

I INDEX

Symbols

@ filename (command file) compiler switch [2-19](#)

μ -law

companders

ADSP-2106x [4-4](#)

ADSP-2116x [5-4](#)

compression function

ADSP-2106x [4-63](#)

ADSP-2116x [5-67](#)

expansion function

ADSP-2106x [4-64](#)

ADSP-2116x [5-68](#)

Numerics

-21060 (ADSP-21060 macro) compiler switch [2-20](#)

-21061 (ADSP-21061 macro) compiler switch [2-20](#)

-21062 (ADSP-21062 macro) compiler switch [2-20](#)

-21065L (ADSP-21065L macro) compiler switch [2-21](#)

-21160 (ADSP-21160 macro) compiler switch [2-21](#)

-21161 (ADSP-21161 macro) compiler switch [2-21](#)

A

-A (assert) compiler switch [2-21](#)

Abend (See abort function)

abort (abnormal program end) function [3-30](#)

Abridged C++ Library [3-21](#)

abs (absolute value, int) function [3-31](#)

Absolute value (See abs, fabs, fabsf, labs functions)

acos, acosf (arc cosine) function [3-32](#)

ADSP-2106x

Built-in DSP functions [4-11](#)

DSP Run-time library reference [4-13–4-84](#)

ADSP-2106x functions

a_compress [4-14](#)

a_expand [4-15](#)

autocoh [4-16](#)

autocorr [4-17](#)

biquad [4-18](#)

cabsf [4-21](#)

cexpf [4-22](#)

cfftN [4-23](#)

copysign, copysignf [4-26](#)

cot, cotf [4-27](#)

crosscoh [4-28](#)

crosscorr [4-29](#)

INDEX

- favg, favgf 4-30
- fclip, fclipf 4-31
- fir 4-32
- fmax, fmaxf 4-34
- fmin, fminf 4-35
- histogram 4-48
- idle 4-50
- ifftN 4-51
- iir 4-54
- matmul 4-58
- matscaltmult 4-60
- matsub 4-61
- mean 4-62
- mu_compress 4-63
- mu_expand 4-64
- poll_flag_in 4-65
- rfftN 4-67
- rms 4-70
- rsqrt, rsqrtf 4-71
- set_flag 4-72
- set_semaphore 4-74
- timer_off 4-75
- timer_on 4-77
- timer_set 4-79
- timer0_off, timer1_off 4-76
- timer0_on, timer1_on 4-78
- timer0_set, timer1_set 4-81
- var 4-83
- zero_cross 4-84
- ADSP-2116x
 - Built-in DSP functions 5-12
 - DSP Run-time library reference 5-15–5-93
- ADSP-2116x functions
 - a_compress 5-16
 - a_expand 5-17
 - autocoh 5-18
 - autocorr 5-19
 - cabsf 5-20
 - cexpf 5-21
 - cfftN 5-22
 - copysign, copysignf 5-25
 - cot, cotf 5-26
 - crosscoh 5-27
 - crosscorr 5-28
 - cvecdot 5-29
 - cvecsadd 5-30
 - cvecsmult 5-31
 - cvecssub 5-32
 - cvecvadd 5-33
 - cvecvmlt 5-34
 - cvecvsub 5-35
 - favg, favgf 5-36
 - fclip, fclipf 5-37
 - fft_mag 5-38
 - fir 5-39
 - fmax, fmaxf 5-41
 - fmin, fminf 5-42
 - histogram 5-54
 - idle 5-56
 - ifftN 5-57
 - iir 5-60
 - matadd 5-62
 - matmul 5-63
 - matscaltmult 5-64
 - matsub 5-65
 - mean 5-66

