VSI Pascal for OpenVMS

Software Product Description

1. Description

This document addresses VSI Pascal Version 6.1 for OpenVMS Alpha and VSI Pascal Version 6.1 for OpenVMS Integrity Servers (I64).

VSI Pascal is an implementation of the Pascal[1] language that accepts programs compatible with either level of the ISO specification for Programming languages - Pascal ([ISO 7185-1987]) as well as (ANSI/IEEE 770X3.97-1987). VSI Pascal also meets the Federal Information Processing Standard Publication ((FIPS-109)) requirements by accepting programs conforming to the ANSI standard. VSI Pascal also accepts many features from the Extended Pascal standard ((ANSI/IEEE 770X3. 160-1989) and (ISO 10206)).

The compiler has been validated for both levels of the ISO unextended Pascal standard and for conforming to FIPS-109. Containing extensions to the standards, VSI Pascal generates optimized, shareable code that takes full advantage of the Alpha and Itanium hardware floating point and character instruction sets and the virtual memory capabilities of the OpenVMS Alpha and I64 Operating Systems. The language contains control statements, data types, and predeclared procedures and functions.

Major Pascal Language Elements:

- INTEGER, REAL, CHAR, BOOLEAN, enumerated, and subrange data types

- ARRAY, RECORD, SET, and FILE structured data types

- Schemata type denoting families of types

- STRING schema denoting variable-length character strings up to 65,535 characters

- FOR, REPEAT, and WHILE repetitive control statements

- CASE, IF-THEN, and IF-THEN-ELSE conditional statements

- BEGIN...END compound statement

- User-defined procedures and functions that can return structured types (other than file types)

- GET, PUT, READ, WRITE, READLN, and WRITELN input and output procedures

- Concatenation operator and set of predefined character string functions including INDEX, LENGTH, and SUBSTR

- Standard set of functions, procedures, and operators

- Module initialization and finalization

- Initial state specification

- Enhanced structured value constructors

- Implementation characteristics (MAXCHAR, MINREAL, MAXREAL, EPSREAL)

- OTHERWISE clause and case ranges for CASE statement and variant records

- Date and Time functions and procedures

---

[1] K.Jensen and N. Birth ,Pascal User Manual and Report,2nd.ed., Spring-Verlag, New York 1974.

- Short Circuit Boolean Operators (AND_THEN and OR_ELSE)

- Non-Decimal representation of numbers

- Exponentiation operator (**)

Pascal Extensions:

- DOUBLE (VAX D_floating ,VAX G_floating, and IEEE T_floating on OpenVMS Alpha and I64) data type that supports the Alpha double range and double precision floating point architectural features

- QUADRUPLE (IEEE X_floating on OpenVMS Alpha and I64) data type that supports the Alpha extended range and extended precision floating point architectural features

- Support for null-terminated strings via the C_STR_T predefined type and the MALLOC_C_STR, C_STR, PAS_STRCPY, and PAS_STR predefined functions

- VARYING data type denoting variable-length character strings up to 65,535 characters

- Language elements providing sequential and random access to RMS relative files, and sequential and keyed access to RMS multikey indexed files

- Optional attributes specification on constants, variables, types, type identifiers, routines, routine parameters, schema discriminants, and compilation units

- MODULE capability for combining procedures, functions, and other declarations for compilation separate from the main program

- ENVIRONMENT and INHERIT attributes to control separate and independent compilation

- UNSIGNED, CARDINAL, INTEGER_ADDRESS, INTEGERnn, UNSIGNEDnn, POINTER, and SINGLE predefined types

- VALUE initialization section and optional value initialization in declaration section program level

- External procedure and function declarations

- Nonpositional passing of parameters

- RETURN, BREAK, CONTINUE, EXIT and NEXT statements

- SELECT, SELECTONE statements (OpenVMS Alpha and OpenVMS I64 systems only)

- Default values for parameters

- Double-quoted character strings with backslash constants

- 31-character identifiers that can include dollar sign ($) and underscore (_)

- Conditional compilation facility including the %IF directive and the /CONSTANT command line qualifier to provide for compilation of code that has many variants or configurations.

- Many compiler directives such as %ARCH_NAME, %SYSTEM_NAME, %COMPILER_VERSION, %FILE, etc. to allow compile-time information to be inserted into the compiled program.

