ECE 571 – Advanced Microprocessor-Based Design Lecture 25

Vince Weaver

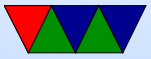
https://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

1 December 2022

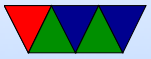
Announcements

- Don't forget projects, presentations next week (Tues and Thurs)
  Will send out prospective schedule
- Final writeup due last day of exams (16th)
- Will try to get homeworks graded soon.
- Reminder: no final



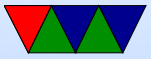
Brief Review of GPU concepts

- CPU better for single-thread, low-latency workloads
- GPU better at massively parallel high-bandwidth workloads
- CPUs becoming more GPU like as GPUs becoming more CPU like



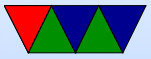
How does Real Hardware

- Biggest impediment was memory size/bandwidth
- 3D Polygon Rendering/Rasterizing – vertices together make polygons, all s shapes broken up into smaller amount of polygons. Textures applied, lighting calculated based on normal to polygon.
- Scanline Rendering – no framebuffer, triangles processed and what line they start/stop are stored. One line drawn at a time
- Tiled rendering, similar, but tiles rather than lines



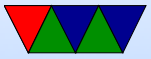
as small as 16x16 or 32x32 (used on various boards, including Videocore IV on Pi)

- 3D Polygon rendering has problems. Lots of things, such as shadows, reflections, and transparency, are effectively faked with hacks.



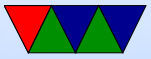
Ray Casting vs Ray Tracing

- Complex, not really good definitions
- Ray casting in general is when you cast rays from eye into scene, but stop when hit first object.
- Ray tracing cast rays from eye into scene, but reflect/refract off of objects until hit light source



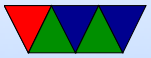
Ray Tracing Hardware

- Anything other than simple ray-casting requires recursion (Each time you hit an object) as well as random-access to the entire 3d-space
- NVIDIA RTX
- Hybrid raytracing – traditional casting+rasterization used for visibility, raytracing for shadows
- Only continue tracing rays of surface has more than threshold of reflectability

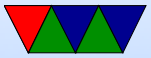


NVIDIA GPUs

Tesla	2006	90-40nm
Fermi	2010	40nm/28nm
Kepler	2012	28nm
Maxwell	2014	28nm
Pascal/Volta	2016	16nm/14nm
Turing	2018	12nm
Ampere	2020	8nm/7nm
AdaLovelace/Hopper	2022	TSMC 4N / N4 (5nm/4nm)

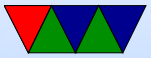


- GeForce – Gaming
- Quadro (old) / Workstation
- Tesla (old) / DataCenter

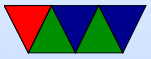


Gaming GPUs?

- Geforce 40
- RTX 4090 16,384 FP32 cores (82.6 TFLOPS)
  24G GDDR6X memory, 72M L2 cache, 450W
- RTX 4080
- CUDA Compute 8.9
- TSMC 4N (custom NVIDIA 5nm, easily confused with TSMC N4 which is different)
- Fourth-gen Tensor cores with FP8, FP16, bloat-16 sparsity acceleration? (sparse arrays? lots of zeros?)

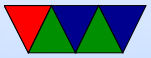


- Third-gen Raytracing
  Opacity Micromap Engine, Displaced Micro-mesh Engine
- Shader Execution Reordering
  When following ray, might end up diverging in behavior and not as vectorable. Allow to re-order so more similar?
- Optical Flow Acceleration (use AI to guess between-frame animations)
- FP64 performance 1/64 that of F32
- Controversy over 4080 naming
- 12VHPWR connector failure (PCIe gen 5 16-pin connector) melting



Note on Modern GPUs

- Apple M1 has whole microcontroller



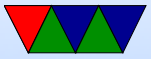
Homework Reading #1

NVIDIA Hopper GPU Architecture and H100 Accelerator Announced: Working Smarter and Harder

https:

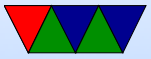
//www.anandtech.com/print/17327/nvidia-hopper-gpu-architecture-and-h100-accelerator-announced

March 2022 – by Ryan Smith



Background

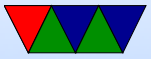
- Hopper
- H100
- 16896 Cores
- 4.8Gbps HBM3 RAM
- 80GB VRAM – 6 16GB stacks with 1 disabled? 3TB/s
- FP32 – 60 TFLOPS
- FP64 – 30 TFLOPS
- INT8 Tensor – 2000 TFLOPS (2 Petaflops?)
- NVLink4 – 900GB/s



NVLink switch 256 GPUs connected together, supplemental to Infiniband

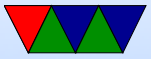
- 814mm^2
- 80 Billion transistors
- 700W
- TSMC 4N
- SXM5 special socket, high power, NVLink, faster than PCIe

  vs PCIe, PCIe5 support
- Transformers, for machine learning, FP16/FP8
- DPX dynamic programming – eliminate redundant



workload?

