2 Invoking the Compiler

The iWarp FORTRAN compiler is an optimizing compiler for the iWarp processor. The iWarp FORTRAN compiler is a cross-compiler that runs on the Sun host and produces code for the iWarp system. The iWarp FORTRAN compiler translates FORTRAN programs into iWarp executable load modules. The iWarp FORTRAN compiler produces object code files in Common Object File Format (COFF).

The Compilation Command: iwf77

To invoke the iWarp FORTRAN compiler, use the iwf77 command as follows:

```
iwf77 [option ...]  filename ...
```

filename can be any of the following:

- A FORTRAN source file (.f file extension).
- A FORTRAN source file with C comments and C preprocessor directives (.F file extension).
- An assembly language source file (.s file extension). These files are passed to the assembler and an object file is produced.
- An intra-file inlined intermediate file (.K file extension). These files are used by the global optimizer for inter-file inlining of any specified source programs. Intra-file and inter-file inlining are described in Chapter 4.
- RATFOR files (.r file extension) or EFL files (.e file extension). These files are transformed by the appropriate preprocessor, then compiled by the FORTRAN compiler. Intel does not distribute RATFOR or EFL preprocessors, but the iWarp FORTRAN driver does support their use.

All other files are passed to the linker.

option can be any of the options listed in the following section.

iwf77 Options

-#	List each phase of compilation being executed and the command options used to invoke that phase.
-##	Same as -#.
-###	List each phase of compilation and the command options used to invoke that phase, but do not execute.
-1	Compile DO loops so they are performed at least once if reached This is the same as -onetrip.

-66	Disable FORTRAN 77 and VMS F77 language features to enhance FORTRAN 66 compatibility.
-77	Disable VMS extensions.
-auto	Make local variables automatic (non-persistent) by default. This option does not affect variables explicitly declared as persistent using SAVE, variables initialized with DATA statements, or variables equivalenced to persistent objects. This option is the default condition.
-c	Suppress the link-edit phase of the compilation and produce an object file for each source file. For a single file, you can change the output file name with the -o option.
-C	Generate code for run-time subscript range checking. If an array subscript is outside the declared array bounds, this option causes compiled code to abort with an error message.
-E*arg*	Use *arg* as an EFL flag argument when processing a .e file.
-f *filename*	Specifies *filename* as an input file.
-F	Run the EFL and RATFOR preprocessor on relevant files, and put the result in files with a .f file extension.
-g	Generate additional information needed by the debugger. This option suppresses all optimizations.
-G*c*	Stop after the specified pass and retain all intermediate files. *c* can be either of the following: 1 first pass q PCO pass
-iwarp	Specifies that the code is to be executed on the iWarp system. This option is useful for makefiles.
-K	Invoke PCO to generate an intra-file inlined intermediate file. Arguments whose names end with .c or .f are compiled and intra-file inlined by PCO. Each intermediate file is left in a file whose name is that of the source file with .K substituted for .c or .f.
-l*x*	Link with the object library lib*x*.a.
-L*dir*	Add *dir* to the list of directories searched for libraries by the linker. Directories specified by the -L option are searched before the standard directory, /iwarp/lib.
-m	Apply the M4 preprocessor to each EFL or RATFOR source file before transforming with the efl or ratfor processors. This option can also be specified as -m4.
-M	Merge the C or FORTRAN and generated iWarp assembler files into a file suffixed .m. For a single file, you can change the output file name with the -o option.

-MM	Same as -M, but incorporate the text of included files in the listing file.
-N*letter nnn*	Change the size of the table defined by *letter* to *nnn*. The compiler provides a diagnostic message when a table overflows. Letter can be any of the following (the default table size, *nnn*, is included: q equivalences (150) x common blocks, subroutine, and function names (200) s statement numbers (2000) c depth of loops, if-then-elses (20) n variable and common block names (401) l labels for computed and assigned gotos and the number of alternate returns (125)
-onetrip	Compile DO loops so they are performed at least once if reached. Without -onetrip, FORTRAN DO loops are not performed at all if the upper limit is smaller than the lower limit.
-o *filename*	Specifies *filename* as the name of the object file, overriding the default name a.out. When used with the -c, -S, or -M options, the default output file name will be overridden only if a single source file is specified.
-O[*level*]	Invoke optimization as indicated by *level*, where *level* is one of the following: 0 no optimizations 1 global optimizations on intermediate code (PCO) 2 basic block compaction, generation of Compute and Access instructions, and PCO optimizations 3 software pipelining, basic block compaction, and PCO optimizations If *level* is not specified, this option will invoke all available optimizations.
-p	Produce code that counts the number of times each routine is called. If link editing takes place, replace the standard startup routine with one that automatically calls the monitor at the start and arranges to write a mon.out# file at normal termination of execution of the object program (# is the cell number). This allows generation of an execution profile with the iwprof command.
-Q*options*	Invoke optimizations as defined by *options*, where *options* can be any or all of the following: G invoke global optimizations (PCO) C compaction of basic blocks, generation of Compute and Access instructions S software pipelining, compaction of basic blocks, and generation of Compute and Access instructions If compile time is an issue, but you want to do some optimization, use the -O*level* option to reduce the amount of optimization.
-R*arg*	Use *arg* as a RATFOR flag argument when processing a .r file.

