KAV30

digital

Software Installation and System Testing Information

Part Number: AA–PEYAA–TE

KAV30

Software Installation and System Testing Information

Order Number: AA–PEYAA–TE

July 1991

This guide describes how to install the KAV30 software. It also describes the KAV30 system exerciser.

Revision Information:	This is a new guide.
Operating System and Version:	VMS Version 5.0 or higher, VAXELN Version 4.2 or higher.
Software Version:	VAXELN KAV Toolkit Extensions for VMS Version 1.0

Digital Equipment Corporation
Maynard, Massachusetts

July 1991

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.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

© Digital Equipment Corporation 1991.

All Rights Reserved.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: CVAX, DEC, DECnet, Digital, VAX Ada, VAX C, VAX FORTRAN, VAX–11/780, VAXcluster, VAXELN, VAXELN Pascal, VMS, and the DIGITAL logo.

68000 is a registered trademarks of Motorola, Inc. Intel is a trademark of Intel Corporation.

This document was prepared using VAX DOCUMENT, Version 2.0.

Contents

Figures

Tables

Preface

This guide describes how to install the VAXELN™ KAV Toolkit Extensions for VMS™ (KAV30 software). It also describes the KAV30 system exerciser.

This guide applies to Version 1.0 of the KAV30 software and all the subsequent maintenance releases up to the next major product release.

Who Should Read This Guide

This guide is for the following people:

- System managers who want to install the KAV30 software
- KAV30 system exerciser users

To install the KAV30 software, you must be familiar with the VMS operating system. To use the KAV30 system exerciser, you must be familiar with VAXELN, the VMEbus environment, and the KAV30 hardware.

Structure of This Guide

This guide is divided into four chapters, two appendixes, a glossary, and an index:

- Chapter 1 describes how to prepare to install the KAV30 software.
- Chapter 2 describes how to install the KAV30 software.
- Chapter 3 describes what you can do after you install the KAV30 software.
- Chapter 4 describes the KAV30 system exerciser.
- Appendix A contains a sample KAV30 software installation.
- Appendix B lists the files that the KAV30 software installation modifies and installs on the system.
- The glossary defines some important terms used in this guide.

Associated Documents

For more information, see the following documents:

- *KAV30 Software Product Description* (AE-PFB5A-TE)
- *KAV30 System Support Addendum* (AE-PFB6A-TE)
- *KAV30 Software Cover Letter* (AV-PEYFA-TE)
- *KAV30 Programmer's Reference Information* (AA-PEYCA-TE)
- *KAV30 Hardware Cover Letter* (AV-PFSSA-TE)
- *KAV30 Hardware Installation and User's Information* (AA-PFM6A-TE)

Conventions

The following conventions are used in this guide:

Convention	Description
Note	A note contains information that is of special importance to the reader.
Ctrl/*x*	Ctrl/*x* indicates that you hold down the Ctrl key while you press another key (indicated here by *x*).
[]	Brackets enclose optional command arguments in command descriptions. Brackets are also used in the syntax of a directory name in a VMS file specification.
italic type	Italic type emphasizes important information and complete titles of manuals.
UPPERCASE	Words in uppercase indicate a command, the name of a file, or an abbreviation for a system privilege.
`Monospace type`	Monospace type indicates both system displays and user input.

1

Preparing for a KAV30 Software Installation

This chapter describes how to prepare for a KAV30 software installation. It gives information on the following:

- Checking installation requirements
- Backing up the system disk
- Registering the license

The bill of materials (BOM) and Indented Bills Report (BIL) specify the number and contents of the media. Verify the contents of the kit with this information. If the kit is damaged, or if you find that parts of it are missing, contact your Digital™ Equipment Corporation representative.

1.1 Checking Installation Requirements

This section gives the KAV30 software installation requirements.

1.1.1 Software Requirements

You must have the following software on the system:

- VMS Version 5.0 or higher
- VAXELN Version 4.2 or higher

1.1.2 Hardware Requirements

You must have either a hardcopy or a video terminal. Use the terminal to communicate with the operating system and to respond to installation prompts.

1.1.3 Privilege and Disk Space Requirements

To install the KAV30 software, you must login to an account that has the SETPRV or SYSPRV privilege. Note that VMSINSTAL turns off the BYPASS privilege at the start of the installation.

You must have read (R) and write (W) access to the ELN$ directory.

You must have a certain amount of free disk space while installing the KAV30 software. After you install the KAV30 software, you need less space. Table 1–1 summarizes the KAV30 software disk space requirements.

Table 1–1 Disk Space Requirements

Kit	Blocks During Installation	Blocks After Installation
KAV30 software Version 1.0	2,500	800

To determine the number of free disk blocks on the current disk, enter the following command at the DCL prompt:

```
$ SHOW DEVICE SYS$SYSDEVICE
```

1.1.4 System Parameter Requirements

The installation account does not require nonstandard system parameter settings. Standard system parameter settings are those that Digital ships with the system account.

1.1.5 Process Account Quota Requirements

The installation account does not require nonstandard process account quotas. Standard process account quotas are those that Digital ships with the system account.

1.1.6 VMSINSTAL Requirements

When you invoke VMSINSTAL, it checks the following:

- If you have set the default device and directory to SYS$UPDATE
- If you are logged in to a privileged account
- If the installation account has adequate process account quotas
- If DECnet™ is running
- If other users are logged into the system

VMSINSTAL requires the installation account to have the following minimum quotas:

ASTLM = 24
BIOLM = 18
BYTLM = 18,000
DIOLM = 18
ENQLM = 30
FILLM = 20

1.2 Backing Up The System Disk

This section gives guidelines on backing up the system disk. Digital recommends that you back up the system disk before installing software. Use the backup procedures that are established at your site. For details on performing a system disk backup, see the Backup Utility manual in the VMS System Management Subkit.

1.3 Registering the License

This section gives guidelines on registering the license. Register and load the KAV30 software license before installing the KAV30 software. Use the VMS license management facility (LMF) to register and load the license. This kit includes a product authorization key (PAK). The PAK contains the information that you must have to register the license.

When you plan to use the KAV30 software from more than one node in a VAXcluster™, follow these steps:

1. Before you install the KAV30 software, register the license on the node from which you are installing
2. After you install the KAV30 software, register the license on the nodes from which you plan to use the KAV30 software

See the License Management Utility manual in the VMS documentation set for complete information on using LMF.

2

Installing the KAV30 Software

This chapter describes how to install the KAV30 software. It gives information on the following:

- Installation procedure
- Error recovery

Before you install the KAV30 software, read Chapter 1, which describes how to prepare for the installation.

The installation takes between five and 20 minutes, depending on the type of media and the system configuration. Installing the KAV30 software and running the Installation Verification Procedure (IVP) on a standalone VAX–11/780™ system takes seven minutes.

2.1 Installation Procedure

This section describes the KAV30 software installation procedure. The KAV30 software installation consists of a series of questions and informational messages. See Appendix A for a sample installation.

Note

When you press Ctrl/Y, the installation deletes all the files that it has created up to that point, and exits.

To install the KAV30 software, follow these steps:

1. **Invoke VMSINSTAL**

```
$ @SYS$UPDATE:VMSINSTAL saveset-name device-name options
```

To start the installation, invoke VMSINSTAL from a privileged account, such as the SYSTEM account. VMSINSTAL is in the SYS$UPDATE directory.

The following list defines the VMSINSTAL parameters:

saveset-name
The name of the saveset that you want to install. For Version 1.0 of the KAV30 software, use the following saveset name:

```
KAV010
```

device-name
The name of the device on which you want to mount the media containing the KAV30 software. For example, MKB500: is the device name for a tape drive. It is not necessary to use the console drive for this installation. However, if you do use the console drive, replace removed media after the installation.

options
An optional parameter specifying the VMSINSTAL options. You can specify the following VMSINSTAL options:

- Release_note option (N): Displays the release notes menu.
- Auto_answer option (A): Creates an answer file. The answer file contains your answers to VMSINSTAL questions. You can use this option (and an answer file) to save time during a reinstallation.
- Get saveset option (G): Temporarily stores product savesets on a magnetic tape or in a disk directory.
- File log option (L): Logs installation actions on the screen.
- Alternate root option (R): Installs the product to a system root other than that of the running system.

