RSX-11M/M-PLUS and Micro/RSX
Crash Dump Analyzer Reference Manual

Order No. AA-FD11A-TC

digital software

RSX–11M/M–PLUS and Micro/RSX
Crash Dump Analyzer Reference Manual

Order No. AA-FD11A-TC

RSX-11M Version 4.2
RSX-11M-PLUS Version 3.0
Micro/RSX Version 3.0

digital equipment corporation · maynard, massachusetts

First Printing, May 1979
Revised, January 1982
Updated, April 1982
Revised, April 1983
Revised, July 1985

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Printed in U.S.A.

The postpaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DEC	DIBOL	PDT
DEC/CMS	EduSystem	RSTS
DEC/MMS	IAS	RSX
DECnet	MASSBUS	UNIBUS
DECsystem-10	MicroPDP-11	VAX
DECSYSTEM-20	Micro/RSTS	VMS
DECUS	Micro/RSX	VT
DECwriter	PDP	digital

ZK2572

HOW TO ORDER ADDITIONAL DOCUMENTATION

In Continental USA and Puerto Rico call 800-258-1710

In New Hampshire, Alaska, and Hawaii call 603-884-6660

In Canada call 800-267-6215

DIRECT MAIL ORDERS (USA & PUERTO RICO)*

Digital Equipment Corporation
P.O. Box CS2008
Nashua, New Hampshire 03061

*Any prepaid order from Puerto Rico must be placed with the local Digital subsidiary (809-754-7575)

DIRECT MAIL ORDERS (CANADA)

Digital Equipment of Canada Ltd.
100 Herzberg Road
Kanata, Ontario K2K 2A6
Attn: Direct Order Desk

DIRECT MAIL ORDERS (INTERNATIONAL)

Digital Equipment Corporation
PSG Business Manager
c/o Digital's local subsidiary or approved distributor

Internal orders should be placed through the Software Distribution Center (SDC), Digital Equipment Corporation, Northboro, Massachusetts 01532

CONTENTS

FIGURES

TABLES

PREFACE

MANUAL OBJECTIVES

This manual describes the operation of the Crash Dump Analyzer (CDA). It does not attempt to describe the operation of the RSX-11M-PLUS Executive or the significance of the individual data structures. The RSX-11M/M-PLUS Executive Reference Manual and the RSX-11M and RSX-11M-PLUS Guide to Writing an I/O Driver describe these data structures.

INTENDED AUDIENCE

This manual is intended for system managers who are responsible for interpreting system failures and for system operators who run CDA to generate dumps. Understanding CDA output requires a working knowledge of assembly language programming and the Executive data structures.

STRUCTURE OF THIS MANUAL

Chapter 1 explains the function of the Crash Dump Analyzer. It describes the system resources necessary and the procedure for obtaining a crash dump. The chapter continues with an overview of loadable crash dump drivers, and explains how to run CDA. The chapter also describes indirect command files as they pertain to CDA.

Chapter 2 describes the two ways that you can use CDA: by issuing the CDA command line or the DCL ANALYZE/CRASH_DUMP command. The chapter begins by describing the CDA command line format, including command line specifications and switches. Two summary tables provide quick reference on switch operation. Finally, the chapter concludes with a description of the DCL ANALYZE/CRASH_DUMP command.

Chapter 3 consists of examples and descriptions of CDA output listings.

Chapter 4 contains helpful hints for interpreting CDA output listings.

Appendix A contains a short description of each CDA error message.

Appendix B lists system macros that supply symbolic offsets for system data structures for RSX-11M.

Appendix C lists system macros that supply symbolic offsets for system data structures for RSX-11M-PLUS.

Appendix D lists error code definitions for Micro/RSX operating systems.

ASSOCIATED MANUALS

Refer to the RSX-11M/RSX-11S Information Directory and Index for a brief description of each manual in the RSX-11M documentation set.

Refer to the RSX-11M-PLUS Information Directory and Index for a brief description of each manual in the RSX-11M-PLUS documentation set.

CONVENTIONS USED IN THIS MANUAL

RET	This symbol indicates that you press the RETURN key.
[]	Square brackets show elements in a command line format that are optional. For example, [/switch] indicates that you can include a switch if you want to, but you do not have to.
[,...]	Square brackets around a comma and an ellipsis mark indicate that you can use a series of optional elements separated by commas. For example, (argument[,...]) means that you can specify a series of optional arguments by enclosing the arguments in parentheses and separating them with commas.
red ink	Red ink in the examples of this manual denotes user input.
pink shading	Pink shading in this manual indicates features that are specific to RSX-11M operating systems only.
gray shading	Gray shading indicates features that are specific to RSX-11M-PLUS operating systems only.

SUMMARY OF TECHNICAL CHANGES

TECHNICAL CHANGES

- The system Assign Table, which is one of the report listings that CDA generates, has been changed. The table now lists logical assignments in two categories: system logical assignments, and user logical assignments. The entry in the table for each assignment includes its size in blocks, its type, and its status.

- Some of the system data structures that are shown in Appendix B and Appendix C have changed as a result of new system features such as support for logical names and networking. Refer to the specific data structures in Appendix B (RSX-11M) or Appendix C (RSX-11M-PLUS) to see the changes.

NEW DEVICE SUPPORT

You can specify any of the following new devices as the crash dump device for your system:

For RSX-11M/M-PLUS Systems

Device Type	Mnemonic
RA60/RX50 disk packs	DU:
RC25 removable disk packs	DU:
TK25/TU80 magnetic tapes	MS:
TK50 magnetic tapes	MU:

For Micro/RSX or Pregenerated RSX-11M-PLUS Systems

Device Type	Mnemonic
RD52 disks	DU:
TK25 magnetic tapes	MS:
TK50 magnetic tapes	MU:

ADDITIONS TO THE CRASH DUMP ANALYZER REFERENCE MANUAL

The following documentation has been added to this manual:

- Chapter 1 now includes a description of loadable crash dump support for Micro/RSX and pregenerated RSX-11M-PLUS systems. You load a crash dump driver by specifying a crash dump device. If the system crashes when the driver is loaded, the driver dumps the contents of memory at the time of the crash

onto the specified crash dump device. When you do not want a crash driver resident in memory, you can unload it to the system disk. Thus, loadable crash drivers allow you to choose when you want crash dump support.

- Chapter 2 now includes a description of the DCL ANALYZE/CRASH_DUMP command. If your terminal supports the DIGITAL Command Language (DCL) command line interpreter, you can use the ANALYZE/CRASH_DUMP command to run CDA. Command qualifiers let you choose which report listings you want CDA to generate. You can also use qualifiers to specify the format of the CDA report listings.

- A new appendix, Appendix D, lists error code definitions for Micro/RSX operating systems, including facility-independent definitions and Bugcheck standard format definitions.

CHAPTER 1

INTRODUCTION

This chapter introduces the Crash Dump Analyzer (CDA). It describes the function of CDA, details the system features that CDA requires, and explains how to obtain a crash dump. The procedures for generating a crash dump vary from system to system. This chapter explains how to obtain a crash dump on different types of systems. Then this chapter describes how to run CDA, and how to use CDA with indirect command files. Finally, the last section of the chapter lists the six basic analysis listings that the Crash Dump Analyzer generates.

1.1 CRASH DUMP ANALYZER FUNCTION

CDA is a specialized utility that helps you establish the cause of system crashes. It is installed in a system as a nonprivileged task that any user can run. CDA reads the contents of a memory dump created by the crash dump routine of the Executive. CDA then uses the data in the Executive symbol table file (RSX11M.STB) to format the binary input of the memory dump into readable analysis listings. Finally, CDA prints the analysis listings on a line printer. Examining the CDA listings can help you to determine the cause of a system crash.

CDA is a nonprivileged task that any user can run.

1.2 SYSTEM REQUIREMENTS

Micro/RSX operating systems with the Advanced Programmer's Kit and pregenerated RSX-11M-PLUS operating systems support loadable crash dump drivers. Refer to Section 1.4 for a description of loadable crash dump support.

On RSX-11M operating systems, and on non-pregenerated RSX-11M-PLUS operating systems, you can select support for crash dump analysis during system generation. Refer to the RSX-11M System Generation and Installation Guide or the RSX-11M-PLUS System Generation and Installation Guide for instructions on how to include CDA in your system. If you select support for crash dump analysis during system generation, you specify a crash notification device and a crash dump device. The system then builds a crash dump routine into the Executive. Thereafter, when the system crashes, the crash dump routine displays a message on the crash notification device and writes the contents of memory onto the specified crash dump device. The contents of memory are the input to CDA. If you decide to change the crash dump or crash notification devices, you must perform another system generation in order to specify the new devices.

Since CDA overwrites the contents of the crash dump device, you should not specify the system device as the crash dump device. Also, the following fixed disks cannot serve as CDA crash dump devices:

RA80
RA81
RD51
RC25

However, you may use any of the following mass storage devices as the crash dump device for your system.

For RSX-11M/M-PLUS Operating Systems

Device	Mnemonic
RP04/RP05/RP06 disk packs	DB:
DECtape II (TU58)	DD:
RK05/J/F disk cartridge	DK:
RL01/RL02 disks	DL:
RK06/RK07 disk cartridges	DM:
RM02/RM03/RM05 disk packs	DR:
DECtape (TU56)	DT:
RC25 removable disk pack	DU:
RA60/RX50 disk packs	DU:
RX01 diskette (RSX-11M only)	DX:
RX02 diskette	DY:
TU45/TU16/TE16/TU77 magnetic tapes	MM:
TS11/TU80/TSV05/TK25 magnetic tapes	MS:
TS03/TU10/TE10 magnetic tapes	MT:
TK50 magnetic tapes	MU:

For Micro/RSX and Pregenerated RSX-11M-PLUS Operating Systems

Device	Mnemonic
RL01/RL02 disks	DL:
RD51/RD52/RX50 disks	DU:
TSV05/TK25 magnetic tape	MS:
TK50 magnetic tapes	MU:

1.3 OBTAINING A CRASH DUMP

To obtain a crash dump, control of the processor must be transferred to the Executive crash dump routine following a system crash. The transfer of processor control depends on how the crash occurred and whether you built the Executive Debugging Tool (XDT) into your system during system generation.

System crashes can result from any of the following causes:

1. The processor encounters a program condition that causes it to trap to location 40 or to XDT.
2. An infinite loop condition occurs.
3. The processor encounters an unintentional HALT instruction in kernel mode (000000).

When a program condition causes a processor trap and XDT is included in your system, control transfers automatically to XDT. You can then type X at the console terminal, and XDT transfers control to the crash dump routine. For example:

```
XDT>X (RET)
```

Refer to the RSX-11M/M-PLUS and Micro/RSX Debugging Reference Manual for a description of XDT.

If your system does not include XDT, a processor trap causes control to be transferred directly to the crash dump routine of the Executive.

When a system crash is the result of a HALT instruction or an infinite loop condition, you must restart the processor manually at location 40.

Regardless of how control is transferred, once the processor enters the crash dump routine, the routine prints the following informational message on the crash notification device:

```
CRASH-CONT WITH SCRATCH MEDIA ON ddnn
```

After displaying the message, the crash dump routine halts the processor so you can put the crash dump device on line. When the device is on line, restart the processor by depressing the Continue switch on the processor console. The crash dump routine then dumps memory on the crash dump device and halts the processor when the dump finishes. The volume in the crash dump device now contains a binary representation of the contents of memory at the time of the crash. These contents are the input to CDA. You can then reboot the system and run CDA to analyze the dump.

If you attempt to crash to an illegal device, the crash dump routine displays the following message on the crash notification device:

```
CRASH -- ILLEGAL CRASH DEVICE
```

After displaying the message, the crash dump routine halts. The illegal crash device error occurs if you specify a fixed media device as the crash dump device. If you have a removable media device on the same controller, you can switch the physical unit number plugs on the devices to assign the removable media device to the crash device. Then press the Continue key on the operator's console and the crash dump routine will attempt the dump again.

1.4 LOADABLE CRASH DUMP DRIVERS

The pregenerated RSX-11M-PLUS operating system and the Privileged Development option of the Micro/RSX Advanced Programmer's Kit include loadable crash dump drivers. Loadable drivers reside on an external storage device when they are not in use. Using loadable drivers for crash dump support reduces the size of the Executive and frees memory space for other purposes.

1.4.1 Crash Devices

Loadable crash dump support is provided by four loadable crash dump drivers, each of which dumps the contents of memory to a specific type of device. The following list shows the crash dump drivers and their corresponding device types.

Crash Dump Driver	Crash Dump Device
DLCRSH.TSK	RL02 cartridge disk
DUCRSH.TSK	RX50 diskette
MSCRSH.TSK	TSV05/TK25 magnetic tape
MUCRSH.TSK	TK50 magnetic tape(Micro/RSX) TK50 magnetic tape(RSX-11M-PLUS)

If the crash dump driver is loaded and the system crashes, the contents of memory are dumped to the designated crash device. You can then use the Crash Dump Analyzer to investigate the cause of the crash. If there is not a crash driver resident in the system when the system crashes, the Bugcheck facility displays the following message:

```
SYSTEM FAULT DETECTED AT PC=xxxxxx FACILITY=xxxxxx ERROR CODE=xxxxxx
CRASH -- CRASH DRIVER NOT LOADED
nnnnnn
@?
```

1.4.2 Loading a Crash Dump Driver

A loadable crash driver resides on the system disk until you specify a crash device. To specify a crash device, use the following command:

```
SET SYSTEM /CRASH_DEVICE:ddn: RET
```

This command loads a specific crash driver into a main memory partition and updates the crash data base. Also, you may use this command to change the crash dump device or to change the unit number of the crash device while the system is running.

When the crash driver is successfully loaded, you receive the following message:

```
SET -- Crash device ddn:  has been successfully loaded
```

If the device that you specified as the crash device is not in the current system, the following error message is displayed:

```
SET -- Device not in system
```

If a crash driver is already loaded and you specify a different device with the SET SYSTEM/CRASH_DEVICE command, the system unloads the resident crash driver, loads the new driver for the device that you specified, and updates the crash data base. If a crash driver is loaded and you specify the same device but a different unit number, the resident driver remains loaded and the system changes the device unit number in the crash data base.

If you specify the system disk as a crash device, you receive the following warning message:

```
SET -- WARNING, System disk chosen as crash device
SET -- Crash device ddn:  has been successfully loaded
```

Note that if your system disk is a removable disk, it is a valid crash dump device. The system warns you that you have specified the system disk, but it loads the crash dump driver for the disk despite the warning message. You should avoid using the system disk as the crash dump device, because the memory dump will overwrite the contents of the disk, unless you remove the system disk and replace it with a scratch disk when the system crashes.

It is not possible to crash to a fixed media device such as the RD51 fixed disk. If you indicate the RD51 or any other fixed disk as a crash device, you receive the following message:

```
CRASH -- ILLEGAL CRASH DEVICE
CRASH -- CONT WITH SCRATCH MEDIA ON ddn
```

At this point, you cannot obtain a crash dump of memory.

NOTE

You cannot select a crash device and unit number once the crash has occurred.

However, you may choose a crash device unit that is not in the current system. To do this, specify the address of the control and status register (CSR) of the device that you want as the crash device. Use the /REGISTER switch to specify the address of the CSR of the desired device:

```
SET SYSTEM /CRASH_DEVICE:ddn:REGISTER:csraddr (RET)
```

To display the current crash dump device unit, use the following command:

```
$ SHOW SYSTEM /CRASH_DEVICE (RET)
```

In response to this command, the system displays the current device unit, as follows:

```
CRASHDEV=ddn:
```

1.4.3 Unloading a Crash Dump Driver

You use the following command to unload a crash dump driver when crash dump support is unnecessary:

```
$ SET SYSTEM /NOCRASH_DEVICE (RET)
```

In response to this command, the system displays the following message:

```
SET -- Crash device ddn:  is being unloaded
SET -- WARNING, Crash dump support is inactive
```

The system then unloads the crash dump driver and updates the crash data base. When there is no crash dump driver resident in memory, the Bugcheck facility services system crashes (refer to Appendix D for a list of error code definitions used by Bugcheck). Unloading the crash dump driver frees the memory space in the crash driver partition until you decide to reactivate crash dump support. You can reactivate crash dump support at any time simply by specifying a new crash device unit.

1.4.4 When the System Crashes

When a Micro/RSX or pregenerated RSX-11M-PLUS operating system crashes, the reaction of the system depends on the type of crash support that is loaded when the crash occurs. There are three types of crash support:

1. The Bugcheck facility, which is a standard part of the operating system, and is therefore resident in memory
2. Loadable crash dump drivers
3. XDT, which is also loadable

Thus, when a system crashes, any of the following conditions may exist:

- A crash dump driver is loaded but XDT is not
- Both a crash dump driver and XDT are loaded
- XDT is loaded but a crash dump driver is not

1.4.4.1 **A System Crash with a Driver Loaded and XDT Unloaded** - If a system crashes when a crash driver is loaded but XDT is not loaded, the crash dump routine notifies you of the crash with the following message:

```
CRASH -- CONT WITH SCRATCH MEDIA ON ddn
```

After displaying this message, the crash routine halts the hardware processor so that you can make sure there is a scratch media in the crash device. When you have the crash device ready, press the P key followed by a carriage return to proceed.

In response to your command to proceed, the crash dump routine dumps memory to the designated crash dump device. When the dump is completed, the processor is again halted. During the memory dump, the processor Run light is on; when the dump is completed, the processor Run light goes off.

At this point, the medium in the crash dump device contains a binary representation of the contents of memory at the time the system crash occurred. This memory dump is the input to CDA. Now you can use the ANALYZE/CRASH_DUMP command, which is described in Section 2.2 of this manual, to control how CDA processes the crash dump. Then you can analyze the output listings that CDA generates to determine why your system crashed.

1.4.4.2 **A System Crash with a Driver Loaded and XDT Loaded** - If a system crashes when a crash driver and XDT are loaded, control is transferred to XDT. After you use XDT to debug the system, if you want to obtain a crash dump, press the X key followed by a carriage return. The following message is then displayed:

```
CRASH -- CONT WITH SCRATCH MEDIA ON ddn
```

Now you can follow the procedure in Section 1.4.4.1 to obtain the crash dump.

1.4.4.3 **A System Crash with only XDT Loaded** - If a system crashes when XDT is loaded but a crash driver is not loaded, control is transferred to XDT. However, when you enter the X command, the following is displayed:

```
SYSTEM FAULT DETECTED AT PC=xxxxxx FACILITY=xxxxxx ERROR CODE=xxxxxx
CRASH -- CRASH DRIVER NOT LOADED
```

1.4.4.4 **Inducing a System Crash** - In some situations, you may want to purposely induce a system crash. Then, if you have a crash driver loaded, you can dump the contents of memory and examine them. For example, suppose that you want to stop the processor from executing in an infinite loop. You can induce a system crash by performing the following procedure:

1. Push the Halt button on the processor. On processors with console ODT, the following is displayed:

   ```
   nnnnnn
   @
   ```

2. Release the Halt button.

3. At the terminal, type 40G. The following is displayed:

   ```
   CRASH -- CONT WITH SCRATCH MEDIA ON ddn

   nnnnnn
   @
   ```

If you have a crash driver loaded, you can obtain a crash dump now by pressing the P key followed by RETURN.

1.5 RUNNING CDA

There are several ways to run CDA, and you can run it as either an installed or an uninstalled task. Also, you can run CDA from either the DCL or MCR command line interpreter (CLI). This section describes the alternative ways of running CDA.

If CDA is an installed task on your system, you can enter the CDA command line at the CLI prompt. After CDA processes your command, the CLI prompt returns. In the following example, MCR is the CLI:

```
>CDA CRASH_DUMP.LST,COPY.CDA=[1,54]/STB,DR5: RET
>
```

If CDA is installed and you want to enter commands directly to CDA, you can invoke the command level of the CDA utility by typing CDA and a carriage return. When you are finished using CDA, you exit from CDA by pressing CTRL/Z, which returns control to the CLI. In the following example, DCL is the CLI:

```
$ CDA RET
CDA>command line RET
CDA>command line RET
CDA>^Z
$
```

If CDA is an uninstalled task, the system has to find and install the CDA task image file before it can run CDA. Therefore, the command you use depends upon the location of the CDA task image file (CDA.TSK). If CDA.TSK is in the system UFD or the system library, type:

```
RUN $CDA (RET)
CDA>command line (RET)
CDA>
```

On RSX-11M operating systems, you can use the RUN $CDA command only if CDA.TSK is present in the UFD that corresponds to the system UIC on device LB:. On RSX-11M-PLUS operating systems, CDA.TSK must be present in the UFD that corresponds to the library UIC.

If CDA.TSK is present in the UFD that corresponds to the current UIC on the default system device (the current UFD for the terminal from which the command is entered), you can run CDA by typing the following command:

```
RUN CDA (RET)
CDA>command line (RET)
```

Finally, you can run CDA by using the DCL ANALYZE/CRASH_DUMP command:

```
$ ANALYZE/CRASH_DUMP (RET)
```

If your CLI is MCR, but your terminal also supports DCL, you can run the ANALYZE/CRASH_DUMP command by typing DCL and a space before the command. For example:

```
>DCL ANALYZE/CRASH_DUMP (RET)
```

Chapter 2 shows you how to use CDA command lines and the ANALYZE/CRASH_DUMP command.

1.6 INDIRECT COMMAND FILES

As with other utilities, you can enter CDA command lines directly from the terminal or from an indirect command file. However, CDA indirect command files must not contain a reference to another command file.

1.7 BASIC CRASH DUMP ANALYZER OUTPUT LISTING

While the Crash Dump Analyzer provides many output listing options, fundamental system information appears on the first six pages of output listing (you can suppress this information by using the /-SYS switch, which is described in Chapter 2). The first six pages of output listings contain the following information:

Page 1 -- Volatile registers

Page 2 -- Kernel stack

Page 3 -- System common

Page 4 -- System common labeled dump

Page 5 -- Pool statistics

Page 6 -- Assign table

Sections 3.1.1 through 3.1.6 describe these pages in detail.

The system information section also includes three more pages if the relevant information is in memory at the time of the crash. These pages display group-global event flags, error log packets, and, on RSX-11M-PLUS systems, the contents of low core memory. Section 3.1.7 describes the group-global event flag page, Section 3.1.8 describes the error log page, and Section 3.1.9 describes the RSX-11M-PLUS low core memory page that is part of the system common dump.

CHAPTER 2

COMMAND LINES

CDA commands control how the Crash Dump Analyzer processes a memory dump and how it formats the output listings that it generates. You can use CDA command lines to enter commands directly to the CDA utility or, if your terminal supports the DIGITAL Command Language (DCL), you can use the DCL ANALYZE/CRASH_DUMP command to run CDA. This chapter describes CDA command lines and the ANALYZE/CRASH_DUMP command by showing the format of the command lines, the command specifications and qualifiers, and examples of how the commands work.

2.1 CDA COMMAND LINES

This section shows the CDA command line format, lists and describes command line switches, and provides some examples of CDA command lines.

The CDA command line has the following format:

```
CDA>[listfile/sw],[binaryfile/sw]=[symbolfile/STB],crash-input[/sw]
```

The CDA command line specifies the input to CDA and the output from CDA. The specifications to the left of the equal sign in the command line are output specifications, and those on the right side of the equal sign are input specifications.

You must include at least one output specification and one input specification in the command line. For output from CDA, you can specify a list file only, a binary file only, or both a list file and a binary file. For input to CDA, you must specify the crash-input, but the symbol file specification is optional.

Output file specifications are position dependent. Position dependent means that when you include both output specifications, you must place them in the positions shown in the command line. If you omit the list file, you must place a comma before the binary file specification.

Input file specifications are position independent and can appear in either order.

The remainder of this section describes CDA command line specifications.

Output Specifications:

listfile

The output specification of the formatted CDA analysis listings. You can use either a device or a file as the list file specification. If you specify a file, CDA creates the file and writes the output listings to the file. By default, CDA then

spools the file to the system line printer queue, unless you specify otherwise. If you specify a device for the list file, CDA displays or prints its output listings on that specific device. For example, if you specify your terminal (TTnn: or TI:) as the list file, CDA displays the output listings on your terminal. Chapter 3 describes the analysis listings that CDA generates.

binaryfile

The file specification for the optional binary file. This file is a copy of the binary data that the crash dump routine wrote on the crash dump device. It allows you to selectively create an historical record of crash dumps. If you create this file during an initial analysis, you can use it for input to CDA at a later time. Since the crash dump routine overwrites the information on the crash dump volume with each successive dump, this feature allows you to use a single volume for all crash dumps.

If the crash dump device on your system is a secondary storage or sequential device, you can reduce CDA analysis time by copying the crash input to a binary file on another device. Then you can use the binary file as input to CDA for analysis.

Input Specifications:

symbolfile/STB

The file specification of the symbol table file for the crashed system. The /STB switch is an integral part of this file specification, because CDA uses the data in the symbol table file to format the binary memory dump into readable formats. If you omit this file specification and switch, CDA uses the default symbol table file, which is the file named RSX11M.STB in the UFD that corresponds to the current UIC.

crash-input

The source of the binary input to CDA. This specification can be either a device name (the crash dump device) or a binary file that was created during a previous CDA analysis. However, if the crash-input specification is a binary file, you cannot also include a binary file output specification in the command line.

Switches:

/sw

An optional CDA switch. The list file, binary file, and crash-input file specifications can include optional switches that modify CDA action. Each specification in the command line has its own switches. Section 2.1.1 describes the CDA switches and lists which specification they apply to.

File specifications in the CDA command line can appear in complete Files-11 format, with device name, UFD, file name, file type, and version number. When you omit any of these elements, CDA uses the defaults shown in Table 2-1. However, not all of the elements in file specifications have defaults.

Table 2-1
File Default Values

File	Device	UFD	Default Value	
			File Name	File Type
List file	SY:	Current	None	.LST
Binary file	SY:	Current	None	.CDA
Symbol file/STB	SY:	Current	RSX11M	.STB
Crash-input	SY:	Current	None	.CDA

See Section 2.1.2 for examples of CDA command lines, which include examples that show how CDA uses default file types.

2.1.1 CDA Command Line Switches

Two kinds of command line switches, analysis switches and function switches, allow you to control CDA operation.

Analysis switches determine which analysis routines CDA applies to the crash input. Thus, you can select the types of data that you want CDA to output. For example, analysis switches can list information about all of the devices in the system, or they can list information about active devices only.

Function switches provide a number of options for controlling CDA output. For example, function switches can terminate an analysis after CDA encounters a specified number of errors, or they can limit the number of pages of output listings.

Both types of switches are file specific. That is, each switch applies to a particular file and may not be used without that file or with any other file.

2.1.1.1 **Analysis Switches** - Table 2-2 summarizes the analysis switches and gives brief descriptions of their effects. Some of the switches in Table 2-2 have synonyms or alternate mnemonics. These are shown under each switch. Expanded descriptions of each switch follow the table.

Table 2-2
Summary of CDA Analysis Switches

Switch	Function	Applies to File
/ACT /ATL	Lists the contents of the Task Control Block (TCB) for each active task	Crash-input
/ADV	Lists information for all devices in the system	Crash-input

(Continued on next page)

Table 2-2 (Cont.)
Summary of CDA Analysis Switches

Switch	Function	Applies to File
/ALL	Lists the output of all analysis routines	Crash-input
/CLI /CPB	Lists the contents of the CLI Parser Blocks in the system	Crash-input
/CLQ	Lists the contents of the clock queue	Crash-input
/CTL	Lists information for each device controller	Crash-input
/DEV /DCB /SCB /UCB	Lists information for all active devices in the system	Crash-input
/DUMP:a:b:[c] /DMP:a:b:[c]	Lists the contents of physical memory between address a and address b; (c is an optional virtual starting address)	Crash-input
/HDR	Lists the contents of the task headers for each task resident in memory	Crash-input
/KDS:a:b	Lists the contents of the kernel data space from virtual address a to virtual address b (RSX-11M-PLUS systems only)	Crash-input
/KIS:a:b	Lists the contents of kernel instruction space from virtual address a to virtual address b (RSX-11M-PLUS systems only)	Crash-input
/PCB /PAR	Lists the contents of each Partition Control Block	Crash-input
/POOL	Lists the contents of the system's pool	Crash-input
/SECPOL	Lists the contents of system secondary pool (RSX-11M-PLUS systems only)	Crash-input
/-SYS	Suppresses listing of the system information	Crash-input
/TASK:name:a:b /TAS:name:a:b /TSK:name:a:b	Lists the contents of task "name" between virtual address a and virtual address b; lists the contents of task data space (if task includes data space) on RSX-11M-PLUS	Crash-input

(Continued on next page)

Table 2-2 (Cont.)
Summary of CDA Analysis Switches

Switch	Function	Applies to File
/TCB /TAL /STD	Lists the contents of the TCB for every task in the System Task Directory	Crash-input
/TDS:name:a:b	Lists the contents of task data space (RSX-11M-PLUS only)	Crash-input
/TIS:name:a:b	Lists the contents of task instruction space (RSX-11M-PLUS systems only)	Crash-input

/ACT or /ATL (Task Control Blocks for Active Tasks)

File: Crash-input

Effect: CDA lists the contents of the Task Control Block (TCB) for each active task.

/ADV (All Devices)

File: Crash-input

Effect: CDA lists the contents of the control blocks for all devices in the system. To list active devices, use the /DEV switch.

/ALL (All Analysis Routines)

File: Crash-input

Effect: CDA applies all of its analysis routines (except those associated with memory and task dumps) to the specified crash-input. The output from these routines is listed in the following order:

1. System information
2. Active tasks information
3. Task headers information
4. Partition information
5. Common Block Directory entries
6. Device information
7. Clock queue contents
8. Device controller information
9. Pool contents

/CLI or /CPB (Command Line Interpreter Parser Blocks)

File: Crash-input

Effect: CDA lists the contents of all Command Line Interpreter Parser Blocks (CPBs) in the system.

/CLQ (Clock Queue)

File: Crash-input

Effect: CDA lists the contents of the clock queue.

/CTL (Device Controllers)

File: Crash-input

Effect: CDA lists the contents of the controller table and Controller Request Block (KRB) for each device controller in the system.

/DEV, /DCB, /SCB, or /UCB (Devices in System)

File: Crash-input

Effect: CDA scans the system device tables and lists the contents of the control blocks for each active device in the system. To list all devices, use the /ADV switch.

/DUMP:a:b:[c] or /DMP (Physical Memory)

File: Crash-input

Effect: If only a and b are specified, CDA dumps the contents of physical addresses a through b inclusive and labels them with their physical addresses. If a, b, and c are specified, CDA dumps the contents of physical addresses a through b, but labels them with dummy virtual addresses, starting at the address specified by c.

CDA allows you to specify a virtual starting address because RSX-11M and RSX-11M-PLUS systems use physical memory in terms of virtual addresses. If you dump physical memory labeled with the corresponding virtual addresses, you do not have to translate physical addresses to virtual addresses as you read the dump.

/HDR (Headers for Memory-Resident Tasks)

File: Crash-input

Effect: CDA lists the contents of the task headers for each task resident in memory.

/KDS:a:b (Kernel Data Space)

File: Crash-input

Effect: CDA lists the contents of kernel data space between the virtual addresses a and b inclusive.

/KIS:a:b (Kernel Instruction Space)

File: Crash-input

Effect: CDA lists the contents of kernel instruction space between the virtual addresses a and b inclusive.

/PCB or /PAR (Partition Control Blocks)

File: Crash-input

Effect: CDA outputs a map that lists all the occupants of memory and the contents of each Partition Control Block (PCB).

/POOL:a:b (System Pool)

File: Crash-input

Effect: CDA lists the system pool in octal, Radix-50, and ASCII.

/SECPOOL[:a:b] (Secondary Pool)

File: Crash-input

Effect: Lists the contents of the secondary pool on RSX-11M-PLUS systems.

/STD, /TCB, or /TAL (System Task Directory)

File: Crash-input

Effect: CDA lists the contents of all of the Task Control Blocks in the System Task Directory (STD) at the time of the crash.

/-SYS (System Information)

File: Crash-input

Effect: CDA suppresses the system information listing.

/TASK:name:a:b, /TAS, or /TSK (Task Virtual Address Space)

File: Crash-input

Effect: CDA lists the virtual address space from the 16-bit virtual address a through b for the task specified by "name." If you do not specify addresses, CDA lists the task's entire virtual address space.

/TDS:name[:a:b] (Task Data Space)

File: Crash-input

Effect: CDA lists the contents of the task data space between the virtual addresses a and b inclusive. If you do not specify addresses, CDA lists the entire task data space.

/TIS:name:a:b (Task Instruction Space)

File: Crash-input

Effect: CDA lists the contents of the task instruction space between the virtual addresses a and b inclusive. If you do not specify addresses, CDA lists the entire task instruction space.

2.1.1.2 **Function Switches** - Table 2-3 summarizes the function switches and gives brief descriptions of their effects. Expanded descriptions of each switch follow the table.

Table 2-3
Summary of CDA Function Switches

Switch	Function	Applies to File	Default[1]
/BL:n	Identifies the starting block number of the crash-input device; the value of n must be less than 65535.	Crash-input	n=1
/DENS:n :HIGH :LOW	Sets density of crash input tape to 800 or 1600 bits per inch (bpi)	Crash-input	n=800
/EXIT:n	Terminates analysis after encountering n analysis errors	List file	--
/LIMIT:n	Limits output listing to n pages	List file	n=300.
/LINES:n	Limits page length to n lines	List file	n=60.
/MEMSIZ:n	Saves nKb memory from crash in a binary file	Binary file	n=124.
/KMR	Forces the assignment of kernel address register values for the crashed system	Crash-input	/-KMR
/-SP	Does not print analysis output listing	List file	/SP
/STB	Identifies the file specification that contains the Executive symbol table	Symbol file	--

1. n can be expressed as an octal or decimal number. A decimal point (.) following the number denotes decimal.

/BL:n (Identify Starting Block Number)

File: Crash-input

Effect: CDA reads the dump from the input device beginning at block n. If the crash dump device is not a disk or a DECtape, CDA ignores this switch.

Default: n = 1

/DENS:n (Sets Tape Density)

:HIGH
:LOW

File: Crash-input

Effect: CDA reads the crash input tape at the density specified: 800 or 1600 bpi. You can also use LOW to indicate 800 bpi or HIGH to indicate 1600 bpi.

Default: n=800

/EXIT:n (Exit After n Errors)

File: List file

Effect: CDA maintains an error count. As it encounters inconsistencies in the system data structures, it increments this count. If you specify the /EXIT:n switch, CDA terminates analysis after n errors. If you specify the /EXIT switch but do not specify n, CDA exits after one error.

Default: CDA runs to completion.

/LIMIT:n (Limit Output Listing)

File: List file

Effect: The /LIMIT:n switch limits the number of pages of analysis output. When CDA has generated n pages, it terminates the analysis and prints a message on the user terminal indicating that it has done so.

Default: n = 300.

/LINES:n (Print n Lines per Page)

File: List file

Effect: This switch lets you specify the number of lines you want CDA to print per page. After n lines are printed, a new page is ejected.

Default: n=60.

/MEMSIZ:n (Establish Size of Binary Output File)

File: Binary file

Effect: This switch causes CDA to create a binary output file 4*n blocks long and to transfer nKb words to it from the crash-input file. The value of n must be greater than 16.

This switch is particularly useful when transferring binary crash dumps from disk or DECtape. Since disks and DECtapes have no physical EOFs, it is necessary to specify the size of the actual memory dump.

When the crash input resides on magnetic tape, the binary output file is filled with zeroes if the EOF is read before nKb words are transferred.

Default: n = 124.

/KMR (Assign Kernel Mapping Register Values)

File: Crash-input

Effect: On mapped systems, when CDA reads incorrect Page Address Register (PAR) values from the crash stack, it aborts the analysis and prints an error message on the terminal. If this happens, you can use the /KMR switch to retry the analysis. When you specify /KMR, CDA uses standard mapping values to convert kernel virtual addresses to physical memory addresses.

Default: CDA uses existing Page Address Registers.

/-SP (Do Not Spool)

File: List file

Effect: CDA does not spool the analysis output listing to the system line printer queue. Instead, it creates an output list file on the device indicated in the output file specification. If you do not specify a device in the output file specification when you use the /-SP qualifier, CDA creates the output list file on SY0:.

Default: /SP

/STB (File Specified Contains the Executive Symbol Table)

File: Symbol file (RSX11M.STB)

Effect: The /STB switch identifies a file containing the Executive symbol table. This file must correspond to the crashed system. CDA opens the symbol file and extracts the necessary symbol values. If it fails to find any required symbol values, CDA aborts the analysis and returns an error message.

Default: [current UIC]RSX11M.STB

2.1.2 CDA Command Line Examples

The following examples illustrate CDA command lines. Assume that the user in these examples is logged in under UIC [301,356], that the crash dump device is DR5:, and that CDA is running as an installed task. Also, note how CDA uses default file types.

Example 1

```
>CDA (RET)
CDA>DUMP,DUMP=RSX11M.STB/STB,DR5: (RET)
```

This command line creates:

- A list file, DUMP.LST, in UFD [301,356], which is printed automatically
- A binary file, DUMP.CDA, in UFD [301,356]

CDA reads the binary crash dump input from the crash dump device (DR5:), makes a binary copy of the crash dump input named DUMP.CDA, analyzes the crash dump input according to the information in the Executive symbol table file named RSX11M.STB in UFD [301,356], and writes a formatted output listing to a file named DUMP.LST. CDA then spools DUMP.LST to the system line printer queue.

Example 2

```
>CDA (RET)
CDA>,DUMP=[1,54]/STB,DR5: (RET)
```

This command line creates a binary file named DUMP.CDA in UFD [301,356].

CDA reads the binary crash dump input from DR5: and analyzes it according to the information in the Executive symbol table file, which is named RSX11M.STB in UFD [1,54].

Example 3

```
>CDA LP:=[1,54]/STB,DUMP (RET)
>
```

This command line creates an output listing on device LP:.

CDA reads the binary input from a previously created binary file named DUMP.CDA, and analyzes it in accordance with the information contained in the Executive symbol table file named RSX11M.STB in UFD [1,54]. The CDA output listings are then printed on LP:.

This command line is also an example of a CDA command that is issued from the CLI prompt. Thus, the CLI prompt returns after the command is issued.

Example 4

```
>CDA TI:=DUMP (RET)
>
```

This command line creates an output listing that is displayed on the terminal from which the command was issued.

CDA reads the binary input from a previously created binary file named DUMP.CDA and analyzes it according to the information in the default symbol table file, (the file named RSX11M.STB in the UFD that currently corresponds to UIC [301,356]). The CDA output listings are then displayed on TI:.

2.2 THE DCL ANALYZE/CRASH_DUMP COMMAND

If your terminal supports the DIGITAL Command Language (DCL) command line interpreter, you can run the CDA utility by using the DCL ANALYZE/CRASH_DUMP command as an alternative to the CDA command line. This section describes the ANALYZE/CRASH_DUMP command line format and qualifiers. The section concludes with some examples of ANALYZE/CRASH_DUMP command lines.

The ANALYZE/CRASH_DUMP command line has the following format:

```
ANALYZE/CRASH_DUMP[/qualifiers] crash-input[/qualifiers]
```

You use the ANALYZE/CRASH_DUMP command to specify CDA input and output. The command qualifiers that you place immediately after the command name specify the CDA output files and, optionally, the symbol table file that CDA uses to process the crash dump input. The crash-input specification is mandatory because it directs CDA to the source of the binary crash dump input.

Output Specifications:

You must specify at least one of the following command qualifiers as an output specification in the command line:

- /LIST: Specifies the output list file
- /BINARY: Specifies a binary copy of the crash-input file
- /SYMBOLS: Specifies the symbol definition file

You can specify /LIST: only, /BINARY: only, or /LIST: and /BINARY: together. You can optionally specify /SYMBOLS: with any combination of the /LIST: and /BINARY: qualifiers. However, if you do specify /SYMBOLS, you must include at least one of the other command qualifiers (because the symbol definition file is not an output file; it is used by CDA to generate an output file). Section 2.2.1.1. provides complete descriptions of the functions of each of the command qualifiers.

If you omit the crash-input specification from the command line, CDA prompts you for it, as shown in the following example:

```
$ ANALYZE/CRASH_DUMP/LIST:LP: (RET)
Crash input? DR5: (RET)
```

If you enter the command name only, CDA prompts you for input and output, as shown in the following example:

```
$ ANALYZE/CRASH_DUMP (RET)
Crash output? /LIST:SY:[301,356]CRASH.LST:/BINARY:COPY.CDA (RET)
Crash input? DUMP.CDA (RET)
```

Note that if you enter an output file in this way, you must include the /LIST: or /BINARY: qualifiers as part of the output file specification.

Input Specification:

crash-input

Specifies the location of the binary input to the ANALYZE/CRASH_DUMP command. The crash-input specification can be the name of the crash dump device, or it can be a binary file that was created during a previous crash dump analysis.

When you enter an ANALYZE/CRASH_DUMP command line, you can include command qualifiers, qualifiers for the crash-input parameter, or both. Section 2.2.1 describes qualifiers.

2.2.1 ANALYZE/CRASH_DUMP Command Qualifiers

You can control the way CDA processes the crash input and how it formats the output listings by using command qualifiers in the command line. You can select the information that you want in the CDA output listings by using qualifiers for the crash-input specification. Section 2.2.1.1 describes command qualifiers. Section 2.2.1.2 describes the qualifiers that you can use when you specify the crash input.

2.2.1.1 **Command Qualifiers** - You can use command qualifiers with the ANALYZE/CRASH_DUMP command to control how CDA processes the binary crash-input, and how it formats the output analysis listings. You place command qualifiers immediately after the command name in the command line. Table 2-4 summarizes the command qualifiers and gives brief descriptions of their effects. Expanded descriptions of each qualifier follow the table.

Table 2-4
Summary of ANALYZE/CRASH_DUMP Command Qualifiers

Command Qualifier	Function	Applies to File
/LIST:listfile[/qualifiers]	Specifies the output list file or device	List file
listfile qualifiers:		
/ERROR_LIMIT	Specifies an error limit at which CDA analysis terminates	List file
/PAGE_COUNT:n	Specifies the number of output pages	List file
/PAGE_LENGTH:n	Specifies the number of output lines per page	List file
/[NO]PRINTER	Specifies whether the output should be printed on the system line printer	List file
/BINARY:binaryfile[/qual]	Specifies an optional copy of the binary input file	Crash-input
binaryfile qualifier:		
/MEMORY_SIZE:n	Copies nKb words of memory from a crashed system	Crash-input
/SYMBOLS:symbolfile	Specifies the symbol definition file	Crash-input

Command Qualifier Descriptions:

/LIST:listfile[/qualifiers]
/ERROR_LIMIT[:n]
/PAGE_COUNT:n
/PAGE_LENGTH:n
/[NO]PRINTER

File: List file

Effect: Specifies the optional formatted CDA output list file. This list file consists of the analysis report listings that are described in Chapter 3. You can also specify a device for the list file, in which case CDA displays or prints its output listings on the specified device. You can control the list file output by using the following file qualifiers.

List File Qualifiers:

/ERROR_LIMIT[:n]

Effect: CDA maintains an error count. As it encounters inconsistencies in the system data structures, it increments the error count. CDA terminates the crash dump analysis when it finds the number of errors that you specify with this qualifier. If you use the /ERROR_LIMIT qualifier without specifying a number, the crash dump analysis terminates after one error.

Default: CDA runs the analysis until it is completed.

/PAGE_COUNT:n

Effect: This qualifier limits the number of pages of analysis output. When CDA has generated n pages, it terminates the analysis and prints a message on the terminal indicating that the analysis has terminated.

Default: Analysis terminates after 300 pages.

/PAGE_LENGTH:n

Effect: This qualifier lets you specify the number of lines that you want CDA to print per output page. After the specified number of lines are printed, CDA breaks to a new page.

Default: CDA prints 60 lines per page.

/[NO]PRINTER

Effect: This qualifier prevents the printing of the analysis output on the system line printer. Instead, CDA creates the output list file on the device in the list file specification. If a device is not specified in the list file specification, CDA creates the output file on the default user disk (SY0:).

Default: CDA prints all output on the system line printer.

/BINARY:binaryfile[/qualifier]
/MEMORY_SIZE:n

File: Crash-input

Effect: Specifies that an optional binary file should be created. This file is a copy of the binary data that the crash dump routine wrote on the crash dump device. If you create the file during an initial analysis, you can use it as input to the ANALYZE/CRASH_DUMP command at a later time. Also, because the crash dump routine overwrites the contents of the crash dump volume with each crash dump, this qualifier allows you to save the results of crash dumps. You can then reuse the same volume for successive crash dumps while maintaining a record of previous crash dumps.

Binary File Qualifier:

/MEMORY_SIZE:n

Effect: Specifies memory size for the binary copy of the crash dump input file. You specify n, where n is the number of Kb words. CDA then creates a binary file 4n words long and transfers nKb words to it from the crash-input file. The value of n must be greater than 16(decimal).

Default: n=124

/SYMBOLS:symbolfile

File: Symbol definition file

Effect: Specifies the symbol definition file for the crashed system, which contains the Executive symbol table. The symbol file must correspond to the crashed system. CDA opens the file and extracts the necessary symbol values. If it fails to find any required symbol values, CDA aborts the analysis and returns an error message. If you omit this file specification, CDA uses the default file, which is the file named RSX11M.STB in the UFD that corresponds to the current UIC.

2.2.1.2 **Crash-input File Qualifiers** - You can select the analysis listings that you want CDA to output by using qualifiers for the crash-input file specification in the ANALYZE/CRASH_DUMP command line. Table 2-5 summarizes the crash-input qualifiers and gives brief descriptions of their effects. Expanded descriptions of each qualifier follow the table.

Table 2-5
Summary of ANALYZE/CRASH_DUMP Crash-input Qualifiers

Qualifier or Argument	Function	Applies to File
/ACTIVE:(arg[,...])	Lists data on active tasks and/or devices	Crash-input
/ACTIVE arguments:		
DEVICES	Lists data about active devices	Crash-input
TASKS	Lists contents of the Task Control Blocks for active tasks	Crash-input
/ALL	Lists all available crash dump data	Crash-input
/BLOCK:n	Specifies the block number where crash dump begins on the crash dump device	Crash-input

(Continued on next page)

Table 2-5 (Cont.)
Summary of ANALYZE/CRASH_DUMP Crash-input Qualifiers

Qualifier or Argument	Function	Applies to File
/CLOCK_QUEUE	Lists the contents of the clock queue	Crash-input
/CONTROLLERS	Lists device controller data	Crash-input
/DATA_STRUCTURES:(arg[,...])	Specifies which data structures are to be formatted and listed	Crash-input
/DATA_STRUCTURES arguments:		
COMMAND_PARSER	Lists contents of CLI Parser Blocks	Crash-input
DEVICE STATUS UNIT	List contents of the control blocks for active devices	Crash-input
PARTITION	Lists contents of Partition Control Blocks	Crash-input
TASK	Lists contents of the Task Control Blocks for tasks in the STD	Crash-input
/DENSITY:n	Specifies bits per inch for input device	Crash-input
/DEVICES	Lists contents of all Device Control Blocks	Crash-input
/DUMP[:(START:n,END:n,ADDRESS:n)]	Lists contents of physical addresses	Crash-input
/HEADERS	Lists contents of resident task headers	Crash-input
/KERNEL:(arg[,...])	Lists kernel contents	Crash-input
/KERNEL arguments:		
DATA:(START:n,END:n)	Lists contents of kernel data space	Crash-input
INSTRUCTION:(START:n,END:n)	Lists contents of kernel instruction space	Crash-input
REGISTERS	Forces assignment of values for the kernel address registers	Crash-input

(Continued on next page)

Table 2-5 (Cont.)
Summary of ANALYZE/CRASH_DUMP Crash-input Qualifiers

Qualifier or Argument	Function	Applies to File
/PARTITION	Lists contents of Partition Control Blocks	Crash-input
/POOL:(START:n,END:n)	Lists pool contents	Crash-input
/SECONDARY_POOL[:(START:n,END:n)]	Lists contents of secondary pool from START to END	Crash-input
/[NO]SYSTEM	Suppresses listing of system information	Crash-input
/TASKS:(arg[,...])	Lists task data	Crash-input
/TASKS arguments:		
DIRECTORY	Lists contents of the Task Control Blocks for tasks in the STD	Crash-input
ADDRESS:(NAME:name[,START:n,END:n])	Lists contents of task addresses from START to END	Crash-input
DATA:(NAME:name[,START:n,END:n])	Lists contents of task data space	Crash-input
INSTRUCTION:(NAME:[,START:n,END:n])	Lists contents of task instruction space	Crash-input

File Qualifier Descriptions:

/ACTIVE:(arg[,...])
DEVICES
TASKS

File: Crash-input

Effect: Lists data on active tasks and devices.

/ACTIVE arguments:

DEVICES

Effect: Lists data on the devices active in the system at the time of the crash. If you want CDA to list data on all of the devices known to the system at the time of the crash, use the /DEVICES qualifier.

TASKS

Effect: Lists the contents of the Task Control Blocks of active tasks. If you want CDA to list the contents of the Task Control Blocks of all installed tasks, both active and dormant, use the /TASKS:(DIRECTORY) qualifier.

/ALL

File: Crash-input

Effect: Analyzes all information available in the crash dump file (except the information associated with memory and task dumps). CDA lists the output in the following order:

1. System information
2. Active tasks information
3. Task headers information
4. Partition information
5. Common Block Directory entries
6. Device information
7. Clock queue contents
8. Device controller information
9. Pool contents

/BLOCK:n

File: Crash-input

Effect: Identifies the starting block number of the crash dump file on the crash input device. The value of n must be less than 65535(decimal).

/CLOCK_QUEUE

File: Crash-input

Effect: Lists the contents of the system clock queue.

/CONTROLLERS

File: Crash-input

Effect: Lists the contents of the controller table and Controller Request Block (KRB) for each device controller in the system.

/DATA_STRUCTURES:(arg[,...])

```
COMMANDPARSER
DEVICE
PARTITION
STATUS
TASK
UNIT
```

File: Crash-input

Effect: Selects which system data structures CDA will format and list.

/DATA_STRUCTURES arguments:

COMMAND_PARSER

Effect: Lists the contents of the Command Line Interpreter (CLI) Parser Blocks.

PARTITION

Effect: Lists the contents of the Partition Control Blocks.

TASK

Effect: Lists the contents of the Task Control Block for every task in the System Task Directory (all installed tasks) at the time of the system crash.

DEVICE

Effect: Lists the contents of the Device Control Blocks for active devices.

STATUS

Effect: Lists the contents of the Status Control Blocks for active devices.

UNIT

Effect: Lists the contents of the Unit Control Blocks for active devices.

/DENSITY:n

File: Crash-input

Effect: Causes a crash input tape to be read at the density specified, 800 or 1600 bpi. The default is 800 bpi.

/DEVICES

File: Crash-input

Effect: Lists the contents of the control blocks for all devices in the system. To list only active devices, use the /ACTIVE:(DEVICES) qualifier.

/DUMP[:(START:a,END:b[,ADDRESS:c])]

File: Crash-input

Effect: Lists the contents of physical addresses a through b inclusive and labels them with their physical addresses. If you include address c, the /DUMP qualifier dumps the contents of physical addresses a through b, but labels them with dummy virtual addresses, starting at c.

/HEADERS

File: Crash-input

Effect: Lists the contents of the task headers for each task resident in memory.

/KERNEL:(arg[,...])
DATA:(START:n,END:n)
INSTRUCTION:(START:n,END:n)
REGISTERS

File: Crash-input

Effect: Lists kernel data.

/KERNEL arguments:

DATA:(START:n,END:n)

Effect: Lists the contents of kernel data space from virtual addresses START:n to END:n.

INSTRUCTION:(START:n,END:n)

Effect: Lists the contents of kernel instruction space from virtual address START:n to END:n.

REGISTERS

Effect: Forces the assignment of the kernel address register values for the crashed system.

/PARTITION

File: Crash-input

Effect: Lists the contents of the Partition Control Blocks.

/POOL:(START:n,END:n)

File: Crash-input

Effect: Lists the contents of system pool between the addresses specified in octal, Radix-50, and ASCII.

/SECONDARY_POOL:(START:n,END:n)]

File: Crash-input

Effect: Lists the contents of system secondary pool between the addresses specified by START and END.

/[NO]SYSTEM

File: Crash-input

Effect: The /NOSYSTEM qualifier suppresses the system information listing. The default action of CDA is /SYSTEM; that is, it lists the system information.

/TASKS:(arg[,...])
DIRECTORY
ADDRESS:(NAME:name,START:n,END:n)
DATA:(NAME:name[,START:n,END:n])
INSTRUCTION:(NAME:name[,START:n,END:n])

File: Crash-input

Effect: Lists task data.

/TASKS arguments:

DIRECTORY

Effect: Lists the contents of the Task Control Block for every task in the System Task Directory (all installed tasks) at the time of the system crash.

ADDRESS:(NAME:name,START:n,END:n)

Effect: Lists the contents of the task specified by NAME between the virtual addresses specified by START and END. Includes the contents of task data space if a task includes data space.

DATA:(NAME:name[,START:n,END:n])

Effect: RSX-11M-PLUS operating systems only. Lists the contents of task data space for the task specified by NAME.

INSTRUCTION:(NAME:name[,START:n,END:n])

Effect: RSX-11M-PLUS operating systems only. Lists the contents of task instruction space for the task specified by NAME.

2.2.2 ANALYZE/CRASH_DUMP Command Examples

The following examples illustrate the ANALYZE/CRASH_DUMP command. Assume that the user in these examples is logged in under UIC [301,356], and that the crash dump device is DR5:. In this way, you can note how CDA uses default file types. Also, assume that CDA is running as an installed task.

Example 1

```
$ ANALYZE/CRASH_DUMP/LIST:CRASH/BINARY:COPY/MEMORYSIZE:250 DR5: RET
```

This command creates:

- An output list file named CRASH.LST in the current UFD for UIC [301,356].
- A binary copy of 250kb words of the crash dump from DR5: (the crash dump device). The copy is named COPY.CDA and is placed in the current UFD for UIC [301,56].

CDA reads the binary crash dump input from the crash dump device and analyzes it according to the default symbol definition file, since a symbol definition file is not specified in the command line. CDA uses the file named RSX11M.STB in the current UIC as the symbol definition file. CDA then generates a list file named CRASH.LST and spools it to the default system line printer queue. CDA also copies the specified amount of memory from the crash dump device to a binary file named COPY.CDA.

Example 2

```
$ ANALYZE/CRASH_DUMP/LIST:LP5:/PAGE_COUNT:5 DR5:/BL:100 RET
```

This command creates a list file that is printed on LP5:.

CDA reads the crash input from DR5:, beginning at block 100, and analyzes it according to the default symbol definition file. CDA then prints the first five pages of its output listing on LP5:.

Example 3

```
$ ANALYZE/CRASH_DUMP/LIST:TI:/SYMBOLS:[1,54] COPY.CDA (RET)
```

This command creates a list file that is displayed on TI: (the terminal at which the command was issued).

CDA reads the previously generated binary file named COPY.CDA, analyzes it according to the file named RSX11M.STB in UFD [1,54], and displays its output listings on TI:.

CHAPTER 3

ANALYSIS LISTINGS

The CDA output listings in this chapter illustrate CDA operation. Each item of each listing is keyed to the brief explanatory text that precedes it.

Dumps shown in offset mode use relative addresses. They are offset from the beginning of the displayed data. They are neither physical nor virtual addresses of the data.

NOTE

These listings came from several different crash dumps. Therefore, values that would usually correlate across the various listings do not necessarily correlate here. Those listings that extend across several pages in an actual dump of a crashed system are truncated here and reflect only a typical printout format for them.

3.1 SYSTEM INFORMATION

The first six pages of a CDA output listing normally contain the system information described in Sections 3.1.1 through 3.1.6. The system information consists of the following:

- Volatile registers
- Kernel stack
- System common
- System common alphabetized dump
- Pool statistics
- Assign table

If Group-global Event Flag Blocks are in memory when the system crashes, the listing described in Section 3.1.7 appears. If error log packets are in memory at the time of the crash, the listing described in Section 3.1.8 appears. On RSX-11M-PLUS systems, CDA generates the listing of low core memory shown in Section 3.1.9 as part of the system common dump.

3.1.1 Volatile Registers

Figure 3-1 is a listing that reflects the state of the hardware registers at the time of the crash. Refer to the appropriate PDP-11 processor handbook for detailed information on these registers. Each item in the following list describes a correspondingly numbered item in Figure 3-1.

Item	Description
1.	Contents of Processor Status Word (PSW) and kernel and user stack pointers after crash
2.	Program counter and PSW (that the system pushed onto the kernel stack) just prior to system crash (These values are valid only if the system trapped.)
3.	Contents of general registers
4.	Contents of memory management registers
5.	Contents of Page Address and Page Description Registers (See Section 4.1.1 for information on how to interpret this information.)
6.	Contents of UNIBUS map registers (This field is suppressed if the processor does not have a UNIBUS map.)
7.	Contents of CPU error register that identifies the source of the abort or trap that used the vector at location 4 (on RSX-11M-PLUS systems, this field is suppressed if the processor does not have a UNIBUS map.)
8.	Contents of memory system error register (On RSX-11M-PLUS systems, this field is suppressed if the processor does not have a UNIBUS map.)
9.	Contents of cache control register (On RSX-11M-PLUS systems, this field is suppressed if the processor does not have a UNIBUS map.)

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85   10:58       PAGE 1
VOLATILE REGISTERS

AFTER CRASH:  PS=000000   SP(K)=000616   SP(U)=001674 (1)

BEFORE CRASH:  PC=000000   PS=120476 (2)

R0=007760   R1=007377   R2=007530   R3=000000   R4=000001   R5=000000 (3)

MMR0=000000   MMR1=000000   MMR2=002256   MMR3=000000 (4)

                U S E R                                  U N I B U S   M A P

   I   S P A C E          D   S P A C E                    1  00000000
   PDR       PAR          PDR       PAR                   2  00020000
  077506    000000       000000    000000                 3  00040000
  077406    000200       000000    000000                 4  00060000
  077506    000400       000000    000000                 5  00100000
  077406    000600       000000    000000                 6  00120000
  077406    001000       000000    000000                 7  00140000
  037506    001600       000000    000000                 8  00327024
  000000    003123       000000    000000                 9  01132134
  077406    177600       000000    000000                10  01152134
                                                          11  01157134
           S U P E R V I S O R                            12  01200134
                                                          13  01674220
   I   S P A C E          D   S P A C E                   14  00430770
   PDR       PAR          PDR       PAR                   15  01636100 (6)
  000000    000000       000000    000000                 16  17416700
  000000    000000       000000    000000                 17  17416700
  000000    000000       000000    000000 (5)             18  17416700
  000000    000000       000000    000000                 19  17416700
  000000    000000       000000    000000                 20  17416700
  000000    000000       000000    000000                 21  17416700
  000000    000000       000000    000000                 22  17416700
  000000    000000       000000    000000                 23  17416700
  000000    000000       000000    000000                 24  17416700
                                                          25  17416700
                K E R N E L                               26  17416700
                                                          27  17416700
   I   S P A C E          D   S P A C E                   28  17416700
   PDR       PAR          PDR       PAR                   29  17416700
  077506    000000       000000    000000                 30  17416700
  077506    000200       000000    000000                 31  17416700
  077506    000400       000000    000000
  077506    000600       000000    000000
  077506    001000       000000    000000
  077406    001600       000000    000000
  077406    003123       000000    000000
  077506    177600       000000    000000
```

Figure 3-1 Volatile Registers

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0     19-APR-85  14:03        PAGE 1
VOLATILE REGISTERS

AFTER CRASH:  PS=000344   SP(K)=000604   SP(S)=001212   SP(U)=120362
                                (7)              (8)              (9)
              CPU ERR = 000100   MEM SYS ERR = 000000   CACHE CTL REG = 000001

BEFORE CRASH:  PC=045210   PS=030005

R0=000401   R1=053550   R2=000000   R3=000010   R4=000102   R5=120526

MMR0=000001   MMR1=000000   MMR2=011710   MMR3=000066

                U S E R                                U N I B U S   M A P

  I   S P A C E          D   S P A C E                    1  00000000
  PDR       PAR          PDR       PAR                    2  00074400
 015006    000000        000000    000000                 3  00114400
 077406    000744        000000    000000                 4  00134400
 077406    001144        000000    000000                 5  00154400
 077406    001344        000000    000000                 6  00243400
 077406    001544        000000    000000                 7  00263400
 037506    001744        000000    000000                 8  00461124
 000000    027255        000000    000000                 9  02642064
 077406    177600        000000    000000                10  02662064
                                                          11  02702064
                                                          12  02722064
          S U P E R V I S O R                             13  02742064
                                                          14  02762064
  I   S P A C E          D   S P A C E                   15  15350622
  PDR       PAR          PDR       PAR                   16  17776366
 077402    006210        077406    020347                 17  07720376
 004402    006410        014406    020547                 18  13340176
 000000    000000        000000    022713                 19  11420312
 000000    000000        000000    023113                 20  14020346
 000000    000000        000000    023313                 21  12100272
 000000    000000        000000    016267                 22  03140176
 000000    000000        074406    004750                 23  04410272
 000000    000000        000000    005142                 24  00020200
                                                          25  06335600
                                                          26  04637510
                K E R N E L                               27  17567456
                                                          28  17357734
  I   S P A C E          D   S P A C E                   29  10217500
  PDR       PAR          PDR       PAR                   30  00217404
 077506    000000        077506    000000                 31  11357560
 077406    000200        077506    000744
 077406    000400        077506    001144
 077406    000600        077506    001344
 077406    001000        077506    001544
 077406    001744        077406    001744
 077406    027255        077406    027255
 077406    177600        077506    177600
```

Figure 3-1 (Cont.) Volatile Registers

3.1.2 Kernel Stack

Figure 3-2 shows the contents of the kernel stack area beginning at V$$CTR and ending at $STACK. The kernel stack pointer points to a location within this area. See Section 4.1.2 for information on interpreting the contents of the kernel stack.

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85    15:13        PAGE 2
KERNEL STACK

  KERNEL STACK:

    000400    000000  000000  000000  000000  000000  000000  000000  000000
    000420    000000  000000  000000  000000  000000  000000  000000  000000
    000440    000000  000000  000000  000000  000000  000000  000000  000000
    000460    000000  000000  000000  000000  000000  000000  000000  000000
    000500    000000  000000  000000  000000  000000  000000  000000  000000
    000520    000000  000000  000000  000000  000000  000000  000000  000000
    000540    000000  000000  000000  000000  000000  000000  000000  000000
    000560    000000  000000  000000  123064  177613  120204  000251  000251
    000600    123224  116506  000000  122710  000014  120344  123064  001446
    000620    120204  123064  123064  001446  120204  161121  006066  110160
    000640    114514  025160  000000  023540  006066  133362  126570  000000
    000660    140672  130054  160020  136744  122026  000000  137062  160020
    000700    130110  130054  006066  105664  000000  120644  022402  011762
    000720    030011  000700  007736  121000  003306  120212  025616  177777
    000740    106036  000000  106004  120220  170000
```

Figure 3-2 Kernel Stack

3.1.3 System Common

The listing in Figure 3-3 provides a selective interpretation of some of the items in system common. Each item in the following describes the corresponding numbered item in Figure 3-3. (Refer to the RSX-11M Guide to Writing an I/O Driver or the RSX-11M-PLUS Guide to Writing an I/O Driver for further information.)

Item	Description
1.	Time and date of crash, as set in the system
2.	The task that was running at the time of the crash (If no task was running, this field contains the null task. This condition could develop if all the active tasks are blocked at the time of the crash. For information on determining which task or driver was mapped at the time of the crash, see Section 4.1.1.)
3.	The address of the Task Control Block (TCB) of the current task
4.	The contents of the 4-byte system ID indicating system base level
5.	The first address available for partitions (the last address of the Executive plus 1)
6.	The system size in 32-word blocks and in total words
7.	System UIC
8.	Stack depth count
9.	Contents of the global event flag words
10.	Name of the system for which dump is generated
11.	Network UIC
12.	Device from which the system was booted
13.	Logical block number (LBN) of the beginning of the system image
14.	Size of system image file in blocks
15.	The octal value of the system feature masks and the meaning of each set bit
16.	Octal dump of system common in offset mode in numerical order by address

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85   15:13       PAGE 3
SYSTEM COMMON


CRASH OCCURRED AT 09:53:51  21-MAY-81 (1)

          (2)                              (3)
CURRENT TASK = LDR...     TCB ADDRESS = 112050
     (4)              (5)              (6)                (7)
$SYSID = 30E     $EXSIZ = 115000     $SYSIZ = 16384./512K     $SYUIC = [2,54]
     (8)                        (9)
$STKDP = 000000      $COMEF: <33-48> 000000  <49-64> 000000
      (10)                    (11)
SYSTEM NAME = QUASAR      $NTUIC = [102,54]
      (12)                    (13)                (14)
LOAD DEVICE = DB0      LBN = 00124461      FILE SIZE = 496.

SYSTEM FEATURE MASK (FIRST WORD) = 033377

      BIT SET              MEANING
      -------              -------
        EXT          22-BIT EXTENDED MEMORY SUPPORT
        MUP          MULTI-USER PROTECTION SUPPORT
        EXV          20K EXEC SUPPORTED
        DRV          LOADABLE DRIVER SUPPORT
        PLA          PLAS SUPPORT
        CAL          DYNAMIC CHECKPOINT SPACE ALLOCATION
        PKT          PREALLOCATION OF I/O PACKETS
        EXP          EXTEND TASK DIRECTIVE SUPPORTED
        OFF          PARENT/OFFSPRING TASKING SUPPORTED
        FDT          FULL DUPLEX TERMINAL DRIVER
        DYM          DYNAMIC MEMORY ALLOCATION SUPPORTED
        CEX          COMMUNICATIONS EXEC IS LOADED


SYSTEM FEATURE MASK (SECOND WORD) = 167400

      BIT SET              MEANING
      -------              -------
        DPR          DIRECTIVE PARTITION SUPPORT
        IRR          INSTALL, REQUEST, AND REMOVE TASK SUPPORT
        GGF          GROUP GLOBAL EVENT FLAG SUPPORT
        RAS          RECEIVE/SEND DATA PACKET SUPPORT
        RBN          ROUND ROBIN SCHEDULING SUPPORTED
        SWP          EXECUTIVE LEVEL DISK SWAPPING SUPPORTED
        STP          EVENT FLAG MASK IS IN THE TCB


SYSTEM FEATURE MASK (THIRD WORD) = 025215

      BIT SET              MEANING
      -------              -------
        CLI          MULTIPLE CLI SUPPORT
        EIS          SYSTEM REQUIRES THE EXTENDED INSTRUCTION SET
        CRA          SYSTEM SPONTANEOUSLY CRASHED (1=YES)
        STM          SYSTEM HAS SET SYSTEM TIME DIRECTIVE
        AST          SYSTEM HAS AST SUPPORT
```

(15)

Figure 3-3 System Common

```
RSX-11M CRASH DUMP ANALYZER      V4.2      19-APR-85   15:13        PAGE 4
SYSTEM COMMON DUMP

                                    (16)
ADDR    LABEL   VALUE          ADDR    LABEL   VALUE          ADDR    LABEL   VALUE

007660          000760         010020  $POLST  000403         010160          000000
                010316                 $PRIHL  003100                         000000
                000207                 $PRILL  001130                         000000
        $HEADR  111700                 $PFRSZ  000310                         000000
                174000                 $POLBP  000063                         000000
        $COMEF  000000                 $POLFL  000200                         000000
                000000                 $POLLW  000144                         000000
        $SYSID  030063                 $PARPT  017226                         000000

007700          020105         010040  $CLKHD  044414         010200          000000
        $TKNPT  107134                 $COPT   037356                         000000
        $SHFPT  106704                 $PARHD  114734                         000000
        $CKCNT  177546                 $LDRPT  112050                         000000
        $CKCSR  177546                 $TSKHD  112050                         000000
        $CKLDC  000000                 $XCOM1  001600                         000000
        $SYUIC  001054                 $XCOM2  001746                         000000
                000000                 $GGEF   000000                         000000

007720  $EXSIZ  115000         010060  $GFTCB  010064         010220          000000
        $PWRFL  000000                 $GEFPT  010064                 $ERBAF  051423
        $SIGFL  000000                 $GEFDM  000356                         030131
        $LOGHD  052254                 $IDLCT  000000                         055472
        $MCRCB  104360                 $IDLPT  103741                         026061
        $LSTLK  103640                 $DYPMN  020035                         056466
                000003                         020037                         040502
        $CRAVL  045074                         020037                         045503

007740          000000         010100          017440         010240          050125
        $ACTHD  112050                         017440                         042456
        $DICSV  001051                         020040                         051122
        $TKTCB  112050                 $BTMSK  000001                         000000
        $LBUIC  000454                         000002                         000000
        $ABTIM  007572                         000004                         000000
        $RQSCH  000000                         000010                         000000
        $STKDP  000000                         000020                         000000

007760  $DEVHD  063460         010120          000040         010260          000000
        $RNDCT  000005                         000100                         000000
        $SWPCT  000036                         000200                         000000
        $ERRPT  000000                         000400                 $ERFID  000000
        $CFLPT  046444                         001000                         000000
                000000                         002000                         000000
                026222                         004000                         000000
        $INTCT  177777                         010000                         000000

010000  $FRKHD  000000         010140          020000         010300  $PRMOD  000000
                010000                         040000                 $SYSIZ  040000
        $FMASK  033377                         100000                         000000
                167400                 $ERHEA  000000                         124461
                025215                         010146                         041104
        $HFMSK  000003                 $ENTSQ  000001                         000760
        $PTTCB  053510                 $ERRSQ  000000                         177777
        $PRISZ  003370                 $ERFLA  000000                         000015
```

Figure 3-3 (Cont.) System Common

3.1.4 System Common Alphabetized Dump

The listing in Figure 3-4 represents an alphabetical list of the locations in system common that have a label associated with them. The octal numbers represent the contents of those locations, not the addresses of the labels. The following summary lists the labels and their meanings. Note that some of these labels may not appear on your listing, or that additional labels may appear, depending upon the options you selected at system generation.