- mu_compress 5-67
- mu_expand 5-68
- poll_flag_in 5-69
- rfft2_N 5-74
- rfftN 5-71
- rms 5-77
- rsqrt, rsqrtf 5-78
- set_flag 5-79
- set_semaphore 5-81
- timer_off 5-82
- timer_on 5-83
- timer_set 5-84
- var 5-85
- vecdot 5-86
- vecsadd 5-87
- vecsmult 5-88
- vecssub 5-89
- vecvadd 5-90
- vecvmlt 5-91
- vecvsub 5-92
- zero_cross 5-93
- Aggregate assignment support
(compiler) 2-79
- Aggregate constructor expression
support 2-79
- A-law
 - companders
 - ADSP-2106x 4-4
 - ADSP-2116x 5-4
 - compression function
 - ADSP-2106x 4-14
 - ADSP-2116x 5-16
 - expansion function
 - ADSP-2106x 4-15
 - ADSP-2116x 5-17
- Algebraic functions (See Math
functions)
- aligned-stack (align stack) compiler
switch 2-22
- Allocate memory (See calloc, free,
malloc, realloc functions)
- Alphabetic character test (See
isalpha function)
- Alphanumeric character test (See
isalnum function)
- alter macro 2-155
- alternate registers 2-129, 2-134
- analog (Analog C compilation)
compiler switch 2-18
- ANSI standard compiler 2-18, 2-26
- Arc (See acos, asin, atan, atan2
functions)
- Arguments & Return Transfer
2-140
- Arithmetic
 - saturated 2-87
- Array search, binary (See bsearch
function)
- Array storage 2-143
- ASCII string (See atof, atoi, atol
functions)
- asin, asinf (arc sine) functions 3-33
- asm compiler keyword 2-55
(See Assembly language support
keyword (asm))
- asm_sprt.h file 2-153
- Assembler (easm21k) 2-177
- Assembler for BLACKfin DSPs
(easmlkfn) 2-2

INDEX

Assembly language support keyword
(asm) [2-58](#), [2-166](#)

Assignments [2-70](#)

atan, atanf (arc tangent) functions
[3-34](#)

atan2, atan2f (arc tangent division)
functions [3-35](#)

atexit (select exit function) function
[3-36](#)

atof (string to double) function [3-40](#)

atoi (string to int) function [3-38](#)

atol (string to long) function [3-39](#)

-auto-inline (auto inline) compiler
switch [2-23](#)

Automatic variables [2-68](#)

average (mean of 2 int) function
[3-40](#)

B

B (Base) Registers [2-177](#)

background registers [2-129](#), [2-134](#)

Binary array search (See bsearch
function)

Boolean Type Support Keywords
(bool, true, false) [2-74](#)

Boolean type support keywords
(bool, true, false) [2-74](#)

bsearch (array search, binary)
function [3-41](#)

-build-lib (build library) compiler
switch [2-23](#)

Built-in DSP functions

ADSP-2106x [4-11](#)

ADSP-2116x [5-12](#)

C

-C (comments) compiler switch
[2-23](#)

-c (compile only) compiler switch
[2-23](#)

C language extensions

C++ style comments [2-56](#)

Preprocessor generated warnings
[2-56](#)

C run-time library reference [3-29](#)–
[3-164](#)

-c++ (C++ mode) compiler switch
[2-18](#)

C++ language extension
fract data type [2-56](#)

C++ member functions in assembly
[2-159](#)

C++ programming examples [2-162](#)
complex support [2-163](#)

fract support [2-163](#)

C++ style comments [2-80](#)

C/Assembly Interfacing (See Mixed
C/assembly programming)

C/C++ compiler
overview [2-1](#)

C/C++ language extensions [2-55](#)
Aggregate assignments [2-56](#)

asm keyword [2-58](#)

bool keyword [2-55](#)

false keyword [2-55](#)

Indexed initializers [2-56](#)

inline keyword [2-57](#)

non-constant initializers [2-56](#)

pm and dm keywords [2-68](#)

- section keyword [2-55](#)
- true keyword [2-55](#)
- variable length arrays [2-56](#)
- C/C++ run-time environment
 - (See [Mixed C/C++/assembly programming](#))
 - definition [2-122](#)
 - for cc21k [2-177](#)
- C/C++ run-time library guide [3-3–3-28](#)
- Call preserved registers [2-131](#)
- Calling library functions [3-3](#)
- calloc (allocate initialized memory)
 - function [3-43](#)
- cc21k (ADSP-21xxx C/C++ compiler) [2-177](#)
- ccall macro [2-154](#)
- ceil, ceilf (ceiling) function [3-45](#)
- char storage [2-143](#)
- Character string search (See [strchr](#) function)
- Character string search, recursive
 - (See [strrchr](#) function)
- clear_interrupt (clear pending)
 - function [3-46](#)
- clip (x by y, int) function [3-51](#)
- command line (see [Compiler command-line syntax](#))
- Compare memory range (See [memcmp](#) function)
- Compare, strings (See [strcmp](#), [strcoll](#), [strcspn](#), [strpbrk](#), [strncmp](#), [strstr](#) functions)
- Compiler
 - glossary [2-177](#)
 - reference [2-121](#)
 - registers [2-129](#)
- Compiler command-line switches
 - nothreads (disable thread-safe build) [2-32](#)
 - proc identifier [2-36](#)
 - threads (enable thread-safe build) [2-39](#)
 - Wremarks [2-41](#)
- Compiler common switches
 - auto-inline (auto inline) [2-23](#)
 - ipa [2-29](#)
 - l (link library) [2-29](#)
 - no-alttok (disable alternative tokens) [2-30](#)
 - no-restrict (disable restrict) [2-32](#)
 - O (enable optimizations) [2-33](#)
 - Os (enable code size optimizations) [2-33](#)
 - PP (omit line numbers and run) [2-33](#)
 - restrict [2-37](#)
 - version (display version) [2-27](#)
- Concatenate, string (See [strcat](#), [strncat](#) function)
- const keyword [2-19](#)
- const-read-write compiler switch [2-24](#)
- Control
 - of inlining [2-51](#)
 - of optimization [2-49](#)