To specify an option, type OPTIONS and the letter representing the option you want to specify. For example, to specify the release notes option, type:

```
OPTIONS N
```

To specify more than one option, type OPTIONS and the letters representing options that you want to specify. Separate the options with commas and enclose the options in parentheses. For example, to specify the release notes and the log file options, type:

```
OPTIONS (N,L)
```

See the VMS System Management Subkit for more information on the options parameter.

The following example invokes VMSINSTAL to install the KAV30 software from tape drive MKB500: and shows the system response. This example uses the OPTIONS N parameter.

```
$ @SYS$UPDATE:VMSINSTAL KAV010 MKB500:

 VAX/VMS Software Product Installation Procedure V5.4-2

It is 6-MAY-1991 at 10:41.

Enter a question mark (?) at any time for help.
```

2. **Confirm DECnet status**

```
%VMSINSTAL-W-DECNET, Your DECnet network is up and running.
   .
   .
   .
* Do you want to continue anyway [NO]?
```

When you install with VMS Version 5.1 or lower, VMSINSTAL notifies you if DECnet is running and displays a list of the active processes. It then asks if you want to continue the installation. Digital recommends that you install the KAV30 software on a standalone system or cluster when DECnet is shut down. NO is the default response to the question. Press Return to choose this default response.

When you install with VMS Version 5.2 or higher, VMSINSTAL does not check the DECnet status. You can install the KAV30 software while DECnet is up and running.

3. **Confirm system disk backup**

```
* Are you satisfied with the backup of your system disk [YES]?
```

VMSINSTAL asks if you are satisfied with the system disk backup:

- Press Return when you are satisfied and want to continue the installation.
- Type NO and press Return when you are not satisfied and want to stop the installation. After you back up the system disk, restart the installation.

4. **Read informational messages**

```
The following products will be processed:

  KAV V1.0

 Beginning installation of KAV V1.0 at 10:41

%VMSINSTAL-I-RESTORE, Restoring product save set A ...
```

VMSINSTAL displays informational messages.

5. **Choose a release notes option**

   If you specified OPTIONS N at the start of the installation, VMSINSTAL now offers you the release notes options.

   When you install with VMS Versions 5.0 or 5.1, VMSINSTAL displays the following options:

```
Release Notes Options:

 1.  Display release notes
 2.  Print release notes
 3.  Both 1 and 2
 4.  Copy release notes to SYS$HELP
 5.  Do not display, print or copy release notes

* Select option [2]:
```

   When you install with VMS Version 5.2 or higher, VMSINSTAL displays the following options:

```
Release notes included with this kit are always copied to SYS$HELP.

Additional Release Notes Options:

 1.  Display release notes
 2.  Print release notes
 3.  Both 1 and 2
 4.  None of the above

* Select option [2]:
```

   Choose option 1 to display the release notes on the terminal. Press Ctrl/C to clear release notes from the screen.

   Choose option 2 to print the release notes. VMSINSTAL prompts you for a print queue name, as follows:

```
* Queue name [SYS$PRINT]:
```

   Take one of the following actions:

   - Enter a print queue name.
   - Press Return. When you press Return you send the file to the default output print device

   Choose option 3 to both display the release notes on the terminal and print the release notes. VMSINSTAL displays the release notes on the terminal, and when you finish reading the release notes, VMSINSTAL prompts you for a print queue name.

When you install with VMS Version 5.0 or 5.1, choose option 4 to copy the release notes to the system help directory. When you install with VMS Version 5.2 or higher, choose option 4 if you have reviewed the release notes and are restarting the installation.

When you install with VMS Version 5.0 or 5.1, choose option 5 if you have reviewed the release notes and are restarting the installation.

```
* Do you want to continue the installation [N]?:
```

VMSINSTAL then asks you if you want to continue the installation:

- Type YES and press Return to continue the installation
- Press Return to stop the installation

```
%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.
```

In either case, VMSINSTAL copies the release notes to the SYS$HELP:KAV010.RELEASE_NOTES file.

Note

The name of the release notes file installed by VMSINSTAL is denoted by the current product name and version number. Do not delete release notes for previous versions of the KAV30 software.

6. Read Informational Messages

```
        DEC VAXELN KAV TOOLKIT EXTENSIONS FOR VMS

%KAV-I-VMS, Checking VMS version
%KAV-I-VERSION, VMS Version 054, RELEASED
```

VMSINSTAL displays informational messages.

7. Respond to license registration query

```
        Product:      KAV-TOOLKIT-V
        Producer:     DEC
        Version:      1.0
        Release Date: 1-NOV-1990

* Does this product have an authorization key registered and loaded?
```

The installation procedure displays license information about the product and asks if you have registered and loaded the Product Authorization Key (PAK):

- Type NO and press Return when you have not registered and loaded the PAK. Register and load the PAK before continuing with the installation.
- Type YES and press Return when you have registered and loaded the PAK.

8. **Choose the IVP option**

```
This kit contains an Installation Verification Procedure (IVP). The
IVP verifies that the installed software is available on your system.
You can run the IVP to verify the DEC VAXELN KAV Toolkit Extensions
for VMS installation.

You can run the IVP either during or after the installation. To run
the IVP after the installation, execute the following command at the
Digital Command Language (DCL) prompt:

$ @SYS$COMMON:[SYSTEST.KAV]KAV$IVP.COM

To run the IVP during the installation procedure, answer YES to the
following question:

* Do you want to run the IVP after the installation [YES]?
```

The installation procedure now asks if you want to run the IVP. The KAV30 IVP verifies that the KAV30 software and hardware are available on the system. Digital recommends that you run the IVP.

9. **Purge files**

```
This installation modifies the following files on your system:

ELN$:RTL.OLB
ELN$:RTLOBJECT.OLB
ELN$:SHARED_STATUS_TEXT.EXE

This installation installs the following files on your system:

ELN$:300KER.EXE
ELN$:300KER.MAP
ELN$:300KER.STB
ELN$:EBUILD.EXE
ELN$:EBUILD.HLB
ELN$:KAV$RTL_OBJLIB.OLB
ELN$:KRDRIVER.EXE
```

```
        ELN$:SCSI53C700.OBJ
        ELN$:SCSI53C700.OPT
        ELN$:SCSI53C700_SCRIPT.OBJ
        ELN$:KAVDEF.H, .PAS, .FOR, .ADA
        ELN$:KAV$EXERCISER.DAT
        ELN$:KAV$EXERCISER.EXE
        ELN$:KAV$EXERCISER.OPT
        ELN$:KAV$EXR_ETHERNET_SECONDARY.EXE
        ELN$:KAV$EXR_INTS_SECONDARY.EXE
        ELN$:KAV$EXR_OBJLIB.OLB
        ELN$:KAV$EXR_SECONDARY.DAT
        ELN$:KAV$EXR_UART_SECONDARY.EXE
        ELN$:KAV$EXR_USER_1_PROCESS.PAS
        ELN$:KAV$EXR_USER_2_PROCESS.PAS
        ELN$:KAV$EXR_USER_3_PROCESS.PAS
        ELN$:KAV$EXR_USER_4_PROCESS.PAS
        ELN$:KAV$EXR_USER_5_PROCESS.PAS
        ELN$:KAV$EXR_USER_6_PROCESS.PAS
        ELN$:KAV$EXR_USER_7_PROCESS.PAS
        ELN$:KAV$EXR_USER_8_PROCESS.PAS
        ELN$:KAV$EXR_VME_DATA_SECONDARY.EXE
        ELN$:KAVEXRDEF.PAS

        This installation installs the following example programs on your
        system:

        [SYSHLP.EXAMPLES.KAV]KAV_MVME.FOR
        [SYSHLP.EXAMPLES.KAV]KAV_TIMER.C
        [SYSHLP.EXAMPLES.KAV]MVMEDEF.H
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER.C
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER_AST.C
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER_ISR.C
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER_ISR_AST.C

        [SYSHLP.EXAMPLES.KAV]VDADDRIVER.C

        [SYSHLP.EXAMPLES.KAV]FIFO_AST_ROUTINES.ADA
        [SYSHLP.EXAMPLES.KAV]FIFO_CONSUMER.ADA
        [SYSHLP.EXAMPLES.KAV]FIFO_CONSUMER.DAT
        [SYSHLP.EXAMPLES.KAV]FIFO_PRODUCER.ADA
        [SYSHLP.EXAMPLES.KAV]FIFO_PRODUCER.DAT
        [SYSHLP.EXAMPLES.KAV]FIFO_SIGNALLER_TASK.ADA

* Do you want to purge files replaced by this installation [YES]?
```

The KAV30 software installation installs and modifies a number of files on the system. See Appendix B for a description of these files. You have the option to purge any previous versions of these files that are on the system. Digital recommends that you purge these files.

