



**Consulting
Software Specialties
Internet / Intranet Servers**

OpenVMS Hobbyist

Multia Support Page

Topics on this page:

[Introduction](#)

[Getting Started](#)

[Multia Notes](#)

[SRM Firmware Update](#)

[Other Hardware Requirements](#)

[OpenVMS Installation - Floppy](#)

[OpenVMS Installation](#)

Introduction

The Multia was originally designed based on a low-cost Alpha CPU and was intended as a Windows/NT workstation. According to information available from Digital (now Compaq), the viability of the Multia as an OpenVMS workstation was reviewed. Some preliminary work was done; however, the idea was deemed to have only a limited degree of sales value.

The Multia is rather slow by most current Alpha workstation standards. Still, it has gained some popularity in the UN*X community since they are available, albeit in limited numbers, and are Alpha workstations, instead of VAX or Intel. This provides a test platform for operating systems such as Linux, FreeBSD, NetBSD, OpenBSD, etc. that even students can afford. Now, it has begun to provide an affordable Alpha platform for the OpenVMS Hobbyist.

Please note that the author of this web page does not (yet?) have a Multia. The information here is based on [documented procedures and file locations](#).

Getting Started

Please read [some important notes](#), courtesy of John Malmberg... (Many thanks to John for

these great notes!)

Getting started with the Multia begins unlike most other Alpha and VAX machines.

In the case of the Multia, before attempting to install OpenVMS on a Multia, you must update the console "SRM" firmware. The SRM firmware required to support OpenVMS is available in the "boot floppy" saveset provided on the OpenVMS freeware V4 CD.

You will also need the floppy disk required during the installation process. The diskette is required only for the installation procedure. Once OpenVMS is installed on the Multia, the install floppy is not required and the system boots up normally from its own internal disk.

SRM Firmware Update

The SRM firmware update is actually an automated process. The problem is just getting the diskette in the first place. The files are buried in a saveset which represents the installation boot floppy. The files can only be extracted on an OpenVMS V6.2 or later system (or V6.1 with an important VAX [BACKUP ECO](#) applied). From there, they must be transferred to a DOS format floppy disk.

The files you're looking for can be found in the installation boot floppy BACKUP/IMAGE saveset, in the [VMS\$COMMON.SRM] path. There are two files:

QBYPASS.SCR
SRM.EXE

You can find these two files in a DOS/Win [.ZIP archive here](#) on our web site.

Unpack the .ZIP archive onto a freshly formatted diskette. Note - please be sure to format the diskette yourself as experience has shown that pre-formatted diskettes are only about 80% reliable.

Making the SRM update diskette is best done on a pre-Win/9x machine. Using Win/9x for this poses no special difficulties; however, there are [some problems with Win/95 and Win/98](#) of which you should be aware.

Once you have transferred the files to a DOS/Win format diskette, simply insert the diskette into your Multia's diskette drive and power up your Multia. The existing firmware will look at the diskette, execute the QBYPASS.SCR script and use the SRM.EXE file to update the SRM firmware.

We're working on a firmware update diskette image that you can copy onto a diskette using your OpenVMS system. Watch this space for more info.

Other Hardware Requirements

As we've already discussed, your Multia needs a diskette drive to facilitate the firmware

update and the OpenVMS install.

Generally, 32 MB of RAM is about the minimum to run OpenVMS V7.1-2 or V7.2 and later. However, if you intend to run DECwindows/Motif, you'll soon wish for more memory as your minimum set of processes will fully consume the free page list resulting in a lot of swapping and a rather slow system. Available information indicates that the Multia will accept up to four(4) SIMMs of up to 32 MB each, total 128 MB. Use parity memory.

The Multia internally accepts only a single 2-1/2 inch SCSI disk if the PCMCIA options are present. If these options are absent, the system box will also accomodate a low-profile (1-inch high) 3-1/2 inch form factor disk. Disks such as RZ26L and disks of larger capacity have reportedly been installed in such machines.