$ABTIM	Absolute time counter
$ACCLK	Absolute time clock for accounting
$ACNFE	Accounting feature mask word
$ACTHD	Active task listhead
$ACTPS	Clock rate for accounting
$APLIM	Free secondary pool space ACNT reserves
$AVRHD	Automatic volume recognition listhead
$BTTIM	Absolute time when system was booted
$CBDHD	Common block directory listhead
$CFLPT	Checkpoint file PCB listhead
$CKCNT	Address of clock count register
$CKCSR	Clock control status register (CSR)
$CKLDC	Clock load count
$CKUAB	User Account Block (UAB) for task run from clock queue
$CKURM	UNIBUS Run Mask (URM) of processor that keeps the clock
$CLICQ	Command queue listhead
$CLKHD	Clock queue listhead
$COPT	Pointer to Console Output (CO0:) Unit Control Block (UCB)
$CPMSK	Processor bit clear mask
$CPPAR	Pointer to partition for CPU local memory
$CPTBL	Pointer to Command Line Interpreter Parser Block (CPB) Table
$CRAVL	Free system pool listhead
$CRCSR	Crash device CSR address with no Controller Request Block (KRB)
$CRFLG	Flag indicating saved registers
$CRFPR	Number of first processor to crash
$CRKRB	Crash dump device KRB address

```
$CRLCK    One CPU dumps memory
$CRSUN    Crash physical unit number
$CTLST    Start of the Controller Table (CTB) list
$CURPR    Current task priority
$CXDBL    Context switching disabled flag
$DEVHD    Pointer to first Device Control Block (DCB)
$DICSV    Temporary storage for directive services
$DRAPR    APR value to map directive partition
$DVSAV    Saved CSR contents for error logging
$ENTSQ    Error log entry sequence number
$ERFLA    Error Logger flag word
$ERHEA    Error Log message queue listhead
$ERRPT    Pointer to Error Logger Task Control Block
$ERRSQ    Universal error sequence number
$EVBSQ    Buffer sequence number
$EVDIS    Buffer position for next event
$EVKS6    KISAR6 offset to buffer
$EVLEN    Pointer to word beyond end of buffer
$EVLOS    Number of events lost through saturation
$EVSEQ    Event sequence number
$EVTCB    TCB address of event logger task
$EXCRC    Executive read-only code cyclic redundancy check (CRC)
$EXECL    Serialize access to executive data lock
$EXSIZ    Executive size
$FMASK    System feature mask
$FORKL    Serialize access to fork list lock
$FRKHD    Fork queue listhead
$FXRPT    Pointer to parity error task
$GEFDM    Group-global dummy mask address word
$GEFPT    Group-global mask address pointer
$GFTCB    Group-global user TCB pointer
$GGEF     Group-global event flags listhead
$GNLST    General use pool packet listhead
```

$HEADR	Pointer to current task header
$HFMSK	Hardware system feature mask
$HRCPT	Pointer to HRC... task (privileged task for reconfiguration services)
$ICAVL	ICB pool; same as core pool if no data space on system
$IDLCT	Idle pattern count byte
$IDLFL	Idle pattern flag in bytes
$IDLPT	Idle pattern word
$IICPU	Mask of interrupted URMs
$IIFNL	Serialize access to $MPTAB lock
$IIMSK	IIst interrupt mask word
$IINXT	Round robin word for $IISVC
$IIPND	Pending URM work word
$INTCT	Clock interrupt ticks count
$LBUIC	Library UIC
$LDPCB	Current loader PCB pointer
$LDRPT	Pointer to loader TCB
$LOGHD	Logical device assignment listhead
$LSTLK	Lock word; TCB address of owner
$MCRPT	Pointer to MCR TCB
$MOULS	Mount listhead
$MXEXT	Last address in system common
$NCPU	Number of processors in system
$NTUIC	Network UIC
$PARHD	Pointer to partition list
$PASTH	Partition AST listhead
$PFRSZ	Minimum size of largest fragment in pool
$PFURM	URM to powerfail
$PKAVL	Pointer to first preallocated I/O packet
$PKMAX	Maximum number of preallocated I/O packets
$PKNUM	Number of preallocated I/O packets currently in list
$PLPAR	Pointer to secondary pool PCB
$POLBP	Minimum priority for nonprivileged task to execute at low pool

$POLFL	Executive pool usage control flags
$POLHD	Listhead for secondary pool free list
$POLST	Executive pool communications word
$PRIFR	Current amount of free pool
$PRIHL	Upper limit for pool monitoring
$PRILL	Lower limit for pool monitoring
$PRISZ	Minimum size of largest pool fragment
$PRMOD	Processor model number
$PRTAB	Processor current task priority table
$PTCBL	Prototype TCB listhead
$PTCPT	KISAR6 bias of prototype TCB
$PTTCB	TCB address of pool recovery task
$PWRFL	Power-fail recovery request flag
$PWRLK	Serialize access to $PWRMK lock
$PWRMK	Mask of CPU in power-fail code
$RNDC	Clock ticks for each scheduling interval
$RNDCT	Number of clock ticks until next scheduled interval
$RNDH	Highest priority class to consider
$RNDL	Lowest priority class to consider
$ROEND	End of read-only part of the Executive
$RQSCH	Schedule request TCB address
$RQTAB	Reschedule pointer to TCB table
$SABPT	Pointer to System Account Block
$SAHDB	Bias of current task header
$SAHPT	Virtual address of current task header
$SAVSP	Saved stack pointer
$SCCTB	CTB if $SCDEV contains KRB
$SCDEV	UCB or KRB for status change
$SCERR	Error return from driver
$SCMOF	Offset to data space
$SCOFL	On-line or off-line parameter
$SECFR	Number of free blocks in secondary pool
$SGFFR	Pointer into stack for $SGFIN

$SHERR	Points to TCB of shadow error task
$SHFCT	Minimum ticks between shuffler requests
$SHFPT	Pointer to Shuffler Task Control Block
$SHFTM	Time remaining before next possible request to Shuffler
$SHLIM	Error packet limit
$SHLOS	Number of packets lost from saturation
$SHPCT	Current shadow error count
$SHUMB	Root for UMB list
$SIGFL	Task waiting for significant event
$STALR	Sanity timer alarm enabled on CPU
$STENB	Sanity timer enabled
$STKDP	Stack depth indicator
$SWPC	Clock ticks for each swapping interval
$SWPCT	Number of clock ticks to next swapping interval
$SWPR	Swapping priority
$SWR	Multiprocessor console switch register
$SYLHD	Listhead for System log input queue
$SYSIZ	Size of memory in 32Kb-word blocks
$SYUAB	Address of UAB for system tasks
$SYUIC	System User Identification Code (UIC)
$TKNPT	Pointer to Task Termination Notification Program (TKTN) Task Control Block
$TKPS	Ticks per second
$TNAME	Multiuser task name
$TSKHD	Pointer to System Task Directory (STD)
$TTNS	Tick of second
$ULDPT	Microcode loader task TCB address
$UMRST	Unibus Run Mask (URM) status table
$VECTR	Highest vector address
$XDTFL	Executive Debugging Tool (XDT) initialization table
$XDTPR	Flag for prompts from XDT

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85     15:13       PAGE 6
SYSTEM COMMON ALPHABETIZED DUMP

$ABTIM 007572        $ERRPT 000000        $NETPF 000000        $SWPCT 000036
$ACTHD 112050        $ERRSQ 000000        $NTUIC 041054        $SYSIZ 040000
$AVRHD 000000        $EXSIZ 115000        $PARHD 114734        $SYSNM 052521
$CFLPT 046444        $FMASK 033377        $PARPT 017226        $SYUIC 001054
$CKCNT 177546        $FRKHD 000000        $PFRSZ 000310        $TEMP0 110160
$CKCSR 177546        $GEFDM 000356        $PKAVL 072230        $TEMP1 064170
$CKLDC 000000        $GEFPT 010064        $PKMAX     017       $TEMP2 013356
$CLICQ 000000        $GFTCB 010064        $PKNUM     012       $TEMP4 000000
$CLKHD 044414        $GGEF  000000        $POLBP 000063        $TKNPT 107134
$COPT  037356        $HEADR 111700        $POLFL 000200        $TKPS  000074
$CPTBL 010472        $HFMSK 000003        $POLLW 000144        $TKTCB 112050
$CRAVL 045074        $IDLCT     000       $POLST 000403        $TSKHD 112050
$CURPR     370       $IDLFL     000       $PRIHL 003100        $TTNS  000066
$CXDBL     000       $IDLPT 103741        $PRILL 001130        $UMRHD 052642
$DEVHD 063460        $INTCT 177777        $PRISZ 003370        $UMRPT 170200
$DICSV 001051        $LBUIC 000454        $PRMOD 000000        $UMRWT 000000
$DPM   000040        $LDRPT 112050        $PTTCB 053510        $WTCSR 017226
$ENTSQ 000001        $LOGHD 052254        $PWRFL 000000        $WTDUM 017226
$ERBAF     023       $LSTLK 103640        $RNDCT 000005        $XCOM1 001600
$ERFID 000000        $MCRCB 104360        $RQSCH 000000        $XCOM2 001746
$ERFLA     000       $MCRPT 110160        $SHFPT 106704
$ERHEA 000000        $MOULS 046614        $SSIGFL 000000
$ERLOF     000       $MXEXT 177777        $STKDP 000000
```

Figure 3-4 System Common Alphabetized Dump

3.1.5 Pool Statistics

The listing in Figure 3-5 contains information concerning the system pool. CDA derives Items 2, 3, and 4 by scanning the free block pointers of the pool. The minimum block size (that is, pool granularity) in Item 5 comes from the contents of $CRAVL-2. Each item in the following list describes a correspondingly numbered item in Figure 3-5.

Item	Description
1.	Pool size in decimal bytes
2.	The largest fragment of pool space
3.	Total number of free bytes in pool
4.	Number of fragments not allocated
5.	Smallest possible block (This is the minimum number of bytes which may be requested at a time. The minimum block size is always four bytes.)
6.	Bit map in octal

Each bit in the bit map represents one 4-byte block. If the bit is set, the block is free. The first block in the pool is bit 0 of the first octal word in the bit map. The bits are numbered as follows:

Bit Number	15	14 13 12	11 10 9	8 7 6	5 4 3	2 1 0
Binary	0	0 1 1	1 0 1	1 1 1	1 1 1	0 0 0
Octal	0	3	5	7	7	0

Any bits left over in the last word of the bit map are cleared.

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85     10:58        PAGE 5
POOL STATISTICS

POOL SIZE (BYTES) = 16532. (1)

LARGEST FREE BLOCK (BYTES) = 12416. (2)

TOTAL FREE BYTES = 12676. (3)

NUMBER OF FRAGMENTS = 10. (4)

MINIMUM BLOCK SIZE (BYTES) = 4. (5)


POOL BITMAP (CONSTRUCTED FROM LINKED POOL, BLOCK FREE IF BIT SET):      (6)

       000000  100434  000000  007000  000040  140000  002001  000000
       000000  000000  000000  000000  000004  000000  000000  000000
       176000  177777  177777  017777  000000  000000  177740  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       177777  177777  177777  177777  177777  177777  177777  177777
       000037  000000  000000  000000  000000  000000  000000  000000
       000000  000000  000000  000000  000000  000000  000000  000000
       000000  000000  000000  000000  000000  000000  000000  000000
       000000  000000  000000  000000  000000  000000  000000  000000
       000000  000000  000000  000000  000000  000000  000000  000000
       000000  000000  000000
```

Figure 3-5 Pool Statistics

3.1.6 Assign Table

Figure 3-6 is a listing of the logical device assignment table.

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0    16-APR-85   09:15
ASSIGN TABLE


System Logicals:

     SY = DR5:
     Block: 1      Status: (Final)

     EDTINI = SYS$LOGIN:EDTINI
     Block: 1



User Logicals:

   Terminal: TT5:
     SYS$LOGIN = DR5:[COVERT]
     Block: 2      Status: (Final, Privileged)

   Terminal: TT24:
     IN = DR5:
     Block: 1      Status: (Final)

     SYS$LOGIN = DR5:[007,325]
     Block: 2      Status: (Final, Privileged)

   Terminal: TT75:
     HOME = DR5:[7,40]
     Block: 1

     DEFCOR = DB3:[61,40]
     Block: 1

     DUMPS = LB:[4,54]
     Block: 1

     SYS$LOGIN = DR5:[7,40]
     Block: 2      Status: (Final, Privileged)
```

Figure 3-6 Logical Assignment Table

The Assign Table lists logical assignment table entries in two categories: system logicals and user logicals. The system logicals listing shows the logical name, the equivalence name, the number of blocks, and the status of an assignment. The user logicals listing also shows the terminal from which an assignment was made.

3.1.7 Group-Global Event Flags

Figure 3-7 shows a group-global event flag dump. If there are no group-global event flags, this dump does not appear.

Item	Description
1.	Group number
2.	Access count
3.	Group-global Event Flag Block dump (The last two words are the group-global event flags.)

```
RSX-11M-PLUS CRASH DUMP ANALYZER     V4.2     19-APR-85  10:17      PAGE 18
GROUP GLOBAL EVENT FLAGS


      (1)                          (2)
GROUP NUMBER =1           ACCESS COUNT =177476

GGEF  DUMP:       (3)

    000000    056660  000401  177476  000015  000000


GROUP NUMBER =7           ACCESS COUNT =006332

GGEF  DUMP:

    000000    020000  000007  006332  000014  000000
```

Figure 3-7 Group-global Event Flags

3.1.8 Error Log Packets

The listing shown in Figure 3-8 contains error logging information that resided in memory at the time of the crash. This page does not appear if no error log packets were in memory at the time of the crash. This data is not written to the Error Log file on disk.

Item	Description
1.	Address of error log buffer
2.	Error log packet entry type code
3.	Error log packet entry type subcode
4.	Time the packet was logged
5.	Dump of error log packet in octal

```
RSX-11M CRASH DUMP ANALYZER      V4.2      19-APR-85  13:58       PAGE 8
ERROR LOG BUFFERS

          (1)                              (2)                          (3)
    BUFFER ADDRESS = 072304    ENTRY TYPE CODE = 000002   ENTRY TYPE SUBCODE = 000001

    TIME = 7-JUL-81   07:46:11
                                      (4)

000000    000000  000210  000034  000055  000401  030463  020040  003401
000020    000025  000013  000402  003521  003407  005456  000106  000001
000040    000046  046504  000001  000001  000000  047115  052105  031526
000060    041117  000112  000000  000000  000000  000001  064766  000000
000100    011532  000000  000000  000030  045662  131574  000424  047503  (5)
000120    000400  001000  000003  000017  024010  177000  000000  000056
000140    101220  172741  044702  000424  004301  100301  000000  000000
000160    000253  177777  033065  022000  004066  000000  010341  100000
000200    101721  000001  000001  105262  030111
```

Figure 3-8 Error Log Packets

3.1.9 Low Core Memory Dump (RSX-11M-PLUS Only)

The listing shown in Figure 3-9 contains a dump of RSX-11M-PLUS low core memory, alphabetized by label.

The following summary lists labels found in RSX-11M-PLUS low core memory and their meanings:

$CRSBF	Internal crash stack
$CRUPC	Scratch user program counter (PC)
$CRUST	Scratch user Processor Status Word (PSW)
$CURPR	Pointer to current task priority
$CXDBL	Context switch disable flag
$DICSV	Temporary storage for directive service
$DXDEP	Entry point to dynamic Executive debugger interface
$DXDK5	Saved KINAR5 for dynamic Executive debugger interface
$DXDRL	Relocation bias for dynamic Executive debugger interface
$HEADR	Pointer to current task header
$PARLV	Interrupt recursion level counter
$RQSCH	Pointer to current reschedule pointer
$SAHDB	Bias of current task header
$SAHPT	Virtual address of current task header
$SAVSP	Saved stack pointer
$SIRWF	Supervisor instruction space read/write flag
$STKDP	Stack depth indicator
$SUPFL	Supervisor window flag
$TKTCB	Pointer to current task TCB

```
RSX-11M-PLUS CRASH DUMP ANALYZER  V3.0     19-APR-85  15:44       PAGE 3
LOWCORE ALPHABETIZED DUMP

$CRSBF 000000            000000            000000        $RQSCH 021712
       000000            000000            000000        $SAHDB 000000
       000000            000000            000000        $SAHPT 000000
       000000            000000            000000        $SAVSP 000000
$CRUPC 000000            000000        $DXDEP 000000     $SIRWF    000
$CRUST 000000            000000        $DXDK5 000000     $STKDP 000000
$CURPR 021722            000000        $DXDRL 000000     $SUPFL    000
$CXDBL    000            000000        $HEADR 000000     $TKTCB 000000
$DICSV 000000            000000        $PARLV 177777
```

Figure 3-9 Low Core Memory

3.2 OPTIONAL INFORMATION

CDA gives you additional information when you use the analysis switches described in Chapter 2. Figures 3-10 through 3-24 illustrate the output that CDA provides when you use these switches.

3.2.1 Active Tasks

The listing shown in Figure 3-10 contains active task information. The Receive Queue, AST Queue, Receive-by-Reference Queue, and Offspring Control Block sections of this example appear only if the task has them; otherwise, they are suppressed. Section 3.2.2 describes the additional information in the active task listing when the active task is MCR.

Item	Description
1.	Task name
2.	Address of Task Control Block (TCB) for the task
3.	Name of the partition in which the task runs
4.	Address of Partition Control Block (PCB)
5.	Base address for the partition in which the task runs
6.	Device that contains task image
7.	Beginning logical block number (LBN) of the task on the device
8.	Running priority
9.	Number of outstanding QIO requests
10.	Current UIC (either the login UIC or the UIC specified with a SET command)
11.	Physical name of task's pseudo device
12.	Maximum size of task image in 32(decimal)-word blocks
13.	State of local event flags for task
14.	First status word (blocking bits), using the following three-letter codes:

-EXE - Task not executing

RDN - I/O rundown in progress

CIP - Task blocked for checkpoint in progress

MSG - Abort message being output

CKR - Task has checkpoint request (RSX-11M-PLUS only)

BLC - Increment blocking count

STP - Task stopped by CLI command

Item	Description
15.	Second status word (state bits), using the following three-letter codes:

AST - Asynchronous system trap (AST) in progress

SIO - Task stopped for buffered I/O

DST - AST recognition disabled

AFF - Task installed with affinity

-CHK - Task not checkpointable

SEF - Stopped for event flag

REX - Exit AST specified

HLT - Task being halted

ABO - Task marked for abort

STP - Task stopped

SPN - Task suspended

WFR - Task in wait-for state

16. Third status word (attribute bits), using the following three-letter codes:

ACP - Task is an Ancillary Control Processor (ACP)

PMD - Task not dumped on synchronous abort

CMD - Task is executing a CLI command

REM - Remove task on exit

PRV - Task is privileged

MCR - Task requested as an external MCR function

SLV - Task is a slave task

CLI - Task is a command line interpreter

RST - Task is restricted

NSD - Task does not allow send data

CAL - Task has checkpoint space in task image

ROV - Task has resident overlays

NET - Network protocol level

GFL - Group-global event flags are locked

SWS - Reserved for Software Services

MPC - Mapping change with outstanding I/O

Item	Description

17. Fourth status word, using the following three-letter codes:

    MUT - Task is a multiuser task

    LDD - Task load device is dismounted

    PRO - TCB is a prototype

    PRV - Task was privileged but has cleared TB.PRV with the GIN directive

    DSP - Task was built for user data space

    SNC - Task uses common synchronization

18. Octal dump of TCB in offset mode

RECEIVE QUEUE (if the task has one)

19. Starting address of receive block

20. Name of task

21. Octal dump of receive block in offset mode

OFFSPRING CONTROL BLOCK (OCB) LIST (if the task has one)

22. Exit event flag number of offspring task

23. Name of parent task

24. Octal dump of offspring control block in offset mode

ASYNCHRONOUS SYSTEM TRAP (AST) QUEUE (if the task has one)

If a task has an AST Queue, CDA lists its contents. If the task also has a Receive Queue, the AST Queue appears immediately after the Receive Queue on the output listing. If the task does not have a Receive Queue, the AST Queue is listed after the fourth status word information (the example in Figure 3-10 does not include an AST Queue).

An item appearing in the AST queue may be one of the following:

- Unsolicited AST
- Floating point AST
- Receive data AST
- Receive-by-reference AST
- Parity error AST
- Requested exit AST
- Power fail
- CLI command arrival AST
- Buffered I/O AST

- Offspring task AST
- Segmented buffered I/O completion AST
- Task force trace bit trap AST
- Delayed I/O completion AST
- Group-global rundown AST
- I/O request packet
  - Address of AST block
  - A 2-byte indicator (The high-order byte is an offset into the header of the AST control block; the low-order byte is the length of the AST control block in bytes.)

NOTE

If the low-order byte is negative, the block is not an AST block, but an I/O request packet internal to the system. If the low-order byte is 0, the block is an unsolicited character AST.

  - Number of bytes allocated on task stack
  - Entry point of AST routine
  - Number of AST parameters
  - Octal dump of the AST block in offset mode (On RSX-11M-PLUS systems, two additional negative offset words appear in the dump.)

RECEIVE-BY-REFERENCE QUEUE (if the task has one)

If a task has a Receive-by-Reference Queue, CDA lists its contents. If the task also has an Offspring Control Block list, the Receive-by-Reference Queue appears immediately before the OCB list on the output listing. If the task does not have an OCB list, the Receive-by-Reference Queue is the last list on the Active Tasks listing (the example in Figure 3-10 does not include a Receive-by-Reference Queue).

- Address of Receive-by-Reference Queue Block
- Address of the Task Control Block (TCB) for the task that initiated the Send by Reference
- Contents of event flag mask
- Address of event flag mask
- Pointer to created attachment descriptor
- Offset into partition as specified in window definition
- Length to be mapped
- The receiving task's access rights to region being mapped
- Octal dump of Receive-by-Reference Block in offset mode

```
RSX-11M-PLUS CRASH DUMP ANALYZER    V3.0     19-APR-85  13:02       PAGE 13
ACTIVE TASKS


...LDR (1)
------
                (2)                    (3)                    (4)
  TCB ADDRESS = 111650      PAR = SYSPAR      PCB ADDRESS = 111434
                (5)                    (6)                    (7)
  LOAD ADDRESS = 00461600     LOAD DEVICE = LB0:     LBN = 00076636
     (8)                 (9)                   (10)                (11)
  PRI = 248.      I/O COUNT = 0.      UIC = [1,24]      TI = CO0:
     (12)                                  (13)
  MAX SIZE = 000035     EVENT FLAGS = <1-16> 000001  <17-32> 000000

  T.STAT:  000000 (14)

  T.ST2:   020020  -CHK  STP (15)

  T.ST3:   050200  -PMD  PRV  NSD (16)

  T.ST4:   000000 (17)

  TCB DUMP:

    000000    000000  000370  000000  131574  045662  000000  111662  000000
    000020    111666  000001  000000  022370  111300  000000  020020  050200
    000040    000370  076636  031420  111434  000035  033500  111426  111426 (18)
    000060    000000  000000  000000  000001  111672  000035  000000  000000
    000100    000000  111750  000000  111754  000000  000000  000000  000000

      .
      .
      .

    RECEIVE QUEUE
    -------------

                 (19)                              (20)
    RECEIVE BLOCK ADDRESS = 062000   TASK NAME = TU$KRD

    000000    000000  100143  043624  062640  050210  016000  121502  024172
    000020    140002  133406  000000  000000  000000  000000  000000  000000 (21)
    000040    046522  000000

      .
      .
      .

  OCB LIST:
  ---------

    EXIT EVENT FLAG (0,EFN) = 000000 (22)

    PARENT TASK NAME = QMG... (23)

    OCB DUMP:

    000000    000000  000000  106424  121350  000000  000000  066117  131574 (24)
    000020    000000  000000  000000  000000  000000  000000
```

Figure 3-10 Active Tasks (Truncated)

3.2.2 Active Task (MCR)

The active task listing for the MCR task (MCR...) contains more information than the active task listing for other tasks. Figure 3-11 shows a listing in which the first 17 items are the same as those in Figure 3-10. The following list describes only the items that are different from those in the previous figure, when MCR... is the active task.

Item	Description
1.	Address of MCR input buffer
2.	Address of Unit Control Block (UCB) of the requesting terminal
3.	Device name and unit number of the terminal that sent block to MCR (ASCII characters)
4.	Octal dump of the MCR input buffer in offset mode
5.	Address of command buffer
6.	Address of Task Control Block of the requesting task
7.	ASCII dump of command buffer
8.	Octal dump of command buffer in offset mode

```
RSX-11M-PLUS CRASH DUMP ANALYZER  V3.0     19-APR-85  16:23       PAGE 10
ACTIVE TASKS


MCR...
------

  TCB ADDRESS = 114610      PAR = SYSPAR      PCB ADDRESS = 037050

  LOAD ADDRESS = 00174400      LOAD DEVICE = LB0       LBN = 00073747

  PRI = 160.      I/O COUNT = 0.      UIC = [1,24]      TI = TT3

  MAX SIZE = 000100      EVENT FLAGS = <1-16> 000001  <17-32> 040000

  T.STAT:  000000

  T.ST2:   000020  STP

  T.ST3:   051300  -PMD PRV  CLI  NSD  CAL

TCB DUMP

    000000    000000  000240  000000  050712  131574  000000  114622  000000
    000020    114626  000001  040000  024552  113760  000000  000020  051300
    000040    000240  073747  035636  037050  000100  112330  036046  036046
    000060    000000  000000  000000  114674  000000  114700  000000  000000
    000100    004256

    RECEIVE QUEUE
    -------------

                                    (1)                    (2)
    COMMAND LINE INPUT BUFFER ADDRESS = 036730     UCB = 030245

    TT50 : (3)

    000000    043214  030245  035770  000050  000020  000000  030244  000400
    000020    120430  001750  140030  000000  001751  140010  000027  000361
    000040    000040  000000  000000  000000  000000  000000  000000  000214   (4)
    000060    026226  000000  000000  117404  005627  000362  000000  000362
    000100    000000  113234  000010  037050  000000  036120  036120  000000

    MCR COMMAND BLOCKS
    ------------------

                  (5)                    (6)
    BUFFER ADDRESS = 103100     TCB = 102730

     ASN DB01:=SY:/LOGIN (7)

    000000    000000  102730  051501  020116  041104  030460  036472  054523
    000020    027472  047514  044507  015516  026226  000000  000000  117404
    000040    024600  000420  000000  000420  000000  103020  000010  104074   (8)
    000060    000000  060440  060440  060370  103370  000114  000612  033406
    000100    170000  126102  000000  000000  000000  000000  000012  000137
    000120    000000  000000
```

Figure 3-11 Active Task (MCR)

3.2.3 Task Headers

Figure 3-12 is an example of a task header listing. The following list describes its contents.

Item	Description
1.	Task name
2.	Pointer to the first word in the task header
3.	Pointer to the first word in the Task Control Block (TCB)
4.	Contents of Processor Status Word (PSW) and Program Counter (PC)
5.	Contents of the general registers
6.	Initial contents of the PSW, the PC, and the stack pointer (SP)
7.	The task header size in decimal bytes, the number of windows required to map the task, and the number of logical unit numbers assigned to the task
8.	Current and default UIC
9.	Pointer to number of window blocks
10.	Pointer to header guard word
11.	Work area extension vector pointer
12.	Priority difference for swapping
13.	Directive Status Word
14.	Address of File Control Services (FCS) impure area
15.	Address of FORTRAN impure area
16.	Address of overlay impure storage
LOGICAL UNIT TABLE	
17.	Logical unit number (LUN)
18.	Physical device name before redirect
19.	Window pointer in header

Item	Description
20.	Low-order byte of this word indicates the number of map entries active; the high-order byte has the following bit assignments: WI.RDV=400 - read virtual address allowed if set WI.WRV=1000 - write virtual block allowed if set WI.EXT=2000 - extend allowed if set WI.LCK=4000 - set if locked against shared access WI.DLK=10000 - set if deaccess lock enabled WI.BPS=100000 - bypass access interlock if set
21.	Address of File Control Block
22.	File number
23.	File sequence number
24.	File Control Block status word
25.	Number of accesses
26.	Number of block locks

WINDOW BLOCKS

27.	The name of the partition in which the task runs
28.	The virtual limits of the task
29.	Address of attachment descriptor
30.	Window size in 32-word blocks
31.	Offset into partition
32.	Address of the first Page Description Register (PDR) used to map the window
33.	Number of PDRs used
34.	The contents of the last PDR used
35.	Octal dump of task header in offset mode

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85     15:13        PAGE 37
TASK HEADERS

TKTN (1)
------
          (2)                                (3)
  HEADER ADDRESS = 076060       TCB ADDRESS = 107134

  PS=170000      PC=122630 (4)

  R0=120254  R1=000065  R2=000060  R3=140354  R4=120702  R5=051024  SP=120226(5)

  INITIAL PS = 170017   INITIAL PC = 120764   INITIAL SP = 120230(6)

  HEADER SIZE = 102.    NO. OF WINDOWS = 1.   NO. OF LUNS = 1.(7)

  CURRENT UIC = [1,24]       DEFAULT UIC = [1,24](8)
       (9)                  (10)                (11)                (12)
  H.WND = 076162     H.GARD = 076224     H.VEXT = 000000     H.SPRI = 5.
       (13)                 (14)                (15)                (16)
  DSW = 000001       H.FCS = 000000      H.FORT = 000000     H.OVLY = 000000


  LOGICAL UNIT TABLE:
(17) (18)    (19)       (20)      (21)      (22)      (23)      (24)    (25)   (26)
  #  DEV  WINDOW    W.CTL     W.FCB     F.FNUM    F.FSEQ    F.STAT    NAC    NLCK
  -  ---  ------    -----     -----     ------    ------    ------    ---    ----
  1  TI0: 000000


  WINDOW BLOCKS:
   (27)        (28)          (29)       (30)       (31)      (32)     (33)     (34)
   PAR      VIRT LIMITS    ATT DESC   WND SIZE    OFFSET    1ST PDR    NO.    LAST PDR
   ---      -----------    --------   --------    ------    -------    ---    --------
  TKNPAR  120000 127777    045220     000100      000000    177612     1      037406


  HEADER:

    000000    120226  000146  140354  163500  000424  000424  170017  120764
    000020    120230  000000  000000  000000  000000  000000  000000  000000
    000040    000000  000000  076162  000001  000000  000000  000000  000000 (35)
    000060    000005  000000  000000  000000  000000  076224  000001  044570
    000100    000000  000001  114450  120000  127777  045220  000100  000000
    000120    000612  037406  170000  122630  051024  120702  140354  000060
    000140    000065  120254  000000
```

Figure 3-12 Task Headers (Truncated)

3.2.4 Command Line Interpreter Parser Block (CPB)

The listing shown in Figure 3-13 contains the Command Line Interpreter Parser Block for MCR. The listing corresponds to the following items:

Item	Description
1.	Task name of the CLI
2.	Starting address of the CPB
3.	C.PSTS, which is the CPB status word
4.	Dump of the CPB in octal

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85     15:13        PAGE 9
CLI PARSER BLOCKS


CLI TASK NAME   MCR... (1)
CPB ADDRESS = 010472     CLI NAME = MCR (2)
C.PSTS:   SGL (3)

CPB DUMP:

    000000    110160  050712  000000  000040  003404  005015  000076  005015 (4)
    000020    041515  037122  000000
```

Figure 3-13 CLI Parser Blocks

3.2.5 Partition Information

CDA outputs partition information in two segments. The listing shown in Figure 3-14 contains system partition information, and the listing shown in Figure 3-15 represents individual partition information. The following list describes elements of Figure 3-14. Individual partitions include Attachment Descriptors and Wait Queues when they apply.

Item	Description
1.	Partition names
2.	Partition Control Block (PCB) address
3.	Base address of partition in memory
4.	Size of the partition
5.	Type of partition
6.	Task or tasks occupying the partition

```
RSX-11M CRASH DUMP ANALYZER      V4.2     19-APR-85    15:13        PAGE 54
PARTITION INFORMATION
```

M E M O R Y M A P

(1) PARTITION	(2) PCB ADR	(3) BASE	(4) SIZE	(5) TYPE	(6) OCCUPIED BY
<EXEC>		00000000	00044700		
<POOL>		00044700	00050100		
CEXPAR	114734	00115000	00003000	MAIN COMMON	
TTPAR	114670	00120000	00040000	MAIN DRIVER	TT:
EXCOM1	114624	00160000	00014600	MAIN COMMON	
EXCOM2	114560	00174600	00006100	MAIN COMMON	
SYSPAR	114514	00202700	00010000	MAIN TASK	MCR...
TKNPAR	114450	00212700	00010000	MAIN TASK	TKTN
DRVPAR	114404	00222700	00030200	MAIN SYS	
	114340	00222700	00002100	SUB DRIVER	DB:
	114240	00225000	00001200	SUB DRIVER	DK:
	114140	00226200	00003000	SUB DRIVER	DM:
	114040	00231200	00003000	SUB DRIVER	DR:
	113740	00234200	00001000	SUB DRIVER	EM:
		00235200	00000100	<HOLE>	
	113640	00235300	00001100	SUB DRIVER	DT:
	113540	00236400	00001400	SUB DRIVER	DX:
	113440	00240000	00002200	SUB DRIVER	DL:
	113340	00242200	00002600	SUB DRIVER	DD:
	113240	00245000	00001300	SUB DRIVER	LP:
	113104	00246300	00004300	SUB DRIVER	MM:
	112450	00252600	00000300	SUB DRIVER	CO:
LDRPAR	112404	00253100	00002500	MAIN TASK	LDR...
BASIC2	112340	00255600	00040000	MAIN COMMON	
FCSRES	112274	00315600	00040000	MAIN COMMON	
TSTPAR	112230	00355600	00100000	MAIN TASK	
GEN	112164	00455600	03322200	MAIN SYS	
	045620	00455600	00024000	SUB TASK	DB2FCP
	057144	00501600	00025500	SUB TASK	NETACP
		00527300	00005200	<HOLE>	
	064060	00534500	00023300	SUB TASK	RMHACP
	053004	00560000	00005200	SUB TASK	CA.T6
	073534	00565200	00005200	SUB TASK	CA.T30
		00572400	00005700	<HOLE>	
	045010	00600300	00005700	SUB TASK	PMT...

Figure 3-14 Partition Information

Each item in the following list describes a correspondingly numbered item in Figure 3-15.

Item	Description

PARTITION CONTROL BLOCK

1. Partition name
2. Pointer to the first word of the PCB
3. Type of partition
4. Name of main partition
5. Physical base address of partition in 32-word blocks
6. Partition size in 32-word blocks
7. Pointer to the first word of the TCB of attached task
8. Partition protection word (mapped system only)
9. Priority of attached task or partition
10. I/O count of attached task or partition
11. Partition status flags, using the following three-letter codes:

    OUT - Partition is out of memory

    CKP - Partition checkpoint in progress

    CKR - Partition checkpoint is requested

    CAF - Checkpoint space allocation failure

    -CHK - Partition is not checkpointable

    FXD - Partition is fixed

    LFR - Last head of region failure

    PER - Parity error in partition

    LIO - Marked by Shuffler for long I/O

    NSF - Partition cannot be shuffled

    COM - Library or common block

    DEL - Partition should be deleted when not attached

12. Octal dump of PCB in offset mode

Item	Description

ATTACHMENT DESCRIPTOR

13. Address of attachment descriptor

14. Partition to which attachment occurs

15. Name of attaching task

16. PCB attachment queue thread word

17. TCB attachment queue thread word

18. Priority of highest priority task attached to this partition

19. I/O count of attached task on RSX-11M systems; I/O count of attached partition on RSX-11M-PLUS systems

20. Number of times task is mapped through this attachment descriptor

21. Attachment descriptor status byte, using the following three-letter codes:

    DEL - Task has delete access

    EXT - Task has extend access

    WRT - Task has write access

    RED - Task has read access

    PRO - TCB is secondary pool TCB bias

    SPB - Cache bypass request

    RBP - Request to not bypass cache

22. Octal dump of attachment descriptors in offset mode

PARTITION WAIT QUEUE

23. Name of the task awaiting access to the partition

24. Address of TCB for the task

25. TI: device for the task

26. Task's priority

27. Second status word (state bits)--same as item 15 of the active task dump (Section 3.2.1)

RSX-11M-PLUS SYSTEMS ONLY ADDITIONAL ITEM DESCRIPTION

28. Resident mapped task count

29. Wait queue contains partition description rather than task description

```
PARTITION  PCB ADR  TYPE   MAIN    BASE     SIZE    P.TCB    PRO     PRI   IOC
---------  ------   ----   ----    ----     ----    -----    ---     ---   ---
 DRVPAR    117270   SUB    DRVPAR  001312   000060  040754   000000  0.    0.

  P.STAT:  SYS DRV

     000000    000000  000000  015746  062072  000000  117734  001312  000060
     000020    000000  000000  000000  040754  000060  000000  000000  000000
     000040    117326
```

```
    (1)        (2)      (3)     (4)     (5)      (6)     (7)      (8)     (9)   (10)
PARTITION  PCB ADR  TYPE   MAIN    BASE     SIZE    P.TCB    PRO     PRI   IOC
---------  ------   ----   ----    ----     ----    -----    ---     ---   ---
 SYSPAR    117074   MAIN   SYSPAR  001372   000105  115264   000000  0.    0.

  P.STAT: (11)

     000000    117030  000000  075273  062072  000000  117074  001372  000105
     000020    115764  115764  100200  115264  000000  042760  000000  042000   } (12)
     000040    041764
```

```
WAIT QUEUE:
    (23)       (24)       (25)     (26)          (27)
  TASK     TCB ADR    TI     T.PRI     STATE BITS (T.ST2)
  ----     -------    --     -----     ------------------
F11ACP     115764     C00    149.        CAF  STP
```

```
ATTACHMENT DESCRIPTORS:
    (13)       (14)          (15)        (16)      (17)      (18)   (19)      (20)
ADDRESS    PARTITION    ATT TASK    A.PCBL    A.TCBL    PRI    IOC    MAP COUNT
-------    ---------    --------    ------    ------    ---    ---    ---------
042000       SYSPAR       ...MCR    041764    000000    160.   0.       0.

A.STAT:  WRT RED (21)

000000    041764  000240  115264  000000  000003  117074 (22)
```

Figure 3-15 Partition Control Blocks and Attachment Descriptors

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0     2-MAY-85  15:36        PAGE 83
PARTITION INFORMATION

                                                                      (28)
  NAME     PCB ADR  TYPE   MAIN   BASE    SIZE   P.OWN     PRO    PRI  RMCT
---------  ------  ----   ----   ----    ----   -----    ---    ---  ----
 GEN       117404  MAIN   GEN    004432  047346  000000  000000  0.   0.

  P.STAT:  000000

  P.ST2:   000000

    000000    000000  000000  026226  000000  035724  117404  004432  047346
    000020    061774  063170  000000  000000  000000  000000  000000  000000
    000040    000000

    WAIT QUEUE: (29)

  NAME     PCB ADR  TYPE   MAIN   BASE    SIZE   P.TCB     PRO    PRI  RMCT
---------  ------  ----   ----   ----    ----   -----    ---    ---  ----
 BAP2      061774  SUB    GEN    012731  000372  061670  000000  50.   0.

  P.STAT:  140010   OUT  CKP  DEL

    000000    063170  000062  026226  000000  063170  117404  012731  000372
    000020    000000  000372  074114  061670  140010  062474  000000  037360
    000040    037360
```

Figure 3-15 (Cont.) Partition Control Blocks and Attachment Descriptors

3.2.6 Common Block Directory

CDA lists partition information, status words, and Partition Control Blocks for each installed, named common region. The listing in Figure 3-16 shows a Common Block Directory entry. The following list describes the items in Figure 3-16.

Item	Description
1.	Name of the installed common region partition
2.	Address of Partition Control Block (PCB)
3.	Type of partition
4.	Name of the main partition
5.	Physical base address of partition
6.	Size of partition in 32(decimal)-word blocks
7.	Owning UIC of the common region
8.	Partition protection word

Item	Description
9.	Resident mapped task count
10.	Partition status words (refer to Section 3.2.4)
11.	Octal dump of PCB
12.	Address of PCB of the common task image file
13.	Address of Unit Control Block (UCB) of the device on which the common resides
14.	Starting logical block number (LBN) of the common task image file
15.	Word that always contains a 0

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0     2-MAY-85  14:03        PAGE 75
COMMON BLOCK DIRECTORY

   (1)          (2)      (3)    (4)     (5)      (6)      (7)       (8)          (9)
  NAME        PCB ADR   TYPE   MAIN    BASE     SIZE    P.OWN      PRO     PRI  RMCT
---------     ------    ----   ----    ----     ----    -----     ---      ---  ----
 BASIC2       043254    SUB    GEN     017747   000400  001454    000000   0.   0.

  P.STAT:     000200    COM
                                                                              } (10)
  P.ST2:      000006    APR

    000000    000000  000000  006273  034330  040370  117404  017747  000400
    000020    053550  000000  045240  001454  000200  000006  000000  000000  } (11)
    000040    043312

    COMMON TASK IMAGE FILE PCB
       (12)        (13)          (14)          (15)
      PCB ADR     P.UCB         P.LBN         P.REL
      -------     -----         -----         -----
      045240      035662  000001,023653       000000

  NAME        PCB ADR   TYPE   MAIN    BASE     SIZE    P.OWN      PRO     PRI  RMCT
---------     ------    ----   ----    ----     ----    -----     ---      ---  ----
 FCSSUP       053550    SUB    GEN     000000   000212  000401    000000   0.   0.

  P.STAT:     100200    OUT  COM

  P.ST2:      000000

    000000    000000  000400  023013  075030  000000  117404  000000  000212
    000020    047304  000212  040114  000401  100200  000000  000000  000000
    000040    053606

    COMMON TASK IMAGE FILE PCB

      PCB ADR     P.UCB         P.LBN         P.REL
      -------     -----         -----         -----
      040114      035662  000002,001071       000000
```

Figure 3-16 Common Block Directory

3.2.7 Device Information

CDA lists information on all devices known to the system. The listing in Figure 3-17 shows a typical terminal device listing with an I/O packet. The Terminal Status Words (items 15, 16, and 17) appear only in listings for terminal devices. The codes for these items apply only to the full-duplex terminal driver. The section labeled I/O REQUEST PACKETS appears only for devices that have an I/O request in progress or an I/O request queued at the time of the system crash. The following list describes the items in Figure 3-17.

Item	Description
1.	Device name
2.	Address of offset 0 in Unit Control Block (UCB)
3.	Address of offset 0 in Device Control Block (DCB)
4.	Address of offset 0 in Status Control Block (SCB)
5.	Device to which unit is redirected
6.	Name of Ancillary Control Processor (ACP)
7.	Name of attached task
8.	Pointer to the UCB name of the owning terminal
9.	UIC used to log into the system
10.	Unit status byte, using the following three-letter codes: BSY - Unit is busy -MNT - Unit is not mounted FOR - Unit is mounted as a foreign volume MDM - Unit is marked for dismount PWF - Power fail occurred WCK - Write check enabled SPU - Unit is spinning up VV - Volume is valid
11.	Unit status extension byte, using the following three-letter codes: OFL - Unit off line -RED - Unit is not redirectable PUB - Unit is public device UMD - Unit attached for diagnostics PDF - Privileged diagnostic functions only

Item	Description
12.	Control Processing flags, using the following three-letter codes:
	-ALG - Byte alignment not allowed
	NPR - Device is a NPR device
	QUE - Call driver before queuing
	PWF - Always call driver at power fail entry point
	ATT - Call driver on attach/detach
	KIL - Always call driver at I/O kill
13.	First device characteristics word, using the following three-letter codes:
	REC - Record-oriented device
	CCL - Carriage-control device
	TTY - Terminal device
	DIR - File-structured device
	SDI - Single directory device
	SQD - Sequential device
	MSD - Mass storage device
	EXT - Unit on extended 22-bit UNIBUS controller
	UMD - User-mode diagnostics supported
	MBC - MASSBUS device
	SWL - Unit software write locked
	ISP - Input spooled device
	OSP - Output spooled device
	PSE - Pseudo device
	COM - Device is mountable as COM channel
	F11 - Device is mountable as Files-11 device
	MNT - Device is mountable
14.	Second device characteristics word, using the following three-letter codes:
	DH1 - Unit is a multiplexer
	DJ1 - Unit is a DJ11
	RMT - Unit is remote
	HFF - Unit handles hardware form feeds
	NEC - Solicited input not echoed

Item	Description
CRT	Unit is a CRT
ESC	Unit generates escape sequences
-LOG	User not logged in on terminal
SLV	Unit is a slave terminal
DZ1	Unit is a DZ11
HLD	Terminal is in hold screen mode
AT.	MCR command AT. is being processed
PRV	Unit is a privileged terminal
L3S	Unit is a LA30S terminal
VT5	Unit is a VT05B terminal
LWC	Lowercase to uppercase conversion

15. Terminal status word, using the following three-letter codes:

    RST - Read with special terminators in progress

    RUB - Rubout sequence (non-CRT) in progress

    ESC - Escape sequence in progress

    RAL - Read pass all in progress

    RNE - Echo suppressed

    CTO - Output disabled

    OBY - Output busy

    IBY - Input busy

    BEL - Bell pending

    DPR - Defer processing of character in buffer

    DEC - Defer echo of character in buffer

    DSI - Input processing disabled

    CTS - Output stopped by CTRL-S

    USI - Unsolicited input in progress

    OBF - Buffered output in progress

    IBF - Buffered input in progress

Item	Description
16.	Second terminal status word using the following three-letter codes: ACR - Wrap-around required CR - Trailing carriage return required on output BRQ - Break-through write is queued WRA - Control for wraparound SRQ - Special request is queued WRB - Low bit in 52-WRA bit pattern ORQ - Output request is queued IRQ - Input request is queued HFL - Horizontal fill required VFL - Vertical fill required HHT - Hardware horizontal tab is present HFF - Hardware form-feed is present FLF - Force line feed before next echo FDX - Line is full duplex mode
17.	Fourth terminal status word, using the following three-letter codes: RAL - Terminal is in read-pass-all mode WES - Task waiting for escape sequence RPO - Read with prompt output in progress TAB - Type-ahead buffer allocation requested 8BC - Pass eight bits on input ABD - Autobaud speed detection enabled RCU - Restore cursor ABP - Autobaud speed detection in progress WAL - Terminal is in write-pass-all mode VER - Last character in type-ahead buffer has a parity error

Item	Description

BCC - Last character in type-ahead buffer has a framing error

DAO - Last character in type-ahead buffer has a data overrun error

PCU - Position cursor

NOTE

On RSX-11M systems that use the half-duplex terminal driver, CDA dumps two terminal status words.

UNIT CONTROL BLOCK

18. Octal dump of Unit Control Block (UCB), including negative offsets, and octal dump of UCB extension if a UCB extension is present

DEVICE CONTROL BLOCK

19. Octal dump of Device Control Block

STATUS CONTROL BLOCK

20. Octal dump of Status Control Block

I/O REQUEST PACKETS

21. Address of the first word of the I/O packet

22. Name of the task requesting I/O

23. Priority of the task requesting I/O

24. Event flag number used to signal I/O completion

25. Logical unit number used by requesting task

NOTE

If the task was checkpointed while the packet was queued, this number may not be correct. If the address in I.LN2 is within the task header, the logical unit number is correct.

26. I/O function codes (for detailed information on the legal I/O function codes for each device, refer to the RSX-11M-PLUS I/O Drivers Reference Manual)

27. Status of the I/O request - current or queued

28. Octal dump of I/O request packet in offset mode

29. I.LN2 - pointer to the second word of the LUN

```
RSX-11M CRASH DUMP ANALYZER      V4.2     2-MAY-85      15:13       PAGE 110
DEVICE INFORMATION

TT11:
-----

  UCB ADR   DCB ADR   SCB ADR   REDIRECT   ACP      ATT     OWNER   LOGIN UIC
  -------   -------   -------   --------   ---      ---     -----   ---------
  040474    037630    044266                        ...MAI  NONE    [7,37]

  U.STS:

  U.ST2:

  U.CTL:    QUE  PWF  ATT  KIL

  U.CW1:    TTY  CCL  REC

  U.CW2:    DH1  CRT  PRV  LWC

  U.TSTA:        IBY  IBF

  U.TSTA+2:      ACR  ORQ  IRQ  HHT  FLF

  U.TSTA+4:      TAB


    UNIT CONTROL BLOCK:

    040466    000000  003437  000000  037630  040474  000474  000010  000007
    040506    102011  000335  000120  044266  056604  145100  100200  050301
    040526    000100  141730  000030  000001  013400  003437  000015  000004
    040546    000000  000000  000000

    UNIT CONTROL BLOCK EXTENSION:

    145100    075670  146730  146734  000044  000054  146730  000000  000000
    145120    127147  000001  146734  146470  000000  000000  002400  000022
    145140    000000  013400  000000  023531

    DEVICE CONTROL BLOCK:

    037630    044434  037674  052124  030001  000060  121274  163177  000130
    037650    160000  000000  000007  000000  000001  000006  114670

    STATUS CONTROL BLOCK:

    044260    100200  000011  000004  000000  044266  034240  001000  000000
    044300    160020  067454  031000  170500  000000  000000  001200  000000
    044320    044316  035240  001000  000002  160040


    I/O REQUEST PACKETS:

    PACKET ADR     REQUESTOR     PRI     EFN     LUN     FUNCTION CODE     STATUS
    ----------     ---------     ---     ---     ---     -------------     ------

      075670         ...MAI      65.     32.      5.         IO.RLB         INPUT


    000000    000000  020101  056604  074354  040474  001000  057220  017645
    000020    140020  000000  000572  140040  000120  100000  000000  000056
    000040    000020  075624
```

Figure 3-17 Device Information and I/O Packet (Truncated)

3.2.8 System Task Directory

CDA scans the System Task Directory and outputs the information contained in Figure 3-18. The information in this format is identical to the first 17 items described in Figure 3-11 of this manual.

```
RSX-11M CRASH DUMP ANALYZER      V4.2     2-MAY-85      15:15       PAGE 31
SYSTEM TASK DIRECTORY

. LDR.
------

  TCB ADDRESS = 041572     PAR = LDR        PCB ADDRESS = 041536

  LOAD ADDRESS = 00000000     LOAD DEVICE = LB0      LBN = 00000000

  PRI = 248.     I/O COUNT = 0.     UIC = [1,1]     TI = C00

  MAX SIZE = 000000     EVENT FLAGS = <1-16> 000001  <17-32> 000000

  T.STAT:

  T.ST2:   -CHK FXD  STP

  T.ST3:   PRV

    000000    000000  000370  000000  127414  015754  000000  041604  000000
    000020    041610  000001  000000  041304  114564  000000  022020  010000
    000040    000370  000000  041354  041536  000000  115264  000000  041646
    000060    000000  000000  000000  041656  000000  041662  000000  000000


TKTN
------

  TCB ADDRESS = 114564     PAR = SYSPAR     PCB ADDRESS = 117074

  LOAD ADDRESS = 00137200     LOAD DEVICE = LB0      LBN = 00057410

  PRI = 248.     I/O COUNT = 0.     UIC = [0,0]     TI = NONE

  MAX SIZE = 000105     EVENT FLAGS = <1-16> 000000  <17-32> 000000

  T.STAT:  -EXE OUT

  T.ST2:

  T.ST3:   -PMD PRV  CAL

    000000    000000  000370  000000  077314  053600  000000  114576  000000
    000020    114602  000000  000000  000000  116064  100400  000000  050100
    000040    000370  057410  041354  117074  000105  000000  000000  000000
    000060    114642  000000  000000  000000  114652  000000  114656  000000
```

Figure 3-18 System Task Directory (Truncated)

3.2.9 Pool Dump

As shown in Figure 3-19, CDA prints the system pool in octal, Radix-50, and ASCII. On RSX-11M-PLUS systems with secondary pool support, CDA prints a dump of secondary pool with the /SECPOL switch. If a line is repeated more than nine times, CDA prints it once and then prints a message indicating the number of identical lines.

The symbols in Figure 3-19 have the following meanings:

- `*` Indicates that the next word is allocated.

- `+` Indicates that the next word is contained in an unused, preallocated I/O packet (in $PKAVL free list).

  NOTE

  $PKAVL is a list containing fixed-size blocks. The blocks in this list are used for fast allocation, and I.LGTH determines the length of these blocks.

- `-` Indicates that the next word is allocated in both $CRAVL and $PKAVL. (This is an error condition).

  NOTE

  $CRAVL is the free pool list head.

RSX-11M CRASH DUMP ANALYZER V4.2 2-MAY-85 15:15 PAGE 42
POOL DUMP

S Y S T E M P O O L

* = NEXT WORD ALLOCATED FIRST FREE BLOCK ($CRAVL) = 041760
\+ = NEXT WORD IS IN $PKAVL LIST $PKAVL = 042120
\- = NEXT WORD IS IN $PKAVL AND ALSO IN $CRAVL

```
041674   * 000000 * 042710 * 042164 * 042164 ¦     KFX J8D J8D¦¦   HEtDtD¦
041704   * 000400 * 000002 * 003204 * 000407 ¦ FP    B AA.  FW¦¦        ¦
041714   * 000007 * 002400 * 002112 * 000401 ¦  G  2   $R  FQ¦¦     J   ¦
041724   * 000000 * 000030 * 160000 * 000000 ¦       X 53X    ¦¦      '  ¦
041734   * 177400 * 002347 * 051522 * 030530 ¦ 2   10 MMJ G52¦¦  g RSX1¦
041744   * 046461 * 046102 * 032462 * 000000 ¦LNA LHB HTR     ¦¦1MBL25  ¦
041754   * 000000 * 044506   042014   000004 ¦     K.8 J5T   D¦¦  FI D  ¦
041764   * 000000 * 000225 * 115764 * 000000 ¦     C/ X8D    ¦¦     t   ¦
041774   * 000003 * 117074 * 041764 * 000240 ¦  C YL. J46  D ¦¦  < tC   ¦
042004   * 115264 * 000000 * 000003 * 117074 ¦X0D        C YL.¦¦4      <  ¦
042014     042064   000004 + 042264 + 000044 ¦J6T   D J9.   6¦¦4D   4D$ ¦
042024   + 000020 + 000000 + 040212 + 001010 ¦  P     JMB  M ¦¦      @   ¦
042034   + 121200 + 002111 + 140000 + 000000 ¦Z    $Q 0.2    ¦¦ "I  @   ¦
042044   + 000020 + 012054 + 001350 + 000000 ¦  P CID  RX     ¦¦  , h    ¦
042054   + 000553 + 001016 + 000000 + 000000 ¦ IC  MF        ¦¦k        ¦
042064     042240   000034   115264   007422 ¦J9H   . X0D BPR¦¦ D  4    ¦
042074     000000   000000   100000   115306 ¦       TSH X0V¦¦      F   ¦
042104     042240   000014   000000   000000 ¦J9H   L       ¦¦ D       ¦
042114     000003   000011 + 042020 + 000044 ¦  C   I J5X   6¦¦       D$ ¦
042124   + 000020 + 000000 + 041016 + 000400 ¦  P     JV0  FP¦¦      B   ¦
042134   + 120426 + 001376 + 140026 + 000000 ¦Y08  SF 0/N    ¦¦ !"  @   ¦
042144   + 001412 + 140047 + 000003 + 000000 ¦ SR 0/1   C    ¦¦  'P     ¦
042154   + 000000 + 000000 + 000000 + 041260 ¦              JZ2¦¦       0B¦
042164   * 000000 * 000001 * 000001 * 000000 ¦      A    A   ¦¦         ¦
042174   * 000401 * 000000 * 000000 * 000000 ¦ FQ           ¦¦         ¦
042204   * 000003 * 000000 * 000000 * 000000 ¦  C           ¦¦         ¦
042214   * 002112 * 000401 * 000000 * 000000 ¦ $R  FQ       ¦¦J        ¦
042224   * 000000 * 000000 * 000000 * 000001 ¦             A¦¦        ¦
042234   * 000000 * 122022   042330   000024 ¦     ZJB K X   T¦¦   $XD   ¦
042244     115264   005374   000000   000000 ¦X0D A0L        ¦¦4 |      ¦
042254     100000   115306   042760   000004 ¦TSH X0V KGX    D¦¦  F pE  ¦
042264   + 042504 + 002114 + 140054 + 000000 ¦KCL  $T 0/6    ¦¦DEL ,@   ¦
042274   + 040212 + 001000 + 120636 + 001400 ¦JMB  L2 Y4N  SH¦¦ @   !   ¦
042304   + 140036 + 000000 + 000000 + 141750 ¦0/V          1M2¦¦ @    hC¦
042314   + 001000 + 000000 + 000040 + 000025 ¦ L2        2   U¦¦       ¦
042324   + 000000 + 000020   042550   000154 ¦       P KDH  B.¦¦    hE1 ¦
042334     050061   020105   030066   000415 ¦L3I EFU G.V  F/¦¦1PE 60  ¦
042344     170017   120374   120252   000000 ¦8P0 Y0L Y.J    ¦¦ p! *    ¦
042354     000000   120712   000016   000000 ¦     Y5R   N   ¦¦  J!     ¦
042364     000000   000000   000000   000000 ¦             ¦¦         ¦
042374     042436   000001   000000   000000 ¦KBN   A       ¦¦ E       ¦
042404     123550   000000   000375   000000 ¦Z1P       FM   ¦¦h' }     ¦
042414     000000   000000   000000   042500 ¦             KCH¦¦      @E¦
042424     000002   040212   000000   041354 ¦  B JMB      J.L¦¦   @  1B¦
042434     000000   000001   117074   120000 ¦       A YL. YX ¦¦    <    ¦
042444     130477   041764   000105   000000 ¦.NO J46  A/     ¦¦?1tCE    ¦
```

Figure 3-19 Pool Dump (Truncated)

```
RSX-11M-PLUS CRASH DUMP ANALYZER    V3.0    2-MAY-85   13:55        PAGE 1
SECONDARY POOL DUMP

                    S E C O N D A R Y   P O O L

    NUMBER OF FREE BYTES = 00037500      FIRST FREE BYTE = 00656700

    LENGTH = 00120000 BYTES

    START ADDRESS = 00625000             ENDING ADDRESS = 00627500

  00625000    000000  000101  000000  131574    |     AY      ...||  A   |3|
  00625010    003273  000000  140012  000000    |ACC     0/B     ||,    0  |
  00625020    140016  002040  000000  000000    |0/F  ZP          || 0      |
  00625030    006314  100000  020000  050000    |BA6 T8H ED2 L2 ||L       P|
  00625040    001101  041072  042334  113634    | NQ JW4 K . XJL||A !B\D   |
  00625050    000415  000000  000000  140054    | F/           0/6||      ,0|
  00625060    000010  000002  000000  000000    |  H   B          ||        |
  00625070    000000  000143  000002  000000    |      BS    B    ||  c     |
  00625100    000000  140100  000000  140104    |     00P         00T||  00  D0|
  00625110    000000  000000  000000  000000    |                    ||      |
  00625120    000000  000000  000000  000000    |                    ||      |
  00625130    000000  000000  000000  000000    |                    ||      |
  00625140    000000  000000  000000  000000    |                    ||      |
  00625150    000000  000000  000000  000000    |                    ||      |
  00625160    000000  000000  000000  000000    |                    ||      |
  00625170    000000  000000  000000  000000    |                    ||      |
  00625200    000000  000106  000000  131574    |     A0      ...||  F   |3|
  00625210    003313  000000  140012  000000    |ACS     0/B     ||K    0  |
  00625220    140016  001660  000000  000000    |0/F  WX          || 00     |
  00625230    006651  100000  020000  050000    |BGQ TSH ED2 L2 ||)       P|
  00625240    000506  002346  042334  113634    | HF  1N K . XJL||F ! \D   |
  00625250    000164  000000  000000  140054    | B6           0/6||t     ,0|
  00625260    000010  000002  000000  000000    |  H   B          ||        |
  00625270    000000  000162  000002  000000    |      B4    B    ||  r     |
  00625300    000000  140100  000000  140104    |     00P         00T||  00  D0|
  00625310    000000  000000  000000  000000    |                    ||      |
  00625320    000000  000000  000000  000000    |                    ||      |
  00625330    000000  000000  000000  000000    |                    ||      |
  00625340    000000  000000  000000  000000    |                    ||      |
  00625350    000000  000000  000000  000000    |                    ||      |
  00625360    000000  000000  000000  000000    |                    ||      |
  00625370    000000  000000  000000  000000    |                    ||      |
  00625400    000000  141504  141510  000044    |     1I. 1I2    6||  DCHC$ |
  00625410    000120  141504  000000  000000    | B   1I.          ||P DC    |
  00625420    022126  000000  000000  142514    |E2V             1V.||V$    LE|
  00625430    000000  000000  000400  000072    |                FP  AR||     : |
  00625440    000000  022126  000000  047517    |     E2V         L$W||  VS  OO|
  00625450    053000  015563  071517  047433    |M0P DO$ RQ9 LZK||  Vs  0s  O|
  00625460    047563  047563  047560  047560    |L.S L.S L.P L.P||s0s0p0p0|
  00625470    047520  077560  077577  077577    |L$X TOX T09 T09||P0p      |
  00625500    077577  077577  077564  077577    |T09 T09 T0. T09||     t   |
  00625510    077541  077577  077577  077577    |TOI T09 T09 T09||a        |
  00625520    077577  077577  077577  077577    |T09 T09 T09 T09||         |
  00625530    077577  077577  077577  077577    |T09 T09 T09 T09||         |
```

Figure 3-19 (Cont.) Pool Dump (Truncated)

3.2.10 Task Dump

CDA prints all or a portion of the task's virtual address space if the /TASK switch is specified. Figure 3-20 and the following list illustrate this output.

Item	Description
1.	Task name
2.	Address of the first word of the Task Control Block for the task
3.	Address of the first word of the task's header

WINDOW BLOCKS

4.	Name of the partition to which the task is mapped
5.	Task virtual address limits
6.	Address of the attachment descriptor
7.	Size of window in 32-word blocks
8.	Offset to memory region within partition in 32-word blocks
9.	First Page Description Register (PDR) used to map the task
10.	Number of PDRs used to map task
11.	Contents of the last PDR used to map the task
12.	Task virtual address limits
13.	Physical starting address of the memory region being dumped
14.	Dump of the data within the window, formatted in octal, Radix-50, and ASCII

```
RSX-11M CRASH DUMP ANALYZER      V4.2     2-MAY-85      13:52       PAGE 1
TASK DUMP

                          TASK DUMP OF ...MCR (1)
                          -----------------

                    (2)                              (3)
          TCB ADDRESS = 115264          HEADER ADDRESS = 042760

WINDOW BLOCKS:
  (4)          (5)            (6)        (7)        (8)       (9)      (10)     (11)
  PAR      VIRT LIMITS      ATT DESC   WND SIZE   OFFSET    1ST PDR    NO.    LAST PDR
  ---      -----------      --------   --------   ------    -------    ---    --------
SYSPAR   120000 130477       042000     000105     000000    177612     1       042006

WINDOW #1 -- TASK VIRTUAL LIMITS 120000-130477 (12)
-----------------------------------------------

  PHYSICAL STARTING ADDRESS = 00137200 (13)
  ------------------------------------

  120000    120362   000162   040000   115310    ¦Y0B  B4 JIX X0X¦¦r r  @H ¦
  120010    000424   000424   170017   120636    ¦ F6  F6 8PO Y4N¦¦    p ¦¦
  120020    120362   000000   000000   000000    ¦Y0B            ¦¦r       ¦
  120030    000000   000000   000000   000000    ¦               ¦¦        ¦
  120040    000000   000000   043076   000001    ¦          KIV   A¦¦    >F  ¦
  120050    000000   000000   121176   000000    ¦          Y98    ¦¦    ~"  ¦
  120060    000004   000000   000000   000000    ¦  D              ¦¦        ¦
  120070    000000   043140   000004   041260    ¦     KJP    D JZ2¦¦   'F  0B¦
  120100    000000   041400   000000   041400    ¦     J.2      J.2¦¦    C   C¦
  120110    000000   041354   000000   000001    ¦     J.L        A¦¦   1B   ¦
  120120    117074   120000   130477   042000    ¦YL.  YX  .NO J5H¦¦<    ?1 D¦
  120130    000105   000000   000612   042006    ¦ A/        I4 J5N¦¦E       D¦
  120140    170000   122360   000000   000000    ¦8P  ZOX         ¦¦ pp$     ¦
  120150    120426   000070   120472   000000    ¦Y08  AP Y14     ¦¦ 18 ¦¦   ¦
  120160    000000   000000   000000   000000    ¦               ¦¦        ¦

                    [ABOVE LINE REPEATED 10. TIMES]

  120310    000000   000000   006001   001010    ¦          A63  M ¦¦      ¦
  120320    000004   000037   121202   000000    ¦  D    1 Z B     ¦¦     "  ¦
  120330    121634   004130   000000   000001    ¦ZGD AMP           A¦¦ #X    ¦
  120340    125512   060100   120515   000000    ¦$PJ  OP  Y2M      ¦¦J+@"M¦  ¦
  120350    121232   001453   001047   125040    ¦Z Z   TK   M1 $H2¦¦ "+ '  *¦
```

(14)

Figure 3-20 Task Dump (Truncated)

3.2.11 Clock Queue

The example in Figure 3-21 shows a clock queue listing. The following list explains the example.

Item **Description**

1. Address of the clock queue entry

2. Types of time schedule requests
   One of the following types:

   Type 0 - Mark time request

   Type 2 - Request with periodic rescheduling

   Type 4 - Single-shot task request

   Type 6 - Single-shot internal system subroutine with system subroutine identification

   Type 10- Single-shot internal system subroutine without system subroutine identification

   Type 12- Clear stop bit (Shuffler)

3. Task Control Block address or system subroutine identification

4. Task issuing the clock request

5. The hour, minute, and second that time request comes due

6. This field varies with each type of time schedule request

   For a Mark Time request, the labels are:

   C.AST - AST address

   C.SRC - Event flag mask word

   C.DST - Event flag mask address

   Event Flag Number

   For a periodic rescheduling request, the labels are:

   C.RSI - Reschedule internal

   C.UIC - Scheduling UIC

   The field for a single-shot task request contains only one label:

   C.UIC - Scheduling UIC

   The field for a single-shot internal subroutine (both with and without system subroutine identification) contains:

   C.SUB - Subroutine address

   C.AR5 - Relocation base address (for loadable drivers)

7. Octal dump of clock queue in offset mode

```
RSX-11M CRASH DUMP ANALYZER      V4.2     2-MAY-85     15:13       PAGE 116
CLOCK QUEUE

          (1)                    (2)                  (3)          (4)
ADDRESS = 054210    REQUEST TYPE = 0    TCB = 107134    TASK = TKTN

TIME REQUEST BECOMES DUE = 09:54:06.8 (5)

C.AST = 000000    C.SRC = 100000    C.DST = 107156    EVENT FLAG = 16. (6)

000000    057250  010000  107134  011374  000000  000000  100000  107156 } (7)


ADDRESS = 057250    REQUEST TYPE = 0    TCB = 053510    TASK = PMT...