10. Read informational messages

```
        The installation procedure will not ask any more questions. The
        remainder of the installation takes between 5 and 20 minutes,
        depending on the processor configuration.

%VMSINSTAL-I-RESTORE, Restoring product save set B ...

        The installation procedure renames all ELN$:300KER.* files to
        ELN$:300KER.OLD_*, and deletes previous versions of ELN$:300KER.OLD_*
        files.

        The installation procedure updates ELN$:RTL.OLB and ELN$:RTLOBJECT.OLB.
        If either of these files is not present, the installation procedure
        aborts.

%KAV-I-UPDLIB, updating ELN$:RTL.OLB...
%KAV-I-UPDOBJ, updating VAXELN Kernel global data...
%KAV-I-UPDOBJ, updating VAXELN global parameter...
%KAV-I-INSOBJ, inserting KAV Kernel global data...
%KAV-I-UPDOBJ, updating VAXELN Kernel vector table...
%KAV-I-UPDOBJ, updating VAXELN Kernel messages/text...
%KAV-I-UPDLIB, updating ELN$:RTLOBJECT.OLB...
%KAV-I-UPDOBJ, updating VAXELN Kernel messages for EPASCAL...
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                          SYS$COMMON:[SYSHLP.EXAMPLES.KAV].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                          SYS$COMMON:[SYSTEST.KAV].
%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...
```

VMSINSTAL displays informational messages.

11. Observe the IVP

```
        The installation procedure invokes the IVP.
```

If you chose to run the IVP, VMSINSTAL runs it now.

```
System image size is 495 pages (248 Kbytes); file size is 496 blocks

        This IVP provides a downline loadable system
        KAV$IVP.SYS in SYS$COMMON:[SYSTEST.KAV].
        This system can be used to verify the correct
        installation of the KAV software and the KAV30
        hardware, it however requires the KAV30 to be
        installed and ready.

        Before you downline load the IVP system
        use the following commands to setup the NCP database:

        $ MC NCP Set Node <Name> Address <Addr>
        $ MC NCP Set Node <Name> Hardware Address 08-00-2B-nn-nn-nn
        $ MC NCP Set Node <Name> Load File SYS$COMMON:[SYSTEST.KAV]KAV$IVP.SYS
```

```
Upon successful installation and bootup, the
KAV30 module will start counting up starting
with 00 in the status display.
```

The IVP displays informational messages as it runs. When the IVP runs successfully, it displays the following message:

```
DEC VAXELN KAV Toolkit Extensions for VMS IVP completed successfully.
```

When the IVP fails, it displays one or more of the following messages:

```
300KER.EXE not found - IVP failed.
300KER.STB not found - IVP failed.
300KER.MAP not found - IVP failed.
KAV Object Library not found - IVP failed.
KAVDEF.H not found - IVP failed.
KAVDEF.PAS not found - IVP failed.
KAVDEF.ADA not found - IVP failed.
KAVDEF.FOR not found - IVP failed.
RTL.OBJ not found - IVP failed.
New module 1 in RTL.OBJ not found - IVP failed.
New module 2 in RTL.OBJ not found - IVP failed.
New module 3 in RTL.OBJ not found - IVP failed.
New module 4 in RTL.OBJ not found - IVP failed.
New module 5 in RTL.OBJ not found - IVP failed.
```

See Section 3.2 for information on what to do when the IVP fails.

12. **End the installation procedure**

```
Installation of KAV V1.0 completed at 10:47

VMSINSTAL procedure done at 10:47
```

VMSINSTAL indicates that the installation is complete. You can now log out of the privileged account:

```
$ LOGOUT
SYSTEM     logged out at 6-MAY-1991 at 10:48:01.39
```

Note

VMSINSTAL deletes or changes entries in the process symbol tables during the installation. If you are going to continue using the system manager's account and you want to restore these symbols, log out of the account and log in again.

2.2 Error Recovery

This section describes how to recover from errors that occur while you install the KAV30 software.

When the installation fails, VMSINSTAL displays the following message:

```
%VMSINSTAL-E-INSFAIL, The installation of KAV V1.0 has failed.
```

Errors occur during the installation if any of the following conditions exist:

- The operating system version is incorrect
- Prerequisite software is absent
- You did not register and load the product license

For descriptions of the error messages generated by these conditions, see the VMS documentation on system messages, recovery procedures, and VMS software installation. When the system indicates that any of these conditions exist, take the appropriate action, as described in the error message.

3

After Installing the KAV30 Software

This chapter describes what you can do after installing the KAV30 software. It gives information on the following:

- Running the IVP
- Reporting problems

3.1 Running the Installation Verification Procedure

This section describes how to run the KAV30 IVP. The IVP verifies that the KAV30 software and hardware function correctly. You usually run the IVP during the installation procedure. However, you might want to run the IVP after the installation procedure. For example, after a system failure, you must run the KAV30 IVP to make sure that users can access the KAV30 software and hardware.

To run the IVP after the installation procedure, follow these steps:

1. Set-up the network control program (NCP) database so that you can downline load the IVP system to the KAV30. To include details of the KAV30 in the NCP database, enter the following commands:

```
$ MC NCP SET NODE <node_name> ADDRESS <node_address>
$ MC NCP SET NODE <node_name> HARDWARE ADDRESS <ethernet_address>
$ MC NCP SET NODE <node-name> LOAD FILE SYS$COMMON:[SYSTEST.KAV]KAV$IVP.SYS
```

   Replace <node_name> with the node name. Replace <node_address> with the node address. Replace <ethernet_address> with the Ethernet address.

2. Downline load the IVP system to the KAV30. The SYS$COMMON:[SYSTEST.KAV]KAV$IVP.SYS file contains the IVP system.

3. Boot the KAV30. When you boot the KAV30, it runs the IVP. When the IVP is successful, it displays the following message:

```
KAV Installation verification procedure completed successfully.
```

When the IVP fails, it displays one or more of the following messages:

```
300KER.EXE not found - IVP failed.
300KER.STB not found - IVP failed.
300KER.MAP not found - IVP failed.
KAV Object Library not found - IVP failed.
KAVDEF.H not found - IVP failed.
KAVDEF.PAS not found - IVP failed.
KAVDEF.ADA not found - IVP failed.
KAVDEF.FOR not found - IVP failed.
RTL.OBJ not found - IVP failed.
New module 1 in RTL.OBJ not found - IVP failed.
New module 2 in RTL.OBJ not found - IVP failed.
New module 3 in RTL.OBJ not found - IVP failed.
New module 4 in RTL.OBJ not found - IVP failed.
New module 5 in RTL.OBJ not found - IVP failed.
```

See Section 3.2 for information on what to do when the IVP fails.

3.2 Reporting Problems

This section describes how to report problems to Digital. When you encounter a problem while using the KAV30 software, report it to Digital. Depending on the nature of the problem and the type of support that is available to you, take one of the following actions:

- Call Digital when the software contract or warranty agreement entitles you to telephone support.
- Submit a Software Performance Report (SPR).
- Fill out and submit a Reader's Comments form when the problem has to do with this documentation. There are two Reader's Comments forms at the back of each KAV30 guide. Use the form from the guide in which you found the error. Include the section and page number.

Review the Software Product Description (SPD) and Warranty addendum for an explanation of warranty. When you encounter a problem during the warranty period, report the problem as indicated above, or follow alternate instructions provided by Digital for reporting SPD nonconformance problems.

