# CMPSCI 691AD – General Purpose Computation on the GPU

*Spring 2009*

Lecture 2: Graphics Hardware I

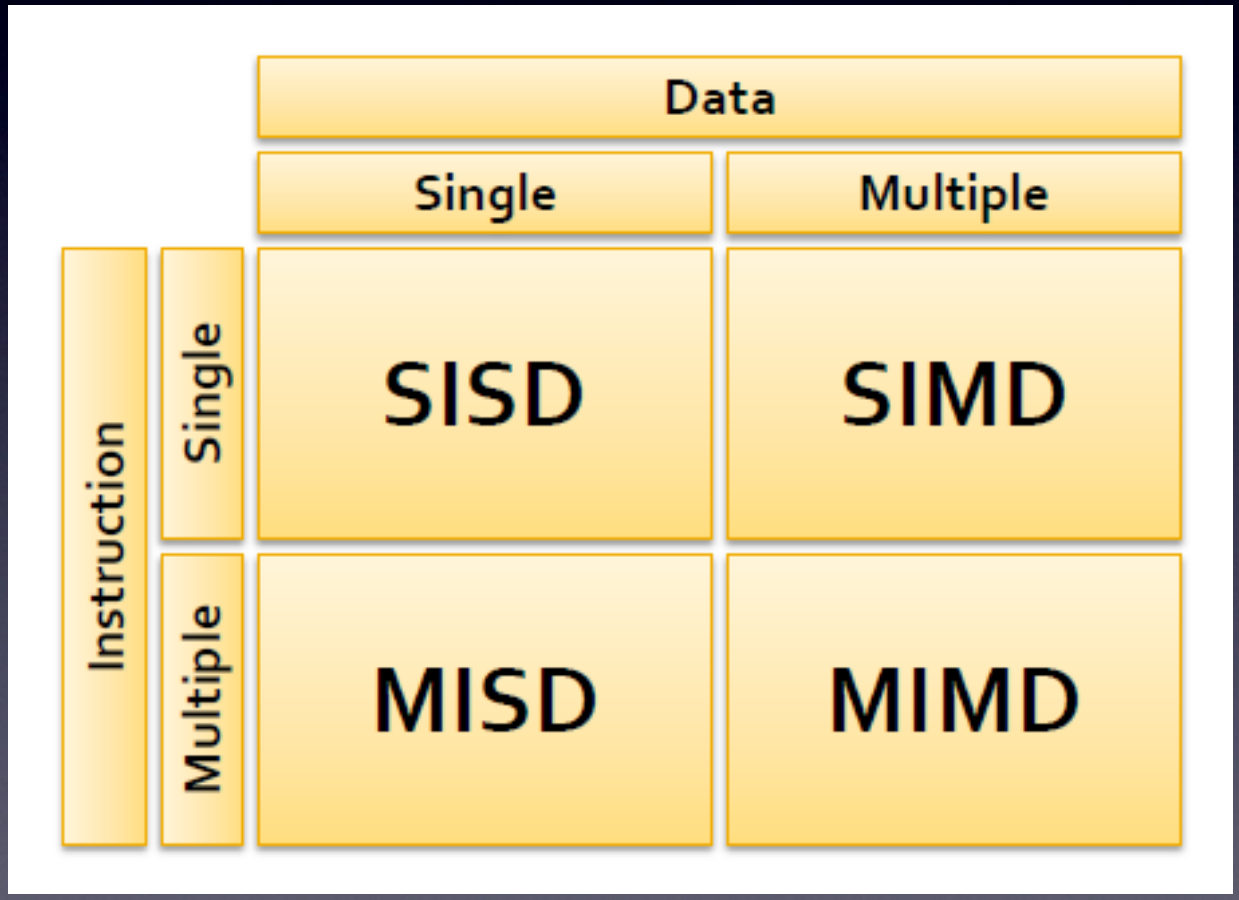
*Rui Wang*

# Terms you should know...

- Host → CPU
- Device → GPU
- PCI: Peripheral Component Interconnect
- PCI-E (PCIe): PCI Express
- SIMD: Single Instruction, Multiple Data

