

I INDEX

Symbols

!= (not equal) preprocessor operator [2-37](#)
#command (see Preprocessor commands)
% (length of) legacy syntax [3-31](#)
% (modulus) preprocessor operator [2-37](#)
& (bitwise AND) preprocessor operator [2-38](#)
&& (logical AND) preprocessor operator [2-38](#)
() assembler operator [2-37](#)
-(subtraction) assembler operator [2-37](#)
-(unary minus) assembler operator [2-37](#)
* (multiply) assembler operator [2-37](#)
+ (addition) assembler operator [2-37](#)
.(DIRECTIVE) (see Assembler directives) [2-44](#)
/ (divide) preprocessor operator [2-37](#)
<= (less than or equal) preprocessor operator [2-37](#)
== (equal) preprocessor operator [2-37](#)
> (greater than) preprocessor operator [2-37](#)
>= (greater than or equal) preprocessor operator [2-37](#)

>> (shift right) assembler operator [2-37](#)
? (generate unique labels) preprocessor operator [4-33](#)
^ (address of) legacy syntax [3-31](#)
^ (bitwise exclusive OR) preprocessor operator [2-38](#)
| (bitwise exclusive OR) assembler operator [2-38](#)
|| (logical OR) preprocessor operator [2-38](#)
~ (one's complement) assembler operator [2-37](#)

Numerics

-2191 assembler switch [2-21](#)
-2192-12 assembler switch [2-21](#)
-219x assembler switch [2-21](#)

A

ABS

legacy module qualifier [3-23](#), [3-24](#)
Absolute placement [2-22](#), [3-26](#)
absolute placement of circular buffers [2-22](#), [3-26](#)
Address expression [2-36](#)
ALIGN (address alignment) assembler directive [2-48](#)

INDEX

- Analog products [1-2](#)
- Archiver [2-15](#)
- Arithmetic
 - fractional [2-41](#)
 - mixed fractional [2-42](#)
- ASCII string initialization support [2-67](#)
- Assembler
 - command-line switch
 - 2191 (ADSP-2191 code) [2-21](#)
 - 2192-12 (ADSP-2192-12 code) [2-21](#)
 - 219x (ADSP-219x code) [2-21](#)
 - Ao (absolute placement) [2-22](#), [3-26](#)
 - c (preserve case) [2-22](#)
 - d (define macro) [2-23](#)
 - flags-pp -opt (pass -opt to the preprocessor) [2-24](#)
 - g (generate debug info) [2-24](#)
 - h (help) [2-24](#)
 - i (include path) [2-24](#)
 - ip (individual placement) [2-25](#)
 - JumpCallShort2Long (short to long) [2-26](#)
 - l (list source file) [2-26](#)
 - legacy (accept legacy code) [2-27](#)
 - li (list source and #include files) [2-26](#)
 - M (make rule only) [2-27](#)
 - MM (make rule and assemble) [2-29](#)
 - Mo (output make rule) [2-28](#)
 - Mt (make rule for named file) [2-28](#)
 - noip (disable individual placement) [2-29](#)
 - o (output) [2-29](#)
 - pp (proceed with preprocessing) [2-29](#)
 - r (remove preprocessor info) [2-30](#)
 - sp (skip preprocessing) [2-30](#)
 - v (verbose) [2-30](#)
 - version (display version) [2-30](#)
 - w (skip warning messages) [2-30](#)
 - directives [2-6](#), [2-44–2-67](#), [2-69](#)
 - glossary section [2-69](#)
 - guide section [2-3–2-14](#)
 - how to use assembler [2-1](#)
 - instruction set [2-6](#)
 - keywords [2-31–2-33](#)
 - legacy directives [3-6–3-29](#)
 - listing files (.LST) [2-5](#), [2-26](#)
 - numeric formats [2-40](#)
 - object files (.DOJ) [2-5](#)
 - overview section [2-1](#)
 - reference section [2-15–2-30](#)
 - source files
 - (.ASM) [2-4](#), [2-17](#)
 - (.DSP) [2-17](#)

Assembler interface (see Command-line syntax)

Assembler legacy directives [3-6](#)

Assembly directive
 - .ALIGN [2-48](#)
 - .EXTERN [2-49](#)