4

KAV30 System Exerciser

The KAV30 system exerciser is a VAXELN application program that allows you to verify that the KAV30 hardware and software function correctly. This chapter describes the KAV30 system exerciser. It gives information on the following:

- Overview of the system exerciser
- Starting the system exerciser
- System exerciser screen
- System exerciser commands
- System exerciser tests
- Creating system exerciser tests

4.1 Overview of the System Exerciser

This section gives an overview of the KAV30 system exerciser. The KAV30 system exerciser is a VAXELN application program that allows you to verify that the KAV30 hardware and software function correctly. It does this by allowing you to test elements of the KAV30 hardware and software functionality, and then allowing you to review the test results.

The system exerciser offers you an interface from which you can perform actions including the following:

- Specify tests to add to a test run
- Set the test run duration
- Perform a test run
- Review test run performance data

You execute system exerciser tests in test runs. A test run is a collection of system exerciser tests. When you perform a test run, the system exerciser executes the specified tests for the duration of the test run.

You can specify tests to monitor the following KAV30 hardware and software functionality:

- Asynchronous system trap (AST) functionality
- The CVAX™ microprocessor
- The VMEbus data interface
- The VSB data interface
- The VMEbus interrupt functionality
- The auxiliary port
- The AUI port
- The small computer system interface (SCSI) port
- The battery backed-up random-access memory (RAM)
- The first-in/first-out (FIFO) buffers
- The calendar/clock
- The 32-bit cascaded timers

4.2 Starting the System Exerciser

This section describes how to start the system exerciser. To start the system exerciser, follow these steps:

1. Build the KAV30 system exerciser into the VAXELN system of the KAV30 that you want to monitor. See Section 4.2.1 for more information.
2. Some system exerciser test require a second KAV30. When you want to run tests that require a second KAV30, build the KAV30 secondary system exerciser into the VAXELN system of the second KAV30. See Section 4.2.2 for more information.
3. When the console terminal is not connected to the KAV30 that you want to monitor, you must connect it. See the *KAV30 Hardware Installation and User's Information* manual for more information.
4. Load and run the VAXELN system that includes the KAV30 system exerciser on the KAV30 that you want to monitor. See Section 4.2.3 for more information.
5. When you want to run tests that require a second KAV30, load and run the VAXELN system that includes the KAV30 secondary system exerciser on the second KAV30. See Section 4.2.3 for more information.

4.2.1 Building the System Exerciser Into a VAXELN System

Use the VAXELN System Builder with the appropriate data file to build the KAV30 system exerciser into a VAXELN system. The VAXELN System Builder offers you a menu interface through which you can enter information about the system that you are building. The data file contains answers to the System Builder prompts. See the *VAXELN Development Utilities Guide* for a complete description of the VAXELN System Builder.

To build the KAV30 system exerciser into a VAXELN system, follow these steps:

1. Create a work directory. Use this directory to preserve original versions of the system exerciser files.
2. Copy the following files to the work directory:
   - ELN$:KAV$EXERCISER.DAT
   - ELN$:KAV$EXERCISER.EXE
3. Change the data file answers that do not apply to the VAXELN system that you are building. The data file has the following specification, ELN$:KAV$EXERCISER.DAT.

   Delete device description entries for devices that the KAV30 does not have. For example, when the KAV30 does not have the SCSI option, delete the DUA description entry.

   Change the System Builder parameters that do not apply to the VAXELN system that you are building. See the *KAV30 Programmers Reference Information* manual for more information about the System Builder parameters.
4. Invoke the System Builder with the following command:

   ```
   $ EBUILD/NOEDIT KAV$EXERCISER
   ```

   The /NOEDIT qualifier specifies that you do not want to enter the menu system interface. The System Builder uses the data file answers to build the new VAXELN system.

4.2.2 Building a Secondary System Exerciser Into a VAXELN System

Use the VAXELN System Builder with the appropriate data file to build the KAV30 secondary system exerciser into a VAXELN system.

To build the KAV30 secondary system exerciser into a VAXELN system, follow these steps:

1. Create a work directory. Use this directory to preserve original versions of the system exerciser files.
2. Copy the following files to the work directory:
   - ELN$:KAV$EXR_SECONDARY.DAT
   - ELN$:KAV$EXR_VME_DATA_SECONDARY.EXE
   - ELN$:KAV$EXR_ETHERNET_SECONDARY.EXE
   - ELN$:KAV$EXR_UART_SECONDARY.EXE
   - ELN$:KAV$EXR_INTS_SECONDARY.EXE
3. Change the data file answers that do not apply to the VAXELN system that you are building. The data file has the following specification, ELN$:KAV$EXR_SECONDARY.DAT.

   Delete device description entries for devices that the KAV30 does not have. For example, when the KAV30 does not have the SCSI option, delete the DUA description entry.

   Change the System Builder parameters that do not apply to the VAXELN system that you are building. See the *KAV30 Programmers Reference Information* manual for more information about the System Builder parameters.
4. Invoke the System Builder with the following command:

   ```
   $ EBUILD/NOEDIT KAV$EXR_SECONDARY
   ```

   The /NOEDIT qualifier specifies that you do not want to enter the menu system interface. The System Builder uses the data file answers to build the new VAXELN system.

4.2.3 Loading and Running the VAXELN System

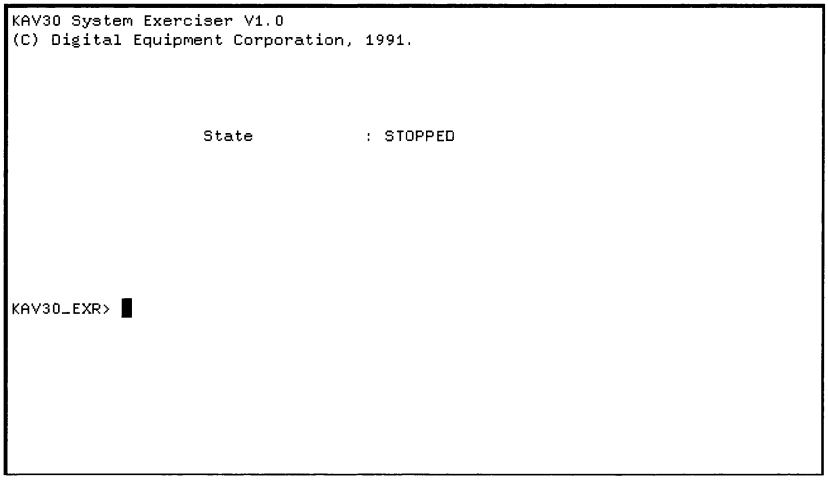
After you build either the KAV30 system exerciser or the KAV30 secondary system exerciser into a VAXELN system, use one of the following methods to load the VAXELN system onto the KAV30 and to run the system:

- Downline load the VAXELN system over the Ethernet from the host system to the KAV30 and boot the system
- Copy the VAXELN system to a DEC™ SCSI floppy disk or hard disk and boot the system from the SCSI disk

For more information about loading and running VAXELN systems, see the *VAXELN Development Utilities Guide*.

When you boot a new VAXELN system containing the KAV30 system exerciser, the KAV30 runs the exerciser and displays the exerciser screen on its console terminal. Figure 4–1 shows the initial KAV30 system exerciser screen.

Figure 4–1 Initial KAV30 System Exerciser Screen



When you boot a new VAXELN system containing the KAV30 secondary system exerciser, the system containing the KAV30 system exerciser can perform system exerciser tests with the new VAXELN system. The new VAXELN system does not display an interface.

4.3 System Exerciser Screen

This section describes the system exerciser screen. Figure 4–2 shows a sample KAV30 system exerciser screen.

Figure 4–2 Sample KAV30 System Exerciser Screen

```
KAV30 System Exerciser V1.0
(C) Digital Equipment Corporation, 1991.




                State           : STOPPED
                Runtime         :         2 minute(s)
                Errors detected :         0







KAV30_EXR> INIT
KAV30_EXR> SET RTC
KAV30_EXR> SET RUN/DURATION="0 00:01:00"
KAV30_EXR> GO
*** Test run complete - Performing Post-run cleanup ***
KAV30_EXR> REVIEW
RTC - no errors detected
RTC ran         5 times, used 600.0000 ms of cpu time
KAV30_EXR>
```

The system exerciser screen has the following characteristics:

- The top half of the screen displays the following information:
  - The test run status
  - The number of minutes the exerciser is performing the present test run
  - The number of errors detected during the present test run
- The bottom half of the screen is a scrolling region that displays the following information:
  - Previously executed exerciser commands
  - Exerciser message text
  - The exerciser command prompt

4.4 System Exerciser Commands

This section describes the system exerciser commands. They are as follows:

- `! <comment>`

  Arguments: `<comment>` specifies comment text.
  Qualifiers: None.
  Description: This command prefixes a comment. The system exerciser ignores text between an exclamation mark (!) and the next end-of-line character.

