Intel® Fortran Compiler Options Quick Reference

Intel® Fortran Compiler for Linux* Systems Options Quick Reference Guide

Legal Information
Document Number: 253258-002

Disclaimer and Legal Information

Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, or life sustaining applications.

This Quick Reference Guide as well as the software described in it is furnished under license and may only be used or copied in accordance with the terms of the license. The information in this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Intel Corporation. Intel Corporation assumes no responsibility or liability for any errors or inaccuracies that may appear in this document or any software that may be provided in association with this document.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The software described in this Quick Reference Guide may contain software defects which may cause the product to deviate from published specifications. Current characterized software defects are available on request.

Intel SpeedStep, Intel Thread Checker, Celeron, Dialogic, i386, i486, iCOMP, Intel, Intel logo, Intel386, Intel486, Intel740, IntelDX2, IntelDX4, IntelSX2, Intel Inside, Intel Inside logo, Intel NetBurst, Intel NetStructure, Intel Xeon, Intel Centrino, Intel XScale, Itanium, MMX, MMX logo, Pentium, Pentium II Xeon, Pentium III Xeon, Intel Pentium M processor, and VTune are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

Table Of Contents

How to Use This Guide

The Intel® Fortran Compiler Options Quick Reference Guide contains the following information:

- New options for the current release
- An alphabetical listing of all options
- A cross reference of Windows* and Linux* compiler options

For complete information on each compiler option, refer to the *Intel® Fortran Compiler User's Guide,* Volumes I and II, and the `ifort` man page.

In this guide, compiler options are available on both IA-32 and Intel® Itanium® processors unless otherwise identified.

Notation Conventions

ON in "Default" column	Indicates that the option is in effect by default when the compiler is invoked. If an option has a value for the ON state, it is indicated in parentheses.
OFF in "Default" column	Indicates that, by default, the option is not used when the compiler is invoked. If an option has a value for the OFF state, it is indicated in parentheses.
`this type style`	Italic, monospaced text indicates placeholders for information that you must supply. Italics are also used to introduce new terms. For example, the option `-Qoption,`*`str,opts`* could appear in the command line as follows: `ifort -Qoption,link,-w myprog.f`
{value | value}	Braces and a vertical bar indicate a choice among two or more items. You must choose one of the items unless all of the items are also enclosed in square brackets.
`-option` *`parameter`*	Indicates that an option requires a parameter; for example, in `-L`*`dir`*, the option `-L` instructs the linker to search directory *`dir`* for libraries.
`-option` *`keyword`*	Indicates that an option requires one of the keyword values.
`-option [`*`keyword`*`]`	Indicates that the option can be used alone or with an optional keyword.

`-option[`*n*`]`	Indicates that the option can be used alone or with an optional value; for example, in `-unroll[`*n*`]`, *n* can be omitted or a valid value can be specified.
`-option[-]`	Indicates that a trailing hyphen disables the option; for example, `-ansi_alias-` disables the `-ansi_alias` option.
`-[no]option`	Indicates that "no" preceding an option disables the option. For example, in `-[no]altparam`, `-altparam` enables the option, while `-noaltparam` disables the option. **Note** The `[no]options` are listed in the alphabetical order of an `option`.

New Options

The following table lists new options in this release.

Some compiler options are only available on certain systems, as indicated by these labels:

Label	Meaning
i32	The option is available on IA-32-based systems
i32em	The option is available on IA-32-based systems with Intel® Extended Memory 64 Technology (Intel® EM64T)
i64	The option is available on Itanium®-based systems

If no label appears, the option is available on all supported systems.

If "only" appears in the label, the option is only available on the identified system.

For more details on these options refer to the *Intel® Fortran Compiler User's Guide*, Volumes I and II, and the `ifort` man page.

For information on conventions used in this table, see Notation Conventions.

Option	Description	Default
`-debug` *`keyword`*	Specifies settings that enhance debugging (requires `-g`). *`keyword`*: `variable_locations`	OFF
`-IPF_fp_relaxed` (i64 only)	Enables use of faster but slightly less accurate code sequences for math functions, such as divide and sqrt. When compared to strict IEEE* precision, this option slightly reduces the accuracy of floating-point calculations performed by these functions, usually limited to the least significant digit.	OFF
`-ipo[`*`n`*`]`	The optional *n* is new. It is an integer that specifies the number of object files the compiler should create. By default, at least one object file is created.	OFF
`-ipo_separate`	Tells the compiler to generate one object file per source file. This option overrides any `-ipo`*`n`* specification.	OFF

`-openmp_profile`	Enables analysis of OpenMP* applications. To use this option, you must have Thread Profiler installed, which is one of the Intel® Threading Tools. If this threading tool is not installed, this option has no effect.	OFF
`-tcheck`	Enables analysis of threaded applications. To use this option, you must have Intel® Thread Checker installed, which is one of the Intel® Threading Tools. If this threading tool is not installed, this option has no effect.	OFF

Alphabetical Quick Reference Guide

The following table summarizes options that you can use for compilations. For more details on these options, refer to the *Intel® Fortran Compiler User's Guide*, Volumes I and II, and the `ifort` man page.

Some compiler options are only available on certain systems, as indicated by these labels:

Label	Meaning
i32	The option is available on IA-32-based systems
i32em	The option is available on IA-32-based systems with Intel® Extended Memory 64 Technology (Intel® EM64T)
i64	The option is available on Itanium®-based systems

If no label appears, the option is available on all supported systems.

If "only" appears in the label, the option is only available on the identified system.

The options that are new for this release are marked with (***new**). If a new keyword that does not change the functionality is added, this mark appears near that keyword value. If a new keyword adds a new functionality, the option with the new keyword takes a separate entry and is marked as (***new**).
For information on conventions used in this table, see Notation Conventions.

Option	Description	Default
`-1`	Executes at least one iteration of DO loops (same as the `-onetrip` option). This option has the same effect as `-f66`.	OFF
`-66`	Enforces FORTRAN-66 semantics (same as the `-f66` option).	`-nof66`
`-72, -80, -132`	Treats the statement field of each fixed-form source line as ending in column 72, 80, or 132. Option `-132` is the same as the `-extend_source` option.	`-72`

`-align` *keyword*	Tells the compiler how to align data items.
	keyword: `all`, `none`, `[no]commons`, `[no]dcommons`, `[no]records`, `rec`*n*`byte`, `[no]sequence`
	For details on these keywords, see your user's guide or the `ifort` man page.
		`-align nocommons`
		`-align nodcommons`
		`-align records`
		`-align nosequence`

`-ansi_alias-`	Tells the compiler to assume the program does *not* adhere to the Fortran 95 Standard type aliasability rules.	`-ansi_alias`
`-arch` *keyword* (i32 only)	Determines the version of the architecture for which the compiler generates instructions. *keyword*: `pn1` - Optimizes for the Intel® Pentium® processor. `pn2` - Optimizes for the Intel® Pentium® Pro, Intel® Pentium® II, and Intel® Pentium® III processors. `pn3` - This is the same as specifying the `-arch pn2` option. `pn4` - Optimizes for the Intel® Pentium® 4 processor. SSE - Optimizes for Intel Pentium 4 processors with Streaming SIMD Extensions (SSE). SSE2 - Optimizes for Intel Pentium 4 processors with Streaming SIMD Extensions 2 (SSE2).	`-arch pn4`

`-assume` *`keyword`*	Specifies assumptions made by the compiler. *`keyword`*: `none`, `[no]bscc`, `[no]buffered_io`, `[no]byterecl`, `[no]cc_omp`, `[no]dummy_aliases`, `[no]minus0`, `[no]protect_constants`, `[no]source_include`, `[no]underscore` If `-openmp` is specified, it sets `-assume cc_omp`. For details on these keywords, see your user's guide or the `ifort` man page.	OFF (`-assume none`)
`-auto`	Places variables, except those declared as SAVE, on the run-time stack (same as `-automatic` or `-nosave`). If you specify `-recursive` or `-openmp`, the default is `-auto`.	`-auto_scalar`
`-auto_ilp32` (i32em, i64)	Specifies that the application cannot exceed a 32-bit address space, which allows the compiler to use 32-bit pointers whenever possible. To use this option, you must also specify `-ipo`. On Intel® EM64T systems, this option has no effect unless you also specify `-xP` or `-axP`. If you use the `-auto_ilp32` option on programs that can exceed 32-bit address space (2**32), unpredictable results may occur during program execution.	OFF
`-auto_scalar`	Makes AUTOMATIC all scalar local variables of intrinsic type INTEGER, REAL, COMPLEX, or LOGICAL. You cannot specify `-save`, `-auto`, or `-automatic` with this option.	ON unless `-recursive` or `-openmp` is specified.

