

Customer Software Release Notice
for
VS C Compiler - Version 7.53.01
March 2000

Copyright Getronics 2000



DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

The staff of Getronics has taken due care in preparing this manual; however, nothing contained herein modifies or alters in any way the standard terms and conditions of the Getronics purchase, lease, or license agreement by which this software package was acquired, nor shall Getronics or its subsidiaries be liable for incidental or consequential damages in connection with or arising from the use of the software package, the accompanying manual, or any related manuals.

NOTICE

All Getronics Program Products are licensed to customers in accordance with the terms and conditions of the Getronics Standard Program Products License; no ownership of Getronics Software is transferred and any use beyond the terms of the aforesaid License, without the written authorization of Getronics is prohibited.



Table of Contents

- 1.0 Release Abstract
- 2.0 Prerequisites and Dependencies
 - 2.1 Hardware Prerequisites
 - 2.2 Software Prerequisites
- 3.0 Restrictions and Special Considerations
- 4.0 Enhancements
 - 4.1 Compiler
 - 4.2 Runtime
- 5.0 Problems Corrected
 - 5.1 Compiler
 - 5.2 Runtime
- 6.0 Media Contents
- 7.0 Installation Instructions

1.0 Release Abstract

This document identifies the features, restrictions, and special considerations for Version 7.53.01 of the VS C Compiler and Runtime System. Prior to installation of version 7.53.01 of the VS C Compiler and its associated Runtime System the customer must have version 3.00.14 installed and running a version of this compiler. See section 3.0 "Restrictions and Special Considerations" for more information.

The VS C Compiler supports the C language as described in Kernighan and Ritchie's "The C Programming Language". The VS C Runtime system provides support for VS DMS record access as well as UNIX-style byte access. In addition, it includes routines to perform the math, type conversion, string manipulation, and memory management functions commonly supplied with a C compiler.

2.0 Prerequisites and Dependencies

2.1 Hardware

2.1.1 Configuration

The VS C Compiler may be run on any VS configuration supported by OS 7.53.00 or greater. At least two mb of modifiable data area (MDA) is recommended.

2.2 Software

2.2.0 VS Operating System

Version 7.53.01 of the VS C compiler and the object code it creates must be run on a VS Operating System version 7.53.00 or greater to have full YEAR 2000 support and take advantage of the new built-in functions and other new features documented further in this document.

2.2.1 C Runtime System

The VS C Runtime System distributed with this Version of C Compiler IS NOT COMPATIBLE with versions of the C Runtime System prior to Version 3.00.03. All C programs making up a given "run unit", that use the I/O portions of the runtime library, but haven't been recompiled against Version 3.00.03 or greater, MUST BE RECOMPILED AND RELINKED against current Version of runtime library.

2.2.2 VS Shell

To execute a program that uses the shell for I/O redirection, you must explicitly link in the runtime. This may be done by specifying the runtime library in LINKLIB and LINKVOL on the compiler's OPTIONS screen or by running the linker after compilation.

2.2.3 Forms File

The compiler's forms file CCFORM, which contains most of the screens displayed by the compiler, must be in @SYSFRM@ on the system (IPL) volume for the compiler to work. The compiler's installation procedure will perform this function. Any attempts to invoke the compiler without a properly installed forms file will result in a FAILED COMPILATION.

If the C compiler and the forms file SETOPTS utility are run at the same time, and the SETOPTS utility is accessing one of the compiler's screens, this option screen will not display during the compilation process.

The forms master file (@FORMS@ in @SYSFRM@) must contain a record for C Compiler referencing the name of the C forms file (CCFORM). The compiler's installation procedure will ensure that this record is present.

3.0 Restrictions and Special Considerations

Because of runtime incompatibilities with releases prior to 3.00.03 and many language incompatibilities with releases prior to 3.00.14 it is highly recommended that 3.00.14 be installed and running prior to upgrading to release 7.53.01 of the compiler. The resulting runtime and syntax problems which may arise cannot be determined nor dealt with if this recommendation is not followed.

4.0 Enhancements

4.0.1 Forms File

The C Compiler forms file now has a PF 16 options to allow the user to exit the compiler without having to cancel the compilation.

4.1 VS C Compiler

4.1.1 When a program is compiled using object form 0 with no static runtime library specified there is a timestamp placed in the lengths block.

This can be used to help determine the DATSEL option by versions 7.52.05 and greater of the LINKER. The compilation time will also appear on the link map as it currently does from all object form 1 object modules.