INDEX

Control character test (See `isctrnl` function)
Convert, characters (See `tolower`, `toupper` functions)
Convert, strings (See `atof`, `atoi`, `atol`, `strtod`, `strtok`, `strtol`, `strtoul`, functions)
Convert, strings (See `atof`, `atoi`, `atol`, `strtok`, `strtol`, `strtoul`, functions)
Copy memory range (See `memcpy` function)
Copy, string (See `strcpy`, `strncpy` function)
`cos`, `cosf` (cosine) functions 3-52
`cosh`, `coshf` (hyperbolic cosine) functions 3-53
C-type functions
 `isalnum` 3-81
 `isalpha` 3-82
 `isctrnl` 3-83
 `isdigit` 3-84
 `isgraph` 3-85
 `islower` 3-86
 `isprint` 3-87
 `ispunct` 3-88
 `isspace` 3-89
 `isupper` 3-90
 `isxdigit` 3-91
 `tolower` 3-159
 `toupper` 3-160
Customer support 1-3

D
-D (define macro) compiler switch 2-24, 2-39
Data storage formats 2-143
Deallocate memory (See `free` function)
Debug information 2-37
-default-linkage (assembler, C, or C++) compiler switch 2-24
Define signal handling (See `signal` function)
Digit character test (See `isdigit` function)
`div` (division, int) function 3-54
Division (See `div`, `ldiv` functions)
`dm` (See Dual memory support keywords (pm dm))
double storage format 2-144
-double-size-32 (IEEE single-precision double) compiler switch 2-24
-double-size-64 (IEEE double-precision double) compiler switch 2-24
-dry (terse -dry-run) compiler switch 2-25
-dry-run (verbose dry-run) compiler switch 2-25
DSP run-time library guide 4-2, 5-2
Dual memory support keywords (pm dm) 2-68

E

-E (stop after preprocessing)

 compiler switch [2-26](#)

-EE (run after preprocessing)

 compiler switch [2-26](#)

Embedded C++ header files

 complex [3-22](#)

 exception [3-22](#)

 fract [3-22](#)

 fstream [3-22](#)

 iomanip [3-23](#)

 ios [3-23](#)

 iosfwd [3-23](#)

 iostream [3-23](#)

 istream [3-23](#)

 new [3-23](#)

 ostream [3-23](#)

 sstream [3-24](#)

 stdexcept [3-24](#)

 streambuf [3-24](#)

 string [3-24](#)

 strstream [3-24](#)

Embedded Standard Template

 Library [3-26](#)

End (See `atexit`, `exit` functions)

entry macro [2-153](#)

Example

 Call Preserved Registers (Pass
 Array) [2-175](#)

 Inline Assembly (Add) [2-166](#)

 Macros to Manage the Stack
 [2-167](#)

 Non-Leaf Routines That Make
 Calls (RMS) [2-173](#)

Registers for Arguments and

 Return (Add 2) [2-172](#)

Scratch Registers (Dot Product)
[2-168](#)

Stack for Arguments and Return
(Add 5) [2-171](#)

Void Functions (Delay) [2-170](#)

`exit` (program termination) function
[3-55](#)

`exit` macro [2-153](#)

`exp`, `expf` (exponential) function
[3-56](#)

Exponential (See `exp`, `expf`, `ldexp`,
`ldexpf` functions)

-extra-keywords (not quite -analog)
 compiler switch [2-26](#)

F

`fabs`, `fabsf` (absolute value, float)
 functions [3-57](#)

`false` (See Boolean type support
 keywords (`bool`, `true`, `false`))

Far jump return (See `longjmp`,
`setjmp` functions)

File

 extension [2-7](#), [2-19](#)

 searches [2-6](#)