# Taxonomy of Parallel Architecture

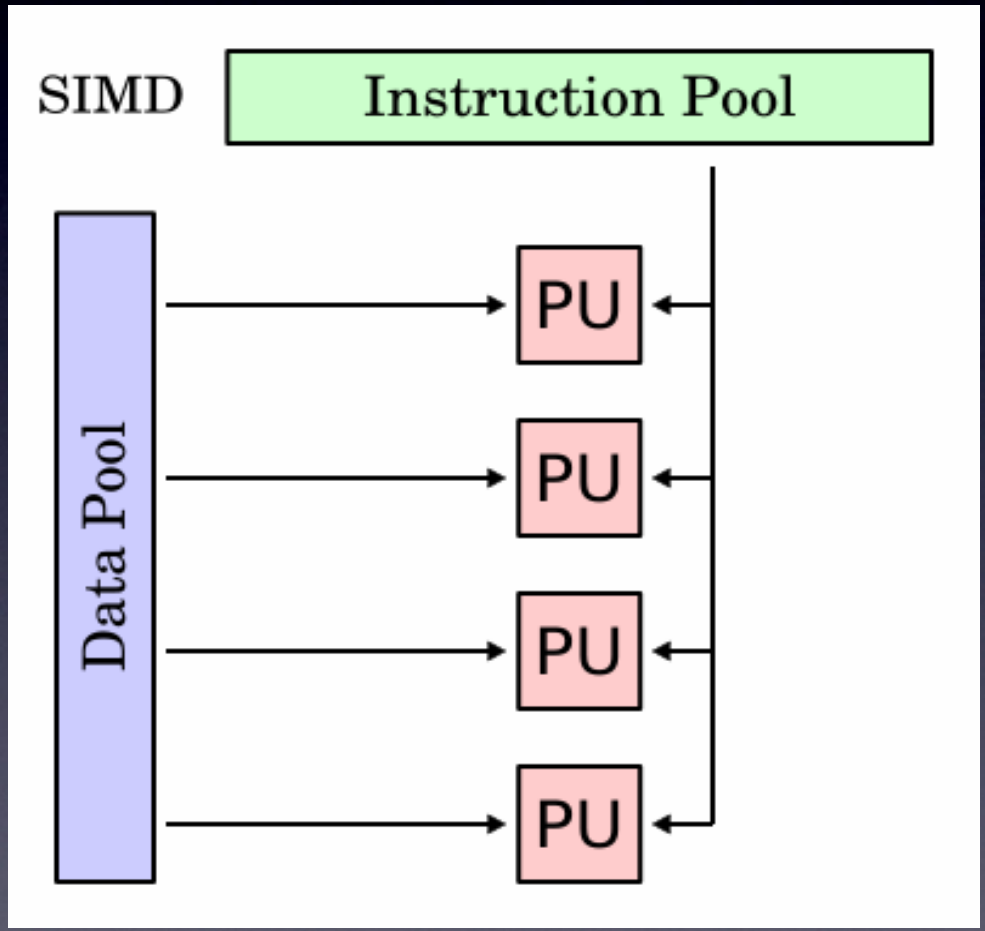
- Flynn's Taxonomy




*http://www.cs.jhu.edu/~bolitho/Teaching/GPGPU/Lectures/*

# Taxonomy of Parallel Architecture

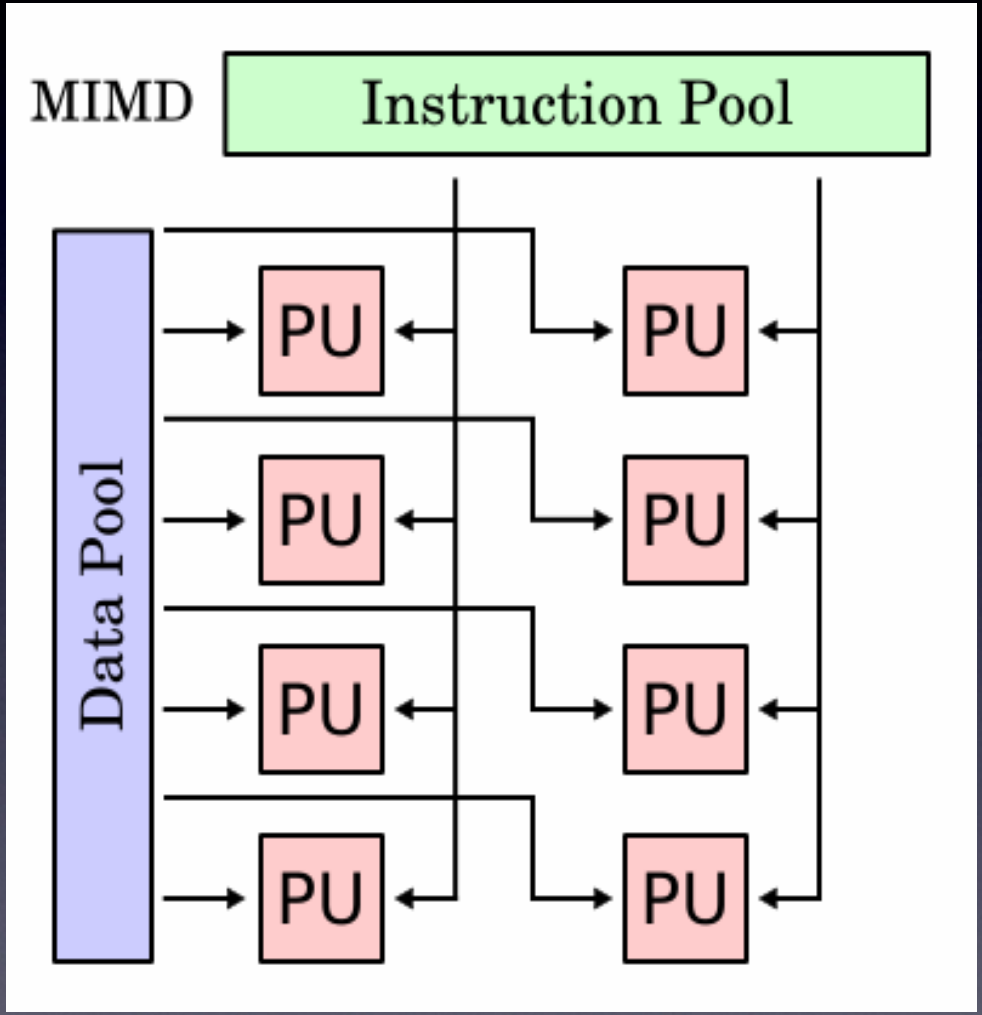
- SIMD
  - SSE
  - GPU
  - …




*Wikipedia*

# Taxonomy of Parallel Architecture

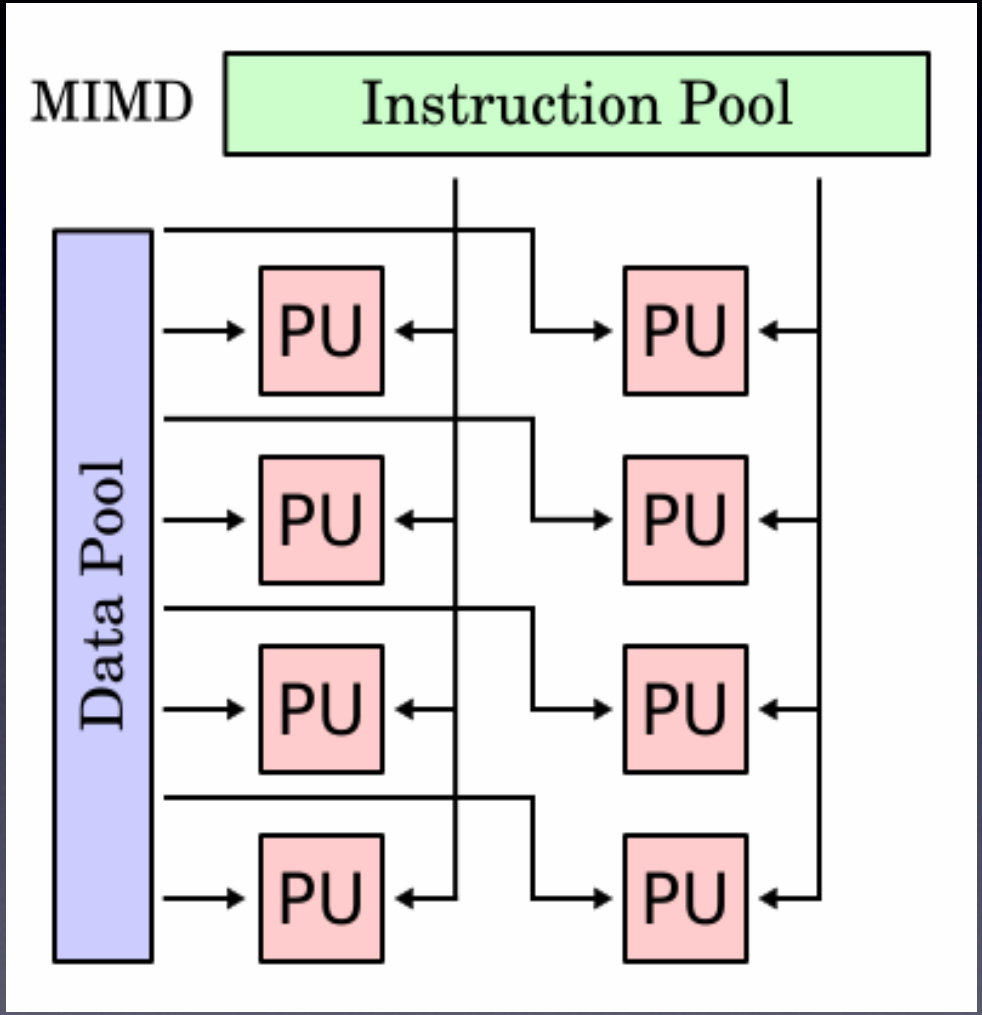
- MIMD
  - Shared Memory Model
    - SMP
    - NUMA
  - Distributed Memory Model
    - MPP
    - Clusters
  - Hybrid
  - Grid
  - …




*Wikipedia*

# Taxonomy of Parallel Architecture

- MIMD
  - Shared Memory Model
    - SMP
    - NUMA
  - Distributed Memory Model
    - MPP
    - Clusters
  - Hybrid
  - Grid
  - …




*Wikipedia*

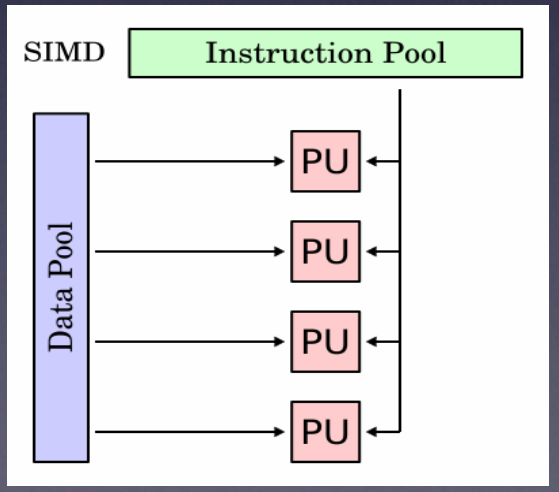
# Amdahl's Law

- The speedup you can gain is limited by the portion that requires sequential computation.
  - Let P be the portion that can be parallelized, S be the speedup of the parallel portion.
  - Overall speedup: $\frac{1}{(1-P)+\frac{P}{S}}$

    Maximum speedup is limited to $\frac{1}{(1-P)}$ even if S is infinity.
- A problem where P=100% is called embarrassingly parallel.

