



Intel C/C++、Fortran编译器的使用

李会民

hmli@ustc.edu.cn

中国科学技术大学 超级运算中心

2011年10月



- 1 Intel C/C++、Fortran编译器简介
- 2 Intel C/C++编译器用法
- 3 Intel Fortran编译器用法
- 4 串行程序调试
- 5 联系信息



Intel C/C++、Fortran编译器简介

Intel C/C++ Fortran编译器是一种主要针对Intel平台的高性能编译器，在AMD Opteron平台上性能也不错，可用于开发复杂且要进行大量计算的程序。

Linux版本的Intel C/C++ Fortran编译器在x86_64（又称AMD64，EM64T）平台上默认安装路径：

- 10系列的C/C++编译器默认安装在`/opt/intel/cce/x`，Fortran编译器在`/opt/intel/fce/x`（x为版本号，如10.1.018）
- 11系列的编译器安装在`/opt/intel/Compiler/y/z`（如安装的为11.0.081，则对应的y为11.0，z为081）
- 12系列的编译器安装在类似下面目录：
 - `/opt/intel/composerxe-2011.3.174`
 - `/opt/intel/composer_xe_2011_sp1.6.233`
- 用户可以查看类似目录下还安装有哪些版本



培训内容

- 1 Intel C/C++、Fortran编译器简介
- 2 Intel C/C++编译器用法
- 3 Intel Fortran编译器用法
- 4 串行程序调试
- 5 联系信息



编译命令基本格式

基本格式:

- C: *icc [options] file1 [file2 ...]*
- C++: *icpc [options] file1 [file2 ...]*

注意:

- *[]*表示是其内部的选项可选
- 文件名和选项区分大小写



输入文件后缀与类型的关系

编译器默认将按照输入文件的后缀判断文件类型，编译时也可以用编译选项强制指定。

| 文件名 | 解释 | 编译时动作 |
|--|--------|---------|
| filename.c | C源文件 | 传给编译器 |
| filename.C filename.CC filename.cc filename.cpp filename.cxx | C++源文件 | 传给编译器 |
| filename.a filename.so | 库文件 | 传递给链接器 |
| filename.i | 预处理文件 | 传递给标准输出 |
| filename.o | 目标文件 | 传递给链接器 |
| filename.s | 汇编文件 | 传递给汇编器 |



输出文件后缀与类型的关系

编译器默认将输出按照文件类型与后缀相对应。

| 文件名 | 解释 |
|------------|---------------|
| filename.i | 预处理文件，由-p选项生成 |
| filename.o | 目标文件，由-c选项生成 |
| filename.s | 汇编文件，由-s选项生成 |
| a.out | 默认生成的可执行文件 |



重要编译选项 I

编译选项对运行速度、兼容性等有影响，以下仅仅介绍部分重要选项，详细的建议仔细看看编译器手册中关于程序优化的部分，特别是IPO、PGO和HLO部分，多加测试，选择适合自己程序的编译选项以提高性能。

- **-Bdynamic**: 在运行时动态链接所需要的库。
- **-Bstatic**: 静态链接用户生成的库。
- **-c**: 仅编译成目标文件 (.o文件)。
- **-fast**: 最大化整个程序的速度。这里是所谓的最大化，还是需要结合程序本身使用合适的选项。
- **-g**: 包含调试信息。
- **--help**: 显示帮助信息。
- **-inline-level=[n]**: 设置inline层数。
- **-ip**: 在单个文件中进行过程间优化(Interprocedural Optimizations-IPO)。



