RA80 PDP-11 UDA DRV FMTR AH-S837C-MC JAN 1983
CZUDECO FICHE 1 OF 1
MADE IN USA

```
.REM ~
```

IDENTIFICATION

PRODUCT CODE: AC-S836C-MC

PRODUCT NAME: CZUDECO UDA DISK FORMATTER

PRODUCT DATE: 27-AUG-82

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: MATT TEDONE

.REM ~

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

This program will format any disk drive connected to a UDA-50 disk controller. 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 UDA-50 disk controller. The host program simply downline loads the DUP program in the UDA-50 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 UDA-50 can only format one disk at a time, so each disk on a UDA-50 will be selected sequentially. If the disk drives to be formatted are connected to different UDA-50s, all UDA-50s 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 UDA-50 and unit 3 is connected to a different UDA-50 (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 UDA-50 subsystems
XXDP+ load media containing this program and the ZUDGA0.PAK data file.

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.

The diagnostic program requires that the data file ZUDGA0.PAK be on the XXDP+ system device. This data file is ordered under the name CZUDEB0. The XXDP+ system device must remain on-line during the execution of the this diagnostic.

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

Once the program is started, 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 followng question will be asked for each disk of be formatted 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 t dious 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 (D) ?  8<CR>

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

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

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

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

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

UNIT 6
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  5<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 7
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  6<CR>
Q-FACTOR (O) 0 ?  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) 0 ?  1,0<CR>

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

UNIT 7
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  6,7<CR>
Q-FACTOR (O) 0 ?  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 Services 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 ZUDEBO"
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"

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 UDA-50:

```
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
```

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

UNIT 0 UDA AT 172150 DRIVE 0   RUNTIME 0:12:20
Format completed
     2 Revectored LBNS
     2 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     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 0:25:18
Format completed
     6 Revectored LBNS
     5 Primary revectored LBNS
     1 Secondary/tertiary revectored LBNS
     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
     4 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

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

  RUNTIME 0:12:45
Format completed
     2 Revectored LBNS
     2 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     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 UDA-50 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 gocd/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 UDA-50 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 UDA-50 being tested. It may be omitted if the error is not specific to one UDA-50.

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 M7161                                           /
```

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, UDA-50 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 UDA-50 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 UDA-50 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 UDA-50, the UDA-50 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 UDA-50 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.

00011 CZUDE DVC FTL ERR 00011 ON UNIT 00 TST 001 SUB 000 PC: xxxxxx
HOST PROGRAM UDA AT 172150 RUNTIME x:xx:xx
WRONG APT FORMATTER IS BEING USED WITH THIS CONTROLLER.
USE CIUDx.

There are two APT compatible formatters. CIUDK is used for UDA-50s with the M7161-2 modules. CIUDI is used for UDA-50s with the M7485-6 modules. If one formatter program is used with the wrong UDA-50 module set, this error will appear. Use the other APT formatter program to successfully format.

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 M7161 OR M7485
OR UNIBUS
OR REPLACE UDA MODULE M7161 OR 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 swiiches) or module M7161 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 x  :xx
        UDA RESIDENT DIAGNOSTICS DETECTED FAILURE
          UDASA CONTAINS  105154
        REPLACE UDA MODULE M7162 OR 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. M7161 and M7485 are the Unibus interface boards. M7162 and M7486 are the SDI interface boards. Module M7161 only works with the module M7182. The same for modules M7485 and M7486. Do not intermix the two boards. It will not work!

```
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 M7161 OR 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 M7161 OR 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 M7161 module or 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 M7161 OR 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 M7161 or 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 00000C	000000 000000
000000 000000	000000 000000
000000 000000	000000 000000

The 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 M7161 OR 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 172150 RUNTIME x:xx:xx
DUP PROGRAM REJECTED ANSWER TO DATE OR SERIAL NUMBER QUESTION

After the cperator 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 M7161 or M7485.

Formatter initialization error

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

Nonexistent 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 M7161 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.

An example of how the errors are presented is below:

```
RUNTIME  0:00:18
Nonexistant 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 in message #2 which were primary revectors.

n Secondary/teritary revectored LBNS

Where n is the number of the LBNs in message #2 which were secondary or tertiary revectors.

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 0:24:57
Format completed
     5 Revectored LBNS
     5 Primary revectored LBNS
     0 Secondary/tertiary revectored LBNS
     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 UDA-50.

PROGRAM HEADER

```
                                .SBTTL  PROGRAM HEADER

                                        BGNMOD
                                ;++
                                ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                                ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                                ;--

                                        POINTER BGNSW, BGNSFT, BGNSETUP

002000                          HEADER  CZUDE,C,0,0,1,PRI07     ;FIELD SERVICE
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  103                                                             .ASCII  /C/
002011                                                                  L$DEPO::
002011  060                                                             .ASCII  /0/
002012                                                                  L$UNIT::
002012  000001                                                          .WORD   T$PTHV
002014                                                                  L$TIML::
002014  000000                                                          .WORD   0
002016                                                                  L$HPCP::
002016  022762                                                          .WORD   L$HARD
002020                                                                  L$SPCP::
002020  023150                                                          .WORD   L$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$PRIO::
002042  000340                                                          .WORD   PRI07
002044                                                                  L$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                  L$EF::
                                        .WORD   0
002054  000000                  L$SPC::
                                        .WORD   0
002056
002056  000000                  L$DEVP::
                                        .WORD   L$DVTYP
002060
002060  003514                  L$REPP::
                                        .WORD   0
002062
002062  000000                  L$EXP4::
                                        .WORD   0
002064
002064  000000                  L$EXP5::
                                        .WORD   0
002066
002066  000000                  L$AUT::
                                        .WORD   0
002070
002070  000000                  L$DUT::
                                        .WORD   0
002072
002072  000000                  L$LUN::
                                        .WORD   0
002074
002074  000000                  L$DESP::
                                        .WORD   L$DESC
002076
002076  003536                  L$LOAD::
                                        EMT     E$LOAD
002100
002100  104035                  L$ETP::
                                        .WORD   0
002102
002102  000000                  L$ICP::
                                        .WORD   L$INIT
002104
002104  020376                  L$CCP::
                                        .WORD   L$CLEAN
002106
002106  022072                  L$ACP::
                                        .WORD   L$AUTO
002110
002110  022070                  L$PRT::
                                        .WORD   L$PROT
002112
002112  020370                  L$TEST::
                                        .WORD   0
002114
002114  000000                  L$DLY::
                                        .WORD   0
002116
002116  000000                  L$HIME::
                                        .WORD   0
002120
002120  000000
```

DISPATCH TABLE

```
                                .SBTTL  DISPATCH TABLE

                                ;++
                                ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                                ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                                ;--

002122                          DISPATCH 1
002122  000001                                                         .WORD   1
002124                                                          L$DISPATCH::
002124  022102                                                          .WORD   T1
```

```
                                .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 DR!VE 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                  PRI03== 140
                000100                  PRI02== 100
                000040                  PRI01== 40
                000000                  PRI00== 0
                                        ;
                                        ;OPERATOR FLAG BITS
                                        ;
                000004                  EVL==       4
                000010                  LOT==      10
                000020                  ADR==      20
                000040                  IDU==      40
                000100                  ISR==     100
                000200                  UAM==     200
                000400                  BOE==     400
                001000                  PNT==    1000
                002000                  PRI==    2000
                004000                  IXE==    4000
                010000                  IBE==   10000
                020000                  IER==   20000
                040000                  LOE==   40000
                100000                  HOE==  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
```

GLOBAL EQUATES SECTION

```
                        ;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
```

GLOBAL EQUATES SECTION

```
                        ;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 POLE 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
```

GLOBAL EQUATES SECTION

```
                        ;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
000122                  HC.BSZ= 82.                     ;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
000306                  HC.BF2= HC.BF1+HC.BSZ           ;SECOND BUFFER

000430                  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 HC.MCT	MESSAGE RING	4 BYTES
HC.CMD HC.CCT	COMMAND RING	4 BYTES
HC.MEV HC.MPK	MESSAGE ENVELOPE	52 BYTES
HC.CEV HC.CPK	COMMAND ENVELOPE	52 BYTES
HC.BF1	BUFFER # 1 (RESPONSE TO DM PROGRAM)	82 BYTES
HC.BF2	BUFFER # 2 (REQUEST FROM DM PROGRAM)	82 BYTES

```
                        ;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. FOE 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
```

GLOBAL EQUATES SECTION

```
                        ;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
```

GLOBAL EQUATES SECTION

```
                        ;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.HSTI= 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.HSTI= 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
```

```
                        ;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
        000400                          CT.U52= BIT8                    ; UDA52 BIT
        001000                          CT.U50= BIT9                    ; UDA50 BIT
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 EXISTS
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

002146                                  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, PNTX, 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  000000          DMPROG: .WORD 0                 ;START ADDRESS OF DM PROGRAM
002166  000000          DMSIZE: .WORD 0                 ;SIZE OF DM PROGRAM
002170                  DMEND:  .BLKW 1                 ;END ADDRESS OF DM PROGRAM(FIRST FREE MEMORY ADR)
002172                  DMENDS: .BLKW 1                 ;FREE MEMORY SIZE FROM END OF DM PROGRAM
002174                  URUN:   .BLKW 1                 ;NUMBER OF UNITS TO RUN AT ONE TIME
002176                  URNING: .BLKW 1                 ;NUMBER OF UNITS STILL RUNNING
002200                  UCNT:   .BLKW 1                 ;COUNTER OF UNITS UNDER TEST
002202  000000          FILOPN: .WORD 0                 ; FILE OPEN
002204  000000          U50UNI: .WORD   0               ; KEEPS TRACK OF ANY UDA 50 CONTROLLERS
002206     125  065  062 U52EXT: .ASCII\U52\
                            .EVEN
002212  000000          TSTTYP: .WORD 0                 ; CONTROLLER TYPE
002214  000000          FNUM:   .WORD 0                 ; FILE NUMBER
002216  000000          INMEM:  .WORD 0                 ; WHICH DM PROG IS IN MEMORY
002220                  UFREEZ: .BLKW 1                 ;FREEZE ON UNIT WHEN NOT ZERO
002222                  NXMAD:  .BLKW 1                 ;SET TO ALL ONES BY NON-EXISTANT ADDRESS
002224  000000          FDATA:  .WORD 0
002226                  FCTBUF: .BLKB 512.              ;STORAGE FOR FCT BLOCK
003226                  FCTNUM: .BLKW 1         ;FCT BLOCK NUMBER
003230                  MODE:   .BLKW 1 ;MODE WORD, SAME BIT DEFS AS SO.BIT

                        ;CLOCK CONTROL

003232  000000          KW.CSR: .WORD 0                 ;CSR OF CLOCK
003234                  KW.BRL: .BLKW 1                 ;BR LEVEL
003236                  KW.VEC: .BLKW 1                 ;VECTOR
003240                  KW.HZ:  .BLKW 1                 ;HERTZ (50. OR 60.)
003242                  KW.EL:  .BLKW 2                 ;ELAPSED TIME

003246  016106          PTYPE:  .WORD PF                ;PRINT TYPE
003250     000  000     ERRCHR: .BYTE 0,0               ;FIRST BYTE LOADED WITH OUTPUT CHARACTER
003252  000000          NULL:   .WORD 0                 ;USED TO PRINT A NULL CHARACTER
003254                  FNAME2: .BLKB 10.
003266     132  125  104 FNAME:  .ASCIZ\ZUDGA0.PAK\
                            .EVEN
```

GLOBAL DATA SECTION

```
      1 003302                                TEMP:   .BLKB 22.                       ;USED TO GET ANSWER FROM GMANID CALL
      2 003330      061       055       112   DATEI:  .ASCIZ\1-JAN-70\                ;DEFAULT DATE
      3 003341                                        .BLKB 3
      4 003344   000000                         DATEO:  .WORD 0 ;DATE STRING IN FORMATTER FORMAT
      5 003346                                        .BLKB 10.       ;(FIRST WORD ZERO SAYS NO DATE HERE YET)
      6 003360      061       070       064   HIGHEST:.ASCIZ\18446744073709551615\    ;HIGHEST DISK SERIAL NUMBER
      7 003405      104       105       103   MONTHS: .ASCII\DEC\     ;NAME OF MONTHS
      8 003410      116       117       126           .ASCII\NOV\
      9 003413      117       103       124           .ASCII\OCT\
     10 003416      123       105       120           .ASCII\SEP\
     11 003421      101       125       107           .ASCII\AUG\
     12 003424      112       125       114           .ASCII\JUL\
     13 003427      112       125       116           .ASCII\JUN\
     14 003432      115       101       131           .ASCII\MAY\
     15 003435      101       120       122           .ASCII\APR\
     16 003440      115       101       122           .ASCII\MAR\
     17 003443      106       105       102           .ASCII\FEB\
     18 003446      112       101       116           .ASCII\JAN\
     19 003451      037                         DAYS:   .BYTE 31.       ;NUMBER OF DAYS IN EACH MONTH
     20 003452      035                                 .BYTE 29.
     21 003453      037                                 .BYTE 31.
     22 003454      036                                 .BYTE 30.
     23 003455      037                                 .BYTE 31.
     24 003456      036                                 .BYTE 30.
     25 003457      037                                 .BYTE 31.
     26 003460      037                                 .BYTE 31.
     27 003461      036                                 .BYTE 30.
     28 003462      037                                 .BYTE 31.
     29 003463      036                                 .BYTE 30.
     30 003464      037                                 .BYTE 31.
     31 003465      061       071       000   YEAR19: .ASCIZ\19\
     32 003470      062       060       000   YEAR20: .ASCIZ\20\
     33                                              .EVEN
     34 003474   000000                         IPADRS: .WORD 0
     35 003476   000000                                 .WORD 0
     36 003500   000000                                 .WORD 0
     37 003502   000000                                 .WORD 0
     38 003504   000000                                 .WORD 0
     39 003506   000000                                 .WORD 0
     40 003510   000000                                 .WORD 0
     41 003512   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
                        ;
 003514                         DEVTYP <UDA-50 CONTROLLER>
 003514
 003514  125  104  101                                          L$DVTYP::
                                                                .ASCIZ  /UDA-50 CONTROLLER/
                                                                .EVEN

                        ; TEST DESCRIPTION
                        ;
 003536                         DESCRIPT <CZUDECO UDA-50 DISK DRV FORMATTER>
 003536
 003536  103  132  125                                          L$DESC::
                                                                .ASCIZ  /CZUDECO UDA-50 DISK
                                                                .EVEN
```

GLOBAL TEXT SECTION

```
      1                                 ;UNFORMATTED MESSAGES
      2
      3 003600      105     116     124 DATEQ:  .ASCIZ\ENTER DATE AS DD-MMM-YY\
      4 003630      040     106     117 FILNAQ: .ASCIZ\ FOR DISK TO BE FORMATTED\
      5 003662      040     000         SERNQ:  .ASCIZ\ \
```

GLOBAL TEXT SECTION

```
                                        ; FORMAT STATEMENTS USED IN PRINT CALLS
                                        
003664      045     124     000 ERRONE: .ASCIZ\%T\
003667      045     116     000 ERRNL:  .ASCIZ\%N\
003672      042     040     040 RNTIM:  .ASCIZ\"  RUNTIME "D16":"\
003715      104     071     042 RNTIM1: .ASCIZ\D9":"\
003723      104     071     000 RNTIM2: .ASCIZ\D9\
003726      042     040     040 ERRME1: .ASCIZ\"  *  *  *  ERROR PROCESSING MESSAGE STRING  *  *  *"N\
004015      116     042     125 MESSG:  .ASCIZ\N"UNIT "D6" UDA AT "O16" DRIVE "D9S\
                                        ;;NOCLOCK:.ASCIZ\"NO LINE CLOCK AVAILABLE"N\
004061      042     116     117 NOCLOCK:.ASCIZ\"NO LINE CLOCK AVAILABLE FOR TIMING EVENTS"N\
004136      042     110     117 BASNO:  .ASCIZ\"HOST PROGRAM"\
004155      042     040     040 BASL2:  .ASCIZ\"  UDA AT "O16\
004174      042     040     040 BASL3:  .ASCIZ\"  DRIVE "D9\
004211      000                 BAS:    .BYTE 0                                 ;NULL TO PRINT NOTHING
                                        
004212      122     066     122 BASLN:  .ASCIZ\R6R6R6R6\                        ;USED TO PRINT BASIC LINE OF ERROR MESSAGE
004223      116     042     123 SERNUM: .ASCIZ\N"SERIAL NUMBER FOR UNIT "D6" UDA AT "O16" DRIVE "D9\
```

GLOBAL TEXT SECTION

```
      1 004310                          X1A:
      2 004310                          X2A:
      3 004310                          X3A:
      4 004310    042     111     040   X8A:    .ASCIZ\"I DON'T LIKE THE ANSWERS YOU GAVE TO THE HARDWARE QUESTIONS'N\
      5 004407    122     065     122   X1:     .ASCIZ\R5R6"UDA HAS MORE THAN ONE VECTOR, BR LEVEL OR BURST RATE'N\
      6 004503    122     065     122   X2:     .ASCIZ\R5R6"TWO UNITS SELECT THE SAME DRIVE'N\
      7 004552    122     065     122   X3:     .ASCIZ\R5R6"MORE THAN EIGHT DRIVES SELECTED ON THIS UDA'N\
      8 004635    122     064     042   X4:     .ASCII\R4"NOT ENOUGH ROOM IN MEMORY TO TEST THE UNITS SELECTED'N\
      9 004726    042     120     114           .ASCIZ\"PLEASE START PROGRAM OVER AND TEST FEWER UNITS AT A TIME'N\
     10 005022    122     064     042   X5:     .ASCIZ\R4"CHECKSUM ERROR IN DM PROGRAM FILE 'N\
     11 005072    122     064     042   X7:     .ASCIZ\R4"ERROR IN DM PROGRAM FILE. DM PROGRAM NOT FOUND'N\
     12 005156    122     065     122   X8:     .ASCIZ\R5R6"TWO UDA'S USE THE SAME VECTOR'N\
     13 005223    122     064     042   X9:     .ASCII\R4"ONLY ONE DISK CAN BE SELECTED IN HW QUESTIONS IN RESTORE MODE.'N\
     14 005326    042     120     114           .ASCIZ\"PLEASE START PROGRAM OVER AND SELECT ONLY ONE DISK.'N\
     15 005415    122     064     042   X10:    .ASCIZ\R4"THIS PROGRAM CAN ONLY REFORMAT A DISK IN UNATTENDED MODE.'N\
     16 005514    122     065     042   X11:    .ASCII\R5"WRONG APT FORMATTER IS BEING USED WITH THIS CONTROLLER.'N\
     23 005610    122     065     042   X20:    .ASCII\R5"MEMORY ERROR TRYING TO READ UDA REGISTERS'N\
     24 005666    042     103     110           .ASCII\"CHECK UNIBUS SELECTION SWITCHES ON UDA MODULE M7161 OR M7485'N\
     25 005765    042     117     122           .ASCII\"OR UNIBUS'N\
     26 006001    042     117     122           .ASCIZ\"OR "R7\
     27 006011    122     065     042   X21:    .ASCII\R5"UDA RESIDENT DIAGNOSTICS DETECTED FAILURE'NR8\
     28 006071    042     122     105           .ASCIZ\"REPLACE UDA MODULE M716"02" OR M748"03N\
     29 006142    122     065     042   X22:    .ASCII\R5"STEP BIT DID NOT SET IN UDASA REGISTER DURING INITIALIZATION'N\
     30 006243    042     123     124           .ASCIZ\"STEP BIT EXPECTED "016NR8R7\
     31 006300    122     065     042   X23A:   .ASCII\R5"UDA DID NOT CLEAR RING STRUCTURE IN HOST MEMORY DURING INITIALIZATION'N\
     32 006412    104     071     042           .ASCII\D9" WORDS WERE TO BE CLEARED STARTING AT ADDRESS "016N\
     33 006500    042     106     111           .ASCII\"FIRST SEVERAL WORDS NOT CLEARED (UP TO 6):'N\
     34 006555    123     066     042           .ASCIZ\S6"ADDRESS"S4"CONTENTS'N\
     35 006606    123     067     117   X23B:   .ASCIZ\S7016S5016N\
     36 006622    122     065     042   X24:    .ASCII\R5"UDASA REGISTER DID NOT GO TO ZERO AFTER STEP 3 WRITE OF INITIALIZATION'N\
     37 006735    042     120     125           .ASCIZ\"PURGE/POLE DIAGNOSTICS WERE REQUESTED'NR8R7\
     38 007012    122     065     042   X25:    .ASCII\R5"UDA DID NOT RETURN CORRECT DATA IN UDASA REGISTER DURING INITIALIZATION'N\
     39 007126    042     040     040           .ASCIZ\"  UDASA EXPECTED  "016NR8R7\
     40 007163    122     065     042   X30:    .ASCIZ\R5"UDA REPORTED FATAL ERROR IN UDASA REGISTER WHILE RUNNING DM PROGRAM'NR8\
     41 007276    122     065     042   X31:    .ASCIZ\R5"DUP PROGRAM IS HUNG'N\
     42 007327    122     065     042   X32:    .ASCIZ\R5"MESSAGE BUFFER RECEIVED FROM DM PROGRAM WITH UNKNOWN REQUEST NUMBER'N\
     43 007440    122     065     042   X36:    .ASCII\R5"NO INTERRUPT RECEIVED FROM UDA FOR 30 SECONDS'N\
     44 007522    042     127     110           .ASCIZ\"WHILE LOADING DM PROGRAM'N\
     45 007556    122     065     042   X37:    .ASCIZ\R5"UDA REPORTED FATAL ERROR IN UDASA REGISTER WHILE LOADING DM PROGRAM'NR8R7\
     46 007673    122     065     042   X100:   .ASCIZ\R5"DUP PROGRAM ASKED UNEXPECTED QUESTION ("D12")'N\
     47 007756    122     065     042   X101:   .ASCIZ\R5"DUP PROGRAM REJECTED ANSWER TO DATE OR SERIAL NUMBER QUESTION'N\
```

GLOBAL TEXT SECTION

```
      1 010061     042     115     105  XMSG1:   .ASCIZ\''MESSAGE BUFFER CONTAINS:''N\
      2 010115     123     063     117  XMSG2:   .ASCIZ\S3016S1016S1016S1016S1016S1016S1016N\
      3 010162     122     065     042  XPKT1:   .ASCII\R5''RESPONSE PACKET FROM UDA DOES NOT CONTAIN EXPECTED DATA''N\
      4 010256     042     105     111           .ASCII\''EITHER UDA RETURNED ERROR STATUS OR PACKET WAS NOT RECEIVED CORRECTLY''N\
      5 010366     123     063     042           .ASCIZ\S3''COMMAND PACKET SENT''S6''RESPONSE PACKET RECEIVED''N\
      6 010453     123     066     117  XPKT2:   .ASCIZ\S6016S1016S14016S1016N\
      7 010502     042     040     040  XSA:     .ASCIZ\''  UDASA CONTAINS  ''016N\
      8 010533     042     122     105  XFRU:    .ASCIZ\''REPLACE UDA MODULE M7161 OR M7485''N\
      9
     10
     11 010600     045     101     111  SERNX:   .ASCIZ\%AINPUT ERROR. ANSWER WITH DECIMAL NUMBER LO= 0  HI= %T\
     12 010670     042     111     116  DATEX:   .ASCIZ\''INPUT ERROR.''\
     13 010707     042     116     101  FILNAM:  .ASCIZ\''NAME OF FILE CONTAINING BAD SECTOR INFORMATION''\
     14                                         .EVEN
```

GLOBAL ERROR REPORT SECTION

```
                                .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

 010770                         BGNMSG ERR001
 010770                                 PNTB X1,#X1A
 010770  012746  004310                                             MOV #X1A,-(SP)
 010774  004137  016240                                             JSR R1,LPNTB
 011000  004407                                                     .WORD X1
 011002  000002                                                     .WORD PNT.CT
 011004                         ENDMSG

 011006                         BGNMSG ERR002
 011006                                 PNTB X2,#X2A
 011006  012746  004310                                             MOV #X2A,-(SP)
 011012  004137  016240                                             JSR R1,LPNTB
 011016  004503                                                     .WORD X2
 011020  000002                                                     .WORD PNT.CT
 011022                         ENDMSG

 011024                         BGNMSG ERR003
 011024                                 PNTB X3,#X3A
 011024  012746  004310                                             MOV #X3A,-(SP)
 011030  004137  016240                                             JSR R1,LPNTB
 011034  004552                                                     .WORD X3
 011036  000002                                                     .WORD PNT.CT
 011040                         ENDMSG

 011042                         BGNMSG ERR004
 011042                                 PNTB X4
 011042  004137  016240                                             JSR R1,LPNTB
 011046  004635                                                     .WORD X4
 011050  000000                                                     .WORD PNT.CT
 011052                         ENDMSG

 011054                         BGNMSG ERR005
 011054                                 PNTB X5
 011054  004137  016240                                             JSR R1,LPNTB
 011060  005022                                                     .WORD X5
 011062  000000                                                     .WORD PNT.CT
 011064                         ENDMSG

 011066                         BGNMSG ERR007
 011066                                 PNTB X7
 011066  004137  016240                                             JSR R1,LPNTB
 011072  005072                                                     .WORD X7
 011074  000000                                                     .WORD PNT.CT
 011076                         ENDMSG
```

GLOBAL ERROR REPORT SECTION

```
    37
    38 011100                           BGNMSG ERR008
    39 011100                                  PNTB X8,#X8A
       011100  012746  004310                                        MOV #X8A,-(SP)
       011104  004137  016240                                        JSR R1,LPNTB
       011110  005156                                                .WORD X8
       011112  000002                                                .WORD PNT.CT
    40 011114                           ENDMSG
    41
    42 011116                           BGNMSG ERR009
    43 011116                                  PNTB X9
       011116  004137  016240                                        JSR R1,LPNTB
       011122  005223                                                .WORD X9
       011124  000000                                                .WORD PNT.CT
    44 011126                           ENDMSG
    45
    46 011130                           BGNMSG ERR010
    47 011130                                  PNTB X10
       011130  004137  016240                                        JSR R1,LPNTB
       011134  005415                                                .WORD X10
       011136  000000                                                .WORD PNT.CT
    48 011140                           ENDMSG
    49
    50 011142                           BGNMSG ERR011
    51 011142                                  PNTB X11
       011142  004137  016240                                        JSR R1,LPNTB
       011146  005514                                                .WORD X11
       011150  000000                                                .WORD PNT.CT
    52 011152                           ENDMSG
    53
    54 011154                           BGNMSG ERR020
    55 011154                                  PNTB X20
       011154  004137  016240                                        JSR R1,LPNTB
       011160  005610                                                .WORD X20
       011162  000000                                                .WORD PNT.CT
    56 011164                           ENDMSG
    57
    58 011166                           BGNMSG ERR021
    59 011166  010201                          MOV R2,R1
    60 011170  000301                          SWAB R1
    61 011172                                  AND 2,R1
       011172  042701  177775                                        BIC #^C<2>,R1
    62 011176  006201                          ASR R1
    63 011200  005201                          INC R1
    64 011202                                  PNTB X21,R2,R1
       011202  010146                                                MOV R1,-(SP)
       011204  010246                                                MOV R2,-(SP)
       011206  004137  016240                                        JSR R1,LPNTB
       011212  006011                                                .WORD X21
       011214  000004                                                .WORD PNT.CT
    65 011216                           ENDMSG
    66
    67 011220                           BGNMSG ERR022
    68 011220  042737  100000  020152          BIC #SA.ERR,UDARSD
    69 011226                                  PNTB X22,UDARSD,R2
       011226  010246                                                MOV R2,-(SP)
       011230  013746  020152                                        MOV UDARSD,-(SP)
```

GLOBAL ERROR REPORT SECTION

```
        011234  004137  016240                                                  JSR R1,LPNTB
        011240  006142                                                           .WORD X22
        011242  000004                                                           .WORD PNT.CT
    70  011244                          PRINTX #XFRU
    71  011264                      ENDMSG
    72
    73  011266                      BGNMSG ERR023
    74  011266                          PNTB X23A,R1,FFREE
        011266  013746  002146                                                  MOV FFREE,-(SP)
        011272  010146                                                          MOV R1,-(SP)
        011274  004137  016240                                                  JSR R1,LPNTB
        011300  006300                                                          .WORD X23A
        011302  000004                                                          .WORD PNT.CT
    75  011304  005742                      TST -(R2)
    76  011306  005712                  ERR23A: TST (R2)
    77  011310  001410                      BEQ ERR23B
    78  011312                          PNTB X23B,R2,(R2)
        011312  011246                                                          MOV (R2),-(SP)
        011314  010246                                                          MOV R2,-(SP)
        011316  004137  016240                                                  JSR R1,LPNTB
        011322  006606                                                          .WORD X23B
        011324  000004                                                          .WORD PNT.CT
    79  011326  005304                      DEC R4
    80  011330  001403                      BEQ ERR23C
    81  011332  005722                  ERR23B: TST (R2)+
    82  011334  005303                      DEC R3
    83  011336  001363                      BNE ERR23A
    84  011340                          ERR23C: PNTB XFRU
        011340  004137  016240                                                  JSR R1,LPNTB
        011344  010533                                                          .WORD XFRU
        011346  000000                                                          .WORD PNT.CT
    85  011350                      ENDMSG
    86
    87  011352                      BGNMSG ERR024
    88  011352                          PNTB X24,R2
        011352  010246                                                          MOV R2,-(SP)
        011354  004137  016240                                                  JSR R1,LPNTB
        011360  006622                                                          .WORD X24
        011362  000002                                                          .WORD PNT.CT
    89  011364                      ENDMSG
    90
    91  011366                      BGNMSG ERR025
    92  011366                          PNTB X25,R1,R2
        011366  010246                                                          MOV R2,-(SP)
        011370  010146                                                          MOV R1,-(SP)
        011372  004137  016240                                                  JSR R1,LPNTB
        011376  007012                                                          .WORD X25
        011400  000004                                                          .WORD PNT.CT
    93  011402                      ENDMSG
    94
    95  011404                      BGNMSG ERR030
    96  011404                          PNTB X30,R1
        011404  010146                                                          MOV R1,-(SP)
        011406  004137  016240                                                  JSR R1,LPNTB
        011412  007163                                                          .WORD X30
        011414  000002                                                          .WORD PNT.CT
    97  011416                      ENDMSG
```

```
 98
 99 011420                          BGNMSG ERR031
100 011420                                 PNTB X31
    011420  004137  016240                                        JSR R1,LPNTB
    011424  007276                                                 .WORD X31
    011426  000000                                                 .WORD PNT.CT
101 011430                          ENDMSG
102
103 011432                          BGNMSG ERR032
104 011432                                 PNTB X32
    011432  004137  016240                                        JSR R1,LPNTB
    011436  007327                                                 .WORD X32
    011440  000000                                                 .WORD PNT.CT
105 011442  004737  011632                   CALL MSGPKT
106 011446                          ENDMSG
107
108 011450                          BGNMSG ERR033
109 011450  004737  011540                   CALL PNTPKT
110 011454                          ENDMSG
111
112 011456                          BGNMSG ERR034
113 011456  004737  011540                   CALL PNTPKT
114 011462                          ENDMSG
115
116 011464                          BGNMSG ERR036
117 011464                                 PNTB X36
    011464  004137  016240                                        JSR R1,LPNTB
    011470  007440                                                 .WORD X36
    011472  000000                                                 .WORD PNT.CT
118 011474                          ENDMSG
119
120 011476                          BGNMSG ERR037
121 011476                                 PNTB X37,R1
    011476  010146                                                 MOV R1,-(SP)
    011500  004137  016240                                        JSR R1,LPNTB
    011504  007556                                                 .WORD X37
    011506  000002                                                 .WORD PNT.CT
122 011510                          ENDMSG
123
124 011512                          BGNMSG ERR100
125 011512                                 PNTB X100,(R4)
    011512  011446                                                 MOV (R4),-(SP)
    011514  004137  016240                                        JSR R1,LPNTB
    011520  007673                                                 .WORD X100
    011522  000002                                                 .WORD PNT.CT
126 011524                          ENDMSG
127
128 011526                          BGNMSG ERR101
129 011526                                 PNTB X101
    011526  004137  016240                                        JSR R1,LPNTB
    011532  007756                                                 .WORD X101
    011534  000000                                                 .WORD PNT.CT
130 011536                          ENDMSG
131
132 011540                          PNTPKT: PNTB XPKT1
    011540  004137  016240                                        JSR R1,LPNTB
    011544  010162                                                 .WORD XPKT1
```

GLOBAL ERROR REPORT SECTION

```
                011546  000000                                                          .WORD PNT.CT
    133         011550  010401                      MOV R4,R1
    134         011552  062701  000104              ADD #HC.CPK,R1
    135         011556  010402                      MOV R4,R2
    136         011560  062702  000020              ADD #HC.MPK,R2
    137         011564  012703  000014              MOV #12.,R3
    138         011570                      PNTPKL: PNTB XPKT2,2(R1),(R1),2(R2),(R2)
                011570  011246                                                          MOV (R2),-(SP)
                011572  016246  000002                                                  MOV 2(R2),-(SP)
                011576  011146                                                          MOV (R1),-(SP)
                011600  016146  000002                                                  MOV 2(R1),-(SP)
                011604  004137  016240                                                  JSR R1,LPNTB
                011610  010453                                                          .WORD XPKT2
                011612  000010                                                          .WORD PNT.CT
    139         011614  062701  000004              ADD #4,R1
    140         011620  062702  000004              ADD #4,R2
    141         011624  005303                      DEC R3
    142         011626  001360                      BNE PNTPKL
    143         011630  000207                      RETURN
    144
    145         011632                      MSGPKT: PNTB XMSG1
                011632  004137  016240                                                  JSR R1,LPNTB
                011636  010061                                                          .WORD XMSG1
                011640  000000                                                          .WORD PNT.CT
    146         011642  016504  000016              MOV C.RING(R5),R4
    147         011646  062704  000306              ADD #HC.BF2,R4
    148         011652  012703  000005              MOV #5,R3
    149         011656                      MSGPKL: PNTB XMSG2,(R4),2(R4),4(R4),6(R4),8.(R4),10.(R4),12.(R4)
                011656  016446  000014                                                  MOV 12.(R4),-(SP)
                011662  016446  000012                                                  MOV 10.(R4),-(SP)
                011666  016446  000010                                                  MOV 8.(R4),-(SP)
                011672  016446  000006                                                  MOV 6(R4),-(SP)
                011676  016446  000004                                                  MOV 4(R4),-(SP)
                011702  016446  000002                                                  MOV 2(R4),-(SP)
                011706  011446                                                          MOV (R4),-(SP)
                011710  004137  016240                                                  JSR R1,LPNTB
                011714  010115                                                          .WORD XMSG2
                011716  000016                                                          .WORD PNT.CT
    150         011720  062704  000016              ADD #14.,R4
    151         011724  005303                      DEC R3
    152         011726  001353                      BNE MSGPKL
    153         011730  000207                      RETURN
```

