host program to print error and summary messages. The DUP program is also commonly called a "diagnostic machine" (DM) program.

This program can only format in one mode at a time. In RESTORE mode, only one disk may be selected in the hardware questions or an error message will result and the program will stop.

In REFORMAT and RECONSTRUCT modes, any number of disk drives may be selected. A UDA50 can only format one disk at a time, so each disk on a UDA50 will be selected sequentially. If the disk drives to be formatted are connected to different UDA50s, all UDA50s will be run simultaneously. For example, lets assume three units are selected for formatting in the hardware questions, units 1 and 2 are connected to one UDA50 and unit 3 is connected to a different UDA50 (Unibus addresses are different). This program will automatically start simultaneous format operations on units 1 and 3. When unit 1 finishes (or errors), unit 2 will be started. After units 2 and 3 are finished, the program stops.

This program will stop after each pass (all units formatted once). There is no need to specify a PASS switch on the command line to the Diagnostic Runtime Services (eg. START/PASS:1).

Special provisions have been made to allow this program to run under an APT system in manufacturing. This system does not allow questions to be asked of an operator. Such a condition also exists under XXDP+ when the UAM flag is set. In this condition, only reformat mode can be selected. Selecting RECONSTRUCT or RESTORE will result in an error. Also, a date of 1-JAN 70 will be written on the disk.

1.2 SYSTEM REQUIREMENTS

This program was designed using the PDP-11 Diagnostic Runtime Services revision C. Run time environments are determined by the Runtime Services and may change as new versions of the Services are developed. The initial version will require the following:

PDP 11 Unibus processor
28K words of memory (minimum)
Console terminal
XXDP+ load media containing this program
One or more UDA50 subsystems. The subsystem controllers
type UDA50-A with microcode level 3 or greater.

A system clock - either type L or P - will be used to time the DUP program and report runtime, if available. If no system clock is available, this program cannot detect a hung DUP program.

2.0 OPERATING INSTRUCTIONS

This section contains a brief description of the Runtime Services. For detailed information, refer to the XXDP+ User's Manual (CHQUS).

2.1 COMMANDS

There are eleven legal commands for the Diagnostic Runtime Services (Supervisor). This section lists the commands and gives a very brief description of them. The XXDP+ User's Manual has more details.

COMMAND	EFFECT
START	Start the diagnostic from an initial state
RESTART	Start the diagnostic without initializing
CONTINUE	Continue at test that was interrupted (after ^C)

PROCEED	Continue from an error halt
EXIT	Return to XXDP+ Monitor (XXDP+ OPERATION ONLY!)
ADD	Activate a unit for testing (all units are considered to be active at start time
DROP	Deactivate a unit
PRINT	Print statistical information (see section 4.0)
DISPLAY	Type a list of all device information
FLAGS	Type the state of all flags (see section 2.3)
ZFLAGS	Clear all flags (see section 2.3)

A command can be recognized by the first three characters. So you may, for example, type 'STA" instead of 'START .

2.2 SWITCHES

There are several switches which are used to modify supervisor operation. These switches are appended to the legal commands. All of the legal switches are tabulated below with a brief description of each. In the descriptions below, a decimal number is designated by "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	Execute only those tests specified in the list. List is a string of test numbers, for example /TESTS:1:5:7-10. This list will cause tests 1,5,7,8,9,10 to be run. All other tests will not be run.
/PASS:DDDDD	Execute DDDDD passes (DDDDD = 1 to 64000)
/FLAGS:FLGS	Set specified flags. Flags are described in section 2.3.
/EOP:DDDDD	Report end of pass message after every DDDDD passes only. (DDDDD = 1 to 64000)
/UNITS:LIST	TEST/ADD/DROP only those units specified in the list. List example - /UNITS:0:5:10-12 use units 0,5,10,11,12 (unit numbers = 0-63).

Example of switch usage:

```
START/TESTS:1-5/PASS:1000/EOP:100
```

The effect of this command will be: 1) tests 1 through 5 will be executed, 2) all units will tested 1000 times and 3) the end of pass messages will be printed after each 100 passes only. A switch can be recognized by the first three characters. You may, for example, type "/TES:1 5' instead of "/TESTS:1-5".

Below is a table that specifies which switches can be used by each command.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

Flags are used to set up certain operational parameters such as looping on error. All flags are cleared at startup and remain cleared until explicitly set using the flags switch. Flags are also cleared after a START or RESTART command unless set using the flag switch. The ZFLAGS command may also be used to clear all flags. With the exception of the START, RESTART and ZFLAGS commands, no commands affect the state of the flags; they remain set or cleared as specified by the last flag switch.

FLAG	EFFECT
HOE	Halt on error - control is returned to runtime services command mode
LOE	Loop on error
IER*	Inhibit all error reports
IBE*	Inhibit all error reports except first level (first level contains error type, number, PC, test and unit)
IXE*	Inhibit extended error reports (those called by PRINTX macro's)
PRI	Direct messages to line printer
PNT	Print test number as test executes
BOE	"BELL" on error
UAM	Unattended mode (no manual intervention)
IDU	Inhibit program dropping of units
LOT	Loop on test

*Error messages are described in section 3.1

See the XXDP+ User's Manual for more details on flags. You may specify more than one flag with the FLAG switch. For example, to cause the program to loop on error, inhibit error reports and type a "BELL" on error, you may use the following string:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

When a diagnostic is STARTed, the Runtime Services will prompt the user for hardware information by typing "CHANGE HW (L) ?" . When you answer this question with a "Y", the Runtime Services will ask for the number of units (in decimal). You will then be asked the following questions for each unit. When you answer this question with an "N", the Runtime Services will use the answers built into the program by the SETUP utility (see chapter 6 of the XXDP+ User's Manual). If you have never run the SETUP utility on this program file, the default values listed below (just before the question mark) will be used.

UNIBUS ADDRESS OF UDA (O) 172150 ?

Answer with the address of the UDAIP register of one UDA as addressed by the processor with memory management turned off (i.e., an even 16-bit address in the range of 160000 to 177774).

VECTOR (O) 154 ?

Answer with the interrupt vector address of the UDA. A vector address in the range of 4 to 774 may be specified. The UDA does not have a vector "hard wired" to it, so any vector not being used by this program and XXDP+ may be used.

BR LEVEL (D) 5 ?

Answer with the interrupt priority used by the UDA. Levels 4 to 7 are accepted. This level must match the level "hard wired" in the UDA by the priority plug.

UNIBUS BURST RATE (D) 63 ?

The UDA allows the ability to control the maximum number of words transferred across the UNIBUS each time the UDA becomes master. The default answer of 63 will allow for the fastest execution of this diagnostic program. You may answer with the value your operating system uses or use zero which will tell the UDA to supply a value that should work on any system. A decimal number in the range of 0 to 63 may be specified and all values should work on any system. A larger value will allow for a faster running program. The value will be passed directly to the UDA during initialization.

DRIVE NUMBER (D) 0 ?

Answer with the drive number of the drive you wish to test. This is the number which appears on the "unit plug" on the front of the disk drive. On a multi-unit drive, each sub-unit number on the drive must be tested as a separate unit to completely test the drive. A maximum of eight logical drives may be tested on one UDA at a time (UDA configuration limit).

2.5 SOFTWARE QUESTIONS

After you have answered the hardware questions or after a RESTART or CONTINUE command, the Runtime Services will ask for software parameters. You will be prompted by 'CHANGE SW (L) ?' If you wish to change any parameters, answer by typing "Y". The software questions and the default values are described in the next paragraphs. You may change the default values with the SETUP utility.

REFORMAT USING EXISTING BAD SECTOR INFORMATION (L) Y ?

If this question is answered "YES", then the user wants the REFORMAT mode format operation. REFORMAT mode will use the bad sector information that is already on the disk. Any other mode will destroy this information. If this question is answered "NO", the following will be asked to be sure the user knows what he is doing.

NOT USING EXISTING INFORMATION WILL DESTROY THE FACTORY BAD SECTOR INFORMATION ON THE DISK.
AGAIN REFORMAT USING EXISTING BAD SECTOR INFORMATION (L) Y ?

This is asked to verify that the user does want to destroy the bad sector information on the disk and run another format mode. If this is answered "YES", then the user wants the REFORMAT mode format operation and use the existing bad block information. If again answered "NO", the following question will be asked.

RECONSTRUCT BAD SECTOR INFORMATION (L) Y ?

A "YES" answer will cause a reconstruct mode format operation. If answered "NO", the following will be asked to verify the user really wants the restore mode format.

DO YOU HAVE A FILE ON THE SYSTEM LOAD DEVICE
CONTAINING BAD SECTOR INFORMATION (L) N ?

Note that such a file will not be provided with the diagnostic and this mode is not recommended. The format will begin only on a "YES" answer. Otherwise the following message will be printed and the program will abort.

YOU CANNOT PROCEED WITHOUT SUCH A FILE.
RESTART PROGRAM AND SELECT TO REFORMAT OR RECONSTRUCT DISK.

2.6 MANUAL INTERVENTION QUESTIONS

When the program starts a warning message is printed to warn of improper use of this formatter.

```
WARNING:
      THIS FORMATTER PROGRAM SHOULD NOT BE USED AS A DIAGNOSTIC
      TOOL.  RUN THIS PROGRAM ONLY AS INSTRUCTED IN THE DISK
      DRIVE'S SERVICE MANUAL.

ARE YOU SURE YOU WANT TO RUN THIS FORMATTER (L) N ?
```

You must answer "YES" or the program will abort immediately. This family of disk drives uses a powerful bad block revectoring mechanism to replace blocks that fall on defective areas of the disk media. As a disk is used and defective blocks are detected, DEC operating systems replace the blocks with other blocks on the disk (reserved for this purpose and otherwise inaccessable) so that the disk constantly appears to have its full storage capacity of error free disk blocks. Formatting a disk of this type destroys this history information and is absolutely not recommended except in the cases specifically described in the disk drive's service manual. These disks are fully formatted when shipped from the factory, therefore there is no reason to run this formatter program at installation.

Upon answering "YES" to the above question, the date will be asked for in the format used by the XXDP+ system.

```
ENTER DATE AS DD-MMM-YY (A) 1-JAN-70 ?
```

The default is provided so the user need not supply the date. The date question will normally only be asked one time. If an improper answer is typed, "INPUT ERROR" is printed and the question is asked again. A two or four digit year may be typed. A four digit year must be 1900 or greater (eg. 14-APR-1982). If only two digits are typed, the year is determined as follows:

1. If the number typed is 70 or greater, a 19 is prefixed. Eg.. 1-JAN-70 translates to year 1970 and 25-DEC-99 translates to year 1999.

2. If the number typed is less than 70, a 20 is prefixed. Eg.. 1-APR-21 is translated to year 2021.

If RECONSTRUCT mode is selected, the following question will be asked for each disk before the format operation begins.

```
SERIAL NUMBER FOR UNIT xx UDA AT xxxxxx DRIVE xxx
 (A) ?
```

A decimal number in the range of 0 to 18446744073709551615 must be entered (no default).

If RESTORE mode is selected, the following question will be asked.

NAME OF FILE CONTAINING BAD SECTOR INFORMATION FOR
DISK TO BE FORMATTED (A) ?

If the file named does not exist on the system load device, the program will abort back to the XXDP> prompt after printing an error message.

2.7 EXTENDED P-TABLE DIALOGUE

When you answer the hardware questions, you are building entries in a table that describes the devices under test. The simplest way to build this table is to answer all questions for each unit to be tested. If you have a multiplexed device such as a mass storage controller with several drives or a communication device with several lines, this becomes tedious since most of the answers are repetitious.

To illustrate a more efficient method, suppose you are testing a fictional device, the XY11. Suppose this device consists of a control module with eight units (sub-devices) attached to it. These units are described by the octal numbers 0 through 7. There is one hardware parameter that can vary among units called the Q-factor. This Q-factor may be 0 or 1. Below is a simple way to build a table for one XY11 with eight units.

```
# UNITS (0) ?  8<CR>

UNIT 1
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  0<CR>
Q-FACTOR (0) 0 ?  1<CR>

UNIT 2
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  1<CR>
Q-FACTOR (0) 1 ?  0<CR>

UNIT 3
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  2<CR>
Q-FACTOR (0) 0 ?  <CR>

UNIT 4
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  3<CR>
Q-FACTOR (0) 0 ?  <CR>

UNIT 5
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  4<CR>
Q FACTOR (0) 0 ?  <CR>
```

```
UNIT 6
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  5<CR>
Q-FACTOR (O) O ?  <CR>

UNIT 7
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  6<CR>
Q FACTOR (O) O ?  1<CR>

UNIT 8
CSR ADDRESS (O)  160000<CR>
SUB DEVICE # (O) ?  7<CR>
Q-FACTOR (O) 1 ?  <CR>
```

Notice that the default value for the Q factor changes when a non-default response is given. Be careful when specifying multiple units!

As you can see from the above example, the hardware parameters do not vary significantly from unit to unit. The procedure shown is not very efficient.

The Runtime Services can take multiple unit specifications however. Let's build the same table using the multiple specification feature.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0,1<CR>
Q-FACTOR (O) O ?  1,0<CR>

UNIT 3
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  2-5<CR>
Q-FACTOR (O) O ?  0<CR>

UNIT 7
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  6,7<CR>
Q-FACTOR (O) O ?  1<CR>
```

As you can see in the above dialogue, the runtime services will build as many entries as it can with the information given in any one pass through the questions. In the first pass, two entries are built since two sub-devices and q-factors were specified. The Services assume that the CSR address is 160000 for both since it was specified only once. In the second pass, four entries were built. This is because four sub-devices were specified. The "-" construct tells the Runtime Services to increment the data from the first number to the second. In this case, sub-devices 2, 3, 4 and 5 were specified. (If the sub device were specified by addresses, the increment would be by 2 since addresses must be on an even boundary.) The CSR addresses and Q factors for the four entries are assumed to be 160000 and 0 respectively since they were only specified once. The last two units are specified in the third pass.

The whole process could have been accomplished in one pass as shown below.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0-7<CR>
Q-FACTOR (O) 0 ?  0,1,0,...,1,1<CR>
```

As you can see from this example, null replies (commas enclosing a null field) tell the Runtime Serv ces to repeat the last reply.

2.8 QUICK START UP PROCEDURE

To start-up this program:

1. Boot XXDP+
2. Give the date and answer the LSI and 50HZ (if there is a clock) questions
3. Type "R ZUDEDO"
4. Type "START"
5. Answer the "CHANGE HW" question with "Y"
6. Answer all the hardware questions
7. Answer the "CHANGE SW" question with "N"
8. Answer "Y" to the "ARE YOU SURE ..." question following the warning. Please read the disk drive's service manual before answering this question.
9. Type today's date.

When you follow this procedure you will be using only the defaults for flags and software parameters. These defaults are described in sections 2.3 and 2.5.

```
Sample of terminal dialogue to test two disks on one UDA50:

DR>STA

CHANGE HW (L)  ? Y

# UNITS (D)  ? 2

UNIT 0
UNIBUS ADDRESS OF UDA (O)  172150 ?
VECTOR (O)  154 ?
BR LEVEL (D) 5 ?
UNIBUS BURST RATE (D)  63 ?
DRIVE NUMBER (D)  0 ? 0,1

CHANGE SW (L)  ? N

WARNING:
      THIS FORMATTER PROGRAM SHOULD NOT BE USED AS A DIAGNOSTIC
      TOOL.  RUN THIS PROGRAM ONLY AS INSTRUCTED IN THE DISK
      DRIVE'S SERVICE MANUAL.

ARE YOU SURE YOU WANT TO RUN THIS FORMATTER (L) N ? Y

ENTER DATE AS DD-MMM-YY (A) 1-JAN-70 ? 14-APR-82

UNIT 0 UDA AT 172150 DRIVE 0   RUNTIME 0:00:20
Format begun Version 8
STOPPING THIS FORMAT AFTER THIS POINT WILL MAKE THE DISK
UNUSABLE, AND WILL CAUSE THE DISK TO BE SPUN DOWN WHEN
BROUGHT ONLINE.

UNIT 1 UDA AT 172150 DRIVE 1   RUNTIME 0:00:23
Format begun Version 8
STOPPING THIS FORMAT AFTER THIS POINT WILL MAKE THE DISK
UNUSABLE, AND WILL CAUSE THE DISK TO BE SPUN DOWN WHEN
BROUGHT ONLINE.

UNIT 0 UDA AT 172150 DRIVE 0   RUNTIME 0:42:20
Format completed
     2 Revectored LBNS
     2 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     0 Bad RBNS
     0 Bad blocks in the RCT area due to data errors
     0 Bad blocks in the DBN area due to data errors
     0 Bad blocks in the XBN area due to data errors
     2 Blocks retried on the check pass
FCT used successfully
```

```
UNIT 1 UDA AT 172150 DRIVE 1   RUNTIME 1:25:18
Format completed
   131 Revectored LBNS
   131 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     0 Bad RBNS
     1 Bad blocks in the RCT area due to data errors
     0 Bad blocks in the DBN area due to data errors
     0 Bad blocks in the XBN area due to data errors
   249 Blocks retried on the check pass
FCT used successfully

CZUDE EOP    1
    0 CUMULATIVE ERRORS
DR>
```

Sample of terminal dialogue going through software questions.
Only one disk is being tested.

```
DR>STA

CHANGE HW (L)  ? N

CHANGE SW (L)  ? Y

REFORMAT USING EXISTING BAD SECTOR INFORMATION (L) Y ? Y

WARNING:
      THIS FORMATTER PROGRAM SHOULD NOT BE USED AS A DIAGNOSTIC
      TOOL.  RUN THIS PROGRAM ONLY AS INSTRUCTED IN THE DISK
      DRIVE'S SERVICE MANUAL.

ARE YOU SURE YOU WANT TO RUN THIS FORMATTER (L) N ? Y

ENTER DATA AS DD-MMM-YY (A) 1 JAN-70 ? 14-APR 82

  RUNTIME 0:00:20
Format begun Version 8
STOPPING THIS FORMAT AFTER THIS POINT WILL MAKE THE DISK
UNUSABLE, AND WILL CAUSE THE DISK TO BE SPUN DOWN WHEN
BROUGHT ONLINE.

  RUNTIME 1:33:45
Format completed
     2 Revectored LBNS
     2 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     0 Bad RBNS
     0 Bad blocks in the RCT area due to data errors
     0 Bad blocks in the DBN area due to data errors
     0 Bad blocks in the XBN area due to data errors
     2 Blocks retried on the check pass
FCT used successfully

CZUDE EOP    1
    0 CUMULATIVE ERRORS
DR>
```

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

There are three levels of error messages that may be issued by a diagnostic: general, basic and extended. General error messages are always printed unless the "IER" flag is set (section 2.3). The general error message is of the form:

```
NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
error message
```

where:
- NAME = diagnostic name
- TYPE = error type (SYS FTL ERR, DEV FTL ERR)
- NUMBER = error number
- UNIT NUMBER = 0 - N (N is last unit in PTABLE)
- TST NUMBER = test and subtest where error occurred
- PC:XXXXXX = address of error message call

System fatal errors (SYS FTL ERR) are used to report errors that are fatal to the entire diagnostic program. The diagnostic stops and the Runtime Services prompt is printed.

Device fatal errors (DVC FTL ERR) are used to report errors that are fatal to the device (may be either a UDA50 or disk drive). Testing stops on that device for the remainder of the current test.

Basic error messages are messages that contain some additional information about the error. These are always printed unless the "IER" or "IBE" flags are set (section 2.3). These messages are printed after the associated general message.

Extended error messages contain supplementary error information such as register contents or good/bad data. These are always printed unless the "IER", "IBE" or "IXE" flags are set (section 2.3). These messages are printed after the associated general error message and any associated basic error messages.

The general and basic error messages from this diagnostic are always one line each. The basic message defines what program detected the error, the UDA50 being used and the time of the error:

```
HOST PROGRAM  UDA AT xxxxxx  RUNTIME hhh:mm:ss
```

The host program (PDP-11) detected the error. UDA AT xxxxx identifies the address of the UDA50 being tested. It may be omitted if the error is not specific to one UDA50.

Sample error message:

```
CZUDE DVC FTL ERR  00021 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx        general message
HOST PROGRAM  UDA AT 172150  RUNTIME 0:00:12                           basic message
UDA RESIDENT DIAGNOSTICS DETECTED FAILURE
  UDASA CONTAINS 104041                                                 extended message
REPLACE UDA MODULE M7485
```

The DUP program may also print error messages. They are printed exactly as presented by the DUP program and cannot be suppressed by any flags.

3.2 SPECIFIC ERROR MESSAGES

3.2.1 HOST PROGRAM ERROR MESSAGES

Following is a list of the error messages that may be printed by the diagnostic program. In the list, some of the numbers that may vary with execution or program version are shown as "xxx". These include program counters and runtime. Other numbers, such as unit number, drive number, UDA50 address and data in registers are filled with sample numbers. Additional information about the error may follow the error message.

00001

```
CZUDE SYS FTL ERR  00001 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
I DON'T LIKE THE ANSWERS YOU GAVE TO THE HARDWARE QUESTIONS
UDA HAS MORE THAN ONE VECTOR, BR LEVEL OR BURST RATE
```

When the hardware questions were answered, two units were selected with the same UNIBUS address but with a different vector, BR level or burst rate. A single UDA50 can have only one vector, BR level or burst rate. The program is aborted and returns to the Runtime Services prompt so that you can change the hardware questions.

00002

```
CZUDE SYS FTL ERR  00002 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
I DON'T LIKE THE ANSWERS YOU GAVE TO THE HARDWARE QUESTIONS
TWO UNITS SELECT THE SAME DRIVE
```

The hardware questions for two units were exactly the same. The program is aborted and returns to the Runtime Services prompt so that you can change the hardware questions.

00003 CZUDE SYS FTL ERR 00003 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
I DON'T LIKE THE ANSWERS YOU GAVE TO THE HARDWARE QUESTIONS
MORE THAN EIGHT DRIVES SELECTED ON THIS UDA

Up to four physical disk drives can be attached to a UDA50 at one time. A physical disk drive may be from one to four logical disk drives. Each logical disk drive is considered one unit to the diagnostic program. Even though more than eight logical disk drives can be attached to one UDA50, the UDA50 only supports eight. The program is aborted and returns to the Runtime Services prompt so that you can change the hardware questions.

00004 CZUDE SYS FTL ERR 00004 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM RUNTIME x:xx:xx
NOT ENOUGH ROOM IN MEMORY TO TEST THE UNITS SELECTED
PLEASE START PROGRAM OVER AND TEST FEWER UNITS AT A TIME

This program does not limit the number of units that can be tested by specifying a maximum number. What limits the number is the amount of memory used to store data on each unit. You have exceeded the number of units that are testable at one time. Start program over and select fewer units.

00008 CZUDE SYS FTL ERR 00008 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
I DON'T LIKE THE ANSWERS YOU GAVE TO THE HARDWARE QUESTIONS
TWO UDA'S USE THE SAME VECTOR

The hardware questions for two units specified different UDA50 Unibus addresses but identical vector addresses. The program is aborted and returns to the Runtime Services prompt so that you can change the hardware questions.

00009 CZUDE DVC FTL ERR 00009 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM RUNTIME x:xx:xx
ONLY ONE DISK CAN BE SELECTED IN HW QUESTIONS IN RESTORE MODE.
PLEASE START PROGRAM OVER AND SELECT ONLY ONE DISK.

If the operator chooses to run the formatter in RESTORE mode, then only one disk can be selected in the hardware questions. RESTORE mode is run in this way because a file containing the bad block information is used and that information matches only one drive.

00010 CZUDE DVC FTL ERR 00010 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM RUNTIME x:xx:xx
THIS PROGRAM CAN ONLY REFORMAT A DISK IN UNATTENDED MODE

This program needs to ask questions of the operator. It refuses to run in RECONSTRUCT and RESTORE modes because the questions obtain data that is absolutely necessary. REFORMAT mode is allowed to run because only a date is needed. The default date of 1 JAN 70 is used.

00014 CZUDE DVC FTL ERR 00014 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
UDA50 CONTROLLER IS AT A REVISION LEVEL NO LONGER SUPPORTED
BY THIS FORMATTER PROGRAM. THIS PROGRAM REQUIRES A UDA50-A
CONTROLLER (MODEL 6) WITH MICROCODE VERSION AT 3 OR GREATER.

CONTROLLER REPORTED MODEL CODE xx AND MICROCODE VERSION xx

All UDA50-0's (modules M7161-2) are not supported by this diag
nostic. The module set M7485-6 is the only one that can be tested
by this diagnostic. If the controller is a UDA50-0 (M7161 2)
it will not be tested. If the controller is a UDA50-A (M7485 6)
and it has old microcode (the microcode version is less than
3) this message will be printed but testing will go on. If
the controller consists of the M7161-2 modules, install one
with M7485-6 modules. Do not intermix the two, it will not
work!

00020 CZUDE DVC FTL ERR 00020 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
MEMORY ERROR TRYING TO READ UDA REGISTERS
CHECK UNIBUS SELECTION SWITCHES ON UDA MODULE M7485
OR UNIBUS
OR REPLACE UDA MODULE M7485

A non-existant memory error occurred when the host program
tried to access the UDAIP and UDASA registers. The UDA
is at another address (check the UNIBUS selection switches)
or module M7485 is broken or the UNIBUS is broken.

```
00021   CZUDE DVC FTL ERR 00021 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME xx:xx:xx
        UDA RESIDENT DIAGNOSTICS DETECTED FAILURE
          UDASA CONTAINS  105154
        REPLACE UDA MODULE M7486
```

The UDA Resident diagnostic detected a failure. The error is displayed in the UDASA. Here are the possible error values and their meaning:

104000	Fatal sequncer error
104040	D processor ALU error
104041	D proc ROM parity error
105102	D proc with no Board #2 or RAM parity error
105105	D proc RAM buffer error
105152	D proc SDI error
105153	D proc write mode wrap SERDES error
105154	D proc read mode SERDES, RSGEN, and ECC error
106040	U proc ALU error
106041	U proc Control Register error
106042	U proc DFAIL/ROM parity error/Board #1 test count is wrong
106047	U proc Constant ROM error with D proc running SDI test
106055	Unexpectant trap found, aborted diagnostic
106071	U proc ROM error
106072	U proc ROM parity error
106200	Step 1 data error (MSB not set)
107103	U proc RAM parity error
107107	U proc RAM buffer error
107115	Board #2 test count was wrong
112300	Step 2 error
122240	NPR error
122300	Step 3 error
142300	Step 4 error

Replace the board specified. M7485 is the Unibus interface board. M7486 is the SDI interface board.

```
00022   CZUDE DVC FTL ERR  00022 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        STEP BIT DID NOT SET IN UDASA REGISTER DURING INITIALIZATION
        STEP BIT EXPECTED 004000
          UDASA CONTAINS  000000
        REPLACE UDA MODULE M7485

                The UDA did not respond as expected during the
                initialization sequence which communicates using data
                in the UDASA register. A normal response from the UDA
                contains either a STEP bit or an ERROR bit defined as
                follows:
                        Bit 15 (100000)   Error bit
                        Bit 14 (040000)   Step 4 bit
                        Bit 13 (020000)   Step 3 bit
                        Bit 12 (010000)   Step 2 bit
                        bit 11 (004000)   Step 1 bit
                The expected step bit nor the error bit set within the
                expected time.

00023   CZUDE DVC FTL ERR  00023 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        UDA DID NOT CLEAR RING STRUCTURE IN HOST MEMORY DURING INITIALIZATION
        6 WORDS WERE TO BE CLEARED STARTING AT ADDRESS 040644
        FIRST SEVERAL WORDS NOT CLEARED (UP TO 6):
              ADDRESS     CONTENTS
               040644      000010
               040650      000010
               040652      000010
        REPLACE UDA MODULE M7485

                The UDA is to clear the ring structure (a communications area
                used by the UDA to talk to the host) in host memory before
                Step 4 of initialization.  If the UDA diagnostics did not
                clear memory and did not flag an error, then error message
                00023 is displayed. The contents of each word in memory
                is set to 177777 before the test. Failure of the UDA
                to clear each word indicates a fault in the address
                interface to the Unibus.

00024   CZUDE DVC FTL ERR  00024 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        UDASA REGISTER DID NOT GO TO ZERO AFTER STEP 3 WRITE OF INITIALIZATION
        PURGE/POLE DIAGNOSTICS WERE REQUESTED
          UDASA CONTENTS  004400

                For better testing, the host can test the PURGE and POLE
                mechanism of the UDA.  To do so the host sets bit15 of the
                step 3 data and sends the data to the UDA.  The UDA must
                go to zero and wait for the purge and pole.  If the UDA
                never went to zero, then error message 00024 is displayed.
                The UDA may have a bad M7485 module or the UNIBUS may be
                broken.
```

00025 CZUDE DVC FTL ERR 00025 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
UDA DID NOT RETURN CORRECT DATA IN UDASA REGISTER DURING INITIALIZATION
UDASA EXPECTED 004400
UDASA CONTAINS 004000
REPLACE UDA MODULE M7485

For each step of initialization, specific data is expected to be displayed in the UDASA. If the UDASA does not match the expected data, then error message 00025 is displayed. Replace UDA module M7485.

00030 CZUDE DVC FTL ERR 00030 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
UDA REPORTED FATAL ERROR IN UDASA REGISTER WHILE RUNNING DM PROGRAM
UDASA CONTAINS 100004

A message from the UDA firmware reports an unexpected failure. An error code is presented in the UDASA. Here is a list of the codes and their meanings:

004400 UDA has been inited by either a bus init or by writing into the UDAIP.
100001 - UNIBUS envelope/packet read error (parity or timeout)
100002 - UNIBUS envelope/packet write error (parity or timeout)
100003 - UDA ROM and RAM parity error
100004 - UDA RAM parity error
100005 - UDA ROM parity error
100006 - UNIBUS ring read error
100007 - UNIBUS ring write error
100010 - UNIBUS interrupt master failure
100011 - Host access timeout error
100012 - Host exceeded credit limit
100013 - UDA SDI hardware fatal error
100014 - DM XFC fatal error
100015 - Hardware timeout of instruction loop
100016 - Invalid virtual circuit identifier
100017 - Interrupt write error on UNIBUS

00031 CZUDE DVC FTL ERR 00031 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
NO INTERRUPT RECEIVED FROM DM PROGRAM FOR 3 MINUTES
ASSUME PROGRAM IS HUNG

All DM programs are required to communicate with the host program; so as to assure the host program that the DM program is not hung up or in an endless loop. If the DM program has not done so, the host program assumes the DM is hung and this message appears.

```
00032   CZUDE DVC FTL ERR  00032 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        MESSAGE BUFFER RECEIVED FROM DM PROGRAM WITH UNKNOWN REQUEST NUMBER
        MESSAGE BUFFER CONTAINS:
           000001 000002 000003 000004 000005 000006 000007
           000008 000009 000010 000011 000012 000013 000014
           000015 000016 000017 000018 000019 000020 000021
           000022 000023 000024 000025 000026 000027 000028
           000029 000030 000031 000032 000033 000034 000035

                The DM program and the host program communicate with
                each other using packets.  Each packet must have a
                request number set up by the DM program and interpreted
                by the host program.  This request number is not a
                known request number.  The problem may be the UNIBUS
                or either one of the UDA modules or a corrupted DM
                program.  Word 1 contains the DM request number, and
                word 2 typically contains the drive number.  The rest
                of the buffer contains information specific to a
                DM request. The numbers in the example show
                the order in which words are displayed.

00033   CZUDE DVC FTL ERR  00033 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
00034   HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        RESPONSE PACKET FROM UDA DOES NOT CONTAIN EXPECTED DATA
        EITHER UDA RETURNED ERROR STATUS OR PACKET WAS NOT RECEIVED CORRECTLY
           COMMAND PACKET SENT      RESPONSE PACKET RECEIVED
              000000 000020            000000 000020
              000000 000000            000000 000000
              000000 000002            000000 000202
              000000 014336            000000 014336
              000000 034674            000000 034674
              000000 000000            000000 000000
              000000 000000            000000 000000
              000000 051232            000000 051232
              000000 000000            000000 000000
              000000 000000            000000 000000
              000000 000000            000000 000000
              000000 000000            000000 000000

                 he host program inspected the response packet which was
                given by to UDA.  The response packet may have been in
                error with one of the following points:

                1)  The end code was not as expected.
                2)  The status code showed an error occurred with the
                    last command.
                3)  The command reference numbers (the first word) did
                    not match.

                If 1 or 3 occurred, there may have been a transmission
                problem between the UDA and the host program.  If 2 occurred,
                check the error code in the MSCP specification for further
                information. The packets are displayed two long words
                per line, low order word and byte to the right
                (corresponding to the MSCP long-word entity).
```

```
00036   CZUDE DVC FTL ERR  00036 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        NO INTERRUPT RECEIVED FROM UDA FOR 30 SECONDS
        WHILE LOADING DM PROGRAM

                After a DM program has been sent to the UDA, the host
                program expects an interrupt within 30 seconds.  The
                interrupt is used to assure the host program that the
                DM program is sane.  If no interrupt occurred, then
                error message 00036 is displayed and the DM program is
                assumed to be hung.

00037   CZUDE DVC FTL ERR  00037 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        UDA REPORTED FATAL ERROR IN UDASA REGISTER WHILE LOADING DM PROGRAM
          UDASA CONTAINS  100004
        REPLACE UDA MODULE M7485

                While loading the DM program to the UDA, the UDASA
                became non-zero.  When this occurs, it signifies that
                the UDA microcode has run across a fatal error.  The
                displayed value is in octal.  Check the error code with
                the list in 00030.

00100   CZUDE DVC FTL ERR  00100 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 172150  RUNTIME x:xx:xx
        DUP PROGRAM ASKED UNEXPECTED QUESTION (25)

                The DUP program sends a value that corresponds to
                a specific question or message.  If this value does not
                fit into the range of questions, then this error appears.

00101   CZUDE DVC FTL ERR  00101 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
        HOST PROGRAM  UDA AT 17215C  RUNTIME x:xx:xx
        DUP PROGRAM REJECTED ANSWER TO DATE OR SERIAL NUMBER QUESTION

                After the operator inputs the date/serial number, the
                DUP program will ask the host program for them.  If for
                some reason the date/serial number was unacceptable to
                the DUP program, this error message will appear.  Retry
                the program and if this error appears again, get out
                of the diagnostic runtime services and back to the
                XXDP> prompt and reload the program.
```

3.2.2 DUP PROGRAM ERROR MESSAGES

Error messages returned by the UDA formatter are as follows:

GET STATUS failure

This could be caused by a number of reasons. Examples: the RUN/STOP switch is out, the WRITE PROTECT switch is in, or the DIAGNOSTIC REQUEST bit is set by the drive.

SDI send error

An attempt to send an SDI command failed. The signal RECEIVER READY was not asserted.

Unsuccessful SDI command

The response from an SDI command was unsuccessful and all commands should be successful for the formatter to work. There may be a cable problem, drive receiver problem or UDA transmitter problem.

SDI receive error

This message is presented for several reasons. The drive timed out, the first word from the drive was not a start frame, there was a framing error on the SDI level 0 read (cable/receiver/transmitter problem), checksum error, or the buffer size given by the formatter wasn't large enough for the UDA. Again, there may be a cable/receiver/transmitter problem.

UNIBUS read error

This is caused by one of two problems. While trying to read an overlay into the UDA buffer memory, the formatter came across a nonexistent memory error. Or, there was a failure while downline loading the bad block information. There may be something wrong with the UNIBUS or the UDA module M7485.

Formatter initialization error

For this error to occur, the UDA must be processing the DM code improperly.

Non existant unit number

The desired disk drive wasn't attached to the UDA.

DBN/XBN format error (drive FORMAT command failed)

All attempts and retries to format a track failed. There may have been a timeout of drive signals, the drive dropped the READ/WRITE READY signal during the format operation or the drive clock timed out (which indicates cable/transmitter/receiver failures).

FCT does not have enough good copies of each block

There must at least two good copies of every block in the FCT. For this error to occur, the media is badly corrupted or the read/write logic is failing.

SEEK error

After a seek command completed successfully, the READ/WRITE READY signal was never set or the ATTENTION signal was set.

RCT does not have enough good copies of each block

There must be at least two good copies of every block in the RCT. For this error to occur, the media is badly corrupted or the read/write logic is failing.

LBN format error (drive FORMAT command failed)

All attempts and retries to format a track failed. There may have been a timeout of drive signals, the drive dropped the READ/WRITE READY signal during the format operation or the drive clock timed out (which indicates cable/transmitter/receiver failures).

FCT write error

A particular block failed to be written into every copy of the FCT. There is either terribly bad media or a write logic failure.

RCT read error

The formatter could not read at least one good copy of a particular block in the RCT area.

RCT write error

A particular block failed to be written into every copy of the RCT. There is either terribly bad media or a write logic failure.

RCT full

There were so many bad blocks on the media that the RCT area was filled and could not hold any more. There could be read/write logic failure or bad cable connection.

FCT read error

The formatter could not read at least one good copy of a particular block in the FCT area.

FCT downline load error

The formatter was led to believe that a bad block information file was larger than it really was. There may be a UNIBUS or M7485 problem.

Drive init timeout

After the drive was inited, the RECEIVER READY signal never asserted.

Illegal response to start-up question

An overflow occurred when the serial number went over 64 bits.

FCT corrupted Format Invalid

A problem was detected while using the data in the FCT. Either the data was not written properly or it has been corrupted since the last format. The format on the disk is no good and the disk will not be usable by any DEC operating system. Running the formatter again may have a slight chance of succeeding. Otherwise, replace the disk or HDA. If you do not have a spare disk or HDA you may try to format the disk in RECONSTRUCT mode. If the disk is not an RA80, order a replacement disk or HDA immediately.

DRIVE ERROR ENCOUNTERED STATUS RESPONSE:
STATUS (R TO L): 1AF1 0304 E100 8800 0080 0013 1000
LAST BLOCK ACCESSED (16 BIT OCTAL): 000000 000000

The disk drive reported an error. You may see the drive s fault light come on. The formatter will attempt to clear the error in the drive and continue. Th's error does not mean that anything is necessarily wrong unless this error is printed many times. If you see many of these errors, you may wish to stop the format and run diagnostics on the disk drive. But remember, if you stop the formatter the disk will not be usable and the diagnost'cs will report the format is bad. The drive s status is presented in hexidecimal in the same format as the diagnost'c programs. The last block accessed is a representation of the last block header written onto the disk.

MORE THAN 12.5% OF TRACK IS BAD

The formatter found more than one eighth of the blocks on a single track bad. This error does not mean that anyth'ng is necessarily wrong unless this error is printed may times. If you see many of these errors, you may wish to stop the format and run diagnostics on the disk drive. But remember, if you stop the formatter the disk will not be usable and the d'agnost'cs will report the format is bad.

An example of how tre errors are presented is below:

```
  RUNTIME  0:00:18
Non ex'stant unit number
```

4.0 PERFORMANCE AND PROGRESS REPORTS

There is no statistical report that can be printed using the Diagnostic Runtime Services PRINT command.

The DUP program issues the following messages upon normal completion:

Format completed

n Revectored LBNS

Where n is the number of LBNs revectored in the user data area.

n Primary revectored LBNS

Where n is the number of LBNs which were primary revectors.

n Secondary/teritary revectored LBNS

Where n is the number of the LBNs which were secondary or tertiary revectors.

n Bad RBNS

Where n is the number of RBNs which were bad.

n Bad blocks in the RCT area due to data errors

Where n is the number of blocks in the total RCT area which were bad.

n Bad blocks in the DBN area due to data errors

Where n is the number of blocks in the total DBN area which were bad.

n Bad blocks in the XBN area due to data errors

Where n is the number of blocks in the total XBN area which were bad.

n Blocks retried on the check pass

where n is the number of blocks which had an error on the first read attempt after formatting.

FCT used successfully or
FCT was not used

Depending on the answers to the software questions and the availability of the bad sector information (FCT), one of these messages will be printed.