- .FILE [2-50](#)
- .GLOBAL [2-51](#)
- .LEFTMARGIN [2-52](#)
- .NEWPAGE [2-53](#)
- .PAGELength [2-54](#)
- .PAGEWIDTH [2-55](#)
- .PREVIOUS [2-56](#)
- .REPEAT/END_REPEAT [2-57](#)
- .SECTION [2-59](#)
- .TYPE [2-61](#)
- .VAR [2-62](#)
- .VAR/CIRC [3-27](#)
- .WEAK [2-68](#)
- Assembly symbols [2-34](#)
- C**
- c (case sensitive) assembler switch [2-22](#)
- C and assembly, interfacing [2-12](#)
- Case-sensitivity [2-22](#)
- Circular buffer [3-27](#)
- Circular buffers with absolute placement [3-26](#)
- Command-line syntax of assembler [2-16–2-30](#)
- Command-line syntax of preprocessor [4-17](#)
- Comment conventions, new and legacy [3-33](#)
- Commentconverter
 - command-line switches
 - csall (detects all comments) [A-2](#)
 - help (displays help screen) [A-2](#)
 - no (disables recognition of cs/rs styles) [A-3](#)
 - noinfo (disables info messages) [A-2](#)
 - nowarn (disables warnings) [A-2](#)
 - o filename (outputs named file) [A-2](#)
 - rs! (replaces "!" comments) [A-2](#)
 - rs{ (replaces "{" comments) [A-2](#)
 - rsall (replaces legacy comments) [A-2](#)
 - v (verbose) [A-3](#)
 - version (displays version) [A-3](#)
- command-line syntax [A-2](#)
- default options [A-3](#)
- Concatenate (##) preprocessor operator [4-32](#)
- Consecutive placement [2-64](#)
- CONST (constant declaration)
 - legacy directive [3-8](#)
- Constant expression [2-36](#)
- Content of an assembly program [2-6](#)
- Conventions [1-7](#)
 - file extensions [2-17](#)
 - file naming [2-17](#)
 - numeric formats [2-40](#)
- cpredef (C-style definitions)
 - preprocessor switch [4-11](#)
- cs! ("!" comment style)
 - preprocessor switch [4-11](#)
- cs/* ("/* */" comment style)
 - preprocessor switch [4-12](#)

INDEX

- cs// ("//" comment style)
 - preprocessor switch [4-12](#)
- cs{ ("{" comment style)
 - preprocessor switch [4-12](#)
- csall (all comment styles)
 - preprocessor switch [4-13](#)
- Customer support [1-2](#)

D

- D|d (define macro)
 - assembler switch [2-23](#)
 - preprocessor switch [4-13](#)
- DAT, data initialization files [2-15](#), [2-62](#), [2-63](#)
- Data Initialization Files (.DAT)
 - [2-63](#)
- Debugging capabilities [3-34](#)
- define (macro or constant)
 - preprocessor command [4-19](#)
- defined preprocessor macros [4-6](#)
- DM or DATA, section type [2-59](#), [3-23](#)
- DMSEG (locate data) legacy
 - directive [3-9](#)

E

- else (alternate instruction)
 - preprocessor command [4-22](#)
- ENTRY (entry point) legacy
 - directive [3-11](#)
- error (error message) preprocessor
 - command [4-24](#)
- Expressions
 - address [2-36](#)

- constant [2-36](#)
- EXTERN (refer to a global)
 - assembler directive [2-49](#)
- EXTERNAL (refer to a global)
 - legacy directive [3-12](#)

F

- FILE (override filename) assembler
 - directive [2-50](#)
- File extension [2-18](#)
 - .APP (preprocessed file) [2-15](#)
 - .ASM (assembly source files) [2-15](#)
 - .DAT (data initialization file)
 - [2-17](#), [2-62](#), [2-63](#)
 - .DLB (library file) [2-15](#)
 - .DOJ (object file) [2-5](#), [2-15](#)
 - .DSP (assembly source file) [2-15](#)
 - .DXE (executable file) [2-15](#)
 - .H (header file) [2-15](#)
 - .IS (preprocessed file) [2-15](#), [2-30](#), [4-1](#)
 - .LST (listing file) [2-5](#), [2-15](#)
- File format
 - DWARF-2 (Debugging Information Format) [2-15](#)
 - ELF (Executable and Linkable Format) [2-15](#)
- File naming conventions [2-17](#)
- flags-pp -switch assembler switch [2-24](#)
- Formats, numeric [2-40](#)
- Fract
 - numeric format [2-1](#), [2-40](#)
 - representation [2-41](#)

G

- g (generate debug info) [2-24](#)
- GLOBAL (global symbol)
 - assembler directive [2-51](#)
- Glossary section [2-69](#)

H

- h (help) assembler switch [2-24](#), [4-13](#)
- Header files [4-3](#)
 - system [4-3](#)
 - user [4-3](#)

