NVIDIA Advanced Features

Lecture 18
May 1, 2025

To Dos

Program #7 results on Tuesday

Reading for next time

Final project assigned

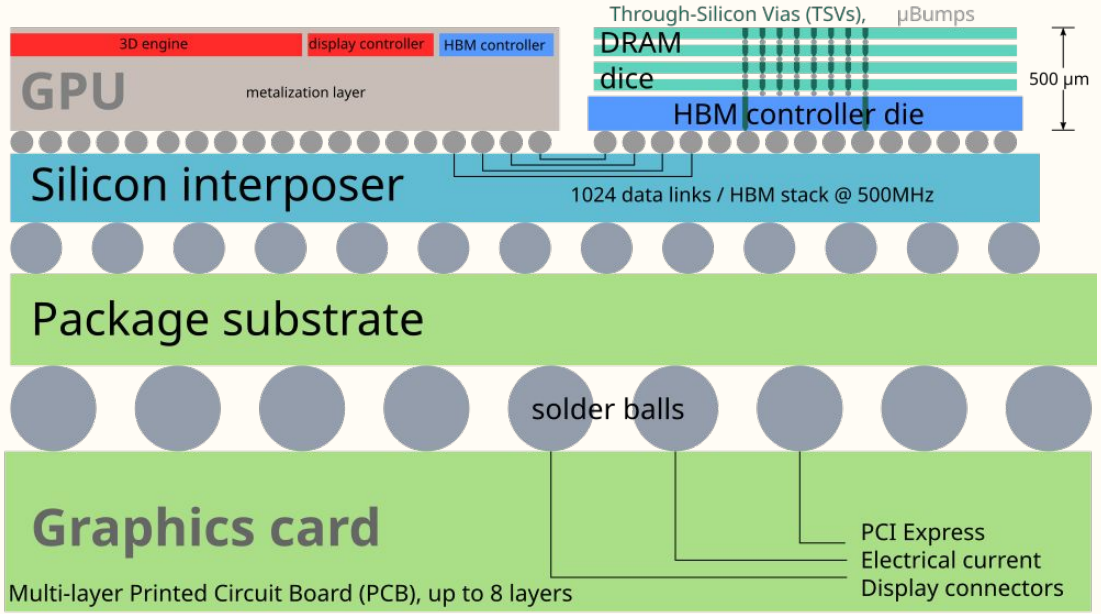
Tensor Cores

- Designed to accelerate matrix operations
- Mixed-precision multiply accumulate operations
  - e.g., Multiply two 4×4 FP16 matrices and then add FP16/FP32 matrix to result
- Trade-off performance and accuracy

	Blackwell	Hopper
Supported Tensor Core precisions	FP64, TF32, BF16, FP16, FP8, INT8, FP6, FP4	FP64, TF32, BF16, FP16, FP8, INT8
Supported CUDA® Core precisions	FP64, FP32, FP16, BF16	FP64, FP32, FP16, BF16, INT8

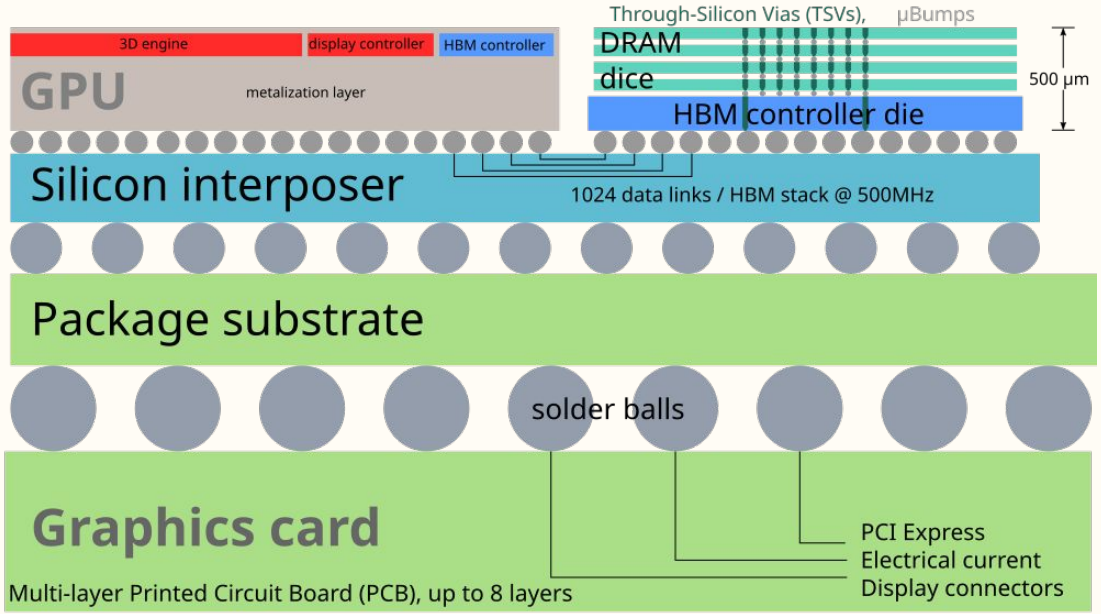
https://www.nvidia.com/en-us/data-center/tensor-cores/