- Compiler directives such as %F_FLOAT, %S_FLOAT, etc. to allow fine grained control of floating literal formats (OpenVMS Alpha and OpenVMS I64 systems only)

As a native-mode language, VSI Pascal is integrated into the Common Language Environment. This integration provides VSI Pascal users with:

- Support for OpenVMS interlanguage calling standard

- Access to all OpenVMS system services

- Access to the facilities of the OpenVMS Symbolic Debugger

- Callable interfaces to the OpenVMS Common Run-time Library

- Oracle CDD/Repository support

- Support for the Language-Sensitive Editor/Source Code Analyzer to provide error diagnostics to Language-Sensitive Editor component, cross reference information for Source Code Analyzer component, and support for low-level program design, including the processing of pseudocode.

- Ability for functions to return structured types (other than file types)

Options available to Pascal users at compile time include:

- Run-time checks for array, character string, and subrange bounds

- Run-time checks for arithmetic overflow, valid case selector values, and null pointer variables

- Run-time checks for invalid declarations and illegal GOTO usage

- Generation of information for use by the OpenVMS Symbolic Debugger and the run-time error traceback mechanism

- Creation of an environment file facilitating separate compilation

- Cross-reference listing

- Creating in the listing file a representation of the object code generated by the compiler

- Printing of information-level messages including flagging uses of extensions to the ISO and ANSI Pascal standards

2. Source Code Information

The following source code modules are provided on all available distribution media for this product:

KITINSTAL.COM, LIBDEF.PAS, MTHDEF.PAS,
PASCAL$IVP.PAS, PASCAL.CLD, PASCAL.HLP,
PASDEF.PAS, PASSTATUS.PAS, SIGDEF.PAS,
DTK_MODULES.DAT, FRONT.PAS,
LIB_MODULES.DAT, MTH_MODULES.DAT,
NCS_MODULES.DAT, OTS_MODULES.DAT,
PASCAL$D_FLOAT.PAS, PASCAL$G_FLOAT.PAS,
PASCAL$CMA ROUTINES.PAS,
PASCAL$CVT ROUTINES.PAS,
PASCAL$DTK_ROUTINES.PAS,
PASCAL$LIB_ROUTINES.PAS,
PASCAL$MTH_ROUTINES.PAS,

PASCAL$NCS_ROUTINES.PAS,
PASCAL$OTS_ROUTINES.PAS,
PASCAL$PPL_ROUTINES.PAS,
PASCAL$SMG_ROUTINES.PAS,
PASCAL$SOR_ROUTINES.PAS,
PASCAL$STR_ROUTINES.PAS,
PASSTR$IVP.PAS,
PPL_MODULES.DAT,
RMSUSR.PAS,
SDLPASCAL.EXE,
SMG_MODULES.DAT,
SOR_MODULES.DAT,
STARLET.PAS,
STARLET_MODULES.DAT,
SOR_MODULES.DAT,
CONSTRUCTOR_1.PAS,
FUNCTION_CALLS.PAS,
HANDLER.PAS,
HELLOWORLD.PAS,
IMPLEMENTATION_MODULE.PAS,
INITIAL_STATE_1.PAS,
INITIAL_STATE_2.PAS,
INTERFACE_MODULE.PAS,
LIB$FIND_FILE.PAS,
MAIN_PROGRAM.PAS,
RFA_READ.PAS,
SCHEMA_PARAMETERS.PAS,
SMG_EXAMPLE.PAS,
SYS$ASCTIM_AND_GETTIM.PAS,
SYS$CHECK_ACCESS.PAS, SYS$DCLEXH.PAS,
SYS$DEVICE_SCAN.PAS, SYS$FAO.PAS,
SYS$GETDVI.PAS, SYS$GETJPI.PAS,
SYS$GETQUI.PAS, SYS$GETSYI.PAS,
SYS$GETUAI.PAS, SYS$PROCESS_SCAN.PAS,
SYS$PUTMSG.PAS, SYS$SNDJBC.PAS,
SYS$TRNLNM.PAS, USE_XABDAT.PAS
PASCAL$ACLEDIT_ROUTINES.PAS
PASCAL$CLI_ROUTINES.PAS
PASCAL$CONV_ROUTINES.PAS
PASCAL$DCX_ROUTINES.PAS
PASCAL$EDT_ROUTINES.PAS
PASCAL$FDL_ROUTINES.PAS
PASCAL$LBR_ROUTINES.PAS
PASCAL$MAIL_ROUTINES.PAS
PASCAL$PSM_ROUTINES.PAS
PASCAL$SMB_ROUTINES.PAS
PASCAL$TPU_ROUTINES.PAS
PASCAL$SHOW_VERSIONS.COM
PASCAL$SET_VERSION.COM
PASCAL$DEFAULT_VERSION.COM

The source code modules are provided in order to install and describe the product. Modules include sample test program, help file, example files, and system definition inclusion files.