TIME REQUEST BECOMES DUE = 09:54:10.2

C.AST = 124602    C.SRC = 000002    C.DST = 053532    EVENT FLAG = 2.

000000    046760  001000  053510  011707  000000  124602  000002  053532


ADDRESS = 046760    REQUEST TYPE = 0    TCB = 053510    TASK = PMT...

TIME REQUEST BECOMES DUE = 09:54:50.2

C.AST = 000000    C.SRC = 000004    C.DST = 053532    EVENT FLAG = 3.

000000    000000  001400  053510  016447  000000  000000  000004  053532
```

Figure 3-21 Clock Queue

3.2.12 Controller Information

Figure 3-22 shows the information associated with a device controller. This information appears only in crash dumps of RSX-11M-PLUS systems. The following list explains the items in Figure 3-22.

Item	Description
1.	Name of the device controller
2.	Address of the Controller Table (CTB)
3.	Address of the Device Controller Block (DCB) for this device
4.	CTB status byte, which may contain any of the following three-letter status codes: CLK - Clock block appears at the top of the CTB MDC - Multidriver CTB CBL - Clock block is linked into the clock queue CIN - Controller uses the common interrupt dispatch table NET - DECnet device
5.	Octal dump of the CTB
6.	Common Interrupt Address

Item	Description
7.	DCB for each device interfaced by this controller
8.	Name of each device interfaced by this controller
9.	Address of the Controller Request Block (KRB)
10.	Controller status. The following is a list of possible status values and their meanings: OFL - Controller is off line MOF - Controller is marked for off line UOP - Controller supports overlapped operations MBC - Device is a MASSBUS controller SDX - Seek operations allowed during data transfers POE - Parallel operations enabled UCB - UCB table present DIP - Data transfer in progress PDF - Privileged diagnostic functions only
11.	Octal dump of KRB in one or two parts

```
RSX-11M-PLUS CRASH DUMP ANALYZER  V3.0     2-MAY-85   15:08       PAGE 149
CONTROLLER INFORMATION


  RH (1)
  --
             (2)                  (3)
   CTB ADDRESS = 020454      L.DCB = 022020
(4) L.STS:  MDC CIN
   CONTROLLER TABLE BLOCK

   020454    022200  044122  022020  005004  022044  022102  022134  022160 } (5)


   DEVICES INTERFACED BY THIS CONTROLLER
   COMMON INTERRUPT ADDRESS = 017526 (6)
  (7) DCB     DEVICE NAME (8)
      ---     -----------
   020474        DB
   021142        DR
   021410        DS
   021616        MM


   KRB ADDRESS = 022044 (9)
   K.STS:  SDX UOP MBC POE UCB DIP (10)

   022026    021410  021616  000000  177777  025640  000400  000374  176700
   022046    000016  000003  020534  021112  021112  000050  020534  020602
   022066    020650  020716  177777  015240                                  } (11)

   022046    000016  000003  020534  021112  021112  000050  020534  020602
   022066    020650  020716


   KRB ADDRESS = 022102
   K.STS:  SDX UOP MBC POE UCB

   022064    020602  020650  020716  177777  015240  000002  000174  176300
   022104    000016  000001  000000  000000  022112  000050  021202  021252
   022124    177777  020640  000004  000030

   022104    000016  000001  000000  000000  022112  000050  021202  021252
```

Figure 3-22 Controller Information

```
KRB ADDRESS = 022134
K.STS:  SDX MBC

022116    000050  021202  021252  177777  020640  000004  000030  172040
022136    000016  000001  021516  000000  022144  000030  022640  000006
022156    000030  172440  000016  000000

022136    000016  000001  021516  000000  022144  000030  022640  000006


KRB ADDRESS = 022160
K.STS:  SDX MBC

022142    021516  000000  022144  000030  022640  000006  000030  172440
022162    000016  000000  000000  000000  022170  000034  004270  022454
022202    045504  022212  000001  022372

022162    000016  000000  000000  000000  022170  000034  004270
```

Figure 3-22 (Cont.) Controller Information

3.2.13 Kernel Data Space

Figure 3-23 shows a dump of kernel data space from the specified starting virtual address to the specified ending virtual address.

3.2.14 Kernel Instruction Space

Figure 3-24 shows a dump of kernel instruction space from the specified starting virtual address to the specified ending virtual address.

3.2.15 Task Data Space

Figure 3-25 shows a dump of task data space. This dump occurs only on RSX-11M-PLUS systems.

3.2.16 Task Instruction Space

Figure 3-26 shows a dump of task instruction space. This dump occurs only on RSX-11M-PLUS systems.

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0    2-MAY-85    12:51       PAGE 6
KERNEL DATA SPACE DUMP



        VIRTUAL ADDRESS: 001000      PHYSICAL ADDRESS: 00001000


    001000    000000  000000  000000  000000  ¦                ¦¦         ¦

                    [ABOVE LINE REPEATED 28. TIMES]

    001350    000000  000000  020422  000000  ¦          EKZ     ¦¦      ¦  ¦
    001360    177400  000000  000000  020416  ¦ 2              EKV¦¦        ¦¦
    001370    005015  025052  042452  042530  ¦AXM F/4 KBZ KC2¦¦  ***EXE¦
    001400    020103  040520  044522  054524  ¦EFS JR  K/J NK.¦¦C PARITY¦
    001410    042440  051122  051117  051440  ¦KBP MF4 MF1 ML ¦¦ ERROR S¦
    001420    047524  025120  025052  005015  ¦L$. F02 F/4 AXM¦¦TOP***  ¦
    001430    000012  177777  172020  003110  ¦  J  80 9B  A H¦¦      tH ¦
    001440    160377  177401  000366  000000  ¦599  2A  FF    ¦¦ '  v    ¦
    001450    000001  001450  001450  001450  ¦  A  TH  TH  TH¦¦  (  (  (  ¦
    001460    001450  001450  001450  001450  ¦ TH  TH  TH  TH¦¦( (  (  (  ¦
    001470    001450  001450  001450  001450  ¦ TH  TH  TH  TH¦¦( (  (  (  ¦
    001500    001450  001450  001450  001450  ¦ TH  TH  TH  TH¦¦( (  (  (  ¦
    001520    177746  177750  177752  177766  ¦ 70  72  74  8F¦¦f h  j  v  ¦
    001530    000000  000000  000000  000000  ¦                ¦¦         ¦
    001540    000000  000000  050044  000000  ¦          L26   ¦¦    $P   ¦
    001550    120030  000000  000000  000001  ¦YXX              A¦¦         ¦
    001560    000001  042220  005015  041412  ¦  A J82 AXM J/B¦¦   D    C¦
    001570    040522  044123  026440  020055  ¦JRB KV5 GH2 EE7¦¦RASH -- ¦
    001600    047503  052116  053440  052111  ¦L$K MSV M7X MSQ¦¦CONT WIT¦
    001610    020110  041523  040522  041524  ¦EFX J05 JRB J06¦¦H SCRATC¦
    001620    020110  042515  044504  020101  ¦EFX KCU K.6 EFQ¦¦H MEDIA ¦
    001630    047117  042040  030113  005015  ¦LUG J6  G/C AXM¦¦ON DK0  ¦
    001640    000012  005015  051103  051501  ¦  J AXM MFS ML3¦¦    CRAS¦
    001650    020110  026455  044440  020117  ¦EFX GIE K.  EF1¦¦H -- IO ¦
    001660    051105  047522  020122  047117  ¦MFU L$Z EF4 LUG¦¦ERROR ON¦
    001670    041440  040522  044123  042040  ¦J/X JRB KV5 J6 ¦¦ CRASH D¦
    001700    046525  020120  042504  044526  ¦LN7 EF2 KCL K/N¦¦UMP DEVI¦
    001710    042503  006400  041412  040522  ¦KCK BCH J/B JRB¦¦CE   CRA¦
    001720    044123  026440  020055  040523  ¦KV5 GH2 EE7 JRC¦¦SH -- SA¦
    001730    044516  054524  052040  046511  ¦K/F NK. MRP LNY¦¦NITY TIM¦
    001740    051105  042440  050130  051111  ¦MFU KBP L4H MFY¦¦ER EXPIR¦
    001750    042105  047440  020116  051120  ¦J67 LZP EF0 MF2¦¦ED ON PR¦
    001760    041517  051505  047523  020122  ¦J01 ML7 L$$ EF4¦¦OCESSOR ¦
    001770    050103  006400  041412  040522  ¦L3$ BCH J/B JRB¦¦CP   CRA¦
    002000    044123  026440  000003  116301  ¦KV5 GH2   C YCI¦¦SH -  A ¦



[END OF ANALYSIS OUTPUT]
```

Figure 3-23 Kernel Data Space

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0    2-MAY-85    12:51       PAGE 5
KERNEL INSTRUCTION SPACE DUMP



        VIRTUAL ADDRESS:  001000      PHYSICAL ADDRESS:  00001000


001000    000000  000000  000000  000000  ¦                ¦¦        ¦

                  [ABOVE LINE REPEATED 28. TIMES]

001350    000000  000000  020422  000000  ¦         EKZ    ¦¦     ¦  ¦
001360    177400  000000  000000  020416  ¦ 2          EKV¦¦        ¦¦
001370    005015  025052  042452  042530  ¦AXM F/4 KBZ KC2¦¦  ***EXE¦
001400    020103  040520  044522  054524  ¦EFS JR  K/J NK.¦¦C PARITY¦
001410    042440  051122  051117  051440  ¦KBP MF4 MF1 ML ¦¦ ERROR S¦
001420    047524  025120  025052  005015  ¦L$. F02 F/4 AXM¦¦TOP***  ¦
001430    000012  177777  172020  003110  ¦  J  80 9B  A H¦¦     tH ¦
001440    160377  177401  000366  000000  ¦599  2A  FF    ¦¦ ^  v   ¦
001450    000001  001450  001450  001450  ¦  A  TH  TH  TH¦¦  ( ( ( ¦
001460    001450  001450  001450  001450  ¦ TH  TH  TH  TH¦¦( ( ( ( ¦
001470    001450  001450  001450  001450  ¦ TH  TH  TH  TH¦¦( ( ( ( ¦
001500    001450  001450  001450  001450  ¦ TH  TH  TH  TH¦¦( ( ( ( ¦
001520    177746  177750  177752  177766  ¦ 70  72  74  8F¦¦f h j v ¦
001530    000000  000000  000000  000000  ¦              ¦¦       ¦
001540    000000  000000  050044  000000  ¦         L26  ¦¦    $P  ¦
001550    120030  000000  000000  000001  ¦YXX           A¦¦       ¦
001560    000001  042220  005015  041412  ¦  A J82 AXM J/B¦¦   D   C¦
001570    040522  044123  026440  020055  ¦JRB KV5 GH2 EE7¦¦RASH -- ¦
001600    047503  052116  053440  052111  ¦L$K MSV M7X MSQ¦¦CONT WIT¦
001610    020110  041523  040522  041524  ¦EFX J05 JRB J06¦¦H SCRATC¦
001620    020110  042515  044504  020101  ¦EFX KCU K.6 EFQ¦¦H MEDIA ¦
001630    047117  042040  030113  005015  ¦LUG J6  G/C AXM¦¦ON DK0  ¦
001640    000012  005015  051103  051501  ¦  J AXM MFS ML3¦¦    CRAS¦
001650    020110  026455  044440  020117  ¦EFX GIE K.  EF1¦¦H -- IO ¦
001660    051105  047522  020122  047117  ¦MFU L$Z EF4 LUG¦¦ERROR ON¦
001670    041440  040522  044123  042040  ¦J/X JRB KV5 J6 ¦¦ CRASH D¦
001700    046525  020120  042504  044526  ¦LN7 EF2 KCL K/N¦¦UMP DEVI¦
001710    042503  006400  041412  040522  ¦KCK BCH J/B JRB¦¦CE   CRA¦
001720    044123  026440  020055  040523  ¦KV5 GH2 EE7 JRC¦¦SH -- SA¦
001730    044516  054524  052040  046511  ¦K/F NK. MRP LNY¦¦NITY TIM¦
001740    051105  042440  050130  051111  ¦MFU KBP L4H MFY¦¦ER EXPIR¦
001750    042105  047440  020116  051120  ¦J67 LZP EF0 MF2¦¦ED ON PR¦
001760    041517  051505  047523  020122  ¦J01 ML7 L$$ EF4¦¦OCESSOR ¦
001770    050103  006400  041412  040522  ¦L3$ BCH J/B JRB¦¦CP   CRA¦
002000    044123  026440  000003  116301  ¦KV5 GH2   C YCI¦¦SH -  A ¦
```

Figure 3-24 Kernel Instruction Space

```
RSX-11M-PLUS CRASH DUMP ANALYZER    V3.0     2-MAY-85   16:13        PAGE 1

                        TASK DUMP OF LITT27
                        -------------------


          D A T A   S P A C E
          -------------------

           TCB ADDRESS = 042204        HEADER ADDRESS = 011323


 WINDOW BLOCKS:

  PAR      VIRT LIMITS    ATT DESC  WND SIZE  OFFSET  1ST PDR  NO.  LAST PDR
  ---      -----------    --------  --------  ------  -------  ---  --------
 GEN      000000 002077   040030    000021    000015  177620    1    010006
          160000 160177   041040    000002    000010  177636    1    000402


 WINDOW #2 -- TASK VIRTUAL LIMITS 000000-002077
 ----------------------------------------------

   PHYSICAL STARTING ADDRESS = 01134000
   ------------------------------------

   000000    000000  000252  140313  160400    ¦     DJ 035 6  ¦¦  * K@ a¦
   000010    003471  003471  170017  160000    ¦AFI AFI 8PO 53X¦¦9 9  p @¦
   000020    001334  000000  000000  000000    ¦ RL              ¦¦\      ¦
   000030    000000  000000  000000  000000    ¦                 ¦¦       ¦
   000040    000000  140252  140126  000001    ¦     03B 008     A¦¦  *@V@  ¦
   000050    000000  000000  000000  000000    ¦                 ¦¦       ¦
   000060    000000  000000  000000  000000    ¦                 ¦¦       ¦
   000070    000000  140250  000006  022520    ¦     03     F E82¦¦  (@  P%¦
   000100    000000  022520  000000  022520    ¦     E82      E82¦¦  P%  P%¦
   000110    000000  022520  000000  037462    ¦     E82      JDR¦¦  P%  2?¦
   000120    000000  037506  000000  000004    ¦     JD8        D¦¦  F?    ¦
   000130    107544  000000  001077  000000    ¦V7.          NO  ¦¦d   ?   ¦
   000140    000011  000004  000600  004006    ¦  I    D   IX AKN¦¦       ¦
   000150    107544  000000  002077  000000    ¦V7.          $G  ¦¦d   ?   ¦
   000160    000021  000021  000620  010006    ¦  Q    Q   J  BVV¦¦       ¦
   000170    040144  160000  160577  000000    ¦JLD 53X 6CG      ¦¦d@ @ a  ¦
   000200    000006  000000  000616  002402    ¦  F        I8  2B¦¦       ¦
   000210    040144  160000  160177  000000    ¦JLD 53X 561      ¦¦d@ @ @  ¦
   000220    000002  000010  000636  000402    ¦  B    H   JN  FR¦¦       ¦
   000230    017747  000004  000000  160400    ¦EDG    D       6 ¦¦g      a¦
   000240    140313  046174  131574  000000    ¦035 LIT ...      ¦¦K@¦L¦3  ¦
   000250    000000  000000  000000  000000    ¦                 ¦¦       ¦

                    [ABOVE LINE REPEATED 63. TIMES]

      .
      .
      .
```

Figure 3-25 Task Data Space

```
RSX-11M-PLUS CRASH DUMP ANALYZER   V3.0    2-MAY-85   13:46       PAGE 1
TASK DUMP


                          TASK DUMP OF ...LDR


          INSTRUCTION   SPACE


           TCB ADDRESS = 117300        HEADER ADDRESS = 117130


 WINDOW BLOCKS:

  PAR      VIRT LIMITS    ATT DESC  WND SIZE  OFFSET  1ST PDR  NO.  LAST PDR
 SYSPAR  120000 123777    117114    000040    000000  177612    1    017406


 WINDOW #1 -- TASK VIRTUAL LIMITS 120000-123777

   PHYSICAL STARTING ADDRESS = 00530100

   120000    000000  000146  140664  140630    !      BV 09. 09 !!   ¢ 4A A!
   120010    000424  000424  170017  120226    ! F6   F6 8PO Y$0!!      p  !
   120020    120166  000000  000000  000000    !YZ8                !!v      !
   120030    000000  000000  000000  000000    !                   !!       !
   120040    000000  000000  000102  000000    !          AZ       !!    B  !
   120050    000000  000000  000000  000000    !                   !!       !
   120060    000000  000000  000000  000000    !                   !!       !
   120070    000000  000144  000001  037602    !      BT    A JFR!!   d     ?!
   120100    000000  000001  117610  120000    !        A YU   YX !!       !
   120110    123777  000000  000040  000000    !Z5G         2      !!  '     !
   120120    000612  017406  000143  000003    ! I4 D8V   BS     C!!     c   !
   120130    000000  140630  140664  045662    !    09   09. LDR!!    A4A2K!
   120140    131574  000000  000000  000000    !...                !!!3     !
   120150    000000  000000  000000  140630    !                09 !!      A!
   120160    000002  120166  102700  006003    !  B YZ8 UP   A65!!   v ¢    !
   120170    001000  000001  000001  120214    ! L2   A    A Y$T!!         !
   120200    000000  140000  101700  000000    !     0.2 UCH     !!   ¢¢    !
   120210    000000  000000  047660  000000    !         L0      !!    00   !
   120220    000000  000000  031401  104376    !         HFQ U60!!     3"  !
   120230    120454  005067  100570  016700    !Y1T AYO T.X D0P!!.!7 x ¢  !
   120240    061024  062700  000012  004767    !O$. PKH    J AW!!! b¢e  w  !
     .
     .
     .
```

Figure 3-26 Task Instruction Space

CHAPTER 4

INTERPRETING A CRASH DUMP LISTING

This chapter introduces basic concepts that can help you to use CDA output listings to analyze the cause of a system crash. However, this chapter is not intended as a complete guide to interpreting a crash dump.

4.1 HELPFUL CONCEPTS

Two concepts are helpful in using CDA output listings to determine the cause of a system failure:

- Determining what was mapped at the time of the crash
- Interpreting stack depth and the kernel stack

4.1.1 Determining What Was Mapped

To determine what was mapped at the time of the crash, look at the dump of the Kernel Page Address Registers on the first page of the CDA listing (instruction space registers on RSX-11M systems, both instruction and data space registers on RSX-11M-PLUS systems). This listing, titled Volatile Registers, is shown in Figure 3-1.

Figure 4-1 is an example of the listing of the contents of the sixth and seventh words from a Kernel Page Address Registers dump. The contents of the sixth and seventh words are the block numbers of the task or driver that was mapped at the time of the crash. You can determine what occupied that portion of memory from the memory map on the first page of partition information in the CDA output listing. Look under the BASE heading (for the base address of the mapped partition).

You can use this information, along with data from the kernel stack and system common listings, to analyze the state of the system at the time of the crash.

K E R N E L

I S P A C E		D S P A C E	
PDR	PAR	PDR	PAR
077506	000000	000000	000000
077506	000200	000000	000000
077506	000400	000000	000000
077406	000600	000000	000000
077506	001000	000000	000000
077406	001372	000000	000000
077406	002077	000000	000000
077506	007600	000000	000000

Figure 4-1 Kernel Page Address Registers

4.1.2 Interpreting the Kernel Stack

$STKDP, which appears on the first page of the system common listing, is the system stack depth indicator. The value of the stack depth indicator shows which state the system was in when it crashed. The following lists shows the possible values of the stack depth indicator and the corresponding system states.

```
$STKDP = 1                Indicates User state
$STKDP = 0                Indicates System state
$STKDP = -1,-2,-3,-4,...  Indicates Interrupt state
```

Note that when the system is in Interrupt state, the stack depth value is negative. Two types of interrupt conditions can decrement stack depth:

1. A synchronous system trap (SST), which can be caused by any of the following:
   - A directive
   - A TRAP instruction
   - An illegal instruction
   - Another SST
2. The interrupt save routine ($INTSV)

When an SST occurs, the following information from the current task is pushed onto the stack:

```
The Processor Status Word (PSW)
The Program Counter (PC)
The address of return to DIRSV
SST specific information
R5  (Mapped systems only)
R4  (Mapped systems only)
R3
R2
R1
R0
```

When an interrupt occurs, the following are pushed onto the stack:

PSW
PC

If a call to $INTSV is then issued, the following are pushed onto the stack after the PC:

Address of return to $INTSV
R5
R4

The stack depth is then decremented. If the value of $STKDP is 0 and the currently mapped driver issues a call to the $FORK routine, the following items are pushed onto the stack:

Return to $FORK
R3 (Which replaces the return to $INTSV)
R2
R1
R0

If the driver issues a call to $FORK when $STKDP is not 0, the registers are saved in a fork block, which is queued for later execution.

Since interrupts can still occur, more of both basic types of stack contents (interrupt or $INTSV) can be pushed. If an SST occurs in the Executive, SST information will be pushed onto the stack, possibly followed by a system crash.

This information, along with the kernel stack pointer SP(K), which appears on the first page of the CDA listing, can be used to interpret the contents of the dump of the kernel stack.

The contents of the PC and the PSW before the crash appear on the first page of the CDA listing. You can compare this information to the contents of the PC and the PSW on the stack to help locate the cause of the crash.

Refer to the RSX-11M Guide to Writing an I/O Driver or to the RSX-11M-PLUS Guide to Writing an I/O Driver for a further description of the contents of the kernel stack.

APPENDIX A

CDA MESSAGES

CDA displays a message on your terminal when it detects one of the error conditions described below. These messages reflect operational conditions. Do not confuse these messages with the diagnostic analysis messages that CDA generates during the analysis and prints in the analysis listing.

Not all of the messages listed below terminate CDA analysis. Some are simply informational messages, and others warn you of nonfatal errors. Also, this list includes some ANALYZE/CRASH_DUMP command error messages.

ANALYZE -- Illegal crash input specification

Type: Fatal

Explanation: You must include a crash-input device or file in the command line. Also, you must use the correct command line syntax when you specify the crash input.

ANALYZE -- Illegal crash output specification

Type: Fatal

Explanation: You must specify at least one output file or device specification. Also, you must use the correct command line syntax when you specify the crash output.

ANALYZE -- Illegal, contradictory, or ambiguous qualifiers

Type: Fatal

Explanation: You must use qualifiers with the correct file or device specification. Refer to the qualifier tables in Chapter 2 if you are unsure which file or device specification a qualifier modifies. Also, when you specify qualifier arguments, be sure to enclose them in parentheses. If you specify more than one argument for the same qualifier, separate the arguments with commas.

ANALYZE -- Sorry, task not installed

Type: Fatal

Explanation: CDA is not presently installed in the system. If you are a nonprivileged user, refer to Chapter 1 for an explanation of how you can run CDA as an uninstalled task, or ask a privileged user (such as your system manager) to install CDA.

If you are a privileged user, you can install CDA and then reenter your command line.

CDA -- ACP out of memory or not in execution

Type: Analysis diagnostic

Explanation: The partition containing the File Control Block (FCB) for the current logical unit number (LUN) was not in memory.

CDA -- Address out of range

Type: Fatal

Explanation: CDA was unable to read a block from the crash-input file. Possible causes for this are:

- A device failure
- A bad block on the volume
- The crashed system had a corrupted data base
- The binary file does not contain all of the crashed system's memory

CDA -- Analysis output must be directed to an explicit device or file

Type: Fatal

Explanation: CDA requires an explicit output file specification. There are no default output file names.

CDA -- Analysis terminated after n. pages

Type: Informational

Explanation: CDA terminated the analysis after generating n pages of analysis output. If you have not specified the /LIMIT switch in the CDA command string, this message indicates that CDA has generated more than 300(decimal) pages of output.

CDA -- Command I/O error

Type: Fatal

Explanation: The system returned an error when CDA attempted to read a command line.

CDA -- Command line syntax error

Type: Fatal

Explanation: CDA detected an error in the syntax of a CDA command line. CDA will point to the beginning of the error within the command line.

CDA -- Crash dump must be input from an explicit device or file

Type: Fatal

Explanation: The crash dump input file specification must be explicit. There is no default file specification for the crash dump input.

CDA -- Device driver missing

Type: Fatal

Explanation: You have not loaded the driver for the crash dump input device.

CDA -- Dump aborted - kernel PARs clobbered

Type: Fatal

Explanation: This message appears on mapped systems only. It indicates that the values contained in the kernel Page Address Registers are invalid. To restart the analysis, you must specify the /KMR switch. This switch forces CDA to use standard mapping values when converting kernel virtual addresses to physical memory addresses.

CDA -- Error reading crash dump

Type: Fatal

Explanation: The system returned an error when CDA attempted to read the crash dump file. This could be caused by:

- A device failure
- A bad block on the volume
- On RSX-11M-PLUS systems, the device might not be mounted foreign

CDA -- Error reading file filename

Type: Fatal

Explanation: The system returned an error when CDA attempted to read the crash dump file. This could be caused by:

- A device failure
- A bad block on the volume

CDA -- Error reading symbol file filename

Type: Fatal

Explanation: The system returned an error when CDA attempted to read the symbol table file indicated. Possible causes for the error are:

- A device failure
- A bad block on the volume
- The specified symbol file was not an STB file

CDA -- Errors detected: n.

Type: Informational

Explanation: CDA has detected n analysis errors during the run.

CDA -- Error writing analysis file

Type: Fatal

Explanation: The system returned an error when CDA attempted to write a line into the analysis listing file. This could be caused by:

- A full volume
- A problem with the device
- A bad block on the volume

CDA -- Error writing dump file filename

Type: Fatal

Explanation: The system returned an error when CDA attempted to write into the binary output file. This condition could be caused by:

- A full volume
- A problem with the device
- A bad block on the volume

CDA -- Exiting due to illegal trap - snapshot dump being attempted

Type: Fatal

Explanation: CDA has aborted after detecting an odd address or some other type of fault. If Postmortem Dump (PMD) is installed in the system, the system will generate a snapshot dump. This is an indication of a software problem. If you send a Software Performance Report (SPR) to DIGITAL for this type of failure, you should include any available dumps.

Also, preserve the following until your SPR is answered:

1. From the crashed system:
   - All applicable user task images
   - RSX11M.SYS
   - RSX11M.STB
   - RSXMC.MAC
   - All applicable privileged task images
   - Crash dump volume
2. From the system used for analysis:
   - RSX11M.SYS
   - RSX11M.STB
   - CDA.TSK

CDA -- Failed to assign LUN to input device ddu

Type: Fatal

Explanation: The Assign LUN (ALUN$) directive failed when CDA attempted to use it to attach the specified input device before reading the crash dump from the device. ALUN$ will fail if the device name in the CDA command line is invalid.

CDA -- Failed to extend page buffer - n. pages available

Type: Informational

Explanation: The Extend Task (EXTK$) directive failed when CDA attempted to use it to expand the page buffer. This problem will cause the analysis to take longer, but the analysis will continue with a buffer of n pages, each 256 words long.

NOTE

CDA uses the Extend Task directive to obtain additional buffering space. CDA does not use space allocated by a /INC= qualifier on the INStall command.

CDA -- Failed to open input file filename

Type: Fatal

Explanation: One of the following conditions exists:

- The specified device does not exist.
- The volume is not mounted.
- A problem exists with the device.

- The specified UFD does not exist.
- The specified file does not exist.
- You do not have read access privileges.

CDA -- Failed to open output file filename

Type: Fatal

Explanation: One of the following conditions exists:

- The specified device does not exist.
- The volume is not mounted.
- A problem exists with the device.
- The specified UFD does not exist.
- The volume is full or the device is write-protected.
- You do not have write access privilege to UFD.

CDA -- Illegal switch

Type: Fatal

Explanation: You specified an unknown switch or used a legal switch with the wrong file specification. CDA will point to the error within the command line.

CDA -- Inconsistency in dynamic storage

Type: Informational

Explanation: CDA detected an inconsistency while scanning the pool pointers. This condition could be associated with the crash. However, it may mean that you specified the wrong executive symbol table file.

CDA -- Indirect command syntax error

Type: Fatal

Explanation: The name of the indirect command file (@filename) is syntactically incorrect.

CDA -- Indirect file open failure

Type: Fatal

Explanation: CDA could not open an indirect command file specified as "@filename" in the CDA command line.

CDA -- Invalid address range

Type: Fatal

Explanation: You specified an address that was not consistent with Active Page Register (APR) mapping.

CDA -- List count expired

Type: Analysis diagnostic

Explanation: The linked list of data structures has too many entries. The list may be corrupted, or contains a loop.

CDA -- Maximum indirect file depth exceeded

Type: Fatal

Explanation: You exceeded the maximum allowable number of nested indirect command files. (CDA only permits one indirect command level.)

CDA -- No input file specified

Type: Fatal

Explanation: You did not specify a crash dump input file.

CDA -- No output file specified

Type: Fatal

Explanation: You did not specify an output file.

CDA -- Output dump filename must be explicit

Type: Fatal

Explanation: You did not specify a valid output file.

CDA -- Pool link error found - continuing

Type: Analysis diagnostic

Explanation: CDA detected a link error while scanning the pool free block pointers ($CRAVL). This condition can be associated with the crash. It can also mean that you specified the wrong Executive symbol table file. If the latter is true, the entire analysis will be meaningless and you should abort CDA.

CDA -- Premature end of dump input - filename being zero-filled

Type: Informational

Explanation: CDA reached the end of the medium (or the end-of-file mark on a magnetic tape) before the crash dump output file had been completely filled. If you expected the file to be completely filled by the dump, this condition could indicate a problem.

CDA -- Processor n failed to dump its registers

Type: Informational

Explanation: On a multiprocessor system, when the system crashes, each on-line processor is notified by an interrupt. If the processor does not respond to the interrupt (for example, if it halted or is off line), it won't dump its registers into the crash buffer. CDA notes this and prints the informational message.

CDA -- Redirect error (U.RED=0)

Type: Analysis diagnostic

Explanation: CDA detected an error in the pointer to the Unit Control Block (UCB) of a redirected unit. This condition may be associated with the cause of the crash.

CDA -- Symbol symbolname not defined in symbol file

Type: Fatal

Explanation: CDA did not find a symbol that it required for the analysis in the specified Executive symbol table file. You have probably entered the wrong file name or have mistakenly used the default file name.

CDA -- Symbol file filename has illegal format

Type: Fatal

Explanation: The specified Executive symbol table file has an improper format, probably caused by entry of the wrong file name. However, this message could also indicate a problem with the device or medium on which the file is located.

CDA -- Task 'taskname' not in memory

Type: Analysis diagnostic

Explanation: This message can be caused by two conditions:

1. You have requested a dump of a task which does not have an entry in the System Task Directory
2. The task has an entry in the System Task Directory, but it is marked out of memory.

You can verify the state of the task by examining a dump of the Task Control Blocks.

CDA -- Transfer complete - ddu may be unloaded

Type: Informational

Explanation: The transfer of the crash dump to the output file is finished; you may unload the crash dump device. This message occurs only when you have specified a binary file in the output of the command string to CDA.

CDA -- Unknown AST type

Type: Analysis diagnostic

Explanation: CDA has detected an AST block which is not one of the following valid types of ASTs:

- Unsolicited character AST
- Buffered I/O AST
- Emit status AST
- Completion AST from:

  QIO$
  MRKT$
  SPWN$
  CNCT$
  CINT$

1. Specified AST from:

   SFPA$
   SRDA$
   SRRA$
   SPFA$

CDA -- Unknown get command line error

Type: Fatal

Explanation: An unrecognized error occurred when CDA attempted to read a command line.

CDA -- $PKVAL link error at n --FWD PTR = n

Type: Analysis diagnostic

Explanation: CDA detected a link error while scanning the pool free packet list $PKVAL. This condition can be associated with the crash. It can also mean that you specified the wrong executive symbol table file.

APPENDIX B

RSX-11M SYSTEM DATA STRUCTURES AND SYMBOLIC DEFINITIONS

This appendix describes the RSX-11M system macros that supply symbolic offsets for data structures listed in Table B-1.

The data structures are defined by macros in the Executive macro library. To reference any of the data structure offsets from your code, include the macro name in an .MCALL directive and invoke the macro. For example:

```
.MCALL  DCBDF$
DCBDF$                  ;Define DCB offsets
```

NOTE

All physical offsets and bit definitions are subject to change in future releases of the operating system. Code that accesses system data structures should always use the symbolic offsets rather than the physical offsets.

The first two arguments, <:> and <=>, make all definitions global. If they are left blank, the definitions will be local.

All of these macros are in the Executive macro library LB:[1,1]EXEMC.MLB. All except ITBDF$ and MTADF$ are also in the Executive definition library LB:[1,1]EXELIB.OLB.

Table B-1
Summary of System Data Structure Macros

Macro	Arguments	Data Structures
ABODF$	<:>,<=>	Task abort and termination notification message codes
CLKDF$	<:>,<=>	Clock queue control block
DCBDF$	<:>,<=>	Device Control Block
EPKDF$	<:>,<=>	Error message block
EVNDF$	<:>,<=>	Terminal Software Architecture (TSA) event packet definitions

(Continued on next page)

Table B-1 (Cont.)
Summary of System Data Structure Macros

Macro	Arguments	Data Structures
F11DF$	<:>,<=>	Files-11 data structures (volume control block, mount list entry, file Control Block, file window block, locked block list node)
HDRDF$	<:>,<=>	Task header and window block
HWDDF$	<:>,<=>	Hardware register addresses and feature mask definitions
ITBDF$	<:>,<=>	Interrupt transfer block
LCBDF$	<:>,<=>	Logical assignment control block
MTADF$	<:>,<=>	ANSI magtape data strucures (volume set control block)
PCBDF$	<:>,<=>	Partition Control Block and attachment descriptor
PKTDF$	<:>,<=>	I/O packet, AST control block, offspring control block, group global event flag control block, and CLI parser block
SCBDF$	<:>,<=>	Status Control Block and UMR assignment block
TCBDF$	<:>,<=>	Task Control Block
UCBDF$	<:>,<=>,TTDEF	Unit Control Block

ABODF$

```
        .MACRO      ABODF$,L,B

;+
; TASK ABORT CODES
;
; NOTE: S.COAD-S.CFLT ARE ALSO SST VECTOR OFFSETS
;-

S.CACT='B'-4.                   ;TASK STILL ACTIVE
S.CEXT='B'-2.                   ;TASK EXITTED NORMALLY
S.COAD='B'0.                    ;ODD ADDRESS AND TRAPS TO 4
S.CSGF='B'2.                    ;SEGMENT FAULT
S.CBPT='B'4.                    ;BREAK POINT OR TRACE TRAP
S.CIOT='B'6.                    ;IOT INSTRUCTION
S.CILI='B'8.                    ;ILLEGAL OR RESERVED INSTRUCTION
S.CEMT='B'10.                   ;NON RSX EMT INSTRUCTION
S.CTRP='B'12.                   ;TRAP INSTRUCTION
S.CFLT='B'14.                   ;11/40 FLOATING POINT EXCEPTION
S.CSST='B'16.                   ;SST ABORT-BAD STACK
S.CAST='B'18.                   ;AST ABORT-BAD STACK
S.CABO='B'20.                   ;ABORT VIA DIRECTIVE
S.CLRF='B'22.                   ;TASK LOAD REQUEST FAILURE
S.CCRF='B'24.                   ;TASK CHECKPOINT READ FAILURE
S.IOMG='B'26.                   ;TASK EXIT WITH OUTSTANDING I/O
S.PRTY='B'28.                   ;TASK MEMORY PARITY ERROR
S.CPMD='B'30.                   ;TASK ABORTED WITH PMD REQUEST
S.CINS='B'32.                   ;TASK INSTALLED IN TWO SYSTEMS
;
; TASK TERMINATION NOTIFICATION MESSAGE CODES
;
T.NDNR='B'0                     ;DEVICE NOT READY
.NDSE='B'2                      ;DEVICE SELECT ERROR
T.NCWF='B'4                     ;CHECKPOINT WRITE FAILURE
T.NCRE='B'6                     ;CARD READER HARDWARE ERROR
T.NDMO='B'8.                    ;DISMOUNT COMPLETE
T.NUER='B'10.                   ;UNRECOVERABLE ERROR
T.NLDN='B'12.                   ;LINK DOWN (NETWORKS)
T.NLUP='B'14.                   ;LINK UP (NETWORKS)
T.NCFI='B'16.                   ;CHECKPOINT FILE INACTIVE
T.NUDE='B'18.                   ;UNRECOVERABLE DEVICE ERROR
T.NMPE='B'20.                   ;MEMORY PARITY ERROR
T.NKLF='B'22.                   ;UCODE LOADER NOT INSTALLED
T.NDEB='B'24.                   ;TASK HAS NO DEBUGGING AID
T.NRCT='B'26.                   ;CONTROL TASK NOT INSTALLED
T.NWBL='B'28.                   ;WRITE BACK CACHING DATA LOST. UNIT
                                ;WRITE LOCKED
T.NPNT='B'30.                   ;REQUIRED PARTITION NOT IN SYSTEM
T.NIOS='B'32.                   ;I/O STALLED
T.NIOR='B'34.                   ;I/O RESUMING

        .MACRO      ABODF$    X,Y
        .ENDM
        .ENDM
```

CLKDF$

```
        .MACRO      CLKDF$,L,B

;+
; CLOCK QUEUE CONTROL BLOCK OFFSET DEFINITIONS
;
; CLOCK QUEUE CONTROL BLOCK
;
; THERE ARE SIX TYPES OF CLOCK QUEUE CONTROL BLOCKS. EACH CONTROL
; BLOCK HAS THE SAME FORMAT IN THE FIRST FIVE WORDS AND DIFFERS
; IN THE REMAINING THREE.
;
; THE FOLLOWING CONTROL BLOCK TYPES ARE DEFINED:
;-

C.MRKT='B'0                     ;MARK TIME REQUEST
C.SCHD='B'2                     ;TASK REQUEST WITH PERIODIC RESCHEDULING
C.SSHT='B'4                     ;SINGLE SHOT TASK REQUEST
C.SYST='B'6                     ;SINGLE SHOT INTERNAL SYSTEM SUBROUTINE
                                ;(IDENT)
C.SYTK='B'8.                    ;SINGLE SHOT INTERNAL SYSTEM SUBROUTINE
                                ;(TASK)
C.CSTP='B'10.                   ;CLEAR STOP BIT (CONDITIONALIZED ON
                                ;SHUFFLING)
;
; CLOCK QUEUE CONTROL BLOCK TYPE INDEPENDENT OFFSET DEFINTIONS
;

        .ASECT
.=0
C.LNK:'L' .BLKW 1               ;CLOCK QUEUE THREAD WORD
C.RQT:'L' .BLKB 1               ;REQUEST TYPE
C.EFN:'L' .BLKB 1               ;EVENT FLAG NUMBER (MARK TIME ONLY)
C.TCB:'L' .BLKW 1               ;TCB ADDRESS OR SYSTEM SUBROUTINE
                                ;IDENTIFICATION
C.TIM:'L' .BLKW 2               ;ABSOLUTE TIME WHEN REQUEST COMES DUE

;
; CLOCK QUEUE CONTROL BLOCK-MARK TIME DEPENDENT OFFSET DEFINITIONS
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
C.AST:'L' .BLKW  1              ;AST ADDRESS
C.SRC:'L' .BLKW  1              ;FLAG MASK WORD FOR 'BIS' SOURCE
C.DST:'L' .BLKW  1              ;ADDRESS OF 'BIS' DESTINATION

;
; CLOCK QUEUE CONTROL BLOCK-PERIODIC RESCHEDULING DEPENDENT OFFSET
; DEFINITIONS
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
C.RSI:'L' .BLKW 2               ;RESCHEDULE INTERVAL IN CLOCK TICKS
C.UIC:'L' .BLKW 1               ;SCHEDULING UIC
```

CLKDF$ (Cont.)

```
;
; CLOCK QUEUE CONTROL BLOCK-SINGLE SHOT DEPENDENT OFFSET DEFINITIONS
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
        .BLKW   2               ;TWO UNUSED WORDS
        .BLKW   1               ;SCHEDULING UIC

;
; CLOCK QUEUE CONTROL BLOCK-SINGLE SHOT INTERNAL SUBROUTINE OFFSET
; DEFINITIONS
;
; THERE ARE TWO TYPE CODES FOR THIS TYPE OF REQUEST:'L'
;
;       TYPE 6=SINGLE SHOT INTERNAL SUBROUTINE WITH A 16-BIT VALUE AS
;              AN IDENTIFIER.
;       TYPE 8=SINGLE SHOT INTERNAL SUBROUTINE WITH A TCB ADDRESS AS
;              AN IDENTIFIER.
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
C.SUB:'L' .BLKW 1               ;SUBROUTINE ADDRESS
C.AR5:'L' .BLKW 1               ;RELOCATION BASE (FOR LOADABLE DRIVERS)
          .BLKW 1               ;ONE UNUSED WORD
C.LGTH='B'.                     ;LENGTH OF CLOCK QUEUE CONTROL BLOCK
          .PSECT

          .MACRO        CLKDF$ X,Y
          .ENDM
          .ENDM
```

DCBDF$

DCBDF

```
      .MACRO      DCBDF$

      ;
      ; DEVICE CONTROL BLOCK
      ;
      ; THE DEVICE CONTROL BLOCK (DCB) DEFINES GENERIC
      ; INFORMATION ABOUT A DEVICE TYPE AND THE LOWEST AND
      ; HIGHEST UNIT NUMBERS. THERE IS AT LEAST ONE DCB FOR
      ; EACH DEVICE TYPE IN A SYSTEM. FOR EXAMPLE, IF THERE
      ; ARE TELETYPES IN A SYSTEM, THEN THERE IS AT LEAST ONE
      ; DCB WITH THE DEVICE NAME 'TT'. IF PART OF THE
      ; TELETYPES WERE INTERFACED VIA DL11-A'S AND THE REST
      ; VIA A DH11, THEN THERE WOULD BE TWO DCB'S. ONE FOR
      ; ALL DL11-A INTERFACED TELETYPES, AND ONE FOR ALL DH11
      ; INTERFACED TELETYPES.
      ;
            .ASECT
      .=0
000000  D.LNK:   .BLKW  1  ;LINK TO NEXT DCB
000002  D.UCB:   .BLKW  1  ;POINTER TO FIRST UNIT CONTROL BLOCK
000004  D.NAM:   .BLKW  1  ;GENERIC DEVICE NAME
000006  D.UNIT:  .BLKB  1  ;LOWEST UNIT NUMBER COVERED BY THIS DCB
000007           .BLKB  1  ;HIGHEST UNIT NUMBER COVERED BY THIS DCB
000010  D.UCBL:  .BLKW  1  ;LENGTH OF UNIT CONTROL BLOCK IN BYTES
000012  D.DSP:   .BLKW  1  ;POINTER TO DRIVER DISPATCH TABLE
000014  D.MSK:   .BLKW  1  ;LEGAL FUNCTION MASK  CODES 0-15.
000016           .BLKW  1  ;CONTROL FUNCTION MASK CODES 0-15.
000020           .BLKW  1  ;NOP'ED FUNCTION MASK CODES 0-15.
000022           .BLKW  1  ;ACP FUNCTION MASK CODES 0-15.
000024           .BLKW  1  ;LEGAL FUNCTION MASK CODES 16.-31.
000026           .BLKW  1  ;CONTROL FUNCTION MASK CODES 16.-31.
000030           .BLKW  1  ;NOP'ED FUNCTION MASK CODES 16.-31.
000032           .BLKW  1  ;ACP FUNCTION MASK CODES 16.-31.
000034  D.PCB:   .BLKW  1  ;LOADABLE DRIVER PCB ADDRESS

            .PSECT


      ;
      ; DRIVER DISPATCH TABLE OFFSET DEFINITIONS
      ;
      D.VDEB=177776      ;DEALLOCATE INTERNAL BUFFERS (FD TTDRV)
      D.VINI=0           ;DEVICE INITIATOR
      D.VCAN=2           ;CANCEL CURRENT I/O FUNCTION
      D.VOUT=4           ;DEVICE TIMEOUT
      D.VPWF=6           ;POWERFAIL RECOVERY
```

EPKDF$

```
        .MACRO  EPKDF$,L,B
;+
; Error Message Block Definitions
;-
        .ASECT

;
; Header Subpacket
;
;
;       +-----------------------------------------------+
;       | Subpacket Length in Bytes                     |
;       +-----------------------------------------------+
;       | Subpacket Flags                               |
;       +-----------------------+-----------------------+
;       | Format Identification | Operating System Code |
;       +-----------------------+-----------------------+
;       | Operating System Identification               |
;       |                                               |
;       +-----------------------+-----------------------+
;       | Flags                 |     Context Code      |
;       +-----------------------+-----------------------+
;       | Entry Sequence                                |
;       +-----------------------------------------------+
;       | Error Sequence                                |
;       +-----------------------+-----------------------+
;       | Entry Type Subcode    |   Entry Type Code     |
;       +-----------------------+-----------------------+
;       | Time Stamp                                    |
;       |                                               |
;       |                                               |
;       +-----------------------+-----------------------+
;       | Reserved              |    Processor Type     |
;       +-----------------------+-----------------------+
;       | Processor Identification (URM)                |
;       +-----------------------------------------------+
;
.=0

E$HLGH:'L'      .BLKW 1         ; Subpacket length in bytes
E$HSBF:'L'      .BLKW 1         ; Subpacket Flags
E$HSYS:'L'      .BLKB 1         ; Operating System Code
E$HIDN:'L'      .BLKB 1         ; Format Identification
E$HSID:'L'      .BLKB 4         ; Operating System Identification
E$HCTX:'L'      .BLKB 1         ; Context Code
E$HFLG:'L'      .BLKB 1         ; Flags
E$HENS:'L'      .BLKW 1         ; Entry Sequence Number
E$HERS:'L'      .BLKW 1         ; Error Sequence Number
E$HENC:'L'                      ; Entry Code
E$HTYC:'L'      .BLKB 1         ; Entry Type Code
E$HTYS:'L'      .BLKB 1         ; Entry Type Subcode
E$HTIM:'L'      .BLKB 6         ; Time Stamp
E$HPTY:'L'      .BLKB 1         ; Processor Type
                .BLKB 1         ; Reserved
E$HURM:'L'      .BLKW 1         ; Processor Identification (URM)

                .EVEN
E$HLEN:'L'                      ; Length
```

EPKDF$ (Cont.)

```
;
; Subpacket Flags for E$HSBF
;
        SM.ERR='B'      1       ; Error Packet
        SM.HDR='B'      1       ; Header Subpacket
        SM.TSK='B'      2       ; Task Subpacket
        SM.DID='B'      4       ; Device Identification Subpacket
        SM.DOP='B'     10       ; Device Operation Subpacket
        SM.DAC='B'     20       ; Device Activity Subpacket
        SM.DAT='B'     40       ; Data Subpacket
        SM.MBC='B'  20000       ; 22-bit massbus controller present
        SM.CMD='B'  40000       ; Error Log Command Packet
        SM.ZER='B' 100000       ; Zero I/O Counts
;
; Codes for field E$HIDN
;
        EH$FOR='B' 1            ; Current packet format


;
; Flags for the error log flags byte ($ERFLA) in the exec.
;
        ES.INI='B'  1       ; Error log initialized
        ES.DAT='B'  2       ; Error log receiving data packets
        ES.LIM='B'  4       ; Error limiting enabled
        ES.LOG='B' 10       ; Error logging enabled

;
; Type and Subtype Codes for fields E$HTYC and E$HTYS
;
;       Symbols with names E$Cxxx are type codes for field E$HTYC,
;       symbols with names E$Sxxx are subtype codes for field E$HTYS.
;
        E$CCMD='B' 1        ; Error Log Control
        E$SSTA='B' 1        ;       Error Log Status Change
        E$SSWI='B' 2        ;       Switch Logging Files
        E$SAPP='B' 3        ;       Append File
        E$SBAC='B' 4        ;       Declare Backup File
        E$SSHO='B' 5        ;       Show
        E$SCHL='B' 6        ;       Change Limits
        E$CERR='B' 2        ; Device Errors
        E$SDVH='B' 1        ;       Device Hard Error
        E$SDVS='B' 2        ;       Device Soft Error
        E$STMO='B' 3        ;       Device Interrupt Timeout (HARD)
        E$SUNS='B' 4        ;       Device Unsolicited Interrupt
        E$STMS='B' 5        ;       Device Interrupt Timeout (SOFT)
        E$CDVI='B' 3        ; Device Information
        E$SDVI='B' 1        ;       Device Information Message

        E$CDCI='B' 4        ; Device Control Information
        E$SMOU='B' 1        ;       Device Mount
        E$SDMO='B' 2        ;       Device Dismount
        E$SRES='B' 3        ;       Device Count Reset
        E$SRCT='B' 4        ;       Block Replacement
        E$CMEM='B' 5        ; Memory Detected Errors
        E$SMEM='B' 1        ;       Memory Error
        E$CSYS='B' 6        ; System Control Information
        E$SPWR='B' 1        ;       Power Recovery
```

EPKDF$ (Cont.)

```
        E$CCTL='B' 7        ; Control Information
        E$STIM='B' 1        ;       Time Change
        E$SCRS='B' 2        ;       System Crash
        E$SLOA='B' 3        ;       Device Driver Load
        E$SUNL='B' 4        ;       Device Driver Unload
        E$SHRC='B' 5        ;       Reconfiguration Status Change
        E$SMES='B' 6        ;       Message

        E$CCPU='B' 10       ; CPU Detected Errors
        E$SINT='B'  1       ;       Unexpected Interrupt

        E$CSDE='B' 11       ; Software Detected Events
        E$SABO='B'  1       ;       Task Abort

;
; Codes for Context Code entry E$HCTX
;
        EH$NOR='B'  1       ; Normal Entry
        EH$STA='B'  2       ; Start Entry
        EH$CRS='B'  3       ; Crash Entry
;
; Codes for Flags entry E$HFLG
;
        EH$VIR='B'  1       ; Addresses are virtual
        EH$EXT='B'  2       ; Addresses are extended
        EH$COU='B'  4       ; Error counts supplied
        EH$QBS='B' 10       ; Q-BUS CPU
        EH$LMR='B' 20       ; Limit reached

;
; Task Subpacket
;
;       +-----------------------------------------------+
;       | Task Subpacket Length                         |
;       +-----------------------------------------------+
;       | Task Name in RAD50                            |
;       |                                               |
;       +-----------------------------------------------+
;       | Task UIC                                      |
;       +-----------------------------------------------+
;       | Task TI: Device Name                          |
;       +-----------------------+-----------------------+
;       | Flags                 | Task TI: Unit Number  |
;       +-----------------------+-----------------------+

.=0

E$TLGH:'L'  .BLKW 1         ; Task Subpacket Length
E$TTSK:'L'  .BLKW 2         ; Task Name in RAD50
E$TUIC:'L'  .BLKW 1         ; Task UIC
E$TTID:'L'  .BLKB 2         ; Task TI: Device Name
E$TTIU:'L'  .BLKB 1         ; Task TI: Unit
E$TFLG:'L'  .BLKB 1         ; Flags

            .EVEN
E$TLEN:'L'

;
; Flags for entry E$TFLG
;
        ET$PRV='B' 1        ; Task is Privileged
        ET$PRI='B' 2        ; Terminal is Privileged
```

EPKDF$ (Cont.)

```
;
; Device Identification Subpacket
;
;       +-----------------------------------------------+
;       | Device Identification Subpacket Length        |
;       +-----------------------------------------------+
;       | Device Mnemonic Name                          |
;       +-----------------------+-----------------------+
;       | Controller Number     |  Device Unit Number   |
;       +-----------------------+-----------------------+
;       | Physical Subunit #    |    Physical Unit #    |
;       +-----------------------+-----------------------+
;       | Physical Device Mnemonic (RSX-11M-PLUS only)  |
;       +-----------------------+-----------------------+
;       | Reserved              | Flags                 |
;       +-----------------------+-----------------------+
;       | Volume Name of Mounted Volume                 |
;       |                                               |
;       |                                               |
;       |                                               |
;       |                                               |
;       +-----------------------------------------------+
;       | Pack Identification                           |
;       |                                               |
;       +-----------------------------------------------+
;       | Device Type Class                             |
;       +-----------------------------------------------+
;       | Device Type                                   |
;       |                                               |
;       +-----------------------------------------------+
;       | I/O Operation Count Longword                  |
;       |                                               |
;       +-----------------------+-----------------------+
;       | Hard Error Count      |  Soft Error Count     |
;       +-----------------------+-----------------------+
;       | Blocks Transferred Count (RSX-11M-PLUS only)  |
;       |                                               |
;       +-----------------------------------------------+
;       | Cylinders Crossed Count (RSX-11M-PLUS only)   |
;       |                                               |
;       +-----------------------------------------------+
;
.=0

E$ILGH:'L' .BLKW 1        ; Device Identification Subpacket Length
E$ILDV:'L' .BLKW 1        ; Device Mnemonic Name
E$ILUN:'L' .BLKB 1        ; Device Unit Number
E$IPCO:'L' .BLKB 1        ; Controller Number
E$IPUN:'L' .BLKB 1        ; Physical Unit Number
E$IPSU:'L' .BLKB 1        ; Physical Subunit Number

               .IF DF R$$MPL

E$IPDV:'L' .BLKW 1        ; Physical Device Mnemonic

               .ENDC ; R$$MPL
```

EPKDF$ (Cont.)

```
E$IFLG:'L' .BLKB   1      ; Flags
           .BLKB   1      ; Reserved
E$IVOL:'L' .BLKB 12.      ; Volume Name
E$IPAK:'L' .BLKB   4      ; Pack Identification
E$IDEV:'L'                ; Device Type
E$IDCL:'L' .BLKW   1      ; Device Type Class
E$IDTY:'L' .BLKW   2      ; Device Type
E$IOPR:'L' .BLKW   2      ; I/O Operation Count Longword
E$IERS:'L' .BLKB   1      ; Soft Error Count
E$IERH:'L' .BLKB   1      ; Hard Error Count

             .IF DF R$$MPL

E$IBLK:'L' .BLKW   2      ; Blocks transferred count
E$ICYL:'L' .BLKW   2      ; Cylinders crossed count

             .ENDC ; R$$MPL

           .EVEN

E$ILEN:'L'                ; Subpacket Length
;
; Flags for field E$IFLG
;
      EI$SUB='B' 1        ; Subcontroller device

             .IF DF R$$MPL

      EI$NUX='B' 2        ; No UCB extension, data invalid

             .ENDC ; R$$MPL

;
; Device Operation Subpacket
;
;       +-----------------------------------------------+
;       | Device Operation Subpacket Length             |
;       +-----------------------------------------------+
;       | Task Name in RAD50                            |
;       |                                               |
;       +-----------------------------------------------+
;       | Task UIC                                      |
;       +-----------------------------------------------+
;       | Task TI: Logical Device Mnemonic              |
;       +-----------------------+-----------------------+
;       | Reserved              | Task TI: Device Unit  |
;       +-----------------------+-----------------------+
;       | I/O Function Code                             |
;       +-----------------------+-----------------------+
;       | Reserved              | Operation Flags       |
;       +-----------------------+-----------------------+
;       | Transfer Operation Address                    |
;       |                                               |
;       +-----------------------------------------------+
;       | Transfer Operation Byte Count                 |
;       +-----------------------------------------------+
;       | Current Operation Retry Count                 |
;       +-----------------------------------------------+
;
.=0
```

EPKDF$ (Cont.)

```
E$OLGN:'L'  .BLKW 1         ; Subpacket Length
E$OTSK:'L'  .BLKW 2         ; Task Name in RAD50
E$OUIC:'L'  .BLKW 1         ; Task UIC
E$OTID:'L'  .BLKB 2         ; Task TI: Logical Device Mnemonic
E$OTIU:'L'  .BLKB 1         ; Task TI: Logical Device Unit
            .BLKB 1         ; Reserved
E$OFNC:'L'  .BLKW 1         ; I/O Function Code
E$OFLG:'L'  .BLKB 1         ; Operation Flags
            .BLKB 1         ; Reserved
E$OADD:'L'  .BLKW 2         ; Transfer Operation Address
E$OSIZ:'L'  .BLKW 1         ; Transfer Operation Byte Count
E$ORTY:'L'  .BLKW 1         ; Current Operation Retry Count
            .EVEN

E$OLEN:'L'                  ; Device Operation Subpacket Length
;
; Flags for field E$OFLG
;
       EO$TRA='B'  1        ; Transfer Operation
       EO$DMA='B'  2        ; DMA Device
       EO$EXT='B'  4        ; Extended Addressing Device
       EO$PIP='B' 10        ; Device is positioning

;
; I/O Activity Subpacket
;
;       +-----------------------------------------------+
;       | I/O Activity Subpacket Length                 |
;       +-----------------------------------------------+
;

.=0

E$ALGH:'L' .BLKW 1          ; Subpacket Length

;
; I/O Activity Subpacket Entry
;
;       +-----------------------------------------------+
;       | Logical Device Name Mnemonic                  |
;       +-----------------------+-----------------------+
;       | Controller Number     |   Logical Device Unit |
;       +-----------------------+-----------------------+
;       | Physical Subunit #    | Physical Unit Number  |
;       +-----------------------+-----------------------+
;       | Physical Device Mnemonic (RSX-11M-PLUS only)  |
;       +-----------------------+-----------------------+
;       | Task TI: logical unit |     Device flags      |
;       +-----------------------+-----------------------+
;       | Requesting Task Name in RAD50                 |
;       |                                               |
;       +-----------------------------------------------+
;       | Requesting Task UIC                           |
;       +-----------------------------------------------+
;       | Task TI: Logical Device Name                  |
;       +-----------------------------------------------+
;       | I/O Function Code                             |
;       +-----------------------+-----------------------+
```

(Continued on next page)

EPKDF$ (Cont.)

```
;       +-----------------------+-----------------------+
;       | Reserved              | Flags                 |
;       +-----------------------+-----------------------+
;       | Transfer Operation Address                    |
;       |                                               |
;       +-----------------------------------------------+
;       | Transfer Operation Byte Count                 |
;       +-----------------------------------------------+
;

.=0

E$ALDV:'L' .BLKW 1        ; Logical Device Name Mnemonic
E$ALUN:'L' .BLKB 1        ; Logical Device Unit
E$APCO:'L' .BLKB 1        ; Controller Number
E$APUN:'L' .BLKB 1        ; Physical Unit Number
E$APSU:'L' .BLKB 1        ; Physical Subunit Number

           .IF DF R$$MPL

E$APDV:'L' .BLKW 1        ; Physical Device Mnemonic

           .ENDC

E$ADFG:'L' .BLKB 1        ; Device flags
E$ATIU:'L' .BLKB 1        ; Task TI: Logical Unit
E$ATSK:'L' .BLKW 2        ; Requesting Task Name in RAD50
E$AUIC:'L' .BLKW 1        ; Requesting Task UIC
E$ATID:'L' .BLKW 1        ; Task TI: Logical Device Name
E$AFNC:'L' .BLKW 1        ; I/O Function Code
E$AFLG:'L' .BLKB 1        ; Flags
           .BLKB 1        ; Reserved
E$AADD:'L' .BLKW 2        ; Transfer Operation Address
E$ASIZ:'L' .BLKW 1        ; Transfer Operation Byte Count

           .EVEN

E$ALEN:'L'                ; Subpacket Entry Length
;
; Flags for field E$ADFG
;
       EA$SUB='B' 1       ; Subcontroller device

              .IF DF R$$MPL

       EA$NUX='B' 2       ; No UCB extension, data invalid

              .ENDC ; R$$MPL
;
; Flags for field E$AFLG
;
       EA$TRA='B'  1      ; Transfer Operation
       EA$DMA='B'  2      ; DMA Device
       EA$EXT='B'  4      ; Device has Extended Addressing
       EA$PIP='B' 10      ; Device is positioning

       .PSECT

       .MACRO  EPKDF$ X,Y
       .ENDM

       .ENDM
```

EVNDF$

```
          .MACRO EVNDF$,L,B,LST

;
; EVNDF$ -- Event Packet Definitions
;
;         This module contains a macro which defines the offsets and field
;         values for TSA Event Packets (TEP's).  These packets are used to
;         pass data and status information between system components that
;         provide Digital's Terminal Software Architecture support on RSX.

;
; Explicit Inputs:
;
;         L                       ":" for global offset definitions
;         B                       "=" for global bit/value definitions
;         LST                     "LIST" for macro expansion listing
;
; Implicit Inputs:
;
;         NONE
;
; Outputs:
;
;         Symbols defined as described above.
;         Listing as described above.
;

;
; General packet header format
;
          .ASECT                    ; Define offsets absolutely
          .=0

E.VLNK:'L'                          .BLKW 1 ; Link word
E.VSIZ:'L'                          .BLKB 1 ; Packet size
E.VTYP:'L'                          .BLKB 1 ; Packet type
E.VUCB:'L'                          .BLKW 1 ; Terminal UCB address

;
; E.VTYP Values
;
ET.LOW='B'0                           ; Lowest valid type code

ET.QIO='B'0                           ; QIO(distinguishes QIO packet from TEP)
ET.BND='B'2                           ; Bind Request
ET.UNB='B'4                           ; Unbind Request
ET.BCP='B'6                           ; Bind Complete
ET.REJ='B'10                          ; Bind Reject
ET.DIS='B'12                          ; Disconnect Notification
ET.DCP='B'14                          ; Disconnect Complete
ET.ICS='B'16                          ; Input Count State Change
ET.OOB='B'20                          ; Out-of-Band (OOB)
ET.ONO='B'22                          ; Abnormal Termination Request
ET.PHO='B'24                          ; Physical Terminal Disconnected

ET.HI='B'24                           ; Highest valid type code
```

EVNDF$ (Cont.)

```
;
;       The following definitions are for packet types that require
;  passing additional information in the packets.  All other packet
;  types use the general packet format described above.
;

;
; Bind Request packet (Terminal Management Mode --> Network)
;
        .=E.VUCB+2

E.VBCT:'L'                      .BLKW 1 ; Count of nodes (One for now)
E.VBND:'L'                      .BLKB 6 ; Node name
E.VBLN:'L'                        ; Length of bind request

;
; Input Count State Change, Out-Of-Band packets (TTDRV --> Network)
; And Modem Hang-up packets (TSA... --> Network)
;
        .=E.VUCB+2

E.VAPR:'L'                      .BLKW 1 ; Doubleword address of packet...
E.VADR:'L'                      .BLKW 1 ; ...queueing routine
E.VFLG:'L'                      .BLKW 1 ; Flag

;
; Input Count State Change
;
        .=E.VFLG+2

E.VSLN:'L'                        ; Length of Input state message

;
; OOB
;
        .=E.VFLG+2

E.VOBM:'L'                      .BLKW 6 ; Out-of-Band bitmasks
E.VHDR:'L'                      .BLKW 2 ; Type-ahead buffer header
E.VTAB:'L'                      .BLKB 10. ; Type-ahead buffer
E.VOLN:'L'                        ; Length of OOB packet

;
; Terminal Management Switch Characters
;
        .=E.VFLG+2

E.VSWC:'L'                      .BLKW 1 ; Terminal management switch characters
E.VTLN:'L'                        ; Length of Switch Character packet

;
; Bit values in flag word (E.VFLG).  For convenience some bits have
;       corresponding bits in the AST Control Block flag word (A.PRM+5).
;
EF.NCO='B'1                       ; All non-control characters are out-of band
EF.NOI='B'2                       ; All non-control OOB are include-OOB
EF.AST='B'10                      ; Reserved bit synonymous with TF.AST
EF.LCK='B'40                      ; Reserved bit synonymous with AF.LCK
EF.QUE='B'100                     ; TEP is queued
EF.MDE='B'200                     ; TEP is marked for delete
```

EVNDF$ (Cont.)

```
;
; Unbind Request packet (TMM --> Network)
;
          .=E.VUCB+2

E.VULN:'L'                          ; Length of Unbind message

;
; Connect Reject notification packet (Network --> TMM)
;
          .=E.VUCB+2

E.VRR:'L' .BLKW                   1 ; Reason for Rejection
E.VRLN:'L'                          ; Length of Reject message

;
; Disconnect Notification packet (Network --> TMM)
;
          .=E.VUCB+2

E.VRD:'L' .BLKW                   1 ; Reason for Disconnect
E.VDLN:'L'                          ; Length of Disconnect message

;
; Disconnect Complete packet (TMM --> Network)
;
          .=E.VUCB+2

E.VDCL:'L'                          ; Length of Disconnect Complete message


          .PSECT

.IF NB LST
          .NLIST                    ; Turn listing back off
.IFF
          .MACRO                  EVNDF$  ; If not listing, redefine
          .ENDM                     ;  macro to nothing
.ENDC
          .ENDM                   EVNDF$
```

F11DF$

```
        .MACRO  F11DF$,L,B

;
; VOLUME CONTROL BLOCK
;

        .ASECT
.=0

V.TRCT:'L'.BLKW 1       ; TRANSACTION COUNT
V.TYPE:'L'.BLKB 1       ; VOLUME TYPE DESCRIPTOR
        VT.FOR='B'  0       ; Foreign volume structure
        VT.SL1='B'  1       ; Files-11 Structure level 1
        VT.SL2='B'  2       ; Files-11 Structure level 2
        VT.ANS='B' 10       ; ANSI labeled tape
        VT.UNL='B' 11       ; Unlabeled tape
V.VCHA:'L'.BLKB 1       ; Volume characteristics
        VC.SLK='B'  1       ; Clear volume valid on dismount
        VC.HLK='B'  2       ; Unload the volume on dismount
        VC.DEA='B'  4       ; Deallocate the volume on dismount
        VC.PUB='B' 10       ; Set (clear) US.PUB on dismount
        VC.DUP='B' 20       ; Duplicate volume name; don't delete
                            ;logicals
V.LABL:'L'.BLKB 14      ; Volume label (ASCII)
V.PKSR:'L'.BLKW  2      ; Pack serial number for error logging
V.SLEN:'L'              ; Length of short VCB
V.IFWI:'L'.BLKW  1      ; INDEX FILE WINDOW
V.FCB:'L' .BLKW  2      ; FILE CONTROL BLOCK LIST HEAD
V.IBLB:'L'.BLKB  1      ; INDEX BIT MAP 1ST LBN HIGH BYTE
V.IBSZ:'L'.BLKB  1      ; INDEX BIT MAP SIZE IN BLOCKS
        .BLKW    1      ; INDEX BITMAP 1ST LBN LOW BITS
V.FMAX:'L'.BLKW  1      ; MAX NO. OF FILES ON VOLUME
V.WISZ:'L'.BLKB  1      ; DEFAULT SIZE OF WINDOW IN RTRV PTRS
                        ; VALUE IS < 128.
V.SBCL:'L'.BLKB  1      ; STORAGE BIT MAP CLUSTER FACTOR
V.SBSZ:'L'.BLKW  1      ; STORAGE BIT MAP SIZE IN BLOCKS
V.SBLB:'L'.BLKB  1      ; STORAGE BIT MAP 1ST LBN HIGH BYTE
V.FIEX:'L'.BLKB  1      ; DEFAULT FILE EXTEND SIZE
          .BLKW  1      ; STORAGE BIT MAP 1ST LBN LOW BITS
V.VOWN:'L'.BLKW  1      ; VOLUME OWNER'S UIC
V.VPRO:'L'.BLKW  1      ; VOLUME PROTECTION
V.FPRO:'L'.BLKW  1      ; VOLUME DEFAULT FILE PROTECTION
V.FRBK:'L'.BLKB  1      ; NUMBER OF FREE BLOCKS ON VOLUME HIGH BYTE
V.LRUC:'L'.BLKB  1      ; COUNT OF AVAILABLE LRU SLOTS IN FCB LIST
          .BLKW  1      ; NUMBER OF FREE BLOCKS ON VOLUME LOW BITS
V.STS:'L' .BLKB  1      ; VOLUME STATUS BYTE, CONTAINING THE
                        ; FOLLOWING
        VS.IFW='B'1     ; INDEX FILE IS WRITE ACCESSED
        VS.BMW='B'2     ; STORAGE BITMAP FILE IS WRITE ACCESSED
V.FFNU:'L'.BLKB  1      ; FIRST FREE INDEX FILE BITMAP BLOCK
V.EXT:'L' .BLKW  1      ; POINTER TO VCB EXTENSION
V.HBLB:'L'.BLKW  2      ; LBN of home block
V.HBCS:'L'.BLKW  2      ; Home block checksums
V.LGTH:'L'              ; SIZE IN BYTES OF VCB
```

F11DF$ (Cont.)