- `EXIT`

  Arguments: None.
  Qualifiers: None.
  Description: This command exits you from the system exerciser. The system exerciser actions depend on the system configuration. For example, when you run the system exerciser from VAXELN Command Language (ECL), the system exerciser returns you to the ECL prompt.

- `GO`

  Arguments: None.
  Qualifiers: None.
  Description: This command performs a test run. When the system exerciser finishes the test run, it performs a cleanup operation for approximately one minute.

- `INITIALIZE`

  Arguments: None.
  Qualifiers: None.
  Description: This command initializes a test run.

- `REVIEW`

  Arguments: None.
  Qualifiers: None.
  Description: This command displays the performance data for the previous test run.

- SET <test_name> [/P1=<parameter> ... /P10=<parameter>]

  Arguments:
  <test_name> specifies a system exerciser test name. See Section 4.5 for more information about system exerciser tests.
  Qualifiers:
  /P1 to /P10 specify parameters to the test specified in <test_name>. Starting with /P1, specify the parameters. You can specify the parameters in any order, for example:

  ```
  SET INTS/P2=1/P1=8
  ```

  Description: This command adds the specified test to the current test run.

- SET RUN/DURATION=<time>

  Arguments: None.
  Qualifiers:
  /DURATION=<time> specifies the duration of the current test run. Specify <time> in standard VMS delta time, and enclose it in quotes ("). The default duration is one minute.
  Description: This command sets the duration of the current test run.
  Example: To set the test run duration to three days, enter the following command:

  ```
  SET RUN/DURATION="3 00:00:00.00"
  ```

- WAIT <time>

  Arguments:
  <time> specifies the amount of time to pause the exercise execution. Specify <time> in standard VMS delta time, and enclose it in quotes (").
  Qualifiers: None.
  Description: This command pauses the exerciser execution for the specified amount of time.

For example, the following commands monitor all the 32-bit cascaded timers and the calendar/clock for one hour:

```
!
! Initialize the test run
!
INIT
```

```
!
! Add tests to the test run
!
SET RTC
SET TIMER/P1=0
SET TIMER/P1=1
SET TIMER/P1=2
SET TIMER/P1=3
SET TIMER/P1=4
!
! Set the test run duration
!
SET RUN/DURATION="0 01:00:00"
!
! Perform the test run
!
GO
!
! Review the test run's performance data
!
REVIEW
```

4.5 System Exerciser Tests

This section describes the system exerciser tests. They are as follows:

- AST

  Description: This test monitors the AST functionality.
  Requirements: None.
  Restrictions: None.
  Parameters: None.

- CVAX

  Description: This test monitors the CVAX microprocessor. The test executes as a low-priority compute-bound process within the system exerciser.
  Requirements: None.
  Restrictions: None.
  Parameters: None.

- DATA P1 P2 P3 P4 P5 P6 P7

  Description: This test monitors VMEbus data transfers between two KAV30s. It performs master accesses, slave accesses, and location monitoring. The system exerciser displays VME_DATA when referring to this test.

  Requirements:

  1. The KAV30 rotary switch is set to the value F
  2. A second KAV30 running a VAXELN system that includes the KAV30 secondary system exerciser

  Restrictions: You can only specify one DATA test in a test run.

  Parameters:

  P1 The number of transfers. The default is one.

  P2 The VMEbus A24 base address of the KAV30 that runs the secondary system exerciser. Specify either the decimal or hexadecimal value. Prefix hexadecimal values with 0X.

  P3 The swapping operation that the KAV30, running the system containing the KAV30 system exerciser, performs. Specify 0 for byte swapping on outgoing accesses. Specify 1 for byte swapping on incoming accesses.

  P4 The address modifier codes that the KAV30 operating in master mode uses. Specify 0 for A24 user mode codes. Specify 1 for A24 supervisor mode codes.

  P5 The byte swapping mode that the KAV30, running the system containing the KAV30 system exerciser, uses. Specify 0 for mode 0 swapping (default). Specify 2 for mode 2 swapping. Specify 3 for mode 3 swapping. The KAV30 running the system containing the KAV30 secondary system exerciser uses mode 0 swapping.

  P6 Whether the KAV30, running the system containing the KAV30 system exerciser, performs unaligned accesses. Specify 0 for aligned accesses only (default). Specify 1 for both aligned and unaligned accesses. The KAV30 running the system containing the KAV30 secondary system exerciser only performs aligned accesses.

  P7 The type of data access to perform. Specify 0 for longword accesses (default). Specify 1 for byte accesses. Specify 2 for word accesses. Specify 4 for longword accesses. Specify 7 for byte, word, and longword accesses. When you specify 7, the system exerciser performs the amount of transfers specified in P1 for bytes, words, and longwords.

- `ETHERNET P1`

  Description: This test monitors Ethernet communication between two KAV30s.
  Requirements:

  1. A second KAV30 running a VAXELN system that includes the KAV30 secondary system exerciser

  Restrictions: None.
  Parameters:

  P1 The number of bytes to transfer. The default is one.

- `FIFOS P1`

  Description: This test monitors the FIFO buffers.
  Requirements: None
  Restrictions: You can only specify four `FIFOS` tests in a test run—one for each FIFO buffer.
  Parameters:

  P1 The FIFO buffer to monitor. Specify 0 for FIFO buffer 0 (default). Specify 1 for FIFO buffer 1. Specify 2 for FIFO buffer 2. Specify 3 for FIFO buffer 3.

- `MEM P1 P2 P3 P4 P5 P6 P7`

  Description: This test monitors memory accesses to another VMEbus or VSB device. The test writes a specified amount of data to the device's memory, reads back the data, and verifies the transfer.
  Requirements:

  1. A VMEbus or VSB device

  Restrictions: In a test run, you can specify one `MEM` test for each device on the VMEbus and VSB
  Parameters:

  P1 The number of transfers. The default is one.

  P2 The base address of the other device for the type of access that you want to perform. Specify either the decimal or hexadecimal value. Prefix hexadecimal values with 0X.

  P3 The bus to use. Specify 0 for the VMEbus (default). Specify 1 for the VSB.

P4 The address modifier codes that the KAV30, operating in master mode, uses. Specify 0 for A24 user mode. Specify 1 for A24 supervisor mode. Specify 2 for A32 user mode. Specify 3 for A32 supervisor mode. Specify 4 for A16 user mode. Specify 5 for A16 supervisor mode.

P5 The byte swapping mode that the KAV30, running the system containing the KAV30 system exerciser, uses. Specify 0 for mode 0 swapping (default). Specify 2 for mode 2 swapping. Specify 3 for mode 3 swapping. The KAV30 running the system containing the KAV30 secondary system exerciser uses mode 0 swapping.

P6 Whether the KAV30, running the system containing the KAV30 system exerciser, performs unaligned accesses. Specify 0 for aligned accesses only (default). Specify 1 for both aligned and unaligned accesses. The KAV30 running the system containing the KAV30 secondary system exerciser only performs aligned accesses.

P7 The type of data access to perform. Specify 0 for longword accesses (default). Specify 1 for byte accesses. Specify 2 for word accesses. Specify 4 for longword accesses. Specify 7 for byte, word, and longword accesses. When you specify 7, the system exerciser performs the amount of transfers specified in P1 for bytes, words, and longwords.

- `INTS P1 P2`

  Description: This test monitors the generation and handling of VMEbus interrupts. You can specify that the KAV30 generates and handles interrupts itself (default), or you can specify that the KAV30 interacts with a second KAV30.

  When you specify that the KAV30 interacts with a second KAV30, the KAV30 generates interrupts which a second KAV30 handles. The second KAV30 then generates an interrupt at the interrupt-request (IRQ) level of the interrupt it handles. The KAV30 then handles this interrupt.