GLOBAL ERROR REPORT SECTION

```
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

011732                          FMERR:  ERRSF 4,,ERR004
011732  104454                                                          TRAP    C$ERSF
011734  000004                                                          .WORD   4
011736  000000                                                          .WORD   0
011740  011042                                                          .WORD   ERR004
011742                                  DOCLN                   ;ABORT
011742  104444                                                          TRAP    C$DCLN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;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.
                                        ;
011744                                  ALOCM:  PUSH FFREE                      ;SAVE FFREE AT ENTRY
011744  013746  002146                                                                  MOV FFREE,-(SP)
011750  160137  002150                          SUB R1,FSIZE                    ;REDUCE SIZE OF FREE MEMORY
011754  002766                                  BLT FMERR                       ;REPORT ERROR IF NOT ENOUGH MEMORY
011756  060101                                  ADD R1,R1                       ;CHANGE WORDS TO BYTES
011760  060137  002146                          ADD R1,FFREE                    ;CALCULATE NEW START OF FREE MEMORY
011764                                          POP R1                          ;GET START OF ALLOCATED MEMORY
011764  012601                                                                          MOV (SP)+,R1
011766  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

011770  012701  000214        HCOMM:  MOV #HC.SIZ/2,R1                ;GET SIZE OF AREA TO ALLOCATE
011774  004737  011744                CALL ALOCM                      ;ALLOCATE THE MEMORY
012000  010165  000016                MOV R1,C.RING(R5)                       ;GET ADDRESS OF HOST COMM AREA
                                                                        ;PLACE IN CONTROLLER TABLE
012004  000207                        RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;TINIT
                                        ;
                                        ;INITIALIZE VARIABLES FOR TEST
                                        ;
                                        ;INPUTS:
                                        ;       R1 - TEST NUMBER
                                        ;OUTPUTS:
                                        ;       LBUFS - CLEARED (DELETES ERROR LOG)
                                        ;       FFREE - FROM FMEM
                                        ;       FSIZE - FROM FMEMS
                                        ;       TNUM - TEST NUMBER FROM R1
                                        ;       ALL REGISTERS CLOBERED
                                        
012006  004737  012064          TINIT:  CALL RESET                      ;RESET ALL DEVICES
012012  013737  002152  002146          MOV FMEM,FFREE                  ;INIT FFREE
012020  013737  002154  002150          MOV FMEMS,FSIZE                 ;INIT FSIZE
012026  020137  002214                  CMP R1,FNUM                     ;SEE IF SAME TEST RUNNING
012032  001007                          BNE TINITR                      ;IF NOT, GO TO READ DM PROGRAM
012034  013737  002170  002146          MOV DMEND,FFREE                 ;CHANGE FREE MEMORY TO LEAVE
012042  013737  002172  002150          MOV DMENDS,FSIZE                ;  DM PROGRAM ALLOCATED
012050  000207                          RETURN
                                        
012052  004737  012210          TINITR: CALL READDM                     ;READ DM PROGRAM
012056  010137  002214                  MOV R1,FNUM                     ;STORE TEST NUMBER TO SHOW DM PROGRAM IN MEMORY
012062  000207                          RETURN
```

```
                                ;RESET
                                ; RESET ALL UDA-50S IN THE CONTROLLER TABLES
                                ;
                                ; INPUTS:
                                ;       IPADRS - CONTAINS ALL IP ADDRESSES
                                ; OUTPUTS:
                                ;       NONE
                                ;
012064                          RESET:  PUSH <R3,R4>
012064  010346                                                          MOV R3,-(SP)
012066  010446                                                          MOV R4,-(SP)
012070  005037  002222                  CLR     NXMAD
012074                                  SETVEC #4,#NXMI,#PRI07
012074  012746  000340                                                  MOV     #PRI07,-(SP)
012100  012746  017152                                                  MOV     #NXMI,-(SP)
012104  012746  000004                                                  MOV     #4,-(SP)
012110  012746  000003                                                  MOV     #3,-(SP)
012114  104437                                                          TRAP    C$SVEC
012116  062706  000010                                                  ADD     #10,SP
012122                                  BREAK
012122  104422                                                          TRAP    C$BRK
012124  012703  000010                  MOV     #8.,R3          ; R3 = COUNTER OF ENTRIES
012130  012704  003474                  MOV     #IPADRS,R4      ; R4 -> IP ADDRESS
012134  005714                  1$:     TST     (R4)            ; IS THERE AN ENTRY?
012136  001406                          BEQ     2$              ; IF NOT, DONE
012140  005034                          CLR     @(R4)+          ; INIT UDA
012142  005737  002222                  TST     NXMAD           ; WAS THERE AN ERROR?
012146  001010                          BNE     3$              ; IF SO, EXIT
012150  005303                          DEC     R3              ; MAKE SURE WE DO NOT EXTEND OVER AREA
012152  001370                          BNE     1$              ; IF NOT DONE, BRANCH
012154                          2$:     CLRVEC  #4
012154  012700  000004                                                  MOV     #4,R0
012160  104436                                                          TRAP    C$CVEC
012162                                  POP     <R4,R3>
012162  012604                                                          MOV (SP)+,R4
012164  012603                                                          MOV (SP)+,R3
012166  000207                          RETURN
                                ;
012170  005744                  3$:     TST     -(R4)           ; R4 -> UDAIP THAT FAILED
012172  010405                          MOV     R4,R5           ; SAVE IN R5 FOR REPORT
012174                                  ERRDF   20,,ERR020
012174  104455                                                          TRAP    C$ERDF
012176  000024                                                          .WORD   20
012200  000000                                                          .WORD   0
012202  011154                                                          .WORD   ERR020
012204  005014                          CLR     (R4)            ; DESTROY ENTRY SO NOT TO FALL INTO RESET ERROR LOOP
012206                                  DOCLN
012206  104444                                                          TRAP    C$DCLN
```

```
                                        ;READDM
                                        ;
                                        ;READ A DM PROGRAM INTO FREE MEMORY
                                        ;
                                        ;INPUTS:
                                        ;       R1 - TEST NUMBER
                                        ;OUTPUTS:
                                        ;       DMPROG - POINTER TO START OF DM PROGRAM IN MEMORY
                                        ;       R1 - UNCHANGED
                                        ;       CARRY CLEAR IF NO ERROR, CARRY SET IF PROGRAM NOT FOUND
                                        ;ALL REGISTERS BUT R1 ARE USED AND PREVIOUS CONTENTS DESTROYED
                                        
012210  013737  002146  002164  READDM: MOV FFREE,DMPROG                ;GET STORAGE ADDRESS
012216  004737  012516                  CALL RDREC
012222  103407                          BCS README                      ;CHECK IF ERROR
012224  013737  002146  002170          MOV FFREE,DMEND                 ;SAVE END OF ADDRESS OF DM PROGRAM
012232  013737  002150  002172          MOV FSIZE,DMENDS                ;  AND CURRENT SIZE OF FREE MEMORY
012240  000207                          RETURN
                                        
012242                          README: ERRSF 7,,ERR007                 ;REPORT DM PROGRAM NOT FOUND
012242  104454                                                                  TRAP    C$ERSF
012244  000007                                                                  .WORD   7
012246  000000                                                                  .WORD   0
012250  011066                                                                  .WORD   ERR007
012252                                  DOCLN
012252  104444                                                                  TRAP    C$DCLN
```

```
      1                                 ;GTDUST
      2                                 ;
      3                                 ;GET DUST STATUS
      4                                 ;
      5 012254  016504  000004  GTDUST: MOV C.VEC(R5),R4          ;GET VECTOR OF UDA
      6 012260                          AND CT.VEC,R4
        012260  042704  177000                                    BIC #^C<CT.VEC>,R4
      7 012264  010501                  MOV R5,R1                 ;GET INTERRUPT SERVICE LINK
      8 012266  062701  000010          ADD #C.JSR,R1
      9 012272                          SETVEC R4,R1,#PRIO7       ;SET UP INTERRUPT VECTOR
        012272  012746  000340                                    MOV     #PRIO7,-(SP)
        012276  010146                                            MOV     R1,-(SP)
        012300  010446                                            MOV     R4,-(SP)
        012302  012746  000003                                    MOV     #3,-(SP)
        012306  104437                                            TRAP    C$SVEC
        012310  062706  000010                                    ADD     #10,SP
     10                                                           ;INITIALIZE UDA WITH SMALLEST
     11 012314  006204                  ASR R4                    ;POSITION VECTOR FOR UDA
     12 012316  006204                  ASR R4
     13 012320  004737  011770          CALL HCOMM                ;ALLOCATE SPACE FOR HOST COMM AREA
     14 012324  004737  017254          CALL UDAINT               ; RING BUFFER AND INTERRUPTS ENABLED
     15 012330  001002                  BNE     1$
     16 012332  000137  016570          JMP LOADER                ;BRANCH IF AN ERROR
     17 012336  012700  000001  1$:     MOV     #OP.GSS,R0        ; R0 HAS OPCODE
     18 012342  004737  016574          CALL    BLDCMD
     19 012346  004737  016660          CALL    SNDCMD
     20 012352  004737  017000          CALL    WAITMS
     21 012356  000207                  RETURN
```

```
                                ;RUNDM
                                ;
                                ;LOAD AND RUN A DM PROGRAM IN THE CONTROLLERS. RETURN WHEN ALL
                                ;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.

012360  010137  002174          RUNDM:  MOV R1,URUN                     ;SAVE NUMBER OF UNITS TO RUN
012364  005037  002176                  CLR URNING                      ;CLEAR NUMBER OF UNITS RUNNING

                                ;LOAD DM PROGRAM INTO EACH CONTROLLER

012370  013737  002174  002200          MOV URUN,UCNT                   ;SET COUNTER OF UNITS
012376  013705  002162                  MOV TSTTAB,R5                   ;GET FIRST CONTROLLER TABLE
012402                          LDDM:
012402  042765  176377  000014          BIC #^C<CT.U52+CT.U50>,C.FLG(R5) ;CLEAR ALL FLAGS
012410  116537  000002  002074          MOVB C.UNIT(R5),L$LUN           ;SEE IF UNIT TO BE TESTED
012416  005765  000002                  TST C.UNIT(R5)
012422  100411                          BMI LDNEXT                      ;IF NOT, DON'T LOAD THIS UNIT
012424                                      ASSUME CT.AVL EQ BIT15
012424  033765  002212  000014          BIT     TSTTYP,C.FLG(R5)        ; DO WE TEST THIS TYPE?
012432  001405                          BEQ     LDNEXT                  ; IF NOT, BRANCH
012434  004737  016372                  CALL LOADDM                     ;LOAD THE DM PROGRAM
012440  001402                          BEQ LDNEXT                      ;IF ERROR, GO TO NEXT CONTROLLER
012442  005237  002176                  INC URNING                      ;IF NO ERROR, COUNT UNIT RUNNING
012446  062705  000054          LDNEXT: ADD #C.SIZE,R5                  ;MOVE TO NEXT CONTROLLER TABLE
012452  005337  002200                  DEC UCNT                        ;CHECK IF MORE CONTROLLERS
012456  001351                          BNE LDDM                        ;LOAD NEXT
012460  005037  002220                  CLR UFREEZ              ;CLEAR UNIT FREEZE FLAG
012464  012737  177777  003226          MOV #-1,FCTNUM  ;INVALIDATE FCT BLOCK NUMBER (BLOCK IN MEMORY)

                                ;CHECK IF ANY CONTROLLERS LOADED

012472  005737  002176                  TST URNING                      ;ANY UNITS LOADED?

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

012476  000207                          RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                ;CLOSEF
                                ;
                                ;CLOSE DATA FILE FOR DM PROGRAMS
                                ;
                                ;INPUTS:
                                ;       FILOPN - ZERO IF FILE NOT OPEN
                                ;OUTPUTS:
                                ;       NONE
                                ;
012500  005737  002202          CLOSEF: TST FILOPN                      ;SEE IF FILE CURRENTLY OPEN
012504  001403                          BEQ 1$
012506                                  CLOSE                           ; IF SO, CLOSE IT
012506  104435                                                                  TRAP    C$CLOS
012510  005037  002202                  CLR FILOPN                      ;AND MARK AS SO
012514  000207                  1$:     RETURN
```

```
                                        ;RDREC
                                        ;
                                        ;READ A RECORD FROM THE INPUT FILE. PLACE DATA INTO FREE MEMORY.
                                        ;
                                        ;INPUTS:
                                        ;       R1 - FILE TYPE
                                        ;               1 - UDA52 FORMATTER
                                        ;               2 - UDA50 FORMATTER
                                        ;OUTPUTS:
                                        ;       DATA FROM RECORD IN MEMORY
                                        ;       CARRY CLEAR IF NO ERROR, CARRY SET IF ERROR
                                        ;
012516                                  RDREC:  PUSH <R0,R1,R2,R3,R4,R5>
012516  010046                                                                  MOV R0,-(SP)
012520  010146                                                                  MOV R1,-(SP)
012522  010246                                                                  MOV R2,-(SP)
012524  010346                                                                  MOV R3,-(SP)
012526  010446                                                                  MOV R4,-(SP)
012530  010546                                                                  MOV R5,-(SP)
012532  005737  002202                          TST FILOPN                      ;SEE IF FILE ALREADY OPEN
012536  001005                                  BNE RDSTS
012540                                          OPEN #FNAME                     ;IF NOT, OPEN FILE NOW
012540  012700  003266                                                          MOV     #FNAME,R0
012544  104434                                                                  TRAP    C$OPEN
012546  005237  002202                          INC FILOPN                      ;AND MARK AS OPEN
012552  005005                          RDSTS:  CLR R5                          ;CLEAR LOAD ADDRESS (SEARCH MODE)
012554                                  RDST:   BREAK                           ;ALLOW PROGRAM TO BE INTERRUPTED
012554  104422                                                                  TRAP    C$BRK
                                                                                ;GETBYTE CALLS DON'T SEEM TO BREAK ON CONTROL-C!
012556                                          GETBYTE R4                      ;GET A BYTE
012556  104426                                                                  TRAP    C$GETB
012560  110004                                                                  MOVB    R0,R4
012562  005704                          TST R4                                  ;IF ZERO
012564  001773                          BEQ RDST                                ; KEEP READING
012566  022704  000001                  CMP #1,R4                               ;WHEN NOT ZERO
012572  001141                          BNE RWRDE1                              ; IT BETTER BE A ONE
012574                                  GETBYTE R0                              ; AND THE NEXT BYTE
012574  104426                                                                  TRAP    C$GETB
012576  060004                          ADD R0,R4
012600  005700                          TST R0                                  ; IF ZERO, PROCESS DATA
012602  001420                          BEQ RDDAT
```

```
      1 012604  020001                CMP R0,R1               ;CHECK IF TYPE OF FILE LOOKING FOR
      2 012606  103416                BLO RDDAT               ;IF TOO SOON IN FILE, KEEP SEARCHING
      3 012610  101120                BHI RDERR               ;IF PAST TYPE, GIVE ERROR RETURN
      4 012612  004737  013030        CALL FWORD              ;GET NEXT TWO WORDS
      5 012616  013702  002224        MOV FDATA,R2
      6 012622  004737  013030        CALL FWORD
      7 012626                        GETBYTE R0              ;GET CHECKSUM
        012626  104426                                          TRAP    C$GETB
      8 012630  060004                ADD R0,R4               ;ADD TO COMPUTED SUM
      9 012632  105704                TSTB R4                 ;SEE IF THIS SUM IS ZERO
     10 012634  001120                BNE RWRDE1              ;  IF NOT, REPORT CHECKSUM ERROR
     11 012636  013705  002146        MOV FFREE,R5            ;GET STORAGE ADDRESS
     12 012642  000744                BR RDST                 ;SWITCH FROM SEARCH TO STORE MODE
```

```
012644  004737  013030  RDDAT:  CALL FWORD          ;READ BYTE COUNT
012650  013703  002224          MOV FDATA,R3        ;SAVE IN R3
012654  004737  013030          CALL FWORD          ;READ LOAD ADDRESS
012660  162703  000006          SUB #6,R3           ;SUBTRACT BYTES ALREADY READ FROM BYTE COUNT
012664  001443                  BEQ RWORDT          ;IF RESULT IS ZERO, THIS IS A
                                                    ;TRANSFER BLOCK
012666  005705                  TST R5              ;IF IN SEARCH MODE,
012670  001425                  BEQ 3$              ;  BYPASS TRANSFER ADDRESS COMPUTATION
012672  013701  002224          MOV FDATA,R1        ;GET LOAD ADDRESS
012676  060301                  ADD R3,R1           ;ADD BYTES IN RECORD
012700  162701  001000          SUB #DMFRST,R1
012704  060501                  ADD R5,R1           ;COMPUTE STORAGE ADDRESS
012706  032701  000001          BIT #1,R1           ;CHECK IF ODD BYTE ADDRESS
012712  001401                  BEQ 1$              ;IF SO,
012714  005201                  INC R1              ;  INCREMENT
012716  163701  002146  1$:     SUB FFREE,R1        ;SEE IF MORE MEMORY NEEDS TO BE ALLOCATED
012722  101403                  BLOS 2$
012724  006001                  ROR R1              ;REDUCE TO WORDS
012726  004737  011744          CALL ALOCM          ;ALLOCATE THE MEMORY
012732  013702  002224  2$:     MOV FDATA,R2        ;GET LOAD ADDRESS
012736  162702  001000          SUB #DMFRST,R2
012742  060502                  ADD R5,R2
012744                  3$:     GETBYTE R0          ;GET DATA BYTE
012744  104426                                                          TRAP    C$GETB
012746  005705                  TST R5              ;IF IN SEARCH MODE,
012750  001401                  BEQ 4$              ;  BYPASS DATA STORAGE
012752  110022                  MOVB R0,(R2)+       ;STORE IN MEMORY
012754  060004          4$:     ADD R0,R4           ;UPDATE CHECKSUM
012756  005303                  DEC R3              ;COUNT THE BYTE
012760  001371                  BNE 3$              ;GET THEM ALL
012762                          GETBYTE R0          ;GET CHECKSUM
012762  104426                                                          TRAP    C$GETB
012764  060004                  ADD R0,R4           ;ADD
012766  105704                  TSTB R4             ;IF CHECKSUM CORRECT,
012770  001671                  BEQ RDST            ;  THEN GO READ NEXT RECORD
012772  000441                  BR RWRDE1           ;  ELSE REPORT ERROR
```

```
      1 012774                          RWORDT: GETBYTE R0                      ;READ CHECKSUM BYTE
        012774  104426                                                                  TRAP    C$GETB
      2 012776  060004                          ADD R0,R4                       ;ADD TO COMPUTED CHECKSUM
      3 013000  105704                          TSTB R4                         ;CHECK LOW BYTE OF SUM
      4 013002  001035                          BNE RWRDE1                      ;BRANCH IF CHECKSUM ERROR
      5 013004  005705                          TST R5                          ;IF IN SEARCH MODE,
      6 013006  001662                          BEQ RDST                        ;  KEEP ON SEARCHING
      7 013010                                  POP <R5,R4,R3,R2,R1,R0>
        013010  012605                                                                  MOV (SP)+,R5
        013012  012604                                                                  MOV (SP)+,R4
        013014  012603                                                                  MOV (SP)+,R3
        013016  012602                                                                  MOV (SP)+,R2
        013020  012601                                                                  MOV (SP)+,R1
        013022  012600                                                                  MOV (SP)+,R0
      8 013024  000241                          CLC
      9 013026  000207                          RETURN
     10
     11 013030                          FWORD:  GETBYTE R0                      ;READ A BYTE FROM FILE
        013030  104426                                                                  TRAP    C$GETB
     12 013032  060004                          ADD R0,R4                       ;UPDATE CHECKSUM ERROR
     13 013034  110037  002224                  MOVB R0,FDATA                   ;START TO BUILD WORD
     14 013040                                  GETBYTE R0                      ;READ ANOTHER BYTE FROM FILE
        013040  104426                                                                  TRAP    C$GETB
     15 013042  060004                          ADD R0,R4                       ;UPDATE CHECKSUM
     16 013044  110037  002225                  MOVB R0,FDATA+1                 ;COMPLETE WORD
     17 013050  000207                          RETURN
```

```
      1 013052  004737  012500          RDERR:  CALL CLOSEF                     ;CLOSE FILE AS POSITION IS LOST
      2 013056                                  POP <R5,R4,R3,R2,R1,R0>
        013056  012605                                                          MOV (SP)+,R5
        013060  012604                                                          MOV (SP)+,R4
        013062  012603                                                          MOV (SP)+,R3
        013064  012602                                                          MOV (SP)+,R2
        013066  012601                                                          MOV (SP)+,R1
        013070  012600                                                          MOV (SP)+,R0
      3 013072  000261                          SEC                           ;ERROR RETURN, FILE NOT FOUND
      4 013074  000207                          RETURN
      5
      6 013076                          RWRDE1: ERRSF 5,,ERR005
        013076  104454                                                          TRAP    C$ERSF
        013100  000005                                                         .WORD   5
        013102  000000                                                         .WORD   0
        013104  011054                                                         .WORD   ERR005
      7 013106                                  DOCLN
        013106  104444                                                          TRAP    C$DCLN
```

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

013110  013705  002162          RESPDM: MOV TSTTAB,R5                   ;GET CONTROLLER TABLE ADDRESS
013114  013737  002174  002200          MOV URUN,UCNT                   ;SET COUNTER OF UNITS
013122                          RESPCT: BREAK   ;ALLOW DRS TO SEE TERMINAL INPUT
013122  104422                                                                  TRAP    C$BRK
013124  016504  000016                  MOV C.RING(R5),R4               ;GET HOST COMM AREA ADDRESS
013130  032765  000002  000014          BIT #CT.RN,C.FLG(R5)            ;CHECK IF PROGRAM RUNNING
013136  001502                          BEQ RSPNXT                      ;IF NOT, LOOK AT NEXT
013140  116537  000002  002074          MOVB C.UNIT(R5),L$LUN           ;STORE UNIT NUMBER UNDER TEST
013146  032765  000010  000014          BIT #CT.MSG,C.FLG(R5)           ;SEE IF INTERRUPT RECEIVED
013154  001150                          BNE RSPIN                       ;IF SO, LOOK AT PACKET
013156  032765  000004  000014          BIT #CT.CMD,C.FLG(R5)           ;SEE IF COMMAND HAS BEEN SENT
013164  001002                          BNE 1$                          ;IF NOT, SEND ONE
013166  000137  013734                  JMP RSPOUT

                                        ;CHECK IF UDA STILL RUNNING

013172  011503                  1$:     MOV (R5),R3                     ;GET ADDRESS OF UDAIP
013174  016301  000002                  MOV 2(R3),R1                    ;LOOK AT UDASA REGISTER
013200  001405                          BEQ RSPTM                       ;IF ZERO, UDA STILL RUNNING
013202                                  ERRDF 30,,ERR030                ;REPORT UDA HAS FATAL ERROR
013202  104455                                                                  TRAP    C$ERDF
013204  000036                                                                  .WORD   30
013206  000000                                                                  .WORD   0
013210  011404                                                                  .WORD   ERR030
013212  000465                          BR RSPDRP                       ;DROP CONTROLLER FROM TESTING

                                        ;CHECK FOR TIMEOUT OF RESPONSE

013214  005765  000044          RSPTM:  TST C.TOT(R5)                   ;SEE IF DUP PROGRAM TO BE TIMED
013220  001451                          BEQ RSPNTO
013222  005737  003232                  TST KW.CSR                      ;SEE IF A CLOCK ON SYSTEM
013226  001446                          BEQ RSPNTO                      ;DON'T TIME IF NO CLOCK
013230  023765  003244  000042          CMP KW.EL+2,C.TOH(R5)           ;COMPARE TO TIMEOUT COUNTER
013236  101005                          BHI RSPTMO
013240  001041                          BNE RSPNTO
013242  023765  003242  000040          CMP KW.EL,C.TO(R5)
013250  103435                          BLO RSPNTO                      ;IF TOO MUCH TIME ELAPSED SINCE LAST INTERRUPT
013252  032765  000040  000014  RSPTMO: BIT #CT.STA,C.FLG(R5)           ;SEE IF A GET DUST STATUS COMMAND OUTSTANDING
013260  001101                          BNE RSPTOE                      ;REPORT ERROR IF SO
013262  005764  000012                  TST HC.CCT(R4)                  ;SEE IF UDA TOOK LAST COMMAND PACKET
013266  100476                          BMI RSPTOE                      ;REPORT ERROR IF NOT
013270  012700  000100                  MOV #CT.TM1,R0                  ;SEE IF FIRST TIMEOUT ALREADY HAPPENED
013274  032765  000100  000014          BIT #CT.TM1,C.FLG(R5)
013302  001401                          BEQ 1$                          ;IF SO,
013304  006300                          ASL R0                          ;SET SECOND TIME OUT FLAG
013306  052700  000040          1$:     BIS #CT.STA,R0                  ;SET THE PROPER TIMEOUT BIT
013312  050065  000014                  BIS R0,C.FLG(R5)                ;  AND STATUS REQUESTED BIT
013316  012700  000001                  MOV #OP.GDS,R0                  ;BUILD GET DUST STATUS COMMAND
013322  004737  016574                  CALL BLDCMD
013326  012764  100000  000012          MOV #RG.OWN,HC.CCT(R4)          ;MARK COMMAND TO UDA
013334  005775  000000                  TST @(R5)                       ;TELL UDA COMMAND IS THERE
013340  000137  014014                  JMP RSPOU4
```

```
     54 013344                          RSPNTO:
```

```
                                        ;SWITCH TO NEXT CONTROLLER
                                        
013344  005737  002220      RSPNXT:   TST UFREEZ       ;FROZEN TO ONE UNIT?
013350  001264                        BNE RESPCT       ;STAY THERE IF SO
013352  062705  000054                ADD #C.SIZE,R5                  ;MOVE TO NEXT TABLE
013356  005337  002200                DEC UCNT                        ;CHECK IF MORE CONTROLLERS
013362  001257                        BNE RESPCT                      ;LOOK AT NEXT CONTROLLER
013364  000651                        BR RESPDM                       ;LOOK AT FIRST CONTROLLER AGAIN
                                        
                                        ;REMOVE A CONTROLLER FROM TESTING
                                        
013366  005065  000014      RSPDRP:   CLR C.FLG(R5)    ;CLEAR PROGRAM RUNNING
013372  005037  002220                CLR UFREEZ
013376  010504                        MOV R5,R4
013400  062704  000020                ADD #C.DRO,R4
013404  012702  000010                MOV #8.,R2
013410  012403              1$:       MOV (R4)+,R3
013412  001420                        BEQ 3$
013414  005763  000002                TST D.UNIT(R3)
013420                                    ASSUME DT.AVL EQ BIT15
013420  100003                        BPL 2$
013422  005302                        DEC R2
013424  001371                        BNE 1$
013426  000412                        BR 3$
013430  052763  100000  000002  2$:       BIS #DT.AVL,D.UNIT(R3)
013436  005302                        DEC R2
013440  001405                        BEQ 3$
013442  005714                        TST (R4)
013444  001403                        BEQ 3$
013446  004737  016372                CALL LOADDM      ;START DM PROGRAM AGAIN
013452  001223                        BNE RESPCT
013454  005337  002176          3$:       DEC URNING                      ;REDUCE RUNNING CONTROLLERS COUNT
013460  001331                        BNE RSPNXT                      ;IF ANY STILL RUNNING, LOOK AT THEM
013462  000207                        RETURN                          ;ELSE RETURN TO TEST SECTION
                                        
013464                      RSPTOE:   ERRDF 31,,ERR031                ;REPORT TIMEOUT ERROR
013464  104455                                                                TRAP    C$ERDF
013466  000037                                                                .WORD   31
013470  000000                                                                .WORD   0
013472  011420                                                                .WORD   ERR031
013474  000734                        BR RSPDRP                       ;DROP CONTROLLER FROM TESTING
```

```
                                        ;CONTROLLER HAS RESPONDED, LOOK AT MESSAGE PACKET
                                        ;
                                        ;CHECK FOR PROPER OPCODE IN END PACKET
                                        ;
013476  012700  000204          RSPIN:  MOV #OP.END+OP.SSD,R0           ;GET SEND DATA END PACKET OPCODE
013502  032765  000020  000014          BIT #CT.REQ,C.FLG(R5)           ;LOOK IF SEND DATA OR RECEIVE DATA
013510  001402                          BEQ RSPMWR
013512  012700  000205                  MOV #OP.END+OP.RSD,R0           ;CHANGE TO RECEIVE DATA END PACKET OPCODE
013516  120064  000030          RSPMWR: CMPB R0,HC.MPK+P.OPCD(R4)       ;COMPARE TO OPCODE IN END PACKET
013522  001145                          BNE RSPERR
                                        ;
                                        ;LOOK AT STATUS CODE
                                        ;
013524  032764  000037  000032          BIT #ST.MSK,HC.MPK+P.STS(R4)    ;CHECK FOR STATUS CODE ST.SUC (ZERO)
013532  001004                          BNE RSPERW
                                        ;
                                        ;CHECK FOR EXPECTED REFERENCE NUMBER
                                        ;
013534  026564  000052  000020          CMP C.REF(R5),HC.MPK+P.CRF(R4)  ;CHECK IF CORRECT REF NUMBER
013542  001405                          BEQ RSPPTW
013544                          RSPERW: ERRDF 33,,ERR033
013544  104455                                                          TRAP    C$ERDF
013546  000041                                                          .WORD   33
013550  000000                                                          .WORD   0
013552  011450                                                          .WORD   ERR033
013554  000704                          BR RSPDRP                       ;DROP UNIT FROM TESTING
                                        ;
                                        ;CHECK IF RESPONSE FROM SEND OR RECEIVE DATA COMMAND
                                        ;
013556  032765  000020  000014  RSPPTW: BIT #CT.REQ,C.FLG(R5)           ;CHECK IF RESPONSE FROM DM PROGRAM
013564  001463                  RSPOU:  BEQ RSPOUT                      ;LOOK AT REQUEST NUMBER IF SO
```

```
                                        ;MAINTENANCE READ END PACKET RECEIVED, LOOK AT REQUEST FROM DM PROGRAM
                                        
013566  016401  000306          RSPPT2: MOV HC.BF2(R4),R1               ;GET REQUEST NUMBER
013572  042701  007777                  BIC     #^C<DU.TYP>,R1          ;CHECK TYPE
013576  001403                          BEQ 1$                          ;IF ZERO, ERROR
013600  020127  060000                  CMP R1,#DU.SPC                  ;CHECK IF IN EXPECTED RANGE
013604  101405                          BLOS RSPPT3
013606                          1$:     ERRDF 32,,ERR032                ;BAD REQUEST NUMBER
013606  104455                                                                  TRAP    C$ERDF
013610  000040                                                                  .WORD   32
013612  000000                                                                  .WORD   0
013614  011432                                                                  .WORD   ERR032
013616  000663                          BR RSPDRP                       ;DROP UNIT FROM TESTING
                                        
013620  016403  000034          RSPPT3: MOV HC.MPK+P.BCNT(R4),R3        ;GET BYTE COUNT OF CHARACTERS RECEIVED IN R3
013624  162703  000002                  SUB #2,R3                       ;(FIRST TWO CHARACTERS ARE TYPE WORD)
013630  012700  000004                  MOV #OP.SSD,R0                  ;BUILD A SEND DATA COMMAND PACKET
013634  004737  016574                  CALL BLDCMD                     ;  FOR ANSWER TO DM PROGRAM
013640  012700  000164                  MOV #HC.BF1,R0                  ;POINT TO BUFFER IN PACKET
013644  004737  016736                  CALL CLRBUF                     ; AND CLEAR BUFFER
013650  010402                          MOV R4,R2                       ;R2 POINTS TO SEND BUFFER
013652  062704  000122                  ADD #HC.BSZ,R4          ;R4 POINTS TO CHARACTERS IN RECEIVE BUFFER
013656  042724  170000                  BIC #DU.TYP,(R4)+       ;CLEAR TYPE FIELD IN BUFFER
013662  000301                          SWAB R1                         ;GET TYPE RIGHT JUSTIFIED
013664  006201                          ASR R1                          ;TIMES TWO
013666  006201                          ASR R1
013670  006201                          ASR R1
013672  010100                          MOV R1,R0                       ;COPY MESSAGE TYPE TO R0
013674  005001                          CLR R1                          ;R1 CONTAINS ZERO SEND BYTE COUNT
013676  004770  014162                  CALL @RSPDSP-2(R0)              ;CALL REQUESTED ROUTINE
013702  001231                          BNE RSPDRP                      ;ROUTINE RETURNS Z CLEAR TO DROP UNIT FROM TESTING
                                                                        ; Z SET IF UNIT TO CONTINUE RUNNING
013704  016504  000016                  MOV C.RING(R5),R4               ;GET RING ADDRESS
013710  032701  000001                  BIT #1,R1                       ;LOOK AT CHARACTER COUNT TO SEND TO DUP PROGRAM
013714  001401                          BEQ 1$                          ;IF AN ODD COUNT
013716  005201                          INC R1                          ;  INCREASE BY ONE
013720  010164  000120          1$:     MOV R1,HC.CPK+P.BCNT(R4)        ;PUT CHARACTER COUNT IN COMMAND PACKET
013724  100003                          BPL RSPOUT                      ;IF NEGATIVE BYTE COUNT RETURNED
013726  042765  000020  000014          BIC #CT.REQ,C.FLG(R5)           ;  DON'T SEND ANY DATA TO UDA
                                        
                                        ;SEND COMMAND BACK TO UDA
                                        
013734  042765  000350  000014  RSPOUT: BIC #CT.MSG+CT.STA+CT.TM1+CT.TM2,C.FLG(R5)      ;CLEAR MESSAGE RECEIVED FLAG
013742  032765  000020  000014          BIT #CT.REQ,C.FLG(R5)           ;CHECK WHICH COMMAND TO SEND
013750  001014                          BNE RSPOU2                      ;BRANCH IF RESPONSE TO REQUEST
                                        
013752  012700  000005                  MOV #OP.RSD,R0                  ;BUILD RECEIVE DATA COMMAND
013756  004737  016574                  CALL BLDCMD
013762  012700  000306                  MOV #HC.BF2,R0                  ;POINT TO MESSAGE BUFFER
013766  004737  016736                  CALL CLRBUF                     ;  AND CLEAR IT
013772  052765  000020  000014          BIS #CT.REQ,C.FLG(R5)           ;SET REQUEST BIT
014000  000403                          BR RSPOU3
                                        
014002  042765  000020  000014  RSPOU2: BIC #CT.REQ,C.FLG(R5)           ;CLEAR REQUEST BIT
014010                          RSPOU3:
014010  004737  016660                  CALL SNDCMD                     ;SEND COMMAND TO UDA
014014  016500  000044          RSPOU4: MOV C.TOT(R5),R0                ;SET TIMEOUT
```

```
                                                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,HC.MPK+P.OPCD(R4)   ;SEE IF GET DUST STATUS OPCODE
                BNE RSPERW
000037          BITB #DF.ACT,HC.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 HC.MPK+P.DPI(R4),C.PRI(R5)  ;COMPARE PROGRESS INDICATOR
                BNE 2$
000050          CMP HC.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 HC.MPK+P.DPI(R4),C.PRI(R5)  ;GET COPY OF PROGRESS INDICATOR
000050          MOV HC.MPK+P.DPI+2(R4),C.PRI+2(R5)      ;GET COPY OF PROGRESS INDICATOR
000006  3$:     MOV #RG.OWN+RG.FLG,HC.MCT(R4)   ;GIVE MESSAGE BUFFER BACK TO UDA
                JMP RSPNXT
        4$:     JMP RSPTOE
```

