NAME

AS − the portable GNU assembler.

SYNOPSIS

as [**−a**[**cdhlns**][=*file*]] [**−D**] [**−−defsym** *sym=val*]
[**−f**] [**−−gstabs**] [**−−gdwarf2**] [**−−help**] [**−I** *dir*]
[**−J**] [**−K**] [**−L**]
[**−−listing−lhs−width**=*NUM*] [**−−listing−lhs−width2**=*NUM*]
[**−−listing−rhs−width**=*NUM*] [**−−listing−cont−lines**=*NUM*]
[**−−keep−locals**] [**−o** *objfile*] [**−R**] [**−−statistics**] [**−v**]
[**−version**] [**−−version**] [**−W**] [**−−warn**] [**−−fatal−warnings**]
[**−w**] [**−x**] [**−Z**] [**−−target−help**] [*target-options*]
[**−−**|*files* ...]

Target Alpha options:
[**−m***cpu*]
[**−mdebug** | **−no−mdebug**]
[**−relax**] [**−g**] [**−G***size*]
[**−F**] [**−32addr**]

Target ARC options:
[**−marc[5|6|7|8]**]
[**−EB**|**−EL**]

Target ARM options:
[**−mcpu**=*processor*[+*extension*...]]
[**−march**=*architecture*[+*extension*...]]
[**−mfpu**=*floating-point-fromat*]
[**−mthumb**]
[**−EB**|**−EL**]
[**−mapcs−32**|**−mapcs−26**|**−mapcs−float**|
−mapcs−reentrant]
[**−mthumb−interwork**] [**−moabi**] [**−k**]

Target CRIS options:
[**−−underscore** | **−−no−underscore**]
[**−−pic**] [**−N**]
[**−−emulation=criself** | **−−emulation=crisaout**]

Target D10V options:
[**−O**]

Target D30V options:
[**−O**|**−n**|**−N**]

Target i386 options:
[**−−32**|**−−64**]

Target i960 options:
[**−ACA**|**−ACA_A**|**−ACB**|**−ACC**|**−AKA**|**−AKB**|
−AKC|**−AMC**]
[**−b**] [**−no−relax**]

Target IP2K options:
[**−mip2022**|**−mip2022ext**]

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**]
[**−mshort**|**−mlong**]
[**−mshort−double**|**−mlong−double**]
[**−−force−long−branchs**] [**−−short−branchs**]
[**−−strict−direct−mode**] [**−−print−insn−syntax**]
[**−−print−opcodes**] [**−−generate−example**]

Target MCORE options:
[**−jsri2bsr**] [**−sifilter**] [**−relax**]
[**−mcpu=[210**|**340]**]

Target MIPS options:
[**−nocpp**] [**−EL**] [**−EB**] [**−n**] [**−O**[*optimization level*]]
[**−g**[*debug level*]] [**−G** *num*] [**−KPIC**] [**−call_shared**]
[**−non_shared**] [**−xgot**] [**−−membedded−pic**]
[**−mabi**=*ABI*] [**−32**] [**−n32**] [**−64**] [**−mfp32**] [**−mgp32**]
[**−march**=*CPU*] [**−mtune**=*CPU*] [**−mips1**] [**−mips2**]
[**−mips3**] [**−mips4**] [**−mips5**] [**−mips32**] [**−mips32r2**]
[**−mips64**]
[**−construct−fbats**] [**−no−construct−fbats**]
[**−trap**] [**−no−break**] [**−break**] [**−no−trap**]
[**−mfix7000**] [**−mno−fix7000**]
[**−mips16**] [**−no−mips16**]
[**−mips3d**] [**−no−mips3d**]
[**−mdmx**] [**−no−mdmx**]
[**−mdebug**] [**−no−mdebug**]

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 PDP11 options:
[**−mpic**|**−mno−pic**] [**−mall**] [**−mno−extensions**]
[**−m***extension*|**−mno−***extension*]
[**−m***cpu*] [**−m***machine*]

Target picoJava options:
[**−mb**|**−me**]