File extension [2-4](#), [2-7](#)

Fill memory range (See `memset`
 function)

Finish processing argument list (See
 `va_end` function)

-flags (command line input)
 compiler switch [2-27](#)

INDEX

float storage format [2-144](#)
Float.h – Floating Point [3-9](#)
Floating point [2-48](#)
floor, floorf (integral value)
 functions [3-58](#)
fmod, fmodf (find remainder)
 functions [3-59](#)
free (deallocate memory) functions
 [3-60](#)
frexp, frexpf (fraction/exponent)
 functions [3-61](#)
fstreams.h [3-28](#)
Function
 arguments/return value transfer
 [2-140](#)
 call return address [2-167](#)
 entry (prologue) [2-135](#), [2-167](#)
 exit (epilogue) [2-135](#), [2-167](#)
functions
 Primitive I/O [3-13](#)

G

-g (generate debug information)
 compiler switch [2-27](#)
gen_bartlett (generate bartlett
 window) function [3-62](#), [4-36](#),
 [5-43](#)
gen_blackman (generate blackman
 window) function [4-38](#), [5-45](#)
gen_gaussian (generate gaussian
 window) function [4-39](#), [5-46](#)
gen_hamming (generate hamming
 window) function [4-40](#), [5-47](#)

gen_hanning (generate hanning
 window) function [4-41](#), [5-48](#)
gen_harris (generate harris window)
 function [4-42](#), [5-49](#)
gen_kaiser (generate kaiser window)
 function [4-43](#), [5-50](#)
gen_rectangular (generate
 rectangular window) function
 [4-45](#), [5-51](#)
gen_triangle (generate triangle
 window) function [4-46](#), [5-52](#)
gen_vonhann (generate von hann
 window) function [4-48](#)
Get locale pointer (See localeconv
 function)
Get next argument in list (See
 va_arg function)
getenv (get string definition from
 operating system) function
 [3-62](#)
gets macro [2-154](#)
Graphical character test (See isgraph
 function)

H

-H (list *.h) compiler switch [2-28](#)
Header files [2-32](#), [2-109](#)
 working with [3-6](#)
Header files (ADSP-2106x) [4-4](#)–
 [4-9](#)
 21060.h [4-4](#)
 21065l.h [4-4](#)
 asm_sprt.h [4-4](#)
 comm.h [4-4](#)

INDEX

heap_switch (set heap for dynamic memory allocation) function [3-76](#)
-help (command-line help) compiler switch [2-28](#)
Hexadecimal digit test (See isxdigit function)
-HH (list *.h and compile) compiler switch [2-28](#)
Hyperbolic (See cosh, coshf, sinh, sinh, tanh, tanhf functions)

I

-I (include search directory) compiler switch [2-28](#), [2-32](#)
I (Index) registers [2-177](#)
IEEE Single / Double Precision Description [2-143](#)
-include (include file) compiler switch [2-29](#)
Indexed initializer support (compiler) [2-78](#)
Initialize argument list (See va_start function)
Inline assembly code [2-121](#)
Inline assembly language support keyword (asm) [2-58](#)
construct I/O operands [2-66](#)
construct optimization [2-65](#)
construct template [2-59](#)
construct template operands [2-62](#)
constructs with multiple instructions [2-64](#)
macros containing asm [2-67](#)

Inline function support keyword (inline) [2-19](#), [2-55](#), [2-57](#)
-instantall (instantiate all) C++ mode compiler switch [2-43](#)
instantiate [2-43](#)
-instantused (instantiate used) C++ mode compiler switch [2-43](#)
int storage format [2-143](#)
Integer [2-48](#)
Interfacing C/C++ & assembly (See Mixed C/C++/assembly programming)
Intermediate files [2-38](#)
interrupt (interrupt handling) functions [3-79](#)
Interrupt table [2-147](#)
Interrupts (See clear_interrupt, interruptf, interrupts, signal, raise functions)
Inverse (See acos, asin, atan, atan2 functions)
iomanip.h header file [3-28](#)
iostream.h header file [3-28](#)
isalnum (alphanumeric character test) function [3-81](#)
isalpha (alphabetic character test) function [3-82](#)
iscntrl (control character test) function [3-83](#)
isdigit (digit character test) function [3-84](#)
isgraph (graphical character test) function [3-85](#)

islower (lower case character test)
 function [3-86](#)
 isprint (printable character test)
 function [3-87](#)
 ispunct (punctuation character test)
 function [3-88](#)
 isspace (white space character test)
 function [3-89](#)
 isupper (upper case character test)
 function [3-90](#)
 isxdigit (hexadecimal digit test)
 function [3-91](#)

