**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 DEB* . 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**.

Step 2: run the installation commands (Note: symbol # means explaination, symbol $ means the commands in the terminal, following the same.)

```
$ sudo dpkg -i cuda-repo-ubuntu1204_6.5-14_amd64.deb
$ sudo apt-get update
$ sudo apt-get install cuda
```

## Step 3: 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-340 path
#  # $ /usr/lib/nvidia-340

#  # Add cuda-6.5 path
#  # $/usr/local/cuda-6.5/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-6.5/bin:$PATH

# save changes and restart your computer.
```

Step 4: Install Cuda samples to verify your installation
# copy the cuda samples to your home directory <dir>, in my case, /home/jiang/
$ cuda-install-samples-6.5.sh <dir>

# check your driver version
$ cat /proc/driver/nvidia/version

# check CUDA TOOLKIT version
$ nvcc -V

# then go to the sample directory and build the samples
$ cd ~/NVIDIA_CUDA-6.5_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-6.5_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-6.5_Samples/bin$ ./x86_64/linux/release/deviceQuery

```
./x86_64/linux/release/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          6.5 / 6.5
  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:                     No
  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 = 6.5, CUDA Runtime Version = 6.5,
NumDevs = 1, Device0 = Quadro K2100M
Result = PASS
jiang@Cansen-HP:~/NVIDIA_CUDA-6.5_Samples/bin$
++++++++++++++++++++++++++++++
```

# then run the bandwidthTest program to ensure the system and the cuda-capable device are able to communicate correctly.

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

```
jiang@Cansen-HP:~/NVIDIA_CUDA-6.5_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                     10288.9

 Device to Host Bandwidth, 1 Device(s)
 PINNED Memory Transfers
   Transfer Size (Bytes)        Bandwidth(MB/s)
   33554432                     10292.5

 Device to Device Bandwidth, 1 Device(s)
 PINNED Memory Transfers
   Transfer Size (Bytes)        Bandwidth(MB/s)
   33554432                     35606.4

Result = PASS
jiang@Cansen-HP:~/NVIDIA_CUDA-6.5_Samples/bin/x86_64/linux/release$
++++++++++++++++++++++++++++++++++++
```

**Congrats! Everything is DONE now!**