

quantum-espresso (原先叫 pwscf) 十分容易安装, 至少比同类的 CPMD, cp2k 容易多了, 作为一个后起之秀, 解决了很多前一代软件安装上的问题。

首先我假定你有一台 AMD 多核 (4 核或 6 核) 的台式机, i7-6 核更好; 当然最好的是 1280 核的 cluster 神码的;

其次, 必须装 Linux, 没听说过在 windows 下跑 quantum-espresso 的, 就像火车必须上火车道。火车不是不能在大马路上跑, 只是那样不叫一个事儿; 如果你不会 Linux 操作, 现学, 单机装 Ubuntu, 上手很快的; 如果你从来没装过, 马上请周围懂 Linux 的朋友 20 分钟即可装好 Ubuntu, 再学一些基本命令行操作; 千万不能被一些本来很容易办到的事情吓住, 对吧。

安装过程大致是这样的, 1. 下载; 2. 解压; 3. 编译; 4. 测试
一步一步来, 没有搞不成的;

1. 下载

这里我教个更简单的下载方法, 打开 Ubuntu linux 命令行终端窗口 terminal, 在 Linux 命令提示符\$或>后面输入:

```
$wget http://qe-forge.org/frs/download.php/167/espresso-4.3.2.tar.gz
```

```
$wget http://qe-forge.org/frs/download.php/167/espresso-4.3.2-examples.tar.gz
```

如果提示找不到 wget 那就先安装 wget 和 curl。Ubuntu Linux 下:

```
$ sudo apt-get install wget
```

```
$ sudo apt-get install curl (运行测试会用到)
```

注意你打开终端窗口以后, 当前目录为/home/xxxx

xxxx 为你的用户名。

2. 解压

下载完后, 输入:

```
$ tar zxvf espresso-4.3.2.tar.gz
```

```
$ tar xzfv espresso-4.3.2-examples.tar.gz
```

解压完成后会生成一个 espresso-4.3.2 的目录，进入

```
$ cd espresso-4.3.2
```

3. 串行编译

编译前先安装 gfortran

```
$ sudo apt-get install build-essential gcc gfortran
```

然后自动配置

```
$ ./configure
```

然后编译

```
$ make all
```

然后就会在当前目录下生成一个 bin 目录，里面就是所有编译好的可执行文件，全部以 .x 结尾；

你要执行它们，以 pw.x 为例，一般是：

```
$ /home/xxxx/espresso-4.3.2/bin/pw.x < input.file > output.file
```

(把 input.file/output.file 换成你的实际的输入输出文件名；)

为了不用每次都输入 pw.x 前面的绝对路径，可以把这个路径加入到环境配置文件 .bashrc 中；

.bashrc 文件在用户根目录 “/home/xxxx” 下面；\$cd 就回到了；

```
$ gedit .bashrc
```

在最下面加入一行

```
export PATH=$PATH:/home/xxxx/espresso-4.3.2/bin
```

保存关闭，然后 source 一下生效（每次开机会自动 source，无须再次 source）

```
$ source .bashrc
```

配置好路径后，每次调用 pw.x 直接

```
$ pw.x < input > output
```

即可

4. 测试

```
$ cd espresso-4.3.2/tests
```

```
$ ./check-pw.x.j
```

这个用来测试是否全部功能正常，注意出错提示；

一般来说，只要常用的 pw.x ph.x 功能正常，或者你用的功能正常就好；用不到功能出错，没必要去折腾；

运行全部 examples

```
$ cd .. 回到 espresso-4.3.2 下
```

```
$ cd examples
```

```
$ ./run_all_examples
```

5. 并行编译

先安装并行编译器和并行编译库 openmpi, 最简单的方法是打开 Ubuntu 的软件中心，搜索 openmpi 然后点 install 会自动配置好的，超级 easy；

否则手动

```
$ sudo apt-get install openmpi-bin
```

然后

```
$ ./configure
```

```
$ make all
```

如果没有错误，完成后在 bin 下得到并行版的 pw.x 等可执行文件；注意这样会覆盖掉之前安装的串行版的 pwscf；

执行的话，一般是(假如用 4 个核来跑)

```
$ mpirun -np 4 pw.x < input > output
```

效率大大提高！

如果你在干净的 Ubuntu Linux 下按照上述步骤，一般不会出错；这得益于 quantum espresso 强大的自动配置 ./configure 会自动检测当前是否有并行环境，是否安装了数学库 lapack, FFT，如果没找到，就自动调用自带的库（下载的压缩文件中自动包含了！）

如果你安装了 intel 编译器，反而会更麻烦一些，因为 openmpi 自动与 Ubuntu 自带的 gfortran 结合，而不与 intel 的 ifort 结合，导致 intel 并行编译出错；欲用 openmpi+intel ifort/MKL，那就的卸掉 openmpi，然后手动重新编译 intel 版的 openmpi，折腾。

并行版的测试与串行版类似，唯一不同点是运行 ./check-pw.x.j 和 ./run_all_examples 之前，修改 espresso-4.3.2/examples 目录下面的 environment_variables 文件，把

```
#PARAM_PREFIX="mpirun -np 2"
```

```
PARAM_PREFIX=""
```

修改成

```
PARAM_PREFIX="mpirun -np 2"
```

```
#PARAM_PREFIX=""
```

把 2 换成你要并行的核数，比如 4。

如果在 configure 或 make all 的过程出错，这个很正常，也很难预料是什么样的错误，那你可以来这里报告，或者去 maillist 上搜搜解决方法。一般你遇到的问题，都是别人遇到过上百次的；