

# Ubuntu14.04 x64 qt 集成 cuda

## 系统环境

- 1: ubuntu 14.04.4 LTS 64 位
- 2: 显卡 GTX 720
- 3: cuda\_7.5.18\_linux.run

环境确认

```
root@coues-Lenovo-H3050:/home/coues# cat /etc/issue
Ubuntu 14.04.4 LTS \n \l

root@coues-Lenovo-H3050:/home/coues# uname -m
x86_64
root@coues-Lenovo-H3050:/home/coues# lspci | grep -i nvidia
01:00.0 VGA compatible controller: NVIDIA Corporation GK208 [GeForce GT 720] (rev a1)
01:00.1 Audio device: NVIDIA Corporation GK208 HDMI/DP Audio Controller (rev a1)
```

## 安装依赖

4: 安装依赖包

```
root@coues-Lenovo-H3050:/home/coues#sudo apt-get install -y freeglut3-dev \
build-essential libx11-dev \
libxmu-dev libxi-dev libgl1-mesa-glx libglu1-mesa-dev
```

5:卸载 ubuntu 原生带有的驱动 ##在我的系统环境没有安装原生驱动

```
sudo nvidia-uninstall
```

6:清除 nvidia 库

```
sudo apt-get --purge remove nvidia-*
```

7:屏蔽原生库

```
cd /etc/modprobe.d/
```

```
ls blacklist.conf
```

vi nvidia-install-disable-nouveau.conf

##没有安装原生驱动 5-7 步均不操作

## 关闭 x 服务

8:关闭 x 服务

[root@coues-Lenovo-H3050](#):/home/coues#service lightdm stop

[root@coues-Lenovo-H3050](#):/home/coues#service gdm stop ##如果启用 gdm 也需要关闭

要关闭

9 : 安装包下载位置



url : <https://developer.nvidia.com/cuda-downloads>

qq 476200175



### Select Target Platform

Click on the green buttons that describe your target platform. Only supported plat

Operating System	Windows	Linux	Mac OSX
Architecture 	x86_64	x86_64le	
Distribution	Fedora	OpenSUSE	RHEL
	Ubuntu		CentOS
Version	15.04	14.04	
Installer Type 	runfile (local)	deb (local)	deb (network)

### Download Target Installer for Linux Ubuntu 14.04 x86\_64

cuda\_7.5.18\_linux.run (md5sum: 4b3bcecf0dfc35928a0898793cf3e4c6)

Download (1.1 GB)

10： 安装

`root@coues-Lenovo-H3050:/home/coues# chmod 777 cuda_7.5.18_linux.run`

```
Do you accept the previously read EULA? (accept/decline/quit): accept
Install NVIDIA Accelerated Graphics Driver for Linux-x86_64 352.39? ((y)es/(n)o/(q)uit): y
Do you want to install the OpenGL libraries? ((y)es/(n)o/(q)uit) [ default is yes ]: y
Install the CUDA 7.5 Toolkit? ((y)es/(n)o/(q)uit): y
Enter Toolkit Location [ default is /usr/local/cuda-7.5 ]:
Do you want to install a symbolic link at /usr/local/cuda? ((y)es/(n)o/(q)uit): y
Install the CUDA 7.5 Samples? ((y)es/(n)o/(q)uit): y
Enter CUDA Samples Location [ default is /home/coues ]:
Installing the NVIDIA display driver...
```

```
Driver:    Installed
Toolkit:   Installed in /usr/local/cuda-7.5
Samples:   Installed in /home/coues, but missing recommended libraries

Please make sure that
- PATH includes /usr/local/cuda-7.5/bin
- LD_LIBRARY_PATH includes /usr/local/cuda-7.5/lib64, or, add /usr/local/cuda-7.5

To uninstall the CUDA Toolkit, run the uninstall script in /usr/local/cuda-7.5/bin
To uninstall the NVIDIA Driver, run nvidia-uninstall

Please see CUDA_Installation_Guide_Linux.pdf in /usr/local/cuda-7.5/doc/pdf for details

Logfile is /tmp/cuda_install_2090.log
```