An example of how the messages are presented is below.

```
  RUNTIME 1:24:57
Format completed
     5 Revectored LBNS
     5 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     0 Bad RBNS
     0 Bad blocks in the RCT area due to data errors
     0 Bad blocks in the DBN area due to data errors
     0 Bad blocks in the XBN area due to data errors
     5 Blocks retried on the check pass
FCT was not used
```

5.0 TEST SUMMARIES

There is only one test in this program - Test #1. Its only purpose is to load and run the format program in a UDA50.

```
                        .SBTTL  PROGRAM HEADER
                        
002000                          BGNMOD
                        ;**
                        ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                        ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                        ;
                        
002000                          POINTER BGNSW, BGNSFT, BGNSETUP
                        
002000                          HEADER  CZUDE,D,0,7200.,1,PRIO7
002000                  L$NAME::
002000  103                             .ASCII /C/
002001  132                             .ASCII /Z/
002002  125                             .ASCII /U/
002003  104                             .ASCII /D/
002004  105                             .ASCII /E/
002005  000                             .BYTE   0
002006  000                             .BYTE   0
002007  000                             .BYTE   0
002010                  L$REV::
002010  104                             .ASCII  /D/
002011                  L$DEPO::
002011  060                             .ASCII  /0/
002012                  L$UNIT::
002012  000001                          .WORD   T$PTHV
002014                  L$TIML::
002014  016040                        .WORD   7200.
002016                  L$HPCP::
002016  022620                        .WORD   L$HARD
002020                  L$SPCP::
002020  023006                        .WORD   L$SOFT
002022                  L$HPTP::
002022  002130                        .WORD   L$HW
002024                  L$SPTP::
002024  002144                        .WORD   L$SW
002026                  L$LADP::
002026  000124'                       .WORD   L$LAST
002030                  L$STA::
002030  000000                        .WORD   0
002032                  L$CO::
002032  000000                        .WORD   0
002034                  L$DTYP::
002034  000001                        .WORD   1
002036                  L$APT::
002036  000000                        .WORD   0
002040                  L$DTP::
002040  002124                        .WORD   L$DISPATCH
002042                  L$PRIO::
002042  000340                        .WORD   PRIO7
002044                  L$ENVI::
002044  000000                        .WORD   0
002046                  L$EXP1::
002046  000000                        .WORD   0
002050                  L$MREV::
002050     003                          .BYTE   C$REVISION
002051     003                          .BYTE   C$EDIT
```

PROGRAM HEADER

```
002052
002052  000000
002054  000000
002056
002056  000000
002060
002060  003456
002062
002062  000000
002064
002064  000000
002066
002066  000000
002070
002070  000000
002072
002072  000000
002074
002074  000000
002076
002076  003500
002100
002100  104035
002102
002102  000000
002104
002104  021100
002106
002106  022116
002110
002110  022114
002112
002112  021072
002114
002114  000000
002116
002116  000000
002120
002120  000000
```

```
L$EF::
        .WORD   0
        .WORD   0
L$SPC::
        .WORD   0
L$DEVP::
        .WORD   L$DVTYP
L$REPP::
        .WORD   0
L$EXP4::
        .WORD   0
L$EXP5::
        .WORD   0
L$AUT::
        .WORD   0
L$DUT::
        .WORD   0
L$LUN::
        .WORD   0
L$DESP::
        .WORD   L$DESC
L$LOAD::
        EMT     E$LOAD
L$ETP::
        .WORD   0
L$ICP::
        .WORD   L$INIT
L$CCP::
        .WORD   L$CLEAN
L$ACP::
        .WORD   L$AUTO
L$PRT::
        .WORD   L$PROT
L$TEST::
        .WORD   0
L$DLY::
        .WORD   0
L$HIME::
        .WORD   0
```

```
                                .SBTTL  DISPATCH TABLE
                                ;**
                                ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                                ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                                ;*

                                DISPATCH 1
002122                                                                          .WORD   1
002122  000001                                                          L$DISPATCH::
002124                                                                          .WORD   *1
002124  022130
```

```
                                .SBTTL  DEFAULT HARDWARE P TABLE
                                ;+
                                ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                                ; THE TEST DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                                ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P TABLES,
                                ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P TABLES.
                                ;-

002126                                  BGNHW   DFPTBL
002126  000005                                                                 .WORD   L10000-L$HW/2
002130                                                                  L$HW::
002130                                                                  DFPTBL::

002130  172150                          .WORD   172150          ; UNIBUS ADDRESS
002132  000154                          .WORD   154             ; VECTOR ADDRESS
002134  000005                          .WORD   5.              ; BR LEVEL
002136  000077                          .WORD   63.             ; UNIBUS BURST RATE
002140  000000                          .WORD   0.              ; LOGICAL DRIVE NUMBER
002142                                  ENDHW
002142                                                          L10000:
```

```
                        .SBTTL  SOFTWARE P TABLE

                        ;++
                        ; THE SOFTWARE TABLE CONTAINS VARIOUS DATA USED BY THE
                        ; PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE
                        ; SET UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR
                        ; AT RUN TIME.
                        ;

002142                          BGNSW   SFPTBL
002142  000001                                                  .WORD   L10001-L$SW/2
002144                                                          L$SW::
002144                                                          SFPTBL::

                                                        ;OFFSET         USE
002144  000007                  .WORD   7              ; 0.     YES/NO ANSWERS
002146                          ENDSW
002146                                                          L10001:

002146                          ENDMOD
```

```
                                .SBTTL  GLOBAL EQUATES SECTION

002146                                  BGNMOD

                                ;++
                                ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                ; ARE USED IN MORE THAN ONE TEST.
                                ;

002146                                  EQUALS
                                ;
                                ; BIT DIFINITIONS
                                ;
100000                          BIT15== 100000
040000                          BIT14== 40000
020000                          BIT13== 20000
010000                          BIT12== 10000
004000                          BIT11== 4000
002000                          BIT10== 2000
001000                          BIT09== 1000
000400                          BIT08== 400
000200                          BIT07== 200
000100                          BIT06== 100
000040                          BIT05== 40
000020                          BIT04== 20
000010                          BIT03== 10
000004                          BIT02== 4
000002                          BIT01== 2
000001                          BIT00== 1
                                ;
001000                          BIT9==  BIT09
000400                          BIT8==  BIT08
000200                          BIT7==  BIT07
000100                          BIT6==  BIT06
000040                          BIT5==  BIT05
000020                          BIT4==  BIT04
000010                          BIT3==  BIT03
000004                          BIT2==  BIT02
000002                          BIT1==  BIT01
000001                          BIT0==  BIT00
                                ;
                                ; EVENT FLAG DEFINITIONS
                                ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                ;
000040                          EF.START==      32.                     ; START COMMAND WAS ISSUED
000037                          EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
000036                          EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
000035                          EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
000034                          EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                ;
                                ;
                                ; PRIORITY LEVEL DEFINITIONS
                                ;
000340                          PRI07== 340
000300                          PRI06== 300
000240                          PRI05== 240
000200                          PRI04== 200
```

GLOBAL EQUATES SECTION

```
                000140                  PRIO3== 140
                000100                  PRIO2== 100
                000040                  PRIO1== 40
                000000                  PRIO0== 0
                                        ;
                                        ;OPERATOR FLAG BITS
                                        ;
                000004                  EVL==       4
                000010                  LOT==      10
                000020                  ADR==      20
                000040                  IDU==      40
                000100                  ISR==     100
                000200                  JAM==     200
                000400                  BOE==     400
                001000                  PNT==    1000
                002000                  PRI==    2000
                004000                  IXE==    4000
                010000                  IBE==   10000
                020000                  IER==   20000
                040000                  LOE==   40000
                100000                  MOE==  100000
    11
    12          000015                  CR=     15                      ;VALUE TO PASS TO PRINT MACRO TO END LINE
```

```
                        ;MACRO DEFINITIONS FOR GLOBAL EQUATES
                        ;
                        ;THESE MACROS ARE USED TO DEFINE INDEXES INTO A TABLE
                        ;
                        ;CALLING SEQUENCE MUST BE
                        ;
                        ;       TABLE
                        ;       ITEM    NAME    BYTES
                        ;       ITEM    NAME    BYTES
                        ;       ITEM    NAME    BYTES
                        ;       END     SIZE
                        ;
                        ;TABLE DEFINES THAT A TABLE IS ABOUT TO BE DEFINED AND END TERMINATES THE DEFINITION.
                        ;ANY NUMBER OF ITEM LINES CAN APPEAR. NAME IS THE NAME OF THE SYMBOL BEING EQUATED TO
                        ;THE INDEX. THE INDEX ALWAYS STARTS AT ZERO. BYTES SPECIFIES THE SIZE OF THE VALUE TO BE
                        ;STORED AT THAT INDEX IN BYTES. THE SIZE ARGUMENT TO THE END STATEMENT IS OPTIONAL, IT
                        ;BE EQUATED TO THE SIZE OF THE TABLE IN BYTES. THE SYMBOL TINDEX IS USED TO KEEP TRACK
                        ;OF THE INDEX VALUE AND WILL BE EQUAL TO THE SIZE OF THE TABLE AFTER THE END STATEMENT.
                        
                        .MACRO TABLE
                                TINDEX=0
                        .ENDM
                        
                        .MACRO ITEM NAME BYTES
                                NAME=TINDEX
                                TINDEX=TINDEX+BYTES
                        .ENDM
                        
                        .MACRO END SIZE
                                .IF NB SIZE
                                SIZE=TINDEX
                                .ENDC
                        .ENDM
```

```
                        ;UDA BIT DEFINITIONS
                        ;
                        ;UDASA REGISTER UNIVERSAL READ BITS
                        ;
004000                  SA.S1= 004000                   ;STEP 1 STATUS BIT
010000                  SA.S2= 010000                   ;STEP 2 STATUS BIT
020000                  SA.S3= 020000                   ;STEP 3 STATUS BIT
040000                  SA.S4= 040000                   ;STEP 4 STATUS BIT
100000                  SA.ERR= 100000                  ;ERROR INDICATOR
                        ;
                        ;UDASA REGISTER ERROR STATUS BITS
                        ;
003777                  SA.ERC= 003777                  ;ERROR CODE
                        ;
                        ;UDASA REGISTER STEP ONE READ BITS
                        ;
002000                  SA.NV= 002000                   ;NON SETTABLE INTERRUPT VECTOR
001000                  SA.A2= 001000                   ;22 BIT ADDRESS BUS
000400                  SA.DI= 000400                   ;ENHANCED DIAGNOSTICS
                        ;       000377                  ;ALL BITS RESERVED
                        ;
                        ;UDASA REGISTER STEP ONE WRITE BITS
                        ;
000177                  SA.VEC= 000177                  ;INTERRUPT VECTOR (DIVIDED BY 4)
000200                  SA.INT= 000200                  ;INTERRUPT ENABLE DURING INITIALIZATION
003400                  SA.MSG= 003400                  ;MESSAGE RING LENGTH
034000                  SA.CMD= 034000                  ;COMMAND RING LENGTH
040000                  SA.WRP= 040000                  ;WRAP BIT
100000                  SA.STP= 100000                  ;STEP - MUST ALWAYS BE WRITTEN A ONE
                        ;
000400                  SA.MS1= 000400                  ;LSB OF MESSAGE RING LENGTH
004000                  SA.CM1= 004000                  ;LSB OF COMMAND RING LENGTH
                        ;
                        ;UDASA REGISTER STEP TWO READ BITS
                        ;
000007                  SA.MSE= 000007                  ;MESSAGE RING LENGTH ECHO
000070                  SA.CME= 000070                  ;COMMAND RING LENGTH ECHO
                        ;       000100                  ;RESERVED
000200                  SA.STE= 000200                  ;STEP ECHO
003400                  SA.CTP= 003400                  ;CONTROLLER TYPE
                        ;
                        ;UDASA REGISTER STEP TWO WRITE BITS
                        ;
000001                  SA.PRG= 000001                  ;ENABLE VAX UNIBUS ADAPTER PURGE INTERRUPT
                        ;       177776                  ;LOW ORDER MESSAGE RING BYTE ADDRESS
```

```
                                ;UDASA REGISTER STEP THREE READ BITS
                                
000177                          SA.VCE= 000177          ;INTERRUPT VECTOR ECHO
000200                          SA.INE= 000200          ;INTERRUPT ENABLE ECHO
000400                          SA.NVE= 000400          ;VECTOR NOT PROGRAMMABLE
                                ;       003000          ;RESERVED
                                
                                ;UDASA REGISTER STEP THREE WRITE BITS
                                
                                ;       077777          ;HIGH ORDER MESSAGE RING BYTE ADDRESS
100000                          SA.TST= 100000          ;PURGE POLL TEST ENABLE
                                
                                ;UDASA REGISTER STEP FOUR READ BITS
                                
000017                          SA.MCV= 000017          ;UDA MICROCODE VERSION
000360                          SA.CNT= 000360          ;CONTROLLER MODEL
                                ;       003400          ;RESERVED
                                
                                ;UDASA REGISTER STEP FOUR WRITE BITS
                                
000001                          SA.GO=  000001          ;GO BIT TO START UDA FIRMWARE
000002                          SA.LFC= 000002          ;LAST FAILURE CODE REQUEST
000374                          SA.BST= 000374          ;BURST LEVEL
```

```
                        ;COMMAND/MESSAGE DESCRIPTOR BIT DEFINITIONS
                        ;
100000                  RG.OWN= 100000                  ;SET WHEN UDA OWNS RING
040000                  RG.FLG= 040000                  ;FLAG BIT
                        ;
                        ;OFFSETS INTO HOST COMMUNICATIONS AREA WITH ONE DESCRIPTOR TO EACH RING
                        ;AND TWO PACKET AND BUFFER AREAS.
                        ;
000004                  HC.ISZ=  4.                     ;SIZE OF INTERRUPT INDICATOR WORDS
000004                  HC.RSZ=  4.                     ;SIZE OF RING IN BYTES
000004                  HC.ESZ=  4.                     ;SIZE OF ENVELOPE WORDS BEFORE PACKET
000060                  HC.PSZ= 48.                     ;SIZE OF COMMAND AND MESSAGE PACKETS
000244                  HC.BSZ= 164.                    ;SIZE OF BUFFER
                        ;
000000                  HC.INT=  0.                     ;INTERRUPT INDICATOR WORDS START
000004                  HC.MSG= HC.INT+HC.ISZ           ;MESSAGE RING START
000006                  HC.MCT= HC.MSG+2.               ;MESSAGE RING CONTROL WORD
000010                  HC.CMD= HC.MSG+HC.RSZ           ;COMMAND RING START
000012                  HC.CCT= HC.CMD+2.               ;COMMAND RING CONTROL WORDS
000014                  HC.MEV= HC.CMD+HC.RSZ           ;MESSAGE ENVELOPE START
000020                  HC.MPK= HC.MEV+HC.ESZ           ;MESSAGE PACKET START
000100                  HC.CEV= HC.MPK+HC.PSZ           ;COMMAND ENVELOPE START
000104                  HC.CPK= HC.CEV+HC.ESZ           ;COMMAND PACKET START
000164                  HC.BF1= HC.CPK+HC.PSZ           ;FIRST BUFFER
000430                  HC.BF2= HC.BF1+HC.BSZ           ;SECOND BUFFER
                        ;
000674                  HC.SIZ= HC.BF2+HC.BSZ           ;TOTAL SIZE OF HOST COMM AREA
                        ;
                        ;VIRTUAL CIRCUIT IDENTIFIERS
                        ;
000000                  MSCP=    0                      ;MSCP CIRCUIT
000001                  LOG=     1                      ;LOG CIRCUIT
177777                  DIAG=    1                      ;DIAGNOSTIC CIRCUIT
001000                  DUP=    1000                    ;DIAGNOSTIC AND UTILITIES PROTOCOL
```

```
                 ;            .------------------------------------------.
                 ;  HC.INT    !     INTERRUPT INDICATORS                 !   4 BYTES
                 ;            !                                          !
                 ;            .------------------------------------------.
                 ;  HC.MSG    !     MESSAGE RING                         !   4 BYTES
                 ;  HC.MCT    !                                          !
                 ;            .------------------------------------------.
                 ;  HC.CMD    !     COMMAND RING                         !   4 BYTES
                 ;  HC.CCT    !                                          !
                 ;            .------------------------------------------.
                 ;  HC.MEV    !     MESSAGE ENVELOPE                     !  52 BYTES
                 ;  HC.MPK    !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            .------------------------------------------.
                 ;  HC.CEV    !     COMMAND ENVELOPE                     !  52 BYTES
                 ;  HC.CPK    !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            .------------------------------------------.
                 ;  HC.BF1    !     BUFFER # 1 (RESPONSE TO DM PROGRAM)  !  82 BYTES
                 ;            !                                          !
                 ;            !                                          !
                 ;            .------------------------------------------.
                 ;  HC.BF2    !     BUFFER # 2 (REQUEST FROM DM PROGRAM) !  82 BYTES
                 ;            !                                          !
                 ;            !                                          !
                 ;            !                                          !
                 ;            .------------------------------------------.
```

GLOBAL EQUATES SECTION

```
                                ;COMMAND PACKET OPCODES

000001                          OP.ABO=  1                      ;ABORT COMMAND
000020                          OP.ACC= 20                      ;ACCESS COMMAND
000010                          OP.AVL= 10                      ;AVAILABLE COMMAND
000021                          OP.CCD= 21                      ;COMPARE CONTROLLER DATA COMMAND
000040                          OP.CMP= 40                      ;COMPARE HOST DATA COMMAND
000022                          OP.ERS= 22                      ;ERASE COMMAND
000023                          OP.FLU= 23                      ;FLUSH COMMAND
000002                          OP.GCS=  2                      ;GET COMMAND STATUS COMMAND
000003                          OP.GUS=  3                      ;GET UNIT STATUS COMMAND
000011                          OP.ONL= 11                      ;ONLINE COMMAND
000041                          OP.RD=  41                      ;READ COMMAND
000024                          OP.RPL= 24                      ;REPLACE COMMAND
000004                          OP.SCC=  4                      ;SET CONTROLLER CHARACTERISTICS COMMAND
000012                          OP.SUC= 12                      ;SET UNIT CHARACTERISTICS COMMAND
000042                          OP.WR=  42                      ;WRITE COMMAND
000030                          OP.MRD= 30                      ;MAINTENANCE READ COMMAND
000031                          OP.MWR= 31                      ;MAINTENANCE WRITE COMMAND
000200                          OP.END=200                      ;END PACKET FLAG
000007                          OP.SEX=  7                      ;SERIOUS EXCEPTION END PACKET
000100                          OP.AVA=100                      ;AVAILABLE ATTENTION MESSAGE
000101                          OP.DUP=101                      ;DUPLICATE UNIT NUMBER ATTENTION MESSAGE
000102                          OP.SHC=102                      ;SHADOW COPY COMPLETE ATTENION MESSAGE
000103                          OP.RLC=103                      ;RESET COMMAND LIMIT ATTENTION MESSAGE

000001                          OP.GDS=  1                      ;DUP GET DUST STATUS
000001                          OP.GSS=  1                      ;DUP GET DUST STATUS
000002                          OP.ESP=  2                      ;DUP EXECUTE SUPPLIED PROGRAM
000003                          OP.ELP=  3                      ;DUP EXECUTE LOCAL PROGRAM
000004                          OP.SSD=  4                      ;DUP SEND STUD DATA
000005                          OP.RSD=  5                      ;DUP RECEIVE STUD DATA

                                ;NOTE: END PACKET OPCODES (ALSO CALLED ENDCODES) ARE FORMED BY ADDING THE END
                                ;PACKET FLAG TO THE COMMAND OPCODE. FOR EXAMPLE, A READ COMMAND'S END PACKET
                                ;CONTAINS THE VALUE OP.RD+OP.END IN ITS OPCODE FIELD. THE INVALID COMMAND END
                                ;PACKET CONTAINS JUST THE END PACKET FLAG (I.E., OP.END) IN ITS OPCODE FIELD.
                                ;THE SERIOUS EXCEPTION END PACKET CONTAINS THE SUM OF THE END PACKET FLAG
                                ;PLUS THE SERIOUS EXCEPTION OPCODE SHOWN ABOVE (I.E., OP.SEX+OP.END) IN ITS
                                ;OPCODE FIELD.
                                ;
                                ;COMMAND OPCODE BITS 3 THROUGH 5 INDICATE THE COMMAND CLASS, WHICH IS ENCODED
                                ;AS FOLLOWS:
                                ;    000  IMMEDIATE COMMANDS
                                ;    001  SEQUENTIAL COMMANDS
                                ;    010  NON-SEQUENTIAL COMMANDS THAT DO NOT INCLUDE A BUFFER DESCRIPTOR
                                ;    100  NON-SEQUENTIAL COMMANDS THAT DO INCLUDE A BUFFER DESCRIPTOR
```

GLOBAL EQUATES SECTION

```
                                ;COMMAND MODIFIERS
                                ;       = 020000        ;CLEAR SERIOUS EXCEPTION
040000                          MD.CMP= 040000          ;COMPARE
100000                          MD.EXP= 100000          ;EXPRESS REQUEST
010000                          MD.ERR= 010000          ;FORCE ERROR
004000                          MD.SCH= 004000          ;SUPPRESS CACHING (HIGH SPEED)
002000                          MD.SCL= 002000          ;SUPPRESS CACHING (LOW SPEED)
000100                          MD.SEC= 000100          ;SUPPRESS ERROR CORRECTION
000400                          MD.SER= 000400          ;SUPPRESS ERROR RECOVERY
000200                          MD.SSH= 000200          ;SUPPRESS SHADOWING
000100                          MD.WBN= 000100          ;WRITE-BACK (NON-VOLATILE)
000400                          MD.WBV= 000400          ;WRITE BACK (VOLATILE)
000020                          MD.SEQ= 000020          ;WRITE SHADOW SET ONE UNIT AT A TIME
000001                          MD.SPD= 000001          ;SPIN-DOWN
000001                          MD.FEU= 000001          ;FLUSH ENTIRE UNIT
000002                          MD.VOL= 000002          ;VOLATILE ONLY
000001                          MD.NXU= 000001          ;NEXT UNIT
000001                          MD.RIP= 000001          ;ALLOW SELF DESTRUCTION
000002                          MD.IMF= 000002          ;IGNORE MEDIA FORMAT ERROR
000004                          MD.SWP= 000004          ;SET WRITE PROTECT
000010                          MD.CWB= 000010          ;CLEAR WRITE-BACK DATA LOST
000001                          MD.PRI= 000001          ;PRIMARY REPLACEMENT BLOCK

                                ;END PACKET FLAGS

000200                          EF.BBR= 000200          ;BAD BLOCK REPORTED
000100                          EF.BBU= 000100          ;BAD BLOCK UNREPORTED
000040                          EF.LOG= 000040          ;ERROR LOG GENERATED
000020                          EF.SEX= 000020          ;SERIOUS EXCEPTION

                                ;CONTROLLER FLAGS

000200                          CF.ATN= 000200          ;ENABLE ATTENTION MESSAGES
000100                          CF.MSC= 000100          ;ENABLE MISCELLANEOUS ERROR LOG MESSAGES
000040                          CF.OTH= 000040          ;ENABLE OTHER HOST'S ERROR LOG MESSAGES
000020                          CF.THS= 000020          ;ENABLE THIS HOST'S ERROR LOG MESSAGES
000002                          CF.SHD= 000002          ;SHADOWING
000001                          CF.576= 000001          ;576 BYTE SECTORS
```

```
                                ;UNIT FLAGS
                                ;
000001                          UF.CMR= 000001                  ;COMPARE READS
000002                          UF.CMW= 000002                  ;COMPARE WRITES
100000                          UF.RPL= 100000                  ;HOST INITIATED BAD BLOCK REPLACEMENT
040000                          UF.INA= 040000                  ;INACTIVE SHADOW SET UNIT
004000                          UF.SCH= 004000                  ;SUPPRESS CACHING (HIGH SPEED)
002000                          UF.SCL= 002000                  ;SUPPRESS CACHING (LOW SPEED)
000100                          UF.WBN= 000100                  ;WRITE BACK (NON-VOLATILE)
020000                          UF.WPH= 020000                  ;WRITE PROTECT (HARDWARE)
001000                          UF.WPS= 001000                  ;WRITE PROTECT (SOFTWARE OR VOLUME
000004                          UF.576= 000004                  ;576 BYTE SECTORS
                                ;
                                ;COMMAND PACKET OFFSETS
                                ;
                                ;
                                GENERIC COMMAND PACKET OFFSETS:
000000                          P.CRF=   0.                     ;COMMAND REFERENCE NUMBER
000004                          P.UNIT=  4.                     ;UNIT NUMBER
000010                          P.OPCD=  8.                     ;OPCODE
000012                          P.MOD=  10.                     ;MODIFIERS
000014                          P.BCNT= 12.                     ;BYTE COUNT
000020                          P.BUFF= 16.                     ;BUFFER DESCRIPTOR
000020                          P.UADR= 16.                     ;UNIBUS ADDRESS OF BUFFER DESCRIPTOR
000034                          P.LBN=  28.                     ;LOGICAL BLOCK NUMBER
                                ;
                                ABORT AND GET COMMAND STATUS COMMAND PACKET OFFSETS:
000014                          P.OTRF= 12.                     ;OUTSTANDING REFERENCE NUMBER
                                ;
                                ONLINE AND SET UNIT CHARACTERISTICS COMMAND PACKET OFFSETS:
000016                          P.UNFL= 14.                     ;UNIT FLAGS
000020                          P.HSTI= 16.                     ;HOST IDENTIFIER / RESERVED
000034                          P.ELGF= 28.                     ;ERROR LOG FLAGS
000040                          P.SHUN= 32.                     ;SHADOW UNIT
000042                          P.CPSP= 34.                     ;COPY SPEED
                                ;
                                REPLACE COMMAND PACKET OFFSETS:
000014                          P.RBN=  12.                     ;REPLACEMENT BLOCK NUMBER
                                ;
                                SET CONTROLLER CHARACTERISTICS COMMAND PACKET OFFSETS:
000014                          P.VRSN= 12.                     ;MSCP VERSION
000016                          P.CNTF= 14.                     ;CONTROLLER FLAGS
000020                          P.HTMO= 16.                     ;HOST TIMEOUT
000022                          P.USEF= 18.                     ;USE FRACTION
000024                          P.TIME= 20.                     ;QUAD-WORD TIME AND DATE
                                ;
                                MAINTENANCE READ AND MAINTENANCE WRITE COMMAND PACKET OFFSETS:
000034                          P.RGID= 28.                     ;REGION ID
000040                          P.RGOF= 32.                     ;REGION OFFSET
                                ;
                                EXECUTE SUPPLIED PROGRAM COMMAND PACKET OFFSETS:
000024                          P.DMDT= 20.                     ;DMDT TERMINAL ADDRESS (MAINT WRITE ONLY)
000034                          P.OVRL= 28.                     ;BUFFER DESCRIPTOR FOR OPERLAYS
```

```
                        ;END PACKET OFFSETS
                        ;
                        ;               GENERIC END PACKET OFFSETS:
000000          P.CRF=   0.             ;COMMAND REFERENCE NUMBER
000004          P.UNIT=  4.             ;UNIT NUMBER
000010          P.OPCD=  8.             ;OPCODE (ALSO CALLED ENDCODE)
000011          P.FLGS=  9.             ;END PACKET FLAGS
000012          P.STS=  10.             ;STATUS
000014          P.BCNT= 12.             ;BYTE COUNT
000034          P.FBBK= 28.             ;FIRST BAD BLOCK
                        ;
                        ;               GET COMMAND STATUS END PACKET OFFSETS:
000014          P.OTRF= 12.             ;OUTSTANDING REFERENCE NUMBER
000020          P.CMST= 16.             ;COMMAND STATUS
                        ;
                        ;               GET UNIT STATUS END PACKET OFFSETS:
000014          P.MLUN= 12.             ;MULTI UNIT CODE
000016          P.UNFL= 14.             ;UNIT FLAGS
000020          P.MSTI= 16.             ;HOST IDENTIFIER
000024          P.UNTI= 20.             ;UNIT IDENTIFIER
000034          P.MEDI= 28.             ;MEDIA TYPE IDENTIFIER
000040          P.SHUN= 32.             ;SHADOW UNIT
000042          P.SHST= 34.             ;SHADOW STATUS
000044          P.TRCK= 36.             ;TRACK SIZE
000046          P.GRP=  38.             ;GROUP SIZE
000050          P.CYL=  40.             ;CYLINDER SIZE
000054          P.RCTS= 44.             ;RCT TABLE SIZE
000056          P.RBNS= 46.             ;RBNS / TRACK
000057          P.RCTC= 47.             ;RCT COPIES
                        ;
                        ;               ONLINE AND SET UNIT CHARACTERISTICS END PACKET AND AVAILABLE
                        ;               ATTENTION MESSAGE OFFSETS:
000014          P.MLUN= 12.             ;MULTI-UNIT CODE
000016          P.UNFL= 14.             ;UNIT FLAGS
000020          P.MSTI= 16.             ;HOST IDENTIFIER
000024          P.UNTI= 20.             ;UNIT IDENTIFIER
000034          P.MEDI= 28.             ;MEDIA TYPE IDENTIFIER
000040          P.SHUN= 32.             ;SHADOW UNIT
000042          P.SHST= 34.             ;SHADOW STATUS
000044          P.UNCL= 36.             ;UNIT COMMAND LIMIT
000050          P.UNSZ= 40.             ;UNIT SIZE
000054          P.VSER= 44.             ;VOLUME SERIAL NUMBER
                        ;
                        ;               SET CONTROLLER CHARACTERISTICS END PACKET OFFSETS:
000014          P.VRSN= 12.             ;MSCP VERSION
000016          P.CNTF= 14.             ;CONTROLLER FLAGS
000020          P.CTMO= 16.             ;CONTROLLER TIMEOUT
000022          P.CNCL= 18.             ;CONTROLLER COMMAND LIMIT
000024          P.CNTI= 20.             ;CONTROLLER ID
                        ;
                        ;               GET DUST STATUS END PACKET OFFSETS:
000014          P.DEXT= 12.             ;DUST PROGRAM EXTENSION
000017          P.DFLG= 15.             ;STATUS FLAGS
000020          P.DPI=  16.             ;PROGRESS INDICATOR
000024          P.DTO=  20.             ;TIMEOUT VALUE
```

GLOBAL EQUATES SECTION

```
                                ;STATUS AND EVENT CODE DEFINITIONS
                                ;
000037                          ST.MSK= 37              ;STATUS / EVENT CODE MASK
000040                          ST.SUB= 40              ;SUB-CODE MULTIPLIER
000000                          ST.SUC=  0              ;SUCCESS
000001                          ST.CMD=  1              ;INVALID COMMAND
000002                          ST.ABO=  2              ;COMMAND ABORTED
000003                          ST.OFL=  3              ;UNIT-OFFLINE
000004                          ST.AVL=  4              ;UNIT-AVAILABLE
000005                          ST.MFE=  5              ;MEDIA FORMAT ERROR
000006                          ST.WPR=  6              ;WRITE PROTECTED
000007                          ST.CMP=  7              ;COMPARE ERROR
000010                          ST.DAT= 10              ;DATA ERROR
000011                          ST.HST= 11              ;HOST BUFFER ACCESS ERROR
000012                          ST.CNT= 12              ;CONTROLLER ERROR
000013                          ST.DRV= 13              ;DRIVE ERROR
000037                          ST.DIA= 37              ;MESSAGE FROM AN INTERNAL DIAGNOSTIC

                                ;GET DUST STATUS FLAGS

000010                          DF.ACT= 010             ;SET IF THIS DUST CURRENTLY ACTIVE
000004                          DF.NES= 004             ;SET IF THIS DUST WILL NOT ACCEPT THE EXECUTE
                                                        ;SUPPLIED PROGRAM COMMAND
000002                          DF.LCL= 002             ;SET IF THIS DUST HAS A LOCAL LOAD MEDIA FOR LOADING
                                                        ;DIAGNOSTICS AND OTHER UTILITIES
000001                          DF.SA=  001             ;SET IF ANY PROGRAM EXECUTION UNDER THIS DUST
                                                        ;DISABLES THE OPERATION OF ALL OTHER SERVERS IN THE
                                                        ;SAME SYSTEM AS THE DUST

                                ;DUP MESSAGE TYPES

010000                          DU.QUE = 10000          ;QUESTION
020000                          DU.DFL = 20000          ;DEFAULT QUESTION
030000                          DU.INF = 30000          ;INFORMATION
040000                          DU.TER = 40000          ;TERMINATOR
050000                          DU.FTL = 50000          ;FATAL ERROR
060000                          DU.SPC = 60000          ;SPECIAL

170000                          DU.TYP= 170000          ;MESSAGE TYPE FIELD

                                ;DM PROGRAM HEADER DEFINITIONS

000000                          DMTRLN= 0               ;OFFSET TO SIZE OF PROGRAM NEEDING DOWNLINE LOAD
000004                          DMOVRL= 4               ;OFFSET TO SIZE OF OVERLAY
000021                          DMTMO=  21              ;TIMEOUT VALUE IN SECONDS (ONE BYTE)
000040                          DMMAIN= 40              ;OFFSET TO FIRST WORD OF MAIN PROGRAM
001000                          DMFRST= 1000            ;ADDRESS IN DM FILE CONTAINING FIRST BYTE OF HEADER
```

```
                                ;CONTROLLER TABLE DEFINITIONS
                                ;
                                ;ONE TABLE WILL BE SET UP BY INITIALIZE SECTION FOR EACH UDA SELECTED
                                ;FOR TESTING.  TABLES ARE CONTIGUOUS.  THE END OF THE TABLES IS
                                ;MARKED BY A WORD OF ZEROS.
                                ;
                                ;THE FIRST TABLE IS POINTED TO BY THE CONTENTS OF CTABS.
                                ;THE NUMBER OF TABLES IS CONTAINED IN CTRLRS.

002146                          TABLE                                   ;START A TABLE DEFINITION

002146                          ITEM C.UADR     2                       ;UNIBUS ADDRESS OF UDAIP REGISTER
002146                          ITEM C.UNIT     2
000077                                  CT.UNT= 000077                  ; LOGICAL UNIT NUMBER (FIRST)
100000                                  CT.AVL= BIT15                   ; SET WHEN NOT AVAILABLE FOR TESTING
002146                          ITEM C.VEC      2
000777                                  CT.VEC= 000777                  ; VECTOR ADDRESS
007000                                  CT.BRL= 007000                  ; BR LEVEL
002146                          ITEM C.BST      2                       ; BURST LEVEL
002146                          ITEM C.JSR      2                       ;INTERRUPT SERVICE ROUTINE FOR CONTROLLER
002146                          ITEM C.JAD      2                       ;  THESE TWO WORDS LOADED WITH (JSR R0,UDASRV)
002146                          ITEM C.FLG      2                       ;FLAGS
000002                                  CT.RN=  BIT1                    ;DM PROGRAM RUNNING
000004                                  CT.CMD= BIT2                    ;COMMAND ISSUED, WAITING FOR RESPONSE
000010                                  CT.MSG= BIT3                    ;MESSAGE RESPONSE RECEIVED
                                                                        ;WHENEVER THIS BIT IS SET, CT.CMD IS CLEARED
000020                                  CT.REQ= BIT4                    ;BUFFER HAS BEEN GIVEN TO UDA FOR REQUEST
                                                                        ;SET WHENEVER READ STUD DATA COMMAND
                                                                        ;GIVEN TO UDA
000040                                  CT.STA= BIT5                    ;GET DUST STATUS COMMAND HAS BEEN SENT
000100                                  CT.TM1= BIT6                    ;ONE TIMEOUT PERIOD HAS EXPIRED BETWEEN SEND OR
                                                                        ;RECEIVE DATA RESPONSE
000200                                  CT.TM2= BIT7                    ;SECOND TIMEOUT HAS EXPIRED
002146                          ITEM C.RING     2                       ;RING BUFFER ADDRESS
002146                          ITEM C.DR0      2                       ;POINTER TO DRIVE TABLES
002146                          ITEM C.DR1      2                       ;  IF ZERO, NO DRIVE TABLE E ISTS
002146                          ITEM C.DR2      2
002146                          ITEM C.DR3      2
002146                          ITEM C.DR4      2
002146                          ITEM C.DR5      2
002146                          ITEM C.DR6      2
002146                          ITEM C.DR7      2
002146                          ITEM C.TO       2                       ;TIMEOUT COUNTER
002146                          ITEM C.TOH      2                       ;  (TWO WORDS)
002146                          ITEM C.TOT      2                       ;DUP PROGRAM TIMEOUT VALUE IN SECONDS
002146                          ITEM C.PRI      4                       ;DUP PROGRAM PROGRESS INDICATOR
002146                          ITEM C.REF      2                       ;COMMAND REFERENCE NUMBER

002146                          END C.SIZE                              ;SIZE OF CONTROLLER TABLE IN BYTES
```

```
                                ;DRIVE TABLE DEFINITIONS
                                ;
                                ;ONE DRIVE TABLE WILL BE SET UP BY THE INITIALIZE SECTION FOR EACH
                                ;DRIVE SELECTED FOR TESTING.  EACH TABLE IS POINTED TO BY A
                                ;WORD IN THE CONTROLLER TABLE ON WHICH THE DRIVE EXISTS.
                                
002146                          TABLE                           ;START A TABLE DEFINITION
                                
002146                          ITEM D.DRV      2               ;DRIVE NUMBER
002146                          ITEM D.UNIT     2
000077                                  DT.UNT= 000077          ; LOGICAL UNIT NUMBER OF DRIVE
100000                                  DT.AVL= BIT15           ; SET WHEN NOT AVAILABLE FOR TESTING
002146                          ITEM D.SERN     22.             ;DISK SERIAL NUMBER
                                
002246                          END D.SIZE                      ;SIZE OF DRIVE TABLE IN BYTES
```

```
                        ;USEFUL  INSTRUCTION DEFINITIONS
                        .MACRO   AND ARG,ADR                    ;LOGICAL AND INSTRUCTION
                                 .LIST
                                                                                BIC #^C<ARG>,ADR
                                 .NLIST
                        .ENDM
                        .MACRO   OR ARG,ADR                     ;LOGICAL OR INSTRUCTION
                                 .LIST
                                                                                BIS #ARG,ADR
                                 .NLIST
                        .ENDM
                        .MACRO   PUSH ARG                       ;PUSH INSTRUCTION
                                 .IRP X,<ARG>
                                 .LIST
                                                                                MOV X,-(SP)
                                 .NLIST
                                 .ENDM
                        .ENDM
                        .MACRO   POP ARG                        ;POP INSTRUCTION
                                 .IRP X,<ARG>
                                 .LIST
                                                                                MOV (SP)+,X
                                 .NLIST
                                 .ENDM
                        .ENDM
                        .MACRO   .BR ADR                        ;A BRANCH TO THE NEXT LOCATION
                                 .IF P2
                                     .IF NE .-ADR
                                         .ERROR ;ILLEGAL .BR TO ADR
                                     .ENDC
                                 .ENDC
                        .ENDM
                        .MACRO   ASSUME FIRST CONDITION SECOND
                                     .IF CONDITION <FIRST>-<SECOND>
                                     .IFF
                                         .ERROR ;BAD ASSUME OF <FIRST> CONDITION <SECOND>
                                     .ENDC
                        .ENDM
```

```
                        ;PRINT CHARACTER
                        ;  ARGUMENT MUST BE SOURCE STATEMENT TO MOVE CHARACTER TO PRINT (MOV ARG,R0)
                        ;       EX: "PRINT R1" WILL PRINT THE CHARACTER IN R1
                        ;  SPECIAL CASE: 'PRINT #CR" WILL PRINT END OF LINE SEQUENCE
                        ;  THE PRINTING IS DONE AT THE MODE OF THE LAST PRINT LINE CALL
                        ;       IE., PNTX, PNTB, PNTY, PNTS

                        .MACRO PRINT ARG1
                                .IF DIF <ARG1>,R0
                                        .LIST
                                                                                MOVB ARG1,R0
                                        .NLIST
                                .ENDC
                                .LIST
                                                                                CALL CPNT
                                .NLIST
                        .ENDM

                        ;PROCESSING MACRO FOR NEXT SET OF FORMATTED MESSAGE MACROS

                        .MACRO PNT... RTN,ADR,ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                                PNT.CT=0
                                .IRP AA,<ARG8,ARG7,ARG6,ARG5,ARG4,ARG3,ARG2,ARG1>
                                        .IF NB,<AA>
                                                .LIST
                                                                                MOV AA, (SP)
                                                .NLIST
                                                PNT.CT=PNT.CT+2
                                        .ENDC
                                .ENDM
                                .LIST
                                                                                JSR R1,RTN
                                                                                .WORD ADR
                                                                                .WORD PNT.CT
                                .NLIST
                        .ENDM
```

```
                        ;PRINT FORMATTED MESSAGE MACROS
                        ;  USE THESE MACROS TO PRINT A FORMATTED MESSAGE
                        ;  FIRST ARGUMENT MUST BE ADDRESS OF FIRST CHARACTER OF MESSAGE STRING
                        ;       TO BE PUT INTO WORD (.WORD ARG)
                        ;  UP TO 8 SOURCE STATEMENTS MAY FOLLOW TO SPECIFY PARAMETERS TO BE
                        ;  USED BY THE FORMAT
                        
                        .MACRO PNTF ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                                PNT... LPNTF ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                        .ENDM
                        .MACRO PNTB ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                                PNT... LPNTB ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                        .ENDM
                        .MACRO PNTX ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                                PNT... LPNTX ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                        .ENDM
                        .MACRO PNTS ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                                PNT... LPNTS ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                        .ENDM
                        .MACRO PNT ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                                PNT... LPNT ADR ARG1,ARG2,ARG3,ARG4,ARG5,ARG6,ARG7,ARG8
                        .ENDM
```

```
                                .SBTTL  GLOBAL DATA SECTION

                                ;++
                                ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                                ; IN MORE THAN ONE TEST.
                                ;

002146                          FFREE:: .BLKW 1                         ;FIRST FREE WORD IN MEMORY
002150                          FSIZE:: .BLKW 1                         ;SIZE OF FREE MEMORY IN WORDS
002152                          FMEM:   .BLKW 1                 ;COPY OF FFREE AT END OF INIT SECTION
002154                          FMEMS:  .BLKW 1                 ;COPY OF FSIZE AT END OF INIT SECTION
002156                          CTABS:: .BLKW 1                         ;START OF CONTROLLER TABLE STORAGE
002160                          CTRLRS: .BLKW 1                         ;COUNT OF UDA CONTROLLERS IN PTABLES
002162                          TSTTAB: .BLKW 1                         ;POINTER TO FIRST CONTROLLER TABLE UNDER TEST
                                        .GLOBL UDAFM
002164  000000G                 DMPROG: .WORD UDAFM                     ;START ADDRESS OF DM PROGRAM
002166                          URUN:   .BLKW 1                         ;NUMBER OF UNITS TO RUN AT ONE TIME
002170                          URNING: .BLKW 1                         ;NUMBER OF UNITS STILL RUNNING
002172                          UCNT:   .BLKW 1                         ;COUNTER OF UNITS UNDER TEST
002174  000000                  FILOPN: .WORD 0                         ; FILE OPEN
002176                          UFREEZ: .BLKW 1                         ;FREEZE ON UNIT WHEN NOT ZERO
002200                          NXMAD:  .BLKW 1                         ;SET TO ALL ONES BY NON-EXISTANT ADDRESS
002202  000000                  FDATA:  .WORD 0
002204                          FCTBUF: .BLKB 512.              ;STORAGE FOR FCT BLOCK
003204                          FCTNUM: .BLKW 1         ;FCT BLOCK NUMBER
003206                          MODE:   .BLKW 1 ;MODE WORD, SAME BIT DEFS AS SO.BIT

                                ;CLOCK CONTROL

003210  000000                  KW.CSR: .WORD 0                         ;CSR OF CLOCK
003212                          KW.BRL: .BLKW 1                         ;BR LEVEL
003214                          KW.VEC: .BLKW 1                         ;VECTOR
003216                          KW.HZ:  .BLKW 1                         ;HERTZ (50. OR 60.)
003220                          KW.EL:  .BLKW 2                         ;ELAPSED TIME

003224  016104                  PTYPE:  .WORD PF                        ;PRINT TYPE
003226     000      000         ERRCHR: .BYTE 0,0                       ;FIRST BYTE LOADED WITH OUTPUT CHARACTER
00323C  000000                  NULL:   .WORD 0                         ;USED TO PRINT A NULL CHARACTER
003232                          FNAME:  .BLKB 10.
```

```
003244                          TEMP:   .BLKB 22.                       ;USED TO GET ANSWER FROM GMANID CALL
003272     061     055     112  DATEI:  .ASCIZ\1-JAN 70\                ;DEFAULT DATE
003303                                  .BLKB 3
003306  000000                  DATEO.  .WORD 0 ;DATE STRING IN FORMATTER FORMAT
003310                                  .BLKB 10.       ;(FIRST WORD ZERO SAYS NO DATE HERE YET)
003322     061     070     064  HIGHEST:.ASCIZ\18446744073709551615\    ;HIGHEST DISK SERIAL NUMBER
003347     104     105     103  MONTHS: .ASCII\DEC\         ;NAME OF MONTHS
003352     116     117     126          .ASCII\NOV\
003355     117     103     124          .ASCII\OCT\
003360     123     105     120          .ASCII\SEP\
003363     101     125     107          .ASCII\AUG\
003366     112     125     114          .ASCII\JUL\
003371     112     125     116          .ASCII\JUN\
003374     115     101     131          .ASCII\MAY\
003377     101     120     122          .ASCII\APR\
003402     115     101     122          .ASCII\MAR\
003405     106     105     102          .ASCII\FEB\
003410     112     101     116          .ASCII\JAN\
003413     037                  DAYS:   .BYTE 31.       ;NUMBER OF DAYS IN EACH MONTH
003414     035                          .BYTE 29.
003415     037                          .BYTE 31.
003416     036                          .BYTE 30.
003417     037                          .BYTE 31.
003420     036                          .BYTE 30.
003421     037                          .BYTE 31.
003422     037                          .BYTE 31.
003423     036                          .BYTE 30.
003424     037                          .BYTE 31.
003425     036                          .BYTE 30.
003426     037                          .BYTE 31.
003427     061     071     000  YEAR19: .ASCIZ\19\
003432     062     060     000  YEAR20: .ASCIZ\20\
                                        .EVEN
003436  000000                  IPADRS: .WORD 0
003440  000000                          .WORD 0
003442  000000                          .WORD 0
003444  000000                          .WORD 0
003446  000000                          .WORD 0
003450  000000                          .WORD 0
003452  000000                          .WORD 0
003454  000000                          .WORD 0
```

```
                                .SBTTL  GLOBAL TEXT SECTION
                                ;
                                ;++
                                ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                ; MORE THAN ONE TEST.
                                ;-
                                ;
                                ;
                                ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                ;
003456                                  DEVTYP <UDA 50 CONTROLLER>
003456                                                                  L$DVTYP::
003456  125     104     101                                             .ASCIZ  /UDA 50 CONTROLLER/
                                                                        .EVEN
                                ;
                                ; TEST DESCRIPTION
                                ;
003500                                  DESCRIPT <CZUDEDO  PDP 11 UDA DRV FMTR>
003500                                                                  L$DESC::
003500  103     132     125                                             .ASCIZ  /CZUDEDO  PDP 11 UDA
DRV FMTR/
                                                                        .EVEN
```

```
                                        ;UNFORMATTED MESSAGES
                                        ;
003536      105     116     124 DATEQ:  .ASCIZ\ENTER DATE AS DD-MMM-YY\
003566      040     106     117 FILNAQ: .ASCIZ\ FOR DISK TO BE FORMATTED\
003620      040     000         SERNQ:  .ASCIZ\ \
003622      101     122     105 UNQUES: .ASCIZ\ARE YOU SURE YOU WANT TO RUN THIS FORMATTER\
```