```
                                ;RESPONSE REQUEST DISPATCH TABLE

014164  014200                RSPDSP: .WORD QUEST             ;QUESTION
014166  014252                        .WORD DQUEST            ;QUESTION WITH DEFAULT ANSWER
014170  014424                        .WORD INFO              ;INFORMATION MESSAGE FOR OPERATOR
014172  014514                        .WORD TERM              ;NORMAL TERMINATION
014174  014524                        .WORD ERRTRM            ;FATAL ERROR TERMINATION
014176  014534                        .WORD SPECL             ;SPECIAL
        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     +-------------------------------+
                        ;
                        ;        .              .
                        ;        .              .
                        ;        .              .
                        ;
                        ;        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
                                        ;
014200  004737  014666          QUEST:  CALL GTDRVT     ;GET POINTER TO DRIVE TABLE
014204  062700  000004                  ADD #D.SERN,R0  ;BUMP POINTER TO SERIAL NUMBER
014210  014403                          MOV -(R4),R3    ;GET QUESTION NUMBER
014212  001411                          BEQ QUE0        ;BRANCH IF QUESTION NUMBER 0
014214  020327  000007                  CMP R3,#7       ;IF NOT, SEE IF QUESTION NUMBER 7
014220  001410                          BEQ QUE7
014222                                  ERRDF 100,,ERR100       ;ANY OTHER NUMBER IS AN ERROR
014222  104455                                                          TRAP    C$ERDF
014224  000144                                                          .WORD   100
014226  000000                                                          .WORD   0
014230  011512                                                          .WORD   ERR100
014232  000244                          CLZ     ;CLEAR Z TO STOP DUP PROGRAM
014234  000207                          RETURN
                                        ;
014236  012700  003344          QUE0:   MOV #DATE0,R0   ;POINT TO DATE STRING
014242                          QUE7:
014242  005201                  QUEL:   INC R1  ;COUNT THE CHARACTERS
014244  112022                          MOVB (R0)+,(R2)+        ; AND PUT THEM IN OUTPUT BUFFER
014246  001375                          BNE QUEL        ;  UNTIL A NUL CHARACTER FOUND
014250  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
                                        
014252  004737  014666        DQUEST: CALL GTDRVT     ;GET DRIVE TABLE ADDRESS INTO R0
014256  014403                        MOV -(R4),R3    ;GET QUESTION NUMBER
014260  020327  000006                CMP R3,#DQUESZ
014264  101035                        BHI DQUEX
014266  006303                        ASL R3
014270  000173  014274                JMP @DQUEJP(R3)
014274  014360                DQUEJP: .WORD DQUEX     ; 0  (NOT USED)
014276  014312                        .WORD DQUNIT    ; 1  ENTER UNIT NUMBER TO FORMAT
014300  014360                        .WORD DQUEX     ; 2  (NOT USED)
014302  014360                        .WORD DQUEX     ; 3  (NOT USED)
014304  014364                        .WORD DQRFMT    ; 4  USE EXISTING BAD SECTOR INFORMATION
014306  014404                        .WORD DQRSTR    ; 5  DOWN-LINE LOAD BAD SECTOR BLOCK INFORMATION
014310  014414                        .WORD DQCONT    ; 6  CONTINUE IF BAD BLOCK INFO INACCESSIBLE
        000006                                DQUESZ=<<.-DQUEJP>/2>-1

                                        ;ENTER UNIT NUMBER TO FORMAT

014312                          DQUNIT: PUSH R5                                         MOV R5,-(SP)
014312  010546
014314  005004                        CLR R4
014316  011003                        MOV (R0),R3     ;GET DRIVE NUMBER
014320                                        ASSUME D.DRV EQ 0
014320  012700  000012                MOV #10.,R0     ;RADIX 10.
014324  004737  016334        DQUNL1: CALL DIVIDE
014330                                PUSH R5                                         MOV R5,-(SP)
014330  010546
014332  005201                        INC R1
014334  005703                        TST R3
014336  001372                        BNE DQUNL1
014340  010100                        MOV R1,R0
014342                          DQUNL2: POP R5                                        MOV (SP)+,R5
014342  012605
014344  062705  000060                ADD #'0,R5
014350  110522                        MOVB R5,(R2)+
014352  005300                        DEC R0
014354  001372                        BNE DQUNL2
014356                                POP R5                                        MOV (SP)+,R5
014356  012605
014360  000264                DQUEX:  SEZ
014362  000207                        RETURN

014364  032737  000003  003230  DQRFMT: BIT #SO.FMT,MODE
```

GLOBAL SUBROUTINES SECTION

```
     54 014372  001410                          BEQ DQNO
     55 014374  112712  000131          DQYES:  MOVB #'Y,(R2)
     56 014400  005201                          INC R1
     57 014402  000766                          BR DQUEX
     58
     59 014404  032737  000010  003230  DQRSTR: BIT #SO.STR,MODE
     60 014412  001370                          BNE DQYES
     61 014414                          DQCONT:
     62 014414  112712  000116          DQNO:   MOVB #'N,(R2)
     63 014420  005201                          INC R1
     64 014422  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
                                        
014424  016400  177776          INFO:   MOV -2(R4),R0   ;GET MESSAGE NUMBER
014430  001417                          BEQ INFOX       ;IF ZERO, IGNORE IT
014432  020027  000100                  CMP R0,#100     ;IF OCTAL 100
014436  001420                          BEQ INFOE       ;  PRINT ERROR MESSAGE
014440  005737  002220                  TST UFREEZ
014444  001007                          BNE INFOP
014446  005237  002220                  INC UFREEZ
014452  004737  014666                  CALL GTDRVT
014456  010002                          MOV R0,R2
014460  004737  014712                  CALL HEADER
014464  004737  014632          INFOP:  CALL MESG       ;PRINT THE MESSAGE
014470  012701  100000          INFOX:  MOV #BIT15,R1                   ;RETURN A NEGATIVE BYTE COUNT
014474  000264                          SEZ
014476  000207                          RETURN                          ;RETURN WITH Z SET
                                        
014500                          INFOE:  ERRDF 101,,ERR101       ;ANSWER WAS REJECTED BY DUP PROGRAM
014500  104455                                                                  TRAP    C$ERDF
014502  000145                                                                  .WORD   101
014504  000000                                                                  .WORD   0
014506  011526                                                                  .WORD   ERR101
014510  000244                  CLZ             ;RETURN WITH Z CLEAR TO STOP DUP PROGRAM
014512  000207                  RETURN
```

```
                                ;MESSAGE TYPE 4
                                ;
                                ;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
                                
014514  004737  014424          TERM:   CALL INFO       ;PRINT THE MESSAGE
014520  000244                          CLZ
014522  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
                                        ;
014524  004737  014424          ERRTRM: CALL INFO
014530  000244                          CLZ
014532  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
                                        
014534  023714  003226        SPECL:  CMP FCTNUM,(R4) ;SEE IF DESIRED BLOCK IS IN MEMORY
014540  001425                        BEQ SPECLX      ; IF SO, SEND TO DUP PROGRAM
014542  002407                        BLT SPECLR      ; IF LOWERED NUMBERED BLOCK IN MEMORY,
                                                        ;   GO READ NEXT BLOCK
014544                          SPECLC:
014544                                  CLOSE   ;OTHERWISE, START READING FROM BEGINNING AGAIN
014544  104435                                                          TRAP    C$CLOS
014546                                  OPEN #FNAME2
014546  012700  003254                                                  MOV     #FNAME2,R0
014552  104434                                                          TRAP    C$OPEN
014554  012737  177777  003226        MOV #-1,FCTNUM
014562  012703  001000        SPECLR: MOV #512.,R3    ;GET BYTE COUNT IN A BLOCK
014566  012701  002226                MOV #FCTBUF,R1  ;POINT TO STORAGE AREA
014572                          SPECLL: GETBYTE (R1)+   ;READ THE FILE
014572  104426                                                          TRAP    C$GETB
014574  110021                                                          MOVB    R0,(R1)+
014576                                  BNCOMPLETE SPECLE       ;PRINT ERROR IF NO MORE BYTES IN FILE
014576  103005                                                          BCC     SPECLE
014600  005303                        DEC R3  ;COUNT THE BYTES
014602  001373                        BNE SPECLL
014604  005237  003226                INC FCTNUM      ;KEEP COUNT OF BLOCK IN MEMORY
014610  000751                        BR SPECL

014612  005212          SPECLE: INC (R2)        ;TELL DUP PROGRAM DATA NOT AVAILABLE
014614  012762  002226  000002  SPECLX: MOV #FCTBUF,2(R2)       ;PUT ADDRESS OF DATA IN OUTPUT BUFFER
014622  012701  000006                MOV #6,R1       ;SEND 3 WORDS TO DUP PROGRAM
014626  000264                        SEZ
014630  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

014632                          MESG:   PRINT #CR
014632  112700  000015                                                  MOVB #CR,R0
014636  004737  016056                                                  CALL CPNT
014642  112400                  1$:     MOVB (R4)+,R0           ;PRINT CHARACTERS FROM DUP PROGRAM
014644  001405                          BEQ 2$                  ;  DISCARDING LF AND NULL CHARACTERS
014646  020027  000012                  CMP R0,#12
014652  001402                          BEQ 2$
014654                                  PRINT R0
014654  004737  016056                                                  CALL CPNT
014660  005303                  2$:     DEC R3                  ;COUNT THE CHARACTERS
014662  003367                          BGT 1$
014664  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)
                                        ;
014666                                  GTDRVT: PUSH R5
014666  010546                                                                  MOV R5,-(SP)
014670  062705  000020                          ADD #C.DR0,R5
014674  012500                          GTDRVL: MOV (R5)+,R0
014676  016037  000002  002074                  MOV D.UNIT(R0),L$LUN
014704                                                  ASSUME DT.AVL EQ BIT15
014704  100773                                  BMI GTDRVL
014706                                          POP R5
014706  012605                                                                  MOV (SP)+,R5
014710  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

014712  022737  000001  002012  HEADER: CMP #1,L$UNIT                  ;IF MORE THAN ONE UNIT BEING TESTED
014720  001411                          BEQ 1$
014722                                  PNTF MESSG,D.UNIT(R2),(R5),(R2)         ;PRINT UDA ADDRESS
014722  011246                                                                  MOV (R2),-(SP)
014724  011546                                                                  MOV (R5),-(SP)
014726  016246  000002                                                          MOV D.UNIT(R2),-(SP)
014732  004137  016230                                                          JSR R1,LPNTF
014736  004015                                                                  .WORD MESSG
014740  000006                                                                  .WORD PNT.CT
014742                                  ASSUME C.UADR EQ 0
014742                                  ASSUME D.DRV EQ 0
014742  000407                          BR 2$
014744  005737  003232          1$:     TST KW.CSR                      ;IF NO CLOCK BEING USED
014750  001406                          BEQ 3$                          ;BYPASS RUNTIME MESSAGE
014752                                  PRINT #CR
014752  112700  000015                                                          MOVB #CR,R0
014756  004737  016056                                                          CALL CPNT
014762  004737  020176          2$:     CALL RNTIME                     ;PRINT RUNTIME IF A CLOCK IN USE
014766  000207                  3$:     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
                                ;
014770  112201          OSTRNG: MOVB (R2)+,R1           ;GET CONTROL CHARACTER
014772  001421                  BEQ OSTRE               ;EXIT IF NULL CHARACTER
014774  012700  015270          MOV #ERRC,R0            ;GET POINTER TO CHARACTER TABLE
015000  120110          NCONS:  CMPB R1,(R0)            ;COMPARE CHARACTER WITH TABLE ENTRY
015002  001407                  BEQ NCONF               ;BRANCH IF MATCH FOUND
015004  105720                  TSTB (R0)+              ;INCREMENT POINTER
015006  001374                  BNE NCONS               ;CONTINUE SEARCH IF NOT END OF TABLE
015010                          PNTF ERRME1             ;REPORT BAD CONTROL CHARACTER
015010  004137  016230                                                  JSR R1,LPNTF
015014  003726                                                          .WORD ERRME1
015016  000000                                                          .WORD PNT.CT
015020  000406                  BR OSTRE
015022  162700  015270  NCONF:  SUB #ERRC,R0            ;GET INCREMENT INTO TABLE
015026  006300                  ASL R0                  ;DOUBLE TO WORD COUNT
015030  004770  015302          CALL @ERRD(R0)          ;DISPATCH TO PRINT ROUTINE
015034  000755                  BR OSTRNG               ;GET NEXT
015036  000207          OSTRE:  RETURN
```

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

015040  112200                CON.QU: MOVB (R2)+,R0           ;GET CHARACTER
015042  120027  000042                CMPB R0,#''             ;CHECK IF ENDING QUOTE
015046  001403                        BEQ CON.QX              ;IF SO, GO GET NEXT CONTROL CHARACTER
015050                                PRINT R0                ;PRINT THE CHARACTER
015050  004737  016056                                                CALL CPNT
015054  000771                        BR CON.QU               ;CONTINUE PRINTING
015056  000207                CON.QX: RETURN

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

015060  004737  015536        CON.A:  CALL GETCNT              ;GET COUNT OF CHARACTERS
015064                        CON.A1: PRINT (R4)+             ;PRINT THE CHARACTER
015064  112400                                                        MOVB (R4)+,R0
015066  004737  016056                                                CALL CPNT
015072  005301                        DEC R1                  ;COUNT THE CHARACTERS
015074  001373                        BNE CON.A1              ;PRINT UNTIL COUNT REACHES ZERO
015076  032704  000001                BIT #1,R4               ;CHECK IF R4 NOW ODD
015102  001401                        BEQ CON.A2
015104  005204                        INC R4                  ;IF SO, INCREMENT TO NEXT EVEN ADDRESS
015106  000207                CON.A2: RETURN                  ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS A D. PRINT DECIMAL NUMBER.

015110  012701  000012        CON.D:  MOV #10.,R1              ;LOAD RADIX
015114  004737  015614                CALL PNTNUM             ;PRINT NUMBER
015120  000207                        RETURN                  ;NOW GET NEXT CONTROL CHARACTER

                                        ;CONTROL CHARACTER WAS AN H. PRINT HEX NUMBER.

015122  012701  000020        CON.H:  MOV #16.,R1              ;LOAD RADIX
015126  004737  015614                CALL PNTNUM             ;PRINT NUMBER
015132  000207                        RETURN                  ;NOW GET NEXT CONTROL CHARACTER
```

```
                                ;CONTROL CHARACTER WAS AN O. PRINT OCTAL NUMBER.
                                
015134  012701  000010          CON.O:  MOV #8.,R1                      ;LOAD RADIX
015140  004737  015614                  CALL PNTNUM                     ;PRINT NUMBER
015144  000207                          RETURN                          ;NOW GET NEXT CONTROL CHARACTER
                                
                                ;CONTROL CHARACTER WAS AN N. PRINT NEW LINE SEQUENCE.
                                
015146  004737  015536          CON.N:  CALL GETCNT                     ;GET COUNT
015152                          CON.N1: PRINT #CR                       ;PRINT NEW LINE SEQUENCE
015152  112700  000015                                                          MOVB #CR,R0
015156  004737  016056                                                          CALL CPNT
015162  005301                          DEC R1                          ;COUNT THE SEQUENCES
015164  001372                          BNE CON.N1
015166  000207                          RETURN                          ;NOW GET NEXT CONTROL CHARACTER
                                
                                ;CONTROL CHARACTER WAS AN R. CALL A PRE-PROGRAMMED ROUTINE.
                                
015170  004737  015536          CON.R:  CALL GETCNT                     ;GET ROUTINE NUMBER
015174  020127  000010                  CMP R1,#ERRRSZ                  ;CHECK IF DEFINED ROUTINE NUMBER
015200  101004                          BHI CON.R1
015202  060101                          ADD R1,R1                       ;DOUBLE COUNT TO GET WORD INDEX
015204  004771  015246                  CALL @ERRRTB-2(R1)              ;CALL ROUTINE
015210  000207                          RETURN                          ;NOW GET NEXT CONTROL CHARACTER
015212                          CON.R1: PNTF ERRME1                     ;REPORT BAD MESSAGE STRING
015212  004137  016230                                                          JSR R1,LPNTF
015216  003726                                                                  .WORD ERRME1
015220  000000                                                                  .WORD PNT.CT
015222                                  POP R1                          ;FIX THE STACK
015222  012601                                                                  MOV (SP)+,R1
015224  000207                          RETURN
                                
                                ;CONTROL CHARACTER WAS AN S. PRINT SPACES.
                                
015226  004737  015536          CON.S:  CALL GETCNT                     ;GET COUNT
015232                          CON.S1: PRINT <#' >                     ;PRINT A SPACE
015232  112700  000040                                                          MOVB #' ,R0
015236  004737  016056                                                          CALL CPNT
015242  005301                          DEC R1                          ;COUNT THE SPACES
015244  001372                          BNE CON.S1
015246  000207                          RETURN                          ;NOW GET NEXT CONTROL CHARACTER
```

```
                                ;ERROR ROUTINE DISPATCH TABLE
                                
015250  015322                  ERRRTB: .WORD CALRE                     ;NOT USED
015252  015322                          .WORD CALRE                     ;NOT USED
015254  015322                          .WORD CALRE                     ;NOT USED
015256  015334                          .WORD CALR4                     ;PRINT BASIC LINE WITHOUT UDA ADDRESS
015260  015410                          .WORD CALR5                     ;PRINT BASIC LINE WITH UDA ADDRESS
015262  015466                          .WORD CALR6                     ;CALL ALTERNATE PRINT STRING IN PDP-11 MEMORY
015264  015502                          .WORD CALR7                     ;PRINT  'REPLACE UDA MODULE M7161 OR M7485''
015266  015520                          .WORD CALR8                     ;PRINT  ''  UDASA CONTAINS  XXXXXX''
                                ;       .WORD CALR9                     ;REPRINT LAST NUMBER
        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
```

GLOBAL SUBROUTINES SECTION

```
                                ;HERE IS FIRST TABLE

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

015270                          ERRC:   BUILD
015270  042                             .BYTE '"
015271  101                             .BYTE 'A
015272  104                             .BYTE 'D
015273  110                             .BYTE 'H
015274  117                             .BYTE 'O
015275  116                             .BYTE 'N
015276  122                             .BYTE 'R
015277  123                             .BYTE 'S
015300  000                             .BYTE 0                 ;FOLLOW WITH A NULL BYTE
                                        .EVEN

                                ;HERE IS SECOND TABLE

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

015302                          ERRD:   BUILD
015302  015040                          .WORD CON.QU
015304  015060                          .WORD CON.A
015306  015110                          .WORD CON.D
015310  015122                          .WORD CON.H
015312  015134                          .WORD CON.O
015314  015146                          .WORD CON.N
015316  015170                          .WORD CON.R
015320  015226                          .WORD CON.S
```

GLOBAL SUBROUTINES SECTION

```
      1                         ;PRE-PROGRAMMED ROUTINES 1, 2 AND 3
      2                         ;NOT USED - PRINTS ERROR MESSAGE
      3
      4 015322                  CALRE:  PNTF ERRME1                 ;PRINT ERROR MESSAGE
        015322  004137  016230                                                JSR R1,LPNTF
        015326  003726                                                       .WORD ERRME1
        015330  000000                                                       .WORD PNT.CT
      5 015332  000207                  RETURN
```

GLOBAL SUBROUTINES SECTION

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

 5 015334                               CALR4:  PNTB BASLN,#BASNO,#BAS,#BAS,#BAS
   015334  012746  004211                                                        MOV #BAS,-(SP)
   015340  012746  004211                                                        MOV #BAS,-(SP)
   015344  012746  004211                                                        MOV #BAS,-(SP)
   015350  012746  004136                                                        MOV #BASNO,-(SP)
   015354  004137  016240                                                        JSR R1,LPNTB
   015360  004212                                                               .WORD BASLN
   015362  000010                                                               .WORD PNT.CT
 6 015364  004737  020176                      CALL RNTIME
 7 015370                                     PRINT #CR
   015370  112700  000015                                                        MOVB #CR,R0
   015374  004737  016056                                                        CALL CPNT
 8 015400  012737  016156  003246             MOV #PX,PTYPE
 9 015406  000207                             RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                ;PRE-PROGRAMMED ROUTINE 5
                                ;PRINT BASIC LINE FOR HOST PROGRAM ERROR WITH UDA ADDRESS
                                ;THEN SWITCH TO EXTENDED FORMAT
                                
 015410                         CALR5:  PNTB BASLN,#BASN0,#BASL2,(R5),#BAS,#BAS
 015410  012746  004211                                                  MOV #BAS,-(SP)
 015414  012746  004211                                                  MOV #BAS,-(SP)
 015420  011546                                                         MOV (R5),-(SP)
 015422  012746  004155                                                  MOV #BASL2,-(SP)
 015426  012746  004136                                                  MOV #BASN0,-(SP)
 015432  004137  016240                                                  JSR R1,LPNTB
 015436  004212                                                         .WORD BASLN
 015440  000012                                                         .WORD PNT.CT
 015442  004737  020176                 CALL RNTIME
 015446                                 PRINT #CR
 015446  112700  000015                                                  MOVB #CR,R0
 015452  004737  016056                                                  CALL CPNT
 015456  012737  016156  003246         MOV #PX,PTYPE
 015464  000207                         RETURN
```

```
                                        ;PRE-PROGRAMMED ROUTINE 6
                                        ;CALL ALTERNATE PRINT ROUTINE IN PDP-11 MEMORY
                                        
4 015466                        CALR6:  PUSH R2                         ;SAVE CURRENT STRING POINTER
  015466  010246                                                                MOV R2,-(SP)
5 015470  012402                        MOV (R4)+,R2                    ;GET NEW STRING POINTER
6 015472  004737  014770                CALL OSTRNG                     ;OUTPUT USING THIS STRING
7 015476                                POP R2                          ;GET OLD POINTER BACK
  015476  012602                                                                MOV (SP)+,R2
8 015500  000207                        RETURN                          ;NOW CONTINUE THE OLD STRING
```

GLOBAL SUBROUTINES SECTION

```
                                        ;PRE-PROGRAMMED ROUTINE 7
                                        ;PRINT ''REPLACE UDA MODULE M7161''
                                        
 4 015502                               CALR7:  PUSH R2                                 MOV R2,-(SP)
   015502  010246
 5 015504  012702  010533                       MOV #XFRU,R2
 6 015510  004737  014770                       CALL OSTRNG
 7 015514                                       POP R2                                  MOV (SP)+,R2
   015514  012602
 8 015516  000207                               RETURN
```

```
                                ;PRE-PROGRAMMED ROUTINE 8
                                ;PRINT   '' UDASA CONTAINS  XXXXXX''

 015520                         CALR8:  PUSH R2
 015520  010246                                                    MOV R2,-(SP)
 015522  012702  010502                 MOV #XSA,R2
 015526  004737  014770                 CALL OSTRNG
 015532                                 POP R2
 015532  012602                                                    MOV (SP)+,R2
 015534  000207                        RETURN
```

```
1                               ;       REPRINT LAST NUMBER
2                               ;               R4 -> TABLE
3                               ;CALR9: TST     -(R4)
4                               ;       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

015536                                  GETCNT: PUSH R0
015536  010046                                                                  MOV R0,-(SP)
015540  005001                                  CLR R1                  ;START WITH ZERO COUNT
015542  121227  000060                  GETCNX: CMPB (R2),#'0           ;CHECK IF CHARACTER A DIGIT
015546  103415                                  BLO GETCDN              ;BRANCH IF LOWER THAN ZERO
015550  121227  000071                          CMPB (R2),#'9
015554  101012                                  BHI GETCDN              ;BRANCH IF HIGHER THAN NINE
015556  006301                                  ASL R1                  ;MULTIPLY NUMBER BY 10
015560  010100                                  MOV R1,R0               ; SAVE 2N
015562  006301                                  ASL R1                  ; COMPUTE 4N
015564  006301                                  ASL R1                  ; COMPUTE 8N
015566  060001                                  ADD R0,R1               ; 8N + 2N = 10N
015570  112200                                  MOVB (R2)+,R0           ;GET DIGIT FROM STING
015572  162700  000060                          SUB #'0,R0              ;GET RID OF ASCII
015576  060001                                  ADD R0,R1               ;ADD TO NUMBER
015600  000760                                  BR GETCNX               ;GO TO NEXT CHARACTER
015602  005701                  GETCDN: TST R1                  ;CHECK IF NUMBER IS ZERO
015604  001001                                  BNE GETCXX              ;IF ZERO, CHANGE
015606  005201                                  INC R1                  ; TO DEFAULT OF ONE
015610                                  GETCXX: POP R0
015610  012600                                                                  MOV (SP)+,R0
015612  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
                                
015614  010100                  PNTNUM: MOV R1,R0                       ;SAVE RADIX
015616  004737  015536                  CALL GETCNT                     ;GET COUNT OF BITS
015622                          PNTNUS: PUSH <R2,R3,R5>
015622  010246                                                                  MOV R2,-(SP)
015624  010346                                                                  MOV R3,-(SP)
015626  010546                                                                  MOV R5,-(SP)
015630  012403                          MOV (R4)+,R3                    ;GET ONE PARAMETER WORD
015632  005005                          CLR R5                          ;CLEAR STORAGE FOR OTHER
015634  020127  000020                  CMP R1,#16.                     ;MORE THAN 16 BITS IN NUMBER?
015640  003401                          BLE 1$
015642  012405                          MOV (R4)+,R5                    ;YES, GET SECOND PARAMETER WORD
015644                          1$:     PUSH R4
015644  010446                                                                  MOV R4,-(SP)
015646  010504                          MOV R5,R4                       ;PUT HIGH WORD IN R4
015650  012702  000020                  MOV #16.,R2                     ;COMPUTE BITS NOT WANTED
015654  160102                          SUB R1,R2                       ;BY SUBTRACTING BITS TO USE
015656  002002                          BGE 2$                          ;FROM 16.
015660  062702  000020                  ADD #16.,R2                     ;IF NEGATIVE, ADD 16 FOR FIRST WORD
015664  001414                  2$:     BEQ 6$                          ;IF ZERO, NO BITS NEED BE CLEARED
015666  012705  100000                  MOV #BIT15,R5                   ;START MASK WITH SIGN BIT SET
015672  005302                  3$:     DEC R2                          ;COUNT BITS IN MASK
015674  001402                          BEQ 4$
015676  006205                          ASR R5                          ;SHIFT MORE BITS TO RIGHT
015700  000774                          BR 3$
015702  020127  000020          4$:     CMP R1,#16.                     ;MORE THAN 16 BITS IN NUMBER?
015706  003402                          BLE 5$
015710  040504                          BIC R5,R4                       ;YES, CLEAR IN HIGH WORD
015712  000401                          BR 6$
015714  040503                  5$:     BIC R5,R3                       ;NO, CLEAR IN LOW WORD
015716  004737  016334          6$:     CALL DIVIDE                     ;DIVIDE BY RADIX IN R0
015722                                  PUSH R5                         ;PUSH REMAINDER ON STACK
015722  010546                                                                  MOV R5,-(SP)
015724  005202                          INC R2                          ;COUNT DIGITS ON STACK
015726  005703                          TST R3                          ;CHECK IF QUOTIENT IS ZERO
015730  001372                          BNE 6$
015732  005704                          TST R4
015734  001370                          BNE 6$
```

GLOBAL SUBROUTINES SECTION

```
 1 015736  020027  000012          CMP R0,#10.             ;IF RADIX IS DECIMAL
 2 015742  001423                  BEQ 10$                 ; JUST GO PRINT DIGITS ON STACK
 3 015744  010103                  MOV R1,R3               ;OTHERWISE COMPUTE NUMBER OF LEADING ZEROS
 4 015746  162700  000014          SUB #12.,R0             ;DIVIDEND IS BITS IN NUMBER
 5 015752  003002                  BGT 7$                  ;DIVISOR IS BITS PER DIGIT PRINTED
 6 015754  012700  000003          MOV #3,R0               ;    (3 OR 4)
 7 015760  004737  016334  7$:     CALL DIVIDE
 8 015764  005705                  TST R5                  ;IF REMAINDER NOT ZERO
 9 015766  001401                  BEQ 8$                  ;INCREMENT QUOTIENT
10 015770  005203                  INC R3
11 015772  160203          8$:     SUB R2,R3               ;SUBTRACT DIGITS ON STACK
12 015774  001406                  BEQ 10$                 ;NO LEADING ZEROS IF ZERO
13 015776                  9$:     PRINT #'0               ;PRINT A ZERO
   015776  112700  000060                                                  MOVB #'0,R0
   016002  004737  016056                                                  CALL CPNT
14 016006  005303                  DEC R3
15 016010  001372                  BNE 9$                  ;REPEAT UNTIL COUNT REACHES ZERO
16
17 016012                  10$:    POP R5                  ;GET CHACACTER FROM STACK
   016012  012605                                                          MOV (SP)+,R5
18 016014  062705  000060          ADD #'0,R5              ;CNVERT TO ASCII DIGIT
19 016020  020527  000071          CMP R5,#'9              ;IF GREATER THAN A 9
20 016024  003402                  BLE 11$                 ; CONVERT TO A OR HIGHER
21 016026  062705  000007          ADD #<'A-'9-1>,R5       ; FOR HEX DIGIT
22 016032                  11$:    PRINT R5                ;PRINT THE CHARACTER
   016032  110500                                                          MOVB R5,R0
   016034  004737  016056                                                  CALL CPNT
23 016040  005302                  DEC R2                  ;REPEAT FOR ALL DIGITS
24 016042  001363                  BNE 10$                 ; ON STACK
25 016044                          POP <R4,R5,R3,R2>
   016044  012604                                                          MOV (SP)+,R4
   016046  012605                                                          MOV (SP)+,R5
   016050  012603                                                          MOV (SP)+,R3
   016052  012602                                                          MOV (SP)+,R2
26 016054  000207                  RETURN
```

```
                                        ;PRINT ONE CHARACTER
                                        ;
                                        ;CALL WITH MACRO PRINT
                                        
016056  110037  003250          CPNT:   MOVB R0,ERRCHR
016062                                  PUSH R1
016062  010146                                                  MOV R1,-(SP)
016064  012701  003664                  MOV #ERRONE,R1
016070  120027  000015                  CMPB R0,#CR
016074  001002                          BNE 1$
016076  012701  003667                  MOV #ERRNL,R1
016102  000177  165140          1$:     JMP @PTYPE
016106                          PF:     PRINTF R1,#ERRCHR
016106  012746  003250                                          MOV     #ERRCHR,-(SP)
016112  010146                                                  MOV     R1,-(SP)
016114  012746  000002                                          MOV     #2,-(SP)
016120  010600                                                  MOV     SP,R0
016122  104417                                                  TRAP    C$PNTF
016124  062706  000006                                          ADD     #6,SP
016130  000435                          BR CPNTX
016132                          PB:     PRINTB R1,#ERRCHR
016132  012746  003250                                          MOV     #ERRCHR,-(SP)
016136  010146                                                  MOV     R1,-(SP)
016140  012746  000002                                          MOV     #2,-(SP)
016144  010600                                                  MOV     SP,R0
016146  104414                                                  TRAP    C$PNTB
016150  062706  000006                                          ADD     #6,SP
016154  000423                          BR CPNTX
016156                          PX:     PRINTX R1,#ERRCHR
016156  012746  003250                                          MOV     #ERRCHR,-(SP)
016162  010146                                                  MOV     R1,-(SP)
016164  012746  000002                                          MOV     #2,-(SP)
016170  010600                                                  MOV     SP,R0
016172  104415                                                  TRAP    C$PNTX
016174  062706  000006                                          ADD     #6,SP
016200  000411                          BR CPNTX
016202                          PS:     PRINTS R1,#ERRCHR
016202  012746  003250                                          MOV     #ERRCHR,-(SP)
016206  010146                                                  MOV     R1,-(SP)
016210  012746  000002                                          MOV     #2,-(SP)
016214  010600                                                  MOV     SP,R0
016216  104416                                                  TRAP    C$PNTS
016220  062706  000006                                          ADD     #6,SP
016224                          CPNTX:  POP R1
016224  012601                                                  MOV (SP)+,R1
016226  000207                          RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;PRINT FORMATTED MESSAGE
                                        ;
                                        ;CALL WITH MACRO PNT, PNTF, PNTB, PNTX, OR PNTS
                                        ;
016230  012737  016106  003246  LPNTF:  MOV #PF,PTYPE
016236  000413                        BR LPNT
016240  012737  016132  003246  LPNTB:  MOV #PB,PTYPE
016246  000407                        BR LPNT
016250  012737  016156  003246  LPNTX:  MOV #PX,PTYPE
016256  000403                        BR LPNT
016260  012737  016202  003246  LPNTS:  MOV #PS,PTYPE
016266                        LPNT:    PUSH <R2,R3,R4,R5>
016266  010246                                                          MOV R2,-(SP)
016270  010346                                                          MOV R3,-(SP)
016272  010446                                                          MOV R4,-(SP)
016274  010546                                                          MOV R5,-(SP)
016276  012102                        MOV (R1)+,R2              ;GET ADDRESS OF STRING
016300  010604                        MOV SP,R4                 ;COMPUTE ADDRESS OF ARGUMENTS
016302  062704  000012                ADD #10.,R4               ;  WHICH ARE NOW ON STACK (IF ANY)
016306                                PUSH R1                   ;SAVE RETURN ADDRESS
016306  010146                                                          MOV R1,-(SP)
016310  004737  014770                CALL OSTRNG               ;PRINT THE FORMATTED MESSAGE
016314                                POP <R0,R5,R4,R3,R2,R1>   ;RESTORE ALL REGISTERS
016314  012600                                                          MOV (SP)+,R0
016316  012605                                                          MOV (SP)+,R5
016320  012604                                                          MOV (SP)+,R4
016322  012603                                                          MOV (SP)+,R3
016324  012602                                                          MOV (SP)+,R2
016326  012601                                                          MOV (SP)+,R1
016330  062006                        ADD (R0)+,SP              ;ADJUST STACK POINTER OVER ARGUMENTS
016332  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
                                        ;
016334                                  DIVIDE: PUSH R2
016334  010246                                                                  MOV R2,-(SP)
016336  012702  000040                          MOV #32.,R2                   ;SET UP SHIFT COUNT
016342  005005                                  CLR R5                        ;START WITH ZERO REMAINDER
016344  006303                          1$:     ASL R3                        ;SHIFT LEFT INTO R5
016346  006104                                  ROL R4
016350  006105                                  ROL R5
016352  020005                                  CMP R0,R5                     ;WILL DIVISOR GO INTO REMAINDER
016354  101002                                  BHI 2$                        ;ONLY SUBTRACT IF IT WILL
016356  160005                                  SUB R0,R5                     ;SUBTRACT DIVISOR
016360  005203                                  INC R3                        ;PUT A ONE INTO QUOTIENT
016362  005302                          2$:     DEC R2                        ;COUNT THE SHIFTS
016364  001367                                  BNE 1$
016366                                          POP R2
016366  012602                                                                  MOV (SP)+,R2
016370  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
                                        ;
016372  013701  002164          LOADDM: MOV DMPROG,R1                   ;GET STORAGE ADDRESS OF DM PROGRAM
016376  116165  000021  000044          MOVB DMTMO(R1),C.TOT(R5)        ;GET TIMEOUT VALUE
016404  105065  000045                  CLRB C.TOT+1(R5)
016410  016504  000004                  MOV C.VEC(R5),R4                ;GET VECTOR OF UDA
016414                                  AND CT.VEC,R4
016414  042704  177000                                                  BIC #^C<CT.VEC>,R4
016420  010501                          MOV R5,R1                       ;GET INTERRUPT SERVICE LINK
016422  062701  000010                  ADD #C.JSR,R1
016426                                  SETVEC R4,R1,#PRI07             ;SET UP INTERRUPT VECTOR
016426  012746  000340                                                  MOV     #PRI07,-(SP)
016432  010146                                                          MOV     R1,-(SP)
016434  010446                                                          MOV     R4,-(SP)
016436  012746  000003                                                  MOV     #3,-(SP)
016442  104437                                                          TRAP    C$SVEC
016444  062706  000010                                                  ADD     #10,SP
                                                                        ;INITIALIZE UDA WITH SMALLEST
016450  004737  017254                  CALL UDAINT                     ; RING BUFFER AND INTERRUPTS ENABLED
016454  001445                          BEQ LOADER                      ;BRANCH IF AN ERROR
```

