



Software Product Description

PRODUCT NAME: VSI C for OpenVMS

SPD DO-DVCSPD-01A

DESCRIPTION

This document addresses VSI C Version 7.4-1 for OpenVMS for VSI Alpha and VSI Integrity.

VSI C is a standard conforming implementation of the C programming language with VSI extensions. The VSI C compiler runs under the OpenVMS Alpha and I64 Operating Systems and generates optimized and position-independent code.

VSI C is a native-mode language product, and is integrated into the Common Language Environments. All OpenVMS system services are available to programs written in VSI C. VSI C programs can invoke, as functions, modules written in other languages.

VSI C supports Record Management Services (RMS) for sequential file organizations and associated access methods. VSI C also supports stream file-access methods common among many C implementations.

VSI C provides extensive standard-conformance checking, as well as many optional code-quality and portability diagnostics, and supports the lint-like features of the VSI Source Code Analyzer. The VSI Source Code Analyzer allows the programmer to check for consistent function usage throughout a program environment.

VSI C also generates complete debug and traceback records for use with OpenVMS Debug. Debug allows the C programmer to set breakpoints, examine and modify the contents of user variables, and selectively halt or continue program execution.

VSI C on OpenVMS provides IEEE floating-point support as defined by, and in conformance with, the IEEE 754 Standard.

FEATURES

VSI C for OpenVMS provides the following features:

- Limited support for installing more than one version of VSI C on the same node, and allowing users to select which version to use on a per-process basis (not on OpenVMS I64, which uses PCSI-based kits).
- Separate modes of compilation to support each of six C dialects:
 - A strict ANSI89 mode that compiles according to the original ANSI C standard (ANSI/ISO/IEC 9899:1990).
 - A strict C99 mode that compiles nearly in accordance with the 1999 version of the C standard (ANSI/ISO/IEC 9899:1999), except that the support is not complete and has not been fully verified against conformance suites. This mode may be useful as a preview of what will be considered conforming source code under the new standard, but should not be relied upon for production use until a future release of both VSI C and the VSI C Run-Time Library for OpenVMS containing complete C99 support.

VSI C for OpenVMS

- A relaxed ANSI mode that compiles according to the latest standard supported by the compiler, but also accepts those VSI extensions that do not directly conflict with the semantics of standard C.
- A common mode that supports many common usage C constructs as implemented on UNIX systems including HP-UX (also called "K&R" C or "pcc" mode)
- A Microsoft compatibility mode that interprets source programs according to certain language rules followed by the C compiler provided with the Microsoft Visual C++ compiler product.

In addition, just the features specified by Addendum 1 to the ISO C standard adopted by ISO in November of 1994 (digraphs and the `__STDC_VERSION__` predefined macro) can be added to each of these dialects except for VAX C mode.

- Data types for numeric, nonnumeric, and systems programming:
 - C99 Universal Character Names (UCNs) are accepted in identifiers, string literals, and character constants (and their wide variations).
 - VSI C supports 8, 16, 32, and 64-bit signed and unsigned integers.
 - VSI C supports an 8-bit `_Bool` data type for C99.
 - VSI C supports 32-bit float and 64-bit double floating-point data types.
 - VSI C also supports IEEE floating-point formats in 32-bit single, 64-bit double, and 128-bit quad-precision double-extended representations. The C language type "long double" normally is represented in 128-bit quad precision IEEE format on these platforms, although there is a compile-time option that allows the user to specify that it should use the same representation as type "double".
 - C99 constants for specific values of Infinity and NaN are supported when using `/float=ieee`.
 - VSI C supports the C99 `_Complex` keyword for specifying three types that represent values in the complex plane, based on Cartesian coordinates of type float, double, or long double, respectively, except that `D_float` representation is not supported for `_Complex` types. Run-time library support for C99 mathematical functions operating on these types is available.
 - VSI C supports passing numeric constants by reference in function calls.
 - VSI C supports the multibyte and wide-character types and features of XPG4, with the locale support available in OpenVMS versions.
 - VSI C supports user-controlled features to specify the use of 64-bit pointers that allow applications to exploit the increased address space capabilities of the Alpha and I64 architectures. These features include command-line qualifiers, `#pragma` directives, and run-time library specifications that allow the programmer to allocate and access data at run time that is to be beyond the range of addressing afforded by 32-bit pointers. By default, programs compiled by earlier versions of the compiler or on earlier versions of OpenVMS continue to behave as before, strictly within 32-bit address space. Explicit use of the new compiler features allow such programs to be extended to exploit the extended address space with minimal changes to the source code.
- Storage allocation using:
 - Reserved words (`globalref`, `globaldef`, and `globalvalue`) for sharing data among program modules
 - Reserved words (`readonly`, `noshare`, and `psect` name specification) for control of data attributes and data placement
 - Reserved words (`_align` and `_unaligned`) for specifying the alignment boundaries of data objects
 - Pragmas to control extern models and structure member alignment and base structure alignment
- Option for running only the preprocessor phase of compilation
- Option for generating include-file dependency information to aid in construction of files for the VSI Module Management System
- Pragmas to control compiler options
- The `C99_Pragma` operator, which effectively allows `pragma` directives to be produced by macro expansion.
- Compilation options allowing a choice between fast turnaround and optimization across compilation units

