

VISUALDSP⁺⁺ 2.0

C /C++ Compiler and

Library

Manual for ADSP-219x DSPs

Revision 3.1, October 2001

Part Number
82-000390-03

Analog Devices, Inc.
Digital Signal Processor Division
One Technology Way
Norwood, Mass. 02062-9106



Copyright Information

©1996–2001 Analog Devices, Inc., ALL RIGHTS RESERVED. This document may not be reproduced in any form without prior, express written consent from Analog Devices, Inc.

Printed in the USA.

Disclaimer

Analog Devices, Inc. reserves the right to change this product without prior notice. Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use; nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under the patent rights of Analog Devices, Inc.

Trademark and Service Mark Notice

The Analog Devices logo, VisualDSP, and the VisualDSP logo are registered trademarks; VisualDSP++, VDK, and the VisualDSP++ logo are trademarks of Analog Devices, Inc.

Adobe and Acrobat are registered trademarks of Adobe Corporation.

Microsoft and Windows are registered trademarks and Windows NT is a trademark of Microsoft Corporation.

All other brand and product names are trademarks or service marks of their respective owners.

Rev. 3.1

CONTENTS

INTRODUCTION

Supported Processors	1-2
For Additional Product Information.....	1-2
For Technical or Customer Support	1-3
Purpose of This Manual.....	1-3
Intended Audience	1-4
Manual Contents Description.....	1-4
What's New in This Manual	1-5
Related Documents	1-5
Conventions.....	1-6

COMPILER

Overview	2-1
Compiler Command-Line Interface.....	2-3
Running the Compiler.....	2-4
Compiler Command Line Switches	2-8
C/C++ Compiler Switch Summaries	2-8
C/C++ Mode Selection Switch Descriptions.....	2-19

CONTENTS

C/C++ Compiler Common Switch Descriptions	2-19
C++ Mode Compiler Switch Descriptions	2-41
Data Type Sizes.....	2-47
C/C++ Compiler Language Extensions	2-49
Inline Function Support Keyword (inline)	2-51
Inline Assembly Language Support Keyword (asm)	2-52
Assembly Construct Template	2-54
Assembly Construct Operand Description.....	2-57
Assembly Constructs with Multiple Instructions.....	2-60
Assembly Construct Reordering and Optimization	2-60
Assembly Constructs with Input and Output Operands	2-61
Assembly Constructs and Macros	2-63
Dual Memory Support Keywords (pm dm)	2-63
Memory Keywords and Assignments/Type Conversions	2-66
Memory Keywords and Function Declarations/Pointers	2-67
Memory Keywords and Function Arguments	2-67
Memory Keywords and Macros	2-68
Placement Support Keyword (section)	2-69
Boolean Type Support Keywords (bool, true, false)	2-69
Pointer Class Support Keyword (restrict)	2-70
Variable Length Array Support	2-70
Non-Constant Aggregate Initializer Support	2-72
Indexed Initializer Support	2-73
Aggregate Constructor Expression Support	2-75

Fractional Type Support	2-76
Format of Fractional Literals	2-76
Conversions Involving Fractional Values	2-77
Fractional Arithmetic Operations	2-77
Mixed Mode Operations	2-78
Saturated Arithmetic	2-78
Preprocessor Generated Warnings	2-79
C++-Style Comments	2-79
Compiler Intrinsic Functions	2-79
Access to System Registers	2-80
I/O Space Read or Write	2-82
Interrupt Control.....	2-83
Mode Control.....	2-83
Preprocessor Features	2-84
Predefined Macros	2-85
Header Files	2-87
Writing Macros	2-88
C/C++ Run-Time Model.....	2-90
Using the Run-Time Header	2-90
Interrupt Table and Interface	2-91
Stack Frame.....	2-92

CONTENTS

Parts of the Stack Frame	2-94
General System-wide Specifications	2-95
Return Values	2-96
Procedure Call and Return	2-97
Miscellaneous Information	2-98
Register Classification	2-99
Complete List of Registers	2-100
C/C++ and Assembly Language Interface	2-103
Writing C/C++-Callable Assembly Routines	2-103
Calling C/C++ Routines From Assembly	2-106
Using Mixed C/C++/Assembly Naming Conventions	2-107
C++ Programming Examples	2-108
Using Fract Support	2-109
Using Complex Support	2-110
C/C++ Compiler Glossary	2-112
C/C++ RUN-TIME LIBRARY	
Overview	3-1
C and C++ Run-Time Libraries Guide	3-3
Calling Library Functions	3-3
Using the Compiler's Built-In C Library Functions	3-4
Linking Library Functions	3-5
Working With Library Source Code	3-6
Working With Library Header Files	3-7
C Run-Time Library Header File Descriptions	3-9

