

[For Home](#)[For Work](#)[Support](#)

Search HP.com

[Software](#) > [OpenVMS Systems](#) > [OpenVMS FAQ](#)> page 23

HP OpenVMS Systems Documentation

The OpenVMS Frequently Asked Questions (FAQ)

[Previous](#)[Contents](#)[Index](#)

14.4.1 on the Alpha Multia?

Yes, there are a set of unsupported images that permit specific OpenVMS Alpha versions to bootstrap on the Multia UDB system. These images and the associated instructions are available at the OpenVMS Freeware website:

- <http://www.hp.com/go/openvms/freeware/>

Look in the Freeware V5.0 /multia/ directory.

Instructions are included IN the kits. READ THE INSTRUCTIONS. PLEASE!

Some of the restrictions involved when running OpenVMS on the Multia system include (but may well not be limited to) the following:

- The PCMCIA support was completely removed, because the Intel chip on the Multia was not compatible with the Cirrus chip on the Alphabook.
This means, of course, that you will not see and cannot use any PCMCIA cards on a Multia.
The Multia uses shared interrupts, and as a result, a special ZLXp-E series graphics device driver---one that does not use interrupts---is needed. This driver is provided in the kit.
- The serial lines don't work.
- If you have a Multia with a PCI slot, you can't use any PCI card that requires interrupts.
- The SRM console on this system is very old and very fragile. (This SRM console was designed only and strictly for diagnostic use, and was not particularly tested or used with OpenVMS.)
- If things don't work for you, don't expect to see any OpenVMS updates, nor SRM console updates, nor any support.
- Do not expect to see any new versions of OpenVMS on the Multia nor on any other unsupported systems. If such new versions do appear and do work, please consider it as a pleasant surprise.

The Multia images are not included on the OpenVMS Freeware V4.0 CD-ROM kit, the kit that was distributed with OpenVMS V7.2. (These images became available after Freeware V4.0 shipped.)

Other sources of information for OpenVMS on Multia include:

- <http://www.djesys.com/vms/hobbyist/multia.html>
- <http://www.djesys.com/vms/hobbyist/multianot.html>
- <http://www.djesys.com/vms/hobbyist/support.html>
- <http://www.netbsd.org/Ports/alpha/multiafaq.html>
- <http://www.brouhaha.com/~eric/computers/udb.html>

14.4.2 on AlphaPC 164LX? AlphaPC 164SX?

OpenVMS Alpha is not supported on the AlphaPC 164LX and 164SX series, though there are folks that have gotten certain of the LX series to load SRM and bootstrap OpenVMS. (The Aspen Durango II variant, specifically.)

One problem has been generally reported: ATA (IDE) bootstraps will fail; SCSI storage and a SCSI CD-ROM device is required.

Also see [Section 14.4.2.1](#).

14.4.2.1 on the NoName AXPpci33 system?

Information on bootstrapping OpenVMS (using the Multia files described in [Section 14.4.1](#)) on the (unsupported) NoName AXPpci33 module is available at:

- <http://www.jyu.fi/~kujala/vms-in-axppci33.txt>

Tips for using the Multia files with the AXPpci33:

- You have to use the Multia kit and follow the directions in ALPHA8, but do *not* load the Multia SRM firmware into the AXPpci33. Rather, download and use the latest firmware for the AXPpci33 from the HP Alpha firmware website instead.
- 64 MB memory is generally necessary.
- you cannot use any PCI cards, and if you plan on networking, you need to find an ISA Ethernet card supported by OpenVMS.
- When the AXPpci33 board bootstraps, it will dump some stuff like a crash dump, but it will continue and---so far---this hasn't caused any particular hassles.
- The system shutdown and reboot procedures do not work properly.
- The serial console is reported to not work, though the serial ports apparently do work. The status of the parallel port is unknown.
- Rumour has it that you have one of the AXPpci33 motherboards with the PS/2 mouse and keyboard connectors and a VGA card (one that will work under DECwindows) and you can run DECwindows on the system.

14.4.3 on the Alpha XL series?

No.

