# Matching SM architectures (CUDA arch and CUDA gencode) for various NVIDIA cards

I’ve seen some confusion regarding NVIDIA’s nvcc sm flags and what they’re used for:

When compiling with NVCC, the arch flag (‘**-arch**‘) specifies the name of the NVIDIA GPU architecture that the CUDA files will be compiled for.  
Gencodes (‘**-gencode**‘) allows for more PTX generations, and can be repeated many times for different architectures.

## When should different ‘gencodes’ or ‘cuda arch’ be used?

When you compile CUDA code, you should always compile only one ‘**-arch**‘ flag that matches your most used GPU cards. This will enable faster runtime, because code generation will occur during compilation.  
If you only mention ‘**-gencode**‘, but omit the ‘**-arch**‘ flag, the GPU code generation will occur on the **JIT** compiler by the CUDA driver.

When you want to speed up CUDA compilation, you want to reduce the amount of irrelevant ‘**-gencode**‘ flags. However, sometimes you may wish to have better CUDA backwards compatibility by adding more comprehensive ‘**-gencode**‘ flags.

Find out which GPU you have, and [which CUDA version you have](http://arnon.dk/check-cuda-installed/) first.

Supported SM and Gencode variations

Below are the supported sm variations and sample cards from that generation

Supported on CUDA 7 and later

* **Fermi** **(CUDA 3.2 until CUDA 8)** (deprecated from CUDA 9):
* SM20 or SM\_20, compute\_30 — Older cards such as **GeForce 400, 500, 600, GT-630**
* **Kepler (CUDA 5 and later)**:
* SM30 or SM\_30, compute\_30 — Kepler architecture (generic — **Tesla K40/K80, GeForce 700, GT-730**)  
  Adds support for unified memory programming
* SM35 or SM\_35, compute\_35 — More specific **Tesla K40**Adds support for dynamic parallelism. Shows no real benefit over SM30 in my experience.
* SM37 or SM\_37, compute\_37 — More specific **Tesla K80**Adds a few more registers. Shows no real benefit over SM30 in my experience
* **Maxwell (CUDA 6 and later)**:
* SM50 or SM\_50, compute\_50 — **Tesla/Quadro M series**
* SM52 or SM\_52, compute\_52 — **Quadro M6000 , GeForce 900, GTX-970, GTX-980, GTX Titan X**
* SM53 or SM\_53, compute\_53 — **Tegra (Jetson) TX1 / Tegra X1**
* **Pascal (CUDA 8 and later)**
* SM60 or SM\_60, compute\_60 — Quadro GP100, **Tesla P100,** DGX-1 (Generic Pascal)
* SM61 or SM\_61, compute\_61 — **GTX 1080, GTX 1070, GTX 1060, GTX 1050, GTX 1030, Titan Xp, Tesla P40, Tesla P4, Discrete GPU on the NVIDIA Drive PX2**
* SM62 or SM\_62, compute\_62 — **Integrated GPU on the NVIDIA Drive PX2, Tegra (Jetson) TX2**
* **Volta (CUDA 9 and later)**
* SM70 or SM\_70, compute\_70 — DGX-1 with Volta, **Tesla V100, GTX 1180 (GV104), Titan V, Quadro GV100**
* SM72 or SM\_72, compute\_72 — Jetson AGX Xavier
* **Turing (CUDA 10 and later)**
* SM75 or SM\_75, compute\_75 — GTX Turing — GTX 1660 Ti, RTX 2060, RTX 2070, **RTX 2080, Titan RTX,**Quadro RTX 4000, Quadro RTX 5000, Quadro RTX 6000, Quadro RTX 8000

## Sample Flags

According to NVIDIA:

*The arch= clause of the -gencode= command-line option to nvcc specifies the front-end compilation target and must always be a PTX version. The code= clause specifies the back-end compilation target and can either be cubin or PTX or both. Only the back-end target version(s) specified by the code= clause will be retained in the resulting binary; at least one must be PTX to provide Volta compatibility.*