重要编译选项 II

- `-ipo[n]`: 在多文件中进行过程间优化, `n`为可产生的目标文件数, 为非负整数。
- `-I<头文件目录>`: 指明头文件的搜索路径。
- `-L<库目录>`: 指明库的搜索路径。
- `-l<库文件>`: 指明所需链接的库名, 如库名为`libxyz.a`, 则可用`-lxyz`指定。
- `-mtune=<CPU架构>`: 指定针对某种CPU架构的处理器进行优化。
- `-openmp`: 编译OpenMP程序, 注意: 只能在同一个节点的CPU上跑OpenMP程序。
- `-O<级别>`: 设定优化级别, 默认为`O2`, `O`与`O2`相同, 推荐使用。`O3`为在`O2`基础之上增加更激进的优化, 比如包含循环和内存读取转换和预取等, 但在有些情况下速度反而慢, 建议在具有大量浮点计算和大数据处理的循环时的程序使用。
- `-p`: 进行概要导向优化(Profile Guided Optimization-PGO)。



重要编译选项 III

- **-shared**: 产生共享目标而不是可执行文件，必须在编译每个目标文件时使用**-fpic**选项。
- **-static**: 静态链接所有库。
- **-std=<标准>**: 标准可以为c89、c99、gnu89、gnu++98或c++0x，分别对应不同的标准。
- **-v**: 显示详细编译过程以及编译参数等。
- **-V**: 显示编译器版本号。
- **-w**: 编译时不显示任何警告，只显示错误。
- **-wall**: 编译时显示所有警告。
- **-x <类型>**: 类型可以为c、c++、c-header、cpp-output、c++-cpp-output、assembler、assembler-with-cpp或none，分别表示c源文件等，以使所有源文件都被认为是此类型的。



串行及OpenMP并程序编译举例

- 将C程序yourprog.c编译为可执行文件yourprog:
icc -o yourprog yourprog.c
- 将C程序yourprog.c编译为目标文件yourprog.o:
icc -c yourprog.c
- 将C程序yourprog.c链接/opt/lib/liblapack.so后编译为可执行文件yourprog:
icc -o yourprog -L/opt/lib -llapack yourprog.c
- 将C程序yourprog.c静态编译为O3优化的可执行文件yourprog:
icc -O3 -static -o yourprog yourprog.c
- 将C++程序yourprog.cpp编译为可执行文件yourprog:
icpc -o yourprog yourprog.cpp
- 将OpenMP指令并行的C程序yourprog-omp.c编译为可执行文件yourprog-omp:
icc -o yourprog-omp -openmp yourprog.c



编译时出错信息格式

```
netlog.c(140): error: identifier "hhh" is undefined
```

```
    for (int hhh=domain_cnt+1; hhh>TMP; hhh--){
```

```
        ^
```

```
netlog.c(156): error: expected an expression
```

```
    for (int i=0; i<32; i++) for (int j=0; j<256; j++) if (ipl[i][j]!=0) fprintf(fin, "202.38.%2d.%3d: %d\n", i,
```

```
        ^
```

- 源文件名(行数): 错误类型: 具体说明
- 源代码, ^指示出错位置



培训内容

- 1 Intel C/C++、Fortran编译器简介
- 2 Intel C/C++编译器用法
- 3 Intel Fortran编译器用法**
- 4 串行程序调试
- 5 联系信息



编译命令基本格式

基本格式:

- Fortran: *ifort* [*options*] *file1* [*file2* ...]

注意:

- []表示是其内部的选项可选
- 文件名和选项区分大小写



输入文件后缀与类型的关系

编译器默认将按照输入文件的后缀判断文件类型，编译时也可以用编译选项强制指定。

| 文件名 | 解释 | 动作 |
|--|-----------------|-----------------------|
| filename.a | 目标库文件 | 传给编译器 |
| filename.f filename.for filename.ftn filename.i | 固定格式的Fortran源文件 | 被Fortran编译器编译 |
| filename.fpp filename.FPP filename.F filename.FOR filename.FTN | 固定格式的Fortran源文件 | 自动被Fortran编译器预处理后再被编译 |
| filename.f90 filename.i90 | 自由格式的Fortran源文件 | 被Fortran编译器编译 |
| filename.F90 | 自由格式的Fortran源文件 | 自动被Fortran编译器预处理后再被编译 |
| filename.s | 汇编文件 | 传递给汇编器 |
| filename.so | 库文件 | 传递给链接器 |
| filename.o | 目标文件 | 传递给链接器 |