- Confidential Computing – cloud computing environments, virtualized GPU, encryption/decryption of date entering GPU

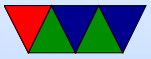


Homework Reading #2

NVIDIA Hopper Architecture In-Depth

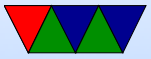
https://developer.nvidia.com/blog/nvidia-hopper-architecture-in-depth/

March 2022 – by Andersch, Palmer, Krashinksky, Stam, Mehta, Brito and Ramaswamy



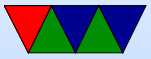
Background

- H100
- 9th gen datacenter GPU
- Order of magnitude better than A100
- Grace Hopper Superchip? CPU+GPU? ARM chip connected to GPU with NVLINK?
- New Streaming Multiprocessor (SM)
  - Fourth-gen tensor cores
    Faster matrix math
  - New DPX instructions



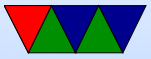
- 3x faster IEEE FP64 and FP32
- New thread block cluster
  Now goes threas, thread blocks, thread block clusters, grids
- Distributed shared memory
- Asynchronous execution?

- Transformer Engine
- HBM3 memory
- 50MB L2
- MIG (Multi-instance GPU), confidentially
  NVDEC (video decoder) and NVJPG (hardware JPEG



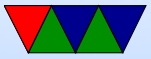
decoder)

- 4th-gen NVLINK
- Third-gen NVSWITCH
- PCIe Gen5



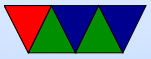
Background

- First asynchronous GPU?
  Not really asynchronous, just means threads can run out of order with each other
- Tensor memory accelerator, only few CUDA threads needed for memory, reset can dedicate to compute



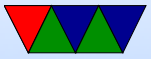
FP8 Data Format

- Two new FP8 types
  E4M3 (4 exponent, 3 mantissa, 1 sign)
  E5M2 (5 exponent, 2 mantissa, 1 sign)



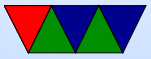
DPX Instructions

- Brute force algorithms, a partial solution might be reused a lot
  Smith-Waterman DP in genomics
  Floyd-Warshall in robotics
- Exponential to Polynomic time
- How?



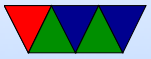
Distributed Shared Memory

- Can directly write memory in other parts of thread block w/o having to go through main memory



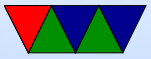
Tensor Memory Allocator

- Transfer large amounts of data and multi-dimensional tensors from global to shared memory
- Asynchronous
- Single thread in warp can issue one



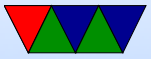
Async Execution

- Asynch memory barriers, if a thread arrives early it can move on to more work while waiting for the others



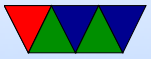
Transformer Engine

- BERT and GPT-3 are transformer models
- Megatron Turing NLG requires 2048 NVIDIA A100 GPUs for 8 weeks to train



Intel Xe-HPG / Arc

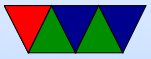
- Each render slice contains 4Xe cores, 4 tracing units
- DirectX12 Ultimate compliant
- Ray Tracing, Variable Rate Shading, Mesh Shading
- 16 256-bit XVE vector engines (rasterization)
- 16 1024-bit XMX matrix engines (machine learning)
  4-deep systolic array can calc 256 multiply-accumulate ops per clock for INT8 inferencing (?)
- Thread sorting unit for Ray-tracing
- Concurrent INT/FP32 pipelines (NVidia introduced in



Turing)

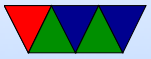
- Media engine, 8K HDR encode/decode, HEVC, VP9, AV1

  AV1 decoding 50x faster than software
- Clock speed comparisons across GPUs difficult



AMD GPUs

Caribbean Islands			Fiji
Sea Islands			
Volcanic Islands			
Polaris RX400	2016	28/14nm	
Polaris / RX500	2017	14/12nm	
Vega	2017	14/7nm	GCN5
Navi / RX5000	2019	7nm	RDNA
Navi2x / RX6000	2020	7nm	RDNA2
	2022	N5	RDNA3



RDNA3

- not supposed to be launched until 13 December 2022 (?)
  Some things announced on November 2nd maybe to steal NVIDIA thunder?
- RX7900
- Use chiplet design
  Can use 5nm for cores and 6nm for memory
- Perf for watt
- GDDR6 vs GDDR6X because uses less power