  The system exerciser displays `VME_INTS` when referring to this test.
  Requirements:

  1. When you want the KAV30 to interact with a second KAV30, you must have a second KAV30 running a VAXELN system that includes the KAV30 secondary system exerciser

Restrictions: You can only specify one INTS test in a test run.
Parameters:

P1 The IRQ level at which to generate the interrupt. Specify 0 for IRQ 1 (default). Specify 1 for IRQ 1. Specify 2 for IRQ 2. Specify 3 for IRQ 3. Specify 4 for IRQ 4. Specify 5 for IRQ 5. Specify 6 for IRQ 6. Specify 7 for IRQ 7. Specify 8 for all the levels (that is, IRQ 1 through to IRQ 7)

P2 The mode of operation. Specify 0 when the KAV30 both generates and handles the interrupt itself (default). Specify 1 when the KAV30 interacts with a second KAV30.

- RD_BBRAM P1 P2 P3

  Description: This test checks that the specified character is at the specified location in the user region of the battery backed-up RAM. The test runs immediately after you issue the command to execute it. Use this test with the WR_BBRAM test.
  Requirements: None.
  Restrictions: None.
  Parameters:

  P1 The number of bytes to read. The default is one byte.

  P2 The offset into the user region of battery backed-up RAM. The default is zero.

  P3 The ASCII character whose value is at the offset. The default is *.

- RTC

  Description: This test monitors the calendar/clock. The test uses the calendar/clock to generate an interrupt every second, and counts the number of interrupts in a ten-second period.
  Requirements: None.
  Restrictions: You can only specify one RTC test in a test run.
  Parameters: None.

- SCSI P1

  Description: This test monitors the KAV30 SCSI interface.
  Requirements:

  1. A Digital SCSI disk that is initialized with a valid FILES-11 disk structure

  Restrictions: None.

Parameters:

P1 The disk drive's SCSI ID. The default is zero.

- `TIMERS P1`

  Description: This test monitors a 32-bit cascaded timer. The test generates an interrupt every second, and counts the number of interrupts in a ten-second period.
  Requirements: None.
  Restrictions: In a test run, you can specify one `TIMER` test for each 32-bit cascaded timer.
  Parameters:

  P1 The timer to monitor. Specify 0 for cascaded timer 0 (default). Specify 1 for cascaded timer 1. Specify 2 for cascaded timer 2. Specify 3 for cascaded timer 3. Specify 4 for cascaded timer 4.

- `U1 to U8`

  Description: The KAV30 system exerciser allows you to create up to eight system exerciser tests (U1 to U8). See Section 4.6 for more information.

- `UART P1`

  Description: This test monitors the KAV30 auxiliary port.
  Requirements:

  1. The KAV30 auxiliary port must be connected either to a loopback connector, or to the auxiliary port of a second KAV30
  2. When you want to connect the KAV30 auxiliary port to a second KAV30, you must have a second KAV30 running a VAXELN system that includes the KAV30 secondary system exerciser

  Restrictions: None.
  Parameters:

  P1 The number of bytes to transfer. The default is one.

- `WR_BBRAM P1 P2 P3`

  Description: This test writes the specified character to the specified offset into the user region of the battery backed-up RAM. The test runs immediately after you issue the command to execute it. Use this test with the RD_BBRAM test.
  Requirements: None.

Restrictions: None.
Parameters:

P1 The number of bytes to write. The default is one.

P2 The offset into the user region of battery backed-up RAM. The default is zero.

P3 The ASCII character to write to the battery backed-up RAM. The default is *.

4.6 Creating System Exerciser Tests

This section tells you how to create system exerciser tests. The KAV30 system exerciser includes a number of tests. However, you can supplement these tests with your own tests. Use one of the eight test templates to write a system exerciser test. The templates, which are in the ELN$ directory, have the following specifications:

- KAV$EXR_USER_1_PROCESS.PAS
- KAV$EXR_USER_2_PROCESS.PAS
- KAV$EXR_USER_3_PROCESS.PAS
- KAV$EXR_USER_4_PROCESS.PAS
- KAV$EXR_USER_5_PROCESS.PAS
- KAV$EXR_USER_6_PROCESS.PAS
- KAV$EXR_USER_7_PROCESS.PAS
- KAV$EXR_USER_8_PROCESS.PAS

To help you write tests, the KAV30 system exerciser includes a VAXELN Pascal™ user-callable interface. This interface consists of a number of functions and procedures that allow you to access KAV30 system exerciser data structures. The functions and procedures are as follows:

- KAV$EXR_WAIT_FOR_START
  When you issue the GO system exerciser command, the exerciser signals an event. This function waits for the system exerciser to signal that event. Call this routine after you perform test-specific initialization.

- KAV$EXR_UPDATE_WORK_DONE
  This procedure updates the number of times the test runs. When you use the REVIEW command at the system exerciser interface, the system exerciser displays the number of times the test ran. Supply a single parameter, index, to this procedure. This procedure passes the index parameter to the system exerciser test as a process argument.

- KAV$EXR_RECORD_START_CPU_TIME
  This procedure records the time that the system exerciser test starts to use the CPU. The system exerciser uses this time to calculate the CPU usage time that it displays when you use the REVIEW command at the system exerciser interface. Supply a single parameter, index, to this procedure. This procedure passes the index parameter to the system exerciser test as a process argument.
- KAV$EXR_RECORD_END_CPU_TIME
  This procedure records the time that the system exerciser test finishes using the CPU. The system exerciser uses this time to calculate the CPU usage time that it displays when you use the REVIEW command at the system exerciser interface. Supply a single parameter, index, to this procedure. This procedure passes the index parameter to the system exerciser test as a process argument.
- KAV$EXR_END_OF_RUN
  This boolean function returns true when the test run finishes. If system exerciser tests contain explicit calls to WAIT_ANY, specify KAV$EXR_ END_OF_RUN as a condition that it waits for.
- KAV$EXR_LOCK_DISPLAY
  KAV$EXR_UNLOCK_DISPLAY
  Surround all WRITEs and WRITELNs with these procedures. These procedures protect the writing of data on the screen.
- KAV$EXR_COPY_QUALIFIER_VALUES
  This procedure obtains the values of the system exerciser test's qualifiers. Supply the following three parameters to this procedure:
  - Index
  - Count
  - Qualifier_values

  This procedure passes the index parameter to the system exerciser test as a process argument. The count parameter is an integer value that specifies the number of qualifiers. The Qualifier_values parameter is a packed array of integer values that returns the qualifier values.

- KAV$EXR_HANDLE_ERROR
  This procedure records the number of errors that this system exerciser test detects. Supply the following four parameters to this procedure:
  - Index
  - Verbose
  - Prefix
  - Status

  This procedure passes the index parameter to the system exerciser test as a process argument. The verbose parameter is a boolean value. Specify TRUE for the verbose parameter if you want the system exerciser test to display an error message when an error occurs. Otherwise, specify FALSE for the verbose parameter. When you specify FALSE, the system exerciser only records that an error occurred. The prefix parameter is a varying_string(80) value that specifies a text string to display as part of the error message. The status parameter is an integer error status value.