OpenVMS Engineering does not formally support the Alpha XL series, nor will OpenVMS (informally) bootstrap on the Alpha XL series.

OpenVMS can not, will not, and does not bootstrap on the Alpha XL series. The Alpha XL series was targeted for use (only) with the Microsoft Windows NT operating system.

The Alpha XL platform does not resemble other supported platforms.

14.4.4 OpenVMS on the Personal Workstation -a and -au series?

Though OpenVMS is not supported on the Personal Workstation -a series platforms, OpenVMS might or might not bootstrap on the platform.

If you wish to attempt this, you must ensure that all graphics and all I/O controllers in the system are supported by OpenVMS. You must also ensure that you have the most current firmware loaded.

Here are some salient differences within the various Personal Workstation series:

- The -a series was designed and was tested for Windows NT use. Only. It is not supported for use with OpenVMS.
- The -au series was designed and tested for Windows, OpenVMS, and Tru64 UNIX compatibility, and is considered a supported system.
- There are at two different and distinct variants of the family, and usually referred to by their internal hardware project names.
 - The Miata MX5. The Miata MX5 variant has no USB ports and no on-board SCSI. The on-board Intel SIO chipset is not supported by OpenVMS, and thus OpenVMS cannot bootstrap ATAPI CD-ROM devices. That said, the Miata MX5 -a series typically came with DEC branded Adaptec 2940UW SCSI controllers, Matrox Millennium graphics cards, no L3 cache module, and an Toshiba IDE CD-Rom. Some came with very high end Powerstorm graphics card if the system was destined to do CAD or movie rendering. Graphics and other I/O can and does vary by package. The Miata MX5 is not supported by OpenVMS.
 - The Miata GL. The Miata GL variant has USB ports and on-board SCSI and bootstraps using the on-board Cypress IDE chipset and an ATAPI CD-ROM are supported by OpenVMS. The Miata GL -a variant is need not be configured with an add-on SCSI controller, given both the ability to bootstrap from ATAPI CD-ROM and the on-board SCSI. Graphics and other I/O can and does vary by package. Various of the Miata GL systems are supported by OpenVMS.

For obvious reasons, most folks will prefer and will select a Miata GL system, given the choice between the Miata MX5 and the Miata GL series. And as for your next question, you cannot necessarily nor easily distinguish the Miata MX5 from the Miata GL based solely on the model number.

See [Section 14.4.4.2](#) for related details.

14.4.4.1 OpenVMS on the Whitebox Windows-Only series Alpha?

Though OpenVMS is not supported on the "Whitebox" series of Alpha platforms, OpenVMS might or might not bootstrap on the platform. These systems were specifically configured, targeted and supported only for use with the Microsoft Windows NT operating system.

On some of the "Whitebox" systems, the following sequence of console commands can potentially be used to convert the system over to unsupported use by and for OpenVMS Hobbyist users. (But please note that if you wish to attempt this, you must ensure that all graphics and all I/O controllers in the system are supported by OpenVMS, and you must ensure that you have the most current SRM firmware loaded. (For information on locating and downloading the most current Alpha SRM firmware, please see [Section 14.3.7.1](#).) And you must realize that the resulting Whitebox configuration will be entirely unsupported and may or may not be stable and useful.)

```
set os_type vms
cat nvram ! too see what is in this, if anything
edit nvram
10 set srm_boot on
20 e
init
```

If your nvram has other contents, you will need to change the line numbers (10 and 20) to reflect the contents of your configuration. To obtain documentation on the commands of the console editor, enter the ? command within the editor.

The above sequence was reportedly tested on the DIGITAL Server 3300 series, a relative of the AlphaServer 800 series. The DIGITAL Server 3300 is not supported by OpenVMS, though the AlphaServer 800 series is a supported platform. The sequence may or may not work on other platforms, and may or may not work on the DIGITAL Server 3300 platform.

Also see [Section 5.33](#).

