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

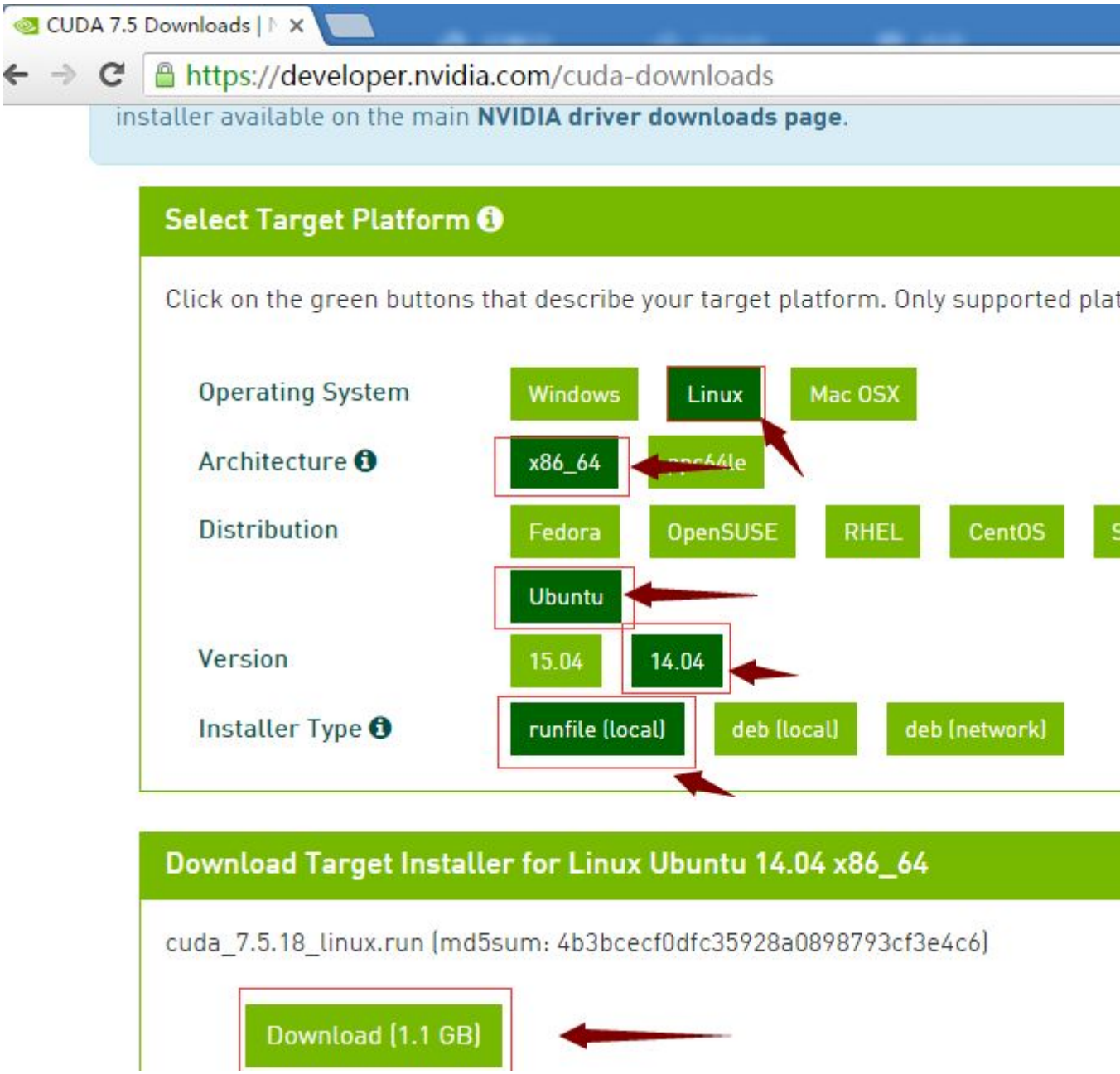
8:关闭 x 服务

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

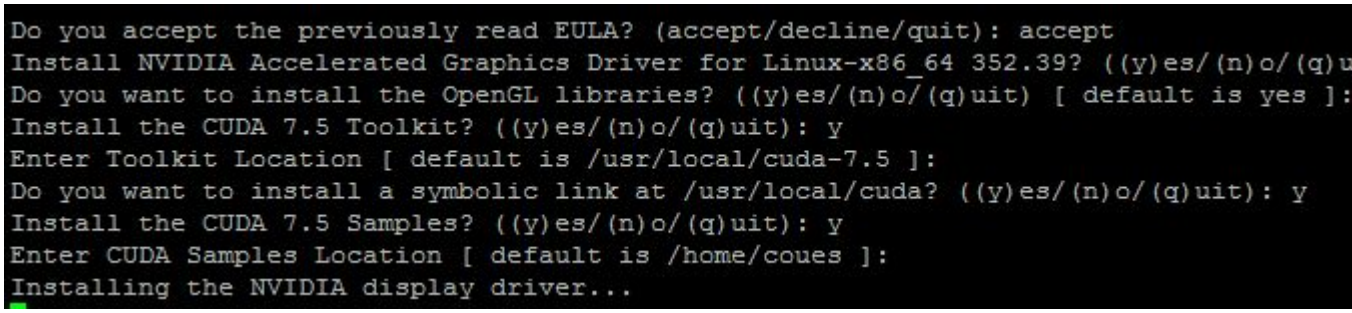
9：安装包下载位置

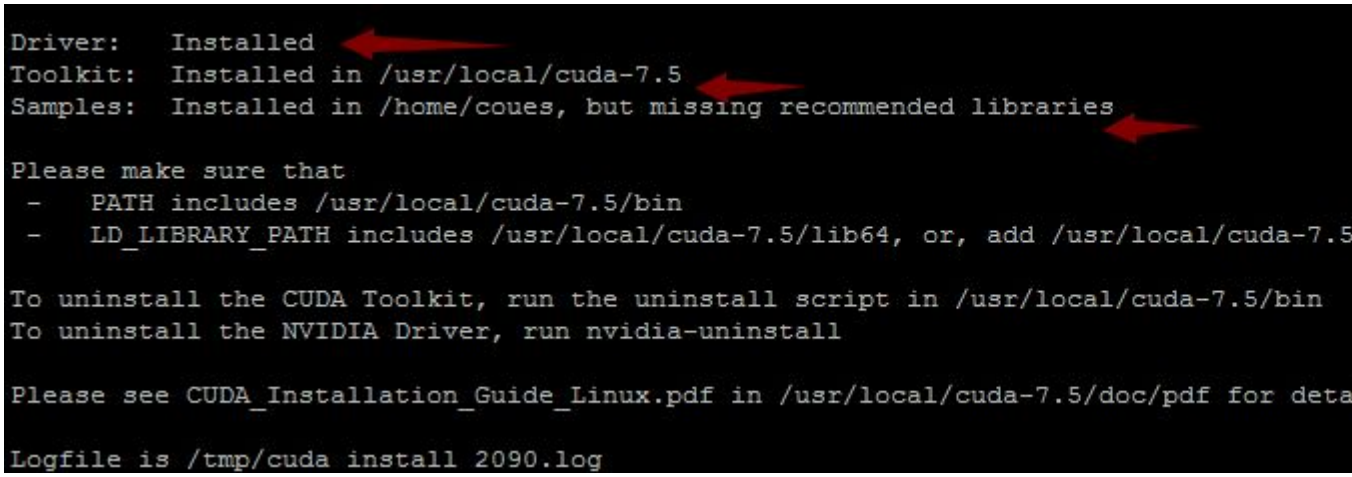
url：https://developer.nvidia.com/cuda-downloads



10： 安装

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





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:$CYDA/lib64:$CUDA/lib:/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