High Bandwidth Memory (HBM and HBM3)



- Optimized for fast data transfer and reduced power consumption
- 3D-stacked synchronous dynamic random access memory (SDRAM)
- Physical organization moves memory closer to compute chip, reducing signal transmission distance and latency
- Wide bus width through multiple independent channels

https://en.wikipedia.org/wiki/High_Bandwidth_Memory#:~:text=In%20August%202023%2C%20Nvidia%20announced,of%20GPUs%20using%20HBM3E%20memory



- 4-7 Stacks per GPU
- Stack <= 8 DRAM dies
- Stack connected to memory controller on chip through substrate (e.g., silicon interposer) or stack directly connected to chip
- Within DRAM die stack, dies are vertically interconnected by through-silicon vias (TSVs) and microbumps (used to create electrical connections)
  - TSV= thin electrical wires run through holes in silicon chips to connect multiple layers to base logic chip

Memory Bandwidth

- Each stack has multiple independent channels
- 1024 bit wide bus
- Each generation sends at higher rate (e.g, DDR)

HBM2

- Up to 8 128 bit channels / stack

HBM3

- Up to 16 64 bit channels / stack

	HBM	GDDR	DDR5
Latest standard	HBM3	GDDR6X	DDR5
Architecture	3D stacked	Planar	Planar
Bus width	1,024-bit	32-bit per chip	64-bit
Bandwidth per stack/chip	819 GBps	Up to 84 GBps	Up to 51.2 GBps
Power efficiency	Highest	Moderate	Lowest of the three
Form factor	Most compact	Moderate	Largest
Integration	On-package with GPU/CPU	Soldered on PCB	DIMM modules
Cost	Highest	Moderate	Lowest
Primary applications	High-end GPUs, AI accelerators	Graphics cards, some AI inference	General computing, servers

https://www.techtarget.com/whatis/definition/high-bandwidth-memory#:~:text=High%2Dbandwidth%20memory%20is%20a,require%20optimized%20high%2Dspeed%20memory.

NVIDIA H200 Tensor Core GPU

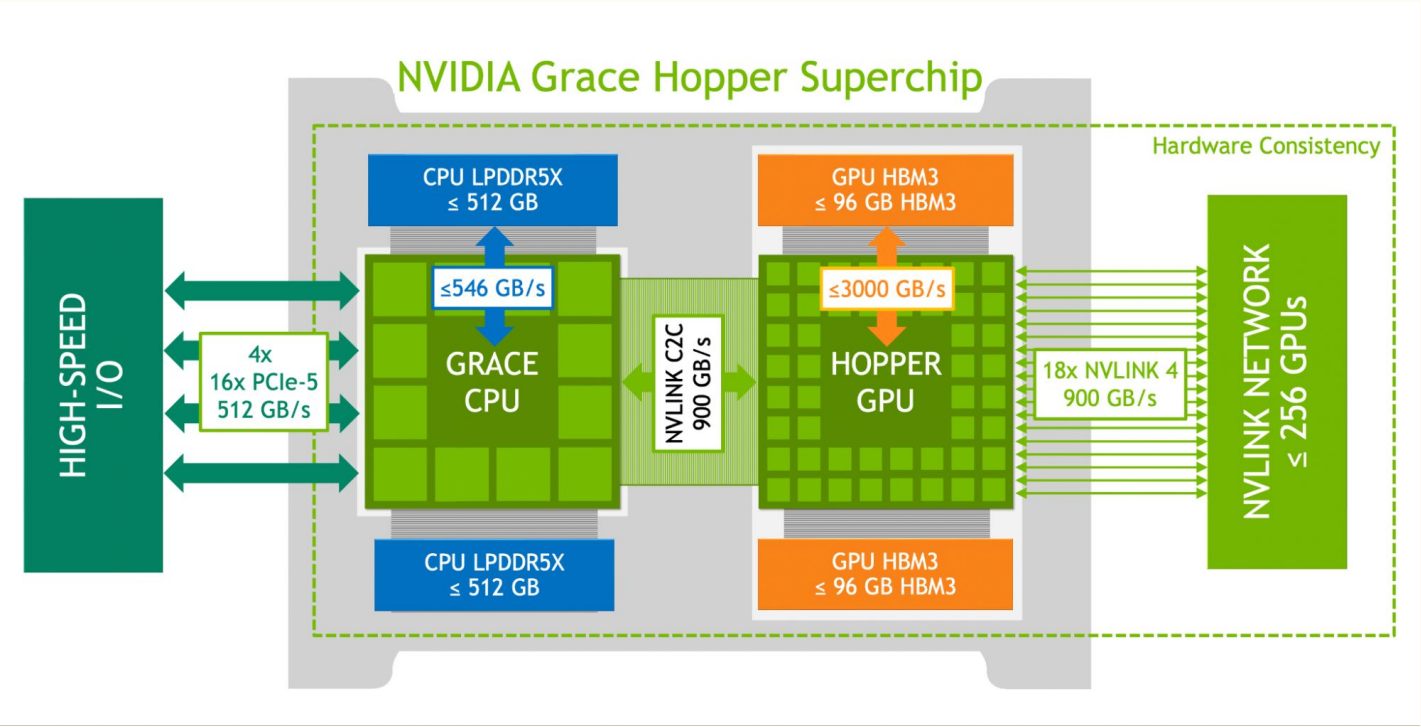
https://www.nvidia.com/en-us/data-center/h200/

