# [Using as](mailto:https:/sourceware.org/binutils/docs/as/index.html?subject=as)

## Table of Contents

* [1 Overview](https://sourceware.org/binutils/docs/as/Overview.html" \l "Overview)
  + [1.1 Structure of this Manual](https://sourceware.org/binutils/docs/as/Manual.html" \l "Manual)
  + [1.2 The GNU Assembler](https://sourceware.org/binutils/docs/as/GNU-Assembler.html" \l "GNU-Assembler)
  + [1.3 Object File Formats](https://sourceware.org/binutils/docs/as/Object-Formats.html" \l "Object-Formats)
  + [1.4 Command Line](https://sourceware.org/binutils/docs/as/Command-Line.html" \l "Command-Line)
  + [1.5 Input Files](https://sourceware.org/binutils/docs/as/Input-Files.html" \l "Input-Files)
  + [1.6 Output (Object) File](https://sourceware.org/binutils/docs/as/Object.html" \l "Object)
  + [1.7 Error and Warning Messages](https://sourceware.org/binutils/docs/as/Errors.html" \l "Errors)
* [2 Command-Line Options](https://sourceware.org/binutils/docs/as/Invoking.html" \l "Invoking)
  + [2.1 Enable Listings: -a[cdghlns]](https://sourceware.org/binutils/docs/as/a.html" \l "a)
  + [2.2 --alternate](https://sourceware.org/binutils/docs/as/alternate.html" \l "alternate)
  + [2.3 -D](https://sourceware.org/binutils/docs/as/D.html" \l "D)
  + [2.4 Work Faster: -f](https://sourceware.org/binutils/docs/as/f.html" \l "f)
  + [2.5 .include Search Path: -I path](https://sourceware.org/binutils/docs/as/I.html" \l "I)
  + [2.6 Difference Tables: -K](https://sourceware.org/binutils/docs/as/K.html" \l "K)
  + [2.7 Include Local Symbols: -L](https://sourceware.org/binutils/docs/as/L.html" \l "L)
  + [2.8 Configuring listing output: --listing](https://sourceware.org/binutils/docs/as/listing.html" \l "listing)
  + [2.9 Assemble in MRI Compatibility Mode: -M](https://sourceware.org/binutils/docs/as/M.html" \l "M)
  + [2.10 Dependency Tracking: --MD](https://sourceware.org/binutils/docs/as/MD.html" \l "MD)
  + [2.11 Output Section Padding](https://sourceware.org/binutils/docs/as/no_002dpad_002dsections.html" \l "no_002dpad_002dsections)
  + [2.12 Name the Object File: -o](https://sourceware.org/binutils/docs/as/o.html" \l "o)
  + [2.13 Join Data and Text Sections: -R](https://sourceware.org/binutils/docs/as/R.html" \l "R)
  + [2.14 Display Assembly Statistics: --statistics](https://sourceware.org/binutils/docs/as/statistics.html" \l "statistics)
  + [2.15 Compatible Output: --traditional-format](https://sourceware.org/binutils/docs/as/traditional_002dformat.html" \l "traditional_002dformat)
  + [2.16 Announce Version: -v](https://sourceware.org/binutils/docs/as/v.html" \l "v)
  + [2.17 Control Warnings: -W, --warn, --no-warn, --fatal-warnings](https://sourceware.org/binutils/docs/as/W.html" \l "W)
  + [2.18 Generate Object File in Spite of Errors: -Z](https://sourceware.org/binutils/docs/as/Z.html" \l "Z)
* [3 Syntax](https://sourceware.org/binutils/docs/as/Syntax.html" \l "Syntax)
  + [3.1 Preprocessing](https://sourceware.org/binutils/docs/as/Preprocessing.html" \l "Preprocessing)
  + [3.2 Whitespace](https://sourceware.org/binutils/docs/as/Whitespace.html" \l "Whitespace)
  + [3.3 Comments](https://sourceware.org/binutils/docs/as/Comments.html" \l "Comments)
  + [3.4 Symbols](https://sourceware.org/binutils/docs/as/Symbol-Intro.html" \l "Symbol-Intro)
  + [3.5 Statements](https://sourceware.org/binutils/docs/as/Statements.html" \l "Statements)
  + [3.6 Constants](https://sourceware.org/binutils/docs/as/Constants.html" \l "Constants)
    - [3.6.1 Character Constants](https://sourceware.org/binutils/docs/as/Characters.html" \l "Characters)
      * [3.6.1.1 Strings](https://sourceware.org/binutils/docs/as/Strings.html" \l "Strings)
      * [3.6.1.2 Characters](https://sourceware.org/binutils/docs/as/Chars.html" \l "Chars)
    - [3.6.2 Number Constants](https://sourceware.org/binutils/docs/as/Numbers.html" \l "Numbers)
      * [3.6.2.1 Integers](https://sourceware.org/binutils/docs/as/Integers.html" \l "Integers)
      * [3.6.2.2 Bignums](https://sourceware.org/binutils/docs/as/Bignums.html" \l "Bignums)
      * [3.6.2.3 Flonums](https://sourceware.org/binutils/docs/as/Flonums.html" \l "Flonums)
* [4 Sections and Relocation](https://sourceware.org/binutils/docs/as/Sections.html" \l "Sections)
  + [4.1 Background](https://sourceware.org/binutils/docs/as/Secs-Background.html" \l "Secs-Background)
  + [4.2 Linker Sections](https://sourceware.org/binutils/docs/as/Ld-Sections.html" \l "Ld-Sections)
  + [4.3 Assembler Internal Sections](https://sourceware.org/binutils/docs/as/As-Sections.html" \l "As-Sections)
  + [4.4 Sub-Sections](https://sourceware.org/binutils/docs/as/Sub_002dSections.html" \l "Sub_002dSections)
  + [4.5 bss Section](https://sourceware.org/binutils/docs/as/bss.html" \l "bss)
* [5 Symbols](https://sourceware.org/binutils/docs/as/Symbols.html" \l "Symbols)
  + [5.1 Labels](https://sourceware.org/binutils/docs/as/Labels.html" \l "Labels)
  + [5.2 Giving Symbols Other Values](https://sourceware.org/binutils/docs/as/Setting-Symbols.html" \l "Setting-Symbols)
  + [5.3 Symbol Names](https://sourceware.org/binutils/docs/as/Symbol-Names.html" \l "Symbol-Names)
  + [5.4 The Special Dot Symbol](https://sourceware.org/binutils/docs/as/Dot.html" \l "Dot)
  + [5.5 Symbol Attributes](https://sourceware.org/binutils/docs/as/Symbol-Attributes.html" \l "Symbol-Attributes)
    - [5.5.1 Value](https://sourceware.org/binutils/docs/as/Symbol-Value.html" \l "Symbol-Value)
    - [5.5.2 Type](https://sourceware.org/binutils/docs/as/Symbol-Type.html" \l "Symbol-Type)
    - [5.5.3 Symbol Attributes: a.out](https://sourceware.org/binutils/docs/as/a_002eout-Symbols.html" \l "a_002eout-Symbols)
      * [5.5.3.1 Descriptor](https://sourceware.org/binutils/docs/as/Symbol-Desc.html" \l "Symbol-Desc)
      * [5.5.3.2 Other](https://sourceware.org/binutils/docs/as/Symbol-Other.html" \l "Symbol-Other)
    - [5.5.4 Symbol Attributes for COFF](https://sourceware.org/binutils/docs/as/COFF-Symbols.html" \l "COFF-Symbols)
      * [5.5.4.1 Primary Attributes](https://sourceware.org/binutils/docs/as/COFF-Symbols.html" \l "Primary-Attributes)
      * [5.5.4.2 Auxiliary Attributes](https://sourceware.org/binutils/docs/as/COFF-Symbols.html" \l "Auxiliary-Attributes)
    - [5.5.5 Symbol Attributes for SOM](https://sourceware.org/binutils/docs/as/SOM-Symbols.html" \l "SOM-Symbols)
* [6 Expressions](https://sourceware.org/binutils/docs/as/Expressions.html" \l "Expressions)
  + [6.1 Empty Expressions](https://sourceware.org/binutils/docs/as/Empty-Exprs.html" \l "Empty-Exprs)
  + [6.2 Integer Expressions](https://sourceware.org/binutils/docs/as/Integer-Exprs.html" \l "Integer-Exprs)
    - [6.2.1 Arguments](https://sourceware.org/binutils/docs/as/Arguments.html" \l "Arguments)
    - [6.2.2 Operators](https://sourceware.org/binutils/docs/as/Operators.html" \l "Operators)
    - [6.2.3 Prefix Operator](https://sourceware.org/binutils/docs/as/Prefix-Ops.html" \l "Prefix-Ops)
    - [6.2.4 Infix Operators](https://sourceware.org/binutils/docs/as/Infix-Ops.html" \l "Infix-Ops)
* [7 Assembler Directives](https://sourceware.org/binutils/docs/as/Pseudo-Ops.html" \l "Pseudo-Ops)
  + [7.1 .abort](https://sourceware.org/binutils/docs/as/Abort.html" \l "Abort)
  + [7.2 .ABORT (COFF)](https://sourceware.org/binutils/docs/as/ABORT-_0028COFF_0029.html" \l "ABORT-_0028COFF_0029)
  + [7.3 .align abs-expr, abs-expr, abs-expr](https://sourceware.org/binutils/docs/as/Align.html" \l "Align)
  + [7.4 .altmacro](https://sourceware.org/binutils/docs/as/Altmacro.html" \l "Altmacro)
  + [7.5 .ascii "string"…](https://sourceware.org/binutils/docs/as/Ascii.html" \l "Ascii)
  + [7.6 .asciz "string"…](https://sourceware.org/binutils/docs/as/Asciz.html" \l "Asciz)
  + [7.7 .balign[wl] abs-expr, abs-expr, abs-expr](https://sourceware.org/binutils/docs/as/Balign.html" \l "Balign)
  + [7.8 Bundle directives](https://sourceware.org/binutils/docs/as/Bundle-directives.html" \l "Bundle-directives)
    - [7.8.1 .bundle\_align\_mode abs-expr](https://sourceware.org/binutils/docs/as/Bundle-directives.html" \l "g_t_002ebundle_005falign_005fmode-abs_002dexpr)
    - [7.8.2 .bundle\_lock and .bundle\_unlock](https://sourceware.org/binutils/docs/as/Bundle-directives.html" \l "g_t_002ebundle_005flock-and-_002ebundle_005funlock)
  + [7.9 .byte expressions](https://sourceware.org/binutils/docs/as/Byte.html" \l "Byte)
  + [7.10 CFI directives](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "CFI-directives)
    - [7.10.1 .cfi\_sections section\_list](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fsections-section_005flist)
    - [7.10.2 .cfi\_startproc [simple]](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fstartproc-_005bsimple_005d)
    - [7.10.3 .cfi\_endproc](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fendproc)
    - [7.10.4 .cfi\_personality encoding [, exp]](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fpersonality-encoding-_005b_002c-exp_005d)
    - [7.10.5 .cfi\_personality\_id id](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fpersonality_005fid-id)
    - [7.10.6 .cfi\_fde\_data [opcode1 [, …]]](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005ffde_005fdata-_005bopcode1-_005b_002c-_2026_005d_005d)
    - [7.10.7 .cfi\_lsda encoding [, exp]](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005flsda-encoding-_005b_002c-exp_005d)
    - [7.10.8 .cfi\_inline\_lsda [align]](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005finline_005flsda-_005balign_005d)
    - [7.10.9 .cfi\_def\_cfa register, offset](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fdef_005fcfa-register_002c-offset)
    - [7.10.10 .cfi\_def\_cfa\_register register](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fdef_005fcfa_005fregister-register)
    - [7.10.11 .cfi\_def\_cfa\_offset offset](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fdef_005fcfa_005foffset-offset)
    - [7.10.12 .cfi\_adjust\_cfa\_offset offset](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fadjust_005fcfa_005foffset-offset)
    - [7.10.13 .cfi\_offset register, offset](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005foffset-register_002c-offset)
    - [7.10.14 .cfi\_val\_offset register, offset](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fval_005foffset-register_002c-offset)
    - [7.10.15 .cfi\_rel\_offset register, offset](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005frel_005foffset-register_002c-offset)
    - [7.10.16 .cfi\_register register1, register2](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fregister-register1_002c-register2)
    - [7.10.17 .cfi\_restore register](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005frestore-register)
    - [7.10.18 .cfi\_undefined register](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fundefined-register)
    - [7.10.19 .cfi\_same\_value register](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fsame_005fvalue-register)
    - [7.10.20 .cfi\_remember\_state and .cfi\_restore\_state](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fremember_005fstate-and-_002ecfi_005frestore_005fstate)
    - [7.10.21 .cfi\_return\_column register](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005freturn_005fcolumn-register)
    - [7.10.22 .cfi\_signal\_frame](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fsignal_005fframe)
    - [7.10.23 .cfi\_window\_save](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fwindow_005fsave)
    - [7.10.24 .cfi\_escape expression[, …]](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fescape-expression_005b_002c-_2026_005d)
    - [7.10.25 .cfi\_val\_encoded\_addr register, encoding, label](https://sourceware.org/binutils/docs/as/CFI-directives.html" \l "g_t_002ecfi_005fval_005fencoded_005faddr-register_002c-encoding_002c-label)
  + [7.11 .comm symbol , length](https://sourceware.org/binutils/docs/as/Comm.html" \l "Comm)
  + [7.12 .data subsection](https://sourceware.org/binutils/docs/as/Data.html" \l "Data)
  + [7.13 .dc[size] expressions](https://sourceware.org/binutils/docs/as/Dc.html" \l "Dc)
  + [7.14 .dcb[size] number [,fill]](https://sourceware.org/binutils/docs/as/Dcb.html" \l "Dcb)
  + [7.15 .ds[size] number [,fill]](https://sourceware.org/binutils/docs/as/Ds.html" \l "Ds)
  + [7.16 .def name](https://sourceware.org/binutils/docs/as/Def.html" \l "Def)
  + [7.17 .desc symbol, abs-expression](https://sourceware.org/binutils/docs/as/Desc.html" \l "Desc)
  + [7.18 .dim](https://sourceware.org/binutils/docs/as/Dim.html" \l "Dim)
  + [7.19 .double flonums](https://sourceware.org/binutils/docs/as/Double.html" \l "Double)
  + [7.20 .eject](https://sourceware.org/binutils/docs/as/Eject.html" \l "Eject)
  + [7.21 .else](https://sourceware.org/binutils/docs/as/Else.html" \l "Else)
  + [7.22 .elseif](https://sourceware.org/binutils/docs/as/Elseif.html" \l "Elseif)
  + [7.23 .end](https://sourceware.org/binutils/docs/as/End.html" \l "End)
  + [7.24 .endef](https://sourceware.org/binutils/docs/as/Endef.html" \l "Endef)
  + [7.25 .endfunc](https://sourceware.org/binutils/docs/as/Endfunc.html" \l "Endfunc)
  + [7.26 .endif](https://sourceware.org/binutils/docs/as/Endif.html" \l "Endif)
  + [7.27 .equ symbol, expression](https://sourceware.org/binutils/docs/as/Equ.html" \l "Equ)
  + [7.28 .equiv symbol, expression](https://sourceware.org/binutils/docs/as/Equiv.html" \l "Equiv)
  + [7.29 .eqv symbol, expression](https://sourceware.org/binutils/docs/as/Eqv.html" \l "Eqv)
  + [7.30 .err](https://sourceware.org/binutils/docs/as/Err.html" \l "Err)
  + [7.31 .error "string"](https://sourceware.org/binutils/docs/as/Error.html" \l "Error)
  + [7.32 .exitm](https://sourceware.org/binutils/docs/as/Exitm.html" \l "Exitm)
  + [7.33 .extern](https://sourceware.org/binutils/docs/as/Extern.html" \l "Extern)
  + [7.34 .fail expression](https://sourceware.org/binutils/docs/as/Fail.html" \l "Fail)
  + [7.35 .file](https://sourceware.org/binutils/docs/as/File.html" \l "File)
  + [7.36 .fill repeat , size , value](https://sourceware.org/binutils/docs/as/Fill.html" \l "Fill)
  + [7.37 .float flonums](https://sourceware.org/binutils/docs/as/Float.html" \l "Float)
  + [7.38 .func name[,label]](https://sourceware.org/binutils/docs/as/Func.html" \l "Func)
  + [7.39 .global symbol, .globl symbol](https://sourceware.org/binutils/docs/as/Global.html" \l "Global)
  + [7.40 .gnu\_attribute tag,value](https://sourceware.org/binutils/docs/as/Gnu_005fattribute.html" \l "Gnu_005fattribute)
  + [7.41 .hidden names](https://sourceware.org/binutils/docs/as/Hidden.html" \l "Hidden)
  + [7.42 .hword expressions](https://sourceware.org/binutils/docs/as/hword.html" \l "hword)
  + [7.43 .ident](https://sourceware.org/binutils/docs/as/Ident.html" \l "Ident)
  + [7.44 .if absolute expression](https://sourceware.org/binutils/docs/as/If.html" \l "If)
  + [7.45 .incbin "file"[,skip[,count]]](https://sourceware.org/binutils/docs/as/Incbin.html" \l "Incbin)
  + [7.46 .include "file"](https://sourceware.org/binutils/docs/as/Include.html" \l "Include)
  + [7.47 .int expressions](https://sourceware.org/binutils/docs/as/Int.html" \l "Int)
  + [7.48 .internal names](https://sourceware.org/binutils/docs/as/Internal.html" \l "Internal)
  + [7.49 .irp symbol,values…](https://sourceware.org/binutils/docs/as/Irp.html" \l "Irp)
  + [7.50 .irpc symbol,values…](https://sourceware.org/binutils/docs/as/Irpc.html" \l "Irpc)
  + [7.51 .lcomm symbol , length](https://sourceware.org/binutils/docs/as/Lcomm.html" \l "Lcomm)
  + [7.52 .lflags](https://sourceware.org/binutils/docs/as/Lflags.html" \l "Lflags)