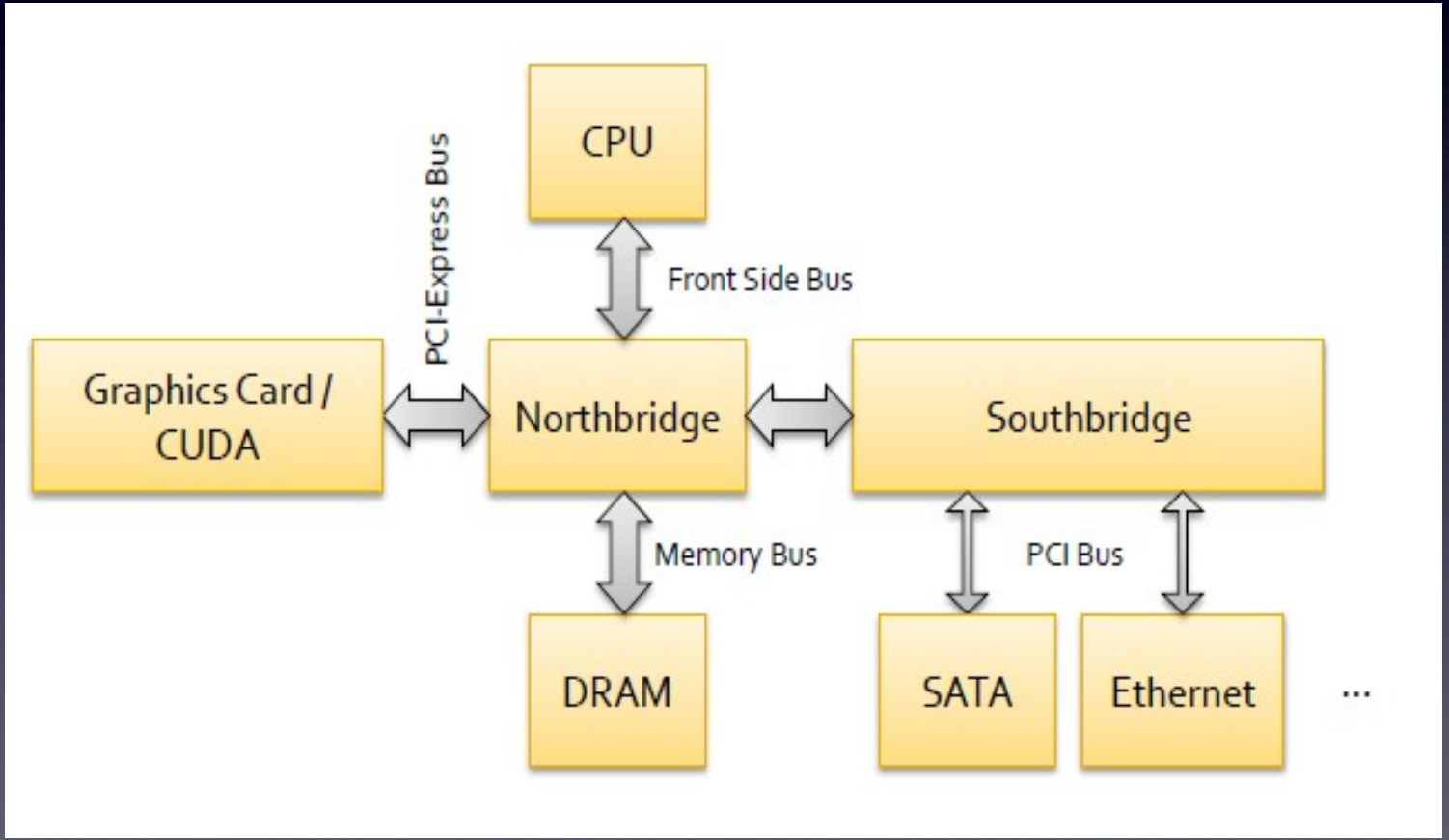
# Taxonomy of Parallel Architecture

- GPU is a SIMD device, works on “streams” of data.
- Each “stream processor” executes one general instruction on the stream of data that it is assigned to handle.
- Executes many threads in parallel
  - Called SIMT (Single Instruction Multiple Threads) by NVIDIA



# PC Architecture

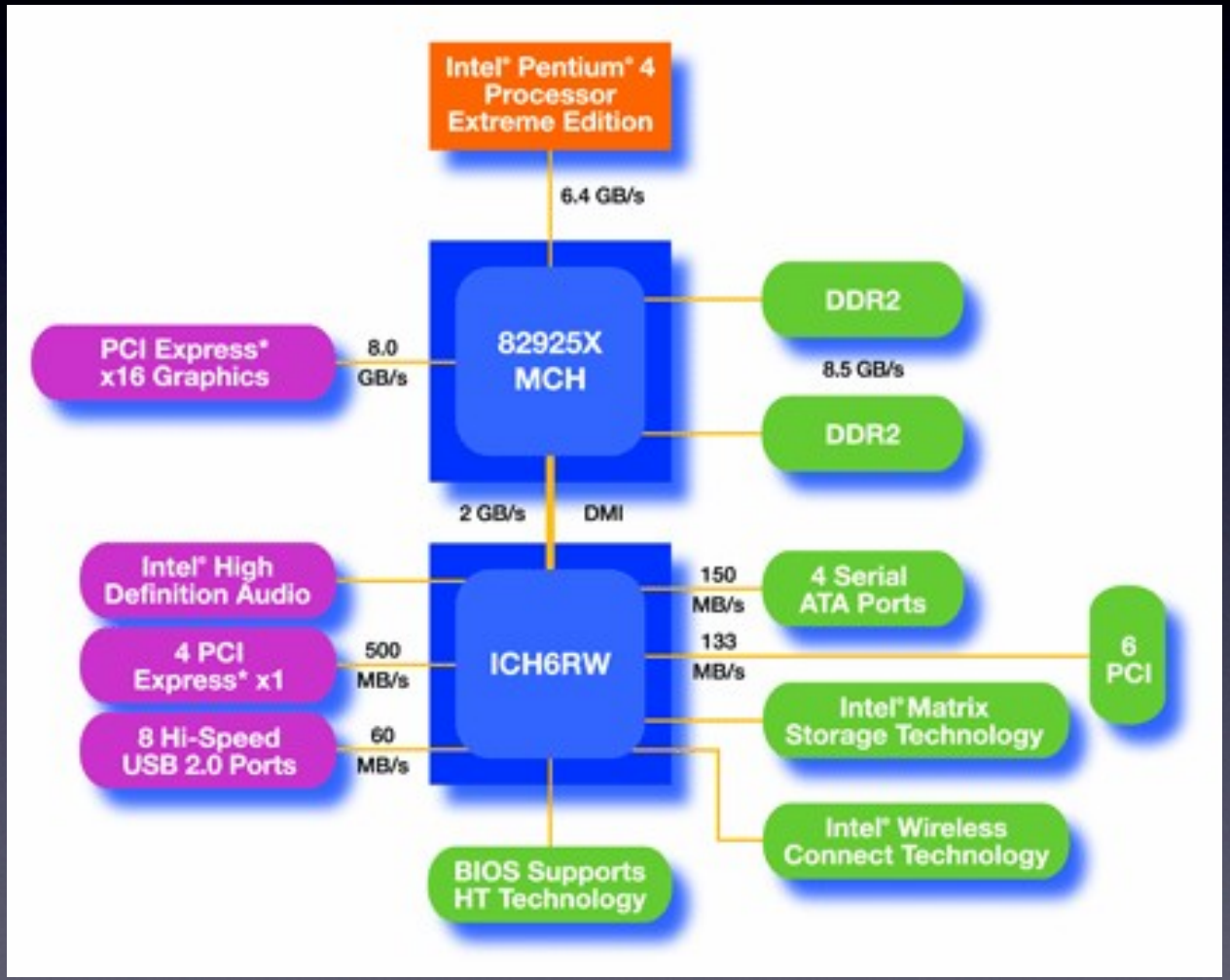
- Northbridge → Memory Controller Hub (MCH)
- Southbridge → I/O Controller Hub (ICH)