K

Keywords (compiler) (See Compiler
 C/C++ extensions)

L

L (Length) registers [2-177](#)
 -L (library search directory)
 compiler switch [2-29](#)
 -l (link library) compiler switch [2-29](#)
 -L (search library) compiler switch
[2-32](#)
 labs (absolute value, long) function
[3-92](#)
 Language extensions (compiler) (See
 Compiler C/C++ extensions)
 ldexp, ldexpf (exponential,
 multiply) function [3-95](#)
 ldiv (division, long) function [3-96](#)
 Leaf assembly routines [2-177](#)
 leaf_entry macro [2-154](#)
 leaf_exit macro [2-154](#)

Library source code
 working with [4-3](#), [5-3](#)
 Linker [2-177](#)
 Linking
 DSP library functions [4-3](#), [5-3](#)
 localeconv (localization pointer)
 function [3-99](#)
 Localization (See localeconv,
 setlocale, strxfrm functions)
 log, logf (log base e) function [3-103](#)
 log10, log10f (log base 10) function
[3-104](#)
 Long jump (See longjmp, setjmp
 functions)
 long storage format [2-143](#)
 longjmp (far jump return) function
[3-105](#)
 Lower case (See islower, tolower
 functions)

M

-M (make only) compiler switch
[2-30](#)
 M (Modify) registers [2-177](#)
 Macros
 ANSI standard run-time
 environment [3-7](#)
 EDOM [3-10](#)
 ERANGE [3-10](#)
 HUGE_VAL [3-10](#)
 Macros and asm() C program
 constructs [2-67](#)
 Macros and the C preprocessor
[2-178](#)

INDEX

- Macros, compound statements
 - [2-109](#)
- Macros, predefined [2-104](#)
- malloc (allocate uninitialized memory) function [3-107](#)
- Managing the stack [2-135](#)
- map (generate a memory map) compiler switch [2-30](#)
- Map file [2-30](#)
- matadd [4-57](#)
- Math functions
 - [acos](#) [3-32](#)
 - [asin](#), [asinf](#) [3-33](#)
 - [atan](#), [atanf](#) [3-34](#)
 - [atan2](#), [atan2f](#) [3-35](#)
 - [ceil](#), [ceilf](#) [3-45](#)
 - [cos](#), [cosf](#) [3-52](#)
 - [cosh](#), [coshf](#) [3-53](#)
 - [fabs](#), [fabsf](#) [3-57](#)
 - [floor](#), [floorf](#) [3-58](#)
 - [fmod](#), [fmodf](#) [3-59](#)
 - [frexp](#), [frexpf](#) [3-61](#)
 - [ldexp](#), [ldexpf](#) [3-95](#)
 - [log](#), [logf](#) [3-103](#)
 - [log10](#), [log10f](#) [3-104](#)
 - [modf](#), [modff](#) [3-115](#)
 - [pow](#), [powf](#) [3-116](#)
 - [rsqrt](#), [rsqrtf](#) [4-71](#)
 - [sinh](#), [sinhf](#) [3-128](#)
 - [sqrt](#), [sqrtf](#) [3-129](#)
 - [tan](#) [tanf](#) [3-157](#)
 - [tanh](#), [tanhf](#) [3-158](#)
- math.h header file [3-9](#)
- max (find larger, int) function
 - [3-108](#)
- mem (enable memory initialization) compiler switch
 - [2-30](#)
- memchr (find character) function
 - [3-109](#)
- memcmp (compare memory range) function [3-110](#)
- memcpy (copy memory range) function [3-111](#)
- memmove (move memory range) function [3-112](#)
- Memory (See [calloc](#), [free](#), [malloc](#), [memcmp](#), [memcpy](#), [memset](#), [memmove](#), [memchar](#), [realloc](#) functions)
- Memory Usage [2-123](#)
- memset (fill memory range) function [3-113](#)
- min (find smaller, int) function
 - [3-114](#)
- Mixed C/assembly
 - reference [2-150](#)
 - run-time model [2-121](#)
- Mixed C/assembly programming
 - [2-121](#)
 - arguments and return [2-140](#)
 - asm() constructs [2-58](#), [2-59](#), [2-62](#), [2-64](#), [2-65](#), [2-66](#), [2-67](#)
 - Call preserved registers [2-131](#)
 - compiler registers [2-129](#)
 - data storage and type sizes [2-143](#)
 - examples [2-165](#)