  + [7.53 .line line-number](https://sourceware.org/binutils/docs/as/Line.html" \l "Line)
  + [7.54 .linkonce [type]](https://sourceware.org/binutils/docs/as/Linkonce.html" \l "Linkonce)
  + [7.55 .list](https://sourceware.org/binutils/docs/as/List.html" \l "List)
  + [7.56 .ln line-number](https://sourceware.org/binutils/docs/as/Ln.html" \l "Ln)
  + [7.57 .loc fileno lineno [column] [options]](https://sourceware.org/binutils/docs/as/Loc.html" \l "Loc)
  + [7.58 .loc\_mark\_labels enable](https://sourceware.org/binutils/docs/as/Loc_005fmark_005flabels.html" \l "Loc_005fmark_005flabels)
  + [7.59 .local names](https://sourceware.org/binutils/docs/as/Local.html" \l "Local)
  + [7.60 .long expressions](https://sourceware.org/binutils/docs/as/Long.html" \l "Long)
  + [7.61 .macro](https://sourceware.org/binutils/docs/as/Macro.html" \l "Macro)
  + [7.62 .mri val](https://sourceware.org/binutils/docs/as/MRI.html" \l "MRI)
  + [7.63 .noaltmacro](https://sourceware.org/binutils/docs/as/Noaltmacro.html" \l "Noaltmacro)
  + [7.64 .nolist](https://sourceware.org/binutils/docs/as/Nolist.html" \l "Nolist)
  + [7.65 .nops size[, control]](https://sourceware.org/binutils/docs/as/Nops.html" \l "Nops)
  + [7.66 .octa bignums](https://sourceware.org/binutils/docs/as/Octa.html" \l "Octa)
  + [7.67 .offset loc](https://sourceware.org/binutils/docs/as/Offset.html" \l "Offset)
  + [7.68 .org new-lc , fill](https://sourceware.org/binutils/docs/as/Org.html" \l "Org)
  + [7.69 .p2align[wl] abs-expr, abs-expr, abs-expr](https://sourceware.org/binutils/docs/as/P2align.html" \l "P2align)
  + [7.70 .popsection](https://sourceware.org/binutils/docs/as/PopSection.html" \l "PopSection)
  + [7.71 .previous](https://sourceware.org/binutils/docs/as/Previous.html" \l "Previous)
  + [7.72 .print string](https://sourceware.org/binutils/docs/as/Print.html" \l "Print)
  + [7.73 .protected names](https://sourceware.org/binutils/docs/as/Protected.html" \l "Protected)
  + [7.74 .psize lines , columns](https://sourceware.org/binutils/docs/as/Psize.html" \l "Psize)
  + [7.75 .purgem name](https://sourceware.org/binutils/docs/as/Purgem.html" \l "Purgem)
  + [7.76 .pushsection name [, subsection] [, "flags"[, @type[,arguments]]]](https://sourceware.org/binutils/docs/as/PushSection.html" \l "PushSection)
  + [7.77 .quad bignums](https://sourceware.org/binutils/docs/as/Quad.html" \l "Quad)
  + [7.78 .reloc offset, reloc\_name[, expression]](https://sourceware.org/binutils/docs/as/Reloc.html" \l "Reloc)
  + [7.79 .rept count](https://sourceware.org/binutils/docs/as/Rept.html" \l "Rept)
  + [7.80 .sbttl "subheading"](https://sourceware.org/binutils/docs/as/Sbttl.html" \l "Sbttl)
  + [7.81 .scl class](https://sourceware.org/binutils/docs/as/Scl.html" \l "Scl)
  + [7.82 .section name](https://sourceware.org/binutils/docs/as/Section.html" \l "Section)
  + [7.83 .set symbol, expression](https://sourceware.org/binutils/docs/as/Set.html" \l "Set)
  + [7.84 .short expressions](https://sourceware.org/binutils/docs/as/Short.html" \l "Short)
  + [7.85 .single flonums](https://sourceware.org/binutils/docs/as/Single.html" \l "Single)
  + [7.86 .size](https://sourceware.org/binutils/docs/as/Size.html" \l "Size)
  + [7.87 .skip size [,fill]](https://sourceware.org/binutils/docs/as/Skip.html" \l "Skip)
  + [7.88 .sleb128 expressions](https://sourceware.org/binutils/docs/as/Sleb128.html" \l "Sleb128)
  + [7.89 .space size [,fill]](https://sourceware.org/binutils/docs/as/Space.html" \l "Space)
  + [7.90 .stabd, .stabn, .stabs](https://sourceware.org/binutils/docs/as/Stab.html" \l "Stab)
  + [7.91 .string "str", .string8 "str", .string16](https://sourceware.org/binutils/docs/as/String.html" \l "String)
  + [7.92 .struct expression](https://sourceware.org/binutils/docs/as/Struct.html" \l "Struct)
  + [7.93 .subsection name](https://sourceware.org/binutils/docs/as/SubSection.html" \l "SubSection)
  + [7.94 .symver](https://sourceware.org/binutils/docs/as/Symver.html" \l "Symver)
  + [7.95 .tag structname](https://sourceware.org/binutils/docs/as/Tag.html" \l "Tag)
  + [7.96 .text subsection](https://sourceware.org/binutils/docs/as/Text.html" \l "Text)
  + [7.97 .title "heading"](https://sourceware.org/binutils/docs/as/Title.html" \l "Title)
  + [7.98 .type](https://sourceware.org/binutils/docs/as/Type.html" \l "Type)
  + [7.99 .uleb128 expressions](https://sourceware.org/binutils/docs/as/Uleb128.html" \l "Uleb128)
  + [7.100 .val addr](https://sourceware.org/binutils/docs/as/Val.html" \l "Val)
  + [7.101 .version "string"](https://sourceware.org/binutils/docs/as/Version.html" \l "Version)
  + [7.102 .vtable\_entry table, offset](https://sourceware.org/binutils/docs/as/VTableEntry.html" \l "VTableEntry)
  + [7.103 .vtable\_inherit child, parent](https://sourceware.org/binutils/docs/as/VTableInherit.html" \l "VTableInherit)
  + [7.104 .warning "string"](https://sourceware.org/binutils/docs/as/Warning.html" \l "Warning)
  + [7.105 .weak names](https://sourceware.org/binutils/docs/as/Weak.html" \l "Weak)
  + [7.106 .weakref alias, target](https://sourceware.org/binutils/docs/as/Weakref.html" \l "Weakref)
  + [7.107 .word expressions](https://sourceware.org/binutils/docs/as/Word.html" \l "Word)
  + [7.108 .zero size](https://sourceware.org/binutils/docs/as/Zero.html" \l "Zero)
  + [7.109 .2byte expression [, expression]\*](https://sourceware.org/binutils/docs/as/2byte.html" \l "g_t2byte)
  + [7.110 .4byte expression [, expression]\*](https://sourceware.org/binutils/docs/as/4byte.html" \l "g_t4byte)
  + [7.111 .8byte expression [, expression]\*](https://sourceware.org/binutils/docs/as/8byte.html" \l "g_t8byte)
  + [7.112 Deprecated Directives](https://sourceware.org/binutils/docs/as/Deprecated.html" \l "Deprecated)
* [8 Object Attributes](https://sourceware.org/binutils/docs/as/Object-Attributes.html" \l "Object-Attributes)
  + [8.1 GNU Object Attributes](https://sourceware.org/binutils/docs/as/GNU-Object-Attributes.html" \l "GNU-Object-Attributes)
    - [8.1.1 Common GNU attributes](https://sourceware.org/binutils/docs/as/GNU-Object-Attributes.html" \l "Common-GNU-attributes)
    - [8.1.2 MIPS Attributes](https://sourceware.org/binutils/docs/as/GNU-Object-Attributes.html" \l "MIPS-Attributes)
    - [8.1.3 PowerPC Attributes](https://sourceware.org/binutils/docs/as/GNU-Object-Attributes.html" \l "PowerPC-Attributes)
    - [8.1.4 IBM z Systems Attributes](https://sourceware.org/binutils/docs/as/GNU-Object-Attributes.html" \l "IBM-z-Systems-Attributes)
  + [8.2 Defining New Object Attributes](https://sourceware.org/binutils/docs/as/Defining-New-Object-Attributes.html" \l "Defining-New-Object-Attributes)
* [9 Machine Dependent Features](https://sourceware.org/binutils/docs/as/Machine-Dependencies.html" \l "Machine-Dependencies)
  + [9.1 AArch64 Dependent Features](https://sourceware.org/binutils/docs/as/AArch64_002dDependent.html" \l "AArch64_002dDependent)
    - [9.1.1 Options](https://sourceware.org/binutils/docs/as/AArch64-Options.html" \l "AArch64-Options)
    - [9.1.2 Architecture Extensions](https://sourceware.org/binutils/docs/as/AArch64-Extensions.html" \l "AArch64-Extensions)
    - [9.1.3 Syntax](https://sourceware.org/binutils/docs/as/AArch64-Syntax.html" \l "AArch64-Syntax)
      * [9.1.3.1 Special Characters](https://sourceware.org/binutils/docs/as/AArch64_002dChars.html" \l "AArch64_002dChars)
      * [9.1.3.2 Register Names](https://sourceware.org/binutils/docs/as/AArch64_002dRegs.html" \l "AArch64_002dRegs)
      * [9.1.3.3 Relocations](https://sourceware.org/binutils/docs/as/AArch64_002dRelocations.html" \l "AArch64_002dRelocations)
    - [9.1.4 Floating Point](https://sourceware.org/binutils/docs/as/AArch64-Floating-Point.html" \l "AArch64-Floating-Point)
    - [9.1.5 AArch64 Machine Directives](https://sourceware.org/binutils/docs/as/AArch64-Directives.html" \l "AArch64-Directives)
    - [9.1.6 Opcodes](https://sourceware.org/binutils/docs/as/AArch64-Opcodes.html" \l "AArch64-Opcodes)
    - [9.1.7 Mapping Symbols](https://sourceware.org/binutils/docs/as/AArch64-Mapping-Symbols.html" \l "AArch64-Mapping-Symbols)
  + [9.2 Alpha Dependent Features](https://sourceware.org/binutils/docs/as/Alpha_002dDependent.html" \l "Alpha_002dDependent)
    - [9.2.1 Notes](https://sourceware.org/binutils/docs/as/Alpha-Notes.html" \l "Alpha-Notes)
    - [9.2.2 Options](https://sourceware.org/binutils/docs/as/Alpha-Options.html" \l "Alpha-Options)
    - [9.2.3 Syntax](https://sourceware.org/binutils/docs/as/Alpha-Syntax.html" \l "Alpha-Syntax)
      * [9.2.3.1 Special Characters](https://sourceware.org/binutils/docs/as/Alpha_002dChars.html" \l "Alpha_002dChars)
      * [9.2.3.2 Register Names](https://sourceware.org/binutils/docs/as/Alpha_002dRegs.html" \l "Alpha_002dRegs)
      * [9.2.3.3 Relocations](https://sourceware.org/binutils/docs/as/Alpha_002dRelocs.html" \l "Alpha_002dRelocs)
    - [9.2.4 Floating Point](https://sourceware.org/binutils/docs/as/Alpha-Floating-Point.html" \l "Alpha-Floating-Point)
    - [9.2.5 Alpha Assembler Directives](https://sourceware.org/binutils/docs/as/Alpha-Directives.html" \l "Alpha-Directives)
    - [9.2.6 Opcodes](https://sourceware.org/binutils/docs/as/Alpha-Opcodes.html" \l "Alpha-Opcodes)
  + [9.3 ARC Dependent Features](https://sourceware.org/binutils/docs/as/ARC_002dDependent.html" \l "ARC_002dDependent)
    - [9.3.1 Options](https://sourceware.org/binutils/docs/as/ARC-Options.html" \l "ARC-Options)
    - [9.3.2 Syntax](https://sourceware.org/binutils/docs/as/ARC-Syntax.html" \l "ARC-Syntax)
      * [9.3.2.1 Special Characters](https://sourceware.org/binutils/docs/as/ARC_002dChars.html" \l "ARC_002dChars)
      * [9.3.2.2 Register Names](https://sourceware.org/binutils/docs/as/ARC_002dRegs.html" \l "ARC_002dRegs)
    - [9.3.3 ARC Machine Directives](https://sourceware.org/binutils/docs/as/ARC-Directives.html" \l "ARC-Directives)
    - [9.3.4 ARC Assembler Modifiers](https://sourceware.org/binutils/docs/as/ARC-Modifiers.html" \l "ARC-Modifiers)
    - [9.3.5 ARC Pre-defined Symbols](https://sourceware.org/binutils/docs/as/ARC-Symbols.html" \l "ARC-Symbols)
    - [9.3.6 Opcodes](https://sourceware.org/binutils/docs/as/ARC-Opcodes.html" \l "ARC-Opcodes)
  + [9.4 ARM Dependent Features](https://sourceware.org/binutils/docs/as/ARM_002dDependent.html" \l "ARM_002dDependent)
    - [9.4.1 Options](https://sourceware.org/binutils/docs/as/ARM-Options.html" \l "ARM-Options)
    - [9.4.2 Syntax](https://sourceware.org/binutils/docs/as/ARM-Syntax.html" \l "ARM-Syntax)
      * [9.4.2.1 Instruction Set Syntax](https://sourceware.org/binutils/docs/as/ARM_002dInstruction_002dSet.html" \l "ARM_002dInstruction_002dSet)
      * [9.4.2.2 Special Characters](https://sourceware.org/binutils/docs/as/ARM_002dChars.html" \l "ARM_002dChars)
      * [9.4.2.3 Register Names](https://sourceware.org/binutils/docs/as/ARM_002dRegs.html" \l "ARM_002dRegs)
      * [9.4.2.4 ARM relocation generation](https://sourceware.org/binutils/docs/as/ARM_002dRelocations.html" \l "ARM_002dRelocations)
      * [9.4.2.5 NEON Alignment Specifiers](https://sourceware.org/binutils/docs/as/ARM_002dNeon_002dAlignment.html" \l "ARM_002dNeon_002dAlignment)
    - [9.4.3 Floating Point](https://sourceware.org/binutils/docs/as/ARM-Floating-Point.html" \l "ARM-Floating-Point)
    - [9.4.4 ARM Machine Directives](https://sourceware.org/binutils/docs/as/ARM-Directives.html" \l "ARM-Directives)
    - [9.4.5 Opcodes](https://sourceware.org/binutils/docs/as/ARM-Opcodes.html" \l "ARM-Opcodes)
    - [9.4.6 Mapping Symbols](https://sourceware.org/binutils/docs/as/ARM-Mapping-Symbols.html" \l "ARM-Mapping-Symbols)
    - [9.4.7 Unwinding](https://sourceware.org/binutils/docs/as/ARM-Unwinding-Tutorial.html" \l "ARM-Unwinding-Tutorial)
  + [9.5 AVR Dependent Features](https://sourceware.org/binutils/docs/as/AVR_002dDependent.html" \l "AVR_002dDependent)
    - [9.5.1 Options](https://sourceware.org/binutils/docs/as/AVR-Options.html" \l "AVR-Options)
    - [9.5.2 Syntax](https://sourceware.org/binutils/docs/as/AVR-Syntax.html" \l "AVR-Syntax)
      * [9.5.2.1 Special Characters](https://sourceware.org/binutils/docs/as/AVR_002dChars.html" \l "AVR_002dChars)
      * [9.5.2.2 Register Names](https://sourceware.org/binutils/docs/as/AVR_002dRegs.html" \l "AVR_002dRegs)
      * [9.5.2.3 Relocatable Expression Modifiers](https://sourceware.org/binutils/docs/as/AVR_002dModifiers.html" \l "AVR_002dModifiers)
    - [9.5.3 Opcodes](https://sourceware.org/binutils/docs/as/AVR-Opcodes.html" \l "AVR-Opcodes)
    - [9.5.4 Pseudo Instructions](https://sourceware.org/binutils/docs/as/AVR-Pseudo-Instructions.html" \l "AVR-Pseudo-Instructions)
  + [9.6 Blackfin Dependent Features](https://sourceware.org/binutils/docs/as/Blackfin_002dDependent.html" \l "Blackfin_002dDependent)
    - [9.6.1 Options](https://sourceware.org/binutils/docs/as/Blackfin-Options.html" \l "Blackfin-Options)
    - [9.6.2 Syntax](https://sourceware.org/binutils/docs/as/Blackfin-Syntax.html" \l "Blackfin-Syntax)
    - [9.6.3 Directives](https://sourceware.org/binutils/docs/as/Blackfin-Directives.html" \l "Blackfin-Directives)
  + [9.7 CR16 Dependent Features](https://sourceware.org/binutils/docs/as/CR16_002dDependent.html" \l "CR16_002dDependent)
    - [9.7.1 CR16 Operand Qualifiers](https://sourceware.org/binutils/docs/as/CR16-Operand-Qualifiers.html" \l "CR16-Operand-Qualifiers)
    - [9.7.2 CR16 Syntax](https://sourceware.org/binutils/docs/as/CR16-Syntax.html" \l "CR16-Syntax)
      * [9.7.2.1 Special Characters](https://sourceware.org/binutils/docs/as/CR16_002dChars.html" \l "CR16_002dChars)
  + [9.8 CRIS Dependent Features](https://sourceware.org/binutils/docs/as/CRIS_002dDependent.html" \l "CRIS_002dDependent)
    - [9.8.1 Command-line Options](https://sourceware.org/binutils/docs/as/CRIS_002dOpts.html" \l "CRIS_002dOpts)
    - [9.8.2 Instruction expansion](https://sourceware.org/binutils/docs/as/CRIS_002dExpand.html" \l "CRIS_002dExpand)
    - [9.8.3 Symbols](https://sourceware.org/binutils/docs/as/CRIS_002dSymbols.html" \l "CRIS_002dSymbols)
    - [9.8.4 Syntax](https://sourceware.org/binutils/docs/as/CRIS_002dSyntax.html" \l "CRIS_002dSyntax)
      * [9.8.4.1 Special Characters](https://sourceware.org/binutils/docs/as/CRIS_002dChars.html" \l "CRIS_002dChars)
      * [9.8.4.2 Symbols in position-independent code](https://sourceware.org/binutils/docs/as/CRIS_002dPic.html" \l "CRIS_002dPic)
      * [9.8.4.3 Register names](https://sourceware.org/binutils/docs/as/CRIS_002dRegs.html" \l "CRIS_002dRegs)
      * [9.8.4.4 Assembler Directives](https://sourceware.org/binutils/docs/as/CRIS_002dPseudos.html" \l "CRIS_002dPseudos)
  + [9.9 C-SKY Dependent Features](https://sourceware.org/binutils/docs/as/C_002dSKY_002dDependent.html" \l "C_002dSKY_002dDependent)
    - [9.9.1 Options](https://sourceware.org/binutils/docs/as/C_002dSKY-Options.html" \l "C_002dSKY-Options)
    - [9.9.2 Syntax](https://sourceware.org/binutils/docs/as/C_002dSKY-Syntax.html" \l "C_002dSKY-Syntax)
  + [9.10 D10V Dependent Features](https://sourceware.org/binutils/docs/as/D10V_002dDependent.html" \l "D10V_002dDependent)
    - [9.10.1 D10V Options](https://sourceware.org/binutils/docs/as/D10V_002dOpts.html" \l "D10V_002dOpts)
    - [9.10.2 Syntax](https://sourceware.org/binutils/docs/as/D10V_002dSyntax.html" \l "D10V_002dSyntax)
      * [9.10.2.1 Size Modifiers](https://sourceware.org/binutils/docs/as/D10V_002dSize.html" \l "D10V_002dSize)
      * [9.10.2.2 Sub-Instructions](https://sourceware.org/binutils/docs/as/D10V_002dSubs.html" \l "D10V_002dSubs)