*http://www.cs.jhu.edu/~bolitho/Teaching/GPGPU/Lectures/Lecture3.pdf*

# PC Architecture

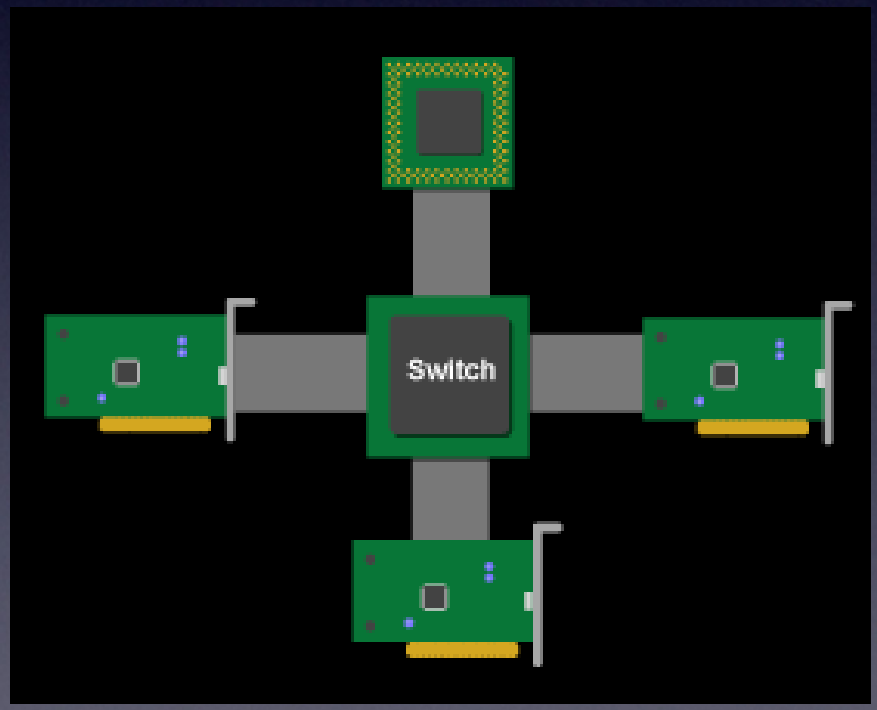
- Example:




*http://courses.ece.uiuc.edu/ece498/al/lectures/*

# PCI Express (PCI-E)

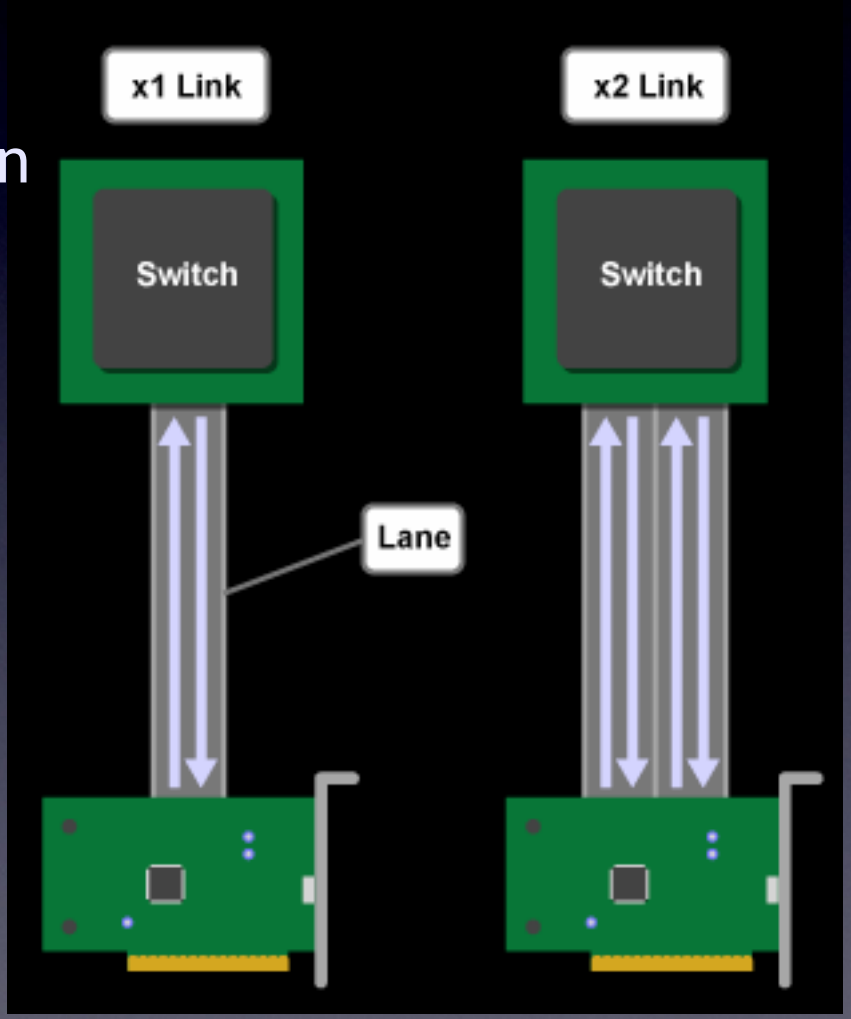
- Switched, Serial, P2P link
- Each card has a dedicated link to the central switch, no bus arbitration.