```
;
; MOUNT LIST ENTRY
;
; EACH ENTRY ALLOWS ACCESS TO A SPECIFIED USER FOR A NON-PUBLIC DEVICE
;
; TO ALLOW EXPANSION, ONLY THE ONLY TYPE CODE DEFINED IS "1" FOR
; DEVICE ACCESS BLOCKS
;

        .ASECT
.=0

M.LNK:'L' .BLKW  1        ; LINK WORD
M.TYPE:'L'.BLKB  1        ; TYPE OF ENTRY
        MT.MLS='B' 1       ; Mounted volume user access list
M.ACC:'L' .BLKB  1        ; NUMBER OF ACCESSES
M.DEV:'L' .BLKW  1        ; DEVICE UCB
M.TI:'L'  .BLKW  1        ; ACCESSOR TI: UCB
M.LEN:'L'                 ; LENGTH OF ENTRY

;
; FILE CONTROL BLOCK
;

        .ASECT
.=0

F.LINK:'L'.BLKW  1        ; FCB CHAIN POINTER
F.FNUM:'L'.BLKW  1        ; FILE NUMBER
F.FSEQ:'L'.BLKW  1        ; FILE SEQUENCE NUMBER
          .BLKB  1        ; NOT USED
F.FSQN:'L'.BLKB  1        ; FILE SEGMENT NUMBER
F.FOWN:'L'.BLKW  1        ; FILE OWNER'S UIC
F.FPRO:'L'.BLKW  1        ; FILE PROTECTION CODE
F.UCHA:'L'.BLKB  1        ; USER CONTROLLED CHARACTERISTICS
F.SCHA:'L'.BLKB  1        ; SYSTEM CONTROLLED CHARACTERISTICS
F.HDLB:'L'.BLKW  2        ; FILE HEADER LOGICAL BLOCK NUMBER
                          ; BEGINNING OF STATISTICS BLOCK
F.LBN:'L'.BLKW   2        ; LBN OF VIRTUAL BLOCK 1 IF CONTIGUOUS
                          ; 0 IF NON CONTIGUOUS
F.SIZE:'L'.BLKW  2        ; SIZE OF FILE IN BLOCKS
F.NACS:'L'.BLKB  1        ; NO. OF ACCESSES
F.NLCK:'L'.BLKB  1        ; NO. OF LOCKS
S.STBK='B'.-F.LBN         ; SIZE OF STATISTICS BLOCK

F.STAT:'L'               ; FCB STATUS WORD
F.NWAC:'L'.BLKB  1        ; NUMBER OF WRITE ACCESSORS
          .BLKB  1        ; STATUS BITS FOR FCB CONSISTING OF
       FC.WAC='B' 100000  ; SET IF FILE ACCESSED FOR WRITE
       FC.DIR='B' 40000   ; SET IF FCB IS IN DIRECTORY LRU
       FC.CEF='B' 20000   ; SET IF DIRECTORY EOF NEEDS UPDATING
       FC.FCO='B' 10000   ; SET IF TRYING TO FORCE DIRECTORY CONTIG
F.DREF:'L'.BLKW  1        ; DIRECTORY EOF BLOCK NUMBER
F.DRNM:'L'.BLKW  1        ; 1ST WORD OF DIRECTORY NAME
F.FEXT:'L'.BLKW  1        ; POINTER TO EXTENSION FCB
F.FVBN:'L'.BLKW  2        ; STARTING VBN OF THIS FILE SEGMENT
F.LKL:'L' .BLKW  1        ; POINTER TO LOCKED BLOCK LIST FOR FILE
F.WIN:'L' .BLKW  1        ; WINDOW BLOCK LIST FOR THIS FILE
F.LGTH:'L'                ; SIZE IN BYTES OF FCB
```

F11DF$ (Cont.)

```
;
; WINDOW
;

        .ASECT
.=0

W.ACT:'L'               ; NUMBER OF ACTIVE MAPPING POINTERS
                        ;  WHEN NO SECONDARY POOL
W.BLKS:'L'              ; BLOCK SIZE OF SECONDARY POOL SEGMENT
                        ;  WHEN SECONDARY POOL
W.CTL:'L'.BLKW  1       ; LOW BYTE = # OF MAP ENTRIES ACTIVE
                        ; HIGH BYTE CONSISTS OF CONTROL BITS
        WI.RDV='B' 400    ; READ VIRTUAL BLOCK ALLOWED IF SET
        WI.WRV='B' 1000   ; WRITE VIRTUAL BLOCK ALLOWED IF SET
        WI.EXT='B' 2000   ; EXTEND ALLOWED IF SET
        WI.LCK='B' 4000   ; SET IF LOCKED AGAINST SHARED ACCESS
        WI.DLK='B' 10000  ; SET IF DEACCESS LOCK ENABLED
        WI.PND='B' 20000  ; WINDOW TURN PENDING BIT
        WI.EXL='B' 40000  ; SET IF MANUAL UNLOCK DESIRED
        WI.WCK='B' 100000 ; Data check all writes to file
W.IOC:'L'.BLKB  1       ; COUNT OF I/O THROUGH THIS WINDOW
          .BLKB  1      ; Reserved
W.FCB:'L'.BLKW  1       ; FILE CONTROL BLOCK ADDRESS
W.TCB:'L'.BLKW  1       ; TCB address of accessor
W.UCB:'L'.BLKW  1       ; Original UCB address of device
W.LKL:'L'.BLKW  1       ; POINTER TO LIST OF USERS LOCKED BLOCKS
W.WIN:'L'.BLKW  1       ; WINDOW BLOCK LIST LINK WORD

        .IF       NB,SYSDEF  ; IF SYSDEF SPECIFIED IN CALL

        .IF       NDF,P$$WND ; IF SECONDARY POOL WINDOWS NOT ALLOWED

;
; NON-SECONDARY POOL WINDOW BLOCK
;       IF SECONDARY POOL WINDOWS ARE NOT ENABLED, THE WINDOW BLOCK
;       CONTAINS THE CONTROL INFORMATION AND RETRIEVAL POINTERS.
;

W.VBN:'L' .BLKB  1      ; HIGH BYTE OF 1ST VBN MAPPED BY WINDOW
W.MAP:'L'               ; DEFINE LABEL WITH ODD ADDRESS TO CATCH BAD
                        ;  REFS
W.WISZ:'L'.BLKB  1      ; SIZE IN RTRV PTRS OF WINDOW (7 BITS)
          .BLKW  1      ; LOW ORDER WORD OF 1ST VBN MAPPED
W.RTRV:'L'              ; OFFSET TO 1ST RETRIEVAL POINTER IN WINDOW

W.SLEN='B'-4            ; Dummy definition to prevent incorrect
                        ; reference
                        ; (-4 when rounded "up" is a VERY large block)

        .IFF            ; IF WINDOWS IN SECONDARY POOL

;
; SECONDARY POOL WINDOW CONTROL AND MAPPING BLOCK
;       IF SECONDARY POOL WINDOW BLOCKS ARE ENABLED, LUTN2 POINTS
;       TO A CONTROL BLOCK IN SYSTEM POOL WHICH CONTAINS THE
;       FOLLOWING CONTROL FIELDS AND THE MAPPING INFORMATION
;       FOR THE SECONDARY POOL WINDOW.
;
```

F11DF$ (Cont.)

```
W.MAP:'L' .BLKW  1        ; ADDR TO THE MAPPING PTRS IN SECONDARY POOL
W.SLEN:'L'                ; Length of primary pool stub

;
; SECONDARY POOL WINDOW
;       IF SECONDARY POOL WINDOW BLOCKS ARE ENABLED, THE RETRIEVAL
;       POINTERS ARE MAINTAINED IN SECONDARY POOL IN THE FOLLOWING
;       FORMAT.
;

.=0

       ASSUME       W.CTL,0

          .BLKB  1        ; NUMBER OF ACTIVE MAPPING POINTERS
W.USE:'L' .BLKB  1        ; STATUS OF BLOCK
W.VBN:'L' .BLKB  1        ; HIGH BYTE OF 1ST VBN MAPPED BY WINDOW
W.WISZ:'L'.BLKB  1        ; SIZE IN RTRV PTRS OF WINDOW (7 BITS)
          .BLKW  1        ; LOW ORDER WORD OF 1ST VBN MAPPED
W.RTRV:'L'                ; OFFSET TO 1ST RETRIEVAL POINTER IN WINDOW

       .ENDC              ; END SECONDARY POOL WINDOW CONDITIONAL

       .ENDC              ; END SYSDEF CONDITIONAL

;
; LOCKED BLOCK LIST NODE
;

       .ASECT
.=0

L.LNK:'L' .BLKW  1        ; LINK TO NEXT NODE IN LIST
L.WI1:'L' .BLKW  1        ; POINTER TO WINDOW FOR FIRST ENTRY
L.VB1:'L' .BLKB  1        ; HIGH ORDER VBN BYTE
L.CNT:'L' .BLKB  1        ; COUNT FOR ENTRY
          .BLKW  1        ; LOW ORDER VBN
L.LKSZ:'L '
;
; END OF DEFINITIONS
;

       .PSECT
       .MACRO   F11DF$  X,Y,Z
       .ENDM    F11DF$
       .ENDM    F11DF$
```

HDRDF$

```
        .MACRO  HDRDF$,L,B

;+
; TASK HEADER OFFSET DEFINITIONS
;-

        .ASECT
.=0
H.CSP:'L' .BLKW  1          ;CURRENT STACK POINTER
H.HDLN:'L'.BLKW  1          ;HEADER LENGTH IN BYTES
H.EFLM:'L'.BLKW  2          ;EVENT FLAG MASK WORD AND ADDRESS
H.CUIC:'L'.BLKW  1          ;CURRENT TASK UIC
H.DUIC:'L'.BLKW  1          ;DEFAULT TASK UIC
H.IPS:'L' .BLKW  1          ;INITIAL PROCESSOR STATUS WORD (PS)
H.IPC:'L' .BLKW  1          ;INITIAL PROGRAM COUNTER (PC)
H.ISP:'L' .BLKW  1          ;INITIAL STACK POINTER (SP)
H.ODVA:'L'.BLKW  1          ;ODT SST VECTOR ADDRESS
H.ODVL:'L'.BLKW  1          ;ODT SST VECTOR LENGTH
H.TKVA:'L'.BLKW  1          ;TASK SST VECTOR ADDRESS
H.TKVL:'L'.BLKW  1          ;TASK SST VECTOR LENGTH
H.PFVA:'L'.BLKW  1          ;POWER FAIL AST CONTROL BLOCK ADDRESS
H.FPVA:'L'.BLKW  1          ;FLOATING POINT AST CONTROL BLOCK ADDRESS
H.RCVA:'L'.BLKW  1          ;RECIEVE AST CONTROL BLOCK ADDRESS
H.EFSV:'L'.BLKW  1          ;EVENT FLAG ADDRESS SAVE ADDRESS
H.FPSA:'L'.BLKW  1          ;POINTER TO FLOATING POINT/EAE SAVE AREA
H.WND:'L' .BLKW  1          ;POINTER TO NUMBER OF WINDOW BLOCKS
H.DSW:'L' .BLKW  1          ;TASK DIRECTIVE STATUS WORD
H.FCS:'L' .BLKW  1          ;FCS IMPURE POINTER
H.FORT:'L'.BLKW  1          ;FORTRAN IMPURE POINTER
H.OVLY:'L'.BLKW  1          ;OVERLAY IMPURE POINTER
H.VEXT:'L'.BLKW  1          ;WORK AREA EXTENSION VECTOR POINTER
H.SPRI:'L'.BLKB  1          ;PRIORITY DIFFERENCE FOR SWAPPING
H.NML:'L' .BLKB  1          ;NETWORK MAILBOX LUN
H.RRVA:'L'.BLKW  1          ;RECEIVE BY REFERENCE AST CONTROL BLOCK
                            ;ADDRESS
H.X25:'L' .BLKB  1          ; FOR USE BY X.25 SOFTWARE
          .BLKB  1          ; FIVE RESERVED BYTES
          .BLKW  2          ;
H.GARD:'L'.BLKW  1          ;POINTER TO HEADER GUARD WORD
H.NLUN:'L'.BLKW  1          ;NUMBER OF LUN'S
H.LUN:'L' .BLKW  2          ;START OF LOGICAL UNIT TABLE


;+
; LENGTH OF FLOATING POINT SAVE AREA
;-

H.FPSL='B'25.*2                     ;
;+
; WINDOW BLOCK OFFSETS
;-

.=0
W.BPCB:'L'.BLKW  1              ;PARTITION CONTROL BLOCK ADDRESS
W.BLVR:'L'.BLKW  1              ;LOW VIRTUAL ADDRESS LIMIT
W.BHVR:'L'.BLKW  1              ;HIGH VIRTUAL ADDRESS LIMIT
W.BATT:'L'.BLKW  1              ;ADDRESS OF ATTACHMENT DESCRIPTOR
W.BSIZ:'L'.BLKW  1              ;SIZE OF WINDOW IN 32W BLOCKS
```

HDRDF$ (Cont.)

```
W.BOFF:'L'.BLKW  1              ;PHYSICAL MEMORY OFFSET IN 32W BLOCKS
W.BFPD:'L'.BLKB  1              ;FIRST PDR ADDRESS
W.BNPD:'L'.BLKB  1              ;NUMBER OF PDR'S TO MAP
W.BLPD:'L'.BLKW  1              ;CONTENTS OF LAST PDR
W.BLGH:'L'                      ;LENGTH OF WINDOW DESCRIPTOR

        .PSECT
        .MACRO HDRDF$ X,Y
        .ENDM
        .ENDM
```

HWDDF$

```
        .MACRO HWDDF$,L,B

;+
; HARDWARE REGISTER ADDRESSES AND STATUS CODES
;-

MPCSR='B'   177746                  ;ADDRESS OF PDP-11/70 MEMORY
                                    ;PARITY REGISTER
MPAR='B'    172100                  ;ADDRESS OF FIRST MEMORY
                                    ;PARITY REGISTER
PIRQ='B'    177772                  ;PROGRAMMED INTERRUPT REQUEST
                                    ;REGISTER
PR0='B'          0                  ;PROCESSOR PRIORITY 0
PR1='B'         40                  ;PROCESSOR PRIORITY 1
PR4='B'        200                  ;PROCESSOR PRIORITY 4
PR5='B'        240                  ;PROCESSOR PRIORITY 5
PR6='B'        300                  ;PROCESSOR PRIORITY 6
PR7='B'        340                  ;PROCESSOR PRIORITY 7
PS='B'      177776                  ;PROCESSOR STATUS WORD
SWR='B'     177570                  ;CONSOLE SWITCH AND DISPLAY
                                    ;REGISTER
TPS='B'     177564                  ;CONSOLE TERMINAL PRINTER STATUS
                                    ;REGISTER

;+
; EXTENDED ARITHMETIC ELEMENT REGISTERS
;-

        .IF DF      E$$EAE

AC='B'      177302                  ;ACCUMULATOR
MQ='B'      177304                  ;MULTIPLIER-QUOTIENT
SC='B'      177310                  ;SHIFT COUNT

        .ENDC

;+
; MEMORY MANAGEMENT HARDWARE REGISTERS AND STATUS CODES
;-

        .IF DF      M$$MGE

KDSAR0='B'      172360              ;KERNEL D PAR 0
KDSDR0='B'      172320              ;KERNEL D PDR 0
KISAR0='B'      172340              ;KERNEL I PAR 0
KINAR0='B'      KISAR0              ;KERNEL I PAR 0
KISAR5='B'      172352              ;KERNEL I PAR 5
KINAR5='B'      KISAR5              ;KERNEL I PAR 5
KISAR6='B'      172354              ;KERNEL I PAR 6
KINAR6='B'      KISAR6              ;KERNEL I PAR 6
KISAR7='B'      172356              ;KERNEL I PAR 7
KINAR7='B'      KISAR7              ;KERNEL I PAR 7
KISDR0='B'      172300              ;KERNEL I PDR 0
KISDR6='B'      172314              ;KERNEL I PDR 6
```

HWDDF$ (Cont.)

```
KISDR7='B'      172316              ;KERNEL I PAR 7
SISDR0='B'      172200              ;SUPERVISOR I PDR 0
UDSAR0='B'      177660              ;USER D PAR 0
UDSDR0='B'      177620              ;USER D PDR 0
UISAR0='B'      177640              ;USER I PAR 0
UISAR4='B'      177650              ;USER I PAR 4
UISAR5='B'      177652              ;USER I PAR 5
UISAR6='B'      177654              ;USER I PAR 6
UISAR7='B'      177656              ;USER I PAR 7
UISDR0='B'      177600              ;USER I PDR 0
UISDR4='B'      177610              ;USER I PDR 4
UISDR5='B'      177612              ;USER I PDR 5
UISDR6='B'      177614              ;USER I PDR 6
UISDR7='B'      177616              ;USER I PDR 7
UBMPR='B'       170200              ;UNIBUS MAPPING REGISTER 0
CMODE='B'       140000              ;CURRENT MODE FIELD OF PS WORD

        .IFTF       ;M$$MGE

PMODE='B'        30000              ;PREVIOUS MODE FIELD OF PS WORD

        .IFT        ;M$$MGE

SR0='B'         177572              ;SEGMENT STATUS REGISTER 0
SR3='B'         172516              ;SEGMENT STATUS REGISTER 3

        .ENDC


;+
; FEATURE SYMBOL DEFINITIONS
;-

FE.EXT='B'        1          ;22-BIT EXTENDED MEMORY SUPPORT
FE.MUP='B'        2          ;MULTI-USER PROTECTION SUPPORT
FE.EXV='B'        4          ;EXECUTIVE IS SUPPORTED TO 20K
FE.DRV='B'       10          ;LOADABLE DRIVER SUPPORT
FE.PLA='B'       20          ;PLAS SUPPORT
FE.CAL='B'       40          ;DYNAMIC CHECKPOINT SPACE ALLOCATION
FE.PKT='B'      100          ;PREALLOCATION OF I/O PACKETS
FE.EXP='B'      200          ;EXTEND TASK DIRECTIVE SUPPORTED
FE.LSI='B'      400          ;PROCESSOR IS AN LSI-11
FE.OFF='B'     1000          ;PARENT OFFSPRING TASKING SUPPORTED
FE.FDT='B'     2000          ;FULL DUPLEX TERMINAL DRIVER
FE.X25='B'     4000          ;X.25 COM EXECUTIVE LOADED (1=YES)
FE.DYM='B'    10000          ;DYNAMIC MEMORY ALLOCATION SUPPORTED
FE.CEX='B'    20000          ;COM EXEC IS LOADED
FE.MXT='B'    40000          ;MCR EXIT AFTER EACH COMMAND MODE
FE.NLG='B'   100000          ;LOGINS DISABLED - MULTI-USER SUPPORT


;+
; SECOND FEATURE MASK SYMBOL DEFINITIONS
;-

F2.DAS='B'        1        ;KERNEL DATA SPACE (M-PLUS ONLY)
F2.LIB='B'        2        ;SUPERVISOR MODE LIBRARIES
F2.MP='B'         4        ;MULTIPROCESSING SUPPORT
```

HWDDF$ (Cont.)

```
F2.EVT='B'      10          ;EVENT TRACE SUPPORT
F2.ACN='B'      20          ;CPU ACCOUNTING
F2.SDW='B'      40          ;SHADOW RECORDING
F2.POL='B'     100          ;SECONDARY POOLS
F2.WND='B'     200          ;SECONDARY POOL FILE WINDOWS
F2.DPR='B'     400          ;DIRECTIVE PARTITION SUPPORT
F2.IRR='B'    1000          ;INSTALL, REQUEST AND REMOVE SUPPORT
F2.GGF='B'    2000          ;GROUP GLOBAL EVENT FLAG SUPPORT
F2.RAS='B'    4000          ;RECEIVE/SEND DATA PACKET SUPPORT
F2.AHR='B'   10000          ;ALT. HEADER REFRESH AREAS SUPPORTED
F2.RBN='B'   20000          ;ROUND ROBIN SCHEDULING SUPPORT
F2.SWP='B'   40000          ;EXECUTIVE LEVEL DISK SWAPPING SUPPORT
F2.STP='B'  100000          ;EVENT FLAG MASK IS IN THE TCB (1=YES)


;+
; THIRD FEATURE MASK SYMBOL DEFINITIONS
;-

F3.CRA='B'        1         ;SPONTANEOUS CRASH (1=YES)
F3.NWK='B'        2         ;SYSTEM HAS NETWORK SUPPORT
F3.EIS='B'        4         ;SYSTEM REQUIRES THE EXTENDED INST. SET
F3.STM='B'       10         ;SYSTEM HAS SET SYSTEM TIME DIRECTIVE
F3.UDS='B'       20         ;USER DATA SPACE (M-PLUS ONLY)
F3.PRO='B'       40         ;PROTO TCBS OUT OF POOL   "
F3.XHR='B'      100         ;EXTERNAL HEADER SUPPORT  "
F3.AST='B'      200         ;SYSTEM HAS AST SUPPORT
F3.11S='B'      400         ;SYSTEM IS RSX-11S
F3.CLI='B'     1000         ;SYSTEM HAS MULTIPLE CLI SUPPORT
F3.TCM='B'     2000         ;TERMINAL COMMON (M-PLUS ONLY)
F3.PMN='B'     4000         ;POOL MONITORING SUPPORT
F3.WAT='B'    10000         ;WATCHDOG TIMER SUPPORT
F3.RLK='B'    20000         ;'RMS' RECORD LOCKING SUPPORT
F3.SHF='B'    40000         ;MEMORY SHUFFLER SUPPORTED
;F3.RES='B'  100000         ;RESERVED FOR FUTURE EXPANSION OF 11M


; HARDWARE FEATURE MASK SYMBOL DEFINITIONS
;-

HF.UBM='B'        1      ;SYSTEM HAS A UNIBUS MAP (1=YES)
HF.EIS='B'        2      ;SYSTEM HAS EXTENDED INSTRUCTION SET
HF.QB='B'         4      ;SYSTEM HAS A QBUS BACKPLANE (1=YES)
HF.CIS='B'      200      ;SYSTEM HAS COMMERCIAL INSTRUCTION SET
HF.FPP='B'   100000      ;SYSTEM SUPPORTS FLOATING POINT (1=NO)

        .MACRO  HWDDF$  X,Y
        .ENDM
        .ENDM
```

ITBDF$

```
;
        .MACRO  ITBDF$  L,B
;
;+
; INTERRUPT TRANSFER BLOCK (ITB) OFFSET DEFINITIONS
;-

          .IF DF A$$TRP

          .MCALL PKTDF$
          PKTDF$                    ; DEFINE AST BLOCK OFFSETS

          .ENDC

          .ASECT
          .=0
X.LNK:'L' .BLKW   1                 ; LINK WORD FOR ITB LIST STARTING IN TCB
X.JSR:'L' JSR     R5,@#0            ; CALL $INTSC
X.PSW:'L' .BLKB   1                 ; LOW BYTE OF PSW FOR ISR
          .BLKB   1                 ; UNUSED
X.ISR:'L' .BLKW   1                 ; ISR ENTRY POINT (APR5 MAPPING)
X.FORK:'L'                          ; FORK BLOCK
          .BLKW   1                 ; THREAD WORD
          .BLKW   1                 ; FORK PC
          .BLKW   1                 ; SAVED R5
          .BLKW   1                 ; SAVED R4

          .IF DF M$$MGE

X.REL:'L' .BLKW   1                 ; RELOCATION BASE FOR APR5

          .ENDC

X.DSI:'L' .BLKW   1                 ; ADDRESS OF DIS.INT. ROUTINE
X.TCB:'L' .BLKW   1                 ; TCB ADDRESS OF OWNING TASK

          .IF NB SYSDEF

          .IF DF A$$TRP

          .BLKW   1                 ; A.DQSR FOR AST BLOCK
X.AST:'L' .BLKB   A.PRM             ; AST BLOCK

          .ENDC

X.VEC:'L' .BLKW   1                 ; VECTOR ADDRESS (IF AST SUPPORT,
                                    ; THIS IS FIRST AND ONLY AST
                                    ;PARAMETER)
X.VPC:'L' .BLKW   1                 ; SAVED VECTOR PC
X.LEN:'L'                           ; LENGTH IN BYTES OF ITB

          .ENDC

          .PSECT
          .MACRO  ITBDF$  X,Y,Z
          .ENDM   ITBDF$
          .ENDM   ITBDF$
```

LCBDF$

```
;
          .MACRO  LCBDF$,L,B
;+
; LOGICAL ASSIGNMENT CONTROL BLOCK
;
; THE LOGICAL ASSIGNMENT CONTROL BLOCK (LCB) IS USED TO ASSOCIATE A
; LOGICAL NAME WITH A PHYSICAL DEVICE UNIT.  LCB'S ARE LINKED TOGETHER
; TO FORM THE LOGICAL ASSIGNMENTS OF A SYSTEM.  ASSIGNMENTS MAY BE ON
; A SYSTEM WIDE OR LOCAL (TERMINAL) BASIS.
;-

       .ASECT
.=0
L.LNK:'L'   .BLKW 1              ;LINK TO NEXT LCB
L.NAM:'L'   .BLKW 1              ;LOGICAL NAME OF DEVICE
L.UNIT:'L'  .BLKB 1              ;LOGICAL UNIT NUMBER
L.TYPE:'L'  .BLKB 1              ;TYPE OF ENTRY (0=SYSTEM WIDE)
L.UCB:'L'   .BLKW 1              ;TI UCB ADDRESS
L.ASG:'L'   .BLKW 1              ;ASSIGNMENT UCB ADDRESS
L.LGTH='B'  .-L.LNK              ;LENGTH OF LCB
            .PSECT

            .MACRO       LCBDF$,X,Y
            .ENDM
            .ENDM
```

MTADF$

```
          .MACRO    MTADF$,L,B
          .ASECT
;
; ANSI MAGTAPE SPECIFIC DATA STRUCTURES
;
; VOLUME SET CONTROL BLOCK OFFSET DEFININTIONS (VSCB)
;
; VOLUME SET AND PROCESS CONTROL SECTION
;
.=0
V.TCNT:'L' .BLKW 1        ;TRANSACTION COUNT
V.TYPE:'L' .BLKB 1        ;VOLUME TYPE DESCRIPTOR
V.VCHA:'L' .BLKB 1        ;VOLUME CHARACTERISTICS
V.LABL:'L' .BLKB 12.      ;FILE SET ID (FIRST SIX BYTES)
V.NXT:'L'  .BLKW 1        ;PTR TO NEXT VSCB NODE
V.MVL:'L'  .BLKW 1        ;PTR TO MOUNTED VOL LIST
V.UVL:'L'  .BLKW 1        ;PTR TO UNMOUNTED VOL LIST
V.ATL:'L'  .BLKW 1        ;ATL ADDR OF ACCESSING TASK
                          ; TCB IN RSX11M
V.UCB:'L'  .BLKW 1        ;ADDR OF CURRENT UCB OR PUD
V.RVOL:'L' .BLKB 1        ;CURRENT RELATIVE VOL #
V.MOU:'L'  .BLKB 1        ;MOUNT MODE BYTE
V.TCHR:'L' .BLKW 1        ;UINT CHAR. FOR ALL UNITS USED FOR VOL SET
V.SEQN:'L' .BLKW 1        ;CURRENT FILE SEQUENCE #
V.SECN:'L' .BLKW 1        ;CURRENT FILE SECTION #
V.TPOS:'L' .BLKB 1        ;POSITION OF TAPE IN TM'S TO NXT HDR1
V.PSTA:'L' .BLKB 1        ;PROCESS STATUS BYTE
V.TIMO:'L' .BLKW 1        ;BLOCKED PROCESS TIMEOUT COUNTER
V.STAT:'L' .BLKW 3        ; STATUS WORDS USED BY COMMAND
                          ;EXECUTION MODULES
V.TRTB:'L' .BLKB 1        ;TRANSLATION CONTROL BYTE
V.EFTV:'L' .BLKB 1        ;FOR MAG TO RETURN IE.EOF, EOT, EOV
;
; LABEL DATA SECTION
;
V.BLKL:'B' .BLKW 1        ;BLOCK LENGTH
V.RECL:'B' .BLKW 1        ;RECORD LENGTH
V.FNAM:'L' .BLKW 3        ;FILE NAME
V.FTYP:'L' .BLKW 1        ;FILE TYPE
V.FVER:'L' .BLKW 1        ;FILE VERSION #
V.CDAT:'L' .BLKW 2        ;CREATION DATE
V.EDAT:'L' .BLKW 2        ;EXPRIATION DATE
V.BLKC:'L' .BLKW 2        ;BLOCK COUNT FOR FILE SECTION
V.RTYP:'L' .BLKB 1        ;RECORD TYPE
V.FATT:'L' .BLKB 1        ;FILE ATTRIBUTES FOR CARRIAGE CONTROL
          .BLKB 30.       ;REMAINDER OF FILE ATTRIBUTES
;
; NULL WINDOW SECTION
;
V.WIND:'L' .BLKW  4.      ; NULL WINDOW
       ;*****
```

MTADF$ (Cont.)

```
V.MST2:'L' .BLKW  1       ;MAGTAPE STATUS BITS
V.FABY:'L' .BLKB  1       ;FILE ACCESSIBILITY BYTE (HDR1)
           .BLKB  1       ;SPARE
V.ANSN:'L' .BLKB 17.      ;ANSI 17 CHARACTER FILE NAME
V.BOFF:'L' .BLKB  1.      ;BUFFER OFFSET
V.DENS:'L' .BLKB  1.      ;REQUESTED UNIT DENSITY
V.DRAT:'L' .BLKB  1.      ;DEFAULT RECORD ATTRIBUTES
V.DBLK:'L' .BLKW  1.      ;DEFAULT BLOCK SIZE
V.DREC:'L' .BLKW  1.      ;DEFAULT RECORD SIZE
       ;*****
S.VSCB=.                  ;SIZE OF VSCB
           .PSECT
;
; DEFINE OFFSETS INTO NULL WINDOW SECTION
;
           .ASECT
.=0
       W.CTL: .BLKW 1           ;CONTROL WORD IN WINDOW
       V.WINC=V.WIND+W.CTL      ; CNTRL WORD IN NULL WINDOW

           .PSECT               ; RELATIVE TO THE VSCB
;
; MOUNTED VOLUME LIST OFFSET DEFININTIONS (MVL)
;
           .ASECT
.=0
           .IF DF R$$11M
M.NXT:'L'  .BLKW 1              ;PTR TO NXT MVL NODE (11M)
           .ENDC
M.UIC:'L'  .BLKW 1              ;OWNER UIC FROM RVOL #1
M.CH:'L'   .BLKW 1              ; U.CH/U.VP (11D)
M.PROT:'L' .BLKW 1              ;PROTECTION U.AR IN 11D
           .IF  NDF R$$11M
           .BLKW 2              ; ACP WORDS 11D
M.NXT:'L'  .BLKW 1              ;PTR TO NEXT MVL NODE (11D)
           .ENDC
M.RVOL:'L' .BLKB 1              ;RELATIVE VOL # OF MOUNTED VOLUME
M.STAT:'L' .BLKB 1              ;VOLUME STATUS
M.VIDP:'L' .BLKW 1              ;VOLUME ID POINTER
M.UCB:'L'  .BLKW 1              ;ADDR OF ASSOC UCB OR PUD
S.MVL=.                         ;SIZE OF MVL NODE
           .PSECT
;
; UNMOUNTED VOLUME AND VOLUME  LIST OFFSET DEFINITIONS (UVL)
;
           .ASECT
.=0
L.NXT:'L'  .BLKW 1              ;PTR TO NXT UVL NODE
L.VOL1:'L' .BLKB 1              ;REL VOL # OF 1'ST VOL IN NODE
L.VOL2:'L' .BLKB 1              ;REL VOL # OF 2'ND VOL IN NODE
L.VID1:'L' .BLKB 6              ;VOL ID OF 1'ST VOL IN NODE
L.VID2:'L' .BLKB 6              ;VOL ID OF 2'ND VOL IN NODE
S.UVL=.                         ;SIZE OF UVL NODE
           .PSECT
;
;
; SYSTEM DATA STRUCTURE CONTENT VALUES
;

; VSCB VALUES
```

MTADF$ (Cont.)

```
; V.MOU VALUES
VM.OLD='B'  200              ;OLD .FL300 VOLUME -- VM.BYP WILL ALSO
                             ; BE SET
VM.BYP='B'  100              ;BYPASS LABEL PROCESSING
VM.ULB='B'   40              ;UNLABELED TAPE
VM.FSC='B'   20              ;OVERRIDE FILE SET ID CHECK
VM.EXC='B'   10              ;OVERRIDE EXPIRATION DATE CHECK

; V.MST2 VALUES[[B

V2.INI='B'  1               ;MAG WANTS US TO INITIALIZE NEXT OUTPUT
V2.XH2='B'  2               ;THIS FILE HAS NO HDR2, DON'T WRITE EOF2
V2.XH3='B'  4               ;THIS FILE HAS NO HDR3, DON'T WRITE EOF3
V2.NH3='B' 10               ;DON'T WRITE HDR3/EOX3 LABELS
V2.OAC='B' 20               ;OVERRIDE FILE/VOLUME ACCESSIBILITY

; V.PSTA VALUES - UNBLOCKED TRANSITION STATE

VP.RM='B'    2               ;READ DATA MODE
VP.WM='B'    4               ;WRITE DATA MODE
VP.UCM='B'   6               ;UNLABELLED CREATE POSITIONING MODE
VP.SM='B'   10               ;SEARCH MODE
VP.MOU='B'  20               ;MOUNT MODE
VP.RWD='B'  40               ;REWIND OR VOL VERIFICATION WAIT
VP.VFY='B' VP.RWD
VP.POS='B' 100               ;PROCESS IN POSITIONING MODE (MULTI-SECTION
                             ;FILE)
;
; BLOCKED STATE = -(UNBLOCKED TRANSITION STATE VALUES)
;
;PROCESS TIMED OUT BIT 0 = 1
       VP.TO=1
;
;
; NULL WINDOW CONTROL BIT DEFINITIONS
;
WI.RDV='B'   400              ;ACCESSED FOR READ
WI.WRV='B'  1000              ;ACCESSED FOR WRITE
WI.EXT='B'  2000              ;ACCESSED FOR EXTEND
WI.LCK='B'  4000              ;LOCKED
;

;
; MVL VALUES IN THE M.STAT FIELD
;
MS.VER='B'  200              ;VOL ID NOT VERIFIED
MS.RID='B'    1              ;VOL ID TO BE READ NOT CHECKED
MS.NMO='B'    2              ;MOUNT MESSAGE NOT GIVEN YET
MS.TMO='B'    4              ;ONE TIMEOUT ALREADY EXPRIED
MS.EXP='B'   10              ;EXPIRATION DATE MESSAGE GIVEN
;
;
; MISC BITS USED IN MOUNT (STORED IN V.STS)
;
MO.OVR='B'    1              ; OVER RIDE VOL NAME SWITCH
MO.UIC='B'    2              ; EXPLICIT UIC GIVEN
MO.PRO='B'    4              ; EXPLICIT PROTECTION GIVEN
MO.160='B'   10              ; 1600 BPI SPECIFIED
;
       .ENDM
```

PCBDF$

```
          .MACRO  PCBDF$  L,B

;+
; PARTITION CONTROL BLOCK OFFSET DEFINITIONS
;-

          .ASECT
.=0
P.LNK:'L' .BLKW  1              ;LINK TO NEXT PARTITION PCB
P.PRI:'L' .BLKB  1              ;PRIORITY OF PARTITION
P.IOC:'L' .BLKB  1              ;I/O + I/O STATUS BLOCK COUNT
P.NAM:'L' .BLKW  2              ;PARTITION NAME IN RAD50
P.SUB:'L' .BLKW  1              ;POINTER TO NEXT SUBPARTITION
P.MAIN:'L'.BLKW  1              ;POINTER TO MAIN PARTITION


          .IF NB  SYSDEF


          .IF NDF M$$MGE

P.HDR:'L'                      ;POINTER TO HEADER CONTROL BLOCK

          .ENDC


          .IFTF

P.REL:'L' .BLKW 1               ;STARTING PHYSICAL ADDRESS OF PARTITION
P.BLKS:'L'
P.SIZE:'L'.BLKW 1               ;SIZE OF PARTITION IN:
                                ;   UNMAPPED SYSTEMS - BYTES
                                ;   MAPPED SYSTEMS   - 32 WORD BLOCKS
P.WAIT:'L'.BLKW 1               ;PARTITION WAIT QUEUE LISTHEAD (2 WORDS)
P.SWSZ:'L'.BLKW 1               ;PARTITION SWAP SIZE (SYSTEM ONLY)
P.BUSY:'L'.BLKB 2               ;PARTITION BUSY FLAGS
P.OWN:'L'
P.TCB:'L' .BLKW 1               ;TCB ADDRESS OF OWNER TASK
P.STAT:'L'.BLKW 1               ;PARTITION STATUS FLAGS

          .IFT


          .IF DF       M$$MGE

P.HDR:'L' .BLKW 1               ;POINTER TO HEADER CONTROL BLOCK

          .ENDC


P.PRO:'L' .BLKW 1               ;PROTECTION WORD [DEWR,DEWR,DEWR,DEWR]
P.ATT:'L' .BLKW 2               ;ATTACHMENT DESCRIPTOR LISTHEAD


          .IF NDF P$$LAS

P.LGTH='B'P.PRO                 ;LENGTH OF PARTITION CONTROL BLOCK

          .IFF
```

PCBDF$ (Cont.)

```
P.LGTH='B'.                     ;LENGTH OF PARTITION CONTROL BLOCK

          .ENDC
          .IFF

          .PSECT

;+
; PARTITION STATUS WORD BIT DEFINITIONS
;-

PS.OUT='B' 100000          ;PARTITION IS OUT OF MEMORY(1=YES)
PS.CKP='B'  40000          ;PARTITION CHECKPOINT IN PROGRESS (1=YES)
PS.CKR='B'  20000          ;PARTITION CHECKPOINT IS REQUESTED (1=YES)
PS.CHK='B'  10000          ;PARTITION IS NOT CHECKPOINTABLE (1=YES)
PS.FXD='B'   4000          ;PARTITION IS FIXED (1=YES)
PS.PER='B'   2000          ;PARITY ERROR IN PARTITION (1=YES)
PS.LIO='B'   1000          ;MARKED BY SHUFFLER FOR LONG I/O (1=YES)
PS.NSF='B'    400          ;PARTITION IS NOT SHUFFLEABLE (1=YES)
PS.COM='B'    200          ;LIBRARY OR COMMON BLOCK (1=YES)
PS.PIC='B'    100          ;POSITION INDEPENDENT LIBRARY OR COMMON
                           ;(1=YES)
PS.SYS='B'     40          ;SYSTEM CONTROLLED PARTITION (1=YES)
PS.DRV='B'     20          ;DRIVER IS LOADED IN PARTITION (1=YES)
PS.DEL='B'     10          ;PARTITION SHOULD BE DELETED WHEN NOT
                           ;ATTACHED (1=YES)
PS.APR='B'      7          ;STARTING APR NUMBER MASK

;+
; ATTACHMENT DESCRIPTOR OFFSETS
;-

       .ASECT
.=0
A.PCBL:'L' .BLKW 1         ;PCB ATTACHMENT QUEUE THREAD WORD
A.PRI:'L'  .BLKB 1         ;PRIORITY OF ATTACHED TASK
A.IOC:'L'  .BLKB 1         ;I/O COUNT THROUGH THIS DESCRIPTOR
A.TCB:'L'  .BLKW 1         ;TCB ADDRESS OF ATTACHED TASK
A.TCBL:'L' .BLKW 1         ;TCB ATTACHMENT QUEUE THREAD WORD
A.STAT:'L' .BLKB 1         ;STATUS BYTE
A.MPCT:'L' .BLKB 1         ;MAPPING COUNT OF TASK THRU THIS DESCRIPTOR
A.PCB:'L'  .BLKW 1         ;PCB ADDRESS OF ATTACHED TASK
A.LGTH='B' .               ;LENGTH OF ATTACHMENT DESCRIPTOR

;+
; ATTACHMENT DESCRIPTOR STATUS BYTE BIT DEFINITIONS
;-

       .PSECT
AS.DEL='B' 10          ;TASK HAS DELETE ACCESS (1=YES)
AS.EXT='B'  4          ;TASK HAS EXTEND ACCESS (1=YES)
AS.WRT='B'  2          ;TASK HAS WRITE ACCESS (1=YES)
AS.RED='B'  1          ;TASK HAS READ ACCESS (1=YES)

       .ENDC


       .MACRO      PCBDF$      X,Y,Z
       .ENDM
       .ENDM
```

```
        .MACRO PKTDF$,L,B

;+
; ASYNCHRONOUS SYSTEM TRAP CONTROL BLOCK OFFSET DEFINITIONS
;
; SOME POSITIONAL DEPENDENCIES BETWEEN THE OCB AND THE AST CONTROL
; BLOCK ARE RELIED UPON IN THE ROUTINE $FINXT IN THE MODULE SYSXT.
;-

        .ASECT
.=177774
A.KSR5:'L' .BLKW 1      ;SUBROUTINE KISAR5 BIAS (A.CBL=0)
A.DQSR:'L' .BLKW 1      ;DEQUEUE SUBROUTINE ADDRESS (A.CBL=0)
           .BLKW 1      ;AST QUEUE THREAD WORD
A.CBL:'L'  .BLKW 1      ;LENGTH OF CONTROL BLOCK IN BYTES
                        ;IF A.CBL = 0, THE AST CONTROL BLOCK IS
                        ;TO BE DEALLOCATED BY THE DEQUEUE SUBROUTINE
                        ;POINTED TO BY A.DQSR MAPPED VIA APR 5
                        ;VALUE A.KSR5. THIS IS CURRENTLY USED ONLY
                        ;BY THE FULL DUPLEX TERMINAL DRIVER FOR
                        ;UNSOLICITED CHARACTER ASTS.
                        ;IF THE LOW BYTE OF A.CBL = 0, AND THE
                        ;HIGH BYTE IS NOT = 0, THE AST CONTROL BLOCK
                        ;IS A SPECIFIABLE AST, WITH LENGTH, C.LGTH.
                        ;IF THE HIGH BYTE OF A.CBL = 0 AND THE LOW
                        ;BYTE > 0, THEN THE LOW BYTE IS THE LENGTH
                        ;OF THE AST CONTROL BLOCK.  IF THE HIGH BYTE
                        ;OF A.CBL = 0 AND THE LOW BYTE IS NEGATIVE,
                        ;THIS IS A KERNEL AST. SEE BELOW FOR
                        ;A DESCRIPTION OF A.CBL FOR KERNEL ASTS.
A.BYT:'L'  .BLKW 1      ;NUMBER OF BYTES TO ALLOCATE ON TASK STACK
A.AST:'L'  .BLKW 1      ;AST TRAP ADDRESS
A.NPR:'L'  .BLKW 1      ;NUMBER OF AST PARAMETERS
A.PRM:'L'  .BLKW 1      ;FIRST AST PARAMETER
;
; THE SPECIFIABLE AST CODES MUST NOT BE 0.
;
AS.FPA='B' 1            ;CODE FOR FLOATING POINT AST
AS.RCA='B' 2            ;CODE FOR RECEIVE DATA AST
AS.RRA='B' 3            ;CODE FOR RECEIVE BY REFERENCE AST
AS.PFA='B' 4            ;CODE FOR POWERFAIL AST
AS.REA='B' 5            ;CODE FOR REQUESTED EXIT (ABORT) AST
AS.CAA='B' 6            ;CODE FOR COMMAND ARRIVAL AST FOR CLIS
;
; BIT VALUES IN A.PRM+5
;
AF.XCC='B'   1          ;ATTACHED FOR ALL BUT CONTROL-C (TF.XCC)
AF.NOT='B'   2          ;ATTACHED FOR ALL NOTIFICATION (TF.NOT)
AF.OOB='B'   4          ;ACB IS FOR OUT-OF-BAND AST
AF.AST='B'  10          ;ACB HANDLES UNSOL. INPUT CHAR AST'S (TF.AST)
AF.ESQ='B'  20          ;ATTACHED FOR ESCAPE SEQUENCES (TF.ESQ)
AF.LCK='B'  40          ;ACB IS LOCKED
AF.QUE='B' 100          ;ACB IS QUEUED
AF.MDE='B' 200          ;ACB IS MARKED FOR DELETE
;
;
;
```

PKTDF$ (Cont.)

```
; ABORTER SUBCODES FOR ABORT AST (AS.REA) TO BE RETURNED ON USER'S
; STACK
;
AB.NPV='B'   1          ;ABORTER IS NONPRIVILEGED (1=YES)
AB.TYP='B'   2          ;ABORT FROM DIRECTIVE (0=YES)
                        ;ABORT FROM CLI COMMAND (1=YES)

;+
; KERNEL AST CONTROL BLOCK DEFINITIONS
;
; THE LOW BYTE OF A.CBL FOR A KERNEL AST HAS THE FOLLOWING FORMAT:
;
;       BIT #200 ALWAYS EQUALS 1
;       BIT #100 IS ZERO IF $SGFIN MUST BE CALLED DURING AST
;       PROCESSING THE REMAINING SIX BITS ARE USED AS THE
;       KERNEL AST TYPE FIELD
;
; BECAUSE THERE ARE ONLY 6 BITS AVAILABLE TO THE KERNEL AST
; INDEX FIELD, ONLY (2**6)-1 KERNEL AST TYPES ARE POSSIBLE.
;-

AK.BUF='B' 200          ;BUFFERED I/O COMPLETION AST
AK.OCB='B' 201          ;OFFSPRING EXIT
AK.GBI='B' 202          ;GENERAL BUFFERED I/O AST
AK.GGF='B' 303          ;GROUP GLOBAL RUNDOWN AST

;+
; OFFSPRING CONTROL BLOCK DEFINITIONS
;
; SOME POSITIONAL DEPENDENCIES EXIST BETWEEN THE OCB AND THE AST
; CONTROL BLOCK IN ROUTINE $FINXT IN MODULE SYSXT
;-

.=0
O.LNK:'L'   .BLKW 1     ;OCB LINK WORD
O.MCRL:'L'  .BLKW 1     ;ADDRESS OF MCR COMMAND LINE
O.PTCB:'L'  .BLKW 1     ;PARENT TCB ADDRESS
O.AST:'L'   .BLKW 1     ;EXIT AST ADDRESS
O.EFN:'L'   .BLKW 1     ;EXIT EVENT FLAG
O.ESB:'L    .BLKW 1     ;EXIT STATUS BLOCK VIRTUAL ADDRESS
O.STAT:'L'  .BLKW 8.    ;EXIT STATUS BUFFER
O.LGTH='B'  .           ;LENGTH OF OCB

;+
; I/O PACKET OFFSET DEFINITIONS
;-

          .ASECT
.=0
I.LNK:'L'   .BLKW 1     ;I/O QUEUE THREAD WORD
I.PRI:'L'   .BLKB 1     ;REQUEST PRIORITY
I.EFN:'L'   .BLKB 1     ;EVENT FLAG NUMBER
I.TCB:'L'   .BLKW 1     ;TCB ADDRESS OF REQUESTOR
I.LN2:'L'   .BLKW 1     ;POINTER TO SECOND LUN WORD
I.UCB:'L'   .BLKW 1     ;POINTER TO UNIT CONTROL BLOCK
I.FCN:'L'   .BLKW 1     ;I/O FUNCTION CODE
```

PKTDF$ (Cont.)

```
I.IOSB:'L' .BLKW 1      ;VIRTUAL ADDRESS OF I/O STATUS BLOCK
           .BLKW 1      ;I/O STATUS BLOCK RELOCATON BIAS
           .BLKW 1      ;I/O STATUS BLOCK ADDRESS
I.AST:'L'  .BLKW 1      ;AST SERVICE ROUTINE ADDRESS
I.PRM:'L'  .BLKW 1      ;RESERVED FOR MAPPING PARAMETER #1
           .BLKW 6      ;PARAMETERS 1 TO 6
           .BLKW 1      ;USER MODE DIAGNOSTIC PARAMETER WORD
;
; FOLLOWING ARE DEFINITIONS FOR FLAG BITS IN I.PRM+11
; (DSA DRIVERS INTERNAL USE ONLY)
;

IP.FAK='B' 20           ;IOP IS PSEUDO IOP
IP.ABO='B' 40           ;(MUDRV)ABORT COMMAND MUST BE ISSUED FOR IOP
IP.PND='B' 100          ;(MUDRV)ABORT COMMAND WAS ISSUED FOR IOP
IP.UMR='B' 200          ;A UMR WAIT BLOCK IS IN USE FOR THIS I/O

I.ATTL='B'.             ;MINIMUM LENGTH OF I/O PACKET (USED BY
                        ;FILE SYSTEM TO CALCULATE MAXIMUM
                        ;NUMBER OF ATTRIBUTES)
I.LGTH='B'.             ;LENGTH OF I/O REQUEST CONTROL BLOCK

;
; DEFINE OFFSETS IN I/O PACKET EXTENSION (IOPX)
;
          .ASECT
.=0

I.XLNK:'L' .BLKW 1      ;LINK WORD
I.XIOP:'L' .BLKW 1      ;I/O PACKET ADDRESS
I.XTCB:'L' .BLKW 1      ;TCB ADDRESS OF REQUESTING TASK
I.XMOD:'L' .BLKW 2      ;MODIFIER WORDS (NOTE: 2ND WORD MUST BE
                        ;SPECIFIED AND MUST BE ZERO.)
I.XRBF:'L' .BLKW 2      ;READ DATA BUFFER ADDRESS APR BIAS
                        ;READ DATA BUFFER VIRTUAL ADDRESS
I.XRBL:'L' .BLKW 1      ;READ DATA BUFFER LENGTH
I.XTMO:'L' .BLKW 1      ;READ TIME-OUT INTERVAL
I.XPBF:'L' .BLKW 2      ;PROMPT BUFFER ADDRESS APR BIAS
                        ;PROMPT BUFFER VIRTUAL ADDRESS
I.XPBL:'L' .BLKW 1      ;PROMPT BUFFER LENGTH
I.XPBV:'L' .BLKW 1      ;PROMPT BUFFER VERTICAL FORMS CONTROL
I.XTTB:'L' .BLKW 2      ;TERMINATOR TABLE ADDRESS APR BIAS
                        ;TERMINATOR TABLE VIRTUAL ADDRESS
I.XTTL:'L' .BLKW 1      ;TERMINATOR TABLE LENGTH
I.XDBF:'L' .BLKW 2      ;DEFAULT INPUT BUFFER ADDRESS APR BIAS
                        ;DEFAULT INPUT BUFFER VIRTUAL ADDRESS
I.XDBL:'L' .BLKW 1      ;DEFAULT INPUT BUFFER LENGTH

;+
; GROUP GLOBAL EVENT FLAG CONTROL BLOCK OFFSETS
;-

.=0
G.LNK:'L'  .BLKW 1      ;LINK WORD
G.GRP:'L'  .BLKB 1      ;GROUP NUMBER
G.STAT:'L' .BLKB 1      ;STATUS BYTE
G.CNT:'L'  .BLKW 1      ;ACCESS COUNT
G.EFLG:'L' .BLKW 2      ;EVENT FLAGS
G.LGTH='B' .            ;LENGTH OF GROUP GLOBAL CONTROL BLOCK
```

PKTDF$ (Cont.)

```
; STATUS BYTE DEFINITIONS

GS.DEL='B'1             ;GROUP MARKED FOR DELETE


;+
; EXECUTIVE POOL MONITOR CONTROL FLAGS
;-

; $POLST IS THE SYNCHRONIZATION WORD BETWEEN THE EXEC AND POOL MONITOR

PC.HIH='B'    1         ;HIGH POOL LIMIT CROSSED (1=YES)
PC.LOW='B'    2         ;LOW POOL LIMIT CROSSED (1=YES)
PC.ALF='B'    4         ;FAILED TO ALLOCATE LARGE BLOCK (1=YES)
PC.XAF='B'   10         ;FAILED TO ALLOCATE SMALL BLOCK (1=YES)
PC.XIT='B'  200         ;FORCE POOL MONITOR TASK TO EXIT
PC.NRM='B'  PC.HIH*400  ;POOL TASK INHIBIT BIT FOR HIGH POOL
PC.ALM='B'  PC.LOW*400  ;POOL TASK INHIBIT BIT FOR LOW POOL

; $POLFL IS THE POOL USAGE CONTROL WORD

PF.INS='B'   40         ;REJECT NONPRIVILEGED INS/RUN/REM
PF.LOG='B'  100         ;LOGINS ARE DISABLED
PF.REQ='B'  200         ;STALL REQUEST OF NONPRIV. TASKS

PF.ALL='B' 177777       ;TAKE ALL POSSIBLE ACTIONS TO SAVE POOL


;+
; CLI PARSER BLOCK (CPB) DEFINITIONS
;-

.=0
C.PTCB:'L' .BLKW 1      ;ADDRESS OF CLI'S TCB
C.PNAM:'L' .BLKW 2      ;CLI NAME
C.PSTS:'L' .BLKW 1      ;STATUS MASK
C.PDPL:'L' .BLKB 1      ;LENGTH OF DEFAULT PROMPT
C.PCPL:'L' .BLKB 1      ;LENGTH OF CNTRL/C PROMPT
C.PRMT:'L'              ;START OF ASCII PROMPT STRINGS
                        ;THE DEFAULT STRING IS CONCANTENATED
                        ;WITH THE ^C STRING
;
; STATUS BIT DEFINITIONS
;
CP.NUL='B'    1         ;PASS EMPTY COMMAND LINES TO CLI
CP.MSG='B'    2         ;CLI DESIRES SYSTEM MESSAGES
CP.LGO='B'    4         ;CLI WANTS COMMANDS FROM LOGGED OFF TTYS
CP.DSB='B'   10         ;CLI IS DISABLED
CP.PRV='B'   20         ;USER MUST BE PRIV TO SET TTY TO THIS CLI
CP.SGL='B'   40         ;DON'T HANDLE CONTINUATIONS (M-PLUS ONLY)
CP.NIO='B'  100         ;MCR..., HEL, BYE DO NO I/O TO TTY
                        ;HEL, BYE ALSO DO NOT SET CLI ETC.
CP.RST='B'  200         ;ABILITY TO SET TO THIS CLI IS RESTRICTED
                        ;TO THE CLI ITSELF
CP.EXT='B'  400         ;PASS TASK EXIT PROMPT REQUESTS TO CLI
;
; IDENTIFIER CODES FOR SYSTEM TO CLI MESSAGES.
;
; CODES 0 - 127. ARE RESERVED FOR USE BY DIGITAL,
; CODES 128. - 255. ARE RESERVED FOR USE BY CUSTOMERS
```

PKTDF$ (Cont.)

```
;
CM.INE='B' 1            ;CLI INITIALIZED ENABLED
CM.IND='B' 2            ;CLI INITIALIZED DISABLED
CM.CEN='B' 3            ;CLI ENABLED
CM.CDS='B' 4            ;CLI DISABLED
CM.ELM='B' 5            ;CLI BEING ELIMINATED
CM.EXT='B' 6            ;CLI MUST EXIT IMMEDIATELY
CM.LKT='B' 7            ;NEW TERMINAL LINKED TO CLI
CM.RMT='B' 8.           ;TERMINAL REMOVED FROM CLI
CM.MSG='B' 9.           ;GENERAL MESSAGE TO CLI


;+
; ANCILLARY CONTROL BLOCK (ACB) DEFINITIONS
;-

.=0
A.REL:'L'   .BLKW 1     ;ACD RELOCATION BIAS
A.DIS:'L'   .BLKW 1     ;ACD DISPATCH TABLE POINTER
A.MAS:'L'   .BLKW 2     ;ACT FUNCTION MASK
A.NUM:'L'   .BLKB 1     ;ACD IDENTIFICATION NUMBER
A.FLEN:'L'  .BLKB 1     ;LENGTH IN BYTES OF FULL ACB
A.LIN:'L'   .BLKW 1     ;ACD LINK WORD
A.ACC:'L'   .BLKB 1     ;ACD ACCESS COUNT
A.STA:'L'   .BLKB 1     ;ACD STATUS BYTE
A.PLEN='B'  .           ;LENGTH IN BYTES OF PROTOTYPE ACB
;
.=A.LIN                 ;FULL ACB OVERLAPS PROTOTYPE ACB
A.IMAP:'L'  .BLKW 1     ;ACD INTERRUPT BUFFER RELOCATION BIAS
A.IBUF:'L'  .BLKW 1     ;ACD INTERRUPT BUFFER ADDRESS
A.ILEN:'L'  .BLKW 1     ;ACD INTERRUPT BUFFER LENGTH
A.SMAP:'L'  .BLKW 1     ;ACD SYSTEM STATE BUFFER RELOCATION BIAS
A.SBUF:'L'  .BLKW 1     ;ACD SYSTEM STATE BUFFER ADDRESS
A.SLEN:'L'  .BLKW 1     ;ACD SYSTEM STATE BUFFER LENGTH
A.IOS:'L'   .BLKW 2     ;ACD I/O STATUS
A.RES='B'   .           ;START OF ACB RESERVED FOR USE BY THE ACD
;
; DEFINE THE FLAG VALUES IN THE OFFSET U.AFLG
;
UA.ACC='B'     1        ;ACCEPT THIS CHARACTER
UA.PRO='B'     2        ;PROCESS THIS CHARACTER
UA.ECH='B'     4        ;ECHO THIS CHARACTER
UA.TYP='B'    10        ;FORCE THIS CHARACTER INTO TYPEAHEAD
UA.SPE='B'    20        ;THIS CHARACTER HAS A SPECIAL ECHO
UA.PUT='B'    40        ;PUT THIS CHARACTER IN THE INPUT BUFFER
UA.CAL='B'   100        ;CALL THE ACD BACK AFTER THE TRANSFER
UA.COM='B'   200        ;COMPLETE THE INPUT REQUEST
;
UA.ALL='B'   400        ;ALLOW PROCESSING OF THIS I/O REQUEST
UA.TRN='B' 1000         ;TRANSLATE CHARACTERS FROM OUTPUT QIO
UA.TRA='B' 2000         ;TRANSFER CHARACTERS WHEN I/O COMPLETES
;
; DEFINE THE ACD ENTRY POINTS (OFFSETS INTO THE DISPATCH TABLE)
;
.=0
A.ACCE:'L'  .BLKW 1     ;I/O REQUEST ACCEPTANCE ENTRY POINT
A.DEQU:'L'  .BLKW 1     ;I/O REQUEST DEQUEUE ENTRY POINT
A.POWE:'L'  .BLKW 1     ;POWER FAILURE ENTRY POINT
A.INPU:'L'  .BLKW 1     ;INPUT COMPLETION ENTRY POINT
```

PKTDF$ (Cont.)

```
A.OUTP:'L' .BLKW 1      ;OUTPUT COMPLETION ENTRY POINT
A.CONN:'L' .BLKW 1      ;CONNECTION ENTRY POINT
A.DISC:'L' .BLKW 1      ;DISCONNECTION ENTRY POINT
A.RECE:'L' .BLKW 1      ;INPUT CHARACTER RECEPTION ENTRY POINT
A.PROC:'L' .BLKW 1      ;INPUT CHARACTER PROCESSING ENTRY POINT
A.TRAN:'L' .BLKW 1      ;OUTPUT QIO CHARACTER TRANSLATION ENTRY POINT
A.CALL:'L' .BLKW 1      ;CALL ACD BACK AFTER TRANSFER ENTRY POINT
;
; DEFINE THE STATUS BITS IN A.STA OF THE PROTOTYPE ACB
;
AS.DLT='B' 1            ;ACD IS MARKED FOR DELETE
AS.DIS='B' 2            ;ACD IS DISABLED
           .PSECT

           .MACRO  PKTDF$ X,Y,Z
           .ENDM
           .ENDM
```

SCBDF$

```
; THE STATUS CONTROL BLOCK (SCB) DEFINES THE STATUS OF A DEVICE
; CONTROLLER.  THERE IS ONE SCB FOR EACH CONTROLLER IN A SYSTEM.
; THE SCB IS POINTED TO BY UNIT CONTROL BLOCKS. TO EXPAND ON THE
; TELETYPE EXAMPLE ABOVE, EACH TELETYPEWRITER TYPE INTERFACED VIA
; A DL11-A WOULD HAVE A SCB SINCE EACH DL11-A IS AN INDEPENDENT
; INTERFACE UNIT. THE TELETYPES INTERFACED VIA THE DH11 WOULD
; ALSO EACH HAVE AN SCB SINCE THE DH11 IS A SINGLE CONTROLLER BUT
; MULTIPLEXES MANY UNITS IN PARALLEL.
;-

            .ASECT
.=177772
S.RCNT:'L'  .BLKB 1         ;NUMBER OF REGISTERS TO COPY ON ERROR
S.ROFF:'L'  .BLKB 1         ;OFFSET TO FIRST DEVICE REGISTER
S.BMSV:'L'  .BLKW 1         ;SAVED I/O ACTIVE BITMAP AND POINTER TO EMB
S.BMSK:'L'  .BLKW 1         ;DEVICE I/O ACTIVE BIT MASK
S.LHD:'L'   .BLKW 2         ;CONTROLLER I/O QUEUE LISTHEAD
S.PRI:'L'   .BLKB 1         ;DEVICE PRIORITY
S.VCT:'L'   .BLKB 1         ;INTERRUPT VECTOR ADDRESS /4
S.CTM:'L'   .BLKB 1         ;CURRENT TIMEOUT COUNT
S.ITM:'L'   .BLKB 1         ;INITIAL TIMEOUT COUNT
S.CON:'L'   .BLKB 1         ;CONTROLLER INDEX
S.STS:'L'   .BLKB 1         ;CONTROLLER STATUS (0=IDLE,1=BUSY)
S.CSR:'L'   .BLKW 1         ;ADDRESS OF CONTROL STATUS REGISTER
S.PKT:'L'   .BLKW 1         ;ADDRESS OF CURRENT I/O PACKET
S.FRK:'L'   .BLKW 1         ;FORK BLOCK LINK WORD
S.DMCS:'L'                  ;DM11-BB CSR FOR FDX TTDRV
            .BLKW 1         ;FORK-PC
            .BLKW 1         ;FORK-R5
            .BLKW 1         ;FORK-R4


            .IF NB SYSDEF


            .IF DF L$$DRV & M$$MGE

            .BLKW 1         ;FORK-DRIVER RELOCATION BASE

            .ENDC


S.PORT:'L'                  ;FIRST THREE CHAR. OF PORT NAME (RAD50)
S.PBIA='B'  S.PORT+2        ;BIAS OF PORT COMMON
S.QST='B'   S.PORT+4        ;ADDRESS OF QST, CONTROLLER STATE TABLE
S.BSYU='B'  S.PORT+6        ;ADDRESS OF UNIT CORRESPONDING TO OLDEST CMD
                            ;  ZERO IF THERE ARE NO OUTSTANDING CMDS.
S.CCB:'L'                   ;MIXED MASSBUS CHANNEL CONTROL BLOCK
S.MPR:'L'   .BLKW 6         ;11/70 EXTENDED MEMORY UNIBUS DEVICE C-BLOCK
            .BLKW 1         ;BUFFER WORD
S.UMHD:'L'  .BLKW 2         ;LIST HEAD FOR UMR ASSIGNMENT BLOCK(S)
S.UMCT:'L'  .BLKW 1         ;COUNT OF AVAILABLE UMR ASSIGNMENT BLOCK(S)

            .IFF

            .PSECT
```

SCBDF$ (Cont.)

```
;+
; STATUS CONTROL BLOCK PRIORITY BYTE CONDITION CODE STATUS BIT
; DEFINITIONS
;-

SP.EIP='B' 1                ;ERROR IN PROGRESS (1=YES)
SP.ENB='B' 2                ;ERROR LOGGING ENABLED (0=YES)
SP.LOG='B' 4                ;ERROR LOGGING AVAILABLE (1=YES)
SPARE=10                    ;SPARE BIT

;+
; MAPPING ASSIGNMENT BLOCK (FOR UNIBUS MAPPING REGISTER ASSIGNMENT)
;-

            .ASECT
.=0
M.LNK:'L'   .BLKW 1         ;LINK WORD
M.UMRA:'L'  .BLKW 1         ;ADDRESS OF FIRST ASSIGNED UMR
M.UMRN:'L'  .BLKW 1         ;NUMBER OF UMR'S ASSIGNED * 4
M.UMVL:'L'  .BLKW 1         ;LOW 16 BITS MAPPED BY 1ST ASSIGNED UMR
M.UMVH:'L'  .BLKB 1         ;HIGH 2 BITS MAPPED IN BITS 4 AND 5
M.BFVH:'L'  .BLKB 1         ;HIGH 6 BITS OF PHYSICAL BUFFER ADDRESS
M.BFVL:'L'  .BLKW 1         ;LOW 16 BITS OF PHYSICAL BUFFER ADDRESS
M.LGTH='B'.                 ;LENGTH OF MAPPING ASSIGNMENT BLOCK

            .ENDC

            .PSECT

            .MACRO SCBDF$,X,Y,Z
            .ENDM
            .ENDM
```

```
        .MACRO TCBDF$,L,B

;+
; TASK CONTROL BLOCK OFFSET AND STATUS DEFINITIONS
;
; TASK CONTROL BLOCK
;-

;
; SEVERAL PIECES OF PRIVILEGED CODE EXIST THAT CREATE TCBS FROM
; OTHER TCBS. SINCE THESE PIECES OF CODE ARE GENERALLY OPTIMIZED
; FOR SPEED AND DO NOT USE THE SYMBOLIC OFFSETS PROVIDED BELOW,
; ANY CHANGE IN THE TCB MUST ALSO BE MADE TO EACH OF THESE PIECES
; OF CODE. THE KNOWN LIST OF SUCH PIECES OF CODE IS AS FOLLOWS:
;
; LIBRARY        MODULE              COMMENT
; -------        ------              -------
;
; RSX11M         DRSPW               TCB CREATED FOR RPOI$ DIRECTIVE
; MCR            MCROV,MCRDIS        MULTIUSER TASK DISPATCHING
;
        .ASECT
.=0
T.LNK:'L'   .BLKW 1          ;UTILITY LINK WORD
T.PRI:'L'   .BLKB 1          ;TASK PRIORITY
T.IOC:'L'   .BLKB 1          ;I/O PENDING COUNT
T.CPCB:'L'  .BLKW 1          ;POINTER TO CHECKPOINT PCB
T.NAM:'L'   .BLKW 2          ;TASK NAME IN RAD50
T.RCVL:'L'  .BLKW 2          ;RECEIVE QUEUE LISTHEAD
T.ASTL:'L'  .BLKW 2          ;AST QUEUE LISTHEAD
T.EFLG:'L'  .BLKW 2          ;TASK LOCAL EVENT FLAGS 1-32
T.UCB:'L'   .BLKW 1          ;UCB ADDRESS FOR PSEUDO DEVICE 'TI'
T.TCBL:'L'  .BLKW 1          ;TASK LIST THREAD WORD
T.STAT:'L'  .BLKW 1          ;FIRST STATUS WORD (BLOCKING BITS)
T.ST2:'L'   .BLKW 1          ;SECOND STATUS WORD (STATE BITS)
T.ST3:'L'   .BLKW 1          ;THIRD STATUS WORD (ATTRIBUTE BITS)
T.DPRI:'L'  .BLKB 1          ;TASK'S DEFAULT PRIORITY
T.LBN:'L'   .BLKB 3          ;LBN OF TASK LOAD IMAGE
T.LDV:'L'   .BLKW 1          ;UCB ADDRESS OF LOAD DEVICE
T.PCB:'L'   .BLKW 1          ;PCB ADDRESS OF TASK PARTITION
T.MXSZ:'L'  .BLKW 1          ;MAXIMUM SIZE OF TASK IMAGE (MAPPED ONLY)
T.ACTL:'L'  .BLKW 1          ;ADDRESS OF NEXT TASK IN ACTIVE LIST
T.SAST:'L'  .BLKW 1          ;SPECIFIED AST LISTHEAD
            .BLKB 1          ;RESERVED BYTE (CURRENTLY MUST BE 0)
T.TIO:'L'   .BLKB 1          ;BUFFERED I/O COUNT
T.TKSZ:'L'  .BLKW 1          ;TASK SIZE (FROM L$BLDZ IN LABEL BLK) IN:
                             ;    UNMAPPED SYSTEMS - BYTES
                             ;    MAPPED SYSTEMS   - 32 WORD BLOCKS
                             ;TASK SIZE (FROM L$BMXZ IN LABEL BLK)
                             ;FOR RSX11S SYSTEMS ONLY
                             ;    MAPPED SYSTEMS   - 32 WORD BLOCKS
                             ;    UNMAPPED SYSTEMS - BYTES

$$$=.                        ;MARK START OF PLAS AREA

T.ATT:'L'   .BLKW 2          ;ATTACHMENT DESCRIPTOR LISTHEAD
T.OFF:'L'   .BLKW 1          ;OFFSET TO TASK IMAGE IN PARTITION
                             ;IF A$$HDR IS DEFINED, THIS WORD ALSO
                             ;INCLUDES THE LENGTH OF THE ALTERNATE
                             ;HEADER REFRESH AREA STORED IN T.HDLN
            .BLKB 1          ;RESERVED
```

TCBDF$ (Cont.)