Library Structure.....	3-13
Documented Library Functions	3-14
Abridged C++ Library Support	3-17
Embedded C++ Library Header Files	3-18
C++ Header Files for Library Facilities.....	3-21
Embedded Standard Template Library Header Files	3-22
C Run-Time Library Reference	3-25
abort	3-26
abs.....	3-27
acos	3-28
asin.....	3-29
atan	3-30
atan2	3-31
atexit	3-32
atof.....	3-33
atoi.....	3-34
atol.....	3-35
bsearch	3-36
calloc	3-38
ceil	3-39
clear_interrupt.....	3-40
cos.....	3-42
cosh.....	3-43
disable_interrupts	3-44

CONTENTS

div	3-45
enable_interrupts.....	3-46
exit.....	3-47
exp	3-48
fabs	3-49
floor	3-50
fmod	3-51
free	3-52
frexp	3-53
interrupt	3-54
io_space_read	3-57
io_space_write.....	3-58
isalnum	3-59
isalpha.....	3-60
iscntrl.....	3-61
isdigit.....	3-62
isgraph	3-63
isinf	3-64
islower.....	3-66
isnan	3-67
isprint	3-69
ispunct	3-70
isspace	3-71
isupper	3-72

isxdigit.....	3-73
labs.....	3-74
ldexp	3-75
ldiv	3-76
log	3-77
log10	3-78
longjmp	3-79
malloc	3-81
memchr	3-82
memcmp	3-83
memcpy	3-84
memcpy_from_shared	3-85
memcpy_to_shared	3-86
memmove	3-87
memset	3-88
mode_change	3-89
modf	3-91
perror	3-92
pow	3-93
printf.....	3-94
qsort.....	3-95
raise.....	3-97
rand.....	3-99
realloc.....	3-100

CONTENTS

setjmp	3-101
signal	3-102
sin.....	3-105
sinh.....	3-106
srand.....	3-107
sqrt	3-108
strcat	3-109
strchr	3-110
strcmp.....	3-111
strcoll.....	3-112
strcpy	3-113
strcspn	3-114
strerror	3-115
strlen.....	3-116
strncat.....	3-117
strncmp.....	3-118
strncpy	3-119
strpbrk	3-120
strrchr	3-121
strspn	3-122
strstr	3-123
strtod	3-124
strtodf	3-126
strtok	3-128

strtol.....	3-130
strtoul.....	3-132
strxfrm	3-134
sysreg_read	3-135
sysreg_write	3-137
tan.....	3-139
tanh.....	3-140
tolower	3-141
toupper.....	3-142
va_arg.....	3-143
va_end.....	3-144
va_start.....	3-145

DSP RUN-TIME LIBRARY

Overview	4-1
DSP Run-Time Library Guide.....	4-2
Calling Library Functions	4-2
Linking DSP Library Functions	4-3
Working with Library Source Code	4-3
Data Types	4-4
DSP Header Files	4-4
DSP Run-Time Library Reference	4-22
a_compress	4-23
a_expand	4-24
arg.....	4-25

CONTENTS

autocoh	4-26
autocorr	4-27
cabs.....	4-28
cadd	4-29
cdiv	4-30
cexp	4-31
cfft	4-32
cffttrad4	4-34
cfft2d	4-36
cfirc	4-38
clip	4-40
cmlt	4-41
conj.....	4-42
convolve.....	4-43
conv2d	4-45
conv2d3x3	4-46
copysign.....	4-47
countones.....	4-48
crosscoh	4-49
crosscorr.....	4-50
csub	4-51
fir	4-52
fir_decima	4-54
fir_interp.....	4-55

gen_bartlett	4-56
gen_blackman	4-58
gen_gaussian	4-59
gen_hamming	4-60
gen_hanning	4-61
gen_harris	4-62
gen_kaiser	4-63
gen_rectangular	4-64
gen_triangle	4-65
gen_vonhann	4-66
histogram	4-67
ifft	4-68
iffttrad4	4-70
ifft2d	4-72
iir	4-74
max	4-76
mean	4-77
min	4-78
mu_compress	4-79
mu_expand	4-80
norm	4-81
polar	4-82
rfft	4-83
rfftrad4	4-85

CONTENTS

rfft2d	4-87
rms	4-89
twidfft.....	4-90
var	4-92
zero_cross.....	4-93

COMPILER LEGACY SUPPORT

Overview.....	A-1
Tools Differences	A-1
C/C++ Compiler and C/C++ Run-time Library	A-3
Segment Placement Support Keyword Changed to Section....	A-3
G21 Compatibility Call.....	A-3
Support for G21-Based Options And Extensions	A-3
ANSI C Extensions.....	A-4
Compiler Switch Modifications.....	A-5
New and Obsolete Warnings	A-8
Run-Time Model.....	A-9
C/C++ Run-Time Library	A-9

INDEX