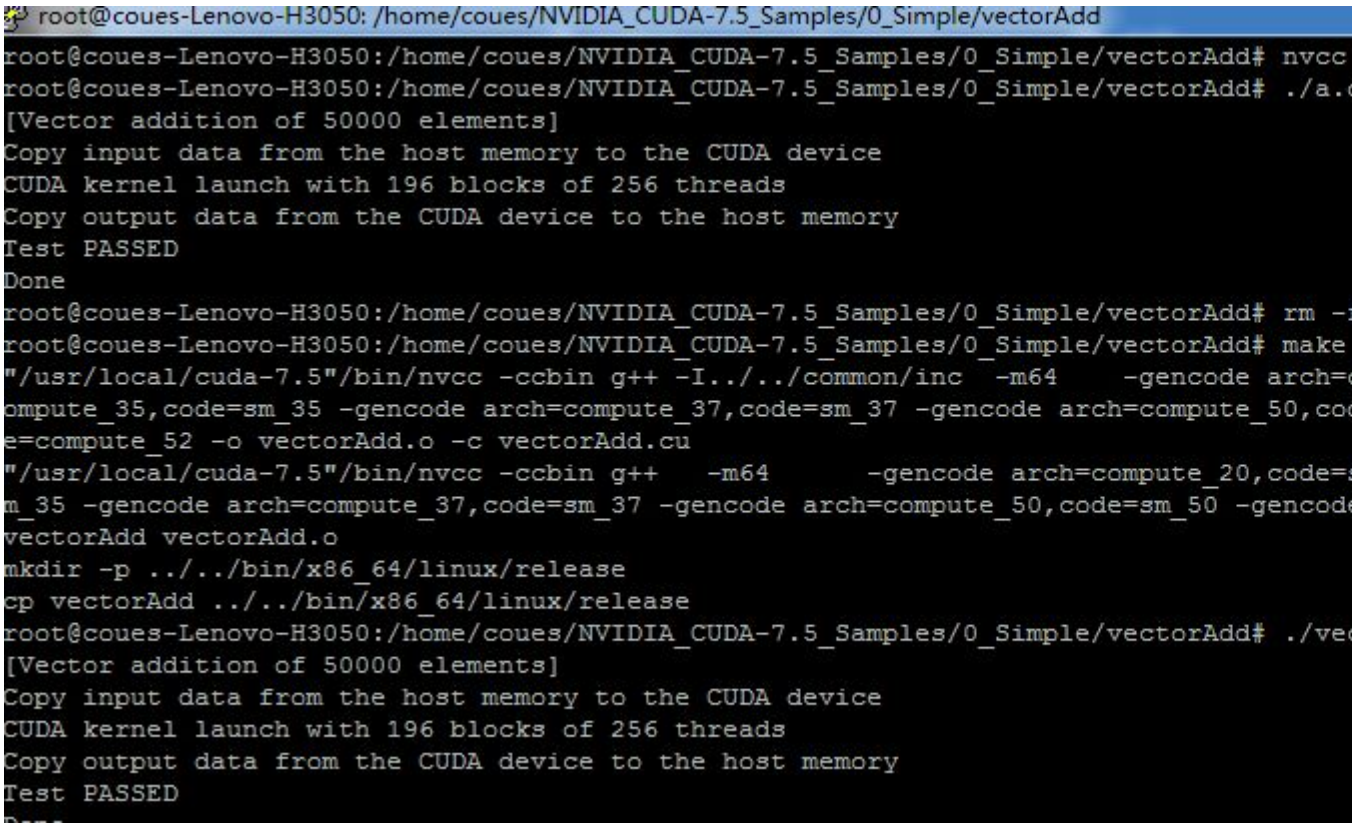
Copyright (c) 2005-2015 NVIDIA Corporation