```
T.SRCT:'L' .BLKB 1          ;SREF WITH EFN COUNT IN ALL RECEIVE QUEUES
T.RRFL:'L' .BLKW 2          ;RECEIVE BY REFERENCE LISTHEAD

           .IF NDF P$$LAS

.=$$$                       ;POINT TO START OF PLAS AREA

           .ENDC            ;P$$LAS

           .IF NB SYSDEF

$$$=.                       ;MARK START OF PARENT OFFSPRING TASKING AREA

T.OCBH:'L' .BLKW 2          ;OFFSPRING CONTROL BLOCK LISTHEAD
T.RDCT:'L' .BLKW 1          ;OUTSTANDING OFFSPRING COUNT

           .IF NDF P$$OFF

.=$$$                       ;POINT TO START OF PARENT OFFSPRING AREA

           .ENDC            ;P$$OFF

$$$=.                       ;MARK START OF EVENT FLAG MASK AREA

T.EFLM:'L' .BLKW 2          ;EVENT FLAG MASK WORD
                            ;EVENT FLAG MASK ADDRESS

           .IF NDF S$$TOP&T$$BUF

.=$$$                       ;POINT TO START OF EVENT FLAG MASK AREA

           .ENDC            ;S$$TOP&T$$BUF


$$$=.

T.HDLN:'L' .BLKB 1          ;TASK HEADER LENGTH IN 32-WORD BLOCKS

           .IF NDF A$$HDR

.=$$$                       ;NOT SUPPORTED IF NDF

           .ENDC            ;A$$HDR

$$$=.

T.GGF:'L'  .BLKB 1          ;GROUP GLOBAL USE COUNT FOR TASK

           .IF NDF R$$SND&G$$EFN!A$$CLI&G$$EFN

.=$$$

           .ENDC

           .EVEN


T.LGTH='B' .                  ;LENGTH OF TASK CONTROL BLOCK
T.EXT='B'0                    ;LENGTH OF TCB EXTENSION

           .IFF
```

TCBDF$ (Cont.)

```
;+
; TASK STATUS DEFINITIONS
;
; FIRST STATUS WORD (BLOCKING BITS)
;-

TS.EXE='B' 100000      ;TASK NOT IN EXECUTION (1=YES)
TS.RDN='B'  40000      ;I/O RUN DOWN IN PROGRESS (1=YES)
TS.MSG='B'  20000      ;ABORT MESSAGE BEING OUTPUT (1=YES)
TS.NRP='B'  10000      ;TASK MAPPED TO NONRESIDENT PARTITION (1=YES)
TS.RUN='B'   4000      ;TASK IS RUNNING ON ANOTHER PROCESSOR (1=YES)
TS.HLD='B'   2000      ;TASK HALF-LOADED BY TASK LOADER
TS.STP='B'   1000      ;TASK EXTERNALLY BLOCKED VIA CLI COMMAND
TS.OUT='B'    400      ;TASK IS OUT OF MEMORY (1=YES)
TS.CKP='B'    200      ;TASK IS BEING CHECKPOINTED (1=YES)
TS.CKR='B'    100      ;TASK CHECKPOINT REQUESTED (1=YES)

;+
; TASK BLOCKING STATUS MASK
;-

TS.BLK='B'TS.CKP!TS.CKR!TS.EXE!TS.MSG!TS.NRP!TS.OUT!TS.RDN!TS.STP

;+
; SECOND STATUS WORD (STATE BITS)
;-

T2.AST='B' 100000           ;AST IN PROGRESS (1=YES)
T2.DST='B'  40000           ;AST RECOGNITION DISABLED (1=YES)
T2.CHK='B'  20000           ;TASK NOT CHECKPOINTABLE (1=YES)
T2.CKD='B'  10000           ;CHECKPOINTING DISABLED (1=YES)
T2.SEF='B'   4000           ;TASK STOPPED FOR EVENT FLAGS (1=YES)
T2.FXD='B'   2000           ;TASK FIXED IN MEMORY (1=YES)
T2.REX='B'   1000           ;ABORT AST EFFECTED OR IN PROGRESS (1=YES)
T2.CAF='B'    400           ;DYN CHECKPOINT SPACE ALLOCATION FAILURE
T2.HLT='B'    200           ;TASK IS BEING HALTED (1=YES)
T2.ABO='B'    100           ;TASK MARKED FOR ABORT (1=YES)
T2.STP='B'     40           ;SAVED T2.STP ON AST IN PROGRESS
T2.STP='B'     20           ;TASK STOPPED (1=YES)
T2.SPN='B'     10           ;SAVED T2.SPN ON AST IN PROGRESS
T2.SPN='B'      4           ;TASK SUSPENDED (1=YES)
T2.WFR='B'      2           ;SAVED T2.WFR ON AST IN PROGRESS
T2.WFR='B'      1           ;TASK IN WAITFOR STATE (1=YES)

;+
; THIRD STATUS WORD (ATTRIBUTE BITS)
;-

T3.ACP='B' 100000        ;ANCILLARY CONTROL PROCESSOR (1=YES)
T3.PMD='B'  40000        ;DUMP TASK ON SYNCHRONOUS ABORT (0=YES)
T3.REM='B'  20000        ;REMOVE TASK ON EXIT (1=YES)
T3.PRV='B'  10000        ;TASK IS PRIVILEGED (1=YES)
T3.MCR='B'   4000        ;TASK REQUESTED AS EXTERNAL MCR FUNCTION
                         ;(1=YES)
T3.SLV='B'   2000        ;TASK IS A SLAVE TASK (1=YES)
T3.CLI='B'   1000        ;TASK IS A COMMAND LINE INTERPRETER (1=YES)
T3.RST='B'    400        ;TASK IS RESTRICTED (1=YES)
T3.NSD='B'    200        ;TASK DOES NOT ALLOW SEND DATA
```

TCBDF$ (Cont.)

```
T3.CAL='B'    100        ;TASK HAS CHECKPOINT SPACE IN TASK IMAGE
T3.ROV='B'     40        ;TASK HAS RESIDENT OVERLAYS
T3.NET='B'     20        ;NETWORK PROTOCOL LEVEL
T3.GFL='B'     10        ;TASK HAS ITS GRP GBL EVENT FLAGS LOCKED
;     ='B'      4        ;RESERVED FOR FUTURE USE
T3.SWS='B'      2        ;RESERVED FOR USE BY SOFTWARE SERVICES
;     ='B'      1        ;RESERVED FOR FUTURE USE

        .ENDC


        .PSECT
        .MACRO TCBDF$ X,Y,Z
        .ENDM
        .ENDM
```

UCBDF$

```
        .MACRO      UCBDF$,L,B,TTDEF,SYSDF

;+
; UNIT CONTROL BLOCK
;
; THE UNIT CONTROL BLOCK (UCB) DEFINES THE STATUS OF AN INDIVIDUAL
; DEVICE UNIT AND IS THE CONTROL BLOCK THAT IS POINTED TO BY THE
; FIRST WORD OF AN ASSIGNED LUN. THERE IS ONE UCB FOR EACH DEVICE
; UNIT OF EACH DCB. THE UCB'S ASSOCIATED WITH A PARTICULAR DCB ARE
; CONTIGUOUS IN MEMORY AND ARE POINTED TO BY THE DCB. UCB'S ARE
; VARIABLE LENGTH BETWEEN DCB'S BUT ARE OF THE SAME LENGTH FOR A
; SPECIFIC DCB. TO FINISH THE TELETYPE EXAMPLE ABOVE, EACH UNIT
; ON BOTH INTERFACES WOULD HAVE A UCB.
;-

            .ASECT

            .IF NB SYSDF

            .IF DF E$$DVC

            .IF DF M$$MUP       ;IS U.OWN THERE?
.=177766
        .IFF
.=177770
        .ENDC

U.IOC:'L'   .BLKW 2             ;I/O COUNT SINCE MOUNT (ERROR LOG DEVS
                                ;ONLY)
U.ERSL:'L'  .BLKB 1             ;SOFT ERROR LIMIT
U.ERHL:'L'  .BLKB 1             ;HARD ERROR LIMIT
U.ERSC:'L'  .BLKB 1             ;SOFT ERROR COUNT
U.ERHC:'L'  .BLKB 1             ;HARD ERROR COUNT

            .ENDC

            .ENDC


.=177772
U.MUP:'L'                       ;MULTIUSER PROTECTION FLAG WORD
U.CLI:'L'   .BLKW 1             ;TCB OF COMMAND LINE INTERPRETER
U.LUIC:'L'  .BLKW 1             ;LOGIN UIC - MULTI USER SYSTEMS ONLY
U.OWN:'L'   .BLKW 1             ;OWNING TERMINAL - MULTI USER SYSTEMS
                                ;ONLY
U.DCB:'L'   .BLKW 1             ;BACK POINTER TO DCB
U.RED:'L'   .BLKW 1             ;POINTER TO REDIRECT UNIT UCB
U.CTL:'L'   .BLKB 1             ;CONTROL PROCESSING FLAGS
U.STS:'L'   .BLKB 1             ;UNIT STATUS
U.UNIT:'L'  .BLKB 1             ;PHYSICAL UNIT NUMBER
U.ST2:'L'   .BLKB 1             ;UNIT STATUS EXTENSION
U.CW1:'L'   .BLKW 1             ;FIRST DEVICE CHARACTERISTICS WORD
U.CW2:'L'   .BLKW 1             ;SECOND DEVICE CHARACTERISTICS WORD
U.CW3:'L'   .BLKW 1             ;THIRD DEVICE CHARACTERISTICS WORD
U.CW4:'L '  .BLKW 1             ;FOURTH DEVICE CHARACTERISTICS WORD
U.SCB:'L'   .BLKW 1             ;POINTER TO SCB
U.ATT:'L'   .BLKW 1             ;TCB ADDRESS OF ATTACHED TASK
U.BUF:'L'   .BLKW 1             ;RELOCATION BIAS OF CURRENT I/O REQUEST
            .BLKW 1             ;BUFFER ADDRESS OF CURRENT I/O REQUEST
```

UCBDF$ (Cont.)

```
U.CNT:'L'   .BLKW 1             ;BYTE COUNT OF CURRENT I/O REQUEST
U.ACP='B'   U.CNT+2             ;ADDRESS OF TCB OF MOUNTED ACP
U.VCB='B'   U.CNT+4             ;ADDRESS OF VOLUME CONTROL BLOCK
U.CBF='B'   U.CNT+2             ;CONTROL BUFFER RELOCATION AND ADDRESS
U.KCSR='B'  U.CNT+2             ;CSR ADDRESS OF KMC-11
U.KCS6='B'  U.KCSR+2            ;CSR+6 OF KMC-11

;
;       MAGTAPE DRIVER DEFINITIONS
;
U.SPC='B'   U.CNT+6             ;SPACING COUNT
U.SUB='B'   U.CNT+6             ;SUBCONTROLLER, PHYSICAL UNIT #.
U.FNUM='B'  U.CNT+10            ;FORMATTER NUMBER
U.FCDE='B'  U.CNT+12            ;FUNCTION CODE AND INDEX

;
;               MSCP/TMSCP DRIVER UCB OFFSETS
;
U.UTIL='B'  U.VCB+2             ;UNIT STATE WORD
;
; DEFINITIONS FOR U.UTIL BITS
;
UU.SER='B'     1                ;SERIAL MODE
UU.RCT='B'     2                ;(DUDRV)RCT IN PROGRESS
UU.AVN='B'     4                ;UNIT IS WAITING FOR OTHER UNITS TO SPIN
                                ;DOWN
UU.GUS='B'    10                ;UNIT MUST HAVE A GUS COMMAND ISSUED
UU.ONL='B'    20                ;UNIT MUST HAVE A ONL COMMAND ISSUED
UU.SPC='B'    40                ;SPECIAL ONLINE TRANSITION
UU.ATN='B'   100                ;UNIT HAS SENT ATTENTION MESSAGE
UU.RDY='B'   200                ;UNIT IS READY
UU.ABO='B'   400                ;IF SET, XXCAN SET UU.SER FLAG FOR UNIT
UU.SIO='B'  1000                ;THIS UNIT CAN STALL I/O
UU.IOS='B'  2000                ;THIS UNIT HAS I/O STALLED
U.MEDI='B'  U.VCB+4             ;MEDIA IDENTIFIER (2 WORDS)
;
; ALL THE FOLLOWING MSCP FIELDS APPLY ONLY TO DISK
;
U.BPKT='B'  U.VCB+10            ;UNIT BAD BLOCK PACKET WAITING LIST
;
; CHARACTERISTICS OBTAINED FROM "GET UNIT STATUS" END PACKETS
;
U.MLUN='B'  U.VCB+14            ;MULTI-UNIT CODE
U.UNFL='B'  U.VCB+16            ;UNIT FLAGS
U.UNTI='B'  U.VCB+24            ;UNIT IDENTIFIER
U.2MED='B'  U.VCB+34            ;ORIGINAL COPY OF MEDIA IDENTIFIER
U.SHUN='B'  U.VCB+40            ;SHADOW UNIT
U.SHST='B'  U.VCB+42            ;SHADOW UNIT STATUS
U.TRCK='B'  U.VCB+44            ;UNIT TRACK SIZE
U.GRP='B'   U.VCB+46            ;UNIT GROUP SIZE
U.CYL='B'   U.VCB+50            ;UNIT CYLINDER SIZE
U.USVR='B'  U.VCB+52            ;UNIT SOFTWARE VERSION
U.UHVR='B'  U.VCB+53            ;UNIT HARDWARE VERSION
U.RCTS='B'  U.VCB+54            ;UNIT RCT TABLE SIZE
U.RBNS='B'  U.VCB+56            ;UNIT RBN 'S / TRACK
U.RCTC='B'  U.VCB+57            ;UNIT RCT COPIES
```

UCBDF$ (Cont.)

```
;
; CHARACTERISTICS OBTAINED FROM "ONLINE" OR "SET UNIT CHARACTERISTICS"
; END PACKETS
;
U.UNSZ='B'U.VCB+60              ;UNIT SIZE
U.VSER='B'U.VCB+64              ;VOLUME SERIAL NUMBER
;
; TERMINAL DRIVER DEFINITIONS
;
.=U.BUF
U.TUX:'L'   .BLKW 1             ;POINTER TO UCB EXTENSION (UCBX)
U.TSTA:'L'  .BLKW 4             ;STATUS QUADRUPLE-WORD
U.TFRQ:'L'  .BLKW 1             ;FORK REQUEST WORD
U.TFLK:'L'  .BLKW 1             ;FORK LIST LINK WORD
U.TCHP:'L'  .BLKB 1             ;CURRENT HORIZONTAL POSITION
U.TCVP:'L'  .BLKB 1             ;CURRENT VERTICAL POSITION
U.UIC:'L'   .BLKW 1             ;TERMINAL UIC
U.TTYP:'L'  .BLKB 1             ;TERMINAL TYPE
U.TMTI:'L'  .BLKB 1             ;MODEM TIMER
U.TTAB:'L'  .BLKW 1             ;IF 0: U.TTAB+1 IS SINGLE-CHARACTER
                                ; TYPE-AHEAD BUFFER, CURRENTLY EMPTY
                                ;IF ODD: U.TTAB+1 IS SINGLE-CHARACTER
                                ; TYPE-AHEAD BUFFER AND HOLDS A
                                ; CHARACTER
                                ;IF NON-0 AND EVEN: POINTER TO
                                ; MULTI-CHARACTER TYPE-AHEAD BUFFER
U.CTYP:'L'  .BLKB 1             ;CONTROLLER TYPE
U.TLPP:'L'  .BLKB 1             ;LINES PER PAGE
U.TST5:'L'  .BLKW 1             ;ADDITIONAL STATUS BITS
U.TST6:'L'  .BLKW 1             ;EXTENDED I/O STATUS BITS
U.TIXL:'L'  .BLKW 1             ;I/O PACKET EXTENSION LISTHEAD
U.ACB:'L'   .BLKW 1             ;ANCILLARY CONTROL DRIVER BLOCK ADDR
U.AFLG:'L'  .BLKW 1             ;ANCILLARY CONTROL DRIVER FLAGS WORD
U.ADMA:'L'  .BLKW 1             ;ANCILLARY CONTROL DRIVER DMA BUFFER
;
; CONSOLE DRIVER DEFINITIONS
;

.=U.BUF+2
U.CTCB:'L'  .BLKW 1             ;ADDRESS OF CONSOLE LOGGER TCB
U.COTQ:'L'  .BLKW 2             ;I/O PACKET LIST QUEUE
U.RED2:'L'  .BLKW 1             ;REDIRECT UCB ADDRESS

;
; DEFINE BITS IN STATUS WORD 1 (U.TSTA)
;
;
; I N P U T   S T A T U S
;

S1.RST='B'       1              ;READ WITH SPECIAL TERMINATORS IN
                                ;PROGRESS
S1.ESC='B'       2              ;ESCAPE SEQUENCE IN PROGRESS
S1.RSP='B'       4              ;READ WITH SPECIAL PROCESSING
S1.PTH='B'      10              ;PASS THRU IS CURRENTLY ACTIVE
S1.RNE='B'      20              ;ECHO SUPPRESSED
S1.TSY='B'      40              ;TERMINAL OUTPUT SYNC IS CURRENTLY
                                ;ENABLED
```

UCBDF$ (Cont.)

```
S1.OBY='B'    100          ;OUTPUT BUSY
S1.IBY='B'    200          ;INPUT BUSY
S1.DPR='B'    400          ;DEFER PROCESSING OF CHAR. IN U.TECB
S1.DEC='B'   1000          ;DEFER ECHO OF CHAR. IN U.TECB
S1.IBF='B'   2000          ;BUFFERED INPUT IN PROGRESS
S1.DSI='B'   4000          ;INPUT PROCESSING DISABLED
S1.RES='B'  10000          ;ESC. SEQ PROCESSING IS ENABLED FOR THE
                           ;CURRENT READ
S1.RNF='B'  20000          ;READ NO FILTER IS ACTIVE (EDIT CHARS.
                           ;ARE DISPLAYED)
S1.TNE='B'  40000          ;TERMINATOR NO ECHO
S1.USI='B' 100000          ;UNSOLICITED INPUT IN PROGRESS

; DEFINE BITS IN STATUS WORD 2 (U.TSTA+2)
;
;    O U T P U T   S T A T U S
;
S2.RCU='B'      1          ;RESTORE CURSOR (MUST = TF.RCU)
S2.WRA='B'      6          ;CONTEXT FOR WRAP-AROUND
S2.WRB='B'      2          ;LOW BIT IN S2.WRA BIT PATTERN
S2.WAL='B'     10          ;WRITE PASS ALL (MUST = TF.WAL)
S2.BRQ='B'     20          ;BREAK-THROUGH-WRITE REQUEST IN QUEUE
S2.SRQ='B'     40          ;SPECIAL REQUEST IN QUEUE
                           ;(IO.ATT, IO.DET, SF.SMC)
S2.ORQ='B'    100          ;OUTPUT REQUEST IN QUEUE (MUST = S1.OBY)
S2.IRQ='B'    200          ;INPUT REQUEST IN QUEUE (MUST = S1.IBY)
S2.FLF='B'    400          ;FORCE LINEFEED BEFORE NEXT ECHO
S2.ELF='B'   1000          ;EAT A LINEFEED (IGNORE A LEADING LF ON
                           ;OUTPUT)
S2.CR='B'    2000          ;TRAILING CR REQUIRED ON OUTPUT
S2.OBF='B'   4000          ;BUFFERED OUTPUT IN PROGRESS
S2.PCU='B'  10000          ;POSITION CURSOR BEFORE WRITE
S2.BEL='B'  20000          ;BELL PENDING
S2.CTO='B'  40000          ;OUTPUT STOPPED BY CTRL-O 266.
S2.CTS='B' 100000          ;OUTPUT STOPPED BY CTRL-S

; DEFINE BITS IN STATUS WORD 3 (U.TSTA+4)
;
;   TERMINAL OPERATION CHARACTERISTICS
;
S3.ACR='B'      1          ;WRAP-AROUND (AUTOMATIC CR-LF) REQUIRED
S3.TAB='B'      2          ;TYPE-AHEAD BUFFER ALLOCATION REQUESTED
S3.CTC='B'      4          ;TERMINAL WANTS CLI TO HAVE ^C NOTIFICATION
S3.RAL='B'     10          ;TERMINAL IS IN READ-PASS-ALL MODE
S3.NEC='B'     20          ;NO ECHO
S3.TSY='B'     40          ;TERMINAL SYNC
S3.8BC='B'    100          ;PASS 8 BITS ON INPUT
S3.FDX='B'    200          ;LINE IS IN FULL DUPLEX MODE
S3.MHE='B'    400          ;NOTIFY ATTACHED TASK OF MODEM HANG-UP
S3.ICE='B'   1000          ;INPUT COUNT STATE ENABLED
S3.TME='B'   2000          ;TERMINAL MANAGEMENT MODE ENABLED
S3.PTH='B'   4000          ;PASS THROUGH REQUESTED
S3.RES='B'  10000          ;TASK WANTS ESCAPE SEQUENCES
S3.PPT='B'  20000          ;TERMINAL HAS PRINTER PORT
S3.RUB='B'  40000          ;RUBOUT SEQUENCE IN PROGRESS (NON-SCOPE)
```

UCBDF$ (Cont.)

```
;
; DEFINE BITS IN STATUS WORD 4 (U.TSTA+6)
;
;
; TERMINAL ATTRIBUTE CHARACTERISTICS
;
S4.HFL='B'       7          ;HORIZONTAL FILL REQUIREMENT
S4.VFL='B'      10          ;VERTICAL FILL REQUIREMENT
S4.HFF='B'      20          ;HARDWARE FORM-FEED PRESENT
S4.HHT='B'      40          ;HARDWARE HORIZONTAL TAB PRESENT
S4.DLO='B'     100          ;DIAL-OUT LINE (IMPLIES U2.RMT)
S4.HSY='B'     200          ;HOST/TERMINAL SYNCHRONIZATION ENABLED
                            ;(1=YES)
S4.ANI='B'     400          ;ANSI CRT TERMINAL
S4.AVO='B'    1000          ;VT100-FAMILY TERMINAL DISPLAY
S4.BLK='B'    2000          ;BLOCK MODE TERMINAL
S4.DEC='B'    4000          ;DIGITAL CRT TERMINAL
S4.EDT='B'   10000          ;TERMINAL HAS LOCAL EDITING FUNCTIONS
S4.RGS='B'   20000          ;TERMINAL SUPPORTS REGIS GRAPHICS
S4.SFC='B'   40000          ;TERMINAL SUPPORTS SOFT CHARACTERS (DRCS)
S4.ABD='B' 100000           ;AUTO-BAUD SPEED DETECTION ENABLED
;
;
; DEFINE BITS IN STATUS WORD U.TST5
;
;  ADDITIONAL STATUS CHARACTERISTICS
;
;
S5.SW1='B'       1          ;FIRST TERMINAL MANAGEMENT SWITCH
                            ;CHARACTER HAS BEEN SEEN
S5.TMM='B'       2          ;TERMINAL IN TERMINAL MANAGEMENT MODE
S5.XOF='B'       4          ;SEND AN XOFF AT FIRST OPPORTUNITY
S5.XON='B'      10          ;SEND AN XON AT FIRST OPPORTUNITY
S5.HPC='B'      14          ;OUTPUT OF HIGH PRIORITY CHARACTERS
                            ;REQUESTED
S5.HPO='B'      20          ;HIGH PRIORITY OUTPUT IN PROGRESS
S5.OXF='B'      40          ;XOFF HAS BEEN OUTPUT
S5.ITI='B'     100          ;IMMEDIATE TIMEOUT ON INPUT
;
S5.RPO='B'    2000          ;READ W/PROMPT OUTPUT IN PROGRESS
S5.VER='B'   10000          ;LAST CHAR. IN TYPE-AHEAD BUFFER
                            ;HAS PARITY ERROR
S5.BCC='B'   20000          ;LAST CHAR. IN TYPE-AHEAD BUFFER
                            ;HAS FRAMING ERROR
S5.DAO='B'   40000          ;LAST CHAR. IN TYPE-AHEAD BUFFER
                            ;HAS DATA OVERRUN ERROR
                            ;NOTE - THE 3 BITS ABOVE MUST CORRESPOND
                            ;TO THE RESPECTIVE ERROR FLAGS IN THE
                            ;HARDWARE RECEIVE BUFFER
S5.ABP='B' 100000           ;AUTO-BAUD SPEED DETECTION IN PROGRESS
;
;
; DEFINE BITS IN EXTENDED I/O STATUS WORD U.TST6
;
;
S6.EIO='B'     400          ;READ WAS AN EXTENDED I/O
S6.RLU='B'    1000          ;READ WITH LOWER CASE TO UPPER CASE
                            ;CONVERSION
S6.RDI='B' 100000           ;READ WITH DEFAULT INPUT
```

UCBDF$ (Cont.)

```
;

;

          .PSECT

;+
; DEVICE TABLE STATUS DEFINITIONS
;
; DEVICE CHARACTERISTICS WORD 1 (U.CW1) DEVICE TYPE DEFINITION BITS.
;-

DV.REC='B'      1           ;RECORD ORIENTED DEVICE (1=YES)
DV.CCL='B'      2           ;CARRIAGE CONTROL DEVICE (1=YES)
DV.TTY='B'      4           ;TERMINAL DEVICE (1=YES)
DV.DIR='B'     10           ;FILE STRUCTURED DEVICE (1=YES)
DV.SDI='B'     20           ;SINGLE DIRECTORY DEVICE (1=YES)
DV.SQD='B'     40           ;SEQUENTIAL DEVICE (1=YES)
DV.MSD='B'    100           ;MASS STORAGE DEVICE (1=YES)
DV.UMD='B'    200           ;USER MODE DIAGNOSTICS SUPPORTED (1=YES)
DV.MBC='B'    400           ;DEVICE IS ON MASSBUS CONTROLLER (1=YES)
DV.EXT='B'    400           ;DEVICE ON EXTENDED ADDRESSING CONTROLLER
DV.SWL='B'   1000           ;UNIT SOFTWARE WRITE LOCKED (1=YES)
DV.ISP='B'   2000           ;INPUT SPOOLED DEVICE (1=YES)
DV.OSP='B'   4000           ;OUTPUT SPOOLED DEVICE (1=YES)
DV.PSE='B'  10000           ;PSEUDO DEVICE (1=YES)
DV.COM='B'  20000           ;DEVICE IS MOUNTABLE AS COM CHANNEL
                            ;(1=YES)
DV.F11='B'  40000           ;DEVICE IS MOUNTABLE AS F11 DEVICE (1=YES)
DV.MNT='B' 100000           ;DEVICE IS MOUNTABLE (1=YES)

;+
; TERMINAL DEPENDENT CHARACTERISTICS WORD 2 (U.CW2) BIT DEFINITIONS
;-

U2.DH1='B' 100000           ;UNIT IS A MULTIPLEXER (1=YES)
U2.DJ1='B'  40000           ;UNIT IS A DJ11 (1=YES)
U2.RMT='B'  20000           ;UNIT IS REMOTE (1=YES)
U2.HFF='B'  10000           ;UNIT HANDLES HARDWARE FORM FEEDS (1=YES)
U2.L8S='B'  10000           ;OLD NAME FOR U2.HFF
U2.NEC='B'   4000           ;DON'T ECHO SOLICITED INPUT (1=YES)
U2.CRT='B'   2000           ;UNIT IS A CRT (1=YES)
U2.ESC='B'   1000           ;UNIT GENERATES ESCAPE SEQUENCES (1=YES)
U2.LOG='B'    400           ;USER LOGGED ON TERMINAL (0=YES)
U2.SLV='B'    200           ;UNIT IS A SLAVE TERMINAL (1=YES)
U2.DZ1='B'    100           ;UNIT IS A DZ11 (1=YES)
U2.HLD='B'     40           ;TERMINAL IS IN HOLD SCREEN MODE (1=YES)
U2.AT.='B'     20           ;MCR COMMAND AT. BEING PROCESSED (1=YES)
U2.PRV='B'     10           ;UNIT IS A PRIVILEGED TERMINAL (1=YES)
U2.L3S='B'      4           ;UNIT IS A LA30S TERMINAL (1=YES)
U2.SCS='B'      4           ;SCS-11 COMMAND TERMINAL (1=YES)
U2.VT5='B'      2           ;UNIT IS A VT05B TERMINAL (1=YES)
U2.LWC='B'      1           ;LOWER CASE TO UPPER CASE CONVERSION
                            ;(0=YES)

;+
; BIT DEFINITIONS FOR U.MUP (SYSTEMS WITH ALTERNATE CLI SUPPORT ONLY)
;-
```

UCBDF$ (Cont.)

```
UM.OVR='B'   1              ;OVERRIDE CLI INDICATOR
UM.CLI='B'  36              ;CLI INDICATOR BITS
UM.DSB='B' 200              ;TERMINAL DISABLED SINCE CLI ELIMINATED
UM.NBR='B' 400              ;NO BROADCAST

;+
; RH11-RS03/RS04 CHARACTERISTICS WORD 2 (U.CW2) BIT DEFINITIONS
;-

U2.R04='B' 100000           ;UNIT IS A RS04 (1=YES)

;+
; RH11-TU16 CHARACTERISTICS WORD 2 (U.CW2) BIT DEFINITIONS
;-

U2.7CH='B' 10000            ;UNIT IS A 7 CHANNEL DRIVE (1=YES)

;+
; TERMINAL DEPENDENT CHARACTERISTICS WORD 3 (U.CW3) BIT DEFINITIONS
;-

U3.UPC='B'  20000           ;UPCASE OUTPUT FLAG
U3.PAR='B'  40000           ;PARITY GENERATION AND CHECKING
U3.OPA='B' 100000           ;PARITY SENSE (1=ODD PARITY)

;+
; TERMINAL DEPENDENT CHARACTERISTICS WORD 4 (U.CW4) BIT DEFINITIONS
;-

U4.CR='B' 100               ;LOOK FOR CARRIAGE RETURN

;+
; UNIT CONTROL PROCESSING FLAG DEFINITIONS
;-

UC.ALG='B' 200              ;BYTE ALIGNMENT ALLOWED (1=NO)
UC.NPR='B' 100              ;DEVICE IS AN NPR DEVICE (1=YES)
UC.QUE='B'  40              ;CALL DRIVER BEFORE QUEUING (1=YES)
UC.PWF='B'  20              ;CALL DRIVER AT POWERFAIL ALWAYS (1=YES)
UC.ATT='B'  10              ;CALL DRIVER ON ATTACH/DETACH (1=YES)
UC.KIL='B'   4              ;CALL DRIVER AT I/O KILL ALWAYS (1=YES)
UC.LGH='B'   3              ;TRANSFER LENGTH MASK BITS

;+
; UNIT STATUS BIT DEFINTIONS
;-

US.BSY='B' 200              ;UNIT IS BUSY (1=YES)
US.MNT='B' 100              ;UNIT IS MOUNTED (0=YES)
US.FOR='B'  40              ;UNIT IS MOUNTED AS FOREIGN VOLUME (1=YES)
US.MDM='B'  20              ;UNIT IS MARKED FOR DISMOUNT (1=YES)
US.PWF='B'  10              ;POWERFAIL OCCURRED (1=YES)

;+
; CARD READER DEPENDENT UNIT STATUS BIT DEFINITIONS
;-

US.ABO='B' 1                ;UNIT IS MARKED FOR ABORT IF NOT READY
                            ; (1=YES)
US.MDE='B' 2                ;UNIT IS IN 029 TRANSLATION NODE (1=YES)
```

UCBDF$ (Cont.)

```
;+
; FILES-11 DEPENDENT UNIT STATUS BITS
;-

US.WCK='B' 10               ;WRITE CHECK ENABLED (1=YES)
US.SPU='B'  2               ;UNIT IS SPINNING UP (1=YES)
US.VV='B'   1               ; VOLUME VALID IS SET (1=YES)

;+
; KMC-11-LP DEPDENDENT UNIT STATUS BITS
;-

US.KPF='B'  1               ;KMC-11 POWERFAIL INTERLOCK

;+
; TERMINAL DEPENDENT UNIT STATUS BIT DEFINITIONS
;-

           .IF NB TTDEF

           .IF DF T$$CPW

US.CRW='B' 4                ;UNIT IS WAITING FOR CARRIER (1=YES)
US.DSB='B' 2                ;UNIT IS DISABLED (1=YES)
US.OIU='B' 1                ;OUTPUT INTERRUPT IS UNEXPECTED ON UNIT
                            ;(1=YES)

           .IFF             ;T$$CPW

US.DSB='B' 10               ;UNIT IS DISABLED (1=YES)
US.CRW='B'  4               ;UNIT IS WAITING FOR CARRIER (1=YES)
US.ECH='B'  2               ;UNIT HAS ECHO IN PROGRESS (1=YES)
US.OUT='B'  1               ;UNIT IS EXPECTING OUTPUT INTERRUPT
                            ;(1=YES)

           .ENDC

           .ENDC            ;TTDEF

;+
; LPS11 DEPENDENT UNIT STATUS BIT DEFINITIONS
;-

US.FRK='B' 2                ;FORK IN PROGRESS (1=YES)
US.SHR='B' 1                ;SHAREABLE FUNCTION IN PROGRESS (0='B'YES)

;+
; MAGTAPE DEPENDENT UNIT STATUS BITS
;-
US.LAB='B' 4                ; UNIT HAS LABELED TAPE ON IT (1=YES)
US.BSP='B' 2                ; INTERNAL BACKSPACE IN PROGRESS (1=YES)

;+
; UNIT STATUS EXTENSION (U.ST2) BIT DEFINITIONS
;-
```

UCBDF$ (Cont.)

```
US.OFL='B'  1               ;UNIT OFFLINE (1=YES)
US.RED='B'  2               ;UNIT REDIRECTABLE (0=YES)
US.PUB='B'  4               ;UNIT IS PUBLIC DEVICE (1=YES)
US.UMD='B' 10               ;UNIT ATTACHED FOR DIAGNOSTICS (1=YES)

;+
; MAG TAPE DENS SUPPORT IDENT IN CHAR WORD 3 (U.CW3) DEFENITION
;       ASSIGNMENTS PER NUMERICAL SEQUENCE 0 - 255.
;-

UD.UNS='B' 0                ; UNSUPPORTED
UD.200='B' 1                ; 200BPI, 7 TRACK
UD.556='B' 2                ; 556BPI, 7 TRACK
UD.800='B' 3                ; 800BPI, 7 OR 9 TRACK
UD.160='B' 4                ;1600BPI, 9 TRACK
UD.625='B' 5                ;6250BPI, 9 TRACK
UD.8K='B'  6                ;8K BPI - SERIAL, SERPENTINE RECORDING

        .MACRO UCBDF$,X,Y,Z,ZZ
        .ENDM
        .ENDM
```

APPENDIX C

RSX-11M-PLUS SYSTEM DATA STRUCTURES AND SYMBOLIC DEFINITIONS

This appendix describes the RSX-11M-PLUS system macros that supply symbolic offsets for data structures listed in Table C-1.

The data structures are defined by macros in the Executive macro library. To reference any of the data structure offsets from your code, include the macro name in an .MCALL directive and invoke the macro. For example:

```
.MCALL  DCBDF$
DCBDF$                  ;Define DBC offsets
```

NOTE

All physical offsets and bit definitions are subject to change in future releases of the operating system. Code that accesses system data structures should always use the symbolic offsets rather than the physical offsets.

The first two arguments, <:> and <=>, make all definitions global. If they are left blank, the definitions will be local.

All of these macros are in the Executive macro library LB:[1,1]EXEMC.MLB. All except F11DF$, ITBDF$, MTADF$, OLRDF$, and SHDDF$ are also in the Executive definition library LB:[1,1]EXELIB.OLB.

Table C-1
Summary of System Data Structure Macros

Macro Arguments	Data Structures
ABODF$ <:>,<=>	Task abort and termination notification message codes
ACNDF$ <:>,<=>	Accounting data structures (user account block, task account block, system account block)
CLKDF$ <:>,<=>	Clock queue control block
CTBDF$ <:>,<=>	Controller table

(Continued on next page)

Table C-1 (Cont.)
Summary of System Data Structure Macros

Macro Arguments	Data Structures
DCBDF$ <:>,<=>	Device Control Block
EPKDF$ <:>,<=>	Error message block
EVNDF$ <:>,<=>	Terminal Software Architecture (TSA) event packet definitions
F11DF$ <:>,<=>	FILES-11 data structures (Volume Control Block, mount list entry, File Control Block, file window block, locked block list node)
HDRDF$ <:>,<=>	Task header and window block
HWDDF$ <:>,<=>	Hardware register addresses and feature mask definitions
ITBDF$ <:>,<=>	Interrupt transfer block
KRBDF$ <:>,<=>	Controller request block
LCBDF$ <:>,<=>	Logical assignment control block
MTADF$ <:>,<=>	ANSI magtape data structures (volume set control block)
OLRDF$	On-line reconfiguration interface
PCBDF$ <:>,<=>	Partition Control Block and attachment descriptor
PKTDF$ <:>,<=>	I/O packet, AST control block, offspring control block, group global event flag control block, and CLI parser block
SCBDF$ <:>,<=>	Status Control Block and UMR assignment block
SHDDF$ <:>,<=>	Shadow recording linkage block
TCBDF$ <:>,<=>	Task Control Block
UCBDF$ <:>,<=>,TTDEF	Unit Control Block

ABODF$

```
        .MACRO  ABODF$,L,B

;+
; TASK ABORT CODES
;
; NOTE: S.COAD-S.CFLT ARE ALSO SST VECTOR OFFSETS
;-

S.CACT='B'-4.                   ;TASK STILL ACTIVE
S.CEXT='B'-2.                   ;TASK EXITED NORMALLY
S.COAD='B'0.                    ;ODD ADDRESS AND TRAPS TO 4
S.CSGF='B'2.                    ;SEGMENT FAULT
S.CBPT='B'4.                    ;BREAK POINT OR TRACE TRAP
S.CIOT='B'6.                    ;IOT INSTRUCTION
S.CILI='B'8.                    ;ILLEGAL OR RESERVED INSTRUCTION
S.CEMT='B'10.                   ;NON RSX EMT INSTRUCTION
S.CTRP='B'12.                   ;TRAP INSTRUCTION
S.CFLT='B'14.                   ;11/40 FLOATING POINT EXCEPTION
S.CSST='B'16.                   ;SST ABORT-BAD STACK
S.CAST='B'18.                   ;AST ABORT-BAD STACK
S.CABO='B'20.                   ;ABORT VIA DIRECTIVE
S.CLRF='B'22.                   ;TASK LOAD REQUEST FAILURE
S.CCRF='B'24.                   ;TASK CHECKPOINT READ FAILURE
S.IOMG='B'26.                   ;TASK EXIT WITH OUTSTANDING I/O
S.PRTY='B'28.                   ;TASK MEMORY PARITY ERROR
S.CPMD='B'30.                   ;TASK ABORTED WITH PMD REQUEST
S.CELV='B'32.                   ;TI: VIRTUAL TERMINAL WAS ELIMINATED
S.CINS='B'34.                   ;TASK INSTALLED IN 2 DIFFERENT SYSTEMS
S.CAFF='B'36.                   ;TASK ABORTED DUE TO BAD AFFINITY (REQUIRED
                                ;BUS RUNS ARE OFFLINE OR NOT PRESENT)
S.CCSM='B'38.                   ;BAD CSM PARAMETERS OR BAD STACK
S.COTL='B'40.                   ;TASK HAS RUN OVER ITS TIME LIMIT
S.CTKN='B'42.                   ;ABORT VIA DIRECTIVE WITH NO TKTN MESSAGE

;
; TERMINATION CODES FOR BOM$
;
; NOTE:
;
; THE NORMAL TKTN ERROR CODES SPAN -4 THROUGH 42. THE BOM CODES,
; ALTHOUGH DEFINED FOR THE TASK, ETC, AS 0 THROUGH N, ARE PASSED TO
; TKTN AS -127 THROUGH -127+N. AN UNRECOGNIZED CODE IS PASSED AS -128,
; UNKNOWN ERROR.
;

S.BUNK='B'-128.                 ;UNKNOWN BOM$ ERROR
S.BFEI='B'0.                    ;ERROR IN HIGH LEVEL LANGUAGE INTERFACE
S.BOVL='B'1.                    ;LOAD OVERLAY FAILURE

;
; KEEP THE FOLLOWING DEFINED AS THE HIGHEST CODE IN USE
;

S.BHI='B'1.                     ;HIGHEST ACCEPTABLE BOM$ CODE

;
; BIT DEFINITIONS FOR BOM$ FLAGS WORD
;
S.BBIF='B'1.                    ;ENABLE CONDITIONAL BREAKPOINT
```

ABODF$ (Cont.)

```
;
; TASK TERMINATION NOTIFICATION MESSAGE CODES
;

T.NDNR='B'0                     ;DEVICE NOT READY
T.NDSE='B'2                     ;DEVICE SELECT ERROR
T.NCWF='B'4                     ;CHECKPOINT WRITE FAILURE
T.NCRE='B'6                     ;CARD READER HARDWARE ERROR
T.NDMO='B'8.                    ;DISMOUNT COMPLETE
T.NUER='B'10.                   ;UNRECOVERABLE ERROR
T.NLDN='B'12.                   ;LINK DOWN (NETWORKS)
T.NLUP='B'14.                   ;LINK UP (NETWORKS)
T.NCFI='B'16.                   ;CHECKPOINT FILE INACTIVE
T.NUDE='B'18.                   ;UNRECOVERABLE DEVICE ERROR
T.NMPE='B'20.                   ;MEMORY PARITY ERROR
T.NKLF='B'22.                   ;UCODE LOADER NOT INSTALLED
T.NAAF='B'24.                   ;ACCOUNTING ALLOCATION FAILURE
T.NTAF='B'26.                   ;ACCOUNTING TAB ALLOCATION FAILURE
T.NDEB='B'28.                   ;TASK HAS NO DEBUGGING AID
T.NRCT='B'30.                   ;REPLACEMENT CONTROL TASK NOT INSTALLED
T.NWBL='B'32.                   ;WRITE BACK CACHING DATA LOST
                                ;UNIT WRITE LOCKED
T.NVER='B'34.                   ;MOUNT VERIFICATION TASK NOT INSTALLED
T.NIOS='B'36.                   ;I/O STALLED TO DEVICE
T.NIOR='B'38.                   ;I/O RESUMING ON DEVICE

        .MACRO  ABODF$  X,Y
        .ENDM
        .ENDM
```

ACNDF$

```
        .MACRO  ACNDF$,L,B

;+
; ACCOUNTING  BLOCK OFFSET AND STATUS DEFINITIONS
; FOR EACH TRANSACTION TYPE.
;
; HEADER COMMON TO ALL TRANSACTIONS
;-

        .ASECT
.=0

B.LNK:'L'.BLKW  1               ;LINK TO NEXT IN SYSLOG QUEUE
B.TYP:'L'.BLKB  1               ;TRANSACTION TYPE
B.LEN:'L'.BLKB  1               ;TRANSACTION LENGTH
B.TIM:'L'.BLKW  3               ;ENDING TIME OF TRANSACTION
B.HID='B'.                      ;START OF HEADER IDENTIFICATION AREA
B.UID:'L'.BLKW  2               ;UNIQUE SESSION IDENT
                                ; FIRST WORD-RAD50, SECOND-BINARY
B.ACN:'L'.BLKW  1               ;ACCOUNT NUMBER
B.TID:'L'.BLKB  1               ;ASCII TERMINAL TYPE (V,T,B OR C)
                                ;(VIRTUAL,REAL,BATCH, OR CONSOLE)
         .BLKB  1               ;UNIT NUMBER
B.HEND='B'.                     ;END OF HEADER ID AREA
$$$HLN=.                        ;HEADER LENGTH


;+
; ACCUMULATION FIELDS FOR TAB, UAB, AND SAB
;-

B.CPU:'L'.BLKW  2               ;TOTAL CPU TIME USED
B.DIR:'L'.BLKW  2               ;TOTAL DIRECTIVE COUNT
B.QIO:'L'.BLKW  2               ;TOTAL QIO$ COUNT
B.TAS:'L'.BLKW  2               ;TOTAL TASK COUNT
B.MEM:'L'.BLKW  3               ;RESERVED
B.BEG:'L'.BLKW  3               ;BEGINNING/LOGIN TIME
B.CPUL:'L'.BLKW 2               ;CPU LIMIT
B.PNT:'L'.BLKW  1               ;POINTER TO HIGHER LEVEL TOTALS
B.STM:'L'.BLKB  1               ;STATUS MASK
$$$TLN=.                        ;TOTAL'S LENGTH


;+
; USER ACCOUNT BLOCK (UAB)
;       NOTE:  UAB'S MUST END ON A WORD BOUNDRY
;-

.=$$$TLN                        ;START AFTER TOTALS
B.USE:'L'.BLKB  1               ;USE COUNT
B.ACT:'L'.BLKW  1               ;NUMBER OF CURRENTLY ACTIVE TASKS
B.UUIC:'L'.BLKW 1               ;LOGIN UIC
B.UCB:'L'.BLKW  1               ;POINTER TO UCB
B.LGO:'L'.BLKW  3               ;LOGOFF TIME
B.ULNK:'L'.BLKW 1               ;LINK TO NEXT UAB
B.RNA:'L'.BLKW  3               ;LOC IN SYSTEM ACCNT FILE
                                ;(OFFSET,VBN-HI,VBN-LO)
```

ACNDF$ (Cont.)

```
B.NAM:'L'.BLKB  14.             ;LAST NAME OF USER
         .BLKB  1               ;FIRST INITIAL OF USER
         .BLKB  1               ;FLAG BYTE FOR UAB (bs.sil) etc.
B.LDS:'L'.BLKB  10.             ;LOGIN DIRECTORY STRING
B.ULEN='B'.                     ;UAB LENGTH
$$$=    <.+77>/100              ;UAB LENGTH (ROUNDED UP TO 32 WORD BOUND)

;+
; TASK ACCOUNT BLOCK (TAB)
;       NOTE: THE TAB MUST END ON A WORD BOUNDARY
;-

.=$$$TLN                        ;STARTS AFTER TOTALS
B.PRI:'L'.BLKB  1               ;HIGHEST RUNNING PRIORITY
B.TNAM:'L'.BLKW 2               ;TASK NAME
B.TCB:'L' .BLKW 1               ;TCB ADDRESS
B.TST3:'L'.BLKW 1               ;T.ST3 FROM TASK'S TCB
          .BLKW 1               ;RESERVED FOR FUTURE STATUS BITS
B.CUIC:'L'.BLKW 1               ;CURRENT UIC OF TASK
B.PUIC:'L'.BLKW 1               ;PROTECTION UIC OF TASK
B.CTXT:'L'.BLKW 2               ;NUMBER OF CONTEXT LOADS
B.TCKP:'L'.BLKW 2               ;TIMES TASK HAS BEEN CHECKPOINTED
B.OVLY:'L'.BLKW 2               ;NUMBER OF DISK OVERLAY LOADS
B.EXST:'L'.BLKW 2               ;EXIT STATUS AND ABORT CODE
B.TLEN='B'.                     ;TAB LENGTH
B.TBLK='B'<.+77>/100            ;NUMBER OF SEC POOL BLOCKS IN TAB


;+
; SYSTEM ACCOUNT BLOCK (SAB)
;-

.=$$$TLN                        ;START AFTER TOTALS
B.SHDN:'L'.BLKB 1               ;ACCOUNTING SHUTDOWN REASON CODE
B.UHD:'L'.BLKW  1               ;UAB LISTHEAD
B.ULO:'L'.BLKW  1               ;NUMBER OF USERS CURRENTLY LOGGED ON
B.ULT:'L'.BLKW  2               ;TOTAL NUMBER OF LOGONS
B.CKP:'L'.BLKW  2               ;TOTAL NUMBER OF CHECKPOINTS
B.SHF:'L'.BLKW  2               ;TOTAL NUMBER OF SHUFFLER RUNS
B.RND:'L'.BLKW  2               ;NUMBER OF CPU INTERVALS ROUNDED UP TO 1
B.FID:'L'.BLKW  3               ;FILE-ID OF TRANSACTION FILE
B.DVNM:'L'.BLKB 2               ;DEVICE  OF TRANSACTION FILE
B.UNIT:'L'.BLKW 1               ;UNIT    OF TRANSACTION FILE
B.EXTS:'L'.BLKW 1               ;EXTEND SIZE FOR TRANSACTION FILE
B.LSCN:'L'.BLKW 3               ;TIME OF LAST SCAN
B.SCNR:'L'.BLKW 1               ;SCAN RATE IN SECONDS
B.DSCN:'L'.BLKW 1               ;STATISTICAL SCAN RATE (IN SEC)
B.STSP:'L'.BLKW 2               ;RESERVED
B.SYSM:'L'.BLKW 1               ;RESERVED
B.CKUS:'L'.BLKW 3               ;RESERVED
B.CKSP:'L'.BLKW 2               ;RESERVED
B.CKAL:'L'.BLKW 1               ;RESERVED
B.SLEN='B'.                     ;SAB LENGTH

; NEW FIELDS FOR EXTENDED ACCOUNTING
```

ACNDF$ (Cont.)

```
B.CPUT:'L'.BLKW 8.              ;CPU TIME USED PER PROCESSOR
B.CTXP:'L'.BLKW 8.              ;NUMBER OF CONTEXT SWITCHES (PER PROC)
B.IDCT:'L'.BLKW 8.              ;NUMBER OF IDLE LOOP ENTRIES (PER PROC)
B.QIOC:'L'.BLKW 8.              ;NUMBER OF I/O INITIATIONS (PER PROC)
B.MIOC:'L'.BLKW 8.              ;MASS STORE I/O COMPLETIONS (PER PROC)
B.AIOC:'L'.BLKW 8.              ;ALL I/O COMPLETIONS (PER PROC)
B.IPSN:'L'.BLKW 8.              ;IP INTERRUPTS SENT (PER PROC)
B.IPRC:'L'.BLKW 8.              ;IP INTERRUPTS RCVD (PER PROC)
B.CKEX:'L'.BLKW 2               ;CHECKPOINT DUE TO EXTEND TASKS
B.CFCL:'L'.BLKW 2               ;CALLS TO CFORK
B.CFRK:'L'.BLKW 2               ;CFORK FORKS
B.TLOD:'L'.BLKW 2               ;TASK LOADS
B.RLOD:'L'.BLKW 2               ;REGION LOADS
          .BLKB 82.             ;BUMP SIZE TO NEXT 32 WORD BLOCK
B.SSBL=.-B.SLEN                 ;EXTRA LENGTH OF SYSTEM STATISTICS BLOCK
$$$=    <.+77>/100              ;SAB LENGTH (ROUNDED UP TO 32 WORD BOUND)


;+
; SYSLOG STARTUP TRANSACTION
;-

.=      $$$HLN                  ;START AFTER HEADER
B.SSLN='B'                      ;TRANSACTION LENGTH

;+
; CRASH RECOVERY TRANSACTION
;-

.=      $$$HLN                  ;START AFTER STANDARD HEADER
B.CTLS:'L'.BLKW 3               ;TIME OF LAST SCAN BEFORE CRASH
B.CSRT:'L'.BLKW 1               ;SCAN RATE BEFORE CRASH
B.CRSN:'L'.BLKB 60.             ;ASCII TEXT EXPLAINING CRASH
B.CLEN='B'.                     ;TRANSACTION LENGTH


;+
; INVALID LOGIN TRANSACTION
;-

.=      $$$HLN                  ;
B.INAM:'L'.BLKB 14.             ;NAME FROM LOGIN LINE
B.IUIC:'L'.BLKB 6.              ;UIC FROM LOGIN LINE
B.IPSW:'L'.BLKB 6.              ;PASSWORD FROM LOGIN LINE
B.ILEN='B'.                     ;TRANSACTION LENGTH


;+
; DEVICE TRANSACTIONS (ALLOCATION, DEALLOCATION, MOUNT, AND DISMOUNT)
;-

.=      $$$HLN                  ;
B.DNAM:'L'.BLKW 1               ;ASCII DEVICE NAME
B.DUNT:'L'.BLKB 1               ;OCTAL DEVICE UNIT NUMBER
B.DLEN='B'.                     ;TRANSACTION LENGTH FOR ALL, DEA, AND DMO
          .BLKB 1               ;UNUSED BYTE
```

ACNDF$ (Cont.)

```
B.DLBL:'L'.BLKW 6               ;VOLUME LABEL
B.DMST:'L'.BLKW 1               ;MOUNT STATUS BITS
B.DUIC:'L'.BLKW 1               ;OWNER UIC
B.DVPR:'L'.BLKW 1               ;VOLUME PROTECTION CODE
B.DACP:'L'.BLKW 2               ;NAME OF ACP FOR DEVICE
B.MLEN='B'.                     ;LENGTH OF MOUNT TRANSACTION

;+
; STATUS BITS FOR MOUNT STATUS MASK (B.DMST)
;-

BM.SHR='B'1                     ;DEVICE IS MOUNTED SHARED
BM.NOS='B'2                     ;DEVICE IS MOUNTED NOSHARE
BM.SYS='B'4                     ;DEVICE IS MOUNTED FOR THE SYSTEM (PUBLIC)
BM.FOR='B'10                    ;DEVICE IS MOUNTED FOREIGN


;+
; SYSTEM TIME CHANGE TRANSACTION
;-

.=      $$$HLN                  ;
B.TOLD:'L'.BLKB 6               ;OLD TIME (YR, MON, DAY, HR, MIN, SEC)
B.TNEW:'L'.BLKB 6               ;NEW TIME (YR, MON, DAY, HR, MIN, SEC)
B.TMLN='B'.                     ;TRANSACTION LENGTH


;+
; PRINT DESPOOLER TRANSACTION
;-

.=      $$$HLN                  ;START AFTER HEADER
B.PNAM:'L'.BLKW 3               ;PRINT JOB NAME  (RAD50)
B.PPGS:'L'.BLKW 1               ;PAGE COUNT
B.PNFI:'L'.BLKW 1               ;NUMBER OF FILES PRINTED
B.PFRM:'L'.BLKB 1               ;FORM NUMBER
B.PPRI:'L'.BLKB 1               ;PRINT PRIORITY
B.PDEV:'L'.BLKW 1               ;PRINT DEVICE NAME  (ASCII)
B.PPUN:'L'.BLKB 1               ;UNIT NUMBER OF PRINT DEVICE
B.PLEN='B'.                     ;TRANSACTION LENGTH

;+
; CARD READER SPOOLING TRANSACTION
;-

.=      $$$HLN                  ;START AFTER HEADER
B.RNAM:'L'.BLKW 3               ;BATCH OR PRINT JOB NAME
B.RCDS:'L'.BLKW 1               ;NUMBER OF CARDS READ
B.RDEV:'L'.BLKW 1               ;READER DEVICE NAME (ASCII)
B.RUNT:'L'.BLKB 1               ;UNIT NUMBER OF READER DEVICE
B.RSOP:'L'.BLKB 1               ;SUBMIT OR PRINT (0=SUBMIT, 1=PRINT)
B.RLEN='B'.                     ;TRANSACTION LENGTH


;+
; LOGIN TRANSACTION
;-
```

ACNDF$ (Cont.)

```
.=      $$$HLN                  ;START AFTER HEADER
B.LUIC:'L'.BLKW 1               ;LOGIN UIC
B.LNAM:'L'.BLKB 14.             ;USER'S LAST NAME
          .BLKB 1               ;AND FIRST INITIAL
B.LLEN='B'.                     ;TRANSACTION LENGTH

;+
;  RESET TRANSACTION PARAMETERS
;-

.=$$$HLN                        ;AFTER HEADER
B.OFID:'L'.BLKW 3               ;FILE-ID OF OLD TRN. FILE
B.ODNM:'L'.BLKB 2               ;DEVICE OF OLD TRN. FILE
B.OUNT:'L'.BLKW 1               ;UNIT OF OLD TRN. FILE
B.NFID:'L'.BLKW 3               ;FILE-ID OF NEW TRN. FILE
B.NDNM:'L'.BLKB 2               ;DEVICE OF NEW TRN. FILE
B.NUNT:'L'.BLKW 1               ;UNIT OF NEW TRN. FILE
B.OEXS:'L'.BLKW 1               ;EXT. SIZE FOR OLD TRN. FILE
B.NEXS:'L'.BLKW 1               ;EXT. SIZE FOR NEW TRN. FILE
B.OSCR:'L'.BLKW 1               ;OLD SCAN RATE IN SECONDS
B.NSCR:'L'.BLKW 1               ;NEW SCAN RATE IN SECONDS
B.ODSC:'L'.BLKW 1               ;OLD STATISTICAL SCAN RATE
B.NDSC:'L'.BLKW 1               ;NEW STATISTICAL SCAN RATE
B.RTLN='B'.

;+
; TRANSACTION TYPES
;
;       000 THRU 127            RESERVED FOR DEC USE
;       128 THRU 255            RESERVED FOR CUSTOMER USE
;
;-

BT.SAB='B'1                     ;SYSTEM ACCOUNT BLOCK (SAB)
BT.UAB='B'2                     ;USER ACCOUNT BLOCK (UAB)
BT.TAB='B'3                     ;TASK ACCOUNT BLOCK (TAB)
BT.SS='B'11                     ;SYSLOG STARTUP TRANSACTION
BT.INV='B'12                    ;INVALID LOGIN TRANSACTION
BT.TIM='B'13                    ;SYSTEM TIME CHANGE TRANSACTION
BT.ALL='B'14                    ;ALLOCATE DEVICE TRANSACTION
BT.DEA='B'15                    ;DEALLOCATE DEVICE TRANSACTION
BT.MOU='B'16                    ;MOUNT DEVICE TRANSACTION
BT.DMO='B'17                    ;DISMOUNT DEVICE TRANSACTION
BT.PRT='B'20                    ;PRINT DESPOOLER TRANSACTION
BT.DIR='B'21                    ;DISK ACCOUNTING BY DIRECTORY
                                ; (UNSUPPORTED)
BT.VOL='B'22                    ;DISK ACCOUNTING BY VOLUME
                                ; (UNSUPPORTED)
BT.LOG='B'23                    ;LOGIN TRANSACTION
BT.CRH='B'24                    ;CRASH RECOVERY TRANSACTION
BT.DST='B'25                    ;DEVICE STATISTICS (UCB EXTENSION)
BT.RTP='B'26                    ;RESET TRANSACTION PARAMETERS
BT.INP='B'27                    ;CARD READER SPOOLING TRANSACTION

;+
; STATUS MASK BIT DEFINITIONS (B.STM)
;-
```

ACNDF$ (Cont.)

```
BS.ACT='B'200                   ;CONTROL BLOCK ACTIVE
BS.CRH='B'100                   ;RECORD FROM "TMP" FILE AFTER SYSTEM CRASH
BS.LGO='B'40                    ;LOGGED OFF WITH OUTSTANDING ACTIVITY (UAB)
BS.CO='B'40                     ;TASK'S TI: IS CO: (TAB ONLY)
BS.TML='B'20                    ;TAB EXISTS ONLY FOR TIME LIMIT (TAB ONLY)
BS.SIL='B'20                    ;SILENT LOGIN/LOGOUT (UAB ONLY)
BS.ZER='B'10                    ;LAST CPU INTERVAL WAS OF LENGTH ZERO
BS.SCN='B'4                     ;TRANSACTION READY FOR WRITE TO SCAN FILE


;+
; ACCOUNTING FEATURE MASK ($ACNFE)
;-

BF.DST='B'40000                 ;STATISTICAL SCAN RATE
BF.WRT='B'2000                  ;FORCE SYSLOG TO WRITE ITS BUFFER
BF.SCN='B'1000                  ;SCAN REQUESTED
BF.SLR='B'400                   ;SYSLOG IS RUNNING (NOT STOPPED)
BF.ERR='B'200                   ;ACCOUNTING STOPPED DUE TO FATAL ERROR
BF.STR='B'100                   ;ACCOUNTING IS STARTING UP / SHUTTING DOWN
BF.LSS='B'40                    ;ACCUMULATE SYSTEM STATISTICS
                                ; (POINT UAB TO SAB)
BF.TRN='B'10                    ;OUTPUT TO TRANSACTION FILE
BF.XTK='B'4                     ;CHECKPOINT REQUEST IS DUE TO EXTK$
BF.TSK='B'2                     ;TASK ACCOUNTING TURNED ON
BF.XAC='B'1                     ;EXTENDED ACCOUNTING ASSEMBLED IN

;+
; SHUTDOWN CODES (B.SHDN)
;-

; 1             MAINTENANCE
; 2             REBOOT
; 3             SCHEDULED SHUTDOWN
; 4             ACCOUNTING SHUTDOWN BY TASK "SHUTUP"
; 5             OTHER


;*****************************************************************
B.MAXL='B'128.                  ;MAXIMUM TRANSACTION LENGTH
B.MINL='B'$$$HLN                ;MINIMUM TRANSACTION LENGTH
;*****************************************************************


        .PSECT
        .MACRO  ACNDF$  X,Y
        .ENDM
        .ENDM

        .MACRO  ACTDF$,L,B
        .ASECT
.=0
A.GRP:'L'       .BLKB   3               ; GROUP CODE (ASCII)
A.MBR:'L'       .BLKB   3               ; MEMBER CODE
A.PSWD:'L'      .BLKB   6               ; PASSWORD
A.LNM:'L'       .BLKB   14.             ; LAST NAME
A.FNM:'L'       .BLKB   12.             ; FIRST NAME
A.LDAT:'L'      .BLKB   6               ; DATE OF LAST LOGON
                                        ;  FORMAT = (DD/MM/YY HH:MM:SS)
```

ACNDF$ (Cont.)

```
A.NLOG:'L'      .BLKB   2               ; TOTAL NUMBER OF LOGONS
A.SYDV:'L'      .BLKB   4               ; DEFAULT SYSTEM DEVICE
A.ACN:'L'       .BLKW   1               ;ACCOUNT NUMBER (BINARY)
A.CLI:'L'       .BLKW   2               ; RAD50 USER CLI
                .BLKW   2               ; UNUSED
A.LPRV:'L'      .BLKW   1               ;LOGIN PRIVILEGE WORD
A.SID:'L'       .BLKW   1               ; SESSION IDENTIFIER
A.DDS:'L'       .BLKB   11.             ;DEFAULT DIRECTORY STRING
                .BLKB   1               ;UNUSED BYTE
A.FPRO:'L'      .BLKW   1               ;DEFAULT FILE PROTECTION
A.RLVL:'L'      .BLKW   1               ;ACCOUNT RECORD REV. LEVEL
AR.LVL='B'401
A.SALT:'L'      .BLKW   1               ;16-BIT ENCRYPTION SALT VALUE
A.ENCT:'L'      .BLKB   1               ;ENCRYPTION TYPE
                                        ;       0 = PLAIN TEXT OR ENCRYPT
                                        ;       1 = PURDY-V ALGORITHM
                .BLKB   1               ;UNUSED
A.HPW:'L'       .BLKW   4               ;HASHED PASSWORD

                .IF DF  A$$LOG

A.TTY:'L'       .BLKB   5               ;TERMINAL TTNNN FOR AUTO LOGIN
A.PRID:'L'      .BLKB   1               ;PRIMARY DAYS MASK
A.SECD:'L'      .BLKB   1               ;SECONDARY DAYS MASK
A.PRIT:'L'      .BLKW   1               ;PRIMARY DAYS TIME
A.SECT:'L'      .BLKW   1               ;SECONDARY DAYS TIME
A.RLEN  ='B'    .

                .ENDC   ; DF A$$LOG

A.LEN   ='B'    128.                    ;LENGTH OF CONTROL BLOCK
;
;
; BIT DEFINITIONS ON A.LPRV - LOGIN PRIVILEGE BITS
;
AL.SLV='B'      1                       ;SLAVE TERMINAL ON LOGIN
AL.DDS='B'      2                       ;INDICATOR FOR PROLOGUE 2 FORMAT
AL.SIL='B'      4                       ;SILENT LOGIN/LOGOUT

                .IF DF A$$LOG

AL.AUT='B'      10                      ;AUTO LOGIN ENABLED     ('*)
AL.BND='B'      20                      ;BINDING ENABLED        ('Y)
AL.RMT='B'      40                      ;REMOTE DIALUP 1=NO
AL.NET='B'      100                     ;NETWORK LOGIN 1=NO
AL.DIS='B'      200                     ;DISABLE THIS ACCOUNT FROM LOGIN
AL.PRI='B'      400                     ;PRIMARY DAYS LIMIT SET
AL.SEC='B'      1000                    ;SECONDARY DAYS LIMIT SET

                .ENDC ; DF A$$LOG

;
        .PSECT
        .ENDM
```

CLKDF$

```
        .MACRO  CLKDF$,L,B

;+
; CLOCK QUEUE CONTROL BLOCK OFFSET DEFINITIONS
;
; CLOCK QUEUE CONTROL BLOCK
;
; THERE ARE FIVE TYPES OF CLOCK QUEUE CONTROL BLOCKS. EACH CONTROL BLOCK
; HAS THE SAME FORMAT IN THE FIRST FIVE WORDS AND DIFFERS IN THE REMAINING
; THREE. THE FOLLOWING CONTROL BLOCK TYPES ARE DEFINED:
;-

C.MRKT='B'0                     ;MARK TIME REQUEST
C.SCHD='B'2                     ;TASK REQUEST WITH PERIODIC RESCHEDULING
C.SSHT='B'4                     ;SINGLE SHOT TASK REQUEST
C.SYST='B'6                     ;SINGLE SHOT INTERNAL SYSTEM SUBROUTINE
                                ; (IDENT)
C.SYTK='B'8.                    ;SINGLE SHOT INTERNAL SYSTEM SUBROUTINE
                                ; (TASK)
C.CSTP='B'10.                   ;CLEAR STOP BIT (CONDITIONALIZED ON
                                ; SHUFFLING)

;
; CLOCK QUEUE CONTROL BLOCK TYPE INDEPENDENT OFFSET DEFINTIONS
;

        .ASECT
.=0
C.LNK:'L' .BLKW 1               ;CLOCK QUEUE THREAD WORD
C.RQT:'L' .BLKB 1               ;REQUEST TYPE
C.EFN:'L' .BLKB 1               ;EVENT FLAG NUMBER (MARK TIME ONLY)
C.TCB:'L' .BLKW 1               ;TCB ADDRESS OR SYSTEM SUBROUTINE
                                ; IDENTIFICATION
C.TIM:'L' .BLKW 2               ;ABSOLUTE TIME WHEN REQUEST COMES DUE

;
; CLOCK QUEUE CONTROL BLOCK-MARK TIME DEPENDENT OFFSET DEFINITIONS
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
C.AST:'L' .BLKW 1               ;AST ADDRESS
C.SRC:'L' .BLKW 1               ;FLAG MASK WORD FOR 'BIS' SOURCE
C.DST:'L' .BLKW 1               ;ADDRESS OF 'BIS' DESTINATION
        .BLKW   1               ;UNUSED

;
; CLOCK QUEUE CONTROL BLOCK-PERIODIC RESCHEDULING DEPENDENT OFFSET DEFINITIONS
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
C.RSI:'L' .BLKW 2               ;RESCHEDULE INTERVAL IN CLOCK TICKS
C.UIC:'L' .BLKW 1               ;SCHEDULING UIC
C.UAB:'L'.BLKW  1               ;POINTER TO ASSOCIATED UAB

;
; CLOCK QUEUE CONTROL BLOCK-SINGLE SHOT DEPENDENT OFFSET DEFINITIONS
;
```

CLKDF$ (Cont.)