`-autodouble`	Defines real variables to be REAL(KIND=8). This option is the same as specifying `-r8`.	OFF
`-automatic`	Places variables, except those declared as SAVE, on the run-time stack (same as the `-auto` or `-nosave` options). If you specify `-recursive` or `-openmp`, the default is `-auto`.	`-auto_scalar`
`-ax{K|W|N|B|P}` (i32, i32em)	Generates processor-specific code if there is a performance benefit. The processor type is indicated by one of the following values: `K` - Intel® Pentium® III processors and compatible Intel processors. `W` - Intel® Pentium® 4 processors and compatible Intel processors. `N` - Intel® Pentium® 4 processors and compatible Intel processors. `B` - Intel® Pentium® M and compatible Intel processors `P` - Intel® Pentium® 4 processor with Streaming SIMD Extensions 3 (SSE3) instruction support The only options available on Intel® EM64T systems are `-axP` and `-axW`. For more details, see your user's guide or the `ifort` man page.	OFF
`-Bdynamic`	Enables dynamic linking of libraries at run time. Smaller executables are created than with static linking.	OFF
`-Bstatic`	Enables static linking of a user's library.	OFF
`-c`	Causes the compiler to compile to an object (`.o`) file only and not link.	OFF

`-CB`	Performs run-time checks on whether array subscript and substring references are within declared bounds. Same as the `-check bounds` option.	OFF
`-ccdefault` *`keyword`*	Specifies the type of carriage control used for units 6 and `*`. *`keyword`*: `default`, `fortran`, `list`, or `none`. The default setting can be affected by the `-vms` option. For details on these keywords, see your user's guide or the `ifort` man page.	`-ccdefault default`
`-check` *`keyword`*	Checks several run-time conditions. *`keyword`*: `all`, `none`, `[no]arg_temp_created`, `[no]bounds`, `[no]format`, `[no]output_conversion`. For details on these keywords, see your user's guide or the `ifort` man page.	OFF (`-nocheck` or `-check none`)
`-cm`	Suppresses all messages about questionable programming practices (same as the `-warn nousage` option).	OFF (`-nocm`)
`-common_args`	Tells the compiler that dummy (formal) arguments to procedures share memory locations with other dummy arguments or with variables shared through use association, host association, or common block use. This is the same as specifying `-assume dummy_aliases`.	OFF (`-no common_args`)
`-complex_limited_range[-]`	Enables the use of basic algebra-ic expansions of some arithmetic operations involving data of type COMPLEX. This can cause some performance improvements in programs that use a lot of COMPLEX arithmetic, but values at the extremes of the exponent range may not compute correctly.	OFF (`-complex_limited_range-`)

`-convert` *keyword*	Specifies the format of unformatted files containing numeric data. *keyword*: `big_endian`, `cray`, `fdx`, `fgx`, `ibm`, `little_endian`, `native`, `vaxd`, `vaxg`.	`-convert native`
`-cpp`	Runs the Fortran preprocessor on source files prior to compilation (same as the `-fpp` option).	OFF (`-nocpp`)
`-D`*name* `-D`*name*`[=value]`	Defines the *name* as a definition to use with conditional compilation directives or the Fortran preprocessor (`-fpp`). The `value` can be an integer or it can be a character string delimited by double quotes; for example, `-Dname="string"`. If no definition is given, *name* is defined as "1".	OFF
`-d_lines`	Compiles debug statements indicated by the letter D in column 1 of the source code; this is the same as specifying `-DD`.	OFF (`-nod_lines`)
`-DD`	Compiles debug statements indicated by the letter D in column 1 of the source code; this is the same as specifying `-d_lines`.	OFF
`-debug` *keyword* (***new**)	Specifies settings that enhance debugging. To use these options, you must also specify `-g`. *keyword*: variable_locations For details on this keyword, see your user's guide or the `ifort` man page.	OFF
`-double_size` *size*	Defines the SIZE of DOUBLE PRECISION and DOUBLE COMPLEX declarations, constants, functions, and intrinsics. *size* can be 64 or 128.	`-double_size 64`
`-dryrun`	Specifies that driver tool commands should be shown but not executed.	OFF (`-nodryrun`)
`-dynamic-linker`*file*	Specifies a dynamic linker in *file* other than the default.	OFF

`-dyncom "a,b,c"`	Enables dynamic allocation of the specified COMMON blocks at run time. The quotes are required.	OFF
`-E`	Causes the Fortran preprocessor to send output to stdout.	OFF
`-e90`	Causes the compiler to issue errors instead of warnings for nonstandard Fortran 90.	OFF
`-e95`	Causes the compiler to issue errors instead of warnings for nonstandard Fortran 95.	OFF
`-EP`	Causes the Fortran preprocessor to send output to stdout, omitting #line directives.	OFF
`-[no]error_limit` *n*	Specifies the maximum number of error-level or fatal-level compiler errors allowed for a file specified on the command line. If you specify `-noerror_limit`, there is no limit to the number of errors that are allowed.	`-error_limit 30`
`-extend_source` [*size*]	Specifies the column number to use to end the statement field in fixed-form source files. *size* can be 72, 80, or 132. Specifying `-noextend_source` implies column 72. If you specify `-extend_source` with no *size*, it's the same as specifying `-extend_source 132`.	OFF (`-no extend_source`)
`-F`	Causes the Fortran preprocessor to send output to a file (same as the `-preprocess_only` and `-P` options). To use this option, you must also specify `-fpp`.	OFF
`-f66`	Tells the compiler to apply FORTRAN-66 semantics; the default is to apply Fortran 95 semantics. For more details, see your user's guide or the `ifort` man page.	OFF (`-nof66`)

-f77rtl	Tells the compiler to use the run-time behavior of FORTRAN 77 instead of Intel® Fortran. For more details, see your user's guide or the ifort man page.	OFF (-nof77rtl)
-fast	Maximizes speed across the entire program. On Itanium®-based systems, this option sets options -O3, -ipo, and -static. On IA-32 and Intel® EM64T systems, this option sets options -O3, -ipo, -static, and -xP. Note that on IA-32 systems, programs compiled with the -xP option will detect non-compatible processors and generate an error message during execution.	OFF (-nofast)
-fcode_asm	Produces an assembly file with optional code annotations. To use this option, you must also specify -S.	OFF
-FI	Specifies source files are in fixed format (same as the -fixed option).	OFF
-fixed	Specifies source files are in fixed format (same as the -FI option). By default, source file format is determined by the file suffix.	OFF
-fltconsistency	Enables improved floating-point consistency. The default setting provides better accuracy and run-time performance at the expense of less consistent floating-point results. For more details, see your user's guide or the ifort man page.	OFF (-no fltconsistency)
-fminshared	Specifies that a compilation unit is a component of a main program and will not be linked as part of a shareable object.	OFF
-fno-alias	Specifies that aliasing should not be assumed in the program.	-falias

`-fno-fnalias`	Specifies that aliasing should not be assumed within functions, but should be assumed across calls.	`-ffnalias`
`-fnsplit` (i64 only)	Enables function splitting if `-prof_use` is also specified. Otherwise, the default value disables the splitting within a routine but leaves function grouping enabled.	OFF (`-fnsplit-`)
`-fp` (i32, i32em)	Disables using EBP as a general purpose register so it can be used as a stack frame printer.	OFF (`-fp-`)
`-fp_port` (i32 only)	Rounds floating-point results after floating-point operations, so rounding to user-declared precision happens at assignments and type conversions; this has some impact on speed. The default is to keep results of floating-point operations in higher precision; this provides better performance but less consistent floating-point results.	OFF
`-fpconstant`	Tells the compiler to extend the precision to double precision for single-precision constants assigned to double-precision variables.	OFF (`-nofpconstant`)
`-fpe`*n*	Specifies floating-point exception handling at run time for the main program. *n*=0, 1, 3. `0` - floating underflow results in zero; all other floating-point exceptions abort execution `1` - floating underflow results in zero; all other floating-point exceptions produce exceptional values (signed Infinities or NaNs) and execution continues `3` - all floating-point exceptions produce exceptional values (signed infinities, denormals, or NaNs) and execution continues; also see `-ftz`.	`-fpe3`

`-fpic`	Generates position-independent code. Can also be specified as `-fPIC`. On Itanium-based systems, this option must be used when building shared objects.	OFF
`-fpp`	Runs the Fortran preprocessor on source files prior to compilation.	OFF
`-fpscomp` [*keyword*]	Specifies the level of compatibility with Microsoft* Fortran PowerStation or Intel® Fortran. *keyword*: `all`, `none`, `[no]filesfromcmd`, `[no]general`, `[no]ioformat`, `[no]libs`, `[no]ldio_spacing`, `[no]logicals`. For details on these keywords, see your user's guide or the `ifort` man page.	For `all` and `nolibs`: `-fpscomp libs` For the rest: `-fpscomp none`
`-fpstkchk` (i32 only)	Generates extra code after every function call to ensure that the FP (floating-point) stack is in the expected state. By default, there is no checking. For more details, see your user's guide or the `ifort` man page.	OFF
`-FR`	Specifies source files are in free format (same as the `-free` option).	Based on source file extension
`-fr32` (i64 only)	Disables use of high floating-point registers. Uses only the lower 32 floating-point registers.	OFF
`-free`	Specifies source files are in free format (same as the `-FR` option). By default, source file format is determined by the file suffix.	OFF
`-fsource_asm`	Produces an assembly file with optional code annotations. To use this option, you must also specify `-S`.	OFF
`-ftz`	Enables floating underflow results set to zero. On Itanium-based systems, option `-O3` sets the `-ftz` option.	OFF `(-ftz-)`