14.4.4.2 OpenVMS and Personal Workstation ATA (IDE) bootstrap?

OpenVMS will boot and is supported on specific Personal Workstation -au series platforms, though OpenVMS will require a SCSI CD-ROM if the Intel Saturn I/O (SIO) IDE chip is present in the configuration--- only the Cypress IDE controller chip is supported by OpenVMS for IDE bootstraps. (Configurations with the Intel SIO are not generally considered to be supported systems.)

If you have an -au series system, you can determine which IDE chip you have using the SRM console command:

SHOW CONFIGURATION

If you see "Cypress PCI Peripheral Controller", you can bootstrap OpenVMS from IDE storage. If you see "Intel SIO 82378", you will need to use and bootstrap from SCSI. (A procedure to load DQDRIVER on the Intel SIO---once the system has bootstrapped from a SCSI device---is expected to be included as part of the contents of the DQDRIVER directory on Freeware V5.0 and later.)

Many of the -a series systems will include the Intel SIO, and thus cannot bootstrap from IDE.

See [Section 14.4.4](#) for related details.

14.4.5 On the Intel Itanium IA-64 platform?

OpenVMS has been ported to the Intel IA-64 architecture; to HP Integrity systems based on the Intel Itanium Processor Family.

The first release of OpenVMS I64 was V8.0, with the first general release of OpenVMS I64 known as V8.2. Yes, there was a V8.1 release, too.

Some Intel and HP terminology: Itanium Processor Family is the name of the current implementation; of the current Intel microprocessor family implementing the IA-64 architecture. IA-64 is the name of the Intel architecture implementing the VLIW (Very Long Instruction Word) design known as EPIC (Explicitly Parallel Instruction Computing).

I64 is the name of a family of HP computer systems that use Intel Itanium processors and that are supported by "HP OpenVMS for Integrity Servers" (and itself more commonly known as "OpenVMS I64"); by one of the HP operating systems that runs on HP Integrity hardware.

The Extensible Firmware Interface (EFI) is the name of the console environment for Itanium systems, and the Baseboard Management Console (BMC) and the optional Management Processor (MP) are the most typical hardware interfaces into the system console.

14.4.5.1 Where can I get Intel Itanium information?

Intel Itanium Processor Family and IA-64 Architecture, Hardware, Software, and related documentation materials are available at:

- <ftp://download.intel.com/design/IA-64/manuals/>
- <ftp://download.intel.com/design/IA-64/Downloads/>
- <ftp://download.intel.com/design/IA-64/Downloads/archSysSoftware.pdf>
- <ftp://download.intel.com/design/IA-64/Downloads/24870101.pdf>

Information on the classic Intel Extensible Firmware Interface (EFI) (for IA-64) and of the multi-platform Unified EFI (UEFI) console project documentation are available at the following URLs:

- Intel
<http://developer.intel.com>
- UEFI
<http://www.uefi.org>

Please see [Section 14.4.5](#) for Intel Itanium terminology.

14.5 What is the least expensive system that will run OpenVMS?

The cheapest systems that are or have been recently offered by HP that will run OpenVMS Alpha are the AlphaServer DS10 server, the AlphaStation XP900 workstation, the AlphaStation VS10 workstation, and the AlphaStation XP1000 workstation. Other companies sell Alpha-powered systems and Alpha motherboards, some of which will run (and can be purchased with) OpenVMS---see the OpenVMS Software Product Description (SPD) for details on the supported systems and configurations. There are also many used AlphaStation, AlphaServer, and DEC 3000 series models available which are quite suitable. For more experienced OpenVMS system managers, the (unsupported) Multia can bootstrap OpenVMS---see [Section 14.4.1](#) for details.

Used Itanium-based systems that a hobbyist could likely use to bootstrap OpenVMS I64 have been seen selling on auction websites for under US\$1000. New Integrity rx1620 series systems (officially supported by OpenVMS I64) have been offered as part of a week-long DSPP porting and training package for US\$2000. See [Section 2.8.3](#) for details on the DSPP program. Also see the HP Renew used-and/or refurbished-equipment program for any hardware that might be available.

Free and commercial VAX software-based hardware emulators are available for various platforms. See [Section 13.12](#) for details on those.