#Sample flags for generation on CUDA 7 for maximum compatibility:

-arch=sm\_30 \  
-gencode=arch=compute\_20,code=sm\_20 \  
-gencode=arch=compute\_30,code=sm\_30 \  
-gencode=arch=compute\_50,code=sm\_50 \  
-gencode=arch=compute\_52,code=sm\_52 \  
-gencode=arch=compute\_52,code=compute\_52  
  
  
#Sample flags for generation on CUDA 8 for maximum compatibility:

-arch=sm\_30 \  
-gencode=arch=compute\_20,code=sm\_20 \  
-gencode=arch=compute\_30,code=sm\_30 \  
-gencode=arch=compute\_50,code=sm\_50 \  
-gencode=arch=compute\_52,code=sm\_52 \  
-gencode=arch=compute\_60,code=sm\_60 \  
-gencode=arch=compute\_61,code=sm\_61 \  
-gencode=arch=compute\_61,code=compute\_61

#Sample flags for generation on CUDA 9 for maximum compatibility with Volta cards.   
Note the removed SM\_20:

-arch=sm\_50 \  
-gencode=arch=compute\_50,code=sm\_50 \  
-gencode=arch=compute\_52,code=sm\_52 \  
-gencode=arch=compute\_60,code=sm\_60 \  
-gencode=arch=compute\_61,code=sm\_61 \  
-gencode=arch=compute\_70,code=sm\_70 \   
-gencode=arch=compute\_70,code=compute\_70

#Sample flags for generation on CUDA 10 for maximum compatibility with Turing cards:

-arch=sm\_50 \   
-gencode=arch=compute\_50,code=sm\_50 \   
-gencode=arch=compute\_52,code=sm\_52 \   
-gencode=arch=compute\_60,code=sm\_60 \   
-gencode=arch=compute\_61,code=sm\_61 \   
-gencode=arch=compute\_70,code=sm\_70 \   
-gencode=arch=compute\_75,code=sm\_75 \  
-gencode=arch=compute\_75,code=compute\_75

PATH=/usr/local/cuda/bin:$PATH   
make -j$(nproc)

[**Install NVIDIA DIGITS On Ubuntu 18.04**](https://medium.com/@patrickorcl/install-nvidia-digits-on-ubuntu-18-04-2d097ddd560?source=follow_footer---------0----------------------------)

Ref Link:  
<https://github.com/dusty-nv/jetson-inference#system-setup>

Installing the NVIDIA driver

Add the NVIDIA Developer repository and install the NVIDIA driver.

$ sudo apt-get install -y apt-transport-https curl build-essential  
$ cat <<EOF | sudo tee /etc/apt/sources.list.d/cuda.list > /dev/null  
deb [https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1604/x86\_64](https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64/) /  
EOF  
$ curl -s \  
 [https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1604/x86\_64/7fa2af80.pub](https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64/7fa2af80.pub) \  
 | sudo apt-key add -  
$ cat <<EOF | sudo tee /etc/apt/preferences.d/cuda > /dev/null  
Package: \*  
Pin: origin developer.download.nvidia.com  
Pin-Priority: 600  
EOF  
$ sudo apt-get update && sudo apt-get install -y --no-install-recommends cuda-drivers  
$ sudo reboot

After reboot, check if you can run nvidia-smi and see if your GPU shows up.

$ nvidia-smi  
Thu May 31 11:56:44 2018  
+-----------------------------------------------------------------------------+  
| NVIDIA-SMI 390.30 Driver Version: 390.30 |  
|-------------------------------+----------------------+----------------------+  
| GPU Name Persistence-M| Bus-Id Disp.A | Volatile Uncorr. ECC |  
| Fan Temp Perf Pwr:Usage/Cap| Memory-Usage | GPU-Util Compute M. |  
|===============================+======================+======================|  
| 0 Quadro GV100 Off | 00000000:01:00.0 …