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



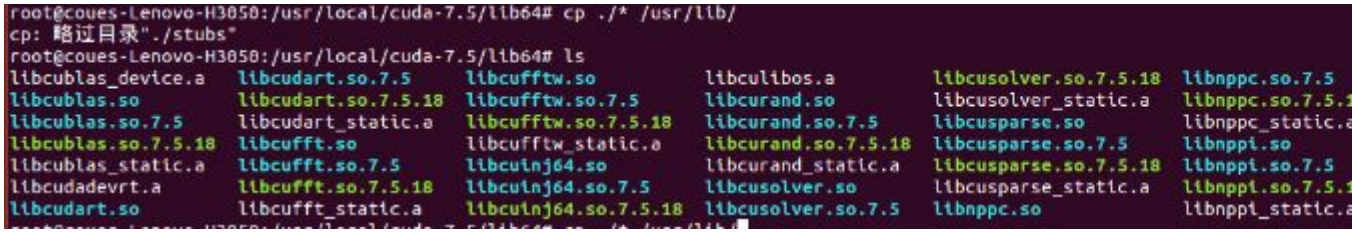
配置 IDE

15:配置 QT creator　开发环境

配置链接库,有两种方法

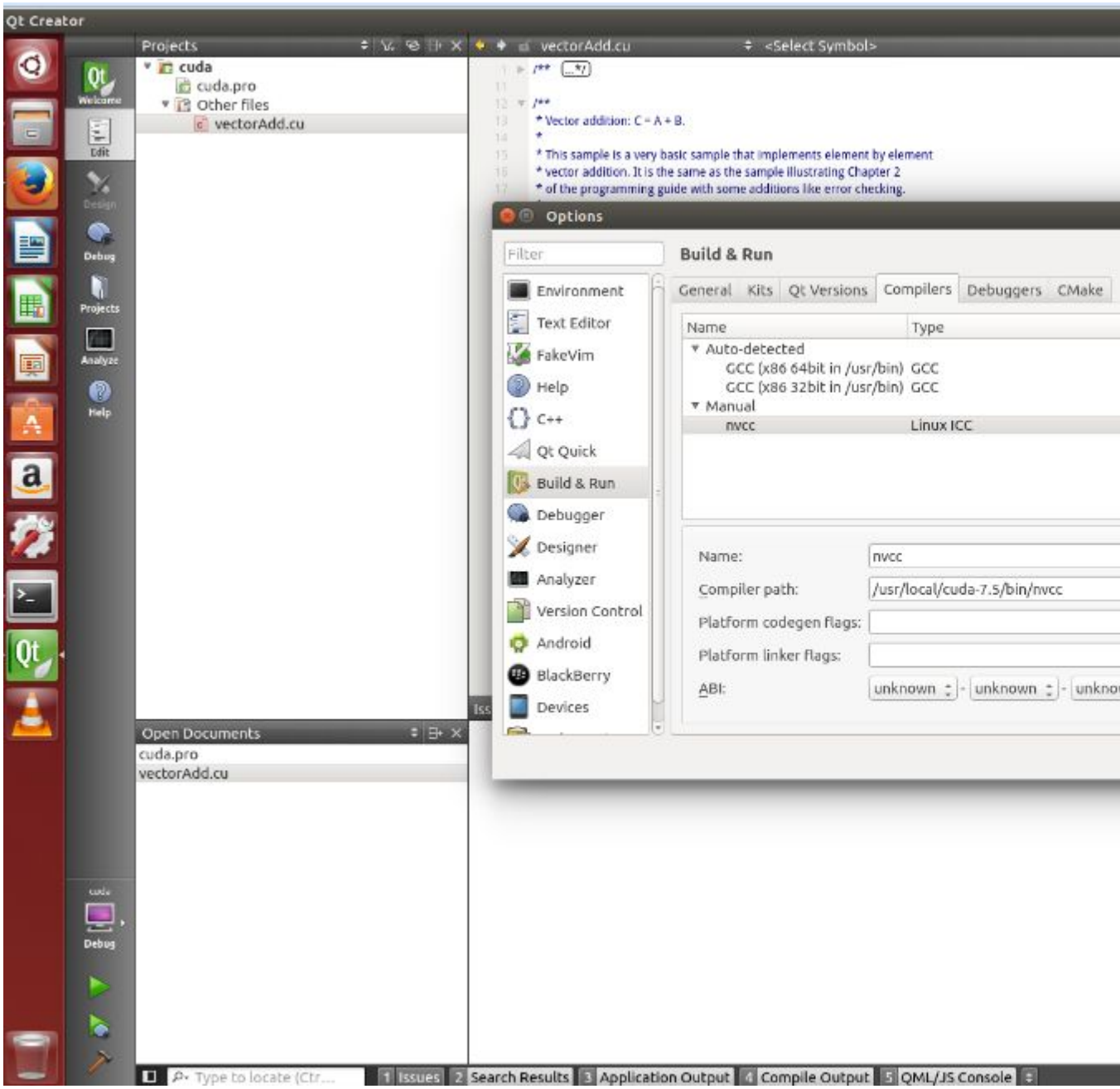
a:在/etc/ld.so.conf 里面配置，加入"include /usr/loca/cuda-7.5/lib64",执行"ldconfig"

b:直接拷贝



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
```

QMAKE_EXTRA_COMPILERS += cuda

}

18:运行结果