GLOBAL TEXT SECTION

```
                                        ; FORMAT STATEMENTS USED IN PRINT CALLS

003676  045  124  000  ERRONE: .ASCIZ\@T\
003701  045  116  000  ERRNL.  .ASCIZ\@N\
003704  042  040  040  RNTIM:  .ASCIZ/"  RUNTIME "D16":"\
003727  104  071  042  RNTIM1: .ASCIZ\D9':"\
003735  104  071  000  RNTIM2: .ASCIZ\D9\
003740  042  040  040  ERRME1: .ASCIZ\"  *  *  *  ERROR PROCESSING MESSAGE STRING  *  *  *'N
004027  116  042  125  MESSG:  .ASCIZ\N"UNIT "D6" UDA AT "O16" DRIVE 'D9S\
004073  042  116  117  NOCLOCK:.ASCIZ\"NO LINE CLOCK AVAILABLE FOR TIMING EVENTS"N\
004150  042  110  117  BASNO:  .ASCIZ\"HOST PROGRAM"\
004167  042  040  040  BASL2:  .ASCIZ\"  UDA AT "O16\
004206  042  040  040  BASL3:  .ASCIZ\"  DRIVE "D9\
004223  000              BAS:    .BYTE 0                          ;NULL TO PRINT NOTHING

004224  122  066  122  BASLN:  .ASCIZ\R6R6R6R6\                  ;USED TO PRINT BASIC LINE OF ERROR MESSAGE
004235  116  042  123  SERNUM: .ASCIZ\N"SERIAL NUMBER FOR UNIT "D6" UDA AT "O16" DRIVE "D9\
004322  042  123  124  WNSTOP: .ASCII\"STOPPING THIS FORMAT AFTER THIS POINT WILL MAKE THE DISK"N\
004415  042  125  116          .ASCII\"UNUSABLE, AND WILL CAUSE THE DISK TO BE SPUN DOWN WHEN"N\
004506  042  102  122          .ASCIZ\"BROUGHT ONLINE."NN\
004532  116  042  127  WNSTRT: .ASCII\N"WARNING:"N\
004546  042  040  040          .ASCII\"      THIS FORMATTER PROGRAM SHOULD NOT BE USED AS A DIAGNOSTIC"N\
004650  042  040  040          .ASCII\"      TOOL.  RUN THIS PROGRAM ONLY AS INSTRUCTED IN THE DISK"N\
004747  042  040  040          .ASCIZ\"      DRIVE'S SERVICE MANUAL."N\
```

```
                                        X1A:
                                        X2A:
                                        X3A:
005010  042     111     040     X8A:    .ASCIZ\"I DON'T LIKE THE ANSWERS YOU GAVE TO THE HARDWARE QUESTIONS"N
005107  122     065     122     X1:     .ASCIZ\R5R6"UDA HAS MORE THAN ONE VECTOR, BR LEVEL OR BURST RATE"N
005203  122     065     122     X2:     .ASCIZ\R5R6"TWO UNITS SELECT THE SAME DRIVE"N\
005252  122     065     122     X3:     .ASCIZ\R5R6"MORE THAN EIGHT DRIVES SELECTED ON THIS UDA"N\
005335  122     064     042     X4:     .ASCII\R4"NOT ENOUGH ROOM IN MEMORY TO TEST THE UNITS SELECTED"N\
005426  042     120     114             .ASCIZ\"PLEASE START PROGRAM OVER AND TEST FEWER UNITS AT A TIME"N\
005522  122     065     122     X8:     .ASCIZ\R5R6"TWO UDA'S USE THE SAME VECTOR"N\
005567  122     064     042     X9:     .ASCII\R4"ONLY ONE DISK CAN BE SELECTED IN HW QUESTIONS IN RESTORE MODE."N
005672  042     120     114             .ASCIZ\"PLEASE START PROGRAM OVER AND SELECT ONLY ONE DISK."N\
005761  122     064     042     X10:    .ASCIZ\R4"THIS PROGRAM CAN ONLY REFORMAT A DISK IN UNATTENDED MODE."N\
006060  122     065     042     X14:    .ASCII\R5"UDA50 CONTROLLER IS AT A REVISION LEVEL NO LONGER SUPPORTED BY"N
006163  042     124     110             .ASCII\"THIS FORMATTER PROGRAM.  THIS PROGRAM REQUIRES A UDA50-A"N\
006256  042     103     117             .ASCII\"CONTROLLER (MODEL 6) WITH MICROCODE VERSION AT 3 OR GREATER."NN\
006356  042     103     117             .ASCIZ\"CONTROLLER REPORTED MODEL CODE "D4" AND MICROCODE VERSION "D4"."N\
006461  122     065     042     X20:    .ASCII\R5"MEMORY ERROR TRYING TO READ UDA REGISTERS"N\
006537  042     103     110             .ASCII\"CHECK UNIBUS SELECTION SWITCHES ON UDA MODULE M7485"N\
006625  042     117     122             .ASCII\"OR UNIBUS"N\
006641  042     117     122             .ASCIZ\"OR "R7\
006651  122     065     042     X21:    .ASCII\R5"UDA RESIDENT DIAGNOSTICS DETECTED FAILURE"NR8\
006731  042     122     105             .ASCIZ\"REPLACE UDA MODULE M748"03N\
006766  122     065     042     X22:    .ASCII\R5"STEP BIT DID NOT SET IN UDASA REGISTER DURING INITIALIZATION"N\
007067  042     123     124             .ASCIZ\"STEP BIT EXPECTED "016NR8R7\
007124  122     065     042     X23A:   .ASCII\R5"UDA DID NOT CLEAR RING STRUCTURE IN HOST MEMORY DURING INITIALIZATION"N\
007236  104     071     042             .ASCII\D9" WORDS WERE TO BE CLEARED STARTING AT ADDRESS "016N\
007324  042     106     111             .ASCII\"FIRST SEVERAL WORDS NOT CLEARED (UP TO 6):"N\
007401  123     066     042             .ASCIZ\S6"ADDRESS"S4"CONTENTS"N\
007432  123     067     117     X23B:   .ASCIZ\S7016S5016N\
007446  122     065     042     X24:    .ASCII\R5"UDASA REGISTER DID NOT GO TO ZERO AFTER STEP 3 WRITE OF INITIALIZATION"N
007561  042     120     125             .ASCIZ\"PURGE/POLE DIAGNOSTICS WERE REQUESTED"NR8R7\
007636  122     065     042     X25:    .ASCII\R5"UDA DID NOT RETURN CORRECT DATA IN UDASA REGISTER DURING INITIALIZATION"N\
007752  042     040     040             .ASCIZ\"  UDASA EXPECTED  "016NR8R7\
010007  122     065     042     X30:    .ASCIZ\R5"UDA REPORTED FATAL ERROR IN UDASA REGISTER WHILE RUNNING DM PROGRAM"NR8
010122  122     065     042     X31:    .ASCIZ\R5"DUP PROGRAM IS HUNG"N\
010153  122     065     042     X32:    .ASCIZ\R5"MESSAGE BUFFER RECEIVED FROM DM PROGRAM WITH UNKNOWN REQUEST NUMBER"N\
010264  122     065     042     X36:    .ASCII\R5"NO INTERRUPT RECEIVED FROM UDA FOR 30 SECONDS"N\
010346  042     127     110             .ASCIZ\"WHILE LOADING DM PROGRAM"N\
010402  122     065     042     X37:    .ASCIZ\R5"UDA REPORTED FATAL ERROR IN UDASA REGISTER WHILE LOADING DM PROGRAM"NR8R7
010517  122     065     042     X100:   .ASCIZ\R5"DUP PROGRAM ASKED UNEXPECTED QUESTION ("D12")"N\
010602  122     065     042     X101:   .ASCIZ\R5"DUP PROGRAM REJECTED ANSWER TO DATE OR SERIAL NUMBER QUESTION"N\
```

GLOBAL TEXT SECTION

```
 1 010705  042  115  105  XMSG1:  .ASCIZ\"MESSAGE BUFFER CONTAINS:"N\
 2 010741  123  063  117  XMSG2:  .ASCIZ\S3016S1016S1016S1016S1016S1016S1016N\
 3 011006  122  065  042  XPKT1:  .ASCII\R5"RESPONSE PACKET FROM UDA DOES NOT CONTAIN EXPECTED DATA"N
 4 011102  042  105  111         .ASCII\"EITHER UDA RETURNED ERROR STATUS OR PACKET WAS NOT RECEIVED CORRECTLY"N
 5 011212  123  063  042         .ASCIZ\S3"COMMAND PACKET SENT S6"RESPONSE PACKET RECEIVED"N\
 6 011277  123  066  117  XPKT2:  .ASCIZ\S6016S1016S14016S1016N\
 7 011326  042  040  040  XSA:    .ASCIZ\"  UDASA CONTAINS  '016N\
 8 011357  042  122  105  XFRU:   .ASCIZ\'REPLACE UDA MODULE M7485'N\
 9
10
11 011413  045  101  111  SERNX:  .ASCIZ\%AINPUT ERROR. ANSWER WITH DECIMAL NUMBER LO= 0  HI= %T
12 011503  042  111  116  DATEX:  .ASCIZ\"INPUT ERROR.'\
13 011522  042  116  101  FILNAM: .ASCIZ\"NAME OF FILE CONTAINING BAD SECTOR INFORMATION"\
14                         .EVEN
```

```
                                .SBTTL  GLOBAL ERROR REPORT SECTION
                                ;+
                                ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
                                ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
                                ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
                                ;-
        177777                  SVCINS= -1                      ; LIST INSTRUCTIONS, SHIFTED RIGHT
        177777                  SVCTST= -1                      ; LIST TEST TAGS, SHIFTED RIGHT
        177777                  SVCSUB= -1                      ; LIST SUBTEST TAGS, SHIFTED RIGHT
        177777                  SVCGBL= -1                      ; LIST GLOBAL TAGS, SHIFTED RIGHT
        177777                  SVCTAG= -1                      ; LIST OTHER TAGS, SHIFTED RIGHT

011604                          BGNMSG ERR001
011604                                  PNTB X1,#X1A
011604  012746  005010                                                  MOV #X1A,-(SP)
011610  004137  016236                                                  JSR R1,LPNTB
011614  005107                                                          .WORD X1
011616  000002                                                          .WORD PNT.CT
011620                          ENDMSG

011622                          BGNMSG ERR002
011622                                  PNTB X2,#X2A
011622  012746  005010                                                  MOV #X2A,-(SP)
011626  004137  016236                                                  JSR R1,LPNTB
011632  005203                                                          .WORD X2
011634  000002                                                          .WORD PNT.CT
011636                          ENDMSG

011640                          BGNMSG ERR003
011640                                  PNTB X3,#X3A
011640  012746  005010                                                  MOV #X3A,-(SP)
011644  004137  016236                                                  JSR R1,LPNTB
011650  005252                                                          .WORD X3
011652  000002                                                          .WORD PNT.CT
011654                          ENDMSG

011656                          BGNMSG ERR004
011656                                  PNTB X4
011656  004137  016236                                                  JSR R1,LPNTB
011662  005335                                                          .WORD X4
011664  000000                                                          .WORD PNT.CT
011666                          ENDMSG

011670                          BGNMSG ERR008
011670                                  PNTB X8,#X8A
011670  012746  005010                                                  MOV #X8A,-(SP)
011674  004137  016236                                                  JSR R1,LPNTB
011700  005522                                                          .WORD X8
011702  000002                                                          .WORD PNT.CT
011704                          ENDMSG

011706                          BGNMSG ERR009
011706                                  PNTB X9
011706  004137  016236                                                  JSR R1,LPNTB
011712  005567                                                          .WORD X9
011714  000000                                                          .WORD PNT.CT
```

```
011716                              ENDMSG

011720                              BGNMSG ERR010
011720                                      PNTB X10
011720  004137  016236                                                  JSR R1,LPNTB
011724  005761                                                          .WORD X10
011726  000000                                                          .WORD PNT.CT
011730                              ENDMSG

011732                              BGNMSG ERR014
011732                                      PNTB X14,R3,R2
011732  010246                                                          MOV R2,-(SP)
011734  010346                                                          MOV R3,-(SP)
011736  004137  016236                                                  JSR R1,LPNTB
011742  006060                                                          .WORD X14
011744  000004                                                          .WORD PNT.CT
011746                              ENDMSG

011750                              BGNMSG ERR020
011750                                      PNTB X20
011750  004137  016236                                                  JSR R1,LPNTB
011754  006461                                                          .WORD X20
011756  000000                                                          .WORD PNT.CT
011760                              ENDMSG

011762                              BGNMSG ERR021
011762  010201                              MOV R2,R1
011764  000301                              SWAB R1
011766                                      AND 2,R1
011766  042701  177775                                                  BIC #^C<2>,R1
011772  006201                              ASR R1
011774  062701  000005                      ADD #5,R1
012000                                      PNTB X21,R2,R1
012000  010146                                                          MOV R1,-(SP)
012002  010246                                                          MOV R2,-(SP)
012004  004137  016236                                                  JSR R1,LPNTB
012010  006651                                                          .WORD X21
012012  000004                                                          .WORD PNT.CT
012014                              ENDMSG

012016                              BGNMSG ERR022
012016  042737  100000  020216              BIC #SA.ERR,UDARSD
012024                                      PNTB X22,UDARSD,R2
012024  010246                                                          MOV R2,-(SP)
012026  013746  020216                                                  MOV UDARSD,-(SP)
012032  004137  016236                                                  JSR R1,LPNTB
012036  006766                                                          .WORD X22
012040  000004                                                          .WORD PNT.CT
012042                              ENDMSG

012044                              BGNMSG ERR023
012044                                      PNTB X23A,R1,FFREE
012044  013746  002146                                                  MOV FFREE,-(SP)
012050  010146                                                          MOV R1,-(SP)
012052  004137  016236                                                  JSR R1,LPNTB
012056  007124                                                          .WORD X23A
012060  000004                                                          .WORD PNT.CT
```

```
012062  005742                   TST    (R2)
012064  005712          ERR23A:  TST    (R2)
012066  001410                   BEQ    ERR23B
012070                           PNTB   X23B,R2,(R2)
012070  011246                                                   MOV (R2),-(SP)
012072  010246                                                   MOV R2,-(SP)
012074  004137  016236                                           JSR R1,LPNTB
012100  007432                                                   .WORD X23B
012102  000004                                                   .WORD PNT.CT
012104  005304                   DEC    R4
012106  001403                   BEQ    ERR23C
012110  005722          ERR23B:  TST    (R2)+
012112  005303                   DEC    R3
012114  001363                   BNE    ERR23A
012116                  ERR23C:  PNTB   XFRU
012116  004137  016236                                           JSR R1,LPNTB
012122  011357                                                   .WORD XFRU
012124  000000                                                   .WORD PNT.CT
012126                  ENDMSG

012130                  BGNMSG ERR024
012130                           PNTB   X24,R2
012130  010246                                                   MOV R2,-(SP)
012132  004137  016236                                           JSR R1,LPNTB
012136  007446                                                   .WORD X24
012140  000002                                                   .WORD PNT.CT
012142                  ENDMSG

012144                  BGNMSG ERR025
012144                           PNTB   X25,R1,R2
012144  010246                                                   MOV R2,-(SP)
012146  010146                                                   MOV R1,-(SP)
012150  004137  016236                                           JSR R1,LPNTB
012154  007636                                                   .WORD X25
012156  000004                                                   .WORD PNT.CT
012160                  ENDMSG

012162                  BGNMSG ERR030
012162                           PNTB   X30,R1
012162  010146                                                   MOV R1,-(SP)
012164  004137  016236                                           JSR R1,LPNTB
012170  010007                                                   .WORD X30
012172  000002                                                   .WORD PNT.CT
012174                  ENDMSG

012176                  BGNMSG ERR031
012176                           PNTB   X31
012176  004137  016236                                           JSR R1,LPNTB
012202  010122                                                   .WORD X31
012204  000000                                                   .WORD PNT.CT
012206                  ENDMSG

012210                  BGNMSG ERR032
012210                           PNTB   X32
012210  004137  016236                                           JSR R1,LPNTB
012214  010153                                                   .WORD X32
012216  000000                                                   .WORD PNT.CT
```

```
 96 012220  004737  012410                 CALL MSGPKT
 97 012224                              ENDMSG
 98
 99 012226                              BGNMSG ERR033
100 012226  004737  012316                 CALL PNTPKT
101 012232                              ENDMSG
102
103 012234                              BGNMSG ERR034
104 012234  004737  012316                 CALL PNTPKT
105 012240                              ENDMSG
106
107 012242                              BGNMSG ERR036
108 012242                                 PNTB X36
    012242  004137  016236                                          JSR R1,LPNTB
    012246  010264                                                  .WORD X36
    012250  000000                                                  .WORD PNT.CT
109 012252                              ENDMSG
110
111 012254                              BGNMSG ERR037
112 012254                                 PNTB X37,R1
    012254  010146                                                  MOV R1, (SP)
    012256  004137  016236                                          JSR R1,LPNTB
    012262  010402                                                  .WORD X37
    012264  000002                                                  .WORD PNT.CT
113 012266                              ENDMSG
114
115 012270                              BGNMSG ERR100
116 012270                                 PNTB X100,(R4)
    012270  011446                                                  MOV (R4), (SP)
    012272  004137  016236                                          JSR R1,LPNTB
    012276  010517                                                  .WORD X100
    012300  000002                                                  .WORD PNT.CT
117 012302                              ENDMSG
118
119 012304                              BGNMSG ERR101
120 012304                                 PNTB X101
    012304  004137  016236                                          JSR R1,LPNTB
    012310  010602                                                  .WORD X101
    012312  000000                                                  .WORD PNT.CT
121 012314                              ENDMSG
122
123 012316                              PNTPKT: PNTB XPKT1
    012316  004137  016236                                          JSR R1,LPNTB
    012322  011006                                                  .WORD XPKT1
    012324  000000                                                  .WORD PNT.CT
124 012326  010401                           MOV R4,R1
125 012330  062701  000104                   ADD #HC.CPK,R1
126 012334  010402                           MOV R4,R2
127 012336  062702  000020                   ADD #HC.MPK,R2
128 012342  012703  000014                   MOV #12.,R3
129 012346                              PNTPKL: PNTB XPKT2,2(R1),(R1),2(R2),(R2)
    012346  011246                                                  MOV (R2), (SP)
    012350  016246  000002                                          MOV 2(R2),-(SP)
    012354  011146                                                  MOV (R1), (SP)
    012356  016146  000002                                          MOV 2(R1),-(SP)
    012362  004137  016236                                          JSR R1,LPNTB
    012366  011277                                                  .WORD XPKT2
```

```
012370  000010                                          .WORD PNT.CT
012372  062701  000004            ADD #4,R1
012376  062702  000004            ADD #4,R2
012402  005303                    DEC R3
012404  001360                    BNE PNTPKL
012406  000207                    RETURN

012410                   MSGPKT:  PNTB XMSG1
012410  004137  016236                                  JSR R1,LPNTB
012414  010705                                          .WORD XMSG1
012416  000000                                          .WORD PNT.CT
012420  016504  000016            MOV C.RING(R5),R4
012424  062704  000430            ADD #HC.BF2,R4
012430  012703  000005            MOV #5,R3
012434                   MSGPKL:  PNTB XMSG2,(R4),2(R4),4(R4),6(R4),8.(R4),10.(R4),12.(R4)
012434  016446  000014                                  MOV 12.(R4),-(SP)
012440  016446  000012                                  MOV 10.(R4),-(SP)
012444  016446  000010                                  MOV 8.(R4),-(SP)
012450  016446  000006                                  MOV 6(R4),-(SP)
012454  016446  000004                                  MOV 4(R4),-(SP)
012460  016446  000002                                  MOV 2(R4),-(SP)
012464  011446                                          MOV (R4),-(SP)
012466  004137  016236                                  JSR R1,LPNTB
012472  010741                                          .WORD XMSG2
012474  000016                                          .WORD PNT.CT
012476  062704  000016            ADD #14.,R4
012502  005303                    DEC R3
012504  001353                    BNE MSGPKL
012506  000207                    RETURN
```

```
1           000001                  SVCINS= 1                       ; LIST INSTRUCTIONS, SHIFTED RIGHT
2           000001                  SVCTST= 1                       ; LIST TEST TAGS, SHIFTED RIGHT
3           000001                  SVCSUB= 1                       ; LIST SUBTEST TAGS, SHIFTED RIGHT
4           000001                  SVCGBL= 1                       ; LIST GLOBAL TAGS, SHIFTED RIGHT
5           000001                  SVCTAG= 1                       ; LIST OTHER TAGS, SHIFTED RIGHT
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

                                ;MEMORY ALLOCATION ERROR
                                ;
                                ;THIS ROUTINE PRINTS A SYSTEM FATAL ERROR AND EXITS THE TEST

012510                          FMERR:  ERRSF 4,,ERR004
012510  104454                                                          TRAP    C$ERSF
012512  000004                                                          .WORD   4
012514  000000                                                          .WORD   0
012516  011656                                                          .WORD   ERR004
012520                                  DOCLN                   ;ABORT
012520  104444                                                          TRAP    C$DCLN
```

```
                                        ;ALOCM
                                        ;
                                        ;ALLOCATE A BLOCK OF FREE MEMORY.  REPORT ERROR IF MEMORY EXHAUSTED.
                                        ;
                                        ;INPUTS:
                                        ;       R1 - NUMBER OF WORDS TO ALLOCATE
                                        ;       FFREE   FIRST FREE WORD IN MEMORY
                                        ;       FSIZE   SIZE OF FREE MEMORY AVAILABLE IN WORDS
                                        ;OUTPUTS:
                                        ;       R1 - ADDRESS OF FIRST WORD OF ALLOCATED MEMORY
                                        ;       FFREE - NEW FIRST FREE WORD IN MEMORY
                                        ;       FSIZE   SIZE OF FREE MEMORY LEFT AFTER ALLOCATION
                                        ;SYSTEM FATAL ERROR WILL BE REPORTED IF NOT ENOUGH MEMORY AVAILABLE
                                        ;AND ENTIRE PROGRAM WILL BE STOPPED.

012522                                  ALOCM:  PUSH FFREE                      ;SAVE FFREE AT ENTRY
012522  013746  002146                                                                MOV FFREE,(SP)
012526  160137  002150                          SUB R1,FSIZE                    ;REDUCE SIZE OF FREE MEMORY
012532  002766                                  BLT FMERR                       ;REPORT ERROR IF NOT ENOUGH MEMORY
012534  060101                                  ADD R1,R1                       ;CHANGE WORDS TO BYTES
012536  060137  002146                          ADD R1,FFREE                    ;CALCULATE NEW START OF FREE MEMORY
012542                                          POP R1                          ;GET START OF ALLOCATED MEMORY
012542  012601                                                                      MOV (SP)+,R1
012544  000207                                  RETURN
```

```
                                        ;HCOMM
                                        ;
                                        ;ALLOCATES MEMORY FOR HOST COMM AREA AND PACKET BUFFERS WITH ONE
                                        ;DESCRIPTOR IN EACH RING.  TO BE CALLED WHEN INITIALIZING
                                        ;A CONTROLLER WITH SA.MSG=0 AND SA.CMD=0.
                                        ;
                                        ;INPUTS:
                                        ;       R5    ADDRESS OF CONTROLLER TABLE
                                        ;OUTPUTS:
                                        ;       CONTROLLER TABLE POINTING TO HOST COMM AREA
                                        ;       R4    ADDRESS OF HOST COMM AREA

012546  012701  000336        HCOMM:  MOV #HC.SIZ/2,R1                ;GET SIZE OF AREA TO ALLOCATE
012552  004737  012522                CALL ALOCM                      ;ALLOCATE THE MEMORY
012556  010165  000016                MOV R1,C.RING(R5)                       ;GET ADDRESS OF HOST COMM AREA
                                                                        ;PLACE IN CONTROLLER TABLE
012562  000207                        RETURN
```

```
                                ;RESET
                                ; RESET ALL UDA SOS IN THE CONTROLLER TABLES
                                ;
                                ; INPUTS:
                                ;       IPADRS   CONTAINS ALL IP ADDRESSES
                                ; OUTPUTS:
                                ;       NONE
                                ;
012564                          RESET:  PUSH <R3,R4>
012564  010346                                                                 MOV R3, (SP)
012566  010446                                                                 MOV R4, (SP)
012570  005037  002200                  CLR     NXMAD
012574                                  SETVEC #4,#NXMI,#PRI07
012574  012746  000340                                                        MOV     #PRI07, (SP)
012600  012746  017146                                                        MOV     #NXMI, (SP)
012604  012746  000004                                                        MOV     #4, (SP)
012610  012746  000003                                                        MOV     #3,-(SP)
012614  104437                                                              TRAP    C$SVEC
012616  062706  000010                                                      ADD     #10,SP
012622                                  BREAK
012622  104422                                                              TRAP    C$BRK
012624  012703  000010                  MOV     #8.,R3          ; R3 = COUNTER OF ENTRIES
012630  012704  003436                  MOV     #IPADRS,R4      ; R4 -> IP ADDRESS
012634  005714                  1$:     TST     (R4)            ; IS THERE AN ENTRY?
012636  001406                          BEQ     2$              ; IF NOT, DONE
012640  005034                          CLR     @(R4)+          ; INIT UDA
012642  005737  002200                  TST     NXMAD           ; WAS THERE AN ERROR?
012646  001010                          BNE     3$              ; IF SO, EXIT
012650  005303                          DEC     R3              ; MAKE SURE WE DO NOT EXTEND OVER AREA
012652  001370                          BNE     1$              ; IF NOT DONE, BRANCH
012654                          2$:     CLRVEC  #4
012654  012700  000004                                                        MOV     #4,R0
012660  104436                                                              TRAP    C$CVEC
012662                                  POP     <R4,R3>
012662  012604                                                              MOV (SP)+,R4
012664  012603                                                              MOV (SP)+,R3
012666  000207                          RETURN

012670  005744                  3$:     TST     -(R4)           ; R4 -> UDAIP THAT FAILED
012672  010405                          MOV     R4,R5           ; SAVE IN R5 FOR REPORT
012674                                  ERRDF   20.,ERR020
012674  104455                                                              TRAP    C$ERDF
012676  000024                                                              .WORD   20
012700  000000                                                              .WORD   0
012702  011750                                                              .WORD   ERR020
012704  005014                          CLR     (R4)            ; DESTROY ENTRY SO NOT TO FALL INTO RESET ERROR LOOP
012706                                  DOCLN
012706  104444                                                              TRAP    C$DCLN
```

```
                                        ;RUNDM
                                        ;
                                        ;LOAD AND RUN A DM PROGRAM IN THE CONTROLLERS. RETURN WHEN AL
                                        ;DM PROGRAMS HAVE TERMINATED.
                                        ;
                                        ;INPUTS:
                                        ;       TSTTAB   POINTER TO FIRST CONTROLLER TABLE
                                        ;       R1   NUMBER OF CONTROLLERS TO TEST
                                        ;IMPLICIT INPUTS:
                                        ;       DMPROG - POINTER TO START OF DM PROGRAM IN MEMORY
                                        ;OUTPUTS:
                                        ;       Z SET IF NO CONTROLLERS SUCCESSFULLY STARTED
                                        ;ALL REGISTERS ARE USED AND PREVIOUS CONTENTS DESTROYED.

012710  010137  002166          RUNDM:  MOV R1,URUN                     ;SAVE NUMBER OF UNITS TO RUN
012714  005037  002170                  CLR URNING                      ;CLEAR NUMBER OF UNITS RUNNING

                                        ;LOAD DM PROGRAM INTO EACH CONTROLLER

012720  013737  002166  002172          MOV URUN,UCNT                   ;SET COUNTER OF UNITS
012726  013705  002162                  MOV TSTTAB,R5                   ;GET FIRST CONTROLLER TABLE
012732                          LDDM:
012732  005065  000014                  CLR C.FLG(R5)                   ;CLEAR ALL FLAGS
012736  116537  000002  002074          MOVB C.UNIT(R5),L$LUN           ;SEE IF UNIT TO BE TESTED
012744  005765  000002                  TST C.UNIT(R5)
012750  100407                          BMI LDNEXT                      ;IF NOT, DON'T LOAD THIS UNIT
012752                                          ASSUME CT.AVL EQ BIT15
012752  004737  012546                  CALL HCOMM                      ;ALLOCATE SPACE FOR HOST COMM AREA
012756  004737  016770                  CALL LOADDM                     ;LOAD THE DM PROGRAM
012762  001402                          BEQ LDNEXT                      ;IF ERROR, GO TO NEXT CONTROLLER
012764  005237  002170                  INC URNING                      ;IF NO ERROR, COUNT UNIT RUNNING
012770  062705  000054          LDNEXT: ADD #C.SIZE,R5                  ;MOVE TO NEXT CONTROLLER TABLE
012774  005337  002172                  DEC UCNT                        ;CHECK IF MORE CONTROLLERS
013000  001354                          BNE LDDM                        ;LOAD NEXT
013002  005037  002176                  CLR UFREEZ                      ;CLEAR UNIT FREEZE FLAG
013006  012737  177777  003204          MOV #-1,FCTNUM  ;INVALIDATE FCT BLOCK NUMBER (BLOCK IN MEMORY)

                                        ;CHECK IF ANY CONTROLLERS LOADED

013014  005737  002170                  TST URNING                      ;ANY UNITS LOADED?

                                        ;THE DM PROGRAMS ARE NOW IN CONTROL
                                        ;RESPDM MUST BE CALLED TO RESPOND TO THEIR REQUESTS

013020  000207                          RETURN
```

```
                                        ;CLOSEF
                                        ;
                                        ;CLOSE DATA FILE FOR DM PROGRAMS
                                        ;
                                        ;INPUTS:
                                        ;       FILOPN   ZERO IF FILE NOT OPEN
                                        ;OUTPUTS:
                                        ;       NONE
                                        ;
013022  005737  002174          CLOSEF: TST FILOPN                      ;SEE IF FILE CURRENTLY OPEN
013026  001403                          BEQ 1$
013030                                  CLOSE                           ; IF SO, CLOSE IT
013030  104435                                                                  TRAP    C$CLOS
013032  005037  002174                  CLR FILOPN                      ;AND MARK AS SO
013036  000207                  1$:     RETURN
```

```
                                        ;RESPDM
                                        ;
                                        ;RESPOND TO DM REQUESTS. RETURN WHEN ALL DM PROGRAMS
                                        ;HAVE TERMINATED.

013040  013705  002162          RESPDM: MOV TSTTAB,R5                   ;GET CONTROLLER TABLE ADDRESS
013044  013737  002166  002172          MOV URUN,UCNT                   ;SET COUNTER OF UNITS
013052                          RESPCT: BREAK   ;ALLOW DRS TO SEE TERMINAL INPUT
013052  104422                                                                  TRAP    C$BRK
013054  016504  000016                  MOV C.RING(R5),R4               ;GET MOST COMM AREA ADDRESS
013060  032765  000002  000014          BIT @CT.RN,C.FLG(R5)            ;CHECK IF PROGRAM RUNNING
013066  001502                          BEQ RSPNXT                      ;IF NOT, LOOK AT NEXT
013070  116537  000002  002074          MOVB C.UNIT(R5),L$LUN           ;STORE UNIT NUMBER UNDER TEST
013076  032765  000010  000014          BIT @CT.MSG,C.FLG(R5)           ;SEE IF INTERRUPT RECEIVED
013104  001150                          BNE RSPIN                       ;IF SO, LOOK AT PACKET
013106  032765  000004  000014          BIT @CT.CMD,C.FLG(R5)           ;SEE IF COMMAND HAS BEEN SENT
013114  001002                          BNE 1$                          ;IF NOT, SEND ONE
013116  000137  013664                  JMP RSPOUT

                                        ;CHECK IF UDA STILL RUNNING

013122  011503                  1$:     MOV (R5),R3                     ;GET ADDRESS OF UDAIP
013124  016301  000002                  MOV 2(R3),R1                    ;LOOK AT UDASA REGISTER
013130  001405                          BEQ RSPTM                       ;IF ZERO, UDA STILL RUNNING
013132                                  ERROF 30.,ERR030                ;REPORT UDA HAS FATAL ERROR
013132  104455                                                                  TRAP    C$EROF
013134  000036                                                                  .WORD   30
013136  000000                                                                  .WORD   0
013140  012162                                                                  .WORD   ERR030
013142  000465                          BR RSPDRP                       ;DROP CONTROLLER FROM TESTING

                                        ;CHECK FOR TIMEOUT OF RESPONSE

013144  005765  000044          RSPTM:  TST C.TOT(R5)                   ;SEE IF DUP PROGRAM TO BE TIMED
013150  001451                          BEQ RSPNTO
013152  005737  003210                  TST KW.CSR                      ;SEE IF A CLOCK ON SYSTEM
013156  001446                          BEQ RSPNTO                      ;DON'T TIME IF NO CLOCK
013160  023765  003222  000042          CMP KW.EL+2,C.TOH(R5)           ;COMPARE TO TIMEOUT COUNTER
013166  101005                          BHI RSPTMO
013170  001041                          BNE RSPNTO
013172  023765  003220  000040          CMP KW.EL,C.TO(R5)
013200  103435                          BLO RSPNTO                      ;IF TOO MUCH TIME ELAPSED SINCE LAST INTERRUPT
013202  032765  000040  000014  RSPTMO: BIT @CT.STA,C.FLG(R5)           ;SEE IF A GET DUST STATUS COMMAND OUTSTANDING
013210  001101                          BNE RSPTOE                      ;REPORT ERROR IF SO
013212  005764  000012                  TST MC.CCT(R4)                  ;SEE IF UDA TOOK LAST COMMAND PACKET
013216  100476                          BMI RSPTOE                      ;REPORT ERROR IF NOT
013220  012700  000100                  MOV @CT.TM1,R0                  ;SEE IF FIRST TIMEOUT ALREADY HAPPENED
013224  032765  000100  000014          BIT @CT.TM1,C.FLG(R5)
013232  001401                          BEQ 1$                          ;IF SO,
013234  006300                          ASL R0                          ;SET SECOND TIME OUT FLAG
013236  052700  000040          1$:     BIS @CT.STA,R0                  ;SET THE PROPER TIMEOUT BIT
013242  050065  000014                  BIS R0,C.FLG(R5)                ;   AND STATUS REQUESTED BIT
013246  012700  000001                  MOV @OP.GDS,R0                  ;BUILD GET DUST STATUS COMMAND
013252  004737  016570                  CALL BLDCMD
013256  012764  100000  000012          MOV @RG.OWN,MC.CCT(R4)          ;MARK COMMAND TO UDA
013264  005775  000000                  TST @(R5)                       ;TELL UDA COMMAND IS THERE
013270  000137  013744                  JMP RSPOU4
```

53 0:3274 RSPNTO:

```
                                        ;SWITCH TO NEXT CONTROLLER
                                        ;
013274  005737  002176          RSPNXT: TST UFREEZ      ;FROZEN TO ONE UNIT?
013300  001264                          BNE RESPCT      ;STAY THERE IF SO
013302  062705  000054                  ADD #C.SIZE,R5                  ;MOVE TO NEXT TABLE
013306  005337  002172                  DEC UCNT                        ;CHECK IF MORE CONTROLLERS
013312  001257                          BNE RESPCT                      ;LOOK AT NEXT CONTROLLER
013314  000651                          BR RESPDM                       ;LOOK AT FIRST CONTROLLER AGAIN
                                        ;
                                        ;REMOVE A CONTROLLER FROM TESTING
                                        ;
013316  005065  000014          RSPDRP: CLR C.FLG(R5)   ;CLEAR PROGRAM RUNNING
013322  005037  002176                  CLR UFREEZ
013326  010504                          MOV R5,R4
013330  062704  000020                  ADD #C.DRO,R4
013334  012702  000010                  MOV #8.,R2
013340  012403                  1$:     MOV (R4)+,R3
013342  001420                          BEQ 3$
013344  005763  000002                  TST D.UNIT(R3)
013350                                          ASSUME DT.AVL EQ BIT15
013350  100003                          BPL 2$
013352  005302                          DEC R2
013354  001371                          BNE 1$
013356  000412                          BR 3$
013360  052763  100000  000002  2$:     BIS #DT.AVL,D.UNIT(R3)
013366  005302                          DEC R2
013370  001405                          BEQ 3$
013372  005714                          TST (R4)
013374  001403                          BEQ 3$
013376  004737  016370                  CALL LOADDM     ;START DM PROGRAM AGAIN
013402  001223                          BNE RESPCT
013404  005337  002170          3$:     DEC URNING                      ;REDUCE RUNNING CONTROLLERS COUNT
013410  001331                          BNE RSPNXT                      ;IF ANY STILL RUNNING, LOOK AT THEM
013412  000207                          RETURN                          ;ELSE RETURN TO TEST SECTION
                                        ;
013414                          RSPTOE: ERRDF 31.,ERR031                ;REPORT TIMEOUT ERROR
013414  104455                                                                  TRAP    C$ERDF
013416  000037                                                                  .WORD   31
013420  000000                                                                  .WORD   0
013422  012176                                                                  .WORD   ERR031
013424  000734                          BR RSPDRP                       ;DROP CONTROLLER FROM TESTING
```

```
      1                                 ;CONTROLLER HAS RESPONDED, LOOK AT MESSAGE PACKET
      2
      3                                 ;CHECK FOR PROPER OPCODE IN END PACKET
      4
      5 013426  012700  000204          RSPIN:  MOV #OP.END+OP.SSD,R0           ;GET SEND DATA END PACKET OPCODE
      6 013432  032765  000020  000014          BIT #CT.REQ,C.FLG(R5)           ;LOOK IF SEND DATA OR RECEIVE DATA
      7 013440  001402                          BEQ RSPMWR
      8 013442  012700  000205                  MOV #OP.END+OP.RSD,R0           ;CHANGE TO RECEIVE DATA END PACKET OPCODE
      9 013446  120064  000030          RSPMWR: CMPB R0,MC.MPK+P.OPCD(R4)       ;COMPARE TO OPCODE IN END PACKET
     10 013452  001145                          BNE RSPERR
     11
     12                                 ;LOOK AT STATUS CODE
     13
     14 013454  032764  000037  000032          BIT #ST.MSK,MC.MPK+P.STS(R4)    ;CHECK FOR STATUS CODE ST.SUC (ZERO)
     15 013462  001004                          BNE RSPERW
     16
     17                                 ;CHECK FOR EXPECTED REFERENCE NUMBER
     18
     19 013464  026564  000052  000020          CMP C.REF(R5),MC.MPK+P.CRF(R4)  ;CHECK IF CORRECT REF NUMBER
     20 013472  001405                          BEQ RSPPTW
     21 013474                          RSPERW: ERRDF 33.,ERR033
        013474  104455                                                          TRAP    C$ERDF
        013476  000041                                                          .WORD   33
        013500  000000                                                          .WORD   0
        013502  012226                                                          .WORD   ERR033
     22 013504  000704                          BR RSPDRP                       ;DROP UNIT FROM TESTING
     23
     24                                 ;CHECK IF RESPONSE FROM SEND OR RECEIVE DATA COMMAND
     25
     26 013506  032765  000020  000014  RSPPTW: BIT #CT.REQ,C.FLG(R5)           ;CHECK IF RESPONSE FROM DM PROGRAM
     27 013514  001463                  RSPOU:  BEQ RSPOUT                      ;LOOK AT REQUEST NUMBER IF SO
```

```
                                        ;MAINTENANCE READ END PACKET RECEIVED, LOOK AT REQUEST FROM DM PROGRAM

013516  016401  000430                  RSPPT2: MOV HC.BF2(R4),R1                      ;GET REQUEST NUMBER
013522  042701  007777                          BIC     #^C<DU.TYP>,R1                 ;CHECK TYPE
013526  001403                                  BEQ 1$                                 ;IF ZERO, ERROR
013530  020127  060000                          CMP R1,#DU.SPC                         ;CHECK IF IN EXPECTED RANGE
013534  101405                                  BLOS RSPPT3
013536                                  1$:     ERRDF 32.,ERR032                       ;BAD REQUEST NUMBER
013536  104455                                                                                              TRAP    C$ERDF
013540  000040                                                                                              .WORD   32
013542  000000                                                                                              .WORD   0
013544  012210                                                                                              .WORD   ERR032
013546  000663                                  BR RSPDRP                              ;DROP UNIT FROM TESTING

013550  016403  000034                  RSPPT3: MOV HC.MPK+P.BCNT(R4),R3               ;GET BYTE COUNT OF CHARACTERS RECEIVED IN R3
013554  162703  000002                          SUB #2,R3                              ;(FIRST TWO CHARACTERS ARE TYPE WORD)
013560  012700  000004                          MOV #OP.SSD,R0                         ;BUILD A SEND DATA COMMAND PACKET
013564  004737  016570                          CALL BLDCMD                            ;  FOR ANSWER TO DM PROGRAM
013570  012700  000164                          MOV #HC.BF1,R0                         ;POINT TO BUFFER IN PACKET
013574  004737  016732                          CALL CLRBUF                            ; AND CLEAR BUFFER
013600  010402                                  MOV R4,R2                              ;R2 POINTS TO SEND BUFFER
013602  062704  000244                          ADD #HC.BSZ,R4                ;R4 POINTS TO CHARACTERS IN RECEIVE BUFFER
013606  042724  170000                          BIC #DU.TYP,(R4)+             ;CLEAR TYPE FIELD IN BUFFER
013612  000301                                  SWAB R1                                ;GET TYPE RIGHT JUSTIFIED
013614  006201                                  ASR R1                                 ;TIMES TWO
013616  006201                                  ASR R1
013620  006201                                  ASR R1
013622  010100                                  MOV R1,R0                              ;COPY MESSAGE TYPE TO R0
013624  005001                                  CLR R1                                 ;R1 CONTAINS ZERO SEND BYTE COUNT
013626  004770  014112                          CALL @RSPDSP-2(R0)                     ;CALL REQUESTED ROUTINE
013632  001231                                  BNE RSPDRP                             ;ROUTINE RETURNS Z CLEAR TO DROP UNIT FROM TESTING
                                                                                       ; Z SET IF UNIT TO CONTINUE RUNNING
013634  016504  000016                          MOV C.RING(R5),R4                      ;GET RING ADDRESS
013640  032701  000001                          BIT #1,R1                              ;LOOK AT CHARACTER COUNT TO SEND TO DUP PROGRAM
013644  001401                                  BEQ 1$                                 ;IF AN ODD COUNT
013646  005201                                  INC R1                                 ;  INCREASE BY ONE
013650  010164  000120                  1$:     MOV R1,HC.CPK+P.BCNT(R4)               ;PUT CHARACTER COUNT IN COMMAND PACKET
013654  100003                                  BPL RSPOUT                             ;IF NEGATIVE BYTE COUNT RETURNED
013656  042765  000020  000014                  BIC #CT.REQ,C.FLG(R5)                  ;  DON'T SEND ANY DATA TO UDA

                                        ;SEND COMMAND BACK TO UDA

013664  042765  000350  000014  RSPOUT: BIC #CT.MSG+CT.STA+CT.TM1+CT.TM2,C.FLG(R5)      ;CLEAR MESSAGE RECEIVED FLAG
013672  032765  000020  000014          BIT #CT.REQ,C.FLG(R5)                          ;CHECK WHICH COMMAND TO SEND
013700  001014                          BNE RSPOU2                                     ;BRANCH IF RESPONSE TO REQUEST

013702  012700  000005                  MOV #OP.RSD,R0                                 ;BUILD RECEIVE DATA COMMAND
013706  004737  016570                  CALL BLDCMD
013712  012700  000430                  MOV #HC.BF2,R0                                 ;POINT TO MESSAGE BUFFER
013716  004737  016732                  CALL CLRBUF                                    ;  AND CLEAR IT
013722  052765  000020  000014          BIS #CT.REQ,C.FLG(R5)                          ;SET REQUEST BIT
013730  000403                          BR RSPOU3

013732  042765  000020  000014  RSPOU2: BIC #CT.REQ,C.FLG(R5)                          ;CLEAR REQUEST BIT
013740                          RSPOU3:
013740  004737  016654                  CALL SNDCMD                                    ;SEND COMMAND TO UDA
013744  016500  000044          RSPOU4: MOV C.TOT(R5),R0                               ;SET TIMEOUT
```

