



PathScale Compiler Suite

Quick Reference Card

<http://www.pathscale.com>

| Man Page | Sec | Description |
|-------------------|-----|---------------------------------|
| pathf95 | 1 | Fortran Compiler |
| pathcc | 1 | C Compiler |
| pathCC | 1 | C++ Compiler |
| pathopt2 | 1 | Application tuning utility |
| pathdb | 1 | Debugger |
| pathhow-compiled | 1 | Display options used utility |
| explain | 1 | Fortran error message utility |
| assign | 1 | Customize Fortran I/O |
| fsymlist | 1 | List symbols from Fortran lib |
| compiler.defaults | 5 | Default options definition file |
| eko | 7 | List of all compiler options |

| Extension | Driver Behavior |
|-----------|---|
| .c | C source, preprocessor |
| .C | C++ source, preprocessor |
| .cc | |
| .cpp | |
| .cxx | |
| .f | Fortran source, fixed format, no preprocessor |
| .f90 | Fortran source, free format, no preprocessor |
| .f95 | |
| .F | Fortran source, fixed formation, preprocessor |
| .F90 | Fortran source, freeform, preprocessor |
| .F95 | |
| .i | Preprocessed C source file |
| .ii | Preprocessed C++ source file |
| .s | Assembly language file |
| .a | Static library of object files |
| .so | Share (dynamic) library of object files |

| Option | Description |
|-----------------|--|
| -c | Generate object file for each source file |
| -g | Produce debugging information |
| -I<path> | Add <path> to directories preprocessor searches |
| -L<path> | Add <path> to directories searched during link for libraries |
| -l<lib> | Searches library <lib> for unresolved symbols. |
| -lm | Link using libm math library |
| -o<file> | Writes generated output to <file> |
| -O or -O2 | [Default] Generates optimized binary |
| -O[0 1 2 3 s] | Controls optimization level (0=min, 3=max) |
| -Ofast | Equivalent to -O3 -ipa -OPT:Ofast -fno-math-errno -ffast-math |
| -ipa | Invoke inter-procedural analysis |
| -ar | Create static archive (.a) |
| -shared | Create shared object library (.so) |
| -D <name>[=DEF] | Define macro <name> with optional value DEF. Default DEF value is 1. |
| -apo | Enable auto-parallelization |

PathScale, LLC
 1320 Chesapeake Terrace
 Sunnyvale, CA 94089
 408-350-5574
 sales@pathscale.com
 support@pathscale.com

Copyright (C) 2008 PathScale LLC. All Rights Reserved
 June 11, 2008

| Option | Description |
|----------------------------------|--|
| ### | Like -v except nothings is run, args are quoted |
| -ansi | Generate messages on non-ANSI Standard constructs |
| -backslash | Treat backslash as a normal character |
| -CG: | Option group to control code generation |
| -CLIST: | Option group to control emission of C from compiler |
| -copyright | Display copyright message |
| -cpp | run cpp before compiling, regardless of file suffix |
| -dumpversion | Display version number of compiler |
| -E | Run only the preprocessor |
| -fb-create <path> | Instrument code for feedback directed optimization. Store data in directory <path> |
| -fb-opt <path> | Optimize using feedback directed optimization data stored in <path> |
| -fb-phase=(0 1 2 3 4) | Compiler phase to instrument feedback directed optimization for. Default is 0 |
| -fe | Stop after front-end |
| -f[no]-exceptions | Controls exception handling |
| -ffast-math | Improve FP perf by relaxing ANSI & IEEE rules |
| -fno-fast-math | Conform to ANSI & IEEE rules |
| -f[no]-fast-stdlib | Controls use of fast standard library functions |
| -ffloat-store | Do not store floating point variables in registers |
| -f[no]-gnu-keywords | Controls typeof keyword |
| -f[no]-implicit-inline-templates | Controls generation of code for inline templates instantiated implicitly |
| -f[no]-implicit-templates | Controls generation of code for non-inline templates instantiated implicitly |
| -finhibit-size-directive | Do not generate .size directives |
| -f[no]-inline | Controls inlining. (Same as -[no]inline) |
| -finstrument-functions | Insert instrumentation calls at entry & return functions |
| -fno-ident | Inore #ident directives |
| -fno-math-errno | Do not set ERRNO after calling math functions |
| -f[no]-permissive | Controls messages about non-conformant code to warnings |
| -f[no]-PIC | Controls generation of position independent code |
| -f[no]-preprocessed | Controls preprocessing of input |
| -funsafe-math-optimizations | Improve FP speed by violating ANSI & IEEE rules |
| -fno-unsafe-math-optimizations | Conform to FP ANSI & IEEE rules |
| -ftest-coverage | Generate coverage files for gcov |
| -fullwarn | Generate more warnings |
| -f[no]-unwind-tables | Controls emission of unwind information |
| -GRA: | Option group to control global register allocator |
| -g[0 1 2] | Generate debuggin info. 0=none, 1=minimal, 2=maximum |
| -H | Print the name of each header file used |
| -help | List available options |
| -help:<string> | List options that contain <string> |
| -ignore-suffix | Determine language by command instead of file suffix |
| -INLINE: | Option group to control inlining |
| -inline | Request inline processing |
| -IPA: | Option group to control Inter-procedural analysis |
| -iquote <dir> | Add to search path for #include "file" before -I dirs |
| -isisroot <dir> | Defne system root for header files |
| -isystem <dir> | Search <dir> for headers after -I dirs, but before system dirs |
| -keep | Keep all intermediate files |
| -LANG: | Option group to control language features |
| -LIST: | Option group to control listing (.lst) file contents |
| -LNO: | Option group to control Loop Nest Optimizer |
| -M | run cpp and print list of make dependencies |