Depending on the OpenVMS version and configuration, the OpenVMS Software Product Description (SPD) is available at:

- <http://www.hp.com/go/openvms/doc/>

When purchasing a system, ensure that the system itself is supported, that the system disk drive is supported or closely compatible, that the optical (CD or DVD) drive is supported or is closely compatible and that (in the case of SCSI devices) it also specifically supports 512-byte block transfers; no equivalent requirement exists for IDE devices. Also particularly ensure that the video controller is supported. Use of supported HP hardware will generally reduce the level of integration effort involved.

A CD-ROM, CD-R or DVD drive is required for OpenVMS Alpha installations, and a DVD-ROM or recordable or rewritable DVD DVD drive is required for OpenVMS I64 installations.

CD-ROM drive compatibility information is available at:

- <http://sites.inka.de/pcde/dec-cdrom-list.txt>

14.6 Where can I get more information on Alpha systems?

HP operates an AlphaServer information center at:

- <http://www.hp.com/go/server>

Alpha Technical information and documentation is available at:

- <ftp://ftp.compaq.com/pub/products/alphaCPUdocs/>
- <http://h18000.www1.hp.com/products/software/alpha-tools/>
- <ftp://ftp.digital.com/pub/DEC/Alpha/systems/>
- <http://ftp.digital.com/pub/Digital/info/semiconductor/literature/dsc-library.html>
- Alpha Systems Update:
http://h18000.www1.hp.com/alphaserver/fb_acu.html

Software Product Description (SPD) information, including platform support documentation:

- <http://h18000.www1.hp.com/info/spd/>
OpenVMS typically uses SPD 25.01.xx, SPD 41.87.xx, and SPD 82.35.xx.

Information on Multia hardware is available at:

- <http://www.netbsd.org/Ports/alpha/multiafaq.html>

Information on DEC 3000 series hardware is available at:

- <http://www.phys.ufl.edu/~prescott/linux/alpha/dec3000-sysinfo.html>
- <http://www.phys.ufl.edu/~prescott/linux/alpha/dec3000-docs.html>
- <http://ftp.netbsd.org/pub/NetBSD/misc/dec-docs/index.html>

The NetBSD folks maintain useful Alpha hardware information at:

- <http://www.netbsd.org/Ports/alpha/models.html>

14.7 Describe Alpha instruction emulation and instruction subsets?

The Alpha architecture is upward- and downward-compatible, and newer instructions are emulated on older platforms, for those cases

where the compiler is explicitly requested to generate the newer Alpha instructions.

In particular, OpenVMS Alpha V7.1 and later include the instruction emulation capabilities necessary for the execution of newer Alpha instructions on older Alpha microprocessors. (Instruction emulation capabilities are available for user-mode application code, and are not available to device drivers or other similar kernel-mode code.)

Alpha instructions are available in groups (or subsets). Obviously, there is the base instruction set that is available on all Alpha microprocessors. Then, the following are the current instruction extension groups (or subsets) that are available on some of various recent Alpha microprocessors:

- byte/word extension (BWX):
LDBU, LDWU, SEXTB, SEXTW, STB, and STW.
- floating-point and square root extension (FIX):
FTOIS, FTOIT, ITOFF, ITOFS, ITOFT, SQRTF, SQRTG, SQRTS, and SQRTT.
- count extension (CIX):
CTLZ, CTPOP, and CTTZ.
- multi-media extension (MVI):
MAXSB8, MAXSW4, MAXUB8, MAXUW4, MINSB8, MINSW4, MINUB8, MINUW4, PERR, PKLB, PKWB, UNPKBL, and UNPKBW.

The typical instruction subset that provides the biggest win---and of course, your mileage may vary---is typically the instruction set that is provided by the EV56 and later; specifically, the byte-word instruction subset. To select this subset, use the following:

`/ARCHITECTURE=EV56/OPTIMIZE=TUNE=GENERIC`

The `/ARCHITECTURE` controls the maximum instruction subset that the compiler will generally use, while the `/OPTIMIZE=TUNE` controls both the instruction-level scheduling and also the instructions generated inside loops---any code resulting from `/OPTIMIZE=TUNE` that is specific to an instruction subset will be generated only inside loops and will also be "protected" by an AMASK-based test that permits the execution of the proper code for the particular current Alpha microprocessor.