4.1.2 Listing

The listing now provides the full date in MM/DD/YYYY format.

4.1.3 Date Functions

The DATE4 TIME STAMP parameter has been added to provide the return of a 4-digit year where the DATE TIME STAMP only provided a 2-digit year.

4.2 VS C Runtime

4.2.1 WRTDATE4 has been added as the runtime module to support 4-digit year functions.

4.2.2 Users can now specify RELEASE=NO for an open.

5.0 Problems Corrected

5.1 VS C Compiler Version 7.53.01

When VSSHELL was called from an application without parameters the program would sometimes fail. The type of failure would vary depending upon what the registers contained at the time. This problem has been corrected.

5.2 VS C Runtime

The following PTRs were corrected since the release of version 3.00.14:

<i>PTR #</i>	<i>PTR Description</i>
G410010249	Opening and closing Version 2 stream files with OPEN/CLOSE or FOPEN/FCLOSE no longer causes stream files to be corrupted if using a READ/WRITE open mode.
M410010881	Version 2 stream file corruption no longer occurs when a write is performed and the new file position is less than the actual file length. The result is that all data from the file position to the end of the file is zeroed out.

G800017924	When LSEEK is called to position a stream file pointer close to a block boundary, a subsequent read over the block would only read data correctly up to the boundary. This problem has been corrected.
G500016840	When doing an FGETS on a null file garbage is no longer returned with a valid pointer.
P400008830	When attempting to write to a file when no more extents are available the file is properly closed instead of the program aborting.
P100015320	Reading a tape file with stream access no longer give an erroneous EOF result.
P100011961	The user is now allowed the option of not defining all alternate indexes and will not receive an error message.

6.0 Media Contents

LIBRARY = CINSTALL

Module	Version	Description
INSTALL	7.53.01	Installation procedure

LIBRARY = CSRNLIB

Module	Version	Description
CC75301	N/A	Customer Software Release Notice

LIBRARY = VSC75301 VS C Compiler

Module	Version	Description
CC	7.53.01	Main compiler module
VSSHLL	"	The VS Shell
WLCPP	"	Preprocessor
WLCPASS1	"	Syntax and semantics
WLCOPT	"	Optimizer
WLCPASS3	"	Code generator
WLCPASS4	"	Program formatter

LIBRARY = VSF75301 - Compiler Forms Files

Module	Version	Description
CCFORM	000004	Forms File
FIM	1.03.00	Forms Index Manager

LIBRARY = VSR75301 - VS C Runtime System

All VS C Runtime modules are version 7.53.01

Module	Description
@CCCRTF@	Static Subroutine File
ACOS	C runtime routine
ASCTIME	"
ASSERTH	"
ASIN	"
ATAN	"
ATAN2	"
ATOF	"
ATOI	"
ATOL	"
BUILTINH	"
CALLOC	"
CEIL	"
CGLOBALC	runtime globals
CLAYERHC	-only header
CLOSE	C runtime routine
COPEN	"
COS	"
COSH	"
CREAT	"
CTIME	"
CTYPEH	C runtime header
CUSERIDC	runtime routine
DELR	"
ERRNOH	C runtime header
EXIT	C runtime routine
EXP	"
FCBH	CRS system layer header
FCLOSE	C runtime routine
FDELR	"
FDOPEN	"
FENDFILE	"

FFLUSH	"
FGETC	"
FGETR	"
FGETS	"
FLOATH	"
FLOOR	"
FMOD	"
FOPEN	"
FPRINTF	"
FPUTC	"
FPUTR	"
FPUTS	"
FREAD	"
FREADR	"
FREE	C runtime routine
FREOPEN	"
FREXP	"
FSCANF	"
FSEEK	"
FTELL	"
FTRUNC	"
FWRITE	"
FWRITER	"
GETRN	"
GETS	"
GETW	"
HEAPH	C runtime header
HYPOT	C runtime routine
IODEFSHC	runtime header
ISOPEN	C runtime routine
ITOA	"
LCLTIME	"
LIMITSHC	runtime header
LOG	C runtime routine
LOG10	"
LSEEK	"
MALLOC	"
MATHERR	"
MATHH	C runtime header
MEMCCPYC	runtime routine
MEMCHR	C runtime routine
MEMCMP	"
MEMCPY	"
MEMORYHC	runtime header
MEMSET	C runtime routine
OPEN	"
PALLOC	CRS -only routine
PFREE	CRS -only routine