Target PowerPC options:
[**−mpwrx**|**−mpwr2**|**−mpwr**|**−m601**|**−mppc**|**−mppc32**|**−m603**|**−m604**|
−m403|**−m405**|**−mppc64**|**−m620**|**−mppc64bridge**|**−mbooke**|
−mbooke32|**−mbooke64**]
[**−mcom**|**−many**|**−maltivec**] [**−memb**]
[**−mregnames**|**−mno−regnames**]
[**−mrelocatable**|**−mrelocatable−lib**]
[**−mlittle**|**−mlittle−endian**|**−mbig**|**−mbig−endian**]
[**−msolaris**|**−mno−solaris**]

Target SPARC options:
[**−Av6**|**−Av7**|**−Av8**|**−Asparclet**|**−Asparclite**
−Av8plus|**−Av8plusa**|**−Av9**|**−Av9a**]
[**−xarch=v8plus**|**−xarch=v8plusa**] [**−bump**]
[**−32**|**−64**]

Target TIC54X options:

[**−mcpu=54[123589]**|**−mcpu=54[56]lp**] [**−mfar−mode**|**−mf**]
[**−merrors−to−file** *<file name>*|**−me** *<file name>*]

Target Xtensa options:
[**−−[no−]density**] [**−−[no−]relax**] [**−−[no−]generics**]
[**−−[no−]text−section−literals**]
[**−−[no−]target−align**] [**−−[no−]longcalls**]

DESCRIPTION

GNU **as** is really a family of assemblers. If you use (or have used) the GNU assembler on one architecture, you should find a fairly similar environment when you use it on another architecture. Each version has much in common with the others, including object file formats, most assembler directives (often called *pseudo-ops*) and assembler syntax.

as is primarily intended to assemble the output of the GNU C compiler for use by the linker . Nevertheless, we've tried to make **as** assemble correctly everything that other assemblers for the same machine would assemble. Any exceptions are documented explicitly. This doesn't mean **as** always uses the same syntax as another assembler for the same architecture; for example, we know of several incompatible versions of 680x0 assembly language syntax.

Each time you run **as** it assembles exactly one source program. The source program is made up of one or more files. (The standard input is also a file.)

You give **as** a command line that has zero or more input file names. The input files are read (from left file name to right). A command line argument (in any position) that has no special meaning is taken to be an input file name.

If you give **as** no file names it attempts to read one input file from the **as** standard input, which is normally your terminal. You may have to type **ctl-D** to tell **as** there is no more program to assemble.

Use **−−** if you need to explicitly name the standard input file in your command line.

If the source is empty, **as** produces a small, empty object file.

as may write warnings and error messages to the standard error file (usually your terminal). This should not happen when a compiler runs **as** automatically. Warnings report an assumption made so that **as** could keep assembling a flawed program; errors report a grave problem that stops the assembly.

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.)

OPTIONS

−a[cdhlmns]

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

−ac
omit false conditionals

−ad
omit debugging directives

−ah
include high-level source

−al include assembly

−am
include macro expansions

−an
omit forms processing

−as include symbols

=fi le
set the name of the listing fi le

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

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

−−defsym *sym=value*
Defi ne the symbol *sym* to be *value* before assembling the input fi le. *value* must be an integer constant. As in C, a leading **0x** indicates a hexadecimal value, and a leading **0** indicates an octal value.

−f ‘‘fast’’−−−skip whitespace and comment preprocessing (assume source is compiler output).

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

−−gdwarf2
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.

−−help
Print a summary of the command line options and exit.

−−target−help
Print a summary of all target specifi c options and exit.

−I *dir*
Add directory *dir* to the search list for `.include` directives.

−J Don’t warn about signed overflow.

−K This option is accepted but has no effect on the TARGET family.

−L
−−keep−locals
Keep (in the symbol table) local symbols. On traditional a.out systems these start with **L**, but different systems have different local label prefi xes.

−−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.

−o *objfi le*
Name the object-fi le output from **as** *objfi le*.