11 : 添加配置到系统环境

[root@coues-Lenovo-H3050](#):/home/coues#vi ~/.bashrc

```
export CUDA=/usr/local/cuda-7.5
```

```
export PATH=$CUDA/bin:$PATH
```

```
export LD_LIBRARY_PATH=/usr/lib:$CUDA/lib64:$CUDA/lib:$LD_LIBRARY_PATH
```

12:使配置环境生效

[root@coues-Lenovo-H3050](#):/home/coues#source ~/.bashrc

13:确认 nvcc 配置正确

[root@coues-Lenovo-H3050](#):/home/coues# nvcc --version

nvcc: NVIDIA (R) Cuda compiler driver

Copyright (c) 2005-2015 NVIDIA Corporation

Built on Tue\_Aug\_11\_14:27:32\_CDT\_2015

Cuda compilation tools, release 7.5, V7.5.17

## 验证环境

### 14 : 跑个 Samples 玩玩

```
root@coues-Lenovo-H3050: /home/coues/NVIDIA_CUDA-7.5_Samples/0_Simple/vectorAdd
root@coues-Lenovo-H3050:/home/coues/NVIDIA_CUDA-7.5_Samples/0_Simple/vectorAdd# nvcc
root@coues-Lenovo-H3050:/home/coues/NVIDIA_CUDA-7.5_Samples/0_Simple/vectorAdd# ./a.o
[Vector addition of 50000 elements]
Copy input data from the host memory to the CUDA device
CUDA kernel launch with 196 blocks of 256 threads
Copy output data from the CUDA device to the host memory
Test PASSED
Done
root@coues-Lenovo-H3050:/home/coues/NVIDIA_CUDA-7.5_Samples/0_Simple/vectorAdd# rm -f
root@coues-Lenovo-H3050:/home/coues/NVIDIA_CUDA-7.5_Samples/0_Simple/vectorAdd# make
"/usr/local/cuda-7.5/bin/nvcc -cbin g++ -I../common/inc -m64 -gencode arch=compute_35,code=sm_35 -gencode arch=compute_37,code=sm_37 -gencode arch=compute_50,code=sm_50 -gencode arch=compute_52,code=sm_52 -o vectorAdd.o -c vectorAdd.cu"
"/usr/local/cuda-7.5/bin/nvcc -cbin g++ -m64 -gencode arch=compute_20,code=sm_20 -gencode arch=compute_35,code=sm_35 -gencode arch=compute_37,code=sm_37 -gencode arch=compute_50,code=sm_50 -gencode arch=compute_52,code=sm_52 -o vectorAdd vectorAdd.o"
mkdir -p ../../bin/x86_64/linux/release
cp vectorAdd ../../bin/x86_64/linux/release
root@coues-Lenovo-H3050:/home/coues/NVIDIA_CUDA-7.5_Samples/0_Simple/vectorAdd# ./vectorAdd
[Vector addition of 50000 elements]
Copy input data from the host memory to the CUDA device
CUDA kernel launch with 196 blocks of 256 threads
Copy output data from the CUDA device to the host memory
Test PASSED
```

## 配置 IDE

### 15:配置 QT creator 开发环境

配置链接库,有两种方法

a:在/etc/ld.so.conf 里面配置 , 加入"include /usr/local/cuda-7.5/lib64",执行"ldconfig"