| | |
|----------------------------|---|
| -m32 | Compile for 32-bit (x86) ABI |
| -m3dnow | Control use of 3DNow instructions |
| -m64 | Compile for 64-bit (x86_64) ABI |
| -march=<cpu-type> | Optimize code for <cpu-type> |
| -mcmode=(small medium) | Code size model used when generating offsets within object files. |
| -mcpu=<cpu-type> | Synonym for -march |
| -MD | Write dependencies to .d output file |
| -MDtarget | Use as the target for make dependencies |
| -MDupdate | Update file with Make dependencies |
| -MF | Write dependencies to specified output file |
| -MG | Treat missing header files as generated files |
| -MM | Output user dependencies of source file |
| -MMD | Write user dependencies to .d output file |
| -mp | Enable support for OpenMP |
| -MP | With -M or -MM, add phony targets for each dependency |
| -MQ | Same as -MT but quote Make's special characters |
| -m[no]-sse | Control use of SSE2 & SSE3 |
| -m[no]-sse2 | Controls SSE2 instruction generation |
| -m[no]-sse3 | Controls SSE3 instruction generation |
| -m[no]-sse4a | Controls SSE4A instruction generation |
| -mtune=<cpu-type> | Synonym for -march |
| -MT | Change the target of the generated dependency rules |
| -mx87-precision=(32 64 80) | Specify the precision of the x87 floating-point calculations |
| -nobool | Disables bool keyword |
| -nocpp | Do not run the preprocessor |
| -nodefaultlibs | Do not use standard system libraries when linking |
| -noexpop | Do not optimize exponentiation operations |
| -noinline | Suppress expansion of inline functions |
| -no-pathcc | Don't define __PATHSCALE__ |
| -nostartfiles | Do not sue standard system startup files when linking |
| -nostdinc | Do not search standard locations for #include directives |
| -nostdlib | No predefined libraries or startfiles |
| -objectlist <file> | Read <file> for a list of files to be linked |
| -openmp | Synonym for -mp |
| -OPT: | Option group to control miscellaneous optimizations |
| -P | With -E, disables generation of #line directives |
| -pathcc | Define __PATHCC__ and other macros |
| -pg | Generate code to profile with gprof |
| -profile | Same as -pg but includes libraries |
| -r | Generate relocatable .o and stop |
| -S | Generate symbolic assembly output |
| -shared-libgcc | Force the use of shared libgcc |
| -show | Print passes as they are executed |
| -show-defaults | Show the compiler "defaults" including compiler.defaults contents |
| -show0 | Show what phases would be executed, but don't do anything |
| -showt | Show time taken by each phase |
| -static-data | Statically allocate all local variables |
| -stdinc | Predefined include search path list |
| -subverbose | Produce subscription manager diagnostic output |
| -sysroot=<dir> | Root directory for system headers and libraries |
| -TENV: | Option group to control Target Environment |
| -trapuv | Trap uninitialized variables |
| -U<name> | Undefine <name> in preprocessor |
| -v | Print the commands executed during compilation |
| -version | Write compiler release version information to stdout |
| -W<c>,<args> | Pass <args> to pass <c> of compiler |
| -Wall | Enable most warning messages |
| -WB,<arg> | Passes <arg> to backend via ipacomp |
| -W[no]-deprecated | Warn on deprecated compiler features |

| | |
|-------------------------------|---|
| -W[no-]disabled-optimizations | Warn if a requested optimization pass is disabled |
| -W[no-]div-by-zero | Warns about compile-time integer divide by zero |
| -W[no-]endif-labels | Warn if #if or #endif is followed by text |
| -W[no-]error | Makes all warnings into errors |
| -WOPT: | Option group to control global optimizer |
| -w | Suppress warning messages |
| -woff | Turn off named warnings |
| -woffall | Turn off all warnings |
| -woffoptions | Turn off warnings about options |
| -woffn | Specify message numbers to suppress |
| -Xlinker <opt> | Pass <option> to the linker |
| -Y<c>,<path> | Set the <path> where the phase <c> is found |
| -zerouv | Set uninitialized variables to zero |