`-fverbose-asm`	Produces an assembly file with compiler comments, including options and version information. To use this option, you must also specify `-S`, which sets `-fverbose-asm`. If you do not want this default when you specify `-S`, specify `-fnoverbose-asm`.	OFF (`-fnoverbose-asm`)
`-fvisibility=`*`keyword`* `-fvisibility-`*`keyword=file`*	The first form specifies the default visibility for global symbols. The second form specifies the visibility for symbols that are in a file (this form overrides the first form). *`file`* is the pathname of a file containing the list of symbols whose visibility you want to set; the symbols are separated by whitespace (spaces, tabs, or newlines). *`keyword`*: `external`, `default`, `protected`, `hidden`, and `internal`. For details on these keywords, see your user's guide or the `ifort` man page.	OFF
`-g`	Produces symbolic debug information in the object file. The compiler does not support the generation of debugging information in assemblable files. If you specify the `-g` option, the resulting object file will contain debugging information, but the assemblable file will not. On IA-32 systems, specifying the `-g` or `-O0` option automatically enables the `-fp` option. For more information, see "Optimizations and Debugging" in Volume II of your user's guide.	OFF

`-help`	Displays the list of compiler options.	OFF
`-I`*`dir`*	Specifies a directory to add to the include path, which is used to search for module files (USE statement) and include files (INCLUDE statement).	OFF
`-i_dynamic`	Links Intel-provided libraries dynamically.	OFF
`-i{2|4|8}`	Specifies the default size of integer and logical variables to be 2, 4, or 8 bytes (same as `-integer_size {16|32|64}`).	`-integer_size 32`
`-implicitnone`	Sets the default type of a variable to undefined (IMPLICIT NONE). Same as the `-u` option.	OFF
`-inline_debug_info`	Produces enhanced source position information for inlined code. It also provides enhanced debug information useful for function call traceback. To use this option for debugging, you must also specify `-g`.	OFF
`-intconstant`	Tells the compiler to use Fortran 77 semantics, rather than Fortran 95/90 semantics, to determine the KIND for integer constants.	OFF (`-nointconstant`)
`-integer_size` *`size`*	Specifies the default size of integer and logical variables. *`size`* can be 16, 32, or 64.	`-integer_size 32`
`-ip`	Enables single-file interprocedural optimizations. If you specify this option, the compiler performs inline function expansion for calls to functions defined within the current source file.	OFF
`-ip_no_inlining`	Disables full and partial inlining enabled by `-ip`. To use this option, you must specify `-ip` or `-ipo`.	OFF
`-ip_no_pinlining`	Disables partial inlining. To use this option, you must specify `-ip` or `-ipo`.	OFF

`-IPF_fltacc` (i64 only)	Disables optimizations that affect floating-point accuracy. If the default setting is used, the compiler may apply optimizations that reduce floating-point accuracy. You can use `-IPF_fltacc` or `-mp` to improve floating-point accuracy, but at the cost of disabling some optimizations.	OFF `(-IPF_fltacc-)`
`-IPF_flt_eval_method0` (i64 only)	Tells the compiler to evaluate the expressions involving floating-point operands in the precision indicated by the variable types declared in the program. By default, intermediate floating-point expressions are maintained in higher precision.	OFF
`-IPF_fma` (i64 only)	Enables the combining of floating-point multiplies and add/subtract operations. Also enables the contraction of floating-point multiply and add/subtract operations into a single operation. The compiler contracts these operations whenever possible. However, if `-mp` is specified, these contractions are disabled.	OFF `(-IPF_fma-)`
`-IPF_fp_relaxed` (***new**) (i64 only)	Enables use of faster but slightly less accurate code sequences for math functions, such as divide and sqrt. When compared to strict IEEE* precision, this option slightly reduces the accuracy of floating-point calculations performed by these functions, usually limited to the least significant digit.	OFF `(-IPF_fp_relaxed-)`

`-IPF_fp_speculation`*`mode`* (i64 only)	Tells the compiler to speculate on floating-point (FP) operations in one of the following `modes`: *`fast`*: speculate on floating-point operations *`safe`*: speculate on floating-point operations only when safe *`strict`*: same as specifying *`off`* *`off`*: disables speculation of floating-point operations	`-IPF_fp_speculation`*`fast`*
`-ipo[`*`n`*`]`	Enables multifile interprocedural (IP) optimizations (between files). When you specify this option, the compiler performs inline function expansion for calls to functions defined in separate files. n is an optional integer that specifies the number of object files the compiler should create. Any integer greater than or equal to 0 is valid. If n is 0, the compiler decides whether to create one or more object files based on an estimate of the size of the object file. It generates one object file for small applications, and two or more object files for large applications. If n is greater than 0, the compiler generates n object files, unless n exceeds the number of source files (`m`), in which case the compiler generates only `m` object files. If you do not specify n, the default is 1.	OFF
`-ipo_c`	Generates a multifile object file (`ipo_out.o`) that can be used in further link steps.	OFF
`-ipo_obj`	Forces the generation of real object files. To use this option, you must specify `-ipo`.	OFF (`-ipo_obj-`)

`-ipo_S`	Generates a multifile assembly file (`ipo_out.s`) that can be used in further link steps.	OFF
`-ipo_separate` (***new**)	Tells the compiler to generate one object file per source file. This option overrides any `-ipo`*n* specification.	OFF
`-ivdep_parallel` (i64 only)	Tells the compiler that there is no loop-carried memory dependency in any loop following an IVDEP directive.	OFF
`-Kpic`	This is a deprecated option; it can also be specified as `-KPIC`. Use `-fpic` instead.	OFF
`-L`*dir*	Tells the linker to search for libraries in *dir* before searching the standard directories.	OFF
`-lowercase`	Causes the compiler to ignore case differences in identifiers and to convert external names to lowercase (same as the `-names lowercase` option).	ON
`-logo`	Displays the compiler version information.	OFF (`-nologo`)
`-mixed_str_len_arg`	Tells the compiler that the hidden length passed for a character argument is to be placed immediately after its corresponding character argument in the argument list. The default places the hidden lengths in sequential order at the end of the argument list.	OFF (`-nomixed_str_len_arg`)
`-module` *dir*	Specifies the directory *dir* where module (`.mod`) files should be placed when created and where they should be searched for (USE statement).	OFF

`-mp`	Maintains floating-point precision (while disabling some optimizations). Restricts optimization to maintain declared precision and to ensure that floating-point arithmetic conforms more closely to the ANSI* and IEEE standards. For most programs, specifying this option adversely affects performance. If you are not sure whether your application needs this option, try compiling and running your program both with and without it to evaluate the effects on both performance and precision.	OFF
`-mp1`	Improves floating-point precision. This option disables fewer optimizations and has less impact on performance than `-mp`.	OFF
`-names` *`keyword`*	Specifies how source code identifiers and external names are interpreted. *`keyword`*: `as_is`, `lowercase`, `uppercase`. For details on these keywords, see your user's guide or the `ifort` man page.	`-names lowercase`
`-nbs`	Tells the compiler to treat a backslash (\) as a normal character, not an escape character (same as the `-assume nobscc` option).	ON
`-no_cpprt`	Prevents linking of the C++ run-time libraries.	OFF
`-noalign`	Prevents the alignment of data items. This is the same as specifying `-align none`.	OFF (`-align`)
`-noaltparam`	Specifies that the alternate form of parameter constant declarations should not be recognized.	OFF (`-altparam`)

`-nobss_init`	Places any variables that are explicitly initialized with zeros in the DATA section. By default, variables explicitly initialized with zeros are placed in the BSS section.	OFF
`-nodefaultlibs`	Prevents the compiler from using standard libraries when linking.	OFF
`-nodefine`	Specifies that all preprocessor definitions apply only to `-fpp` and not to Intel Fortran conditional compilation directives.	OFF
`-nodps`	Specifies that the alternate form of parameter constant declarations (without parentheses) should not be recognized (same as the `-noaltparam` option).	OFF (`-dps`)
`-nofor_main`	Specifies the main program is not written in Fortran, and prevents the compiler from linking `for_main.o` into applications.	OFF (`-for_main`)
`-noinclude`	Prevents the compiler from searching in `/usr/include` for files specified in an INCLUDE statement. You can specify the `-I`*`dir`* option along with this option. This option does not affect `cpp(1)` behavior, and is not related to the Fortran 95/90 USE statement.	OFF
`-nolib_inline`	Disables inline expansion of intrinsic functions.	OFF
`-nostartfiles`	Prevents the compiler from using standard startup files when linking.	OFF
`-nostdinc`	Removes standard directories from the include file search path (same as the `-X` option).	OFF
`-nostdlib`	Prevents the compiler from using standard libraries and startup files when linking.	OFF