- return address [2-167](#)
- scratch registers [2-132](#)
- stack registers [2-133](#)
- stack usage [2-135](#)
- user registers [2-130](#)
- Mixed C/C++/assembly
 - programming [2-58](#), [2-121](#), [2-147](#), [2-148](#)
- MM (make and compile) compiler switch [2-30](#)
- MODE (1 & 2) registers [2-178](#)
- modf, modff (modulus, float) functions [3-115](#)
- Move memory range (See memmove function)
- MRB (Multiplier Results, Background) registers [2-178](#)
- MRF (Multiplier Results, Foreground) registers [2-178](#)
- Multi-line asm() C program constructs [2-64](#)

- N
- Naming conventions
 - assembly and C [2-158](#)
 - assembly and C++ [2-158](#)
 - C and assembly [2-157](#)
- Natural logarithm (See log, logf functions)
- new.h header file [3-28](#)
- no-aligned-stack (do not align stack) compiler switch [2-30](#)
- no-alttok (disable tokens) C++ mode compiler switch [2-30](#)
- no-builtin (no builtin functions) compiler switch [2-31](#)
- no-def (disable definitions) compiler switch [2-31](#)
- no-extra-keywords (not quite -ansi) compiler switch [2-31](#)
- no-inline (disable inline keyword) compiler switch [2-31](#)
- no-mem (disable memory initialization) compiler switch [2-31](#)
- Non-constant initializer support (compiler) [2-77](#)
- Non-leaf assembly routines [2-165](#), [2-178](#)
- no-restrict (no restrict support) C++ mode compiler switch [2-32](#)
- Normalized fraction (See frexp, frexpf function)
- no-std-def (disable standard definitions) compiler switch [2-32](#)
- no-std-inc (disable standard include search) compiler switch [2-32](#)
- no-std-lib (disable standard library search) compiler switch [2-32](#)
- Notation conventions [1-7](#)
- nothreads (disable thread-safe build) compiler switch [2-32](#)

INDEX

O

- O (enable optimization) compiler switch [2-33](#)
- o (output) compiler switch [2-33](#)
- Optimizing `asm()` C program constructs [2-65](#)

P

- P (omit #line) compiler switch [2-33](#)
- Parameter passing [2-140](#)
- path-install (installation location) compiler switch [2-34](#)
- path-output (non-temporary files location) compiler switch [2-34](#)
- path-temp (temporary files location) compiler switch [2-34](#)
- path-tool (tool location) compiler switch [2-34](#)
- pedantic (ANSI standard warnings) compiler switch [2-35](#)
- pedantic-errors (ANSI standard errors) compiler switch [2-35](#)
- Placement support keyword (section) [2-73](#)
- pm [2-55](#)
- pm (See Dual memory support keywords (pm dm))
- Pointer class support keyword (restrict) [2-55](#), [2-74](#)
- Polymorphic functions [3-29](#)
- pow, powf (power, x^y) functions [3-116](#)

- Power (See `exp`, `expf`, `pow`, `powf` functions)
- pplist (preprocessor listing) compiler switch [2-35](#)
- Predefined macros [2-104](#)
- Prelinker [2-43](#), [2-51](#)
- Preprocessing a program [2-102](#)
- Preprocessor
 - generated warnings [2-80](#)
- primitive I/O functions [3-13](#)
- Printable character test (See `isprint` function)
- proc (target processor) compiler switch [2-36](#)
- Program control functions
 - `calloc` [3-43](#)
 - `free` [3-60](#)
 - `malloc` [3-107](#)
 - `realloc` [3-121](#)
- ptr storage format [2-143](#)
- Punctuation character test (`ispunct`) function [3-88](#)
- `puts` macro [2-154](#)

Q

- qsort (quicksort) function [3-117](#)

R

- R (Register File) registers [2-178](#)
- R (search for source files) compiler switch [2-36](#)
- raise (force a signal) function [3-119](#)
- rand (Random number generator) function [3-120](#)