−R Fold the data section into the text section.

−−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.

−− | *file* ...
Standard input, or source files to assemble.

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

−marc[5|6|7|8]
This option selects the core processor variant.

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

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.

−mthumb
Enable Thumb only instruction decoding.

−mapcs−32 | −mapcs−26 | −mapcs−fbat | −mapcs−reentrant | −moabi
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.

−k Specify that PIC code has been generated.

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

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 Intel 80960 processor.

-ACA | -ACA_A | -ACB | -ACC | -AKA | -AKB | -AKC | -AMC
Specify which variant of the 960 architecture is the target.

-b Add code to collect statistics about branches taken.

-no-relax
Do not alter compare-and-branch instructions for long displacements; error if necessary.

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 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.

For details about the PDP-11 machine dependent features options, see `@ref{PDP-11-Options}`.

-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.

−m*extension* | **−mno−***extension*
Enable (or disable) a particular instruction set extension.

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

−m*machine*
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.

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

−m68hc11 | −m68hc12 | −m68hcs12
Specify what processor is the target. The default is defined by the configuration option when building the assembler.

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

−S | −−short−branchs
Do not turn relative branchs 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
−mips64
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**, and **−mips64** correspond to generic **MIPS V**, **MIPS32**, **MIPS32 Release 2**, and **MIPS64** 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.

−mdebug
−no−mdebug
Cause stabs-style debugging output to go into an ECOFF-style .mdebug section instead of the standard ELF .stabs 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.

−mips16
−no−mips16
Generate code for the MIPS 16 processor. This is equivalent to putting `.set mips16` at the start of the assembly file. **−no−mips16** 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.

−−construct−fbats
−−no−construct−fbats
The **−−no−construct−fbats** 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−fbats** is selected, allowing construction of these floating point constants.

−−emulation=*name*
This option causes **as** to emulate **as** configured for some other target, in all respects, including output format (choosing between ELF and ECOFF only), handling of pseudo-opcodes which may generate debugging information or store symbol table information, and default endianness. The available configuration names are: **mipsecoff**, **mipself**, **mipslecoff**, **mipsbecoff**, **mipslelf**, **mipsbelf**. The first two do not alter the default endianness from that of the primary target for which the assembler was configured; the others change the default to little− or big-endian as indicated by the **b** or **l** in the name. Using **−EB** or **−EL** will override the endianness selection in any case.

This option is currently supported only when the primary target **as** is configured for is a MIPS ELF or ECOFF target. Furthermore, the primary target or others specified with **−−enable−targets=...** at configuration time must include support for the other format, if both are to be available. For example, the Irix 5 configuration includes support for both.

Eventually, this option will support more configurations, with more fine-grained control over the assembler's behavior, and will be supported for more processors.

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

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

−−density | −−no−density
Enable or disable use of instructions from the Xtensa code density option. This is enabled by default when the Xtensa processor supports the code density option.

−−relax | −−no−relax
Enable or disable instruction relaxation. This is enabled by default. Note: In the current implementation, these options also control whether assembler optimizations are performed, making these options equivalent to **−−generics** and **−−no−generics**.

−−generics | −−no−generics
Enable or disable all assembler transformations of Xtensa instructions. The default is **−−generics**; **−−no−generics** should be used only in the rare cases when the instructions must be exactly as specified in the assembly source.

−−text−section−literals | −−no−text−section−literals
With **−−text−section−literals**, literal pools are interspersed in the text section. The default is **−−no−text−section−literals**, which places literals in a separate section in the output file.

−−target−align | −−no−target−align
Enable or disable automatic alignment to reduce branch penalties at the expense of some code density. The default is **−−target−align**.

−−longcalls | −−no−longcalls
Enable or disable transformation of call instructions to allow calls across a greater range of addresses. The default is **−−no−longcalls**.

SEE ALSO

gcc (1), *ld* (1), and the Info entries for *binutils* and *ld*.

COPYRIGHT

no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts. A copy of the license is included in the section entitled ‘‘GNU Free Documentation License’’.