GLOBAL SUBROUTINES SECTION

```
                                                MOV R5,R1
                                                ADD #C.TO,R1                    ;PUT TIME IN CONTROLLER TABLE
                                                CALL SETTO
                                                JMP RSPNXT                      ;NOW WAIT FOR END PACKET
000030  RSPERR: CMPB #OP.END+OP.GDS,MC.MPK+P.OPCD(R4)   ;SEE IF GET DUST STATUS OPCODE
                BNE RSPERR
000037          BITB #DF.ACT,MC.MPK+P.DFLG(R4)  ;IF DUST NO LONGER RUNNING
                BEQ RSPTOE                      ;  REPORT ERROR
000014          BIC #CT.STA+CT.MSG,C.FLG(R5)    ;CLEAR CONTROL BITS
000014          BIT #CT.TM2,C.FLG(R5)           ;IF AT SECOND TIMEOUT
                BEQ 1$
000046          CMP MC.MPK+P.DPI(R4),C.PRI(R5)  ;COMPARE PROGRESS INDICATOR
                BNE 2$
000050          CMP MC.MPK+P.DPI+2(R4),C.PRI+2(R5)      ;COMPARE PROGRESS INDICATOR
                BEQ 4$                          ;REPORT ERROR IF NOT CHANGED
000014  2$:     BIC #CT.TM2,C.FLG(R5)           ;CLEAR TIMEOUT 2 FLAG
000014  1$:     BIT #CT.TM1,C.FLG(R5)           ;IF AT FIRST TIMEOUT
                BEQ 3$
000046          MOV MC.MPK+P.DPI(R4),C.PRI(R5)  ;GET COPY OF PROGRESS INDICATOR
000050          MOV MC.MPK+P.DPI+2(R4),C.PRI+2(R5)      ;GET COPY OF PROGRESS INDICATOR
000006  3$:     MOV #RG.OWN+RG.FLG,MC.MCT(R4)   ;GIVE MESSAGE BUFFER BACK TO UDA
                JMP RSPNXT
                JMP PSPTOE
```

```
ZUDEDO PDP 11 UDA DRV MTR MACRO V05.00 Tuesday 24 May 83 10:10 Page 69
GLOBAL SUBROUTINES SECTION

      1                                 ;RESPONSE REQUEST DISPATCH TABLE
      2
      3 014114  014130                  RSPDSP: .WORD QUEST                     ;QUESTION
      4 014116  014202                          .WORD DQUEST                    ;QUESTION WITH DEFAULT ANSWER
      5 014120  014354                          .WORD INFO                      ;INFORMATION MESSAGE FOR OPERATOR
      6 014122  014502                          .WORD TERM                      ;NORMAL TERMINATION
      7 014124  014512                          .WORD ERRTRM                    ;FATAL ERROR TERMINATION
      8 014126  014522                          .WORD SPECL                     ;SPECIAL
      9         000006                          DSPSIZ=.-RSPDSP/2               ;LEGAL NUMBERS ARE LOWER THAN THIS
```

```
                        ;NORMAL DUP RECEIVE DATA BUFFER DESCRIPTION
                        ;
                        ;BYTE OFFSET FROM
                        ;START OF BUFFER
                        ;        0          ! TYPE !     MESSAGE NUMBER     !  USED TO SELECT ROUTINE
                        ;                   !                               !
                        ;        2          !           DATA BYTES          !  R4 CONTAINS THIS ADDRESS
                        ;                   !                               !
                        ;        4          !           DATA BYTES          !
                        ;                   !                               !
                        ;        6          !           DATA BYTES          !
                        ;                   !                               !
                        ;        8          !           DATA BYTES          !
                        ;                   !                               !
                        ;       10          !           DATA BYTES          !
                        ;                   !                               !
                        ;       12          !           DATA BYTES          !
                        ;                   !                               !
                        ;       14          !           DATA BYTES          !
                        ;                   !                               !
                        ;       16          !           DATA BYTES          !
                        ;                   !                               !
                        ;       18          !           DATA BYTES          !
                        ;                   !                               !
                        ;       20          !           DATA BYTES          !
                        ;                   !                               !
                        ;       22          !           DATA BYTES          !
                        ;                   !                               !
                        ;                   !                               !
                        ;        .                        .
                        ;                   !                               !
                        ;        .                        .
                        ;                   !                               !
                        ;        .                        .
                        ;                   !                               !
                        ;                   !                               !
                        ;       80          !           DATA BYTES          !
                        ;                   !                               !
```

```
                ;NORMAL DUP SEND DATA BUFFER DESCRIPTION GIVEN IN RESPONSE TO ABOVE PACKET
                ;
                ;BYTE OFFSET FROM
                ;START OF BUFFER
                ;       0               !          DATA BYTES          !  R2 CONTAINS THIS ADDRESS
                ;                       !                              !
                ;       2               !          DATA BYTES          !
                ;                       !                              !
                ;       4               !          DATA BYTES          !
                ;                       !                              !
                ;       6               !          DATA BYTES          !
                ;                       !                              !
                ;       8               !          DATA BYTES          !
                ;                       !                              !
                ;       10              !          DATA BYTES          !
                ;                       !                              !
                ;       12              !          DATA BYTES          !
                ;                       !                              !
                ;       14              !          DATA BYTES          !
                ;                       !                              !
                ;       16              !          DATA BYTES          !
                ;                       !                              !
                ;       18              !          DATA BYTES          !
                ;                       !                              !
                ;       20              !          DATA BYTES          !
                ;                       !                              !
                ;       22              !          DATA BYTES          !
                ;                       !                              !
                ;       .                             .
                ;       .                             .
                ;       .                             .
                ;                       !                              !
                ;       80              !          DATA BYTES          !
                ;                       !                              !
```

```
                                        ;MESSAGE TYPE 1
                                        ;
                                        ;ANSWER QUESTION FOR DUP PROGRAM
                                        ;
                                        ;INPUT:
                                        ;       R5   ADDRESS OF CONTROLLER TABLE
                                        ;       R4   POINTER TO DATA IN RECEIVE BUFFER
                                        ;       R3   CHARACTER COUNT IN RECEIVE BUFFER
                                        ;       R2   POINTER TO SEND BUFFER (BUFFER IS CLEARED)
                                        ;       R1   ZERO
                                        ;OUTPUT:
                                        ;       R1   COUNT OF CHARACTERS IN SEND BUFFER
                                        ;       Z SET TO CONTINUE RUNNING DUP PROGRAM
                                        ;       Z CLEAR TO STOP THE DUP PROGRAM

014130  004737  014654        QUEST:  CALL GTDRVT     ;GET POINTER TO DRIVE TABLE
014134  062700  000004                ADD #D.SERN,R0  ;BUMP POINTER TO SERIAL NUMBER
014140  014403                        MOV -(R4),R3    ;GET QUESTION NUMBER
014142  001411                        BEQ QUE0        ;BRANCH IF QUESTION NUMBER 0
014144  020327  000007                CMP R3,#7       ;IF NOT, SEE IF QUESTION NUMBER 7
014150  001410                        BEQ QUE7
014152                                ERRDF 100.,ERR100       ;ANY OTHER NUMBER IS AN ERROR
014152  104455                                                          TRAP    C$ERDF
014154  000144                                                          .WORD   100
014156  000000                                                          .WORD   0
014160  012270                                                          .WORD   ERR10C
014162  000244                        CLZ     ;CLEAR Z TO STOP DUP PROGRAM
014164  000207                        RETURN

014166  012700  003306        QUE0:   MOV #DATE0,R0   ;POINT TO DATE STRING
014172                        QUE7:
014172  005201                QUEL:   INC R1  ;COUNT THE CHARACTERS
014174  112022                        MOVB (R0)+,(R2)+        ; AND PUT THEM IN OUTPUT BUFFER
014176  001375                        BNE QUEL        ;  UNTIL A NUL CHARACTER FOUND
014200  000207                        RETURN  ;RETURN WITH Z SET
```

```
                                        ;MESSAGE TYPE 2
                                        ;
                                        ;ANSWER QUESTION FOR DUP PROGRAM WITH DEFAULT ANSWER
                                        ;
                                        ;INPUT:
                                        ;       R5   ADDRESS OF CONTROLLER TABLE
                                        ;       R4   POINTER TO DATA IN RECEIVE BUFFER
                                        ;       R3   CHARACTER COUNT IN RECEIVE BUFFER
                                        ;       R2 - POINTER TO SEND BUFFER (BUFFER IS CLEARED)
                                        ;       R1   ZERO
                                        ;OUTPUT:
                                        ;       R1   COUNT OF CHARACTERS IN SEND BUFFER
                                        ;       Z SET TO CONTINUE RUNNING DUP PROGRAM
                                        ;       Z CLEAR TO STOP THE DUP PROGRAM

014202  004737  014654          DQUEST: CALL GTDRVT     ;GET DRIVE TABLE ADDRESS INTO R0
014206  014403                          MOV -(R4),R3    ;GET QUESTION NUMBER
014210  020327  000006                  CMP R3,#DQUESZ
014214  101035                          BHI DQUEX
014216  006303                          ASL R3
014220  000173  014224                  JMP @DQUEJP(R3)
014224  014310          DQUEJP: .WORD DQUEX     ; 0  (NOT USED)
014226  014242                  .WORD DQUNIT    ; 1  ENTER UNIT NUMBER TO FORMAT
014230  014310                  .WORD DQUEX     ; 2  (NOT USED)
014232  014310                  .WORD DQUEX     ; 3  (NOT USED)
014234  014314                  .WORD DQRFMT    ; 4  USE EXISTING BAD SECTOR INFORMATION
014236  014334                  .WORD DQRSTR    ; 5  DOWN-LINE LOAD BAD SECTOR BLOCK INFORMATION
014240  014344                  .WORD DQCONT    ; 6  CONTINUE IF BAD BLOCK INFO INACCESSIBLE
        000006                          DQUESZ=<<.-DQUEJP>/2>-1

                                        ;ENTER UNIT NUMBER TO FORMAT

014242                          DQUNIT: PUSH R5                                         MOV R5, (SP)
014242  010546
014244  005004                          CLR R4
014246  011003                          MOV (R0),R3     ;GET DRIVE NUMBER
014250                                          ASSUME D.DRV EQ 0
014250  012700  000012                  MOV #10.,R0     ;RADIX 10.
014254  004737  016332          DQUNL1: CALL DIVIDE
014260                                  PUSH R5                                         MOV R5, (SP)
014260  010546
014262  005201                          INC R1
014264  005703                          TST R3
014266  001372                          BNE DQUNL1
014270  010100                          MOV R1,R0
014272                          DQUNL2: POP R5                                          MOV (SP)+,R5
014272  012605
014274  062705  000060                  ADD #'0,R5
014300  110522                          MOVB R5,(R2)+
014302  005300                          DEC R0
014304  001372                          BNE DQUNL2
014306                                  POP R5                                          MOV (SP)+,R5
014306  012605
014310  000264          DQUEX:  SEZ
014312  000207                  RETURN

014314  032737  000003  003206  DQRFMT: BIT #SO.FMT,MODE
```