```
      1 016456  017701  162302                MOV @DMPROG,R1                  ;GET SIZE OF PROGRAM
      2 016462  012700  000002        LOADB:  MOV #OP.ESP,R0                  ;BUILD EXECUTE SUPPLIED PROGRAM COMMAND PACKET
      3 016466  004737  016574                CALL BLDCMD
      4 016472  013764  002164  000124        MOV DMPROG,HC.CPK+P.UADR(R4)    ;LOAD MAIN PROGRAM ADDRESS
      5 016500  010164  000120                MOV R1,HC.CPK+P.BCNT(R4)        ;  AND SIZE
      6 016504  013764  002164  000140        MOV DMPROG,HC.CPK+P.OVRL(R4)    ;LOAD OVERLAY ADDRESS
      7 016512  067764  163446  000140        ADD @DMPROG,HC.CPK+P.OVRL(R4)
      8 016520  004737  016660                CALL SNDCMD                     ;SEND COMMAND TO UDA
      9 016524  004737  017000                CALL WAITMS                     ;WAIT FOR MESSAGE RESPONSE
     10 016530  001417                        BEQ LOADER      ;ABORT IF NO RESPONSE
     11 016532  032764  000037  000032        BIT #ST.MSK,HC.MPK+P.STS(R4)    ;CHECK FOR ERRORS
     12 016540  001007                        BNE LOADE1
     13 016542  042765  000024  000014        BIC #CT.CMD+CT.REQ,C.FLG(R5)    ;CLEAR COMMAND OUTSTANDING FLAG
     14 016550  052765  000002  000014        BIS #CT.RN,C.FLG(R5)            ;SET DM PROGRAM RUNNING FLAG
     15 016556  000207                        RETURN
```

```
                                ;UDA FAILED TO DOWNLINE LOAD DM PROGRAM
                                LOADE1: ERRDF 34,,ERR034
016560                                                                  TRAP    C$ERDF
016560  104455                                                          .WORD   34
016562  000042                                                          .WORD   0
016564  000000                                                          .WORD   ERR034
016566  011456
016570  000264                  LOADER: SEZ                     ;SET Z TO INDICATE ERROR OCCURRED
016572  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
                                
 016574                         BLDCMD: PUSH <R1,R0>
 016574  010146                                                                        MOV R1,-(SP)
 016576  010046                                                                        MOV R0,-(SP)
 016600  016504  000016                 MOV C.RING(R5),R4       ;GET ADDRESS OF HOST COMM AREA
 016604  010400                         MOV R4,R0               ;COPY TO R0
 016606  062700  000100                 ADD #HC.CEV,R0          ;COMPUTE ADDRESS OF COMMAND ENVELOPE
 016612  012720  000060                 MOV #HC.PSZ,(R0)+       ;LOAD PACKET LENGTH
 016616  012701  001000                 MOV #DUP,R1             ;LOAD DIAG CIRCUIT IDENTIFIER
 016622  022716  000031                 CMP #OP.MWR,(SP)        ;IF CODE IS MAINTENANCE WRITE
 016626  001002                         BNE BLDC0               ;  GET OTHER CIRCUIT IDENTIFIER
 016630  012701  177777                 MOV #DIAG,R1
 016634  010120                 BLDC0:  MOV R1,(R0)+            ;PUT IDENTIFIER INTO PACKET
 016636  012701  000030                 MOV #<HC.PSZ>/2,R1      ;GET WORDS TO CLEAR
 016642  005020                 BLDC1:  CLR (R0)+               ;CLEAR PACKET
 016644  005301                         DEC R1
 016646  001375                         BNE BLDC1
 016650                                 POP HC.CPK+P.OPCD(R4)   ;PUT OPCODE IN PACKET
 016650  012664  000114                                                            MOV (SP)+,HC.CPK+P.OPCD(R4)
 016654                                 POP R1                  ;RESTORE R1
 016654  012601                                                                    MOV (SP)+,R1
 016656  000207                         RETURN
```

GLOBAL SUBROUTINES SECTION

```
                                        ;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
                                        ;
                                        SNDCMD: PUSH <R0,R1>
016660  010046                                                                    MOV R0,-(SP)
016662  010146                                                                    MOV R1,-(SP)
016664  016504  000016                       MOV C.RING(R5),R4                    ;LOAD R4 WITH HOST COMM AREA ADDRESS
016670  005265  000052                       INC C.REF(R5)                        ;INCREMENT CMD REFERENCE NUMBER
016674  016564  000052  000104               MOV C.REF(R5),HC.CPK+P.CRF(R4)       ;PUT IN PACKET
016702  012764  140000  000006               MOV #RG.OWN+RG.FLG,HC.MCT(R4)        ;MARK MESSAGE PACKET AVAILABLE
016710  012764  100000  000012               MOV #RG.OWN,HC.CCT(R4)               ;MARK COMMAND TO UDA
016716  005775  000000                       TST @(R5)                            ;TELL UDA COMMAND IS THERE
016722  052765  000004  000014               BIS #CT.CMD,C.FLG(R5)                ;MARK COMMAND ISSUED
016730                                       POP <R1,R0>
016730  012601                                                                    MOV (SP)+,R1
016732  012600                                                                    MOV (SP)+,R0
016734  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
                                ;
016736                          CLRBUF: PUSH <R0,R1>
016736  010046                                                          MOV R0,-(SP)
016740  010146                                                          MOV R1,-(SP)
016742  060400                          ADD R4,R0                       ;ADD START OF HOST COMM AREA TO OFFSET
016744  010064  000124                 MOV R0,HC.CPK+P.UADR(R4)        ;PUT BUFFER ADDRESS IN COMMAND PACKET
016750  012764  000122  000120         MOV #HC.BSZ,HC.CPK+P.BCNT(R4)   ;PUT SIZE OF BUFFER IN COMMAND PACKET
016756  010004                         MOV R0,R4                       ;PUT BUFFER ADDRESS IN R4
016760  012701  000051                 MOV #HC.BSZ/2,R1                ;GET SIZE OF BUFFER IN WORDS
016764  005020                  CLRBFL: CLR (R0)+                       ;CLEAR ALL THE WORDS
016766  005301                          DEC R1
016770  001375                          BNE CLRBFL
016772                                  POP <R1,R0>
016772  012601                                                          MOV (SP)+,R1
016774  012600                                                          MOV (SP)+,R0
016776  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
                                ;
017000                          WAITMS: PUSH <R0,R1>
017000  010046                                                                 MOV R0,-(SP)
017002  010146                                                                 MOV R1,-(SP)
017004  012700  000036                  MOV #30.,R0             ;SET TIME OUT VALUE OF 30 SECONDS
017010  010501                          MOV R5,R1               ;POINT TO TIME OUT COUNTER
017012  062701  000040                  ADD #C.TO,R1
017016  004737  017172                  CALL SETTO
017022  011500                  1$:     MOV (R5),R0             ;GET ADDRESS OF UDAIP REGISTER
017024  032765  000010  000014          BIT #CT.MSG,C.FLG(R5)   ;LOOK IF INTERRUPT OCCURRED
017032  001030                          BNE 3$                  ;BRANCH IF SO
017034  016001  000002                  MOV 2(R0),R1            ;LOOK AT UDASA REGISTER
017040  001034                          BNE 4$                  ;BRANCH IF ERROR CODE PRESENT
017042                                  BREAK
017042  104422                                                                 TRAP    C$BRK
017044  005737  003232                  TST KW.CSR              ;SEE IF A CLOCK ON SYSTEM
017050  001764                          BEQ 1$
017052  023765  003244  000042          CMP KW.EL+2,C.TOH(R5)   ;CHECK IF TIMEOUT HAS HAPPENED
017060  101005                          BHI 2$
017062  001357                          BNE 1$
017064  023765  003242  000040          CMP KW.EL,C.TO(R5)
017072  103753                          BLO 1$
017074                          2$:     ERRDF 36,,ERR036
017074  104455                                                                 TRAP    C$ERDF
017076  000044                                                                 .WORD   36
017100  000000                                                                 .WORD   0
017102  011464                                                                 .WORD   ERR036
017104                                  POP <R1,R0>
017104  012601                                                                 MOV (SP)+,R1
017106  012600                                                                 MOV (SP)+,R0
017110  000264                          SEZ
017112  000207                          RETURN
```

GLOBAL SUBROUTINES SECTION

```
      1 017114  042765  00C010  000014  3$:     BIC #CT.MSG,C.FLG(R5)          ;CLEAR MESSAGE RECEIVED FLAG
      2 017122                                  POP <R1,R0>
        017122  012601                                                          MOV (SP)+,R1
        017124  012600                                                          MOV (SP)+,R0
      3 017126  000244                          CLZ                           ;GIVE NO ERROR RETURN
      4 017130  000207                          RETURN
      5 017132                            4$:   ERRDF 37,,ERR037
        017132  104455                                                          TRAP    C$ERDF
        017134  000045                                                          .WORD   37
        017136  000000                                                          .WORD   0
        017140  011476                                                          .WORD   ERR037
      6 017142                                  POP <R1,R0>
        017142  012601                                                          MOV (SP)+,R1
        017144  012600                                                          MOV (SP)+,R0
      7 017146  000264                          SEZ
      8 017150  000207                          RETURN
```

```
      1                                 ;NXMI
      2                                 ;
      3                                 ;NON-EXISTANT MEMORY SERVICE ROUTINE
      4                                 ;
      5                                 ;INPUTS:
      6                                 ;       NXMAD SET TO ZERO
      7                                 ;OUTPUTS:
      8                                 ;       NXMAD SET TO ONES IF NON-EXISTANT TRAP OCCURED
      9
     10 017152                          BGNSRV NXMI
        017152                                                                  NXMI::
     11
     12 017152  012737  177777  002222          MOV #-1,NXMAD
     13
     14 017160                          ENDSRV
        017160                                                                  L10033:
        017160  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
                                
 017162                         BGNSRV UDASRV
 017162                                                                 UDASRV::
 017162  052710  000010                 BIS #CT.MSG,(R0)               ;SET CT.MSG
 017166                                 POP R0                         ;RESTORE R0
 017166  012600                                                                 MOV (SP)+,R0
 017170                         ENDSRV
 017170                                                                 L10034:
 017170  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

017172                          SETTO:  PUSH <R2,R3>
017172  010246                                                          MOV R2,-(SP)
017174  010346                                                          MOV R3,-(SP)
017176  005002                          CLR R2                  ;CLEAR PRODUCT
017200  013703  003240                  MOV KW.HZ,R3            ;GET MULTIPLICAND
017204  006200                  SETOO:  ASR R0                  ;SHIFT MULTIPLIER TO RIGHT
017206  103001                          BCC SETO1               ;IF A ONE BIT SHIFTED OUT
017210  060302                          ADD R3,R2               ;  ADD MULTIPLICAND TO PRODUCT
017212  006303                  SETO1:  ASL R3                  ;DOUBLE THE MULTIPLICAND
017214  005700                          TST R0
017216  001372                          BNE SETOO               ;CONTINUE UNTIL MULTIPLIER IS ZERO

                                ;GET CURRENT TIME

017220  013700  003242          SETO2:  MOV KW.EL,R0            ;GET TIME
017224  013703  003244                  MOV KW.EL+2,R3
017230  020037  003242                  CMP R0,KW.EL            ;IF CHANGED DURING RETRIEVAL
017234  001371                          BNE SETO2               ;  GET IT AGAIN

                                ;ADD TIME TIL TIMEOUT

017236  060200                          ADD R2,R0               ;ADD
017240  005503                          ADC R3

                                ;PUT RESULT IN STORAGE

017242  010021                          MOV R0,(R1)+
017244  010311                          MOV R3,(R1)

017246                                  POP <R3,R2>
017246  012603                                                          MOV (SP)+,R3
017250  012602                                                          MOV (SP)+,R2
017252  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

017254  016502  000016        UDAINT: MOV C.RING(R5),R2                       ;GET FIRST ADDRESS OF RING BUFFER
017260  012703  000006                MOV #<HC.RSZ*2+HC.ISZ>/2,R3                     ;GET SIZE OF RING BUFFER
017264  012722  177777        UDAI1L: MOV #-1,(R2)+                   ;WRITE ONES TO BUFFER
017270  005303                        DEC R3                          ;COUNT THE WORDS IN BUFFER
017272  003374                        BGT UDAI1L                      ;LOOP UNTIL ENTIRE BUFFER WRITTEN

                                ;DO THE INITIALIZATION

017274  004737  017470                CALL UDAIST                     ;DO FIRST THREE STEPS
017300  103471                        BCS UDAIEX                      ;GET OUT IF UDA MICROCODE REPORTED FAILURE
017302  012364  000002                MOV (R3)+,2(R4)                 ;WRITE NEXT WORD TO UDASA REGISTER
017306  012703  000310                MOV #200.,R3                    ;GET TRY COUNTER
017312  016402  000002        UDAI1A: MOV 2(R4),R2                    ;LOOK AT UDASA
017316  001410                        BEQ UDAI1C
017320  100005                        BPL UDAI1B
017322                                ERRDF 24,,ERR024
017322  104455                                                                  TRAP    C$ERDF
017324  000030                                                                  .WORD   24
017326  000000                                                                  .WORD   0
017330  011352                                                                  .WORD   ERR024
017332  000454                        BR UDAIEX
017334  005303                UDAI1B: DEC R3
017336  001365                        BNE UDAI1A
017340  010264  000002        UDAI1C: MOV R2,2(R4)                    ;WRITE 0 TO UDASA (PURGE)
017344  011402                        MOV (R4),R2                     ;READ FROM UDAIP (POLL)
017346  004737  020014                CALL UDARSP                     ;WAIT FOR STEP OR ERROR BIT
017352  103444                        BCS UDAIEX                      ;GET OUT IF UDA MICROCODE REPORTED FAILURE
```

```
                                        ;CHECK HOST COMMUNICATION AREA FOR ALL ZEROS
                                        
017354  016502  000016      UDAI2:  MOV C.RING(R5),R2                ;GET FIRST ADDRESS OF RING BUFFER
017360  012703  000006              MOV #<HC.RSZ*2+HC.ISZ>/2,R3              ;GET SIZE OF RING BUFFER
017364  005722              UDAI2L: TST (R2)+                        ;CHECK WORD IN BUFFER
017366  001003                      BNE UDAI2E                       ;GO TO ERROR REPORTER IF NOT ZERO
017370  005303                      DEC R3                           ;COUNT THE WORDS IN BUFFER
017372  003374                      BGT UDAI2L                       ;LOOP UNTIL ALL WORDS CHECKED
017374  000405                      BR UDAI3
                                    
017376                      UDAI2E: ERRDF 23,,ERR023                 ;REPORT BUFFER NOT CLEARED
017376  104455                                                               TRAP    C$ERDF
017400  000027                                                               .WORD   23
017402  000000                                                               .WORD   0
017404  011266                                                               .WORD   ERR023
017406  000426                      BR UDAIEX
                                    
                                    ;SEND GO BIT TO UDASA REGISTER TO END INITIALIZATION
                                    
017410                      UDAI3:
017410  016500  000006              MOV C.BST(R5),R0                 ;GET BURST VALUE
017414  006300                      ASL R0                           ;SHIFT TO POSITION
017416  006300                      ASL R0
017420  052700  000001              BIS #SA.GO,R0                    ;SET THE GO BIT
017424  010064  000002              MOV R0,2(R4)                     ;SEND TO UDA
017430  016501  000016              MOV C.RING(R5),R1
017434  010161  000004              MOV R1,HC.MSG(R1)
017440  062761  000020  000004      ADD #HC.MPK,HC.MSG(R1)
017446  010161  000010              MOV R1,HC.CMD(R1)
017452  062761  000104  000010      ADD #HC.CPK,HC.CMD(R1)
017460  000244                      CLZ                              ;CLEAR Z AS NO ERROR INDICATION
017462  000207                      RETURN
                                    
                                    ;ERROR RETURN
                                    
017464  000264              UDAIEX: SEZ                              ;SET Z TO INDICATE ERROR OCCURRED
017466  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

017470                          UDAIST: BREAK
017470  104422                                                          TRAP    C$BRK
017472                                  PUSH R1
017472  010146                                                          MOV R1,-(SP)
017474  016504  000004                 MOV C.VEC(R5),R4
017500                                  AND CT.VEC,R4
017500  042704  177000                                                  BIC #^C<CT.VEC>,R4
017504  006204                          ASR R4
017506  006204                          ASR R4
017510  052704  100000                 BIS #SA.STP,R4          ;SET STEP BIT IN DATA WORD
017514  010437  017706                 MOV R4,UDAID1           ;LOAD INTERRUPT VECTOR
017520  016537  000016  017712         MOV C.RING(R5),UDAID2   ;LOAD MEMORY ADDRESS
017526  062737  000004  017712         ADD #HC.MSG,UDAID2      ;  OF FIRST RESPONSE RING

                                ;START THE INITIALIZATION BY WRITING TO UDAIP REGISTER

017534  016504  000000                 MOV C.UADR(R5),R4       ;GET ADDRESS OF UDAIP REGISTER
017540  005037  002222                 CLR NXMAD               ;CLEAR MEMORY ERROR FLAG
017544                                  SETVEC #4,#NXMI,#PRI07  ;SET UP VECTOR 4
017544  012746  000340                                                  MOV     #PRI07,-(SP)
017550  012746  017152                                                  MOV     #NXMI,-(SP)
017554  012746  000004                                                  MOV     #4,-(SP)
017560  012746  000003                                                  MOV     #3,-(SP)
017564  104437                                                          TRAP    C$SVEC
017566  062706  000010                                                  ADD     #10,SP
017572  005764  000002                 TST 2(R4)               ;ACCESS UDASA REGISTER
017576  005014                          CLR (R4)                ;WRITE TO UDAIP
017600                                  CLRVEC #4               ;GIVE UP THE VECTOR
017600  012700  000004                                                  MOV     #4,R0
017604  104436                                                          TRAP    C$CVEC
017606  005737  002222                 TST NXMAD               ;SEE IF A MEMORY ERROR OCCURRED
017612  001406                          BEQ UDAISG
017614                                  ERRDF 20,,ERR020
017614  104455                                                          TRAP    C$ERDF
017616  000024                                                          .WORD   20
017620  000000                                                          .WORD   0
017622  011154                                                          .WORD   ERR020
017624  000261                          SEC
017626  000424                          BR UDAISE
```

```
                                        ;SET UP LOOP PARAMETERS TO EXECUTE THE FOUR STEPS OF INITIALIZATION
                                        
017630  012737  004000  020152  UDAISG: MOV #SA.S1,UDARSD                ;STORE RESPONSE MASK
017636  012703  017704                  MOV #UDAIDT,R3                  ;AND INDEX TO TABLE
                                        
                                        ;WAIT FOR AND CHECK RESPONSE DATA
                                        
017642  004737  020014          UDAISL: CALL UDARSP                     ;WAIT FOR STEP OR ERROR BITS
017646  103414                          BCS UDAISE                      ;EXIT IF ERROR
017650  004733                          CALL @(R3)+                     ;CALL RESPONSE CHECKER FOR STEP
017652  103412                          BCS UDAISE                      ;GET OUT IF ERROR
017654  006337  020152                  ASL UDARSD                      ;SHIFT TO NEXT STEP BIT
017660  032737  040000  020152          BIT #SA.S4,UDARSD               ;CHECK IF NOW AT STEP 4
017666  001003                          BNE UDAISX                      ;GET OUT IF SO
017670  012364  000002                  MOV (R3)+,2(R4)                 ;WRITE DATA TO UDASA REGISTER
017674  000762                          BR UDAISL                       ;STAY IN LOOP
                                        
017676  000241          UDAISX: CLC                                     ;CLEAR CARRY FOR NO ERROR INDICATION
017700                  UDAISE: POP R1
017700  012601                                                          MOV (SP)+,R1
017702  000207                          RETURN
```

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

017704  017722                  UDAIDT: .WORD UDAIR1                    ;FIRST WORD RESPONSE CHECK ROUTINE
017706  000000                  UDAID1: .WORD 0                         ;FIRST WORD TO SEND TO UDASA
017710  017730                          .WORD UDAIR2                    ;SECOND WORD RESPONSE CHECK ROUTINE
017712  000000                  UDAID2: .WORD 0                         ;SECOND WORD TO SEND TO UDASA
017714  017750                          .WORD UDAIR3                    ;THIRD WORD RESPONSE CHECK ROUTINE
017716  100000                  UDAID3: .WORD SA.TST                    ;THIRD WORD TO SEND TO UDASA
017720  017766                          .WORD UDAIR4                    ;FOURTH WORD RESPONSE CHECK ROUTINE

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

017722  012701  004400          UDAIR1: MOV #SA.S1+SA.DI,R1             ;SET STEP ONE BIT
017726  000422                          BR UDAIRC                       ;NOW COMPARE

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

017730  013701  017706          UDAIR2: MOV UDAID1,R1                   ;GET WORD SENT TO UDASA
017734  000301                          SWAB R1                         ;GET HIGH 8 BITS
017736  042701  177400                  BIC #177400,R1
017742  052701  010000                  BIS #SA.S2,R1                   ;SET STEP 2 BIT
017746  000412                          BR UDAIRC                       ;NOW COMPARE

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

017750  013701  017706          UDAIR3: MOV UDAID1,R1                   ;GET WORD SENT TO UDASA
017754  042701  177400                  BIC #177400,R1                  ;JUST LOW 8 BITS
017760  052701  020000                  BIS #SA.S3,R1                   ;SET STEP 3 BIT
017764  000403                          BR UDAIRC                       ;NOW COMPARE

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

017766  010201                  UDAIR4: MOV R2,R1                       ;GET RESPONSE FROM UDA
017770  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

017774  020102                  UDAIRC: CMP R1,R2                       ;COMPARE THE DATA
017776  001405                          BEQ UDAIRX                      ;EXIT IF COMPARED CORRECTLY
020000                                  ERRDF 25,,ERR025                ;REPORT ERROR
020000  104455                                                                  TRAP    C$ERDF
020002  000031                                                                  .WORD   25
020004  000000                                                                  .WORD   0
020006  011366                                                                  .WORD   ERR025
020010  000261                          SEC
020012  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

020014                          UDARSP: PUSH R1                                         MOV R1,-(SP)
020014  010146
020016  052737  100000  020152          BIS #SA.ERR,UDARSD              ;SET ERROR BIT IN MASK WORD
020024  012700  000012                  MOV #10.,R0                     ;SET UP FOR 10 SECOND TIMEOUT
020030  010501                          MOV R5,R1                       ;POINT TO COUNTER IN CONTROLLER TABLE
020032  062701  000040                  ADD #C.TO,R1
020036  004737  017172                  CALL SETTO
020042                                  POP R1                                          MOV (SP)+,R1
020042  012601
020044  033764  020152  000002  UDARS1: BIT UDARSD,2(R4)                 ;LOOK AT ERROR AND STEP BIT
020052  001024                          BNE UDARS2                      ;BRANCH IF EITHER SET
020054                                  BREAK                                           TRAP    C$BRK
020054  104422
020056  005737  003232                  TST KW.CSR                      ;SEE IF CLOCK ON SYSTEM
020062  001770                          BEQ UDARS1
020064  023765  003244  000042          CMP KW.EL+2,C.TOH(R5)           ;CHECK IF TIME OUT OCCURRED
020072  101005                          BHI 1$
020074  001363                          BNE UDARS1
020076  023765  003242  000040          CMP KW.EL,C.TO(R5)
020104  103757                          BLO UDARS1
020106  016402  000002          1$:     MOV 2(R4),R2                    ;GET REGISTER CONTENTS
020112                                  ERRDF 22,,ERR022                ;REPORT TIME OUT ERROR
020112  104455                                                                  TRAP    C$ERDF
020114  000026                                                                  .WORD   22
020116  000000                                                                  .WORD   0
020120  011220                                                                  .WORD   ERR022
020122  000407                          BR UDARSE
```

GLOBAL SUBROUTINES SECTION

```
                                        ;CHECK IF ERROR BIT SET
                                        
020124  016402  000002          UDARS2: MOV 2(R4),R2                    ;GET REGISTER CONTENTS
020130  100006                          BPL UDARSX                      ;EXIT IF ERROR NOT SET
020132                                  ERRDF 21,,ERR021                ;REPORT ERROR INFO
020132  104455                                                                  TRAP    C$ERDF
020134  000025                                                                  .WORD   21
020136  000000                                                                  .WORD   0
020140  011166                                                                  .WORD   ERR021
020142  000261                  UDARSE: SEC
020144  000207                          RETURN
                                        
                                        ;NORMAL EXIT
                                        
020146  000241                  UDARSX: CLC                             ;CLEAR CARRY AS NO ERROR INDICATION
020150  000207                          RETURN
                                        
                                        ;LOCATION FOR STEP BIT MASK
                                        
020152  000000                  UDARSD: .WORD 0                         ;LOAD BY CALLING ROUTINE
```

```
                        ;CLOG
                        ;
                        ;COMPUTE LOGARITHMIC VALUE OF NUMBER TO BASE 2.
                        ;
                        ;INPUTS:
                        ;       R0 - LOGARITHM TO BE CONVERTED
                        ;OUTPUTS:
                        ;       R1 - VALUE OF 2 RAISED TO POWER OF INPUT NUMBER
                        ;
                        ;CLOG:  PUSH R0
                        ;       CLR R1                          ;SET UP ZERO START VALUE
                        ;       SEC                             ;WITH CARRY READY TO SHIFT IN
                        ;CLOGLP:        ROL R1                          ;SHIFT TO LEFT
                        ;       DEC R0                          ; UNTIL R0
                        ;       BPL CLOGLP                      ; GOES NEGATIVE
                        ;       POP R0
                        ;       RETURN
```

```
                                ;KW11I
                                ;
                                ;CLOCK INTERRUPT SERVICE ROUTINE
                                ;
020154                          BGNSRV KW11I
020154                                                                  KW11I::
020154  062737  000001  003242          ADD #1,KW.EL                  ;COUNT THE INTERRUPT
020162  005537  003244                  ADC KW.EL+2
020166  012777  000105  163036          MOV #KWOUT.,@KW.CSR           ;RESTART THE CLOCK
020174                          ENDSRV
020174                                                                  L10035:
020174  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
                                ;
020176  005737  003232  RNTIME: TST KW.CSR                      ;CHECK IF A CLOCK PRESENT
020202  001465                  BEQ RNTIMX                      ;BRANCH IF NOT
020204                          PUSH <R0,R3,R4,R5>
020204  010046                                                          MOV R0,-(SP)
020206  010346                                                          MOV R3,-(SP)
020210  010446                                                          MOV R4,-(SP)
020212  010546                                                          MOV R5,-(SP)
020214  013703  003242          MOV KW.EL,R3                    ;GET ELAPSED TIME
020220  013704  003244          MOV KW.EL+2,R4
020224  013700  003240          MOV KW.HZ,R0                    ;GET SPEED OF CLOCK
020230  004737  016334          CALL DIVIDE                     ;COMPUTE SECONDS OF ELAPSED TIME
020234  012700  000074          MOV #60.,R0                     ;NOW DIVIDE BY 60
020240  004737  016334          CALL DIVIDE                     ; TO COMPUTE MINUTES
020244                          PUSH R5                         ;SAVE REMAINDER AS SECONDS
020244  010546                                                          MOV R5,-(SP)
020246  004737  016334          CALL DIVIDE                     ;DIVIDE BY 60 AGAIN
020252                          PNT RNTIM,R3                    ;PRINT HOURS
020252  010346                                                          MOV R3,-(SP)
020254  004137  016266                                                  JSR R1,LPNT
020260  003672                                                          .WORD RNTIM
020262  000002                                                          .WORD PNT.CT
020264  020527  000011          CMP R5,#9.                      ;IF MINUTES 9 OR LESS
020270  003004                  BGT 1$
020272                          PRINT #'0                       ;PRINT A LEADING ZERO
020272  112700  000060                                                  MOVB #'0,R0
020276  004737  016056                                                  CALL CPNT
020302                  1$:     PNT RNTIM1,R5                   ;NOW PRINT MINUTES
020302  010546                                                          MOV R5,-(SP)
020304  004137  016266                                                  JSR R1,LPNT
020310  003715                                                          .WORD RNTIM1
020312  000002                                                          .WORD PNT.CT
020314                          POP R5                          ;GET SECONDS
020314  012605                                                          MOV (SP)+,R5
020316  020527  000011          CMP R5,#9.                      ;IF 9 OR LESS
020322  003004                  BGT 2$
020324                          PRINT #'0                       ;PRINT A LEADING ZERO
020324  112700  000060                                                  MOVB #'0,R0
020330  004737  016056                                                  CALL CPNT
020334                  2$:     PNT RNTIM2,R5                   ;NOW PRINT SECONDS
020334  010546                                                          MOV R5,-(SP)
020336  004137  016266                                                  JSR R1,LPNT
020342  003723                                                          .WORD RNTIM2
020344  000002                                                          .WORD PNT.CT
020346                          POP <R5,R4,R3,R0>               ;HOURS IN R3
020346  012605                                                          MOV (SP)+,R5
```

GLOBAL SUBROUTINES SECTION

```
                020350  012604                                                  MOV (SP)+,R4
                020352  012603                                                  MOV (SP)+,R3
                020354  012600                                                  MOV (SP)+,R0
     35         020356                  RNTIMX: PRINT <#' >             ;PRINT A SPACE
                020356  112700  000040                                          MOVB #' ,R0
                020362  004737  016056                                          CALL CPNT
     36         020366  000207                  RETURN
     37
     38         020370                          ENDMOD
```

