Announcements

• Core Wars will be out in the next couple of days
  – Aim at having fun!
  – Number of points allocated to it is small (10-20% of other assignments)
  – 5 points for turning something in, 1 point more for going up the ladder

• Pizza party on last day of class
  – Showdown
  – Friday Nov 30th

• Final project (distributed ray tracer) out last week
  – Demoed: Dec 13 2-4:30
Memory Hierarchy

<table>
<thead>
<tr>
<th></th>
<th>16 KB</th>
<th>512 KB</th>
<th>2 GB</th>
<th>300 GB</th>
<th>1 TB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>registers/L1</td>
<td>L2</td>
<td>DRAM</td>
<td>Disk</td>
<td>Tape</td>
</tr>
<tr>
<td>Access Time</td>
<td>2 ns, random access</td>
<td>5 ns, random access</td>
<td>20-80 ns, random access</td>
<td>2-8 ms, random access</td>
<td>100s, sequential access</td>
</tr>
<tr>
<td>Capacity</td>
<td>16 KB</td>
<td>512 KB</td>
<td>2 GB</td>
<td>300 GB</td>
<td>1 TB</td>
</tr>
</tbody>
</table>

Kavita Bala, Computer Science, Cornell University

Tapes

- Same basic principle for 8-tracks cassettes, VHS, atari tape drive, tape storage

Kavita Bala, Computer Science, Cornell University
Disks & CDs

- Disks use same magnetic medium as tapes
  - concentric rings (not a spiral)

- CDs & DVDs use optics, and a single spiral track
  - Non-volatile

Disk Physics

Typical parameters:
- 1 spindle
- 1 arm assembly
- 1-4 platters
- 1-2 sides/platter
- 1 head per side
  (but only 1 active head at a time)
- 4,200 – 15,000 RPM
Disk Accesses

Accessing a disk requires:
- specify sector: C (cylinder), H (head), and S (sector)
- specify size: number of sectors to read or write
- specify memory address: bus address to DMA to

Performance:
- seek time: move the arm assembly to track
- Rotational delay: wait for sector to come around
- transfer time: get the bits off the disk

Example

- Average time to read/write 512-byte sector
  - Disk rotation at 10,000 RPM
  - Seek time: 6ms
  - Transfer rate: 50 MB/sec
  - Controller overhead: 0.2 ms

- Average time:
  - Seek time + rotational delay + transfer time + controller overhead
  - 6ms + 0.5 rotation/(10,000 RPM) + 0.5KB/(50 MB/sec) + 0.2ms
  - 6.0 + 3.0 + 0.01 + 0.2 = 9.2ms
Disk Scheduling

- Goal: minimize seek time
  - secondary goal: minimize rotational latency
- FCFS (First come first served)
- Shortest seek time
- SCAN/Elevator
  - First service all requests in one direction
  - Then reverse and serve in opposite direction
- Circular SCAN
  - Go off the edge and come to the beginning and start all over again

What we didn’t talk about

- RAID
  - Redundancy for fault tolerance
  - Speed
- Solid State drives
  - Very expensive still
GPUs
NVidia G80 Architecture

Kavita Bala, Computer Science, Cornell University
Traditional Graphics Pipeline

Application
Transform & Lighting
Projection & Clipping
Triangle Setup
Rasterization
Display

Projection & Clipping

View frustum
Eye view

Application
Transform & Lighting
Projection & Clipping
Triangle Setup
Rasterization
Display

Kavita Bala, Computer Science, Cornell University
The dark ages (early-mid 1990’s), when there were only frame buffers for normal PC’s.

Some accelerators were no more than a simple chip that sped up linear interpolation along a single span, so increasing fill rate.

This is where pipelines start for PC commodity graphics, prior to Fall of 1999.

This part of the pipeline reaches the consumer level with the introduction of the NVIDIA GeForce256.

Hardware today is moving traditional application processing (surface generation, occlusion culling) into the graphics accelerator.
Moore’s Law

• 1965
  – number of transistors that can be integrated on a die would double every 18 to 24 months (i.e., grow exponentially with time).

• Amazingly visionary
  – 2300 transistors, 1 MHz clock (Intel 4004) – 1971
  – 16 Million transistors (Ultra Sparc III) – 1998
  – 42 Million transistors, 2 GHz clock (Intel Xeon) – 2001
  – 55 Million transistors, 3 GHz, 130nm technology, 250mm² die (Intel Pentium 4) – 2004
  – 290+ Million transistors, 3 GHz (Intel Core 2 Duo) – 2007

Processor Performance Increase

![Processor Performance Increase Diagram]

Kavita Bala, Computer Science, Cornell University
Faster than Moore’s Law

Why are GPUs so fast?

- Pipelined and parallel
- Very, very deep pipeline: 800-1000 deep
GPU Parallelism

G70 Hardware Architecture
Parallelism

• Critical to achieving performance
• Flynn’s taxonomy
  – SISD (Single instruction, single data)
    • Boring CPU
  – MISD (Multiple instruction, single data)
    • Redundant processing
  – SIMD (Single instruction, multiple data)
    • GPUs
  – MIMD (Multiple instruction, multiple data)
    • Multicore, Cell processor

Parallelism

• *Must* exploit parallelism for performance
  – Lot of parallelism in graphics applications
• SIMD: single instruction, multiple data
  – Perform same operation in parallel on many data items
  – Data parallelism
• MIMD: multiple instruction, multiple data
  – Run separate programs in parallel (on different data)
  – Task parallelism
Performance Tuning

1. Identify bottleneck (slowest stage)
2. Improve performance of slowest stage
   Repeat as necessary

Amdahl’s Law
Amdahl’s Law

• Task: serial part, parallel part
• As number of processors increases,
  – time to execute parallel part goes to zero
  – time to execute serial part remains the same
• Serial part eventually dominates
• Must parallelize ALL parts of task

\[ \text{Speedup}(E) = \frac{\text{Execution Time without } E}{\text{Execution Time with } E} \]

Kavita Bala, Computer Science, Cornell University

Amdahl’s Law

• Consider an improvement E
• F of the execution time is affected
• S is the speedup

\[
\text{Execution time (with } E) = ((1 - F) + F/S) \cdot \text{Execution time (without } E) \\
\text{Speedup (with } E) = \frac{1}{(1 - F) + F/S}
\]
Load Balancing

- Need to manage work so all units are actually operating
GPGPUs

• Scientific Computing
  – MATLAB codes
• Convex hulls
• Molecular Dynamics
• Etc.