

CS4803DGC Design Game Consoles

Spring 2010

Prof. Hyesoon Kim



**Georgia
Tech**



College of
Computing



Lab #2 Solution





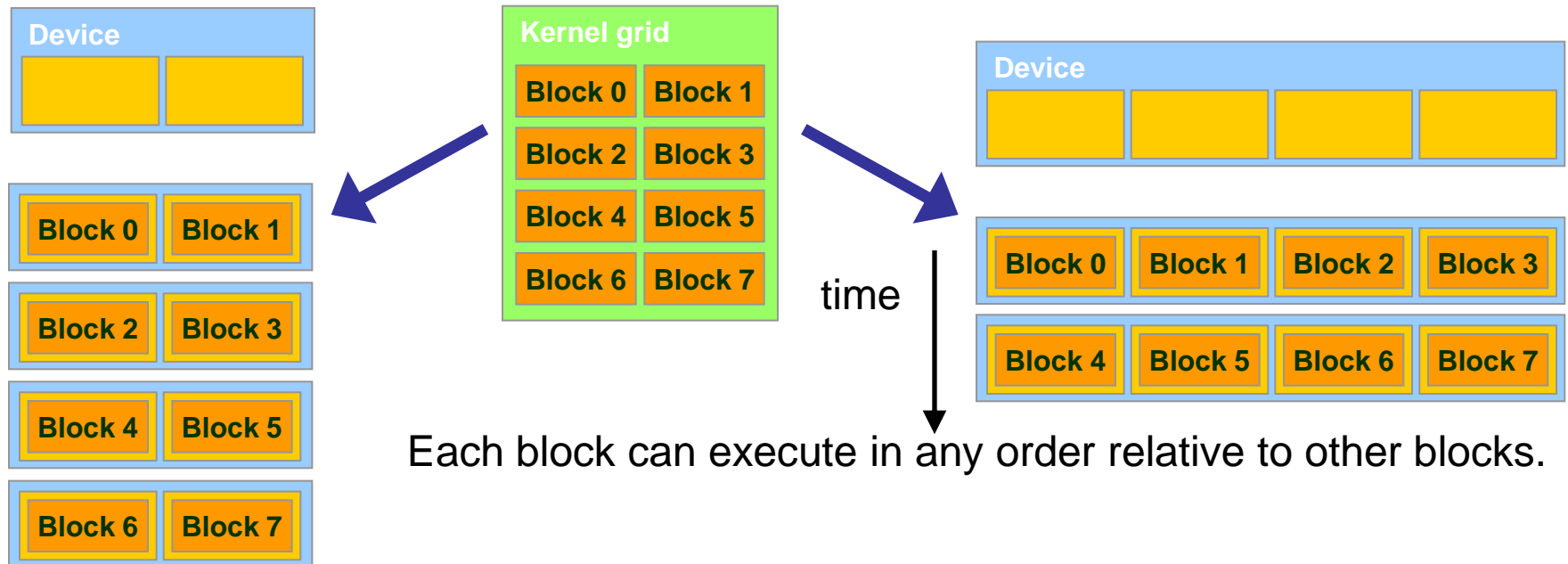
Warp in CUDA

- A group of threads that are executed together as a lock step
- Similar to SIMD instruction
- Hardware's minimum execution unit.
- G80 architecture: 32 threads
 - The number can be changed. microarchitecture feature
- All hardware unit is allocated per warp



Block Dynamic Scheduling

- Up to 8 blocks
- Each SM can take up to 768 threads (Tesla 1024 threads)
- 16KB Shared memory limit