GLOBAL SUBROUTINES SECTION

```
    54 014322  001410                          BEQ DQNO
    55 014324  112712  000131          DQYES:  MOVB #'Y,(R2)
    56 014330  005201                          INC R1
    57 014332  000766                          BR DQUEX
    58
    59 014334  032737  000010  003206  DQRSTR: BIT #SO.STR,MODE
    60 014342  001370                          BNE DQYES
    61 014344                          DQCONT:
    62 014344  112712  000116          DQNO:   MOVB #'N,(R2)
    63 014350  005201                          INC R1
    64 014352  000756                          BR DQUEX
```

```
                                ;MESSAGE TYPE 3
                                ;
                                ;PRINT INFORMATION FROM DUP PROGRAM
                                ;
                                ;INPUT:
                                ;       R5   POINTER TO CONTROLLER TABLE
                                ;       R4   POINTER TO DATA IN RECEIVE BUFFER
                                ;       R3   CHARACTER COUNT IN RECEIVE BUFFER
                                ;       R2 - POINTER TO SEND BUFFER (BUFFER IS CLEARED)
                                ;       R1   ZERO
                                ;OUTPUT:
                                ;       R1   BIT 15 SET TO PREVENT SENDING DATA TO DUP PROGRAM
                                ;       Z SET TO CONTINUE RUNNING DUP PROGRAM
                                
014354  016400  177776          INFO:   MOV -2(R4),R0   ;GET MESSAGE NUMBER
014360  001434                  BEQ INFOB       ;IF ZERO, PRINT BEGUN MESSAGE
014362  020027  000100          CMP R0,#100     ;IF OCTAL 100
014366  001423                  BEQ INFOE       ;  PRINT ERROR MESSAGE
014370  020027  000200          CMP R0,#200     ;SEE IF 200 OR GREATER
014374  002005                  BGE INFOM       ;  IF SO, PRINT WITHOUT FREEZING
014376  005737  002176          TST UFREEZ
014402  001007                  BNE INFOP
014404  005237  002176          INC UFREEZ
014410  004737  014654          INFOM:  CALL GTDRVT
014414  010002                  MOV R0,R2
014416  004737  014700          CALL HEADER
014422  004737  014620          INFOP:  CALL MESG       ;PRINT THE MESSAGE
014426  012701  100000          INFOX:  MOV #BIT15,R1                   ;RETURN A NEGATIVE BYTE COUNT
014432  000264                  SEZ
014434  000207                  RETURN                                  ;RETURN WITH Z SET

014436                          INFOE:  ERRDF 101,,ERR101       ;ANSWER WAS REJECTED BY DUP PROGRAM
014436  104455                                                                  TRAP    C$ERDF
014440  000145                                                                  .WORD   101
014442  000000                                                                  .WORD   0
014444  012304                                                                  .WORD   ERR101
014446  000244                  CLZ     ;RETURN WITH Z CLEAR TO STOP DUP PROGRAM
014450  000207                  RETURN

014452  004737  014654          INFOB:  CALL GTDRVT      ;PRINT FORMAT BEGUN MESSAGE
014456  010002                  MOV R0,R2
014460  004737  014700          CALL HEADER
014464  004737  014620          CALL MESG
014470                          PNT WNSTOP      ;PRINT WARNING NOT TO STOP NOW
014470  004137  016264                                                        JSR R1,LPNT
014474  004322                                                                .WORD WNSTOP
014476  000000                                                                .WORD PNT.C*
014500  000752                  BR INFOX
```

```
                                        ;MESSAGE TYPE --
                                        ;
                                        ;TERMINATION MES AGE
                                        ;
                                        ;INPUT:
                                        ;       R5  - POINTER TO CONTROLLER TABLE
                                        ;       R4  - POINTER TO DATA IN RECEIVE BUFFER
                                        ;       R3  - CHARACTER COUNT IN RECEIVE BUFFER
                                        ;       R2 - POINTER TO SEND BUFFER (BUFFER IS CLEARED)
                                        ;       R1 - ZERO
                                        ;OUTPUT:
                                        ;       Z CLEAR TO TERMINATE DUP PROGRAM
                                        ;
014502  004737  014354          TERM:   CALL INFO       ;PRINT THE MESSAGE
014506  000244                          CLZ
014510  000207                          RETURN                          ;RETURN Z CLEAR TO TERMINATE DUP PROGRAM
```

```
                                ;MESSAGE TYPE 5
                                ;
                                ;ERROR TERMINATION MESSAGE
                                ;
                                ;INPUT:
                                ;       R5 - POINTER TO CONTROLLER TABLE
                                ;       R4   POINTER TO DATA IN RECEIVE BUFFER
                                ;       R3 - CHARACTER COUNT IN RECEIVE BUFFER
                                ;       R2   POINTER TO SEND BUFFER (BUFFER IS CLEARED)
                                ;       R1   ZERO
                                ;OUTPUT:
                                ;       Z CLEAR TO TERMINATE DUP PROGRAM

014512  004737  014354          ERRTRM: CALL INFO
014516  000244                          CLZ
014520  000207                          RETURN                          ;RETURN Z CLEAR TO TERMINATE DUP PROGRAM
```

```
                                        ;MESSAGE TYPE 6
                                        ;
                                        ;SPECIAL TYPE - READ FCT BLOCK FROM FILE
                                        ;
                                        ;INPUT:
                                        ;       R5   POINTER TO CONTROLLER TABLE
                                        ;       R4   POINTER TO DATA IN RECEIVE BUFFER
                                        ;       R3 - CHARACTER COUNT IN RECEIVE BUFFER
                                        ;       R2 - POINTER TO SEND BUFFER (BUFFER IS CLEARED)
                                        ;       R1 - ZERO
                                        ;OUTPUT:
                                        ;       Z SET TO SEND DATA TO PROGRAM
                                        
014522  023714  003204        SPECL:  CMP FCTNUM,(R4) ;SEE IF DESIRED BLOCK IS IN MEMORY
014526  001425                        BEQ SPECLX      ; IF SO, SEND TO DUP PROGRAM
014530  002407                        BLT SPECLR      ; IF LOWER NUMBERED BLOCK IN MEMORY,
                                                        ;   GO READ NEXT BLOCK
014532                          SPECLC:
014532                                  CLOSE   ;OTHERWISE, START READING FROM BEGINNING AGAIN
014532  104435                                                                  TRAP    C$CLOS
014534                                  OPEN @FNAME
014534  012700  003232                                                         MOV     @FNAME,R0
014540  104434                                                                 TRAP    C$OPEN
014542  012737  177777  003204        MOV #-1,FCTNUM
014550  012703  001000        SPECLR: MOV #512.,R3    ;GET BYTE COUNT IN A BLOCK
014554  012701  002204                MOV #FCTBUF,R1  ;POINT TO STORAGE AREA
014560                          SPECLL: GETBYTE (R1)+   ;READ THE FILE
014560  104426                                                                 TRAP    C$GETB
014562  110021                                                                 MOVB    R0,(R1)+
014564                                  BNCOMPLETE SPECLE       ;PRINT ERROR IF NO MORE BYTES IN FILE
014564  103005                                                                 BCC     SPECLE
014566  005303                          DEC R3  ;COUNT THE BYTES
014570  001373                          BNE SPECLL
014572  005237  003204                 INC FCTNUM      ;KEEP COUNT OF BLOCK IN MEMORY
014576  000751                          BR SPECL
                                        
014600  005212                SPECLE: INC (R2)        ;TELL DUP PROGRAM DATA NOT AVAILABLE
014602  012762  002204  000002 SPECLX: MOV #FCTBUF,2(R2)      ;PUT ADDRESS OF DATA IN OUTPUT BUFFER
014610  012701  000006                MOV #6,R1       ;SEND 3 WORDS TO DUP PROGRAM
014614  000264                        SEZ
014616  000207                        RETURN                  ;RETURN WITH Z SET TO SEND DATA TO DUP PROGRAM
```

```
                                        ;PRINT A MESSAGE IN THE RECEIVE BUFFER FROM THE DUP PROGRAM
                                        ;
                                        ;INPUT:
                                        ;       R4 - POINTER TO DATA IN RECEIVE BUFFER
                                        ;       R3   CHARACTER COUNT IN RECEIVE BUFFER
                                        ;OUTPUT:
                                        ;       R4   POINTER TO CHARACTER AFTER MESSAGE IN RECEIVE BUFFER
                                        ;       R3   ZERO
                                        ;       R1   BIT 15 SET TO PREVENT SENDING DATA TO DUP PROGRAM
                                        ;       R0   CONTENTS DESTROYED
                                        ;       Z SET TO CONTINUE RUNNING DUP PROGRAM

014620                                  MESG:
014620  112400                          1$:     MOVB (R4)+,R0                   ;PRINT CHARACTERS FROM DUP PROGRAM
014622  001405                                  BEQ 2$                          ;  DISCARDING LF AND NULL CHARACTERS
014624  020027  000012                          CMP R0,#12
014630  001402                                  BEQ 2$
014632                                          PRINT R0
014632  004737  016054                                                          CALL CPNT
014636  005303                          2$:     DEC R3                          ;COUNT THE CHARACTERS
014640  003367                                  BGT 1$
014642                                          PRINT #CR
014642  112700  000015                                                          MOVB #CR,R0
014646  004737  016054                                                          CALL CPNT
014652  000207                                  RETURN
```

```
                                ;GTDRVT
                                ;
                                ;GET DRIVE TABLE ADDRESS FROM CONTROLLER TABLE
                                ;
                                ;INPUTS:
                                ;       R5 - CONTROLLER TABLE ADDRESS
                                ;OUTPUTS:
                                ;       R0 - ADDRESS OF FIRST DRIVE TABLE AVAILABLE FOR TESTING
                                ;               (WITH DT.AVL BIT CLEAR)
                                ;
014654                          GTDRVT: PUSH R5
014654  010546                                                                 MOV R5, -(SP)
014656  062705  000020                  ADD #C.DR0,R5
014662  012500                  GTDRVL: MOV (R5)+,R0
014664  016037  000002  002074          MOV D.UNIT(R0),L$LUN
                                                ASSUME DT.AVL EQ BIT15
014672                                  BMI GTDRVL
014674                                  POP R5
014674  012605                                                                 MOV (SP)+,R5
014676  000207                          RETURN
```

```
                                        ;HEADER
                                        ;
                                        ;PRINT A HEADER IN FRONT OF EACH MESSAGE FROM DUP PROGRAM.
                                        ;A UDA ADDRESS IS PRINTED IF MORE THAN ONE UDA IS IN HARDWARE P TABLE.
                                        ;A RUNTIME IS PRINTED IF A CLOCK IS BEING USED TO TIME PROGRAM EXECUTION.
                                        ;
                                        ;INPUT:
                                        ;       R5 - POINTER TO CONTROLLER TABLE
                                        ;OUTPUT:
                                        ;       R0 - POINTER TO DRIVE TABLE
                                        ;       PRINTED MESSAGE

014700  022737  000001  002012  HEADER: CMP #1,L$UNIT                  ;IF MORE THAN ONE UNIT BEING TESTED
014706  001411                        BEQ 1$
                                        PNTF MESSG,D.UNIT(R2),(R5),(R2)         ;PRINT UDA ADDRESS
014710  011246                                                                    MOV (R2), (SP)
014712  011546                                                                    MOV (R5),-(SP)
014714  016246  000002                                                            MOV D.UNIT(R2), (SP)
014720  004137  016226                                                            JSR R1,LPNTF
014724  004027                                                                    .WORD MESSG
014726  000006                                                                    .WORD PNT.CT
                                                ASSUME C.UADR EQ 0
                                                ASSUME D.DRV EQ 0
014730  000407                        BR 2$
014732  005737  003210          1$:   TST KW.CSR                        ;IF NO CLOCK BEING USED
014736  001406                        BEQ 3$                            ;BYPASS RUNTIME MESSAGE
                                        PRINT #CR
014740  112700  000015                                                            MOVB #CR,R0
014744  004737  016054                                                            CALL CPNT
014750  004737  020242          2$:   CALL RNTIME                       ;PRINT RUNTIME IF A CLOCK IN USE
014754                          3$:   PRINT #CR
014754  112700  000015                                                            MOVB #CR,R0
014760  004737  016054                                                            CALL CPNT
014764  000207                        RETURN
```

```
                                        ;OSTRNG
                                        ;
                                        ;FORMAT OF THE ASCIZ STRING IS AS FOLLOWS:
                                        ;
                                        ;CHARACTERS ENCLOSED IN QUOTES ARE TO BE PRINTED AS THEY ARE.
                                        ;
                                        ;OTHERWISE CODE IS A SINGLE LETTER FOLLOWED BY AN OPTIONAL DECIMAL
                                        ;NUMBER:
                                        ;   ON - PRINT OCTAL NUMBER. N REPRESENTS SIZE OF BINARY NUMBER PASSED
                                        ;        IN PARAMETER IN BITS. MAY BE IN RANGE 1 TO 32. IF N>16, TWO PARAMETER
                                        ;        WORDS ARE USED, OTHERWISE ONLY ONE WORD. LEADING ZEROS ARE PRINTED.
                                        ;        N IS ALWAYS SPECIFIED.
                                        ;   DN - PRINT UNSIGNED DECIMAL NUMBER FROM N BIT PARAMETER. LEADING ZEROS
                                        ;        ARE NOT PRINTED. A 16 BIT NUMBER EQUAL TO ZERO WILL PRINT "0".
                                        ;   HN - PRINT HEX NUMBER FROM PARAMETER OF N BITS. IF N>16 TWO PARAMETERS
                                        ;        ARE USED, OTHERWISE ONLY ONE PARAMETER. LEADING ZEROS ARE PRINTED.
                                        ;   SN   PRINT N SPACES. N ASSUMED TO BE 1.
                                        ;   NN - START NEW LINE (CR-LF SEQUENCE). N ASSUMED TO BE 1.
                                        ;   AN - PRINT N ASCII CHARACTERS FROM PARAMETERS, N ASSUMED TO BE 1.
                                        ;        N/2 PARAMETER WORDS USED.
                                        ;   RN   EXECUTE ROUTINE #N. N MUST BE GIVEN AND DEFINED IN HOST PROGRAM.
                                        ;
                                        ;A NULL CHARACTER MEANS END OF MESSAGE. A NULL AS FIRST CHARACTER IN STRING
                                        ;MUST BE IGNORED.
                                        ;
                                        ;OUTPUT A MESSAGE ACCORDING TO A FORMAT STRING
                                        ;
                                        ;INPUTS:
                                        ;       R2 - ADDRESS OF START OF FORMAT STRING
                                        ;       R4   ADDRESS OF PARAMETERS
                                        ;OUTPUTS:
                                        ;       R2 AND R4 UPDATED TO END OF STRING AND PARAMETERS
                                        ;
014766  112201                  OSTRNG: MOVB (R2)+,R1                   ;GET CONTROL CHARACTER
014770  001421                          BEQ OSTRE                       ;EXIT IF NULL CHARACTER
014772  012700  015266                  MOV #ERRC,R0                    ;GET POINTER TO CHARACTER TABLE
014776  120110                  NCONS:  CMPB R1,(R0)                    ;COMPARE CHARACTER WITH TABLE ENTRY
015000  001407                          BEQ NCONF                       ;BRANCH IF MATCH FOUND
015002  105720                          TSTB (R0)+                      ;INCREMENT POINTER
015004  001374                          BNE NCONS                       ;CONTINUE SEARCH IF NOT END OF TABLE
015006                                  PNTF ERRME1                     ;REPORT BAD CONTROL CHARACTER
015006  004137  016226                                                          JSR R1,LPNTF
015012  003740                                                                  .WORD ERRME1
015014  000000                                                                  .WORD PNT.CT
015016  000406                          BR OSTRE
015020  162700  015266          NCONF:  SUB #ERRC,R0                    ;GET INCREMENT INTO TABLE
015024  006300                          ASL R0                          ;DOUBLE TO WORD COUNT
015026  004770  015300                  CALL @ERRD(R0)                  ;DISPATCH TO PRINT ROUTINE
015032  000755                          BR OSTRNG                       ;GET NEXT
015034  000207                  OSTRE:  RETURN
```

```
                                        ;CONTROL CHARACTER WAS A QUOTE. PRINT ALL CHARACTERS TO THE NEXT QUOTE.

015036  112200          CON.QU: MOVB (R2)+,R0           ;GET CHARACTER
015040  120027  000042          CMPB R0,#'"             ;CHECK IF ENDING QUOTE
015044  001403                  BEQ CON.QX              ;IF SO, GO GET NEXT CONTROL CHARACTER
015046                          PRINT R0                ;PRINT THE CHARACTER
015046  004737  016054                                                  CALL CPNT
015052  000771                  BR CON.QU               ;CONTINUE PRINTING
015054  000207          CON.QX: RETURN

                                        ;CONTROL CHARACTER WAS AN A. PRINT ASCII CHARACTERS FROM PARAMETERS.

015056  004737  015534  CON.A:  CALL GETCNT              ;GET COUNT OF CHARACTERS
015062  112400          CON.A1: PRINT (R4)+             ;PRINT THE CHARACTER
015062                                                          MOVB (R4)+,R0
015064  004737  016054                                          CALL CPNT
015070  005301                  DEC R1                  ;COUNT THE CHARACTERS
015072  001373                  BNE CON.A1              ;PRINT UNTIL COUNT REACHES ZERO
015074  032704  000001          BIT #1,R4               ;CHECK IF R4 NOW ODD
015100  001401                  BEQ CON.A2
015102  005204                  INC R4                  ;IF SO, INCREMENT TO NEXT EVEN ADDRESS
015104  000207          CON.A2: RETURN                  ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS A D. PRINT DECIMAL NUMBER.

015106  012701  000012  CON.D:  MOV #10.,R1              ;LOAD RADIX
015112  004737  015612          CALL PNTNUM             ;PRINT NUMBER
015116  000207                  RETURN                  ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS AN H. PRINT HEX NUMBER.

015120  012701  000020  CON.H:  MOV #16.,R1              ;LOAD RADIX
015124  004737  015612          CALL PNTNUM             ;PRINT NUMBER
015130  000207                  RETURN                  ;NOW GET NEXT CONTROL CHARACTER
```

```
                                ;CONTROL CHARACTER WAS AN O. PRINT OCTAL NUMBER.
                                ;
015132  012701  000010          CON.O:  MOV #8.,R1              ;LOAD RADIX
015136  004737  015612                  CALL PNTNUM             ;PRINT NUMBER
015142  000207                          RETURN                  ;NOW GET NEXT CONTROL CHARACTER
                                ;
                                ;CONTROL CHARACTER WAS AN N. PRINT NEW LINE SEQUENCE.
                                ;
015144  004737  015534          CON.N:  CALL GETCNT             ;GET COUNT
015150                          CON.N1: PRINT #CR               ;PRINT NEW LINE SEQUENCE
015150  112700  000015                                                  MOVB #CR,R0
015154  004737  016054                                                  CALL CPNT
015160  005301                          DEC R1                  ;COUNT THE SEQUENCES
015162  001372                          BNE CON.N1
015164  000207                          RETURN                  ;NOW GET NEXT CONTROL CHARACTER
                                ;
                                ;CONTROL CHARACTER WAS AN R. CALL A PRE-PROGRAMMED ROUTINE.
                                ;
015166  004737  015534          CON.R:  CALL GETCNT             ;GET ROUTINE NUMBER
015172  020127  000010                  CMP R1,#ERRSZ           ;CHECK IF DEFINED ROUTINE NUMBER
015176  101004                          BMI CON.R1
015200  060101                          ADD R1,R1               ;DOUBLE COUNT TO GET WORD INDEX
015202  004771  015244                  CALL @ERRRTB 2(R1)      ;CALL ROUTINE
015206  000207                          RETURN                  ;NOW GET NEXT CONTROL CHARACTER
015210                          CON.R1: PNTF ERRME1             ;REPORT BAD MESSAGE STRING
015210  004137  016226                                                  JSR R1,L.PNTF
015214  003740                                                          .WORD ERRME1
015216  000000                                                          .WORD PNT.CT
015220                                  POP R1                  ;FIX THE STACK
015220  012601                                                          MOV (SP)+,R1
015222  000207                          RETURN
                                ;
                                ;CONTROL CHARACTER WAS AN S. PRINT SPACES.
                                ;
015224  004737  015534          CON.S:  CALL GETCNT             ;GET COUNT
015230                          CON.S1: PRINT <#' '>            ;PRINT A SPACE
015230  112700  000040                                                  MOVB #' ',R0
015234  004737  016054                                                  CALL CPNT
015240  005301                          DEC R1                  ;COUNT THE SPACES
015242  001372                          BNE CON.S1
015244  000207                          RETURN                  ;NOW GET NEXT CONTROL CHARACTER
```

```
                                ;ERROR ROUTINE DISPATCH TABLE
                                ;
015246  015320                ERRRTB: .WORD CALRE                     ;NOT USED
015250  015320                        .WORD CALRE                     ;NOT USED
015252  015320                        .WORD CALRE                     ;NOT USED
015254  015332                        .WORD CALR4                     ;PRINT BASIC LINE WITHOUT UDA ADDRESS
015256  015406                        .WORD CALR5                     ;PRINT BASIC LINE WITH UDA ADDRESS
015260  015464                        .WORD CALR6                     ;CALL ALTERNATE PRINT STRING IN PDP-11 MEMOR
015262  015500                        .WORD CALR7                     ;PRINT  "REPLACE UDA MODULE M7485"
015264  015516                        .WORD CALR8                     ;PRINT  "   UDASA CONTAINS  XXXXXX"
        000010                      ERRRSZ=<.-ERRRTB>/2

                                ;BUILD TWO TABLES
                                ;       FIRST CONTAINING CONTROL CHARACTERS
                                ;       SECOND CONTAINING ROUTINE ADDRESSES

                                .MACRO BUILD
                                        ENTRY ",CON.QU
                                        ENTRY A,CON.A
                                        ENTRY D,CON.D
                                        ENTRY H,CON.H
                                        ENTRY O,CON.O
                                        ENTRY N,CON.N
                                        ENTRY R,CON.R
                                        ENTRY S,CON.S
                                .ENDM
```

```
                                ;HERE IS FIRST TABLE

                                .MACRO ENTRY ARG1,ARG2
                                        .LIST
                                        .BYTE   ARG1
                                        .NLIST
                                .ENDM

015266                          ERRC:   BUILD
015266  042                             .BYTE '
015267  101                             .BYTE  A
015270  104                             .BYTE  D
015271  110                             .BYTE 'H
015272  117                             .BYTE 'O
015273  116                             .BYTE  N
015274  122                             .BYTE 'R
015275  123                             .BYTE  S
015276  000                             .BYTE 0                 ;FOLLOW WITH A NULL BYTE
                                        .EVEN

                                ;HERE IS SECOND TABLE

                                .MACRO ENTRY ARG1,ARG2
                                        .LIST
                                        .WORD ARG2
                                        .NLIST
                                .ENDM

015300                          ERRD:   BUILD
015300  015036                          .WORD CON.QU
015302  015056                          .WORD CON.A
015304  015106                          .WORD CON.D
015306  015120                          .WORD CON.H
015310  015132                          .WORD CON.O
015312  015144                          .WORD CON.N
015314  015166                          .WORD CON.R
015316  015224                          .WORD CON.S
```

```
                                ;PRE-PROGRAMMED ROUTINES 1, 2 AND 3
                                ;NOT USED   PRINTS ERROR MESSAGE

4  015320                       CALRE.  PNTF ERRME1                ;PRINT ERROR MESSAGE
   015320  004137  016226                                                  JSR R1,LPNTF
   015324  003740                                                          .WORD ERRME1
   015326  000000                                                          .WORD PNT.CT
5  015330  000207                       RETURN
```

```
                                ;PRE-PROGRAMMED ROUTINE 4
                                ;PRINT BASIC LINE FOR HOST PROGRAM ERROR WITHOUT UDA ADDRESS
                                ;THEN SWITCH TO EXTENDED FORMAT

015332                          CALR4:  PNTB BASLN,@BASNO,@BAS,@BAS,@BAS
015332  012746  004223                                                    MOV @BAS,-(SP)
015336  012746  004223                                                    MOV @BAS,-(SP)
015342  012746  004223                                                    MOV @BAS,-(SP)
015346  012746  004150                                                    MOV @BASNO,-(SP)
015352  004137  016236                                                    JSR R1,LPNTB
015356  004224                                                           .WORD BASLN
015360  000010                                                           .WORD PNT.CT
015362  004737  020242                  CALL RNTIME
015366                                  PRINT @CR
015366  112700  000015                                                    MOVB @CR,R0
015372  004737  016054                                                    CALL LPNT
015376  012737  016154  003224          MOV @PX,PTYPE
015404  000207                          RETURN
```

```
                                ;PRE PROGRAMMED ROUTINE 5
                                ;PRINT BASIC LINE FOR HOST PROGRAM ERROR WITH UDA ADDRESS
                                ;THEN SWITCH TO EXTENDED FORMAT
                                
015406                          CALR5:  PNTB BASLN,@BASNO,@BASL2,(R5),@BAS,@BAS
015406  012746  004223                                                          MOV @BAS,-(SP)
015412  012746  004223                                                          MOV @BAS,-(SP)
015416  011546                                                                 MOV (R5),-(SP)
015420  012746  004167                                                          MOV @BASL2,(SP)
015424  012746  004150                                                          MOV @BASNO,-(SP)
015430  004137  016236                                                          JSR R1,LPNTB
015434  004224                                                                 .WORD BASLN
015436  000012                                                                 .WORD PNT.CT
015440  004737  020242                  CALL RNTIME
015444                                  PRINT @CR
015444  112700  000015                                                          MOVB @CR,R0
015450  004737  016054                                                          CALL CPNT
015454  012737  016154  003224          MOV @Px,PTYPE
015462  000207                          RETURN
```

```
                                ;PRE PROGRAMMED ROUTINE 6
                                ;CALL ALTERNATE PRINT ROUTINE IN PDP 11 MEMORY
                                
4 015464                        CALR6.  PUSH R2                 ;SAVE CURRENT STRING POINTER
  015464  010246                                                        MOV R2, (SP)
5 015466  012402                        MOV (R4)+,R2            ;GET NEW STRING POINTER
6 015470  004737  014766                CALL OSTRNG             ;OUTPUT USING THIS STRING
7 015474                                POP R2                  ;GET OLD POINTER BACK
  015474  012602                                                        MOV (SP)+,R2
8 015476  000207                        RETURN                  ;NOW CONTINUE THE OLD STRING
```

```
                                ;PRE-PROGRAMMED ROUTINE 7
                                ;PRINT "REPLACE UDA MODULE M7485"
                                
015500                          CALR7.  PUSH R2                         MOV R2, (SP)
015500  010246
015502  012702  011357                  MOV #XFRU,R2
015506  004737  014766                  CALL OSTRNG
015512                                  POP R2                          MOV (SP)+,R2
015512  012602
015514  000207                          RETURN
```

```
                                ;PRE-PROGRAMMED ROUTINE 8
                                ;PRINT  '  UDASA CONTAINS  XXXXXX"
                                
015516                          CALR8.  PUSH R2
015516  010246                                                    MOV R2, (SP)
015520  012702  011326                  MOV #XSA,R2
015524  004737  014766                  CALL OSTRNG
015530                                  POP R2
015530  012602                                                    MOV (SP)+,R2
015532  000207                          RETURN
```

```
                                ;GETCNT
                                ;
                                ;GET COUNT IN NEXT CHARACTERS OF STRING POINTED TO BY R2
                                ;NUMBER WILL BE IN DECIMAL. IF NO NUMBER, RETURN A
                                ;DEFAULT OF 1.
                                ;
                                ;INPUTS:
                                ;       R2 - POINTER TO ASCII STRING
                                ;OUTPUTS:
                                ;       R1   NUMBER READ OR A ONE
                                ;       R2 - POINTING TO CHARACTER AFTER NUMBER

015534                          GETCNT: PUSH R0
015534  010046                                                          MOV R0, (SP)
015536  005001                          CLR R1                  ;START WITH ZERO COUNT
015540  121227  000060          GETCNX: CMPB (R2),#'0           ;CHECK IF CHARACTER A DIGIT
015544  103415                          BLO GETCDN              ;BRANCH IF LOWER THAN ZERO
015546  121227  000071                  CMPB (R2),#'9
015552  101012                          BHI GETCDN              ;BRANCH IF HIGHER THAN NINE
015554  006301                          ASL R1                  ;MULTIPLY NUMBER BY 10
015556  010100                          MOV R1,R0               ; SAVE 2N
015560  006301                          ASL R1                  ; COMPUTE 4N
015562  006301                          ASL R1                  ; COMPUTE 8N
015564  060001                          ADD R0,R1               ; 8N + 2N = 10N
015566  112200                          MOVB (R2)+,R0           ;GET DIGIT FROM STING
015570  162700  000060                  SUB #'0,R0              ;GET RID OF ASCII
015574  060001                          ADD R0,R1               ;ADD TO NUMBER
015576  000760                          BR GETCNX               ;GO TO NEXT CHARACTER
015600  005701                  GETCDN: TST R1                  ;CHECK IF NUMBER IS ZERO
015602  001001                          BNE GETCXX              ;IF ZERO, CHANGE
015604  005201                          INC R1                  ; TO DEFAULT OF ONE
015606                          GETCXX: POP R0
015606  012600                                                          MOV (SP)+,R0
015610  000207                          RETURN
```

```
                                        ;PNTNUM
                                        ;
                                        ;PRINT A NUMBER
                                        ;
                                        ;INPUTS:
                                        ;       R1   RADIX OF NUMBER
                                        ;       R2   ASCII STRING TO COUNT OF BITS IN NUMBER
                                        ;       R4   POINTER TO NUMBER (LOW WORD)
                                        ;OUTPUTS:
                                        ;       NUMBER IS PRINTED. LEADING ZEROS ARE PRINTED EXCEPT FOR
                                        ;            DECIMAL NUMBERS.
                                        ;       R0   CONTENTS DESTROYED
                                        
015612  010100                  PNTNUM: MOV R1,R0                       ;SAVE RADIX
015614  004737  015534                  CALL GETCNT                     ;GET COUNT OF BITS
015620                          PNTNUS: PUSH <R2,R3,R5>
015620  010246                                                                  MOV R2, (SP)
015622  010346                                                                  MOV R3, (SP)
015624  010546                                                                  MOV R5,-(SP)
015626  012403                          MOV (R4)+,R3                    ;GET ONE PARAMETER WORD
015630  005005                          CLR R5                          ;CLEAR STORAGE FOR OTHER
015632  020127  000020                  CMP R1,#16.                     ;MORE THAN 16 BITS IN NUMBER?
015636  003401                          BLE 1$
015640  012405                          MOV (R4)+,R5                    ;YES, GET SECOND PARAMETER WORD
015642                          1$:     PUSH R4
015642  010446                                                                  MOV R4, (SP)
015644  010504                          MOV R5,R4                       ;PUT HIGH WORD IN R4
015646  012702  000020                  MOV #16.,R2                     ;COMPUTE BITS NOT WANTED
015652  160102                          SUB R1,R2                       ;BY SUBTRACTING BITS TO USE
015654  002002                          BGE 2$                          ;FROM 16.
015656  062702  000020                  ADD #16.,R2                     ;IF NEGATIVE, ADD 16 FOR FIRST WORD
015662  001414                  2$:     BEQ 6$                          ;IF ZERO, NO BITS NEED BE CLEARED
015664  012705  100000                  MOV #BIT15,R5                   ;START MASK WITH SIGN BIT SET
015670  005302                  3$:     DEC R2                          ;COUNT BITS IN MASK
015672  001402                          BEQ 4$
015674  006205                          ASR R5                          ;SHIFT MORE BITS TO RIGHT
015676  000774                          BR 3$
015700  020127  000020          4$:     CMP R1,#16.                     ;MORE THAN 16 BITS IN NUMBER?
015704  003402                          BLE 5$
015706  040504                          BIC R5,R4                       ;YES, CLEAR IN HIGH WORD
015710  000401                          BR 6$
015712  040503                  5$:     BIC R5,R3                       ;NO, CLEAR IN LOW WORD
015714  004737  016332          6$:     CALL DIVIDE                     ;DIVIDE BY RADIX IN R0
01572C                                  PUSH R5                         ;PUSH REMAINDER ON STACK
015720  010546                                                                  MOV R5, (SP)
015722  005202                          INC R2                          ;COUNT DIGITS ON STACK
015724  005703                          TST R3                          ;CHECK IF QUOTIENT IS ZERO
015726  001372                          BNE 6$
015730  005704                          TST R4
015732  001370                          BNE 6$
```

```
                                                ;IF RADIX IS DECIMAL
                                                ; JUST GO PRINT DIGITS ON STACK
                                                ;OTHERWISE COMPUTE NUMBER OF LEADING ZEROS
                                                ;DIVIDEND IS BITS IN NUMBER
                                                ;DIVISOR IS BITS PER DIGIT PRINTED
                                                ;   (3 OR 4)

                                                ;IF REMAINDER NOT ZERO
                                                ;INCREMENT QUOTIENT

                                                ;SUBTRACT DIGITS ON STACK
                                                ;NO LEADING ZEROS IF ZERO
                                                ;PRINT A ZERO
                                                        MOVB #'0,R0
                                                        CALL CPNT

                                                ;REPEAT UNTIL COUNT REACHES ZERO

                                                ;GET CHACACTER FROM STACK
                                                        MOV (SP)+,R5
                                                ;CNVERT TO ASCII DIGIT
                                                ;IF GREATER THAN A 9
                                                ; CONVERT TO A OR HIGHER
                                                ; FOR HEX DIGIT
                                                ;PRINT THE CHARACTER
                                                        MOVB R5,R0
                                                        CALL CPNT
                                                ;REPEAT FOR ALL DIGITS
                                                ; ON STACK
                                                        MOV (SP)+,R4
                                                        MOV (SP)+,R5
                                                        MOV (SP)+,R3
                                                        MOV (SP)+,R2
```

```
                                ;PRINT ONE CHARACTER
                                ;
                                ;CALL WITH MACRO PRINT
                        CPNT:   MOVB R0,ERRCHR
                                PUSH R1
016060  010146                                         MOV R1, (SP)
                                MOV @ERRONE,R1
                                CMPB R0,@CR
                                BNE 1$
                                MOV @ERRNL,R1
                        1$:     JMP @PTYPE
                        PF:     PRINTF R1,@ERRCHR
016104  012746  003226                                 MOV     @ERRCHR,-(SP)
016110  010146                                         MOV     R1, (SP)
016112  012746  000002                                 MOV     @2, (SP)
016116  010600                                         MOV     SP,R0
016120  104417                                         TRAP    C$PNTF
016122  062706  000006                                 ADD     @6,SP
                                BR CPNTX
                        PB:     PRINTB R1,@ERRCHR
016130  012746  003226                                 MOV     @ERRCHR, (SP)
016134  010146                                         MOV     R1,-(SP)
016136  012746  000002                                 MOV     @2,-(SP)
016142  010600                                         MOV     SP,R0
016144  104414                                         TRAP    C$PNTB
016146  062706  000006                                 ADD     @6,SP
                                BR CPNTX
                        PX:     PRINTX R1,@ERRCHR
016154  012746  003226                                 MOV     @ERRCHR, (SP)
016160  010146                                         MOV     R1,-(SP)
016162  012746  000002                                 MOV     @2,-(SP)
016166  010600                                         MOV     SP,R0
016170  104415                                         TRAP    C$PNTX
016172  062706  000006                                 ADD     @6,SP
                                BR CPNTX
                        PS:     PRINTS R1,@ERRCHR
016200  012746  003226                                 MOV     @ERRCHR, (SP)
016204  010146                                         MOV     R1, (SP)
016206  012746  000002                                 MOV     @2,-(SP)
016212  010600                                         MOV     SP,R0
016214  104416                                         TRAP    C$PNTS
016216  062706  000006                                 ADD     @6,SP
                        CPNTX:  POP R1
016222  012601                                         MOV (SP)+,R1
                                RETURN
```

```
                                        ;PRINT FORMATTED MESSAGE
                                        ;
                                        ;CALL WITH MACRO PNT, PNTF, PNTB, PNTX, OR PNTS

 5 016226  012737  016104  003224  LPNTF:  MOV @PF,PTYPE
 6 016234  000413                  LPNTB:  BR LPNT
 7 016236  012737  016130  003224  LPNTB:  MOV @PB,PTYPE
 8 016244  000407                  LPNTX:  BR LPNT
 9 016246  012737  016154  003224  LPNTX:  MOV @PX,PTYPE
10 016254  000403                  LPNTS:  BR LPNT
11 016256  012737  016200  003224  LPNTS:  MOV @PS,PTYPE
12 016264                          LPNT:   PUSH <R2,R3,R4,R5>
   016264  010246                                                  MOV R2, (SP)
   016266  010346                                                  MOV R3, (SP)
   016270  010446                                                  MOV R4, (SP)
   016272  010546                                                  MOV R5, (SP)
13 016274  012102                  MOV (R1)+,R2                    ;GET ADDRESS OF STRING
14 016276  010604                  MOV SP,R4                       ;COMPUTE ADDRESS OF ARGUMENTS
15 016300  062704  000012          ADD #10.,R4                     ;  WHICH ARE NOW ON STACK (IF ANY)
16 016304                          PUSH R1                         ;SAVE RETURN ADDRESS
   016304  010146                                                  MOV R1, (SP)
17 016306  004737  014766          CALL OSTRNG                     ;PRINT THE FORMATTED MESSAGE
18 016312                          POP <R0,R5,R4,R3,R2,R1>         ;RESTORE ALL REGISTERS
   016312  012600                                                  MOV (SP)+,R0
   016314  012605                                                  MOV (SP)+,R5
   016316  012604                                                  MOV (SP)+,R4
   016320  012603                                                  MOV (SP)+,R3
   016322  012602                                                  MOV (SP)+,R2
   016324  012601                                                  MOV (SP)+,R1
19 016326  062006                  ADD (R0)+,SP                    ;ADJUST STACK POINTER OVER ARGUMENTS
20 016330  000110                  JMP @R0                         ;RETURN
```

```
                                ;DIVIDE
                                ;
                                ;DIVIDE A 32 BIT UNSIGNED NUMBER BY A 16 BIT UNSIGNED NUMBER.
                                ;REPLACE DIVIDEND WITH QUOTIENT AND RETURN REMAINDER.
                                ;WILL NOT CHECK FOR DIVIDE BY ZERO.
                                ;
                                ;INPUTS:
                                ;       R3   LOW 16 BITS OF DIVIDEND
                                ;       R4   HIGH 16 BITS OF DIVIDEND
                                ;       R0   DIVISOR
                                ;OUTPUTS:
                                ;       R3   LOW 16 BITS OF QUOTIENT
                                ;       R4 - HIGH 16 BITS OF QUOTIENT
                                ;       R5 - REMAINDER
                                ;
016352                          DIVIDE: PUSH R2
016352  010246                                                          MOV R2, (SP)
016354  012702  000040                  MOV #32.,R2             ;SET UP SHIFT COUNT
016360  005005                  CLR R5                          ;START WITH ZERO REMAINDER
016362  006303                  1$:     ASL R3                  ;SHIFT LEFT INTO R5
016364  006104                  ROL R4
016366  006105                  ROL R5
016370  020005                  CMP R0,R5                       ;WILL DIVISOR GO INTO REMAINDER
016372  101002                  BHI 2$                          ;ONLY SUBTRACT IF IT WILL
016374  160005                  SUB R0,R5                       ;SUBTRACT DIVISOR
016376  005203                  INC R3                          ;PUT A ONE INTO QUOTIENT
016400  005302                  2$:     DEC R2                  ;COUNT THE SHIFTS
016402  001367                  BNE 1$
016404                          POP R2
016404  012602                                                          MOV (SP)+,R2
016406  000207                  RETURN
```

```
                                        ;LOADDM
                                        ;
                                        ;LOAD AND START A DM PROGRAM INTO A CONTROLLER
                                        ;
                                        ;INPUTS:
                                        ;       R5 - CONTROLLER TABLE ADDRESS
                                        ;       DMPROG   POINTER TO START OF DM PROGRAM IN MEMORY
                                        ;OUTPUTS:
                                        ;       IF LOAD SUCCEEDS   Z CLEAR
                                        ;                          CONTROLLER TABLE MARKED LOADED
                                        ;       IF ERROR   Z SET
                                        ;
016370  013701  002164          LOADDM: MOV DMPROG,R1                   ;GET STORAGE ADDRESS OF DM PROGRAM
016374  116165  000021  000044          MOVB DMTMO(R1),C.TOT(R5)        ;GET TIMEOUT VALUE
016402  105065  000045                  CLRB C.TOT+1(R5)
016406  016504  000004                  MOV C.VEC(R5),R4                ;GET VECTOR OF UDA
016412                                  AND CT.VEC,R4
016412  042704  177000                                                          BIC #^C<CT.VEC>,R4
016416  010501                          MOV R5,R1                       ;GET INTERRUPT SERVICE LINK
016420  062701  000010                  ADD #C.JSR,R1
016424                                  SETVEC R4,R1,#PRIO7             ;SET UP INTERRUPT VECTOR
016424  012746  000340                                                          MOV     #PRIO7,(SP)
016430  010146                                                                  MOV     R1,(SP)
016432  010446                                                                  MOV     R4,(SP)
016434  012746  000003                                                          MOV     #3,-(SP)
016440  104437                                                                  TRAP    C$SVEC
016442  062706  000010                                                          ADD     #10,SP
                                                                        ;INITIALIZE UDA WITH SMALLEST
016446  004737  017250                  CALL UDAINT                     ; RING BUFFER AND INTERRUPTS ENABLED
016452  001444                          BEQ LOADER                      ;BRANCH IF AN ERROR
```

```
 1 016454  012700  000002          MOV #OP.ESP,R0                  ;BUILD EXECUTE SUPPLIED PROGRAM COMMAND PACKET
 2 016460  004737  016570          CALL BLDCMD
 3 016464  013764  002164  000124  MOV DMPROG,MC.CPK+P.UADR(R4)    ;LOAD MAIN PROGRAM ADDRESS
 4 016472  017764  163466  000120  MOV @DMPROG,MC.CPK+P.BCNT(R4)   ;   AND SIZE
 5 016500  013764  002164  000140  MOV DMPROG,MC.CPK+P.OVRL(R4)    ;LOAD OVERLAY ADDRESS
 6 016506  063764  163452  000140  ADD @DMPROG,MC.CPK+P.OVRL(R4)
 7 016514  004737  016654          CALL SNDCMD                     ;SEND COMMAND TO UDA
 8 016520  004737  016774          CALL WAITMS                     ;WAIT FOR MESSAGE RESPONSE
 9 016524  001417                  BEQ LOADER      ;ABORT IF NO RESPONSE
10 016526  032764  000037  000032  BIT #ST.MSK,MC.MPK+P.STS(R4)    ;CHECK FOR ERRORS
11 016534  001007                  BNE LOADE1
12 016536  042765  000024  000014  BIC #CT.CMD+CT.REQ,C.FLG(R5)    ;CLEAR COMMAND OUTSTANDING FLAG
13 016544  052765  000002  000014  BIS #C*.RN,C.FLG(R5)            ;SET DM PROGRAM RUNNING FLAG
14 016552  000207                  RETURN
```

```
                        ;UDA FAILED TO DOWNLINE LOAD DM PROGRAM
                        ;
016554                  LOADE1: ERROF  34,,ERR034
016554  104455                                                          TRAP    C%ERDF
016556  000042                                                          .WORD   34
016560  000000                                                          .WORD   0
016562  012234                                                          .WORD   ERR034
016564  000264          LOADER: SEZ                     ;SET Z TO INDICATE ERROR OCCURRED
016566  000207                  RETURN
```

```
                        ;BLDCMD
                        ;
                        ;BUILD A COMMAND IN COMMAND PACKET
                        ;
                        ;INPUTS:
                        ;       R5 - CONTROLLER TABLE ADDRESS
                        ;       R0   COMMAND CODE
                        ;OUTPUTS:
                        ;       R4   ADDRESS OF HOST COMM AREA
                        ;       COMMAND PACKET CONTAINING REF NUMBER AND OPCODE, ALL OTHER FIELDS CLEARED.
                        ;       CMD REFERRENCE NUMBER IN CONTROLLER TABLE INCREMENTED AND RESULT
                        ;       IN COMMAND PACKET.
                        ;       R0   CONTENTS DESTROYED
                        ;
016570                  BLDCMD: PUSH <R1,R0>
016570  010146                                                                    MOV R1,-(SP)
016572  010046                                                                    MOV R0,-(SP)
016574  016504  000016          MOV C.RING(R5),R4          ;GET ADDRESS OF HOST COMM AREA
016600  010400                  MOV R4,R0                  ;COPY TO R0
016602  062700  000100          ADD #HC.CEV,R0             ;COMPUTE ADDRESS OF COMMAND ENVELOPE
016606  012720  000060          MOV #HC.PSZ,(R0)+          ;LOAD PACKET LENGTH
016612  012701  001000          MOV #DUP,R1                ;LOAD DIAG CIRCUIT IDENTIFIER
016616  022716  000031          CMP #OP.MWR,(SP)           ;IF CODE IS MAINTENANCE WRITE
016622  001002                  BNE BLDC0                  ;  GET OTHER CIRCUIT IDENTIFIER
016624  012701  177777          MOV #DIAG,R1
016630  010120          BLDC0:  MOV R1,(R0)+               ;PUT IDENTIFIER INTO PACKET
016632  012701  000030          MOV #<HC.PSZ>/2,R1         ;GET WORDS TO CLEAR
016636  005020          BLDC1:  CLR (R0)+                  ;CLEAR PACKET
016640  005301                  DEC R1
016642  001375                  BNE BLDC1
016644                          POP HC.CPK+P.OPCD(R4)      ;PUT OPCODE IN PACKET
016644  012664  000114                                                            MOV (SP)+,HC.CPK+P.OPCD(R4)
016650                          POP R1                     ;RESTORE R1
016650  012601                                                                    MOV (SP)+,R1
016652  000207                  RETURN
```

```
                                ;SNDCMD
                                ;
                                ;SEND A COMMAND TO THE UDA.
                                ;MARK BOTH PACKETS AVAILABLE TO THE
                                ;UDA. SET COMMAND ISSUED BIT IN CONTROLLER TABLE AND INITIALIZE
                                ;TIMEOUT COUNTER.
                                ;
                                ;INPUTS:
                                ;       R5    CONTROLLER TABLE ADDRESS
                                ;OUTPUTS:
                                ;       R4    ADDRESS OF HOST COMM AREA


016654                          SNDCMD: PUSH <R0,R1>
016654  010046                                                                    MOV R0, (SP)
016656  010146                                                                    MOV R1, -(SP)
016660  016504  000016                MOV C.RING(R5),R4           ;LOAD R4 WITH HOST COMM AREA ADDRESS
016664  005265  000052                INC C.REF(R5)               ;INCREMENT CMD REFERENCE NUMBER
016670  016564  000052  000104        MOV C.REF(R5),MC.CPK+P.CRF(R4)  ;PUT IN PACKET
016676  012764  140000  000006        MOV #RG.OWN+RG.FLG,MC.MCT(R4)   ;MARK MESSAGE PACKET AVAILABLE
016704  012764  100000  000012        MOV #RG.OWN,MC.CCT(R4)      ;MARK COMMAND TO UDA
016712  005775  000000                TST @(R5)                   ;TELL UDA COMMAND IS THERE
016716  052765  000004  000014        BIS #CT.CMD,C.FLG(R5)       ;MARK COMMAND ISSUED
016724                                POP <R1,R0>
016724  012601                                                                    MOV (SP)+,R1
016726  012600                                                                    MOV (SP)+,R0
016730  000207                        RETURN
```

```
                                        ;CLRBUF
                                        ;
                                        ;CLEAR THE SPECIFIED DATA BUFFER IN THE HOST COMM AREA
                                        ;AND LOAD BUFFER DESCRIPTOR IN COMMAND PACKET TO THE BUFFER
                                        ;
                                        ;INPUTS:
                                        ;       R5   CONTROLLER TABLE ADDRESS
                                        ;       R4   ADDRESS OF HOST COMM AREA
                                        ;       R0   OFFSET INTO HOST COMM AREA TO DATA BUFFER
                                        ;OUTPUTS:
                                        ;       DATA BUFFER CLEARED
                                        ;       COMMAND PACKET POINTING TO BUFFER
                                        ;       BYTE COUNT SET TO SIZE OF BUFFER
                                        ;       R4 - ADDRESS OF DATA BUFFER
                                        ;
016732                                  CLRBUF: PUSH <R0,R1>
016732  010046                                                                    MOV R0,-(SP)
016734  010146                                                                    MOV R1,-(SP)
016736  060400                                  ADD R4,R0                       ;ADD START OF HOST COMM AREA TO OFFSET
016740  010064  000124                          MOV R0,HC.CPK+P.UADR(R4)        ;PUT BUFFER ADDRESS IN COMMAND PACKET
016744  012764  000244  000120                  MOV #HC.BSZ,HC.CPK+P.BCNT(R4)   ;PUT SIZE OF BUFFER IN COMMAND PACKET
016752  010004                                  MOV R0,R4                       ;PUT BUFFER ADDRESS IN R4
016754  012701  000122                          MOV #HC.BSZ/2,R1                ;GET SIZE OF BUFFER IN WORDS
016760  005020                          CLRBFL: CLR (R0)+                       ;CLEAR ALL THE WORDS
016762  005301                                  DEC R1
016764  001375                                  BNE CLRBFL
016766                                          POP <R1,R0>
016766  012601                                                                    MOV (SP)+,R1
016770  012600                                                                    MOV (SP)+,R0
016772  000207                                  RETURN
```

```
                                        ;WAITMS
                                        ;
                                        ;WAIT FOR UDA TO RESPOND WITH A MESSAGE PACKET
                                        ;
                                        ;INPUTS:
                                        ;       R5 - ADDRESS OF CONTROLLER TABLE
                                        ;OUTPUTS:
                                        ;       Z CLEAR IF NO ERROR
                                        ;       Z SET IF ERROR, MESSAGE PRINTED
                                        ;
016774                                  WAITMS: PUSH <R0,R1>
016774  010046                                                                    MOV R0, (SP)
016776  010146                                                                    MOV R1, (SP)
017000  012700  000036                          MOV #30.,R0             ;SET TIME OUT VALUE OF 30 SECONDS
017004  010501                                  MOV R5,R1               ;POINT TO TIME OUT COUNTER
017006  062701  000040                          ADD #C.TO,R1
017012  004737  017166                          CALL SETTO
017016  011500                          1$:     MOV (R5),R0             ;GET ADDRESS OF UDAIP REGISTER
017020  032765  000010  000014                  BIT #CT.MSG,C.FLG(R5)   ;LOOK IF INTERRUPT OCCURRED
017026  001030                                  BNE 3$                  ;BRANCH IF SO
017030  016001  000002                          MOV 2(R0),R1            ;LOOK AT UDASA REGISTER
017034  001034                                  BNE 4$                  ;BRANCH IF ERROR CODE PRESENT
017036                                          BREAK
017036  104422                                                                    TRAP    C$BRK
017040  005737  003210                          TST KW.CSR              ;SEE IF A CLOCK ON SYSTEM
017044  001764                                  BEQ 1$
017046  023765  003222  000042                  CMP KW.EL+2,C.TOH(R5)   ;CHECK IF TIMEOUT HAS HAPPENED
017054  101005                                  BHI 2$
017056  001357                                  BNE 1$
017060  023765  003220  000040                  CMP KW.EL,C.TO(R5)
017066  103753                                  BLO 1$
017070                                  2$:     ERRDF 36.,ERR036
017070  104455                                                                    TRAP    C$ERDF
017072  000044                                                                    .WORD   36
017074  000000                                                                    .WORD   0
017076  012242                                                                    .WORD   ERR036
017100                                          POP <R1,R0>
017100  012601                                                                    MOV (SP)+,R1
017102  012600                                                                    MOV (SP)+,R0
017104  000264                                  SEZ
017106  000207                                  RETURN
```

```
      1 017110  042765  000010  000014  3$:     BIC #CT.MSG.C.FLG(R5)          ;CLEAR MESSAGE RECEIVED FLAG
      2 017116                                  POP <R1,R0>
        017116  012601                                                          MOV (SP)+,R1
        017120  012600                                                          MOV (SP)+,R0
      3 017122  000244                          CLZ                           ;GIVE NO ERROR RETURN
      4 017124  000207                          RETURN
      5 017126                            4$:   ERRDF 37,,ERR037
        017126  104455                                                          TRAP    C$ERDF
        017130  000045                                                         .WORD   37
        017132  000000                                                         .WORD   0
        017134  012254                                                         .WORD   ERR037
      6 017136                                  POP <R1,R0>
        017136  012601                                                          MOV (SP)+,R1
        017140  012600                                                          MOV (SP)+,R0
      7 017142  000264                          SEZ
      8 017144  000207                          RETURN
```

```
                                ;NXMI
                                ;
                                ;NON-EXISTANT MEMORY SERVICE ROUTINE
                                ;
                                ;INPUTS:
                                ;       NXMAD SET TO ZERO
                                ;OUTPUTS:
                                ;       NXMAD SET TO ONES IF NON EXISTANT TRAP OCCURED
                                
017146                          BGNSRV NXMI
017146                                                                  NXMI::
                                
017146  012737  177777  002200          MOV #-1,NXMAD
                                
017154                          ENDSRV
017154                                                                  L10031:
017154  000002                                                                RTI
```

```
                                ;UDASRV
                                ;
                                ;UDA INTERRUPT SERVICE ROUTINE. MARKS UDA CONTROLLER TABLE THAT AN
                                ;INTERRUPT HAS BEEN RECEIVED.
                                ;
                                ;THIS ROUTINE IS CALLED BY A (JSR R0,UDASRV) INSTRUCTION FROM WITHIN
                                ;THE CONTROLLER TABLE. THE PC STORED IN R0 IS THE ADDRESS OF THE C.FLG
                                ;WORD IN THE CONTROLLER TABLE. THE STACK CONTAINS THE SAVED CONTENTS
                                ;OF R0 FOLLOWED BY THE INTERRUPTED PC AND PS.
                                ;
                                ;INPUTS:
                                ;       R0 - ADDRESS OF C.FLG WORD IN CONTROLLER TABLE
                                ;       STACK   SAVED CONTENTS OF R0
                                ;OUTPUTS:
                                ;       CT.CMD CLEARED AND CT.MSG SET IN C.FLG WORD OF CONTROLLER TABLE
                                ;       R0   RESTORED FROM STACK
                                
017156                          BGNSRV UDASRV
017156                                                                  UDASRV::
017156  052710  000010                  BIS #CT.MSG,(R0)            ;SET CT.MSG
017162                                  POP R0                      ;RESTORE R0
017162  012600                                                              MOV (SP)+,R0
017164                          ENDSRV
017164                                                                  L10032:
017164  000002                                                              RTI
```

```
                                ;SETTO
                                ;
                                ;SET TIMEOUT COUNTER TO SOME NUMBER OF SECONDS FROM CURRENT TIME.
                                ;
                                ;INPUTS:
                                ;       R0  - NUMBER OF SECONDS FOR TIMEOUT
                                ;       R1  - ADDRESS WHERE TWO WORD TIME TO BE PUT
                                ;OUTPUTS:
                                ;       R0  - CONTENTS DESTROYED
                                ;       R1  - INCREMENTED BY 2
                                ;
                                ;COMPUTE CLOCK TICKS TIL TIMEOUT
                                ;
017166                          SETTO:  PUSH <R2,R3>
017166  010246                                                          MOV R2, (SP)
017170  010346                                                          MOV R3, (SP)
017172  005002                          CLR R2                  ;CLEAR PRODUCT
017174  013703  003216                  MOV KW.HZ,R3            ;GET MULTIPLICAND
017200  006200                  SET00:  ASR R0                  ;SHIFT MULTIPLIER TO RIGHT
017202  103001                          BCC SETD1               ;IF A ONE BIT SHIFTED OUT
017204  060302                          ADD R3,R2               ;  ADD MULTIPLICAND TO PRODUCT
017206  006303                  SET01:  ASL R3                  ;DOUBLE THE MULTIPLICAND
017210  005700                          TST R0
017212  001372                          BNE SET00               ;CONTINUE UNTIL MULTIPLIER IS ZERO
                                ;
                                ;GET CURRENT TIME
                                ;
017214  013700  003220          SET02:  MOV KW.EL,R0            ;GET TIME
017220  013703  003222                  MOV KW.EL+2,R3
017224  020037  003220                  CMP R0,KW.EL            ;IF CHANGED DURING RETRIEVAL
017230  001371                          BNE SET02               ;  GET IT AGAIN
                                ;
                                ;ADD TIME TIL TIMEOUT
                                ;
017232  060200                          ADD R2,R0               ;ADD
017234  005503                          ADC R3
                                ;
                                ;PUT RESULT IN STORAGE
                                ;
017236  010021                          MOV R0,(R1)+
017240  010311                          MOV R3,(R1)
                                ;
017242                                  POP <R3,R2>
017242  012603                                                          MOV (SP)+,R3
017244  012602                                                          MOV (SP)+,R2
017246  000207                          RETURN
```

```
                                        ;UDAINT
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       SUBROUTINE TO INITIALIZE A UDA AND BRING IT ON LINE.
                                        ;       ALL STEPS ARE CHECKED. AN ERROR MESSAGE IS REPORTED IF ANY ERROR
                                        ;       DETECTED.
                                        ;
                                        ;INPUTS:
                                        ;       R5 - ADDRESS OF CONTROLLER TABLE.
                                        ;IMPLICIT INPUTS:
                                        ;       C.RING(R5) - ADDRESS GIVEN TO UDA AS START OF RING BUFFER.
                                        ;       LENGTH OF RING STRUCTURE IS ONE ENTRY EACH.
                                        ;OUTPUTS:
                                        ;       CONDITION Z   SET IF ANY ERROR REPORTED. CLEAR IF NO ERROR.
                                        ;       R4   ADDRESS OF UDAIP REGISTER IN UDA
                                        ;       R5 - UNCHANGED.

                                        ;FILL HOST COMMUNICATION AREA WITH ALL ONES

017250  016502  000016          UDAINT: MOV C.RING(R5),R2               ;GET FIRST ADDRESS OF RING BUFFER
017254  012703  000006                  MOV #<MC.RSZ*2+MC.ISZ>/2,R3     ;GET SIZE OF RING BUFFER
017260  012722  177777          UDAI1L: MOV #-1,(R2)+                   ;WRITE ONES TO BUFFER
017264  005303                          DEC R3                          ;COUNT THE WORDS IN BUFFER
017266  003374                          BGT UDAI1L                      ;LOOP UNTIL ENTIRE BUFFER WRITTEN

                                        ;DO THE INITIALIZATION

017270  004737  017534                  CALL UDAIST                     ;DO FIRST THREE STEPS
017274  103515                          BCS UDAIEX                      ;GET OUT IF UDA MICROCODE REPORTED FAILURE
017276  012364  000002                  MOV (R3)+,2(R4)                 ;WRITE NEXT WORD TO UDASA REGISTER
017302  012703  000310                  MOV #200.,R3                    ;GET TRY COUNTER
017306  016402  000002          UDAI1A: MOV 2(R4),R2                    ;LOOK AT UDASA
017312  001410                          BEQ UDAI1C
017314  100005                          BPL UDAI1B
017316                                  ERRDF 24.,ERR024
017316  104455                                                                  TRAP    C$ERDF
017320  000030                                                                  .WORD   24
017322  000000                                                                  .WORD   0
017324  012130                                                                  .WORD   ERR024
017326  000500                          BR UDAIEX
017330  005303          UDAI1B: DEC R3
017332  001365                          BNE UDAI1A
017334  010264  000002          UDAI1C: MOV R2,2(R4)                    ;WRITE 0 TO UDASA (PURGE)
017340  011402                          MOV (R4),R2                     ;READ FROM UDAIP (POLL)
017342  004737  020060                  CALL UDARSP                     ;WAIT FOR STEP OR ERROR BIT
017346  103470                          BCS UDAIEX                      ;GET OUT IF UDA MICROCODE REPORTED FAILURE
017350  010203                          MOV R2,R3                       ;GET COPY OF MODEL CODE
017352  042703  177417                  BIC #^C<SA.CNT>,R3              ;CLEAR OTHER BITS
017356  006203                          ASR R3                          ;MOVE TO RIGHT OF REGISTER
017360  006203                          ASR R3
017362  006203                          ASR R3
017364  006203                          ASR R3
017366  042702  177760                  BIC #^C<SA.MCV>,R2              ;MICROCODE VERSION IN R2
017372  020327  000006                  CMP R3,#6                       ;CONTROLLER MODEL MUST BE 6
017376  001003                          BNE UDAI1D
017400  020227  000003                  CMP R2,#3                       ;MICROCODE VERSION MUST BE
017404  002005                          BGE UDAI2                       ;  3 OR GREATER
```

```
54 017406                UDAI1D: ERRDF  14,,ERR014         ;REPORT CONTROLLER NEEDS NEW REVISION
   017406  104455                                                  TRAP    C$ERDF
   017410  000016                                                  .WORD   14
   017412  000000                                                  .WORD   0
   017414  011732                                                  .WORD   ERR014
55 017416  000444                BR UDAIEX
```

```
                                        ;CHECK HOST COMMUNICATION AREA FOR ALL ZEROS

017420  016502  000016        UDAI2:  MOV C.RING(R5),R2               ;GET FIRST ADDRESS OF RING BUFFER
017424  012703  000006                MOV #<HC.RSZ*2+HC.ISZ>/2,R3     ;GET SIZE OF RING BUFFER
017430  005722                UDAI2L: TST (R2)+                       ;CHECK WORD IN BUFFER
017432  001003                        BNE UDAI2E                      ;GO TO ERROR REPORTER IF NOT ZERO
017434  005303                        DEC R3                          ;COUNT THE WORDS IN BUFFER
017436  003374                        BGT UDAI2L                      ;LOOP UNTIL ALL WORDS CHECKED
017440  000405                        BR UDAI3

017442                          UDAI2E: ERRDF 23,,ERRO23                ;REPORT BUFFER NOT CLEARED
017442  104455                                                                  TRAP    CSERRDF
017444  000027                                                                  .WORD   23.
017446  000000                                                                  .WORD   0
017450  012044                                                                  .WORD   ERRO23
017452  000426                        BR UDAIEX

                                        ;SEND GO BIT TO UDASA REGISTER TO END INITIALIZATION

017454                          UDAI3:
017454  016500  000006                MOV C.BST(R5),R0                ;GET BURST VALUE
017460  006300                        ASL R0                          ;SHIFT TO POSITION
017462  006300                        ASL R0
017464  052700  000001                BIS #SA.GO,R0                   ;SET THE GO BIT
017470  010064  000002                MOV R0,2(R4)                    ;SEND TO UDA
017474  016501  000016                MOV C.RING(R5),R1
017500  010161  000004                MOV R1,HC.MSG(R1)
017504  062761  000020  000004        ADD #HC.MPK,HC.MSG(R1)
017512  010161  000010                MOV R1,HC.CMD(R1)
017516  062761  000104  000010        ADD #HC.CPK,HC.CMD(R1)
017524  000244                        CLZ                             ;CLEAR Z AS NO ERROR INDICATION
017526  000207                        RETURN

                                        ;ERROR RETURN

017530  000264                UDAIEX: SEZ                             ;SET Z TO INDICATE ERROR OCCURRED
017532  000207                        RETURN
```

```
                                        ;UDAIST
                                        ;
                                        ;START THE INITIALIZATION PROCESS ON THE SELECTED UDA.
                                        ;STOP BEFORE WRITING THE THIRD WORD SO UDA DOES NOT
                                        ;ATTEMPT ANY UNIBUS TRANSFERS.
                                        ;
                                        ;INPUTS:
                                        ;       R5   ADDRESS OF CONTROLLER TABLE

                                        ;LOAD TABLE OF DATA TO SEND TO UDASA REGISTER

017534                                  UDAIST: BREAK
017534  104422                                                                  TRAP    C$BRK
017536                                          PUSH R1
017536  010146                                                                  MOV R1, -(SP)
017540  016504  000004                        MOV C.VEC(R5),R4
017544                                          AND CT.VEC,R4
017544  042704  177000                                                          BIC #^C<CT.VEC>,R4
017550  006204                                  ASR R4
017552  006204                                  ASR R4
017554  052704  100000                        BIS #SA.STP,R4                  ;SET STEP BIT IN DATA WORD
017560  010437  017752                        MOV R4,UDAID1                   ;LOAD INTERRUPT VECTOR
017564  016537  000016  017756                MOV C.RING(R5),UDAID2           ;LOAD MEMORY ADDRESS
017572  062737  000004  017756                ADD #HC.MSG,UDAID2              ;  OF FIRST RESPONSE RING

                                        ;START THE INITIALIZATION BY WRITING TO UDAIP REGISTER

017600  016504  000000                        MOV C.UADR(R5),R4               ;GET ADDRESS OF UDAIP REGISTER
017604  005037  002200                        CLR NXMAD                       ;CLEAR MEMORY ERROR FLAG
017610                                          SETVEC #4,#NXMI,#PRI07          ;SET UP VECTOR 4
017610  012746  000340                                                          MOV     #PRI07,-(SP)
017614  012746  017146                                                          MOV     #NXMI,-(SP)
017620  012746  000004                                                          MOV     #4,-(SP)
017624  012746  000003                                                          MOV     #3,-(SP)
017630  104437                                                                  TRAP    C$SVEC
017632  062706  000010                                                          ADD     #10,SP
017636  005764  000002                        TST 2(R4)                       ;ACCESS UDASA REGISTER
017642  005014                                  CLR (R4)                        ;WRITE TO UDAIP
017644                                          CLRVEC #4                       ;GIVE UP THE VECTOR
017644  012700  000004                                                          MOV     #4,R0
017650  104436                                                                  TRAP    C$CVEC
017652  005737  002200                        TST NXMAD                       ;SEE IF A MEMORY ERROR OCCURRED
017656  001406                                  BEQ UDAISG
017660                                          ERRDF 20.,ERR020
017660  104455                                                                  TRAP    C$ERDF
017662  000024                                                                  .WORD   20
017664  000000                                                                  .WORD   0
017666  011750                                                                  .WORD   ERR020
017670  000261                                  SEC
017672  000424                                  BR UDAISE
```

```
                                        ;SET UP LOOP PARAMETERS TO EXECUTE THE FOUR STEPS OF INITIALIZATION

017674  012737  004000  020216  UDAISG: MOV #SA.S1,UDARSD               ;STORE RESPONSE MASK
017702  012703  017750                  MOV #UDAIDT,R3                  ;AND INDEX TO TABLE

                                        ;WAIT FOR AND CHECK RESPONSE DATA

017706  004737  020060          UDAISL: CALL UDARSP                     ;WAIT FOR STEP OR ERROR BITS
017712  103414                          BCS UDAISE                      ;EXIT IF ERROR
017714  004733                          CALL @(R3)+                     ;CALL RESPONSE CHECKER FOR STEP
017716  103412                          BCS UDAISE                      ;GET OUT IF ERROR
017720  006337  020216                  ASL UDARSD                      ;SHIFT TO NEXT STEP BIT
017724  032737  040000  020216          BIT #SA.S4,UDARSD               ;CHECK IF NOW AT STEP 4
017732  001003                          BNE UDAISX                      ;GET OUT IF SO
017734  012364  000002                  MOV (R3)+,2(R4)                 ;WRITE DATA TO UDASA REGISTER
017740  000762                          BR UDAISL                       ;STAY IN LOOP

017742  000241          UDAISX: CLC                                     ;CLEAR CARRY FOR NO ERROR INDICATION
017744                  UDAISE: POP R1
017744  012601                                                          MOV (SP)+,R1
017746  000207                          RETURN
```

```
                                  ;DATA TO BE SENT AND RECEIVED BY UDA INITIALIZATION

017750  017766                    UDAIDT: .WORD UDAIR1              ;FIRST WORD RESPONSE CHECK ROUTINE
017752  000000                    UDAID1: .WORD 0                   ;FIRST WORD TO SEND TO UDASA
017754  017774                            .WORD UDAIR2              ;SECOND WORD RESPONSE CHECK ROUTINE
017756  000000                    UDAID2: .WORD 0                   ;SECOND WORD TO SEND TO UDASA
017760  020014                            .WORD UDAIR3              ;THIRD WORD RESPONSE CHECK ROUTINE
017762  100000                    UDAID3: .WORD SA.TST              ;THIRD WORD TO SEND TO UDASA
017764  020032                            .WORD UDAIR4              ;FOURTH WORD RESPONSE CHECK ROUTINE

                                  ;RESPONSE CHECK FOR FIRST WORD FROM UDASA
                                  ;CHECK FOR PROPER CONTROLLER TYPE

017766  012701  004400            UDAIR1: MOV #SA.S1+SA.DI,R1       ;SET STEP ONE BIT
017772  000422                            BR UDAIRC                 ;NOW COMPARE

                                  ;RESPONSE CHECK FOR SECOND WORD FROM UDASA
                                  ;CHECK FOR ECHO OF INTI AND VECTOR

017774  013701  017752            UDAIR2: MOV UDAID1,R1             ;GET WORD SENT TO UDASA
020000  000301                            SWAB R1                   ;GET HIGH 8 BITS
020002  042701  177400                    BIC #177400,R1
020006  052701  010000                    BIS #SA.S2,R1             ;SET STEP 2 BIT
020012  000412                            BR UDAIRC                 ;NOW COMPARE

                                  ;RESPONSE CHECK FOR THIRD WORD FROM UDASA
                                  ;CHECK FOR ECHO OF MESSAGE AND COMMAND RING LENGTHS

020014  013701  017752            UDAIR3: MOV UDAID1,R1             ;GET WORD SENT TO UDASA
020020  042701  177400                    BIC #177400,R1            ;JUST LOW 8 BITS
020024  052701  020000                    BIS #SA.S3,R1             ;SET STEP 3 BIT
020030  000403                            BR UDAIRC                 ;NOW COMPARE

                                  ;RESPONSE CHECK FOR FOURTH WORD FROM UDASA
                                  ;CHECK FOR ECHO OF PURGE AND LFAIL BITS

020032  010201                    UDAIR4: MOV R2,R1                 ;GET RESPONSE FROM UDA
020034  042701  137760                    BIC #^C<SA.S4+SA.MCV>,R1  ;KEEP MICROCODE VERSION AND STEP 4

                                  ;COMPARE EXPECTED DATA IN R1 WITH ACTUAL DATA IN R2

020040  020102                    UDAIRC: CMP R1,R2                 ;COMPARE THE DATA
020042  001405                            BEQ UDAIRX                ;EXIT IF COMPARED CORRECTLY
020044                                    ERRDF 25,,ERR025          ;REPORT ERROR
020044  104455                                                                           TRAP    C$ERDF
020046  000031                                                                           .WORD   25
020050  000000                                                                           .WORD   0
020052  012144                                                                           .WORD   ERR025
020054  000261                            SEC
020056  000207                    UDAIRX: RETURN
```

```
                                ;UDARSP
                                ;
                                ;WAIT FOR UDA TO RESPOND WITH DATA IN UDASA REGISTER.
                                ;EITHER STEP BIT FROM MASK IN LOCATION UDARSD OR ERROR BIT
                                ;WILL CAUSE A TERMINATION.
                                ;AN ERROR MESSAGE WILL BE PRINTED IF THE UDA DOES NOT RESPOND
                                ;IN 10 SECONDS OR IF ERROR SETS.
                                ;
                                ;INPUTS:
                                ;       UDASRD - MASK OF STEP BIT TO LOOK FOR
                                ;       R5   ADDRESS OF CONTROLLER TABLE
                                ;       R4   ADDRESS OF UDAIP REGISTER
                                ;OUTPUTS:
                                ;       ERROR MESSAGE IF TIME OUT ON RESPONSE OR ERROR BIT SETS
                                ;       R2   DATA FROM UDASA REGISTER
                                ;       CARRY SET IF ERROR BIT SETS OR TIME OUT
                                ;
020060                          UDARSP: PUSH R1
020060  010146                                                                  MOV R1, (SP)
020062  052737  100000  020216          BIS #SA.ERR,UDARSD              ;SET ERROR BIT IN MASK WORD
020070  012700  000012                  MOV #10.,R0                     ;SET UP FOR 10 SECOND TIMEOUT
020074  010501                          MOV R5,R1                       ;POINT TO COUNTER IN CONTROLLER TABLE
020076  062701  000040                  ADD #C.TO,R1
020102  004737  017166                  CALL SETTO
020106                                  POP R1
020106  012601                                                                  MOV (SP)+,R1
020110  033764  020216  000002  UDARS1: BIT UDARSD,2(R4)                 ;LOOK AT ERROR AND STEP BIT
020116  001024                          BNE UDARS2                      ;BRANCH IF EITHER SET
020120                                  BREAK
020120  104422                                                                  TRAP    C$BRK
020122  005737  003210                  TST KW.CSR                      ;SEE IF CLOCK ON SYSTEM
020126  001770                          BEQ UDARS1
020130  023765  003222  000042          CMP KW.EL+2,C.TOH(R5)           ;CHECK IF TIME OUT OCCURRED
020136  101005                          BHI 1$
020140  001363                          BNE UDARS1
020142  023765  003220  000040          CMP KW.EL,C.TO(R5)
020150  103757                          BLO UDARS1
020152  016402  000002          1$:     MOV 2(R4),R2                    ;GET REGISTER CONTENTS
020156                                  ERRDF 22.,ERR022                ;REPORT TIME OUT ERROR
020156  104455                                                                  TRAP    C$ERDF
020160  000026                                                                  .WORD   22
020162  000000                                                                  .WORD   0
020164  012016                                                                  .WORD   ERR022
020166  000407                          BR UDARSE
```

```
                                        ;CHECK IF ERROR BIT SET

020170  016402  000002        UDARS2: MOV 2(R4),R2            ;GET REGISTER CONTENTS
020174  100006                        BPL UDARSX              ;EXIT IF ERROR NOT SET
020176                                 ERRDF 21.,ERR021        ;REPORT ERROR INFO
020176  104455                                                         TRAP    C$ERDF
020200  000025                                                         .WORD   21.
020202  000000                                                         .WORD   0
020204  011762                                                         .WORD   ERR021
020206  000261                 UDARSE: SEC
020210  000207                         RETURN

                                        ;NORMAL EXIT

020212  000241                 UDARSX: CLC                     ;CLEAR CARRY AS NO ERROR INDICATION
020214  00020                          RETURN

                                        ;LOCATION FOR STEP BIT MASK

020216  000000                 UDARSD: .WORD 0                 ;LOAD BY CALLING ROUTINE
```

```
                                        ;KW11I
                                        ;
                                        ;CLOCK INTERRUPT SERVICE ROUTINE
                                        ;
020220                                  BGNSRV KW11I
020220                                                                          KW11I::
020220  062737  000001  003220                  ADD #1,KW.EL                  ;COUNT THE INTERRUPT
02022€  005537  003222                          ADC KW.EL+2
020232  012777  000105  162750                  MOV #KWOUT,@KW.CSR            ;RESTART THE CLOCK
020240                                  ENDSRV
020240                                                                          L10033:
020240  000002                                                                        RTI
```

```
                                ;RNTIME
                                ;
                                ;PRINT RUNTIME
                                ;
                                ;INPUTS:
                                ;       KW.EL - CONTAINS ELAPSED TIME
                                ;       KW.HZ   HERTZ OF CLOCK
                                ;OUTPUTS:
                                ;       IF CLOCK ON SYSTEM:
                                ;          "  RUNTIME HH:MM:SS " PRINTED
                                ;       IF NO CLOCK: ONE SPACE IS PRINTED
                                ;
020242  005737  003210          RNTIME: TST KW.CSR                      ;CHECK IF A CLOCK PRESENT
020246  001465                          BEQ RNTIMX                      ;BRANCH IF NOT
020250                                  PUSH <R0,R3,R4,R5>
020250  010046                                                                  MOV R0,-(SP)
020252  010346                                                                  MOV R3,-(SP)
020254  010446                                                                  MOV R4,-(SP)
020256  010546                                                                  MOV R5,-(SP)
020260  013703  003220                  MOV KW.EL,R3                    ;GET ELAPSED TIME
020264  013704  003222                  MOV KW.EL+2,R4
020270  013700  003216                  MOV KW.HZ,R0                    ;GET SPEED OF CLOCK
020274  004737  016332                  CALL DIVIDE                     ;COMPUTE SECONDS OF ELAPSED TIME
020300  012700  000074                  MOV #60.,R0                     ;NOW DIVIDE BY 60
020304  004737  016332                  CALL DIVIDE                     ; TO COMPUTE MINUTES
020310                                  PUSH R5                         ;SAVE REMAINDER AS SECONDS
020310  010546                                                                  MOV R5,-(SP)
020312  004737  016332                  CALL DIVIDE                     ;DIVIDE BY 60 AGAIN
020316                                  PNT RNTIM,R3                    ;PRINT HOURS
020316  010346                                                                  MOV R3,-(SP)
020320  004137  016264                                                          JSR R1,LPNT
020324  003704                                                                  .WORD RNTIM
020326  000002                                                                  .WORD PNT.CT
020330  020527  000011                  CMP R5,#9.                      ;IF MINUTES 9 OR LESS
020334  003004                          BGT 1$
020336                                  PRINT #'0                       ;PRINT A LEADING ZERO
020336  112700  000060                                                          MOVB #'0,R0
020342  004737  016054                                                          CALL CPNT
020346                          1$:     PNT RNTIM1,R5                   ;NOW PRINT MINUTES
020346  010546                                                                  MOV R5,-(SP)
020350  004137  016264                                                          JSR R1,LPNT
020354  003727                                                                  .WORD RNTIM1
020356  000002                                                                  .WORD PNT.CT
020360                                  POP R5                          ;GET SECONDS
020360  012605                                                                  MOV (SP)+,R5
020362  020527  000011                  CMP R5,#                        ;IF 9 OR LESS
020366  003004                          BGT 2$
020370                                  PRINT #'                        ;PRINT A LEADING ZERO
020370  112700  000060                                                          MOVB #'0,R0
020374  004737  016054                                                          CALL CPNT
020400                          2$:     PNT RNT    ...                  ;NOW PRINT SECONDS
020400  010546                                                                  MOV R5,-(SP)
020402  004137  016264                                                          JSR R1,LPNT
020406  003735                                                                  .WORD RNTIM2
020410  000002                                                                  .WORD PNT.CT
020412                                  POP <R5,R4,R3,R0>               ;HOURS IN R3
020412  012605                                                                  MOV (SP)+,R5
```

GLOBAL SUBROUTINES SECTION

```
     0204.4  012604                                                  MOV (SP)+,R4
     020416  012603                                                  MOV (SP)+,R3
     020420  012600                                                  MOV (SP)+,R0
35   020422                          RNTIMX: PRINT <0  >             ;PRINT A SPACE
     020422  112700  000040                                          MOVB #' ,R0
     020426  004737  016054                                          CALL CPNT
36   020432  000207                  RETURN
```

```
  020434                         DATE:   GMANID DATEQ,DATEI,A, 1,1,11.,YES       ;GET DATE
  020434  104443                                                                            TRAP    C$GMAN
  020436  000406                                                                            BR      10000$
  020440  003272                                                                            .WORD   DATEI
  020442  000152                                                                            .WORD   T$CODE
  020444  003536                                                                            .WORD   DATEQ
  020446  177777                                                                            .WORD   -1
  020450  000001                                                                            .WORD   T$LOLIM
  020452  000013                                                                            .WORD   T$HILIM
  020454                                                                            10000$:
  020454  012705  003272                 MOV #DATEI,R5   ;GET POINTER TO ANSWER
  020460  121527  000060                 CMPB (R5),#'0
  020464  103443                         BLO DERR
  020466  122527  000071         DAY:    CMPB (R5)+,#'9
  020472  101040                         BHI DERR
  020474  121527  000055                 CMPB (R5),#'-
  020500  001406                         BEQ DAS1
  020502  121527  000060                 CMPB (R5),#'0
  020506  103432                         BLO DERR
  020510  122527  000071                 CMPB (R5)+,#'9
  020514  101027                         BHI DERR
  020516  122527  000055         DAS1:   CMPB (R5)+,#'-
  020522  001024                         BNE DERR
  020524  012704  000014                 MOV #12.,R4     ;GET NUMBER OF MONTH
  020530  012703  003347                 MOV #MONTHS,R3  ;GET POINTER TO MONTH NAMES
  020534  005000                 MON1:   CLR R0
  020536  121523                         CMPB (R5),(R3)+
  020540  001401                         BEQ MON2
  020542  005200                         INC R0
  020544  126523  000001         MON2:   CMPB 1(R5),(R3)+
  020550  001401                         BEQ MON3
  020552  005200                         INC R0
  020554  126523  000002         MON3:   CMPB 2(R5),(R3)+
  020560  001401                         BEQ MON4
  020562  005200                         INC R0
  020564  005700                 MON4:   TST R0
  020566  001407                         BEQ MON5
  020570  005304                         DEC R4
  020572  001360                         BNE MON1
  020574                         DERR:   PNTF DATEX
  020574  004137  016226                                                                    JSR R1,LPNTF
  020600  011503                                                                            .WORD DATEX
  020602  000000                                                                            .WORD PNT.CT
  020604  000713                         BR DATE
  020606  012701  003306         MON5:   MOV #DATEO,R1   ;GET POINTER TO DATE FOR FORMATTER
  020612  010403                         MOV R4,R3       ;GET COPY OF MONTH NUMBER
  020614  020327  000012                 CMP R3,#10.     ;  IF 10 OR GREATER
  020620  103404                         BLO MON6
  020622  112721  000061                 MOVB #'1,(R1)+  ;PUT A "1" IN OUTPUT
  020626  162703  000012                 SUB #10.,R3
  020632  062703  000060         MON6:   ADD #'0,R3      ;CONVERT MONTH NUMBER TO ASCII
  020636  110321                         MOVB R3,(R1)+   ;PUT A NUMBER IN OUTPUT
  020640  112721  000055                 MOVB #'-,(R1)+  ;PUT A "-" IN OUTPUT
  020644  062704  003412                 ADD #DAYS-1,R4  ;GET POINTER TO DAYS IN MONTH
                                                                 ;INDEXED BY NUMBER OF MONTH
  020650  012703  003272                 MOV #DATEI,R3   ;GET POINTER TO DATE INPUT
  020654  005000                         CLR R0
```

```
 46 020656  121327  000055          DAY1:   CMPB (R3),#'
 47 020662  001413                          BEQ DAY2
 48 020664  111321                          MOVB (R3),(R1)+ ;PUT DAY CHARACTER IN OUTPUT
 49 020666  006300                          ASL R0
 50 020670  010002                          MOV R0,R2
 51 020672  006300                          ASL R0
 52 020674  006300                          ASL R0
 53 020676  060200                          ADD R2,R0
 54 020700  112302                          MOVB (R3)+,R2
 55 020702  162702  000060                  SUB #'0,R2
 56 020706  060200                          ADD R2,R0
 57 020710  000762                          BR DAY1
 58 020712  120014                  DAY2:   CMPB R0,(R4)
 59 020714  101327                          BHI DERR
 60 020716  005700                          TST R0  ;SEE IF DATE IS ZERO
 61 020720  001725                          BEQ DERR        ;ERROR IF SO
 62 020722  062705  000003                  ADD #3,R5
 63 020726  121527  000055                  CMPB (R5),#'-   ;CHECK FOR " " BETWEEN DAY
 64 020732  001320                          BNE DERR        ;  AND YEAR IN OUTPUT
 65 020734  112521                          MOVB (R5)+,(R1)+        ;PUT "-" IN OUTPUT
 66 020736  010504                          MOV R5,R4       ;GET COPY OF INPUT STRING POINTER
 67 020740  005000                          CLR R0
 68 020742  005002                          CLR R2
 69 020744  121427  000060          YER1:   CMPB (R4),#'0
 70 020750  103416                          BLO YER2
 71 020752  121427  000071                  CMPB (R4),#'9
 72 020756  101013                          BHI YER2
 73 020760  006300                          ASL R0
 74 020762  010003                          MOV R0,R3
 75 020764  006300                          ASL R0
 76 020766  006300                          ASL R0
 77 020770  060300                          ADD R3,R0
 78 020772  112403                          MOVB (R4)+,R3
 79 020774  162703  000060                  SUB #'0,R3
 80 021000  060300                          ADD R3,R0
 81 021002  005202                          INC R2
 82 021004  000757                          BR YER1
 83 021006  105714                  YER2:   TSTB (R4)
 84 021010  001271                          BNE DERR
 85 021012  020227  000002                  CMP R2,#2
 86 021016  001407                          BEQ YER3
 87 021020  020227  000004                  CMP R2,#4
 88 021024  001263                          BNE DERR
 89 021026  020027  003554                  CMP R0,#1900.
 90 021032  103660                          BLO DERR
 91 021034  000413                          BR YER5
 92 021036  012702  003427          YER3:   MOV #YEAR19,R2
 93 021042  020027  000106                  CMP R0,#70.
 94 021046  103002                          BHIS YER4
 95 021050  012702  003432                  MOV #YEAR20,R2
 96 021054  105712                  YER4:   TSTB (R2)
 97 021056  001402                          BEQ YER5
 98 021060  112221                          MOVB (R2)+,(R1)+
 99 021062  000774                          BR YER4
100 021064  112521                  YER5:   MOVB (R5)+,(R1)+
101 021066  001376                          BNE YER5
102 021070  000207                          RETURN
```

```
    103
104 021072                          ENDMOD
```

```
                                .SBTTL  PROTECTION TABLE
                        
021072                          BGNMOD
                        
                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--
                        
021072                          BGNPROT
021072                                                                  L$PROT::
                        
021072  177777                   1                                ;OFFSET INTO P-TABLE FOR CSR ADDRESS
021074  177777                   1                                ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
021076  177777                  -1                                ;OFFSET INTO P TABLE FOR DRIVE NUMBER
                        
021100                          ENDPROT
```

```
                                            .SBTTL  INITIALIZE SECTION

                                            ;++
                                            ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                            ; AT THE BEGINNING OF EACH PASS.
                                            ;

021100                                              BGNINIT
021100                                                                                L$INIT::

021100                                              READEF  #EF.START               ;CHECK IF STARTED BY OPERATOR
021100  012700  000040                                                                          MOV     #EF.START,R0
021104  104447                                                                                  TRAP    C$REFG
021106                                              BCOMPLETE INIT1                 ;  IF NOT,
021106  103415                                                                                  BCS     INIT1
021110                                              READEF  #EF.RESTART
021110  012700  000037                                                                          MOV     #EF.RESTART,R0
021114  104447                                                                                  TRAP    C$REFG
021116                                              BCOMPLETE INIT1
021116  103411                                                                                  BCS     INIT1
021120                                              READEF  #EF.CONTINUE
021120  012700  000036                                                                          MOV     #EF.CONTINUE,R0
021124  104447                                                                                  TRAP    C$REFG
021126                                              BCOMPLETE INIT1
021126  103405                                                                                  BCS     INIT1
021130                                              READEF  #EF.PWR
021130  012700  000034                                                                          MOV     #EF.PWR,R0
021134  104447                                                                                  TRAP    C$REFG
021136                                              BCOMPLETE INIT1
021136  103401                                                                                  BCS     INIT1
021140                                      INITQT: DOCLN                           ;  ABORT PROGRAM
021140  104444                                                                                  TRAP    C$DCLN

021142  012700  000003                      INIT1:  MOV #SO.FMT,R0  ;BUILD MODE WORD FROM SOFTWARE QUESTIONS
021146  030037  002144                              BIT R0,SFPTBL   ;SEE IF REFORMAT
021152  001011                                      BNE INIT2       ;  BRANCH IF SO
021154  012700  000004                              MOV #SO.CNS,R0  ;SEE IF RECONSTRUCT
021160  030037  002144                              BIT R0,SFPTBL
021164  001004                                      BNE INIT2       ;  BRANCH IF SO
021166  006300                                      ASL R0   ;SEE IF RESTORE
021170                                                      ASSUME SO.STR EQ SO.CNS*2
021170  030037  002144                              BIT R0,SFPTBL
021174  001761                                      BEQ INITQT      ;IF NOT, ABORT PROGRAM
021176  010037  003206                      INIT2:  MOV R0,MODE     ;SAVE MODE FLAGS
021202  004737  012564                      INIT3:  CALL    RESET                   ;RESET ALL UNITS
021206                                              MEMORY FFREE                    ;RESET START OF FREE MEMORY
021206  104431                                                                                  TRAP    C$MEM
021210  010037  002146                                                                          MOV     R0,FFREE
021214  017737  160726  002150                      MOV @FFREE,FSIZE                ;RESET SIZE OF FREE MEMORY

                                            ;INITIALIZE CLOCK

        000105                                       KWOUT.=105                     ;DATA TO SEND TO KW11 TO START CLOCK
021222  005037  003210                              CLR KW.CSR                      ;MARK CLOCK AS NOT ON SYSTEM
021226  005037  003220                              CLR KW.EL                       ;CLEAR ELAPSED TIME
021232  005037  003222                              CLR KW.EL+2
021236                                              CLOCK L,R0                      ;SEE IF AN L CLOCK PRESENT
```