The Multia's stand is an integral part of the cooling needed by the system board. In the Multia, cooling is partially fan-forced, partially convection cooling. Specifically, the system **MUST** be operated in a vertical position. The cooling fan draws air over the CPU heat sink; however, other areas of the system board depend on convection to allow heated air to move away from the heat-producing components and be exhausted by the fan.

OpenVMS Installation - Floppy

Installing OpenVMS on a Multia requires a special floppy disk. This provides the needed special bootstrap and some drivers that are needed to support OpenVMS on the Multia.

The boot floppy can be made in couple of ways:

- Using another [OpenVMS system which has a floppy drive](#). The floppy can be made by performing an image restore of a saveset which can be found on the V4 Freeware CD from Compaq.
- Using another [OpenVMS system which has no floppy drive](#). The floppy can be made by first creating a floppy-sized logical disk, then performing an image restore of a saveset which can be found on the V4 Freeware CD from Compaq. You can then transfer the diskette image (logical disk container file) to your MS-DOS, Windows/9x or UN*X system and employ the third method:
- Using an [MS-DOS, Windows/9x](#) or [UN*X](#) system to write a diskette image to a diskette. For MS-DOS or Windows/9x, you'll need a program called [RAWRITE](#). You'll also need the boot diskette image for OpenVMS [V7.1-2](#) or [V7.2](#) (available via these links as compressed .ZIP archives).

Notice that Win/NT is explicitly not mentioned here. [Reports](#) indicate that the RAWRITE

program will not run reliably on Win/NT in a "DOS" window. At one time, "OpenDOS" (formerly known as DR-DOS) was available as shareware (free trial software) on the [Caldera](#) website. That may be worth looking into. There's also [FreeDOS](#), but this may still be in a state which is too immature to allow the use of RAWRITE.

Writing the Boot Floppy on OpenVMS

There are two ways to use an OpenVMS system to create the diskette.

The first way is to use an OpenVMS-Alpha system which has a diskette drive. All you need for this is a freshly formatted 3-1/2 inch HD (high-density) floppy and the floppy disk /IMAGE saveset. Insert the diskette, MOUNT it /FOREIGN and use BACKUP/IMAGE to do the restore.

For example:

```
$ BACKUP/IMAGE MULTIA_V72.BCK/SAVE DVA0:
```

Writing the Boot Floppy on OpenVMS With Help From DOS/Win

The second way to create the boot diskette is to use an OpenVMS-Alpha system to create a diskette image file (logical disk container file), then transfer that to your DOS/Win machine and use RAWRITE to create the diskette.

You'll need the [Logical Disk Driver](#) free software from Digital.

First, create a Logical Disk Container File the size of a High-Density diskette:

```
$ LD CREATE MULTIA.FLP/SIZE=2880
```

Then, connect it to a Logical Disk device, INITIALIZE it, and MOUNT it /FOREIGN (the device name LDA1: is used as an example only):

```
$ LD CONNECT MULTIA.FLP LDA1:
```

```
$ INITIALIZE LDA1: MULTIA
```

```
$ MOUNT/FOREIGN LDA1:
```

Now, use BACKUP to restore the diskette /IMAGE saveset to the logical disk:

```
$ BACKUP/IMAGE MULTIA_V72.BCK/SAVE LDA1:/INITIALIZE
```

Dismount the Logical Disk device and disconnect it from the container file.

```
$ DISMOUNT LDA1:
```

```
$ LD DISCONNECT LDA1:
```

Now, transfer the container file to your DOS/Win machine (remember to transfer it as BINARY!) and use the [RAWRITE](#) program to write it to a diskette. See the next section for how to use RAWRITE to create the diskette on your DOS/Win machine.

Writing the Boot Floppy on Your DOS/Win Machine

Once you've transferred the diskette image file (or logical disk container file) to your

DOS/Win machine, or after you've unzipped the compressed diskette image archive you downloaded, you can use the [RAWRITE](#) program to create a boot floppy from the image file.