- Random number (See `rand`, `srand` functions) [3-120](#)
 - `reads` macro [2-154](#)
 - `realloc` (allocate used memory) function [3-121](#)
 - Real-time signals (See `clear_interrupt`, `interruptf`, `interrupts`, `poll_flag_in`, `raise`, `signal` functions)
 - Reciprocal square root function (See `rsqrt`, `rsqrtf` function)
 - Register usage (See Mixed C/assembly programming)
 - Registers for `asm()` constructs [2-62](#)
 - Reordering `asm()` C program constructs [2-65](#)
 - `-reserve` (reserve register) compiler switch [2-37](#)
 - `restore_reg` macro [2-155](#)
 - `restrict` (See Pointer class support keyword (`restrict`))
 - `-restrict` (support `restrict` keyword) C++ mode compiler switch [2-37](#)
 - Return Value Transfer [2-140](#)
 - Run-time environment (see Mixed C/assembly programming)
- S**
- `-S` (stop after compilation) compiler switch [2-37](#)
 - `-s` (strip debug information) compiler switch [2-37](#)
 - Saturated arithmetic [2-87](#)
 - `save_reg` macro [2-155](#)
 - `-save-temps` (save intermediate files) compiler switch [2-38](#)
 - Scratch registers [2-132](#)
 - Search character string (See `strchr`, `strchr` functions)
 - Search memory, character (See `memchar` function)
 - Search path for include files [2-28](#)
 - Search path for library files [2-29](#)
 - secondary registers [2-129](#), [2-134](#)
 - segment (See Placement Support Keyword (section))
 - segment keyword (See section keyword)
 - Send string to operating system (See `system` function)
 - Set jump (See `longjmp`, `setjmp` functions)
 - `setjmp` (long jump) function [3-123](#)
 - `setlocale` (set localization) function [3-124](#)
 - Setting compiler options [2-121](#)
 - short storage format [2-143](#)
 - `-show` (display command line) compiler switch [2-38](#)
 - signal (define signal handling) function [3-125](#)
 - Signals (See `clear_interrupt`, `interruptf`, `interrupts`, `poll_flag_in`, `raise`, `signal` functions)
 - signed keyword [2-19](#)

INDEX

- signed-char (make char signed)
 - compiler switch 2-38
- SIMD
 - C/C++ callable subroutines 2-161
 - Pitfalls 5-13
 - Support annotation (#pragma SIMD_for) 2-88
- sin, sinf (sine) functions 3-127
- Sine (See sin, sinf, sinh, sinh functions)
- sinh, sinh (sine hyperbolic) functions 3-128
- sqrt, sqrtf (square root) functions 3-129
- Square root (sqrt, sqrtf) functions 3-129
- srand (Random number seed) function 3-130
- Stack 2-133, 2-135
 - registers 2-133
- Standard argument functions
 - va_arg 3-161
 - va_end 3-163
 - va_start 3-164
- Standard C Library functions 3-7–3-19
- Standard header files
 - assert.h 3-8
 - ctype.h 3-8
 - errno.h 3-8
 - float.h 3-9
 - limits.h 3-9
 - locale.h 3-9
 - math.h 3-9
 - setjmp.h 3-10
 - stddef.h 3-13
- Standard library functions
 - abort 3-30
 - abs 3-31
 - acos, acosf 3-32
 - atexit 3-36
 - atof 3-37
 - atol 3-39
 - bsearch 3-41
 - div 3-54
 - exit 3-55
 - free 3-60
 - getenv 3-62
 - labs 3-92
 - ldiv 3-96
 - malloc 3-107
 - rand 3-120
 - srand 3-130
 - strtol 3-146, 3-150
 - strtoul 3-152
 - system 3-156
- stdio header file 3-13
- Stop (See atexit, exit functions)
- strcat (concatenate string) function 3-131
- strchr (search character string) function 3-132
- strcmp (compare strings) function 3-133
- strcoll (compare strings, localized) function 3-134
- strcpy (copy string) function 3-135

- strcspn (compare string span)
 - function [3-136](#)
- strerror (get error message string)
 - function [3-137](#)
- String compare (See strcmp, strcoll, strcspn, strncmp, strpbrk, strstr functions)
- String concatenate (See strcat, strcat functions)
- String conversion (See atof, atoi, atol, strtok, strtol, strxfrm functions)
- String copy (See strcpy, strncpy function)
- String functions
 - memchar [3-109](#)
 - memcmp [3-110](#)
 - memcpy [3-111](#)
 - memmove [3-112](#)
 - memset [3-113](#)
 - strcat [3-131](#)
 - strchr [3-132](#)
 - strcmp [3-133](#)
 - strcoll [3-134](#)
 - strcpy [3-135](#)
 - strcspn [3-136](#)
 - strerror [3-137](#)
 - strlen [3-138](#)
 - strncat [3-139](#)
 - strncmp [3-140](#)
 - strncpy [3-141](#)
 - strpbrk [3-142](#)
 - strchr [3-143](#), [3-144](#)
 - strspn [3-144](#)
 - strstr [3-145](#)
 - strtok [3-148](#)
 - strxfrm [3-154](#)
- String length (See strlen function)
- strlen (string length) function [3-138](#)
- strncat (concatenate characters from string) function [3-139](#)
- strncmp (compare characters in strings) function [3-140](#)
- strncpy (copy characters in string) function [3-141](#)
- strpbrk (compare strings, pointer break) function [3-142](#)
- strrchr (search character string, recursive) function [3-143](#)
- strspn (string span) function [3-144](#)
- strstr (compare string, string) function [3-145](#)
- strtod (convert string to a double) function [3-146](#)
- strtok (convert token to string) function [3-148](#)
- strtol (convert string to long) function [3-150](#)
- strtoul (convert string to unsigned long) function [3-152](#)
- strxfrm (localization transform) function [3-154](#)
- Support for multiple heaps [2-111](#)
- syntax-only (just check syntax)
 - compiler switch [2-38](#)
- sysreg_read [2-82](#)
- sysreg_write [2-82](#)

