Previously…

- Boolean logic
- Arithmetic: addition, subtraction
- Reuse:
  - pipelining
  - bit-serial (vectorization)
- State, FSMs
- Area/Time Tradeoffs
- Latency and Throughput

Today

- Memory
  - features
  - Area, delay, (energy) intuition and modeling
  - design
  - technology

Preclass 1

When is:

$$2000\sqrt{N} + N < 100N$$

Preclass 2

Find $m$ to minimize:

$$\frac{C_1 N \log_2(N)}{m} + C_2 m + C_3 N$$

Memory

- What’s a memory?
- What’s special about a memory?
Memory Function

- Typical:
  - Data Input Bus
  - Data Output Bus
  - Address
    - (location or name)
    - read/write control

Memory

- Block for storing data for later retrieval
- State element

Collection of Registers

- What’s different between a memory and a collection of registers?

Memory Uniqueness

- Cost
- Compact state element
- Packs data very tightly (Area)
- At the expense of sequentializing access
- Example of Area-Time tradeoff
  - and a key enabler

Memory Organization

- **Key idea:** sharing
  - factor out common components among state elements
  - can have big elements if amortize costs
  - state element unique → small

SRAM Memory bit

Source: http://commons.wikimedia.org/wiki/File:6t-SRAM-cell.png
Memory Organization

- Share: Interconnect
  - Input bus
  - Output bus
  - Control routing
- **very** topology/wire cost aware design
- Note: local, abuttment wiring

Share Interconnect

- Input Sharing
  - wiring
  - drivers
- Output Sharing
  - wiring
  - sensing
  - driving

Address/Control

- Addressing and Control
  - an overhead
  - paid to allow this sharing

Preclass 2

- How related?

\[
\frac{C_1N \log_2(N)}{m} + C_2m + C_3N
\]

- Related?
  - m=bw
  - bw*bd=N
  - So bd=N/m

Preclass 2

- \(C_1bd\log_2(N) + C_2bw + C_3N\)
  - \(C_3\) is for area of single memory cell
  - \(\log_2(n)\) is for decoder
    - In cyan
  - \(C_1\) is for gates in decoder
  - \(C_2\) is for amps at bottom of row and drivers at top
Preclass 1
\[
\frac{C_1 N \log_2(N)}{m} + C_3 m + C_3 N
\]
When we solved for \(N\), we found \(m\) is proportional to \(\sqrt{N \log_2(N)}\)
\[
C_4 \sqrt{N \log_2(N)} + C_3 N
\]
If we approximate \(\log_2(N)\) as a constant, we get
\[
C_6 \sqrt{N} + C_3 N
\]
\[2000 \sqrt{N} + N^{19}\]

Sharing Impact
\[
C_1 b d \log_2(N) + C_2 b w + C_3 N
\]
\[
C_4 \sqrt{N \log_2(N)} + C_3 N
\]
• For large \(N\), \(C_3\) term dominates

Compactness vs. Memory Size

Memory Organization

Dynamic RAM
• Goes a step further
• Share refresh/restoration logic as well
• Minimal storage is a capacitor
• “Feature” DRAM process is ability to make capacitors efficiently

Some Numbers (memory)
• Unit of area = \(F^2\) \((F=2x)\)
• Register as stand-alone element \(\approx 1000 F^2\)
  – e.g. as needed/used Day 4
• Static RAM cell \(\approx 250 F^2\)
  – SRAM Memory (single ported)
• Dynamic RAM cell (DRAM process) \(\approx 25 F^2\)
• Dynamic RAM cell (SRAM process) \(\approx 75 F^2\)
DRAM Layout

Fig 3. 6P Memory Cell Layout
The 6P cell layout used by Micron Technology has active regions used for transistor channels, etc., formed diagonally. Layout presented at the 2007 Symposium on VLSI Technology.

Source: http://techon.nikkeibp.co.jp/article/HONSHI/20071219/144399/

Memory Access Timing

- RAS/CAS access
- Optimization for access within a row

Table 1: Key Timing Parameters

<table>
<thead>
<tr>
<th>Mode</th>
<th>tRAS</th>
<th>tRP</th>
<th>tRTP</th>
<th>tRAS+</th>
<th>tRP+</th>
</tr>
</thead>
<tbody>
<tr>
<td>64MB</td>
<td>8NS</td>
<td>2NS</td>
<td>5NS</td>
<td>13NS</td>
<td>22NS</td>
</tr>
<tr>
<td>16MB</td>
<td>8NS</td>
<td>2NS</td>
<td>5NS</td>
<td>13NS</td>
<td>22NS</td>
</tr>
<tr>
<td>4MB</td>
<td>8NS</td>
<td>2NS</td>
<td>5NS</td>
<td>13NS</td>
<td>22NS</td>
</tr>
<tr>
<td>1MB</td>
<td>8NS</td>
<td>2NS</td>
<td>5NS</td>
<td>13NS</td>
<td>22NS</td>
</tr>
</tbody>
</table>

DRAM Latency

Processor-Memory Performance Gap:
(grows 50% / year)

1000
100
10
1

µProc 60%/yr.
DRAM 7%/yr.