PROTECTION TABLE

```
                                .SBTTL  PROTECTION TABLE

020370                                  BGNMOD

                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--

020370                                  BGNPROT
020370                                                                  L$PROT::

020370  177777                          -1                      ;OFFSET INTO P-TABLE FOR CSR ADDRESS
020372  177777                          -1                      ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
020374  177777                          -1                      ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

020376                                  ENDPROT
```

```
                                .SBTTL  INITIALIZE SECTION

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

020376                                  BGNINIT
020376                                                                  L$INIT::

                                        READEF #EF.START                ;CHECK IF STARTED BY OPERATOR
020376  012700  000040                                                  MOV     #EF.START,R0
020402  104447                                                         TRAP    C$REFG
020404                                  BCOMPLETE INIT0         ;  IF NOT,
020404  103415                                                         BCS     INIT0
020406                                  READEF #EF.RESTART
020406  012700  000037                                                  MOV     #EF.RESTART,R0
020412  104447                                                         TRAP    C$REFG
020414                                  BCOMPLETE INIT1
020414  103415                                                         BCS     INIT1
020416                                  READEF #EF.CONTINUE
020416  012700  000036                                                  MOV     #EF.CONTINUE,R0
020422  104447                                                         TRAP    C$REFG
020424                                  BCOMPLETE INIT1
020424  103411                                                         BCS     INIT1
020426                                  READEF #EF.PWR
020426  012700  000034                                                  MOV     #EF.PWR,R0
020432  104447                                                         TRAP    C$REFG
020434                                  BCOMPLETE INIT1
020434  103405                                                         BCS     INIT1
020436                          INITQT: DOCLN                           ;  ABORT PROGRAM
020436  104444                                                         TRAP    C$DCLN
020440                          INIT0:
020440  004737  012500                  CALL    CLOSEF  ; MAKE SURE  FILE IS CLOSED
020444  005037  002214                  CLR     FNUM    ; NO FILE READ
020450  012700  000003          INIT1:  MOV #SO.FMT,R0  ;BUILD MODE WORD FROM SOFTWARE QUESTIONS
020454  030037  002144                  BIT R0,SFPTBL   ;SEE IF REFORMAT
020460  001011                          BNE INIT2       ;  BRANCH IF SO
020462  012700  000004                  MOV #SO.CNS,R0  ;SEE IF RECONSTRUCT
020466  030037  002144                  BIT R0,SFPTBL
020472  001004                          BNE INIT2       ;  BRANCH IF SO
020474  006300                          ASL R0  ;SEE IF RESTORE
020476                                  .ASSUME SO.STR EQ SO.CNS*2
020476  030037  002144                  BIT R0,SFPTBL
020502  001755                          BEQ INITQT      ;IF NOT, ABORT PROGRAM
020504  010037  003230          INIT2:  MOV R0,MODE      ;SAVE MODE FLAGS
020510  005737  003344                  TST DATEQ       ;SEE IF ALREADY ASKED FOR DATE
020514  001002                          BNE INIT3
020516  004737  021430                  CALL DATE       ;IF NOT, GET IT NOW
020522  004737  012064          INIT3:  CALL    RESET                   ;RESET ALL UNITS
020526                                  MEMORY FFREE                    ;RESET START OF FREE MEMORY
020526  104431                                                         TRAP    C$MEM
020530  010037  002146                                                  MOV     R0,FFREE
020534  017737  161406  002150                MOV @FFREE,FSIZE                ;RESET SIZE OF FREE MEMORY

                                ;INITIALIZE CLOCK
```

```
                000105                          KWOUT.=105                      ;DATA TO SEND TO KW11 TO START CLOCK
020542  005037  003232                          CLR KW.CSR                      ;MARK CLOCK AS NOT ON SYSTEM
020546  005037  003242                          CLR KW.EL                       ;CLEAR ELAPSED TIME
020552  005037  003244                          CLR KW.EL+2
020556                                          CLOCK L,R0                      ;SEE IF AN L CLOCK PRESENT
020556  012700  000114                                                                  MOV     #'L,R0
020562  104462                                                                          TRAP    C$CLCK
020564                                          BCOMPLETE KYES
020564  103413                                                                          BCS     KYES
020566                                          CLOCK P,R0                      ;SEE IF A P CLOCK PRESENT
020566  012700  000120                                                                  MOV     #'P,R0
020572  104462                                                                          TRAP    C$CLCK
020574                                          BCOMPLETE KYES
020574  103407                                                                          BCS     KYES
020576  005037  003232                          CLR KW.CSR                      ;IF NEITHER, CLEAR CSR STORAGE WORD
020602                                          PNTF NOCLOCK
020602  004137  016230                                                                  JSR R1,LPNTF
020606  004061                                                                          .WORD NOCLOCK
020610  000000                                                                          .WORD PNT.CT
020612  000426                                  BR KNO
020614  012037  003232          KYES:           MOV (R0)+,KW.CSR                ;STORE DATA RETURNED
020620  012037  003234                          MOV (R0)+,KW.BRL
020624  012037  003236                          MOV (R0)+,KW.VEC
020630  012037  003240                          MOV (R0)+,KW.HZ
020634                                          SETVEC KW.VEC,#KW11I,KW.BRL     ;SET THE VECTOR
020634  013746  003234                                                                  MOV     KW.BRL,-(SP)
020640  012746  020154                                                                  MOV     #KW11I,-(SP)
020644  013746  003236                                                                  MOV     KW.VEC,-(SP)
020650  012746  000003                                                                  MOV     #3,-(SP)
020654  104437                                                                          TRAP    C$SVEC
020656  062706  000010                                                                  ADD     #10,SP
020662  012777  000105  162342                  MOV #KWOUT.,@KW.CSR             ;START THE CLOCK
020670                  KNO:
```

```
                                        ;INITIALIZE CONTROLLER TABLE STORAGE WITH A WORD OF ZEROS

020670  013737  002146  002156            MOV FFREE,CTABS             ;STORE START OF CONTROLLER TABLES
020676  005077  161254                    CLR @CTABS                  ;ZEROS MARKS END CONTROLLER TABLES
020702  005037  002160                    CLR CTRLRS                  ;CLEAR CONTROLLER COUNT
020706  012701  003474                    MOV     #IPADRS,R1          ; R1 -> IP ADDRESS
020712  012702  000010                    MOV     #8.,R2              ; R2 IS A COUNTER
020716  005021                    1$:     CLR     (R1)+               ; CLEAR ENTRY
020720  005302                            DEC     R2                  ; DONE?
020722  001375                            BNE     1$                  ; IF NOT, BRANCH

                                        ;GET A P-TABLE FROM DRS

020724  005002                            CLR R2                      ;LOGICAL UNIT NUMBER IN R2
020726                          INIT4:  GPHARD R2,R0                ;GET POINTER TO A P-TABLE
020726  010200                                                              MOV     R2,R0
020730  104442                                                              TRAP    C$GPHRD
020732                                  BNCOMPLETE NXTTAB           ;IGNORE IF NO TABLE RETURNED
020732  103110                                                              BCC     NXTTAB

                                        ;SEE IF A CONTROLLER TABLE ALREADY EXISTS FOR CONTROLLER IN P-TABLE

020734  013703  002156                    MOV CTABS,R3                ;GET ADDRESS OF CONTROLLER TABLES
020740  005713                  INIT5:    TST (R3)                    ;CHECK IF ANY MORE TABLES
020742  001416                            BEQ NEWTAB                  ;BUILD NEW TABLE IF FOUND ZERO WORD
020744  021013                            CMP (R0),(R3)               ;CHECK IF SAME UNIBUS ADDRESS
020746                                          ASSUME C.UADR EQ 0
020746                                          ASSUME H0.UBA EQ 0
020746  001463                            BEQ SAMTAB                  ;CHECK TABLE IF ALREADY EXISTS
020750  016301  000004                    MOV C.VEC(R3),R1            ;GET VECTOR FROM EXISTING CONTROLLER TABLE
020754  042701  177000                    BIC #^C<CT.VEC>,R1
020760  026001  000002                    CMP H0.VEC(R0),R1           ;SEE IF DIFFERENT VECTOR
020764  001002                            BNE 1$
020766  000137  021414                    JMP SAMVEC                  ;ERROR, CAN'T HAVE TWO UDA'S WITH SAME VECTOR
020772  062703  000054          1$:       ADD #C.SIZE,R3              ;MOVE TO NEXT TABLE
020776  000760                            BR INIT5
```

```
                                        ;BUILD A CONTROLLER TABLE
                                        
                                NEWTAB: MOV     #8.,R3          ;R3 IS A COUNTER
                                        MOV     #IPADRS,R4      ;R4 -> IP ADDRESSES
                                1$:     TST     (R4)            ; FOUND AN OPEN ENTRY?
                                        BEQ     2$              ; IF SO, GO FILL ENTRY
                                        TST     (R4)+           ; NEXT ENTRY
                                        DEC     R3              ; SEARCH THROUGH ENTIRE TABLE?
                                        BNE     1$              ; IF NOT, BRANCH
                                        BR      3$              ; ELSE, TABLE FULL
                                2$:     MOV     (R0),(R4)       ; STORE ENTRY INTO TABLE
                                3$:     MOV #C.SIZE/2,R1        ;GET WORDS IN CONTROLLER TABLE
                                        CALL ALOCM              ;ALLOCATE SPACE FOR IT
                                        MOV (R0),(R1)+          ;STORE UNIBUS ADDRESS
                                        MOV R2,(R1)+            ;UNIT NUMBER
                                        MOV H0.BRL(R0),R4       ;GET BR LEVEL
                                        SWAB R4                 ;SWAP TO HIGH BYTE
                                        ROL R4                  ;SHIFT ONE MORE TO LEFT
                                        BIS H0.VEC(R0),R4       ;ADD VECTOR ADDRESS
                                        MOV R4,(R1)+            ;  TO TABLE
                                        MOV H0.BST(R0),(R1)+
                                        MOV #4037,(R1)+         ;PUT [JSR R0,UDASRV]
                                        MOV #UDASRV,(R1)+       ;  INTO TABLE
                                        MOV #16.,R3             ;CLEAR POINTERS TO DRIVE TABLES,
                                                                ; TIMEOUT COUNTER, FLAGS, REF. NUMBER
                                INIT7:  CLR (R1)+
                                        DEC R3
                                        BNE INIT7               ;LOOP TIL ALL CLEARED
                                        INC CTRLRS              ;COUNT THE CONTROLLER
                                        CLR (R1)                ;CLEAR TABLE END MARKER
                                        BR NXTTAB               ;NOW GO TO NEXT P-TABLE
```

INITIALIZE SECTION

```
                                        ;SHOULD BE SAME CONTROLLER, CHECK THAT OTHER PARAMETERS MATCH

021116  016004  000004        SAMTAB: MOV HO.BRL(R0),R4       ;GET BR LEVEL FROM P-TABLE
021122  000304                        SWAB R4                 ;SWAP TO HIGH BYTE
021124  006104                        ROL R4                  ;SHIFT ONE MORE TO LEFT
021126  056004  000002                BIS HO.VEC(R0),R4       ;ADD VECTOR ADDRESS
021132  020463  000004                CMP R4,C.VEC(R3)        ;COMPARE WITH CONTROLLER TABLE
021136  001004                        BNE 1$
021140  026063  000006  000006       CMP HO.BST(R0),C.BST(R3) ;COMPARE BURST RATES
021146  001402                        BEQ NXTTAB
021150  000137  021344        1$:     JMP CTABER              ;FATAL ERROR IF NOT SAME

                                        ;GET NEXT P-TABLE

021154  005202                NXTTAB: INC R2                  ;INCREMENT LOGICAL UNIT NUMBER
021156  023702  002012                CMP L$UNIT,R2           ;CHECK IF GOT ALL TABLES
021162  003261                        BGT INIT4               ;IF NOT, GO BACK FOR NEXT

021164  012701  000001                MOV #1,R1               ;ALLOCATE SPACE FOR ZERO END WORD
021170  004737  011744                CALL ALOCM              ;AFTER CONTROLLER TABLES
```

```
                                        ;NOW BUILD DRIVE TABLES
                                        
021174  005002                          CLR R2                          ;LOGICAL UNIT NUMBER IN R2
021176                          INIT8:  GPHARD R2,R0            ;GET POINTER TO A P-TABLE
021176  010200                                                          MOV     R2,R0
021200  104442                                                          TRAP    C$GPHRD
021202                                  BNCOMPLETE INIT14               ;IF NOT AVAILABLE, GO GET NEXT
021202  103040                                                          BCC     INIT14
                                        
                                        ;FIND CONTROLLER TABLE
                                        
021204  013703  002156                  MOV CTABS,R3                    ;GET ADDRESS OF CONTROLLER TABLES
021210  021013                  INIT10: CMP (R0),(R3)           ;CHECK IF SAME UNIBUS ADDRESS
021212  001403                          BEQ INIT11                      ;BRANCH IF TABLE FOUND
021214  062703  000054                  ADD #C.SIZE,R3                  ;MOVE TO NEXT TABLE
021220  000773                          BR INIT10
```

INITIALIZE SECTION

```
                                        ;BUILD DRIVE TABLE
                                        
021222  012701  000015   INIT11: MOV #D.SIZE/2,R1        ;GET SIZE OF DRIVE TABLE
021226  004737  011744           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
021232  010337  003302           MOV R3,TEMP                     ;SAVE CONTROLLER TABLE ADDRESS
                                                                 ;IN CASE AN ERROR IS DETECTED
021236  062703  000020           ADD #C.DR0,R3                   ;BUILD POINTER TO C.DR ENTRY IN CONTROLLER TABLE
021242  012704  000010           MOV #8.,R4                      ;GET MAX COUNT OF DRIVES ON ONE CONTROLLER
021246  005713           INIT12: TST (R3)                ;CHECK IF ENTRY CONTAINS POINTER TO DRIVE TABLE
021250  001411                   BEQ INIT13
021252  026033  000010           CMP HD.LDR(R0),@(R3)+           ;CHECK DRIVE NUMBER IN DRIVE TABLE
021256  001002                   BNE 1$
021260  000137  021360           JMP MLDRER                      ;IF SAME, TWO P-TABLES POINT TO SAME DRIVE
021264  005304           1$:     DEC R4                          ;COUNT DRIVES
021266  001367                   BNE INIT12                      ;IF EIGHT DRIVE TABLES EXIST,
021270  000137  021376           JMP TOOMER                      ;  THEN REPORT ERROR
021274  010113           INIT13: MOV R1,(R3)             ;LOAD DRIVE TABLE POINTER
021276  016021  000010           MOV HD.LDR(R0),(R1)+            ;LOAD DRIVE NUMBER
021302  010221                   MOV R2,(R1)+                    ;LOAD UNIT NUMBER
```

```
      1                                 ;GO TO NEXT DRIVE TABLE
      2
      3 021304  005202                  INIT14: INC R2                  ;INCREMENT LOGICAL UNIT NUMBER
      4 021306  023702  002012                  CMP L$UNIT,R2                   ;CHECK IF GOT ALL TABLES
      5 021312  003331                          BGT INIT8                       ;IF NOT, GET NEXT TABLE
      6
      7                                 ;SAVE CURRENT PARAMETERS TO FREE MEMORY
      8
      9 021314  013737  002146  002152  INIT15: MOV FFREE,FMEM         ;SAVE START ADDRESS
     10 021322  013737  002150  002154          MOV FSIZE,FMEMS                 ;SAVE SIZE
     11
     12 021330                          INITXX: SETPRI  #PRI00                  ; SET RUNNING PRIORITY TO ZERO
        021330  012700  000000                                                          MOV     #PRI00,R0
        021334  104441                                                                  TRAP    C$SPRI
     14 021336                                  CLOSE                           ;MAKE SURE DATA FILE IS CLOSED
        021336  104435                                                                  TRAP    C$CLOS
     16 021340                                  EXIT INIT
        021340  104432                                                                  TRAP    C$EXIT
        021342  000524                                                                  .WORD   L10037-.
```

INITIALIZE SECTION

```
                                ;DIFFERENT VECTORS, BR LEVELS OR BURST RATES FOR ONE CONTROLLER
021344  010305                  CTABER: MOV R3,R5                       ;GET CONTROLLER ADDRESS
021346                                  ERRSF 1,,ERR001
021346  104454                                                                  TRAP    C$ERSF
021350  000001                                                                  .WORD   1
021352  000000                                                                  .WORD   0
021354  010770                                                                  .WORD   ERR001
021356                                  DOCLN
021356  104444                                                                  TRAP    C$DCLN

                                ;TWO P-TABLES FOR SAME DRIVE
021360  013705  003302          MLDRER: MOV TEMP,R5                     ;GET CONTROLLER ADDRESS
021364                                  ERRSF 2,,ERR002
021364  104454                                                                  TRAP    C$ERSF
021366  000002                                                                  .WORD   2
021370  000000                                                                  .WORD   0
021372  011006                                                                  .WORD   ERR002
021374                                  DOCLN
021374  104444                                                                  TRAP    C$DCLN

                                ;MORE THAN EIGHT DRIVES SELECTED ON ONE CONTROLLER

021376  013705  003302          TOOMER: MOV TEMP,R5                     ;GET CONTROLLER ADDRESS
021402                                  ERRSF 3,,ERR003
021402  104454                                                                  TRAP    C$ERSF
021404  000003                                                                  .WORD   3
021406  000000                                                                  .WORD   0
021410  011024                                                                  .WORD   ERR003
021412                                  DOCLN
021412  104444                                                                  TRAP    C$DCLN

                                ;TWO UDA'S USE THE SAME VECTOR

021414  010305                  SAMVEC: MOV R3,R5                       ;GET CONTROLLER ADDRESS
021416                                  ERRSF 8,ERR008
021416  104454                                                                  TRAP    C$ERSF
021420  000010                                                                  .WORD   8
021422  011100                                                                  .WORD   ERR008
021424  000000                                                                  .WORD   0
021426                                  DOCLN
021426  104444                                                                  TRAP    C$DCLN
```

```
     1 021430                          DATE:   GMANID DATEQ,DATEI,A,-1,1,11.,YES       ;GET DATE
       021430  104443                                                                          TRAP    C$GMAN
       021432  000406                                                                          BR      10000$
       021434  003330                                                                          .WORD   DATEI
       021436  000152                                                                          .WORD   T$CODE
       021440  003600                                                                          .WORD   DATEQ
       021442  177777                                                                          .WORD   -1
       021444  000001                                                                          .WORD   T$LOLIM
       021446  000013                                                                          .WORD   T$HILIM
       021450                                                                        10000$:
     2 021450  012705  003330                    MOV #DATEI,R5   ;GET POINTER TO ANSWER
     3 021454  121527  000060                    CMPB (R5),#'0
     4 021460  103443                           BLO DERR
     5 021462  122527  000071          DAY:     CMPB (R5)+,#'9
     6 021466  101040                           BHI DERR
     7 021470  121527  000055                    CMPB (R5),#'-
     8 021474  001406                           BEQ DAS1
     9 021476  121527  000060                    CMPB (R5),#'0
    10 021502  103432                           BLO DERR
    11 021504  122527  000071                    CMPB (R5)+,#'9
    12 021510  101027                           BHI DERR
    13 021512  122527  000055          DAS1:    CMPB (R5)+,#'-
    14 021516  001024                           BNE DERR
    15 021520  012704  000014                    MOV #12.,R4     ;GET NUMBER OF MONTH
    16 021524  012703  003405                    MOV #MONTHS,R3  ;GET POINTER TO MONTH NAMES
    17 021530  005000                 MON1:     CLR R0
    18 021532  121523                           CMPB (R5),(R3)+
    19 021534  001401                           BEQ MON2
    20 021536  005200                           INC R0
    21 021540  126523  000001          MON2:    CMPB 1(R5),(R3)+
    22 021544  001401                           BEQ MON3
    23 021546  005200                           INC R0
    24 021550  126523  000002          MON3:    CMPB 2(R5),(R3)+
    25 021554  001401                           BEQ MON4
    26 021556  005200                           INC R0
    27 021560  005700                 MON4:     TST R0
    28 021562  001407                           BEQ MON5
    29 021564  005304                           DEC R4
    30 021566  001360                           BNE MON1
    31 021570                          DERR:    PNTF DATEX
       021570  004137  016230                                                                  JSR R1,LPNTF
       021574  010670                                                                          .WORD DATEX
       021576  000000                                                                          .WORD PNT.CT
    32 021600  000713                           BR DATE
    33 021602  012701  003344          MON5:    MOV #DATEO,R1   ;GET POINTER TO DATE FOR FORMATTER
    34 021606  010403                           MOV R4,R3       ;GET COPY OF MONTH NUMBER
    35 021610  020327  000012                    CMP R3,#10.     ;  IF 10 OR GREATER
    36 021614  103404                           BLO MON6
    37 021616  112721  000061                    MOVB #'1,(R1)+  ;PUT A "1" IN OUTPUT
    38 021622  162703  000012                    SUB #10.,R3
    39 021626  062703  000060          MON6:    ADD #'0,R3      ;CONVERT MONTH NUMBER TO ASCII
    40 021632  110321                           MOVB R3,(R1)+   ;PUT A NUMBER IN OUTPUT
    41 021634  112721  000055                    MOVB #'-,(R1)+  ;PUT A "-" IN OUTPUT
    42 021640  062704  003450                    ADD #DAYS-1,R4  ;GET POINTER TO DAYS IN MONTH
    43                                                                 ;INDEXED BY NUMBER OF MONTH
    44 021644  012703  003330                    MOV #DATEI,R3   ;GET POINTER TO DATE INPUT
    45 021650  005000                           CLR R0
```

```
     46 021652  121327  000055          DAY1:   CMPB (R3),#'-
     47 021656  001413                          BEQ DAY2
     48 021660  111321                          MOVB (R3),(R1)+ ;PUT DAY CHARACTER IN OUTPUT
     49 021662  006300                          ASL R0
     50 021664  010002                          MOV R0,R2
     51 021666  006300                          ASL R0
     52 021670  006300                          ASL R0
     53 021672  060200                          ADD R2,R0
     54 021674  112302                          MOVB (R3)+,R2
     55 021676  162702  000060                  SUB #'0,R2
     56 021702  060200                          ADD R2,R0
     57 021704  000762                          BR DAY1
     58 021706  120014                  DAY2:   CMPB R0,(R4)
     59 021710  101327                          BHI DERR
     60 021712  005700                          TST R0  ;SEE IF DATE IS ZERO
     61 021714  001725                          BEQ DERR        ;ERROR IF SO
     62 021716  062705  000003                  ADD #3,R5
     63 021722  121527  000055                  CMPB (R5),#'-   ;CHECK FOR "-" BETWEEN DAY
     64 021726  001320                          BNE DERR        ;  AND YEAR IN OUTPUT
     65 021730  112521                          MOVB (R5)+,(R1)+        ;PUT "-" IN OUTPUT
     66 021732  010504                          MOV R5,R4       ;GET COPY OF INPUT STRING POINTER
     67 021734  005000                          CLR R0
     68 021736  005002                          CLR R2
     69 021740  121427  000060          YER1:   CMPB (R4),#'0
     70 021744  103416                          BLO YER2
     71 021746  121427  000071                  CMPB (R4),#'9
     72 021752  101013                          BHI YER2
     73 021754  006300                          ASL R0
     74 021756  010003                          MOV R0,R3
     75 021760  006300                          ASL R0
     76 021762  006300                          ASL R0
     77 021764  060300                          ADD R3,R0
     78 021766  112403                          MOVB (R4)+,R3
     79 021770  162703  000060                  SUB #'0,R3
     80 021774  060300                          ADD R3,R0
     81 021776  005202                          INC R2
     82 022000  000757                          BR YER1
     83 022002  105714                  YER2:   TSTB (R4)
     84 022004  001271                          BNE DERR
     85 022006  020227  000002                  CMP R2,#2
     86 022012  001407                          BEQ YER3
     87 022014  020227  000004                  CMP R2,#4
     88 022020  001263                          BNE DERR
     89 022022  020027  003554                  CMP R0,#1900.
     90 022026  103660                          BLO DERR
     91 022030  000413                          BR YER5
     92 022032  012702  003465          YER3:   MOV #YEAR19,R2
     93 022036  020027  000106                  CMP R0,#70.
     94 022042  103002                          BHIS YER4
     95 022044  012702  003470                  MOV #YEAR20,R2
     96 022050  105712                  YER4:   TSTB (R2)
     97 022052  001402                          BEQ YER5
     98 022054  112221                          MOVB (R2)+,(R1)+
     99 022056  000774                          BR YER4
    100 022060  112521                  YER5:   MOVB (R5)+,(R1)+
    101 022062  001376                          BNE YER5
    102 022064  000207                          RETURN
```

INITIALIZE SECTION

```
    103
    104 022066                          ENDINIT
         022066                                    L10037:
         022066  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.
                                ;--

 022070                                 BGNAUTO
 022070                                                                 L$AUTO::

 022070                                 ENDAUTO
 022070                                                                 L10040:
 022070  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.
                                ;--

022072                                  BGNCLN
022072                                                                  L$CLEAN::

022072                                  CLOSE                   ;CLOSE DATA FILE
022072  104435                                                          TRAP    C$CLOS
022074  004737  012064                  CALL    RESET           ;RESET ALL UDAS

022100                                  ENDCLN
022100                                                                  L10041:
022100  104412                                                          TRAP    C$CLEAN

022102                                  ENDMOD
```

```
                                .SBTTL  TEST 1: DUP PROGRAM DRIVER

                                        BGNMOD

                                        BGNTST
                                                                        T1::
                                        BIT #SO.FMT,MODE
                                        BNE T1FMT
                                        MANUAL
                                                                                TRAP    C$MANI
                                        BCOMPLETE T1GO
                                                                                BCS     T1GO
                                        ERRSF 10,,ERR010
                                                                                TRAP    C$ERSF
                                                                                .WORD   10
                                                                                .WORD   0
                                                                                .WORD   ERR010
                                        EXIT TST
                                                                                TRAP    C$EXIT
                                                                                .WORD   L10042-.
                                T1GO:   BIT #SO.STR,MODE
                                        BEQ T1CNS
                                        CMP L$UNIT,#1
                                        BEQ T1RST
                                        ERRSF 9,,ERR009
                                                                                TRAP    C$ERSF
                                                                                .WORD   9
                                                                                .WORD   0
                                                                                .WORD   ERR009
                                        EXIT TST
                                                                                TRAP    C$EXIT
                                                                                .WORD   L10042-.

                                T1RST:  PNTF FILNAM
                                                                                JSR R1,LPNTF
                                                                                .WORD FILNAM
                                                                                .WORD PNT.CT
                                        GMANID FILNAQ,FNAME2,A,-1,1,10.,NO      ;GET FILE NAME
                                                                                TRAP    C$GMAN
                                                                                BR      10000$
                                                                                .WORD   FNAME2
                                                                                .WORD   T$CODE
                                                                                .WORD   FILNAQ
                                                                                .WORD   -1
                                                                                .WORD   T$LOLIM
                                                                                .WORD   T$HILIM
                                                                        10000$:
                                        OPEN #FNAME2
                                                                                MOV     #FNAME2,R0
                                                                                TRAP    C$OPEN
                                        BR T1FMT
                                T1CNS:  MOV CTABS,R5
                                T1SER1: MOV R5,R4
                                        ADD #C.DRO,R4
                                        MOV #8.,R3
                                T1SER2: MOV (R4),R2     ;GET DRIVE TABLE POINTER
                                        BEQ T1SERN
```

```
                                        PNTF SERNUM,D.UNIT(R2),(R5),(R2)
022250  011246                                                          MOV (R2),-(SP)
022252  011546                                                          MOV (R5),-(SP)
022254  016246  000002                                                  MOV D.UNIT(R2),-(SP)
022260  004137  016230                                                  JSR R1,LPNTF
022264  004223                                                          .WORD SERNUM
022266  000006                                                          .WORD PNT.CT
                                            ASSUME C.UADR EQ 0
                                            ASSUME D.DRV EQ 0
                                T1SER3: GMANID SERNQ,TEMP,A,-1,1,20.,NO ;GET SERIAL NUMBER
022270  104443                                                          TRAP    C$GMAN
022272  000406                                                          BR      10001$
022274  003302                                                          .WORD   TEMP
022276  000142                                                          .WORD   T$CODE
022300  003662                                                          .WORD   SERNQ
022302  177777                                                          .WORD   -1
022304  000001                                                          .WORD   T$LOLIM
022306  000024                                                          .WORD   T$HILIM
022310                                                                  10001$:
                                        MOV #TEMP,R1
                                        CLR R0
                                T1SER4: TSTB (R1)
                                        BEQ T1SER5
                                        INC R0
                                        CMPB (R1),#'0
                                        BLO T1SER7
                                        CMPB (R1)+,#'9
                                        BLOS T1SER4
                                        BR T1SER7
                                T1SER5: CMP R0,#20.
                                        BLO T1SER8
                                        MOV #TEMP,R1
                                        MOV #HIGHEST,R0
                                T1SER6: CMPB (R1)+,(R0)+
                                        BEQ T1SER6
                                        BLO T1SER8
                                T1SER7: PRINTF #SERNX,#HIGHEST
022366  012746  003360                                                  MOV     #HIGHEST,-(SP)
022372  012746  010600                                                  MOV     #SERNX,-(SP)
022376  012746  000002                                                  MOV     #2,-(SP)
022402  010600                                                          MOV     SP,R0
022404  104417                                                          TRAP    C$PNTF
022406  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
                        T1FMT:  MOV CTABS,TSTTAB                ;GET FIRST TABLE ADDRESS
                                MOV CTRLRS,R1                   ;RUN DM PROGRAM ON ALL CONTROLLERS
```

```
                                        ; FIND OUT WHICH CONTROLLER IS WHICH
                                TSTCNT: MOV     R1,URUN
                                        MOV     FFREE,DMPROG
                                        MOV     #1,R1                   ; MUST READ AT LEAST ON DMPROG
                                        CALL    TINIT                   ; READ DMCODE
                                        MOV     URUN,UCNT
                                        MOV     TSTTAB,R5               ; POINT TO CONTROLLER TABLE
                                        CLR     U50UNI                  ; CLEAR NUMBER OF UDA50'S
                                        TST     C.UNIT(R5)
                                        BMI     2$                      ; IF UNIT DROPPED, BRANCH
                                        CALL    GTDUST                  ; GET DUST STATUS
                                        CMP     U52EXT,HC.MPK+P.DEXT(R4) ; IS IT A 52?
                                        BNE     1$                      ; IF NOT, BRANCH
                                        BIS     #CT.U52,C.FLG(R5)       ; SET FOR 52
                                        BIC     #CT.U50,C.FLG(R5)
                                        BR      2$
                                1$:
                                        BIC     #CT.U52,C.FLG(R5)       ; SET FOR 50
                                        BIS     #CT.U50,C.FLG(R5)
                                        INC     U50UNI                  ; KEEP TRACK IF WE NEED TO READ UDA50 FORMATTER
                                2$:     CLR     @(R5)                   ; INIT CONTROLLER
                                        ADD     #C.SIZE,R5              ; POINT TO NEXT CONTROLLER
                                        DEC     UCNT                    ; CONTINUE UNTIL DONE
                                        BNE     TSTCNT
                                ;
                                        MOV     #CT.U52,TSTTYP          ;INIT CONTROLLER TYPE
                                3$:     MOV     CTRLRS,UCNT             ; GET COMM AREA FOR EACH CONTROLLER
                                        MOV     CTABS,R5
                                        MOV     DMPROG,R1               ; GET DMSIZE
                                        MOV     (R1),DMSIZE
                                        ADD     4(R1),DMSIZE
                                        MOV     CTABS,TSTTAB            ; INIT PARAMETERS FOR RUNDM
                                        MOV     CTRLRS,R1
                                        CALL RUNDM                      ;  RUN ALL CONTROLLERS OF ONE TYPE AT ONCE
                                        BEQ 6$
                                        CALL RESPDM
                                6$:     ASL     TSTTYP                  ; NEXT CONTROLLER TYPE
                                        MOV     FMEM,FFREE              ; RESTORE FREE MEMORY
                                        MOV     FMEMS,FSIZE
                                        BIT     #CT.U50,TSTTYP          ; ALL DONE?
                                        BEQ     7$                      ; IF SO, EXIT
                                        TST     U50UNI                  ; DO WE CONTINUE FOR UDA50?
                                        BEQ     7$                      ; IF NOT, EXIT
                                        MOV     #2,R1
                                        CALL    TINIT                   ; ELSE READ IN UDA50 FORMATTER
                                        BR      3$
                                7$:     EXIT TST
022752 104432                                                           TRAP    C$EXIT
022754 000002                                                           .WORD   L10042-.
                                        ENDTST
022756                                                          L10042:
022756 104401                                                           TRAP    C$ETST
                                        ENDMOD
```

```
                                .SBTTL  HARDWARE PARAMETER CODING SECTION

022760                                  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.
                                ;--

022760                                  BGNHRD
022760  000027                                                  .WORD L10043-L$HARD/2
022762                                                  L$HARD::
                                ;FORMAT OF HARDWARE P-TABLE IS AS FOLLOWS:

022762                          TABLE                           ;START A TEBLE DEFINITION

022762                          ITEM HO.UBA     2               ; UNIBUS ADDRESS
022762                          ITEM HO.VEC     2               ; UDA VECTOR
022762                          ITEM HO.BRL     2               ; BR LEVEL
022762                          ITEM HO.BST     2               ; BURST RATE
022762                          ITEM HO.LDR     2               ; DRIVE NUMBER
022762                          END
```

HARDWARE PARAMETER CODING SECTION

```
     1 022762                          GPRMA   H.UBA,HO.UBA,O,160000,177774,YES        ;BUS ADDRESS
       022762  000031                                                                    .WORD   T$CODE
       022764  023040                                                                    .WORD   H.UBA
       022766  160000                                                                    .WORD   T$LOLIM
       022770  177774                                                                    .WORD   T$HILIM
     2 022772                          GPRMA   H.VEC,HO.VEC,O,4,774,YES          ; VECTOR
       022772  001031                                                                    .WORD   T$CODE
       022774  023066                                                                    .WORD   H.VEC
       022776  000004                                                                    .WORD   T$LOLIM
       023000  000774                                                                    .WORD   T$HILIM
     3 023002                          GPRMD   H.BRL,HO.BRL,D,-1,4.,7.,YES      ; BR LEVEL
       023002  002052                                                                    .WORD   T$CODE
       023004  023075                                                                    .WORD   H.BRL
       023006  177777                                                                    .WORD   -1
       023010  000004                                                                    .WORD   T$LOLIM
       023012  000007                                                                    .WORD   T$HILIM
     4 023014                          GPRMD   H.BST,HO.BST,D,-1,0.,63.,YES      ; BURST RATE
       023014  003052                                                                    .WORD   T$CODE
       023016  023106                                                                    .WORD   H.BST
       023020  177777                                                                    .WORD   -1
       023022  000000                                                                    .WORD   T$LOLIM
       023024  000077                                                                    .WORD   T$HILIM
     5 023026                          GPRMD   H.LDR,HO.LDR,D,-1,0.,255.,YES    ; DRIVE SELECT NUMBER
       023026  004052                                                                    .WORD   T$CODE
       023030  023130                                                                    .WORD   H.LDR
       023032  177777                                                                    .WORD   -1
       023034  000000                                                                    .WORD   T$LOLIM
       023036  000377                                                                    .WORD   T$HILIM
     6 023040                          ENDHRD
                                                                                         .EVEN
       023040                                                                    L10043:
     7
     8 023040      125     116     111  H.UBA:  .ASCIZ  \UNIBUS ADDRESS OF UDA\
     9 023066      126     105     103  H.VEC:  .ASCIZ  \VECTOR\
    10 023075      102     122     040  H.BRL:  .ASCIZ  \BR LEVEL\
    11 023106      125     116     111  H.BST:  .ASCIZ  \UNIBUS BURST RATE\
    12 023130      104     122     111  H.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.
                                ;--

023146                                  BGNSFT
023146  000022                                                    .WORD L10044-L$SOFT/2
023150                                                          L$SOFT::

                                ;FORMAT OF SOFTWARE P-TABLE IS AS FOLLOWS:

023150                          TABLE                           ;START A TABLE DEFINITION

023150                          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

023150                          END
```

```
      1 023150                          GPRML S.FMT,SO.BIT,SO.FM1,YES   ;REFORMAT?
        023150  000130                                                          .WORD   T$CODE
        023152  023365                                                          .WORD   S.FMT
        023154  000001                                                          .WORD   SO.FM1
      2 023156                          XFERT SWEND
        023156  017024                                                          .WORD   T$CODE
      3 023160                          GPRML S.NRF,SO.BIT,SO.FM2,YES   ;AGAIN - REFORMAT?
        023160  000130                                                          .WORD   T$CODE
        023162  023214                                                          .WORD   S.NRF
        023164  000002                                                          .WORD   SO.FM2
      4 023166                          XFERT SWEND
        023166  013024                                                          .WORD   T$CODE
      5 023170                          GPRML S.CNS,SO.BIT,SO.CNS,YES   ;RECONSTRUCT
        023170  000130                                                          .WORD   T$CODE
        023172  023444                                                          .WORD   S.CNS
        023174  000004                                                          .WORD   SO.CNS
      6 023176                          XFERT SWEND
        023176  007024                                                          .WORD   T$CODE
      7 023200                          GPRML S.RST,SO.BIT,SO.STR,YES   ;RESTORE?
        023200  000130                                                          .WORD   T$CODE
        023202  023507                                                          .WORD   S.RST
        023204  000010                                                          .WORD   SO.STR
      8 023206                          XFERT SWEND
        023206  003024                                                          .WORD   T$CODE
      9 023210                          DISPLAY S.NOF   ;WARNING
        023210  000003                                                          .WORD   T$CODE
        023212  023630                                                          .WORD   S.NOF
     10 023214                  SWEND:  ENDSFT                                  .EVEN
        023214                                                          L10044:
     11
     12 023214      015     012         S.NRF:  .BYTE 15,12
     13 023216      116     117     124         .ASCII\NOT USING EXISTING INFORMATION WILL DESTROY THE FACTORY BAD SECTOR\
     14 023320      015     012                 .BYTE 15,12
     15 023322      111     116     106         .ASCII\INFORMATION ON THE DISKS.\
     16 023353      015     012                 .BYTE 15,12
     17 023355      101     107     101         .ASCII\AGAIN - \
     18 023365      122     105     106 S.FMT:  .ASCIZ\REFORMAT USING EXISTING BAD SECTOR INFORMATION\
     19 023444      122     105     103 S.CNS:  .ASCIZ\RECONSTRUCT BAD SECTOR INFORMATION\
     20 023507      104     117     040 S.RST:  .ASCII\DO YOU HAVE A FILE ON THE SYSTEM LOAD DEVICE\
     21 023563      015     012                 .BYTE 15,12
     22 023565      040     103     117         .ASCIZ\ CONTAINING BAD SECTOR INFORMATION\
     23 023630      131     117     125 S.NOF:  .ASCIZ\YOU CANNOT PROCEED WITHOUT SUCH A FILE.\
     24 023700      122     105     123         .ASCIZ\RESTART PROGRAM AND SELECT TO REFORMAT OR RECONSTRUCT DISK.\
     25 023774      000                         .BYTE 0
     26                                         .EVEN
     27
     28                                 .DSABL AMA
     29 000000                          .PSECT END
```

PATCH AREA

```
                        .SBTTL  PATCH AREA
                
000000                  $PATCH::
        000050                  .REPT 40.
                                .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   L10047-./2-1
        000130                                                                L10045:
      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                                                                L10047:
     12
     13 000142                                  ENDSETUP
     14
     15
     16
     17
     18
     19
     20
     21          000001              .END
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  29440 WORDS  ( 115 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  70 PAGES
A:ZUDEC0,B:ZUDEC0/C=[20,0]SVC34R.MLB/P:1,ZUDEC0.DOC,ZUDEC0.MAC
```