INITIALIZE SECTION

```
        021236  012700  000114                                                MOV     #'L,R0
        021242  104462                                                      TRAP    C$CLCK
43      021244                          BCOMPLETE KYES
        021244  103413                                                      BCS     KYES
45      021246                          CLOCK P,R0                  ;SEE IF A P CLOCK PRESENT
        021246  012700  000120                                                MOV     #'P,R0
        021252  104462                                                      TRAP    C$CLCK
44      021254                          BCOMPLETE KYES
        021254  103407                                                      BCS     KYES
45      021256  005037  003210          CLR KW.CSR                  ;IF NEITHER, CLEAR CSR STORAGE WORD
46      021262                          PNTF NOCLOCK
        021262  004137  016226                                                JSR R1,LPNTF
        021266  004073                                                      .WORD NOCLOCK
        021270  000000                                                      .WORD PNT.CT
47      021272  000426                  BR KNO
48      021274  012037  003210  KYES:   MOV (R0)+,KW CSR             ;STORE DATA RETURNED
49      021300  012037  003212          MOV (R0)+,KW.BRL
50      021304  012037  003214          MOV (R0)+,KW.VEC
51      021310  012037  003216          MOV (R0)+,KW.MZ
52      021314                          SETVEC KW.VEC,#KW11I,KW.BRL ;SET THE VECTOR
        021314  013746  003212                                                MOV     KW.BRL,-(SP)
        021320  012746  020220                                                MOV     #KW11I,(SP)
        021324  013746  003214                                                MOV     KW.VEC,(SP)
        021330  012746  000003                                                MOV     #3,(SP)
        021334  104437                                                      TRAP    C$SVEC
        021336  062706  000010                                                ADD     #10,SP
55      021342  012777  000105  161640          MOV #KWOUT.,@KW.CSR         ;START THE CLOCK
54      021350                  KNO:
```

```
                                        ;INITIALIZE CONTROLLER TABLE STORAGE WITH A WORD OF ZEROS
                                        ;
021350  013737  002146  002156            MOV FFREE,CTABS             ;STORE START OF CONTROLLER TABLES
021356  005077  160574                    CLR @CTABS                  ;ZEROS MARKS END CONTROLLER TABLES
021362  005037  002160                    CLR CTRLRS                  ;CLEAR CONTROLLER COUNT
021366  012701  003436                    MOV     #IPADRS,R1          ; R1 -> IP ADDRESS
021372  012702  000010                    MOV     #8.,R2              ; R2 IS A COUNTER
021376  005021                  1$:       CLR     (R1)+               ; CLEAR ENTRY
021400  005302                            DEC     R2                  ; DONE?
021402  001375                            BNE     1$                  ; IF NOT, BRANCH
                                        ;
                                        ;GET A P TABLE FROM DRS
                                        ;
021404  005002                            CLR R2                      ;LOGICAL UNIT NUMBER IN R2
021406                          INIT4:    GPHARD R2,R0                ;GET POINTER TO A P TABLE
021406  010200                                                                MOV     R2,R0
021410  104442                                                                TRAP    C$GPHRD
021412                                    BNCOMPLETE NXTTAB           ;IGNORE IF NO TABLE RETURNED
021412  103110                                                                BCC     NXTTAB
                                        ;
                                        ;SEE IF A CONTROLLER TABLE ALREADY EXISTS FOR CONTROLLER IN P-TABLE
                                        ;
021414  013703  002156                    MOV CTABS,R3                ;GET ADDRESS OF CONTROLLER TABLES
021420  005713                  INIT5:    TST (R3)                    ;CHECK IF ANY MORE TABLES
021422  001416                            BEQ NEWTAB                  ;BUILD NEW TABLE IF FOUND ZERO WORD
021424  021013                            CMP (R0),(R3)               ;CHECK IF SAME UNIBUS ADDRESS
021426                                            ASSUME C.UADR EQ 0
021426                                            ASSUME MO.UBA EQ 0
021426  001463                            BEQ SAMTAB                  ;CHECK TABLE IF ALREADY EXISTS
021430  016301  000004                    MOV C.VEC(R3),R1            ;GET VECTOR FROM EXISTING CONTROLLER TABLE
021434  042701  177000                    BIC #^C<CT.VEC>,R1
021440  026001  000002                    CMP MO.VEC(R0),R1           ;SEE IF DIFFERENT VECTOR
021444  001002                            BNE 1$
021446  000137  022076                    JMP SAMVEC                  ;ERROR, CAN'T HAVE TWO UDA'S WITH SAME VECTOR
021452  062703  000054            1$:     ADD #C.SIZE,R3              ;MOVE TO NEXT TABLE
021456  000760                            BR INIT5
```

INITIALIZE SECTION

```
                                        ;BUILD A CONTROLLER TABLE
                                        ;
021460  012703  000010        NEWTAB: MOV     #8.,R3          ;R3 IS A COUNTER
021464  012704  003436                MOV     #IPADRS,R4      ;R4 -> IP ADDRESSES
021470  005714                1$:     TST     (R4)            ; FOUND AN OPEN ENTRY?
021472  001404                        BEQ     2$              ; IF SO, GO FILL ENTRY
021474  005724                        TST     (R4)+           ; NEXT ENTRY
021476  005303                        DEC     R3              ; SEARCH THROUGH ENTIRE TABLE?
021500  001373                        BNE     1$              ; IF NOT, BRANCH
021502  000401                        BR      3$              ; ELSE, TABLE FULL
021504  011014                2$:     MOV     (R0),(R4)       ; STORE ENTRY INTO TABLE
021506  012701  000026        3$:     MOV #C.SIZE/2,R1        ;GET WORDS IN CONTROLLER TABLE
021512  004737  012522                CALL ALOCM              ;ALLOCATE SPACE FOR IT
021516  011021                        MOV (R0),(R1)+          ;STORE UNIBUS ADDRESS
021520  010221                        MOV R2,(R1)+            ;UNIT NUMBER
021522  016004  000004                MOV H0.BRL(R0),R4       ;GET BR LEVEL
021526  000304                        SWAB R4                 ;SWAP TO HIGH BYTE
021530  006104                        ROL R4                  ;SHIFT ONE MORE TO LEFT
021532  056004  000002                BIS H0.VEC(R0),R4       ;ADD VECTOR ADDRESS
021536  010421                        MOV R4,(R1)+            ;  TO TABLE
021540  016021  000006                MOV H0.BST(R0),(R1)+
021544  012721  004037                MOV #4037,(R1)+         ;PUT [JSR R0,UDASRV]
021550  012721  017156                MOV #UDASRV,(R1)+       ;  INTO TABLE
021554  012703  000020                MOV #16.,R3             ;CLEAR POINTERS TO DRIVE TABLES,
                                                            ; TIMEOUT COUNTER, FLAGS, REF. NUMBER
                                        ;
021560  005021        INIT7:  CLR (R1)+
021562  005303                DEC R3
021564  001375                BNE INIT7               ;LOOP TIL ALL CLEARED
021566  005237  002160        INC CTRLRS              ;COUNT THE CONTROLLER
021572  005011                CLR (R1)                ;CLEAR TABLE END MARKER
021574  000417                BR NXTTAB               ;NOW GO TO NEXT P TABLE
```

```
                                        ;SHOULD BE SAME CONTROLLER, CHECK THAT OTHER PARAMETERS MATCH
                                        ;
021576  016004  000004          SAMTAB: MOV MO.BRL(R0),R4       ;GET BR LEVEL FROM P TABLE
021602  000304                          SWAB R4                 ;SWAP TO HIGH BYTE
021604  006104                          ROL R4                  ;SHIFT ONE MORE TO LEFT
021606  056004  000002                  BIS MO.VEC(R0),R4        ;ADD VECTOR ADDRESS
021612  020463  000004                  CMP R4,C.VEC(R3)         ;COMPARE WITH CONTROLLER TABLE
021616  001004                          BNE 1$
021620  026063  000006  000006         CMP MO.BST(R0),C.BST(R3) ;COMPARE BURST RATES
021626  001402                          BEQ NXTTAB
021630  000137  022026          1$:     JMP CTABER               ;FATAL ERROR IF NOT SAME
                                        ;
                                        ;GET NEXT P TABLE
                                        ;
021634  005202                  NXTTAB: INC R2                   ;INCREMENT LOGICAL UNIT NUMBER
021636  023702  002012                  CMP L$UNIT,R2            ;CHECK IF GOT ALL TABLES
021642  003261                          BGT INIT4                ;IF NOT, GO BACK FOR NEXT
                                        ;
021644  012701  000001                  MOV #1,R1                ;ALLOCATE SPACE FOR ZERO END WORD
021650  004737  012522                  CALL ALOCM               ;AFTER CONTROLLER TABLES
```

```
                                        ;NOW BUILD DRIVE TABLES

  021654  005002                                CLR R2                                  ;LOGICAL UNIT NUMBER IN R2
  021656                                INIT8.  GPHARD R2,R0                    ;GET POINTER TO A P-TABLE
  021656  010200                                                                                        MOV     R2,R0
  021660  104442                                                                                        TRAP    C$GPHRD
  021662                                        BNCOMPLETE INIT14                       ;IF NOT AVAILABLE, GO GET NEXT
  021662  103040                                                                                        BCC     INIT14

                                        ;FIND CONTROLLER TABLE

  021664  013703  002156                        MOV CTABS,R3                            ;GET ADDRESS OF CONTROLLER TABLES
  021670  021013                        INIT10: CMP (R0),(R3)                   ;CHECK IF SAME UNIBUS ADDRESS
  021672  001403                                BEQ INIT11                              ;BRANCH IF TABLE FOUND
  021674  062703  000054                        ADD #C.SIZE,R3                          ;MOVE TO NEXT TABLE
  021700  000773                                BR INIT10
```

```
                                        ;BUILD DRIVE TABLE
                                        
021702  012701  000015          INIT11: MOV #D.SIZE/2,R1        ;GET SIZE OF DRIVE TABLE
021706  004737  012522                  CALL ALOCM                      ;ALLOCATE SPACE FROM FREE MEMORY
                                ;       R0 POINTS TO P-TABLE
                                ;       R1 POINTS TO DRIVE TABLE
                                ;       R3 POINTS TO CONTROLLER TABLE
                                ;       R2 IS UNIT NUMBER
021712  010337  003244                  MOV R3,TEMP                     ;SAVE CONTROLLER TABLE ADDRESS
                                                                        ;IN CASE AN ERROR IS DETECTED
021716  062703  000020                  ADD #C.DR0,R3                   ;BUILD POINTER TO C.DR ENTRY IN CONTROLLER TABLE
021722  012704  000010                  MOV #8.,R4                      ;GET MAX COUNT OF DRIVES ON ONE CONTROLLER
021726  005713                  INIT12: TST (R3)                ;CHECK IF ENTRY CONTAINS POINTER TO DRIVE TABLE
021730  001411                          BEQ INIT13
021732  026033  000010                  CMP H0.LDR(R0),@(R3)+           ;CHECK DRIVE NUMBER IN DRIVE TABLE
021736  001002                          BNE 1$
021740  000137  022042                  JMP MLDRER                      ;IF SAME, TWO P TABLES POINT TO SAME DRIVE
021744  005304                  1$:     DEC R4                          ;COUNT DRIVES
021746  001367                          BNE INIT12                      ;IF EIGHT DRIVE TABLES EXIST,
021750  000137  022060                  JMP TOOMER                      ;  THEN REPORT ERROR
021754  010113                  INIT13: MOV R1,(R3)             ;LOAD DRIVE TABLE POINTER
021756  016021  000010                  MOV H0.LDR(R0),(R1)+            ;LOAD DRIVE NUMBER
021762  010221                          MOV R2,(R1)+                    ;LOAD UNIT NUMBER
```

```
                                        ;GO TO NEXT DRIVE TABLE
                                        
021764  005202                  INIT14: INC R2                  ;INCREMENT LOGICAL UNIT NUMBER
021766  023702  002012                  CMP L$UNIT,R2                   ;CHECK IF GOT ALL TABLES
021772  003331                          BGT INIT8                       ;IF NOT, GET NEXT TABLE
                                        
                                        ;SAVE CURRENT PARAMETERS TO FREE MEMORY
                                        
021774  013737  002146  002152  INIT15: MOV FFREE,FMEM           ;SAVE START ADDRESS
022002  013737  002150  002154          MOV FSIZE,FMEMS                 ;SAVE SIZE
                                        
022010                          INITXX: SETPRI  #PRI00                  ; SET RUNNING PRIORITY TO ZERO
022010  012700  000000                                                  MOV     #PRI00,R0
022014  104441                                                          TRAP    C$SPRI
022016  004737  013022                  CALL CLOSEF                     ;MAKE SURE DATA FILE IS CLOSED
022022                                  EXIT INIT
022022  104432                                                          TRAP    C$EXIT
022024  000066                                                          .WORD   L10035 .
```

```
                                ;DIFFERENT VECTORS, BR LEVELS OR BURST RATES FOR ONE CONTROLLER
022026  010305                  CTABER: MOV R3,R5                       ;GET CONTROLLER ADDRESS
022030                                  ERRSF 1,,ERR001
022030  104454                                                                  TRAP    C$ERSF
022032  000001                                                                  .WORD   1
022034  000000                                                                  .WORD   0
022036  011604                                                                  .WORD   ERR001
022040                                  DOCLN
022040  104444                                                                  TRAP    C$DCLN

                                ;TWO P TABLES FOR SAME DRIVE
022042  013705  003244          MLDRER: MOV TEMP,R5             ;GET CONTROLLER ADDRESS
022046                                  ERRSF 2,,ERR002
022046  104454                                                                  TRAP    C$ERSF
022050  000002                                                                  .WORD   2
022052  000000                                                                  .WORD   0
022054  011622                                                                  .WORD   ERR002
022056                                  DOCLN
022056  104444                                                                  TRAP    C$DCLN

                                ;MORE THAN EIGHT DRIVES SELECTED ON ONE CONTROLLER

022060  013705  003244          TOOMER: MOV TEMP,R5             ;GET CONTROLLER ADDRESS
022064                                  ERRSF 3,,ERR003
022064  104454                                                                  TRAP    C$ERSF
022066  000003                                                                  .WORD   3
022070  000000                                                                  .WORD   0
022072  011640                                                                  .WORD   ERR003
022074                                  DOCLN
022074  104444                                                                  TRAP    C$DCLN

                                ;TWO UDA'S USE THE SAME VECTOR

022076  010305                  SAMVEC: MOV R3,R5                       ;GET CONTROLLER ADDRESS
022100                                  ERRSF 8,ERR008
022100  104454                                                                  TRAP    C$ERSF
022102  000010                                                                  .WORD   8
022104  011670                                                                  .WORD   ERR008
022106  000000                                                                  .WORD   0
022110                                  DOCLN
022110  104444                                                                  TRAP    C$DCLN

022112                                  ENDINIT
022112                                                                  L10035:
022112  104411                                                                  TRAP    C$INIT
```

```
                        .SBTTL  AUTODROP SECTION
                        ;++
                        ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
                        ; THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
                        ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                        ; DROPPED FROM TESTING.
                        ;
022114                          BGNAUTO
022114                                                          L$AUTO::
022114                          ENDAUTO
022114                                                          L10036:
022114  104461                                                          TRAP    C$AUTO
```

```
                                .SBTTL  CLEANUP CODING SECTION

                                ;**
                                ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
                                ;

022116                                  BGNCLN
022116                                                                  L$CLEAN::

022116  004737  013022                  CALL CLOSEF                     ;CLOSE DATA FILE
022122  004737  012564                  CALL    RESET                   ;RESET ALL UDAS

022126                                  ENDCLN
022126                                                                  L10037:
022126  104412                                                                  TRAP    C$CLEAN

022130                                  ENDMOD
```

```
                                .SBTTL  TEST 1: DUP PROGRAM DRIVER
                                
022130                                  BGNMOD
                                
022130                                  BGNTST
022130                                                                  T1::
022130                                  PNTX WNSTRT             ;PRINT WARNING MESSAGE
022130  004137  016246                                                  JSR R1,LPNTX
022134  004532                                                        .WORD WNSTRT
022136  000000                                                        .WORD PNT.CT
022140                                  MANUAL                  ;SEE IF MANUAL INTERVENTION ALLOWED
022140  104450                                                        TRAP    C$MANI
022142                                  BNCOMPLETE T1MODE       ;IF NOT, JUST RUN THE PROGRAM
022142  103020                                                        BCC     T1MODE
022144  005037  003244                  CLR TEMP                ;CLEAR WORD FOR ANSWER
022150                                  GMANIL WNQUES,TEMP,1,YES        ;ASK IF STILL WANT TO RUN
022150  104443                                                        TRAP    C$GMAN
022152  000404                                                        BR      100008
022154  003244                                                        .WORD   TEMP
022156  000130                                                        .WORD   T$CODE
022160  003622                                                        .WORD   WNQUES
022162  000001                                                        .WORD   1
022164                                                          100008:
022164  005737  003244                  TST TEMP                        ;LOOK AT ANSWER
022170  001417                          BEQ T1QUIT              ;IF NO, QUIT NOW
022172  005737  003306                  TST DATEO       ;SEE IF ALREADY ASKED FOR DATE
022176  001002                          BNE T1MODE
022200  004737  020*34                  CALL DATE       ;IF NOT, GET IT NOW
                                
022204  032737  000003  003206  T1MODE: BIT #SO.FMT,MODE
022212  001164                          BNE T1FMT
022214                                  MANUAL
022214  104450                                                        TRAP    C$MANI
022216                                  BCOMPLETE T1GO
022216  103406                                                        BCS     T1GO
022220                                  ERRSF 10.,ERR010
022220  104454                                                        TRAP    C$ERSF
022222  000012                                                        .WORD   10
022224  000000                                                        .WORD   0
022226  011720                                                        .WORD   ERR010
022230                          T1QUIT: EXIT TST
022230  104432                                                        TRAP    C$EXIT
022232  000362                                                        .WORD   L10040
022234  032737  000010  003206  T1GO:   BIT #SO.STR,MODE
022242  001435                          BEQ T1CNS
022244  023727  002012  000001          CMP L$UNIT,#1
022252  001406                          BEQ T1RST
022254                                  ERRSF 9.,ERR009
022254  104454                                                        TRAP    C$ERSF
022256  000011                                                        .WORD   9
022260  000000                                                        .WORD   0
022262  011706                                                        .WORD   ERR009
022264                                  EXIT TST
022264  104432                                                        TRAP    C$EXIT
022266  000326                                                        .WORD   L10040
                                
022270                          T1RST:  PNTF FILNAM
```

```
022270 004137 016226                                              JSR R1,LPNTF
022274 011522                                                      .WORD FILNAM
022276 000000                                                     .WORD PNT.CT
022300                    GMANID FILNAQ,FNAME,A,-1,1,10.,NO       ;GET FILE NAME
022300 104443                                                     TRAP    C$GMAN
022302 000406                                                     BR      10001$
022304 003232                                                     .WORD   FNAME
022306 000142                                                     .WORD   T$CODE
022310 003566                                                     .WORD   FILNAQ
022312 177777                                                     .WORD    1
022314 000001                                                     .WORD   T$LOLIM
022316 000012                                                     .WORD   T$HILIM
022320                                                    10001$:
022320                    OPEN #FNAME
022320 012700 003232                                              MOV     #FNAME,R0
022324 104434                                                     TRAP    C$OPEN
022326 012737 177777 002174  MOV #-1,FILOPN  ;MARK FLAG AS FILE OPEN
022334 000513                BR T1FMT
022336 013705 002156  T1CNS:  MOV CTABS,R5
022342 010504         T1SER1: MOV R5,R4
022344 062704 000020          ADD #C.DRO,R4
022350 012703 000010          MOV #8.,R3
022354 011402         T1SER2: MOV (R4),R2      ;GET DRIVE TABLE POINTER
022356 001476                 BEQ T1SERN
022360                        PNTF SERNUM,D.UNIT(R2),(R5),(R2)
022360 011246                                                     MOV (R2),-(SP)
022362 011546                                                     MOV (R5),-(SP)
022364 016246 000002                                              MOV D.UNIT(R2),-(SP)
022370 004137 016226                                              JSR R1,LPNTF
022374 004235                                                     .WORD SERNUM
022376 000006                                                     .WORD PNT.CT
022400                             ASSUME C.UADR EQ 0
022400                             ASSUME D.DRV EQ 0
022400                T1SER3: GMANID SERNQ,TEMP,A,-1,1,20.,NO ;GET SERIAL NUMBER
022400 104443                                                     TRAP    C$GMAN
022402 000406                                                     BR      10002$
022404 003244                                                     .WORD   TEMP
022406 000142                                                     .WORD   T$CODE
022410 003620                                                     .WORD   SERNQ
022412 177777                                                     .WORD    1
022414 000001                                                     .WORD   T$LOLIM
022416 000024                                                     .WORD   T$HILIM
022420                                                    10002$:
022420 012701 003244          MOV #TEMP,R1
022424 005000                 CLR R0
022426 105711         T1SER4: TSTB (R1)
022430 001410                 BEQ T1SER5
022432 005200                 INC R0
022434 121127 000060          CMPB (R1),#'0
022440 103420                 BLO T1SER7
022442 122127 000071          CMPB (R1)+,#'9
022446 101767                 BLOS T1SER4
022450 000414                 BR T1SER7
022452 020027 000024  T1SER5: CMP R0,#20.
022456 103424                 BLO T1SER8
022460 012701 003244          MOV #TEMP,R1
022464 012700 003322          MOV #HIGHEST,R0
```

```
                                T1SER6: TSTB (R0)
                                        BEQ T1SER8
                                        CMPB (R1)+,(R0)+
                                        BEQ T1SER6
                                        BLO T1SER8
                                T1SER7: PRINTF #SERNX,#HIGHEST
022502  012746  003322                                                    MOV     #HIGHEST,-(SP
022506  012746  011413                                                    MOV     #SERNX,-(SP)
022512  012746  000002                                                    MOV     #2,-(SP)
022516  010600                                                            MOV     SP,R0
022520  104417                                                            TRAP    C$PNTF
022522  062706  000006                                                    ADD     #6,SP
                                        BR T1SER3
                                T1SER8: ADD #D.SERN,R2  ;PUT ANSWER INTO DRIVE TABLE
                                        MOV #TEMP,R1
                                T1SER9: MOVB (R1)+,(R2)+
                                        BNE T1SER9
                                        DEC R3
                                        BEQ T1SERN
                                        TST (R4)+
                                        BR T1SER2
                                T1SERN: ADD #C.SIZE,R5
                                        TST (R5)+
                                        BNE T1SER1
                        002156  002162  T1FMT:  MOV CTABS,TSTTAB              ;GET FIRST TABLE ADDRESS
                                               MOV CTRLRS,R1                ;RUN DM PROGRAM ON ALL CONTROLLERS
                                               CALL RUNDM                   ;  RUN ALL CONTROLLERS OF ONE TYPE AT ONCE
                                               BEQ 6$
                                               CALL RESPDM
                                        6$:    EXIT TST
022610  104432                                                                  TRAP    C$EXIT
022612  000002                                                                  .WORD   L10040-.
                                        ENDTST
022614                                                                  L10040:
022614  104401                                                                  TRAP    C$ETST
                                        ENDMOD
```

```
                                .SBTTL  HARDWARE PARAMETER CODING SECTION
                        ;
022616                          BGNMOD
                        ;
                        ;++
                        ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                        ; THAT ARE USED BY THE SUPERVISOR TO BUILD P TABLES.  THE
                        ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                        ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                        ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                        ; WITH THE OPERATOR.
                        ;-
                        
022616                          BGNHRD
022616  000027                                                  .WORD L10041-L$HARD/2
022620                                                  L$HARD::
                        
                        ;FORMAT OF HARDWARE P-TABLE IS AS FOLLOWS:
                        
022620                  TABLE                                   ;START A TEBLE DEFINITION
                        
022620                  ITEM HO.UBA     2                       ; UNIBUS ADDRESS
022620                  ITEM HO.VEC     2                       ; UDA VECTOR
022620                  ITEM HO.BRL     2                       ; BR LEVEL
022620                  ITEM HO.BST     2                       ; BURST RATE
022620                  ITEM HO.LDR     2                       ; DRIVE NUMBER
022620                  END
```

```
  1 022620                                GPRMA   M.UBA,MO.UBA,0,160000,177774,YES        ;BUS ADDRESS
    022620  000031                                                                         .WORD   T$CODE
    022622  022676                                                                         .WORD   M.UBA
    022624  160000                                                                         .WORD   T$LOLIM
    022626  177774                                                                         .WORD   T$HILIM
  2 022630                                GPRMA   M.VEC,MO.VEC,0,4,774,YES         ; VECTOR
    022630  001031                                                                         .WORD   T$CODE
    022632  022724                                                                         .WORD   M.VEC
    022634  000004                                                                         .WORD   T$LOLIM
    022636  000774                                                                         .WORD   T$HILIM
  3 022640                                GPRMD   M.BRL,MO.BRL,D, 1,4.,7.,YES      ; BR LEVEL
    022640  002052                                                                         .WORD   T$CODE
    022642  022733                                                                         .WORD   M.BRL
    022644  177777                                                                         .WORD   -1
    022646  000004                                                                         .WORD   T$LOLIM
    022650  000007                                                                         .WORD   T$HILIM
  4 022652                                GPRMD   M.BST,MO.BST,D, 1,0.,63.,YES     ; BURST RATE
    022652  003052                                                                         .WORD   T$CODE
    022654  022744                                                                         .WORD   M.BST
    022656  177777                                                                         .WORD   -1
    022660  000000                                                                         .WORD   T$LOLIM
    022662  000077                                                                         .WORD   T$HILIM
  5 022664                                GPRMD   M.LDR,MO.LDR,D, 1,0.,255.,YES    ; DRIVE SELECT NUMBER
    022664  004052                                                                         .WORD   T$CODE
    022666  022766                                                                         .WORD   M.LDR
    022670  177777                                                                         .WORD   -1
    022672  000000                                                                         .WORD   T$LOLIM
    022674  000377                                                                         .WORD   T$HILIM
  6 022676                                ENDMRD                                          .EVEN
    022676                                                                      L10041:
  7
  8 022676      125     116     111  M.UBA:  .ASCIZ  \UNIBUS ADDRESS OF UDA
  9 022724      126     105     103  M.VEC:  .ASCIZ  \VECTOR\
 10 022733      102     122     040  M.BRL:  .ASCIZ  \BR LEVEL\
 11 022744      125     116     111  M.BST:  .ASCIZ  \UNIBUS BURST RATE\
 12 022766      104     122     111  M.LDR:  .ASCIZ  \DRIVE NUMBER\
 13                                         .EVEN
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION
                                ;**
                                ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;

023004                                  BGNSFT
023004  000022                                                            .WORD L10042-L$SOFT/2
023006                                                                  L$SOFT::

                                ;FORMAT OF SOFTWARE P-TABLE IS AS FOLLOWS:

023006                          TABLE                                   ;START A TABLE DEFINITION

023006                          ITEM SO.BIT     2                       ;YES/NO ANSWERS
        000001                          SO.FM1 = BIT0                   ;  REFORMAT MODE
        000002                          SO.FM2 = BIT1                   ;  (AGAIN)
        000003                            SO.FMT = SO.FM1+SO.FM2
        000004                          SO.CNS = BIT2                   ;  RECONSTRUCT MODE
        000010                          SO.STR = BIT3                   ;  RESTORE MODE

023006                          END
```

```
  1 023006                                  GPRML S.FMT,SO.BIT,SO.FM1,YES   ;REFORMAT?
    023006  000130                                                            .WORD   T$CODE
    023010  023223                                                            .WORD   S.FMT
    023012  000001                                                            .WORD   SO.FM1
  2 023014                                  xFERT SWEND
    023014  017024                                                            .WORD   T$CODE
  3 023016                                  GPRML S.NRF,SO.BIT,SO.FM2,YES   ;AGAIN   REFORMAT?
    023016  000130                                                            .WORD   T$CODE
    023020  023052                                                            .WORD   S.NRF
    023022  000002                                                            .WORD   SO.FM2
  4 023024                                  xFERT SWEND
    023024  013024                                                            .WORD   T$CODE
  5 023026                                  GPRML S.CNS,SO.BIT,SO.CNS,YES   ;RECONSTRUCT
    023026  000130                                                            .WORD   T$CODE
    023030  023302                                                            .WORD   S.CNS
    023032  000004                                                            .WORD   SO.CNS
  6 023034                                  xFERT SWEND
    023034  007024                                                            .WORD   T$CODE
  7 023036                                  GPRML S.RST,SO.BIT,SO.STR,YES   ;RESTORE?
    023036  000130                                                            .WORD   T$CODE
    023040  023345                                                            .WORD   S.RST
    023042  000010                                                            .WORD   SO.STR
  8 023044                                  xFERT SWEND
    023044  003024                                                            .WORD   T$CODE
  9 023046                                  DISPLAY S.NOF   ;WARNING
    023046  000003                                                            .WORD   T$CODE
    023050  023466                                                            .WORD   S.NOF
 10 023052                          SWEND:  ENDSFT
                                                                               .EVEN
    023052                                                          L10042:
 11
 12 023052      015     012         S.NRF:  .BYTE 15,12
 13 023054      116     117     124         .ASCII\NOT USING EXISTING INFORMATION WILL DESTROY THE FACTORY BAD SECTOR\
 14 023156      015     012                 .BYTE 15,12
 15 023160      111     116     106         .ASCII\INFORMATION ON THE DISKS.\
 16 023211      015     012                 .BYTE 15,12
 17 023213      101     107     101         .ASCII\AGAIN - \
 18 023223      122     105     106 S.FMT:  .ASCIZ\REFORMAT USING EXISTING BAD SECTOR INFORMATION\
 19 023302      122     105     103 S.CNS:  .ASCIZ\RECONSTRUCT BAD SECTOR INFORMATION\
 20 023345      104     117     040 S.RST:  .ASCII\DO YOU HAVE A FILE ON THE SYSTEM LOAD DEVICE\
 21 023421      015     012                 .BYTE 15,12
 22 023423      040     103     117         .ASCIZ\ CONTAINING BAD SECTOR INFORMATION\
 23 023466      131     117     125 S.NOF:  .ASCIZ\YOU CANNOT PROCEED WITHOUT SUCH A FILE.\
 24 023536      122     105     123         .ASCIZ\RESTART PROGRAM AND SELECT TO REFORMAT OR RECONSTRUCT DISK.\
 25 023632      000                         .BYTE 0
 26                                         .EVEN
 27
 28                                         .DSABL AMA
 29 000000                                  .PSECT END
```

```
                                        .SBTTL  PATCH AREA
                                ;
000000                          $PATCH::
                                        .REPT 40.
000050                                  .WORD 0
                                        .ENDR
                                ;
000120                                  LASTAD
                                                                        .EVEN
000120  000142                                                         .WORD T$FREE
000122  000007                                                         .WORD T$SIZE
000124                          L$LAST::
                                ;
000124                                  ENDMOD
```

PATCH AREA

```
      1 000124                                  BGNSETUP        1
      2
      3 000124                                  BGNPTAB
        000124  000000                                                          .WORD   0
        000126  000005                                                          .WORD   L10045 ./2-1
        000130                                                          L10043:
      4
      5 000130  172150                          .WORD   172150          ; UNIBUS ADDRESS
      6 000132  000154                          .WORD   154             ; VECTOR ADDRESS
      7 000134  000005                          .WORD   5.              ; BR LEVEL
      8 000136  000077                          .WORD   63.             ; UNIBUS BURST RATE
      9 000140  000000                          .WORD   0.              ; LOGICAL DRIVE NUMBER
     10
     11 000142                                  ENDPTAB
        000142                                                          L10045:
     12
     13 000142                                  ENDSETUP
     14
     15
     16
     17
     18
     19
     20
     21         000001                  .END
Errors detected:  0

*** Assembler statistics


Work  file  reads: 276
Work  file writes: 268
Size of work file: 28936 Words  ( 114 Pages)
Size of core pool: 17152 Words  ( 67 Pages)
Operating  system: RT-11 (Under RSTS/E)

Elapsed time: 00:02:44.47
ZUDEDO.OBJ,ZUDEDO/C=[20,0]SVC34R.MLB/P:1,ZUDEDO.DOC,ZUDEDO
```

Cross reference table (CREF V05.00)

```
$PATCH 135 30
ADR      34-10#
ALOCM    60-16#   61 14    122-13   123-20   125 4
ASSEMB   30-8     30-8
BAS      54-14#   87-5      87-5     87-5     88 5     88 5
BASL2    54 12#   88-5
BASL3    54 13#
BASLN    54-16#   87-5      88 5
BASNO    54-11#   87-5      88-5
BIT0     34-10#  133-19
BIT00    34-10    34-10#
BIT01    34-10    34-10#
BIT02    34-10    34-10#
BIT03    34-10    34-10#
BIT04    34-10    34-10#
BIT05    34-10    34-10#
BIT06    34 10    34-10#
BIT07    34-10    34-10#
BIT08    34-10    34-10#
BIT09    34-10    34-10#
BIT1     34-10#   45-23    133-20
BIT10    34-10#
BIT11    34-10#
BIT12    34-10#
BIT13    34-10#
BIT14    34-10#
BIT15    34-10#   45-15     46-12    63-27    66-20    74-28    79-15    93-29
BIT2     34-10#   45-24    133-22
BIT3     34-10#   45-25    133-23
BIT4     34-10#   45-27
BIT5     34-10#   45-30
BIT6     34-10#   45-31
BIT7     34-10#   45-33
BIT8     34-10#
BIT9     34-10#
BLDC0   101-22    101-24#
BLDC1   101-26#   101-28
BLDCMD   65-49     68-14     68 44    99-2     101-15#
BOE      34-10#
C$AU     30-8#
C$AUTC   30-8#    128-12
C$BRK    30-8#     62-12     65-8     104-21   111 12   114-27
C$BSEG   30-8#
C$BSUB   30-8#
C$CEFG   30-8#
C$CLCK   30-8#    120-41    120-43
C$CLEA   30-8#    129-13
C$CLOS   30-8#     64-12     77-19
C$CLP1   30-8#
C$CVEC   30-8#     62 22    111-30
C$DCLN   30-8#     59-8      62-30    120 18   127 4    127-9    127-15   127-21
C$DODU   30-8#
C$DRPT   30-6#
C$DU     30-8#
C$EDIT   30-8#     30 34
C$ERDF   30-8#     62 28     65-24    66-36    67 21    68-8     72-22    74 32   100 3    104-29   105-5    109-35   109 54   110-11
```

Cross reference table (CREF V05.00)

```
        111-33   113 44   114-56   115 5
C$ERHR   30-80
C$ERRO   30-80
C$ERSF   30-80    59-7     127-3    127-8    127-14   127-20   130-21   130-27
C$ERSO   30-80
C$ESCA   30-80
C$ESEG   30-80
C$ESUB   30-80
C$ETST   30-80   130 83
C$EXIT   30 80   126 14   130 22   130-28   130-82
C$GETB   30-80    77 24
C$GETW   30-80
C$GMAN   30-80   118-1    130-10   130 31   130-44
C$GPHR   30-80   121 15   124 4
C$GPLO   30-80
C$GPRI   30-80
C$INIT   30 80   127-23
C$INLP   30-80
C$MANI   30-80   130-7    130-19
C$MEM    30-80   120-32
C$MSG    30-80    57-16    57-20    57-24    57 28    57-32    57-36    57-40    57-44    57-48    57-57    57-62    57 76    57-80
         57-84    57-88    57-92    57-97    57-101   57-105   57-109   57-113   57 117   57-121
C$OPEN   30-80    77-20    130-32
C$PNTB   30-80    95 14
C$PNTF   30-80    95-12    130-64
C$PNTS   30 80    95-18
C$PNTX   30 80    95 16
C$QIO    30-80
C$RDBU   30-80
C$REFG   30 80   120-10   120-12   120 14   120 16
C$RESE   30-80    30-80
C$REVI   30-80    30 34
C$RFLA   30-80
C$RPT    30-80
C$SEFG   30-80
C$SPRI   30-80   126-12
C$SVEC   30 80    62-11    98 20    111-27   120-52
C$TPRI   30-80
C.BST    45-190  110-17   123-9
C.DR0    45 350   66-15    79 12    125-11   130-37
C.DR1    45-360
C.DR2    45-370
C.DR3    45-380
C.DR4    45-390
C.DR5    45-400
C.DR6    45-410
C.DR     45-420
C.FLG    45-220   63-23•   65-10    65-13    65-15    65-38    65-43    65-47•   66-12•   67-6     67-26    68 35•   68 39•   68-40
         68-47•   68 50•   68-62•   68-63    68 69•   68-70    99-12•   99-13•   102-21•  104-17   105-1•
C.JAD    45-210
C.JSR    45-200   98 19
C.PRI    45-460   68-65    68 67    68-72•   68-73•
C.REF    45-470   67-19    102-16•  102-17
C.RING   45-340  57-137    61 15•   65-9     68-29    101-16   102-15   109 20   110-3    110-22   111 20
C.SIZE   45-490   63-32    66-5     121-32   122-12   124-12   130-74
C.TO     45-430   65-36    68-55    104-14   104-27   114-22   114 33
```

```
C.TOH    45-44#   65-33    104-24   114-30
C.TOT    45-45#   65 29     68 53    98-14*   98-15*
C.UADR   45 12#   80-16    111-25   121-24   130-42
C.UNIT   45 13#   63 24     63-25    65-12
C.VEC    45-16#   98-16    111-14   121-27   123 7
CALR4    84 6     87-5#
CALR5    84-7     88-5#
CALR6    84-8     89-4#
CALR7    84-9     90 4#
CALR8    84-10    91 4#
CALRE    84-3     84 4      84 5     86 4#
CF.576   41-3[illegible]
CF.ATN   41-34#
CF.MSC   41-35#
CF.OTH   41-36#
CF.SHD   41-38#
CF.TMS   41 37#
CLOSEF   64-10#   126-13   129-10
CLRBFL  103-22#  103-24
CLRBUF   68-16    68-46    103-16#
CON.A    82-12#   85-21
CON.A1   82-13#   82-15
CON.A2   82-17    82-19#
CON.D    82-23#   85-21
CON.H    82-29#   85 21
CON.N    83-9#    85-21
CON.N1   83-10#   83-12
CON.O    83-3#    85-21
CON.QU   82-3#    82-7      85 21
CON.QX   82-5     82-8#
CON.R    83 17#   85-21
CON.R1   83-19    83-23#
CON.S    83-29#   85-21
CON.S1   83-30#   83-32
CPNT     78-18    78-21     80 21    80-23    82-6     82-13    83-10    83-30    87-7     88-7     94 13    94-22    95-5#    117 27
        117-32   117-35
CPNTx    95-13    95-15     95-17    95-19#
CR       34-12#   78-21     80-21    80-23    83-10    87-7     88-7     95-8
CT.AVL   45-15#   63-27
CT.BRL   45-18#
CT.CMD   45-24#   65-15     99-12    102-21
CT.MSG   45-25#   65 13     68-39    68-62    104-17   105-1    107-19
CT.REQ   45-27#   67-6      67-26    68-35    68-40    68-47    68-50    99-12
CT.RN    45-23#   65-10     99-13
CT.STA   45-30#   65-38     65-46    68-39    68-62
CT.TM1   45-31#   65-42     65-43    68-39    68-70
CT.TM2   45-33#   68-39     68 63    68-69
CT.UNT   45-14#
CT.VEC   45-17#   98-17    111-15   121-28
CTABER  123-11   127 2#
CTABS    50-12#   121 3*    121-4*   121-20   124 9    130-35   130-77
CTRLRS   50-13#   121-5*    122-29*  130-78
D.DRV    46-9#    73-36     80-17    130-43
D.SERN   46-13#   72-17     130-66
D.SIZE   46-15#   125-3
D.UNIT   46 10#   66 19     66-25*   79-14    80-15    130-41
```