To create a system exerciser test, follow these steps:

1. Create a work directory. Use this directory to preserve original versions of system exerciser files.
2. Copy the test template that you want to use to the work directory.
3. Enter the relevant information in the original test template.
4. Compile the test template. For example, enter the following command to compile KAV$EXR_USER_1_PROCESS.PAS:

   ```
   $ EPASCAL KAV$EXR_USER_1_PROCESS + ELN$EXR_OBJLIB/LIB + KAV$RTL_OBJLIB/LIB -
    -  + RTLOBJECT/LIB
   ```

5. Replace the test template in KAV$EXR_OBJLIB.OLB. For example, to replace KAV$EXR_USER_1_PROCESS.PAS, enter the following command:

   ```
   $ LIB/REPLACE ELN$:KAV$EXR_OBJLIB KAV$EXR_USER_1_PROCESS
   ```

6. Link the system exerciser as follows:

   ```
   LINK ELN$:KAV$EXERCISER_OPT/OPT
   ```

7. Start the system exerciser. See Section 4.2 for more information.

A

Sample Installation

This appendix contains a sample KAV30 software installation.

```
$ @SYS$UPDATE:VMSINSTAL KAV010 MKB500: OPTIONS N

 VAX/VMS Software Product Installation Procedure V5.4-2

It is 6-MAY-1991 at 10:41.

Enter a question mark (?) at any time for help.

* Are you satisfied with the backup of your system disk [YES]? YES

The following products will be processed:

  KAV V1.0

 Beginning installation of KAV V1.0 at 10:41

%VMSINSTAL-I-RESTORE, Restoring product save set A ...

    Additional Release Notes Options:

 1.  Display release notes
 2.  Print release notes
 3.  Both 1 and 2
 4.  None of the above

* Select option [2]: 2
* Queue name [SYS$PRINT]: SYS$PRINT

* Do you want to continue the installation [N]?: YES
%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

        DEC VAXELN KAV TOOLKIT EXTENSIONS FOR VMS

%KAV-I-VMS, Checking VMS version
%KAV-I-VERSION, VMS Version 054, RELEASED

        Product:      KAV-TOOLKIT-V
        Producer:     DEC
        Version:      1.0
        Release Date: 1-NOV-1990

* Does this product have an authorization key registered and loaded? YES
```

```
        This kit contains an Installation Verification Procedure (IVP). The
        IVP verifies that the installed software is available on your system.
        You can run the IVP to verify the DEC VAXELN KAV Toolkit Extensions
        for VMS installation.

        You can run the IVP either during or after the installation. To run
        the IVP after the installation, execute the following command at the
        Digital Command Language (DCL) prompt:

        $ @SYS$COMMON:[SYSTEST.KAV]KAV$IVP.COM

        To run the IVP during the installation procedure, answer YES to the
        following question:

* Do you want to run the IVP after the installation [YES]? YES

        This installation modifies the following files on your system:

        ELN$:RTL.OLB
        ELN$:RTLOBJECT.OLB
        ELN$:SHARED_STATUS_TEXT.EXE

        This installation installs the following files on your system:

        ELN$:300KER.EXE
        ELN$:300KER.MAP
        ELN$:300KER.STB
        ELN$:EBUILD.EXE
        ELN$:EBUILD.HLB
        ELN$:KAV$RTL_OBJLIB.OLB
        ELN$:KRDRIVER.EXE
        ELN$:SCSI53C700.OBJ
        ELN$:SCSI53C700.OPT
        ELN$:SCSI53C700_SCRIPT.OBJ
        ELN$:KAVDEF.H, .PAS, .FOR, .ADA
        ELN$:KAV$EXERCISER.DAT
        ELN$:KAV$EXERCISER.EXE
        ELN$:KAV$EXERCISER.OPT
        ELN$:KAV$EXR_ETHERNET_SECONDARY.EXE
        ELN$:KAV$EXR_INTS_SECONDARY.EXE
        ELN$:KAV$EXR_OBJLIB.OLB
        ELN$:KAV$EXR_SECONDARY.DAT
        ELN$:KAV$EXR_UART_SECONDARY.EXE
        ELN$:KAV$EXR_USER_1_PROCESS.PAS
        ELN$:KAV$EXR_USER_2_PROCESS.PAS
        ELN$:KAV$EXR_USER_3_PROCESS.PAS
        ELN$:KAV$EXR_USER_4_PROCESS.PAS
        ELN$:KAV$EXR_USER_5_PROCESS.PAS
        ELN$:KAV$EXR_USER_6_PROCESS.PAS
        ELN$:KAV$EXR_USER_7_PROCESS.PAS
        ELN$:KAV$EXR_USER_8_PROCESS.PAS
        ELN$:KAV$EXR_VME_DATA_SECONDARY.EXE
        ELN$:KAVEXRDEF.PAS
```

```
        This installation installs the following example programs on your
        system:

        [SYSHLP.EXAMPLES.KAV]KAV_MVME.FOR
        [SYSHLP.EXAMPLES.KAV]KAV_TIMER.C
        [SYSHLP.EXAMPLES.KAV]MVMEDEF.H
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER.C
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER_AST.C
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER_ISR.C
        [SYSHLP.EXAMPLES.KAV]MVMEDRIVER_ISR_AST.C

        [SYSHLP.EXAMPLES.KAV]VDADDRIVER.C

        [SYSHLP.EXAMPLES.KAV]FIFO_AST_ROUTINES.ADA
        [SYSHLP.EXAMPLES.KAV]FIFO_CONSUMER.ADA
        [SYSHLP.EXAMPLES.KAV]FIFO_CONSUMER.DAT
        [SYSHLP.EXAMPLES.KAV]FIFO_PRODUCER.ADA
        [SYSHLP.EXAMPLES.KAV]FIFO_PRODUCER.DAT
        [SYSHLP.EXAMPLES.KAV]FIFO_SIGNALLER_TASK.ADA

* Do you want to purge files replaced by this installation [YES]? YES

        The installation procedure will not ask any more questions. The
        remainder of the installation takes between 5 and 20 minutes,
        depending on the processor configuration.

VMSINSTAL-I-RESTORE, Restoring product save set B ...

        The installation procedure renames all ELN$:300KER.* files to
        ELN$:300KER.OLD_*, and deletes previous versions of ELN$:300KER.OLD_*
        files.


        The installation procedure updates ELN$:RTL.OLB and ELN$:RTLOBJECT.OLB.
        If either of these files is not present, the installation procedure
        aborts.

KAV-I-UPDLIB, updating ELN$:RTL.OLB...
KAV-I-UPDOBJ, updating VAXELN Kernel global data...
KAV-I-UPDOBJ, updating VAXELN global parameter...
KAV-I-INSOBJ, inserting KAV Kernel global data...
KAV-I-UPDOBJ, updating VAXELN Kernel vector table...
KAV-I-UPDOBJ, updating VAXELN Kernel messages/text...
KAV-I-UPDLIB, updating ELN$:RTLOBJECT.OLB...
KAV-I-UPDOBJ, updating VAXELN Kernel messages for EPASCAL...
VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                        SYS$COMMON:[SYSHLP.EXAMPLES.KAV].
VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                        SYS$COMMON:[SYSTEST.KAV].
VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...

        The installation procedure invokes the IVP.

System image size is 495 pages (248 Kbytes); file size is 496 blocks
```

```
        This IVP provides a downline loadable system
        KAV$IVP.SYS in SYS$COMMON:[SYSTEST.KAV].
        This system can be used to verify the correct
        installation of the KAV software and the KAV30
        hardware, it however requires the KAV30 to be
        installed and ready.

        Before you downline load the IVP system
        use the following commands to setup the NCP database:

        $ MC NCP Set Node <Name> Address <Addr>
        $ MC NCP Set Node <Name> Hardware Address 08-00-2B-nn-nn-nn
        $ MC NCP Set Node <Name> Load File SYS$COMMON:[SYSTEST.KAV]KAV$IVP.SYS

        Upon successful installation and bootup, the
        KAV30 module will start counting up starting
        with 00 in the status display.

        DEC VAXELN KAV Toolkit Extensions for VMS IVP completed successfully.

 Installation of KAV V1.0 completed at 10:47

 VMSINSTAL procedure done at 10:47

$ LOGOUT
SYSTEM     logged out at 6-MAY-1991 at 10:48:01.39
```

B

Files Modified and Installed

This appendix lists the files that the KAV30 software installation modifies and installs on the system.