I

- i (include path)
 - assembler switch [2-24](#)
- I*i* (include path)
 - preprocessor switch [4-13](#)
- if (test if true) preprocessor
 - command [4-25](#)
- ifdef (test if defined) preprocessor
 - command [4-26](#)
- ifndef (test if not defined)
 - preprocessor command [4-27](#)
- INCLUDE (include file) legacy
 - directive [3-13](#)
- include (include file) preprocessor
 - command [4-28](#)
- Included files
 - system header files [4-3](#)
 - user header files [4-3](#)
- INDENT (indent 1st file) legacy
 - directive [3-15](#)

- INIT & INIT24 (initialize symbols)
 - legacy directive [3-16](#)
- Instruction set [2-6](#), [2-69](#)

J

- JumpCallShort2Long (short to long) assembler switch [2-26](#)

K

- Keywords, list of [2-34](#)

L

- l (listing file) assembler switch [2-26](#)
- legacy (accept legacy code)
 - assembler switch [2-27](#)
- Legacy directives
 - .CONST [3-8](#)
 - .DMSEG [3-9](#)
 - .ENTRY [3-11](#)
 - .EXTERNAL [3-12](#)
 - .INCLUDE [3-13](#)
 - .INDENT [3-15](#)
 - .INIT & .INIT24 [3-16](#)
 - .LOCAL [3-19](#)
 - .MACRO & .ENDMACRO [3-21](#)
 - .MODULE & .ENDMODULE [3-23](#)
 - .PMSEG [3-9](#)
- Legacy module qualifiers
 - /ABS [3-26](#)
 - /SEG [3-24](#)
- line (output line number)
 - preprocessor command [4-29](#)

INDEX

Linker [2-15](#)

Linker Description File [2-8](#), [2-69](#)

Linker's RESOLVE() command
[2-22](#), [3-26](#)

Listing file [2-3](#), [2-13](#)

address [2-13](#)

line number [2-14](#)

opcode [2-13](#)

source code line [2-14](#)

M

-M (make rule only)

assembler switch [2-27](#)

preprocessor switch [4-14](#)

Macros

compound statements [4-6](#)

overview [4-4](#)

predefined [4-6](#)

Manual

audience [1-3](#)

contents description [1-4](#)

conventions [1-7](#)

new in this edition [1-5](#)

related documents [1-6](#)

Memory type [2-7](#)

DM or DATA (data) [2-59](#), [3-23](#)

PM or CODE (code and data)
[2-59](#), [3-23](#)

RAM (random access memory)
[3-23](#)

ROM (read only memory) [3-23](#)

-MM (make rule and assemble)

assembler switch [2-28](#)

preprocessor switch [4-14](#)

-Mo (output make rule)

assembler switch [2-28](#)

preprocessor switch [4-15](#)

Modified assembler operators [3-31](#)

Modified numeric conventions [3-32](#)

Module qualifier

ABS [3-23](#)

SEG [3-23](#)

MODULE/ ENDMODULE

(mark code module) legacy

directive [3-23](#)

-Mt (output make rule for named
file)

assembler switch [2-29](#)

preprocessor switch [4-15](#)

N

New assembler switches [3-3](#)

NEWPAGE (.lst page break)

assembler directive [2-69](#)

Notation conventions of this

manual [1-7](#)

Numeric formats [2-40](#)

O

-o (output)

assembler switch [2-29](#)

preprocessor switch [4-16](#)

Object file [2-3](#)

Operator

- (subtraction) [2-37](#)

- (unary minus) [2-37](#)

!= (not equal) [2-37](#)

% (modulus) [2-37](#)

- & (bitwise AND) [2-38](#)
- && (logical AND) [2-38](#)
- + (addition) [2-37](#)
- / (divide) [2-37](#)
- <= (less than or equal) [2-37](#)
- == (equal) [2-37](#)
- > (greater than) [2-37](#)
- >= (greater than or equal) [2-37](#)
- >> (shift right) [2-37](#)
- \ (divide) operator of 6.1 Release [2-37](#)
- ^(bitwise exclusive OR) [2-38](#)
- | (bitwise inclusive OR) [2-38](#)
- || (logical OR) [2-38](#)
- ~(one's complement) [2-37](#)
- Operator precedence chart [2-37](#)

P

- PAGE (page of) operator legacy syntax [3-31](#)
- PAGELength (.lst page size) assembler directive [2-54](#)
- PAGEWIDTH (.lst page width) assembler directive [2-55](#)
- PM or CODE, section type [2-59](#), [3-23](#)
- PMSEG (locate code & data) legacy directive [3-9](#)
- pp (preprocessing only) assembler switch [2-29](#)
- Preprocessor
 - command syntax [4-17](#)
 - command, list of [4-17](#)
 - command-line switch