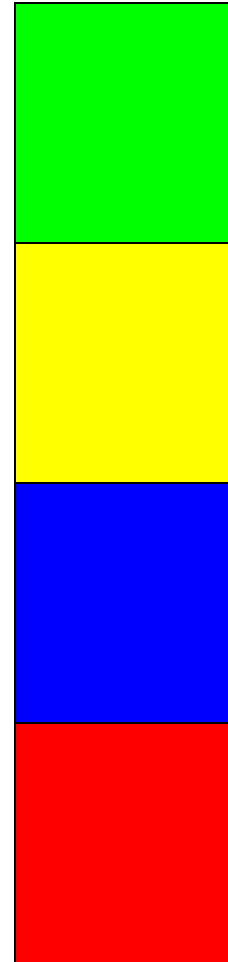




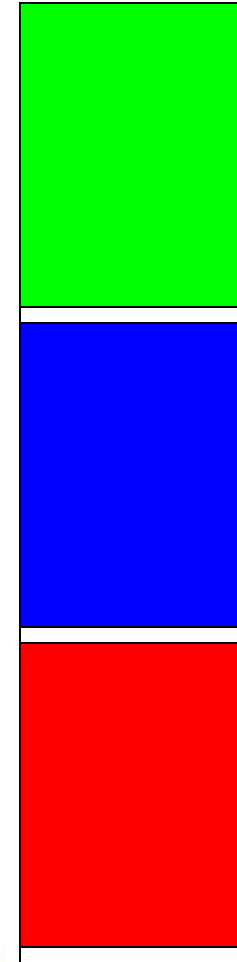
Programmer View of Register File

- There are 8192 registers in each SM in G80
 - This is an implementation decision, not part of CUDA
 - Registers are dynamically partitioned across all Blocks assigned to the SM
 - Once assigned to a Block, the register is NOT accessible by threads in other Blocks
 - Each thread in the same Block only access registers assigned to itself

4 blocks



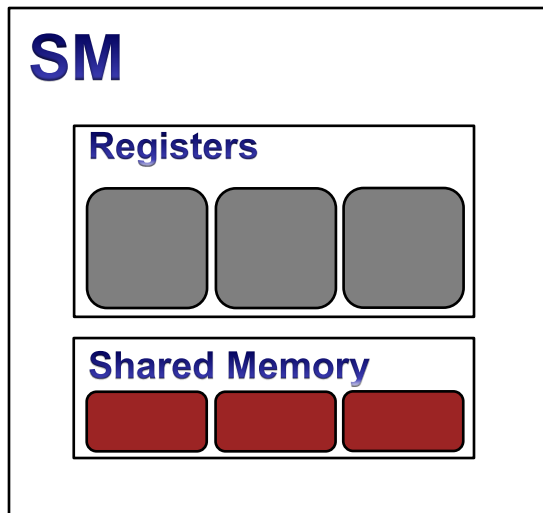
3 blocks



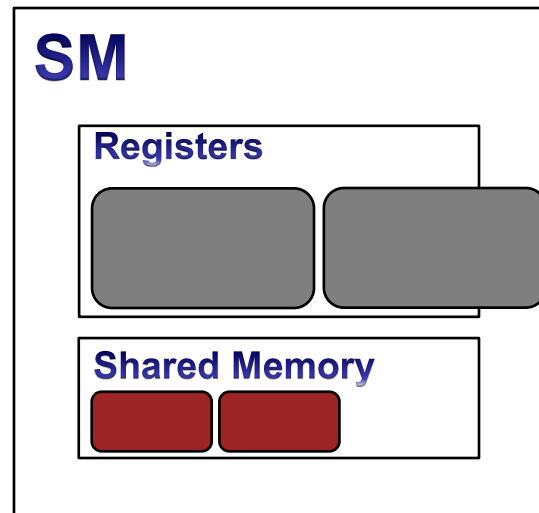


Occupancy



- Shows how many warps are assigned to the SM
- Warps are assigned at block granularity
- Programmer specifies the number of threads per block