Typically `/OPTIMIZE=TUNE=GENERIC` is the appropriate choice for tuning, and the `/ARCHITECTURE` selects the minimum target architecture for general use throughout the generated code.

generated for later architectures and instruction subsets will run on older Alpha systems due to the emulation, but if `/ARCHITECTURE` is a significant benefit, then the emulation might be a performance penalty.

Please see the OpenVMS Ask The Wizard area for the source code of a (non-privileged) tool that looks at the instruction subsets available on the particular Alpha microprocessor that the tool is run on. This tool demonstrates the use of the Alpha AMASK and IMPLVER instructions.

Please see [Section 10.22](#) and [Section 14.9](#) for additional details and related considerations.

14.8 So how do I open up the DEC 3000 chassis?

After removing those two little screws, tilt the back end of the top shell upwards---then you can remove the lid.

14.9 What is byte swizzling?

"Swizzling" is the term used to describe the operation needed to do partial longword (i.e. byte or word) accesses to I/O space on those systems that don't support it directly. It involved shifting the offset into an address space by 5 (or 7 for one older system), and ORing this into the base address. It then required the size of the operation to be ORed into the low order bits.

That is, because the EV4 and EV5 CPUs did not bring bits 0 and 1 off the chip, to do programmed I/O for bytes/words, the information on the size/offset of the transfer was encoded into the address data. The data itself then had to be shifted into the correct "byte lane" ; into the required offset position within a longword transfer;

The EV56 microprocessor supports byte/word instruction references in memory space, however only specific EV56 systems can support byte/word accesses into I/O space; device drivers may or may not be able to utilize to byte/word instructions to access device registers. Further, even on an EV56 system with hardware support for byte/word accesses into I/O space, the relevant OpenVMS routines typically do not support byte/word access into I/O space.

Systems based on the EV6 microprocessor (with the salient exception of the AlphaServer GS60 and AlphaServer GS140 series, for reasons of platform compatability) support a flat, byte addressable I/O space.

If a device driver uses CRAM or IOC\$WRITE_IO/IOC\$READ_IO, then OpenVMS will correctly process the swizzling requirements without requiring changes the driver; OpenVMS will transparently swizzle and unswizzle the I/O space references, if needed for the particular target platform. (Access and use of these routines may or may not be feasible within the requirements for a particular device driver, with the decision typically based on the target performance requirements and the expected frequency of device references and thus the expected frequency of calls to these or other similar routines.)

To use byte/word operations on MEMORY, you need to tell the compiler to use the EV56 or EV6 architecture (`/ARCHITECTURE=EV56`). Memory operations did not swizzle, but the compiler would do long/quad access, and extract/insert bytes as needed. Using `/ARCHITECTURE=EV56` allows smaller, more efficient byte/word access logic to memory.

If the application is directly referencing I/O space access across a range of Alpha systems such as is done with the X Windows device drivers, then the driver will need to know how to do swizzling for old platforms, and byte access for new platforms. Device drivers for new graphics controllers can specifically target and specifically require platforms based on EV6 and later Alpha microprocessors because of this requirement, for instance.

Please see [Section 10.22](#) and [Section 14.7](#) for additional details and related considerations.

Previous	Next	Contents	Index
--------------------------	----------------------	--------------------------	-----------------------

About HP

Contact us
Newsroom
Investor relations
Living Progress
Accessibility
Events
HP Labs
Jobs

Social Media

Consumer support forum
Enterprise business community
Developer community
Corporate blogs

HP Partners

Become a partner
Find a reseller
PartnerOne

Customer Support

Power cord replacement
Download drivers
Register your product
HP replacement parts
Authorized service providers
Training & certification
Product recycling

[Home](#) | [Email sign-up](#) | [Site map](#) | [Privacy](#) | [Cookies & ad choices](#) | [Terms of use](#) | [Recalls](#)

© 2015 Hewlett-Packard Development Company, L.P.

[View mobile site](#)