      * [9.10.2.3 Special Characters](https://sourceware.org/binutils/docs/as/D10V_002dChars.html" \l "D10V_002dChars)
      * [9.10.2.4 Register Names](https://sourceware.org/binutils/docs/as/D10V_002dRegs.html" \l "D10V_002dRegs)
      * [9.10.2.5 Addressing Modes](https://sourceware.org/binutils/docs/as/D10V_002dAddressing.html" \l "D10V_002dAddressing)
      * [9.10.2.6 @WORD Modifier](https://sourceware.org/binutils/docs/as/D10V_002dWord.html" \l "D10V_002dWord)
    - [9.10.3 Floating Point](https://sourceware.org/binutils/docs/as/D10V_002dFloat.html" \l "D10V_002dFloat)
    - [9.10.4 Opcodes](https://sourceware.org/binutils/docs/as/D10V_002dOpcodes.html" \l "D10V_002dOpcodes)
  + [9.11 D30V Dependent Features](https://sourceware.org/binutils/docs/as/D30V_002dDependent.html" \l "D30V_002dDependent)
    - [9.11.1 D30V Options](https://sourceware.org/binutils/docs/as/D30V_002dOpts.html" \l "D30V_002dOpts)
    - [9.11.2 Syntax](https://sourceware.org/binutils/docs/as/D30V_002dSyntax.html" \l "D30V_002dSyntax)
      * [9.11.2.1 Size Modifiers](https://sourceware.org/binutils/docs/as/D30V_002dSize.html" \l "D30V_002dSize)
      * [9.11.2.2 Sub-Instructions](https://sourceware.org/binutils/docs/as/D30V_002dSubs.html" \l "D30V_002dSubs)
      * [9.11.2.3 Special Characters](https://sourceware.org/binutils/docs/as/D30V_002dChars.html" \l "D30V_002dChars)
      * [9.11.2.4 Guarded Execution](https://sourceware.org/binutils/docs/as/D30V_002dGuarded.html" \l "D30V_002dGuarded)
      * [9.11.2.5 Register Names](https://sourceware.org/binutils/docs/as/D30V_002dRegs.html" \l "D30V_002dRegs)
      * [9.11.2.6 Addressing Modes](https://sourceware.org/binutils/docs/as/D30V_002dAddressing.html" \l "D30V_002dAddressing)
    - [9.11.3 Floating Point](https://sourceware.org/binutils/docs/as/D30V_002dFloat.html" \l "D30V_002dFloat)
    - [9.11.4 Opcodes](https://sourceware.org/binutils/docs/as/D30V_002dOpcodes.html" \l "D30V_002dOpcodes)
  + [9.12 Epiphany Dependent Features](https://sourceware.org/binutils/docs/as/Epiphany_002dDependent.html" \l "Epiphany_002dDependent)
    - [9.12.1 Options](https://sourceware.org/binutils/docs/as/Epiphany-Options.html" \l "Epiphany-Options)
    - [9.12.2 Epiphany Syntax](https://sourceware.org/binutils/docs/as/Epiphany-Syntax.html" \l "Epiphany-Syntax)
      * [9.12.2.1 Special Characters](https://sourceware.org/binutils/docs/as/Epiphany_002dChars.html" \l "Epiphany_002dChars)
  + [9.13 H8/300 Dependent Features](https://sourceware.org/binutils/docs/as/H8_002f300_002dDependent.html" \l "H8_002f300_002dDependent)
    - [9.13.1 Options](https://sourceware.org/binutils/docs/as/H8_002f300-Options.html" \l "H8_002f300-Options)
    - [9.13.2 Syntax](https://sourceware.org/binutils/docs/as/H8_002f300-Syntax.html" \l "H8_002f300-Syntax)
      * [9.13.2.1 Special Characters](https://sourceware.org/binutils/docs/as/H8_002f300_002dChars.html" \l "H8_002f300_002dChars)
      * [9.13.2.2 Register Names](https://sourceware.org/binutils/docs/as/H8_002f300_002dRegs.html" \l "H8_002f300_002dRegs)
      * [9.13.2.3 Addressing Modes](https://sourceware.org/binutils/docs/as/H8_002f300_002dAddressing.html" \l "H8_002f300_002dAddressing)
    - [9.13.3 Floating Point](https://sourceware.org/binutils/docs/as/H8_002f300-Floating-Point.html" \l "H8_002f300-Floating-Point)
    - [9.13.4 H8/300 Machine Directives](https://sourceware.org/binutils/docs/as/H8_002f300-Directives.html" \l "H8_002f300-Directives)
    - [9.13.5 Opcodes](https://sourceware.org/binutils/docs/as/H8_002f300-Opcodes.html" \l "H8_002f300-Opcodes)
  + [9.14 HPPA Dependent Features](https://sourceware.org/binutils/docs/as/HPPA_002dDependent.html" \l "HPPA_002dDependent)
    - [9.14.1 Notes](https://sourceware.org/binutils/docs/as/HPPA-Notes.html" \l "HPPA-Notes)
    - [9.14.2 Options](https://sourceware.org/binutils/docs/as/HPPA-Options.html" \l "HPPA-Options)
    - [9.14.3 Syntax](https://sourceware.org/binutils/docs/as/HPPA-Syntax.html" \l "HPPA-Syntax)
    - [9.14.4 Floating Point](https://sourceware.org/binutils/docs/as/HPPA-Floating-Point.html" \l "HPPA-Floating-Point)
    - [9.14.5 HPPA Assembler Directives](https://sourceware.org/binutils/docs/as/HPPA-Directives.html" \l "HPPA-Directives)
    - [9.14.6 Opcodes](https://sourceware.org/binutils/docs/as/HPPA-Opcodes.html" \l "HPPA-Opcodes)
  + [9.15 80386 Dependent Features](https://sourceware.org/binutils/docs/as/i386_002dDependent.html" \l "i386_002dDependent)
    - [9.15.1 Options](https://sourceware.org/binutils/docs/as/i386_002dOptions.html" \l "i386_002dOptions)
    - [9.15.2 x86 specific Directives](https://sourceware.org/binutils/docs/as/i386_002dDirectives.html" \l "i386_002dDirectives)
    - [9.15.3 i386 Syntactical Considerations](https://sourceware.org/binutils/docs/as/i386_002dSyntax.html" \l "i386_002dSyntax)
      * [9.15.3.1 AT&T Syntax versus Intel Syntax](https://sourceware.org/binutils/docs/as/i386_002dVariations.html" \l "i386_002dVariations)
      * [9.15.3.2 Special Characters](https://sourceware.org/binutils/docs/as/i386_002dChars.html" \l "i386_002dChars)
    - [9.15.4 i386-Mnemonics](https://sourceware.org/binutils/docs/as/i386_002dMnemonics.html" \l "i386_002dMnemonics)
      * [9.15.4.1 Instruction Naming](https://sourceware.org/binutils/docs/as/i386_002dMnemonics.html" \l "Instruction-Naming)
      * [9.15.4.2 AT&T Mnemonic versus Intel Mnemonic](https://sourceware.org/binutils/docs/as/i386_002dMnemonics.html" \l "AT_0026T-Mnemonic-versus-Intel-Mnemonic)
    - [9.15.5 Register Naming](https://sourceware.org/binutils/docs/as/i386_002dRegs.html" \l "i386_002dRegs)
    - [9.15.6 Instruction Prefixes](https://sourceware.org/binutils/docs/as/i386_002dPrefixes.html" \l "i386_002dPrefixes)
    - [9.15.7 Memory References](https://sourceware.org/binutils/docs/as/i386_002dMemory.html" \l "i386_002dMemory)
    - [9.15.8 Handling of Jump Instructions](https://sourceware.org/binutils/docs/as/i386_002dJumps.html" \l "i386_002dJumps)
    - [9.15.9 Floating Point](https://sourceware.org/binutils/docs/as/i386_002dFloat.html" \l "i386_002dFloat)
    - [9.15.10 Intel’s MMX and AMD’s 3DNow! SIMD Operations](https://sourceware.org/binutils/docs/as/i386_002dSIMD.html" \l "i386_002dSIMD)
    - [9.15.11 AMD’s Lightweight Profiling Instructions](https://sourceware.org/binutils/docs/as/i386_002dLWP.html" \l "i386_002dLWP)
    - [9.15.12 Bit Manipulation Instructions](https://sourceware.org/binutils/docs/as/i386_002dBMI.html" \l "i386_002dBMI)
    - [9.15.13 AMD’s Trailing Bit Manipulation Instructions](https://sourceware.org/binutils/docs/as/i386_002dTBM.html" \l "i386_002dTBM)
    - [9.15.14 Writing 16-bit Code](https://sourceware.org/binutils/docs/as/i386_002d16bit.html" \l "i386_002d16bit)
    - [9.15.15 Specifying CPU Architecture](https://sourceware.org/binutils/docs/as/i386_002dArch.html" \l "i386_002dArch)
    - [9.15.16 AT&T Syntax bugs](https://sourceware.org/binutils/docs/as/i386_002dBugs.html" \l "i386_002dBugs)
    - [9.15.17 Notes](https://sourceware.org/binutils/docs/as/i386_002dNotes.html" \l "i386_002dNotes)
  + [9.16 IA-64 Dependent Features](https://sourceware.org/binutils/docs/as/IA_002d64_002dDependent.html" \l "IA_002d64_002dDependent)
    - [9.16.1 Options](https://sourceware.org/binutils/docs/as/IA_002d64-Options.html" \l "IA_002d64-Options)
    - [9.16.2 Syntax](https://sourceware.org/binutils/docs/as/IA_002d64-Syntax.html" \l "IA_002d64-Syntax)
      * [9.16.2.1 Special Characters](https://sourceware.org/binutils/docs/as/IA_002d64_002dChars.html" \l "IA_002d64_002dChars)
      * [9.16.2.2 Register Names](https://sourceware.org/binutils/docs/as/IA_002d64_002dRegs.html" \l "IA_002d64_002dRegs)
      * [9.16.2.3 IA-64 Processor-Status-Register (PSR) Bit Names](https://sourceware.org/binutils/docs/as/IA_002d64_002dBits.html" \l "IA_002d64_002dBits)
      * [9.16.2.4 Relocations](https://sourceware.org/binutils/docs/as/IA_002d64_002dRelocs.html" \l "IA_002d64_002dRelocs)
    - [9.16.3 Opcodes](https://sourceware.org/binutils/docs/as/IA_002d64-Opcodes.html" \l "IA_002d64-Opcodes)
  + [9.17 IP2K Dependent Features](https://sourceware.org/binutils/docs/as/IP2K_002dDependent.html" \l "IP2K_002dDependent)
    - [9.17.1 IP2K Options](https://sourceware.org/binutils/docs/as/IP2K_002dOpts.html" \l "IP2K_002dOpts)
    - [9.17.2 IP2K Syntax](https://sourceware.org/binutils/docs/as/IP2K_002dSyntax.html" \l "IP2K_002dSyntax)
      * [9.17.2.1 Special Characters](https://sourceware.org/binutils/docs/as/IP2K_002dChars.html" \l "IP2K_002dChars)
  + [9.18 LM32 Dependent Features](https://sourceware.org/binutils/docs/as/LM32_002dDependent.html" \l "LM32_002dDependent)
    - [9.18.1 Options](https://sourceware.org/binutils/docs/as/LM32-Options.html" \l "LM32-Options)
    - [9.18.2 Syntax](https://sourceware.org/binutils/docs/as/LM32-Syntax.html" \l "LM32-Syntax)
      * [9.18.2.1 Register Names](https://sourceware.org/binutils/docs/as/LM32_002dRegs.html" \l "LM32_002dRegs)
      * [9.18.2.2 Relocatable Expression Modifiers](https://sourceware.org/binutils/docs/as/LM32_002dModifiers.html" \l "LM32_002dModifiers)
      * [9.18.2.3 Special Characters](https://sourceware.org/binutils/docs/as/LM32_002dChars.html" \l "LM32_002dChars)
    - [9.18.3 Opcodes](https://sourceware.org/binutils/docs/as/LM32-Opcodes.html" \l "LM32-Opcodes)
  + [9.19 M32C Dependent Features](https://sourceware.org/binutils/docs/as/M32C_002dDependent.html" \l "M32C_002dDependent)
    - [9.19.1 M32C Options](https://sourceware.org/binutils/docs/as/M32C_002dOpts.html" \l "M32C_002dOpts)
    - [9.19.2 M32C Syntax](https://sourceware.org/binutils/docs/as/M32C_002dSyntax.html" \l "M32C_002dSyntax)
      * [9.19.2.1 Symbolic Operand Modifiers](https://sourceware.org/binutils/docs/as/M32C_002dModifiers.html" \l "M32C_002dModifiers)
      * [9.19.2.2 Special Characters](https://sourceware.org/binutils/docs/as/M32C_002dChars.html" \l "M32C_002dChars)
  + [9.20 M32R Dependent Features](https://sourceware.org/binutils/docs/as/M32R_002dDependent.html" \l "M32R_002dDependent)
    - [9.20.1 M32R Options](https://sourceware.org/binutils/docs/as/M32R_002dOpts.html" \l "M32R_002dOpts)
    - [9.20.2 M32R Directives](https://sourceware.org/binutils/docs/as/M32R_002dDirectives.html" \l "M32R_002dDirectives)
    - [9.20.3 M32R Warnings](https://sourceware.org/binutils/docs/as/M32R_002dWarnings.html" \l "M32R_002dWarnings)
  + [9.21 M680x0 Dependent Features](https://sourceware.org/binutils/docs/as/M68K_002dDependent.html" \l "M68K_002dDependent)
    - [9.21.1 M680x0 Options](https://sourceware.org/binutils/docs/as/M68K_002dOpts.html" \l "M68K_002dOpts)
    - [9.21.2 Syntax](https://sourceware.org/binutils/docs/as/M68K_002dSyntax.html" \l "M68K_002dSyntax)
    - [9.21.3 Motorola Syntax](https://sourceware.org/binutils/docs/as/M68K_002dMoto_002dSyntax.html" \l "M68K_002dMoto_002dSyntax)
    - [9.21.4 Floating Point](https://sourceware.org/binutils/docs/as/M68K_002dFloat.html" \l "M68K_002dFloat)
    - [9.21.5 680x0 Machine Directives](https://sourceware.org/binutils/docs/as/M68K_002dDirectives.html" \l "M68K_002dDirectives)
    - [9.21.6 Opcodes](https://sourceware.org/binutils/docs/as/M68K_002dopcodes.html" \l "M68K_002dopcodes)
      * [9.21.6.1 Branch Improvement](https://sourceware.org/binutils/docs/as/M68K_002dBranch.html" \l "M68K_002dBranch)
      * [9.21.6.2 Special Characters](https://sourceware.org/binutils/docs/as/M68K_002dChars.html" \l "M68K_002dChars)
  + [9.22 M68HC11 and M68HC12 Dependent Features](https://sourceware.org/binutils/docs/as/M68HC11_002dDependent.html" \l "M68HC11_002dDependent)
    - [9.22.1 M68HC11 and M68HC12 Options](https://sourceware.org/binutils/docs/as/M68HC11_002dOpts.html" \l "M68HC11_002dOpts)
    - [9.22.2 Syntax](https://sourceware.org/binutils/docs/as/M68HC11_002dSyntax.html" \l "M68HC11_002dSyntax)
    - [9.22.3 Symbolic Operand Modifiers](https://sourceware.org/binutils/docs/as/M68HC11_002dModifiers.html" \l "M68HC11_002dModifiers)
    - [9.22.4 Assembler Directives](https://sourceware.org/binutils/docs/as/M68HC11_002dDirectives.html" \l "M68HC11_002dDirectives)
    - [9.22.5 Floating Point](https://sourceware.org/binutils/docs/as/M68HC11_002dFloat.html" \l "M68HC11_002dFloat)
    - [9.22.6 Opcodes](https://sourceware.org/binutils/docs/as/M68HC11_002dopcodes.html" \l "M68HC11_002dopcodes)
      * [9.22.6.1 Branch Improvement](https://sourceware.org/binutils/docs/as/M68HC11_002dBranch.html" \l "M68HC11_002dBranch)
  + [9.23 S12Z Dependent Features](https://sourceware.org/binutils/docs/as/S12Z_002dDependent.html" \l "S12Z_002dDependent)
    - [9.23.1 S12Z Options](https://sourceware.org/binutils/docs/as/S12Z_002dOpts.html" \l "S12Z_002dOpts)
    - [9.23.2 Syntax](https://sourceware.org/binutils/docs/as/S12Z_002dSyntax.html" \l "S12Z_002dSyntax)
    - [9.23.3 Assembler Directives](https://sourceware.org/binutils/docs/as/S12Z_002dDirectives.html" \l "S12Z_002dDirectives)
    - [9.23.4 Opcodes](https://sourceware.org/binutils/docs/as/S12Z_002dopcodes.html" \l "S12Z_002dopcodes)
  + [9.24 Meta Dependent Features](https://sourceware.org/binutils/docs/as/Meta_002dDependent.html" \l "Meta_002dDependent)
    - [9.24.1 Options](https://sourceware.org/binutils/docs/as/Meta-Options.html" \l "Meta-Options)
    - [9.24.2 Syntax](https://sourceware.org/binutils/docs/as/Meta-Syntax.html" \l "Meta-Syntax)
      * [9.24.2.1 Special Characters](https://sourceware.org/binutils/docs/as/Meta_002dChars.html" \l "Meta_002dChars)
      * [9.24.2.2 Register Names](https://sourceware.org/binutils/docs/as/Meta_002dRegs.html" \l "Meta_002dRegs)
  + [9.25 MicroBlaze Dependent Features](https://sourceware.org/binutils/docs/as/MicroBlaze_002dDependent.html" \l "MicroBlaze_002dDependent)
    - [9.25.1 Directives](https://sourceware.org/binutils/docs/as/MicroBlaze-Directives.html" \l "MicroBlaze-Directives)
    - [9.25.2 Syntax for the MicroBlaze](https://sourceware.org/binutils/docs/as/MicroBlaze-Syntax.html" \l "MicroBlaze-Syntax)
      * [9.25.2.1 Special Characters](https://sourceware.org/binutils/docs/as/MicroBlaze_002dChars.html" \l "MicroBlaze_002dChars)
  + [9.26 MIPS Dependent Features](https://sourceware.org/binutils/docs/as/MIPS_002dDependent.html" \l "MIPS_002dDependent)
    - [9.26.1 Assembler options](https://sourceware.org/binutils/docs/as/MIPS-Options.html" \l "MIPS-Options)
    - [9.26.2 High-level assembly macros](https://sourceware.org/binutils/docs/as/MIPS-Macros.html" \l "MIPS-Macros)
    - [9.26.3 Directives to override the size of symbols](https://sourceware.org/binutils/docs/as/MIPS-Symbol-Sizes.html" \l "MIPS-Symbol-Sizes)
    - [9.26.4 Controlling the use of small data accesses](https://sourceware.org/binutils/docs/as/MIPS-Small-Data.html" \l "MIPS-Small-Data)
    - [9.26.5 Directives to override the ISA level](https://sourceware.org/binutils/docs/as/MIPS-ISA.html" \l "MIPS-ISA)
    - [9.26.6 Directives to control code generation](https://sourceware.org/binutils/docs/as/MIPS-assembly-options.html" \l "MIPS-assembly-options)
    - [9.26.7 Directives for extending MIPS 16 bit instructions](https://sourceware.org/binutils/docs/as/MIPS-autoextend.html" \l "MIPS-autoextend)
    - [9.26.8 Directive to mark data as an instruction](https://sourceware.org/binutils/docs/as/MIPS-insn.html" \l "MIPS-insn)
    - [9.26.9 Directives to control the FP ABI](https://sourceware.org/binutils/docs/as/MIPS-FP-ABIs.html" \l "MIPS-FP-ABIs)
      * [9.26.9.1 History of FP ABIs](https://sourceware.org/binutils/docs/as/MIPS-FP-ABI-History.html" \l "MIPS-FP-ABI-History)