TFLOP:
1 trillion
FLOP/s

	H200 SXM[1]	H200 NVL[1]
FP64	34 TFLOPS	30 TFLOPS
FP64 Tensor Core	67 TFLOPS	60 TFLOPS
FP32	67 TFLOPS	60 TFLOPS
TF32 Tensor Core[2]	989 TFLOPS	835 TFLOPS
BFLOAT16 Tensor Core[2]	1,979 TFLOPS	1,671 TFLOPS
FP16 Tensor Core[2]	1,979 TFLOPS	1,671 TFLOPS
FP8 Tensor Core[2]	3,958 TFLOPS	3,341 TFLOPS
INT8 Tensor Core[2]	3,958 TFLOPS	3,341 TFLOPS
GPU Memory	141GB	141GB
GPU Memory Bandwidth	4.8TB/s	4.8TB/s

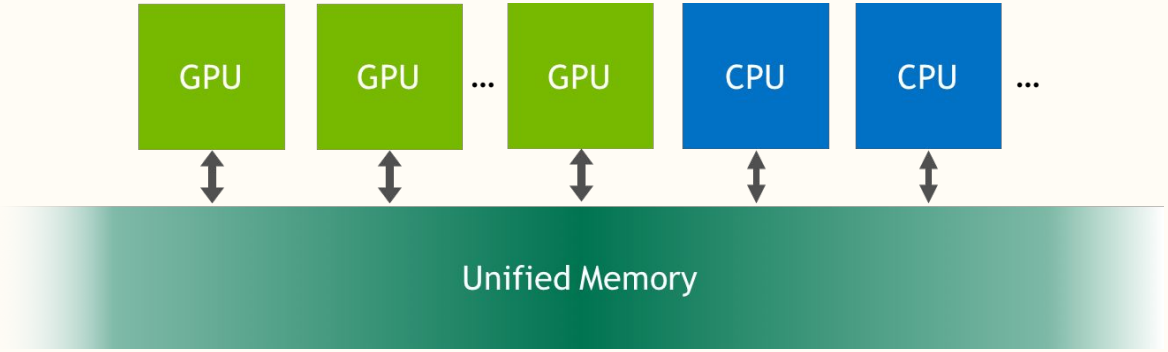
The NVIDIA H200 GPU contains 16,896 CUDA cores for handling parallel computations efficiently. It also has 528 fourth-generation Tensor Cores that support 8-bit, 16-bit, 32-bit, and 64-bit floating point operations.

https://www.trgdatacenters.com/resource/nvidia-h200/#:~:text=How%20many%20cores%20are%20in,64%2Dbit%20floating%20point%20operations.



- CPU and GPU threads can access CPU and GPU memory directly
- CPU:
  - <= 72 Arm cores (including SIMD units), 117MB L3 $, 512 GB memory
- GPU:
  - <= 144 SMs with tensor cores, <= 96GB HBM3, 60 MB L2 $

Unified Memory



- Single memory address space accessible from any processor
- Processor either accesses data in remote memory or system migrates data between memory based on usage
- Allows oversubscription (i.e., using more memory than available on GPU)
- Use cudaMallocManaged() for allocation instead of malloc() or new()

https://developer.nvidia.com/blog/unified-memory-cuda-beginners/

L0 $ accessed directly by functional units

Figure 4. GH100 streaming multiprocessor

Reconfiguration between L1 and shared memory on per-SM basis

https://developer.nvidia.com/blog/nvidia-hopper-architecture-in-depth/#:~:text=H100%20HBM%20and%20L2%20cache%20memory%20architectures,GPU%20further%20increased%20HBM2%20performance%20and%20capacity.

Turning off and bypassing the cache

"Caching can be controlled on a per instruction basis using inline PTX. The L1 cache can also be disabled using the compiler option -dlcm."

https://forums.developer.nvidia.com/t/switch-off-l1-cache/37274/2

NVIDIA Collective Communications Library (NCCL)

- Multi-GPU and multi-node communication
- Collective communication:
  - all-gather, all-reduce, broadcast, reduce, reduce-scatter
- Point-to-point
  - Send and receive