VSI C for OpenVMS

- Option to generate a file of prototype-style function declarations suitable for use in a header file from the function definitions (both prototype-style and old-style) contained in a source file.
- Enhanced diagnostic message controls with the command-line qualifier /WARNINGS, including the following features:
 - Specify whether a message is issued only once per compilation, or at each occurrence
 - Specify severity of any message with a default severity of information or warning
 - Control optional messages using a single numeric "importance level"
 - Control optional messages using functional groups
- Compiler-generated listing file including optional:
 - Annotations that provide information about certain optimizations that were performed or not performed
 - Source Code
 - Include-file contents
 - Machine code
 - Macro expansion
 - Compilation statistics
 - Symbol table with attributes of source program identifiers
 - Symbol cross reference, showing for each symbol the source lines where it is defined or used, annotated with type of use
- Built-in functions allow access to a subset of machine instructions. VSI C OpenVMS Alpha inline-assembly code is also supported giving access to all Alpha machine-code instructions and PAL calls.
- Integration into the OpenVMS Common Language Environments:
 - Generation of complete debug and traceback records for Debug support
 - Conformance to the Calling Standard
 - Access to the Common Run-Time Library for general purpose routines and support of multi-language environments
 - Access to the data management facilities of OpenVMS Record Management Services (RMS) by direct calls to the Common Run-Time Library
 - Support for providing error diagnostics to the VSI Language-Sensitive Editor and cross-reference information for the VSI Source Code Analyzer
 - Support for Common Data Dictionary (CDD)
 - VSI C support for interaction with routines executing in translated mode. On OpenVMS Alpha, native Alpha images can link against and interoperate with images translated from OpenVMS VAX. On OpenVMS I64, native I64 images can link against and interoperate with images translated from OpenVMS Alpha (including Alpha images translated from OpenVMS VAX).
- Extensive global and local optimizations of generated code for increased performance under OpenVMS
- Extensive control over optimization behavior
- Interface to the curses screen-manipulation package
- Installation kit cooperates with VSI C++.

VSI C for OpenVMS

Compatibility with Other C Implementations

VSI C is a conforming hosted implementation of ANSI X3.159-1989 Programming Language C (ISO/IEC 9899:1990[1994]). Its VAXC, common C, and Microsoft C compatibility modes provide many features to ease porting from other environments, though they do not provide 100% emulation of every feature of a particular version of the compilers used in those environments. In addition, the relaxed ANSI mode accepts all features from the currently-supported standard (C99) and also accepts a number of features present in those special dialects that do not conflict with the standard, as well as features from the GNU C compiler (gcc) that are sometimes used in Open Source applications and header files on the Linux platform (e.g. the `__typeof__` operator). While many programs written in C for other compilers can be successfully recompiled under VSI C, some incompatibilities among implementations exist.

Run-Time Library for C Applications

The complete VSI C Run-Time Library that is needed for use with VSI C is distributed with the OpenVMS operating systems. The VSI C Run-Time Library provides routines to perform input/output, character and string handling, mathematical computations, memory allocation, and emulation of selected UNIX[R] features. These routines are provided both in shared image and object module library form.

Run-time Library Redistribution

The VSI C kit may include run-time library components in either shareable image or object library form. VSI grants the user a nonexclusive royalty-free worldwide right to reproduce and distribute these Run-Time Libraries ("the RTLs") provided that the user:

- distributes the RTLs only in conjunction with and as a part of the user's software application product, which is designed to operate in the OpenVMS environment;
- does not use VSI's name, logo, or trademarks to market the user's software application product;
- includes VSI's copyright notice for VSI C on one of the following:
 - the user's product disk label
 - each copy of the application
 - the title or copyright page of the documentation for the software application product
- agrees to indemnify, hold harmless, and defend VSI from and against any claims or lawsuits, including attorney's fees, that arise or result from the use or distribution of the software application product. Except as expressly provided herein, VSI grants no implied or express license under any of its patents, copyrights, trade secrets, trademarks, or any license or other proprietary interests and rights.

HARDWARE REQUIREMENTS

Processors Supported:

- Integrity: Any Integrity system capable of running the VSI OpenVMS Integrity Operating System Version 8.4-2 or higher.
- Alpha: Any AlphaServer system capable of running the VSI OpenVMS Alpha Operating System Version 8.4-2L1 or higher.