CROSS REFERENCE TABLE (CREF V04.00)

```
$PATCH 143-3#
A1          1-5#      1-11      1-70      28-35     50-19     53-17     138-88
A2          1-6#      1-12      1-151     28-36     50-22     53-20     138-99
ADR        32-10#
ALOCM      58-16#    59-14     68-19     129-13    130-20    132-4
ASS         1-8#      1-10      1-11      1-12      1-16      1-70      1-151     27-37     28-34     28-35     28-36     48-7      50-16     50-19
           50-22     53-17     53-20     60-1      61-31     64-28     65-1      83-19     127-20    133-13    137-10    138-21    138-69    138-84
          138-88    138-94    138-99    138-108
ASSEMB     28-8      28-8
BAS        52-15#    93-5      93-5      93-5      94-5      94-5
BASL2      52-13#    94-5
BASL3      52-14#
BASLN      52-17#    93-5      94-5
BASNO      52-12#    93-5      94-5
BIT0       32-10#    141-19
BIT00      32-10     32-10#
BIT01      32-10     32-10#
BIT02      32-10     32-10#
BIT03      32-10     32-10#
BIT04      32-10     32-10#
BIT05      32-10     32-10#
BIT06      32-10     32-10#
BIT07      32-10     32-10#
BIT08      32-10     32-10#
BIT09      32-10     32-10#
BIT1       32-10#    43-23     141-20
BIT10      32-10#
BIT11      32-10#
BIT12      32-10#
BIT13      32-10#
BIT14      32-10#
BIT15      32-10#    43-15     44-12     64-27     72-20     80-26     85-15     100-29
BIT2       32-10#    43-24     141-22
BIT3       32-10#    43-25     141-23
BIT4       32-10#    43-27
BIT5       32-10#    43-30
BIT6       32-10#    43-31
BIT7       32-10#    43-33
BIT8       32-10#    43-34
BIT9       32-10#    43-35
BLDC0     108-22    108-24#
BLDC1     108-26#   108-28
BLDCMD     63-18     71-50     74-14     74-44     106-3     108-15#
BOE        32-10#
C$AU       28-8#
C$AUTO     28-8#     136-12
C$BRK      28-8#     61-12     66-19     71-9      111-21    118-12    121-27
C$BSEG     28-8#
C$BSUB     28-8#
C$CEFG     28-8#
C$CLCK     28-8#     127-48    127-50
C$CLEA     28-8#     137-15
C$CLOS     28-8#     65-13     83-20     133-14    137-11
C$CLP1     28-8#
C$CVEC     28-8#     61-22     118-30
C$DCLN     28-8#     57-8      61-30     62-21     70-7      127-18    134-4     134-9     134-15    134-21
```

CROSS REFERENCE TABLE (CREF V04.00)

```
C$DODU   28-8#
C$DRPT   28-8#
C$DU     28-8#
C$EDIT   28-8#     28-34
C$ERDF   28-8#     61-28    71-25    72-36    73-21    74-8     78-22    80-30    107-3    111-29   112-5    116-35   117-11   118-33
        120-44    121-36   122-5
C$ERHR   28-8#
C$ERRO   28-8#
C$ERSF   28-8#     57-7     62-20    70-6     134-3    134-8    134-14   134-20   138-10   138-16
C$ERSO   28-8#
C$ESCA   28-8#
C$ESEG   28-8#
C$ESUB   28-8#
C$ETST   28-8#     138-138
C$EXIT   28-8#     133-16   138-11   138-17   138-137
C$GETB   28-8#     66-21    66-26    67-7     68-23    68-30    69-1     69-11    69-14    83-26
C$GETW   28-8#
C$GMAN   28-8#     135-1    138-20   138-34
C$GPHR   28-8#     128-15   131-4
C$GPLO   28-8#
C$GPRI   28-8#
C$INIT   28-8#     135-104
C$INLP   28-8#
C$MANI   28-8#     138-8
C$MEM    28-8#     127-39
C$MSG    28-8#     55-16    55-20    55-24    55-28    55-32    55-36    55-40    55-44    55-48    55-52    55-56    55-65    55-71
         55-85     55-89    55-93    55-97    55-101   55-106   55-110   55-114   55-118   55-122   55-126   55-130
C$OPEN   28-8#     66-16    83-21    138-22
C$PNTB   28-8#     102-14
C$PNTF   28-8#     102-12   138-52
C$PNTS   28-8#     102-18
C$PNTX   28-8#     55-70    102-16
C$QIO    28-8#
C$RDBU   28-8#
C$REFG   28-8#     127-10   127-12   127-14   127-16
C$RESE   28-8#     28-8#
C$REVI   28-8#     28-34
C$RFLA   28-8#
C$RPT    28-8#
C$SEFG   28-8#
C$SPRI   28-8#     133-12
C$SVEC   28-8#     61-11    63-9     105-20   118-27   127-59
C$TPRI   28-8#
C.BST    43-19#    117-17   130-9
C.DR0    43-37#    72-15    85-12    132-11   138-27
C.DR1    43-38#
C.DR2    43-39#
C.DR3    43-40#
C.DR4    43-41#
C.DR5    43-42#
C.DR6    43-43#
C.DR7    43-44#
C.FLG    43-22#    64-23*   64-29    71-11    71-14    71-16    71-39    71-44    71-48*   72-12*   73-6     73-26    74-35*   74-39*
         74-40     74-47*   74-50*   74-62*   74-63    74-69*   74-70    106-13*  106-14*  109-21*  111-17   112-1*   138-85*  138-86*
        138-95*   138-96*
C.JAD    43-21#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
C.JSR    43-20#   63-8     105-19
C.PRI    43-48#   74-65     74-67    74-72*   74-73*
C.REF    43-49#   73-19    109-16*  109-17
C.RING   43-36#   55-146    59-15*   71-10    74-29   108-16   109-15   116-20   117-3    117-22   118-20
C.SIZE   43-51#   64-37     72-5     128-32   129-12  131-12   138-62   138-104
C.TO     43-45#   71-37     74-55    111-14   111-27  121-22   121-33
C.TOH    43-46#   71-34    111-24    121-30
C.TOT    43-47#   71-30     74-53    105-14*  105-15*
C.UADR   43-12#   86-16    118-25    128-24   138-32
C.UNIT   43-13#   64-24     64-25     71-13   138-79
C.VEC    43-16#   63-5     105-16    118-14   128-27   130-7
CALR4    90-6     93-5#
CALR5    90-7     94-5#
CALR6    90-8     95-4#
CALR7    90-9     96-4#
CALR8    90-10    97-4#
CALRE    90-3     90-4      90-5     92-4#
CF.576   39-39#
CF.ATN   39-34#
CF.MSC   39-35#
CF.OTH   39-36#
CF.SHD   39-38#
CF.THS   39-37#
CLOSEF   65-11#   70-1     127-21
CLRBFL  110-22#  110-24
CLRBUF   74-16    74-46    110-16#
CON.A    88-12#   91-21
CON.A1   88-13#   88-15
CON.A2   88-17    88-19#
CON.D    88-23#   91-21
CON.H    88-29#   91-21
CON.N    89-9#    91-21
CON.N1   89-10#   89-12
CON.O    89-3#    91-21
CON.QU   88-3#    88-7     91-21
CON.QX   88-5     88-8#
CON.R    89-17#   91-21
CON.R1   89-19    89-23#
CON.S    89-29#   91-21
CON.S1   89-30#   89-32
CPNT     84-13    84-18     86-21    88-6     88-13   89-10   89-30   93-7    94-7    101-13   101-22   102-5#   125-27   125-32
         125-35
CPNTX   102-13   102-15    102-17   102-19#
CR       32-12#   84-13     86-21    89-10    93-7    94-7    102-8
CT.AVL   43-15#   64-27
CT.BRL   43-18#
CT.CMD   43-24#   71-16    106-13   109-21
CT.MSG   43-25#   71-14     74-39    74-62   111-17  112-1   114-19
CT.REQ   43-27#   73-6      73-26    74-35    74-40   74-47   74-50   106-13
CT.RN    43-23#   71-11    106-14
CT.STA   43-30#   71-39     71-47    74-39    74-62
CT.TM1   43-31#   71-43     71-44    74-39    74-70
CT.TM2   43-33#   74-39     74-63    74-69
CT.U50   43-35#   64-23    138-86   138-96   138-123
CT.U52   43-34#   64-23    138-85   138-95   138-109
CT.UNT   43-14#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
CT.VEC   43-17#   63-6     105-17   118-15   128-28
CTABER  130-11    134-2#
CTABS    48-15#   128-3*   128-4*   128-20   131-9    138-25   138-65   138-111  138-115
CTRLRS   48-16#   128-5*   129-29*  138-66   138-110  138-116
D.DRV    44-9#    79-36    86-17    138-33
D.SERN   44-13#   78-17    138-54
D.SIZE   44-15#   132-3
D.UNIT   44-10#   72-19    72-25*   85-14    86-15    138-31
DAS1    135-8     135-13#
DATE    127-37    135-1#   135-32
DATEI    49-2#    135-1    135-2    135-44
DATEO    49-4#    78-26    127-35   135-33
DATEQ    51-3#    135-1
DATEX    54-12#   135-31
DAY     135-5#
DAY1    135-46#   135-57
DAY2    135-47    135-58#
DAYS     49-19#   135-42
DERR    135-4     135-6    135-10   135-12   135-14   135-31#  135-59   135-61   135-64   135-84   135-88   135-90
DF.ACT   42-21#   74-60
DF.LCL   42-24#
DF.NES   42-22#
DF.SA    42-26#
DFPTBL   30-10#
DIAG     36-33#   108-23
DIAGMC   28-8     28-8
DIVIDE   79-38    100-39   101-7    104-16#  125-19   125-21   125-23
DMEND    48-21#   60-20    62-16*
DMENDS   48-22#   60-21    62-17*
DMFRST   42-47#   68-11    68-21
DMMAIN   42-46#
DMOVRL   42-44#
DMPROG   48-19#   62-13*   105-13   106-1    106-4    106-6    106-7    138-70*  138-112
DMSIZE   48-20#   138-113* 138-114*
DMTMO    42-45#   105-14
DMTRLN   42-43#
DQCONT   79-28    79-61#
DQNO     79-54    79-62#
DQRFMT   79-26    79-53#
DQRSTR   79-27    79-59#
DQUEJP   79-21    79-22#   79-29
DQUEST   75-4     79-16#
DQUESZ   79-18    79-29#
DQUEX    79-19    79-22    79-24    79-25    79-50#   79-57    79-64
DQUNIT   79-23    79-33#
DQUNL1   79-38#   79-42
DQUNL2   79-44#   79-48
DQYES    79-55#   79-60
DSPSIZ   75-9#
DT.AVL   44-12#   72-20    72-25    85-15
DT.UNT   44-11#
DU.DFL   42-33#
DU.FTL   42-36#
DU.INF   42-34#
DU.QUE   42-32#
DU.SPC   42-37#   74-6
```

CROSS REFERENCE TABLE (CREF V04.00)

```
DU.TER   42-35#
DU.TYP   42-39#    74-4      74-19
DUP      36-34#   108-20
E$END    28-8#
E$LOAD   28-8#     28-34
EF.BBR   39-27#
EF.BBU   39-28#
EF.CON   32-10#   127-14
EF.LOG   39-29#
EF.NEW   32-10#
EF.PWR   32-10#   127-16
EF.RES   32-10#   127-12
EF.SEX   39-30#
EF.STA   32-10#   127-10
ERR001   55-14#   134-3
ERR002   55-18#   134-8
ERR003   55-22#   134-14
ERR004   55-26#    57-7
ERR005   55-30#    70-6
ERR007   55-34#    62-20
ERR008   55-38#   134-20
ERR009   55-42#   138-16
ERR010   55-46#   138-10
ERR011   55-50#
ERR020   55-54#    61-28    118-33
ERR021   55-58#   122-5
ERR022   55-67#   121-36
ERR023   55-73#   117-11
ERR024   55-87#   116-35
ERR025   55-91#   120-44
ERR030   55-95#    71-25
ERR031   55-99#    72-36
ERR032   55-103#   74-8
ERR033   55-108#   73-21
ERR034   55-112#  107-3
ERR036   55-116#  111-29
ERR037   55-120#  112-5
ERR100   55-124#   78-22
ERR101   55-128#   80-30
ERR23A   55-76#    55-83
ERR23B   55-77     55-81#
ERR23C   55-80     55-84#
ERRC     87-36     87-43     91-9#
ERRCHR   48-49#   102-5*    102-12    102-14    102-16    102-18
ERRD     87-45     91-21#
ERRME1   52-8#     87-41     89-23     92-4
ERRNL    52-4#    102-10
ERRONE   52-3#    102-7
ERRRSZ   89-18     90-12#
ERRRTB   89-21     90-3#     90-12
ERRTRM   75-7      82-14#
EVL      32-10#
F$AU     28-8#
F$AUTO   28-8#    136-10    136-12
F$BGN    28-8#     28-26     31-16     32-3      55-14     55-18     55-22     55-26     55-30     55-34     55-38     55-42     55-46     55-50
         55-54     55-58     55-67     55-73     55-87     55-91     55-95     55-99     55-103    55-108    55-112    55-116    55-120    55-124
```

CROSS REFERENCE TABLE (CREF V04.00)

```
        55-128  113-10  114-18  124-5   125-38  126-3   126-10  127-8   133-16  136-10  137-8   137-17  138-3   138-5
       138-11   138-17  138-137 138-138 138-139 139-3   139-14  141-12  142-9   142-9   143-10  144-1   144-3   144-3
       144-11   144-13
F$CLEA  28-8#   137-8   137-15
F$DU    28-8#
F$END   28-8     28-8    28-8    28-8    28-8    28-8    28-8    28-8    28-8    28-8    28-8    28-8    28-8    28-8
        28-8     28-8    28-8#   28-26   31-16   32-3    55-16   55-20   55-24   55-28   55-32   55-36   55-40   55-44
        55-48    55-52   55-56   55-65   55-71   55-85   55-89   55-93   55-97   55-101  55-106  55-110  55-114  55-118
        55-122   55-126  55-130  113-14  114-21  124-9   125-38  126-3   133-16  135-104 136-12  137-15  137-17  138-3
       138-5    138-5   138-5   138-11  138-17  138-137 138-138 138-138 138-139 139-3   140-6   142-10  143-10  144-1
       144-3    144-11  144-13
F$HARD  28-8#   139-14  140-6   142-2   142-4   142-6   142-8
F$HW    28-8#    30-10   30-17
F$INIT  28-8#   127-8   135-104
F$JMP   28-8#   133-16  138-11  138-17  138-137
F$MOD   28-8#    28-26   31-16   32-3   125-38  126-3   137-17  138-3   138-139 139-3   143-10
F$MSG   28-8#    55-14   55-16   55-18   55-20   55-22   55-24   55-26   55-28   55-30   55-32   55-34   55-36   55-38
        55-40    55-42   55-44   55-46   55-48   55-50   55-52   55-54   55-56   55-58   55-65   55-67   55-71   55-73
        55-85    55-87   55-89   55-91   55-93   55-95   55-97   55-99   55-101  55-103  55-106  55-108  55-110  55-112
        55-114   55-116  55-118  55-120  55-122  55-124  55-126  55-128  55-130
F$PROT  28-8#   126-10  126-16
F$PWR   28-8#
F$RPT   28-8#
F$SEG   28-8#
F$SOFT  28-8#   141-12  142-2   142-4   142-6   142-8   142-10
F$SRV   28-8#   113-10  113-14  114-18  114-21  124-5   124-9
F$SUB   28-8#
F$SW    28-8#    31-10   31-14
F$TEST  28-8#   138-5   138-138
FCTBUF  48-36#   83-25   83-34
FCTNUM  48-37#   64-41*  83-14   83-23*  83-30*
FDATA   48-35#   67-5    68-2    68-9    68-20   69-13*  69-16*
FFREE   48-11#   55-74   58-16   58-20*  60-16*  60-20*  62-13   62-16   67-11   68-16   127-39* 127-40  128-3   133-9
       138-70   138-121*
FILNAM  54-13#  138-19
FILNAQ  51-4#   138-20
FILOPN  48-26#   65-11   65-14*  66-14   66-17*
FMEM    48-13#   60-16  133-9*  138-121
FMEMS   48-14#   60-17  133-10* 138-122
FMERR   57-7#    58-18
FNAME   48-52#   66-16
FNAME2  48-51#   83-21  138-20  138-22
FNUM    48-31#   60-18   60-25*  127-22*
FS       1-4#     1-8     1-10    1-16   27-37   28-34   48-7    50-16   60-1    61-31   64-28   65-1    83-19   127-20
       133-13   137-10  138-21  138-69  138-84  138-94  138-108
FSIZE   48-12#   58-17*  60-17*  60-21*  62-17   127-40* 133-10  138-122*
FWORD   67-4     67-6    68-1    68-3    69-11#
G$CNTO  28-8#
G$DELM  28-8#
G$DISP  28-8#   142-9
G$EXCP  28-8#
G$HILI  28-8#
G$LOLI  28-8#
G$NO    28-8#   138-20  138-34
G$OFFS  28-8#   135-1   138-20  138-34  140-1   140-2   140-3   140-4   140-5   142-1   142-3   142-5   142-7
G$OFSI  28-8#   135-1   138-20  138-34  140-1   140-2   140-3   140-4   140-5   142-1   142-3   142-5   142-7
```

CROSS REFERENCE TABLE (CREF V04.00)

```
G$PRMA   28-8#    140-1     140-2
G$PRMD   28-8#    135-1     138-20    138-34    140-3     140-4     140-5
G$PRML   28-8#    142-1     142-3     142-5     142-7
G$RADA   28-8#    135-1     138-20    138-34
G$RADB   28-8#
G$RADD   28-8#    140-3     140-4     140-5
G$RADL   28-8#    142-1     142-3     142-5     142-7
G$RADO   28-8#    140-1     140-2
G$XFER   28-8#    142-2     142-4     142-6     142-8
G$YES    28-8#    135-1     140-1     140-2     140-3     140-4     140-5     142-1     142-3     142-5     142-7
GETCDN   99-16     99-18     99-28#
GETCNT   88-12     89-9      89-17     89-29     99-13#    100-15
GETCNX   99-15#    99-27
GETCXX   99-29     99-31#
GTDRVL   85-13#    85-16
GTDRVT   78-16     79-16     80-22     85-11#
GTDUST   63-5#    138-81
H.BRL    140-3    140-10#
H.BST    140-4    140-11#
H.LDR    140-5    140-12#
H.UBA    140-1    140-8#
H.VEC    140-2    140-9#
HC.BF1    36-24#    36-25     74-15
HC.BF2    36-25#    36-27     55-147    74-3      74-45
HC.BSZ    36-13#    36-25     36-27     74-18     110-19    110-21
HC.CCT    36-19#    71-41     71-51*    109-19*
HC.CEV    36-22#    36-23     108-18
HC.CMD    36-18#    36-19     36-20     117-25*   117-26*
HC.CPK    36-23#    36-24     55-134    74-33*    106-4*    106-5*    106-6*    106-7*    108-29*   109-17*   110-18*   110-19*   117-26
HC.ESZ    36-11#    36-21     36-23
HC.INT    36-15#    36-16
HC.ISZ    36-9#     36-16     116-21    117-4
HC.MCT    36-17#    74-74*    109-18*
HC.MEV    36-20#    36-21
HC.MPK    36-21#    36-22     55-136    73-9      73-14     73-19     74-11     74-58     74-60     74-65     74-67     74-72     74-73     106-11
          117-24    138-82
HC.MSG    36-16#    36-17     36-18     117-23*   117-24*   118-21
HC.PSZ    36-12#    36-22     36-24     108-19    108-25
HC.RSZ    36-10#    36-18     36-20     116-21    117-4
HC.SIZ    36-27#    59-13
HCOMM     59-13#    63-13
HEADER    80-24     86-13#
HIGHES    49-6#    138-48    138-52
HO.BRL   129-16    130-3     139-22#   140-3     140-3     140-3
HO.BST   129-21    130-9     139-23#   140-4     140-4     140-4
HO.LDR   132-15    132-22    139-24#   140-5     140-5     140-5
HO.UBA   128-25    139-20#   140-1     140-1     140-1
HO.VEC   128-29    129-19    130-6     139-21#   140-2     140-2     140-2
HOE       32-10#
I$AU      28-8#
I$AUTO    28-8#    136-10#   136-12#
I$CLN     28-8#    137-8#    137-15#
I$DU      28-8#
I$HRD    139-14#   140-6#    142-9
I$INIT    28-8#    127-8#    133-16    135-104#
I$MOD     28-8#     28-26     28-26#    31-16     31-16#    32-3      32-3#     125-38    125-38#   126-3     126-3#    137-17    137-17#   138-3
```

CROSS REFERENCE TABLE (CREF V04.00)

```
        138-3#   138-139  138-139# 139-3    139-3#   143-10   143-10#
I$MSG    28-8#    55-14#   55-16#   55-18#   55-20#   55-22#   55-24#   55-26#   55-28#   55-30#   55-32#   55-34#   55-36#   55-38#
         55-40#   55-42#   55-44#   55-46#   55-48#   55-50#   55-52#   55-54#   55-56#   55-58#   55-65#   55-67#   55-71#   55-73#
         55-85#   55-87#   55-89#   55-91#   55-93#   55-95#   55-97#   55-99#   55-101#  55-103#  55-106#  55-108#  55-110#  55-112#
         55-114#  55-116#  55-118#  55-120#  55-122#  55-124#  55-126#  55-128#  55-130#
I$PROT   28-8#   126-10#
I$PTAB   28-8#   144-3    144-3#   144-11   144-11#
I$PWR    28-8#
I$RPT    28-8#
I$SEG    28-8#   138-5
I$SETU   28-8#   144-1    144-1#   144-3    144-13   144-13#
I$SFT   141-12#  142-9    142-10#
I$SRV    28-8#   113-10#  113-14#  114-18#  114-21#  124-5#   124-9#
I$SUB    28-8#   138-5
I$TST    28-8#   138-5    138-5#   138-11   138-17   138-137  138-138  138-138# 138-138#
IBE      32-10#
IDU      32-10#
IER      32-10#
INFO     75-5     80-15#   81-14    82-14
INFOE    80-18    80-30#
INFOP    80-20    80-25#
INFOX    80-16    80-26#
INIT0   127-11   127-19#
INIT1   127-13   127-15   127-17   127-24#
INIT10  131-10#  131-13
INIT11  131-11   132-3#
INIT12  132-13#  132-19
INIT13  132-14   132-21#
INIT14  131-5    133-3#
INIT15  133-9#
INIT2   127-26   127-29   127-34#
INIT3   127-36   127-38#
INIT4   128-15#  130-17
INIT5   128-21#  128-33
INIT7   129-26#  129-28
INIT8   131-4#   133-5
INITQT  127-18#  127-33
INITXX  133-12#
INMEM    48-32#
IPADRS   49-34#   61-14    128-6    129-4
ISR      32-10#
IXE      32-10#
J$JMP    28-8#
KNO     127-54   127-61#
KW.BRL   48-43#  127-56*  127-59
KW.CSR   48-42#   71-32    86-19   111-22   121-28   124-8*   125-13   127-45*  127-52*  127-55*  127-60*
KW.EL    48-46#   71-34    71-37   111-24   111-27   115-26   115-27   115-28   121-30   121-33   124-6*   124-7*   125-16   125-17
        127-46*  127-47*
KW.HZ    48-45#  115-16   125-18   127-58*
KW.VEC   48-44#  127-57*  127-59
KW11I   124-5#   127-59
KWOUT.  124-8    127-44#  127-60
KYES    127-49   127-51   127-55#
L$ACP    28-34#
L$APT    28-34#
L$AUT    28-34#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
L$AUTO  28-34   136-10#
L$CCP   28-34#
L$CLEA  28-34   137-8#
L$CO    28-34#
L$DEPO  28-34#
L$DESC  28-34    50-17#
L$DESP  28-34#
L$DEVP  28-34#
L$DISP  28-34    29-9#
L$DLY   28-34#
L$DTP   28-34#
L$DTYP  28-34#
L$DUT   28-34#
L$DVTY  28-34    50-12#
L$EF    28-34#
L$ENVI  28-34#
L$ETP   28-34#
L$EXP1  28-34#
L$EXP4  28-34#
L$EXP5  28-34#
L$HARD  28-34   139-14   139-14#
L$HIME  28-34#
L$HPCP  28-34#
L$HPTP  28-34#
L$HW    28-34    30-10    30-10#
L$ICP   28-34#
L$INIT  28-34   127-8#
L$LADP  28-34#
L$LAST  28-34   143-8#   144-13
L$LOAD  28-34#
L$LUN   28-34#   64-24*   71-13*   85-14*
L$MREV  28-34#
L$NAME  28-34#
L$PRIO  28-34#
L$PROT  28-34   126-10#
L$PRT   28-34#
L$REPP  28-34#
L$REV   28-34#
L$SOFT  28-34   141-12   141-12#
L$SPC   28-34#
L$SPCP  28-34#
L$SPTP  28-34#
L$STA   28-34#
L$SW    28-34    31-10    31-10#
L$TEST  28-34#
L$TIML  28-34#
L$UNIT  28-34#   86-13   130-16   133-4    138-14
L10000  30-10    30-17#
L10001  31-10    31-14#
L10002  55-16#
L10003  55-20#
L10004  55-24#
L10005  55-28#
L10006  55-32#
L10007  55-36#
L10010  55-40#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
L10011  55-44#
L10012  55-48#
L10013  55-52#
L10014  55-56#
L10015  55-65#
L10016  55-71#
L10017  55-85#
L10020  55-89#
L10021  55-93#
L10022  55-97#
L10023  55-101#
L10024  55-106#
L10025  55-110#
L10026  55-114#
L10027  55-118#
L10030  55-122#
L10031  55-126#
L10032  55-130#
L10033 113-14#
L10034 114-21#
L10035 124-9#
L10037 133-16   135-104#
L10040 136-12#
L10041 137-15#
L10042 138-11   138-17   138-137  138-138#
L10043 139-14   140-6#
L10044 141-12   142-10#
L10045 144-3#
L10047 144-3    144-11#
LDDM    64-22#   64-39
LDNEXT  64-26    64-30    64-35    64-37#
LOADB  106-2#
LOADDM  64-34    72-30   105-13#
LOADE1 106-12   107-3#
LOADER  63-16   105-23   106-10   107-4#
LOE     32-10#
LOG     36-32#
LOT     32-10#
LPNT   103-6    103-8    103-10   103-12#  125-24   125-28   125-33
LPNTB   55-15    55-19    55-23    55-27    55-31    55-35    55-39    55-43    55-47    55-51    55-55    55-64    55-69    55-74
        55-78    55-84    55-88    55-92    55-96    55-100   55-104   55-117   55-121   55-125   55-129   55-132   55-138   55-145
        55-149   93-5     94-5    103-7#
LPNTF   86-15    87-41    89-23    92-4     103-5#   127-53   135-31   138-19   138-31
LPNTS  103-11#
LPNTX  103-9#
MD.CMP  39-4#
MD.CWB  39-22#
MD.ERR  39-6#
MD.EXP  39-5#
MD.FEU  39-16#
MD.IMF  39-20#
MD.NXU  39-18#
MD.PRI  39-23#
MD.RIP  39-19#
MD.SCH  39-7#
MD.SCL  39-8#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
MD.SEC   39-9#
MD.SEQ   39-14#
MD.SER   39-10#
MD.SPD   39-15#
MD.SSH   39-11#
MD.SWP   39-21#
MD.VOL   39-17#
MD.WBN   39-12#
MD.WBV   39-13#
MESG     80-25    84-13#
MESSG    52-9#    86-15
MLDRER  132-17   134-7#
MODE     48-38#   79-53    79-59   127-34*  138-6    138-12
MON1    135-17#  135-30
MON2    135-19   135-21#
MON3    135-22   135-24#
MON4    135-25   135-27#
MON5    135-28   135-33#
MON6    135-36   135-39#
MONTHS   49-7#   135-16
MSCP     36-31#
MSGPKL   55-149#  55-152
MSGPKT   55-105   55-145#
NCONF    87-38    87-43#
NCONS    87-37#   87-40
NEWTAB  128-22   129-3#
NOCLOC   52-11#  127-53
NULL     48-50#
NXMAD    48-34#   61-10*   61-18   113-12*  118-26*  118-31
NXMI     61-11   113-10#  118-27
NXTTAB  128-16   129-31   130-10   130-15#
O$APTS   28-8#    28-34
O$AU     28-8#    28-34
O$BGNR   28-8#    28-34
O$BGNS   28-8#    28-32#   28-34
O$DU     28-8#    28-34
O$ERRT   28-8#    28-34
O$GNSW   28-8#    28-32#   28-34
O$POIN   28-8#    28-32    28-32#   28-32#   28-32#   28-34
O$SETU   28-8#    28-32#   28-34   143-8
OP.ABO   38-3#
OP.ACC   38-4#
OP.AVA   38-22#
OP.AVL   38-5#
OP.CCD   38-6#
OP.CMP   38-7#
OP.DUP   38-23#
OP.ELP   38-30#
OP.END   38-20#   73-5     73-8     74-58
OP.ERS   38-8#
OP.ESP   38-29#  106-2
OP.FLU   38-9#
OP.GCS   38-10#
OP.GDS   38-27#   71-49    74-58
OP.GSS   38-28#   63-17
OP.GUS   38-11#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
OP.MRD   38-18#
OP.MWR   38-19#  108-21
OP.ONL   38-12#
OP.RD    38-13#
OP.RLC   38-25#
OP.RPL   38-14#
OP.RSD   38-32#   73-8     74-43
OP.SCC   38-15#
OP.SEX   38-21#
OP.SHC   38-24#
OP.SSD   38-31#   73-5     74-13
OP.SUC   38-16#
OP.WR    38-17#
OSTRE    87-35    87-42    87-47#
OSTRNG   87-34#   87-46    95-6     96-6     97-6     103-17
P.BCNT   40-21#   41-9#    74-11    74-33*   106-5*   110-19*
P.BUFF   40-22#
P.CMST   41-14#
P.CNCL   41-48#
P.CNTF   40-40#   41-46#
P.CNTI   41-49#
P.CPSP   40-34#
P.CRF    40-17#   41-4#    73-19    109-17*
P.CTMO   41-47#
P.CYL    41-26#
P.DEXT   41-52#  138-82
P.DFLG   41-53#   74-60
P.DMDT   40-50#
P.DPI    41-54#   74-65    74-67    74-72    74-73
P.DTO    41-55#
P.ELGF   40-32#
P.FBBK   41-10#
P.FLGS   41-7#
P.GRP    41-25#
P.HSTI   40-31#   41-19#   41-35#
P.HTMO   40-41#
P.LBN    40-24#
P.MEDI   41-21#   41-37#
P.MLUN   41-17#   41-33#
P.MOD    40-20#
P.OPCD   40-19#   41-6#    73-9     74-58    108-29*
P.OTRF   40-27#   41-13#
P.OVRL   40-51#  106-6*   106-7*
P.RBN    40-36#
P.RBNS   41-28#
P.RCTC   41-29#
P.RCTS   41-27#
P.RGID   40-46#
P.RGOF   40-47#
P.SHST   41-23#   41-39#
P.SHUN   40-33#   41-22#   41-38#
P.STS    41-8#    73-14    106-11
P.TIME   40-43#
P.TRCK   41-24#
P.UADR   40-23#  106-4*   110-18*
P.UNCL   41-40#
```

CROSS REFERENCE TABLE (CREF V04.00)

