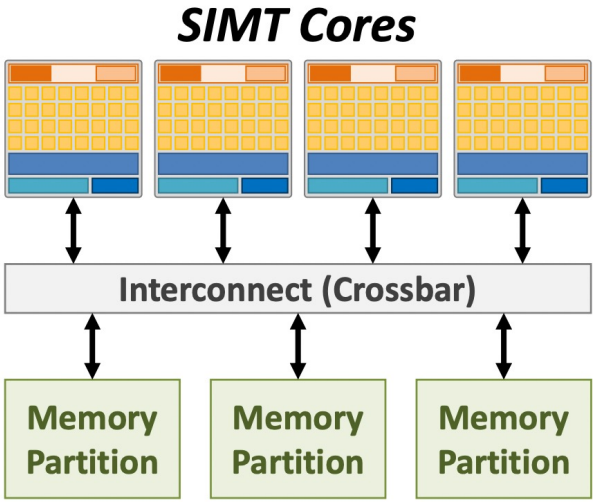
GPU Architectures
CS533, UIUC

Why GPUs?

- SIMD → difficult to program
  - Must use specialized instructions + need to think about data widths
  - Bad support for memory instructions
- SIMT = Single Instruction Multiple Threads
  - Write a scalar thread
  - Run multiple copies of the thread in a lockstep on different pieces of data
    - All threads execute the same instruction at the same time
  - Why not a manycore machine?

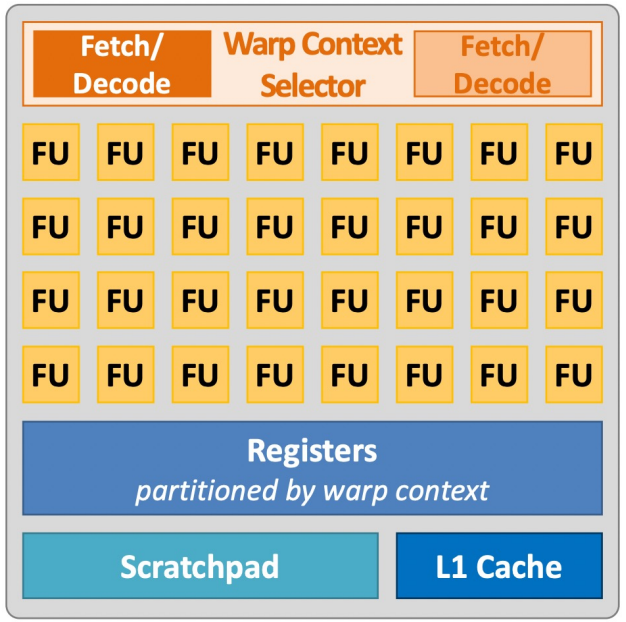
What are GPUs?

- A cluster of SIMT cores
  - SMs or shaders
  - Share a memory hierarchy
- Each SIMT core operates on one or more warps
- Originally used for graphics workloads
- Shift to general-purpose GPUs
  - Today, very popular for ML/AI workloads



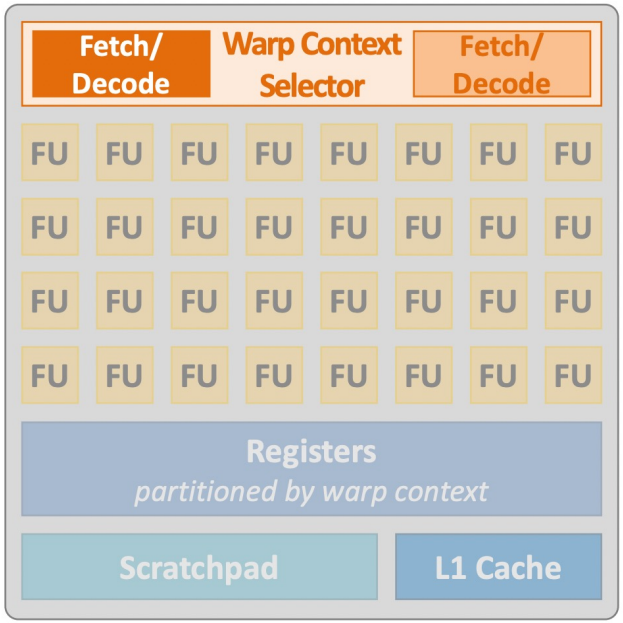
GPU Architecture: GPU Core

- A streaming multiprocessor or a shader core
- No hardware coordination across cores
- NVIDIA GTX 1080 (2016)
  - 20 SMs
  - Each SM runs 64 warps, 32 threads per warp
    - → 40960 threads
  - 1.6GHz and 180W TDP
- NVIDIA H100 SXM5 (2023)
  - 132 SMs
  - ~2GHz and **700W TDP**