- cs! ("!" comment style) [4-11](#)
- cs/* ("/* */" comment style) [4-12](#)
- cs// ("//" comment style) [4-12](#)
- cs{ ("{" comment style) [4-12](#)
- csall (all comment styles) [4-13](#)
- D (define macro) [4-13](#)
- h (help) [4-13](#)
- I|i (include path) [4-13](#)
- M (make rule only) [4-14](#)
- MM (make rule and assemble) [4-14](#)
- Mo (output make rule) [4-15](#)
- Mt (output make rule for named file) [4-15](#)
- o (output) [4-16](#)
- v (verbose) [4-16](#)
- version (display version) [4-16](#)
- commands [2-7](#)
- guide section [4-2–4-16](#)
- how to use [4-2](#)
- option setting [4-7](#)
- overview [4-1](#)
- preprocessed assembly files [4-1](#)
- preprocessing of a program [2-13](#)
- reference section [4-8–4-32](#)
- Preprocessor commands
 - #define [4-19](#)
 - #elif [4-21](#)
 - #else [4-22](#)
 - #endif [4-23](#)
 - #error [4-24](#)
 - #if [4-25](#)
 - #ifdef [4-26](#)

INDEX

- `#ifndef` [4-27](#)
- `#include` [4-28](#)
- `#line` [4-29](#)
- `#undef` [4-30](#)
- `#warning` [4-30](#)
- Preprocessor directives (see Preprocessor commands)
- Preprocessor operators
 - `#` (to string) [4-31](#)
 - `##` (concatenate) [4-32](#)
 - `?` (generate unique labels) [3-19](#), [4-33](#)
- Preprocessor switch (see Preprocessor command-line switch)
- PREVIOUS (section embedding)
 - assembler directive [2-62](#)
- Program [2-6](#)
 - assembling [2-4](#)
 - content [2-6–2-7](#)
 - interfacing requirements [2-12](#)
 - preprocessing [2-13](#)
 - reading a listing file [2-13](#)
 - structure [2-7–2-10](#)
 - writing an assembly program [2-3–2-5](#)
- Project Options
 - setting assembler options [2-14](#)
 - setting preprocessor options [4-7](#)
- R**
- `-r` (remove preprocessor info)
 - assembler switch [2-30](#)
- Related documents [1-6](#)
- REPEAT/END_REPEAT
 - assembler directive [2-57](#)
- RESOLVE() linker command [2-22](#), [3-26](#)
- S**
- SECTION (program or data section) assembler directive [2-59](#)
- Section qualifier
 - DM (DATA) [2-59](#), [3-23](#)
 - PM (CODE) [2-59](#), [3-23](#)
- SEG, legacy module qualifier [3-24](#)
- Setting assembler options [2-14](#)
- Setting preprocessor options
 - from command line [4-7](#)
 - from VisualDSP++ [4-7](#)
 - through build tools [4-7](#)
- `-sp` (skip preprocessing) assembler switch [2-30](#)
- `-stringize` preprocessor switch [4-16](#)
- Structure of a program [2-7](#)
- Switches (see Assembler command-line switch)
- Symbols (see Assembly symbols)
- Syntax
 - assembler command line [2-16](#)
 - assembler directives [2-44](#)
 - assembler expressions [2-36](#)
 - compound statement [4-6](#)
 - instruction set [2-6](#)
 - macro [4-4](#)
 - preprocessor command [4-17](#)
 - preprocessor command line [4-10](#)

System header files [4-3](#)

T

Technical support [1-2](#)

TYPE (change default type)
 assembler directive [2-61](#), [2-69](#)

U

undef (undefine) preprocessor
 command [4-30](#)

User header files [4-3](#)

Utility

 comment conversion [4-8](#), [A-1](#)
 elfdump [2-59](#)

V

-v (verbose)
 assembler switch [2-30](#)
 preprocessor switch [4-16](#)

VAR & VAR/INIT24 (declare
 variable) assembler directive
 [2-62](#)

-version (display version)
 assembler switch [2-30](#)
 preprocessor switch [4-16](#)

VisualDSP++

 assembling from [2-1](#)
 Project Options dialog box [2-14](#),
 [2-31](#)
 setting preprocessor options [4-7](#)

W

-w (skip warning messages)
 assembler switch [2-30](#)
 warning (warning message)
 preprocessor command [4-30](#)

WEAK (weak definition &
 reference) assembler directive
 [2-68](#)

Writing assembly programs [2-3](#)

INDEX