`-nus`	Prevents the compiler from appending an underscore character to external user-defined names. This option is the same as the `-assume nounderscore` option, and is the opposite of `-us`.	ON
`-o`*`name`*	Specifies the *name* for an output file as follows: If `-c` is specified, `-o` specifies the name of an object file. If `-S` is specified, `-o` specifies the name of an assembly listing file. Otherwise, `-o` specifies the name of the executable file.	OFF
`-O0`	Disables `-O`*`n`* optimizations. On IA-32 and Intel® EM64T systems, this option sets the `-fp` option.	OFF
`-O1`	On IA-32 and Intel® EM64T systems, enables optimizations for speed. Also disables intrinsic recognition and the `-fp` option. This option is the same as the `-O2` option. On Itanium-based systems, enables optimizations for server applications (straight-line and branch-like code with flat profile). Enables optimizations for speed, while being aware of code size. For example, this option disables software pipelining and loop unrolling.	OFF

`-O2` or `-O`	This option is the default for optimizations. However, if `-g` is specified, the default is `-O0`. On IA-32 and Intel® EM64T systems, this option is the same as the `-O1` option. On Itanium-based systems, the `-O2` option enables optimizations for speed, including global code scheduling, software pipelining, predication, and speculation. On these systems, it enables inlining of intrinsics. It also enables the following capabilities for performance gain: constant propagation, copy propagation, dead-code elimination, global register allocation, global instruction scheduling and control speculation, loop unrolling, optimized code selection, partial redundancy elimination, strength reduction/induction variable simplification, variable renaming, exception handling optimizations, tail recursions, peephole optimizations, structure assignment lowering and optimizations, and dead store elimination.	ON

`-O3`	Enables `-O2` optimizations plus more aggressive optimizations, such as prefetching, scalar replacement, and loop transformations. Enables optimizations for maximum speed, but does not guarantee higher performance unless loop and memory access transformation take place. On IA-32 and Intel® EM64T systems, when the `-O3` option is used with the `-ax` and `-x` options, it causes the compiler to perform more aggressive data dependency analysis than for `-O2`, which may result in longer compilation times. On Itanium-based systems, the `-O3` option enables optimizations for technical computing applications (loop-intensive code): loop optimizations and data prefetch.	OFF
`-onetrip`	Executes at least one iteration of DO loops (same as the `-1` option). This option has the same effect as `-f66`.	OFF
`-openmp`	Enables the parallelizer to generate multithreaded code based on OpenMP* directives. The code can be executed in parallel on both uniprocessor and multiprocessor systems. The `-openmp` option works with both `-O0` (no optimization) and any optimization level of `-O`*n*. Specifying `-O0` with `-openmp` helps to debug OpenMP applications.	OFF

`-openmp_profile` (***new**)	Enables analysis of OpenMP* applications. To use this option, you must have Thread Profiler installed, which is one of the Intel® Threading Tools. If this threading tool is not installed, this option has no effect.	OFF
`-openmp_report[`*n*`]`	Controls the OpenMP parallelizer's level of diagnostic messages. *n*=0, 1, 2. `0` - Displays no diagnostic information. `1` - Displays diagnostics indicating loops, regions, and sections successfully parallelized. `2` - Displays the diagnostics specified by `-openmp_report1` plus diagnostics indicating successful handling of MASTER constructs, SINGLE constructs, CRITICAL constructs, ORDERED constructs, ATOMIC directives, etc..	`-openmp_report1`
`-openmp_stubs`	Tells the compiler to generate sequential code. The OpenMP directives are ignored and a stub OpenMP library is linked.	OFF
`-opt_report`	Tells the compiler to generate an optimization report to `stderr`.	OFF
`-opt_report_file` *`file`*	Tells the compiler to generate an optimization report and name it *`file`*.	OFF
`-opt_report_help`	Lists the logical names of optimizers available for report generation (for `-opt_report_phase`).	OFF
`-opt_report_level`*`level`*	Specifies the detail level of the optimization report. *`level`* can be `min`, `med`, or `max`.	`-opt_report_levelmin`

`-opt_report_phase`*`phase`*	Specifies the optimizer phase to generate reports for. The compiler generates reports for the optimizer you specify in *`phase`*. This option can be used multiple times on the same command line to generate reports for multiple optimizers. Currently, the following optimizer reports are supported: `ipo` • Interprocedural Optimizer `hlo` • High Level Optimizer `ilo` • Intermediate Language Scalar Optimizer `ecg` • Code Generator `omp` • Open MP `all` • All phases When one of the above logical names for optimizers is specified for *`phase`*, all reports from that optimizer are generated.	OFF
`-opt_report_routine` [*`substring`*]	Generates a report on the routines containing the specified *`substring`*. If *`substring`* is not specified, reports from all routines are generated.	OFF
`-p`	Compiles and links for function profiling with `gprof(1)`. This is the same as specifying `-pg` or `-qp`.	OFF
`-P`	Causes the Fortran preprocessor to send output to a file (same as the `-preprocess_only` and `-F` options). To use this option, you must also specify `-fpp`.	OFF
`-pad`	Enables the changing of the variable and array memory layout.	OFF (`-nopad`)
`-pad_source`	Specifies that fixed-form source records shorter than the statement field width are to be padded with spaces (on the right) to the end of the statement field. This affects the interpretation of character and Hollerith literals that are continued across source records.	OFF (`-no pad_source`)

`-par_report[`*n*`]`	Controls the auto-parallelizer's level of diagnostic messages. *n*=0, 1, 2, 3. `0` - Displays no diagnostic information. `1` - Displays diagnostics indicating loops successfully auto-parallelized. Issues a "LOOP AUTO-PARALLELIZED" message for parallel loops. `2` - Displays diagnostics indicating loops successfully auto-parallelized, as well as unsuccessful loops. `3` - Displays the diagnostics specified by `-par_report2` plus additional information about any proven or assumed dependencies inhibiting auto-parallelization (reasons for not parallelizing).	`-par_report1`
`-par_threshold`*n*	Sets a threshold for the auto-parallelization of loops based on the probability of profitable execution of the loop in parallel. This option is used for loops whose computation work volume cannot be determined at compile-time. The threshold is usually relevant when the loop trip count is unknown at compile-time. n=0 to 100. The compiler applies a heuristic that tries to balance the overhead of creating multiple threads versus the amount of work available to be shared amongst the threads.	`-par_threshold100`
`-parallel`	Tells the auto-parallelizer to generate multithreaded code for loops that can be safely executed in parallel. To use this option, you must also specify `-O2` or `-O3`.	OFF

-pc*n* (i32, i32em)	Enables control of floating-point significand precision. Some floating-point algorithms are sensitive to the accuracy of the significand, or fractional part of the floating-point value. For example, iterative operations like division and finding the square root can run faster if you lower the precision with the -pc*n* option. *n*=32, 64, 80. 32 - Rounds the significand to 24 bits 64 - Rounds the significand to 53 bits 80 - Rounds the significand to 64 bits	-pc64
-pg	Compiles and links for function profiling with gprof(1). This is the same as specifying -p or -qp.	OFF
-prec_div (i32, i32em)	Improves precision of floating-point divides; it has some speed impact. With some optimizations, such as -xN and -xB, the compiler may change floating-point division computations into multiplication by the reciprocal of the denominator. For example, A/B is computed as A * (1/B) to improve the speed of the computation. However, for values of B greater than 2126, the value of 1/B is "flushed" (changed) to 0. When it is important to maintain the value of 1/B, use -prec_div to disable the floating-point division-to-multiplication optimization. The result of -prec_div is more accurate, with some loss of performance.	OFF
-prefetch- (i32 only)	Disables prefetch insertion optimization. To use this option, you must also specify -O3.	OFF (-prefetch)

`-preprocess_only`	Causes the Fortran preprocessor to send output to a file (same as the `-F` and `-P` options). To use this option, you must also specify `-fpp`.	OFF
`-prof_dir`*`dir`*	Specifies a directory (*`dir`*) for profiling output files `*.dyn` and `*dpi`.	OFF
`-prof_file`*`file`*	Specifies a file name (*`file`*) for the profiling summary file.	OFF
`-prof_format_32` (i32, i64)	Produces profile data with 32-bit counters; allows compatibility with earlier compilers. The default is to produce profile data with 64-bit counters to handle large numbers of events.	OFF
`-prof_gen`	Instruments a program for profiling.	OFF
`-prof_use`	Enables use of profiling information during optimization.	OFF
`-Qinstall`*`dir`*	Sets *`dir`* as a root directory for compiler installation.	OFF
`-Qlocation,`*`str`*`,`*`dir`*	Sets *`dir`* as the location of the tools specified by *`str`*.	OFF
`-Qoption,`*`str`*`,`*`opts`*	Passes options (*`opts`*) to the tools specified by *`str`*, which can be `fpp`, `f`, `c`, `link`, `asm` (on IA-32 systems), or `ias` (on Itanium-based systems).	OFF
`-qp`	Compiles and links for function profiling with `gprof(1)`. This is the same as specifying `-p` or `-pg`.	OFF
`-r8`	Defines REAL declarations, constants, functions, and intrinsics as DOUBLE PRECISION REAL*8, and defines COMPLEX declarations, constants, functions, and intrinsics as DOUBLE COMPLEX (COMPLEX*16). This option is the same as specifying `-real_size 64` or `-autodouble`.	OFF