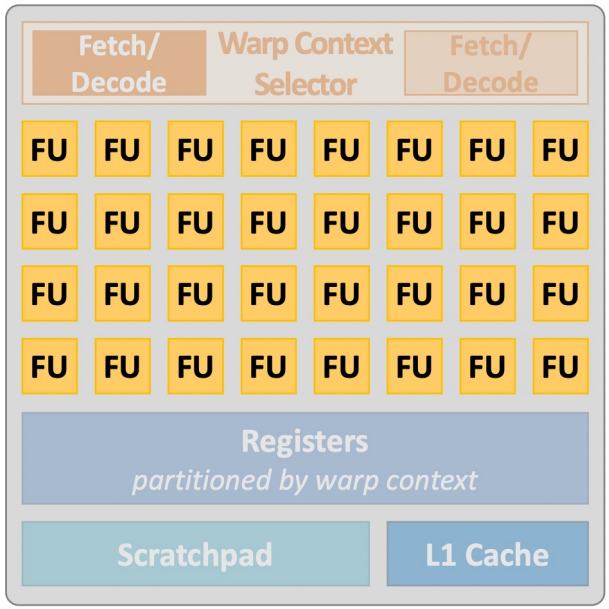
GPU Architecture: Warp Context Selector

- Chooses which warp should issue instructions to the FUs in the next cycle
- Contains a single fetch/decode logic block for all threads in a warp
- Enables fine-grained multithreading
  - Each cycle a different warp is selected to run on the pipelined FUs
- To hide latency, need to have enough warps to frequently context switch across them



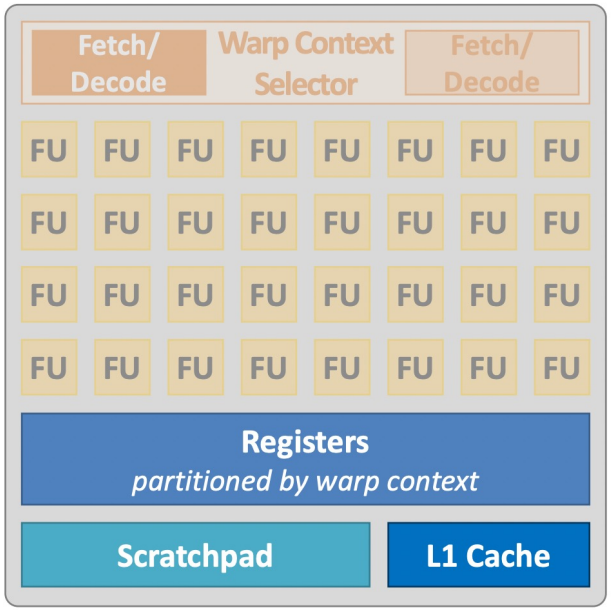
GPU Architecture: Functional Units

- Pipelined functional units
  - Can perform one MUL-ADD per cycle
  - Also includes special function units and load/store
- One FU per thread
  - Controlled by the warp context
  - All FUs for a warp operate in **lockstep**



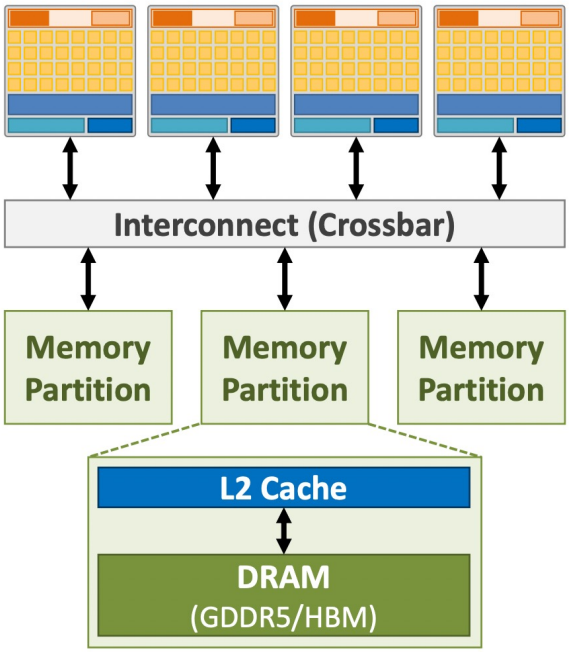
GPU Architecture: Registers and Caches

- Warp registers
  - Assigned per warp
  - Allocated by the compiler/programmer
- Scratchpad
  - Managed by software, known as shared memory
- L1 cache
  - Managed by hardware, communicates with memory
- NVIDIA GTX 1080
  - 48KB/SM L1 cache, 2MB L2 cache
- NVIDIA H100 SXM5
  - 256KB/SM L1 cache, 50MB L2 cache