输出文件的后缀与类型的关系

编译器默认将输出按照文件类型与后缀相对应。

| 文件名 | 解释 | 生成方式 |
|--------------|------------|----------------------------|
| filename.o | 目标文件 | 编译时添加-c选项生成 |
| filename.so | 共享库文件 | 编译时指定为共享型，如添加-shared，并不含-c |
| filename.mod | 模块文件 | 编译含有MODULE声明时的源文件生成 |
| filename.s | 汇编文件 | 编译时添加-S选项生成 |
| a.out | 默认生成的可执行文件 | 编译时没有指定-c时生成 |



重要编译选项 I

编译选项对运行速度、兼容性等有影响，以下仅仅介绍部分重要选项，详细的建议仔细看看编译器手册中关于程序优化的部分，特别是IPO、PGO和HLO部分，多加测试，选择适合自己程序的编译选项以提高性能。

- **-Bdynamic**: 运行时动态链接所需要的库。
- **-Bstatic**: 静态链接用户生成的库。
- **-c**: 仅编译成目标文件（.o文件）。
- **-convert [关键字]**: 转换无格式数据的类型，比如关键字为**big_endian**和**little_endian**时，分别表示无格式的输入输出为**big_endian**和**little_endian**格式，更多格式类型请看编译器手册。
- **-cpp**: 对源代码进行预处理，等价于**-fpp**。
- **-extend-source[size]**: 指明固定格式的Fortran源代码宽度，选项size可为72、80和132。也可直接用**-72**、**-80**和**-132**指定，默认为72字符。



重要编译选项 II

- **-fast**: 最大化整个程序的速度。这里是所谓的最大化，还是需要结合程序本身使用合适的选项。
- **-fixed**: 指明Fortran源代码为固定格式，默认由文件后缀决定格式类别。
- **-free**: 指明Fortran源程序为自由格式，默认由文件后缀决定格式类别。
- **-fpic**: 生成位置无关代码，当编译成共享目标文件时必须使用此选项，等价于-fPIC，默认为-fno-pic。
- **-g**: 包含调试信息。
- **--help**: 显示帮助信息。
- **-ip**: 在单个文件中进行过程间优化(Interprocedural Optimizations-IPO)。
- **-ipo[n]**: 在多文件中进行过程间优化，非负整数n为可生成的目标文件数。



- `-I<头文件目录>`: 指明头文件的搜索路径。
- `-implicitnone`: 指明默认变量名为未定义, 建议在写程序时添加`implicit none`语句, 以避免出现由于默认类型造成的错误。
- `-L<库目录>`: 指明库的搜索路径。
- `-l<库文件>`: 指明所需链接的库名, 如库文件为`libxyz.a`, 则可用`-lxyz`指定。
- `-nofree`: 指明Fortran源程序为固定格式。
- `-openmp`: 编译OpenMP指令并程序, 注意: 在集群上只能在同一个节点内CPU上跑OpenMP程序, 提交作业时请保证在同一个节点上运行。
- `-O<级别>`: 设定优化级别。默认为O2, O与O2相同, 推荐使用。O3为在O2基础之上增加更激进的优化, 比如包含循环和内存读取转换和预取等, 但在有些情况下速度反而慢, 建议在具有大量浮点计算和大数据处理的循环时的程序使用。



