Install Cuda 6.5 with Nvidia Qadro K2100M Graphics in Ubuntu 12.04

Step 1: download the cuda installer from the official website. Select Linux x86, choose Ubuntu 12.04 RUN* . You could also download the installation guide from the website, but basically this tutorial is enough.
Download the Cuda .deb installer for ubuntu (much simplier than .run installer) , because the .Deb file will **automatically uninstall the previous Nvidia driver and automatically install the dependencies**.

 # However, .RUN file is prefer if you have the Nvidia driver installed in you machine.

Step 2: Install Nvidia driver using Ubuntu external driver detector, and install the Nvidia driver, the good point is that it will automatically choose the compatible Nvidia driver for your Graphic card. Sometimes, the official one has confliction with Ubuntu. Then you can use Synaptic to install the Nvidia-Cuda-Toolkit, and all the dependencies will be installed automatically.

Step 3. Run the installation commands (Note: symbol # means explaination, symbol $ means the commands in the terminal, following the same.)

$ *sudo sh cuda_5.5.22_linux_64.run*

during the installation, you will be asked to choose some options:
1. install Nvidia driver? No.
2. install Cuda-Toolkit? Yes.
3. install Samples? Yes.
4. create link …? Yes.

Environment Setup:

----------**ADD LIBRARY**----------
go to the folder contains the configure files

$ *cd /etc/ld.so.conf.d/*

add your own library path

$ *sudo nano filename.conf*

add the path of the library inside the filename.conf file
for example:
    # Add nvidia-331 path, this is the Nvidia driver installed by Ubuntu.
    /usr/lib/nvidia-331

Add cuda-5.5 path
/usr/local/cuda-5.5/lib64

then reload the configure file to active the created configure file.
$ *sudo ldconfig*
----------**Add PATH**----------
open the .bashrc file

$ *cd ~*
$ *sudo .bashrc*

Add the path like the following format, to the end of files

export PATH=/usr/local/cuda-5.5/bin:$PATH

save changes and **REBOOT** your system.

Step 4: Install Cuda samples to verify your installation
check your driver version
$ *cat /proc/driver/nvidia/version*
++++++++++++++++++++++++++++++++++++
jiang@Cansen-HP:~$ cat /proc/driver/nvidia/version
NVRM version: NVIDIA UNIX x86_64 Kernel Module 331.38 Wed Jan 8 19:32:30 PST 2014
GCC version: gcc version 4.6.3 (Ubuntu/Linaro 4.6.3-1ubuntu5)
++++++++++++++++++++++++++++++++++++

check CUDA TOOLKIT version
$ *nvcc -V*
++
jiang@Cansen-HP:~$ nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2013 NVIDIA Corporation
Built on Wed_Jul_17_18:36:13_PDT_2013
Cuda compilation tools, release 5.5, V5.5.0
jiang@Cansen-HP:~$
++

then go to the sample directory and build the samples
$ *cd NVIDIA_CUDA-5.5_Samples/*
$ *make*

you will see the following info:
++
….
Finished building CUDA samples

$jiang@Cansen-HP:~$
++

Step 5. Run samples
go to release folder and run deviceQuery
$ *cd ~/NVIDIA_CUDA-5.5_Samples/bin/linux/release/*
$ *./deviceQuery*

if you see the **error**

```
$ sudo ./deviceQuery
        • ./deviceQuery Starting...
        •
        • CUDA Device Query (Runtime API) version (CUDART static linking)
        •
        • FATAL: Module nvidia_uvm not found.
        • cudaGetDeviceCount returned 30
        • -> unknown error
        • Result = FAIL
```

Solution is here:

```
$ sudo update-alternatives --config x86_64-linux-gnu_gl_conf
  Selection    Path                                            Priority   Status
------------------------------------------------------------
  0            /usr/lib/nvidia-331/ld.so.conf                   8604      auto mode
* 1            /usr/lib/nvidia-331-prime/ld.so.conf             8603      manual mode
  2            /usr/lib/nvidia-331/ld.so.conf                   8604      manual mode
  3            /usr/lib/x86_64-linux-gnu/mesa/ld.so.conf        500       manual mode

# basically, the solution here you need to change the status to be prime.
```

run again the command, you will see:
++++++++++++++++++++++++++
jiang@Cansen-HP:~/NVIDIA_CUDA-5.5_Samples/NVIDIA_CUDA-5.5_Samples/bin/x86_64/linux/release$ ./deviceQuery
./deviceQuery Starting...