```
.=C.TIM+4                       ;START OF DEPENDENT AREA
        .BLKW   2               ;TWO UNUSED WORDS
        .BLKW   1               ;SCHEDULING UIC
        .BLKW   1               ;C.UAB

;
; CLOCK QUEUE CONTROL BLOCK-SINGLE SHOT INTERNAL SUBROUTINE OFFSET
;   DEFINITIONS
;
; THERE ARE TWO TYPE CODES FOR THIS TYPE OF REQUEST:'L'
;
;   TYPE 6=SINGLE SHOT INTERNAL SUBROUTINE WITH A 16 BIT VALUE AS AN
;     IDENTIFIER.
;   TYPE 8=SINGLE SHOT INTERNAL SUBROUTINE WITH A TCB ADDRESS AS AN
;     IDENTIFIER.
;

.=C.TIM+4                       ;START OF DEPENDENT AREA
C.SUB:'L' .BLKW 1               ;SUBROUTINE ADDRESS
C.AR5:'L' .BLKW 1               ;RELOCATION BASE (FOR LOADABLE DRIVERS)
C.URM:'L'.BLKW  1               ;URM TO EXECUTE ROUTINE ON
                                ;(MP SYSTEMS, C.SYST ONLY)
        .BLKW   1               ;UNUSED
C.LGTH='B'.                     ;LENGTH OF CLOCK QUEUE CONTROL BLOCK

;
;       NAMED DIRECTORY SUPPORT
;
; OFFSET C.EFN WILL BE REUSED IN SCHEDULING REQUESTS TO INDICATE IF C.UIC
; HAS A VALID UIC (C.NAM=0) OR IF C.UIC POINTS TO A CONTEXT BLOCK WITH A
; DDS. IN SCHEDULE REQUESTS, C.EFN WILL BE REFERRED TO AS C.NAM.

C.NAM='B'C.EFN                  ;FLAG WORD FOR USE WITH NAME DIRECTORIES
;
; DEPENDING ON THE VALUE IN C.NAM, C.UIC WILL CONTAIN A UIC OR A POINTER
; TO A CONTEXT BLOCK.  C.UIC WILL BE REFERED TO AS C.CTX WHEN IT CONTAINS
; A POINTER TO A CONTEXT BLOCK.
;

C.CTX='B'C.UIC                  ;POINTER TO CONTEXT BLOCK

        .PSECT

        .MACRO  CLKDF$  X,Y
        .ENDM
        .ENDM
```

CTBDF$

```
        .MACRO      CTBDF$

        ;
        ; CONTROLLER TABLE (CTB)
        ;
        ; THE CONTROLLER TABLE IS A CONTROL BLOCK THAT CONTAINS A
        ; VECTOR OF KRB ADDRESSES.  THIS VECTOR MAY BE ADDRESSED
        ; BY THE CONTROLLER INDEX TAKEN FROM THE  INTERRUPT PS BY
        ; $INTSV/$INTSE.
        ;
              .ASECT
        .=177756
177756  L.CLK:  .BLKW   8.      ;START OF CLOCK BLOCK (IF ANY)
177776  L.ICB:  .BLKW   1       ;ICB CHAIN FOR THIS CTB
000000  L.LNK:  .BLKW   1       ;CTB LINK WORD
000002  L.NAM:  .BLKW   1       ;GENERIC CONTROLLER NAME (ASCII)
000004  L.DCB:  .BLKW   1       ;DCB ADDRESS OF THIS DEVICE
000006  L.NUM:  .BLKB   1       ;NUMBER OF KRB ADDRESSES IN TABLE
000007  L.STS:  .BLKB   1       ;CTB STATUS BYTE
000010  L.KRB:  .BLKW   1       ;START OF KRB ADDRESSES


        ;
        ; NOTE: THE SYMBOL $XYCTB:: IS DEFINED FOR EACH CTB,
        ; WHERE THE CHARACTERS XY ARE THE SAME AS THOSE STORED IN
        ; L.NAM.  THESYMBOL IS NOT THE START OF THE CTB, BUT THE
        ; START OFTHE KRB TABLE AT THE END OF THE CTB (L.KRB).
        ;
              .PSECT


        ;
        ; CONTROLLER TABLE STATUS BYTE BIT DEFINITIONS
        ;
        LS.CLK=1                ;CLOCK BLOCK AT TOP OF CTB (1=YES)
        LS.MDC=2                ;MULTIDRIVER CTB (1=YES)
        LS.CBL=4                ;CLOCK BLK LINKED INTO CLK Q (1=YES)
        LS.CIN=10               ;CONT. USE COMMON INT TABLE (1=YES)
        LS.NET=20               ;THIS IS DECNET DEVICE.
                                ;ICB'S IN K.PRM
                                ;(1=YES)
        ;
        ; COMMON INTERRUPT TABLE DISPATCH ENTRY POINTS
        ;
        CI.CSR=-6               ;CSR TEST ENTRY POINT
        CI.KRB=-4               ;KRB STATUS CHANGE ENTRY POINT
        CI.PWF=-2               ;POWERFAIL ENTRY POINT
        CI.INT=0                ;COMMON INTERRUPT ADDRESS
        CI.DCB=2                ;START OF DCB TABLE (0 ENDS TABLE)
```

DCBDF$

```
        .MACRO  DCBDF$,L,B
;+
;
; DEVICE CONTROL BLOCK
;
; THE DEVICE CONTROL BLOCK (DCB) DEFINES GENERIC INFORMATION ABOUT A DEVICE
; TYPE AND THE LOWEST AND HIGHEST UNIT NUMBERS. THERE IS AT LEAST ONE DCB
; FOR EACH DEVICE TYPE IN A SYSTEM. FOR EXAMPLE, IF THERE ARE TELETYPES IN A
; SYSTEM, THEN THERE IS AT LEAST ONE DCB WITH THE DEVICE NAME 'TT'. IF PART
; OF THE TELETYPES WERE INTERFACED VIA DL11-A'S AND THE REST VIA A DH11, THEN
; THERE WOULD BE TWO DCB'S. ONE FOR ALL DL11-A INTERFACED TELETYPES, AND ONE
; FOR ALL DH11 INTERFACED TELETYPES.
;-

        .ASECT
.=0
D.LNK:'L' .BLKW 1               ;LINK TO NEXT DCB
D.UCB:'L' .BLKW 1               ;POINTER TO FIRST UNIT CONTROL BLOCK
D.NAM:'L' .BLKW 1               ;GENERIC DEVICE NAME
D.UNIT:'L' .BLKB 1              ;LOWEST UNIT NUMBER COVERED BY THIS DCB
        .BLKB   1               ;HIGHEST UNIT NUMBER COVERED BY THIS DCB
D.UCBL:'L' .BLKW 1              ;LENGTH OF EACH UNIT CONTROL BLOCK IN BYTES
D.DSP:'L' .BLKW 1               ;POINTER TO DRIVER DISPATCH TABLE
D.MSK:'L' .BLKW 1               ;LEGAL FUNCTION MASK  CODES 0-15.
          .BLKW 1               ;CONTROL FUNCTION MASK CODES 0-15.
          .BLKW 1               ;NOP'ED FUNCTION MASK CODES 0-15.
          .BLKW 1               ;ACP FUNCTION MASK CODES 0-15.
          .BLKW 1               ;LEGAL FUNCTION MASK CODES 16.-31.
          .BLKW 1               ;CONTROL FUNCTION MASK CODES 16.-31.
          .BLKW 1               ;NOP'ED FUNCTION MASK CODES 16.-31.
          .BLKW 1               ;ACP FUNCTION MASK CODES 16.-31.
D.PCB:'L' .BLKW 1               ;LOADABLE DRIVER PCB ADDRESS

        .PSECT

;+
; DRIVER DISPATCH TABLE OFFSET DEFINITIONS
;-

D.VDEB='B'-2                    ;DEALLOCATE BUFFER(S)
D.VCHK='B'-4                    ;ADDRESS OF ROUTINE CALLED TO VALIDATE
                                ;AND CONVERT THE LBN.  USED BY DRIVERS
                                ;THAT SUPPORT SEEK OPTIMIZATION.
D.VNXC='B'-4                    ;ADDRESS OF ROUTINE IN TTDRV CALLED TO
                                ;HAVE IT SEND THE NEXT COMMAND IN THE
                                ;TYPEAHEAD BUFFER TO MCR...
D.VTOU='B'-10                   ;ADDRESS OF ROUTINE IN TTDRV CALLED
                                ;FOR OUTPUT COMPLETION
D.VTIN='B'-6                    ;ADDRESS OF ROUTINE IN TTDRV CALLED
                                ;FOR INPUT FROM THE CT FIRMWARE TASK
D.VINI='B'0                     ;DEVICE INITIATOR
D.VCAN='B'2                     ;CANCEL CURRENT I/O FUNCTION
D.VOUT='B'4                     ;DEVICE TIMEOUT
D.VPWF='B'6                     ;POWERFAIL RECOVERY
D.VKRB='B'10                    ;CONTROLLER STATUS CHANGE ENTRY
D.VUCB='B'12                    ;UNIT STATUS CHANGE ENTRY
```

DCBDF$ (Cont.)

```
        .IF NB  SYSDEF

D.VINT='B'14                        ;BEGINNING OF INTERRUPT DATA

        .ENDC


        .MACRO  DCBDF$,X,Y,Z
        .ENDM
        .ENDM
```

EPKDF$

```
        .MACRO  EPKDF$,L,B

;+
; Error Message Block Definitions
;-
        .ASECT
;
; Header Subpacket
;
;
;       +-----------------------------------------------+
;       | Subpacket Length in Bytes                     |
;       +-----------------------------------------------+
;       | Subpacket Flags                               |
;       +-----------------------+-----------------------+
;       | Format Identification | Operating System Code |
;       +-----------------------+-----------------------+
;       | Operating System Identification               |
;       |                                               |
;       +-----------------------+-----------------------+
;       | Flags                 | Context Code          |
;       +-----------------------+-----------------------+
;       | Entry Sequence                                |
;       +-----------------------------------------------+
;       | Error Sequence                                |
;       +-----------------------+-----------------------+
;       | Entry Type Subcode    | Entry Type Code       |
;       +-----------------------+-----------------------+
;       | Time Stamp                                    |
;       |                                               |
;       |                                               |
;       +-----------------------+-----------------------+
;       | Reserved              | Processor Type        |
;       +-----------------------+-----------------------+
;       | Processor Identification (URM)                |
;       +-----------------------------------------------+
;
.=0

E$HLGH:'L'      .BLKW   1       ; Subpacket length in bytes
E$HSBF:'L'      .BLKW   1       ; Subpacket Flags
E$HSYS:'L'      .BLKB   1       ; Operating System Code
E$HIDN:'L'      .BLKB   1       ; Format Identification
E$HSID:'L'      .BLKB   4       ; Operating System Identification
E$HCTX:'L'      .BLKB   1       ; Context Code
E$HFLG:'L'      .BLKB   1       ; Flags
E$HENS:'L'      .BLKW   1       ; Entry Sequence Number
E$HERS:'L'      .BLKW   1       ; Error Sequence Number
E$HENC:'L'                      ; Entry Code
E$HTYC:'L'      .BLKB   1       ; Entry Type Code
E$HTYS:'L'      .BLKB   1       ; Entry Type Subcode
E$HTIM:'L'      .BLKB   6       ; Time Stamp
E$HPTY:'L'      .BLKB   1       ; Processor Type
                .BLKB   1       ; Reserved
E$HURM:'L'      .BLKW   1       ; Processor Identification (URM)

                .EVEN
E$HLEN:'L'                      ; Length
```

EPKDF$ (Cont.)

```
;
; Subpacket Flags for E$HSBF
;
        SM.ERR  ='B'          1  ; Error Packet
        SM.HDR  ='B'          1  ; Header Subpacket
        SM.TSK  ='B'          2  ; Task Subpacket
        SM.DID  ='B'          4  ; Device Identification Subpacket
        SM.DOP  ='B'         10  ; Device Operation Subpacket
        SM.DAC  ='B'         20  ; Device Activity Subpacket
        SM.DAT  ='B'         40  ; Data Subpacket
        SM.MBC  ='B'      20000  ; 22-bit massbus controller present
        SM.CMD  ='B'      40000  ; Error Log Command Packet
        SM.ZER  ='B'     100000  ; Zero I/O Counts
;
; Codes for field E$HIDN
;
        EH$FOR  ='B'          2  ; Current packet format

;
; Flags for the error log flags byte ($ERFLA) in the Executive
;
        ES.INI  ='B'          1  ; Error log initialized
        ES.DAT  ='B'          2  ; Error log receiving data packets
        ES.LIM  ='B'          4  ; Error limiting enabled
        ES.LOG  ='B'         10  ; Error logging enabled

;
; Type and Subtype Codes for fields E$HTYC and E$HTYS
;
;       Symbols with names E$Cxxx are type codes for field E$HTYC,
;       Symbols with names E$Sxxx are subtype codes for field E$HTYS.
;
        E$CCMD  ='B'          1  ; Error Log Control
        E$SSTA  ='B'          1  ;       Error Log Status Change
        E$SSWI  ='B'          2  ;       Switch Logging Files
        E$SAPP  ='B'          3  ;       Append File
        E$SBAC  ='B'          4  ;       Declare Backup File
        E$SSHO  ='B'          5  ;       Show
        E$SCHL  ='B'          6  ;       Change Limits
        E$CERR  ='B'          2  ; Device Errors
        E$SDVH  ='B'          1  ;       Device Hard Error
        E$SDVS  ='B'          2  ;       Device Soft Error
        E$STMO  ='B'          3  ;       Device Interrupt Timeout (HARD)
        E$SUNS  ='B'          4  ;       Device Unsolicited Interrupt
        E$STMS  ='B'          5  ;       Device Interrupt Timeout (SOFT)

        E$CDVI  ='B'          3  ; Device Information
        E$SDVI  ='B'          1  ;       Device Information Message

        E$CDCI  ='B'          4  ; Device Control Information
        E$SMOU  ='B'          1  ;       Device Mount
        E$SDMO  ='B'          2  ;       Device Dismount
        E$SRES  ='B'          3  ;       Device Count Reset
        E$SRCT  ='B'          4  ;       Block Replacement

        E$CMEM  ='B'          5  ; Memory Detected Errors
        E$SMEM  ='B'          1  ;       Memory Error
```

EPKDF$ (Cont.)

```
        E$CSYS  ='B'          6  ; System Control Information
        E$SPWR  ='B'          1  ;       Power Recovery

        E$CCTL  ='B'          7  ; Control Information
        E$STIM  ='B'          1  ;       Time Change
        E$SCRS  ='B'          2  ;       System Crash
        E$SLOA  ='B'          3  ;       Device Driver Load
        E$SUNL  ='B'          4  ;       Device Driver Unload
        E$SHRC  ='B'          5  ;       Reconfiguration Status Change
        E$SMES  ='B'          6  ;       Message

        E$CCPU  ='B'         10  ; CPU Detected Errors
        E$SINT  ='B'          1  ;       Unexpected Interrupt
        E$SINT  ='B'          2  ;       Unexpected Interrupt

;
; Subtype code 2 is reserved.  Use 3 for the next following Subtype code

        E$CSDE  ='B'         11  ; Software Detected Events
        E$SABO  ='B'          1  ;       Task Abort

;
; Codes for Context Code entry E$HCTX
;
        EH$NOR  ='B'          1  ; Normal Entry
        EH$STA  ='B'          2  ; Start Entry
        EH$CRS  ='B'          3  ; Crash Entry

;
; Codes for Flags entry E$HFLG
;
        EH$VIR  ='B'          1  ; Addresses are virtual
        EH$EXT  ='B'          2  ; Addresses are extended
        EH$COU  ='B'          4  ; Error counts supplied
        EH$QBS  ='B'         10  ; Q-BUS CPU
        EH$LMR  ='B'         20  ; Limit reached

;
; Task Subpacket
;
;       +-----------------------------------------------+
;       | Task Subpacket Length                         |
;       +-----------------------------------------------+
;       | Task Name in RAD50                            |
;       |                                               |
;       +-----------------------------------------------+
;       | Task UIC                                      |
;       +-----------------------------------------------+
;       | Task TI: Device Name                          |
;       +-----------------------+-----------------------+
;       | Flags                 | Task TI: Unit Number  |
;       +-----------------------+-----------------------+

.=0
```

EPKDF$ (Cont.)

```
E$TLGH:'L'      .BLKW   1       ; Task Subpacket Length
E$TTSK:'L'      .BLKW   2       ; Task Name in RAD50
E$TUIC:'L'      .BLKW   1       ; Task UIC
E$TTID:'L'      .BLKB   2       ; Task TI: Device Name
E$TTIU:'L'      .BLKB   1       ; Task TI: Unit
E$TFLG:'L'      .BLKB   1       ; Flags

        .EVEN
E$TLEN:'L'

;
; Flags for entry E$TFLG
;
        ET$PRV  ='B'            1  ; Task is Privileged
        ET$PRI  ='B'            2  ; Terminal is Privileged

;
; Device Identification Subpacket
;
;       +-----------------------------------------------+
;       | Device Identification Subpacket Length        |
;       +-----------------------------------------------+
;       | Device Mnemonic Name                          |
;       +-----------------------+-----------------------+
;       | Controller Number     | Device Unit Number    |
;       +-----------------------+-----------------------+
;       | Physical Subunit #    | Physical Unit #       |
;       +-----------------------+-----------------------+
;       | Physical Device Mnemonic (RSX-11M-PLUS only)  |
;       +-----------------------+-----------------------+
;       | Reserved              | Flags                 |
;       +-----------------------+-----------------------+
;       | Volume Name of Mounted Volume                 |
;       |                                               |
;       |                                               |
;       |                                               |
;       |                                               |
;       +-----------------------------------------------+
;       | Pack Identification                           |
;       |                                               |
;       +-----------------------------------------------+
;       | Device Type Class                             |
;       +-----------------------------------------------+
;       | Device Type                                   |
;       |                                               |
;       +-----------------------------------------------+
;       | I/O Operation Count Longword                  |
;       |                                               |
;       +-----------------------+-----------------------+
;       | Hard Error Count      | Soft Error Count      |
;       +-----------------------+-----------------------+
;       | Blocks Transferred Count (RSX-11M-PLUS only)  |
;       |                                               |
;       +-----------------------------------------------+
;       | Cylinders Crossed Count (RSX-11M-PLUS only)   |
;       |                                               |
;       +-----------------------------------------------+
;
```

EPKDF$ (Cont.)

```
.=0

E$ILGH:'L'      .BLKW   1       ; Device Identification Subpacket Length
E$ILDV:'L'      .BLKW   1       ; Device Mnemonic Name
E$ILUN:'L'      .BLKB   1       ; Device Unit Number
E$IPCO:'L'      .BLKB   1       ; Controller Number
E$IPUN:'L'      .BLKB   1       ; Physical Unit Number
E$IPSU:'L'      .BLKB   1       ; Physical Subunit Number

                .IF DF R$$MPL

E$IPDV:'L'      .BLKW   1       ; Physical Device Mnemonic

                .ENDC ; R$$MPL

E$IFLG:'L'      .BLKB   1       ; Flags
                .BLKB   1       ; Reserved
E$IVOL:'L'      .BLKB   12.     ; Volume Name
E$IPAK:'L'      .BLKB   4       ; Pack Identification
E$IDEV:'L'                      ; Device Type
E$IDCL:'L'      .BLKW   1       ; Device Type Class
E$IDTY:'L'      .BLKW   2       ; Device Type
E$IOPR:'L'      .BLKW   2       ; I/O Operation Count Longword
E$IERS:'L'      .BLKB   1       ; Soft Error Count
E$IERH:'L'      .BLKB   1       ; Hard Error Count

                .IF DF R$$MPL

E$IBLK:'L'      .BLKW   2       ; Blocks transferred count
E$ICYL:'L'      .BLKW   2       ; Cylinders crossed count

                .ENDC ; R$$MPL

        .EVEN

E$ILEN:'L'                      ; Subpacket Length
;
; Flags for field E$IFLG
;
        EI$SUB  ='B'        1   ; Subcontroller device

                .IF DF R$$MPL

        EI$NUX  ='B'        2   ; No UCB extension, data invalid

                .ENDC ; R$$MPL

;
; Device Operation Subpacket
;
;       +-----------------------------------------------+
;       | Device Operation Subpacket Length             |
;       +-----------------------------------------------+
;       | Task Name in RAD50                            |
;       |                                               |
;       +-----------------------------------------------+
;       | Task UIC                                      |
;       +-----------------------------------------------+
;       | Task TI: Logical Device Mnemonic              |
;       +-----------------------+-----------------------+
```

(Continued on next page)

EPKDF$ (Cont.)

```
;       +-----------------------+-----------------------+
;       | Reserved              | Task TI: Device Unit  |
;       +-----------------------+-----------------------+
;       | I/O Function Code                             |
;       +-----------------------+-----------------------+
;       | Reserved              | Operation Flags       |
;       +-----------------------+-----------------------+
;       | Transfer Operation Address                    |
;       |                                               |
;       +-----------------------------------------------+
;       | Transfer Operation Byte Count                 |
;       +-----------------------------------------------+
;       |    Maximum Retries    |     Retries Left      |
;       +-----------------------+-----------------------+
;
.=0

E$OLGN:'L'      .BLKW   1       ; Subpacket Length
E$OTSK:'L'      .BLKW   2       ; Task Name in RAD50
E$OUIC:'L'      .BLKW   1       ; Task UIC
E$OTID:'L'      .BLKB   2       ; Task TI: Logical Device Mnemonic
E$OTIU:'L'      .BLKB   1       ; Task TI: Logical Device Unit
                .BLKB   1       ; Reserved
E$OFNC:'L'      .BLKW   1       ; I/O Function Code
E$OFLG:'L'      .BLKB   1       ; Operation Flags
                .BLKB   1       ; Reserved
E$OADD:'L'      .BLKW   2       ; Transfer Operation Address
E$OSIZ:'L'      .BLKW   1       ; Transfer Operation Byte Count
E$ORTY:'L'      .BLKB   1       ; Retries Left
                .BLKB   1       ; Maximum Retries

        .EVEN

E$OLEN:'L'                      ; Device Operation Subpacket Length

;
; Flags for field E$OFLG
;
        EO$TRA  ='B'      1     ; Transfer Operation
        EO$DMA  ='B'      2     ; DMA Device
        EO$EXT  ='B'      4     ; Extended Addressing Device
        EO$PIP  ='B'     10     ; Device is positioning
        EO$IIO  ='B'     20     ; Internal I/O operation
;
; I/O Activity Subpacket
;
;       +-----------------------------------------------+
;       | I/O Activity Subpacket Length                 |
;       +-----------------------------------------------+
;

.=0

E$ALGH:'L'      .BLKW   1       ; Subpacket Length
```

EPKDF$ (Cont.)

```
;
; I/O Activity Subpacket Entry
;
;       +-----------------------------------------------+
;       | Logical Device Name Mnemonic                  |
;       +-----------------------+-----------------------+
;       | Controller Number     | Logical Device Unit   |
;       +-----------------------+-----------------------+
;       | Physical Subunit #    | Physical Unit Number  |
;       +-----------------------+-----------------------+
;       | Physical Device Mnemonic (RSX-11M-PLUS only)  |
;       +-----------------------+-----------------------+
;       | Task TI: logical unit | Device flags          |
;       +-----------------------+-----------------------+
;       | Requesting Task Name in RAD50                 |
;       +-----------------------------------------------+
;       | Requesting Task UIC                           |
;       +-----------------------------------------------+
;       | Task TI: Logical Device Name                  |
;       +-----------------------------------------------+
;       | I/O Function Code                             |
;       +-----------------------+-----------------------+
;       | Reserved              | Flags                 |
;       +-----------------------+-----------------------+
;       | Transfer Operation Address                    |
;       +-----------------------------------------------+
;       | Transfer Operation Byte Count                 |
;       +-----------------------------------------------+

.=0

E$ALDV:'L'      .BLKW   1       ; Logical Device Name Mnemonic
E$ALUN:'L'      .BLKB   1       ; Logical Device Unit
E$APCO:'L'      .BLKB   1       ; Controller Number
E$APUN:'L'      .BLKB   1       ; Physical Unit Number
E$APSU:'L'      .BLKB   1       ; Physical Subunit Number

                .IF DF R$$MPL

E$APDV:'L'      .BLKW   1       ; Physical Device Mnemonic

                .ENDC

E$ADFG:'L'      .BLKB   1       ; Device flags
E$ATIU:'L'      .BLKB   1       ; Task TI: Logical Unit
E$ATSK:'L'      .BLKW   2       ; Requesting Task Name in RAD50
E$AUIC:'L'      .BLKW   1       ; Requesting Task UIC
E$ATID:'L'      .BLKW   1       ; Task TI: Logical Device Name
E$AFNC:'L'      .BLKW   1       ; I/O Function Code
E$AFLG:'L'      .BLKB   1       ; Flags
                .BLKB   1       ; Reserved
E$AADD:'L'      .BLKW   2       ; Transfer Operation Address
E$ASIZ:'L'      .BLKW   1       ; Transfer Operation Byte Count

        .EVEN
```

EPKDF$ (Cont.)

```
E$ALEN:'L'                          ; Subpacket Entry Length
;
; Flags for field E$ADFG
;
        EA$SUB  ='B'          1     ; Subcontroller device

                .IF DF R$$MPL

        EA$NUX  ='B'          2     ; No UCB extension, data invalid

                .ENDC ; R$$MPL
;
; Flags for field E$AFLG
;
        EA$TRA  ='B'          1     ; Transfer Operation
        EA$DMA  ='B'          2     ; DMA Device
        EA$EXT  ='B'          4     ; Device has Extended Addressing
        EA$PIP  ='B'         10     ; Device is positioning
        EA$IIO  ='B'         20     ; Internal I/O operation
        .PSECT

;
; FLAG DEFINITIONS FOR ERROR LOG FEATURE MASK
;
        EL.ICM  ='B'    1           ;SET - Inhibit Operator Console Messages
        EL.SEF  ='B'    2           ;SET - Special File Formats Enabled
        EL.MOU  ='B'    4           ;SET - Process MOU/DMO In Special Files


        .MACRO  EPKDF$  X,Y
        .ENDM

        .ENDM
```

EVNDF$

```
        .MACRO  EVNDF$  L,B,LST

;
; EVNDF$ -- Event Packet Definitions
;
;       This module contains a macro which defines the offsets and field
;       values for TSA Event Packets (TEP's).  These packets are used to
;       pass data and status information between system components that
;       provide Digital's Terminal Software Architecture support on RSX.
;+

;
; Explicit Inputs:
;
;       L       ":" for global offset definitions
;       B       "=" for global bit/value definitions
;       LST     "LIST" for macro expansion listing
;
; Implicit Inputs:
;
;       NONE
;
; Outputs:
;
;       Symbols defined as described above.
;       Listing as described above.
;-


;
; General packet header format
;
        .ASECT                  ; Define offsets absolutely
        .=0

E.VLNK:'L'      .BLKW   1       ; Link word
E.VSIZ:'L'      .BLKB   1       ; Packet size
E.VTYP:'L'      .BLKB   1       ; Packet type
E.VUCB:'L'      .BLKW   1       ; Terminal UCB address

;
; E.VTYP Values
;
ET.LOW='B'0                     ; Lowest valid type code

ET.QIO='B'0                     ; QIO (distinguishes QIO packet from TEP)
ET.BND='B'2                     ; Bind Request
ET.UNB='B'4                     ; Unbind Request
ET.BCP='B'6                     ; Bind Complete
ET.REJ='B'10                    ; Bind Reject
ET.DIS='B'12                    ; Disconnect Notification
ET.DCP='B'14                    ; Disconnect Complete
ET.ICS='B'16                    ; Input Count State Change
ET.OOB='B'20                    ; Out-of-Band (OOB)
ET.ONO='B'22                    ; Abnormal Termination Request
ET.PHO='B'24                    ; Physical Terminal Disconnected

ET.HI='B'24                     ; Highest valid type code
```

EVNDF$ (Cont.)

```
;
;       The following definitions are for packet types that require
;       passing additional information in the packets.  All other
;       packet types use the general packet format described above.
;

;
; Bind Request packet (Terminal Management Mode --> Network)
;
        .=E.VUCB+2

E.VBCT:'L'      .BLKW   1       ; Count of nodes (One for now)
E.VBND:'L'      .BLKB   6       ; Node name
E.VBLN:'L'                      ; Length of bind request

;
; Input Count State Change, Out-Of-Band packets (TTDRV --> Network)
; And Modem Hang-up packets (TSA... --> Network)
;
        .=E.VUCB+2

E.VAPR:'L'      .BLKW   1       ; Doubleword address of packet...
E.VADR:'L'      .BLKW   1       ; ...queueing routine
E.VFLG:'L'      .BLKW   1       ; Flag

;
; Input Count State Change
;
        .=E.VFLG+2

E.VSLN:'L'                      ; Length of Input state message

;
; OOB
;
        .=E.VFLG+2

E.VOBM:'L'      .BLKW   6       ; Out-of-Band bitmasks
E.VHDR:'L'      .BLKW   2       ; Type-ahead buffer header
E.VTAB:'L'      .BLKB   10.     ; Type-ahead buffer
E.VOLN:'L'                      ; Length of OOB packet

;
; Terminal Management Switch Characters
;
        .=E.VFLG+2

E.VSWC:'L'      .BLKW   1       ; Terminal management switch characters
E.VTLN:'L'                      ; Length of Switch Character packet

;
;       Bit values in flag word (E.VFLG).  For convenience some bits have
;       corresponding bits in the AST Control Block flag word (A.PRM+5).
;
```

EVNDF$ (Cont.)

```
EF.NCO='B'1                         ; All non-control characters are out-of band
EF.NOI='B'2                         ; All non-control OOB are include-OOB
EF.AST='B'10                        ; Reserved bit synonymous with TF.AST
EF.LCK='B'40                        ; Reserved bit synonymous with AF.LCK
EF.QUE='B'100                       ; TEP is queued
EF.MDE='B'200                       ; TEP is marked for delete

;
; Unbind Request packet (TMM --> Network)
;
        .=E.VUCB+2

E.VULN:'L'                          ; Length of Unbind message

;
; Connect Reject notification packet (Network --> TMM)
;
        .=E.VUCB+2

E.VRR:'L'       .BLKW   1           ; Reason for Rejection
E.VRLN:'L'                          ; Length of Reject message

;
; Disconnect Notification packet (Network --> TMM)
;
        .=E.VUCB+2

E.VRD:'L'       .BLKW   1           ; Reason for Disconnect
E.VDLN:'L'                          ; Length of Disconnect message

;
; Disconnect Complete packet (TMM --> Network)
;
        .=E.VUCB+2

E.VDCL:'L'                          ; Length of Disconnect Complete message


        .PSECT

.IF NB LST
        .NLIST                      ; Turn listing back off
.IFF
        .MACRO  EVNDF$              ; If not listing, redefine
        .ENDM                       ;  macro to nothing
.ENDC
        .ENDM   EVNDF$
```

F11DF$

```
        .MACRO  F11DF$,L,B

;
; VOLUME CONTROL BLOCK
;

        .ASECT
.=0

V.TRCT:'L'.BLKW 1       ; TRANSACTION COUNT
V.TYPE:'L'.BLKB 1       ; VOLUME TYPE DESCRIPTOR
        VT.FOR='B' 0    ; FOREIGN VOLUME STRUCTURE
        VT.SL1='B' 1    ; FILES-11 STRUCTURE LEVEL 1
        VT.SL2='B' 2    ; FILES-11 STRUCTURE LEVEL 2
        VT.ANS='B' 10   ; ANSI LABELED TAPE
        VT.UNL='B' 11   ; UNLABELED TAPE
V.VCHA:'L'.BLKB 1       ; VOLUME CHARACTERISTICS
        VC.SLK='B' 1    ; CLEAR VOLUME VALID ON DISMOUNT
        VC.HLK='B' 2    ; UNLOAD THE VOLUME ON DISMOUNT
        VC.DEA='B' 4    ; DEALLOCATE THE VOLUME ON DISMOUNT
        VC.PUB='B' 10   ; SET (CLEAR) US.PUB ON DISMOUNT
        VC.DUP='B' 20   ; DUPLICATE VOLUME NAME; DON'T DELETE LOGICALS
        VC.SIL='B' 40   ; SILENT MODE; SUPPRESS DISMOUNT COMPLETE MESSAGE
V.LABL:'L'.BLKB 14      ; VOLUME LABEL (ASCII)
V.PKSR:'L'.BLKW 2       ; PACK SERIAL NUMBER FOR ERROR LOGGING
V.SLEN:'L'              ; LENGTH OF SHORT VCB
V.IFWI:'L'.BLKW 1       ; INDEX FILE WINDOW
V.FCB:'L'.BLKW  2       ; FILE CONTROL BLOCK LIST HEAD
V.IBLB:'L'.BLKB 1       ; INDEX BIT MAP 1ST LBN HIGH BYTE
V.IBSZ:'L'.BLKB 1       ; INDEX BIT MAP SIZE IN BLOCKS
        .BLKW   1       ; INDEX BITMAP 1ST LBN LOW BITS
V.FMAX:'L'.BLKW 1       ; MAX NO. OF FILES ON VOLUME
V.WISZ:'L'.BLKB 1       ; DEFAULT SIZE OF WINDOW IN RTRV PTRS
                        ; VALUE IS < 128.
V.SBCL:'L'.BLKB 1       ; STORAGE BIT MAP CLUSTER FACTOR
V.SBSZ:'L'.BLKW 1       ; STORAGE BIT MAP SIZE IN BLOCKS
V.SBLB:'L'.BLKB 1       ; STORAGE BIT MAP 1ST LBN HIGH BYTE
V.FIEX:'L'.BLKB 1       ; DEFAULT FILE EXTEND SIZE
        .BLKW   1       ; STORAGE BIT MAP 1ST LBN LOW BITS
;
;  WARNING
;
;       THE FOLLOWING CELLS OF THE VCB ARE ORDER DEPENDENT.
;       THEY ARE RETURNED BY A READ ATTRIBUTES FUNCTION AND
;       MUST BE KEPT CONTIGUOUS.  IF THE ORDER OF THE CELLS
;       IS BROKEN, THE CODE MAY BREAK AS WELL.
;

V.VOWN:'L'.BLKW 1       ; VOLUME OWNER'S UIC
V.VPRO:'L'.BLKW 1       ; VOLUME PROTECTION
V.FPRO:'L'.BLKW 1       ; VOLUME DEFAULT FILE PROTECTION
V.FRBK:'L'.BLKB 1       ; NUMBER OF FREE BLOCKS ON VOLUME HIGH BYTE
V.LRUC:'L'.BLKB 1       ; COUNT OF AVAILABLE LRU SLOTS IN FCB LIST
        .BLKW   1       ; NUMBER OF FREE BLOCKS ON VOLUME LOW BITS
```

F11DF$ (Cont.)

```
;
;  WARNING
;
;       THE ABOVE CELLS OF THE VCB ARE ORDER DEPENDENT.
;       THEY ARE RETURNED BY A READ ATTRIBUTES FUNCTION AND
;       MUST BE KEPT CONTIGUOUS.  IF THE ORDER OF THE CELLS
;       IS BROKEN, THE CODE MAY BREAK AS WELL.
;
;
V.STS:'L'.BLKB  1       ; VOLUME STATUS BYTE, CONTAINING THE FOLLOWING
        VS.IFW='B' 1    ; INDEX FILE IS WRITE ACCESSED
        VS.BMW='B' 2    ; STORAGE BITMAP FILE IS WRITE ACCESSED
V.FFNU:'L'.BLKB 1       ; FIRST FREE INDEX FILE BITMAP BLOCK
V.EXT:'L'.BLKW  1       ; POINTER TO VCB EXTENSION
V.HBLB:'L'.BLKW 2       ; LBN OF HOME BLOCK
V.HBCS:'L'.BLKW 2       ; HOME BLOCK CHECKSUMS
V.LGTH:'L'              ; SIZE IN BYTES OF VCB

;
; MOUNT LIST ENTRY
;
; EACH ENTRY ALLOWS ACCESS TO A SPECIFIED USER FOR A NON-PUBLIC DEVICE
;
; TO ALLOW EXPANSION, ONLY THE ONLY TYPE CODE DEFINED IS "1" FOR
; DEVICE ACCESS BLOCKS
;

        .ASECT
.=0

M.LNK:'L'.BLKW  1       ; LINK WORD
M.TYPE:'L'.BLKB 1       ; TYPE OF ENTRY
        MT.MLS='B' 1    ; MOUNTED VOLUME USER ACCESS LIST
M.ACC:'L'.BLKB  1       ; NUMBER OF ACCESSES
M.DEV:'L'.BLKW  1       ; DEVICE UCB
M.TI:'L'.BLKW   1       ; ACCESSOR TI: UCB
M.LEN:'L'               ; LENGTH OF ENTRY

;
; FILE CONTROL BLOCK
;

        .ASECT
.=0

F.LINK:'L'.BLKW 1       ; FCB CHAIN POINTER
F.FNUM:'L'.BLKW 1       ; FILE NUMBER
F.FSEQ:'L'.BLKW 1       ; FILE SEQUENCE NUMBER
        .BLKB   1       ; NOT USED
F.FSQN:'L'.BLKB 1       ; FILE SEGMENT NUMBER
F.FOWN:'L'.BLKW 1       ; FILE OWNER'S UIC
F.FPRO:'L'.BLKW 1       ; FILE PROTECTION CODE
F.UCHA:'L'.BLKB 1       ; USER CONTROLLED CHARACTERISTICS
F.SCHA:'L'.BLKB 1       ; SYSTEM CONTROLLED CHARACTERISTICS
F.HDLB:'L'.BLKW 2       ; FILE HEADER LOGICAL BLOCK NUMBER
```

F11DF$ (Cont.)

```
                        ; BEGINNING OF STATISTICS BLOCK
F.LBN:'L'.BLKW  2       ; LBN OF VIRTUAL BLOCK 1 IF CONTIGUOUS
                        ; 0 IF NON CONTIGUOUS
F.SIZE:'L'.BLKW 2       ; SIZE OF FILE IN BLOCKS
F.NACS:'L'.BLKB 1       ; NO. OF ACCESSES
F.NLCK:'L'.BLKB 1       ; NO. OF LOCKS
S.STBK='B'.-F.LBN       ; SIZE OF STATISTICS BLOCK

F.STAT:'L'              ; FCB STATUS WORD
F.NWAC:'L'.BLKB 1       ; NUMBER OF WRITE ACCESSORS
        .BLKB   1       ; STATUS BITS FOR FCB CONSISTING OF
        FC.WAC='B' 100000 ; SET IF FILE ACCESSED FOR WRITE
        FC.DIR='B' 40000  ; SET IF FCB IS IN DIRECTORY LRU
        FC.CEF='B' 20000  ; SET IF DIRECTORY EOF NEEDS UPDATING
        FC.FCO='B' 10000  ; SET IF TRYING TO FORCE DIRECTORY CONTIG
F.DREF:'L'.BLKW 1       ; DIRECTORY EOF BLOCK NUMBER
F.DRNM:'L'.BLKW 1       ; 1ST WORD OF DIRECTORY NAME
F.FEXT:'L'.BLKW 1       ; POINTER TO EXTENSION FCB
F.FVBN:'L'.BLKW 2       ; STARTING VBN OF THIS FILE SEGMENT
F.LKL:'L'.BLKW  1       ; POINTER TO LOCKED BLOCK LIST FOR FILE
F.WIN:'L'.BLKW  1       ; WINDOW BLOCK LIST FOR THIS FILE
F.LGTH:'L'              ; SIZE IN BYTES OF FCB


;
; WINDOW
;

        .ASECT
.=0

W.ACT:'L'               ; NUMBER OF ACTIVE MAPPING POINTERS
                        ;  WHEN NO SECONDARY POOL
W.BLKS:'L'              ; BLOCK SIZE OF SECONDARY POOL SEGMENT
                        ;  WHEN SECONDARY POOL
W.CTL:'L'.BLKW  1       ; LOW BYTE = # OF MAP ENTRIES ACTIVE
                        ; HIGH BYTE CONSISTS OF CONTROL BITS
        WI.RDV='B' 400    ; READ VIRTUAL BLOCK ALLOWED IF SET
        WI.WRV='B' 1000   ; WRITE VIRTUAL BLOCK ALLOWED IF SET
        WI.EXT='B' 2000   ; EXTEND ALLOWED IF SET
        WI.LCK='B' 4000   ; SET IF LOCKED AGAINST SHARED ACCESS
        WI.DLK='B' 10000  ; SET IF DEACCESS LOCK ENABLED
        WI.PND='B' 20000  ; WINDOW TURN PENDING BIT
        WI.EXL='B' 40000  ; SET IF MANUAL UNLOCK DESIRED
        WI.WCK='B' 100000 ; DATA CHECK ALL WRITES TO FILE
W.IOC:'L'.BLKB  1       ; COUNT OF I/O THROUGH THIS WINDOW
        .BLKB   1       ; RESERVED
W.FCB:'L'.BLKW  1       ; FILE CONTROL BLOCK ADDRESS
W.TCB:'L'.BLKW  1       ; TCB ADDRESS OF ACCESSOR
W.UCB:'L'.BLKW  1       ; ORIGINAL UCB ADDRESS OF DEVICE
W.LKL:'L'.BLKW  1       ; POINTER TO LIST OF USERS LOCKED BLOCKS
W.WIN:'L'.BLKW  1       ; WINDOW BLOCK LIST LINK WORD

        .IF     NB,SYSDEF  ; IF SYSDEF SPECIFIED IN CALL

        .IF     NDF,P$$WND ; IF SECONDARY POOL WINDOWS NOT ALLOWED
```

F11DF$ (Cont.)

```
;
; NON-SECONDARY POOL WINDOW BLOCK
;
;       IF SECONDARY POOL WINDOWS ARE NOT ENABLED, THE WINDOW BLOCK
;       CONTAINS THE CONTROL INFORMATION AND RETRIEVAL POINTERS.
;

W.VBN:'L'.BLKB  1       ; HIGH BYTE OF 1ST VBN MAPPED BY WINDOW
W.MAP:'L'               ; DEFINE LABEL WITH ODD ADDRESS TO CATCH BAD REFS
W.WISZ:'L'.BLKB 1       ; SIZE IN RTRV PTRS OF WINDOW (7 BITS)
        .BLKW   1       ; LOW ORDER WORD OF 1ST VBN MAPPED
W.RTRV:'L'              ; OFFSET TO 1ST RETRIEVAL POINTER IN WINDOW

W.SLEN='B'-4            ; DUMMY DEFINITION TO PREVENT INCORRECT REFERENCE
                        ; (-4 WHEN ROUNDED "UP" IS A VERY LARGE BLOCK)

        .IFF            ; IF WINDOWS IN SECONDARY POOL

;
;SECONDARY POOL WINDOW CONTROL AND MAPPING BLOCK
;
;       IF SECONDARY POOL WINDOW BLOCKS ARE ENABLED, LUTN2 POINTS
;       TO A CONTROL BLOCK IN SYSTEM POOL WHICH CONTAINS THE
;       FOLLOWING CONTROL FIELDS AND THE MAPPING INFORMATION
;       FOR THE SECONDARY POOL WINDOW.
;

W.MAP:'L'.BLKW  1       ; ADDR TO THE MAPPING PTRS IN SECONDARY POOL
W.SLEN:'L'              ; LENGTH OF PRIMARY POOL STUB

;
; SECONDARY POOL WINDOW
;       IF SECONDARY POOL WINDOW BLOCKS ARE ENABLED, THE RETRIEVAL
;       POINTERS ARE MAINTAINED IN SECONDARY POOL IN THE FOLLOWING
;       FORMAT.
;

.=0

        ASSUME  W.CTL,0

        .BLKB   1       ; NUMBER OF ACTIVE MAPPING POINTERS
W.USE:'L'.BLKB  1       ; STATUS OF BLOCK
W.VBN:'L'.BLKB  1       ; HIGH BYTE OF 1ST VBN MAPPED BY WINDOW
W.WISZ:'L'.BLKB 1       ; SIZE IN RTRV PTRS OF WINDOW (7 BITS)
        .BLKW   1       ; LOW ORDER WORD OF 1ST VBN MAPPED
W.RTRV:'L'              ; OFFSET TO 1ST RETRIEVAL POINTER IN WINDOW

        .ENDC           ; END SECONDARY POOL WINDOW CONDITIONAL

        .ENDC           ; END SYSDEF CONDITIONAL

;
; LOCKED BLOCK LIST NODE
;

        .ASECT
.=0
```

F11DF$ (Cont.)

```
L.LNK:'L'.BLKW  1       ; LINK TO NEXT NODE IN LIST
L.WI1:'L'.BLKW  1       ; POINTER TO WINDOW FOR FIRST ENTRY
L.VB1:'L'.BLKB  1       ; HIGH ORDER VBN BYTE
L.CNT:'L'.BLKB  1       ; COUNT FOR ENTRY
        .BLKW   1       ; LOW ORDER VBN
L.LKSZ:'L'

; END OF DEFINITIONS

        .PSECT
        .MACRO  F11DF$  X,Y,Z
        .ENDM   F11DF$
        .ENDM   F11DF$
```

HDRDF$

```
        .MACRO  HDRDF$,L,B

;+
; TASK HEADER OFFSET DEFINITIONS
;-

        .ASECT
.=0
H.CSP:'L'.BLKW  1               ;CURRENT STACK POINTER
H.HDLN:'L'.BLKW 1               ;HEADER LENGTH IN BYTES
H.SMAP:'L'.BLKB 1               ;SUPERVISOR D SPACE OVERMAP MASK
H.DMAP:'L'.BLKB 1               ;USER D SPACE OVERMAP MASK
H.FMAP:'L'.BLKW 1               ;POINTER TO FAST MAP SECTION OF HDR
H.CUIC:'L'.BLKW 1               ;CURRENT TASK UIC
H.DUIC:'L'.BLKW 1               ;DEFAULT TASK UIC
H.IPS:'L'.BLKW  1               ;INITIAL PROCESSOR STATUS WORD (PS)
H.IPC:'L'.BLKW  1               ;INITIAL PROGRAM COUNTER (PC)
H.ISP:'L'.BLKW  1               ;INITIAL STACK POINTER (SP)
H.ODVA:'L'.BLKW 1               ;ODT SST VECTOR ADDRESS
H.ODVL:'L'.BLKW 1               ;ODT SST VECTOR LENGTH
H.TKVA:'L'.BLKW 1               ;TASK SST VECTOR ADDRESS
H.TKVL:'L'.BLKW 1               ;TASK SST VECTOR LENGTH
H.PFVA:'L'.BLKW 1               ;POWER FAIL AST CONTROL BLOCK ADDRESS
H.FPVA:'L'.BLKW 1               ;FLOATING POINT AST CONTROL BLOCK ADDRESS
H.RCVA:'L'.BLKW 1               ;RECIEVE AST CONTROL BLOCK ADDRESS
H.EFSV:'L'.BLKW 1               ;EVENT FLAG ADDRESS SAVE ADDRESS
H.FPSA:'L'.BLKW 1               ;POINTER TO FLOATING POINT/EAE SAVE AREA
H.WND:'L'.BLKW  1               ;POINTER TO NUMBER OF WINDOW BLOCKS
H.DSW:'L'.BLKW  1               ;TASK DIRECTIVE STATUS WORD
H.FCS:'L'.BLKW  1               ;FCS IMPURE POINTER
H.FORT:'L'.BLKW 1               ;FORTRAN IMPURE POINTER
H.OVLY:'L'.BLKW 1               ;OVERLAY IMPURE POINTER
H.VEXT:'L'.BLKW 1               ;WORK AREA EXTENSION VECTOR POINTER
H.SPRI:'L'.BLKB 1               ;PRIORITY DIFFERENCE FOR SWAPPING
H.NML:'L'.BLKB  1               ;NETWORK MAILBOX LUN
H.RRVA:'L'.BLKW 1               ;RECEIVE BY REFERENCE AST CONTROL BLOCK
                                ; ADDRESS
H.X25:'L'.BLKB  1               ;FOR USE BY X25 SOFTWARE
        .BLKB   1               ;5 RESERVED BYTES
        .BLKW   2               ;
H.GARD:'L'.BLKW 1               ;POINTER TO HEADER GUARD WORD
H.NLUN:'L'.BLKW 1               ;NUMBER OF LUN'S
H.LUN:'L'.BLKW  2               ;START OF LOGICAL UNIT TABLE

;+
; LENGTH OF FLOATING POINT SAVE AREA
;-

H.FPSL='B'25.*2                 ;

;+
; WINDOW BLOCK OFFSETS
;-

.=0
W.BPCB:'L'.BLKW 1               ;PARTITION CONTROL BLOCK ADDRESS
W.BLVR:'L'.BLKW 1               ;LOW VIRTUAL ADDRESS LIMIT
W.BHVR:'L'.BLKW 1               ;HIGH VIRTUAL ADDRESS LIMIT
W.BATT:'L'.BLKW 1               ;ADDRESS OF ATTACHMENT DESCRIPTOR
W.BSIZ:'L'.BLKW 1               ;SIZE OF WINDOW IN 32W BLOCKS
```

HDRDF$ (Cont.)

```
W.BOFF:'L'.BLKW 1               ;PHYSICAL MEMORY OFFSET IN 32W BLOCKS
W.BFPD:'L'.BLKB 1               ;FIRST PDR ADDRESS
W.BNPD:'L'.BLKB 1               ;NUMBER OF PDR'S TO MAP
W.BLPD:'L'.BLKW 1               ;CONTENTS OF LAST PDR
W.BLGH:'L'                      ;LENGTH OF WINDOW DESCRIPTOR

;
; BIT DEFINITION FOR W.BLPD
;

WB.NBP='B'20                    ;CACHE BYPASS NOT DESIRED FOR THIS WINDOW
WB.BPS='B'40                    ;ALWAYS BYPASS THE CACHE FOR THIS WINDOW


        .PSECT

        .MACRO  HDRDF$  X,Y
        .ENDM
        .ENDM
```

```
        .MACRO HWDDF$,L,B

;+
; MACROS FOR DEFINING MAPPING REGISTER DEFINITIONS
;-

        .MACRO  CRESET  NAM,ADDR
$$$=0
        .REPT   8.
        CRENAM  NAM,ADDR+<$$$2>,$$$
$$$=$$$+1
        .ENDR
        .ENDM

        .MACRO  CRENAM  NAM,ADDR,N
'NAM''N'==ADDR
        .ENDM

;+
; HARDWARE REGISTER ADDRESSES AND STATUS CODES
;-

MPCSR='B'177746                 ;ADDRESS OF PDP-11/70 MEMORY PARITY
                                ; REGISTER
MPAR='B'172100                  ;ADDRESS OF FIRST MEMORY PARITY REGISTER
PIRQ='B'177772                  ;PROGRAMMED INTERRUPT REQUEST REGISTER
PR0='B'0                        ;PROCESSOR PRIORITY 0
PR1='B'40                       ;PROCESSOR PRIORITY 1
PR4='B'200                      ;PROCESSOR PRIORITY 4
PR5='B'240                      ;PROCESSOR PRIORITY 5
PR6='B'300                      ;PROCESSOR PRIORITY 6
PR7='B'340                      ;PROCESSOR PRIORITY 7
PS='B'177776                    ;PROCESSOR STATUS WORD
SWR='B'177570                   ;CONSOLE SWITCH AND DISPLAY REGISTER
TPS='B'177564                   ;CONSOLE TERMINAL PRINTER STATUS REGISTER

;+
; EXTENDED ARITHMETIC ELEMENT REGISTERS
;-


        .IF DF  E$$EAE

AC='B'177302                    ;ACCUMULATOR
MQ='B'177304                    ;MULTIPLIER-QUOTIENT
SC='B'177310                    ;SHIFT COUNT

        .ENDC

;+
; MEMORY MANAGEMENT HARDWARE REGISTERS AND STATUS CODES
;-

        .IF NB  B

        CRESET  KINAR,172340    ;KERNEL I PAR'S
        CRESET  KINDR,172300    ;KERNEL I PDR'S
        CRESET  KDSAR,172360    ;KERNEL D PAR'S
        CRESET  KDSDR,172320    ;KERNEL D PDR'S
```

HWDDF$ (Cont.)

```
        CRESET  SISAR,172240    ;SUPERVISOR I PAR'S
        CRESET  SISDR,172200    ;SUPERVISOR I PDR'S
        CRESET  SDSAR,172260    ;SUPERVISOR D PAR'S
        CRESET  SDSDR,172220    ;SUPERVISOR D PDR'S
        CRESET  UINAR,177640    ;USER I PAR'S
        CRESET  UINDR,177600    ;USER I PDR'S
        CRESET  UDSAR,177660    ;USER D PAR'S
        CRESET  UDSDR,177620    ;USER D PDR'S

        .ENDC

        .IF NB  SYSDEF

        .IF DF  K$$DAS

        CRESET  KISAR,172360    ;KERNEL D PAR'S
        CRESET  KISDR,172320    ;KERNEL D PDR'S

        .IFF

        CRESET  KISAR,172340    ;KERNEL I PAR'S
        CRESET  KISDR,172300    ;KERNEL I PDR'S

        .ENDC

        .IF DF  U$$DAS

        CRESET  UISAR,177660    ;USER D PAR'S
        CRESET  UISDR,177620    ;USER D PDR'S

        .IFF    ; DF U$$DAS

        CRESET  UISAR,177640    ;USER I PAR'S
        CRESET  UISDR,177600    ;USER I PDR'S

        .ENDC   ; DF U$$DAS
        .ENDC

UBMPR='B'170200                 ;UNIBUS MAPPING REGISTER 0
CMODE='B'140000                 ;CURRENT MODE FIELD OF PS WORD
PMODE='B'30000                  ;PREVIOUS MODE FIELD OF PS WORD
CSMODE='B'40000                 ;CURRENT MODE = SUPERVISOR PS WORD BITS
PSMODE='B'10000                 ;PREVIOUS MODE = SUPERVISOR PS WORD BITS
SR0='B'177572                   ;SEGMENT STATUS REGISTER 0
SR3='B'172516                   ;SEGMENT STATUS REGISTER 3
CPUERR='B'177766                ;CPU ERROR REGISTER
MEMERR='B'177744                ;MEMORY SYSTEM ERROR REGISTER
MEMCTL='B'177746                ;MEMORY CONTROL REGISTER

;+
; DEFINE THE LOCATIONS USED IN THE NON-VOLATILE RAM (NVR)
; FOR XT SYSTEMS
;-

N.KEY='B'173054                 ;NUMBER OF KEYS PRESSED
N.UPT='B'173064                 ;UPTIME IN MINUTES
N.DZA='B'173074                 ;NUMBER OF I/OS DONE ON THE DZ
N.DWA='B'173104                 ;NUMBER OF I/OS DONE ON THE DW
N.DAY='B'173114                 ;DATE THAT THE NVR WAS LAST INITIALIZED
N.MON='B'173116                 ;...
N.YEA='B'173120                 ;...
```

HWDDF$ (Cont.)

```
;+
; FEATURE SYMBOL DEFINITIONS
;-

FE.EXT='B'1                     ;22-BIT EXTENDED MEMORY SUPPORT
FE.MUP='B'2                     ;MULTI-USER PROTECTION SUPPORT
FE.EXV='B'4                     ;EXECUTIVE IS SUPPORTED TO 20K
FE.DRV='B'10                    ;LOADABLE DRIVER SUPPORT
FE.PLA='B'20                    ;PLAS SUPPORT
FE.CAL='B'40                    ;DYNAMIC CHECKPOINT SPACE ALLOCATION
FE.PKT='B'100                   ;PREALLOCATION OF I/O PACKETS
FE.EXP='B'200                   ;EXTEND TASK DIRECTIVE SUPPORTED
FE.LSI='B'400                   ;PROCESSOR IS AN LSI-11
FE.OFF='B'1000                  ;PARENT/OFFSPRING TASKING SUPPORTED
FE.FDT='B'2000                  ;FULL DUPLEX TERMINAL DRIVER SUPPORTED
FE.X25='B'4000                  ;X.25 CEX IS LOADED
FE.DYM='B'10000                 ;DYNAMIC MEMORY ALLOCATION SUPPORTED
FE.CEX='B'20000                 ;COM EXEC IS LOADED
FE.MXT='B'40000                 ;MCR EXIT AFTER EACH COMMAND MODE
FE.NLG='B'100000                ;LOGINS DISABLED - MULTI-USER SUPPORT

;+
; FEATURE MASK DEFINITIONS (SECOND WORD)
;-

F2.DAS='B'1                     ;KERNEL DATA SPACE SUPPORTED
F2.LIB='B'2                     ;SUPERVISOR MODE LIBRARIES SUPPORTED
F2.MP='B'4                      ;SYSTEM SUPPORTS MULTIPROCESSING
F2.EVT='B'10                    ;SYSTEM SUPPORTS EVENT TRACE FEATURE
F2.ACN='B'20                    ;SYSTEM SUPPORTS CPU ACCOUNTING
F2.SDW='B'40                    ;SYSTEM SUPPORTS SHADOW RECORDING
F2.POL='B'100                   ;SYSTEM SUPPORTS SECONDARY POOLS
F2.WND='B'200                   ;SYSTEM SUPPORTS SECONDARY POOL FILE WINDOWS
F2.DPR='B'400                   ;SYSTEM HAS A SEPARATE DIRECTIVE PARTITION
F2.IRR='B'1000                  ;INSTALL, RUN, AND REMOVE SUPPORT
F2.GGF='B'2000                  ;GROUP GLOBAL EVENT FLAG SUPPORT
F2.RAS='B'4000                  ;RECEIVE/SEND DATA PACKET SUPPORT
F2.AHR='B'10000                 ;ALT. HEADER REFRESH AREA SUPPORT
F2.RBN='B'20000                 ;ROUND ROBIN SCHEDULING SUPPORT
F2.SWP='B'40000                 ;EXECUTIVE LEVEL DISK SWAPPING SUPPORT
F2.STP='B'100000                ;EVENT FLAG MASK IS IN THE TCB(1=YES)

;+
; THIRD FEATURE MASK SYMBOL DEFINITIONS
;-

F3.CRA='B'1                     ;SYSTEM SPONTANEOUSLY CRASHED (1=YES)
F3.XCR='B'2                     ;SYSTEM CRASHED FROM XDT (1=YES)
F3.EIS='B'4                     ;SYSTEM REQUIRES EXTENDED INSTRUCTION SET
F3.STM='B'10                    ;SYSTEM HAS SET SYSTEM TIME DIRECTIVE
F3.UDS='B'20                    ;SYSTEM SUPPORTS USER DATA SPACE
F3.PRO='B'40                    ;SYSTEM SUPPORTS SEC. POOL PROTO TCBS
F3.XHR='B'100                   ;SYSTEM SUPPORTS EXTERNAL TASK HEADERS
F3.AST='B'200                   ;SYSTEM HAS AST SUPPORT
F3.11S='B'400                   ;RSX-11S SYSTEM
F3.CLI='B'1000                  ;MULTIPLE CLI SUPPORT
F3.TCM='B'2000                  ;SYSTEM HAS SEPARATE TERMINAL DRIVER POOL
F3.PMN='B'4000                  ;SYSTEM SUPPORTS POOL MONITORING
F3.WAT='B'10000                 ;SYSTEM HAS WATCHDOG TIMER SUPPORT
F3.RLK='B'20000                 ;SYSTEM SUPPORTS RMS RECORD LOCKING
F3.SHF='B'40000                 ;SYSTEM SUPPORTS SHUFFLER TASK
```

HWDDF$ (Cont.)

```
;+
; FOURTH FEATURE MASK BITS
;-

F4.CXD='B'1                     ;COMM EXEC IS DEALLOCATED (NON-I/D ONLY)
F4.XT='B'2                      ;SYSTEM IS AN XT SYSTEM (1=YES)
F4.ERL='B'4                     ;SYSTEM SUPPORTS ERROR LOGGING (1=YES)
F4.PTY='B'10                    ;SYSTEM SUPPORTS PARITY MEMORY (1=YES)
F4.DVN='B'20                    ;SYSTEM SUPPORTS DECIMAL VERSIONS (1=YES)
F4.LCD='B'40                    ;SYSTEM SUPPORTS LOADABLE CRASH (1=YES)
F4.NIM='B'100                   ;SYSTEM SUPPORTS DELETED TASK IMAGES (1=YES)
F4.CHE='B'200                   ;SYSTEM SUPPORTS DISK DATA CACHING (1=YES)
F4.LOG='B'400                   ;SYSTEM SUPPORTS LOGICAL NAMES (1=YES)
F4.NAM='B'1000                  ;SYSTEM SUPPORTS NAMED DIRECTORIES (1=YES)
F4.FMP='B'2000                  ;SYSTEM SUPPORTS FAST MAP DIRECTIVE
F4.DCL='B'4000                  ;DCL IS DEFAULT CLI (1=YES)
F4.DDS='B'10000                 ;NAMED DIRECTORY MODE IS THE DEFAULT (1=YES)
F4.ACD='B'20000                 ;SYSTEM SUPPORTS ACD'S

;+
; HARDWARE FEATURE MASK BIT DEFINITIONS
;
;       HF.CIS,HF.FPP DEFINED AS SIGN BITS FOR RUN TIME SPEED
;-

HF.UBM='B'1                     ;PROCESSOR HAS A UNIBUS MAP (1=YES)
HF.EIS='B'2                     ;PROCESSOR HAS EXTENDED INSTRUCION SET
HF.QB='B'4                      ;SYSTEM HAS A QBUS (1=YES)
HF.DSP='B'10                    ;HARDWARE SUPPORTS DATA SPACE
HF.CIS='B'200                   ;PROCESSOR SUPPORTS COMMERCIAL INSTRUCTION SET
HF.FPP='B'100000                ;(1=PROC. HAS NO FLOATING POINT UNIT)

;+
; SECOND HARDWARE FEATURE MASK BIT DEFINITIONS
; THIS WORD IS RESERVED FOR XT HARDWARE FEATURES
;-

H2.NVR='B'1                     ;XT NON-VOLATILE RAM PRESENT (1=YES)
H2.INV='B'2                     ;NON-VOLATILE RAM IS INVALID (1=YES)
H2.CLK='B'4                     ;XT CLOCK IS PRESENT (1=YES)
H2.ITF='B'10                    ;INVALID TIME FORMAT IN NON-VOLATILE RAM
                                ; (1=YES)
H2.PRO='B'20                    ;RUNNING ON PRO/3XX HARDWARE
H2.BRG='B'100000                ;XT BRIDGE MODULE PRESENT (1=YES)

;+
; SYSGEN FEATURE SELECTIONS MASK. THIS IS INTENDED TO RECORD IN A
; BIT MASK THE CHOICES MADE AT SYSGEN TIME. FEATURES ARE LISTED HERE FOR
; OUR INFORMATIONAL PURPOSES ONLY. THEY CANNOT BE TESTED LIKE BITS IN THE
; FEATURE MASK SINCE THIS ONLY EXISTS IN THE RSX11M.STB FILE. NO BITS IN
; MEMORY ; ARE USED. THEY ARE ONLY INTENDED TO BE PRINTED FROM THE STB FILE
; BY CDA.
;-

SF.STD='B'1                     ;STANDARD EXEC SELECTED
SF.PGN='B'2                     ;SYSTEM WAS PRE-GENERATED
                                ; (EX. RL02/RC25 SYSTEM)
```

HWDDF$ (Cont.)

```
;+
; MULTIPROCESSOR STATUS TABLE DEFINITIONS (TEMPORARY)
;-

MP.CRH='B'100000                ;CRASH PROCESSOR IMMEDIATELY
MP.PWF='B'40000                 ;POWERFAIL ON ONE CPU
MP.RSM='B'20000                 ;RESET INTERRUPT MASKS
MP.NOP='B'10000                 ;NOP FUNCTION FOR TRANSMISSION CHECK
MP.STP='B'4                     ;STOP PROCESSOR IN ORDERLY FASHION
MP.INT='B'7777                  ;BIC MASK FOR INTERRUPT LVL FUNCTIONS

        .MACRO  HWDDF$  X,Y,Z
        .ENDM
        .ENDM
```

ITBDF$

```
        .MACRO  ITBDF$  L,B

;
; INTERRUPT TRANSFER BLOCK (ITB) OFFSET DEFINITIONS
;

        .IF DF  A$$TRP

        .MCALL  PKTDF$
        PKTDF$                      ; DEFINE AST BLOCK OFFSETS

        .ENDC

        .ASECT
        .=0
X.LNK:'L' .BLKW 1                   ; LINK WORD FOR ITB LIST STARTING IN TCB
X.JSR:'L' JSR   R5,@ 0              ; CALL $INTSC
X.PSW:'L' .BLKB 1                   ; LOW BYTE OF PSW FOR ISR
          .BLKB 1                   ; UNUSED
X.ISR:'L' .BLKW 1                   ; ISR ENTRY POINT (APR5 MAPPING)
X.FORK:'L'                          ; FORK BLOCK
          .BLKW   1                 ; THREAD WORD
          .BLKW   1                 ; FORK PC
          .BLKW   1                 ; SAVED R5
          .BLKW   1                 ; SAVED R4

        .IF DF  M$$MGE

X.REL:'L' .BLKW 1                   ; RELOCATION BASE FOR APR5

          .ENDC

X.DSI:'L' .BLKW 1                   ; ADDRESS OF DIS.INT. ROUTINE
X.TCB:'L' .BLKW 1                   ; TCB ADDRESS OF OWNING TASK

        .IF NB  SYSDEF

        .IF DF  A$$TRP

          .BLKW   1                 ; A.DQSR FOR AST BLOCK
X.AST:'L' .BLKB A.PRM               ; AST BLOCK

        .ENDC
X.VEC:'L' .BLKW 1                   ; VECTOR ADDRESS (IF AST SUPPORT,
                                    ; THIS IS FIRST AND ONLY AST PARAMETER)
X.VPC:'L' .BLKW 1                   ; SAVED VECTOR PC
X.LEN:'L'                           ; LENGTH IN BYTES OF ITB

        .ENDC

        .PSECT

        .MACRO  ITBDF$  X,Y,Z

        .ENDM   ITBDF$
```

KRBDF$

```
        ;.MACRO       KRBDF$

        ;
        ; CONTROLLER REQUEST BLOCK (KRB)
        ;
        ; THE CONTROLLER REQUEST BLOCK DEFINES THE ENVIRONMENT OF A DEVICE
        ; CONTROLLER.  EXACTLY ONE KRB EXISTS FOR EVERY DEVICE CONTROLLER
        ; IN AN RSX-11M+ SYSTEM.  THE KRB CONTAINS CERTAIN DEVICE STATUS
        ; INCLUDING THE CSR AND VECTOR ADDRESS FOR THE CONTROLLER.
        ;
                .ASECT
        .=177770
177770  K.PRM:  .BLKW   1       ;DEVICE DEPENDANT PARAMETER WORD
177772  K.PRI:  .BLKB   1       ;CONTROLLER PRIORITY
177773  K.VCT:  .BLKB   1       ;INTERRUPT VECTOR ADDRESS
177774  K.CON:  .BLKB   1       ;CONTROLLER INDEX WITHIN THE SYSTEM
177775  K.IOC:  .BLKB   1       ;CONTROLLER I/O COUNT
177776  K.STS:  .BLKW   1       ;CONTROLLER STATUS
000000  K.CSR:  .BLKW   1       ;ADDRESS OF CONTROL STATUS REGISTER
        ;
        ; NOTE: K.CSR MUST BE THE ZERO OFFSET!
        ;
000002  K.OFF:  .BLKW   1       ;OFFSET TO UCB/UMR/RHBAE TABLE
000004  K.HPU:  .BLKB   1       ;HIGHEST PHYSICAL UNIT NUMBER
000005          .BLKB   1       ;UNUSED BYTE
000006  K.OWN:  .BLKW   1       ;OWNER OF CONTROLLER
000010  K.CRQ:  .BLKW   2       ;CONTROLLER REQUEST QUEUE
000014  K.URM:  .BLKW   1       ;CONTROLLER UNIBUS RUN MASK
000016  K.FRK:  .BLKW   1       ;POSSIBLE KRB FORK BLOCK

        ;
        ; OFFSETS FOR THE KRB EXTENSION REACHED BY ADDING (K.OFF) TO
        ; THE STARTING ADDRESS OF THE KRB.
        ;


        ;
        ; DEFINE OFFSETS IN SCB/KRB FOR DISK MSCP CONTROLLERS
        ;
        .=-20.
177754  KE.UMH: .BLKW   2       ;LIST HEAD FOR UMR WAITING ASSIGNMENT
                                    ;BLK(S)
177760  KE.UMC: .BLKW   1       ;COUNT OF AVAILABLE UMR WAITING
                                ;ASSIGNMENT BLOCK(S)
        .=177776
177776  KE.RHB: .BLKW   1       ;OFFSET TO RHBAE REGISTER (IF ANY)
        ;
        ; WHEN ONE ADDS (K.OFF) TO THE KRB ADDRESS, IT YIELDS AN
        ;ADDRESS WHICH POINTS TO HERE.
        ;
000000  KE.UCB: .BLKW   1       ;OFFSET TO UCB TABLE (IF KS.UCB SET)

                .PSECT
```

KRBDF$ (Cont.)