Refer to the latest VSI OpenVMS Integrity or Alpha Software Product Description for information about supported servers.

Disk Space Requirements (Block Cluster Size = 1)

The following counts refer to the disk space required on the system disk. The sizes are approximate. Actual sizes may vary depending on the user's system environment, configuration, and software options.

	VSI C for OpenVMS Alpha	VSI C for OpenVMS Integrity
Disk space required for kit installation (without docs):	150,000 blocks (75 MB)	225,000 (110MB)
Disk space required for kit installation: (with docs):	250,000 blocks (125 MB)	325,000 (159MB)
Disk space required for use (permanent) without docs:	100,000 blocks (50 MB)	150,000 (75MB)
Disk space required for use (permanent) with docs:	160,000 blocks (80 MB)	210,000 (106MB)

VSI C for OpenVMS

Memory Requirements for DECwindows Support

The minimum supported memory for this application running in a standalone DECwindows environment with both the client and server executing on that same system is 8 Mbytes.

OPTIONAL HARDWARE

Floating-point-intensive applications should be run on configurations with the appropriate hardware support for the floating-point data types being used. For OpenVMS I64 in particular, note that only the IEEE format has hardware support; the VAX format floating-point types on OpenVMS I64 systems are implemented in software and incur significant software run-time emulation overhead. Floating-point code that is performance-critical should always use the IEEE format on OpenVMS I64. Consult the base operating system Software Product Description (SPD) for the appropriate floating-point accelerator or other floating-point hardware appropriate for your configuration.

SOFTWARE REQUIREMENTS

On Integrity servers, VSI OpenVMS Integrity Version 8.4-2 or higher is the required operating system version for this product. On AlphaServer systems, VSI OpenVMS Alpha Version 8.4-2L1 or higher is the required operating system version for this product.

SOFTWARE LICENSING

A software license is required in order to use the VSI C software product.

- For Integrity servers, the license is a Concurrent Use license. Version update licenses are not available for the Integrity servers platform. Rights to use future revisions of VSI C are available only through a Support Agreement or through a new license purchase.
- For AlphaServer systems, the license to use VSI C is included in the ALPHA-LP license.

For more information about OpenVMS licensing terms and policies, contact your VSI account representative.

Information is also available at the following website:

<http://vmsssoftware.com/services>

LICENSE MANAGEMENT FACILITY SUPPORT

VSI C for OpenVMS supports the *OpenVMS License Management Facility*.

For more information about the License Management Facility, refer to the *VSI OpenVMS License Management Utility Manual* in the OpenVMS documentation set.

CLUSTER ENVIRONMENT

This layered product is fully supported when installed on any valid and licensed OpenVMS Cluster configuration, which are fully described in the *OpenVMS Cluster Software Product Description (SPD DO-VIBHAA-032)*. See the HARDWARE REQUIREMENTS section in this document for hardware requirements.

OPTIONAL SOFTWARE

VSI DECset for OpenVMS:

- Language-Sensitive Editor/Source Code Analyzer (LSE/SCA) for OpenVMS Systems
- DIGITAL Test Manager (DTM) for OpenVMS Systems
- Performance and Coverage Analyzer (PCA) for OpenVMS Systems
- Code Management System (CMS) for OpenVMS Systems
- Module Management System (MMS) for Open- VMS Systems

GROWTH CONSIDERATIONS

The minimum hardware and software requirements for any future version of this product may be different from the requirements for the current version.

VSI C for OpenVMS

ORDERING INFORMATION

For VSI C on OpenVMS, licenses are available as electronic licenses (E-LTU) or physical licenses (P-LTU):

For VSI Integrity:

- VSI C for VMS I64 Concurrent E-LTU SL-LICC0E-73V
- VSI C for VMS I64 Concurrent P-LTU SL-LICC0P-73V

For VSI Alpha:

- VSI C for OpenVMS Included in the ALPHA-LP license bundle

SOFTWARE PRODUCT SERVICES

A variety of service options are available from VSI. For more information, contact your VSI account representative or distributor. Information is also available at the following website:

<http://vmssoftware.com/services>

SOFTWARE WARRANTY

This software product is provided by VSI with a 90-day conformance warranty in accordance with the VSI warranty terms applicable to the license purchase.

Copyright © 2017 VMS Software, Inc., Bolton Massachusetts, USA

Confidential computer software. Valid license from VSI required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for VSI products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. VSI shall not be liable for technical or editorial errors or omissions contained herein.

HPE, HPE Integrity, HPE Alpha, and HPE Proliant are trademarks or registered trademarks of Hewlett Packard Enterprises.

Microsoft, Windows, Windows-NT and Microsoft XP are U.S. registered trademarks of Microsoft Corporation.

The VSI OpenVMS documentation set is available on DVD.