重要编译选项 IV

- **-p**: 进行概要导向优化(Profile Guided Optimization-PGO)。
- **-shared**: 生成共享目标而不是可执行文件，必须在编译每个目标文件时使用**-fpic**选项。
- **-stand <标准>**: 以指定Fortran标准进行编译，编译时显示源文件中不符合此标准的信息。标准可为**f03**、**f90**、**f95**和**none**，分别对应显示不符合Fortran 2003、90、95的代码信息和不显示任何非标准的代码信息，也可写为**-std<标准>**，此时标准不带**f**，可为**03**、**90**、**95**。
- **-static**: 静态链接所有库。
- **-unroll[n]**: 循环最大可展开的层数，与性能相关。
- **-us**: 编译时给外部用户定义的函数名添加一个下划线，等价于**-assume underscore**，如果编译时显示 **_** 函数找不到时也许添加此选项即可解决。
- **-v**: 显示详细编译过程以及编译参数等。
- **-V**: 显示编译器版本号。



- **-w**: 编译时不显示任何警告，只显示错误。
- **-Wall**: 编译时显示所有警告。
- **-X**: 编译时不用默认的头文件搜索目录，与**-I**结合可使用指定的头文件目录。



- 将Fortran 77程序yourprog.for编译为可执行文件yourprog:
ifort -o yourprog yourprog.for
- 将Fortran 77程序yourprog.for编译为目标文件yourprog.o:
ifort -c yourprog.for
- 将使用lapack库的Fortran 90程序yourprog.f90编译为可执行文件yourprog:
ifort -o yourprog -L/opt/lib -llapack yourprog.f90
- 将Fortran 90程序yourprog.f90编译为目标文件yourprog.o:
ifort -c yourprog.f90
- 将Fortran 90程序yourprog.f90静态编译为O3优化的可执行文件yourprog:
ifort -O3 -static -o yourprog yourprog.f90
- 将Fortran 90程序yourprog.f90静态编译为可执行文件yourprog:
ifort -o yourprog -static yourprog.f90
- 将OpenMP指令并行的Fortran 90程序yourprog-omp.f90编译为可执行文件yourprog-omp:
ifort -o yourprog-omp -openmp yourprog.f90



```
NOlihm.f90(146): error #6404: This name does not have a type, and must have an explicit type.  
[NPR]
```

```
    n2nd=0;  npr=0
```

```
-----^  
NOlihm.f90(542): remark #8290: Recommended relationship between field width 'W' and the number of  
6060 format(/ i2, '-th layer ', i2, '-th element: z=', i3, ' a=', f9.5/' Ef=', f7.5, ' (keV)'/ ' c =', f5.3, '  
( at% )')
```

```
-----^
```

- 源文件名(行数): 错误类型:具体说明
- 源代码, ^指示出错位置



- 1 Intel C/C++、Fortran编译器简介
- 2 Intel C/C++编译器用法
- 3 Intel Fortran编译器用法
- 4 串行程序调试
- 5 联系信息



串程序调试: idb

- 利用Intel编译器带有调试器: 命令行的*idbc*和图形界面的*idb*, 并且可以调试OpenMP和MPI并程序, 进入后运行*help*可以查看具体命令
- 编译时需要添加*-g*参数, 比如:
ifort -o yourprog -g yourprog.f90
- 启动调试:
 - 无初始调试命令文件方式: *idbc ./yourprog*
 - 有初始调试命令文件方式: *idbc -command dbg.txt ./yourprog**dbg.txt*文件存储调试命令, 每行一条命令, 启动时自动按顺序执行, 其内容类似:

```
break 139
```

```
run
```

详细用法, 参见

- Getting Started with the Intel® Debugger
- Intel® Debugger Online Help



- 1 Intel C/C++、Fortran编译器简介
- 2 Intel C/C++编译器用法
- 3 Intel Fortran编译器用法
- 4 串行程序调试
- 5 联系信息



李会民:

- 办公室: 科大东区新科研楼A座网络信息中心二楼205室
- 办公电话: 0551-3602248
- 电子信箱: hml@ustc.edu.cn
- 个人主页: <http://hml.ustc.edu.cn>
- 中国科大超算平台: <http://scc.ustc.edu.cn>