Time

[From Patterson et al., IRAM: http://iram.cs.berkeley.edu]

1 Gigabit DDR2 SDRAM

**DRAM Latency**

- "Moore's Law"  
- Processor-Memory Performance Gap: (grows 50%/year)
- DRAM: 7%/yr.

![Graph showing DRAM Latency over time](http://iram.cs.berkeley.edu)

---

**Basic Memory Design Space**

- **Width**
- **Depth**
- **Internal vs. External Width**
- **Banking** – To come

![Diagram of basic memory design space](http://iram.cs.berkeley.edu)

---

**Depth**

- What happens as depth makes deeper?  
  - Delay?  
  - Energy?

![Diagram showing depth effects](http://iram.cs.berkeley.edu)

---

**Banking**

- Tile Banks/memory blocks
- What does banking do for us? (E, D, A)?

![Diagram showing banking effects](http://iram.cs.berkeley.edu)

---

**Compactness vs. Memory Size**

- Area/bit vs. Memory Capacity

![Graph showing compactness vs. memory size](http://iram.cs.berkeley.edu)

---

**1 Gigabit DDR2 SDRAM**

Independent Bank Access

- Utility? Costs and benefits?

Memory

- Key Idea
  - Memories hold state compactly
  - Do so by minimizing key state storage and amortizing rest of structure across large array

Yesterday vs. Today (Memory Technology)

- What’s changed?
  - Capacity
    - single chip
  - Integration
    - memory and logic
    - dram and logic
    - embedded memories
    - Multicore
  - Energy dominance
  - Room on chip for big memories
  - And many memories...
  - Don’t have to make a chip crossing to get to memory

Yesterday vs. Today (Memory Technology)

- What’s changed?
  - Capacity
    - single chip
  - Integration
    - memory and logic
    - dram and logic
    - embedded memories
    - Multicore
  - Energy dominance
  - Room on chip for big memories
  - And many memories...
  - Don’t have to make a chip crossing to get to memory

[Memory Technology Graph]

[Table: Key Timing Parameters]

On-Chip vs. Off-Chip BW

- Use Micron 1Gb DRAM as example

<table>
<thead>
<tr>
<th>Options</th>
<th>Marking</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Mb</td>
<td>GAMBO 128Mb 256Mb</td>
</tr>
<tr>
<td>8 Mb</td>
<td>128Mb</td>
</tr>
<tr>
<td>16 Mb</td>
<td>128Mb</td>
</tr>
<tr>
<td>32 Mb</td>
<td>128Mb</td>
</tr>
<tr>
<td>64 Mb</td>
<td>256Mb (256Mb)</td>
</tr>
</tbody>
</table>

| On Chip BW: assume 8 1024b banks? |

- multistage pipelining
- multithreading
- distributed memory
- multithreading

<table>
<thead>
<tr>
<th>Important Technology Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IO between chips &lt;&lt; IO on chip</td>
</tr>
<tr>
<td>- pad spacing</td>
</tr>
<tr>
<td>- area vs. perimeter (4s vs. s^2)</td>
</tr>
<tr>
<td>- wiring technology</td>
</tr>
<tr>
<td>• BIG factor in multi-chip system designs</td>
</tr>
<tr>
<td>• Memories nice</td>
</tr>
<tr>
<td>- very efficient with IO cost vs. internal area</td>
</tr>
</tbody>
</table>
Memory on a Virtex-7

- 28nm process node
- 1.2M programmable 6-input gates (6-LUTs)
- 1,880 memory banks ~ 512×72
  - Total ~ 65Mbits
- Operating at 600MHz
- On chip bandwidth?

Costs Change

- Design space changes when whole system goes on single chip
- Can afford
  - wider busses
  - more banks
  - memory tailored to application/architecture
- Beware of old (stale) answers
  - their cost model was different

What is Importance of Memory?

- Radical Hypothesis:
  - Memory is simply a very efficient organization which allows us to store data compactly
    - (at least, in the technologies we’ve seen to date)
  - A great engineering trick to optimize resources
- Alternative:
  - memory is a primary
- State is a primary, but state≠memory

Big Ideas

- Memory: efficient way to hold state
- Resource sharing: key trick to reduce area
- Memories are a great example of resource sharing

Big Ideas [MSB-1 Ideas]

- Tradeoffs in memory organization
- Changing cost of memory organization as we go to on-chip, embedded memories

Admin

- Piazza
- HW2 late graded (as of 6pm last night)
- HW3 due today
- HW4 out
- Reading on web
- Drop Date Friday
- No office hours Tuesday 2/25
  - Piazza, email, schedule apt. as needed