      * [9.26.9.2 Supported FP ABIs](https://sourceware.org/binutils/docs/as/MIPS-FP-ABI-Variants.html" \l "MIPS-FP-ABI-Variants)
      * [9.26.9.3 Automatic selection of FP ABI](https://sourceware.org/binutils/docs/as/MIPS-FP-ABI-Selection.html" \l "MIPS-FP-ABI-Selection)
      * [9.26.9.4 Linking different FP ABI variants](https://sourceware.org/binutils/docs/as/MIPS-FP-ABI-Compatibility.html" \l "MIPS-FP-ABI-Compatibility)
    - [9.26.10 Directives to record which NaN encoding is being used](https://sourceware.org/binutils/docs/as/MIPS-NaN-Encodings.html" \l "MIPS-NaN-Encodings)
    - [9.26.11 Directives to save and restore options](https://sourceware.org/binutils/docs/as/MIPS-Option-Stack.html" \l "MIPS-Option-Stack)
    - [9.26.12 Directives to control generation of MIPS ASE instructions](https://sourceware.org/binutils/docs/as/MIPS-ASE-Instruction-Generation-Overrides.html" \l "MIPS-ASE-Instruction-Generation-Overrides)
    - [9.26.13 Directives to override floating-point options](https://sourceware.org/binutils/docs/as/MIPS-Floating_002dPoint.html" \l "MIPS-Floating_002dPoint)
    - [9.26.14 Syntactical considerations for the MIPS assembler](https://sourceware.org/binutils/docs/as/MIPS-Syntax.html" \l "MIPS-Syntax)
      * [9.26.14.1 Special Characters](https://sourceware.org/binutils/docs/as/MIPS_002dChars.html" \l "MIPS_002dChars)
  + [9.27 MMIX Dependent Features](https://sourceware.org/binutils/docs/as/MMIX_002dDependent.html" \l "MMIX_002dDependent)
    - [9.27.1 Command-line Options](https://sourceware.org/binutils/docs/as/MMIX_002dOpts.html" \l "MMIX_002dOpts)
    - [9.27.2 Instruction expansion](https://sourceware.org/binutils/docs/as/MMIX_002dExpand.html" \l "MMIX_002dExpand)
    - [9.27.3 Syntax](https://sourceware.org/binutils/docs/as/MMIX_002dSyntax.html" \l "MMIX_002dSyntax)
      * [9.27.3.1 Special Characters](https://sourceware.org/binutils/docs/as/MMIX_002dChars.html" \l "MMIX_002dChars)
      * [9.27.3.2 Symbols](https://sourceware.org/binutils/docs/as/MMIX_002dSymbols.html" \l "MMIX_002dSymbols)
      * [9.27.3.3 Register names](https://sourceware.org/binutils/docs/as/MMIX_002dRegs.html" \l "MMIX_002dRegs)
      * [9.27.3.4 Assembler Directives](https://sourceware.org/binutils/docs/as/MMIX_002dPseudos.html" \l "MMIX_002dPseudos)
    - [9.27.4 Differences to mmixal](https://sourceware.org/binutils/docs/as/MMIX_002dmmixal.html" \l "MMIX_002dmmixal)
  + [9.28 MSP 430 Dependent Features](https://sourceware.org/binutils/docs/as/MSP430_002dDependent.html" \l "MSP430_002dDependent)
    - [9.28.1 Options](https://sourceware.org/binutils/docs/as/MSP430-Options.html" \l "MSP430-Options)
    - [9.28.2 Syntax](https://sourceware.org/binutils/docs/as/MSP430-Syntax.html" \l "MSP430-Syntax)
      * [9.28.2.1 Macros](https://sourceware.org/binutils/docs/as/MSP430_002dMacros.html" \l "MSP430_002dMacros)
      * [9.28.2.2 Special Characters](https://sourceware.org/binutils/docs/as/MSP430_002dChars.html" \l "MSP430_002dChars)
      * [9.28.2.3 Register Names](https://sourceware.org/binutils/docs/as/MSP430_002dRegs.html" \l "MSP430_002dRegs)
      * [9.28.2.4 Assembler Extensions](https://sourceware.org/binutils/docs/as/MSP430_002dExt.html" \l "MSP430_002dExt)
    - [9.28.3 Floating Point](https://sourceware.org/binutils/docs/as/MSP430-Floating-Point.html" \l "MSP430-Floating-Point)
    - [9.28.4 MSP 430 Machine Directives](https://sourceware.org/binutils/docs/as/MSP430-Directives.html" \l "MSP430-Directives)
    - [9.28.5 Opcodes](https://sourceware.org/binutils/docs/as/MSP430-Opcodes.html" \l "MSP430-Opcodes)
    - [9.28.6 Profiling Capability](https://sourceware.org/binutils/docs/as/MSP430-Profiling-Capability.html" \l "MSP430-Profiling-Capability)
  + [9.29 NDS32 Dependent Features](https://sourceware.org/binutils/docs/as/NDS32_002dDependent.html" \l "NDS32_002dDependent)
    - [9.29.1 NDS32 Options](https://sourceware.org/binutils/docs/as/NDS32-Options.html" \l "NDS32-Options)
    - [9.29.2 Syntax](https://sourceware.org/binutils/docs/as/NDS32-Syntax.html" \l "NDS32-Syntax)
      * [9.29.2.1 Special Characters](https://sourceware.org/binutils/docs/as/NDS32_002dChars.html" \l "NDS32_002dChars)
      * [9.29.2.2 Register Names](https://sourceware.org/binutils/docs/as/NDS32_002dRegs.html" \l "NDS32_002dRegs)
      * [9.29.2.3 Pseudo Instructions](https://sourceware.org/binutils/docs/as/NDS32_002dOps.html" \l "NDS32_002dOps)
  + [9.30 Nios II Dependent Features](https://sourceware.org/binutils/docs/as/NiosII_002dDependent.html" \l "NiosII_002dDependent)
    - [9.30.1 Options](https://sourceware.org/binutils/docs/as/Nios-II-Options.html" \l "Nios-II-Options)
    - [9.30.2 Syntax](https://sourceware.org/binutils/docs/as/Nios-II-Syntax.html" \l "Nios-II-Syntax)
      * [9.30.2.1 Special Characters](https://sourceware.org/binutils/docs/as/Nios-II-Chars.html" \l "Nios-II-Chars)
    - [9.30.3 Nios II Machine Relocations](https://sourceware.org/binutils/docs/as/Nios-II-Relocations.html" \l "Nios-II-Relocations)
    - [9.30.4 Nios II Machine Directives](https://sourceware.org/binutils/docs/as/Nios-II-Directives.html" \l "Nios-II-Directives)
    - [9.30.5 Opcodes](https://sourceware.org/binutils/docs/as/Nios-II-Opcodes.html" \l "Nios-II-Opcodes)
  + [9.31 NS32K Dependent Features](https://sourceware.org/binutils/docs/as/NS32K_002dDependent.html" \l "NS32K_002dDependent)
    - [9.31.1 Syntax](https://sourceware.org/binutils/docs/as/NS32K-Syntax.html" \l "NS32K-Syntax)
      * [9.31.1.1 Special Characters](https://sourceware.org/binutils/docs/as/NS32K_002dChars.html" \l "NS32K_002dChars)
  + [9.32 OPENRISC Dependent Features](https://sourceware.org/binutils/docs/as/OpenRISC_002dDependent.html" \l "OpenRISC_002dDependent)
    - [9.32.1 OpenRISC Syntax](https://sourceware.org/binutils/docs/as/OpenRISC_002dSyntax.html" \l "OpenRISC_002dSyntax)
      * [9.32.1.1 Special Characters](https://sourceware.org/binutils/docs/as/OpenRISC_002dChars.html" \l "OpenRISC_002dChars)
      * [9.32.1.2 Register Names](https://sourceware.org/binutils/docs/as/OpenRISC_002dRegs.html" \l "OpenRISC_002dRegs)
      * [9.32.1.3 Relocations](https://sourceware.org/binutils/docs/as/OpenRISC_002dRelocs.html" \l "OpenRISC_002dRelocs)
    - [9.32.2 Floating Point](https://sourceware.org/binutils/docs/as/OpenRISC_002dFloat.html" \l "OpenRISC_002dFloat)
    - [9.32.3 OpenRISC Machine Directives](https://sourceware.org/binutils/docs/as/OpenRISC_002dDirectives.html" \l "OpenRISC_002dDirectives)
    - [9.32.4 Opcodes](https://sourceware.org/binutils/docs/as/OpenRISC_002dOpcodes.html" \l "OpenRISC_002dOpcodes)
  + [9.33 PDP-11 Dependent Features](https://sourceware.org/binutils/docs/as/PDP_002d11_002dDependent.html" \l "PDP_002d11_002dDependent)
    - [9.33.1 Options](https://sourceware.org/binutils/docs/as/PDP_002d11_002dOptions.html" \l "PDP_002d11_002dOptions)
      * [9.33.1.1 Code Generation Options](https://sourceware.org/binutils/docs/as/PDP_002d11_002dOptions.html" \l "Code-Generation-Options)
      * [9.33.1.2 Instruction Set Extension Options](https://sourceware.org/binutils/docs/as/PDP_002d11_002dOptions.html" \l "Instruction-Set-Extension-Options)
      * [9.33.1.3 CPU Model Options](https://sourceware.org/binutils/docs/as/PDP_002d11_002dOptions.html" \l "CPU-Model-Options)
      * [9.33.1.4 Machine Model Options](https://sourceware.org/binutils/docs/as/PDP_002d11_002dOptions.html" \l "Machine-Model-Options)
    - [9.33.2 Assembler Directives](https://sourceware.org/binutils/docs/as/PDP_002d11_002dPseudos.html" \l "PDP_002d11_002dPseudos)
    - [9.33.3 PDP-11 Assembly Language Syntax](https://sourceware.org/binutils/docs/as/PDP_002d11_002dSyntax.html" \l "PDP_002d11_002dSyntax)
    - [9.33.4 Instruction Naming](https://sourceware.org/binutils/docs/as/PDP_002d11_002dMnemonics.html" \l "PDP_002d11_002dMnemonics)
    - [9.33.5 Synthetic Instructions](https://sourceware.org/binutils/docs/as/PDP_002d11_002dSynthetic.html" \l "PDP_002d11_002dSynthetic)
  + [9.34 picoJava Dependent Features](https://sourceware.org/binutils/docs/as/PJ_002dDependent.html" \l "PJ_002dDependent)
    - [9.34.1 Options](https://sourceware.org/binutils/docs/as/PJ-Options.html" \l "PJ-Options)
    - [9.34.2 PJ Syntax](https://sourceware.org/binutils/docs/as/PJ-Syntax.html" \l "PJ-Syntax)
      * [9.34.2.1 Special Characters](https://sourceware.org/binutils/docs/as/PJ_002dChars.html" \l "PJ_002dChars)
  + [9.35 PowerPC Dependent Features](https://sourceware.org/binutils/docs/as/PPC_002dDependent.html" \l "PPC_002dDependent)
    - [9.35.1 Options](https://sourceware.org/binutils/docs/as/PowerPC_002dOpts.html" \l "PowerPC_002dOpts)
    - [9.35.2 PowerPC Assembler Directives](https://sourceware.org/binutils/docs/as/PowerPC_002dPseudo.html" \l "PowerPC_002dPseudo)
    - [9.35.3 PowerPC Syntax](https://sourceware.org/binutils/docs/as/PowerPC_002dSyntax.html" \l "PowerPC_002dSyntax)
      * [9.35.3.1 Special Characters](https://sourceware.org/binutils/docs/as/PowerPC_002dChars.html" \l "PowerPC_002dChars)
  + [9.36 PRU Dependent Features](https://sourceware.org/binutils/docs/as/PRU_002dDependent.html" \l "PRU_002dDependent)
    - [9.36.1 Options](https://sourceware.org/binutils/docs/as/PRU-Options.html" \l "PRU-Options)
    - [9.36.2 Syntax](https://sourceware.org/binutils/docs/as/PRU-Syntax.html" \l "PRU-Syntax)
      * [9.36.2.1 Special Characters](https://sourceware.org/binutils/docs/as/PRU-Chars.html" \l "PRU-Chars)
    - [9.36.3 PRU Machine Relocations](https://sourceware.org/binutils/docs/as/PRU-Relocations.html" \l "PRU-Relocations)
    - [9.36.4 PRU Machine Directives](https://sourceware.org/binutils/docs/as/PRU-Directives.html" \l "PRU-Directives)
    - [9.36.5 Opcodes](https://sourceware.org/binutils/docs/as/PRU-Opcodes.html" \l "PRU-Opcodes)
  + [9.37 RISC-V Dependent Features](https://sourceware.org/binutils/docs/as/RISC_002dV_002dDependent.html" \l "RISC_002dV_002dDependent)
    - [9.37.1 RISC-V Options](https://sourceware.org/binutils/docs/as/RISC_002dV_002dOptions.html" \l "RISC_002dV_002dOptions)
    - [9.37.2 RISC-V Directives](https://sourceware.org/binutils/docs/as/RISC_002dV_002dDirectives.html" \l "RISC_002dV_002dDirectives)
    - [9.37.3 Instruction Formats](https://sourceware.org/binutils/docs/as/RISC_002dV_002dFormats.html" \l "RISC_002dV_002dFormats)
    - [9.37.4 RISC-V Object Attribute](https://sourceware.org/binutils/docs/as/RISC_002dV_002dATTRIBUTE.html" \l "RISC_002dV_002dATTRIBUTE)
  + [9.38 RL78 Dependent Features](https://sourceware.org/binutils/docs/as/RL78_002dDependent.html" \l "RL78_002dDependent)
    - [9.38.1 RL78 Options](https://sourceware.org/binutils/docs/as/RL78_002dOpts.html" \l "RL78_002dOpts)
    - [9.38.2 Symbolic Operand Modifiers](https://sourceware.org/binutils/docs/as/RL78_002dModifiers.html" \l "RL78_002dModifiers)
    - [9.38.3 Assembler Directives](https://sourceware.org/binutils/docs/as/RL78_002dDirectives.html" \l "RL78_002dDirectives)
    - [9.38.4 Syntax for the RL78](https://sourceware.org/binutils/docs/as/RL78_002dSyntax.html" \l "RL78_002dSyntax)
      * [9.38.4.1 Special Characters](https://sourceware.org/binutils/docs/as/RL78_002dChars.html" \l "RL78_002dChars)
  + [9.39 RX Dependent Features](https://sourceware.org/binutils/docs/as/RX_002dDependent.html" \l "RX_002dDependent)
    - [9.39.1 RX Options](https://sourceware.org/binutils/docs/as/RX_002dOpts.html" \l "RX_002dOpts)
    - [9.39.2 Symbolic Operand Modifiers](https://sourceware.org/binutils/docs/as/RX_002dModifiers.html" \l "RX_002dModifiers)
    - [9.39.3 Assembler Directives](https://sourceware.org/binutils/docs/as/RX_002dDirectives.html" \l "RX_002dDirectives)
    - [9.39.4 Floating Point](https://sourceware.org/binutils/docs/as/RX_002dFloat.html" \l "RX_002dFloat)
    - [9.39.5 Syntax for the RX](https://sourceware.org/binutils/docs/as/RX_002dSyntax.html" \l "RX_002dSyntax)
      * [9.39.5.1 Special Characters](https://sourceware.org/binutils/docs/as/RX_002dChars.html" \l "RX_002dChars)
  + [9.40 IBM S/390 Dependent Features](https://sourceware.org/binutils/docs/as/S_002f390_002dDependent.html" \l "S_002f390_002dDependent)
    - [9.40.1 Options](https://sourceware.org/binutils/docs/as/s390-Options.html" \l "s390-Options)
    - [9.40.2 Special Characters](https://sourceware.org/binutils/docs/as/s390-Characters.html" \l "s390-Characters)
    - [9.40.3 Instruction syntax](https://sourceware.org/binutils/docs/as/s390-Syntax.html" \l "s390-Syntax)
      * [9.40.3.1 Register naming](https://sourceware.org/binutils/docs/as/s390-Register.html" \l "s390-Register)
      * [9.40.3.2 Instruction Mnemonics](https://sourceware.org/binutils/docs/as/s390-Mnemonics.html" \l "s390-Mnemonics)
      * [9.40.3.3 Instruction Operands](https://sourceware.org/binutils/docs/as/s390-Operands.html" \l "s390-Operands)
      * [9.40.3.4 Instruction Formats](https://sourceware.org/binutils/docs/as/s390-Formats.html" \l "s390-Formats)
      * [9.40.3.5 Instruction Aliases](https://sourceware.org/binutils/docs/as/s390-Aliases.html" \l "s390-Aliases)
      * [9.40.3.6 Instruction Operand Modifier](https://sourceware.org/binutils/docs/as/s390-Operand-Modifier.html" \l "s390-Operand-Modifier)
      * [9.40.3.7 Instruction Marker](https://sourceware.org/binutils/docs/as/s390-Instruction-Marker.html" \l "s390-Instruction-Marker)
      * [9.40.3.8 Literal Pool Entries](https://sourceware.org/binutils/docs/as/s390-Literal-Pool-Entries.html" \l "s390-Literal-Pool-Entries)
    - [9.40.4 Assembler Directives](https://sourceware.org/binutils/docs/as/s390-Directives.html" \l "s390-Directives)
    - [9.40.5 Floating Point](https://sourceware.org/binutils/docs/as/s390-Floating-Point.html" \l "s390-Floating-Point)
  + [9.41 SCORE Dependent Features](https://sourceware.org/binutils/docs/as/SCORE_002dDependent.html" \l "SCORE_002dDependent)
    - [9.41.1 Options](https://sourceware.org/binutils/docs/as/SCORE_002dOpts.html" \l "SCORE_002dOpts)
    - [9.41.2 SCORE Assembler Directives](https://sourceware.org/binutils/docs/as/SCORE_002dPseudo.html" \l "SCORE_002dPseudo)
    - [9.41.3 SCORE Syntax](https://sourceware.org/binutils/docs/as/SCORE_002dSyntax.html" \l "SCORE_002dSyntax)
      * [9.41.3.1 Special Characters](https://sourceware.org/binutils/docs/as/SCORE_002dChars.html" \l "SCORE_002dChars)
  + [9.42 Renesas / SuperH SH Dependent Features](https://sourceware.org/binutils/docs/as/SH_002dDependent.html" \l "SH_002dDependent)
    - [9.42.1 Options](https://sourceware.org/binutils/docs/as/SH-Options.html" \l "SH-Options)
    - [9.42.2 Syntax](https://sourceware.org/binutils/docs/as/SH-Syntax.html" \l "SH-Syntax)
      * [9.42.2.1 Special Characters](https://sourceware.org/binutils/docs/as/SH_002dChars.html" \l "SH_002dChars)
      * [9.42.2.2 Register Names](https://sourceware.org/binutils/docs/as/SH_002dRegs.html" \l "SH_002dRegs)
      * [9.42.2.3 Addressing Modes](https://sourceware.org/binutils/docs/as/SH_002dAddressing.html" \l "SH_002dAddressing)
    - [9.42.3 Floating Point](https://sourceware.org/binutils/docs/as/SH-Floating-Point.html" \l "SH-Floating-Point)
    - [9.42.4 SH Machine Directives](https://sourceware.org/binutils/docs/as/SH-Directives.html" \l "SH-Directives)
    - [9.42.5 Opcodes](https://sourceware.org/binutils/docs/as/SH-Opcodes.html" \l "SH-Opcodes)
  + [9.43 SPARC Dependent Features](https://sourceware.org/binutils/docs/as/Sparc_002dDependent.html" \l "Sparc_002dDependent)
    - [9.43.1 Options](https://sourceware.org/binutils/docs/as/Sparc_002dOpts.html" \l "Sparc_002dOpts)
