UC San Diego

CSE 291: Operating Systems in Datacenters

Amy Ousterhout

Nov. 3, 2022

UC San Diego

Agenda for Today

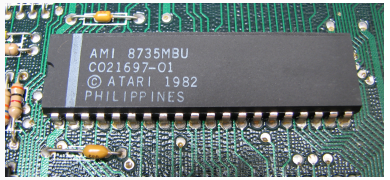
- GPUs overview
- PTask discussion

UC San Diego

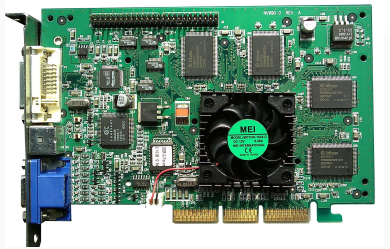
GPUs

History of GPUs

- Originally designed to create images to display
  - 1970s: video processors for arcade games
  - 1980s: graphics processors for PCs
  - 1990s: 3D graphics
    - 1999: “the world’s first GPU”
  - 2000s: more programmability
- Applied to general purpose compute tasks
  - GPGPUs
  - Linear algebra (2003)
  - Scientific computing
  - Mining bitcoin (today)



Atari ANTIC microprocessor

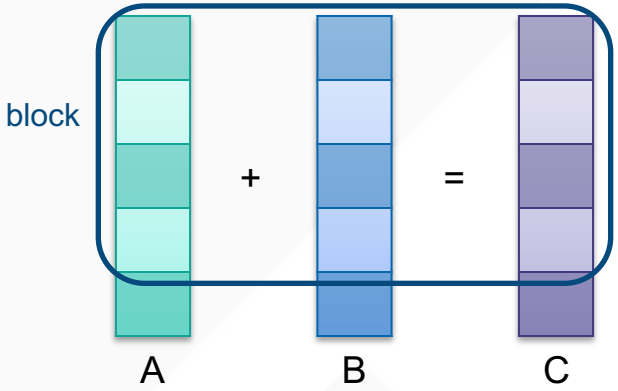


Nvidia GeForce 256

https://commons.wikimedia.org/wiki/File:ANTIC_chip_on_an_Atari_130XE_motherboard.jpg
https://commons.wikimedia.org/wiki/File:RTX_3090_Founders_Edition!.jpg

Data Parallelism

- GPUs are designed for data-parallel tasks
- Example: add two arrays/vectors



Sequential (e.g., on a CPU):

```
void sequential_add(int n, float *A, float *B, float *C) {
    for (int i = 0; i < n; i++)
        C[i] = A[i] + B[i];
}
```

Parallel (e.g., on a GPU):

```
void parallel_add(int n, float *A, float *B, float *C) {
    int i = current_block * block_size + thread_index;
    if (i < n)
        C[i] = A[i] + B[i];
}
```

Systems Research on GPUs

- How should we program GPUS?
  - CUDA, OpenCL, etc.
- How can we process packets on GPUs?
  - PacketShader, SSLShader
- How can we schedule and manage memory on GPUs?
  - TimeGraph, PTask ⟵ today
- How can we share GPUs across multiple apps?
- How can use use GPUs to accelerate ML workloads?

UC San Diego

PTask Discussion