*http://courses.ece.uiuc.edu/ece498/al/lectures/*

# PCI Express (PCI-E)

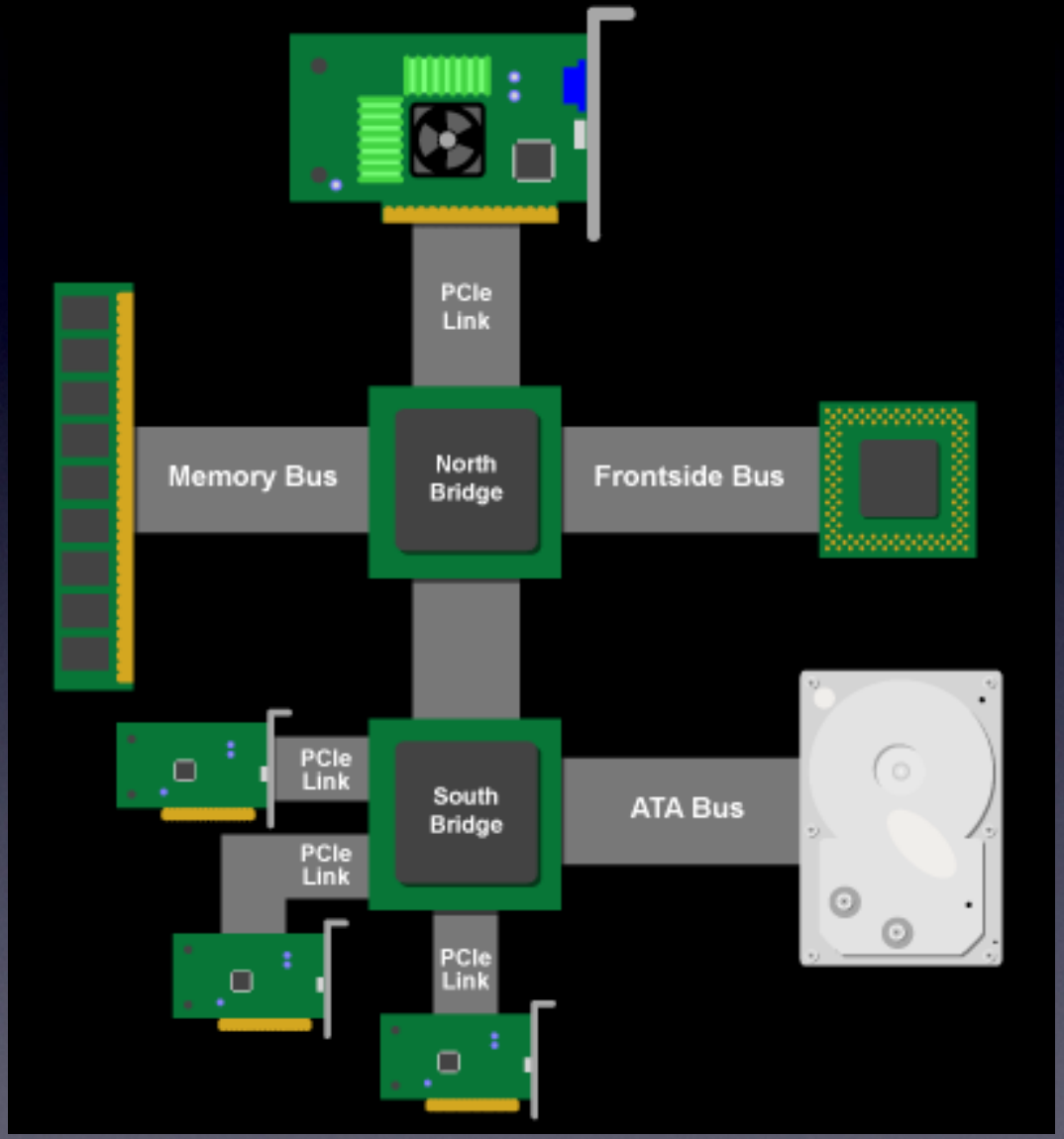
- Each link is duplex
- PCIe 1.0:
  250 MB/s per link each direction
- Can have multiple links:
  x1, x2, x4, x8, x16
- Therefore 4GB/s each direction for x16
- PCIe 2.0:
  500 MB/s
- 3.0:
  1 GB/s




*http://courses.ece.uiuc.edu/ece498/al/lectures/*

# PCI Express (PCI-E)

- PCIe forms the interconnect backbone




*http://courses.ece.uiuc.edu/ece498/al/lectures/*

# GPU

- The GPU is connected to the CPU PCIe x16
  - The idea is to use the GPU as a co-processor
  - Dispatch big parallelizable tasks to the GPU
  - Keep the CPU busy with the control of the execution
  - No direct access to CPU memory or devices