100% Occupancy



Only one block is allocated

-  Register requirements per *block*
-  Shared memory requirements per *block*



Granularity Considerations

- For Matrix Multiplication, should I use 4X4, 8X8, 16X16 or 32X32 tiles?
 - For 4X4, we have 16 threads per block, Since each SM can take up to 768 threads, the thread capacity allows 48 blocks. However, each SM can only take up to 8 blocks, thus there will be only 128 threads in each SM!
 - There are 8 warps but each warp is only half full.
 - For 8X8, we have 64 threads per Block. Since each SM can take up to 768 threads, it could take up to 12 Blocks. However, each SM can only take up to 8 Blocks, only 512 threads will go into each SM!
 - There are 16 warps available for scheduling in each SM
 - Each warp spans four slices in the y dimension
 - For 16X16, we have 256 threads per Block. Since each SM can take up to 768 threads, it can take up to 3 Blocks and achieve full capacity unless other resource considerations overrule.
 - There are 24 warps available for scheduling in each SM
 - Each warp spans two slices in the y dimension
 - For 32X32, we have 1024 threads per Block. Not even one can fit into an SM!



CUDA Optimization Strategies

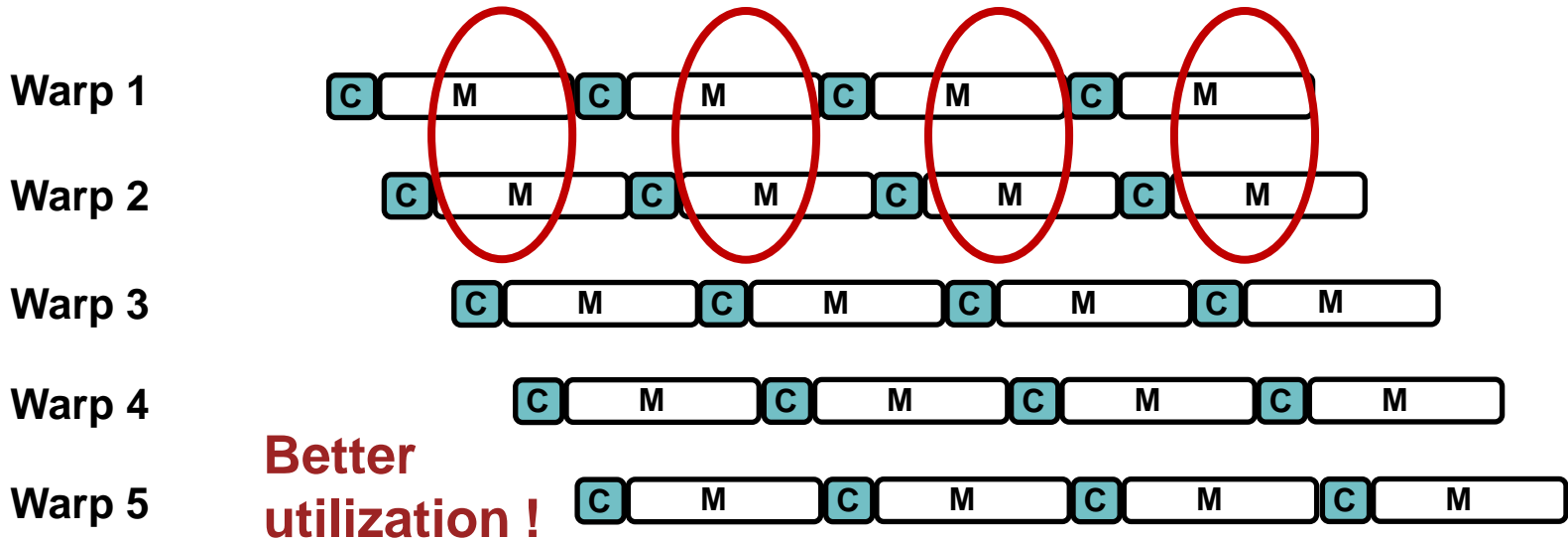
- Optimize Algorithms for the GPU
 - Reduce communications between the CPU and GPU
- Increase occupancy
- Optimize Memory Access Coherence
- Take Advantage of On-Chip Shared Memory
- Use Parallelism Efficiently



Higher Occupancy

- Better processor utilization
- Hide the memory latency

Processor is not utilized





Ann.

- First Quiz (Next Monday)
- Until Today's lecture
- CUDA, architecture, Performance calculation, Xbox 360