Here's a sample RAWRITE session:

```
C:\DNLOAD>dir *.flp
```

```
Volume in drive C has no label
Volume Serial Number is 2E6B-14F2
Directory of C:\DNLOAD
```

```
MLTIA712 FLP      1,474,560  07-15-99  9:03p MLTIA712.FLP
MLTIAV72 FLP      1,474,560  07-15-99  9:06p MLTIAV72.FLP
      2 file(s)          2,949,120 bytes
      0 dir(s)           1,652.25 MB free
```

```
C:\DNLOAD>rawrite
```

```
RaWrite 1.3 - Write disk file to raw floppy diskette
```

```
Enter source file name: mltiav72.flp
```

```
Enter destination drive: a:
```

```
Please insert a formatted diskette into drive A: and press -ENTER- :
```

```
Number of sectors per track for this disk is 18
```

```
Writing image to drive A:. Press ^C to abort.
```

```
Track: 79 Head: 1 Sector: 16
```

```
Done.
```

```
C:\DNLOAD>
```

The above is just an example, of course. Your directory name, diskette drive letter, filename, etc. may differ from the above example.

Notice that RAWRITE should be used in a "DOS Window" on Win/9x. If possible, it is preferable to start your machine without the GUI. You can do this by pressing F8 while the "Starting Windows 95" message is displayed. You will then be presented with a boot menu. From that menu, select the "Command Line only" option. This will help prevent any problems that Win/9x may introduce due to certain "features" in [Windows/95](#) and [Windows/98](#).

The safest move is to run RAWRITE on a machine running an earlier version of DOS, or to start your Win/9x machine without the GUI.

Writing the Boot Floppy on a UN*X System

You can use your Linux, BSD, etc. machine to write the boot diskette, also. If you download a .ZIP file containing a compressed diskette image, you'll need the appropriate "unzip" program for your operating system. See the [InfoZip download site](#) if you need unzip for your platform.

Once you've transferred the diskette image file (or logical disk container file) to your UN*X machine, or after you've unzipped the compressed diskette image archive you downloaded, you can use the "dd" program to create a boot floppy from the image file.

Remember to format the diskette before using "dd" to write it. Only formatting is necessary; creating a filesystem on the diskette is not necessary. Some UN*X systems have an "fdformat" command. Refer to the man page on fdformat for usage information. The example shown here is based on the FreeBSD documentation.

```
# fdformat -f 1440 fd0.1440
```

See the man page for "dd" on your system for command line options; however, none should be needed other than to identify the input and output streams. Again, the example shown here is based on the FreeBSD documentation.

```
# dd if=mltiav72.flp of=/dev/rfd0
```

OpenVMS Installation

Once you have made the floppy disk that will be needed during the OpenVMS Installation, all you need is the OpenVMS Operating System CD-ROM. About the best source for this is the [media order page](#) on the [OpenVMS Hobbyist home site](#). You can also order OpenVMS Operating System CD-ROM directly from [Compaq](#).

You'll need the special boot floppy twice: once to boot the Operating System CD and again to boot the system once you've installed OpenVMS on it. After that, you will be able to boot your Multia normally from its own hard disk without the floppy.

To proceed with installing OpenVMS, refer to the [OpenVMS-Alpha Installation Page](#).

Other OpenVMS-related pages:

[OpenVMS Page](#)

[OpenVMS Hobbyist Page](#)

[OpenVMS Freeware Page](#)

[ZIP](#) and [UNZIP](#) for OpenVMS

[Using Zip and Unzip on OpenVMS](#)

[The REAL Story about "tarballs"](#) (GZIP and TAR for OpenVMS)

[VAX Linux](#)

"Vaporware"

[Free VMS \(for Intel\)](#)

[Home Page](#) [Operating Systems](#) [Products](#) [Specialties](#) [Freeware](#)

[FreeVMS Page](#) [Bulletin Board](#) [Contact Us](#) [About Us](#)

Tradenames, trademarks, service marks, etc. appearing on this page are the property of their registered owners.

"DJE Systems" is a service mark of [David J. Dachtera](#), doing business as DJE Systems.
This page Copyright ©1998, 1999 by David J. Dachtera, All Rights Reserved.