POW	C runtime routine
POWER10	"
PRINTF	C Runtime Routine
PUTRN	"
PUTS	"
PUTW	C runtime routine
QSORT	"
RAND	"
READ	"
READR	"
REALLOC	"
REWIND	"
SCANF	"
SETBUF	"
SETJMP	"
SETJMPHC	runtime header
SIN	C runtime routine
SINH	"
SPALLOC	"
SPFREE	"
SPRINTF	"
SQRT	"
SSCANF	"
SSORT	"
STDDEFHC	runtime header
STDIOH	"
STDLIBH	"
STDTYPEH	"
STRCAT	C runtime routine
STRCHR	"
STRCMP	"
STRCPY	"
STRCSPN	"
STRINGHC	runtime header
STRLEN	C runtime routine
STRMCMVRT	C stream convert routine
STRNCATC	runtime routine
STRNCMP	"
STRNCPY	"
STRPBRK	"
STRRCHR	"
STRSPN	"
STRTOK	"
STRTOL	"
STRTOUL	"
SYSERR	HCRS system header
SYSINTH	"
TAN	C runtime routine

TANH	"
TIME	C runtime routine
TIMEH	C runtime header
TOLOWERC	runtime routine
TOUPPER	"
TRUNC	"
UEXIT	"
UFILBUF	"
UFLSBUF	"
UNGETC	"
UNLINK	"
V3STREAM	C stream convert proc.
VS2UNERR	C -only routine
VSTIME	C runtime routine
WANGH	Wang VS C Standard Header
WLPASGNB	C -only routine
WLPGOTO	"
WLPSSIG	"
WLPSTKCL	"
WLRATA	"
WLRBMSG	"
WLRCVBF	"
WLRCVDF	"
WLRCVF	"
WLRDASNC	"
WLRDATAN	"
WLRDEXP	"
WLRDLG	"
WLRDSNCH	"
WLRDSNCO	"
WLRDSORT	"
WLRDTANH	"
WLRDTNCT	"
WLRDXPDC	"
WLREVOL	"
WLRMTHCH	"
WLRSEERR	"
WLXHALT	"
WRITE	C runtime routine
WRITER	"
WRS2WRTH	CRS conversion header
WRTALLOC	CRS system layer routine
WRTBTRAN	"
WRTCFCB	"
WRTCLOSE	"
WRTCOMIT	"
WRTRDATE	"
WRTRDATE4	"

WRTDELET	"
WRTDEXT	"
WRDISP	"
WRTDMS	CRS -only routine
WRTEXT	CRS system layer routine
WRTFFCB	"
WRTFIND	"
WRTFLUSH	"
WRTFREE	"
WRTFRES	"
WRTFXRT	"
WRTGET	"
WRTGXRT	"
WRTHOLD	CRS system layer routine
WRTMODE	"
WRTOPEN	"
WRTPUT	"
WRTREAD	"
WRTRELS	"
WRTROLBK	"
WRTRWRT	"
WRTSCRAT	"
WRTSECND	"
WRTSEEK	"
WRTSETRC	"
WRTSHDLR	CRS -only routine
WRTTFIND	CRS system layer routine
WRTTIME	"
WRTTOP	"
WRTTRUNC	"
WRTWRITE	"
WRTWSBUF	CRS -only routine
XFCBH	C -only header
XFIDTYPE	C runtime routine
XLABEL	"
XLRSEERR	"
XPRINTF	"
XPRNAME	"
XPTH#FCB	"
XPTHNAME	"
XSCANF	"

7.0 Installation Instructions

The distribution volume for the VS C Compiler and Runtime System contains the following five libraries:

Library CINSTALL:

File INSTALL - an installation procedure for the compiler and runtime system.

Library CSRNLIB:

File CC75301 - the customer software release notice.

Library VSC75301:

The seven compiler modules (CC is the compiler name).

Library VSF75301:

The forms file (CCFORM) containing the various compiler screens.

Library VSR75301:

The VS C Runtime System routines and the Version 3 stream file conversion utility.

To install the compiler and runtime take the following steps:

- 1) Make sure that the compiler is not currently in use on your system and that it is not currently 'open' under the auspices of the VS FASTLINK utility.
- 2) Mount the distribution volume.
- 3) Run VS BACKUP and restore the volume to any volume on the system.
- 4) Run the procedure INSTALL from CINSTALL. It will install the compiler and the runtime system on the volumes of your choice. The procedure will also take note of existing versions and offer to back them up.