```
        ; CONTROLLER REQUEST BLOCK (KRB) STATUS BIT DEFINITIONS
        ;
        KS.OFL=1                ;CONTROLLER OFFLINE (1=YES)
        KS.MOF=2                ;CONTROLLER MARKED FOR OFFLINE (1=YES)
        KS.UOP=4                ;SUPPORTS OVERLAPPED OPERATION (1=YES)
        KS.MBC=10               ;DEVICE IS MASSBUS CONTROLLER (1=YES)
        KS.SDX=20               ;SEEKS ALLOWED DURING DATA XFERS (1=YES)
        KS.POE=40               ;PARALLEL OPERATION ENABLED (1=YES)
        KS.UCB=100              ;UCB TABLE PRESENT (1=YES)
        KS.DIP=200              ;DATA TRANSFER IN PROGRESS (1=YES)
        KS.PDF=400              ;PRIVILEGED DIAGNOSTIC FUNCTIONS ONLY
                                ;BLOCK(S)(1=YES)
        KS.EXT=1000             ;EXTENDED 22-BIT UNIBUS CONTROLLER
                                ;BLOCK(S);(1=YES)
        KS.SLO=2000             ;CONTROLLER IS SLOW COMING ONLINE
                                ;BLOCK(S)(1=YES)


        ;
        ; DEFINE THE CONTIGUOUS SCB OFFSETS
        ;
                .ASECT
        .=177762
177762  S.PRI:   .BLKB   1       ;CONTROLLER PRIORITY
177763  S.VCT:   .BLKB   1       ;INTERRUPT VECTOR ADDRESS
177764  S.CON:   .BLKB   1       ;CONTROLLER INDEX
177765           .BLKB   1
177766           .BLKW   1
177770  S.CSR:   .BLKW   1       ;CONTROL AND STATUS REGISTER
177772           .BLKW   1
177774           .BLKB   1
177775           .BLKB   1
177776  S.OWN:   .BLKW   1       ;DISTRIBUTED CNTBL
        ;
        ; SUBCONTROLLER REQUEST BLOCK (KRB1)
        ;
        ; THE SUBCONTROLLER REQUEST BLOCK DEFINES THE ENVIRONMENT OF A
        ; DEVICE SUBCONTROLLER. EXACTLY ONE KRB1 EXISTS FOR EVERY DEVICE
        ; SUBCONTROLLER IN AN RSX-11M+ SYSTEM.
        ;
                .ASECT
        .=-4
177774  K1.CON: .BLKB   1       ;SUBCONTROLLER INDEX WITHIN THE SYSTEM
177775           .BLKB   1       ;UNUSED BYTE
177776  K1.STS: .BLKW   1       ;SUBCONTROLLER STATUS
000000  K1.MAS: .BLKW   1       ;UCB ADDRESS OF THE MASTER UNIT
        ;
        ; NOTE: K1.MAS MUST BE THE ZERO OFFSET
        ;
000002  K1.OWN: .BLKW   1       ;OWNER OF SUBCONTROLLER
000004  K1.CRQ: .BLKW   2       ;SUBCONTROLLER REQUEST QUEUE
000010  K1.UCB:                 ;START OF THE UCB TABLE (IF ANY)

                .PSECT
```

LCBDF$

```
        ;.MACRO       LCBDF$

        ;
        ; LOGICAL ASSIGNMENT CONTROL BLOCK
        ;
        ; THE LOGICAL ASSIGNMENT CONTROL BLOCK (LCB) IS USED TO
        ; ASSOCIATE A LOGICAL NAME WITH A PHYSICAL DEVICE UNIT.
        ; LOGICAL CONTROL BLOCKS  ARE LINKED TO FORM THE LOGICAL
        ; ASSIGNMENTS OF A SYSTEM. LOGICAL ASSIGNMENTS CAN BE MADE
        ; ON A SYSTEM-WIDE OR LOCAL (TERMINAL) BASIS.
        ;
                .ASECT
        .=0
000000  L.LNK:  .BLKW   1       ;LINK TO NEXT LCB
000002  L.NAM:  .BLKW   1       ;LOGICAL NAME OF DEVICE
000004  L.UNIT: .BLKB   1       ;LOGICAL UNIT NUMBER
000005  L.TYPE: .BLKB   1       ;TYPE OF ENTRY (0=SYSTEM WIDE)
000006  L.UCB:  .BLKW   1       ;TI UCB ADDRESS
000010  L.ASG:  .BLKW   1       ;ASSIGNMENT UCB ADDRESS
000012  L.LGTH=.-L.LNK          ;LENGTH OF LCB

                .PSECT
```

MTADF$

```
        .MACRO  MTADF$,L,B
        .ASECT
;
; ANSI MAGTAPE SPECIFIC DATA STRUCTURES
;
; VOLUME SET CONTROL BLOCK OFFSET DEFINITIONS (VSCB)
;
; VOLUME SET AND PROCESS CONTROL SECTION
;
.=0
V.TCNT:'L' .BLKW 1              ;TRANSACTION COUNT
V.TYPE:'L' .BLKB 1              ;VOLUME TYPE DESCRIPTOR
V.VCHA:'L' .BLKB 1              ;VOLUME CHARACTERISTICS
V.LABL:'L' .BLKB 12.            ;FILE SET ID (FIRST SIX BYTES)
V.NXT:'L' .BLKW 1               ;PTR TO NEXT VSCB NODE
V.MVL:'L' .BLKW 1               ;PTR TO MOUNTED VOL LIST
V.UVL:'L' .BLKW 1               ;PTR TO UNMOUNTED VOL LIST
V.ATL:'L' .BLKW 1               ;ATL ADDR OF ACCESSING TASK
                                ; TCB IN RSX11M
V.UCB:'L' .BLKW 1               ;ADDR OF CURRENT UCB OR PUD
V.RVOL:'L' .BLKB 1              ;CURRENT RELATIVE VOL #
V.MOU:'L' .BLKB 1               ;MOUNT MODE BYTE
V.TCHR:'L' .BLKW 1              ;UINT CHAR. FOR ALL UNITS USED FOR VOL SET
V.SEQN:'L' .BLKW 1              ;CURRENT FILE SEQUENCE #
V.SECN:'L' .BLKW 1              ;CURRENT FILE SECTION #
V.TPOS:'L' .BLKB 1              ;POSITION OF TAPE IN TM'S TO NXT HDR1
V.PSTA:'L' .BLKB 1              ;PROCESS STATUS BYTE
V.TIMO:'L' .BLKW 1              ;BLOCKED PROCESS TIMEOUT COUNTER
V.STAT:'L' .BLKW 3              ; STATUS WORDS USED BY COMMAND
                                ;EXECUTION MODULES
V.TRTB:'L' .BLKB 1              ;TRANSLATION CONTROL BYTE
V.EFTV:'L' .BLKB 1              ;FOR MAG TO RETURN IE.EOF, EOT, EOV
;
; LABEL DATA SECTION
;
V.BLKL:'L' .BLKW 1              ;BLOCK LENGTH
V.RECL:'L' .BLKW 1              ;RECORD LENGTH
V.FNAM:'L' .BLKW 3              ;FILE NAME
V.FTYP:'L' .BLKW 1              ;FILE TYPE
V.FVER:'L' .BLKW 1              ;FILE VERSION #
V.CDAT:'L' .BLKW 2              ;CREATION DATE
V.EDAT:'L' .BLKW 2              ;EXPRIATION DATE
V.BLKC:'L' .BLKW 2              ;BLOCK COUNT FOR FILE SECTION
V.RTYP:'L' .BLKB 1              ;RECORD TYPE
V.FATT:'L' .BLKB 1              ;FILE ATTRIBUTES FOR CARRIAGE CONTROL
           .BLKB 30.            ;REMAINDER OF FILE ATTRIBUTES

; NULL WINDOW SECTION
;
V.WIND:'L' .BLKW  4.            ; NULL WINDOW
V.MST2:'L' .BLKW  1             ;MAGTAPE STATUS BITS
V.FABY:'L' .BLKB  1             ;FILE ACCESSIBILITY BYTE (HDR1)
           .BLKB  1             ;SPARE
V.ANSN:'L' .BLKB 17.            ;ANSI 17 CHARACTER FILE NAME
V.BOFF:'L' .BLKB  1.            ;BUFFER OFFSET
V.DENS:'L' .BLKB  1.            ;REQUESTED UNIT DENSITY
V.DRAT:'L' .BLKB  1.            ;DEFAULT RECORD ATTRIBUTES
V.DBLK:'L' .BLKW  1.            ;DEFAULT BLOCK SIZE
V.DREC:'L' .BLKW  1.            ;DEFAULT RECORD SIZE
S.VSCB='B'.                     ;SIZE OF VSCB
```

MTADF$ (Cont.)

```
        .PSECT
;
; DEFINE OFFSETS INTO NULL WINDOW SECTION
;
        .ASECT
.=0
W.CTL:'L'.BLKW 1                ;CONTROL WORD IN WINDOW
V.WINC='B'V.WIND+W.CTL          ; CNTRL WORD IN NULL WINDOW

        .PSECT
                                ; RELATIVE TO THE VSCB
;
; MOUNTED VOLUME LIST OFFSET DEFINITIONS (MVL)
;
        .ASECT
.=0
        .IF     DF      R$$11M

M.NXT:'L' .BLKW 1               ;PTR TO NXT MVL NODE (11M)

        .ENDC

M.UIC:'L' .BLKW 1               ;OWNER UIC FROM RVOL #1
M.CH:'L' .BLKW 1                ; U.CH/U.VP (11D)
M.PROT:'L' .BLKW 1              ;PROTECTION U.AR IN 11D

        .IF     NDF     R$$11M

           .BLKW 2              ; ACP WORDS 11D
M.NXT:'L'  .BLKW 1              ;PTR TO NEXT MVL NODE (11D)
           .ENDC
M.RVOL:'L' .BLKB 1              ;RELATIVE VOL # OF MOUNTED VOLUME
M.STAT:'L' .BLKB 1              ;VOLUME STATUS
M.VIDP:'L' .BLKW 1              ;VOLUME ID POINTER
M.UCB:'L'  .BLKW 1              ;ADDR OF ASSOC UCB OR PUD

S.MVL='B'.                      ;SIZE OF MVL NODE

        .PSECT
;
; UNMOUNTED VOLUME AND VOLUME  LIST OFFSET DEFINITIONS (UVL)
;
        .ASECT
.=0
L.NXT:'L' .BLKW 1               ;PTR TO NXT UVL NODE
L.VOL1:'L' .BLKB 1              ;REL VOL # OF 1'ST VOL IN NODE
L.VOL2:'L' .BLKB 1              ;REL VOL # OF 2'ND VOL IN NODE
L.VID1:'L' .BLKB 6              ;VOL ID OF 1'ST VOL IN NODE
L.VID2:'L' .BLKB 6              ;VOL ID OF 2'ND VOL IN NODE
S.UVL='B'.                      ;SIZE OF UVL NODE

        .PSECT
;
; SYSTEM DATA STRUCTURE CONTENT VALUES
;

; VSCB VALUES
```

MTADF$ (Cont.)

```
; V.MOU VALUES
VM.OLD  ='B'    200             ;OLD .FL300 VOLUME -- VM.BYP WILL ALSO BE SET
VM.BYP  ='B'    100             ;BYPASS LABEL PROCESSING
VM.ULB  ='B'    40              ;UNLABELED TAPE
VM.FSC  ='B'    20              ;OVERRIDE FILE SET ID CHECK
VM.EXC  ='B'    10              ;OVERRIDE EXPRIATION DATE CHECK

; V.MST2 VALUES

V2.INI  ='B'    1               ;MAG WANTS US TO INITIALIZE NEXT OUTPUT
V2.XH2  ='B'    2               ;THIS FILE HAS NO HDR2, DON'T WRITE EOF2
V2.XH3  ='B'    4               ;THIS FILE HAS NO HDR3, DON'T WRITE EOF3
V2.NH3  ='B'    10              ;DON'T WRITE HDR3/EOX3 LABELS
V2.OAC  ='B'    20              ;OVERRIDE FILE/VOLUME ACCESIBILITY

; V.PSTA VALUES - UNBLOCKED TRANSITION STATE

VP.RM   ='B'    2               ;READ DATA MODE
VP.WM   ='B'    4               ;WRITE DATA MODE
VP.UCM  ='B'    6               ;UNLABELLED CREATE POSITIONING MODE
VP.SM   ='B'    10              ;SEARCH MODE
VP.MOU  ='B'    20              ;MOUNT MODE
VP.RWD  ='B'    40              ;REWIND OR VOL VERIFICATION WAIT
VP.VFY  ='B'    VP.RWD
VP.POS  ='B'    100             ;PROCESS IN POSITIONING MODE
                                ; (MULTI-SECTION FILE)

;
; BLOCKED STATE = -(UNBLOCKED TRANSITION STATE VALUES)
;
;PROCESS TIMED OUT BIT 0 = 1
        VP.TO='B'1
;
;
; NULL WINDOW CONTROL BIT DEFINITIONS
;
WI.RDV  ='B'    400             ;ACCESSED FOR READ
WI.WRV  ='B'    1000            ;ACCESSED FOR WRITE
WI.EXT  ='B'    2000            ;ACCESSED FOR EXTEND
WI.LCK  ='B'    4000            ;LOCKED
;

;
; MVL VALUES IN THE M.STAT FIELD
;
MS.VER  ='B'    200             ;VOL ID NOT VERIFIED
MS.RID  ='B'    1               ;VOL ID TO BE READ NOT CHECKED
MS.NMO  ='B'    2               ;MOUNT MESSAGE NOT GIVEN YET
MS.TMO  ='B'    4               ;ONE TIMEOUT ALREADY EXPRIED
MS.EXP  ='B'    10              ;EXPIRATION DATE MESSAGE GIVEN
;
;
; MISC BITS USED IN MOUNT (STORED IN V.STS)
;
MO.OVR  ='B'    1               ; OVER RIDE VOL NAME SWITCH
MO.UIC  ='B'    2               ; EXPLICIT UIC GIVEN
MO.PRO  ='B'    4               ; EXPLICIT PROTECTION GIVEN
MO.160  ='B'    10              ; 1600 BPI SPECIFIED
;
        .ENDM
```

OLRDF$

```
        .MACRO  OLRDF$  $$$GBL

;
; THIS MODULE DEFINES THE ONLINE RECONFIGURATION INTERFACE
; AS IMPLEMENTED BETWEEN THE RSX-11M-PLUS TASKS CON, HRC, AND
; THE RDDRV.
;

;
; DEFINE THE I/O FUNCTION CODES FOR ONLINE RECONFIGURATION CONTROL.
;

        .MCALL  .WORD.,DEFIN$
        .IF IDN <$$$GBL>,<DEF$G>
...GBL=1
        .IFF
...GBL=0
        .ENDC

;
; THE FOLLOWING MACRO DEFINES THE SUB-FUNCTION CODES FOR EACH OF THE
; OPERATIONS PERFORMED BY THE HRC TASK AND A PARAMETER DESCRIBING THE
; ARGUMENTS REQUIRED FOR EACH FUNCTION. IN A MACRO CALL THE FOLLOWING
; ARE THE LEGAL COMBINATIONS FOR THE 'MASK' PARAMETER:
;
;               <>      SIGNIFYING NO PARAMETERS
;               <D>     SIGNIFYING ONE BUFFER DESCRIPTOR
;               <D,D>   SIGNIFYING TWO BUFFER DESCRIPTORS
;               <D,CT>  SIGNIFYING ONE DESCRIPTOR AND 'CT'
;                         BYTES OF PARAMETERS
;               <CT>    SIGNIFYING 'CT' BYTES OF PARAMETERS
;

        .MACRO FUNC     NAME,SUBF,FUN,MASK
        .WORD.  IO.'NAME,SUBF,FUN
        FUNCA NAME,<MASK>
        .ENDM

        .MACRO FUNCA NAME,MSK
        PARCT=0
        DESCT=0
        .IRP X,<MSK>
        .IIF IDN <X>,<P> PARCT=PARCT+1
        .IIF IDN <X>,<D> DESCT=DESCT+1
        .IIF GT <PARCT-17> .ERROR INVALID PARAMETER COUNT
        .IIF GT <DESCT-17> .ERROR INVALID DESCRIPTOR COUNT
        .ENDR

        TEMP=<DESCT*4>+<PARCT*2>
        .WORD.  IO$'NAME,<<DESCT*20+PARCT>>,TEMP
        .ENDM

;
; DEFINE ONLINE RECONFIGURATION I/O FUNCTIONS
;

        .WORD.  IO.MFC,000,001  ; MULTI-FUNCTION MODIFY CONFIGURATION
        .WORD.  IO.RSC,000,002  ; READ SYSTEM CONFIGURATION
        .WORD.  IO.WSC,000,006  ; MODIFY DEVICE CONFIGURATION
```

OLRDF$ (Cont.)

```
;
; DEFINE SUBFUNCTIONS TO MODIFY DEVICE CONFIGURATION
;
        FUNC    ONL,001,006,<D,D>       ; SET DEVICE ONLINE
        FUNC    OFL,002,006,<D,D>       ; SET DEVICE OFFLINE
        FUNC    MAI,003,006,<D,D>       ; SET DEVICE IN MAINTAINENCE MODE
        FUNC    CAC,004,006,<>          ; CACHE CONTROL
        FUNC    MEM,005,006,<>          ; MIND CONTROL
        FUNC    STN,006,006,<P,P>       ; RECONFIGURATION CONTROL,
                                        ; SPECIFY TASK NAME
        FUNC    HRC,007,006,<P,P>       ; RECONFIGURATION CONTROL,
                                        ; HRC OPERATING MODE
        FUNC    ONE,010,006,<P,P>       ; ON <CONDITION> <COMMAND>
        FUNC    STA,011,006,<D>         ; RETURN DEVICE STATE
        FUNC    IF ,012,006,<P,P>       ; IF <CONDITION> <COMMAND>
        FUNC    RLI,013,006,<D,D,D,D>   ; LINK UNIBUS RUN
        FUNC    RUL,014,006,<D,D,D,D>   ; UNLINK UNIBUS RUN
        FUNC    MBO,015,006,<P,P,D,D,D,D,D,D,D,D,D> ; MEMORY BOX ONLINE
        FUNC    RSW,016,006,<D,D,D,D>   ; SWITCH BUS
        FUNC    WAT,017,006,<D>         ; WRITE ATTRIBUTES
        FUNC    RAT,020,006,<D,D>       ; READ ATTRIBUTES
        FUNC    MBF,021,006,<P,P,D,D,D,D,D,D,D,D,D> ; MEMORY BOX OFFLINE

        IO$MAX=21                       ; DEFINE MAXIMUM SUBFUNCTION


        DEFIN$  IS.HRG,6.       ; STOP PROCESSING CONDITION ENCOUNTERED
                                ; SECOND STATUS WORD IS ARGUMENT

;
; DEFINE A MACRO, WHICH WHEN EXPANDED WITH THE APPROPRIATE DEFINITION
; FOR .IOER. WILL DEFINE THE PRIVATE ERROR CODES USED BY HRC AND CON.
;
        .MACRO  OLREM$

        $$$VAL=-256.            ; DEFINE INITIAL ERROR NUMBER VALUE

        .IOER.  IE$DAL,<DEVICE already linked>
        .IOER.  IE$DNL,<DEVICE not linked>
        .IOER.  IE$PRM,<Parameter error>
        .IOER.  IE$SYN,<Syntax error>
        .IOER.  IE$AFE,<Attribute format error>
        .IOER.  IE$TMU,<HRC... Internal tables insufficient for this system>
        .IOER.  IE$CAB,<Unable to access busrun>
        .IOER.  IE$TRP,<HRC... internal addressing error>
        .IOER.  IE$ALG,<Memory box parameter error>
        .IOER.  IE$TQU,<Timeout on unit quieting operation>
        .IOER.  IE$EPO,<ONLINE CPU failure>
        .IOER.  IE$EUO,<ONLINE UNIT failure>
        .IOER.  IE$ECO,<ONLINE CONTROLLER failure>
        .IOER.  IE$EPF,<OFFLINE CPU failure>
        .IOER.  IE$EUF,<OFFLINE UNIT failure>
        .IOER.  IE$ECF,<OFFLINE CONTROLLER failure>
        .IOER.  IE$CFU,<Attempt to quiet unit for controller failed>
        .IOER.  IE$CSR,<CSR for controller not present in I/O page>
        .IOER.  IE$SWF,<Unable to switch unit away from current controller>
        .IOER.  IE$ICE,<HRC... detected I/O database consistancy error>
        .IOER.  IE$SCE,<Executive or Driver status change error>
        .IOER.  IE$MDE,<HRC... Memory descriptor format error>
```

OLRDF$ (Cont.)

```
        .IOER.  IE$NFW,<No path to target device is available>
        .IOER.  IE$CXT,<Unable to take unit with context offline.>
        .IOER.  IE$IDU,<Invalid device descriptor>
        .IOER.  IE$UNK,<Device is unknown in this configuration>
        .IOER.  IE$SZE,<HRC... Unable to access device to size drive>
        .IOER.  IE$POB,<HRC... Can't take box offline. Partition overmaps box>
        .IOER.  IE$NLB,<HRC... Can't take box offline. Not last box in memory>
        .IOER.  IE$OMP,<HRC... Can't modify partition size. Overmap exists>
        .IOER.  IE$POC,<HRC... Can't modify partition size. Occupied>
        .IOER.  IE$DFE,<HRC... Request format error.>
        .IOER.  IE$IDS,<HRC... Invalid device specification.>
        .IOER.  IE$UOE,<HRC... Unkown error from online/offline call>
        .ENDM

;
; CONDITION CODES FOR CONDITIONS TESTED BY IO.ONE AND IO.IF FUNCTIONS
;

        CO$ONL = 1              ; IF DEVICE NOW ONLINE
        CO$OFL = 2              ; IF DEVICE NOW OFFLINE
        CO$UNK = 3              ; UNKNOWN DEVICE
        CO$ACC = 4              ; ACCESSABLE (ACCESS PATH EXISTS)
        CO$ANY = 5              ; ANY ERROR CONDITION
        CO$MAI = 6              ; MAINTENANCE MODE

        CO$MAX = 6              ; MAXIMUM CODE

;
; CONDITION COMMAND CODES FOR IO.ONE AND IO.IF FUNCTIONS
;

        CD$STO = 2              ; 'STOP' COMMAND
        CD$GOT = 4              ; 'GOTO'
        CD$CON = 6              ; 'CONTINUE'

        CD$MAX = 6              ; MAXIMUM CONDITION DEFINED

;
; ARGUMENT DEFINITION FOR IO.HRC FUNCTION
;

        M$LOG = 1               ; SUPRESS CONFIGURATION TRANSMISSION TO ERRLOG
        M$INIT = 2              ; INITALIZE HRC
        M$DEBG = 4              ; SET HRC INTO DEBUG MODE. (DEVELOPMENT ONLY)
        M$EXIT = 10             ; EXIT REQUEST (FROM ABORT AST REQUEST)


;
; DEFINE TABLE OFFSETS AND STATUS BITS RETURNED IN RESPONSE TO
; A 'READ CONFIGURATION' QIO
;

        .ASECT
        .=0
C$DTYP: .BLKB   1               ; ENTRY TYPE FIELD

;
; ENTRY TYPE CODES ARE AS FOLLOWS
;
```

OLRDF$ (Cont.)

```
        ET$HDR = 1              ; CONFIGURATION HEADER ENTRY
        ET$END = 2              ; END OF CONFIGURATION DATA
        ET$DEV = 'A             ; MINIMUM VALUE FOR DEVICE SPECIFICATION
                                ;  ENTRY

C$DECT: .BLKB   1               ; COUNT OF TABLE ENTRIES (CPUS+SWITCHED
                                ; BUS RUNS+CONTROLLERS+UNITS)
C$DVER: .BLKB   1               ; VERSION OF RECONFIGURATION TASK PROTOCAL
C$DSTD: .BLKB   1               ; SIZE OF HEADER
C$DMUB: .BLKB   1               ; MAXIMUM UNIBUS RUNS SUPPORTED
C$DMCT: .BLKB   1               ; MAXIMUM CONTROLLERS OF A GIVEN TYPE
                                ;  SUPPORTED
        .EVEN
C$DFAC: .BLKW   2               ; FACILITES SUPPORTED IN HOST SYSTEM
C$DIDN: .BLKW   9.              ; HRC VERSION AND BUILD TIMESTAMP
C$STD:                          ; SIZE OF THE TABLE HEADER

;
; OFFSETS WITHIN THE FIXED PORTION OF A GIVEN ENTRY
;
        .=0

C$DTYP:                         ; ENTRY TYPE CODE
C$DNAM: .BLKW   1               ; TWO ASCII CHARACTER UNIT OR CONTROLLER NAME
C$DPUN: .BLKB   1               ; CONTROLLER NUMBER (0-255.)
C$DLUN: .BLKB   1               ; LOGICAL UNIT NUMBER IF THIS DEVICE IS A UNIT
C$DSCT: .BLKB   1               ; SUB-CONTROLLER NUMBER
C$DEVT: .BLKB   1               ; DEVICE TYPE CODE
C$DSTS: .BLKW   1               ; DEVICE STATUS MASK

;
; FLAG VALUES FOR C$DSTS
;

        CS$ATR=1                ; VARIABLE LENGTH ATTRIBUTE INFO IS APPENDED
        CS$EXF=76               ; FIELD IN C$DSTS CONTAINING COUNT OF ADDITIONAL
                                ; BYTES IN THIS DEVICE ENTRY
        CS$SUB=100              ; THIS IS A SUB-CONTROLLER DEVICE
        ;CS$XXX=200            ; UNUSED
        CS$OFL=400              ; 1=>DEVICE IS OFFLINE, 0=>DEVICE IS ONLINE
        CS$PDF=1000             ; DEVICE IS RESTRICTED TO PRIVILEGED DIAG FNS
        CS$POR=2000             ; THIS IS A MULTIPORT DEVICE
        CS$MBD=4000             ; DEVICE IS A MASS BUS DEVICE
        CS$UNK=10000            ; DEVICE IS UNKNOWN
        CS$ACC=20000            ; AN ONLINE ACCESS PATH EXISTS TO THIS DEVICE
        CS$MTD=40000            ; DEVICE IS MOUNTED(DISK) OR LOGGED IN (TERM)
        CS$DRV=100000           ; A DRIVER IS LOADED FOR THIS DEVICE

C$DST2: .BLKW   1               ; STATUS EXTENSION

        CS$PUN=20               ; 1=> THIS DEVICE SPECIFIED WITH PHYSICAL
                                ;       UNIT NUMBER
        CS$CRD=40               ; 1=> THIS IS A CONTROLLER RELATIVE DEVICE SPEC
        CS$PRC=100              ; 1=> THIS IS A PORT RELATIVE CONTROLLER SPEC
        CS$CTL=200              ; DEVICE IS A CONTROLLER (MUST BE SIGN BIT)
        CS$DCL=3400             ; DEVICE CLASS CODE FIELD. MUST BE LOW ORDER
                                ;  BIT OF HIGH BYTE.
```

OLRDF$ (Cont.)

```
;
; DEVICE CLASS VALUES
;
        DC$UNI = 0              ; UNIT
        DC$CTL = 1              ; CONTROLLER
        DC$MKU = 2              ; MEMORY BOX UNIT
        DC$MKC = 3              ; MEMORY BOX CONTROLLER
        DC$SBU = 4              ; SWITCHED BUS UNIT
        DC$SBC = 5              ; SWITCHED BUS CONTROLLER
        DC$CPU = 6              ; CPU
        ;DC$XXX = 7             ; UNUSED

C$DDAT: .BLKW   2               ; DEVICE DEPENDANT DATA
C$SME:                          ; SIZE IF A MINIMUM ENTRY

;
; VARIABLE PORTION OF A GIVEN ENTRY
;

;
; FOR CONTROLLERS
;
        .=C$SME

C$DKPO: .BLKW   1               ; PORT-STATUS-WORD. THIS DESCRIBES THE BUS RUN
                                ; CPU OR SWITCHED BUS, TO WHICH THIS
                                ; CONTROLLER IS CONNECTED.
C$SCT:                          ; MIMIMUM SIZE OF A CONTROLLER ENTRY

;
; FOR UNIT ENTRIES
;
        .=C$SME

C$DCTN: .BLKW   1               ; CONTROLLER NAME. TWO CHARACTER ASCII CODE
                                ; OF THE CONTROLLER TO WHICH THIS UNIT IS
                                ; ATTACHED.
C$DUPO: .BLKW   1               ; PORT-STATUS-WORD. THIS IS THE
                                ; FIRST OF THE PSWS DESCRIBING THE CONTROLLER(S)
                                ; TO WHICH THIS UNIT IS CONNECTED.
C$SUN:                          ; MIMIMUM SIZE OF A UNIT ENTRY

;
; FOR CPU-S
;

        .=C$SME

C$DCPO: .BLKW   1               ; PORT-STATUS-WORD. THIS IS THE BUS
                                ; NUMBER FOR THIS CPU.
C$SCP:                          ; MINIMUM SIZE OF A CPU ENTRY

;
; FOR MEMORY BOXES
;
```

OLRDF$ (Cont.)

```
        .=C$SME

C$DCTN: .BLKW   1               ; CONTROLLER NAME.
        .BLKW   4               ; MAXIMUM OF 4 PORTS FOR MEMORY CONTROLLERS

C$SMB:                          ; MAXIMUM SIZE OF A MEMORY BOX ENTRY


;
; STATUS BIT DEFINITIONS FOR THE PORT STATUS WORD
;

        CP$OFL=400              ; 1=> PORT IS OFFLINE
        CP$XXX=1000             ; UNUSED
        CP$CUR=2000             ; THIS PORT IS THE CURRENT PORT (S.KRB
                                ; REFERENCES THIS PORT
        CP$XXX=4000             ; UNUSED
        CP$XXX=10000            ; UNUSED
        CP$ACC=20000            ; THIS PORT HAS AN ACCESS PATH
        CP$MTD=40000            ; PORT HAS CONTEXT OR SERVICES A DEVICE HAVING
                                ; CONTEXT
        CP$XXX=100000           ; UNUSED

;
; DEVICE ATTRIBUTES CODES
;

        .MACRO ATT NAME,SIZ
        $$$TMP=$$$TMP+1
        DEFIN$ DA$'NAME,$$$TMP!<400*SIZ>
        .ENDM

        $$$TMP=0

        ATT     CSR,2           ; CSR ADDRESS
        ATT     VEC,2           ; VECTOR ADDRESS
        ATT     UBR,2           ; UNIBUS RUN
        ATT     TYP,2           ; DEVICE TYPE, READ ONLY
        ATT     VOL,12.         ; MOUNTED VOLUME NAME, READ ONLY
        ATT     ERR,10          ; DEVICE ERROR COUNTERS, READ/WRITE
        ATT     PRI,2           ; DEVICE INTERRUPT PRIORITY
        ATT     MBP,6           ; MEMORY BOX PARAMETER
        ATT     STE,2           ; SANITY TIMER ENABLE/DISABLE
        ATT     SAL,2           ; ALARM ENABLE/DISABLE
        ATT     DSN,2           ; DEVICE SERIAL NUMBER
        ATT     CSN,10          ; CPU SERIAL NUMBERS

;
; MEMORY BOX ATTRIBUTE BUFFER
;

        .ASECT
        .=0

C$MBAS: .BLKW   1               ; BASE ADDRESS OF BOX
C$MINT: .BLKB   1               ; INTERLEAVE FACTOR
        .BLKB   1               ; FREE BYTE
C$MSIZ: .BLKW   1               ; SIZE OF BOX IN 32 WORD BLOCKS
C$MGRN: .BLKW   1               ; BOX GRANULARITY. "BYTES-PER-UNIT"

C$MDSC:                         ; SIZE OF BOX ATTRIBUTE BUFFER
```

OLRDF$ (Cont.)

```
        .PSECT

;
; REDEFINE MACRO TO NULL
;

        .MACRO  OLRDF$  X
        .ENDM
        .MACRO ATT X
        .ENDM

        .ENDM

;
; MACRO FOR THE DEFINITION OF DEVICE TYPE CODES
;

        .MACRO DEVCD$ $$$GBL

        .MCALL DEFIN$

        .IF IDN <$$$GBL>,<DEF$G>
...GBL=1
        .IFF
...GBL=0
        .ENDC

        .MACRO DEV X
        DEFIN$ D$'X,$$$TMP
        $$$TMP=$$$TMP+1
        .ENDM

        $$$TMP = 0

        DEV UDET        ; UNDETERMINED DEVICE TYPE
        DEV UKNO        ; UNKNOWN DEVICE TYPE

        DEV RK03        ; RK03
        DEV RK05        ; RK05
        DEV RK5F        ; RK05-F (DUAL DENSITY FIXED CARTRIDGE)

        DEV RX01        ; RX01
        DEV RX02        ; RX02 (DUAL DENSITY RX01)

        DEV RL01        ; RL01
        DEV RL02        ; RL02

        DEV RP02        ; RP02
        DEV RP03        ; RP03
        DEV RP04        ; RP04
        DEV RP05        ; RP05
        DEV RP06        ; RP06
        DEV RP07        ; RP07

        DEV RK06        ; RK06
        DEV RK07        ; RK07
```

OLRDF$ (Cont.)

```
DEV RM02        ; RM02
DEV RM03        ; RM03
DEV RM05        ; RM05
DEV RM80        ; RM80

DEV RS03        ; RS03
DEV RS04        ; RS04 (DUAL DENSITY RS03)

DEV RF11        ; RF11/RS08

DEV TK25        ; TK25
DEV TK50        ; TK50
DEV TU10        ; TU10
DEV TU16        ; TU16
DEV TU45        ; TU45
DEV TU77        ; TU77
DEV TU78        ; TU78
DEV TS11        ; TS11
DEV TSU0        ; TSU05
DEV TSV0        ; TSV05
DEV TU80        ; TU80
DEV TU81        ; TU81

DEV TM02        ; TM02
DEV TM03        ; TM03
DEV TM78        ; TM78

DEV TU56        ; TU56
DEV TU58        ; TU58
DEV TU60        ; TU60

DEV MSCP        ; UDA50
DEV RA60        ; RA60
DEV RA80        ; RA80
DEV RA81        ; RA81
DEV RC25        ; RC25 (AZTEC)

DEV RD50        ; RD50
DEV RD51        ; RD51
DEV RX50        ; RX50

DEV ML11        ; ML11

DEV TERM        ; TERMINAL

$$$TMP=370
DEV USR0        ; USER TYPE 0
DEV USR1        ; USER TYPE 1
DEV USR2        ; USER TYPE 2
DEV USR3        ; USER TYPE 3
DEV USR4        ; USER TYPE 4
DEV USR5        ; USER TYPE 5
DEV USR6        ; USER TYPE 6
DEV USR7        ; USER TYPE 7


.MACRO DEVCD$
.ENDM
.MACRO DEV X
.ENDM

.ENDM
```

PCBDF$

```
        .MACRO  PCBDF$  L,B

;+
; MAIN PARTITION PCB
;-

        .ASECT
.=0
P.LNK:'L'.BLKW  1               ;LINK TO NEXT MAIN PARTITION PCB
          .BLKW  1              ;(UNUSED)
P.NAM:'L'.BLKW  2               ;PARTITION NAME IN RAD50
P.SUB:'L'.BLKW  1               ;POINTER TO FIRST SUBPARTITION
P.MAIN:'L'.BLKW 1               ;POINTER TO SELF
P.REL:'L'.BLKW  1               ;STARTING PHYSICAL ADDRESS IN 32W BLOCKS
P.BLKS:'L'
P.SIZE:'L'.BLKW 1               ;SIZE OF PARTITION IN 32W BLOCKS
P.WAIT:'L'.BLKW 2               ;PARTITION WAIT QUEUE LISTHEAD
          .BLKW  2              ;(UNUSED)
P.STAT:'L'.BLKW 1               ;PARTITION STATUS FLAGS
P.ST2:'L' .BLKW 1               ;STATUS EXTENSION FOR COMMON AND MAIN PCB'S
          .BLKW  3              ;(UNUSED)

P.HDLN:'L'.BLKB 1               ;SIZE OF EXTERNAL HEADER IN 32W BLOCKS
P.IOC:'L' .BLKB 1               ;PARTITION I/O COUNT

$$$=.

P.RRM:'L'.BLKW  1               ;REQUIRED RUN MASK


        .IF NDF M$$PRO

.=$$$

        .ENDC


        .IF NB  SYSDEF

P.LGTH='B'.                     ;PARTITION CONTROL BLOCK LENGTH

        .ENDC



;+
; TASK REGION PCB
;-

.=0
P.LNK:'L'.BLKW  1               ;UTILITY LINK WORD
P.PRI:'L'.BLKB  1               ;PRIORITY OF PARTITION
P.RMCT:'L'.BLKB 1               ;RESIDENT MAPPED TASKS COUNT
P.NAM:'L'.BLKW  2               ;PARTITION NAME IN RAD50
P.SUB:'L'.BLKW  1               ;POINTER TO NEXT SUBPARTITION
P.MAIN:'L'.BLKW 1               ;POINTER TO MAIN PARTITION
P.REL:'L'.BLKW  1               ;STARTING PHYSICAL ADDRESS IN 32W BLOCKS
P.BLKS:'L'
```

PCBDF$ (Cont.)

```
P.SIZE:'L'.BLKW 1               ;SIZE OF PARTITION IN 32W BLOCKS
          .BLKW 1               ;(UNUSED)
P.SWSZ:'L'.BLKW 1               ;PARTITION SWAP SIZE
P.DPCB:'L'.BLKW 1               ;CHECKPOINT ALLOCATION PCB
P.TCB:'L'.BLKW  1               ;TCB ADDRESS OF OWNER TASK
P.STAT:'L'.BLKW 1               ;PARTITION STATUS FLAGS
P.HDR:'L' .BLKW 1               ;POINTER TO HEADER CONTROL BLOCK
          .BLKW 1               ;(UNUSED)
P.ATT:'L' .BLKW 2               ;ATTACHMENT DESCRIPTOR LISTHEAD

P.HDLN:'L'.BLKB 1               ;SIZE OF EXTERNAL HEADER IN 32W BLOCKS
P.IOC:'L' .BLKB 1               ;PARTITION I/O COUNT

$$$=.

P.RRM:'L'.BLKW  1               ;REQUIRED RUN MASK


        .IF NDF M$$PRO

.=$$$

        .ENDC


;+
; COMMON REGION PCB
;-

.=0
P.LNK:'L'.BLKW  1               ;UTILITY LINK WORD
P.PRI:'L'.BLKB  1               ;PRIORITY OF PARTITION
P.RMCT:'L'.BLKB 1               ;RESIDENT MAPPED TASKS COUNT
P.NAM:'L'.BLKW  2               ;PARTITION NAME IN RAD50
P.SUB:'L'.BLKW  1               ;POINTER TO NEXT SUBPARTITION
P.MAIN:'L'.BLKW 1               ;POINTER TO MAIN PARTITION
P.REL:'L'.BLKW  1               ;STARTING PHYSICAL ADDRESS IN 32W BLOCKS
P.BLKS:'L'
P.SIZE:'L'.BLKW 1               ;SIZE OF PARTITION IN 32W BLOCKS
P.CBDL:'L'.BLKW 1               ;COMMON BLOCK DIRECTORY LINK
P.CSBA:'L'                      ; CACHE STATISTICS BLOCK LISTHEAD
                                ; ...(IF P2.CHE IS SET, PARTITION WON'T SWAP)
P.SWSZ:'L'.BLKW 1               ;PARTITION SWAP SIZE
P.DPCB:'L'.BLKW 1               ;POINTER TO DISK PCB
P.OWN:'L' .BLKW 1               ;OWNING UIC OF REGION
P.STAT:'L'.BLKW 1               ;PARTITION STATUS FLAGS
P.ST2:'L' .BLKW 1               ;STATUS EXTENSION FOR COMMON AND MAIN PCB'S
P.PRO:'L' .BLKW 1               ;PROTECTION WORD [DEWR,DEWR,DEWR,DEWR]
P.ATT:'L' .BLKW 2               ;ATTACHMENT DESCRIPTOR LISTHEAD

P.HDLN:'L'.BLKB 1               ;SIZE OF EXTERNAL HEADER IN 32W BLOCKS
P.IOC:'L' .BLKB 1               ;PARTITION I/O COUNT

$$$=.

P.RRM:'L'.BLKW  1               ;REQUIRED RUN MASK
```

PCBDF$ (Cont.)

```
        .IF NDF M$$PRO
.=$$$
        .ENDC

        .PSECT

;+
; PARTITION STATUS WORD BIT DEFINITIONS
;-

PS.OUT='B'100000                ;PARTITION IS OUT OF MEMORY(1=YES)
PS.CKP='B'40000                 ;PARTITION CHECKPOINT IN PROGRESS (1=YES)
PS.CKR='B'20000                 ;PARTITION CHECKPOINT IS REQUESTED (1=YES)
PS.CHK='B'10000                 ;PARTITION IS NOT CHECKPOINTABLE (1=YES)
PS.FXD='B'4000                  ;PARTITION IS FIXED (1=YES)
PS.CAF='B'2000                  ;CHECKPOINT SPACE ALLOCATION FAILURE (1=YES)
PS.LIO='B'1000                  ;MARKED BY SHUFFLER FOR LONG I/O (1=YES)
PS.NSF='B'400                   ;PARTITION IS NOT SHUFFLEABLE (1=YES)
PS.COM='B'200                   ;LIBRARY OR COMMON BLOCK (1=YES)
PS.LFR='B'100                   ;LAST LOAD OF REGION FAILED (1=YES)
PS.PER='B'40                    ;PARTIY ERROR OCCURED IN THIS REGION (1=YES)
PS.NWB='B'20                    ;COMMON SHOULDN'T BE WRITTEN BACK
PS.DEL='B'10                    ;PARTITION SHOULD BE DELETED WHEN NOT ATTACHED
                                : (1=YES)
PS.AST='B'4                     ;PARTITION HAS REGION LOAD AST PENDING

;+
; REQUIRED RUN MASK
;-

PR.UBT='B'100000                ;UNIBUS RUN T
PR.UBS='B'40000                 ;UNIBUS RUN S
PR.UBR='B'20000                 ;UNIBUS RUN R
PR.UBP='B'10000                 ;UNIBUS RUN P
PR.UBN='B'4000                  ;UNIBUS RUN N
PR.UBM='B'2000                  ;UNIBUS RUN M
PR.UBL='B'1000                  ;UNIBUS RUN L
PR.UBK='B'400                   ;UNIBUS RUN K
PR.UBJ='B'200                   ;UNIBUS RUN J
PR.UBH='B'100                   ;UNIBUS RUN H
PR.UBF='B'40                    ;UNIBUS RUN F
PR.UBE='B'20                    ;UNIBUS RUN E
PR.CPD='B'10                    ;PROCESSOR D
PR.CPC='B'4                     ;PROCESSOR C
PR.CPB='B'2                     ;PROCESSOR B
PR.CPA='B'1                     ;PROCESSOR A

;+
; STATUS EXTENSION WORD BIT DEFINITIONS
;       (THESE BITS CAN ONLY BE EXAMINED IN COMMON OR MAIN PCB'S)
;-
```

PCBDF$ (Cont.)

```
P2.LMA='B'40000                 ;DON'T SHUFFLE,DELETE SPINDLE OR MUTILATE
                                ;THIS PARTITION
P2.CPC='B'20000                 ;CPCR INITIATED CHECKPOINT PENDING
P2.CHE='B'10000                 ;CACHE PARTITION
P2.SEC='B'4000                  ;THIS IS RO SECTION OF MU TASK
                                ;WITH TCB IN SEC. POOL
P2.PAR='B'2000                  ;THE FIXER TASK HAS HANDLED A PARITY ERROR
P2.POL='B'1000                  ;SECONDARY POOL PARTITION
P2.CPU='B'400                   ;MULTIPROCESSOR CPU PARTITION
P2.PIC='B'200                   ;POSITION INDEPENDENT LIBRARY OR COMMON (1=YES)
P2.RON='B'100                   ;READ-ONLY COMMON (1=YES)
P2.DRV='B'40                    ;DRIVER COMMON PARTITION (1=YES)
P2.APR='B'7                     ;STARTING APR NUMBER MASK FOR NON-PIC COMMON

;+
; CHECKPOINT FILE PCB
;-

        .ASECT
.=0
P.LNK:'L' .BLKW 1               ;LINK WORD OF CHECKPOINT FILE PCB'S
P.UCB:'L' .BLKW 1               ;UCB ADDRESS OF CHECKPOINT FILE DEVICE
P.LBN:'L' .BLKW 1               ;HIGH PART OF STARTING LBN
          .BLKW 1               ;LOW PART OF STARTING LBN
P.SUB:'L' .BLKW 1               ;POINTER TO FIRST CHECKPOINT ALLOCATION PCB
P.MAIN:'L' .BLKW 1              ;MUST BE 0 (FOR $RLPR1)
P.REL:'L' .BLKW 1               ;CONTAINS 0 IF FILE IN USE, 1 IF NOT IN USE
P.SIZE:'L' .BLKW 1              ;SIZE OF CHECKPOINT FILE IN 256W BLOCKS
P.DLGH='B'.                     ;LENGTH OF ALL DISK PCB'S

;+
; CHECKPOINT ALLOCATION PCB
;-

.=0
        .BLKW 4                 ;(UNUSED)
P.SUB:'L' .BLKW 1               ;LINK TO NEXT CHECKPOINT ALLOCATION PCB
P.MAIN:'L' .BLKW 1              ;ADDRESS OF CHECKPOINT FILE PCB
P.REL:'L' .BLKW 1               ;RELATIVE POSITION IN FILE IN 256W BLOCKS
P.SIZE:'L' .BLKW 1              ;SIZE ALLOCATED IN 256W BLOCKS

;+
; COMMON TASK IMAGE FILE PCB
;-

.=0
P.FID1:'L' .BLKW 1              ;FILE ID WORD FOR SAVE
P.UCB:'L' .BLKW 1               ;UCB ADDRESS OF DEVICE ON WHICH COMMON RESIDES
P.LBN:'L' .BLKW 1               ;HIGH PART OF STARTING LBN
        .BLKW 1                 ;LOW PART OF STARTING LBN
P.FID2:'L' .BLKW 1              ;FILE ID WORD FOR SAVE
P.MAIN:'L'.BLKW 1               ;POINTER TO SELF
P.REL:'L' .BLKW 1               ;ALWAYS CONTAINS A 0
P.FID3:'L' .BLKW 1              ;FILE ID WORD FOR SAVE

;+
; ATTACHMENT DESCRIPTOR OFFSETS
;-
```

PCBDF$ (Cont.)

```
        .ASECT
.=0
A.PCBL:'L'.BLKW 1               ;PCB ATTACHMENT QUEUE THREAD WORD
A.PRI:'L'.BLKB  1               ;PRIORITY OF ATTACHED TASK
A.IOC:'L'.BLKB  1               ;I/O COUNT THROUGH THIS DESCRIPTOR
A.TCB:'L'.BLKW  1               ;TCB ADDRESS OF ATTACHED TASK
A.TCBL:'L'.BLKW 1               ;TCB ATTACHMENT QUEUE THREAD WORD
A.STAT:'L'.BLKB 1               ;STATUS BYTE
A.MPCT:'L'.BLKB 1               ;MAPPING COUNT OF TASK THRU THIS DESCRIPTOR
A.PCB:'L'.BLKW  1               ;PCB ADDRESS OF ATTACHED TASK
A.LGTH='B'.                     ;LENGTH OF ATTACHMENT DESCRIPTOR

;+
; ATTACHMENT DESCRIPTOR STATUS BYTE BIT DEFINITIONS
;-

        .PSECT
AS.PRO='B'100                   ;A.TCB IS SEC POOL TCB BIAS (1=YES)
AS.SBP='B'20                    ;CACHE BYPASS REQUESTED
AS.RBP='B'40                    ;REQUEST TO NOT BYPASS CACHE
AS.DEL='B'10                    ;TASK HAS DELETE ACCESS (1=YES)
AS.EXT='B'4                     ;TASK HAS EXTEND ACCESS (1=YES)
AS.WRT='B'2                     ;TASK HAS WRITE ACCESS (1=YES)
AS.RED='B'1                     ;TASK HAS READ ACCESS (1=YES)

        .MACRO  PCBDF$  X,Y,Z
        .ENDM
        .ENDM
```

PKTDF$

```
        .MACRO  PKTDF$,L,B

;+
; ASYNCHRONOUS SYSTEM TRAP CONTROL BLOCK OFFSET DEFINITIONS
;
; SOME POSITIONAL DEPENDENCIES BETWEEN THE OCB AND THE AST CONTROL BLOCK
; ARE RELIED UPON IN THE ROUTINE $FINXT IN THE MODULE SYSXT.
;-

        .ASECT
.=177774
A.KSR5:'L' .BLKW 1              ;SUBROUTINE KISAR5 BIAS (A.CBL=0)
A.DQSR:'L' .BLKW 1              ;DEQUEUE SUBROUTINE ADDRESS (A.CBL=0)
        .BLKW   1               ;AST QUEUE THREAD WORD
A.CBL:'L' .BLKW 1               ;LENGTH OF CONTROL BLOCK IN BYTES
                                ;IF A.CBL = 0, THE AST CONTROL BLOCK IS
                                ;TO BE DEALLOCATED BY THE DEQUEUE SUBROUTINE
                                ;POINTED TO BY A.DQSR MAPPED VIA APR 5
                                ;VALUE A.KSR5.  THIS IS CURRENTLY USED ONLY
                                ;BY THE FULL DUPLEX TERMINAL DRIVER FOR
                                ;UNSOLICITED CHARACTER ASTS.
                                ;IF THE LOW BYTE OF A.CBL = 0, AND THE
                                ;HIGH BYTE IS NOT = 0, THE AST CONTROL BLOCK
                                ;IS A SPECIFIED AST, WITH LENGTH, C.LGTH.
                                ;IF THE HIGH BYTE OF A.CBL=0
                                ;AND THE LOW BYTE > 0, THEN
                                ;THE LOW BYTE IS THE LENGTH OF THE
                                ;AST CONTROL BLOCK.
                                ;IF HIGH BYTE = 0 AND LOW BYTE IS NEGATIVE,
                                ;THEN THE BLOCK IS A KERNEL AST
                                ;BIT 6 IS SET IF $SGFIN SHOULD
                                ;NOT BE CALLED PRIOR TO DISPATCHING
                                ;THE AST, AND THE LOW SIX BITS (5-0)
                                ;REPRESENT THE INDEX/2 INTO THE
                                ;KERNEL AST DISPATCH TABLE ($KATBL)
A.BYT:'L' .BLKW 1               ;NUMBER OF BYTES TO ALLOCATE ON TASK STACK
A.AST:'L' .BLKW 1               ;AST TRAP ADDRESS
A.NPR:'L' .BLKW 1               ;NUMBER OF AST PARAMETERS
A.PRM:'L' .BLKW 1               ;FIRST AST PARAMETER
AS.FPA='B'1                     ;CODE FOR FLOATING POINT AST
AS.RCA='B'2                     ;CODE FOR RECEIVE DATA AST
AS.RRA='B'3                     ;CODE FOR RECEIVE BY REFERENCE AST
AS.PEA='B'4                     ;CODE FOR PARITY ERROR AST
AS.REA='B'5                     ;CODE FOR REQUESTED EXIT AST
AS.PFA='B'6                     ;CODE FOR POWER FAIL AST
AS.CAA='B'7                     ;CODE FOR CLI COMMAND ARRIVAL AST

; BIT VALUES IN A.PRM+5
;
AF.XCC='B'1                     ;ATTACHED FOR ALL BUT CONTROL-C (TF.XCC)
AF.NOT='B'2                     ;ATTACHED FOR ALL NOTIFICATION (TF.NOT)
AF.OOB='B'4                     ;ACB IS FOR OUT-OF-BAND AST
AF.AST='B'10                    ;ACB HANDLES UNSOL. INPUT CHAR AST'S (TF.AST)
AF.ESQ='B'20                    ;ATTACHED FOR ESCAPE SEQUENCES (TF.ESQ)
AF.LCK='B'40                    ;ACB IS LOCKED
AF.QUE='B'100                   ;ACB IS QUEUED
AF.MDE='B'200                   ;ACB IS MARKED FOR DELETE
```

PKTDF$ (Cont.)

```
;
; ABORTER SUBCODES FOR ABORT AST (AS.REA) TO BE RETURNED ON USER'S STACK
;
AB.NPV='B'1                     ;ABORTER IS NONPRIVILEGED (1=YES)
AB.TYP='B'2                     ;ABORT FROM DIRECTIVE (0=YES)
                                ;ABORT FROM CLI COMMAND (1=YES)
A.PLGH='B'70                    ;SIZE OF PARITY ERROR AST CONTROL BLOCK
A.DUCB='B'10                    ;UCB OF TERM ISSUING DEBUG COMMAND
A.DLGH='B'10.                   ;LENGTH OF DEBUG (AK.TBT) AST BLOCK

;       KERNEL AST CONTROL CODES (A.CBL)

AK.BUF='B'200                   ;BUFFERED I/O COMPLETION
                                ;THIS CODE MUST BE 200 UNTIL ALL
                                ;REFERENCES IN TTDRV ARE FIXED
AK.OCB='B'201                   ;OFFSPRING TASK EXIT
AK.GBI='B'202                   ;SEGMENTED BUFFERED I/O COMPLETION
AK.TBT='B'203                   ;TASK FORCE T-BIT TRAP (DEBUG CMD)
AK.DIO='B'204                   ;DELAYED I/O COMPLETION
AK.GGF='B'205                   ;GRP. GBL. RUNDWN
;+
; BIT DEFINITIONS FOR THE GET/SET INFORMATION DIRECTIVE.
;-

SF.PRV='B'100000                ;FUNCTION IS PRIVILEGED
SF.IN='B' 40000                 ;FUNCTION IS AN INPUT FUNCTION

;+
; GROUP GLOBAL EVENT FLAG BLOCK OFFSETS
;-

.=0
G.LNK:'L'.BLKW  1               ;LINK WORD
G.GRP:'L'.BLKB  1               ;GROUP NUMBER
G.STAT:'L'.BLKB 1               ;STATUS BYTE
G.CNT:'L'.BLKW  1               ;ACCESS COUNT
G.EFLG:'L'.BLKW 2               ;EVENT FLAGS

G.LGTH='B'.                     ;LENGTH OF GROUP GLOBAL EVENT FLAG
                                ;BLOCK

GS.DEL='B'1                     ;STATUS BIT -- MARKED FOR DELETE

;+
; EXECUTIVE POOL MONITOR CONTROL FLAGS
;-

; $POLST IS THE SYNCHRONIZATION WORD BETWEEN THE EXEC AND POOL MONITOR

PC.HIH='B'1                     ;HIGH POOL LIMIT CROSSED (1=YES)
PC.LOW='B'2                     ;LOW POOL LIMIT CROSSED (1=YES)
PC.ALF='B'4                     ;POOL ALLOCATION FAILURE (1=YES)
PC.XIT='B'200                   ;FORCE POOL MONITOR TASK TO EXIT (MUST
                                ;BE COUPLED WITH SETTING FE.MXT IN THE
                                ;FEATURE MASK)
PC.NRM='B'PC.HIH*400            ;POOL TASK INHIBIT BIT FOR HIGH POOL
PC.ALM='B'PC.LOW*400            ;POOL TASK INHIBIT BIT FOR LOW POOL
```

PKTDF$ (Cont.)

```
; $POLFL IS THE POOL USAGE CONTROL WORD

PF.INS='B'40                    ;REJECT NONPRIVILEGED INS/RUN/REM
PF.LOG='B'100                   ;NONPRIVILEGED LOGINS ARE DISABLED
PF.REQ='B'200                   ;STALL REQUEST OF NONPRIV. TASKS

PF.ALL='B'177777                ;TAKE ALL POSSIBLE ACTIONS TO SAVE POOL

;+
; OFFSPRING CONTROL BLOCK DEFINITIONS
;
; SOME POSITIONAL DEPENDENCIES ARE DEPENDED ON BETWEEN THE OCB AND THE
; AST BLOCK IN THE ROUTINE $FINXT IN THE MODULE SYSXT.
;-

.=0
O.LNK:'L'.BLKW  1               ;OCB LINK WORD
O.MCRL:'L'.BLKW 1               ;ADDRESS OF MCR COMMAND LINE
O.PTCB:'L'.BLKW 1               ;PARENT TCB ADDRESS
O.AST:'L'.BLKW  1               ;EXIT AST ADDRESS
O.EFN:'L'.BLKW  1               ;EXIT EVENT FLAG
O.ESB:'L'.BLKW  1               ;EXIT STATUS BLOCK VIRTUAL ADDRESS
O.STAT:'L'.BLKW 8.              ;EXIT STATUS BUFFER
O.LGTH='B'.                     ;LENGTH OF OCB

;+
; I/O PACKET OFFSET DEFINITIONS
;-

        .ASECT
.=0
I.LNK:'L' .BLKW 1               ;I/O QUEUE THREAD WORD
I.PRI:'L' .BLKB 1               ;REQUEST PRIORITY
I.EFN:'L' .BLKB 1               ;EVENT FLAG NUMBER
I.TCB:'L' .BLKW 1               ;TCB ADDRESS OF REQUESTOR
I.LN2:'L' .BLKW 1               ;POINTER TO SECOND LUN WORD
I.UCB:'L' .BLKW 1               ;POINTER TO UNIT CONTROL BLOCK
I.FCN:'L' .BLKW 1               ;I/O FUNCTION CODE
I.IOSB:'L' .BLKW 1              ;VIRTUAL ADDRESS OF I/O STATUS BLOCK
          .BLKW 1               ;I/O STATUS BLOCK RELOCATON BIAS
          .BLKW 1               ;I/O STATUS BLOCK ADDRESS
I.AST:'L' .BLKW 1               ;AST SERVICE ROUTINE ADDRESS
I.PRM:'L' .BLKW 1               ;RESERVED FOR MAPPING PARAMETER #1
          .BLKW 6               ;PARAMETERS 1 TO 6
          .BLKW 1               ;USER MODE DIAGNOSTIC PARAMETER WORD

;
; FOLLOWING ARE DEFINITIONS FOR FLAG BITS IN I.PRM+11
; (DSA DRIVERS INTERNAL USE ONLY)
;

IP.FAK ='B' 20                  ;IOP IS PSEUDO IOP
IP.ABO ='B' 40                  ;(MUDRV)ABORT COMMAND MUST BE ISSUED FOR IOP
IP.PND ='B' 100                 ;(MUDRV)ABORT COMMAND WAS ISSUED FOR IOP
IP.UMR ='B'200                  ;A UMR WAIT BLOCK IS IN USE FOR THIS I/O
```

PKTDF$ (Cont.)

```
I.ATTL='B'.                     ;MINIMUM LENGTH OF I/O PACKET (USED BY
                                ;FILE SYSTEM TO CALCULATE MAXIMUM
                                ;NUMBER OF ATTRIBUTES)
I.AADA:'L' .BLKW 2              ;STORAGE FOR ATT DESCR PTRS WITH I/O
I.LGTH='B'.                     ;LENGTH OF I/O REQUEST CONTROL BLOCK
I.ATRL='B'6*8.                  ;LENGTH OF FILE SYSTEM ATTRIBUTE BLOCK

;
; DEFINE OFFSETS IN I/O PACKET EXTENSION (IOPX)
;
        .ASECT
.=      0

I.XLNK:'L' .BLKW  1             ;LINK WORD
I.XIOP:'L' .BLKW  1             ;I/O PACKET ADDRESS
I.XTCB:'L' .BLKW  1             ;TCB ADDRESS OF REQUESTING TASK
I.XMOD:'L' .BLKW  2             ;MODIFIER WORDS (NOTE: 2ND WORD MUST BE
                                ;SPECIFIED AND MUST BE ZERO.)
I.XRBF:'L' .BLKW  2             ;READ DATA BUFFER ADDRESS APR BIAS
                                ;READ DATA BUFFER VIRTUAL ADDRESS
I.XRBL:'L' .BLKW  1             ;READ DATA BUFFER LENGTH
I.XTMO:'L' .BLKW  1             ;READ TIME-OUT INTERVAL
I.XPBF:'L' .BLKW  2             ;PROMPT BUFFER ADDRESS APR BIAS
                                ;PROMPT BUFFER VIRTUAL ADDRESS
I.XPBL:'L' .BLKW  1             ;PROMPT BUFFER LENGTH
I.XPBV:'L' .BLKW  1             ;PROMPT BUFFER VERTICAL FORMS CONTROL
I.XTTB:'L' .BLKW  2             ;TERMINATOR TABLE ADDRESS APR BIAS
                                ;TERMINATOR TABLE VIRTUAL ADDRESS
I.XTTL:'L' .BLKW  1             ;TERMINATOR TABLE LENGTH
I.XDBF:'L' .BLKW  2             ;DEFAULT INPUT BUFFER ADDRESS APR BIAS
                                ;DEFAULT INPUT BUFFER VIRTUAL ADDRESS
I.XDBL:'L' .BLKW  1             ;DEFAULT INPUT BUFFER LENGTH

;+
; CLI PARSER BLOCK (CPB) DEFINITIONS
;-

.=0

C.PTCB:'L' .BLKW 1              ;ADDRESS OF CLI'S TCB
C.PNAM:'L' .BLKW 2              ;CLI NAME
C.PSTS:'L' .BLKW 1              ;STATUS MASK
C.PDPL:'L' .BLKB 1              ;LENGTH OF DEFAULT PROMPT
C.PCPL:'L' .BLKB 1              ;LENGTH O CNTRL/C PROMPT
C.PRMT:'L'                      ;START OF PROMPT STRINGS. DEFAULT
                                ;IS CONCATENATED WITH CONTROL C PROMPT

;
; STATUS BIT DEFINITIONS
;

CP.NUL='B'1                     ;PASS EMPTY COMMANDS TO CLI
CP.MSG='B'2                     ;CLI DESIRES SYSTEM MESSAGES
CP.LGO='B'4                     ;CLI WANTS COMMANDS FROM LOGGED OFF TTYS
CP.DSB='B'10                    ;CLI IS DISABLED
CP.PRV='B'20                    ;USER MUST BE PRIV TO SET TTY TO THIS CLI
CP.SGL='B'40                    ;DON'T HANDLE CONTINUATIONS (M-PLUS ONLY)
CP.NIO='B'100                   ;MCR..., HEL, BYE DO NO I/O TO TTY
                                ;HEL, BYE DO NOT SET CLI ETC.
```

PKTDF$ (Cont.)

```
CP.RST='B'200                       ;ABILITY TO SET TO THIS CLI IS RESTRICTED
                                    ;TO THE CLI ITSELF
CP.EXT='B'400                       ;PASS TASK EXIT PROMPT REQUESTS TO CLI
CP.POL='B'1000                      ;CLI TCB IS IN SECONDARY POOL
CP.CTC='B'2000                      ;^C NOTIFICATION PACKETS ARE WANTED

;+
; SECONDARY POOL COMMAND BUFFER BLOCKS
;-

.=0
C.CLK:'L'   .BLKW       1           ;LINK WORD
C.CTCB:'L'  .BLKW       1           ;TCB ADDRESS OF TASK TO RECEIVE COMMAND
C.CUCB:'L'  .BLKW       1           ;UCB ADDRESS OF RESPONSIBLE TERMINAL
C.CCT:'L'   .BLKW       1           ;CHARACTER COUNT, EXCLUDING TRAILING CR
C.CSTS:'L'  .BLKW       1           ;STATUS MASK
C.CMCD:'L'                          ;SYSTEM MESSAGE CODE
C.CSO:'L'   .BLKW       1           ;STARTING OFFSET OF VALID COMMAND TEXT
C.CTR:'L'   .BLKB       1           ;TERMINATOR CHARACTER
C.CBLK:'L'  .BLKB       1           ;SIZE OF PACKET IN SEC POOL (32 WD.) BLOCKS
C.CTXT:'L'                          ;COMMAND TEXT, FOLLOWED BY CR

;+
; STATUS BITS FOR COMMAND BLOCKS
;-

CC.MCR='B'1                         ;FORCE COMMAND TO MCR
CC.PRM='B'2                         ;ISSUE DEFAULT PROMPT
CC.EXT='B'4                         ;TASK EXIT PROMPT REQUEST
CC.KIL='B'10                        ;DELETE ALL CONTINUATION PIECES FROM THIS TTY
CC.CLI='B'20                        ;COMMAND TO BE RETREIVED BY GCCI$ ONLY
CC.MSG='B'40                        ;PACKET CONTAINS SYSTEM MESSAGE TO CLI
CC.TTD='B'100                       ;COMMAND CAME FROM TTDRV
CC.CTC='B'200                       ;^C NOTIFICATION PACKET

; IDENTIFIER CODES FOR SYSTEM TO CLI MESSAGES
;
; CODES 0-127. ARE RESERVED FOR USE BY DIGITAL
; CODES 128.-255. ARE RESERVED FOR USE BY CUSTOMERS
;
CM.INE='B'1                         ;CLI INITIALIZED ENABLED
CM.IND='B'2                         ;CLI INITIALIZED DISABLED
CM.CEN='B'3                         ;CLI ENABLED
CM.CDS='B'4                         ;CLI DISABLED
CM.ELM='B'5                         ;CLI BEING ELIMINATED
CM.EXT='B'6                         ;CLI MUST EXIT IMMEDIATELY
CM.LKT='B'7                         ;NEW TERMINAL LINKED TO CLI
CM.RMT='B'8.                        ;TERMINAL REMOVED FROM CLI
CM.MSG='B'9.                        ;GENERAL MESSAGE TO CLI

;+
; ANCILLARY CONTROL BLOCK (ACB) DEFINITIONS
;-

.=0
A.REL:'L' .BLKW 1                   ;ACD RELOCATION BIAS
A.DIS:'L' .BLKW 1                   ;ACD DISPATCH TABLE POINTER
A.MAS:'L' .BLKW 2                   ;ACD FUNCTION MASK WORDS
A.NUM:'L' .BLKB 1                   ;ACD IDENTIFICATION NUMBER
A.FLEN:'L'.BLKB 1                   ;LENGTH IN BYTES OF FULL ACB
```

PKTDF$ (Cont.)