C/C++ Only

| | |
|--|--|
| -C | Keep comments after preprocessing |
| -clist | Enables C listing |
| -fabi-version=N | (C++) Use version N of the C++ ABI. 1=G++ 3.2 0=current pathCC ABI |
| -f[no-]gnu-exceptions | Controls exception handling. Same as -fexceptions |
| -f[no-]inline-functions | Controls automatic simple function inlining |
| -fkeep-inline-functions | Generate code for functions even if fully inlined |
| -fms-extensions | Accept broken MFC extensions without warning |
| -fno-asm | Do not recognize the 'asm' keyword |
| -fno-builtin | Do not recognize any built in functions |
| -fpack-struct | Pack structure members together without holes |
| -fprefix-function-name | Add a prefix to all function names |
| -frandom-seed=<string> | Seed random number generator for generated file names |
| -f[no-]rtti | [Controls generation of runtime type information |
| -f[no-]signed-bitfields | Make bitfields signed by default |
| -f[no-]signed-char | Make char signed by default |
| -f[no-]strict-aliasing | Assume strictest aliasing rules |
| -fshared-data | Mark data as shared rather than private |
| -fshort-double | Same size for double and float |
| -fshort-enums | Use smallest fitting integer for enums |
| -fshort-wchar | Use short unsigned int for wchar_t |
| -fuse-cxa-atexit | (C++) Register static destructors with __cxa_atexit instead of atexit |
| -fwritable-strings | Support K&R style C writable-strings |
| -gnuN | Emit code compatible with N series GNU compilers. 3 = GCC 3.3, 4 = GCC 4.2 |
| -pedantic-errors | Generate warnings on non-ANSI C compliant code |
| -std=<option> | Controls flavor of C/C++ standard adherence |
| -traditional | Use K&R style C |
| -W[no-]aggregate-return | Warns about returning structures, arrays, or unions |
| -W[no-]bad-function-cast | Attempts to support K&R style C writable strings |
| -W[no-]cast-align | Warns about pointer casts that increase alignment |
| -W[no-]char-subscripts | Warn about subscripts whose type is 'char' |
| -W[no-]comment | Warns on nested comments |
| -W[no-]conversion | Warn about possibly confusing type conversions |
| -Wdeclaration-after-statement | Warn about declarations after statements |
| -Werror-implicit-function-declarations | Give an error when a function is used without being declared |
| -W[no-]float-equal | Warns if floating point values are compared for equality |
| -W[no-]format | Warns about printf format anomalies |
| -W[no-]format-nonliteral | With -Wformat, warn if format string is not a literal |
| -W[no-]format-security | With -Wformat, warn on potentially insecure format functions |
| -W[no-]id-clash <N> | Warn if two identifiers have same N chars |