CROSS REFERENCE TABLE (CREF V04.00)

```
RNTIM1  52-6#   125-28
RNTIM2  52-7#   125-33
RNTIME  86-22    93-6     94-6     125-13#
RNTIMX 125-14   125-35#
RSPDRP  71-26    72-12#   72-37    73-22    74-9     74-27
RSPDSP  74-26    75-3#    75-9
RSPERR  73-10    74-58#
RSPERW  73-15    73-21#   74-59
RSPIN   71-15    73-5#
RSPMWR  73-7     73-9#
RSPNTO  71-31    71-33    71-36    71-38    71-54#
RSPNXT  71-12    72-3#    72-33    74-57    74-75
RSPOU   73-27#
RSPOU2  74-41    74-50#
RSPOU3  74-48    74-51#
RSPOU4  71-53    74-53#
RSPOUT  71-18    73-27    74-34    74-39#
RSPPT2  74-3#
RSPPT3  74-7     74-11#
RSPPTW  73-20    73-26#
RSPTM   71-24    71-30#
RSPTMO  71-35    71-39#
RSPTOE  71-40    71-42    72-36#   74-61    74-76
RUNDM   64-15#  138-117
RWORDT  68-5     69-1#
RWRDE1  66-25    67-10    68-34    69-4     70-6#
S$LSYM  28-8#    30-17#   31-14#   55-16#   55-20#   55-24#   55-28#   55-32#   55-36#   55-40#   55-44#   55-48#   55-52#   55-56#
        55-65#   55-71#   55-85#   55-89#   55-93#   55-97#   55-101#  55-106#  55-110#  55-114#  55-118#  55-122#  55-126#  55-130#
       113-14#  114-21#  124-9#   135-1    135-1    135-1    135-1#   135-104# 136-12#  137-15#  138-20   138-20   138-20   138-20#
       138-34   138-34   138-34   138-34#  138-138# 140-6#   142-10#
S.CNS  142-5    142-19#
S.FMT  142-1    142-18#
S.NOF  142-9    142-23#
S.NRF  142-3    142-12#
S.RST  142-7    142-20#
SA.A2   34-18#
SA.BST  35-23#
SA.CM1  34-32#
SA.CMD  34-27#
SA.CME  34-37#
SA.CNT  35-16#
SA.CTP  34-40#
SA.DI   34-19#  120-14
SA.ERC  34-13#
SA.ERR  34-9#    55-68    121-19
SA.GO   35-21#  117-20
SA.INE  35-4#
SA.INT  34-25#
SA.LFC  35-22#
SA.MCV  35-15#  120-38
SA.MS1  34-31#
SA.MSE  34-36#
SA.MSG  34-26#
SA.NV   34-17#
SA.NVE  35-5#
SA.PRG  34-44#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
SA.S1    34-5#   119-3   120-14
SA.S2    34-6#   120-23
SA.S3    34-7#   120-31
SA.S4    34-8#   119-13  120-38
SA.STE   34-39#
SA.STP   34-29#  118-18
SA.TST   35-11#  120-8
SA.VCE   35-3#
SA.VEC   34-24#
SA.WRP   34-28#
SAMTAB  128-26   130-3#
SAMVEC  128-31   134-19#
SERNQ    51-5#   138-34
SERNUM   52-18#  138-31
SERNX    54-11#  138-52
SETO0   115-17#  115-22
SETO1   115-18   115-20#
SETO2   115-26#  115-29
SETTO    74-56   111-15  115-14#  121-23
SFPTBL   31-10#  127-25  127-28   127-32
SNDCMD   63-19    74-52  106-8    109-14#
SO.BIT  141-18#  142-1   142-1    142-1    142-3   142-3   142-3   142-5   142-5   142-5   142-7   142-7   142-7
SO.CNS  127-27   127-31  141-22#  142-5
SO.FM1  141-19#  141-21  142-1
SO.FM2  141-20#  141-21  142-3
SO.FMT   79-53   127-24  138-6    141-21#
SO.STR   79-59   127-31  138-12   141-23#  142-7
SPECL    75-8     83-14#  83-31
SPECLC   83-18#
SPECLE   83-27    83-33#
SPECLL   83-26#   83-29
SPECLR   83-16    83-24#
SPECLX   83-15    83-34#
ST.ABO   42-7#
ST.AVL   42-9#
ST.CMD   42-6#
ST.CMP   42-12#
ST.CNT   42-15#
ST.DAT   42-13#
ST.DIA   42-17#
ST.DRV   42-16#
ST.HST   42-14#
ST.MFE   42-10#
ST.MSK   42-3#   73-14   106-11
ST.OFL   42-8#
ST.SUB   42-4#
ST.SUC   42-5#
ST.WPR   42-11#
SVCGBL   28-8#   28-13#  28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34
         28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34
         28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   28-34   29-9    30-10
         30-10   31-10   31-10   50-12   50-17   55-11#  55-14   55-14   55-14   55-18   55-18   55-18   55-22   55-22
         55-22   55-26   55-26   55-26   55-30   55-30   55-30   55-34   55-34   55-34   55-38   55-38   55-38   55-42
         55-42   55-42   55-46   55-46   55-46   55-50   55-50   55-50   55-54   55-54   55-54   55-58   55-58   55-58
         55-67   55-67   55-67   55-73   55-73   55-73   55-87   55-87   55-87   55-91   55-91   55-91   55-95   55-95
         55-95   55-99   55-99   55-99   55-103  55-103  55-103  55-108  55-108  55-108  55-112  55-112  55-112  55-116
```

```
                                                                    G 14
CZUDECO UDA-50 DISK FORMATTER MACRO V04.00 30-AUG-82 12:58:39 PAGE S-16                                                     SEQ 0175
CROSS REFERENCE TABLE (CREF V04.00 )

         55-116   55-116   55-120   55-120   55-120   55-124   55-124   55-124   55-128   55-128   55-128   56-4#   113-10   114-18
       124-5    126-10   127-8    136-10   137-8    139-14   141-12   143-8    143-8    143-8    143-8#
SVCINS  28-8#    28-10#   28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    29-9     29-9     29-9     29-9     29-9     29-9     30-10    30-10    30-10    31-10    31-10
        31-10    50-12    50-12    50-12    50-12    50-12    50-12    50-17    50-17    50-17    50-17    50-17    50-17    55-8#
        55-16    55-20    55-24    55-28    55-32    55-36    55-40    55-44    55-48    55-52    55-56    55-65    55-70    55-70
        55-70    55-70    55-70    55-71    55-85    55-89    55-93    55-97    55-101   55-106   55-110   55-114   55-118   55-122
        55-126   55-130   56-1#    57-7     57-7     57-7     57-7     57-7     57-7     57-7     57-7     57-7     57-7     57-7
        57-7     57-8     57-8     57-8     61-11    61-11    61-11    61-11    61-11    61-11    61-11    61-11    61-11    61-11
        61-11    61-11    61-11    61-11    61-11    61-11    61-11    61-11    61-12    61-12    61-12    61-22    61-22    61-22
        61-22    61-22    61-22    61-28    61-28    61-28    61-28    61-28    61-28    61-28    61-28    61-28    61-28    61-28
        61-28    61-30    61-30    61-30    62-20    62-20    62-20    62-20    62-20    62-20    62-20    62-20    62-20    62-20
        62-20    62-20    62-21    62-21    62-21    63-9     63-9     63-9     63-9     63-9     63-9     63-9     63-9     63-9
        63-9     63-9     63-9     63-9     63-9     63-9     63-9     63-9     63-9     65-13    65-13    65-13    66-16    66-16
        66-16    66-16    66-16    66-16    66-19    66-19    66-19    66-21    66-21    66-21    66-21    66-21    66-21    66-26
        66-26    66-26    67-7     67-7     67-7     68-23    68-23    68-23    68-30    68-30    68-30    69-1     69-1     69-1
        69-11    69-11    69-11    69-14    69-14    69-14    70-6     70-6     70-6     70-6     70-6     70-6     70-6     70-6
        70-6     70-6     70-6     70-6     70-7     70-7     70-7     71-9     71-9     71-9     71-25    71-25    71-25    71-25
        71-25    71-25    71-25    71-25    71-25    71-25    71-25    71-25    72-36    72-36    72-36    72-36    72-36    72-36
        72-36    72-36    72-36    72-36    72-36    72-36    73-21    73-21    73-21    73-21    73-21    73-21    73-21    73-21
        73-21    73-21    73-21    73-21    74-8     74-8     74-8     74-8     74-8     74-8     74-8     74-8     74-8     74-8
        74-8     74-8     78-22    78-22    78-22    78-22    78-22    78-22    78-22    78-22    78-22    78-22    78-22    78-22
        80-30    80-30    80-30    80-30    80-30    80-30    80-30    80-30    80-30    80-30    80-30    80-30    83-20    83-20
        83-20    83-21    83-21    83-21    83-21    83-21    83-21    83-26    83-26    83-26    83-26    83-26    83-26    83-27
        83-27    83-27   102-12   102-12   102-12   102-12   102-12   102-12   102-12   102-12   102-12   102-12   102-12   102-12
       102-12   102-12   102-12   102-12   102-12   102-12   102-14   102-14   102-14   102-14   102-14   102-14   102-14   102-14
       102-14   102-14   102-14   102-14   102-14   102-14   102-14   102-14   102-14   102-14   102-16   102-16   102-16   102-16
       102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16   102-16
       102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18   102-18
       102-18   102-18   102-18   102-18   105-20   105-20   105-20   105-20   105-20   105-20   105-20   105-20   105-20   105-20
       105-20   105-20   105-20   105-20   105-20   105-20   105-20   105-20   107-3    107-3    107-3    107-3    107-3    107-3
       107-3    107-3    107-3    107-3    107-3    107-3    111-21   111-21   111-21   111-29   111-29   111-29   111-29   111-29
       111-29   111-29   111-29   111-29   111-29   111-29   111-29   112-5    112-5    112-5    112-5    112-5    112-5    112-5
       112-5    112-5    112-5    112-5    112-5    113-14   113-14   113-14   114-21   114-21   114-21   116-35   116-35   116-35
       116-35   116-35   116-35   116-35   116-35   116-35   116-35   116-35   116-35   117-11   117-11   117-11   117-11   117-11
       117-11   117-11   117-11   117-11   117-11   117-11   117-11   118-12   118-12   118-12   118-27   118-27   118-27   118-27
       118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27   118-27
       118-30   118-30   118-30   118-30   118-30   118-30   118-33   118-33   118-33   118-33   118-33   118-33   118-33   118-33
       118-33   118-33   118-33   118-33   120-44   120-44   120-44   120-44   120-44   120-44   120-44   120-44   120-44   120-44
       120-44   120-44   121-27   121-27   121-27   121-36   121-36   121-36   121-36   121-36   121-36   121-36   121-36   121-36
       121-36   121-36   121-36   122-5    122-5    122-5    122-5    122-5    122-5    122-5    122-5    122-5    122-5    122-5
       122-5    124-9    124-9    124-9    127-10   127-10   127-10   127-10   127-10   127-10   127-11   127-11   127-11   127-12
       127-12   127-12   127-12   127-12   127-12   127-13   127-13   127-13   127-14   127-14   127-14   127-14   127-14   127-14
       127-15   127-15   127-15   127-16   127-16   127-16   127-16   127-16   127-16   127-17   127-17   127-17   127-18   127-18
       127-18   127-39   127-39   127-39   127-39   127-39   127-39   127-48   127-48   127-48   127-48   127-48   127-48   127-49
       127-49   127-49   127-50   127-50   127-50   127-50   127-50   127-50   127-51   127-51   127-51   127-59   127-59   127-59
       127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59   127-59
```

CROSS REFERENCE TABLE (CREF V04.00)

```
        127-59  128-15  128-15  128-15  128-15  128-15  128-15  128-16  128-16  128-16  131-4   131-4   131-4   131-4
        131-4   131-4   131-5   131-5   131-5   133-12  133-12  133-12  133-12  133-12  133-12  133-14  133-14  133-14
        133-16  133-16  133-16  133-16  133-16  133-16  134-3   134-3   134-3   134-3   134-3   134-3   134-3   134-3
        134-3   134-3   134-3   134-3   134-4   134-4   134-4   134-8   134-8   134-8   134-8   134-8   134-8   134-8
        134-8   134-8   134-8   134-8   134-8   134-9   134-9   134-9   134-14  134-14  134-14  134-14  134-14  134-14
        134-14  134-14  134-14  134-14  134-14  134-14  134-15  134-15  134-15  134-20  134-20  134-20  134-20  134-20
        134-20  134-20  134-20  134-20  134-20  134-20  134-20  134-21  134-21  134-21  135-1   135-1   135-1   135-1
        135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1   135-1
        135-1   135-1   135-1   135-1   135-1   135-1   135-104 135-104 135-104 136-12  136-12  136-12  137-11  137-11
        137-11  137-15  137-15  137-15  138-8   138-8   138-8   138-9   138-9   138-9   138-10  138-10  138-10  138-10
        138-10  138-10  138-10  138-10  138-10  138-10  138-10  138-10  138-11  138-11  138-11  138-11  138-11  138-11
        138-16  138-16  138-16  138-16  138-16  138-16  138-16  138-16  138-16  138-16  138-16  138-16  138-17  138-17
        138-17  138-17  138-17  138-17  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20
        138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20  138-20
        138-22  138-22  138-22  138-22  138-22  138-22  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34
        138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34  138-34
        138-34  138-34  138-52  138-52  138-52  138-52  138-52  138-52  138-52  138-52  138-52  138-52  138-52  138-52
        138-52  138-52  138-52  138-52  138-52  138-52  138-137 138-137 138-137 138-137 138-137 138-137 138-138 138-138
        138-138 139-14  139-14  139-14  140-1   140-1   140-1   140-1   140-1   140-1   140-1   140-1   140-1   140-1
        140-1   140-1   140-2   140-2   140-2   140-2   140-2   140-2   140-2   140-2   140-2   140-2   140-2   140-2
        140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3   140-3
        140-3   140-4   140-4   140-4   140-4   140-4   140-4   140-4   140-4   140-4   140-4   140-4   140-4   140-4
        140-4   140-4   140-5   140-5   140-5   140-5   140-5   140-5   140-5   140-5   140-5   140-5   140-5   140-5
        140-5   140-5   140-5   140-6   140-6   140-6   141-12  141-12  141-12  142-1   142-1   142-1   142-1   142-1
        142-1   142-1   142-1   142-1   142-2   142-2   142-2   142-3   142-3   142-3   142-3   142-3   142-3   142-3
        142-3   142-3   142-4   142-4   142-4   142-5   142-5   142-5   142-5   142-5   142-5   142-5   142-5   142-5
        142-6   142-6   142-6   142-7   142-7   142-7   142-7   142-7   142-7   142-7   142-7   142-7   142-8   142-8
        142-8   142-9   142-9   142-9   142-9   142-9   142-9   142-10  142-10  142-10  143-8   143-8   143-8   143-8
        143-8   143-8   143-8   143-8   143-8   144-3   144-3   144-3   144-3   144-3   144-3
SVCSUB   28-8#   28-12#  55-10#  56-3#
SVCTAG   28-8#   28-14#  30-17   31-14   55-12#  55-16   55-16   55-16   55-20   55-20   55-20   55-24   55-24   55-24
         55-28   55-28   55-28   55-32   55-32   55-32   55-36   55-36   55-36   55-40   55-40   55-40   55-44   55-44
         55-44   55-48   55-48   55-48   55-52   55-52   55-52   55-56   55-56   55-56   55-65   55-65   55-65   55-71
         55-71   55-71   55-85   55-85   55-85   55-89   55-89   55-89   55-93   55-93   55-93   55-97   55-97   55-97
         55-101  55-101  55-101  55-106  55-106  55-106  55-110  55-110  55-110  55-114  55-114  55-114  55-118  55-118
         55-118  55-122  55-122  55-122  55-126  55-126  55-126  55-130  55-130  55-130  56-5#   113-14  114-21  124-9
        135-1   135-104 136-12  137-15  138-20  138-34  138-138 140-6   142-10  144-3   144-11
SVCTST   28-8#   28-11#  55-9#   56-2#   138-5
SWEND   142-2   142-4   142-6   142-8   142-10#
T$$AUT  136-10#  136-12
T$$CLE  137-8#   137-15
T$$DAT  144-3    144-3#  144-11
T$$HAR  139-14   139-14#  140-6
T$$HW    30-10    30-10#   30-17
T$$INI  127-8#   133-16  135-104
T$$MSG   55-14#   55-16   55-18#  55-20   55-22#  55-24   55-26#  55-28   55-30#  55-32   55-34#  55-36   55-38#  55-40
         55-42#  55-44   55-46#  55-48   55-50#  55-52   55-54#  55-56   55-58#  55-65   55-67#  55-71   55-73#  55-85
         55-87#  55-89   55-91#  55-93   55-95#  55-97   55-99#  55-101  55-103# 55-106  55-108# 55-110  55-112# 55-114
        55-116#  55-118  55-120#  55-122  55-124#  55-126  55-128#  55-130
T$$PC   144-1#   144-13
T$$PRO  126-10#
T$$PTA  144-1#   144-3    144-3#
T$$SOF  141-12   141-12#  142-10
T$$SRV  113-10#  113-14   114-18#  114-21   124-5#   124-9
T$$SW    31-10    31-10#   31-14
T$$TES  138-5#   138-11   138-17   138-137  138-138
```

CROSS REFERENCE TABLE (CREF V04.00)

```
T$ARGC  28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   55-70    55-70    55-70#   102-12   102-12   102-12   102-12#  102-12#  102-14   102-14   102-14
        102-14#  102-14#  102-16   102-16   102-16   102-16#  102-16#  102-18   102-18   102-18   102-18#  102-18#  138-52   138-52
        138-52   138-52#  138-52#
T$CODE  135-1    135-1    135-1    135-1#   135-1#   135-1#   138-20   138-20   138-20   138-20#  138-20#  138-20#  138-34   138-34
        138-34   138-34#  138-34#  138-34#  140-1    140-1    140-1    140-1#   140-1#   140-1#   140-2    140-2    140-2    140-2#
        140-2#   140-2#   140-3    140-3    140-3    140-3#   140-3#   140-3#   140-4    140-4    140-4    140-4#   140-4#   140-4#
        140-5    140-5    140-5    140-5#   140-5#   140-5#   142-1    142-1    142-1    142-1#   142-1#   142-1#   142-2    142-2
        142-2    142-2    142-2    142-2    142-2#   142-2#   142-2#   142-2#   142-3    142-3    142-3    142-3#   142-3#   142-3#
        142-4    142-4    142-4    142-4    142-4    142-4    142-4#   142-4#   142-4#   142-4#   142-5    142-5    142-5    142-5#
        142-5#   142-5#   142-6    142-6    142-6    142-6    142-6    142-6    142-6#   142-6#   142-6#   142-6#   142-7    142-7
        142-7    142-7#   142-7#   142-7#   142-8    142-8    142-8    142-8    142-8    142-8    142-8#   142-8#   142-8#   142-8#
        142-9    142-9#
T$ERRN   28-8#    57-7     57-7#    61-28    61-28#   62-20    62-20#   70-6     70-6#    71-25    71-25#   72-36    72-36#   73-21
         73-21#   74-8     74-8#    78-22    78-22#   80-30    80-30#   107-3    107-3#   111-29   111-29#  112-5    112-5#   116-35
        116-35#  117-11   117-11#  118-33   118-33#  120-44   120-44#  121-36   121-36#  122-5    122-5#   134-3    134-3#   134-8
        134-8#   134-14   134-14#  134-20   134-20#  138-10   138-10#  138-16   138-16#
T$EXCP  135-1    135-1#   138-20   138-20#  138-34   138-34#  140-1    140-1#   140-2    140-2#   140-3    140-3#   140-4    140-4#
        140-5    140-5#
T$FLAG  133-16   133-16   133-16#  133-16#  138-11   138-11   138-11#  138-11#  138-17   138-17   138-17#  138-17#  138-137  138-137
        138-137# 138-137#
T$FREE  143-8    144-13#
T$GMAN   28-8#   135-1    135-1#   135-1#   138-20   138-20#  138-20#  138-34   138-34#  138-34#
T$HILI  135-1    135-1#   138-20   138-20#  138-34   138-34#  140-1    140-1#   140-2    140-2#   140-3    140-3#   140-4    140-4#
        140-5    140-5#
T$LAST   28-8#   143-8#   144-1
T$LOLI  135-1    135-1#   138-20   138-20#  138-34   138-34#  140-1    140-1#   140-2    140-2#   140-3    140-3#   140-4    140-4#
        140-5    140-5#
T$LSYM   28-8     28-8#    30-17    31-14    55-16    55-20    55-24    55-28    55-32    55-36    55-40    55-44    55-48    55-52
         55-56    55-65    55-71    55-85    55-89    55-93    55-97    55-101   55-106   55-110   55-114   55-118   55-122   55-126
        55-130   113-14   114-21   124-9    135-104  136-12   137-15   138-138  140-6    142-10
T$LTNO  143-8#
T$NEST   28-8#    28-26    28-26    28-26#   30-10    30-10    30-10#   30-17    30-17    30-17    30-17#   31-10    31-10    31-10#
         31-14    31-14    31-14    31-14#   31-16    31-16    31-16    31-16#   32-3     32-3     32-3#    55-14    55-14    55-14#
         55-16    55-16    55-16    55-16#   55-18    55-18    55-18#   55-20    55-20    55-20    55-20#   55-22    55-22    55-22#
         55-24    55-24    55-24    55-24#   55-26    55-26    55-26#   55-28    55-28    55-28    55-28#   55-30    55-30    55-30#
         55-32    55-32    55-32    55-32#   55-34    55-34    55-34#   55-36    55-36    55-36    55-36#   55-38    55-38    55-38#
         55-40    55-40    55-40    55-40#   55-42    55-42    55-42#   55-44    55-44    55-44    55-44#   55-46    55-46    55-46#
         55-48    55-48    55-48    55-48#   55-50    55-50    55-50#   55-52    55-52    55-52    55-52#   55-54    55-54    55-54#
         55-56    55-56    55-56    55-56#   55-58    55-58    55-58#   55-65    55-65    55-65    55-65#   55-67    55-67    55-67#
         55-71    55-71    55-71    55-71#   55-73    55-73    55-73#   55-85    55-85    55-85    55-85#   55-87    55-87    55-87#
         55-89    55-89    55-89    55-89#   55-91    55-91    55-91#   55-93    55-93    55-93    55-93#   55-95    55-95    55-95#
         55-97    55-97    55-97    55-97#   55-99    55-99    55-99#   55-101   55-101   55-101   55-101#  55-103   55-103   55-103#
         55-106   55-106   55-106   55-106#  55-108   55-108   55-108#  55-110   55-110   55-110   55-110#  55-112   55-112   55-112#
         55-114   55-114   55-114   55-114#  55-116   55-116   55-116#  55-118   55-118   55-118   55-118#  55-120   55-120   55-120#
         55-122   55-122   55-122   55-122#  55-124   55-124   55-124#  55-126   55-126   55-126   55-126#  55-128   55-128   55-128#
         55-130   55-130   55-130   55-130#  113-10   113-10   113-10#  113-14   113-14   113-14   113-14#  114-18   114-18   114-18#
        114-21   114-21   114-21   114-21#  124-5    124-5    124-5#   124-9    124-9    124-9    124-9#   125-38   125-38   125-38
        125-38#  126-3    126-3    126-3#   126-10   126-10   126-10#  126-16   126-16   126-16   126-16#  127-8    127-8    127-8#
        135-104  135-104  135-104  135-104# 136-10   136-10   136-10#  136-12   136-12   136-12   136-12#  137-8    137-8    137-8#
        137-15   137-15   137-15   137-15#  137-17   137-17   137-17   137-17#  138-3    138-3    138-3#   138-5    138-5    138-5#
        138-138  138-138  138-138  138-138# 138-139  138-139  138-139  138-139# 139-3    139-3    139-3#   139-14   139-14   139-14#
        140-6    140-6    140-6    140-6#   141-12   141-12   141-12#  142-2    142-4    142-6    142-8    142-10   142-10   142-10
        142-10#  143-10   143-10   143-10   143-10#
T$NS0    28-26#   31-16    32-3#    125-38   126-3#   137-17   138-3#   138-139  139-3#   143-10
T$NS1    30-10#   30-17    31-10#   31-14    55-14#   55-16    55-18#   55-20    55-22#   55-24    55-26#   55-28    55-30#   55-32
```

CROSS REFERENCE TABLE (CREF V04.00)

```
        55-34#  55-36   55-38#  55-40   55-42#  55-44   55-46#  55-48   55-50#  55-52   55-54#  55-56   55-58#  55-65
        55-67#  55-71   55-73#  55-85   55-87#  55-89   55-91#  55-93   55-95#  55-97   55-99#  55-101  55-103# 55-106
        55-108# 55-110  55-112# 55-114  55-116# 55-118  55-120# 55-122  55-124# 55-126  55-128# 55-130  113-10# 113-14
       114-18# 114-21  124-5#  124-9   126-10# 126-16  127-8#  135-104 136-10# 136-12  137-8#  137-15  138-5#  138-138
       139-14# 140-6   141-12# 142-2   142-4   142-6   142-8   142-10
T$PCNT 144-1#  144-3   144-3   144-3#
T$PTAB 144-3   144-3#
T$PTHV  28-34  144-13#
T$PTNU  28-8#  144-3   144-3#  144-13  144-13
T$SAVL  28-8#
T$SEGL  28-8#
T$SIZE 143-8   144-13#
T$SUBN  28-8#  138-5#
T$TAGL  28-8#
T$TAGN  28-8#   30-10   30-10   30-10#  31-10   31-10   31-10#  55-14   55-14   55-14#  55-18   55-18   55-18#  55-22
        55-22   55-22#  55-26   55-26   55-26#  55-30   55-30   55-30#  55-34   55-34   55-34#  55-38   55-38   55-38#
        55-42   55-42   55-42#  55-46   55-46   55-46#  55-50   55-50   55-50#  55-54   55-54   55-54#  55-58   55-58
        55-58#  55-67   55-67   55-67#  55-73   55-73   55-73#  55-87   55-87   55-87#  55-91   55-91   55-91#  55-95
        55-95   55-95#  55-99   55-99   55-99#  55-103  55-103  55-103# 55-108  55-108  55-108# 55-112  55-112  55-112#
        55-116  55-116  55-116# 55-120  55-120  55-120# 55-124  55-124  55-124# 55-128  55-128  55-128# 113-10  113-10
       113-10# 114-18  114-18  114-18# 124-5   124-5   124-5#  126-10  126-10  126-10# 127-8   127-8   127-8#  136-10
       136-10  136-10# 137-8   137-8   137-8#  138-5   138-5   138-5#  139-14  139-14  139-14# 141-12  141-12  141-12#
       144-1   144-1   144-1#  144-3   144-3   144-3   144-3   144-3#  144-3#
T$TEMP  29-9    29-9    29-9#   29-9#   30-17   30-17#  31-14   31-14#  31-16   31-16#  55-16   55-16#  55-20   55-20#
        55-24   55-24#  55-28   55-28#  55-32   55-32#  55-36   55-36#  55-40   55-40#  55-44   55-44#  55-48   55-48#
        55-52   55-52#  55-56   55-56#  55-65   55-65#  55-71   55-71#  55-85   55-85#  55-89   55-89#  55-93   55-93#
        55-97   55-97#  55-101  55-101# 55-106  55-106# 55-110  55-110# 55-114  55-114# 55-118  55-118# 55-122  55-122#
        55-126  55-126# 55-130  55-130# 113-14  113-14# 114-21  114-21# 124-9   124-9#  125-38  125-38# 126-16  126-16#
       133-16  133-16# 135-1   135-1   135-1   135-1#  135-1#  135-1#  135-104 135-104# 136-12 136-12# 137-15  137-15#
       137-17  137-17# 138-11  138-11# 138-17  138-17#  138-20  138-20  138-20  138-20#  138-20# 138-20# 138-34  138-34
       138-34  138-34# 138-34# 138-34# 138-137 138-137# 138-138 138-138# 138-139 138-139# 140-1   140-1   140-1   140-1#
       140-1#  140-1#  140-2   140-2   140-2   140-2#   140-2#  140-2#  140-3   140-3   140-3   140-3#  140-3#  140-3#
       140-4   140-4   140-4   140-4#  140-4#  140-4#   140-5   140-5   140-5   140-5#  140-5#  140-5#  140-6   140-6#
       142-1   142-1   142-1   142-1#  142-1#  142-1#   142-3   142-3   142-3   142-3#  142-3#  142-3#  142-5   142-5
       142-5   142-5#  142-5#  142-5#  142-7   142-7    142-7   142-7#  142-7#  142-7#  142-10  142-10# 143-10  143-10#
T$TEST  28-8#  138-5   138-5   138-5#  143-8
T$TSTM  28-8#   55-16   55-20   55-24   55-28   55-32   55-36   55-40   55-44   55-48   55-52   55-56   55-65   55-70
        55-71   55-85   55-89   55-93   55-97   55-101  55-106  55-110  55-114  55-118  55-122  55-126  55-130  57-7
        57-8    61-11   61-12   61-22   61-28   61-30   62-20   62-21   63-9    65-13   66-16   66-19   66-21   66-26
        67-7    68-23   68-30   69-1    69-11   69-14   70-6    70-7    71-9    71-25   72-36   73-21   74-8    78-22
        80-30   83-20   83-21   83-26  102-12  102-14  102-16  102-18  105-20  107-3   111-21  111-29  112-5   116-35
       117-11  118-12  118-27  118-30  118-33  120-44  121-27  121-36  122-5   127-10  127-12  127-14  127-16  127-18
       127-39  127-48  127-50  127-59  128-15  131-4   133-12  133-14  133-16  134-3   134-4   134-8   134-9   134-14
       134-15  134-20  134-21  135-1   135-104 136-12  137-11  137-15  138-8   138-10  138-11  138-16  138-17  138-20
       138-22  138-34  138-52  138-137 138-138
T$TSTS  28-8#  138-5#
T1      29-9   138-5#
T1CNS  138-13  138-25#
T1FMT  138-7   138-24   138-65#
T1GO   138-9   138-12#
T1RST  138-15  138-19#
T1SER1 138-26# 138-64
T1SER2 138-29# 138-61
T1SER3 138-34# 138-53
T1SER4 138-37# 138-43
T1SER5 138-38  138-45#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
T1SER6  138-49#  138-50
T1SER7  138-41   138-44   138-52#
T1SER8  138-46   138-51   138-54#
T1SER9  138-56#  138-57
T1SERN  138-30   138-59   138-62#
TEMP     49-1#   132-9*   134-7    134-13   138-34   138-35   138-47   138-55
TERM     75-6     81-14#
TINDEX   43-10#   43-12    43-12    43-12#   43-13    43-13    43-13#   43-16    43-16    43-16#   43-19    43-19    43-19#   43-20
         43-20    43-20#   43-21    43-21    43-21#   43-22    43-22    43-22#   43-36    43-36    43-36#   43-37    43-37    43-37#
         43-38    43-38    43-38#   43-39    43-39    43-39#   43-40    43-40    43-40#   43-41    43-41    43-41#   43-42    43-42
         43-42#   43-43    43-43    43-43#   43-44    43-44    43-44#   43-45    43-45    43-45#   43-46    43-46    43-46#   43-47
         43-47    43-47#   43-48    43-48    43-48#   43-49    43-49    43-49#   43-51    44-7#    44-9     44-9     44-9#    44-10
         44-10    44-10#   44-13    44-13    44-13#   44-15    139-18#  139-20   139-20   139-20#  139-21   139-21   139-21#  139-22
        139-22   139-22#  139-23   139-23   139-23#  139-24   139-24   139-24#  141-16#  141-18   141-18   141-18#
TINIT    60-15#  138-72   138-128
TINITR   60-19    60-24#
TOOMER  132-20   134-13#
TSTCNT  138-79#  138-106
TSTTAB   48-17#   64-21    71-7     138-65*  138-77   138-115*
TSTTYP   48-30#   64-29   138-109* 138-120* 138-123
U50UNI   48-27#  138-78*  138-97*  138-125
U52EXT   48-28#  138-82
UAM      32-10#
UCNT     48-25#   64-20*   64-38*   71-8*    72-6*    138-76*  138-105* 138-110*
UDAI1A  116-32#  116-38
UDAI1B  116-34   116-37#
UDAI1C  116-33   116-39#
UDAI1L  116-22#  116-24
UDAI2   117-3#
UDAI2E  117-6    117-11#
UDAI2L  117-5#   117-8
UDAI3   117-9    117-16#
UDAID1  118-19*  120-4#   120-20   120-29
UDAID2  118-20*  118-21*  120-6#
UDAID3  120-8#
UDAIDT  119-4    120-3#
UDAIEX  116-29   116-36   116-42   117-12   117-32#
UDAINT   63-14   105-22   116-20#
UDAIR1  120-3    120-14#
UDAIR2  120-5    120-20#
UDAIR3  120-7    120-29#
UDAIR4  120-9    120-37#
UDAIRC  120-15   120-24   120-32   120-42#
UDAIRX  120-43   120-46#
UDAISE  118-35   119-9    119-11   119-19#
UDAISG  118-32   119-3#
UDAISL  119-8#   119-16
UDAIST  116-28   118-12#
UDAISX  119-14   119-18#
UDARS1  121-25#  121-29   121-32   121-34
UDARS2  121-26   122-3#
UDARSD   55-68*   55-69   119-3*   119-12*  119-13   121-19*  121-25   122-16#
UDARSE  121-37   122-6#
UDARSP  116-41   119-8    121-18#
UDARSX  122-4    122-11#
UDASRV  114-18#  129-23
```

CROSS REFERENCE TABLE (CREF V04.00)

```
UF.576  40-12#
UF.CMR  40-3#
UF.CMW  40-4#
UF.INA  40-6#
UF.RPL  40-5#
UF.SCH  40-7#
UF.SCL  40-8#
UF.WBN  40-9#
UF.WPH  40-10#
UF.WPS  40-11#
UFREEZ  48-33#   64-40*   72-3     72-13*   80-19    80-21*
URNING  48-24#   64-16*   64-36*   64-45    72-32*
URUN    48-23#   64-15*   64-20    71-8     138-68*  138-76
WAITMS  63-20    106-9    111-11#
X$ALWA  28-8#
X$FALS  28-8#
X$OFFS  28-8#    142-2    142-4    142-6    142-8
X$TRUE  28-8#    142-2    142-4    142-6    142-8
X1      53-5#     55-15
X10     53-15#    55-47
X100    53-46#    55-125
X101    53-47#    55-129
X11     53-16#    55-51
X1A     53-1#     55-15
X2      53-6#     55-19
X20     53-23#    55-55
X21     53-27#    55-64
X22     53-29#    55-69
X23A    53-31#    55-74
X23B    53-35#    55-78
X24     53-36#    55-88
X25     53-38#    55-92
X2A     53-2#     55-19
X3      53-7#     55-23
X30     53-40#    55-96
X31     53-41#    55-100
X32     53-42#    55-104
X36     53-43#    55-117
X37     53-45#    55-121
X3A     53-3#     55-23
X4      53-8#     55-27
X5      53-10#    55-31
X7      53-11#    55-35
X8      53-12#    55-39
X8A     53-4#     55-39
X9      53-13#    55-43
XFRU    54-8#     55-70    55-84    96-5
XMSG1   54-1#     55-145
XMSG2   54-2#     55-149
XPKT1   54-3#     55-132
XPKT2   54-6#     55-138
XSA     54-7#     97-5
YEAR19  49-31#   135-92
YEAR20  49-32#   135-95
YER1   135-69#   135-82
YER2   135-70    135-72   135-83#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
YER3    135-86   135-92#
YER4    135-94   135-96#  135-99
YER5    135-91   135-97   135-100# 135-101
```

CROSS REFERENCE TABLE (CREF V04.00)

```
.BR      45-31#
AND      45-3#    55-61    63-6     105-17   118-15
ASSUME   45-39#   64-27    72-20    79-36    85-15    86-16    86-17    127-31   128-24   128-25   138-32   138-33
BCOMPL   127-11   127-13   127-15   127-17   127-49   127-51   138-9
BGNAUT   136-10
BGNCLN   137-8
BGNHRD   139-14
BGNHW    30-10
BGNINI   127-8
BGNMOD   28-26    32-3     126-3    138-3    139-3
BGNMSG   55-14    55-18    55-22    55-26    55-30    55-34    55-38    55-42    55-46    55-50    55-54    55-58    55-67    55-73
         55-87    55-91    55-95    55-99    55-103   55-108   55-112   55-116   55-120   55-124   55-128
BGNPRO   126-10
BGNPTA   144-3
BGNSET   144-1
BGNSFT   141-12
BGNSRV   113-10   114-18   124-5
BGNSW    31-10
BGNTST   138-5
BNCOMP   83-27    128-16   131-5
BREAK    61-12    66-19    71-9     111-21   118-12   121-27
BUILD    90-18#   91-9     91-21
CLOCK    127-48   127-50
CLOSE    65-13    83-20    133-14   137-11
CLRVEC   61-22    118-30
DESCRI   50-17
DEVTYP   50-12
DISPAT   29-9
DISPLA   142-9
DOCLN    57-8     61-30    62-21    70-7     127-18   134-4    134-9    134-15   134-21
END      33-29#   43-51    44-15    139-25   141-25
ENDAUT   136-12
ENDCLN   137-15
ENDHRD   140-6
ENDHW    30-17
ENDINI   135-104
ENDMOD   31-16    125-38   137-17   138-139  143-10
ENDMSG   55-16    55-20    55-24    55-28    55-32    55-36    55-40    55-44    55-48    55-52    55-56    55-65    55-71    55-85
         55-89    55-93    55-97    55-101   55-106   55-110   55-114   55-118   55-122   55-126   55-130
ENDPRO   126-16
ENDPTA   144-11
ENDSET   144-13
ENDSFT   142-10
ENDSRV   113-14   114-21   124-9
ENDSW    31-14
ENDTST   138-138
ENTRY    91-3#    91-9     91-9     91-9     91-9     91-9     91-9     91-9     91-9     91-15#   91-21    91-21    91-21    91-21
         91-21    91-21    91-21    91-21
EQUALS   32-10
ERRDF    61-28    71-25    72-36    73-21    74-8     78-22    80-30    107-3    111-29   112-5    116-35   117-11   118-33   120-44
         121-36   122-5
ERRSF    57-7     62-20    70-6     134-3    134-8    134-14   134-20   138-10   138-16
EXIT     133-16   138-11   138-17   138-137
GETBYT   66-21    66-26    67-7     68-23    68-30    69-1     69-11    69-14    83-26
GMANID   135-1    138-20   138-34
GPHARD   128-15   131-4
```

