全文参考链接如下：

【1】[**https://xyce.sandia.gov/documentation/BuildingGuide.html#rhelPreReq**](https://xyce.sandia.gov/documentation/BuildingGuide.html#rhelPreReq)

**【2】**

[**https://xyce.sandia.gov/documentation/RunningTheTests.html**](https://xyce.sandia.gov/documentation/RunningTheTests.html)

**1.Prerequisite libraries available**

gcc (see the [Trilinos note](https://xyce.sandia.gov/documentation/BuildingGuide.html#TrilinosNote)中 Debian Linux and Variants part)

g++

gfortran

make

cmake

bison

flex

libfl-dev

libfftw3-dev

libsuitesparse-dev

libblas-dev

liblapack-dev

libtool

If you are building Xyce from our Github repository, you will also need:

autoconf

automake

git

Install these packages using

|  |
| --- |
| sudo apt-get install [包名] |

**2.Installing Trilinos(安装12.12.1版本)**

**note:**download **version 12.12.1 of Trilinos** from the Trilinos git repository at <https://github.com/trilinos/Trilinos/releases/tag/trilinos-release-12-12-1/>, and unpack the source code. ***Xyce is not guaranteed to build properly with other versions of Trilinos.***

***WSL系统下安装流程：***

***#################################################***

**step1:创建安装目标文件夹(以/opt/software文件夹为例)**

|  |
| --- |
| sudo mkdir /opt/software |

**【附加】git clone生效问题(以Xyce安装包为例)**

|  |
| --- |
| ssh-keygen -t ed25519 -C "your-own-email" |

**拷贝~/.ssh/id\_ed25519.pub内容到github的ssh key中**

|  |
| --- |
| git clone git@github.com:Xyce/Xyce.git |

**存在问题：在非~/.ssh/目录下：**



**但在~/.ssh/目录下：就成功了**



#######################################################################

**step2:复制Trilinos-trilinos-release-12-12-1.tar.gz安装包到/opt/software/中**

|  |
| --- |
| sudo cp -r ~/.ssh/Trilinos-trilinos-release-12-12-1.tar.gz /opt/software/ |

遇到如下困难：



if you do not allowed to write file into the directory with "sudo cp -r ..."

那么需要修改目标文件夹的权限，使其拥有写入权限。

|  |
| --- |
| sudo chmod 777 /opt/software |

**step3:选择解压安装包存放路径（以/opt/Trilinos12.12为例）**

|  |
| --- |
| sudo mkdir /opt/Trilinos12.12cd /opt/Trilinos12.12sudo tar -xzf /opt/software/Trilinos-trilinos-release-12-12-1.tar.gz |

**3.Building Trilinos(安装12.12.1版本)**

**拟定安装在/opt/XyceLibs/Serial目录下**

|  |
| --- |
| sudo mkdir /opt/XyceLibs/Serial |

note:Set the variable "ARCHDIR" to the location where you want the Trilinos libraries and headers to be installed. In this example, we use $HOME/XyceLibs/Serial. The install process will create "include" and "lib" subdirectories of this directory.

**step1:编写reconfigure脚本于/opt/Trilinos12.12文件夹下**

|  |
| --- |
| #!/bin/shSRCDIR=/opt/Trilinos12.12/Trilinos-trilinos-release-12-12-1ARCHDIR=/opt/XyceLibs/SerialFLAGS="-O3 -fPIC"cmake \-G "Unix Makefiles" \-DCMAKE\_C\_COMPILER=gcc \-DCMAKE\_CXX\_COMPILER=g++ \-DCMAKE\_Fortran\_COMPILER=gfortran \-DCMAKE\_CXX\_FLAGS="$FLAGS" \-DCMAKE\_C\_FLAGS="$FLAGS" \-DCMAKE\_Fortran\_FLAGS="$FLAGS" \-DCMAKE\_INSTALL\_PREFIX=$ARCHDIR \-DCMAKE\_MAKE\_PROGRAM="make" \-DTrilinos\_ENABLE\_NOX=ON \-DNOX\_ENABLE\_LOCA=ON \-DTrilinos\_ENABLE\_EpetraExt=ON \-DEpetraExt\_BUILD\_BTF=ON \-DEpetraExt\_BUILD\_EXPERIMENTAL=ON \-DEpetraExt\_BUILD\_GRAPH\_REORDERINGS=ON \-DTrilinos\_ENABLE\_TrilinosCouplings=ON \-DTrilinos\_ENABLE\_Ifpack=ON \-DTrilinos\_ENABLE\_AztecOO=ON \-DTrilinos\_ENABLE\_Belos=ON \-DTrilinos\_ENABLE\_Teuchos=ON \-DTeuchos\_ENABLE\_COMPLEX=ON \-DTrilinos\_ENABLE\_Amesos=ON \-DAmesos\_ENABLE\_KLU=ON \-DTrilinos\_ENABLE\_Sacado=ON \-DTrilinos\_ENABLE\_Stokhos=ON \-DTrilinos\_ENABLE\_Kokkos=ON \-DTrilinos\_ENABLE\_ALL\_OPTIONAL\_PACKAGES=OFF \-DTrilinos\_ENABLE\_CXX11=ON \-DTPL\_ENABLE\_AMD=ON \-DAMD\_LIBRARY\_DIRS="/usr/lib" \-DTPL\_AMD\_INCLUDE\_DIRS="/usr/include/suitesparse" \-DTPL\_ENABLE\_BLAS=ON \-DTPL\_ENABLE\_LAPACK=ON \$SRCDIR |