| | |
|---------------------------------------|--|
| -W[no-]implicit | Warn about implicit declarations of functions or variables |
| -W[no-]implicit-function-declarations | Warn when a function is used before being declared |
| -W[no-]implicit-int | Warns when a declaration does not specify a type |
| -W[no-]import | Warn about the use of #import directive |
| -W[no-]inline | Warn if a function declared as inline cannot be inlined |
| -W[no-]larger-than-N | Warn if an object is larger than N bytes |
| -W[no-]main | Warns about suspicious declarations of main |
| -W[no-]missing-braces | Warns about possibly missing braces around initializers |
| -W[no-]missing-declarations | Warns about global functions without previous declarations |
| -W[no-]missing-format-attributes | With -Wformat, warn on candidates for 'format' attributes |
| -W[no-]missing-noreturn | Warns about functions that are candidates for 'noreturn' attribute |
| -W[no-]missing-prototypes | Warns about global functions without prototypes |
| -W[no-]multichar | Warns if multi-character constants are used |
| -W[no-]nested-externs | Warns about externs not at file scope level |
| -Wno-cast-qual | Warns about casts that discard qualifiers |
| -Wno-deprecated-declarations | Do not warn about deprecated declarations in code |
| -Wno-format-extra-args | Do not warn about extra arguments to printf-like functions |
| -Wno-format-y2k | Do not warn about strftime formats that yield two-digit years |
| -Wno-long-long | Warns if long long type is used |
| -Wno-non-ateemplate-friend | (C++) Do not warn about friend functions declared in templates |
| -Wno-pmf-conversions | (C++) Do not warn about converting PMFs to plain pointers |
| -W[no-]non-virtual-dtor | (C++) Warn when a class declares a dtor that should be virtual |
| -Wnonnull | Warn when passing null to functions requiring non-null pointers |
| -W[no-]old-style-cast | Warn when a C-style cast to a non-void type is used |
| -W[no-]overloaded-virtual | (C++) Warn when a function declaration hides virtual functions |
| -W[no-]packed | Warns when packed attributes has no effect |
| -W[no-]padded | Warns when padding is included in a struct |
| -W[no-]parentheses | Warns about possible missing parentheses |
| -W[no-]pointer-arith | Warns about function pointer arithmetic |
| -W[no-]redundant-decls | Warns about multiple declarations of same object |
| -W[no-]reorder | Warns when reordering member initializers |
| -W[no-]return-type | Warns when a function return type defaults to int |
| -W[no-]sequence-point | Warns about code violating sequence point rules |
| -W[no-]shadow | Warns when one local variables shadows another |
| -W[no-]sign-compare | Warns about signed/unsigned comparisons |
| -W[no-]sign-promo | Warns when overload resolution promotes from unsigned to signed |
| -W[no-]strict-aliasing | Warns about code that breaks strict aliasing rules |
| -W[no-]strict-prototypes | Warns about non-prototyped function declarations |
| -W[no-]switch | Warns when a switch statement is incorrectly indexed with an enum |
| -W[no-]switch-default | Warn when a switch statement has no default |
| -W[no-]switch-enum | Warn when a switch statement is missing a case for an enum member |
| -W[no-]system-headers | Warns for constructs in system header files |
| -W[no-]synth | Warns about synthesis that is not backwards compatible with cfront |

| | |
|-------------------------|---|
| -W[no-]traditional | Warns about constructs whose meanings change in ANSI C |
| -W[no-]trigraphs | Warns when trigraphs are encountered |
| -W[no-]undefs | Warns if an undefined identifier appears in a #if directive |
| -W[no-]uninitialized | Warns about uninitialized automatic variables |
| -W[no-]unknown-pragmas | Warn when an unknown #pragma is encountered |
| -W[no-]unreachable-code | Warns about code that will never be executed |
| -W[no-]unused | Warns when a variable is unused |
| -W[no-]unused-function | Warns about unused static and inline functions |
| -W[no-]unused-label | Warns about unused labels |
| -W[no-]unused-parameter | Warns about unused function parameters |
| -W[no-]unused-value | Warns about statements whose results are not used |
| -W[no-]unused-variable | Warn about local and static variables that are not used |
| -W[no-]write-strings | Mark strings as 'const char' |

Fortran Only

| | |
|--------------------------------|---|
| -align(32 64) | Align common block data on 32 or 64 bit boundaries |
| -auto-use <module>[, <module>] | Compiler behaves as if "USE <module>" is in every program unit. |
| -byteswapio | Swap bytes during I/O |
| -C | Enable runtime subscript checking |
| -colN | Set line width to N for fixed-format source |
| -d-lines | compile lines with a D in column 1 |
| -default64 | synonym for -r8 -i8 |
| -extend-source | 132 character line length |
| -fcoco | Use Fortran ISO/ANSI Standard conditional compilation |
| -fdecorate <path> | Specifies how to decorate external Fortran identifiers |
| -f[no-]directives | Controls recognition of directives in comments |
| -ff2c-abi <file> | Use GNU f2c ABI when calling functions listed in <file> |
| -ffortran2003 | Force Fortran 2003 intrinsic conversion behavior |
| -ffortran-bounds-check | Enable bounds checking |
| -fixedform | Treat all input as fixed form source. |
| -flist | Invoke all Fortran listing options |
| -freeform | All input source files treated as freeform |
| -f[no-]second-underscore | Append second underscore to symbols that already have one |
| -ftpp | Use traditional Fortran preprocessor instead of cpp for .F* files |
| -f[no-]underscoring | Append underscores to symbols |
| -iN | Length of default integer constants: 4 = 32-bit, 8 = 64-bit |
| -[no-]intrinsic=<name> | Control the set of intrinsic subprograms recognized |
| -keepdollar | Treat \$ as normal last character in symbol names |
| -macro-expand | Enable macroexpansion |
| -module <dir> | Generate module (.mod) files to <dir> |
| -noextend-source | Restrict Fortran source code lines to columns 1-72 |
| -nog77mangle | Default is to mangle names like g77 |
| -pad-char-literals | Pad characteral s used as arguments to size of integer |
| -r(4 8) | Length of default real type. 4: KIND=4, 8: KIND=8 |
| -u<var> | Make default type of variable undefined |