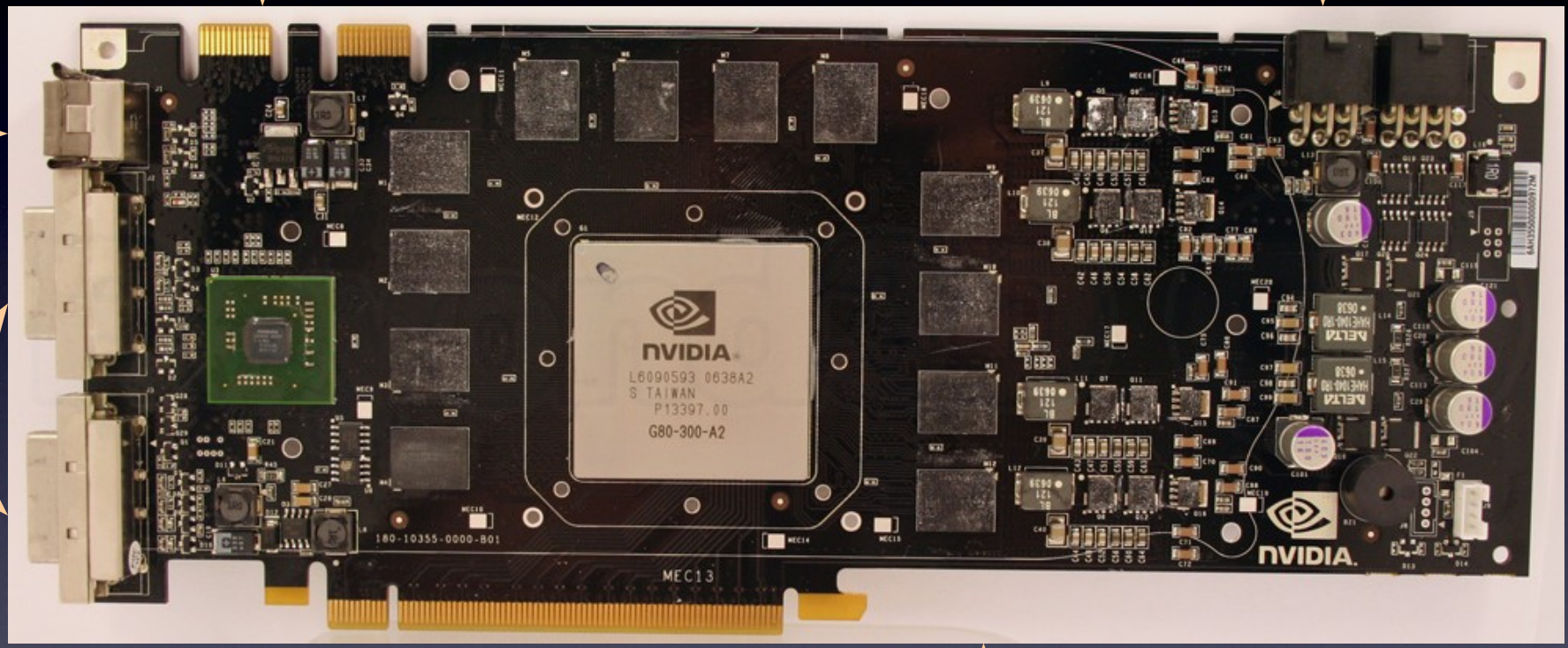
# GPU


*http://www.beyond3d.com/content/reviews/1/3*

# GPU

SLI Connector

PCIe Power Connector




HDMI

Dual DVI

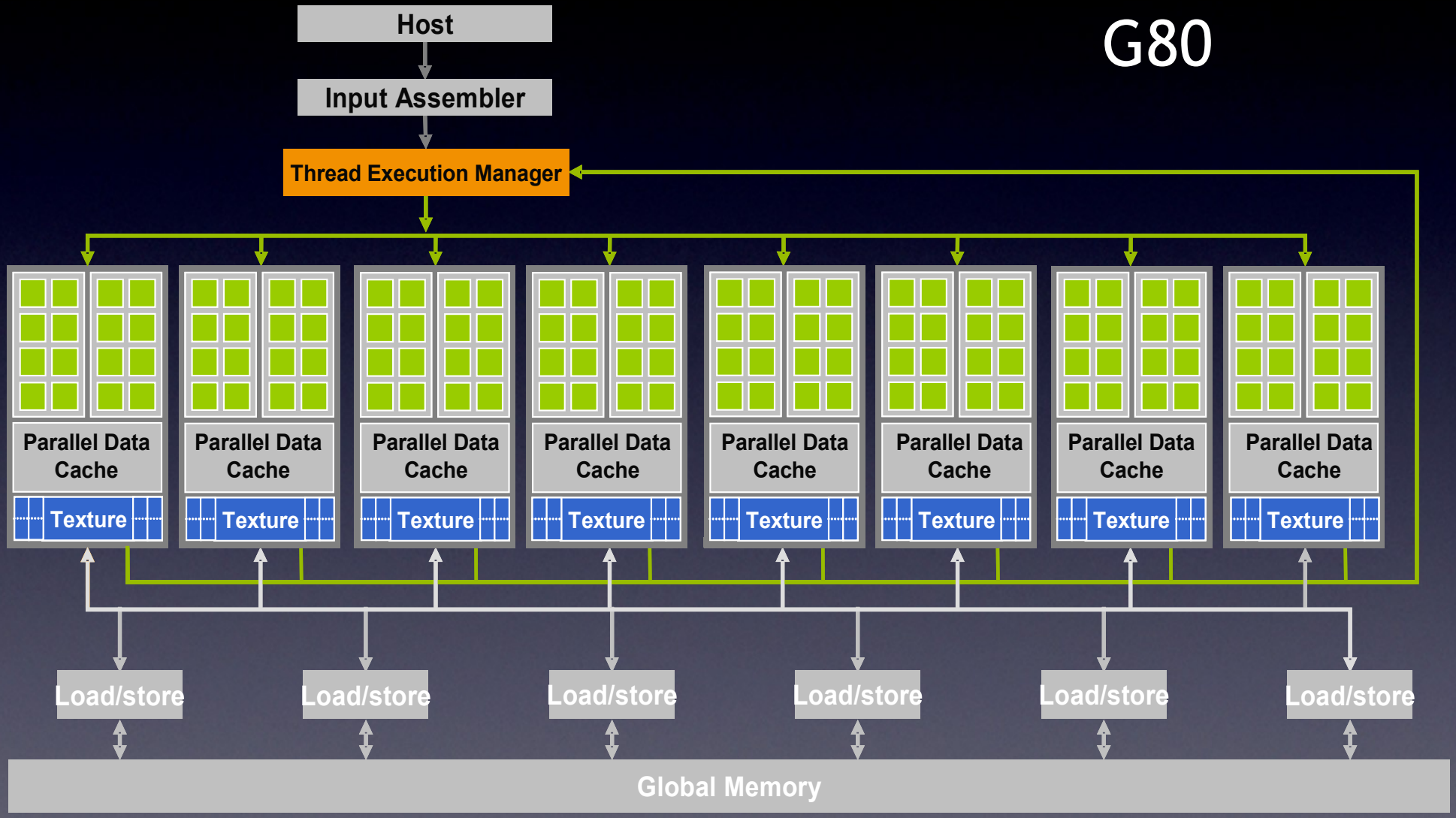
DDR3 Memory

*http://www.beyond3d.com/content/reviews/1/3*

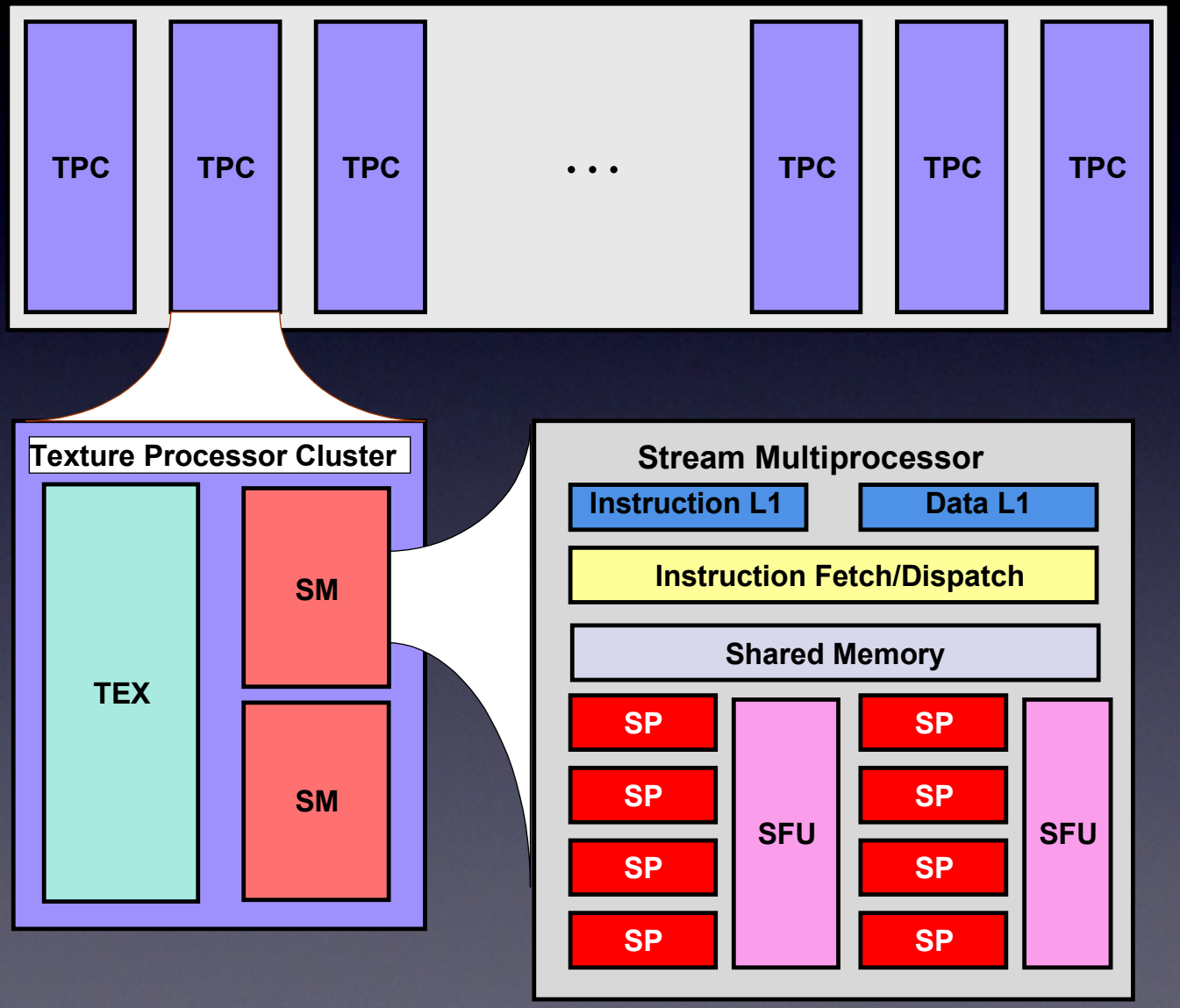
# GPU

- NVIDIA GeForce 8 Series, 9 Series
  - Duel GPU: 9800 GX2, 295
- 200 Series
  - Double precision: 285, 295
- NVIDIA Tesla