CUDA Device Query (Runtime API) version (CUDART static linking)

Detected 1 CUDA Capable device(s)

Device 0: "Quadro K2100M"
  CUDA Driver Version / Runtime Version   6.0 / 5.5
  CUDA Capability Major/Minor version number: 3.0
  Total amount of global memory: 2048 MBytes (2147287040 bytes)
  (3) Multiprocessors, (192) CUDA Cores/MP: 576 CUDA Cores
  GPU Clock rate: 667 MHz (0.67 GHz)
  Memory Clock rate: 1504 Mhz

Memory Bus Width: 128-bit
L2 Cache Size: 262144 bytes
Maximum Texture Dimension Size (x,y,z) 1D=(65536), 2D=(65536, 65536), 3D=(4096, 4096, 4096)
Maximum Layered 1D Texture Size, (num) layers 1D=(16384), 2048 layers
Maximum Layered 2D Texture Size, (num) layers 2D=(16384, 16384), 2048 layers
Total amount of constant memory: 65536 bytes
Total amount of shared memory per block: 49152 bytes
Total number of registers available per block: 65536
Warp size: 32
Maximum number of threads per multiprocessor: 2048
Maximum number of threads per block: 1024
Max dimension size of a thread block (x,y,z): (1024, 1024, 64)
Max dimension size of a grid size (x,y,z): (2147483647, 65535, 65535)
Maximum memory pitch: 2147483647 bytes
Texture alignment: 512 bytes
Concurrent copy and kernel execution: Yes with 1 copy engine(s)
Run time limit on kernels: Yes
Integrated GPU sharing Host Memory: No
Support host page-locked memory mapping: Yes
Alignment requirement for Surfaces: Yes
Device has ECC support: Disabled
Device supports Unified Addressing (UVA): Yes
Device PCI Bus ID / PCI location ID: 1 / 0
Compute Mode:
< Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >

deviceQuery, CUDA Driver = CUDART, CUDA Driver Version = 6.0, CUDA Runtime Version = 5.5, NumDevs = 1, Device0 = Quadro K2100M
Result = PASS
++++++++++++++++++++++++++++

then run the bandwidthTest program to ensure the system and the cuda-capable device are able to communicate correctly.
++
jiang@Cansen-HP:~/NVIDIA_CUDA-5.5_Samples/NVIDIA_CUDA-5.5_Samples/bin/x86_64/linux/release$ ./bandwidthTest
[CUDA Bandwidth Test] - Starting...
Running on...

Device 0: Quadro K2100M
Quick Mode

Host to Device Bandwidth, 1 Device(s)
PINNED Memory Transfers

Transfer Size (Bytes)	Bandwidth(MB/s)
33554432	10059.9

Device to Host Bandwidth, 1 Device(s)
PINNED Memory Transfers

Transfer Size (Bytes)	Bandwidth(MB/s)
33554432	10054.9

Device to Device Bandwidth, 1 Device(s)
PINNED Memory Transfers

Transfer Size (Bytes)	Bandwidth(MB/s)
33554432	34225.1

Result = PASS

jiang@Cansen-HP:~/NVIDIA_CUDA-5.5_Samples/NVIDIA_CUDA-5.5_Samples/bin/x86_64/linux/release$ cd NVIDIA_CUDA-5.5_Samples/

++++++++++++++++++++++++++++++++++

Congrats! Everything is DONE now!

Contact: Cansen.Jiang@u-bourgogne.fr