`-r16`	Defines REAL and DOUBLE PRECISION declarations, constants, functions, and intrinsics as REAL*16, and defines COMPLEX and DOUBLE COMPLEX declarations, constants, functions, and intrinsics as COMPLEX*32. This option is the same as specifying `-real_size 128`.	OFF
`-rcd` (i32 only)	Enables fast float-to-integer conversions. Disables the change to truncation of the rounding mode for all floating-point calculations, including floating point-to-integer conversions. This option can improve performance, but floating-point conversions to integer will not conform to Fortran semantics. For more details, see your user's guide or the `ifort` man page.	OFF
`-real_size` *`size`*	Defines the size of REAL and COMPLEX declarations, constants, functions, and intrinsics. *`size`* can be 32, 64, or 128.	`-real_size 32`
`-recursive`	Specifies that all routines should be compiled for possible recursive execution. This option sets the `-auto` option.	OFF (`-norecursive`)
`-reentrancy` *`keyword`*	Specifies that the compiler should generate reentrant code that supports a multithreaded application. *`keyword`*: `none`, `threaded`, `async`. For details on these keywords, see your user's guide or the `ifort` man page.	OFF (`-reentrancy none`)
`-S`	Causes the compiler to compile to an assembly file (`.s`) only and not link.	OFF
`-safe_cray_ptr`	Specifies that CRAY* pointers do not alias other variables.	OFF

`-save`	Places variables, except those declared as AUTOMATIC, in static memory (same as `-noauto` or `-noautomatic`). If you specify `-recursive` or `-openmp`, the default is `-auto`.	OFF (`-auto_scalar`)
`-scalar_rep` (i32 only)	Enables scalar replacement performed during loop transformation. To use this option, you must also specify `-O3`.	OFF (`-scalar_rep-`)
`-shared`	Instructs the compiler to build a Dynamic Shared Object (DSO) instead of an executable. On Itanium-based systems, you must specify `-fpic` for the compilation of each object file you want to include in the shared library.	OFF
`-shared-libcxa`	Links the Intel `libcxa` C++ library dynamically, overriding the default behavior when `-static` is used. This option is the opposite of `-static-libcxa`.	ON
`-sox[-]` (i32, i32em)	Tells the compiler to save the compiler options and version in the executable.	OFF (`-sox-`)
`-stand [`*`keyword`*`]`	Causes the compiler to issue compile-time messages for nonstandard language elements. *`keyword`*: `f90`, `f95`, `none`. For details on these keywords, see your user's guide or the `ifort` man page.	OFF (`-stand none`)
`-static`	Prevents linking with shared libraries. Causes the executable to link all libraries statically.	OFF
`-static-libcxa`	Links the Intel `libcxa` C++ library statically. This option is the opposite of `-shared-libcxa`.	OFF
`-std90`	Causes the compiler to issue messages for language elements that are not standard in Fortran 90 (same as the `-stand f90` option).	OFF

`-std95` or `-std`	Causes the compiler to issue messages for language elements that are not standard in Fortran 95 (same as the `-stand f95` option). This option is set if you specify `-warn stderrors`.	OFF
`-syntax_only`	Specifies that the source file should be checked only for correct syntax (same as the `-syntax` and `-y` options). No code is generated, no object file is produced, and some error checking done by the optimizer is bypassed. This option lets you do a quick syntax check of your source file.	OFF (`-no syntax_only`)
`-T` *`file`*	Tells the linker to read link commands from the specified *`file`*.	OFF
`-tcheck` (***new**)	Enables analysis of threaded applications. To use this option, you must have Intel® Thread Checker installed, which is one of the Intel® Threading Tools. If this threading tool is not installed, this option has no effect.	OFF
`-Tf`*`file`*	Specifies that *`file`* should be compiled as a Fortran source file. This option is useful when you have a file with a nonstandard filename suffix.	OFF
`-threads`	Specifies that multithreaded libraries should be linked. This option sets the `-reentrancy threaded` option.	OFF (`-nothreads`)
`-tpp{1|2}` (i64 only)	`-tpp1` - Optimizes for the Intel® Itanium® processor. `-tpp2` - Optimizes for the Intel® Itanium® 2 processor.	`-tpp2`

`-tpp{5|6|7}` (i32, i32em)	`-tpp5` - Optimizes for the Intel Pentium® processor. `-tpp6` - Optimizes for the Intel Pentium Pro, Pentium II, and Pentium III processors. `-tpp7` - Optimizes for the Intel® Pentium® 4 processors, Intel® Xeon(TM) processors, Intel® Pentium® M processors, and Intel® Pentium® 4 processor with Streaming SIMD Extensions 3 (SSE3) instruction support. The only option available on Intel® EM64T systems is `-tpp7`.	`-tpp7`
`-traceback`	Tells the compiler to generate extra information in the object file to allow the display of source file traceback information at run time when a severe error occurs.	OFF (`-notraceback`)
`-tune` *keyword* (i32 only)	Determines the version of the architecture for which the compiler generates instructions. Indicated by *keyword*: `pn1` - Optimizes for the Intel® Pentium® processor. `pn2` - Optimizes for the Intel® Pentium® Pro, Intel® Pentium® II, and Intel® Pentium® III processors. `pn3` - Optimizes for the Intel® Pentium® Pro, Intel® Pentium® II, and Intel® Pentium® III processors. This is the same as specifying the `-tune pn2` option. `pn4` - Optimizes for the Intel® Pentium® 4 processor.	`-tune pn4`
`-u`	Sets the default type of a variable to undefined (IMPLICIT NONE). This is the same as specifying the `-implicitnone` option.	ON
`-U`*name*	Removes the predefined macro *name*.	OFF

`-unroll[`*n*`]`	Sets the maximum number of times to unroll loops. *n*=0 (`-unroll0`) disables loop unrolling. If you omit *n*, the compiler uses default heuristics. On Itanium-based systems, the compiler currently recognizes only *n*=`0`; any other value is ignored.	`-unroll`
`-uppercase`	Causes the compiler to ignore case differences in identifiers and to convert external names to uppercase (same as the `-names uppercase` option).	OFF (`-lowercase` or `-names lowercase`)
`-us`	Tells the compiler to append an underscore character to external user-defined names. This option is the same as the `-assume underscore` option, and is the opposite of `-nus`.	OFF
`-use_asm`	Tells the compiler to produce objects through the assembler.	OFF
`-V`	Displays the compiler version information.	OFF
`-v`	Tells the driver that tool commands should be shown and executed.	OFF
`-vec_report[`*n*`]` (i32, i32em)	Controls amount of vectorizer diagnostic information. *n*=1, 2, 3, 4, or 5: `0` - Produces no information `1` - Indicates vectorized loops `2` - Indicates vectorized and non-vectorized loops `3` - Indicates vectorized and non-vectorized loops and prohibiting data dependence information `4` - Indicates non-vectorized loops `5` - Indicates non-vectorized loops and prohibiting data dependence information.	`-vec_report1`

`-vms`	Causes the run-time system to behave like HP* Fortran for OpenVMS Alpha systems and VAX systems (VAX FORTRAN*) in certain ways. For details on the affect of this option, see your user's guide or the `ifort` man page.	OFF
`-w`	Disables all warning messages (same as the `-nowarn` and `-warn nogeneral` options).	OFF
`-W`*n*	Disables warnings (n =0) or enables warnings (n =1). `-W1` is the same as specifying `-warn general`. `-W0` is the same as specifying `-warn nogeneral`, `-nowarn`, or `-w`.	`-W1`
`-w90` or `-w95`	Suppresses warning messages about Fortran features that are deprecated or obsolescent in Fortran 95.	OFF
`-warn` *keyword*	Specifies the level of warning messages issued by the compiler. *keyword*: `all`, `none`, `[no]alignments`, `[no]declarations`, `[no]errors`, `[no]general`, `[no]ignore_loc`, `[no]stderrors`, `[no]truncated_source`, `[no]uncalled`, `[no]unused`, `[no]usage`. For details on these keywords, see your user's guide or the `ifort` man page.	OFF (`-warn none` or `-nowarn`)
`-what`	Displays the version strings of the Fortran command and the compiler.	OFF
`-Wl o1 [,o2,...]`	Passes options `-o1`, `-o2`, etc. to the linker for processing.	OFF
`-Wp o1 [,o2,...]`	Passes options `-o1`, `-o2`, etc. to the preprocessor.	OFF