    - [9.43.2 Enforcing aligned data](https://sourceware.org/binutils/docs/as/Sparc_002dAligned_002dData.html" \l "Sparc_002dAligned_002dData)
    - [9.43.3 Sparc Syntax](https://sourceware.org/binutils/docs/as/Sparc_002dSyntax.html" \l "Sparc_002dSyntax)
      * [9.43.3.1 Special Characters](https://sourceware.org/binutils/docs/as/Sparc_002dChars.html" \l "Sparc_002dChars)
      * [9.43.3.2 Register Names](https://sourceware.org/binutils/docs/as/Sparc_002dRegs.html" \l "Sparc_002dRegs)
      * [9.43.3.3 Constants](https://sourceware.org/binutils/docs/as/Sparc_002dConstants.html" \l "Sparc_002dConstants)
      * [9.43.3.4 Relocations](https://sourceware.org/binutils/docs/as/Sparc_002dRelocs.html" \l "Sparc_002dRelocs)
      * [9.43.3.5 Size Translations](https://sourceware.org/binutils/docs/as/Sparc_002dSize_002dTranslations.html" \l "Sparc_002dSize_002dTranslations)
    - [9.43.4 Floating Point](https://sourceware.org/binutils/docs/as/Sparc_002dFloat.html" \l "Sparc_002dFloat)
    - [9.43.5 Sparc Machine Directives](https://sourceware.org/binutils/docs/as/Sparc_002dDirectives.html" \l "Sparc_002dDirectives)
  + [9.44 TIC54X Dependent Features](https://sourceware.org/binutils/docs/as/TIC54X_002dDependent.html" \l "TIC54X_002dDependent)
    - [9.44.1 Options](https://sourceware.org/binutils/docs/as/TIC54X_002dOpts.html" \l "TIC54X_002dOpts)
    - [9.44.2 Blocking](https://sourceware.org/binutils/docs/as/TIC54X_002dBlock.html" \l "TIC54X_002dBlock)
    - [9.44.3 Environment Settings](https://sourceware.org/binutils/docs/as/TIC54X_002dEnv.html" \l "TIC54X_002dEnv)
    - [9.44.4 Constants Syntax](https://sourceware.org/binutils/docs/as/TIC54X_002dConstants.html" \l "TIC54X_002dConstants)
    - [9.44.5 String Substitution](https://sourceware.org/binutils/docs/as/TIC54X_002dSubsyms.html" \l "TIC54X_002dSubsyms)
    - [9.44.6 Local Labels](https://sourceware.org/binutils/docs/as/TIC54X_002dLocals.html" \l "TIC54X_002dLocals)
    - [9.44.7 Math Builtins](https://sourceware.org/binutils/docs/as/TIC54X_002dBuiltins.html" \l "TIC54X_002dBuiltins)
    - [9.44.8 Extended Addressing](https://sourceware.org/binutils/docs/as/TIC54X_002dExt.html" \l "TIC54X_002dExt)
    - [9.44.9 Directives](https://sourceware.org/binutils/docs/as/TIC54X_002dDirectives.html" \l "TIC54X_002dDirectives)
    - [9.44.10 Macros](https://sourceware.org/binutils/docs/as/TIC54X_002dMacros.html" \l "TIC54X_002dMacros)
    - [9.44.11 Memory-mapped Registers](https://sourceware.org/binutils/docs/as/TIC54X_002dMMRegs.html" \l "TIC54X_002dMMRegs)
    - [9.44.12 TIC54X Syntax](https://sourceware.org/binutils/docs/as/TIC54X_002dSyntax.html" \l "TIC54X_002dSyntax)
      * [9.44.12.1 Special Characters](https://sourceware.org/binutils/docs/as/TIC54X_002dChars.html" \l "TIC54X_002dChars)
  + [9.45 TIC6X Dependent Features](https://sourceware.org/binutils/docs/as/TIC6X_002dDependent.html" \l "TIC6X_002dDependent)
    - [9.45.1 TIC6X Options](https://sourceware.org/binutils/docs/as/TIC6X-Options.html" \l "TIC6X-Options)
    - [9.45.2 TIC6X Syntax](https://sourceware.org/binutils/docs/as/TIC6X-Syntax.html" \l "TIC6X-Syntax)
    - [9.45.3 TIC6X Directives](https://sourceware.org/binutils/docs/as/TIC6X-Directives.html" \l "TIC6X-Directives)
  + [9.46 TILE-Gx Dependent Features](https://sourceware.org/binutils/docs/as/TILE_002dGx_002dDependent.html" \l "TILE_002dGx_002dDependent)
    - [9.46.1 Options](https://sourceware.org/binutils/docs/as/TILE_002dGx-Options.html" \l "TILE_002dGx-Options)
    - [9.46.2 Syntax](https://sourceware.org/binutils/docs/as/TILE_002dGx-Syntax.html" \l "TILE_002dGx-Syntax)
      * [9.46.2.1 Opcode Names](https://sourceware.org/binutils/docs/as/TILE_002dGx-Opcodes.html" \l "TILE_002dGx-Opcodes)
      * [9.46.2.2 Register Names](https://sourceware.org/binutils/docs/as/TILE_002dGx-Registers.html" \l "TILE_002dGx-Registers)
      * [9.46.2.3 Symbolic Operand Modifiers](https://sourceware.org/binutils/docs/as/TILE_002dGx-Modifiers.html" \l "TILE_002dGx-Modifiers)
    - [9.46.3 TILE-Gx Directives](https://sourceware.org/binutils/docs/as/TILE_002dGx-Directives.html" \l "TILE_002dGx-Directives)
  + [9.47 TILEPro Dependent Features](https://sourceware.org/binutils/docs/as/TILEPro_002dDependent.html" \l "TILEPro_002dDependent)
    - [9.47.1 Options](https://sourceware.org/binutils/docs/as/TILEPro-Options.html" \l "TILEPro-Options)
    - [9.47.2 Syntax](https://sourceware.org/binutils/docs/as/TILEPro-Syntax.html" \l "TILEPro-Syntax)
      * [9.47.2.1 Opcode Names](https://sourceware.org/binutils/docs/as/TILEPro-Opcodes.html" \l "TILEPro-Opcodes)
      * [9.47.2.2 Register Names](https://sourceware.org/binutils/docs/as/TILEPro-Registers.html" \l "TILEPro-Registers)
      * [9.47.2.3 Symbolic Operand Modifiers](https://sourceware.org/binutils/docs/as/TILEPro-Modifiers.html" \l "TILEPro-Modifiers)
    - [9.47.3 TILEPro Directives](https://sourceware.org/binutils/docs/as/TILEPro-Directives.html" \l "TILEPro-Directives)
  + [9.48 v850 Dependent Features](https://sourceware.org/binutils/docs/as/V850_002dDependent.html" \l "V850_002dDependent)
    - [9.48.1 Options](https://sourceware.org/binutils/docs/as/V850-Options.html" \l "V850-Options)
    - [9.48.2 Syntax](https://sourceware.org/binutils/docs/as/V850-Syntax.html" \l "V850-Syntax)
      * [9.48.2.1 Special Characters](https://sourceware.org/binutils/docs/as/V850_002dChars.html" \l "V850_002dChars)
      * [9.48.2.2 Register Names](https://sourceware.org/binutils/docs/as/V850_002dRegs.html" \l "V850_002dRegs)
    - [9.48.3 Floating Point](https://sourceware.org/binutils/docs/as/V850-Floating-Point.html" \l "V850-Floating-Point)
    - [9.48.4 V850 Machine Directives](https://sourceware.org/binutils/docs/as/V850-Directives.html" \l "V850-Directives)
    - [9.48.5 Opcodes](https://sourceware.org/binutils/docs/as/V850-Opcodes.html" \l "V850-Opcodes)
  + [9.49 VAX Dependent Features](https://sourceware.org/binutils/docs/as/Vax_002dDependent.html" \l "Vax_002dDependent)
    - [9.49.1 VAX Command-Line Options](https://sourceware.org/binutils/docs/as/VAX_002dOpts.html" \l "VAX_002dOpts)
    - [9.49.2 VAX Floating Point](https://sourceware.org/binutils/docs/as/VAX_002dfloat.html" \l "VAX_002dfloat)
    - [9.49.3 Vax Machine Directives](https://sourceware.org/binutils/docs/as/VAX_002ddirectives.html" \l "VAX_002ddirectives)
    - [9.49.4 VAX Opcodes](https://sourceware.org/binutils/docs/as/VAX_002dopcodes.html" \l "VAX_002dopcodes)
    - [9.49.5 VAX Branch Improvement](https://sourceware.org/binutils/docs/as/VAX_002dbranch.html" \l "VAX_002dbranch)
    - [9.49.6 VAX Operands](https://sourceware.org/binutils/docs/as/VAX_002doperands.html" \l "VAX_002doperands)
    - [9.49.7 Not Supported on VAX](https://sourceware.org/binutils/docs/as/VAX_002dno.html" \l "VAX_002dno)
    - [9.49.8 VAX Syntax](https://sourceware.org/binutils/docs/as/VAX_002dSyntax.html" \l "VAX_002dSyntax)
      * [9.49.8.1 Special Characters](https://sourceware.org/binutils/docs/as/VAX_002dChars.html" \l "VAX_002dChars)
  + [9.50 Visium Dependent Features](https://sourceware.org/binutils/docs/as/Visium_002dDependent.html" \l "Visium_002dDependent)
    - [9.50.1 Options](https://sourceware.org/binutils/docs/as/Visium-Options.html" \l "Visium-Options)
    - [9.50.2 Syntax](https://sourceware.org/binutils/docs/as/Visium-Syntax.html" \l "Visium-Syntax)
      * [9.50.2.1 Special Characters](https://sourceware.org/binutils/docs/as/Visium-Characters.html" \l "Visium-Characters)
      * [9.50.2.2 Register Names](https://sourceware.org/binutils/docs/as/Visium-Registers.html" \l "Visium-Registers)
    - [9.50.3 Opcodes](https://sourceware.org/binutils/docs/as/Visium-Opcodes.html" \l "Visium-Opcodes)
  + [9.51 WebAssembly Dependent Features](https://sourceware.org/binutils/docs/as/WebAssembly_002dDependent.html" \l "WebAssembly_002dDependent)
    - [9.51.1 Notes](https://sourceware.org/binutils/docs/as/WebAssembly_002dNotes.html" \l "WebAssembly_002dNotes)
    - [9.51.2 Syntax](https://sourceware.org/binutils/docs/as/WebAssembly_002dSyntax.html" \l "WebAssembly_002dSyntax)
      * [9.51.2.1 Special Characters](https://sourceware.org/binutils/docs/as/WebAssembly_002dChars.html" \l "WebAssembly_002dChars)
      * [9.51.2.2 Relocations](https://sourceware.org/binutils/docs/as/WebAssembly_002dRelocs.html" \l "WebAssembly_002dRelocs)
      * [9.51.2.3 Signatures](https://sourceware.org/binutils/docs/as/WebAssembly_002dSignatures.html" \l "WebAssembly_002dSignatures)
    - [9.51.3 Floating Point](https://sourceware.org/binutils/docs/as/WebAssembly_002dFloating_002dPoint.html" \l "WebAssembly_002dFloating_002dPoint)
    - [9.51.4 Regular Opcodes](https://sourceware.org/binutils/docs/as/WebAssembly_002dOpcodes.html" \l "WebAssembly_002dOpcodes)
    - [9.51.5 WebAssembly Module Layout](https://sourceware.org/binutils/docs/as/WebAssembly_002dmodule_002dlayout.html" \l "WebAssembly_002dmodule_002dlayout)
  + [9.52 XGATE Dependent Features](https://sourceware.org/binutils/docs/as/XGATE_002dDependent.html" \l "XGATE_002dDependent)
    - [9.52.1 XGATE Options](https://sourceware.org/binutils/docs/as/XGATE_002dOpts.html" \l "XGATE_002dOpts)
    - [9.52.2 Syntax](https://sourceware.org/binutils/docs/as/XGATE_002dSyntax.html" \l "XGATE_002dSyntax)
    - [9.52.3 Assembler Directives](https://sourceware.org/binutils/docs/as/XGATE_002dDirectives.html" \l "XGATE_002dDirectives)
    - [9.52.4 Floating Point](https://sourceware.org/binutils/docs/as/XGATE_002dFloat.html" \l "XGATE_002dFloat)
    - [9.52.5 Opcodes](https://sourceware.org/binutils/docs/as/XGATE_002dopcodes.html" \l "XGATE_002dopcodes)
  + [9.53 XStormy16 Dependent Features](https://sourceware.org/binutils/docs/as/XSTORMY16_002dDependent.html" \l "XSTORMY16_002dDependent)
    - [9.53.1 Syntax](https://sourceware.org/binutils/docs/as/XStormy16-Syntax.html" \l "XStormy16-Syntax)
      * [9.53.1.1 Special Characters](https://sourceware.org/binutils/docs/as/XStormy16_002dChars.html" \l "XStormy16_002dChars)
    - [9.53.2 XStormy16 Machine Directives](https://sourceware.org/binutils/docs/as/XStormy16-Directives.html" \l "XStormy16-Directives)
    - [9.53.3 XStormy16 Pseudo-Opcodes](https://sourceware.org/binutils/docs/as/XStormy16-Opcodes.html" \l "XStormy16-Opcodes)
  + [9.54 Xtensa Dependent Features](https://sourceware.org/binutils/docs/as/Xtensa_002dDependent.html" \l "Xtensa_002dDependent)
    - [9.54.1 Command-line Options](https://sourceware.org/binutils/docs/as/Xtensa-Options.html" \l "Xtensa-Options)
    - [9.54.2 Assembler Syntax](https://sourceware.org/binutils/docs/as/Xtensa-Syntax.html" \l "Xtensa-Syntax)
      * [9.54.2.1 Opcode Names](https://sourceware.org/binutils/docs/as/Xtensa-Opcodes.html" \l "Xtensa-Opcodes)
      * [9.54.2.2 Register Names](https://sourceware.org/binutils/docs/as/Xtensa-Registers.html" \l "Xtensa-Registers)
    - [9.54.3 Xtensa Optimizations](https://sourceware.org/binutils/docs/as/Xtensa-Optimizations.html" \l "Xtensa-Optimizations)
      * [9.54.3.1 Using Density Instructions](https://sourceware.org/binutils/docs/as/Density-Instructions.html" \l "Density-Instructions)
      * [9.54.3.2 Automatic Instruction Alignment](https://sourceware.org/binutils/docs/as/Xtensa-Automatic-Alignment.html" \l "Xtensa-Automatic-Alignment)
    - [9.54.4 Xtensa Relaxation](https://sourceware.org/binutils/docs/as/Xtensa-Relaxation.html" \l "Xtensa-Relaxation)
      * [9.54.4.1 Conditional Branch Relaxation](https://sourceware.org/binutils/docs/as/Xtensa-Branch-Relaxation.html" \l "Xtensa-Branch-Relaxation)
      * [9.54.4.2 Function Call Relaxation](https://sourceware.org/binutils/docs/as/Xtensa-Call-Relaxation.html" \l "Xtensa-Call-Relaxation)
      * [9.54.4.3 Jump Relaxation](https://sourceware.org/binutils/docs/as/Xtensa-Jump-Relaxation.html" \l "Xtensa-Jump-Relaxation)
      * [9.54.4.4 Other Immediate Field Relaxation](https://sourceware.org/binutils/docs/as/Xtensa-Immediate-Relaxation.html" \l "Xtensa-Immediate-Relaxation)
    - [9.54.5 Directives](https://sourceware.org/binutils/docs/as/Xtensa-Directives.html" \l "Xtensa-Directives)
      * [9.54.5.1 schedule](https://sourceware.org/binutils/docs/as/Schedule-Directive.html" \l "Schedule-Directive)
      * [9.54.5.2 longcalls](https://sourceware.org/binutils/docs/as/Longcalls-Directive.html" \l "Longcalls-Directive)
      * [9.54.5.3 transform](https://sourceware.org/binutils/docs/as/Transform-Directive.html" \l "Transform-Directive)
      * [9.54.5.4 literal](https://sourceware.org/binutils/docs/as/Literal-Directive.html" \l "Literal-Directive)
      * [9.54.5.5 literal\_position](https://sourceware.org/binutils/docs/as/Literal-Position-Directive.html" \l "Literal-Position-Directive)
      * [9.54.5.6 literal\_prefix](https://sourceware.org/binutils/docs/as/Literal-Prefix-Directive.html" \l "Literal-Prefix-Directive)
      * [9.54.5.7 absolute-literals](https://sourceware.org/binutils/docs/as/Absolute-Literals-Directive.html" \l "Absolute-Literals-Directive)
  + [9.55 Z80 Dependent Features](https://sourceware.org/binutils/docs/as/Z80_002dDependent.html" \l "Z80_002dDependent)
    - [9.55.1 Options](https://sourceware.org/binutils/docs/as/Z80-Options.html" \l "Z80-Options)
    - [9.55.2 Syntax](https://sourceware.org/binutils/docs/as/Z80-Syntax.html" \l "Z80-Syntax)
      * [9.55.2.1 Special Characters](https://sourceware.org/binutils/docs/as/Z80_002dChars.html" \l "Z80_002dChars)
      * [9.55.2.2 Register Names](https://sourceware.org/binutils/docs/as/Z80_002dRegs.html" \l "Z80_002dRegs)
      * [9.55.2.3 Case Sensitivity](https://sourceware.org/binutils/docs/as/Z80_002dCase.html" \l "Z80_002dCase)
    - [9.55.3 Floating Point](https://sourceware.org/binutils/docs/as/Z80-Floating-Point.html" \l "Z80-Floating-Point)
    - [9.55.4 Z80 Assembler Directives](https://sourceware.org/binutils/docs/as/Z80-Directives.html" \l "Z80-Directives)
    - [9.55.5 Opcodes](https://sourceware.org/binutils/docs/as/Z80-Opcodes.html" \l "Z80-Opcodes)
  + [9.56 Z8000 Dependent Features](https://sourceware.org/binutils/docs/as/Z8000_002dDependent.html" \l "Z8000_002dDependent)
    - [9.56.1 Options](https://sourceware.org/binutils/docs/as/Z8000-Options.html" \l "Z8000-Options)
    - [9.56.2 Syntax](https://sourceware.org/binutils/docs/as/Z8000-Syntax.html" \l "Z8000-Syntax)
      * [9.56.2.1 Special Characters](https://sourceware.org/binutils/docs/as/Z8000_002dChars.html" \l "Z8000_002dChars)
      * [9.56.2.2 Register Names](https://sourceware.org/binutils/docs/as/Z8000_002dRegs.html" \l "Z8000_002dRegs)
      * [9.56.2.3 Addressing Modes](https://sourceware.org/binutils/docs/as/Z8000_002dAddressing.html" \l "Z8000_002dAddressing)
    - [9.56.3 Assembler Directives for the Z8000](https://sourceware.org/binutils/docs/as/Z8000-Directives.html" \l "Z8000-Directives)
    - [9.56.4 Opcodes](https://sourceware.org/binutils/docs/as/Z8000-Opcodes.html" \l "Z8000-Opcodes)
* [10 Reporting Bugs](https://sourceware.org/binutils/docs/as/Reporting-Bugs.html" \l "Reporting-Bugs)
  + [10.1 Have You Found a Bug?](https://sourceware.org/binutils/docs/as/Bug-Criteria.html" \l "Bug-Criteria)
  + [10.2 How to Report Bugs](https://sourceware.org/binutils/docs/as/Bug-Reporting.html" \l "Bug-Reporting)
* [11 Acknowledgements](https://sourceware.org/binutils/docs/as/Acknowledgements.html" \l "Acknowledgements)
* [Appendix A GNU Free Documentation License](https://sourceware.org/binutils/docs/as/GNU-Free-Documentation-License.html" \l "GNU-Free-Documentation-License)
* [AS Index](https://sourceware.org/binutils/docs/as/AS-Index.html" \l "AS-Index)

