CUDA for Tegra Notes

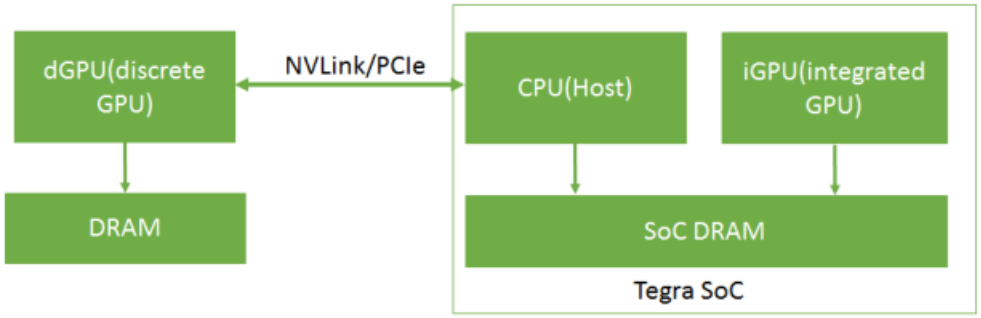
CUDA version: 10.2 (Jetson Nano CUDA version)

The document provide consideration for porting code from GPU (dGPU) with x86 system to Tegra system (iGPU).

Memory Management

In Tegra device both the CPU (Host) and the iGPU share DRAM memory.

An overview of a dGPU-connected Tegra® memory system is shown in Figure 1.



Porting Considerations

- Select an appropriate memory buffer
- Select between iGPU and dGPU

Memory Selection

Device Memory
Should be used for buffers whose accessibility is limited to the iGPU.

Pageable Host Memory
Should be used for buffers whose accessibility is limited to the CPU.

Pinned Memory
For compute capabilities greater or equal to 7.2, pinned memory are I/O coherent and CPU access time of pinned memory is as good as pageable host because it is cached on the CPU. Otherwise, CPU access time is higher.

Unified Memory
Unified memory is cached on the iGPU and the CPU. On Tegra system, using unified memory. On system with compute capability of 7.2 or greater, large buffers which are frequently accessed by the iGPU and the CPU and the accesses on iGPU are repetitive, unified memory is preferable. On system with compute capability less than 7.2, when iGPU accesses are not repetitive, unified memory is still preferable.

GPU Selection

Kernel execution time, data transfer time and data locality and latency should be taken into consideration. To run an application on dGPU, data must be transferred between SoC and the dGPU. If application run on iGPU, such data transfer can be avoided.