-X	Removes standard directories from the include file search path (same as the -nostdinc option). You can use the -X option with the -I option to prevent the compiler from searching the default path for include files and direct it to use an alternate path.	OFF
-x{K|W|N|B|P} (i32, i32em)	Generates specialized and optimized code for the processor that executes your program. The processor type is indicated by one of the following values: K - Intel® Pentium® III processors and compatible Intel processors. W - Intel® Pentium® 4 processors and compatible Intel processors. N - Intel® Pentium® 4 processors and compatible Intel processors B - Intel® Pentium® M and compatible Intel processors P - Intel® Pentium® 4 processor with Streaming SIMD Extensions 3 (SSE3) instruction support. When the main program is compiled with the -xN, -xB, or -xP option, it will detect non-compatible processors and generate a fatal error message during execution. These options also enable new optimizations in addition to Intel processor-specific optimizations. The only options available on Intel® EM64T systems are -xP and -xW. For more details, see your user's guide or the ifort man page.	OFF

`-Xlinker` *`value`*	Passes *`value`* directly to the linker for processing.	OFF
`-y`	Specifies that the source file should be checked only for correct syntax (same as the `-syntax_only` and `-syntax` options).	OFF
`-zero`	Initializes to zero all local scalar variables of intrinsic type INTEGER, REAL, COMPLEX, or LOGICAL that are saved but not yet initialized. Use `-save` on the command line to make all local variables specifically marked as SAVE.	OFF (`-zero-`)
`-Zp[`*`n`*`]`	Aligns fields of records and components of derived types on the smaller of the size boundary specified or the boundary that will naturally align them (same as the `-align rec`*`n`*`byte` option). *n* can be:1, 2, 4, 8, or 16. If you do not specify *n*, you get `-Zp16`.	`-Zp16`

Compiler Options Cross Reference

This section provides a cross-reference table of the Intel® Fortran compiler options used on the Windows* and Linux* operating systems.

Some compiler options are only available on certain systems, as indicated by these labels:

Label	Meaning
i32	The option is available on IA-32-based systems
i32em	The option is available on IA-32-based systems with Intel® Extended Memory 64 Technology (Intel® EM64T)
i64	The option is available on Itanium®-based systems

If no label appears, the option is available on all supported systems.

If "only" appears in the label, the option is only available on the identified system.

For more details on the Linux options refer to the Alphabetical Quick Reference Guide, the *Intel® Fortran Compiler User's Guide*, Volumes I and II, and the `ifort` man page. For more details on the Windows options refer to the Windows documentation for the Intel Fortran Compiler.

The following table is based on the alphabetical order of Linux compiler options, which appear in the first column.

For information on conventions used in this table, see Notation Conventions.

Linux Option	Windows Option	Description	Default
`-1`	`/1`	Executes at least one iteration of DO loops.	OFF
`-66`	None	Tells the compiler to use FORTRAN-66 semantics.	OFF
`-72`, `-80`, `-132`	`/4L{72|80|132}`	Treats the statement field of each fixed-form source line as ending in column 72, 80, or 132.	`-72` `/4L72`

`-align` *keyword*	`/align:`*keyword*	Tells the compiler how to align data items.	*keyword*s:
			`nocommons`
			`nodcommons`
			`records`
			`rec8byte`
			`nosequence`

-ansi_alias[-]	/Qansi_alias[-]	Determines whether the compiler assumes the program adheres to the Fortran 95 Standard type aliasability rules.	ON
-arch *keyword* (i32 only)	/arch:*keyword* (i32 only)	Determines the version of the architecture for which the compiler generates instructions.	*keyword*: pn4
-assume *keyword*	/assume:*keyword*	Specifies assumptions made by the compiler.	OFF
-auto	/Qauto	Places variables, except those declared as SAVE, on the run-time stack. The default setting can be affected by other compiler options.	OFF
-auto_ilp32 (i32em, i64)	/Qauto_ilp32 (i32em, i64)	Specifies that the application cannot exceed a 32-bit address space.	OFF
-auto_scalar	/Qauto_scalar	Makes AUTOMATIC all scalar local variables of intrinsic type INTEGER, REAL, COMPLEX, or LOGICAL. The default setting can be affected by other compiler options.	ON
-autodouble	/Qautodouble	Defines real variables to be REAL(KIND=8).	OFF
-automatic	/automatic	Places variables, except those declared as SAVE, on the run-time stack.	OFF
-ax*p* (i32, i32em)	/Qax*p* (i32, i32em)	Generates processor-specific code if there is a performance benefit. The *p* indicates the processor type.	OFF
-Bdynamic	None	Enables dynamic linking of libraries at run time.	OFF
-Bstatic	None	Enables static linking of a user's library.	OFF
-c	/c	Causes the compiler to compile to an object file only and not link.	OFF

-CB	/CB	Performs run-time checks on whether array subscript and substring references are within declared bounds.	OFF
-ccdefault *keyword*	/ccdefault:*keyword*	Specifies the type of carriage control used for units 6 and *. The default setting can be affected by other compiler options.	*keyword*: default
-check *keyword*	/check:*keyword*	Checks several run-time conditions.	OFF
-common_args	/Qcommon_args	Tells the compiler that dummy (formal) arguments to procedures share memory locations with other dummy arguments or with COMMON variables that are assigned.	OFF
-complex_limited_range	/Qcomplex_limited_range	Enables the use of basic algebraic expansions of some arithmetic operations involving data of type COMPLEX.	OFF
-convert *keyword*	/convert:*keyword*	Specifies the format of unformatted files containing numeric data.	OFF
-cpp	/Qcpp	Runs the Fortran preprocessor on source files prior to compilation.	OFF
-D*name*[=*value*]	/D*name*[=*value*]	Defines a *name*, and optional *value*, as a definition to use with conditional compilation directives or the Fortran preprocessor.	OFF
-d_lines	/d_lines	Compiles debugging statements indicated by the letter D in column 1 of the source code.	OFF
-DD	/Qd_lines	Compiles debugging statements indicated by the letter D in column 1 of the source code.	OFF
-debug *keyword*	None	Specifies settings that enhance debugging.	OFF

-double_size *size*	/double_size:*size*	Defines the size of DOUBLE PRECISION and DOUBLE COMPLEX declarations, constants, functions, and intrinsics.	*size*: 64
-dryrun	None	Specifies that driver tool commands should be shown but not executed.	OFF
-dynamic -linker*file*	None	Specifies a dynamic linker in *file* other than the default.	OFF
-dyncom "a,b,c"	/Qdyncom:A,B,C	Enables dynamic allocation of the specified COMMON blocks at run time.	OFF
-E	/E	Causes the Fortran preprocessor to send output to stdout.	OFF
-e95, -e90	/4Ys	Causes the compiler to issue errors instead of warnings for nonstandard Fortran 95 or Fortran 90.	OFF
-EP	/EP	Causes the Fortran preprocessor to send output to stdout, omitting #line directives.	OFF
-error_limit *n*	/error_limit:*n*	Specifies the maximum number of error-level or fatal-level compiler errors allowed for a file specified on the command line.	*n*: 30
-extend_source *size*	/Qextend_source: *size*	Specifies the column number to use to end the statement field in fixed-form source files.	*size*: 72
-F	None	Causes the Fortran preprocessor to send output to a file (requires -fpp).	OFF
-f66	/f66	Tells the compiler to use FORTRAN-66 semantics.	OFF
-f77rtl	/f77rtl	Tells the compiler to use Fortran 77 run-time behavior.	OFF
-fast	/fast	Maximizes speed across the entire program.	OFF

`-fcode-asm`	`/FAc`	Produces an assembly file with optional machine code annotations.	OFF
`-FI`	`/FI`	Specifies source files are in fixed format.	determined by file suffix
`-fixed`	`/fixed`	Specifies source files are in fixed format.	determined by file suffix
`-fltconsistency`	`/fltconsistency`	Enables improved floating-point consistency.	OFF
`-fminshared`	None	Tells the compiler to treat a compilation unit as a component of a main program and not to link it as a shareable object.	OFF
`-fno-alias`	None	Specifies that aliasing should not be assumed in the program.	`-falias`
`-fno-fnalias`	None	Specifies that aliasing should not be assumed within functions, but should be assumed across calls.	`-ffnalias`
`-fnsplit` (i64 only)	`/Qfnsplit` (i64 only)	Enables function splitting.	OFF
`-fp` (i32, i32em)	`/Oy-` (i32 only)	Disables using EBP as a general purpose register so it can be used as a stack frame printer.	OFF
`-fp_port` (i32 only)	`/Qfp_port` (i32 only)	Rounds floating-point results after floating-point operations, so rounding to user-declared precision happens at assignments and type conversions (some impact on speed).	OFF
`-fpconstant`	`/fpconstant`	Tells the compiler to extend the precision to double precision for single-precision constants assigned to double-precision variables.	OFF
`-fpe`*n*	`/fpe:`*n*	Specifies floating-point exception handling at run time for the main program.	`-fpe3` `/fpe:3`
`-fpic, -fPIC`	None	Generates position-	OFF