Cross reference table (CREF V05.00)

```
DAS1    118-8    118-130
DATE    118 10   118-32   130-15
DATEI    51-20   118-1    118-2    118-44
DATEO    51 40    72-26   118-33   130 13
DATEQ    53-30   118-1
DATEX    56 120  118-31
DAY     118 50
DAY1    118-460  118-57
DAY2    118-47   118-580
DAYS     51 190  118-42
DERR    118-4    118-6    118-10   118-12   118 14   118-310  118-59   118-61   118-64   118-84   118 A8   118 90
DF.ACT   44-210   68-60
DF.LCL   44 240
DF.NES   44-220
DF.SA    44-260
DFPTBL   32-100
DIAG     38 330  101 23
DIAGMC   30-8     30-8
DIVIDE   73-38    93-39    94-7     97-160   117-19   117-21   117-23
DMFRST   44-470
DMMAIN   44-460
DMOVRL   44-440
DMPROG   50-160   98-13    99-3     99-4     99-5     99-6
DMTMO    44-450   98-14
DMTRLN   44-430
DQCONT   73-28    73-610
DQNO     73-54    73-620
DQRFMT   73-26    73-530
DQRSTR   73-27    73-590
DQUEJP   73-21    73-220   73-29
DQUEST   69-4     73-160
DQUESZ   73-18    73-290
DQUEX    73-19    73-22    73-24    73-25    73-500   73-57    73-64
DQUNIT   73-23    73-330
DQUNL1   73-380   73-42
DQUNL2   73-440   73-48
DQYES    73-550   73-60
DSPSIZ   69-90
DT.AVL   46-120   66-20    66-25    79-15
DT.UNT   46-110
DU.DFL   44-330
DU.FTL   44-360
DU.INF   44-340
DU.QUE   44-320
DU.SPC   44-370   68-6
DU.TER   44-350
DU.TYP   44-390   68-4     68-19
DUP      38-340  101-20
E$END    30 80
E$LOAD   30-80    30-34
EF.BBR   41-270
EF.BBU   41-280
EF.CON   34-100  120-14
EF.LOG   41-290
EF.NEW   34-100
EF.PWR   34-100  120 16
```

Cross reference table (CREF V05.00)

```
EF.RES   34 100  120 12
EF.SEX   41-300
EF.STA   34-100  120-10
ERR001   57 140  127-3
ERR002   57-180  127-8
ERR003   57-220  127-14
ERR004   57-260   59-7
ERR008   57-300  127-20
ERR009   57 340  130-27
ERR010   57-380  130-21
ERR014   57-420  109-54
ERR020   57 460   62-28   111-33
ERR021   57-500  115-5
ERR022   57-590  114-36
ERR023   57-640  110-11
ERR024   57 780  109 35
ERR025   57 820  113-44
ERR030   57-860   65-24
ERR031   57-900   66-36
ERR032   57-940   68-8
ERR033   57-990   67-21
ERR034   57-1030 100-3
ERR036   57-1070 104-29
ERR037   57-1110 105-5
ERR100   57-1150  72-22
ERR101   57 1190  74-32
ERR23A   57-670   57-74
ERR23B   57 68    57-720
ERR23C   57 71    57-750
ERRC     81-36    81 43    85-90
ERRCHR   50 370   95-5     95 12    95 14    95-16    95 18
ERRD     81-45    85-210
ERRME1   54-80    81 41    83 23    86-4
ERRNL    54 40    95-10
ERRONE   54-50    95 7
ERRRSZ   83 18    84-110
ERRRTB   83-21    84-30    84 11
ERRTRM   69-7     76 140
EVL      34-100
F$AU     30-80
F$AUTO   30-80    128 10   128-12
F$BGN    30-80    30-26    33-16    34-3     57-14    57-18    57-22    57-26    57-30    57-34    57 38    57 42    57-46    57-50
         57 59    57-64    57-78    57-82    57-86    57-90    57-94    57-99    57-103   57-107   57-111   57-115   57-119   106-10
        107-18   116-5    118-104  119-3    119-10   120-8    126-14   128-10   129-8    129-15   130-3    130-5    130-22   130 28
        130-82   130-83   130-84   131-3    131-14   133-12   134-9    134 9    135-10   136-1    136-3    136 3    136-11   136-13
F$CLEA   30-80    129-8    129-13
F$DU     30 80
F$END    30-8     30-8     30 8     30-8     30-8     30-8     30-8     30-8     30-8     30-8     30-8     30 8     30-8     30-8
         30-8     30-8     30-80    30-26    33 16    34-3     57-16    57-20    57-24    57-28    57-32    57 36    57-40    57-44
         57-48    57 57    57 62    57-76    57-80    57-84    57-88    57-92    57-97    57-101   57-105   57 109   57-113   57 11
         57-121   106 14   107-2    116-9    118-104  119-3    126-14   127-23   128-12   129-13   129-15   130 3    130-5    130 5
        130-5    130-22   130-28   130 82   130-83   130-83   130-84   131 3    132-6    134-10   135-10   136 1    136-3    136 11
        136 15
F$HARD   30-80    131 14   132-6    134 2    134 4    134-6    134-8
F$MU     30-80    32-10    32-17
F$INIT   30 80    120-8    127 23
```

Cross reference table (CREF V05.00)

```
F$JMP    30 80   126 14   130-22   130 28   130 82
F$MOD    30 80    30-26    33-16    34 3    118-104  119 3    129 15   130 3    130 84   131 3    135 10
F$MSG    30 80    57-14    57-16    57-18    57 20    57-22    57 24    57-26    57 28    57-30    57 32    57-34    57 36    57 38
         57 40    57 42    57-44    57-46    57-48    57-50    57-57    57-59    57-62    57 64    57 76    57 78    57 80    57 82
         57 84    57 86    57 88    57-90    57-92    57-94    57-97    57 99    57-101   57 103   57 105   57 107   57 109   57 111
         57 113   57 115   57 117   57 119   57 121
F$PROT   30-80    119 10   119 16
F$PWR    30-80
F$RPT    30 80
F$SEG    30-80
F$SOFT   30-80    133-12   134-2    134-4    134 6    134-8    134 10
F$SRv    30-80    106-10   106-14   107-18   107-21   116 5    116-9
F$SUB    30-80
F$SW     30-80     33-10    33 14
F$TEST   30-80    130-5    130-83
FCTBUF   50 240    77 23    77-32
FCTNUM   50 250    63-36*   77-14    77-21*   77-28*
FDATA    50-230
FFREE    50-80     57-65    60-16    60-20*   120-32*  120-33   121-3    126 9
FILNAM   56 130   130-30
FILNAQ   53-40    130-31
FILOPN   50-200    64-10    64-13*   130-33*
FMEM     50-100   126-9*
FMEMS    50-110   126-10*
FMERR    59-70     60-18
FNAME    50 390    77-20    130-31   130 32
FSIZE    50-90     60-17*   120-53*  126-10
G$CNTO   30-80
G$DELM   30-80
G$DISP   30-80    134-9
G$EXCP   30-80
G$MILI   30-80
G$LOLI   30-80
G$NO     30-80    130-31   130-44
G$OFFS   30-80    118-1    130-10   130-31   130-44   132-1    132-2    132-3    132-4    132-5    134 1    134 3    134-5    134-7
G$OFSI   30-80    118-1    130-10   130-31   130-44   132-1    132-2    132-3    132-4    132-5    134-1    134 3    134 5    134-7
G$PRMA   30-80    132-1    132-2
G$PRMD   30-80    118 1    130-31   130-44   132-3    132-4    132-5
G$PRML   30-80    130-10   134-1    134-3    134-5    134-7
G$RADA   30-80    118-1    130-31   130-44
G$RADB   30-80
G$RADD   30-80    132-3    132-4    132-5
G$RADL   30 80    130-10   134-1    134-3    134-5    134-7
G$RADO   30 80    132-1    132-2
G$XFER   30-80    134-2    134-4    134-6    134-8
G$YES    30-80    118-1    130-10   132 1    132-2    132-3    132-4    132-5    134 1    134 3    134 5    134 7
GETCDN   92-16     92-18    92-28*
GETCNT   82 12     83-9     83-17    83-29    92 130   93-15
GETCNX   92-150    92-27
GETCXX   92-29     92 310
GTDRVL   79-130    79 16
GTDRVT   72-16     73-16    74 24    74 36    79-110
M.BRL   132-3    132-100
M.BST   132-4    132-110
M.LDR   132-5    132 120
M.JBA   132-1    132 80
```

```
H.VEC   132-2     132-9#
HC.BF1   38-24#    38-25     68-15
HC.BF2   38-25#    38-27     57-138    68-3      68-45
HC.BSZ   38-13#    38-25     38-27     68-18     103-19    103-21
HC.CCT   38-19#    65-40     65-50#    102-19#
HC.CEV   38-22#    38-23     101-18
HC.CMD   38-18#    38-19     38-20     110-25#   110-26#
HC.CPK   38-23#    38-24     57-125    68-33#    99-3#     99-4#     99-5#     99-6#     101-29#   102-17#   103-18#   103-19#   110-26
HC.ESZ   38-11#    38-21     38-23
HC.INT   38-15#    38-16
HC.ISZ   38-9#     38-16     109-21    110-4
HC.MCT   38-17#    68-74#    102-18#
HC.MEV   38-20#    38-21
HC.MPK   38-21#    38-22     57-127    67-9      67-14     67-19     68-11     68-58     68-60     68-65     68-67     68-72     68-73     99-10
        110-24
HC.MSG   38-16#    38-17     38-18     110-23#   110-24#   111-21
HC.PSZ   38-12#    38-22     38-24     101-19    101-25
HC.RSZ   38-10#    38-18     38-20     109-21    110-4
HC.SIZ   38-27#    61-13
HCOMM    61-13#    63-28
HEADER   74-26     74-38     80-13#
HIGHES   51-6#     130-58    130-64
HO.BRL  122-16    123-3     131-22#   132-3     132-3     132-3
HO.BST  122-21    123-9     131-23#   132-4     132-4     132-4
HO.LDR  125-15    125-22    131-24#   132-5     132-5     132-5
HO.UBA  121-25    131-20#   132-1     132-1     132-1
HO.VEC  121-29    122-19    123-6     131-21#   132-2     132-2     132-2
HOE      34-10#
I$AU     30-8#
I$AUTO   30-8#    128-10#   128-12#
I$CLN    30-8#    129-8#    129-13#
I$DU     30-8#
I$HRD   131-14#   132-6#    134-9
I$INIT   30-8#    120-8#    126-14    127-23#
I$MOD    30-8#     30-26     30-26#    33-16     33-16#    34-3      34-3#     118-104   118-104#  119-3     119-3#    129-15    129-15#   130-3
        130-3#    130-84    130-84#   131-3     131-3#    135-10    135-10#
I$MSG    30-8#     57-14#    57-16#    57-18#    57-20#    57-22#    57-24#    57-26#    57-28#    57-30#    57-32#    57-34#    57-36#    57-38#
         57-40#    57-42#    57-44#    57-46#    57-48#    57-50#    57-57#    57-59#    57-62#    57-64#    57-76#    57-78#    57-80#    57-82#
         57-84#    57-86#    57-88#    57-90#    57-92#    57-94#    57-97#    57-99#    57-101#   57-103#   57-105#   57-107#   57-109#   57-111#
         57-113#   57-115#   57-117#   57-119#   57-121#
I$PROT   30-8#    119-10#
I$PTAB   30-8#    136-3     136-3#    136-11    136-11#
I$PWR    30-8#
I$RPT    30-8#
I$SEG    30-8#    130-5
I$SETU   30-8#    136-1     136-1#    136-3     136-13    136-13#
I$SFT   133-12#   134-9     134-10#
I$SRV    30-8#    106-10#   106-14#   107-18#   107-21#   116-5#    116-9#
I$SUB    30-8#    130-5
I$TST    30-8#    130-5     130-5#    130-22    130-28    130-82    130-83    130-83#   130-83#
IBE      34-10#
IDU      34-10#
IER      34-10#
INFO     69-5      74-15#    75-14     76-14
INFOB    74-16     74-36#
INFOE    74-18     74-32#
```

Cross reference table (CREF V05.00)

```
INFOH    74 20     74-24#
INFOP    74-22     74-27#
INFOX    74-28#    74-41
INIT1   120 11    120-13    120-15    120 17    120 20#
INIT10  124-10#   124-15
INIT11  124 11    125-3#
INIT12  125-13#   125-19
INIT13  125 14    125-21#
INIT14  124-5     126 3#
INIT15  126 9#
INIT2   120-22    120-25    120 30#
INIT3   120-31#
INIT4   121-15#   123-17
INIT5   121-21#   121-33
INIT7   122-26#   122-28
INIT8   124-4#    126-5
INITQT  120-18#   120-29
INITXX  126-12#
IPADRS   51-34#    62 14    121-6     122 4
ISR      34-10#
IXE      34-10#
J$JMP    30-8#
KNO     120-47    120-54#
KW.BRL   50-31#   120-49*   120 52
KW.CSR   50-30#    65-31     80-19    104 22    114-28    116-8*    117-13    120-38*   120-45*   120-48*   120-53*
KW.EL    50-34#    65-33     65-36    104-24    104-27    108-26    108-27    108-28    114 30    114-33    116-6*    116-7*    117 16    117-17
        120-39*   120-40*
KW.MZ    50-33#   108-16    117-18    120-51*
KW.VEC   50-32#   120-50*   120 52
KW11I   116-5#    120-52
KWOUT.  116-8     120-37#   120-53
KYES    120-42    120-44    120-48#
L$ACP    30-34#
L$APT    30-34#
L$AUT    30-34#
L$AUTO   30-34    128-10#
L$CCP    30-34#
L$CLEA   30-34    129-8#
L$CO     30 34#
L$DEPO   30-34#
L$DESC   30-34     52 16#
L$DESP   30-34#
L$DEVP   30 34#
L$DISP   30-34     31 9#
L$DLY    30-34#
L$DTP    30-34#
L$DTYP   30-34#
L$DUT    30-34#
L$DVTY   30-34     52-12#
L$EF     30-34#
L$ENVI   30-34#
L$ETP    30-34#
L$EXP1   30-34#
L$EXP4   30-34#
L$EXP5   30-34#
L$HARD   30-34    131 14    131 14#
```

Cross reference table (CREF V05.00)

```
L$MIME   30 34#
L$MPCF   30-34#
L$MPTP   30 34#
L$MW     30 34     32-10     32-10#
L$ICP    30 34#
L$INIT   30 34    120-8#
L$LADP   30 34#
L$LAST   30 34    135-8#    136 13
L$LOAD   30-34#
L$LUN    30-34#    63-24*    65 12*    79 14*
L$MREV   30 34#
L$NAME   30 34#
L$PRIO   30-34#
L$PROT   30-34    119 10#
L$PRT    30-34#
L$REPP   30 34#
L$REV    30 34#
L$SOFT   30-34    133-12    133 12#
L$SPC    30-34#
L$SPCP   30-34#
L$SPTP   30 34#
L$STA    30-34#
L$SW     30-34     33 10     33-10#
L$TEST   30-34#
L$TIML   30-34#
L$UNIT   30 34#    80-13    123-16    126-4     130-25
L10000   32-10     32-17#
L10001   33-10     33-14#
L10002   57-16#
L10003   57-20#
L10004   57-24#
L10005   57-28#
L10006   57-32#
L10007   57-36#
L10010   57-40#
L10011   57-44#
L10012   57 48#
L10013   57-57#
L10014   57-62#
L10015   57-76#
L10016   57-80#
L10017   57-84#
L10020   57-88#
L10021   57-92#
L10022   57-97#
L10023   57 101#
L10024   57-105#
L10025   57-109#
L10026   57 113#
L10027   57-117#
L10030   57-121#
L10031  106-14#
L10032  107-21#
L10033  116-9#
L10035  126-14    127-23#
L10036  128-12#
```

Cross reference table (CREF V05.00)

```
L10037  129 13#
L10040  130 22    130-28    130-8?    130 83#
L10041  131 14    132 6#
L10042  133 12    134 10#
L10043  136 3#
L10045  136 3     136 11#
LDDM     63 22#    63 34
LDNEXT   63-26     63-30     63 32#
LOADDM   63 29     66 30     98 13#
LOADE1   99-11    100-3#
LOADER   98-23     99 9     100 4#
LOE      34 10#
LOG      38 32#
LOT      54 10#
LPNT     74 40     96-6      96-8      96-10     96-12#    117-24    117-28    117-33
LPNTB    57 15     57 19     57-23     57-27     57-31     57-35     57-39     57-43     57-47     57-56     57 61     57-65     57-69     57-75
         57 79     57-83     57 87     57-91     57-95     57-108    57-112    57-116    57-120    57 123    57-129    57-136    57-140    87-5
         88-5      96-7#
LPNTF    80-15     81 41     83-23     86 4      96-5#     118-31    120-46    130-30    130-41
LPNTS    96 11#
LPNTX    96-9#    130-6
MD.CMP   41-4#
MD.CNB   41-22#
MD.ERR   41-6#
MD.EXP   41-5#
MD.FEU   41-16#
MD IMF   41 20#
MD.NXU   41 18#
MD.PRI   41 23#
MD.RIP   41-19#
MD.SCM   41-7#
MD.SCL   41-8#
MD.SEC   41-9#
MD.SEQ   41 14#
MD.SER   41-10#
MD.SPD   41-15#
MD.SSM   41-11#
MD.SWP   41-21#
MD.VOL   41 17#
MD.WBN   41-12#
MD.WBV   41-13#
MESG     74-27     74-39     78-13#
MESSG    54-9#     80-15
MLDRER  125-17    127-7#
MODE     50-26#    73-53     73-59    120-30#   130-17    130-23
MON1    118-17#   118-30
MON2    118-19    118-21#
MON3    118-22    118-24#
MON4    118-25    118-27#
MON5    118-28    118 33#
MON6    118-36    118 39#
MONTHS   51-7#    118-16
MSCP     38-31#
MSGPKL   57 140#   57-143
MSGPKT   57-96     57-136#
NCONF    81 38     81 43#
```

Cross reference table (CREF V05.00)

```
NCONS    81 37#   81 40
NEWTAB  121-22    122-3#
NOCLOC   54-10#   120 46
NULL     50 38#
NXMAD    50-22#    62-10*   62 18    106-12*  111-26*  111 31
NXMI     62-11    106-10#  111-27
NXTTAB  121 16    122-31   123 10    123 15#
O$APTS   30-8#     30 34
O$AU     30-8#     30 34
O$BGNR   30-8#     30 34
O$BGNS   30-8#     30-32#   30-34
O$DU     30 8#     30 34
O$ERRT   30-8#     30-34
O$GNSW   30 8#     30 32#   30 34
O$POIN   30-8#     30 32    30 32#    30-32#   30-32#   30-34
O$SETU   30 8#     30 32#   30 34    135-8
OP.ABO   40 3#
OP.ACC   40-4#
OP.AVA   40-22#
OP.AVL   40-5#
OP.CCD   40 6#
OP.CMP   40 7#
OP.DUP   40-23#
OP.ELP   40 30#
OP.END   40-20#    67 5      67-8      68-58
OP.ERS   40-8#
OP.ESP   40-29#    99-1
OP.FLU   40-9#
OP.GCS   40-10#
OP.GDS   40-27#    65 48     68 58
OP.GSS   40-28#
OP.GUS   40-11#
OP.MRD   40-18#
OP.MUR   40-19#    101 21
OP.ONL   40-12#
OP.RD    40 13#
OP.RLC   40-25#
OP.RPL   40-14#
OP.RSD   40-32#    67-8      68-43
OP.SCC   40-15#
OP.SEX   40-21#
OP.SHC   40-24#
OP.SSD   40-31#    67 5      68-13
OP.SUC   40-16#
OP.WR    40-17#
OSTRE    81-35     81 42     81 47#
OSTRNG   81-34#    81-46     89-6      90-6      91-6      96-17
P.BCNT   42-21#    43-9#     68-11     68 33*    99 4*    103-19*
P.BUFF   42-22#
P.CMST   43-14#
P.CNCL   43 48#
P.CNTF   42-40#    43-46#
P.CNTI   43-49#
P.CPSP   42-34#
P.CRF    42 17#    43-4#     67-19    102-17*
P.CTMO   43 47#
```

Cross reference table (CREF V05.00)

```
P.CYL    43 26#
P.DEXT   43-52#
P.DFLG   43-53#    68 60
P.DMDT   42 50#
P.DPI    43-54#    68-65     68 67     68-72     68 73
P.DTO    43 55#
P.ELGF   42 32#
P.FBBK   43 10#
P.FLGS   43 7#
P.GAP    43 25#
P.HSTI   42 31#    43 19#    43 35#
P.HTMO   42-41#
P.LBN    42-24#
P.MEDI   43-21#    43 37#
P.MLUN   43-17#    43-33#
P.MOD    42-20#
P.OPCD   42 19#    43-6#     67-9      68-58     101-29#
P.OTRF   42-27#    43-13#
P.OVRL   42-51#    99 5#     99-6#
P.RBN    42-36#
P.RBNS   43-28#
P.RCTC   43-29#
P.RCTS   43-27#
P.RGID   42-46#
P.RGOF   42-47#
P.SHST   43-23#    43 39#
P.SHUN   42-33#    43-22#    43 38#
P.STS    43-8#     67-14     99-10
P.TIME   42-43#
P.TRCK   43-24#
P.UADR   42-23#    99-3#     103 18#
P.UNCL   43-40#
P.UNFL   42-30#    43-18#    43 34#
P.UNIT   42-18#    43-5#
P.UNSZ   43-41#
P.UNTI   43-20#    43-36#
P.USEF   42-42#
P.VRSN   42-39#    43-45#
P.VSER   43-42#
PB       95-14#    96-7
PF       50-36     95 12#    96-5
PNT      34-10#
PNT.CT   57-15     57-15     57-15#    57-15#    57-19     57-19     57-19#    57-19#    57-23     57-23     57-23#    57-23#    57-27     57-27#
         57-31     57-31     57-31#    57-31#    57-35     57-35#    57-39     57-39#    57-43     57-43     57-43     57-43#    57-43#    57-43#
         57-47     57-47#    57-56     57-56     57-56     57-56#    57-56#    57 56#    57-61     57-61     57-61     57 61#    57-61#    57 61#
         57-65     57-65     57-65     57-65#    57-65#    57-65#    57-69     57-69     57-69     57-69#    57-69#    57-69#    57-75     57-75#
         57-79     57-79     57-79#    57-79#    57-83     57-83     57-83     57-83#    57-83#    57-83#    57-87     57-87     57-87#    57-87#
         57-91     57-91#    57-95     57-95#    57-108    57-108#   57-112    57-112    57-112#   57-112#   57-116    57-116    57-116#   57-116#
         57-120    57-120#   57-123    57-123#   57-129    57-129    57-129    57-129    57-129    57-129#   57-129#   57-129#   57-129#   57-129#
         57-136    57 136#   57-140    57-140    57-140    57-140    57-140    57-140    57-140    57-140    57-140#   57-140#   57-140#   57-140#
         57-140#   57 140#   57-140#   57-140#   74-40     74-40#    80-15     80-15     80-15     80-15     80-15#    80-15#    80-15#    80-15#
         81-41     81-41#    83-23     83-23#    86-4      86-4#     87-5      87-5      87-5      87-5      87-5      87-5#     87 5#     87-5#
         87-5#     87-5#     88-5      88-5      88-5      88-5      88-5      88-5      88-5#     88-5#     88-5#     88-5#     88-5#     88-5#
        117-24    117-24    117-24#   117-24#   117 28    117-28    117-28#   117 28#   117-33    117-33    117-33#   117-33#   118-31    118-31#
        120-46    120-46#   130-6     130-6#    130-30    130-30#   130-41    130-41    130-41    130-41    130 41#   130-41#   130 41#   130-41#
PNTNUM   82 24     82 30     83 4      93-14#
```

Cross reference table (CREF V05.00)

```
PNTNUS   93 16#
PNTPKL   57-129#  57 133
PNTPKT   57-100   57 104   57 123#
PRI      34 10#
PRI00    34 10#   126 12
PRI01    34 10#
PRI02    34 10#
PRI03    34 10#
PRI04    34-10#
PRI05    34 10#
PRI06    34-10#
PRI07    30 34    34-10#   62-11    98 20    111 27
PS       95-18#   96-11
PTYPE    50-36#   87 8#    88 8#    95 11    96-5#    96-7#    96-9#    96-11#
PX       87-8     88-8     95 16#   96-9
QUE0     72-19    72-26#
QUE7     72 21    72-27#
QUEL     72-28#   72-30
QUEST    69-3     72-16#
RESET    62-9#    120 31   129-11
RESPCT   65-8#    66-4     66-7     66 31
RESPDM   65-6#    66-8     130-81
RG.FLG   38-4#    68 74    102-18
RG.OWN   38-3#    65-50    68 74    102-18   102-19
RNTIM    54-5#    117-24
RNTIM1   54-6#    117-28
RNTIM2   54-7#    117-33
RNTIME   80-22    87-6     88-6     117-13#
RNTIMX   117-14   117-35#
RSPDRP   65-25    66-12#   66-37    67-22    68-9     68-27
RSPDSP   68-26    69-3#    69-9
RSPERR   67-10    68-58#
RSPERW   67-15    67-21#   68 59
RSPIN    65-14    67-5#
RSPMWR   67-7     67-9#
RSPNTO   65-30    65-32    65-35    65-37    65-53#
RSPNXT   65-11    66-3#    66 33    68-57    68-75
RSPOU    67-27#
RSPOU2   68-41    68-50#
RSPOU3   68-48    68-51#
RSPOU4   65-52    68 53#
RSPOUT   65-17    67-27    68-34    68-39#
RSPPT2   68-3#
RSPPT3   68-7     68-11#
RSPPTW   67-20    67-26#
RSPTM    65-23    65-29#
RSPTMO   65-34    65-38#
RSPTOE   65-39    65-41    66-36#   68-61    68-76
RUNDM    63-15#   130-79
S$LSYM   30-8#    32-17#   33-14#   57-16#   57-20#   57-24#   57-28#   57-32#   57-36#   57-40#   57-44#   57-48#   57 57#   57 62#
         57-76#   57 80#   57-84#   57-88#   57-92#   57-97#   57-101#  57-105#  57-109#  57-113#  57-117#  57-121#  106-14#  107-21#
         116-9#   118-1    118-1    118 1    118-1#   127-23#  128-12#  129-13#  130-10   130-10   130-10   130-10#  130-31   130 31
         130-31   130-31#  130 44   130-44   130-44   130-44#  130-83#  132-6#   134-10#
S.CNS    134-5    134-19#
S.FMT    134 1    134-18#
S.NOF    134 9    134 23#
```

Cross reference table (CREF V05.00)

```
S.NRF   134-3    134-12#
S.RST   134-7    134-20#
SA.A2    36-18#
SA.BST   37-23#
SA.CM1   36-32#
SA.CMD   36-27#
SA.CME   36-37#
SA.CNT   37-16#  109-44
SA.CTP   36-40#
SA.DI    36-19#  113-14
SA.ERC   36-13#
SA.ERR   36-9#    57-60   114-19
SA.GO    37-21#  110-20
SA.INE   37-4#
SA.INT   36-25#
SA.LFC   37-22#
SA.MCV   37-15#  109-49   113-38
SA.MS1   36-31#
SA.MSE   36-36#
SA.MSG   36-26#
SA.NV    36-17#
SA.NVE   36-5#
SA.PRG   36-44#
SA.S1    36-5#   112-3    113-14
SA.S2    36-6#   113-23
SA.S3    36-7#   113-31
SA.S4    36-8#   112-13   113-38
SA.STE   36-39#
SA.STP   36-29#  111-18
SA.TST   37-11#  113-8
SA.VCE   37-3#
SA.VEC   36-24#
SA.WRP   36-28#
SAMTAB  121-26   123-3#
SAMVEC  121-31   127-19#
SERNO    53-5#   130-44
SERNUM   54-17#  130-41
SERNX    56-11#  130-64
SETOO   108-17#  108-22
SETO1   108-18   108-20#
SETO2   108-26#  108-29
SETTO    68-56   104-15   108-14#  114-23
SFPTBL   33-10#  120-21   120-24   120-28
SNDCMD   68-52    99-7    102-14#
SO.BIT  133-18#  134-1    134-1    134-1    134-3    134-3    134-3    134-5    134-5    134-5    134-7    134-7    134-7
SO.CNS  120-23   120-27   133-22#  134-5
SO.FM1  133-19#  133-21   134-1
SO.FM2  133-20#  133-21   134-3
SO.FMT   73-53   120-20   130-17   133-21#
SO.STR   73-59   120-27   130-23   133-23#  134-7
SPECL    69-8     77-14#   77-29
SPECLC   77-18#
SPECLE   77-25    77-31#
SPECLL   77-24#   77-27
SPECLR   77-16    77-22#
SPECLX   77-15    77-32#
```

Cross reference table (CHEF V05.00)

ST.ABO	44 7#													
ST.AVL	44 9#													
ST.CMD	44-6#													
ST.CMP	44 12#													
ST.CNT	44-15#													
ST.DAT	44 13#													
ST.DIA	44 17#													
ST.DRV	44 16#													
ST.MST	44 14#													
ST.MFE	44 10#													
ST.MSM	44-3#	67 14	99 10											
ST.OFL	44-8#													
ST.SUB	44 4#													
ST.SUC	44 5#													
ST.WPR	44-11#													
SVCJBL	30 8#	30 13#	30 34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34
	30 34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	31 9	32-10
	32-10	33-10	33-10	52-12	52-16	57-11#	57-14	57-14	57-14	57-18	57-18	57-18	57-22	57-22
	57-22	57-26	57-26	57-26	57-30	57-30	57-30	57-34	57-34	57-34	57-38	57-38	57-38	57-42
	57-42	57-42	57-46	57-46	57-46	57-50	57-50	57-50	57-59	57-59	57-59	57-64	57 64	57-64
	57-78	57-78	57-78	57-82	57-82	57-82	57-86	57-86	57-86	57-90	57-90	57-90	57-94	57-94
	57-94	57-99	57-99	57-99	57-103	57-103	57-103	57-107	57-107	57-107	57-111	57-111	57 111	57-115
	57-115	57-115	57-119	57-119	57-119	58-4#	106-10	107-18	116-5	119-10	120-8	128-10	129 8	131-14
	133-12	135-8	135-8	135-8	135-8#									
SVCINS	30-8#	30-10#	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34	30 34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34	30-34	30 34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34	30-34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30 34
	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34
	30-34	30-34	30-34	30-34	30-34	30-54	30-34	30-34	30-34	30-34	30-34	30-34	30-34	30-34
	30-34	30-34	30-34	31-9	31-9	31-9	31-9	31-9	31-9	32-10	32-10	32-10	33-10	33-10
	33-10	52-12	52-12	52-12	52-12	52-12	52-12	52-16	52-16	52-16	52-16	52-16	52-16	57-8#
	57-16	57-20	57-24	57-28	57-32	57-36	57-40	57-44	57-48	57-57	57-62	57-76	57-80	57-84
	57-88	57-92	57-97	57-101	57-105	57-109	57-113	57-117	57-121	58-1#	59-7	59-7	59 7	59-7
	59-7	59-7	59-7	59-7	59-7	59-7	59-7	59-7	59-8	59-8	59-8	62-11	62-11	62-11
	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11	62-11
	62-11	62-12	62-12	62-12	62-22	62-22	62-22	62-22	62-22	62-22	62-28	62-28	62 28	62-28
	62 28	62-28	62-28	62-28	62-28	62-28	62-28	62-28	62-30	62-30	62-30	64-12	64-12	64-12
	65-8	65-8	65-8	65-24	65-24	65-24	65-24	65-24	65-24	65-24	65-24	65-24	65-24	65-24
	65-24	66-36	66-36	66-36	66-36	66-36	66-36	66-36	66-36	66-36	66-36	66-36	66-36	67-21
	67-21	67-21	67-21	67-21	67-21	67-21	67-21	67-21	67-21	67-21	67-21	68-8	68-8	68-8
	68-8	68-8	68-8	68-8	68-8	68-8	68-8	68-8	68-8	72-22	72-22	72 22	72-22	72-22
	72-22	72-22	72-22	72-22	72-22	72-22	72-22	74-32	74-32	74-32	74-32	74-32	74-32	74-32
	74-32	74-32	74-32	74-32	74-32	77-19	77-19	77-19	77-20	77-20	77-20	77-20	77-20	77-20
	77-24	77-24	77-24	77-24	77-24	77-24	77-25	77-25	77-25	95-12	95-12	95-12	95-12	95 12
	95 12	95 12	95-12	95-12	95-12	95-12	95-12	95-12	95-12	95-12	95-12	95-12	95-12	95-14
	95-14	95 14	95-14	95-14	95-14	95-14	95-14	95-14	95-14	95-14	95-14	95-14	95-14	95-14
	95-14	95-14	95-14	95-16	95-16	95-16	95-16	95-16	95-16	95-16	95-16	95-16	95-16	95-16
	95-16	95-16	95 16	95-16	95-16	95-16	95-16	95-18	95-18	95-18	95-18	95-18	95-18	95-18
	95-18	95-18	95-18	95-18	95-18	95-18	95-18	95-18	95-18	95 18	98-20	98-20	98-20	98-20
	98-20	98-20	98-20	98-20	98 20	98-20	98-20	98-20	98-20	98-20	98-20	98-20	98-20	98-20

Cross reference table (CREF V05.00)

	98-20	100-3	100-3	100-3	100-3	100-3	100-3	100-3	100-3	100-3	100-3	100 3	100-3	104-21
	104-21	104-21	104-29	104-29	104-29	104-29	104-29	104-29	104-29	104-29	104-29	104 29	104 29	104-29
	105-5	105-5	105-5	105-5	105-5	105-5	105-5	105-5	105-5	105-5	105-5	105-5	106-14	106-14
	106 14	107-21	107-21	107-21	109-35	109-35	109-35	109-35	109-35	109-35	109-35	109-35	109-35	109 35
	109-35	109-35	109-54	109-54	109-54	109-54	109-54	109-54	109-54	109-54	109-54	109 54	109-54	109-54
	110 11	110-11	110-11	110-11	110-11	110-11	110-11	110-11	110-11	110-11	110-11	110-11	111-12	111-12
	111-12	111-27	111-27	111-27	111-27	111-27	111-27	111-27	111-27	111-27	111-27	111-27	111-27	111 27
	111-27	111-27	111-27	111-27	111-27	111-30	111-30	111-30	111-30	111-30	111-30	111-33	111-33	111-33
	111-33	111-33	111-33	111-33	111-33	111-33	111-33	111-33	111-33	113-44	113-44	113-44	113-44	113-44
	113-44	113-44	113-44	113-44	113-44	113-44	113-44	114-27	114-27	114-27	114-36	114-36	114-36	114-36
	114-36	114-36	114-36	114-36	114-36	114-36	114-36	114-36	115-5	115-5	115-5	115-5	115-5	115 5
	115-5	115-5	115-5	115-5	115-5	115-5	116-9	116-9	116-9	118-1	118-1	118-1	118-1	118-1
	118 1	118-1	118-1	118 1	118-1	118-1	118-1	118-1	118-1	118-1	118-1	118-1	118-1	118-1
	118-1	118-1	118-1	118-1	118-1	120-10	120-10	120-10	120-10	120-10	120-10	120-11	120-11	120-11
	120-12	120-12	120-12	120-12	120-12	120-12	120-13	120-13	120-13	120-14	120-14	120-14	120-14	120-14
	120-14	120-15	120-15	120-15	120-16	120-16	120-16	120-16	120-16	120-16	120-17	120-17	120-17	120-18
	120-18	120-18	120-32	120-32	120-32	120-32	120-32	120-32	120-41	120-41	120-41	120-41	120-41	120-41
	120-42	120-42	120-42	120-43	120-43	120-43	120-43	120-43	120-43	120-44	120-44	120-44	120-52	120-52
	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52	120-52
	120-52	120-52	121-15	121-15	121-15	121-15	121-15	121-15	121-16	121-16	121-16	124-4	124 4	124-4
	124-4	124-4	124-4	124-5	124-5	124-5	126-12	126-12	126-12	126-12	126-12	126-12	126 14	126-14
	126-14	126-14	126-14	126-14	127-3	127-3	127-3	127-3	127-3	127-3	127-3	127-3	127-3	127-3
	127-3	127-3	127-4	127-4	127-4	127-8	127-8	127-8	127-8	127-8	127-8	127-8	127-8	127-8
	127-8	127-8	127-8	127-9	127-9	127-9	127-14	127-14	127-14	127-14	127-14	127-14	127-14	127-14
	127-14	127-14	127-14	127-14	127-15	127-15	127-15	127-20	127-20	127-20	127-20	127-20	127-20	127 20
	127-20	127-20	127-20	127-20	127-20	127-21	127-21	127-21	127-23	127-23	127-23	128-12	128-12	128-12
	129-13	129-13	129-13	130-7	130-7	130-7	130-8	130-8	130-8	130-10	130-10	130-10	130-10	130-10
	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-10	130-19
	130-19	130-19	130-20	130-20	130-20	130-21	130-21	130-21	130-21	130-21	130-21	130-21	130-21	130-21
	130-21	130-21	130-21	130-22	130-22	130-22	130-22	130-22	130-22	130-27	130-27	130-27	130-27	130-27
	130-27	130-27	130-27	130-27	130-27	130-27	130-27	130-28	130-28	130-28	130-28	130-28	130-28	130-31
	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31
	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-32	130-32	130-32	130-32	130-32
	130-32	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44
	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-64	130-64	130-64
	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64	130-64
	130-64	130-82	130-82	130-82	130-82	130-82	130-82	130-83	130-83	130-83	131-14	131-14	131-14	132-1
	132-1	132-1	132-1	132-1	132-1	132-1	132-1	132-1	132-1	132-1	132-1	132-2	132-2	132-2
	132-2	132-2	132-2	132-2	132-2	132-2	132-2	132-2	132-2	132-3	132-3	132-3	132-3	132-3
	132-3	132-3	132-3	132-3	132-3	132-3	132-3	132-3	132-3	132-3	132-4	132-4	132-4	132-4
	132-4	132-4	132-4	132-4	132-4	132-4	132-4	132-4	132-4	132-4	132-4	132 5	132-5	132-5
	132-5	132-5	132-5	132-5	132-5	132-5	132-5	132-5	132-5	132-5	132-5	132-5	132-6	132-6
	132-6	133-12	133-12	133-12	134-1	134-1	134-1	134-1	134-1	134-1	134-1	134-1	134-1	134-2
	134-2	134-2	134-3	134-3	134-3	134-3	134-3	134-3	134-3	134-3	134-3	134 4	134-4	134-4
	134-5	134-5	134-5	134-5	134-5	134-5	134-5	134-5	134-5	134-6	134-6	134-6	134-7	134 7
	134-7	134-7	134-7	134-7	134-7	134-7	134-7	134-8	134-8	134-8	134-9	134-9	134-9	134-9
	134-9	134-9	134-10	134-10	134-10	135-8	135-8	135-8	135-8	135-8	135-8	135-8	135-8	135-8
	136-3	136-3	136-3	136-3	136-3	136-3								
SVCSUB	30-80	30-120	57-100	58-30										
SVCTAG	30-80	30-140	32-17	33-14	57-120	57-16	57-16	57-16	57-20	57-20	57-20	57-24	57-24	57 24
	57-28	57-28	57-28	57-32	57-36	57-32	57-36	57-36	57-36	57-40	57-40	57-40	57-44	57-44
	57-44	57-48	57-48	57-48	57-57	57-57	57-57	57-62	57-62	57-62	57-76	57-76	57-76	57 80
	57-80	57-80	57-84	57-84	57-84	57-88	57-88	57-88	57-92	57-92	57-92	57-97	57-97	57 97
	57-101	57-101	57-101	57-105	57-105	57-105	57-109	57-109	57-109	57-113	57-113	57-113	57-117	57 117
	57-117	57-121	57-121	57-121	58-50	106-14	107-21	116-9	118-1	127-23	128-12	129-13	130 10	130-31
	130-44	130-83	132-6	134-10	136-3	136-11								

```
SVCTST  30-8#    30-11#   57-9#    58-2#    130-5
SWEND   134-2    134-4    134 6    134-8    134-10#
T$$AUT  128-10#  128-12
T$$CLE  129-8#   129 13
T$$DAT  136-3    136-3#   136-11
T$$HAR  131-14   131-14#  132-6
T$$HW    32-10    32-10#   32 17
T$$INI  120-8#   126-14   127-23
T$$MSG   57-14#   57-16    57-18#   57-20    57-22#   57-24    57-26#   57-28    57-30#   57-32    57-34#   57 36    57-38#   57-40
         57-42#   57-44    57-46#   57-48    57 50#   57-57    57-59#   57-62    57-64#   57-76    57-78#   57-80    57-82#   57-84
         57-86#   57-88    57-90#   57-92    57 94#   57-97    57-99#   57-101   57-103#  57-105   57-107#  57-109   57-111#  57 113
         57-115#  57-117   57-119#  57 121
T$$PC   136-1#   136-13
T$$PRO  119-10#
T$$PTA  136-1#   136-3    136-3#
T$$SOF  133-12   133-12#  134-10
T$$SRV  106-10#  106-14   107-18#  107-21   116-5#   116-9
T$$SW    33-10    33-10#   33-14
T$$TES  130-5#   130-22   130-28   130-82   130-83
T$ARGC   30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34#   30-34#   30-34#
         30-34#   30-34#   30-34#   95-12    95-12    95-12    95-12#   95-12#   95-14    95-14    95-14    95-14#   95 14#   95 16
         95-16    95-16    95-16#   95-16#   95-18    95-18    95-18    95-18#   95-18#   130-64   130-64   130-64   130-64#  130-64#
T$CODE  118-1    118-1    118-1    118-1#   118-1#   118-1#   130-10   130-10   130-10   130-10#  130-10#  130-10#  130-31   130-31
        130-31   130-31#  130-31#  130-31#  130-44   130-44   130-44   130-44#  130-44#  132-1    132-1    132-1    132-1    132-1#
        132-1#   132-1#   132-2    132-2    132-2    132-2#   132-2#   132-2#   132-3    132-3    132-3    132-3#   132-3#   132-3#
        132-4    132-4    132-4    132-4#   132-4#   132-4#   132-5    132-5    132-5    132-5#   132-5#   132-5#   134-1    134-1
        134-1    134-1#   134-1#   134-1#   134-2    134-2    134-2    134-2    134-2    134-2    134-2#   134-2#   134-2#   134-2#
        134-3    134-3    134-3    134-3#   134-3#   134-3#   134-4    134-4    134-4    134-4    134-4    134-4    134-4#   134-4#
        134-4#   134-4#   134-5    134-5    134-5    134-5#   134-5#   134-5#   134-6    134-6    134-6    134-6    134-6    134-6
        134-6#   134-6#   134-6#   134-6#   134-7    134-7    134-7    134-7#   134-7#   134-7#   134-8    134-8    134-8    134-8
        134-8    134-8    134-8#   134-8#   134-8#   134-8#   134-9    134-9#
T$ERRN   30-8#    59-7     59-7#    62-28    62-28#   65-24    65-24#   66-36    66-36#   67-21    67-21#   68-8     68-8#    72-22
         72-22#   74-32    74-32#   100-3    100-3#   104-29   104-29#  105-5    105-5#   109-35   109-35#  109-54   109-54#  110-11
        110-11#  111-33   111-33#  113-44   113-44#  114-36   114-36#  115-5    115-5#   127-3    127-3#   127-8    127-8#   127 14
        127-14#  127-20   127-20#  130-21   130-21#  130-27   130-27#
T$EXCP  118-1    118-1#   130-31   130-31#  130-44   130-44#  132-1    132-1#   132-2    132-2#   132-3    132-3#   132-4    132-4#
        132-5    132-5#
T$FLAG  126-14   126-14   126-14#  126-14#  130-22   130-22   130-22#  130-22#  130-28   130-28   130-28#  130-28#  130-82   130-82
        130-82#  130-82#
T$FREE  135-8    136-13#
T$GMAN   30-8#   118-1    118-1#   118-1#   130-31   130-31#  130-31#  130-44   130-44#  130-44#
T$HILI  118-1    118-1#   130-31   130-31#  130-44   130-44#  132-1    132-1#   132-2    132-2#   132-3    132-3#   132-4    132-4#
        132-5    132-5#
T$LAST   30-8#   135-8#   136-1
T$LOLI  118-1    118-1#   130-31   130-31#  130-44   130-44#  132-1    132-1#   132-2    132-2#   132-3    132-3#   132-4    132-4#
        132-5    132-5#
T$LSYM   30-8     30-8#    32-17    33-14    57-16    57-20    57-24    57-28    57-32    57-36    57-40    57-44    57-48    57-57
         57-62    57-76    57-80    57-84    57-88    57-92    57-97    57-101   57-105   57-109   57-113   57-117   57-121   106-14
        107-21   116-9    127-23   128-12   129-13   130-83   132-6    134-10
T$LTNO  135-8#
T$NEST   30-8#    30-26    30-26    30-26#   32-10    32-10    32-10#   32-17    32-17    32-17    32-17#   33-10    33-10    33-10#
         33-14    33-14    33-14    33-14#   33-16    33-16    33-16    33-16#   34-3     34-3     34-3#    57-14    57-14    57-14#
         57-16    57-16    57-16    57-16#   57-18    57-18    57-18#   57-20    57-20    57-20    57-20#   57-22    57-22    57-22#
         57-24    57-24    57 24    57-24#   57-26    57-26    57-26#   57-28    57-28    57-28    57-28#   57-30    57-30    57-30#
         57-32    57-32    57-32    57-32#   57-34    57-34    57-34#   57-36    57-36    57-36    57-36#   57-38    57-38    57-38#
         57-40    57-40    57 40    57-40#   57-42    57-42    57-42#   57-44    57-44    57-44    57-44#   57-46    57-46    57-46#
```

```
            57 48    57 48    57-48    57-48#   57-50    57-50    57 50#   57 57    57 57    57 57    57 57#   57 59    57 59    57 59#
            57-62    57-62    57-62    57 62#   57-64    57-64    57 64#   57 76    57 76    57-76    57-76#   57 78    57 78    57 78#
            57-80    57-80    57-80    57-80#   57-82    57-82    57 82#   57-84    57 84    57-84    57 84#   57-86    57 86    57-86#
            57 88    57-88    57-88    57-88#   57-90    57-90    57-90#   57-92    57-92    57-92    57 92#   57-94    57 94    57 94#
            57-97    57-97    57-97    57-97#   57-99    57-99    57-99#   57 101   57-101   57-101   57-101#  57-103   57-103   57 103#
            57 105   57-105   57-105   57-105#  57-107   57 107   57-107#  57 109   57-109   57-109   57-109#  57-111   57-111   57-111#
            57 113   57-113   57 113   57-113#  57-115   57-115   57-115#  57-117   57 117   57 117   57 117#  57 119   57-119   57-119#
            57-121   57-121   57 121   57-121#  106-10   106 10   106-10#  106 14   106 14   106 14   106 14#  107 18   107-18   107-18#
           107 21   107-21   107-21   107-21#  116 5    116-5    116-5#   116-9    116-9    116-9    116-9#   118-104  118-104  118-104
           118-104# 119-3    119-3    119-3#   119-10   119-10   119-10#  119-16   119-16   119 16   119-16#  120 8    120-8    120-8#
           127-23   127-23   127-23   127-23#  128 10   128-10   128-10#  128-12   128-12   128-12   128-12#  129-8    129-8    129 8#
           129 13   129-13   129-13   129-13#  129-15   129-15   129-15   129-15#  130-3    130 3    130-3#   130-5    130-5    130-5#
           130-83   130-83   130-83   130-83#  130-84   130 84   130-84   130-84#  131-3    131-3    131-3#   131 14   131-14   131 14#
           132 6    132 6    132 6    132-6#   133-12   133-12   133 12#  134 2    134-4    134-6    134-8    134 10   134-10   134 10
           134-10#  135 10   135 10   135-10   135-10#
TSNSO       30 26#   33 16    34-3#    118-104  119-3#   129 15   130 3#   130-84   131-3#   135 10
TSNS1       32 10#   32-17    33-10#   33-14    57-14#   57-16    57-18#   57-20    57-22#   57 24    57-26#   57-28    57-30#   57 32
            57-34#   57-36    57 38#   57-40    57-42#   57-44    57-46#   57-48    57-50#   57-57    57-59#   57-62    57-64#   57-76
            57 78#   57-80    57-82#   57 84    57-86#   57-88    57-90#   57-92    57-94#   57 97    57-99#   57-101   57-103#  57-105
            57-107#  57-109   57-111#  57-113   57-115#  57 117   57-119#  57-121   106-10#  106-14   107-18#  107-21   116 5#   116-9
           119 10#  119-16   120-8#   127-23   128 10#  128-12   129-8#   129 13   130-5#   130-83   131-14#  132-6    133 12#  134-2
           134-4    134-6    134-8    134-10
TSPCNT     136-1#   136-3    136-3    136-3#
TSPTAB     136-3    136-3#
TSPTHV      30-34   136-13#
TSPTNU      30-8#   136-3    136 3#   136 13   136 13
TSSAVL      30-8#
TSSEGL      30-8#
TSSIZE     135-8    136-13#
TSSUBN      30-8#   130-5#
TSTAGL      30-8#
TSTAGN      30-8#    32 10    32-10    32-10#   33-10    33-10    33-10#   57-14    57-14    57-14#   57-18    57-18    57-18#   57 22
            57-22    57-22#   57-26    57-26    57-26#   57-30    57-30    57-30#   57-34    57-34    57-34#   57-38    57-38    57 38#
            57-42    57-42    57-42#   57-46    57-46    57-46#   57-50    57-50    57-50#   57-59    57 59    57-59#   57-64    57-64
            57-64#   57-78    57-78    57-78#   57-82    57-82    57-82#   57-86    57-86    57-86#   57-90    57-90    57-90#   57 94
            57-94    57-94#   57-99    57-99    57-99#   57-103   57-103   57-103#  57-107   57-107   57-107#  57-111   57-111   57-111#
            57-115   57-115   57-115#  57-119   57-119   57-119#  106-10   106-10   106-10   107-18   107 18   107-18#  116-5    116-5
           116-5#   119-10   119-10   119-10#  120-8    120-8    120-8#   128-10   128-10   128-10#  129-8    129-8    129-8#   130-5
           130-5    130-5#   131-14   131-14   131-14#  133-12   133-12   133-12#  136 1    136-1    136-1#   136-3    136 3    136-3
           136-3    136-3#   136-3#
TSTEMP      31-9     31-9     31-9#    31-9#    32-17    32-17#   33-14    33-14#   33-16    33-16#   57-16    57 16#   57-20    57-20#
            57-24    57-24#   57-28    57-28#   57-32    57-32#   57-36    57-36#   57-40    57-40#   57-44    57-44#   57-48    57-48#
            57-57    57-57#   57-62    57-62#   57-76    57-76#   57-80    57-80#   57-84    57-84#   57-88    57-88#   57 92    57-92#
            57-97    57-97#   57-101   57-101#  57-105   57-105#  57-109   57-109#  57-113   57-113#  57-117   57-117#  57-121   57-1
           106-14   106-14#  107-21   107-21#  116-9    116-9#   118-1    118-1    118-1    118-1#   118-1#   118-1#   118-104  118-104#
           119-16   119-16#  126-14   126-14#  127-23   127-23#  128-12   128-12#  129-13   129-13#  129-15   129-15#  130-10   130-10
           130-10   130-10#  130-10#  130-10#  130-22   130-22#  130-28   130-28#  130-31   130-31   130-31   130-31#  130-31#  130-31#
           130-44   130-44   130-44   130-44#  130-44#  130-44#  130-82   130-82#  130-83   130-83#  130-84   130-84#  132-1    132-1
           132-1    132-1#   132-1#   132-1#   132-2    132-2    132-2    132-2#   132-2#   132-2#   132-3    132 3    132-3    132-3#
           132 3#   132-3#   132-4    132-4    132-4    132-4#   132-4#   132-4#   132-5    132-5    132-5    132-5#   132-5#   132 5#
           132-6    132 6#   134-1    134-1    134-1    134-1#   134-1#   134-1#   134-3    134-3    134-3    134-3#   134-3#   134-3#
           134-5    134-5    134-5    134-5#   134-5#   134-5#   134-7    134-7    134-7    134-7#   134-7#   134-7#   134-10   134-10#
           135-10   135-10#
TSTEST      30-8#   130-5    130 5    130-5#   135-8
TSTSTM      30 8#    57-16    57-20    57-24    57-28    57-32    57-36    57-40    57-44    57-48    57-52    57-62    57 76    57-80
            57 84    57 88    57 92    57-97    57 101   57-105   57-109   57 113   57-117   57-121   59-7     59 8     62 11    62-12
```