This source code is provided on an "AS IS" basis without any warranty of any kind either express or implied.

Run-Time Library Redistribution

The VSI Pascal kit may include updated Pascal Run-Time Library shareable images. VSI grants the user a nonexclusive royalty-free worldwide right to reproduce and distribute the executable version of the Run-Time Library designated as PASRTL.EXE or PAS$RTL.EXE and PAS$MSG.EXE (Alpha and I64) (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 Pascal on the user's product disk label and/or on the title page of the documentation for 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.

3. Hardware Requirements

3.1. Processors Supported

Any Alpha system capable of running the OpenVMS Alpha Operating System Version 6.1 to 8.3. Any OpenVMS Integrity Server capable of running the OpenVMS Integrity Operating System Version 8.2 to 8.3.

Refer to the OpenVMS Operating System's Software Product Description (SPD 82.35.XX) for details.

3.2. Disk Space Requirements (Block Cluster Size = 1)

For VSI Pascal for OpenVMS Alpha Systems:

VSI Pascal Compiler:

Disk space required for installation:	25,000 blocks
	(12.5 MB)
Disk space required for permanent use:	21,000 blocks
	(10.5 MB)

Starlet Library Files:

Disk space required for installation:	15,000 blocks
Disk space required for permanent use:	15,000 blocks

VSI Pascal Example Files:

Disk space required for installation:	150 blocks
Disk space required for permanent use:	150 blocks

For VSI Pascal for OpenVMS I64 Systems:

VSI Pascal Compiler:

Disk space required for installation:	45,000 blocks
	(22.5 MB)
Disk space required for permanent use:	45,000 blocks
	(22.5 MB)

Starlet Library Files:

Disk space required for installation:	15,000 blocks
Disk space required for permanent use:	15,000 blocks

VSI Pascal Example Files:

Disk space required for installation:	150 blocks
Disk space required for permanent use:	150 blocks

These block 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 selected.

3.3. Optional Hardware

Floating point intensive applications should be run on configurations with the appropriate hardware support for the floating point data types being used. Consult the Base Operating System Software Product Description for the Floating Point Accelerator or other floating point hardware appropriate for your configuration.

4. Software Licensing

A software license is required in order to use the VSI Fortran 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 Fortran are available only through a Support Agreement or through a new license purchase.

● For AlphaServer systems, the license to use VSI Pascal 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: https://vmssoftware.com/

4.1. License Management Facility Support

VSI Pascal 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.

4.2. Cluster Environment

This layered product is fully supported when installed on any valid and licensed OpenVMS Cluster* configuration without restrictions. Section 3, "Hardware Requirements" of this product's Software Product Description detail any special hardware required by this product.

OpenVMS Cluster configurations are fully described in the OpenVMS Cluster Software Product Description (29.78.XX) and include SEA, Ethernet, and Mixed Interconnect configurations.

4.3. OpenVMS Tailoring Classes

The following OpenVMS classes are required for full functionality of this layered product:

- OpenVMS Required Saveset

- Programming Support

- Utilities

4.4. Optional Software

For Alpha and I64 Systems:

- Oracle CDD/Repository for OpenVMS

- VSI DECset Release 12.7 for OpenVMS includes:

  ○ VSI Language-Sensitive Editor/Source Code Analyzer (LSE/SCA) for OpenVMS Alpha

  ○ VSI Digital Test Manager (DTM) for OpenVMS Alpha

  ○ VSI Performance and Coverage Analyzer (PCA) for OpenVMS Alpha

  ○ VSI Code Management System (CMS) for OpenVMS Alpha

  ○ VSI Module Management System (MMS) for OpenVMS Alpha

4.5. 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.

4.6. Distribution Media

VSI Pascal versions for all supported architectures are available at:
https://vmssoftware.com/products/pascal/

4.7. 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.

4.8. Ordering Information

For information about ordering VSI Pascal for OpenVMS, contact VSI Sales at: `<sales@vmssoftware.com>`.

5. Software Product Services

A variety of service options are available from VSI. For more information, contact VSI or visit https://vmssoftware.com/.