Next: [Overview](https://sourceware.org/binutils/docs/as/Overview.html" \l "Overview), Up: [(dir)](https://sourceware.org/binutils/docs/dir/index.html)   [[Contents](https://sourceware.org/binutils/docs/as/index.html" \l "SEC_Contents" \o "Table of contents)][[Index](https://sourceware.org/binutils/docs/as/AS-Index.html" \l "AS-Index" \o "Index)]

# Using as

This file is a user guide to the GNU assembler as (GNU Binutils) version 2.32.

This document is distributed under the terms of the GNU Free Documentation License. A copy of the license is included in the section entitled “GNU Free Documentation License”.

|  |  |  |
| --- | --- | --- |
| • [Overview](https://sourceware.org/binutils/docs/as/Overview.html" \l "Overview): |  | Overview |
| • [Invoking](https://sourceware.org/binutils/docs/as/Invoking.html" \l "Invoking): |  | Command-Line Options |
| • [Syntax](https://sourceware.org/binutils/docs/as/Syntax.html" \l "Syntax): |  | Syntax |
| • [Sections](https://sourceware.org/binutils/docs/as/Sections.html" \l "Sections): |  | Sections and Relocation |
| • [Symbols](https://sourceware.org/binutils/docs/as/Symbols.html" \l "Symbols): |  | Symbols |
| • [Expressions](https://sourceware.org/binutils/docs/as/Expressions.html" \l "Expressions): |  | Expressions |
| • [Pseudo Ops](https://sourceware.org/binutils/docs/as/Pseudo-Ops.html" \l "Pseudo-Ops): |  | Assembler Directives |
| • [Object Attributes](https://sourceware.org/binutils/docs/as/Object-Attributes.html" \l "Object-Attributes): |  | Object Attributes |
| • [Machine Dependencies](https://sourceware.org/binutils/docs/as/Machine-Dependencies.html" \l "Machine-Dependencies): |  | Machine Dependent Features |
| • [Reporting Bugs](https://sourceware.org/binutils/docs/as/Reporting-Bugs.html" \l "Reporting-Bugs): |  | Reporting Bugs |
| • [Acknowledgements](https://sourceware.org/binutils/docs/as/Acknowledgements.html" \l "Acknowledgements): |  | Who Did What |
| • [GNU Free Documentation License](https://sourceware.org/binutils/docs/as/GNU-Free-Documentation-License.html" \l "GNU-Free-Documentation-License): |  | GNU Free Documentation License |
| • [AS Index](https://sourceware.org/binutils/docs/as/AS-Index.html" \l "AS-Index): |  | AS Index |

Next: [Overview](https://sourceware.org/binutils/docs/as/Overview.html" \l "Overview), Up: [(dir)](https://sourceware.org/binutils/docs/dir/index.html)   [[Contents](https://sourceware.org/binutils/docs/as/index.html" \l "SEC_Contents" \o "Table of contents)][[Index](https://sourceware.org/binutils/docs/as/AS-Index.html" \l "AS-Index" \o "Index)]

## 1 Overview

Here is a brief summary of how to invoke as. For details, see [Command-Line Options](https://sourceware.org/binutils/docs/as/Invoking.html" \l "Invoking).

as [**-a**[**cdghlns**][=file]] [**–alternate**] [**-D**]

[**–compress-debug-sections**] [**–nocompress-debug-sections**]

[**–debug-prefix-map** old=new]

[**–defsym** sym=val] [**-f**] [**-g**] [**–gstabs**]

[**–gstabs+**] [**–gdwarf-2**] [**–gdwarf-sections**]

[**–help**] [**-I** dir] [**-J**]

[**-K**] [**-L**] [**–listing-lhs-width**=NUM]

[**–listing-lhs-width2**=NUM] [**–listing-rhs-width**=NUM]

[**–listing-cont-lines**=NUM] [**–keep-locals**]

[**–no-pad-sections**]

[**-o** objfile] [**-R**]

[**–hash-size**=NUM] [**–reduce-memory-overheads**]

[**–statistics**]

[**-v**] [**-version**] [**–version**]

[**-W**] [**–warn**] [**–fatal-warnings**] [**-w**] [**-x**]

[**-Z**] [**@FILE**]

[**–sectname-subst**] [**–size-check=[error|warning]**]

[**–elf-stt-common=[no|yes]**]

[**–generate-missing-build-notes=[no|yes]**]

[**–target-help**] [target-options]

[**–**|files …]

Target AArch64 options:

[**-EB**|**-EL**]

[**-mabi**=ABI]

Target Alpha options:

[**-mcpu**]

[**-mdebug** | **-no-mdebug**]

[**-replace** | **-noreplace**]

[**-relax**] [**-g**] [**-Gsize**]

[**-F**] [**-32addr**]

Target ARC options:

[**-mcpu=cpu**]

[**-mA6**|**-mARC600**|**-mARC601**|**-mA7**|**-mARC700**|**-mEM**|**-mHS**]

[**-mcode-density**]

[**-mrelax**]

[**-EB**|**-EL**]

Target ARM options:

[**-mcpu**=processor[+extension…]]

[**-march**=architecture[+extension…]]

[**-mfpu**=floating-point-format]

[**-mfloat-abi**=abi]

[**-meabi**=ver]

[**-mthumb**]

[**-EB**|**-EL**]

[**-mapcs-32**|**-mapcs-26**|**-mapcs-float**|

**-mapcs-reentrant**]

[**-mthumb-interwork**] [**-k**]

Target Blackfin options:

[**-mcpu**=processor[-sirevision]]

[**-mfdpic**]

[**-mno-fdpic**]

[**-mnopic**]

Target CRIS options:

[**–underscore** | **–no-underscore**]

[**–pic**] [**-N**]

[**–emulation=criself** | **–emulation=crisaout**]

[**–march=v0\_v10** | **–march=v10** | **–march=v32** | **–march=common\_v10\_v32**]

Target C-SKY options:

[**-march=arch**] [**-mcpu=cpu**]

[**-EL**] [**-mlittle-endian**] [**-EB**] [**-mbig-endian**]

[**-fpic**] [**-pic**]

[**-mljump**] [**-mno-ljump**]

[**-force2bsr**] [**-mforce2bsr**] [**-no-force2bsr**] [**-mno-force2bsr**]

[**-jsri2bsr**] [**-mjsri2bsr**] [**-no-jsri2bsr** ] [**-mno-jsri2bsr**]

[**-mnolrw** ] [**-mno-lrw**]

[**-melrw**] [**-mno-elrw**]

[**-mlaf** ] [**-mliterals-after-func**]

[**-mno-laf**] [**-mno-literals-after-func**]

[**-mlabr**] [**-mliterals-after-br**]

[**-mno-labr**] [**-mnoliterals-after-br**]

[**-mistack**] [**-mno-istack**]

[**-mhard-float**] [**-mmp**] [**-mcp**] [**-mcache**]

[**-msecurity**] [**-mtrust**]

[**-mdsp**] [**-medsp**] [**-mvdsp**]

Target D10V options:

[**-O**]

Target D30V options:

[**-O**|**-n**|**-N**]

Target EPIPHANY options:

[**-mepiphany**|**-mepiphany16**]

Target H8/300 options:

[-h-tick-hex]

Target i386 options:

[**–32**|**–x32**|**–64**] [**-n**]

[**-march**=CPU[+EXTENSION…]] [**-mtune**=CPU]

Target IA-64 options:

[**-mconstant-gp**|**-mauto-pic**]

[**-milp32**|**-milp64**|**-mlp64**|**-mp64**]

[**-mle**|**mbe**]

[**-mtune=itanium1**|**-mtune=itanium2**]

[**-munwind-check=warning**|**-munwind-check=error**]

[**-mhint.b=ok**|**-mhint.b=warning**|**-mhint.b=error**]

[**-x**|**-xexplicit**] [**-xauto**] [**-xdebug**]

Target IP2K options:

[**-mip2022**|**-mip2022ext**]

Target M32C options:

[**-m32c**|**-m16c**] [-relax] [-h-tick-hex]

Target M32R options:

[**–m32rx**|**–[no-]warn-explicit-parallel-conflicts**|

**–W[n]p**]

Target M680X0 options:

[**-l**] [**-m68000**|**-m68010**|**-m68020**|…]

Target M68HC11 options:

[**-m68hc11**|**-m68hc12**|**-m68hcs12**|**-mm9s12x**|**-mm9s12xg**]

[**-mshort**|**-mlong**]

[**-mshort-double**|**-mlong-double**]

[**–force-long-branches**] [**–short-branches**]

[**–strict-direct-mode**] [**–print-insn-syntax**]

[**–print-opcodes**] [**–generate-example**]

Target MCORE options:

[**-jsri2bsr**] [**-sifilter**] [**-relax**]

[**-mcpu=[210|340]**]

Target Meta options:

[**-mcpu=cpu**] [**-mfpu=cpu**] [**-mdsp=cpu**]Target MICROBLAZE options:

Target MIPS options:

[**-nocpp**] [**-EL**] [**-EB**] [**-O**[optimization level]]

[**-g**[debug level]] [**-G** num] [**-KPIC**] [**-call\_shared**]

[**-non\_shared**] [**-xgot** [**-mvxworks-pic**]

[**-mabi**=ABI] [**-32**] [**-n32**] [**-64**] [**-mfp32**] [**-mgp32**]

[**-mfp64**] [**-mgp64**] [**-mfpxx**]

[**-modd-spreg**] [**-mno-odd-spreg**]

[**-march**=CPU] [**-mtune**=CPU] [**-mips1**] [**-mips2**]

[**-mips3**] [**-mips4**] [**-mips5**] [**-mips32**] [**-mips32r2**]

[**-mips32r3**] [**-mips32r5**] [**-mips32r6**] [**-mips64**] [**-mips64r2**]

[**-mips64r3**] [**-mips64r5**] [**-mips64r6**]

[**-construct-floats**] [**-no-construct-floats**]

[**-mignore-branch-isa**] [**-mno-ignore-branch-isa**]

[**-mnan=encoding**]

[**-trap**] [**-no-break**] [**-break**] [**-no-trap**]

[**-mips16**] [**-no-mips16**]

[**-mmips16e2**] [**-mno-mips16e2**]

[**-mmicromips**] [**-mno-micromips**]

[**-msmartmips**] [**-mno-smartmips**]

[**-mips3d**] [**-no-mips3d**]

[**-mdmx**] [**-no-mdmx**]

[**-mdsp**] [**-mno-dsp**]

[**-mdspr2**] [**-mno-dspr2**]

[**-mdspr3**] [**-mno-dspr3**]

[**-mmsa**] [**-mno-msa**]

[**-mxpa**] [**-mno-xpa**]

[**-mmt**] [**-mno-mt**]

[**-mmcu**] [**-mno-mcu**]

[**-mcrc**] [**-mno-crc**]

[**-mginv**] [**-mno-ginv**]

[**-mloongson-mmi**] [**-mno-loongson-mmi**]

[**-mloongson-cam**] [**-mno-loongson-cam**]

[**-mloongson-ext**] [**-mno-loongson-ext**]

[**-mloongson-ext2**] [**-mno-loongson-ext2**]

[**-minsn32**] [**-mno-insn32**]

[**-mfix7000**] [**-mno-fix7000**]

[**-mfix-rm7000**] [**-mno-fix-rm7000**]

[**-mfix-vr4120**] [**-mno-fix-vr4120**]

[**-mfix-vr4130**] [**-mno-fix-vr4130**]

[**-mfix-r5900**] [**-mno-fix-r5900**]

[**-mdebug**] [**-no-mdebug**]

[**-mpdr**] [**-mno-pdr**]

Target MMIX options:

[**–fixed-special-register-names**] [**–globalize-symbols**]

[**–gnu-syntax**] [**–relax**] [**–no-predefined-symbols**]

[**–no-expand**] [**–no-merge-gregs**] [**-x**]

[**–linker-allocated-gregs**]

Target Nios II options:

[**-relax-all**] [**-relax-section**] [**-no-relax**]

[**-EB**] [**-EL**]

Target NDS32 options:

[**-EL**] [**-EB**] [**-O**] [**-Os**] [**-mcpu=cpu**]

[**-misa=isa**] [**-mabi=abi**] [**-mall-ext**]

[**-m[no-]16-bit**] [**-m[no-]perf-ext**] [**-m[no-]perf2-ext**]

[**-m[no-]string-ext**] [**-m[no-]dsp-ext**] [**-m[no-]mac**] [**-m[no-]div**]

[**-m[no-]audio-isa-ext**] [**-m[no-]fpu-sp-ext**] [**-m[no-]fpu-dp-ext**]

[**-m[no-]fpu-fma**] [**-mfpu-freg=FREG**] [**-mreduced-regs**]

[**-mfull-regs**] [**-m[no-]dx-regs**] [**-mpic**] [**-mno-relax**]

[**-mb2bb**]

Target PDP11 options:

[**-mpic**|**-mno-pic**] [**-mall**] [**-mno-extensions**]

[**-m**extension|**-mno-**extension]

[**-m**cpu] [**-m**machine]

Target picoJava options:

[**-mb**|**-me**]

Target PowerPC options:

[**-a32**|**-a64**]

[**-mpwrx**|**-mpwr2**|**-mpwr**|**-m601**|**-mppc**|**-mppc32**|**-m603**|**-m604**|**-m403**|**-m405**|

**-m440**|**-m464**|**-m476**|**-m7400**|**-m7410**|**-m7450**|**-m7455**|**-m750cl**|**-mgekko**|

**-mbroadway**|**-mppc64**|**-m620**|**-me500**|**-e500x2**|**-me500mc**|**-me500mc64**|**-me5500**|

**-me6500**|**-mppc64bridge**|**-mbooke**|**-mpower4**|**-mpwr4**|**-mpower5**|**-mpwr5**|**-mpwr5x**|

**-mpower6**|**-mpwr6**|**-mpower7**|**-mpwr7**|**-mpower8**|**-mpwr8**|**-mpower9**|**-mpwr9-ma2**|

**-mcell**|**-mspe**|**-mspe2**|**-mtitan**|**-me300**|**-mcom**]

[**-many**] [**-maltivec**|**-mvsx**|**-mhtm**|**-mvle**]

[**-mregnames**|**-mno-regnames**]

[**-mrelocatable**|**-mrelocatable-lib**|**-K PIC**] [**-memb**]

[**-mlittle**|**-mlittle-endian**|**-le**|**-mbig**|**-mbig-endian**|**-be**]

[**-msolaris**|**-mno-solaris**]

[**-nops=count**]

Target PRU options:

[**-link-relax**]

[**-mnolink-relax**]

[**-mno-warn-regname-label**]

Target RISC-V options:

[**-fpic**|**-fPIC**|**-fno-pic**]

[**-march**=ISA]

[**-mabi**=ABI]

Target RL78 options:

[**-mg10**]

[**-m32bit-doubles**|**-m64bit-doubles**]

Target RX options:

[**-mlittle-endian**|**-mbig-endian**]

[**-m32bit-doubles**|**-m64bit-doubles**]

[**-muse-conventional-section-names**]

[**-msmall-data-limit**]

[**-mpid**]

[**-mrelax**]

[**-mint-register=number**]

[**-mgcc-abi**|**-mrx-abi**]

Target s390 options:

[**-m31**|**-m64**] [**-mesa**|**-mzarch**] [**-march**=CPU]

[**-mregnames**|**-mno-regnames**]

[**-mwarn-areg-zero**]

Target SCORE options:

[**-EB**][**-EL**][**-FIXDD**][**-NWARN**]

[**-SCORE5**][**-SCORE5U**][**-SCORE7**][**-SCORE3**]

[**-march=score7**][**-march=score3**]

[**-USE\_R1**][**-KPIC**][**-O0**][**-G** num][**-V**]

Target SPARC options:

[**-Av6**|**-Av7**|**-Av8**|**-Aleon**|**-Asparclet**|**-Asparclite**

**-Av8plus**|**-Av8plusa**|**-Av8plusb**|**-Av8plusc**|**-Av8plusd**

**-Av8plusv**|**-Av8plusm**|**-Av9**|**-Av9a**|**-Av9b**|**-Av9c**

**-Av9d**|**-Av9e**|**-Av9v**|**-Av9m**|**-Asparc**|**-Asparcvis**

**-Asparcvis2**|**-Asparcfmaf**|**-Asparcima**|**-Asparcvis3**

**-Asparcvisr**|**-Asparc5**]

[**-xarch=v8plus**|**-xarch=v8plusa**]|**-xarch=v8plusb**|**-xarch=v8plusc**

**-xarch=v8plusd**|**-xarch=v8plusv**|**-xarch=v8plusm**|**-xarch=v9**

**-xarch=v9a**|**-xarch=v9b**|**-xarch=v9c**|**-xarch=v9d**|**-xarch=v9e**

**-xarch=v9v**|**-xarch=v9m**|**-xarch=sparc**|**-xarch=sparcvis**

**-xarch=sparcvis2**|**-xarch=sparcfmaf**|**-xarch=sparcima**

**-xarch=sparcvis3**|**-xarch=sparcvisr**|**-xarch=sparc5**

**-bump**]

[**-32**|**-64**]

[**–enforce-aligned-data**][**–dcti-couples-detect**]

Target TIC54X options:

[**-mcpu=54[123589]**|**-mcpu=54[56]lp**] [**-mfar-mode**|**-mf**]

[**-merrors-to-file** <filename>|**-me** <filename>]

Target TIC6X options:

[**-march=arch**] [**-mbig-endian**|**-mlittle-endian**]

[**-mdsbt**|**-mno-dsbt**] [**-mpid=no**|**-mpid=near**|**-mpid=far**]

[**-mpic**|**-mno-pic**]

Target TILE-Gx options:

[**-m32**|**-m64**][**-EB**][**-EL**]

Target Visium options:

[**-mtune=arch**]

Target Xtensa options:

[**–[no-]text-section-literals**] [**–[no-]auto-litpools**]

[**–[no-]absolute-literals**]

[**–[no-]target-align**] [**–[no-]longcalls**]

[**–[no-]transform**]

[**–rename-section** oldname=newname]

[**–[no-]trampolines**]

Target Z80 options:

[**-z80**] [**-r800**]

[ **-ignore-undocumented-instructions**] [**-Wnud**]

[ **-ignore-unportable-instructions**] [**-Wnup**]

[ **-warn-undocumented-instructions**] [**-Wud**]

[ **-warn-unportable-instructions**] [**-Wup**]

[ **-forbid-undocumented-instructions**] [**-Fud**]

[ **-forbid-unportable-instructions**] [**-Fup**]

@file

Read command-line options from file. The options read are inserted in place of the original @file option. If file does not exist, or cannot be read, then the option will be treated literally, and not removed.

Options in file are separated by whitespace. A whitespace character may be included in an option by surrounding the entire option in either single or double quotes. Any character (including a backslash) may be included by prefixing the character to be included with a backslash. The file may itself contain additional @file options; any such options will be processed recursively.

-a[cdghlmns]

Turn on listings, in any of a variety of ways:

-ac

omit false conditionals

-ad

omit debugging directives

-ag

include general information, like as version and options passed

-ah

include high-level source

-al

include assembly

-am

include macro expansions

-an

omit forms processing

-as

include symbols

=file

set the name of the listing file

You may combine these options; for example, use ‘-aln’ for assembly listing without forms processing. The ‘=file’ option, if used, must be the last one. By itself, ‘-a’ defaults to ‘-ahls’.

--alternate

Begin in alternate macro mode. See [.altmacro](https://sourceware.org/binutils/docs/as/Altmacro.html" \l "Altmacro).

--compress-debug-sections

Compress DWARF debug sections using zlib with SHF\_COMPRESSED from the ELF ABI. The resulting object file may not be compatible with older linkers and object file utilities. Note if compression would make a given section larger then it is not compressed.

--compress-debug-sections=none

--compress-debug-sections=zlib

--compress-debug-sections=zlib-gnu

--compress-debug-sections=zlib-gabi

These options control how DWARF debug sections are compressed. --compress-debug-sections=none is equivalent to --nocompress-debug-sections. --compress-debug-sections=zlib and --compress-debug-sections=zlib-gabi are equivalent to --compress-debug-sections. --compress-debug-sections=zlib-gnu compresses DWARF debug sections using zlib. The debug sections are renamed to begin with ‘.zdebug’. Note if compression would make a given section larger then it is not compressed nor renamed.

--nocompress-debug-sections

Do not compress DWARF debug sections. This is usually the default for all targets except the x86/x86\_64, but a configure time option can be used to override this.

-D

Ignored. This option is accepted for script compatibility with calls to other assemblers.

--debug-prefix-map old=new

When assembling files in directory old, record debugging information describing them as in new instead.