Cross reference table (CREF V05.00)

	62 22	62 28	62-30	64 12	65 8	65-24	66 36	67 21	68-8	72 22	74 52	77 19	77 20	77 24
	95-12	95 14	95-16	95 18	98-20	100-3	104-21	104-29	105 5	109-35	109 54	110 11	111 12	111 27
	111-30	111-33	113-44	114-27	114 36	115-5	118 1	120-10	120 12	120-14	120 16	120-18	120 32	120-41
	120 43	120 52	121-15	124 4	126-12	126-14	127-3	127-4	127-8	127 9	127 14	127 15	127 20	127 21
	127-23	128-12	129-13	130-7	130 10	130 19	130 21	130-22	130-27	130 28	130 31	130 32	130-44	130 64
	130 82	130-83												
T1STS	30-8#	130-5#												
T1	31 9	130-5#												
T1CNS	130-24	130-35#												
T1FMT	130 18	130-34	130 77#											
T1GO	130-20	130-23#												
T1MODE	130-8	130-14	130 17#											
T1QUIT	130-12	130-22#												
T1RST	130-26	130-30#												
T1SER1	130-36#	130-76												
T1SER2	130 39#	130-73												
T1SER3	130 44#	130-65												
T1SER4	130-47#	130-53												
T1SER5	130-48	130-55#												
T1SER6	130 59#	130-62												
T1SER7	130-51	130-54	130-64#											
T1SER8	130-56	130-60	130-63	130-66#										
T1SER9	130 68#	130-69												
T1SERN	130 40	130-71	130-74#											
TEMP	51-1#	125-9#	127-7	127 13	130-9#	130-10	130-11	130-44	130 45	130 57	130-67			
TERM	69-6	75-14#												
TINDEX	45 10#	45-12	45-12	45-12#	45-13	45-13	45-13#	45-16	45-16	45-16#	45-19	45-19	45 19#	45-20
	45-20	45-20#	45-21	45-21	45-21#	45-22	45-22	45-22#	45-34	45-34	45-34#	45-35	45-35	45-35#
	45-36	45-36	45-36#	45-37	45-37	45-37#	45-38	45-38	45-38#	45-39	45-39	45-39#	45-40	45-40
	45-40#	45-41	45-41	45-41#	45-42	45-42	45-42#	45-43	45-43	45-43#	45-44	45 44	45 44#	45-45
	45-45	45-45#	45-46	45-46	45-46#	45-47	45-47	45-47#	45-49	46-7#	46 9	46-9	46 9#	46-10
	46-10	46-10#	46-13	46-13	46-13#	46-15	131-18#	131-20	131-20	131-20#	131-21	131 21	131-21#	131 22
	131-22	131-22#	131-23	131-23	131-23#	131-24	131-24	131-24#	133-16#	133-18	133-18	133 18#		
TOOMER	125-20	127-13#												
TSTTAB	50-14#	63-21	65-6	130-77#										
UAM	34-10#													
UCNT	50-19#	63-20#	63-33#	65-7#	66-6#									
UDAFM	50-15	50-16												
UDAI1A	109-32#	109-38												
UDAI1B	109-34	109-37#												
UDAI1C	109-33	109 39#												
UDAI1D	109-51	109-54#												
UDAI1L	109-22#	109-24												
UDAI2	109-53	110-3#												
UDAI2E	110-6	110-11#												
UDAI2L	110-5#	110-8												
UDAI3	110-9	110-16#												
UDAID1	111-19#	113-4#	113-20	113 29										
UDAID2	111-20#	111-21#	113-6#											
UDAID3	113-8#													
UDAIDT	112-4	113-3#												
UDAIEX	109-29	109-36	109-42	109 55	110-12	110-32#								
UDAINT	98-22	109-20#												
UDAIR1	113-3	113-14#												
UDAIR2	113 5	113-20#												
UDAIR3	113-7	113 29#												

ZUDEDO PDP 11 UDA DRV MTR MACRO V05.00 Tuesday, 14 May 85 10:10 Page S 20
Cross reference table (CREF V05.00)

```
UDAIR4  113 9     113 5*
UDAIRC  113-15    113-24    113-3*    113 42*
UDAIRX  113-43    113-46*
UDAISE  111 35    112-9     112-11    112 19*
UDAISG  111-32    112-3*
UDAISL  112-8*    112-16
UDAIST  109-28    111-12*
UDAISX  112-14    112-18*
UDARS1  114-25*   114-29    114-32    114 34
UDARS2  114-26    115-3*
UDARSD   57-60*    57-61    112 3*    112 12*   112 13    114 19*   114 25    115 16*
UDARSE  114 37    115-6*
UDARSP  109-41    112-8     114 18*
UDARSX  115-4     115 11*
UDASRV  107-18*   122-23
UF.576   42 12*
UF.CMR   42 3*
UF.CMW   42-4*
UF.INA   42-6*
UF.RPL   42-5*
UF.SCM   42 7*
UF.SCL   42-8*
UF.WBN   42-9*
UF.WPM   42-10*
UF.WPS   42-11*
UFREEZ   50-21*    63-35*    66-5      66-13*    74-21     74-23*
URNING   50-18*    63-16*    63-31*    63-40     66-32*
URUN     50-17*    63-15*    63-20     65-7
WAITMS   99-8      104-11*
WNQUES   53-6*     130-10
WNSTOP   54-18*     74-40
WNSTRT   54-21*    130-6
X$ALWA   30-8*
X$FALS   30-8*
X$OFFS   30-8*     134-2     134-4     134-6     134-8
X$TRUE   30-8*     134-2     134-4     134-6     134-8
X1       55-5*      57-15
X10      55-13*     57-39
X100     55-41*     57-116
X101     55-42*     57-120
X14      55-14*     57-43
X1A      55-1*      57-15
X2       55-6*      57-19
X20      55-18*     57-47
X21      55-22*     57-56
X22      55-24*     57-61
X23A     55-26*     57-65
X23B     55 30*     57-69
X24      55-31*     57-79
X25      55-33*     57-83
X2A      55-2*      57-19
X3       55-7*      57 23
X30      55-35*     57-87
X31      55-36*     57-91
X32      55-37*     57-95
X56      55-38*     57 108
```

```
X3*      55 40#   57-112
X3A      55-3#    57-23
X4       55-8#    57-27
X8       55 10#   57-31
X8A      55-4#    57 31
X9       55 11#   57-35
XFRU     56-8#    57-75     90 5
XMSG1    56-1#    57-136
XMSG2    56 2#    57-140
XPKT1    56 3#    57-123
XPKT2    56-6#    57 129
XSA      56 7#    91-5
YEAR19   51-31#  118-92
YEAR20   51-32#  118 95
YER1    118-69#  118-82
YER2    118 70   118-72   118-83#
YER3    118 86   118-92#
YER4    118-94   118-96#  118-99
YER5    118-91   118-97   118-100# 118 101
```

```
.BR      47 31#
AND      47 30#    57-53     98 17     111 15
ASSUME   47 39#    63-27     66 20      73-36     79 15     80-16     80 17    120-27    121 24    121-25    130 42    130-43
BCOMPL  120 11    120-13    120-15    120-17    120 42    120-44    130-20
BGNAUT  128 10
BGNCLN  129 8
BGNHRD  131 14
BGNHW    32-10
BGNINI  120-8
BGNMOD   30-26     34 3     119-3     130 3     131-3
BGNMSG   57-14     57-18     57-22     57 26     57 30     57-34     57-38     57-42     57-46     57 50     57-59     57-64     57-78     57 82
         57-86     57-90     57 94     57 99     57-103    57 107    57-111    57-115    57-119
BGNPRO  119-10
BGNPTA  136-3
BGNSET  136-1
BGNSFT  133 12
BGNSRV  106 10    107 18    116-5
BGNSW    33 10
BGNTST  130-5
BNCOMP   77-25    121-16    124-5     130-8
BREAK    62-12     65-8     104-21    111-12    114 27
BUILD    84-17#    85-9      85-21
CLOCK   120-41    120-43
CLOSE    64-12     77-19
CLRVEC   62-22    111-30
DESCRI   52-16
DEVTYP   52-12
DISPAT   31-9
DISPLA  134-9
DOCLN    59 8      62-30    120-18    127-4     127-9     127-15    127-21
END      35-29#    45-49     46-15    131-25    133-25
ENDAUT  128-12
ENDCLN  129-13
ENDHRD  132-6
ENDHW    32-17
ENDINI  127 23
ENDMOD   33-16    118-104   129-15    130-84    135-10
ENDMSG   57-16     57 20     57-24     57-28     57-32     57-36     57-40     57-44     57-48     57-57     57-62     57 76     57-80     57 84
         57-88     57-92     57-97     57-101    57-105    57-109    57-113    57-117    57-121
ENDPRO  119-16
ENDPTA  136-11
ENDSET  136-13
ENDSFT  134-10
ENDSRV  106-14    107-21    116-9
ENDSW    33-14
ENDTST  130-83
ENTRY    85-3#     85-9      85-9      85-9      85-9      85-9      85-9      85-9      85-9      85-15#    85-21     85-21     85-21     85-21
         85-21     85-21     85-21     85-21
EQUALS   34-10
ERRDF    62-28     65-24     66-36     67-21     68-8      72-22     74-32    100-3     104-29    105-5     109-35    109 54    110 11    111-33
        113-44    114-36    115-5
ERRSF    59-7      127-3     127-8     127-14    127-20    130-21    130-27
EXIT    126-14    130-22    130-28    130-82
GETBYT   77-24
GMANID  118-1     130-31    130 44
GMANIL  130 10
```

Cross reference table (CREF V05.00)

```
GPHARD 121 15   124 4
GPRMA  132 1    132-2
GPRMD  118-1    118-1#   130-31   130-31#  130 44   130-44#  132 3    132-4    132 5
GPRML  130 10   130 10#  134-1    134 3    134-5    134 7
HEADER  30-34
ITEM    35 24#   45-12    45-13    45-16    45-19    45-20    45-21    45 22    45-34    45-35    45-36    45-37    45 38    45 39
        45 40    45-41    45 42    45 43    45 44    45-45    45-46    45-47    46 9     46 10    46 13    131 20   131 21   131 22
       131-23   131 24   133 18
LASTAD 135-8
MSBYTE  30-34    30-34    30-34    30-34#
MSCMEC 126-14   126 14#  130-22   130-22#  130-28   130-28#  130-82   130-82#
MSCNTD 118-1    118-1#   130-10   130-10#  130-31   130-31#  130-44   130-44#  132-1    132 1#   132-2    132 2#   132 3    132 30
       132-4    132-4#   132 5    132-5#   134-1    134 1#   134-3    134-3#   134-5    134-5#   134-7    134 7#
MSCOUN  95-12    95-12#   95-14    95-14#   95-16    95-16#   95-18    95-18#   130-64   130-64#
MSDATA  30-34    30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34
        30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34
        30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34#   30-34#   52-12    52-12#   52 16
        52-16#
MSDECR  32-17    32 17#   33-14    33-14#   33-16    33-16#   57-16    57-16#   57-20    57-20#   57-24    57 24#   57-28    57-28#
        57-32    57-32#   57-36    57-36#   57-40    57-40#   57-44    57-44#   57-48    57-48#   57-57    57-57#   57 62    57-62#
        57-76    57-76#   57-80    57-80#   57-84    57-84#   57-88    57-88#   57-92    57-92#   57-97    57-97#   57 101   57-101#
        57-105   57-105#  57 109   57-109#  57-113   57-113#  57-117   57-117#  57-121   57-121#  106-14   106-14#  107-21   107-21#
       116-9    116-9#   118-104  118-104#  119-16   119-16#  127-23   127-23#  128-12   128-12#  129-13   129-13#  129 15   129-15#
       130-83   130-83#  130-84   130-84#   132-6    132-6#   134-10   134-10#  135-10   135-10#  136-3    136-3#
MSDEFA 118-1    118-1#   130-10   130-10#  130-31   130-31#  130-44   130-44#  132-1    132-1#   132-2    132-2#   132 3    132-3#
       132-4    132-4#   132-5    132-5#   134-1    134-1#   134-3    134-3#   134-5    134-5#   134-7    134-7#
MSENDE  32-17#   33-14#   33-16#   57-16#   57-20#   57-24#   57-28#   57-32#   57-36#   57-40#   57-44#   57-48#   57 57#   57-62#
        57-76#   57-80#   57-88#   57-92#   57-97#   57-101#  57-105#  57-109#  57-113#  57-117#  57 121#  106-14#  107-21#
       116-9#   118-104# 127-23#  128-12#  129-13#  129-15#  130-83#  130-84#  132-6#   134-10#  135-10#
MSERRI  59 7     59-7#    62-28    62-28#   65-24    65-24#   66-36    66-36#   67-21    67-21#   68-8     68-8#    72 22    72-22#
        74-32    74-32#   100-3    100-3#   104-29   104-29#  105-5    105-5#   109-35   109-35#  109-54   109-54#  110 11   110-11#
       111-33   111-33#  113-44   113-44#  114-36   114-36#  115-5    115-5#   127-3    127-3#   127-8    127-8#   127-14   127-14#
       127-20   127-20#  130-21   130-21#  130-27   130-27#
MSEXCP 118-1    118-1    118-1#   130-31   130-31   130-31#  130-44   130-44   130-44#  132-1    132-1    132 1#   132-2    132-2
       132-2#   132-3    132-3    132-3#   132-4    132-4    132-4#   132-5    132-5    132-5#
MSEXIT 126-14   126-14#  130-22   130-22#  130-28   130-28#  130-82   130-82#
MSFXSE 126 14#  130-22#  130-28#  130-82#
MSEXTJ 126-14#  130-22#  130-28#  130-82#
MSGEN   30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34    30-34    30 34    30 34
        30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34
        30-34    30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34#   30-34#   30 34#   30-34#
        30 34#   30 34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#
        30 34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30 34#   30 34#
        30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   31-9     31-9#    32-10    32-10    32-10#   32-10#   32-17    32-17#
        33 10    33-10    33-10#   33-10#   33-14    33-14#   52-12    52-12#   52-16    52-16#   57-14    57-14#   57 16    57-16#
        57-18    57-18#   57-20    57-20#   57-22    57-22#   57-24    57-24#   57-26    57-26#   57-28    57-28#   57-30    57-30#
        57 32    57-32#   57-34    57-34#   57-36    57-36#   57-38    57-38#   57-40    57-40#   57-42    57-42#   57-44    57-44#
        57-46    57-46#   57-48    57-48#   57-50    57-50#   57-57    57-57#   57-59    57-59#   57-62    57-62#   57-64    57-64#
        57-76    57-76#   57-78    57 78#   57-80    57-80#   57-82    57-82#   57-84    57-84#   57-86    57-86#   57-88    57 88#
        57-90    57 90#   57-92    57-92#   57-94    57-94#   57-97    57-97#   57-99    57-99#   57-101   57-101#  57-103   57-103#
        57 105   57 105#  57-107   57-107#  57-109   57-109#  57-111   57-111#  57-113   57-113#  57-115   57 115#  57-117   57-117#
        57-119   57-119#  57-121   57-121#  106-10   106-10#  106-14   106-14#  107-18   107-18#  107-21   107 21#  116-5    116-5#
       116 9    116 9#   118-1    118-1#   119-10   119-10#  120-8    120-8#   127-23   127-23#  128-10   128-10#  128-12   128-12#
       129-8    129-8#   129 13   129-13#  130-5    130-5#   130-10   130-10#  130-31   130-31#  130 44   130-44#  130-83   130 83#
       131-14   131-14#  132 6    132 6#   133-12   133-12#  134-10   134-10#  135-8    135-8#   136 3    136 3#   136 11   136-11#
MSGENR 118-1    118 1#   130-10   130-10#  130-31   130-31#  130-44   130-44#
```

Cross reference table (CREF V05.00)

```
M$GETS   32 17    32 17#   33-14    33 14#   33-16    33-16#   57 16    57 16#   57-20    57 20#   57 24    57-24#   57-28    57 28#
         57-32    57-32#   57-36    57-36#   57-40    57-40#   57-44    57-44#   57 48    57-48#   57-57    57-57#   57-62    57-62#
         57 76    57-76#   57-80    57-80#   57 84    57-84#   57-88    57-88#   57 92    57-92#   57-97    57-97#   57 101   57-101#
         57 105   57-105#  57-109   57-109#  57-113   57-113#  57-117   57-117#  57-121   57-121#  106 14   106 14#  107 21   107 21#
        116 9     116-9#   118-104  118-104# 119-16   119-16#  127-23   127-23#  128-12   128-12#  129 13   129-13#  129-15   129-15#
        130 83    130-83#  130-84   130-84#  132-6    132-6#   134-2    134 2#   134-4    134-4#   134-6    134-6#   134 8    134-8#
        134-10    134-10#  135-10   135-10#
M$GETT  126 14#   130-22#  130-28#  130-82#  134-2    134-2#   134 4    134-4#   134 6    134-6#   134-8    134 8#
M$GNGB   30-26#    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34    30-34
         30-34     30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34    30-34    30-34
         30-34     30-34    30-34    30-34    30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34#   30-34#   30 34#
         30 34#    30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30 34#   30-34#
         30-34#    30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#
         30 34#    30-34#   30 34#   30-34#   30-34#   30-34#   30-34#   30-34#   31-9     31-9#    32-10    32-10    32-10#   33-10
         33-10#    34-3#    52-12    52-12#   52-16    52-16#   57-14    57-14#   57 18    57-18#   57-22    57-22#   57-26    57-26#
         57 30     57 30#   57-34    57-34#   57-38    57-38#   57-42    57-42#   57-46    57-46#   57 50    57-50#   57-59    57 59#
         57 64     57-64#   57-78    57-78#   57-82    57-82#   57-86    57-86#   57-90    57-90#   57-94    57-94#   57-99    57-99#
         57-103    57-103#  57-107   57-107#  57-111   57-111#  57-115   57-115#  57-119   57-119#  106-10   106-10#  107-18   107 18#
        116 5     116-5#   119-3#   119-10   119-10#  120-8    120-8#   128-10   128-10#  129-8    129-8#   130 3#   131-3#   131-14
        131-14#   133-12   133-12#  135-8    135-8#
M$GNIN   30 34     30-34    30-34    30-34    30 34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30 34    30-34
         30-34     30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34
         30-34     30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34    30-34
         30-34     30-34    30-34    30-34    30-34    30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30 34#   30-34#
         30 34#    30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30 34#
         30 34#    30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#   30-34#
         30 34#    30 34#   30-34#   30-34#   30-34#   30-34#   31-9     31-9#    31-9#    31-9#    32-10    32-10#   33 10    33-10#
         52-12     52-12    52-12#   52-12#   52-16    52-16    52-16#   52-16#   57-16    57-16#   57-20    57-20#   57-24    57-24#
         57-28     57-28#   57-32    57-32#   57-36    57-36#   57-40    57-40#   57-44    57-44#   57-48    57-48#   57-57    57-57#
         57-62     57 62#   57-76    57-76#   57-80    57-80#   57-84    57-84#   57-88    57-88#   57-92    57-92#   57-97    57-97#
         57-101    57-101#  57-105   57-105#  57-109   57-109#  57-113   57-113#  57-117   57-117#  57-121   57-121#  59-7     59-7
         59-7      59-7     59-7#    59-7#    59-7#    59-7#    59-7#    59-7#    59-8     59-8#    62-11    62-11    62-11    62-11
         62-11     62-11#   62-11#   62-11#   62-11#   62-11#   62-11#   62-11#   62-12    62-12#   62-22    62-22    62-22#   62-22#
         62-28     62-28    62-28    62-28#   62-28#   62-28#   62-28#   62-28#   62-28#   62-30    62-30#   64-12    64-12#   65-8     65-8#
         65-24     65-24    65-24    65-24    65-24#   65-24#   65-24#   65-24#   65-24#   65-24#   66-36    66-36    66-36    66-36#
         66 36#    66-36#   66-36#   66-36#   67-21    67-21    67-21    67-21    67-21#   67-21#   67-21#   67-21#   67-21#   68-8
         68-8      68-8     68-8     68-8#    68-8#    68-8#    68-8#    68-8#    72-22    72-22    72-22    72-22    72-22#   72-22#
         72-22#    72-22#   72-22#   74-32    74-32    74-32    74-32    74-32#   74-32#   74-32#   74-32#   74-32#   77-19    77-19#
         77-20     77-20    77-20#   77-20#   77-24    77-24    77-24#   77-24#   77-24#   77-25    77-25#   95-12    95-12    95-12
         95 12     95-12    95 12#   95-12#   95-12#   95-12#   95-12#   95-12#   95-14    95-14    95-14    95-14    95-14    95-14
         95-14#    95-14#   95-14#   95-14#   95-16    95-16    95-16    95-16    95-16    95-16    95-16#   95-16#   95-16#   95-16#
         95-16#    95-18    95-18    95-18    95-18    95-18    95-18    95-18    95-18#   95-18#   95-18#   95-18#   98-20    98-20
         98-20     98-20    98-20    98-20    98-20#   98-20#   98-20#   98-20#   98-20#   98-20#   98-20#   100-3    100 3    100-3
        100-3#    100-3#   100-3#   100-3#   100-3#   104-21   104-21#  104-29   104-29#  104-29   104-29#  104-29#  104-29#  104-29#
        104-29#   104-29#  105-5    105-5    105-5    105-5    105-5#   105-5#   105-5#   105-5#   105-5#   106-14   106-14#  107-21
        107-21#   109-35   109-35   109-35   109-35   109-35#  109-35#  109-35#  109-35#  109-35#  109-54   109-54   109 54   109-54
        109-54#   109-54#  109-54#  109-54#  109-54#  110-11   110-11   110-11   110-11   110-11#  110-11#  110-11#  110-11#  110-11#
        111-12    111-12#  111-27   111-27   111-27   111-27   111-27   111-27   111-27#  111-27#  111-27#  111-27#  111-27#  111-27#
        111-30    111-30   111-30#  111-30#  111-33   111-33   111-33   111-33   111-33#  111-33#  111-33#  111 33#  111-33#  113-44
        113-44    113-44   113-44   113-44#  113-44#  113-44#  113-44#  113-44#  114-27   114-27#  114-36   114-36   114-36   114 36
        114-36#   114 36#  114-36#  114-36#  114-36#  115-5    115-5    115-5    115-5    115-5#   115-5#   115 5#   115-5#   115-5#
        116-9     116-9#   118-1    118-1    118-1    118-1    118-1    118-1    118-1    118-1    118-1#   118-1#   118-1#   118 1#
        120-10    120-10   120-10#  120-10#  120-11   120-11#  120-12   120-12   120-12#  120-12#  120-13   120 13#  120-14   120 14
        120 14#   120-14#  120-15   120-15#  120-16   120-16   120-16#  120-16#  120-17   120-17#  120 18   120-18#  120-32   120 32
        120-32#   120-32#  120-41   120-41   120-41#  120-41#  120-41#  120 42   120-42#  120-43   120 43   120-43#  120 43#  120 43#
        120-44    120 44#  120-52   120-52   120-52   120-52   120-52   120 52   120-52#  120-52#  120-52#  120-52#  120 52#  120 52#
```

	121 15	121 15	121-15#	121-15#	121-15#	121 16	121-16#	124-4	124-4	124-4#	124 4#	124-4#	124-5	124 5#
	126 12	126-12	126-12#	126-12#	126-14	126-14	126-14#	126-14#	127-3	127-3	127-3	127-3	127-3#	127-3#
	127-3#	127-3#	127-3#	127-4	127 4#	127-8	127-8	127-8	127 8	127-8#	127-8#	127-8#	127-8#	127-8#
	127 9	127-9#	127-14	127-14	127-14	127 14	127-14#	127-14#	127-14#	127-14#	127-14#	127-15	127-15#	127-20
	127 20	127-20	127-20	127-20#	127-20#	127-20#	127-20#	127-20#	127-21	127-21#	127-23	127-23#	128-12	128-12#
	129 15	129-13#	130-7	130-7#	130-8	130-8#	130-10	130 10	130-10	130-10	130-10	130-10	130-10#	130-10#
	130 10#	130-10#	130-19	130-19#	130-20	130-20#	130-21	130-21	130-21	130 21	130-21#	130 21#	130-21#	130-21#
	130 21#	130-22	130-22	130-22#	130-22#	130-27	130 27	130-27	130-27	130-27#	130 27#	130 27#	130-27#	130-27#
	130 28	130-28	130-28#	130-28#	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31	130-31#	130-31#
	130 31#	130-31#	130-32	130-32	130-32#	130-32#	130-44	130-44	130-44	130-44	130-44	130-44	130-44	130-44
	130-44#	130-44#	130-44#	130-44#	130-64	130-64	130-64	130-64	130-64	130-64	130-64#	130-64#	130-64#	130 64#
	130-64#	130-82	130-82	130-82#	130-82#	130-83	130-83#	131-14	131-14#	132 1	132-1	132 1	132-1	132-1#
	132-2	132-2	132-2	132-2	132-2#	132 3	132-3	132-3	132-3	132-3	132-3#	132-4	132-4	132-4
	132-4	132 4	132 4#	132-5	132-5	132-5	132-5	132-5	132-5#	132-6	132-6#	133-12	133-12#	134-1
	134-1	134-1	134-1#	134-2	134-2#	134-3	134-3	134-3	134-3#	134-4	134-4#	134-5	134-5	134-5
	134 5#	134-6	134-6#	134-7	134-7	134 7	134-7#	134-8	134-8#	134-9	134-9	134-9#	134-9#	134 10
	134 10#	135-8	135-8	135-8	135-8#	136-3	136-3	136-3#	136-3#					
M$GNLS	118-1	118-1#	130-10	130-10#	130-31	130-31#	130-44	130-44#						
M$GNTA	32-17	32-17#	33-14	33 14#	57-16	57-16#	57-20	57-20#	57-24	57-24#	57-28	57-28#	57-32	57-32#
	57 36	57-36#	57-40	57-40#	57-44	57-44#	57-48	57-48#	57-57	57-57#	57-62	57-62#	57 76	57-76#
	57-80	57-80#	57-84	57-84#	57-88	57-88#	57-92	57-92#	57-97	57-97#	57-101	57-101#	57 105	57-105#
	57-109	57-109#	57-113	57-113#	57-117	57-117#	57-121	57-121#	106-14	106-14#	107-21	107-21#	116-9	116-9#
	127 23	127-23#	128-12	128-12#	129-13	129-13#	130-83	130-83#	132-6	132-6#	134-10	134-10#	136 3	136-3#
	136-11	136-11#												
M$GNTE	130-5	130-5#												
M$HAPT	30-34	30-34#												
M$HNAP	30-34	30-34#												
M$INCR	30 26	30-26#	32-10	32-10	32-10#	32-10#	33-10	33-10	33-10#	33-10#	34-3	34-3#	57-14	57-14
	57-14#	57-14#	57-16#	57-18	57-18	57-18#	57-18#	57-20#	57-22	57-22	57-22#	57-22#	57-24#	57-26
	57-26	57-26#	57-26#	57-28#	57-30	57-30	57-30#	57-30#	57-32#	57-34	57-34	57-34#	57-34#	57-36#
	57-38	57-38	57-38#	57-38#	57-40#	57-42	57-42	57-42#	57-42#	57-44#	57-46	57-46	57-46#	57 46#
	57-48#	57-50	57-50	57-50#	57-50#	57-57#	57-59	57-59	57-59#	57-59#	57-62#	57-64	57-64	57 64#
	57-64#	57-76#	57-78	57-78	57-78#	57-78#	57-80#	57-82	57-82	57-82#	57-82#	57-84#	57-86	57 86
	57-86#	57-86#	57-88#	57-90	57-90	57-90#	57-90#	57-92#	57-94	57-94	57-94#	57-94#	57-97#	57-99
	57-99	57-99#	57-99#	57-101#	57-103	57-103	57-103#	57-103#	57-105#	57-107	57 107	57-107#	57-107#	57 109#
	57-111	57-111	57-111#	57-111#	57-113#	57-115	57-115	57-115#	57-115#	57-117#	57-119	57-119	57-119#	57 119#
	57 121#	59-7#	59 8#	62-11#	62-12#	62-22#	62-28#	62-30#	64-12#	65-8#	65-24#	66-36#	67 21#	68-8#
	72-22#	74-32#	77-19#	77-20#	77-24#	95-12#	95-14#	95-16#	95-18#	98-20#	100-3#	104-21#	104-29#	105 5#
	106-10	106-10	106-10#	106-10#	107-18	107-18	107-18#	107-18#	109-35#	109-54#	110-11#	111-12#	111 27#	111-30#
	111-33#	113-44#	114-27#	114-36#	115-5#	116-5	116-5	116-5#	116-5#	118-1	118-1#	118 1#	119-3	119-3#
	119-10	119-10	119-10#	119-10#	120-8	120-8	120-8#	120-8#	120-10#	120-12#	120-14#	120 16#	120-18#	120-32#
	120-41#	120-43#	120-52#	121-15#	124-4#	126-12#	126-14#	127-3#	127-4#	127-8#	127-9#	127-14#	127 15#	127-20#
	127-21#	127-23#	128-10	128-10	128-10#	128-10#	128-12#	129-8	129-8	129-8#	129-8#	129-13#	130 3	130-3#
	130-5	130-5	130-5	130-5#	130-5#	130-5#	130-7#	130-10	130-10#	130-10#	130-19#	130-21#	130-22#	130 27#
	130-28#	130-31	130-31#	130-31#	130-32#	130-44	130-44#	130-44#	130-64#	130-82#	130-83#	131-3	131-3#	131-14
	131-14	131-14#	131-14#	133-12	133-12	133-12#	133-12#	136-1	136-1#	136-3	136-3	136-3	136-3#	
M$LDRO	62-22	62-22#	77-20	77-20#	111-30	111-30#	120-10	120-10#	120-12	120-12#	120-14	120-14#	120-16	120-16#
	120-41	120-41#	120-43	120-43#	121-15	121-15#	124-4	124-4#	126-12	126-12#	130-32	130 32#		
M$MCHI	30-8	30-8#												
M$MCLO	30-8	30-8#												
M$POP	32-17	32-17#	33-14	33-14#	33-16	33-16#	57-16	57-16#	57-20	57-20#	57-24	57-24#	57-28	57-28#
	57-32	57-32#	57-36	57-36#	57-40	57-40#	57-44	57-44#	57-48	57-48#	57 57	57 57#	57-62	57-62#
	57-76	57-76#	57-80	57-80#	57-84	57-84#	57-88	57-88#	57-92	57-92#	57-97	57-97#	57-101	57-101#
	57-105	57-105#	57-109	57-109#	57-113	57-113#	57-117	57-117#	57-121	57-121#	106-14	106-14#	107-21	107-21#
	116-9	116-9#	118-104	118-104#	119-16	119-16#	127-23	127-23#	128-12	128-12#	129-13	129-13#	129-15	129-15#
	130-83	130-83#	130 84	130-84#	132-6	132-6#	134-10	134-10#	135-10	135-10#				

Cross reference table (CREF V05.00)

M$PRIN	95 12	95 12#	95-14	95 14#	95-16	95-16#	95 18	95-18#	130-64	130-64#				
M$PUSH	30-26	30-26#	32-10	32 10#	33 10	33-10#	34-3	34 3#	57-14	57-14#	57-18	57 18#	57 22	57-22#
	57-26	57-26#	57 30	57-30#	57 34	57-34#	57-38	57-38#	57 42	57-42#	57 46	57-46#	57 50	57-50#
	57 59	57 59#	57-64	57 64#	57-78	57-78#	57-82	57-82#	57-86	57-86#	57 90	57-90#	57 94	57 94#
	57 99	57-99#	57-103	57-103#	57-107	57 107#	57-111	57-111#	57-115	57-115#	57-119	57-119#	106-10	106 10#
	107 18	107-18#	116-5	116-5#	119-3	119-3#	119-10	119-10#	120-8	120-8#	128 10	128-10#	129-8	129-8#
	130 3	130-3#	130 5	130 5#	131-3	131-3#	131-14	131-14#	133-12	133 12#				
M$PUT	62-11	62-11	62 11	62-11	62-11#	95-12	95-12	95-12	95-12#	95-14	95-14	95-14	95-14#	95-16
	95 16	95-16	95-16#	95 18	95-18	95-18	95-18#	98-20	98-20	98-20	98-20	98-20#	111-27	111-27
	111 27	111-27	111-27#	120-52	120-52	120-52	120-52	120-52#	130-64	130-64	130-64	130-64#		
M$PUT1	62 11	62 11	62-11	62-11	62-11#	62-11#	62-11#	62-11#	95-12	95-12	95-12	95 12#	95-12#	95 12#
	95 14	95-14	95-14	95-14#	95-14#	95-14#	95-16	95-16	95-16	95-16#	95-16#	95-16#	95-18	95-18
	95 18	95-18#	95-18#	95-18#	98-20	98 20	98-20	98-20	98-20#	98-20#	98-20#	98-20#	111-27	111-27
	111 27	111 27	111-27#	111-27#	111-27#	111-27#	120-52	120-52	120-52	120-52	120-52#	120-52#	120-52#	120 52#
	130-64	130-64	130-64	130-64#	130-64#	130-64#								
M$RADI	118-1	118 1#	130-10	130-10#	130-31	130 31#	130-44	130-44#	132-1	132-1#	132-2	132-2#	132-3	132 3#
	132 4	132-4#	132-5	132-5#	134-1	134-1#	134-3	134-3#	134-5	134-5#	134-7	134-7#		
M$RBRO	77-24	77-24#												
M$RNRO	120 32	120-32#	120-41	120-41#	120-43	120-43#	121-15	121 15#	124-4	124-4#				
M$SETS	30-26	30-26#	32-10	32-10#	33-10	33-10#	34-3	34-3#	57-14	57-14#	57-18	57-18#	57 22	57-22#
	57 26	57-26#	57-30	57-30#	57-34	57-34#	57-38	57-38#	57-42	57-42#	57-46	57-46#	57 50	57-50#
	57-59	57-59#	57-64	57-64#	57-78	57-78#	57-82	57-82#	57-86	57-86#	57-90	57-90#	57-94	57-94#
	57-99	57-99#	57-103	57-103#	57-107	57-107#	57-111	57-111#	57-115	57-115#	57-119	57-119#	106 10	106 10#
	107 18	107-18#	116-5	116-5#	119-3	119-3#	119-10	119-10#	120-8	120-8#	128-10	128-10#	129-8	129 8#
	130-3	130-3#	130-5	130-5#	131-3	131-3#	131-14	131-14#	133-12	133-12#				
M$SVC	57 16	57-16#	57-20	57-20#	57-24	57-24#	57-28	57-28#	57-32	57-32#	57-36	57-36#	57-40	57-40#
	57-44	57-44#	57-48	57-48#	57-57	57-57#	57-62	57-62#	57-76	57-76#	57-80	57-80#	57 84	57 84#
	57-88	57 88#	57-92	57-92#	57-97	57-97#	57-101	57-101#	57-105	57-105#	57-109	57-109#	57-113	57-113#
	57 117	57-117#	57-121	57-121#	59-7	59-8	59-8#	62-11	62-11#	62-12	62-12#	62-22	62-22#	62-28
	62-30	62-30#	64-12	64-12#	65-8	65-8#	65-24	66-36	67-21	68-8	72-22	74-32	77 19	77-19#
	77-20	77-20#	77-24	77-24#	95-12	95-12#	95-14	95-14#	95-16	95-16#	95-18	95-18#	98-20	98-20#
	100-3	104-21	104-21#	104-29	105-5	109-35	109-54	110-11	111-12	111-12#	111-27	111-27#	111-30	111-30#
	111-33	113-44	114-27	114-27#	114-36	115-5	118-1	118-1#	120-10	120-10#	120-12	120-12#	120-14	120 14#
	120-16	120-16#	120-18	120-18#	120-32	120-32#	120-41	120-41#	120-43	120-43#	120-52	120-52#	121-15	121-15#
	124-4	124-4#	126-12	126-12#	126-14	126-14#	127-3	127-4	127-4#	127 8	127-9	127-9#	127-14	127-15
	127-15#	127-20	127-21	127-21#	127-23	127-23#	128-12	128-12#	129-13	129-13#	130-7	130-7#	130 10	130-10#
	130 19	130-19#	130-21	130-22	130-22#	130-27	130-28	130-28#	130-31	130-31#	130-32	130-32#	130-44	130 44#
	130-64	130-64#	130-82	130-82#	130-83	130-83#								
M$TLAB	57-16#	57-20#	57-24#	57-28#	57-32#	57-36#	57-40#	57-44#	57-48#	57-57#	57-62#	57-76#	57 80#	57-84#
	57-88#	57-92#	57-97#	57-101#	57-105#	57-109#	57-113#	57-117#	57-121#	59-7#	59-8#	62-11#	62-12#	62-22#
	62-28#	62-30#	64-12#	65-8#	65-24#	66-36#	67-21#	68-8#	72-22#	74-32#	77-19#	77-20#	77-24#	95 12#
	95-14#	95-16#	95-18#	98-20#	100-3#	104-21#	104-29#	105-5#	109-35#	109-54#	110-11#	111-12#	111-27#	111-30#
	111-33#	113-44#	114-27#	114-36#	115-5#	118-1#	120-10#	120-12#	120-14#	120-16#	120-18#	120-32#	120-41#	120 43#
	120-52#	121-15#	124-4#	126-12#	126-14#	127-3#	127-4#	127-8#	127-9#	127-14#	127-15#	127-20#	127-21#	127-23#
	128-12#	129-13#	130-7#	130-10#	130-19#	130-21#	130-22#	130-27#	130-28#	130-31#	130-32#	130 44#	130-64#	130-82#
	130-83#													
M$TSTL	57 16	57-16#	57-20	57-20#	57-24	57-24#	57-28	57-28#	57 32	57-32#	57-36	57-36#	57-40	57 40#
	57-44	57-44#	57-48	57-48#	57-57	57-57#	57-62	57-62#	57-76	57-76#	57-80	57-80#	57-84	57-84#
	57 88	57-88#	57-92	57-92#	57-97	57-97#	57-101	57-101#	57-105	57-105#	57-109	57-109#	57-113	57 113#
	57 117	57-117#	57-121	57-121#	59-7	59-7#	59-7#	59-8	59-8#	62-11	62-11#	62-12	62-12#	62-22
	62-22#	62-28	62-28#	62-28#	62-30	62-30#	64-12	64-12#	65-8	65-8#	65-24	65-24#	65-24#	66 36
	66-36#	66-36#	67-21	67-21#	67-21#	68-8	68-8#	68-8#	72-22	72-22#	72-22#	74-32	74-32#	74-32#
	77-19	77-19#	77-20	77-20#	77-24	77-24#	95-12	95-12#	95-14	95-14#	95-16	95-16#	95 18	95-18#
	98-20	98-20#	100-3	100-3#	100-3#	104-21	104-21#	104-29	104-29#	104-29#	105-5	105-5#	105-5#	109-35
	109-35#	109-35#	109-54	109-54#	109-54#	110-11	110-11#	110-11#	111-12	111-12#	111-27	111-27#	111 30	111-30#
	111-33	111-33#	111 33#	113-44	113-44#	113-44#	114-27	114 27#	114 36	114-36#	114-36#	115-5	115 5#	115-5#

Cross reference table (CREF V05.00)