-S — Compile the specified FORTRAN programs and leave the assembler-language output in corresponding files suffixed .s. For a single file, the output file name can be changed with the -o option.

-static — Make local variables persistent. This option does not affect variables explicitly declared as AUTOMATIC. Use this option only when it is necessary to run non-standard programs that require static variables. Use of this option negatively impacts optimization.

-T — Retain all intermediate files in the same directory as the source code.

-target iwarp — Specifies that the compiled code will be executed on the iWarp system. This option is useful for makefiles.

-u — Make the default type of a variable undefined.

-U — Treat upper and lower cases as distinct.

-V — Display the version of the F77 compiler and all invoked passes, and perform compilation as specified.

-V*options* — Invoke the specified *options* during compilation. *options* can be any of the following. The default conditions are in bold typeface:

check	same as check=bounds, check=overflow, check=underflow
nocheck	same as check=nobounds, check=nooverflow, check=nounderflow
check=all	same as -Vcheck
check=none	same as -Vnocheck
check=bounds	check array subscript bounds
check=nobounds	do not check array subscript bounds
check=overflow	check for floating-point overflow
check=nooverflow	do not check for floating-point overflow
check=underflow	check for floating-point underflow
check= nounderflow	do not check for floating-point underflow
d_lines	compile lines with ‘‘d’’ or ‘‘D’’ in column 1
nod_lines	treat lines with ‘‘d’’ or ‘‘D’’ in column 1 as comments
debug	same as -Vd_lines
nodebug	same as -Vnod_lines
extend_source	extend the range of FORTRAN source to columns 1-132; lines over 132 characters will stop compilation
noextend_source	restrict source to columns 1-72
f77	compile as standard FORTRAN 77 code
nof77	compile using FORTRAN 66 interpretations in cases of incompatibility

-vms — Enable VMS extensions. This is the default.

-VV — Display the version of the F77 compiler and all possible passes, but do not perform compilation.

-w	Suppress all warning messages.
-w66	Suppress only FORTRAN 66 compatibility warnings.
-Wc, *arg1* [, *arg2*...]	Hand off any specified arguments to pass *c*, where *c* is one of the following:

0	compiler first pass
q	PCO pass
1	compiler second pass
a	assembler
l	link editor
f	iwcrt0 and mcrt0
m	listing merger
s	sort
c	iwcc
e	EFL
r	RATFOR

-Yc, *dir*	Get pass *c* from directory *dir*, where *c* is one of the following:

0	compiler first pass
q	PCO pass
1	compiler second pass
a	assembler
l	link editor
f	iwcrt0 and mcrt0
m	listing merger
s	sort
c	iwcc
e	EFL
r	RATFOR

Files

The following sections list files that are part of the compilation process. These files include:

- input files
- intermediate and output files
- elements of the compiler

Input Files

file.f	FORTRAN input file
file.F	FORTRAN input file with C comments and cpp directives
file.r	RATFOR input file
file.e	EFL input file
file.s	assembler input file
file.c	C input file
file.K	intra-file inlined intermediate file previously produced with the -K option
/iwarp/include/f77	directory searched by the FORTRAN INCLUDE statement

Intermediate and Output Files

a.out	executable output file
file.o	object file (assembler output)
file.1	pass one output
file.I	empty file generated for compatibility with the compiler
file.K	intra-file inlined intermediate file produced by the -K option
file.q	PCO output
file.s	assembler source (pass two output)
file.w	inline file (list of functions to inline)
/usr/tmp/F77*	temporary files; you can use a different directory by setting the TMPDIR environment variable to the name of a directory

Parts of the Compiler

/iwarp/lib/iwfcom0	pass one of the compiler
/iwarp/lib/iwccom1	pass two of the compiler
/iwarp/lib/iwccomq	optional optimizer (PCO)
/iwarp/lib/iwliw	optional LIW optimizer
/iwarp/lib/iwas	iWarp assembler