step2:依次执行以下命令

|  |
| --- |
| sudo chmod u+x reconfiguresudo ./reconfiguresudo makesudo make install |

note：如之前增加了Amesos2部分的编译，其分别对应有：

-DTrilinos\_ENABLE\_Amesos2=ON \

-DAmesos2\_ENABLE\_KLU2=ON \

-DAmesos2\_ENABLE\_Basker=ON \

会出现如下中断异常：



解决方案：（不编译Amesos2部分）

Again, be sure to compile version 12.12.1. Other versions will not work with the following process,as they require slightly different build options.

**4.Obtaining Xyce through Github(见上)**

**5.Installing Xyce(以文件所建路径/opt/spice/Xyce为例)**

|  |
| --- |
| cd /opt/spice/Xycesudo ./bootstrap |

Bootstrap will create the configure script and various "Makefile.in" files needed to build Xyce.

The git repository does not have a copy of the "configure" script that the "tarball" download contains, and in order to proceed to the build you must first "bootstrap" the source code to create it. This operation requires that you have GNU Autoconf, Automake, and Libtool installed.

**6.Building Xyce**

**参考：**[**https://xyce.sandia.gov/documentation/BuildingGuide.html#rhelPreReq**](https://xyce.sandia.gov/documentation/BuildingGuide.html#rhelPreReq)

**step1:编写reconfigure脚本于/opt/spice/Xyce文件夹下**

|  |
| --- |
| #!/bin/sh/opt/spice/Xyce/configure \CXXFLAGS="-O3" \ARCHDIR="/opt/XyceLibs/Serial" \CPPFLAGS="-I /usr/include/suitesparse" \--enable-stokhos \--prefix=/opt/spice/XyceInstall/Serial |

note：如果要将编译和安装分别安装在两个不同文件夹下

（在这儿，编译放在/opt/spice/Xyce\_build\_serial文件夹下；

安装放在/opt/spice/XyceInstall/Serial文件夹下）

step2:依次执行以下命令

|  |
| --- |
| chmod u+x reconfiguresudo mkdir /opt/spice/Xyce\_build\_serialcd /opt/spice/Xyce\_build\_serialsudo /opt/spice/Xyce/reconfiguresudo make |

**7.Running the Xyce Regression Suite（suggestXyceTaglist.sh测试）**

**参考：**[**https://xyce.sandia.gov/documentation/RunningTheTests.html**](https://xyce.sandia.gov/documentation/RunningTheTests.html)

**step1:拿取测试套件**

|  |
| --- |
| cd ~/.sshgit clone git@github.com:Xyce/Xyce\_Regression.gitsudo cp -r Xyce\_Regression/ /opt/spice/ |

**step2:Prerequisites**

**perl/python/numpy/Scipy**

**step3:编写myscript于/opt/spice/Xyce\_build\_serial文件夹下**

|  |
| --- |
| EXECSTRING=/opt/spice/Xyce\_build\_serial/src/Xyceeval `/opt/spice/Xyce\_Regression/TestScripts/suggestXyceTagList.sh $EXECSTRING`/opt/spice/Xyce\_Regression/TestScripts/run\_xyce\_regression \--output=/opt/spice/Xyce\_build\_serial/Xyce\_Test \--xyce\_test="/opt/spice/Xyce\_Regression" \--resultfile=/opt/spice/Xyce\_build\_serial/serial\_results \--taglist="${TAGLIST}" \$EXECSTRING |

测试命令：

|  |
| --- |
| sudo ./myscript |

测试结果查询：（/opt/spice/Xyce\_build\_serial/Xyce\_Test/）





以上少数特殊案例失败是由于所用脚本包含的可执行文件没有覆盖到测试失败的案例中，总体来说算编译成功。

参考：[**https://xyce.sandia.gov/documentation/RunningTheTests.html**](https://xyce.sandia.gov/documentation/RunningTheTests.html)

The reason for using suggestXyceTagList.sh above is to allow for the possibility of testing binaries with incomplete feature sets (such as disabling ADMS sensitivity or the models generated from Verilog-A).  suggestXyceTagList.sh runs "Xyce -capabilities" to determine what features were compiled into the binary and emits a tag list to exclude tests that use features not present.  Prior versions of this document had suggested a tags list directly that was only appropriate for testing complete builds directly out of their build directories, and would show a few failing tests if run on partial builds or with an installed binary.

通过查询测试编译组件是否成功所用的网表案例，可以发现我们的编译组件基本满足需求。