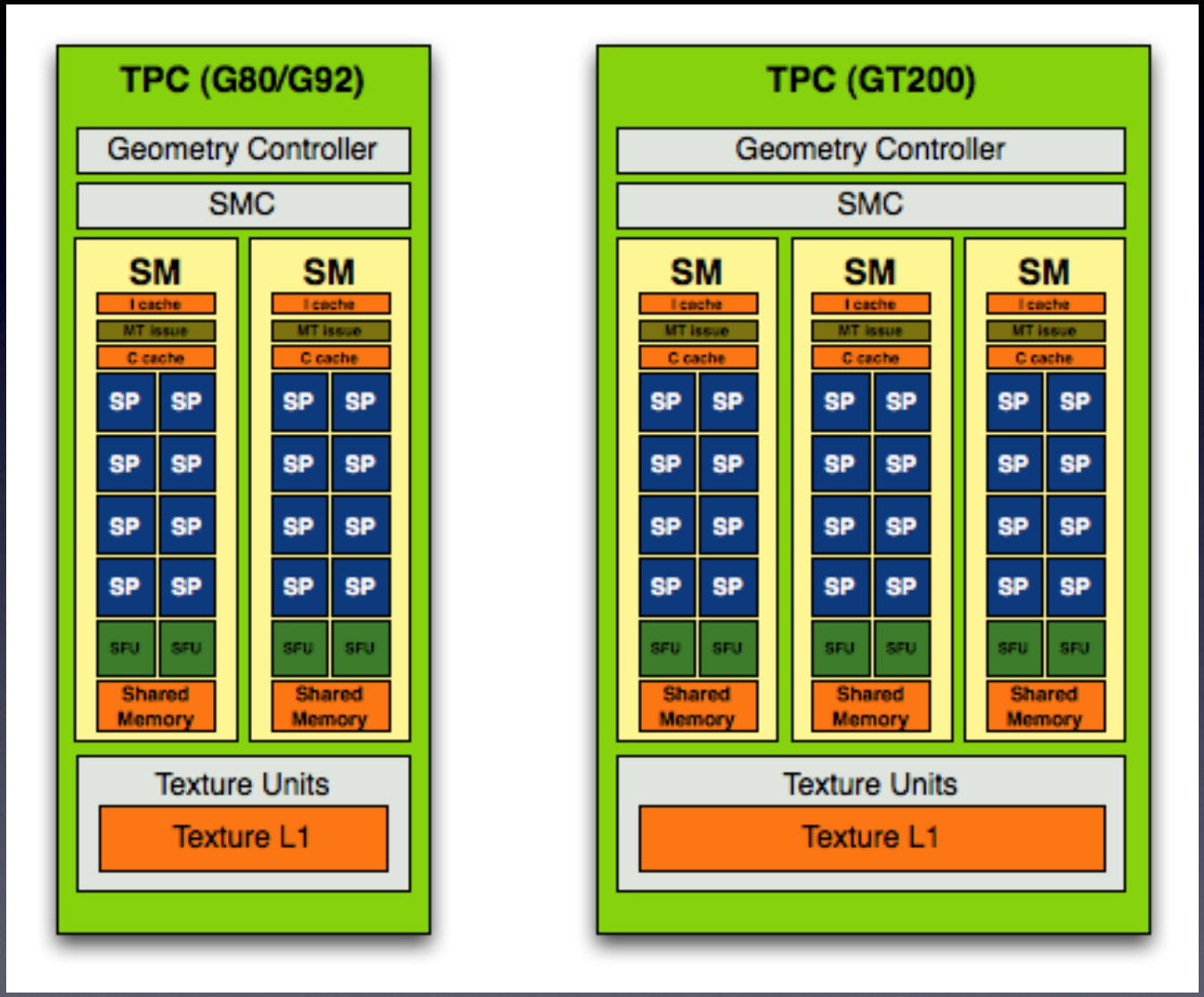
# Modern GPU Architecture



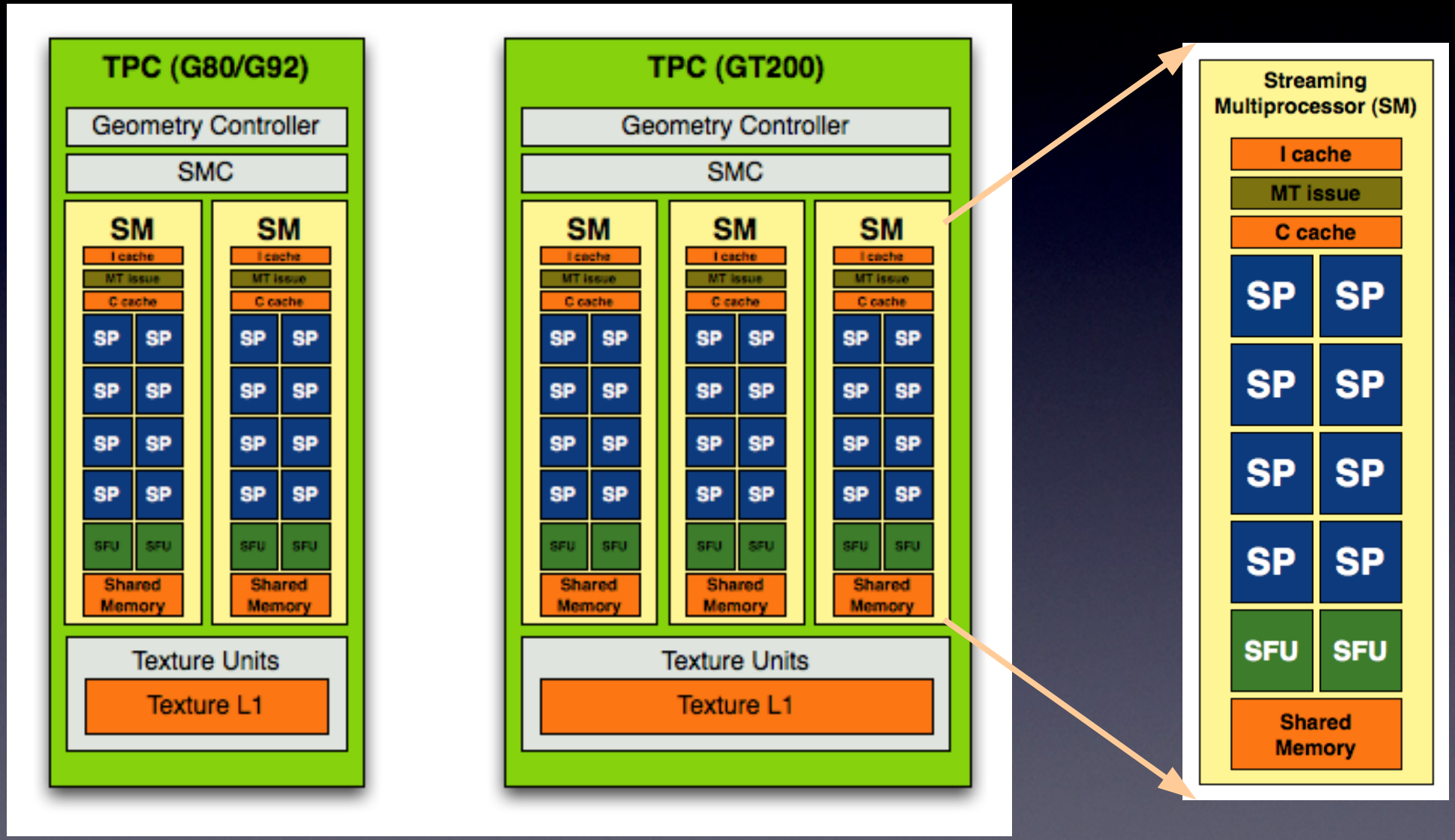
# Modern GPU Architecture



# Modern GPU Architecture



# Modern GPU Architecture

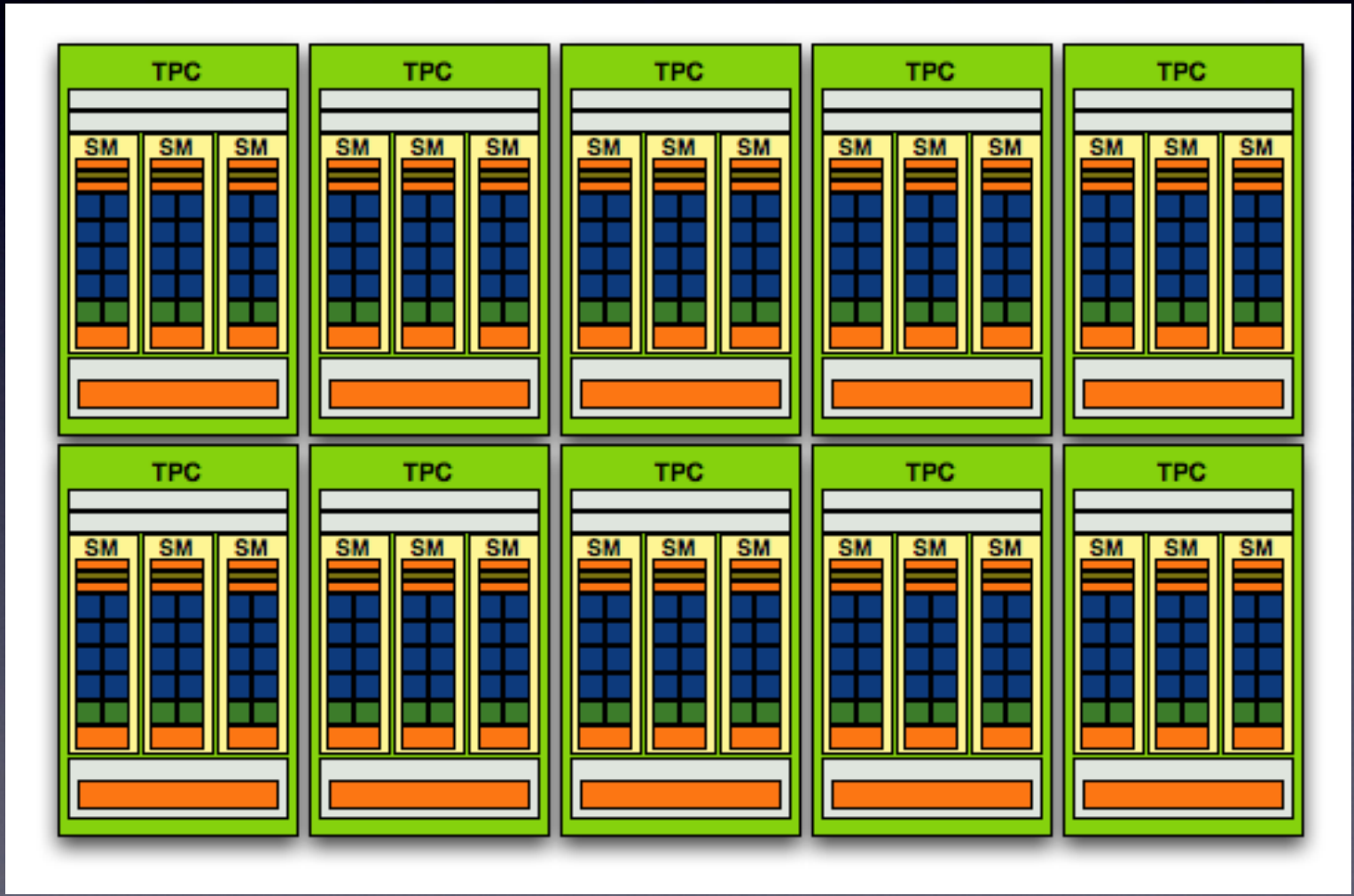


# GPU

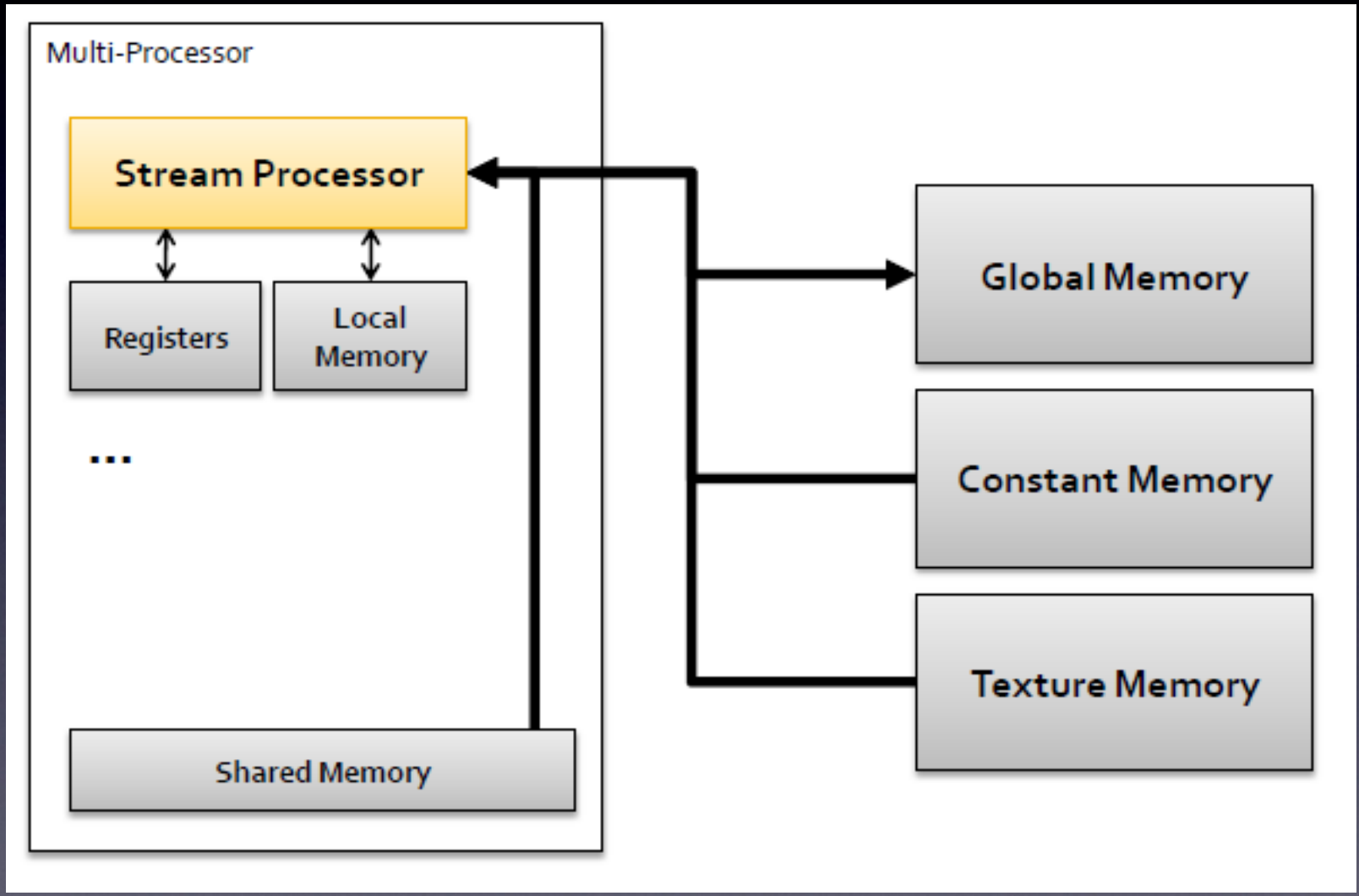
- One Stream Processor Array (SPA)…
  - Which has a collection of Texture Processor Clusters (8 in 8000 series and 10 in 200 series)
  - Each TPC has two or three Stream Multiprocessors (SM)
  - Each SM is made up of eight Scalar Processor (SP), and has its own shared memory space
  - Each SP has a multiply-add (MAD) unit, and an additional multiply (MUL) unit
  - There are also special function units (SFU) that perform FP functions such as SQRT, RCP SQRT etc.

# Modern GPU Architecture

200 Series



# Modern GPU Architecture



# Debugging with Device Emulate Mode

- An executable compiled in device emulate mode runs completely on the CPU using the CUDA runtime
  - `nvcc … -deviceemu`
  - No need for a CUDA-enabled GPU
  - Each thread emulated by a CPU thread (slow)
- Advantages
  - Debugging support
  - Access device-specific data
  - Call host functions (such as printf)

# Compiling with nvcc

- PTX: Parallel Thread Execution