		independent code.	
-fpp[*n*]	/Qfpp[:*n*]	Runs the Fortran preprocessor on source files prior to compilation.	OFF
-fpscomp *keyword*	/fpscomp[:*keyword*]	Specifies the level of compatibility with Microsoft* Fortran PowerStation or Intel® Fortran.	*keyword*: libs
-fpstkchk (i32 only)	/Qfpstkchk (i32 only)	Generates extra code after every function call to ensure that the FP (floating-point) stack is in the expected state.	OFF
-FR	/FR	Specifies source files are in free format.	determined by file suffix
-fr32 (i64 only)	None	Disables use of high floating-point registers.	OFF
-free	/free	Specifies source files are in free format.	determined by file suffix
-fsource-asm	/FAs	Produces an assembly file with optional source code annotations.	OFF
-ftz	/Qftz	Enables floating underflow results set to zero.	OFF
-fverbose-asm	None	Produces an assembly file with compiler comments, including options and version information.	OFF
-fvisibility=*keyword* -fvisibility-*keyword=file*	None	Specifies the default visibility for global symbols; 2nd form indicates symbols in a file.	OFF
-g	/Zi, /Z7	Produces symbolic debug information in the object file.	OFF
-help	/help	Displays the list of compiler options.	OFF
-I*dir*	/I*dir*	Specifies a directory where the compiler can search for module files and include files.	OFF
-i_dynamic	None	Links Intel-provided libraries dynamically.	OFF

`-i{2|4|8}`	`/4I{2|4|8}`	Defines the default KIND in bytes for integer variables and constants.	`-i4` `/4I4`
`-implicitnone`	None	Sets the default type of a variable to undefined.	OFF
`-inline _debug_info`	`/Qinline_ debug_info`	Produces enhanced source position information for inlined code.	OFF
`-intconstant`	`/intconstant`	Tells the compiler to use Fortran 77 semantics to determine the KIND for integer constants.	OFF
`-integer_size` *`size`*	`/integer_size:`*`size`*	Defines the size of INTEGER and LOGICAL variables.	*`size`*: 32
`-ip`	`/Qip`	Enables single-file interprocedural optimizations.	OFF
`-ip_no_inlining`	`/Qip_no _inlining`	Disables full and partial inlining enabled by `-ip`.	OFF
`-ip_no_pinlining`	`/Qip_no _pinlining`	Disables partial inlining.	OFF
`-IPF_flt_eval_ method0` (i64 only)	`/QIPF_flt_eval_ method0` (i64 only)	Tells the compiler to evaluate the expressions involving floating-point operands in the precision indicated by the variable types declared in the program.	OFF
`-IPF_fltacc` (i64 only)	`/QIPF_fltacc` (i64 only)	Disables optimizations that affect floating-point accuracy.	OFF
`-IPF_fma` (i64 only)	`/QIPF_fma` (i64 only)	Enables the combining of floating-point multiplies and add/subtract operations. This option can be affected by other compiler options.	OFF
`-IPF_fp_relaxed` (i64 only)	`/QIPF_fp_relaxed` (i64 only)	Enables use of faster but slightly less accurate code sequences for math functions, such as divide and sqrt.	OFF
`-IPF_fp_ speculation`*`mode`* (i64 only)	`/QIPF_fp_ speculation`*`mode`* (i64 only)	Enables or disables floating-point speculations.	*`mode`*: `fast`
`-ipo[`*`n`*`]`	`/Qipo[`*`n`*`]`	Enables multifile IP optimizations between files.	OFF

`-ipo_c`	`/Qipo_c`	Generates a multifile object file that can be used in further link steps.	OFF
`-ipo_obj`	`/Qipo_obj`	Forces the generation of real object files. Requires `-ipo`.	OFF
`-ipo_S`	`/Qipo_S`	Generates a multifile assembly file that can be used in further link steps.	OFF
`-ipo_separate`	`/Qipo_separate`	Generates one object file per source file.	OFF
`-ivdep_parallel` (i64 only)	`/Qivdep_parallel` (i64 only)	Tells the compiler that there is no loop-carried memory dependency in any loop following an IVDEP directive.	OFF
`-Kpic, -KPIC`	None	This is a deprecated option; use `-fpic` instead.	OFF
`-L`*`dir`*	None	Tells the linker where to search for libraries before searching the standard directories.	OFF
`-lowercase`	`/Qlowercase`	Causes the compiler to ignore case differences in identifiers and to convert external names to lowercase.	Linux: ON Windows: OFF
`-logo`	`/logo`	Displays compiler version information.	Linux: OFF Windows: ON
`-mixed_str_len_arg`	`/iface:mixed_str_len_arg`	Tells the compiler that the hidden length passed for a character argument is to be placed immediately after its corresponding character argument in the argument list.	OFF
`-module` *`dir`*	`/module:`*`path`*	Specifies the directory where module files should be placed when created and where they should be searched for.	OFF
`-mp`	`/Op`	Maintains floating-point precision while disabling some optimizations; can adversely affect performance.	OFF
`-mp1`	`/Qprec`	Improves floating-point	OFF

		precision; disables fewer optimizations and has less impact on performance than `-mp`.	
`-names` *keyword*	`/names:`*keyword*	Specifies how source code identifiers and external names are interpreted.	OFF
`-nbs`	`/nbs`	Tells the compiler to treat a backslash as a normal character, not as an escape character.	ON
`-no_cpprt`	None	Prevents linking of the C++ run-time libraries.	OFF
`-noalign`	`/align:none`	Prevents the alignment of data items.	OFF
`-noaltparam`	`/noaltparam`	Specifies that the alternate form of parameter constant declarations (without parentheses) should not be recognized.	OFF
`-nobss_init`	`/Qnobss_init`	Places any variables that are explicitly initialized with zeros in the DATA section.	OFF
`-nodefaultlibs`	None	Prevents the compiler from using standard libraries when linking.	OFF
`-nodefine`	`/nodefine`	Specifies that all preprocessor definitions apply only to `-fpp` and not to conditional compilation directives.	OFF
`-nodps`	`/Qdps-`	Specifies that the alternate form of parameter constant declarations (without parentheses) should not be recognized.	OFF
`-nofor_main`	None	Specifies the main program is not written in Fortran, and prevents the compiler from linking `for_main.o` into applications.	OFF
`-noinclude`	`/noinclude`	Prevents the compiler from searching in a directory	OFF

		previously added to the include path for files specified in an INCLUDE statement.	
-nolib_inline	/Oi-	Disables inline expansion of intrinsic functions.	OFF
-nostartfiles	None	Prevents the compiler from using standard startup files when linking.	OFF
-nostdinc	None	Removes standard directories from the include file search path.	OFF
-nostdlib	None	Prevents the compiler from using standard libraries and startup files when linking.	OFF
-nus	None	Disables appending an underscore to external subroutine names.	ON
-o*name*	None	Specifies the name for an output file.	OFF
-O0	/Od	Disables all -O*n* optimizations.	OFF
-O1	/O1	On IA-32 systems, optimizes for speed. On Itanium-based systems, optimizes for server applications, and enables optimizations for speed, while being aware of code size.	OFF
-O2	/O2	Optimizes for speed. The default setting can be affected by other compiler options.	ON
-O3	/O3	Enables -O2 optimizations plus more aggressive optimizations.	OFF
-onetrip	/Qonetrip	Executes at least one iteration of DO loops.	OFF
-openmp	/Qopenmp	Enables the parallelizer to generate multithreaded code based on OpenMP* directives.	OFF
-openmp_profile	/Qopenmp_profile	Enables analysis of OpenMP* applications.	OFF

-openmp_report[*n*]	/Qopenmp_report*n*	Controls the OpenMP parallelizer's level of diagnostic messages.	*n*: 1
-openmp_stubs	/Qopenmp_stubs	Tells the compiler to generate sequential code.	OFF
-opt_report	/Qopt_report	Tells the compiler to generate an optimization report to stderr.	OFF
-opt_report_file*file*	/Qopt_report_file*file*	Tells the compiler to generate an optimization report named *file*.	OFF
-opt_report_help	/Qopt_report_help	Displays the logical names of optimizers available for report generation (using -opt_report_phase).	OFF
-opt_report_level*level*	/Qopt_report_level*level*	Specifies the detail level of the optimization report.	*level*: min
-opt_report_phase*phase*	/Qopt_report_phase*phase*	Specifies the optimizer phase to generate reports for.	OFF
-opt_report_routine[*routine*]	/Qopt_report_routine*routine*	Generates a report on the routines containing the specified routine.	OFF
-p	None	Compiles and links for function profiling with gprof(1).	OFF
-P	/P	Causes the Fortran preprocessor to send output to a file (requires -fpp)	OFF
-pad	/Qpad	Enables the changing of the variable and array memory layout.	OFF
-pad_source	/pad_source	Specifies that fixed-form source records shorter than the statement field width should be padded with spaces (on the right) to the end of the statement field.	OFF
-par_report[*n*]	Qpar_report*n*	Controls the auto-parallelizer's level of diagnostic messages.	*n*: 1