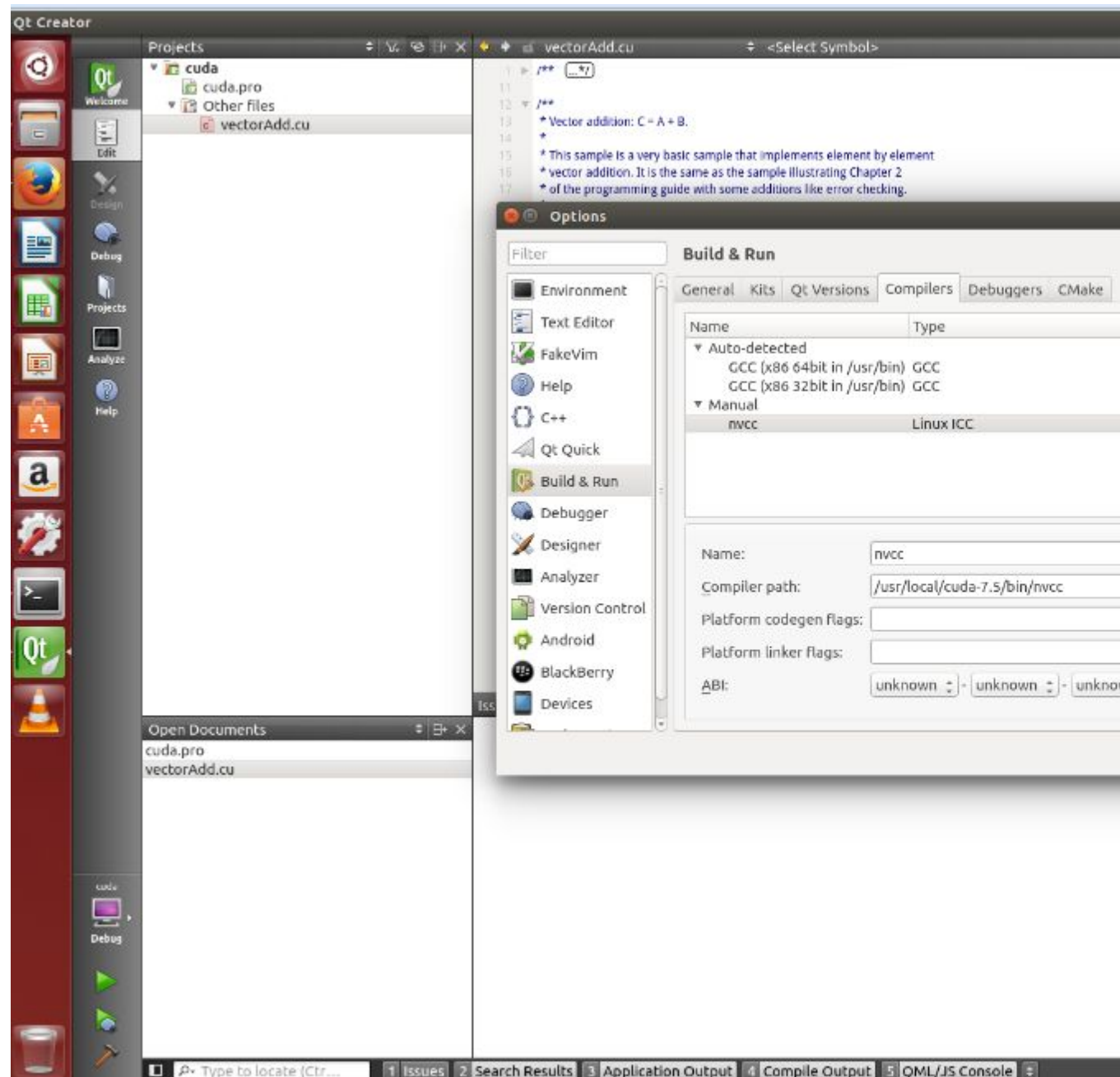
b:直接拷贝

```
root@coues-Lenovo-H3050:/usr/local/cuda-7.5/lib64# cp ./lib* /usr/lib/
cp: 略过目录 './stubs'
root@coues-Lenovo-H3050:/usr/local/cuda-7.5/lib64# ls
libcublas_device.a  libcudart.so.7.5  libcufftw.so  libcurand.so  libcusolver.so.7.5.18  libnppc.so.7.5.18
libcublas.so  libcudart.so.7.5.18  libcufftw.so.7.5  libcurand.so.7.5  libcusolver_static.a  libnppc.so.7.5.18
libcublas.so.7.5  libcudart_static.a  libcufftw.so.7.5.18  libcurand.so.7.5.18  libcusparse.so  libnppc_static.a
libcublas.so.7.5.18  libcufftw.so  libcufftw_static.a  libcurand_static.a  libcusparse.so.7.5  libnppi.so
libcublas_static.a  libcufftw.so.7.5  libcufftw_static.a  libcurand_static.a  libcusparse.so.7.5.18  libnppi.so.7.5
libcudadevrt.a  libcufftw.so.7.5.18  libcufftw_static.a  libcurand_static.a  libcusparse_static.a  libnppi.so.7.5.18
libcudart.so  libcufftw_static.a  libcufftw_static.a  libcurand_static.a  libcusparse_static.a  libnppi_static.a
libcudart.so.7.5  libcufftw_static.a  libcufftw_static.a  libcurand_static.a  libcusparse_static.a  libnppi_static.a
```