The installation modifies the following files in the ELN$ directory:

File	Modification
RTL.OLB	Add KAV data structures, KAV routine entry points, and extended VAXELN kernel message codes
RTLOBJECT.OLB	Add VAXELN Pascal extended VAXELN kernel message codes
SHARED_STATUS_TEXT.EXE	Add KAV kernel-specific message codes and message texts

The installation installs the following files in the ELN$ directory:

File	Description
300KER.EXE	KAV30 kernel image
300KER.MAP	KAV30 link map
300KER.STB	KAV30 symbol table
EBUILD.EXE	VAXELN System Builder image
EBUILD.HLB	VAXELN System Builder help library
KAV$RTL_OBJLIB.OLB	KAV30 runtime library
KRDRIVER.EXE	SCSI port driver image
SCSI53C700.OBJ	SCSI NCR 53C700 port driver object code
SCSI53C700.OPT	SCSI NCR 53C700 port driver option file
SCSI53C700_SCRIPT.OBJ	SCSI NCR 53C700 port driver script

Files Modified and Installed

File	Description
KAVDEF.H	KAV30 VAX C™ language definitions
KAVDEF.PAS	KAV30 VAXELN Pascal language definitions
ELN$:KAVDEF.FOR	KAV30 VAX FORTRAN™ language definitions
KAVDEF.ADA	KAV30 VAX Ada™ language definitions
KAV$EXERCISER.DAT	KAV30 system exerciser VAXELN System Builder data file
KAV$EXERCISER.EXE	KAV30 system exerciser image
KAV$EXERCISER.OPT	KAV30 system exerciser linker options
KAV$EXR_ETHERNET_SECONDARY.EXE	KAV30 secondary system exerciser image
KAV$EXR_INTS_ SECONDARY.EXE	KAV30 secondary system exerciser image
KAV$EXR_OBJLIB.OLB	KAV30 system exerciser object library
KAV$EXR_SECONDARY.DAT	KAV30 secondary system exerciser VAXELN System Builder data file
KAV$EXR_UART_ SECONDARY.EXE	KAV30 secondary system exerciser image
KAV$EXR_USER_1_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_2_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_3_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_4_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_5_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_6_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_7_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_USER_8_PROCESS.PAS	KAV30 system exerciser test template
KAV$EXR_VME_DATA_SECONDARY.EXE	KAV30 secondary system exerciser image
KAVEXRDEF.PAS	KAV30 system exerciser interface declarations

The installation installs the following example programs in the [SYSHLP.EXAMPLES.KAV] directory:

File	Description
KAV_MVME.FOR	VAX FORTRAN program that implements an MVME 335 device driver
KAV_TIMER.C	VAX C program that shows you how to use the KAV$TIMERS VAXELN KAV Toolkit Extensions for VMS system service
MVMEDEF.H	Contains definitions required the MVME 335 card
MVMEDRIVER.C	VAX C program that implements an MVME 335 device driver
MVMEDRIVER_AST.C	VAX C program that uses ASTs to implement an MVME 335 device driver
MVMEDRIVER_ISR.C	VAX C program containing an interrupt service routine for the MVMEDRIVER.C program
MVMEDRIVER_ISR_AST.C	VAX C program containing an interrupt service routine for the MVMEDRIVER_AST.C program
VDADDRIVER.C	A VAX C program that implements a VDAD device driver
FIFO_AST_ ROUTINES.ADA	VAX Ada program containing the AST routines that the FIFO_CONSUMER.ADA and FIFO_ PRODUCER.ADA programs use
FIFO_CONSUMER.ADA	VAX Ada program that implements a KAV30 FIFO consumer
FIFO_CONSUMER.DAT	VAXELN System Builder data file for the FIFO_ CONSUMER.ADA program.
FIFO_PRODUCER.ADA	VAX Ada program that implements a KAV30 FIFO consumer
FIFO_PRODUCER.DAT	VAXELN System Builder data file for the FIFO_ PRODUCER.ADA program.
FIFO_SIGNALLER_TASK.ADA	VAX ADA program containing the signaler task that the FIFO_CONSUMER.ADA and FIFO_ PRODUCER.ADA programs use

Glossary

The glossary defines some important terms used in this guide.

application program

A program that performs an end-user task.

AST

Asynchronous system trap. A procedure that the operating system calls when a particular event occurs.

autovectored interrupt

An interrupt for which the interrupt handler provides the interrupt vector address.

backup process

The process of making copies of the data stored on the disk, so that you can recover that data after an accidental loss. Make backup copies on RX33 diskettes, TK50 tape cartridges, or over a network.

backup copy

A copy of the data stored on the disk.

BIL

Indented Bills Report. A list of specific types and amounts of direct materials that a Digital customer is charged for.

big-endian device

A device based on the 68000® family of processors. *See also* little-endian.

BOM

Bill of materials. A list of specific types and amounts of direct materials that a Digital customer receives.

cluster

A group of computers networked together that share disk storage, application programs, and other computer resources.

CVAX

A real-time microprocessor that is based on the VAX™ architecture.

ECL

VAXELN command language.

FIFO

First-in/first-out. The order in which processing is performed. For example, a FIFO queue processes data on a first-come, first-served basis.

FIFO buffer

An area in which devices can store and retrieve data.

host system

The primary or controlling computer in a multiple-computer network.

interactive

A method of communicating with the system. In an interactive session, you type a command at the keyboard and the system executes the command and responds with a prompt character for another command.

interrupt

A break in the usual flow of a program to process an external request.

interrupt handler

A device that executes interrupt service routines for interrupt requesters. The device receives interrupt requests from the bus.

interrupt requester

A device that requests the execution of an interrupt service routine. The device sends an interrupt request on the bus, which an interrupt handler responds to.

IRQ

Interrupt-request signal. A request, issued by a device, to execute an interrupt service routine.

IVP

Installation Verification Procedure. The IVP verifies that the LAT-11 server and local area transport (LAT) node systems are working properly.

little-endian device

A device based on the Intel™ family of processors.

LMF

License management facility. A VMS utility that allows you to register and load Digital software licenses.

NCP

Network control program. The block that contains the necessary information to set up a virtual circuit or to accept or reject a request to set up a virtual circuit.

network

A group of individual computer systems that are connected by communications lines to share information and resources.

PAK

Product authorization key. The product authorization key is a paper certificate that contains software license registration information.

operating system

A collection of system programs that control the operation of the system and allow the user access to data files, input/output devices, and application programs. The operating system software performs such tasks as assigning memory to programs and data, processing requests, and scheduling jobs.

RAM

Random-access memory. A read/write memory device.

SCSI

Small computer systems interface. An interface designed for connecting disks and other peripheral devices to computer systems. SCSI is defined by an American National Standards Institute (ANSI) standard.

SPD

Software Product Description. Defines the function of a product and minimum hardware needed to support it. It describes software, components, and service.

SPR

Software Performance Report. A form given to customers so that Digital can receive feedback on the performance of software.

system exerciser

An application program that allows you test that a system's hardware and software function correctly.

test run

System exercisers execute tests in test runs. A test run is a collection of system exerciser tests. The system exerciser executes all the tests simultaneously.

vectored interrupt

An interrupt for which the interrupt requester provides the interrupt vector address.

VMSINSTAL

A Digital software product that helps end users to install Digital software on computers running the VMS operating system.

Index

P

R

S

T

U

V

W

Reader's Comments

KAV30 Software Installation and System Testing Information

AA-PEYAA-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Completeness (enough information)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Page layout (easy to find information)	☐	☐	☐	☐

I would like to see more/less _____

_____

What I like best about this manual is _____

_____

What I like least about this manual is _____

_____

I found the following errors in this manual:

Page Description

_____ _____

_____ _____

_____ _____

Additional comments or suggestions to improve this manual:

_____

_____

_____

I am using **Version** _____ of the software this manual describes.

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Phone _____

Do Not Tear - Fold Here and Tape

digital™

AFFIX
STAMP
HERE

DIGITAL EQUIPMENT CORPORATION
Corporate User Information Products
ZKO1–3/J35
110 SPIT BROOK RD
NASHUA, NH 03062–9987

Do Not Tear - Fold Here

Reader's Comments

KAV30 Software Installation and System Testing Information

AA-PEYAA-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Completeness (enough information)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Page layout (easy to find information)	☐	☐	☐	☐

I would like to see more/less _____

_____

What I like best about this manual is _____

_____

What I like least about this manual is _____

_____

I found the following errors in this manual:

Page Description

_____ _____

_____ _____

_____ _____

Additional comments or suggestions to improve this manual:

_____

_____

_____

I am using **Version** _____ of the software this manual describes.

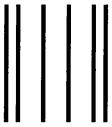
Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Phone _____

Do Not Tear - Fold Here and Tape



AFFIX
STAMP
HERE

DIGITAL EQUIPMENT CORPORATION
Corporate User Information Products
ZKO1–3/J35
110 SPIT BROOK RD
NASHUA, NH 03062–9987

Do Not Tear - Fold Here

digital

Printed in Europe