`-par_threshold[`*n*`]`	`/Qpar_threshold[:`*n*`]` or `/Qpar_threshold[`*n*`]`	Sets a threshold for the auto-parallelization of loops based on the probability of profitable execution of the loop in parallel.	*n*: `100`
`-parallel`	`/Qparallel`	Tells the auto-parallelizer to generate multithreaded code for loops that can be safely executed in parallel.	OFF
`-pc`*n* (i32, i32em)	`/Qpc`*n* (i32 only)	Enables control of floating-point significand precision.	*n*: `64`
`-pg`	None	Compiles and links for function profiling with `gprof(1)`.	OFF
`-prec_div` (i32, i32em)	`/Qprec_div` (i32, i32em)	Disables floating point division-to-multiplication optimization resulting in more accurate division results; some speed impact.	OFF
`-prefetch-` (i32 only)	`/Qprefetch-` (i32 only)	Disables prefetch insertion optimization (requires `-O3`).	OFF
`-preprocess_only`	None	Causes the Fortran preprocessor to send output to a file (requires `-fpp`).	OFF
`-prof_dir` *`dir`*	`/Qprof_dir` *`dir`*	Specifies a directory for profiling output files (*.dyn and *.dpi).	OFF
`-prof_file` *`file`*	`/Qprof_file` *`file`*	Specifies a file name (*`file`*) for the profiling summary file.	OFF
`-prof_format_32` (i32, i64)	`/Qprof_format_32` (i32, i64)	Produces profile data with 32-bit counters; allows compatibility with earlier compilers.	OFF
`-prof_gen`	`/Qprof_gen`	Instruments a program for profiling.	OFF
`-prof_use`	`/Qprof_use`	Enables use of profiling information during optimization.	OFF
`-Qinstall`*`dir`*	None	Sets *`dir`* as the root directory for compiler installation.	OFF

`-Qlocation,`*`str`*`,` *`dir`*	`/Qlocation,tool,` *`dir`*	Specifies a directory as the location of the specified tool.	OFF
`-Qoption,`*`str`*`,`*`opts`*	`/Qoption,tool,` *`options`*	Passes options to the specified tool.	OFF
`-qp`	None	Compiles and links for function profiling with `prof(1)`.	OFF
`-r8`	`/4R8`	Defines REAL declarations, constants, functions, and intrinsics as DOUBLE PRECISION (REAL*8), and defines COMPLEX declarations, constants, functions, and intrinsics as DOUBLE COMPLEX (COMPLEX*16).	OFF
`-r16`	`/4R16`	Defines REAL and DOUBLE PRECISION declarations, constants, functions, and intrinsics as REAL*16 and defines COMPLEX and DOUBLE COMPLEX declarations, constants, functions, and intrinsics as COMPLEX*32.	OFF
`-rcd` (i32 only)	`/Qrcd` (i32 only)	Enables fast float-to-integer conversions.	OFF
`-real size `*`size`*	`/real size:`*`size`*	Defines the size of REAL and COMPLEX declarations, constants, functions, and intrinsics.	*`size`*: 32
`-recursive`	`/recursive`	Specifies that all routines should be compiled for possible recursive execution.	OFF
`-reentrancy` *`keyword`*	`/reentrancy:` *`keyword`*	Specifies that the compiler should generate reentrant code that supports a multithreaded application.	OFF
`-S`	`/S`	Causes the compiler to compile to an assembly file (.s) only and not link.	OFF
`-safe_cray_ptr`	`/Qsafe_cray_ptr`	Specifies that Cray pointers do not alias other variables.	OFF

`-save`	`/Qsave`	Places variables, except those declared as AUTOMATIC, in static memory.	OFF
`-scalar_rep` (i32 only)	`/Qscalar_rep` (i32 only)	Enables scalar replacement performed during loop transformation (requires `-O3`).	OFF
`-shared`	None	Tells the compiler to produce a dynamic shared object instead of an executable.	OFF
`-shared-libcxa`	None	Links the Intel libcxa C++ library dynamically.	OFF
`-sox` (i32, i32em)	`/Qsox` (i32, i32em)	Tells the compiler to save the compiler options and version in the executable.	OFF
`-stand` *keyword*	`/stand:`*keyword*	Causes the compiler to issue compile-time messages for nonstandard language elements.	OFF
`-static`	`/static`	Prevents linking with shared libraries. Causes the executable to link all libraries statically.	OFF
`-static-libcxa`	None	Links the Intel libcxa C++ library statically.	OFF
`-std90` or `-stand f90`	`/stand:f90`	Causes the compiler to issue messages for language elements that are not standard in Fortran 90.	OFF
`-std95` or `-std` or `-stand f95`	`/stand:f95`	Causes the compiler to issue messages for language elements that are not standard in Fortran 95.	OFF
`-syntax_only`	`/syntax_only`	Specifies that the source file should be checked only for correct syntax.	OFF
`-T` *file*	None	Tells the linker to read link commands from the specified *file*.	OFF
`-tcheck`	`/Qtcheck`	Enables analysis of threaded applications.	OFF
`-Tf` *file*	`/Tf` *file*	Specifies that *file* should be	OFF

		compiled as a Fortran source file.	
`-threads`	`/threads`	Specifies that multithreaded libraries should be linked.	OFF
`-tpp{1|2}` (i64 only)	`/G{1|2}` (i64 only)	Optimizes for certain Intel® processors.	`-tpp2` `/G2`
`-tpp{5|6|7}` (i32, i32em)	`/G{5|6|7}` (i32, i32em)	Optimizes for certain Intel® processors.	`-tpp7` `/G7`
`-traceback`	`/traceback`	Tells the compiler to generate extra information in the object file to allow the display of source file traceback information at run time when a severe error occurs.	OFF
`-tune` *keyword* (i32 only)	None	Determines the version of the architecture for which the compiler generates instructions.	*keyword*: `pn4`
`-u`	None	Sets the default type of a variable to undefined (IMPLICIT NONE).	OFF
`-U`*name*	`/U`*name*	Removes the predefined macro named *name*.	OFF
`-unroll[`*n*`]`	`/unroll[:`*n*`]`	Sets the maximum number of times to unroll loops.	OFF
`-uppercase`	`/uppercase`	Causes the compiler to ignore case differences in identifiers and to convert external names to uppercase.	Linux: OFF Windows: ON
`-us`	`/us`	Tells the compiler to append an underscore character to external user-defined names.	OFF
`-use_asm`	None	Tells the compiler to produce objects through the assembler.	OFF
`-V`	/logo	Displays the compiler version information.	OFF
`-v`	None	Tells the driver that tool commands should be shown and executed.	OFF
`-vec_report[`*n*`]` (i32, i32em)	`/Qvec_report[`*n*`]` (i32, i32em)	Specifies the amount of vectorizer diagnostic information to report.	*n*: `1`

-vms	/vms	Causes the run-time system to behave like HP* Fortran on OpenVMS* Alpha systems and VAX* systems (VAX FORTRAN*) in certain ways.	OFF
-w	/w	Disables all warning messages	OFF
-W*n*	/W*n*	Disables (*n*=0) or enables (*n*=0) warnings.	*n*: 1
-w90 or -w95	/w90 or /w95	Suppresses warning messages about Fortran features that are deprecated or obsolescent in Fortran 95.	OFF
-warn *keyword*	/warn:*keyword*	Specifies the level of warning messages issued by the compiler.	OFF
-what	/what	Displays the version strings of the Fortran command and the compiler.	OFF
-Wl,*o1*[,*o2*,...]	/link *o1*[,*o2*,...]	Passes options (*o1*, *o2*, and so forth) to the linker for processing.	OFF
-Wp,*o1*[,*o2*,...]	/Qoption,fpp,*o1*[,*o2*,...]	Passes options (*o1*, *o2*, and so forth) to the preprocessor.	OFF
-X	/X	Removes standard directories from the include file search path.	OFF
-x*p* (i32, i32em)	/Qx*p* (i32, i32em)	Generates the minimum set of processor-specific instructions required for the processor that executes your program. The *p* indicates the processor type.	OFF
-Xlinker *value*	/link *value*	Passes *value* directly to the linker for processing.	OFF
-y	/Zs	Specifies that the source file should be checked only for correct syntax.	OFF
-zero	/Qzero	Initializes to zero all local scalar variables of intrinsic type INTEGER, REAL, COMPLEX, or LOGICAL that	OFF

		are saved but not yet initialized.	
-Zp[*n*]	/Zp[*n*]	Aligns fields of records and components of derived types on the smaller of the size boundary specified or the boundary that will naturally align them.	*n*: 8

Deprecated Compiler Options

This version of the Intel® Fortran compiler no longer supports the following compiler options:

- -axi
- -axM
- -xi
- -xM

Index