```
A.LIN:'L' .BLKW 1               ;ACD LINK WORD
A.ACC:'L' .BLKB 1               ;ACD ACCESS COUNT
A.STA:'L' .BLKB 1               ;ACD STATUS BYTE
A.PLEN='B'.                     ;LENGTH IN BYTES OF PROTOTYPE ACB
;
.=A.LIN                         ;FULL ACB OVERLAPS PROTOTYPE ACB
A.IMAP:'L' .BLKW 1              ;ACD INTERRUPT BUFFER RELOCATION BIAS
A.IBUF:'L' .BLKW 1              ;ACD INTERRUPT BUFFER ADDRESS
A.ILEN:'L' .BLKW 1              ;ACD INTERRUPT BUFFER LENGTH
A.SMAP:'L' .BLKW 1              ;ACD SYSTEM STATE BUFFER RELOCATION BIAS
A.SBUF:'L' .BLKW 1              ;ACD SYSTEM STATE BUFFER ADDRESS
A.SLEN:'L' .BLKW 1              ;ACD SYSTEM STATE BUFFER LENGTH
A.IOS:'L' .BLKW 2               ;ACD I/O STATUS
A.RES='B'.                      ;START OF ACB RESERVED FOR USE BY THE ACD
;
; DEFINE THE FLAG VALUES IN THE OFFSET U.AFLG
;
UA.ACC='B'1                     ;ACCEPT THIS CHARACTER
UA.PRO='B'2                     ;PROCESS THIS CHARACTER
UA.ECH='B'4                     ;ECHO THIS CHARACTER
UA.TYP='B'10                    ;FORCE THIS CHARACTER INTO TYPEAHEAD
UA.SPE='B'20                    ;THIS CHARACTER HAS A SPECIAL ECHO
UA.PUT='B'40                    ;PUT THIS CHARACTER IN THE INPUT BUFFER
UA.CAL='B'100                   ;CALL THE ACD BACK AFTER THE TRANSFER
UA.COM='B'200                   ;COMPLETE THE INPUT REQUEST
;
UA.ALL='B'400                   ;ALLOW PROCESSING OF THIS I/O REQUEST
UA.TRN='B'1000                  ;TRANSLATE CHARACTERS FROM OUTPUT QIO
UA.TRA='B'2000                  ;TRANSFER CHARACTERS WHEN I/O COMPLETES
;
; DEFINE THE ACD ENTRY POINTS (OFFSETS INTO THE DISPATCH TABLE)
;
.=0
A.ACCE:'L' .BLKW 1              ;I/O REQUEST ACCEPTANCE ENTRY POINT
A.DEQU:'L' .BLKW 1              ;I/O REQUEST DEQUEUE ENTRY POINT
A.POWE:'L' .BLKW 1              ;POWER FAILURE ENTRY POINT
A.INPU:'L' .BLKW 1              ;INPUT COMPLETION ENTRY POINT
A.OUTP:'L' .BLKW 1              ;OUTPUT COMPLETION ENTRY POINT
A.CONN:'L' .BLKW 1              ;CONNECTION ENTRY POINT
A.DISC:'L' .BLKW 1              ;DISCONNECTION ENTRY POINT
A.RECE:'L' .BLKW 1              ;INPUT CHARACTER RECEPTION ENTRY POINT
A.PROC:'L' .BLKW 1              ;INPUT CHARACTER PROCESSING ENTRY POINT
A.TRAN:'L' .BLKW 1              ;OUTPUT QIO CHARACTER TRANSLATION ENTRY POINT
A.CALL:'L' .BLKW 1              ;CALL ACD BACK AFTER TRANSFER ENTRY POINT

;
; DEFINE THE STATUS BITS IN A.STA OF THE PROTOTYPE ACB
;
AS.DLT='B'1                     ;ACD IS MARKED FOR DELETE
AS.DIS='B'2                     ;ACD IS DISABLED
        .PSECT

        .MACRO  PKTDF$  X,Y,Z
        .ENDM
        .ENDM
```

SCBDF$

```
        .MACRO  SCBDF$,L,B

;+
; STATUS CONTROL BLOCK
;
; THE STATUS CONTROL BLOCK (SCB) DEFINES THE STATUS OF A DEVICE CONTROLLER.
; THERE IS ONE SCB FOR EACH CONTROLLER IN A SYSTEM. THE SCB IS POINTED TO
; BY UNIT CONTROL BLOCKS. TO EXPAND ON THE TELETYPE EXAMPLE ABOVE, EACH TELE-
; TYPE INTERFACED VIA A DL11-A WOULD HAVE A SCB SINCE EACH DL11-A IS AN IN-
; DEPENDENT INTERFACE UNIT. THE TELETYPES INTERFACED VIA THE DH11 WOULD ALSO
; EACH HAVE AN SCB SINCE THE DH11 IS A SINGLE CONTROLLER BUT MULTIPLEXES MANY
; UNITS IN PARALLEL.
;-

        .IF NB  SYSDEF

        .ASECT
.=0
S.LHD:'L' .BLKW 2               ;CONTROLLER I/O QUEUE LISTHEAD
S.URM:'L'                       ;REFERENCE LABEL


        .IF DF  M$$PRO

        .BLKW   1               ;UNIBUS RUN MASK FOR THE FORK BLOCK

        .ENDC


S.FRK:'L' .BLKW 1               ;FORK BLOCK LINK WORD
        .BLKW   1               ;FORK-PC
        .BLKW   1               ;FORK-R5
        .BLKW   1               ;FORK-R4


        .IF DF  L$$DRV

S.KS5:'L' .BLKW 1               ;FORK KISAR5

        .ENDC


S.PKT:'L' .BLKW 1               ;ADDRESS OF CURRENT I/O PACKET
S.CTM:'L' .BLKB 1               ;CURRENT TIMEOUT COUNT
S.ITM:'L' .BLKB 1               ;INITIAL TIMEOUT COUNT
S.STS:'L' .BLKB 1               ;STATUS (0=FREE, NE 0=BUSY)
S.ST3:'L'.BLKB  1               ;STATUS EXTENSION BYTE
S.ST2:'L' .BLKW 1               ;STATUS EXTENSION
S.KRB:'L' .BLKW 1               ;ADDRESS OF KRB
S.RCNT:'L'.BLKB 1               ;NUMBER OF REGISTERS TO COPY
S.ROFF:'L'.BLKB 1               ;OFFSET TO FIRST DEV REG TO COPY
S.EMB:'L'.BLKW  1               ;ERROR MESSAGE BLOCK POINTER
S.KTB:'L' .BLKW 1               ;START OF MULTI-ACCESS KRBS

        .PSECT
```

SCBDF$ (Cont.)

```
;+
; OFFSETS FOR MSCP/TMSCP DRIVER DATA BASES (MUDRV, DUDRV)
;-

S.PORT='B'S.EMB+2               ;FIRST 3 CHAR. OF PORT NAME IN RAD50
S.PBIA='B'S.EMB+4               ;BIAS OF PORT
S.QST='B'S.EMB+6                ;ADDRESS OF QST (MU,DU CONTR. TABLE)
S.BSYU='B'S.EMB+10              ;UNIT ASSOCIATED WITH OLDEST CMD TO CONTR.

        .IFF


;+
; STATUS CONTROL BLOCK  STATUS EXTENSION BIT DEFINITIONS
;-

S2.EIP='B'1                     ;ERROR IN PROGRESS (1=YES)
S2.ENB='B'2                     ;ERROR LOGGING ENABLED (0=YES)
S2.LOG='B'4                     ;ERROR LOGGING SUPPORTED (1=YES)
S2.MAD='B'10                    ;MULTIACCESS DEVICE (1=YES)
S2.LDS='B'40                    ;LOAD SHARING ENABLED (1=YES)
S2.OPT='B'100                   ;SUPPORTS SEEK OPTIMIZATION (1=YES)
S2.CON='B'200                   ;SCB AND KRB ARE CONTIGUOUS (1=YES)
S2.OP1='B'400                   ;THESE TWO BITS DEFINE THE OPTIMIZATION
S2.OP2='B'1000                  ;METHOD.
                                ;OP2,OP1=0,0 INDICATES NEAREST CYLINDER
                                ;OP2,OP1=0,1 INDICATES ELEVATOR
                                ;OP2,OP1=1,0 INDICATES C-SCAN
                                ;OP2,OP1=1,1 RESERVED
S2.ACT='B'2000                  ;DRIVER HAS OPERATION OUTSTANDING (1=YES)
S2.XHR='B'4000                  ;EXTERNAL HEADER AND NEW I.LN2 SUPPORT
S2.KRQ='B'10000                 ;SCB IS QUEUED IN CONTROLLER REQUEST QUEUE

;+
; STATUS CONTROL BLOCK STATUS EXTENSION (S.ST3) DEFINITIONS
;-

S3.DRL='B'1                     ;MULTI-ACCESS DRIVE IN RELEASED STATE (1=YES)
S3.NRL='B'2                     ;DRIVER SHOULDN'T RLS MULTI-ACCESS DRIVE (1=YES)
S3.SIP='B'4                     ;SEEK IN PROGRESS (1=YES)
S3.ATN='B'10                    ;DRIVER MUST CLEAR ATTENTION BIT (1=YES)
S3.SLV='B'20                    ;DEVICE USES SLAVE UNITS (1=YES)
S3.SPA='B'40                    ;PORT 'A' SPINNING UP
S3.SPB='B'100                   ;PORT 'B' SPINNING UP
S3.OPT='B'200                   ;SEEK OPTIMIZATION ENABLED (1=YES)
S3.SPU='B'S3.SPA!S3.SPB         ;.OR. OF PORT SPINUP BITS

;+
; KRB ADDRESS TABLE (S.KTB) PORT OFFLINE FROM THIS SCB FLAG.
;-

KP.OFL='B'1                     ;KRB ADDRESS POINTS TO OFFLINE PORT (1=YES)

;+
; MAPPING ASSIGNMENT BLOCK (FOR UNIBUS MAPPING REGISTER ASSIGNMENT)
;-
```

SCBDF$ (Cont.)

```
        .ASECT
.=0
M.LNK:'L'   .BLKW 1             ;LINK WORD
M.UMRA:'L'  .BLKW 1             ;ADDRESS OF FIRST ASSIGNED UMR
M.UMRN:'L'  .BLKW 1             ;NUMBER OF UMR'S ASSIGNED * 4
M.UMVL:'L'  .BLKW 1             ;LOW 16 BITS MAPPED BY 1ST ASSIGNED UMR
M.UMVH:'L'  .BLKB 1             ;HIGH 2 BITS MAPPED IN BITS 4 AND 5
M.BFVH:'L'  .BLKB 1             ;HIGH 6 BITS OF PHYSICAL BUFFER ADDRESS
M.BFVL:'L'  .BLKW 1             ;LOW 16 BITS OF PHYSICAL BUFFER ADDRESS
M.LGTH='B'.                     ;LENGTH OF MAPPING ASSIGNMENT BLOCK

        .ENDC

        .PSECT

        .MACRO SCBDF$,X,Y,Z
        .ENDM
        .ENDM
```

SHDDF$

```
        .MACRO  SHDDF$,L,B

;
; FIRST, WE MUST DEFINE THE I/O PACKET DEFINITIONS, SINCE WE
; USE THEM IN OUR DEFINITIONS.
;
        PKTDF$                  ;DEFINE I/O PACKET DEFINITIONS

;+
; SHADOW RECORDING LINKAGE BLOCK (UMB)
;
; THE UMB LINKS TOGETHER TWO UCB'S AS A SHADOW SET.  ONE IS THE
; PRIMARY UCB, THE OTHER THE SECONDARY UCB.  THE EXISTANCE OF A
; UMB SIGNALS THAT SHADOW RECORDING IS ENABLED ON A PARTICULAR
; UCB.
;-

        .ASECT

.=0
M.LNK:'L' .BLKW 1               ;LINKAGE OF ALL UMB'S IN THE SYSTEM
M.LHD:'L' .BLKW 1               ;LISTHEAD OF ALL ML NODES FOR THIS SET
M.UCBS:'L'.BLKW 2               ;PRIMARY AND SECONDARY UCB ADDRESSES
M.STS:'L' .BLKW 1               ;STATUS WORD
M.LBN:'L' .BLKB 1               ;HIGH ORDER BYTE OF FENCE
          .BLKB 1               ;UNUSED BYTE (MAYBE STATUS?)
          .BLKW 1               ;LOW ORDER WORD OF FENCE
M.LGH=.

;+
; UMB STATUS BIT DEFINITIONS
;-

        .PSECT

MS.MDA'B'=1                     ;UMB MARKED FOR DEALLOCATION (1=YES)
MS.CHP'B'=2                     ;CATCHUP IN PROGRESS (1=YES)

;+
; DEFINE THE OFFSETS FOR THE ML NODE, LINKED OFF OF THE UMB
; THROUGH CELL M.LHD.  THIS NODE CONTAINS THE SECONDARY I/O
; PACKET, AND DOUBLES AS THE ERROR PACKET TO THE ERROR MESSAGE
; TASK.
;-

        .ASECT

.=0
ML.LNK:'L' .BLKW 1              ;LINKAGE OF ALL ML NODES ON UMB
ML.LEN:'L' .BLKB 1              ;LENGTH OF ML NODE FOR DEALLOCATION
ML.TYP:'L' .BLKB 1              ;TYPE OF ML NODE FOR ERROR TASK
ML.DNC:'L' .BLKB 1              ;DONE COUNT OF PACKETS
           .BLKB 1              ;UNUSED
ML.PRI:'L' .BLKW 1              ;PRIMARY I/O PACKET ADDRESS
ML.PKT:'L' .BLKB I.LGTH         ;SECONDARY I/O PACKET
ML.LGH=.
```

SHDDF$ (Cont.)

```
;+
; ML NODE TYPE CODES
;-

        .PSECT

MT.PKT'B'=1                     ;ML NODE IS I/O PACKET TYPE

;+
; I/O PACKET OFFSET DEFNS FOR USE BY SHADOW RECORDING
;-

I.R0'B'=I.PRM                   ;STATUS STORAGE FOR R0 STATUS
I.R1'B'=I.PRM+2                 ;STATUS STORAGE FOR R1 STATUS

;+
; DEFINE THE ERROR MESSAGE POINTERS THAT RESIDE IN THE I/O PACKET.
;-

        .PSECT

ML.FID'B'=ML.PKT+I.IOSB         ;FILE ID WHICH CONTAINS ERROR
ML.FSEQ'B'=ML.PKT+I.IOSB+2      ;FILE SEQUENCE NUMBER OF FILE IN ERROR
ML.LBN'B'=ML.PKT+I.PRM+10       ;HIGH ORDER LBN OF BLOCK(S) IN ERROR
ML.CNT'B'=ML.PKT+I.PRM+4        ;NUMBER OF BLOCKS IN BAD XFER
ML.TCB'B'=ML.PKT+I.TCB          ;TCB OF TASK WITH BAD REQUEST
ML.SR0'B'=ML.PKT+I.R0           ;R0 OF SECONDARY I/O PACKET
ML.SR1'B'=ML.PKT+I.R1           ;R1 OF SECONDARY I/O PACKET
ML.PR0'B'=ML.PKT+I.PRM+14       ;R0 OF PRIMARY I/O PACKET
ML.PR1'B'=ML.PKT+I.PRM+16       ;R1 OF PRIMARY I/O PACKET


        .MACRO  SHDDF$,X,Y,Z
        .ENDM
        .ENDM
```

TCBDF$

```
        .MACRO  TCBDF$,L,B

;+
; TASK CONTROL BLOCK OFFSET AND STATUS DEFINITIONS
;
; TASK CONTROL BLOCK
;-

        .ASECT
.=0
T.LNK:'L' .BLKW 1               ;UTILITY LINK WORD
T.PRI:'L' .BLKB 1               ;TASK PRIORITY
T.IOC:'L' .BLKB 1               ;I/O PENDING COUNT
T.PCBV:'L' .BLKW 1              ;POINTER TO COMMON PCB VECTOR
T.NAM:'L' .BLKW 2               ;TASK NAME IN RAD50
T.RCVL:'L' .BLKW 2              ;RECEIVE QUEUE LISTHEAD
T.ASTL:'L' .BLKW 2              ;AST QUEUE LISTHEAD
T.EFLG:'L' .BLKW 2              ;TASK LOCAL EVENT FLAGS 1-32
T.UCB:'L' .BLKW 1               ;UCB ADDRESS FOR PSEUDO DEVICE 'TI'
T.TCBL:'L' .BLKW 1              ;TASK LIST THREAD WORD
T.STAT:'L' .BLKW 1              ;FIRST STATUS WORD (BLOCKING BITS)
T.ST2:'L' .BLKW 1               ;SECOND STATUS WORD (STATE BITS)
T.ST3:'L' .BLKW 1               ;THIRD STATUS WORD (ATTRIBUTE BITS)
T.DPRI:'L' .BLKB 1              ;TASK'S DEFAULT PRIORITY
T.LBN:'L' .BLKB 3               ;LBN OF TASK LOAD IMAGE
T.LDV:'L' .BLKW 1               ;UCB ADDRESS OF LOAD DEVICE
T.PCB:'L' .BLKW 1               ;PCB ADDRESS OF TASK PARTITION
T.MXSZ:'L' .BLKW 1              ;MAXIMUM SIZE OF TASK IMAGE (MAPPED ONLY)
T.ACTL:'L' .BLKW 1              ;ADDRESS OF NEXT TASK IN ACTIVE LIST
T.ATT:'L' .BLKW 2               ;ATTACHMENT DESCRIPTOR LISTHEAD
T.ST4:'L' .BLKW 1               ;FOURTH TASK STATUS WORD
T.HDLN:'L' .BLKB 1              ;LENGTH OF HEADER (0 IF HDR IN POOL)
        .BLKB   1               ;UNUSED
T.GGF:'L' .BLKB 1               ;GROUP GLOBAL USE COUNT FOR TASK
T.TIO:'L' .BLKB 1               ;BUFFERED I/O IN PROGRESS COUNT
T.EFLM:'L' .BLKW 2              ;TASK WAITFOR MASK/ADDRESS
T.TKSZ:'L' .BLKW 1              ;TASK LOAD SIZE IN 32 WD BLOCKS

$$$=.                           ;MARK START OF PLAS AREA

T.OFF:'L' .BLKW 1               ;OFFSET TO TASK IMAGE IN PARTITION
        .BLKB 1                 ;RESERVED
T.SRCT:'L' .BLKB 1              ;SREF WITH EFN COUNT IN ALL RECEIVE QUEUES
T.RRFL:'L' .BLKW 2              ;RECEIVE BY REFERENCE LISTHEAD

        .IF NDF P$$LAS

.=$$$                           ;MOVE LC BACK TO START OF PLAS AREA

        .ENDC


        .IF NB  SYSDEF


$$$=.
```

TCBDF$ (Cont.)

```
T.CTX:'L' .BLKW 1               ;POINTER TO CONTEXT BLOCK (DDS)

        .IF NDF N$$DIR

.=$$$

        .ENDC   ; NDF N$$DIR

$$$=.                           ;MARK START OF PARENT/OFFSPRING AREA

T.OCBH:'L' .BLKW 2              ;OFFSPRING CONTROL BLOCK LISTHEAD
T.RDCT:'L' .BLKW 1              ;OUTSTANDING OFFSPRING AND VT: COUNT


        .IF NDF P$$OFF

.=$$$

        .ENDC


T.SAST:'L' .BLKW 1              ;SPECIFY AST LIST HEAD

$$$=.

T.RRM:'L'.BLKW  1               ;REQUIRED RUN MASK
T.IRM:'L'.BLKW  1               ;INITIAL RUN MASK SET UP BY INSTALL
                                ;**** THIS WORD IS NO LONGER NECESSARY
                                ;**** HOWEVER, INSTALL (INSLB), MCR (SPAWN),
                                ;**** AND VMR MUST BE MODIFIED
T.CPU:'L'.BLKB  1               ;PROCESSOR NUMBER ON WHICH TASK LAST EXECUTED
        .BLKB   1               ;(UNUSED)

        .IF NDF M$$PRO

.=$$$

        .ENDC


$$$=.

T.ACN:'L'.BLKW  1               ;POINTER TO ACCOUNTING BLOCK


        .IF NDF A$$CNT

.=$$$

        .ENDC


$$$=.

T.ISIZ:'L'.BLKW 1               ;SIZE OF ROOT I SPACE

        .IF NDF U$$DAS
```

TCBDF$ (Cont.)

```
.=$$$

        .ENDC   ; NDF U$$DAS

T.LGTH='B'.                     ;LENGTH OF TASK CONTROL BLOCK
T.EXT='B'0                      ;LENGTH OF TCB EXTENSION

        .IFF

;+
; TASK STATUS DEFINITIONS
;
; FIRST STATUS WORD (BLOCKING BITS)
;-

TS.EXE='B'100000                ;TASK NOT IN EXECUTION (1=YES)
TS.RDN='B'40000                 ;I/O RUN DOWN IN PROGRESS (1=YES)
TS.MSG='B'20000                 ;ABORT MESSAGE BEING OUTPUT (1=YES)
TS.CIP='B'10000                 ;TASK BLOCKED FOR CHECKPOINT IN PROGRESS (1=YES)
TS.RUN='B'4000                  ;TASK IS RUNNING ON ANOTHER PROCESSOR (1=YES)
TS.STP='B'1000                  ;TASK BLOCKED BY CLI COMMAND
TS.CKR='B'100                   ;TASK HAS CKP REQUEST (MP SYSTEM ONLY) (1=YES)
TS.BLC='B'37                    ;INCREMENT BLOCKING COUNT MASK

;+
; TASK BLOCKING STATUS MASK
;-

TS.BLK='B'177777

;+
; SECOND STATUS WORD (STATE BITS)
;-

T2.AST='B'100000                ;AST IN PROGRESS (1=YES)
T2.DST='B'40000                 ;AST RECOGNITION DISABLED (1=YES)
T2.CHK='B'20000                 ;TASK NOT CHECKPOINTABLE (1=YES)
T2.REX='B'10000                 ;REQUESTED EXIT AST SPECIFIED
T2.SEF='B'4000                  ;TASK STOPPED FOR EVENT FLAG(S) (1=YES)
T2.SIO='B'1000                  ;TASK STOPPED FOR BUFFERED I/O
T2.AFF='B'400                   ;TASK IS INSTALLED WITH AFFINITY
T2.HLT='B'200                   ;TASK IS BEING HALTED (1=YES)
T2.ABO='B'100                   ;TASK MARKED FOR ABORT (1=YES)
T2.STP='B'40                    ;SAVED T2.SPN ON AST IN PROGRESS
T2.STP='B'20                    ;TASK STOPPED (1=YES)
T2.SPN='B'10                    ;SAVED T2.SPN ON AST IN PROGRESS
T2.SPN='B'4                     ;TASK SUSPENDED (1=YES)
T2.WFR='B'2                     ;SAVED T2.WFR ON AST IN PROGRESS
T2.WFR='B'1                     ;TASK IN WAITFOR STATE (1=YES)

;+
; THIRD STATUS WORD (ATTRIBUTE BITS)
;-

T3.ACP='B'100000                ;ANCILLARY CONTROL PROCESSOR (1=YES)
T3.PMD='B'40000                 ;DUMP TASK ON SYNCHRONOUS ABORT (0=YES)
T3.REM='B'20000                 ;REMOVE TASK ON EXIT (1=YES)
T3.PRV='B'10000                 ;TASK IS PRIVILEGED (1=YES)
```

TCBDF$ (Cont.)

```
T3.MCR='B'4000                  ;TASK REQUESTED AS EXTERNAL MCR FUNCTION (1=YES)
T3.SLV='B'2000                  ;TASK IS A SLAVE TASK (1=YES)
T3.CLI='B'1000                  ;TASK IS A COMMAND LINE INTERPRETER (1=YES)
T3.RST='B'400                   ;TASK IS RESTRICTED (1=YES)
T3.NSD='B'200                   ;TASK DOES NOT ALLOW SEND DATA
T3.CAL='B'100                   ;TASK HAS CHECKPOINT SPACE IN TASK IMAGE
T3.ROV='B'40                    ;TASK HAS RESIDENT OVERLAYS
T3.NET='B'20                    ;NETWORK PROTOCOL LEVEL
T3.MPC='B'10                    ;MAPPING CHANGE WITH OUTSTANDING I/O (1=YES)
T3.CMD='B'4                     ;TASK IS EXECUTING A CLI COMMAND
T3.SWS='B'2                     ;RESERVED FOR SOFTWARE SERVICES USE
T3.GFL='B'1                     ;GROUP GLOBAL EVENT FLAG LOCK

;+
; STATUS BIT DEFINITIONS FOR FOURTH STATUS WORD (T.ST4)
;-

T4.FMP='B'200                   ;TASK HAS FAST MAP HDR EXT.
T4.CTC='B'100                   ;TASK HAS BEEN PROCESSED BY GIN ^C ABORT
T4.MUT='B'40                    ;TASK IS A MULTI-USER TASK
T4.LDD='B'20                    ;TASK'S LOAD DEVICE HAS BEEN DISMOUNTED
T4.PRO='B'10                    ;TCB IS (OR SHOULD BE) A PROTOTYPE
T4.PRV='B'4                     ;TASK WAS PRIV, BUT HAS CLEARED T3.PRV
                                ;WITH GIN (MAY RESET WITH GIN IF T4.PRV SET)
T4.DSP='B'2                     ;TASK WAS BUILT FOR USER I/D SPACE
T4.SNC='B'1                     ;TASK USES COMMONS FOR SYNCHRONIZATION

;+
; REQUIRED RUN MASK
;-

TR.UBT='B'100000                ;UNIBUS RUN T
TR.UBS='B'40000                 ;UNIBUS RUN S
TR.UBR='B'20000                 ;UNIBUS RUN R
TR.UBP='B'10000                 ;UNIBUS RUN P
TR.UBN='B'4000                  ;UNIBUS RUN N
TR.UBM='B'2000                  ;UNIBUS RUN M
TR.UBL='B'1000                  ;UNIBUS RUN L
TR.UBK='B'400                   ;UNIBUS RUN K
TR.UBJ='B'200                   ;UNIBUS RUN J
TR.UBH='B'100                   ;UNIBUS RUN H
TR.UBF='B'40                    ;UNIBUS RUN F
TR.UBE='B'20                    ;UNIBUS RUN E
TR.CPD='B'10                    ;PROCESSOR D
TR.CPC='B'4                     ;PROCESSOR C
TR.CPB='B'2                     ;PROCESSOR B
TR.CPA='B'1                     ;PROCESSOR A

        .ENDC


        .PSECT
        .MACRO  TCBDF$  X,Y,Z
        .ENDM
        .ENDM
```

UCBDF$

```
        .MACRO  UCBDF$,L,B,TTDEF

;+
; UNIT CONTROL BLOCK
;
; THE UNIT CONTROL BLOCK (UCB) DEFINES THE STATUS OF AN INDIVIDUAL DEVICE
; UNIT AND IS THE CONTROL BLOCK THAT IS POINTED TO BY THE FIRST WORD OF
; AN ASSIGNED LUN. THERE IS ONE UCB FOR EACH DEVICE UNIT OF EACH DCB. THE
; UCB'S ASSOCIATED WITH A PARTICULAR DCB ARE CONTIGUOUS IN MEMORY AND ARE
; POINTED TO BY THE DCB. UCB'S ARE VARIABLE LENGTH BETWEEN DCB'S BUT ARE
; OF THE SAME LENGTH FOR A SPECIFIC DCB. TO FINISH THE TELETYPE EXAMPLE
; ABOVE, EACH UNIT ON BOTH INTERFACES WOULD HAVE A UCB.
;-

        .ASECT

.=177772

        .IF NB  SYSDEF

        .IF DF A$$CNT
.=.-2
        .ENDC   ;DF A$$CNT

        .IF DF L$$GCL
.=.-2
        .ENDC   ;DF L$$GCL

.=.-2

        .IF DF  N$$DIR
.=.-2
        .ENDC   ;DF N$$DIR

U.UAB:'L'

        .IF DF  A$$CNT

        .BLKW   1               ;POINTER TO USER ACCOUNT BLOCK

        .ENDC   ;DF A$$CNT

U.LOG:'L'

        .IF DF  L$$GCL

        .BLKW 1         ;POINTER TO USER LOGICAL HASH TABLE

        .ENDC   ;DF L$$GCL

U.FPRO:'L'
        .BLKW 1         ;DEFAULT FILE PROTECTION WORD
```

UCBDF$ (Cont.)

```
U.CTX:'L'

        .IF DF N$$DIR

        .BLKW 1          ;POINTER TO TERMINAL CONTEXT BLOCK

        .ENDC    ;DF N$$DIR


        .ENDC    ;NB SYSDEF


U.MUP:'L' .BLKW 1               ;MULTI-USER PROTECTION WORD
U.LUIC:'L' .BLKW 1              ;LOGIN UIC - MULTI USER SYSTEMS ONLY
U.OWN:'L' .BLKW 1               ;OWNING TERMINAL - MULTI USER SYSTEMS ONLY
U.DCB:'L' .BLKW 1               ;BACK POINTER TO DCB
U.RED:'L' .BLKW 1               ;POINTER TO REDIRECT UNIT UCB
U.CTL:'L' .BLKB 1               ;CONTROL PROCESSING FLAGS
U.STS:'L' .BLKB 1               ;UNIT STATUS
U.UNIT:'L' .BLKB 1              ;PHYSICAL UNIT NUMBER
U.ST2:'L' .BLKB 1               ;UNIT STATUS EXTENSION
U.CW1:'L' .BLKW 1               ;FIRST DEVICE CHARACTERISTICS WORD
U.CW2:'L' .BLKW 1               ;SECOND DEVICE CHARACTERISTICS WORD
U.CW3:'L' .BLKW 1               ;THIRD DEVICE CHARACTERISTICS WORD
U.CW4:'L' .BLKW 1               ;FOURTH DEVICE CHARACTERISTICS WORD
U.SCB:'L' .BLKW 1               ;POINTER TO SCB
U.ATT:'L' .BLKW 1               ;TCB ADDRESS OF ATTACHED TASK
U.BUF:'L' .BLKW 1               ;RELOCATION BIAS OF CURRENT I/O REQUEST
          .BLKW 1               ;BUFFER ADDRESS OF CURRENT I/O REQUEST
U.CNT:'L' .BLKW 1               ;BYTE COUNT OF CURRENT I/O REQUEST
U.UCBX='B'U.CNT+2               ;POINTER TO UCB EXTENSION IN SECONDARY POOL
U.ACP='B'U.CNT+4                ;ADDRESS OF TCB OF MOUNTED ACP
U.VCB='B'U.CNT+6                ;ADDRESS OF VOLUME CONTROL BLOCK
U.CBF='B'U.CNT+2                ;CONTROL BUFFER RELOCATION AND ADDRESS
U.UMB='B'U.CNT+10               ;ADDRESS OF UMB FOR SHADOW RECORDING
U.PRM='B'U.CNT+12               ;DISK SIZE PARAMETER WORDS
U.ICSR='B'U.CNT+16              ;CSR ADDRESS (P/OS)
U.SLT='B'U.CNT+20               ;SLOT ADRESS (P/OS)
U.SPRM='B'U.CNT+22              ;4 WD SAVED I/O PACKET AREA (R$$AMD)
U.UTIL='B'U.CNT+16              ;STATE WORD FOR UNIT

;
; DEFINITIONS FOR U.UTIL BITS
;

UU.SER  ='B'1                   ;SERIAL MODE
UU.RCT  ='B'2                   ;(DUDRV)RCT IN PROGRESS
UU.AVN  ='B'4                   IS WAITING FOR OTHER UNITS TO SPIN DOWN
UU.GUS  ='B'10                  ;UNIT MUST HAVE A GUS COMMAND ISSUED
UU.ONL  ='B'20                  ;UNIT MUST HAVE A ONL COMMAND ISSUED
UU.SPC  ='B'40                  ;SPECIAL ONLINE TRANSITION
UU.ATN  ='B'100                 ;UNIT HAS SENT ATTENTION MESSAGE
UU.RDY  ='B'200                 ;UNIT IS READY
UU.ABO  ='B'400                 ;IF SET, XXCAN SET UU.SER FLAG FOR UNIT
UU.SIO  ='B'1000                ;THIS UNIT CAN STALL I/O
UU.IOS  ='B'2000                ;THIS UNIT HAS I/O STALLED
UU.BLK  ='B'4000                ;THIS UNIT DOESN'T ACCEPT DENSITY SETTINGS
```

UCBDF$ (Cont.)

```
U.BPKT='B'U.CNT+20              ;UNIT BAD BLOCK REPLACEMENT WAITING LIST
U.MEDI='B'U.BPKT                ;MEDIA IDENTIFIER FOR MU TAPE
U.UC2X='B'U.CNT+24              ;POINTER TO SECOND EXTENSION IN SECONDARY POOL

;
; MAGTAPE DEVICE DEPENDENT UCB OFFSETS
;

U.SNUM='B'U.CNT+10              ;SLAVE UNIT NUMBER
U.FCDE='B'U.CNT+12              ;FUNCTION CODE
U.KRB1='B'U.CNT+14              ;SUBCONTROLLER KRB1 POINTER
;

;
; DEFINE SECONDARY POOL UCB EXTENSION OFFSETS (ERROR LOGGING DEVICES ONLY)
;
.=0
        .BLKW   9.              ;FIXED ACCOUNTING TRANSACTION HEADER
X.NAME:'L' .BLKW 2              ;DRIVE NAME IN RAD50
X.IOC:'L' .BLKW 2               ;I/O COUNT
X.ERSL:'L' .BLKB 1              ;SOFT ERROR LIMIT
X.ERHL:'L' .BLKB 1              ;HARD ERROR LIMIT
X.ERSC:'L' .BLKB 1              ;SOFT ERROR COUNT
X.ERHC:'L' .BLKB 1              ;HARD ERROR COUNT
X.WCNT:'L' .BLKW 2              ;WORDS TRANSFERED COUNT

;
; DEFINE OFFSETS FOR SEEK OPTIMIZATION DEVICES
;

X.CYLC:'L' .BLKW 2              ;CYLINDERS CROSSED COUNT
X.CCYL:'L' .BLKW 1              ;CURRENT CYLINDER
X.FCUR:'L' .BLKB 1              ;CURRENT FAIRNESS COUNT
X.FLIM:'L'                      ;FAIRNESS COUNT LIMIT
X.DSKD:'L' .BLKB 1              ;DISK DIRECTION (HIGH BIT 1=OUT)

X.DNAM:'L' .BLKW 1              ;DEVICE NAME FOR ACCOUNTING
X.UNIT:'L' .BLKB 1              ;UNIT NUMBER FOR ACCOUNTING
X.CSTS:'L' .BLKB 1              ;CACHE STATUS BITS
X.CPCB:'L' .BLKW 1              ;CACHE PARTITION PCB ADDRESS
X.CSBA:'L' .BLKW 1              ;CACHE STATISTICS BUFFER ADDRESS (BIAS)
X.CCED:'L' .BLKW 2              ;CACHE EXTENT DESCRIPTOR LISTHEAD
X.XDAT:'L' .BLKB 1              ;CACHE VIRTUAL EXTENT SIZE
X.XRDA:'L' .BLKB 1              ;CACHE READAHEAD EXTENT SIZE
X.XDIR:'L' .BLKB 1              ;CACHE DIRECTORY EXTENT SIZE
X.XLOG:'L' .BLKB 1              ;CACHE LOGICAL EXTENT SIZE
X.XOVR:'L' .BLKB 1              ;CACHE OVERLAY EXTENT SIZE
           .BLKB 1              ;RESERVED
X.LGTH='B'.                     ;LENGTH OF THE UCB EXTENSION
X.DFFL='B'10.                   ;DEFAULT FAIRNESS COUNT LIMIT
X.DFSL='B'8.                    ;DEFAULT SOFT ERROR LIMIT
X.DFHL='B'5.                    ;DEFAULT HARD ERROR LIMIT

;
; CACHE STATUS BITS IN X.CSTS
;
```

UCBDF$ (Cont.)

```
XC.ENA='B'200                   ;AUTOCACHE ENABLED (1=YES)
XC.ACT='B'100                   ;CACHE ACTIVE FOR DEVICE (1=YES)
XC.DIR='B'020                   ;CACHE DIRECTORY REQUESTS (1=YES)
XC.OVR='B'010                   ;CACHE OVERLAY REQUESTS (1=YES)
XC.DAT='B'004                   ;CACHE VIRTUAL REQUESTS (1=YES)
XC.LOG='B'002                   ;CACHE LOGICAL REQUESTS (1=YES)
XC.RDA='B'001                   ;CACHE VIRTUAL READ AHEAD (1=YES)

;
; DEFINE CACHE MAXIMUM AND DEFAULT EXTENT SIZES
;

XX.MAX='B'15.                   ; MAXIMUM EXTENT SIZE
XX.DAT='B'5.                    ; DEFAULT VIRTUAL EXTENT SIZE
XX.RDA='B'5.                    ; DEFAULT READAHEAD EXTENT SIZE
XX.DIR='B'1.                    ; DEFAULT DIRECTORY EXTENT SIZE
XX.LOG='B'1.                    ; DEFAULT LOGICAL EXTENT SIZE
XX.OVR='B'4.                    ; DEFAULT OVERLAY EXTENT SIZE

;
; DEFINE OFFSETS FOR DISK MSCP CONTROLLERS (SECOND UCB EXTENSION)
;

;
; CHARACTERISTICS OBTAINED FROM "GET UNIT STATUS" END PACKETS
;

.=0
X.MLUN:'L'.BLKW 1               ;MULTI-UNIT CODE
X.UNFL:'L'.BLKW 1               ;UNIT FLAGS
          .BLKW 2               ;RESERVED
X.UNTI:'L'.BLKW 4               ;UNIT IDENTIFIER
X.MEDI:'L'.BLKW 2               ;MEDIA IDENTIFIER
X.SHUN:'L'.BLKW 1               ;SHADOW UNIT
X.SHST:'L'.BLKW 1               ;SHADOW UNIT STATUS
X.TRCK:'L'.BLKW 1               ;UNIT TRACK SIZE
X.GRP:'L'.BLKW  1               ;UNIT GROUP SIZE
X.CYL:'L'.BLKW  1               ;UNIT CYLINDER SIZE
X.USVR:'L'.BLKB 1               ;UNIT SOFTWARE VERSION
X.UHVR:'L'.BLKB 1               ;UNIT HARDWARE VERSION
X.RCTS:'L'.BLKW 1               ;UNIT RCT TABLE SIZE
X.RBNS:'L'.BLKB 1               ;UNIT RBN 'S / TRACK
X.RCTC:'L'.BLKB 1               ;UNIT RCT COPIES
;
; CHARACTERISTICS OBTAINED FROM "ONLINE" OR "SET UNIT CHARACTERISTICS" END
; PACKETS
;
X.UNSZ:'L'.BLKW 2               ;UNIT SIZE
X.VSER:'L'.BLKW 2               ;VOLUME SERIAL NUMBER

X.DUSZ='B'.                     ;SIZE OF DISK MSCP CONTROLLER UCB EXTENTION


        .IF NB  TTDEF

;
; TERMINAL DRIVER DEFINITIONS
;
```

UCBDF$ (Cont.)

```
.=U.BUF
U.TAPR:'L'                      ;APR VALUE FOR START OF UCBX
U.TUX:'L' .BLKW 1               ;POINTER TO UCB EXTENSION (UCBX)
U.TSTA:'L' .BLKW 4              ;STATUS QUADRUPLE-WORD
U.UIC:'L' .BLKW 1               ;DEFAULT UIC
U.TFRQ:'L' .BLKW 1              ;FORK REQUEST WORD
U.TFLK:'L' .BLKW 1              ;FORK LIST LINK WORD
U.TCHP:'L' .BLKB 1              ;CURRENT HORIZONTAL POSITION
U.TCVP:'L' .BLKB 1              ;CURRENT VERTICAL POSITION
U.TTYP:'L' .BLKB 1              ;TERMINAL TYPE
U.TMTI:'L' .BLKB 1              ;MODEM TIMER
U.TTAB:'L' .BLKW 1              ;IF 0: U.TTAB+1 IS SINGLE-CHARACTER TYPE-AHEAD
                                ;       BUFFER, CURRENTLY EMPTY
                                ;IF ODD: U.TTAB+1 IS SINGLE-CHARACTER TYPE-AHEAD
                                ;       BUFFER AND HOLDS A CHARACTER
                                ;IF NON-0 AND EVEN: POINTER TO MULTI-CHARACTER
                                ;       TYPE-AHEAD BUFFER
        .=.-2                   ;THE NEXT TWO OFFSETS OVERLAP U.TTAB WHEN THE
                                ;TYPEAHEAD BUFFER IS IN SECONDARY POOL
U.TECO:'L' .BLKB 1              ;ECHO BUFFER FOR DMA OPERATIONS WHEN UCBX IS
                                ;IN SECONDARY POOL AND THUS NOT MAPPED BY A UMR
U.TBSZ:'L' .BLKB 1              ;TYPEAHEAD BUFFER SIZE
U.TLPP:'L' .BLKW 1              ;LINES PER PAGE
U.TST5:'L' .BLKW 1              ;ADDITIONAL STATUS BITS
U.TST6:'L' .BLKW 1              ;EXTENDED I/O STATUS WORD
U.TIXL:'L' .BLKW 1              ;I/O PACKET EXTENSION LISTHEAD
U.ACB:'L' .BLKW 1               ;ANCILLARY CONTROL DRIVER BLOCK ADDR
U.AFLG:'L' .BLKW 1              ;ANCILLARY CONTROL DRIVER FLAGS WORD
U.ADMA:'L' .BLKW 1              ;ANCILLARY CONTROL DRIVER DMA BUFFER

        .IF DF T$$LTH

;
; LAT Host Support
;
                        ;STATUS/CONTROL INFORMATION
U.LINS:'L' .BLKB 1              ;LINK STATUS
U.CREN:'L' .BLKB 1              ;TRANSMIT CREDITS COUNTER
                        ;SERVER/CIRCUIT IDENTIFICATION
U.SRVN:'L' .BLKW 1              ;SERVER NUMBER
U.SESN:'L' .BLKB 1              ;SESSION NUMBER
U.SLSZ:'L' .BLKB 1              ;MAXIMUM SLOT SIZE ON XMT
                        ;PARAMETERS ON RECEIVE DATA
U.RSBB:'L' .BLKW 1              ;RECEIVE SLOT BIAS
U.RBHA:'L' .BLKW 1              ;RECEIVE SLOT HEADER VIRTUAL
U.RSDV:'L' .BLKW 1              ;RECEIVE DATA VIRTUAL
                        ;XMT INTERMEDIATE BUFFER CHAIN
U.TRLH:'L' .BLKW 1              ;XMT LISTHEAD ADDRESS
U.TRSC:'L' .BLKW 1              ;XMT REMAINED BYTES IN BUFFER
                        ;VIRTUAL CIRCUIT CCB
U.CCBA:'L' .BLKW 1              ;CCB ADDRESS (IN POOL)
;
; LINK STATUS WORD IN U.LINS
;
UL.TRS='B'1                     ;1-XMT STOPPED, 0-NOT
                                ;>255 CHAR. FOR SLOT. HOLD IT.
```

UCBDF$ (Cont.)

```
UL.TDA='B'2                     ;1-XMT DATA AVAIL, 0-NOT
                                ;DATA IN XMT QUEUE (SET BY PORT)
UL.LST='B'4                     ;1-LINE STOPPED, 0-NOT
                                ;LINE STOPPED BY USER (XOFF
                                ;CAME FROM REMOTE TERMINAL)
UL.RDA='B'10                    ;1-RCV DATA AVAIL, 0-NOT
                                ;DATA IN RCV QUEUE (SET BY PROC)
UL.ECH='B'20                    ;BUFFER ALLOCATION FAILURE
                                ;FOR 1 ECHO CHARACTER
UL.RSS='B'100                   ;1-RCV STOPPED, 0-NOT
                                ;RCV STOPPED BY TTDRV ON RECEIVE
                                ;IF NO RESOURCES AVAILABLE
UL.LEN='B'200                   ;1-LINK ENABLE, 0-NOT
                                ;TOGGLED BY START/STOP LINK CALL
        .ENDC   ; .IF DF T$$LTH
;
; DEFINE BITS IN STATUS WORD 1 (U.TSTA)
;
;
; I N P U T   S T A T U S
;
S1.RST='B'1                     ;READ WITH SPECIAL TERMINATORS IN PROGRESS
S1.ESC='B'2                     ;ESCAPE SEQUENCE IN PROGRESS
S1.RSP='B'4                     ;READ WITH SPECIAL PROCESSING
S1.PTH='B'10                    ;PASS THRU IS CURRENTLY ACTIVE
S1.RNE='B'20                    ;ECHO SUPPRESSED
S1.TSY='B'40                    ;TERMINAL OUTPUT SYNC IS CURRENTLY ENABLED
S1.OBY='B'100                   ;OUTPUT BUSY
S1.IBY='B'200                   ;INPUT BUSY
S1.DPR='B'400                   ;DEFER PROCESSING OF CHAR. IN U.TECB
S1.DEC='B'1000                  ;DEFER ECHO OF CHAR. IN U.TECB
S1.IBF='B'2000                  ;BUFFERED INPUT IN PROGRESS
S1.DSI='B'4000                  ;INPUT PROCESSING DISABLED
S1.RES='B'10000                 ;ESC. SEQ PROCESSING IS ENABLED FOR THE CURRENT
                                ; READ
S1.RNF='B'20000                 ;READ NO FILTER IS ACTIVE (EDIT CHARACTERS ARE
                                ; DISPLAYED)
S1.TNE='B'40000                 ;TERMINATOR NO ECHO
S1.USI='B'100000                ;UNSOLICITED INPUT IN PROGRESS

; DEFINE BITS IN STATUS WORD 2 (U.TSTA+2)
;
;    O U T P U T   S T A T U S
;
S2.RCU='B'1                     ;RESTORE CURSOR (MUST = TF.RCU)
S2.WRA='B'6                     ;CONTEXT FOR WRAP-AROUND
S2.WRB='B'2                     ;LOW BIT IN S2.WRA BIT PATTERN
S2.WAL='B'10                    ;WRITE PASS ALL (MUST = TF.WAL)
S2.BRQ='B'20                    ;BREAK-THROUGH-WRITE REQUEST IN QUEUE
S2.SRQ='B'40                    ;SPECIAL REQUEST IN QUEUE
                                ;(IO.ATT, IO.DET, SF.SMC)
S2.ORQ='B'100                   ;OUTPUT REQUEST IN QUEUE (MUST = S1.OBY)
S2.IRQ='B'200                   ;INPUT REQUEST IN QUEUE (MUST = S1.IBY)
S2.FLF='B'400                   ;FORCE LINEFEED BEFORE NEXT ECHO
S2.ELF='B'1000                  ;EAT A LINEFEED (IGNORE A LEADING LF ON OUTPUT)
S2.CR='B'2000                   ;TRAILING CR REQUIRED ON OUTPUT
S2.OBF='B'4000                  ;BUFFERED OUTPUT IN PROGRESS
```

UCBDF$ (Cont.)

```
S2.PCU='B'10000                 ;POSITION CURSOR BEFORE WRITE
S2.BEL='B'20000                 ;BELL PENDING
S2.CTO='B'40000                 ;OUTPUT STOPPED BY CTRL-O 266.
S2.CTS='B'100000                ;OUTPUT STOPPED BY CTRL-S

; DEFINE BITS IN STATUS WORD 3 (U.TSTA+4)
;
;   TERMINAL OPERATION CHARACTERISTICS
;
S3.ACR='B'1                     ;WRAP-AROUND (AUTOMATIC CR-LF) REQUIRED
S3.TAB='B'2                     ;TYPE-AHEAD BUFFER ALLOCATION REQUESTED
S3.CTC='B'4                     ;TERMINAL WANTS CLI TO HAVE ^C NOTIFICATION
S3.RAL='B'10                    ;TERMINAL IS IN READ-PASS-ALL MODE
S3.NEC='B'20                    ;NO ECHO
S3.TSY='B'40                    ;TERMINAL SYNC
S3.8BC='B'100                   ;PASS 8 BITS ON INPUT
S3.FDX='B'200                   ;LINE IS IN FULL DUPLEX MODE
S3.MHE='B'400                   ;NOTIFY ATTACHED TASK OF MODEM HANG-UP
S3.ICE='B'1000                  ;INPUT COUNT STATE ENABLED
S3.TME='B'2000                  ;TERMINAL MANAGEMENT MODE ENABLED
S3.PTH='B'4000                  ;PASS THROUGH REQUESTED
S3.RES='B'10000                 ;TASK WANTS ESCAPE SEQUENCES
S3.PPT='B'20000                 ;TERMINAL HAS PRINTER PORT
S3.RUB='B'40000                 ;RUBOUT SEQUENCE IN PROGRESS (NON-SCOPE)

;
; DEFINE BITS IN STATUS WORD 4 (U.TSTA+6)
;
;
; TERMINAL ATTRIBUTE CHARACTERISTICS
;
S4.HFL='B'7                     ;HORIZONTAL FILL REQUIREMENT
S4.VFL='B'10                    ;VERTICAL FILL REQUIREMENT
S4.HFF='B'20                    ;HARDWARE FORM-FEED PRESENT
S4.HHT='B'40                    ;HARDWARE HORIZONTAL TAB PRESENT
S4.DLO='B'100                   ;DIAL-OUT LINE (IMPLIES U2.RMT)
S4.HSY='B'200                   ;HOST/TERMINAL SYNCHRONIZATION ENABLED (1=YES)
S4.ANI='B'400                   ;ANSI CRT TERMINAL
S4.AVO='B'1000                  ;VT100-FAMILY TERMINAL DISPLAY
S4.BLK='B'2000                  ;BLOCK MODE TERMINAL
S4.DEC='B'4000                  ;DIGITAL CRT TERMINAL
S4.EDT='B'10000                 ;TERMINAL HAS LOCAL EDITING FUNCTIONS
S4.RGS='B'20000                 ;TERMINAL SUPPORTS REGIS GRAPHICS
S4.SFC='B'40000                 ;TERMINAL SUPPORTS SOFT CHARACTERS (DRCS)
S4.ABD='B'100000                ;AUTO-BAUD SPEED DETECTION ENABLED
;
;
; DEFINE BITS IN STATUS WORD U.TST5
;
;  ADDITIONAL STATUS CHARACTERISTICS
;
;
S5.SW1='B'1                     ;FIRST TERMINAL MANAGEMENT SWITCH
                                ;CHARACTER HAS BEEN SEEN
S5.TMM='B'2                     ;TERMINAL IN TERMINAL MANAGEMENT MODE
S5.XOF='B'4                     ;SEND AN XOFF AT FIRST OPPORTUNITY
S5.XON='B'10                    ;SEND AN XON AT FIRST OPPORTUNITY
S5.HPC='B'14                    ;OUTPUT OF HIGH PRIORITY CHARACTERS REQUESTED
```

UCBDF$ (Cont.)

```
S5.HPO='B'20                    ;HIGH PRIORITY OUTPUT IN PROGRESS
S5.OXF='B'40                    ;XOFF HAS BEEN OUTPUT
S5.ITI='B'100                   ;IMMEDIATE TIMEOUT ON INPUT
;
S5.RPO='B'2000                  ;READ W/PROMPT OUTPUT IN PROGRESS
S5.VER='B'10000                 ;LAST CHAR. IN TYPE-AHEAD BUFFER
                                ;HAS PARITY ERROR
S5.BCC='B'20000                 ;LAST CHAR. IN TYPE-AHEAD BUFFER
                                ;HAS FRAMING ERROR
S5.DAO='B'40000                 ;LAST CHAR. IN TYPE-AHEAD BUFFER
                                ;HAS DATA OVERRUN ERROR
                                ;NOTE - THE 3 BITS ABOVE MUST CORRESPOND
                                ;TO THE RESPECTIVE ERROR FLAGS IN THE
                                ;HARDWARE RECEIVE BUFFER
S5.ABP='B'100000                ;AUTO-BAUD SPEED DETECTION IN PROGRESS
;
;
; DEFINE BITS IN EXTENDED I/O STATUS WORD U.TST6
;
;
S6.EIO='B'400                   ;READ WAS AN EXTENDED I/O
S6.RLU='B'1000                  ;READ WITH LOWER CASE TO UPPER CASE CONVERSION
S6.RDI='B'100000                ;READ WITH DEFAULT INPUT

;

        .ENDC

;
; VIRTUAL TERMINAL UCB DEFINITIONS
;

.=U.UNIT
U.OCNT:'L'.BLKB 1               ;OFFSPRING WITH THIS AS TI:
.=U.BUF
U.RPKT:'L'.BLKW 1               ;CURRENT OFFSPRING READ I/O PACKET
U.WPKT:'L'.BLKW 1               ;CURRENT OFFSPRING WRITE I/O PACKET
U.IAST:'L'.BLKW 1               ;INPUT AST ROUTINE ADDRESS
U.OAST:'L'.BLKW 1               ;OUTPUT AST ROUTINE ADDRESS
U.AAST:'L'.BLKW 1               ;ATTACH AST ROUTINE ADDRESS


        .IF NB  TTDEF

.IIF NE U.AAST+2-U.UIC  .ERROR  ;ADJACENCY ASSUMED

        .ENDC


.=U.AAST+4
U.PTCB:'L'.BLKW 1               ;PARENT TCB ADDRESS

;
; CONSOLE DRIVER DEFINITIONS
;

.=U.BUF+2
U.CTCB:'L' .BLKW 1              ;ADDRESS OF CONSOLE LOGGER TCB
U.COTQ:'L' .BLKW 2              ;I/O PACKET LIST QUEUE
U.RED2:'L' .BLKW 1              ;REDIRECT UCB ADDRESS
```

UCBDF$ (Cont.)

```
        .PSECT

;+
; DEVICE TABLE STATUS DEFINITIONS
;
; DEVICE CHARACTERISTICS WORD 1 (U.CW1) DEVICE TYPE DEFINITION BITS.
;-

DV.REC='B'1                     ;RECORD ORIENTED DEVICE (1=YES)
DV.CCL='B'2                     ;CARRIAGE CONTROL DEVICE (1=YES)
DV.TTY='B'4                     ;TERMINAL DEVICE (1=YES)
DV.DIR='B'10                    ;FILE STRUCTURED DEVICE (1=YES)
DV.SDI='B'20                    ;SINGLE DIRECTORY DEVICE (1=YES)
DV.SQD='B'40                    ;SEQUENTIAL DEVICE (1=YES)
DV.MSD='B'100                   ;MASS STORAGE DEVICE (1=YES)
DV.UMD='B'200                   ;USER MODE DIAGNOSTICS SUPPORTED (1=YES)
DV.MBC='B'400                   ;MASSBUS CONTROLLER (11M COMPATIBILITY ONLY)
DV.EXT='B'400                   ;UNIT ON EXTENDED 22-BIT UNIBUS CNTROLER (1=YES)
DV.SWL='B'1000                  ;UNIT SOFTWARE WRITE LOCKED (1=YES)
DV.ISP='B'2000                  ;INPUT SPOOLED DEVICE (1=YES)
DV.OSP='B'4000                  ;OUTPUT SPOOLED DEVICE (1=YES)
DV.PSE='B'10000                 ;PSEUDO DEVICE (1=YES)
DV.COM='B'20000                 ;DEVICE IS MOUNTABLE AS COM CHANNEL (1=YES)
DV.F11='B'40000                 ;DEVICE IS MOUNTABLE AS F11 DEVICE (1=YES)
DV.MNT='B'100000                ;DEVICE IS MOUNTABLE (1=YES)

;+
; TERMINAL DEPENDENT CHARACTERISTICS WORD 2 (U.CW2) BIT DEFINITIONS
;-

U2.DH1='B'100000                ;UNIT IS A MULTIPLEXER (1=YES)
U2.DJ1='B'40000                 ;UNIT IS A DJ11 (1=YES)
U2.RMT='B'20000                 ;UNIT IS REMOTE (1=YES)
U2.HFF='B'10000                 ;UNIT HANDLES HARDWARE FORM FEEDS (1=YES)
U2.L8S='B'10000                 ;OLD NAME FOR U2.HFF
U2.NEC='B'4000                  ;DON'T ECHO SOLICITED INPUT (1=YES)
U2.CRT='B'2000                  ;UNIT IS A CRT (1=YES)
U2.ESC='B'1000                  ;UNIT GENERATES ESCAPE SEQUENCES (1=YES)
U2.LOG='B'400                   ;USER LOGGED ON TERMINAL (0=YES)
U2.SLV='B'200                   ;UNIT IS A SLAVE TERMINAL (1=YES)
U2.DZ1='B'100                   ;UNIT IS A DZ11 (1=YES)
U2.HLD='B'40                    ;TERMINAL IS IN HOLD SCREEN MODE (1=YES)
U2.AT.='B'20                    ;MCR COMMAND AT. BEING PROCESSED (1=YES)
U2.PRV='B'10                    ;UNIT IS A PRIVILEGED TERMINAL (1=YES)
U2.L3S='B'4                     ;UNIT IS A LA30S TERMINAL (1=YES)
U2.VT5='B'2                     ;UNIT IS A VT05B TERMINAL (1=YES)
U2.LWC='B'1                     ;LOWER CASE TO UPPER CASE CONVERSION (0=YES)

;+
; BIT DEFINITIONS FOR U.MUP
;-
```

UCBDF$ (Cont.)

```
UM.OVR='B'1                     ;OVERRIDE CLI INDICATOR
UM.CLI='B'36                    ;CLI INDICATOR BITS
UM.DSB='B'200                   ;TERMINAL DISABLED SINCE CLI ELIMINATED
UM.NBR='B'400                   ;NO BROADCAST
UM.CNT='B'1000                  ;CONTINUATION LINE IN PROGRESS
UM.CMD='B'2000                  ;COMMAND IN PROGRESS
UM.SER='B'4000                  ;SERIAL COMMAND RECOGNITION ENABLED
UM.KIL='B'10000                 ;TTDRV SHOULD SEND KILL PKT ON CNTRL/C

;+
; RH11-RS03/RS04 CHARACTERISTICS WORD 2 (U.CW2) BIT DEFINITIONS
;-
U2.R04='B'100000                ;UNIT IS A RS04 (1=YES)

;+
; RH11-TU16 CHARACTERISTICS WORD 2 (U.CW2) BIT DEFINITIONS
;-

U2.7CH='B'10000                 ;UNIT IS A 7 CHANNEL DRIVE (1=YES)

;+
; TERMINAL DEPENDENT CHARACTERISTICS WORD 3 (U.CW3) BIT DEFINITIONS
;-

U3.UPC='B'20000                 ;UPCASE OUTPUT FLAG
U3.PAR='B'40000                 ;PARITY GENERATION AND CHECKING
U3.OPA='B'100000                ;PARITY SENSE (1=ODD PARITY)

;+
; VIRTUAL TERMINAL 3RD CHARACTERISTICS WORD DEFINITIONS
;-

U3.FDX='B'1                     ;FULL DUPLEX MODE (1=YES)
U3.DBF='B'2                     ;INTERMEDIATE BUFFERING DISABLED (1=YES)
U3.RPR='B'4                     ;READ W/PROMPT IN PROGRESS (1=YES)

;+
; TERMINAL DEPENDENT CHARACTERISTICS WORD 4 (U.CW4) BIT DEFINITIONS
;-

U4.CR='B'100                    ;LOOK FOR CARRIAGE RETURN

;+
; UNIT CONTROL PROCESSING FLAG DEFINITIONS
;-

UC.ALG='B'200                   ;BYTE ALIGNMENT ALLOWED (1=NO)
UC.NPR='B'100                   ;DEVICE IS AN NPR DEVICE (1=YES)
UC.QUE='B'40                    ;CALL DRIVER BEFORE QUEUING (1=YES)
UC.PWF='B'20                    ;CALL DRIVER AT POWERFAIL ALWAYS (1=YES)
UC.ATT='B'10                    ;CALL DRIVER ON ATTACH/DETACH (1=YES)
UC.KIL='B'4                     ;CALL DRIVER AT I/O KILL ALWAYS (1=YES)
UC.LGH='B'3                     ;TRANSFER LENGTH MASK BITS

;+
; UNIT STATUS BIT DEFINTIONS
;-
```

UCBDF$ (Cont.)

```
US.BSY='B'200                   ;UNIT IS BUSY (1=YES)
US.MNT='B'100                   ;UNIT IS MOUNTED (0=YES)
US.FOR='B'40                    ;UNIT IS MOUNTED AS FOREIGN VOLUME (1=YES)
US.MDM='B'20                    ;UNIT IS MARKED FOR DISMOUNT (1=YES)
US.PWF='B'10                    ;POWERFAIL OCCURED (1=YES).

;+
; CARD READER DEPENDENT UNIT STATUS BIT DEFINITIONS
;-

US.ABO='B'1                     ;UNIT IS MARKED FOR ABORT IF NOT READY (1=YES)
US.MDE='B'2                     ;UNIT IS IN 029 TRANSLATION NODE (1=YES)

;+
; FILES-11 DEPENDENT UNIT STATUS BITS
;-

US.WCK='B'10                    ;WRITE CHECK ENABLED (1=YES)
US.SPU='B'2                     ;UNIT IS SPINNING UP (1=YES)
US.VV='B'1                      ;VOLUME VALID IS SET (1=YES)

;+
; TERMINAL DEPENDENT UNIT STATUS BIT DEFINITIONS
;-

US.CRW='B'4                     ;UNIT IS WAITING FOR CARRIER (1=YES)
US.DSB='B'2                     ;UNIT IS DISABLED (1=YES)
US.OIU='B'1                     ;OUTPUT INTERRUPT IS UNEXPECTED ON UNIT (1=YES)

;+
; LPS11 DEPENDENT UNIT STATUS BIT DEFINITIONS
;-

US.FRK='B'2                     ;FORK IN PROGRESS (1=YES)
US.SHR='B'1                     ;SHAREABLE FUNCTION IN PROGRESS (0='B'YES)

;+
; ANSI MAGTAPE DEPENDENT UNIT STATUS BITS
;-
US.LAB='B'4                     ; UNIT HAS LABELED TAPE ON IT (1=YES)

;+
; UNIT STATUS EXTENSION (U.ST2) BIT DEFINITIONS
;-

US.OFL='B'1                     ;UNIT OFFLINE (1=YES)
US.RED='B'2                     ;UNIT REDIRECTABLE (0=YES)
US.PUB='B'4                     ;UNIT IS PUBLIC DEVICE (1=YES)
US.UMD='B'10                    ;UNIT ATTACHED FOR DIAGNOSTICS (1=YES)
US.PDF='B'20                    ;PRIVILEGED DIAGNOSTIC FUNCTIONS ONLY (1=YES)
US.MUN='B'40                    ;MULTI-UNIT FLAG
US.TRN='B'100                   ;UNIT TRANSITION HAS OCCURRED (1=YES)
US.SIO='B'200                   ;STALL I/O TO UNIT (1=YES)
;+
; MAGTAPE DENSITY SUPPORT DEFINITION IN U.CW3
;-
```

UCBDF$ (Cont.)

```
UD.UNS='B'0                     ; UNSUPPORTED
UD.200='B'1                     ; 200BPI, 7 TRACK
UD.556='B'2                     ; 556BPI, 7 TRACK
UD.800='B'3                     ; 800BPI, 7 OR 9 TRACK
UD.160='B'4                     ;1600BPI, 9 TRACK
UD.625='B'5                     ;6250BPI, 9 TRACK
UD.8K='B'6                      ;8K BPI - SERIAL, SERPENTINE RECORDING.

        .MACRO  UCBDF$,X,Y,Z,A
        .ENDM
        .ENDM
```

APPENDIX D

MICRO/RSX COMMON ERROR CODE DEFINITIONS

This appendix lists:

1. Facility-independent error code definitions

2. Standard Bugcheck formats for facility-defined error codes

```
;       ------------------------------------------------
;       Common (Facility-Independent) Error Code Definitions
;       ------------------------------------------------

;               SST-Type Errors - Error Code 1

BE.ODD = 000100     ; Odd address or other trap four
BE.SGF = 000102     ; Segment fault
BE.BPT = 000104     ; Breakpoint or T-bit trap
BE.IOT = 000106     ; IOT instruction
BE.ILI = 000110     ; Illegal instruction
BE.EMT = 000112     ; EMT instruction
BE.TRP = 000114     ; Trap instruction
BE.STK = 000116     ; Stack overflow

;               Internal Inconsistency Errors - Error Code 2

BE.NPA = 000200     ; Task with no parent aborted
BE.SGN = 000201     ; Feature not included in system
BE.2FR = 000202     ; Double fork
BE.ISR = 000203     ; Interrupt service routine clobbered register
BE.FHW = 000204     ; Fatal hardware error
BE.CSR = 000205     ; Device CSR disappeared
BE.IDC = 000206     ; Internal database consistency error
BE.ACP = 000207     ; ACP task aborted
BE.HSP = 000210     ; Header subpacket problem
BE.NCT = 000211     ; No current task

;               System Pool Related Errors - Error Code 3

BE.NPL = 000300     ; No pool for operation
BE.DDA = 000301     ; Double deallocation
BE.SIZ = 000302     ; Size of block invalid
BE.BAK = 000303     ; Deallocated block below pool
BE.POV = 000304     ; Deallocation overlaps end of pool

;               Group-global Event Flag Errors - Error code 4

BE.GGF = 000400     ; Task locked to non-existent flags
```

```
;       ------------------------------------------------
;       Standard Bugcheck Format Facility Code Definitions
;       ------------------------------------------------

;       I/O Driver Subsystem - Facility Code 2

BF.TTD = 000200     ; Terminal driver

;       Executive Components - Facility Code 3

BF.EXE = 000300     ; Exec - General and miscellaneous
BF.XDT = 000301     ; Exec - Executive Debugging Tool
BF.POL = 000303     ; Exec - Pool handling routines (CORAL)
BF.ERR = 000304     ; Exec - Hardware error processing subsystem
BF.INT = 000305     ; Exec - Internal consistency checking routine
BF.INI = 000306     ; Exec - INITL - initialization module
BF.DVI = 000307     ; Exec - DVINT common interrupt handler
BF.PAR = 000310     ; Exec - Parity memory support
BF.XIT = 000311     ; Exec - Task exit/abort procesing
BF.QIO = 000312     ; Exec - QIO directive
BF.OPT = 000313     ; Exec - Seek optimization
BF.ACC = 000314     ; Exec - System resource accounting
BF.KAS = 000315     ; Exec - Kernel AST support
BF.DIR = 000316     ; Exec - Miscellaneous directives
BF.SAN = 000317     ; Exec - Crash with sanity timer message
```

INDEX

RSX-11M/M-PLUS
and Micro/RSX
Crash Dump Analyzer
Reference Manual
AA-FD11A-TC

READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well organized? Please make suggestions for improvement.

_____

_____

_____

_____

_____

_____

_____

_____

_____

Did you find errors in this manual? If so, specify the error and the page number.

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify) _____

Name _____ Date _____

Organization _____

Street _____

City _____ State _____ Zip Code _____
or Country

— — Do Not Tear - Fold Here and Tape —

digital

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

SSG PUBLICATIONS ZK1-3/J35
DIGITAL EQUIPMENT CORPORATION
110 SPIT BROOK ROAD
NASHUA, NEW HAMPSHIRE 03062-2698

— — Do Not Tear - Fold Here —

Cut Along Dotted Line

Printed in U.S.A.