--defsym sym=value

Define the symbol sym to be value before assembling the input file. value must be an integer constant. As in C, a leading ‘0x’ indicates a hexadecimal value, and a leading ‘0’ indicates an octal value. The value of the symbol can be overridden inside a source file via the use of a .set pseudo-op.

-f

“fast”—skip whitespace and comment preprocessing (assume source is compiler output).

-g

--gen-debug

Generate debugging information for each assembler source line using whichever debug format is preferred by the target. This currently means either STABS, ECOFF or DWARF2.

--gstabs

Generate stabs debugging information for each assembler line. This may help debugging assembler code, if the debugger can handle it.

--gstabs+

Generate stabs debugging information for each assembler line, with GNU extensions that probably only gdb can handle, and that could make other debuggers crash or refuse to read your program. This may help debugging assembler code. Currently the only GNU extension is the location of the current working directory at assembling time.

--gdwarf-2

Generate DWARF2 debugging information for each assembler line. This may help debugging assembler code, if the debugger can handle it. Note—this option is only supported by some targets, not all of them.

--gdwarf-sections

Instead of creating a .debug\_line section, create a series of .debug\_line.foo sections where foo is the name of the corresponding code section. For example a code section called .text.funcwill have its dwarf line number information placed into a section called .debug\_line.text.func. If the code section is just called .text then debug line section will still be called just .debug\_line without any suffix.

--size-check=error

--size-check=warning

Issue an error or warning for invalid ELF .size directive.

--elf-stt-common=no

--elf-stt-common=yes

These options control whether the ELF assembler should generate common symbols with the STT\_COMMON type. The default can be controlled by a configure option --enable-elf-stt-common.

--generate-missing-build-notes=yes

--generate-missing-build-notes=no

These options control whether the ELF assembler should generate GNU Build attribute notes if none are present in the input sources. The default can be controlled by the --enable-generate-build-notes configure option.

--help

Print a summary of the command-line options and exit.

--target-help

Print a summary of all target specific options and exit.

-I dir

Add directory dir to the search list for .include directives.

-J

Don’t warn about signed overflow.

-K

Issue warnings when difference tables altered for long displacements.

-L

--keep-locals

Keep (in the symbol table) local symbols. These symbols start with system-specific local label prefixes, typically ‘.L’ for ELF systems or ‘L’ for traditional a.out systems. See [Symbol Names](https://sourceware.org/binutils/docs/as/Symbol-Names.html" \l "Symbol-Names).

--listing-lhs-width=number

Set the maximum width, in words, of the output data column for an assembler listing to number.

--listing-lhs-width2=number

Set the maximum width, in words, of the output data column for continuation lines in an assembler listing to number.

--listing-rhs-width=number

Set the maximum width of an input source line, as displayed in a listing, to number bytes.

--listing-cont-lines=number

Set the maximum number of lines printed in a listing for a single line of input to number + 1.

--no-pad-sections

Stop the assembler for padding the ends of output sections to the alignment of that section. The default is to pad the sections, but this can waste space which might be needed on targets which have tight memory constraints.

-o objfile

Name the object-file output from as objfile.

-R

Fold the data section into the text section.

--hash-size=number

Set the default size of GAS’s hash tables to a prime number close to number. Increasing this value can reduce the length of time it takes the assembler to perform its tasks, at the expense of increasing the assembler’s memory requirements. Similarly reducing this value can reduce the memory requirements at the expense of speed.

--reduce-memory-overheads

This option reduces GAS’s memory requirements, at the expense of making the assembly processes slower. Currently this switch is a synonym for ‘--hash-size=4051’, but in the future it may have other effects as well.

--sectname-subst

Honor substitution sequences in section names. See [.section name](https://sourceware.org/binutils/docs/as/Section.html" \l "Section-Name-Substitutions).

--statistics

Print the maximum space (in bytes) and total time (in seconds) used by assembly.

--strip-local-absolute

Remove local absolute symbols from the outgoing symbol table.

-v

-version

Print the as version.

--version

Print the as version and exit.

-W

--no-warn

Suppress warning messages.

--fatal-warnings

Treat warnings as errors.

--warn

Don’t suppress warning messages or treat them as errors.

-w

Ignored.

-x

Ignored.

-Z

Generate an object file even after errors.

-- | files …

Standard input, or source files to assemble.

See [AArch64 Options](https://sourceware.org/binutils/docs/as/AArch64-Options.html" \l "AArch64-Options), for the options available when as is configured for the 64-bit mode of the ARM Architecture (AArch64).

See [Alpha Options](https://sourceware.org/binutils/docs/as/Alpha-Options.html" \l "Alpha-Options), for the options available when as is configured for an Alpha processor.

The following options are available when as is configured for an ARC processor.

-mcpu=cpu

This option selects the core processor variant.

-EB | -EL

Select either big-endian (-EB) or little-endian (-EL) output.

-mcode-density

Enable Code Density extenssion instructions.

The following options are available when as is configured for the ARM processor family.

-mcpu=processor[+extension…]

Specify which ARM processor variant is the target.

-march=architecture[+extension…]

Specify which ARM architecture variant is used by the target.

-mfpu=floating-point-format

Select which Floating Point architecture is the target.

-mfloat-abi=abi

Select which floating point ABI is in use.

-mthumb

Enable Thumb only instruction decoding.

-mapcs-32 | -mapcs-26 | -mapcs-float | -mapcs-reentrant

Select which procedure calling convention is in use.

-EB | -EL

Select either big-endian (-EB) or little-endian (-EL) output.

-mthumb-interwork

Specify that the code has been generated with interworking between Thumb and ARM code in mind.

-mccs

Turns on CodeComposer Studio assembly syntax compatibility mode.

-k

Specify that PIC code has been generated.

See [Blackfin Options](https://sourceware.org/binutils/docs/as/Blackfin-Options.html" \l "Blackfin-Options), for the options available when as is configured for the Blackfin processor family.

See the info pages for documentation of the CRIS-specific options.

See [C-SKY Options](https://sourceware.org/binutils/docs/as/C_002dSKY-Options.html" \l "C_002dSKY-Options), for the options available when as is configured for the C-SKY processor family.

The following options are available when as is configured for a D10V processor.

-O

Optimize output by parallelizing instructions.

The following options are available when as is configured for a D30V processor.

-O

Optimize output by parallelizing instructions.

-n

Warn when nops are generated.

-N

Warn when a nop after a 32-bit multiply instruction is generated.

The following options are available when as is configured for the Adapteva EPIPHANY series.

See [Epiphany Options](https://sourceware.org/binutils/docs/as/Epiphany-Options.html" \l "Epiphany-Options), for the options available when as is configured for an Epiphany processor.

See [i386-Options](https://sourceware.org/binutils/docs/as/i386_002dOptions.html" \l "i386_002dOptions), for the options available when as is configured for an i386 processor.

The following options are available when as is configured for the Ubicom IP2K series.

-mip2022ext

Specifies that the extended IP2022 instructions are allowed.

-mip2022

Restores the default behaviour, which restricts the permitted instructions to just the basic IP2022 ones.

The following options are available when as is configured for the Renesas M32C and M16C processors.

-m32c

Assemble M32C instructions.

-m16c

Assemble M16C instructions (the default).

-relax

Enable support for link-time relaxations.

-h-tick-hex

Support H’00 style hex constants in addition to 0x00 style.

The following options are available when as is configured for the Renesas M32R (formerly Mitsubishi M32R) series.

--m32rx

Specify which processor in the M32R family is the target. The default is normally the M32R, but this option changes it to the M32RX.

--warn-explicit-parallel-conflicts or --Wp

Produce warning messages when questionable parallel constructs are encountered.

--no-warn-explicit-parallel-conflicts or --Wnp

Do not produce warning messages when questionable parallel constructs are encountered.

The following options are available when as is configured for the Motorola 68000 series.

-l

Shorten references to undefined symbols, to one word instead of two.

-m68000 | -m68008 | -m68010 | -m68020 | -m68030

| -m68040 | -m68060 | -m68302 | -m68331 | -m68332

| -m68333 | -m68340 | -mcpu32 | -m5200

Specify what processor in the 68000 family is the target. The default is normally the 68020, but this can be changed at configuration time.

-m68881 | -m68882 | -mno-68881 | -mno-68882

The target machine does (or does not) have a floating-point coprocessor. The default is to assume a coprocessor for 68020, 68030, and cpu32. Although the basic 68000 is not compatible with the 68881, a combination of the two can be specified, since it’s possible to do emulation of the coprocessor instructions with the main processor.

-m68851 | -mno-68851

The target machine does (or does not) have a memory-management unit coprocessor. The default is to assume an MMU for 68020 and up.

See [Nios II Options](https://sourceware.org/binutils/docs/as/Nios-II-Options.html" \l "Nios-II-Options), for the options available when as is configured for an Altera Nios II processor.

For details about the PDP-11 machine dependent features options, see [PDP-11-Options](https://sourceware.org/binutils/docs/as/PDP_002d11_002dOptions.html" \l "PDP_002d11_002dOptions).

-mpic | -mno-pic

Generate position-independent (or position-dependent) code. The default is -mpic.

-mall

-mall-extensions

Enable all instruction set extensions. This is the default.

-mno-extensions

Disable all instruction set extensions.

-mextension | -mno-extension

Enable (or disable) a particular instruction set extension.

-mcpu

Enable the instruction set extensions supported by a particular CPU, and disable all other extensions.

-mmachine

Enable the instruction set extensions supported by a particular machine model, and disable all other extensions.

The following options are available when as is configured for a picoJava processor.

-mb

Generate “big endian” format output.

-ml

Generate “little endian” format output.

See [PRU Options](https://sourceware.org/binutils/docs/as/PRU-Options.html" \l "PRU-Options), for the options available when as is configured for a PRU processor.

The following options are available when as is configured for the Motorola 68HC11 or 68HC12 series.

-m68hc11 | -m68hc12 | -m68hcs12 | -mm9s12x | -mm9s12xg

Specify what processor is the target. The default is defined by the configuration option when building the assembler.

--xgate-ramoffset

Instruct the linker to offset RAM addresses from S12X address space into XGATE address space.

-mshort

Specify to use the 16-bit integer ABI.

-mlong

Specify to use the 32-bit integer ABI.

-mshort-double

Specify to use the 32-bit double ABI.

-mlong-double

Specify to use the 64-bit double ABI.

--force-long-branches

Relative branches are turned into absolute ones. This concerns conditional branches, unconditional branches and branches to a sub routine.

-S | --short-branches

Do not turn relative branches into absolute ones when the offset is out of range.

--strict-direct-mode

Do not turn the direct addressing mode into extended addressing mode when the instruction does not support direct addressing mode.

--print-insn-syntax

Print the syntax of instruction in case of error.

--print-opcodes

Print the list of instructions with syntax and then exit.

--generate-example

Print an example of instruction for each possible instruction and then exit. This option is only useful for testing as.

The following options are available when as is configured for the SPARC architecture:

-Av6 | -Av7 | -Av8 | -Asparclet | -Asparclite

-Av8plus | -Av8plusa | -Av9 | -Av9a

Explicitly select a variant of the SPARC architecture.

‘-Av8plus’ and ‘-Av8plusa’ select a 32 bit environment. ‘-Av9’ and ‘-Av9a’ select a 64 bit environment.

‘-Av8plusa’ and ‘-Av9a’ enable the SPARC V9 instruction set with UltraSPARC extensions.

-xarch=v8plus | -xarch=v8plusa

For compatibility with the Solaris v9 assembler. These options are equivalent to -Av8plus and -Av8plusa, respectively.

-bump

Warn when the assembler switches to another architecture.

The following options are available when as is configured for the ’c54x architecture.

-mfar-mode

Enable extended addressing mode. All addresses and relocations will assume extended addressing (usually 23 bits).

-mcpu=CPU\_VERSION

Sets the CPU version being compiled for.

-merrors-to-file FILENAME

Redirect error output to a file, for broken systems which don’t support such behaviour in the shell.

The following options are available when as is configured for a MIPS processor.

-G num

This option sets the largest size of an object that can be referenced implicitly with the gp register. It is only accepted for targets that use ECOFF format, such as a DECstation running Ultrix. The default value is 8.

-EB

Generate “big endian” format output.

-EL

Generate “little endian” format output.

-mips1

-mips2

-mips3

-mips4

-mips5

-mips32

-mips32r2

-mips32r3

-mips32r5

-mips32r6

-mips64

-mips64r2

-mips64r3

-mips64r5

-mips64r6

Generate code for a particular MIPS Instruction Set Architecture level. ‘-mips1’ is an alias for ‘-march=r3000’, ‘-mips2’ is an alias for ‘-march=r6000’, ‘-mips3’ is an alias for ‘-march=r4000’ and ‘-mips4’ is an alias for ‘-march=r8000’. ‘-mips5’, ‘-mips32’, ‘-mips32r2’, ‘-mips32r3’, ‘-mips32r5’, ‘-mips32r6’, ‘-mips64’, ‘-mips64r2’, ‘-mips64r3’, ‘-mips64r5’, and ‘-mips64r6’ correspond to generic MIPS V, MIPS32, MIPS32 Release 2, MIPS32 Release 3, MIPS32 Release 5, MIPS32 Release 6, MIPS64, MIPS64 Release 2, MIPS64 Release 3, MIPS64 Release 5, and MIPS64 Release 6 ISA processors, respectively.

-march=cpu

Generate code for a particular MIPS CPU.

-mtune=cpu

Schedule and tune for a particular MIPS CPU.

-mfix7000

-mno-fix7000

Cause nops to be inserted if the read of the destination register of an mfhi or mflo instruction occurs in the following two instructions.

-mfix-rm7000

-mno-fix-rm7000

Cause nops to be inserted if a dmult or dmultu instruction is followed by a load instruction.

-mfix-r5900

-mno-fix-r5900

Do not attempt to schedule the preceding instruction into the delay slot of a branch instruction placed at the end of a short loop of six instructions or fewer and always schedule a nopinstruction there instead. The short loop bug under certain conditions causes loops to execute only once or twice, due to a hardware bug in the R5900 chip.

-mdebug

-no-mdebug

Cause stabs-style debugging output to go into an ECOFF-style .mdebug section instead of the standard ELF .stabs sections.

-mpdr

-mno-pdr

Control generation of .pdr sections.

-mgp32

-mfp32

The register sizes are normally inferred from the ISA and ABI, but these flags force a certain group of registers to be treated as 32 bits wide at all times. ‘-mgp32’ controls the size of general-purpose registers and ‘-mfp32’ controls the size of floating-point registers.

-mgp64

-mfp64

The register sizes are normally inferred from the ISA and ABI, but these flags force a certain group of registers to be treated as 64 bits wide at all times. ‘-mgp64’ controls the size of general-purpose registers and ‘-mfp64’ controls the size of floating-point registers.

-mfpxx

The register sizes are normally inferred from the ISA and ABI, but using this flag in combination with ‘-mabi=32’ enables an ABI variant which will operate correctly with floating-point registers which are 32 or 64 bits wide.

-modd-spreg

-mno-odd-spreg

Enable use of floating-point operations on odd-numbered single-precision registers when supported by the ISA. ‘-mfpxx’ implies ‘-mno-odd-spreg’, otherwise the default is ‘-modd-spreg’.

-mips16

-no-mips16

Generate code for the MIPS 16 processor. This is equivalent to putting .module mips16 at the start of the assembly file. ‘-no-mips16’ turns off this option.

-mmips16e2

-mno-mips16e2

Enable the use of MIPS16e2 instructions in MIPS16 mode. This is equivalent to putting .module mips16e2 at the start of the assembly file. ‘-mno-mips16e2’ turns off this option.

-mmicromips

-mno-micromips

Generate code for the microMIPS processor. This is equivalent to putting .module micromips at the start of the assembly file. ‘-mno-micromips’ turns off this option. This is equivalent to putting .module nomicromips at the start of the assembly file.

-msmartmips

-mno-smartmips

Enables the SmartMIPS extension to the MIPS32 instruction set. This is equivalent to putting .module smartmips at the start of the assembly file. ‘-mno-smartmips’ turns off this option.

-mips3d

-no-mips3d

Generate code for the MIPS-3D Application Specific Extension. This tells the assembler to accept MIPS-3D instructions. ‘-no-mips3d’ turns off this option.

-mdmx

-no-mdmx

Generate code for the MDMX Application Specific Extension. This tells the assembler to accept MDMX instructions. ‘-no-mdmx’ turns off this option.

-mdsp

-mno-dsp

Generate code for the DSP Release 1 Application Specific Extension. This tells the assembler to accept DSP Release 1 instructions. ‘-mno-dsp’ turns off this option.

-mdspr2

-mno-dspr2

Generate code for the DSP Release 2 Application Specific Extension. This option implies ‘-mdsp’. This tells the assembler to accept DSP Release 2 instructions. ‘-mno-dspr2’ turns off this option.

-mdspr3

-mno-dspr3

Generate code for the DSP Release 3 Application Specific Extension. This option implies ‘-mdsp’ and ‘-mdspr2’. This tells the assembler to accept DSP Release 3 instructions. ‘-mno-dspr3’ turns off this option.

-mmsa

-mno-msa

Generate code for the MIPS SIMD Architecture Extension. This tells the assembler to accept MSA instructions. ‘-mno-msa’ turns off this option.

-mxpa

-mno-xpa

Generate code for the MIPS eXtended Physical Address (XPA) Extension. This tells the assembler to accept XPA instructions. ‘-mno-xpa’ turns off this option.

-mmt

-mno-mt

Generate code for the MT Application Specific Extension. This tells the assembler to accept MT instructions. ‘-mno-mt’ turns off this option.

-mmcu

-mno-mcu

Generate code for the MCU Application Specific Extension. This tells the assembler to accept MCU instructions. ‘-mno-mcu’ turns off this option.

-mcrc

-mno-crc

Generate code for the MIPS cyclic redundancy check (CRC) Application Specific Extension. This tells the assembler to accept CRC instructions. ‘-mno-crc’ turns off this option.

-mginv

-mno-ginv

Generate code for the Global INValidate (GINV) Application Specific Extension. This tells the assembler to accept GINV instructions. ‘-mno-ginv’ turns off this option.

-mloongson-mmi

-mno-loongson-mmi

Generate code for the Loongson MultiMedia extensions Instructions (MMI) Application Specific Extension. This tells the assembler to accept MMI instructions. ‘-mno-loongson-mmi’ turns off this option.

-mloongson-cam

-mno-loongson-cam

Generate code for the Loongson Content Address Memory (CAM) instructions. This tells the assembler to accept Loongson CAM instructions. ‘-mno-loongson-cam’ turns off this option.

-mloongson-ext

-mno-loongson-ext

Generate code for the Loongson EXTensions (EXT) instructions. This tells the assembler to accept Loongson EXT instructions. ‘-mno-loongson-ext’ turns off this option.

-mloongson-ext2

-mno-loongson-ext2

Generate code for the Loongson EXTensions R2 (EXT2) instructions. This option implies ‘-mloongson-ext’. This tells the assembler to accept Loongson EXT2 instructions. ‘-mno-loongson-ext2’ turns off this option.

-minsn32

-mno-insn32

Only use 32-bit instruction encodings when generating code for the microMIPS processor. This option inhibits the use of any 16-bit instructions. This is equivalent to putting .set insn32at the start of the assembly file. ‘-mno-insn32’ turns off this option. This is equivalent to putting .set noinsn32 at the start of the assembly file. By default ‘-mno-insn32’ is selected, allowing all instructions to be used.

--construct-floats

--no-construct-floats

The ‘--no-construct-floats’ option disables the construction of double width floating point constants by loading the two halves of the value into the two single width floating point registers that make up the double width register. By default ‘--construct-floats’ is selected, allowing construction of these floating point constants.