INDEX

system (send string to operating system) function [3-156](#)

System header file [2-153](#)

System header files [2-109](#)

T

-T (linker description file) compiler switch [2-38](#)

tan, tanf (tangent) function [3-157](#)

Tangent (See atan, atanf, atan2, atan2f, cot, cotf, tan, tanf, tanh, tanhf functions)

tanh, tanhf (Hyperbolic tangent) functions [3-158](#)

Technical support [1-3](#)

Template for asm() in C programs [2-59](#)

Template Library header files

<algorithm> [3-26](#)

<deque> [3-26](#)

<functional> [3-26](#)

<hash_map> [3-26](#)

<hash_set> [3-26](#)

<iterator> [3-27](#)

<list> [3-27](#)

<map> [3-27](#)

<memory> [3-27](#)

<numeric> [3-27](#)

<queue> [3-27](#)

<set> [3-27](#)

<stack> [3-27](#)

<utility> [3-27](#)

<vector> [3-28](#)

Temporary files [2-38](#)

Terminate (See atexit, exit functions)

-threads (enable thread-safe build) compiler switch [2-39](#)

-time (tell time) switch [2-39](#)

Tokens, string convert (See strtok function)

tolower (convert characters to lower case) function [3-159](#)

toupper (convert characters to upper case) function [3-160](#)

-traditional (traditional compilation) compiler switch [2-19](#)

Transferring function arguments and return value [2-140](#)

Trigonometric (See Math functions)

true (See Boolean type support keywords (bool, true, false))

Type sizes, data [2-143](#)

U

-U (undefine macro) compiler switch [2-24](#), [2-39](#)

-unsigned-char (make char unsigned) compiler switch [2-39](#)

Upper case (See isupper, toupper functions)

User header files [2-109](#)

User registers [2-130](#)

Using

mixed C/assembly naming conventions [2-157](#)

- mixed C/assembly support macros
 - 2-153
 - the run-time header 2-147
 - the Visual DSP compiler (Guide) 2-1
 - USTAT (1 and 2) registers 2-178
 - Utility functions
 - getenv 3-62
 - system 3-156
- V**
- v (version & verbose) compiler switch
 - 2-39
 - va_arg (get next argument in list)
 - function 3-161
 - va_end (finish processing argument list)
 - function 3-163
 - va_start (initialize argument list)
 - function 3-164
 - Variable length array support 2-75
 - verbose (display command line)
 - compiler switch 2-40
 - version (display version) compiler
 - switch 2-27, 2-40
 - Visual DSP++ Compiler (cc21k) Guide
 - 2-1
 - VisualDSP++ compiler (cc21k) C/C++
 - language extensions 2-54
 - volatile and asm() C program constructs
 - 2-65
 - volatile keyword 2-19
- W**
- w (disable all warnings) switch 2-41
 - W (override error) compiler switch
 - 2-40
 - Warning messages 2-80
 - Warn-protos (warn if incomplete
 - prototype) compiler switch 2-40
 - Wdriver-limit (maximum process
 - errors) compiler switch 2-40
 - Werror-limit (maximum compiler
 - errors) compiler switch 2-41
 - White space character test (See isspace
 - function)
 - Window generators 4-9, 5-10
 - Wremarks 2-41
 - Wremarks (enable diagnostic
 - warnings) compiler switch 2-41
 - write-files (enable driver I/O pipe)
 - compiler switch 2-41
 - Writing macros 2-109
 - Wterse (enable terse warnings)
 - compiler switch 2-41