CROSS REFERENCE TABLE (CREF V04.00)

```
GPRMA   140-1    140-2
GPRMD   135-1    135-1#   138-20   138-20#  138-34   138-34#  140-3    140-4    140-5
GPRML   142-1    142-3    142-5    142-7
HEADER   28-34
ITEM     33-24#   43-12    43-13    43-16    43-19    43-20    43-21    43-22    43-36    43-37    43-38    43-39    43-40    43-41
         43-42    43-43    43-44    43-45    43-46    43-47    43-48    43-49    44-9     44-10    44-13    139-20   139-21   139-22
         139-23   139-24   141-18
LASTAD  143-8
M$BYTE   28-34    28-34    28-34    28-34#
M$CHEC  133-16   133-16#  138-11   138-11#  138-17   138-17#  138-137  138-137#
M$CNTO  135-1    135-1#   138-20   138-20#  138-34   138-34#  140-1    140-1#   140-2    140-2#   140-3    140-3#   140-4    140-4#
        140-5    140-5#   142-1    142-1#   142-3    142-3#   142-5    142-5#   142-7    142-7#
M$COUN   55-70    55-70#   102-12   102-12#  102-14   102-14#  102-16   102-16#  102-18   102-18#  138-52   138-52#
M$DATA   28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
         28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
         28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34#   28-34#   50-12    50-12#   50-17
         50-17#
M$DECR   30-17    30-17#   31-14    31-14#   31-16    31-16#   55-16    55-16#   55-20    55-20#   55-24    55-24#   55-28    55-28#
         55-32    55-32#   55-36    55-36#   55-40    55-40#   55-44    55-44#   55-48    55-48#   55-52    55-52#   55-56    55-56#
         55-65    55-65#   55-71    55-71#   55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#
         55-106   55-106#  55-110   55-110#  55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#
        113-14   113-14#  114-21   114-21#  124-9    124-9#   125-38   125-38#  126-16   126-16#  135-104  135-104#  136-12   136-12#
        137-15   137-15#  137-17   137-17#  138-138  138-138#  138-139  138-139#  140-6    140-6#   142-10   142-10#  143-10   143-10#
        144-3    144-3#
M$DEFA  135-1    135-1#   138-20   138-20#  138-34   138-34#  140-1    140-1#   140-2    140-2#   140-3    140-3#   140-4    140-4#
        140-5    140-5#   142-1    142-1#   142-3    142-3#   142-5    142-5#   142-7    142-7#
M$ENDE   30-17#   31-14#   31-16#   55-16#   55-20#   55-24#   55-28#   55-32#   55-36#   55-40#   55-44#   55-48#   55-52#   55-56#
         55-65#   55-71#   55-85#   55-89#   55-93#   55-97#   55-101#  55-106#  55-110#  55-114#  55-118#  55-122#  55-126#  55-130#
        113-14#  114-21#  124-9#   125-38#  135-104#  136-12#  137-15#  137-17#  138-138#  138-139#  140-6#   142-10#  143-10#
M$ERRI   57-7    57-7#    61-28    61-28#   62-20    62-20#   70-6     70-6#    71-25    71-25#   72-36    72-36#   73-21    73-21#
         74-8     74-8#    78-22    78-22#   80-30    80-30#   107-3    107-3#   111-29   111-29#  112-5    112-5#   116-35   116-35#
        117-11   117-11#  118-33   118-33#  120-44   120-44#  121-36   121-36#  122-5    122-5#   134-3    134-3#   134-8    134-8#
        134-14   134-14#  134-20   134-20#  138-10   138-10#  138-16   138-16#
M$EXCP  135-1    135-1    135-1#   138-20   138-20   138-20#  138-34   138-34   138-34#  140-1    140-1    140-1#   140-2    140-2
        140-2#   140-3    140-3    140-3#   140-4    140-4    140-4#   140-5    140-5    140-5#
M$EXIT  133-16   133-16#  138-11   138-11#  138-17   138-17#  138-137  138-137#
M$EXSE  133-16#  138-11#  138-17#  138-137#
M$EXTJ  133-16#  138-11#  138-17#  138-137#
M$GEN    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
         28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
         28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34#   28-34#   28-34#   28-34#
         28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
         28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
         28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   29-9     29-9#    30-10    30-10    30-10#   30-10#   30-17    30-17#
         31-10    31-10    31-10#   31-10#   31-14    31-14#   50-12    50-12#   50-17    50-17#   55-14    55-14#   55-16    55-16#
         55-18    55-18#   55-20    55-20#   55-22    55-22#   55-24    55-24#   55-26    55-26#   55-28    55-28#   55-30    55-30#
         55-32    55-32#   55-34    55-34#   55-36    55-36#   55-38    55-38#   55-40    55-40#   55-42    55-42#   55-44    55-44#
         55-46    55-46#   55-48    55-48#   55-50    55-50#   55-52    55-52#   55-54    55-54#   55-56    55-56#   55-58    55-58#
         55-65    55-65#   55-67    55-67#   55-71    55-71#   55-73    55-73#   55-85    55-85#   55-87    55-87#   55-89    55-89#
         55-91    55-91#   55-93    55-93#   55-95    55-95#   55-97    55-97#   55-99    55-99#   55-101   55-101#  55-103   55-103#
         55-106   55-106#  55-108   55-108#  55-110   55-110#  55-112   55-112#  55-114   55-114#  55-116   55-116#  55-118   55-118#
         55-120   55-120#  55-122   55-122#  55-124   55-124#  55-126   55-126#  55-128   55-128#  55-130   55-130#  113-10   113-10#
        113-14   113-14#  114-18   114-18#  114-21   114-21#  124-5    124-5#   124-9    124-9#   126-10   126-10#  127-8    127-8#
        135-1    135-1#   135-104  135-104#  136-10   136-10#  136-12   136-12#  137-8    137-8#   137-15   137-15#  138-5    138-5#
        138-20   138-20#  138-34   138-34#  138-138  138-138#  139-14   139-14#  140-6    140-6#   141-12   141-12#  142-10   142-10#
        143-8    143-8#   144-3    144-3#   144-11   144-11#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
M$GENB 135-1    135-1#   138-20   138-20#  138-34   138-34#
M$GETS  30-17    30-17#   31-14    31-14#   31-16    31-16#   55-16    55-16#   55-20    55-20#   55-24    55-24#   55-28    55-28#
        55-32    55-32#   55-36    55-36#   55-40    55-40#   55-44    55-44#   55-48    55-48#   55-52    55-52#   55-56    55-56#
        55-65    55-65#   55-71    55-71#   55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#
        55-106   55-106#  55-110   55-110#  55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#
       113-14   113-14#  114-21   114-21#  124-9    124-9#   125-38   125-38#  126-16   126-16#  135-104  135-104#  136-12   136-12#
       137-15   137-15#  137-17   137-17#  138-138  138-138#  138-139  138-139#  140-6    140-6#   142-2    142-2#   142-4    142-4#
       142-6    142-6#   142-8    142-8#   142-10   142-10#  143-10   143-10#
M$GETT 133-16#  138-11#  138-17#  138-137#  142-2   142-2#   142-4    142-4#   142-6    142-6#   142-8    142-8#
M$GNGB  28-26#   28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   29-9     29-9#    30-10    30-10    30-10#   31-10    31-10
        31-10#   32-3#    50-12    50-12#   50-17    50-17#   55-14    55-14#   55-18    55-18#   55-22    55-22#   55-26    55-26#
        55-30    55-30#   55-34    55-34#   55-38    55-38#   55-42    55-42#   55-46    55-46#   55-50    55-50#   55-54    55-54#
        55-58    55-58#   55-67    55-67#   55-73    55-73#   55-87    55-87#   55-91    55-91#   55-95    55-95#   55-99    55-99#
        55-103   55-103#  55-108   55-108#  55-112   55-112#  55-116   55-116#  55-120   55-120#  55-124   55-124#  55-128   55-128#
       113-10   113-10#  114-18   114-18#  124-5    124-5#   126-3#   126-10   126-10#  127-8    127-8#   136-10   136-10#  137-8
       137-8#   138-3#   139-3#   139-14   139-14#  141-12   141-12#  143-8    143-8#
M$GNIN  28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34    28-34
        28-34    28-34    28-34    28-34    28-34    28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   28-34#
        28-34#   28-34#   28-34#   28-34#   28-34#   28-34#   29-9     29-9     29-9#    29-9#    30-10    30-10#   31-10    31-10#
        50-12    50-12    50-12#   50-12#   50-17    50-17    50-17#   50-17#   55-16    55-16#   55-20    55-20#   55-24    55-24#
        55-28    55-28#   55-32    55-32#   55-36    55-36#   55-40    55-40#   55-44    55-44#   55-48    55-48#   55-52    55-52#
        55-56    55-56#   55-65    55-65#   55-70    55-70    55-70    55-70    55-70    55-70#   55-70#   55-70#   55-70#   55-71
        55-71#   55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#  55-106   55-106#  55-110
        55-110#  55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#  57-7     57-7     57-7
        57-7     57-7#    57-7#    57-7#    57-7#    57-7#    57-8     57-8#    61-11    61-11    61-11    61-11    61-11    61-11
        61-11#   61-11#   61-11#   61-11#   61-11#   61-11#   61-12    61-12#   61-22    61-22    61-22#   61-22#   61-28    61-28
        61-28    61-28    61-28#   61-28#   61-28#   61-28#   61-28#   61-30    61-30#   62-20    62-20    62-20    62-20    62-20#
        62-20#   62-20#   62-20#   62-20#   62-21    62-21#   63-9     63-9     63-9     63-9     63-9     63-9     63-9#    63-9#
        63-9#    63-9#    63-9#    63-9#    65-13    65-13#   66-16    66-16    66-16#   66-16#   66-19    66-19#   66-21    66-21
        66-21#   66-21#   66-26    66-26#   66-26#   67-7     67-7#    67-7#    68-23    68-23#   68-23#   68-30    68-30#   68-30#
        69-1     69-1#    69-1#    69-11    69-11#   69-11#   69-14    69-14#   69-14#   70-6     70-6     70-6     70-6     70-6#
        70-6#    70-6#    70-6#    70-6#    70-7     70-7#    71-9     71-9#    71-25    71-25    71-25    71-25    71-25#   71-25#
        71-25#   71-25#   71-25#   72-36    72-36    72-36    72-36    72-36#   72-36#   72-36#   72-36#   72-36#   73-21    73-21
        73-21    73-21    73-21#   73-21#   73-21#   73-21#   73-21#   74-8     74-8     74-8     74-8     74-8#    74-8#    74-8#
        74-8#    74-8#    78-22    78-22    78-22    78-22    78-22#   78-22#   78-22#   78-22#   78-22#   80-30    80-30    80-30
        80-30    80-30#   80-30#   80-30#   80-30#   80-30#   83-20    83-20#   83-21    83-21    83-21#   83-21#   83-26    83-26
        83-26#   83-26#   83-27    83-27#  102-12   102-12   102-12   102-12   102-12   102-12   102-12#  102-12#  102-12#  102-12#
       102-12#  102-14   102-14   102-14   102-14   102-14   102-14   102-14#  102-14#  102-14#  102-14#  102-14#  102-16   102-16
       102-16   102-16   102-16   102-16   102-16#  102-16#  102-16#  102-16#  102-16#  102-18   102-18   102-18   102-18   102-18
       102-18   102-18#  102-18#  102-18#  102-18#  102-18#  105-20   105-20   105-20   105-20   105-20   105-20   105-20#  105-20#
       105-20#  105-20#  105-20#  105-20#  107-3    107-3    107-3    107-3    107-3#   107-3#   107-3#   107-3#   107-3#   111-21
       111-21#  111-29   111-29   111-29   111-29   111-29#  111-29#  111-29#  111-29#  111-29#  112-5    112-5    112-5    112-5
       112-5#   112-5#   112-5#   112-5#   112-5#   113-14   113-14#  114-21   114-21#  116-35   116-35   116-35   116-35   116-35#
       116-35#  116-35#  116-35#  116-35#  117-11   117-11   117-11   117-11   117-11#  117-11#  117-11#  117-11#  117-11#  118-12
       118-12#  118-27   118-27   118-27   118-27   118-27   118-27   118-27#  118-27#  118-27#  118-27#  118-27#  118-27#  118-30
       118-30   118-30#  118-30#  118-33   118-33   118-33   118-33   118-33#  118-33#  118-33#  118-33#  118-33#  120-44   120-44
       120-44   120-44   120-44#  120-44#  120-44#  120-44#  120-44#  121-27   121-27#  121-36   121-36   121-36   121-36   121-36#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
        121-36#  121-36#  121-36#  121-36#  122-5    122-5    122-5    122-5    122-5#   122-5#   122-5#   122-5#   122-5#   124-9
        124-9#   127-10   127-10   127-10#  127-10#  127-11   127-11#  127-12   127-12   127-12#  127-12#  127-13   127-13#  127-14
        127-14   127-14#  127-14#  127-15   127-15#  127-16   127-16   127-16#  127-16#  127-17   127-17#  127-18   127-18#  127-39
        127-39   127-39#  127-39#  127-48   127-48   127-48#  127-48#  127-48#  127-49   127-49#  127-50   127-50   127-50#  127-50#
        127-50#  127-51   127-51#  127-59   127-59   127-59   127-59   127-59   127-59   127-59#  127-59#  127-59#  127-59#  127-59#
        127-59#  128-15   128-15   128-15#  128-15#  128-15#  128-16   128-16#  131-4    131-4    131-4#   131-4#   131-4#   131-5
        131-5#   133-12   133-12   133-12#  133-12#  133-14   133-14#  133-16   133-16   133-16#  133-16#  134-3    134-3    134-3
        134-3    134-3#   134-3#   134-3#   134-3#   134-3#   134-4    134-4#   134-8    134-8    134-8#   134-8    134-8#   134-8#
        134-8#   134-8#   134-8#   134-9    134-9#   134-14   134-14   134-14   134-14   134-14#  134-14#  134-14#  134-14#  134-14#
        134-15   134-15#  134-20   134-20   134-20   134-20   134-20#  134-20#  134-20#  134-20#  134-20#  134-21   134-21#  135-1
        135-1    135-1    135-1    135-1    135-1    135-1    135-1    135-1#   135-1#   135-1#   135-1#   135-104  135-104#  136-12
        136-12#  137-11   137-11#  137-15   137-15#  138-8    138-8#   138-9    138-9#   138-10   138-10   138-10   138-10   138-10#
        138-10#  138-10#  138-10#  138-10#  138-11   138-11   138-11#  138-11#  138-16   138-16   138-16   138-16   138-16#  138-16#
        138-16#  138-16#  138-16#  138-17   138-17   138-17#  138-17#  138-20   138-20   138-20   138-20   138-20   138-20   138-20
        138-20   138-20#  138-20#  138-20#  138-20#  138-22   138-22   138-22#  138-22#  138-34   138-34   138-34   138-34   138-34
        138-34   138-34   138-34   138-34#  138-34#  138-34#  138-34#  138-52   138-52   138-52   138-52   138-52   138-52   138-52#
        138-52#  138-52#  138-52#  138-52#  138-137  138-137  138-137#  138-137#  138-138  138-138#  139-14   139-14#  140-1    140-1
        140-1    140-1    140-1#   140-2    140-2    140-2    140-2    140-2#   140-3    140-3    140-3    140-3    140-3    140-3#
        140-4    140-4    140-4    140-4    140-4    140-4#   140-5    140-5    140-5    140-5    140-5    140-5#   140-6    140-6#
        141-12   141-12#  142-1    142-1    142-1    142-1#   142-2    142-2#   142-3    142-3    142-3    142-3#   142-4    142-4#
        142-5    142-5    142-5    142-5#   142-6    142-6#   142-7    142-7    142-7    142-7#   142-8    142-8#   142-9    142-9
        142-9#   142-9#   142-10   142-10#  143-8    143-8    143-8    143-8#   144-3    144-3    144-3#   144-3#
M$GNLS  135-1    135-1#   138-20   138-20#  138-34   138-34#
M$GNTA   30-17    30-17#   31-14    31-14#   55-16    55-16#   55-20    55-20#   55-24    55-24#   55-28    55-28#   55-32    55-32#
         55-36    55-36#   55-40    55-40#   55-44    55-44#   55-48    55-48#   55-52    55-52#   55-56    55-56#   55-65    55-65#
         55-71    55-71#   55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#  55-106   55-106#
         55-110   55-110#  55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#  113-14   113-14#
        114-21   114-21#  124-9    124-9#   135-104  135-104#  136-12   136-12#  137-15   137-15#  138-138  138-138#  140-6    140-6#
        142-10   142-10#  144-3    144-3#   144-11   144-11#
M$GNTE  138-5    138-5#
M$HAPT   28-34    28-34#
M$HNAP   28-34    28-34#
M$INCR   28-26    28-26#   30-10    30-10    30-10#   30-10#   31-10    31-10    31-10#   31-10#   32-3     32-3#    55-14    55-14
         55-14#   55-14#   55-16#   55-18    55-18    55-18#   55-18#   55-20#   55-22    55-22    55-22#   55-22#   55-24#   55-26
         55-26    55-26#   55-26#   55-28#   55-30    55-30    55-30#   55-30#   55-32#   55-34    55-34    55-34#   55-34#   55-36#
         55-38    55-38    55-38#   55-38#   55-40#   55-42    55-42    55-42#   55-42#   55-44#   55-46    55-46    55-46#   55-46#
         55-48#   55-50    55-50    55-50#   55-50#   55-52#   55-54    55-54    55-54#   55-54#   55-56#   55-58    55-58    55-58#
         55-58#   55-65#   55-67    55-67    55-67#   55-67#   55-70#   55-71#   55-73    55-73    55-73#   55-73#   55-85#   55-87
         55-87    55-87#   55-87#   55-89#   55-91    55-91    55-91#   55-91#   55-93#   55-95    55-95    55-95#   55-95#   55-97#
         55-99    55-99    55-99#   55-99#   55-101#  55-103   55-103   55-103#  55-103#  55-106#  55-108   55-108   55-108#  55-108#
         55-110#  55-112   55-112   55-112#  55-112#  55-114#  55-116   55-116   55-116#  55-116#  55-118#  55-120   55-120   55-120#
         55-120#  55-122#  55-124   55-124   55-124#  55-124#  55-126#  55-128   55-128   55-128#  55-128#  55-130#  57-7#    57-8#
         61-11#   61-12#   61-22#   61-28#   61-30#   62-20#   62-21#   63-9#    65-13#   66-16#   66-19#   66-21#   66-26#   67-7#
         68-23#   68-30#   69-1#    69-11#   69-14#   70-6#    70-7#    71-9#    71-25#   72-36#   73-21#   74-8#    78-22#   80-30#
         83-20#   83-21#   83-26#   102-12#  102-14#  102-16#  102-18#  105-20#  107-3#   111-21#  111-29#  112-5#   113-10   113-10
        113-10#  113-10#  114-18   114-18   114-18#  114-18#  116-35#  117-11#  118-12#  118-27#  118-30#  118-33#  120-44#  121-27#
        121-36#  122-5#   124-5    124-5    124-5#   124-5#   126-3    126-3#   126-10   126-10   126-10#  126-10#  127-8    127-8
        127-8#   127-8#   127-10#  127-12#  127-14#  127-16#  127-18#  127-39#  127-48#  127-50#  127-59#  128-15#  131-4#   133-12#
        133-14#  133-16#  134-3#   134-4#   134-8#   134-9#   134-14#  134-15#  134-20#  134-21#  135-1    135-1#   135-1#   135-104#
        136-10   136-10   136-10#  136-10#  136-12#  137-8    137-8    137-8#   137-8#   137-11#  137-15#  138-3    138-3#   138-5
        138-5    138-5    138-5#   138-5#   138-5#   138-8#   138-10#  138-11#  138-16#  138-17#  138-20   138-20#  138-20#  138-22#
        138-34   138-34#  138-34#  138-52#  138-137#  138-138#  139-3    139-3#   139-14   139-14   139-14#  139-14#  141-12   141-12
        141-12#  141-12#  144-1    144-1#   144-3    144-3    144-3    144-3#
M$LDRO   61-22    61-22#   66-16    66-16#   83-21    83-21#   118-30   118-30#  127-10   127-10#  127-12   127-12#  127-14   127-14#
        127-16   127-16#  127-48   127-48#  127-50   127-50#  128-15   128-15#  131-4    131-4#   133-12   133-12#  138-22   138-22#
M$MCHI   28-8     28-8#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
M$MCLO  28-8     28-8#
M$POP   30-17    30-17#   31-14    31-14#   31-16    31-16#   55-16    55-16#   55-20    55-20#   55-24    55-24#   55-28    55-28#
        55-32    55-32#   55-36    55-36#   55-40    55-40#   55-44    55-44#   55-48    55-48#   55-52    55-52#   55-56    55-56#
        55-65    55-65#   55-71    55-71#   55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#
        55-106   55-106#  55-110   55-110#  55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#
       113-14   113-14#  114-21   114-21#  124-9    124-9#   125-38   125-38#  126-16   126-16#  135-104  135-104#  136-12   136-12#
       137-15   137-15#  137-17   137-17#  138-138  138-138#  138-139  138-139#  140-6    140-6#   142-10   142-10#   143-10   143-10#
M$PRIN  55-70    55-70#   102-12   102-12#  102-14   102-14#  102-16   102-16#  102-18   102-18#  138-52   138-52#
M$PUSH  28-26    28-26#   30-10    30-10#   31-10    31-10#   32-3     32-3#    55-14    55-14#   55-18    55-18#   55-22    55-22#
        55-26    55-26#   55-30    55-30#   55-34    55-34#   55-38    55-38#   55-42    55-42#   55-46    55-46#   55-50    55-50#
        55-54    55-54#   55-58    55-58#   55-67    55-67#   55-73    55-73#   55-87    55-87#   55-91    55-91#   55-95    55-95#
        55-99    55-99#   55-103   55-103#  55-108   55-108#  55-112   55-112#  55-116   55-116#  55-120   55-120#  55-124   55-124#
        55-128   55-128#  113-10   113-10#  114-18   114-18#  124-5    124-5#   126-3    126-3#   126-10   126-10#  127-8    127-8#
       136-10   136-10#  137-8    137-8#   138-3    138-3#   138-5    138-5#   139-3    139-3#   139-14   139-14#  141-12   141-12#
M$PUT   55-70    55-70    55-70#   61-11    61-11    61-11    61-11    61-11#   63-9     63-9     63-9     63-9     63-9#    102-12
       102-12   102-12   102-12#  102-14   102-14   102-14   102-14#  102-16   102-16   102-16   102-16#  102-18   102-18   102-18
       102-18#  105-20   105-20   105-20   105-20   105-20#  118-27   118-27   118-27   118-27   118-27#  127-59   127-59   127-59
       127-59   127-59#  138-52   138-52   138-52   138-52#
M$PUT1  55-70    55-70    55-70#   55-70#   61-11    61-11    61-11    61-11    61-11#   61-11#   61-11#   61-11#   63-9     63-9
        63-9     63-9     63-9#    63-9#    63-9#    63-9#    102-12   102-12   102-12   102-12#  102-12#  102-12#  102-14   102-14
       102-14   102-14#  102-14#  102-14#  102-16   102-16   102-16   102-16#  102-16#  102-16#  102-18   102-18   102-18   102-18#
       102-18#  102-18#  105-20   105-20   105-20   105-20   105-20#  105-20#  105-20#  105-20#  118-27   118-27   118-27   118-27
       118-27#  118-27#  118-27#  118-27#  127-59   127-59   127-59   127-59   127-59#  127-59#  127-59#  127-59#  138-52   138-52
       138-52   138-52#  138-52#  138-52#
M$RADI  135-1    135-1#   138-20   138-20#  138-34   138-34#  140-1    140-1#   140-2    140-2#   140-3    140-3#   140-4    140-4#
       140-5    140-5#   142-1    142-1#   142-3    142-3#   142-5    142-5#   142-7    142-7#
M$RBRO  66-21    66-21#   66-26    66-26#   67-7     67-7#    68-23    68-23#   68-30    68-30#   69-1     69-1#    69-11    69-11#
        69-14    69-14#   83-26    83-26#
M$RNRO 127-39   127-39#  127-48   127-48#  127-50   127-50#  128-15   128-15#  131-4    131-4#
M$SETS  28-26    28-26#   30-10    30-10#   31-10    31-10#   32-3     32-3#    55-14    55-14#   55-18    55-18#   55-22    55-22#
        55-26    55-26#   55-30    55-30#   55-34    55-34#   55-38    55-38#   55-42    55-42#   55-46    55-46#   55-50    55-50#
        55-54    55-54#   55-58    55-58#   55-67    55-67#   55-73    55-73#   55-87    55-87#   55-91    55-91#   55-95    55-95#
        55-99    55-99#   55-103   55-103#  55-108   55-108#  55-112   55-112#  55-116   55-116#  55-120   55-120#  55-124   55-124#
        55-128   55-128#  113-10   113-10#  114-18   114-18#  124-5    124-5#   126-3    126-3#   126-10   126-10#  127-8    127-8#
       136-10   136-10#  137-8    137-8#   138-3    138-3#   138-5    138-5#   139-3    139-3#   139-14   139-14#  141-12   141-12#
M$SVC   55-16    55-16#   55-20    55-20#   55-24    55-24#   55-28    55-28#   55-32    55-32#   55-36    55-36#   55-40    55-40#
        55-44    55-44#   55-48    55-48#   55-52    55-52#   55-56    55-56#   55-65    55-65#   55-70    55-70#   55-71    55-71#
        55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#  55-106   55-106#  55-110   55-110#
        55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#  57-7     57-8     57-8#    61-11
        61-11#   61-12    61-12#   61-22    61-22#   61-28    61-30    61-30#   62-20    62-21    62-21#   63-9     63-9#    65-13
        65-13#   66-16    66-16#   66-19    66-19#   66-21    66-21#   66-26    66-26#   67-7     67-7#    68-23    68-23#   68-30
        68-30#   69-1     69-1#    69-11    69-11#   69-14    69-14#   70-6     70-7     70-7#    71-9     71-9#    71-25    72-36
        73-21    74-8     78-22    80-30    83-20    83-20#   83-21    83-21#   83-26    83-26#   102-12   102-12#  102-14   102-14#
       102-16   102-16#  102-18   102-18#  105-20   105-20#  107-3    111-21   111-21#  111-29   112-5    116-35   117-11   118-12
       118-12#  118-27    118-27#  118-30   118-30#  118-33   120-44   121-27   121-27#  121-36   122-5    127-10   127-10#  127-12
       127-12#  127-14    127-14#  127-16   127-16#  127-18   127-18#  127-39   127-39#  127-48   127-48#  127-50   127-50#  127-59
       127-59#  128-15    128-15#  131-4    131-4#   133-12   133-12#  133-14   133-14#  133-16   133-16#  134-3    134-4    134-4#
       134-8    134-9     134-9#   134-14   134-15   134-15#  134-20   134-21   134-21#  135-1    135-1#   135-104  135-104#  136-12
       136-12#  137-11    137-11#  137-15   137-15#  138-8    138-8#   138-10   138-11   138-11#  138-16   138-17   138-17#  138-20
       138-20#  138-22    138-22#  138-34   138-34#  138-52   138-52#  138-137  138-137#  138-138  138-138#
M$TLAB  55-16#   55-20#   55-24#   55-28#   55-32#   55-36#   55-40#   55-44#   55-48#   55-52#   55-56#   55-65#   55-70#   55-71#
        55-85#   55-89#   55-93#   55-97#   55-101#  55-106#  55-110#  55-114#  55-118#  55-122#  55-126#  55-130#  57-7#    57-8#
        61-11#   61-12#   61-22#   61-28#   61-30#   62-20#   62-21#   63-9#    65-13#   66-16#   66-19#   66-21#   66-26#   67-7#
        68-23#   68-30#   69-1#    69-11#   69-14#   70-6#    70-7#    71-9#    71-25#   72-36#   73-21#   74-8#    78-22#   80-30#
        83-20#   83-21#   83-26#   102-12#  102-14#  102-16#  102-18#  105-20#  107-3#   111-21#  111-29#  112-5#   116-35#  117-11#
       118-12#  118-27#  118-30#  118-33#  120-44#  121-27#  121-36#  122-5#   127-10#  127-12#  127-14#  127-16#  127-18#  127-39#
```

CROSS REFERENCE TABLE (CREF V04.00)

```
                127-48#  127-50#  127-59#  128-15#  131-4#   133-12#  133-14#  133-16#  134-3#   134-4#   134-8#   134-9#   134-14#  134-15#
                134-20#  134-21#  135-1#   135-104#  136-12#  137-11#  137-15#  138-8#   138-10#  138-11#  138-16#  138-17#  138-20#  138-22#
                138-34#  138-52#  138-137#  138-138#
M$TSTL          55-16    55-16#   55-20    55-20#   55-24    55-24#   55-28    55-28#   55-32    55-32#   55-36    55-36#   55-40    55-40#
                55-44    55-44#   55-48    55-48#   55-52    55-52#   55-56    55-56#   55-65    55-65#   55-70    55-70#   55-71    55-71#
                55-85    55-85#   55-89    55-89#   55-93    55-93#   55-97    55-97#   55-101   55-101#  55-106   55-106#  55-110   55-110#
                55-114   55-114#  55-118   55-118#  55-122   55-122#  55-126   55-126#  55-130   55-130#  57-7     57-7#    57-7#    57-8
                57-8#    61-11    61-11#   61-12    61-12#   61-22    61-22#   61-28    61-28#   61-28#   61-30    61-30#   62-20    62-20#
                62-20#   62-21    62-21#   63-9     63-9#    65-13    65-13#   66-16    66-16#   66-19    66-19#   66-21    66-21#   66-26
                66-26#   67-7     67-7#    68-23    68-23#   68-30    68-30#   69-1     69-1#    69-11    69-11#   69-14    69-14#   70-6
                70-6#    70-6#    70-7     70-7#    71-9     71-9#    71-25    71-25#   71-25#   72-36    72-36#   72-36#   73-21    73-21#
                73-21#   74-8     74-8#    74-8#    78-22    78-22#   78-22#   80-30    80-30#   80-30#   83-20    83-20#   83-21    83-21#
                83-26    83-26#   102-12   102-12#  102-14   102-14#  102-16   102-16#  102-18   102-18#  105-20   105-20#  107-3    107-3#
                107-3#   111-21   111-21#  111-29   111-29#  111-29#  112-5    112-5#   112-5#   116-35   116-35#  116-35#  117-11   117-11#
                117-11#  118-12   118-12#  118-27   118-27#  118-30   118-30#  118-33   118-33#  118-33#  120-44   120-44#  120-44#  121-27
                121-27#  121-36   121-36#  121-36#  122-5    122-5#   122-5#   127-10   127-10#  127-12   127-12#  127-14   127-14#  127-16
                127-16#  127-18   127-18#  127-39   127-39#  127-48   127-48#  127-50   127-50#  127-59   127-59#  128-15   128-15#  131-4
                131-4#   133-12   133-12#  133-14   133-14#  133-16   133-16#  134-3    134-3#   134-3#   134-4    134-4#   134-8    134-8#
                134-8#   134-9    134-9#   134-14   134-14#  134-14#  134-15   134-15#  134-20   134-20#  134-20#  134-21   134-21#  135-1
                135-1#   135-104  135-104#  136-12   136-12#  137-11   137-11#  137-15   137-15#  138-8    138-8#   138-10   138-10#  138-10#
                138-11   138-11#  138-16   138-16#  138-16#  138-17   138-17#  138-20   138-20#  138-22   138-22#  138-34   138-34#  138-52
                138-52#  138-137  138-137#  138-138  138-138#
MSWORD          28-34    28-34#   29-9     29-9     29-9#    57-7     57-7     57-7     57-7#    61-28    61-28    61-28    61-28#   62-20
                62-20    62-20    62-20#   70-6     70-6     70-6     70-6#    71-25    71-25    71-25    71-25#   72-36    72-36    72-36
                72-36#   73-21    73-21    73-21    73-21#   74-8     74-8     74-8     74-8#    78-22    78-22    78-22    78-22#   80-30
                80-30    80-30    80-30#   107-3    107-3    107-3    107-3#   111-29   111-29   111-29   111-29#  112-5    112-5    112-5
                112-5#   116-35   116-35   116-35   116-35#  117-11   117-11   117-11   117-11#  118-33   118-33   118-33   118-33#  120-44
                120-44   120-44   120-44#  121-36   121-36   121-36   121-36#  122-5    122-5    122-5    122-5#   133-16#  134-3    134-3
                134-3    134-3#   134-8    134-8    134-8    134-8#   134-14   134-14   134-14   134-14#  134-20   134-20   134-20   134-20#
                135-1    135-1    135-1#   135-1#   138-10   138-10   138-10   138-10#  138-11#  138-16   138-16   138-16   138-16#  138-17#
                138-20   138-20   138-20#  138-20#  138-34   138-34   138-34#  138-34#  138-137#  140-1   140-1#   140-2    140-2#   140-3
                140-3#   140-4    140-4#   140-5    140-5#   142-1    142-1#   142-2    142-2#   142-3    142-3#   142-4    142-4#   142-5
                142-5#   142-6    142-6#   142-7    142-7#   142-8    142-8#   142-9    142-9    142-9#   144-3    144-3#
MSXFER          142-2    142-2#   142-4    142-4#   142-6    142-6#   142-8    142-8#
MANUAL          138-8
MEMORY          127-39
OPEN            66-16    83-21    138-22
OR              45-9#
PNT             47-20#   125-24   125-28   125-33
PNT...          46-21#   55-15    55-19    55-23    55-27    55-31    55-35    55-39    55-43    55-47    55-51    55-55    55-64    55-69
                55-74    55-78    55-84    55-88    55-92    55-96    55-100   55-104   55-117   55-121   55-125   55-129   55-132   55-138
                55-145   55-149   86-15    87-41    89-23    92-4     93-5     94-5     125-24   125-28   125-33   127-53   135-31   138-19
                138-31
PNTB            47-11#   55-15    55-19    55-23    55-27    55-31    55-35    55-39    55-43    55-47    55-51    55-55    55-64    55-69
                55-74    55-78    55-84    55-88    55-92    55-96    55-100   55-104   55-117   55-121   55-125   55-129   55-132   55-138
                55-145   55-149   93-5     94-5
PNTF            47-8#    86-15    87-41    89-23    92-4     127-53   135-31   138-19   138-31
PNTS            47-17#
PNTX            47-14#
POINTE          28-32
POP             45-23#   58-21    61-23    69-7     70-2     79-44    79-49    85-17    89-24    95-7     96-7     97-7     99-31    101-17
                101-25   102-19   103-18   104-28   108-29   108-30   109-22   110-25   111-30   112-2    112-6    114-20   115-41   119-19
                121-24   125-29   125-34
PRINT           46-8#    84-13    84-18    86-21    88-6     88-13    89-10    89-30    93-7     94-7     101-13   101-22   125-27   125-32
                125-35
PRINTB          102-14
```

CROSS REFERENCE TABLE (CREF V04.00)

```
PRINTF  102-12   138-52
PRINTS  102-18
PRINTX   55-70   102-16
PUSH     45-15#   58-16    61-9     66-13    79-33    79-39    85-11    95-4     96-4     97-4     99-13    100-16   100-22   100-40
        102-6    103-12   103-16   104-16   108-15   109-14   110-16   111-11   115-14   118-13   121-18   125-15   125-22
READEF  127-10   127-12   127-14   127-16
SETPRI  133-12
SETVEC   61-11    63-9    105-20   118-27   127-59
SVC      28-7#    28-8
TABLE    33-20#   43-10    44-7     139-18   141-16
XFER    133-16#  138-11#  138-17#  138-137#
XFERT   142-2    142-4    142-6    142-8
```