GPU Architecture: Memory Hierarchy

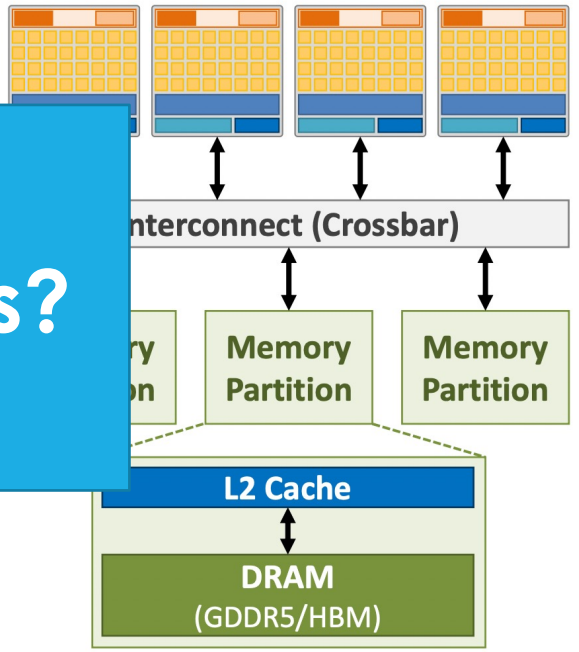
- All SMs connected to a crossbar interconnect
  - Connect one SM to one memory partition
- Memory partition = slice of L2 + DRAM
  - Shared across all SMs
  - DRAM is off-chip, logically sliced into partitions
  - New 3D-stacked DRAM (HBM)
- NVIDIA GTX 1080
  - 8GB of GDDR5X, 320.3GB/s
- NVIDIA H100 SXM5
  - 80GB of HBM3, 3.36TB/s



GPU Architecture: Memory Hierarchy

- All SMs connected to a crossbar interconnect
  - Connect one SM to one memory partition
- Memory partitio
  - Shared across
  - DRAM is off-ch
  - New 3D-stack
- NVIDIA GTX 1080
  - 8GB of GDDR5X, 320.3GB/s
- NVIDIA H100 SXM5
  - 80GB of HBM3, 3.36TB/s

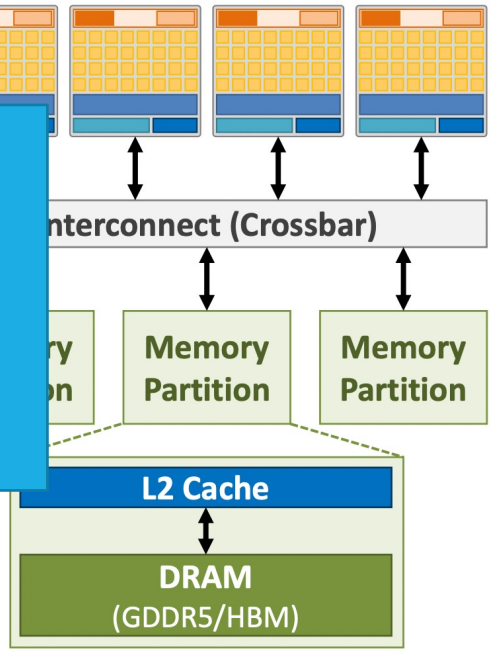
Challenges of GPUs?



GPU Architecture: Memory Hierarchy

- All SMs connected to a crossbar interconnect
  - Connect one SM to one memory partition
- Memory partitio
  - Shared across
  - DRAM is off-ch
  - New 3D-stack
- NVIDIA GTX 1080
  - 8GB of GDDR5X, 320.3GB/s
- NVIDIA H100 SXM5
  - 80GB of HBM3, 3.36TB/s

Branch divergence
Memory coalescing
Resource allocation



Next: GPUs in a Datacenter

- High power density
  - Challenge to provide enough power and cooling capacity
- High cost for datacenter owners
  - TCO = total cost of ownership
- We will discuss about TAPAS (ASPLOS'25)