--relax-branch

--no-relax-branch

The ‘--relax-branch’ option enables the relaxation of out-of-range branches. By default ‘--no-relax-branch’ is selected, causing any out-of-range branches to produce an error.

-mignore-branch-isa

-mno-ignore-branch-isa

Ignore branch checks for invalid transitions between ISA modes. The semantics of branches does not provide for an ISA mode switch, so in most cases the ISA mode a branch has been encoded for has to be the same as the ISA mode of the branch’s target label. Therefore GAS has checks implemented that verify in branch assembly that the two ISA modes match. ‘-mignore-branch-isa’ disables these checks. By default ‘-mno-ignore-branch-isa’ is selected, causing any invalid branch requiring a transition between ISA modes to produce an error.

-mnan=encoding

Select between the IEEE 754-2008 (-mnan=2008) or the legacy (-mnan=legacy) NaN encoding format. The latter is the default.

--emulation=name

This option was formerly used to switch between ELF and ECOFF output on targets like IRIX 5 that supported both. MIPS ECOFF support was removed in GAS 2.24, so the option now serves little purpose. It is retained for backwards compatibility.

The available configuration names are: ‘mipself’, ‘mipslelf’ and ‘mipsbelf’. Choosing ‘mipself’ now has no effect, since the output is always ELF. ‘mipslelf’ and ‘mipsbelf’ select little- and big-endian output respectively, but ‘-EL’ and ‘-EB’ are now the preferred options instead.

-nocpp

as ignores this option. It is accepted for compatibility with the native tools.

--trap

--no-trap

--break

--no-break

Control how to deal with multiplication overflow and division by zero. ‘--trap’ or ‘--no-break’ (which are synonyms) take a trap exception (and only work for Instruction Set Architecture level 2 and higher); ‘--break’ or ‘--no-trap’ (also synonyms, and the default) take a break exception.

-n

When this option is used, as will issue a warning every time it generates a nop instruction from a macro.

The following options are available when as is configured for an MCore processor.

-jsri2bsr

-nojsri2bsr

Enable or disable the JSRI to BSR transformation. By default this is enabled. The command-line option ‘-nojsri2bsr’ can be used to disable it.

-sifilter

-nosifilter

Enable or disable the silicon filter behaviour. By default this is disabled. The default can be overridden by the ‘-sifilter’ command-line option.

-relax

Alter jump instructions for long displacements.

-mcpu=[210|340]

Select the cpu type on the target hardware. This controls which instructions can be assembled.

-EB

Assemble for a big endian target.

-EL

Assemble for a little endian target.

See [Meta Options](https://sourceware.org/binutils/docs/as/Meta-Options.html" \l "Meta-Options), for the options available when as is configured for a Meta processor.

See the info pages for documentation of the MMIX-specific options.

See [NDS32 Options](https://sourceware.org/binutils/docs/as/NDS32-Options.html" \l "NDS32-Options), for the options available when as is configured for a NDS32 processor.

See [PowerPC-Opts](https://sourceware.org/binutils/docs/as/PowerPC_002dOpts.html" \l "PowerPC_002dOpts), for the options available when as is configured for a PowerPC processor.

See [RISC-V-Options](https://sourceware.org/binutils/docs/as/RISC_002dV_002dOptions.html" \l "RISC_002dV_002dOptions), for the options available when as is configured for a RISC-V processor.

See the info pages for documentation of the RX-specific options.

The following options are available when as is configured for the s390 processor family.

-m31

-m64

Select the word size, either 31/32 bits or 64 bits.

-mesa

-mzarch

Select the architecture mode, either the Enterprise System Architecture (esa) or the z/Architecture mode (zarch).

-march=processor

Specify which s390 processor variant is the target, ‘g5’ (or ‘arch3’), ‘g6’, ‘z900’ (or ‘arch5’), ‘z990’ (or ‘arch6’), ‘z9-109’, ‘z9-ec’ (or ‘arch7’), ‘z10’ (or ‘arch8’), ‘z196’ (or ‘arch9’), ‘zEC12’ (or ‘arch10’), ‘z13’ (or ‘arch11’), or ‘z14’ (or ‘arch12’).

-mregnames

-mno-regnames

Allow or disallow symbolic names for registers.

-mwarn-areg-zero

Warn whenever the operand for a base or index register has been specified but evaluates to zero.

See [TIC6X Options](https://sourceware.org/binutils/docs/as/TIC6X-Options.html" \l "TIC6X-Options), for the options available when as is configured for a TMS320C6000 processor.

See [TILE-Gx Options](https://sourceware.org/binutils/docs/as/TILE_002dGx-Options.html" \l "TILE_002dGx-Options), for the options available when as is configured for a TILE-Gx processor.

See [Visium Options](https://sourceware.org/binutils/docs/as/Visium-Options.html" \l "Visium-Options), for the options available when as is configured for a Visium processor.

See [Xtensa Options](https://sourceware.org/binutils/docs/as/Xtensa-Options.html" \l "Xtensa-Options), for the options available when as is configured for an Xtensa processor.

The following options are available when as is configured for a Z80 family processor.

-z80

Assemble for Z80 processor.

-r800

Assemble for R800 processor.

-ignore-undocumented-instructions

-Wnud

Assemble undocumented Z80 instructions that also work on R800 without warning.

-ignore-unportable-instructions

-Wnup

Assemble all undocumented Z80 instructions without warning.

-warn-undocumented-instructions

-Wud

Issue a warning for undocumented Z80 instructions that also work on R800.

-warn-unportable-instructions

-Wup

Issue a warning for undocumented Z80 instructions that do not work on R800.

-forbid-undocumented-instructions

-Fud

Treat all undocumented instructions as errors.

-forbid-unportable-instructions

-Fup

Treat undocumented Z80 instructions that do not work on R800 as errors.

## 2 Command-Line Options

This chapter describes command-line options available in all versions of the GNU assembler; see [Machine Dependencies](https://sourceware.org/binutils/docs/as/Machine-Dependencies.html" \l "Machine-Dependencies), for options specific to particular machine architectures.

If you are invoking as via the GNU C compiler, you can use the ‘-Wa’ option to pass arguments through to the assembler. The assembler arguments must be separated from each other (and the ‘-Wa’) by commas. For example:

gcc -c -g -O -Wa,-alh,-L file.c

This passes two options to the assembler: ‘-alh’ (emit a listing to standard output with high-level and assembly source) and ‘-L’ (retain local symbols in the symbol table).

Usually you do not need to use this ‘-Wa’ mechanism, since many compiler command-line options are automatically passed to the assembler by the compiler. (You can call the GNU compiler driver with the ‘-v’ option to see precisely what options it passes to each compilation pass, including the assembler.)

|  |  |  |
| --- | --- | --- |
| • [a](https://sourceware.org/binutils/docs/as/a.html" \l "a): |  | -a[cdghlns] enable listings |
| • [alternate](https://sourceware.org/binutils/docs/as/alternate.html" \l "alternate): |  | –alternate enable alternate macro syntax |
| • [D](https://sourceware.org/binutils/docs/as/D.html" \l "D): |  | -D for compatibility |
| • [f](https://sourceware.org/binutils/docs/as/f.html" \l "f): |  | -f to work faster |
| • [I](https://sourceware.org/binutils/docs/as/I.html" \l "I): |  | -I for .include search path |
| • [K](https://sourceware.org/binutils/docs/as/K.html" \l "K): |  | -K for difference tables |
|  | | |
| • [L](https://sourceware.org/binutils/docs/as/L.html" \l "L): |  | -L to retain local symbols |
| • [listing](https://sourceware.org/binutils/docs/as/listing.html" \l "listing): |  | –listing-XXX to configure listing output |
| • [M](https://sourceware.org/binutils/docs/as/M.html" \l "M): |  | -M or –mri to assemble in MRI compatibility mode |
| • [MD](https://sourceware.org/binutils/docs/as/MD.html" \l "MD): |  | –MD for dependency tracking |
| • [no-pad-sections](https://sourceware.org/binutils/docs/as/no_002dpad_002dsections.html" \l "no_002dpad_002dsections): |  | –no-pad-sections to stop section padding |
| • [o](https://sourceware.org/binutils/docs/as/o.html" \l "o): |  | -o to name the object file |
| • [R](https://sourceware.org/binutils/docs/as/R.html" \l "R): |  | -R to join data and text sections |
| • [statistics](https://sourceware.org/binutils/docs/as/statistics.html" \l "statistics): |  | –statistics to see statistics about assembly |
| • [traditional-format](https://sourceware.org/binutils/docs/as/traditional_002dformat.html" \l "traditional_002dformat): |  | –traditional-format for compatible output |
| • [v](https://sourceware.org/binutils/docs/as/v.html" \l "v): |  | -v to announce version |
| • [W](https://sourceware.org/binutils/docs/as/W.html" \l "W): |  | -W, –no-warn, –warn, –fatal-warnings to control warnings |
| • [Z](https://sourceware.org/binutils/docs/as/Z.html" \l "Z): |  | -Z to make object file even after errors |

## 3 Syntax

This chapter describes the machine-independent syntax allowed in a source file. as syntax is similar to what many other assemblers use; it is inspired by the BSD 4.2 assembler, except that as does not assemble Vax bit-fields.

### 3.1 Preprocessing

The as internal preprocessor:

* adjusts and removes extra whitespace. It leaves one space or tab before the keywords on a line, and turns any other whitespace on the line into a single space.
* removes all comments, replacing them with a single space, or an appropriate number of newlines.
* converts character constants into the appropriate numeric values.

It does not do macro processing, include file handling, or anything else you may get from your C compiler’s preprocessor. You can do include file processing with the .include directive (see [.include](https://sourceware.org/binutils/docs/as/Include.html" \l "Include)). You can use the GNU C compiler driver to get other “CPP” style preprocessing by giving the input file a ‘.S’ suffix. See [Options Controlling the Kind of Output](https://sourceware.org/binutils/docs/gcc info/Overall-Options.html" \l "Overall-Options) in Using GNU CC.

Excess whitespace, comments, and character constants cannot be used in the portions of the input text that are not preprocessed.

If the first line of an input file is #NO\_APP or if you use the ‘-f’ option, whitespace and comments are not removed from the input file. Within an input file, you can ask for whitespace and comment removal in specific portions of the by putting a line that says #APP before the text that may contain whitespace or comments, and putting a line that says #NO\_APP after this text. This feature is mainly intend to support asm statements in compilers whose output is otherwise free of comments and whitespace.

### 3.2 Whitespace

Whitespace is one or more blanks or tabs, in any order. Whitespace is used to separate symbols, and to make programs neater for people to read. Unless within character constants (see [Character Constants](https://sourceware.org/binutils/docs/as/Characters.html" \l "Characters)), any whitespace means the same as exactly one space.

### 3.3 Comments

There are two ways of rendering comments to as. In both cases the comment is equivalent to one space.

Anything from ‘/\*’ through the next ‘\*/’ is a comment. This means you may not nest these comments.

/\*

The only way to include a newline ('\n') in a comment

is to use this sort of comment.

\*/

/\* This sort of comment does not nest. \*/

Anything from a line comment character up to the next newline is considered a comment and is ignored. The line comment character is target specific, and some targets multiple comment characters. Some targets also have line comment characters that only work if they are the first character on a line. Some targets use a sequence of two characters to introduce a line comment. Some targets can also change their line comment characters depending upon command-line options that have been used. For more details see the Syntax section in the documentation for individual targets.

If the line comment character is the hash sign (‘#’) then it still has the special ability to enable and disable preprocessing (see [Preprocessing](https://sourceware.org/binutils/docs/as/Preprocessing.html" \l "Preprocessing)) and to specify logical line numbers:

To be compatible with past assemblers, lines that begin with ‘#’ have a special interpretation. Following the ‘#’ should be an absolute expression (see [Expressions](https://sourceware.org/binutils/docs/as/Expressions.html" \l "Expressions)): the logical line number of the next line. Then a string (see [Strings](https://sourceware.org/binutils/docs/as/Strings.html" \l "Strings)) is allowed: if present it is a new logical file name. The rest of the line, if any, should be whitespace.

If the first non-whitespace characters on the line are not numeric, the line is ignored. (Just like a comment.)

# This is an ordinary comment.

# 42-6 "new\_file\_name" # New logical file name

# This is logical line # 36.

This feature is deprecated, and may disappear from future versions of as.

### 3.4 Symbols

A symbol is one or more characters chosen from the set of all letters (both upper and lower case), digits and the three characters ‘\_.$’. On most machines, you can also use $ in symbol names; exceptions are noted in [Machine Dependencies](https://sourceware.org/binutils/docs/as/Machine-Dependencies.html" \l "Machine-Dependencies). No symbol may begin with a digit. Case is significant. There is no length limit; all characters are significant. Multibyte characters are supported. Symbols are delimited by characters not in that set, or by the beginning of a file (since the source program must end with a newline, the end of a file is not a possible symbol delimiter). See [Symbols](https://sourceware.org/binutils/docs/as/Symbols.html" \l "Symbols).

Symbol names may also be enclosed in double quote " characters. In such cases any characters are allowed, except for the NUL character. If a double quote character is to be included in the symbol name it must be preceeded by a backslash \ character.

### 3.5 Statements

A statement ends at a newline character (‘\n’) or a line separator character. The line separator character is target specific and described in the Syntax section of each target’s documentation. Not all targets support a line separator character. The newline or line separator character is considered to be part of the preceding statement. Newlines and separators within character constants are an exception: they do not end statements.

It is an error to end any statement with end-of-file: the last character of any input file should be a newline.

An empty statement is allowed, and may include whitespace. It is ignored.

A statement begins with zero or more labels, optionally followed by a key symbol which determines what kind of statement it is. The key symbol determines the syntax of the rest of the statement. If the symbol begins with a dot ‘.’ then the statement is an assembler directive: typically valid for any computer. If the symbol begins with a letter the statement is an assembly language instruction: it assembles into a machine language instruction. Different versions of as for different computers recognize different instructions. In fact, the same symbol may represent a different instruction in a different computer’s assembly language.

A label is a symbol immediately followed by a colon (:). Whitespace before a label or after a colon is permitted, but you may not have whitespace between a label’s symbol and its colon. See [Labels](https://sourceware.org/binutils/docs/as/Labels.html" \l "Labels).

For HPPA targets, labels need not be immediately followed by a colon, but the definition of a label must begin in column zero. This also implies that only one label may be defined on each line.

label: .directive followed by something

another\_label: # This is an empty statement.

instruction operand\_1, operand\_2, …

### 3.6 Constants

A constant is a number, written so that its value is known by inspection, without knowing any context. Like this:

.byte 74, 0112, 092, 0x4A, 0X4a, 'J, '\J # All the same value.

.ascii "Ring the bell\7" # A string constant.

.octa 0x123456789abcdef0123456789ABCDEF0 # A bignum.

.float 0f-314159265358979323846264338327\

95028841971.693993751E-40 # - pi, a flonum.

#### 3.6.1 Character Constants

There are two kinds of character constants. A character stands for one character in one byte and its value may be used in numeric expressions. String constants (properly called string literals) are potentially many bytes and their values may not be used in arithmetic expressions.

#### 3.6.1.1 Strings

A string is written between double-quotes. It may contain double-quotes or null characters. The way to get special characters into a string is to escape these characters: precede them with a backslash ‘\’ character. For example ‘\\’ represents one backslash: the first \ is an escape which tells as to interpret the second character literally as a backslash (which prevents as from recognizing the second \ as an escape character). The complete list of escapes follows.

\b

Mnemonic for backspace; for ASCII this is octal code 010.

backslash-f

Mnemonic for FormFeed; for ASCII this is octal code 014.

\n

Mnemonic for newline; for ASCII this is octal code 012.

\r

Mnemonic for carriage-Return; for ASCII this is octal code 015.

\t

Mnemonic for horizontal Tab; for ASCII this is octal code 011.

\ digit digit digit

An octal character code. The numeric code is 3 octal digits. For compatibility with other Unix systems, 8 and 9 are accepted as digits: for example, \008 has the value 010, and \009 the value 011.

\x hex-digits...

A hex character code. All trailing hex digits are combined. Either upper or lower case x works.

\\

Represents one ‘\’ character.

\"

Represents one ‘"’ character. Needed in strings to represent this character, because an unescaped ‘"’ would end the string.

\ anything-else

Any other character when escaped by \ gives a warning, but assembles as if the ‘\’ was not present. The idea is that if you used an escape sequence you clearly didn’t want the literal interpretation of the following character. However as has no other interpretation, so as knows it is giving you the wrong code and warns you of the fact.

Which characters are escapable, and what those escapes represent, varies widely among assemblers. The current set is what we think the BSD 4.2 assembler recognizes, and is a subset of what most C compilers recognize. If you are in doubt, do not use an escape sequence.

#### 3.6.1.2 Characters

A single character may be written as a single quote immediately followed by that character. Some backslash escapes apply to characters, \b, \f, \n, \r, \t, and \" with the same meaning as for strings, plus \' for a single quote. So if you want to write the character backslash, you must write '\\ where the first \ escapes the second \. As you can see, the quote is an acute accent, not a grave accent. A newline immediately following an acute accent is taken as a literal character and does not count as the end of a statement. The value of a character constant in a numeric expression is the machine’s byte-wide code for that character. as assumes your character code is ASCII: 'A means 65, 'B means 66, and so on.

#### 3.6.2 Number Constants

as distinguishes three kinds of numbers according to how they are stored in the target machine. Integers are numbers that would fit into an int in the C language. Bignums are integers, but they are stored in more than 32 bits. Flonums are floating point numbers, described below.

#### 3.6.2.1 Integers

A binary integer is ‘0b’ or ‘0B’ followed by zero or more of the binary digits ‘01’.

An octal integer is ‘0’ followed by zero or more of the octal digits (‘01234567’).

A decimal integer starts with a non-zero digit followed by zero or more digits (‘0123456789’).

A hexadecimal integer is ‘0x’ or ‘0X’ followed by one or more hexadecimal digits chosen from ‘0123456789abcdefABCDEF’.

Integers have the usual values. To denote a negative integer, use the prefix operator ‘-’ discussed under expressions (see [Prefix Operators](https://sourceware.org/binutils/docs/as/Prefix-Ops.html" \l "Prefix-Ops)).

#### 3.6.2.2 Bignums

A bignum has the same syntax and semantics as an integer except that the number (or its negative) takes more than 32 bits to represent in binary. The distinction is made because in some places integers are permitted while bignums are not.

#### 3.6.2.3 Flonums

A flonum represents a floating point number. The translation is indirect: a decimal floating point number from the text is converted by as to a generic binary floating point number of more than sufficient precision. This generic floating point number is converted to a particular computer’s floating point format (or formats) by a portion of as specialized to that computer.

A flonum is written by writing (in order)

* The digit ‘0’. (‘0’ is optional on the HPPA.)
* A letter, to tell as the rest of the number is a flonum. e is recommended. Case is not important.

On the H8/300 and Renesas / SuperH SH architectures, the letter must be one of the letters ‘DFPRSX’ (in upper or lower case).

On the ARC, the letter must be one of the letters ‘DFRS’ (in upper or lower case).

On the HPPA architecture, the letter must be ‘E’ (upper case only).

* An optional sign: either ‘+’ or ‘-’.
* An optional integer part: zero or more decimal digits.
* An optional fractional part: ‘.’ followed by zero or more decimal digits.
* An optional exponent, consisting of:
  + An ‘E’ or ‘e’.
  + Optional sign: either ‘+’ or ‘-’.
  + One or more decimal digits.

At least one of the integer part or the fractional part must be present. The floating point number has the usual base-10 value.

as does all processing using integers. Flonums are computed independently of any floating point hardware in the computer running as.