16 : 新建一个工程,删除 CPP 文件,添加个编译器

##貌似多余



17 , 修改 pro 文件

```
#-----
```

```
#
```

```
# Project created by QtCreator 2016-04-11T11:27:57
```

```
#
```

```
#-----
```

```
QT     += core
```

```
QT     -= gui
```

```
TARGET = testcuda
```

```
CONFIG += console
```

```
CONFIG -= app_bundle
```

```
TEMPLATE = app
```

```
# This makes the .cu files appear in your project
```

```
OTHER_FILES += ./vectorAdd.cu
```

```
# CUDA settings <-- may change depending on your system

CUDA_SOURCES += ./vectorAdd.cu

CUDA_SDK = "/usr/local/cuda-7.5" # Path to cuda SDK install

CUDA_DIR = "/usr/local/cuda-7.5" # Path to cuda toolkit install


# DO NOT EDIT BEYOND THIS UNLESS YOU KNOW WHAT YOU ARE DOING....


SYSTEM_NAME = unix # Depending on your system either 'Win32', 'x64', or '
Win64'

SYSTEM_TYPE = 64 # '32' or '64', depending on your system

CUDA_ARCH = sm_21 # Type of CUDA architecture, for example 'compute_1
0', 'compute_11', 'sm_10'

NVCC_OPTIONS = --use_fast_math

# include paths

INCLUDEPATH += $$CUDA_DIR/include

# library directories

QMAKE_LIBDIR += $$CUDA_DIR/lib/

CUDA_OBJECTS_DIR = ./

# Add the necessary libraries

#CUDA_LIBS = -lcuda -lcudart ##这行正确配置库以后可以不使用

# The following makes sure all path names (which often include spaces) are put be
tween quotation marks
```



```

CUDA_INC = $$join(INCLUDEPATH," -I","-I","")

#LIBS += $$join(CUDA_LIBS,'.so ', ' ', '.so')

LIBS += $$CUDA_LIBS

# Configuration of the Cuda compiler

CONFIG(debug, debug|release) {

    # Debug mode

    cuda_d.input = CUDA_SOURCES

    cuda_d.output = $$CUDA_OBJECTS_DIR/${QMAKE_FILE_BASE}_cuda.o

    cuda_d.commands = $$CUDA_DIR/bin/nvcc -D_DEBUG $$NVCC_OPTIONS $$C
UDA_INC $$NVCC_LIBS --machine $$SYSTEM_TYPE -arch=$$CUDA_ARCH -c -o ${
QMAKE_FILE_OUT} ${QMAKE_FILE_NAME}

    cuda_d.dependency_type = TYPE_C

    QMAKE_EXTRA_COMPILERS += cuda_d

}

else {

    # Release mode

    cuda.input = CUDA_SOURCES

    cuda.output = $$CUDA_OBJECTS_DIR/${QMAKE_FILE_BASE}_cuda.o

    cuda.commands = $$CUDA_DIR/bin/nvcc $$NVCC_OPTIONS $$CUDA_INC $$N
VCC_LIBS --machine $$SYSTEM_TYPE -arch=$$CUDA_ARCH -c -o ${QMAKE_FILE_
OUT} ${QMAKE_FILE_NAME}

    cuda.dependency_type = TYPE_C

```

}

[illegible]