CS/ECE 5780/6780
Embedded Systems Design

Lecture 9: Memory Technologies

Jon Davies

February 11, 2014
# Memory technologies landscape

<table>
<thead>
<tr>
<th></th>
<th>Volatile</th>
<th>Non-Volatile</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RAM</strong></td>
<td>Static RAM (SRAM)</td>
<td>EEPROM</td>
</tr>
<tr>
<td></td>
<td>Dynamic RAM (DRAM)</td>
<td>Flash Memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FRAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MRAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FRAM</td>
</tr>
<tr>
<td><strong>ROM</strong></td>
<td>n/a</td>
<td>Mask ROM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PROM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EPROM</td>
</tr>
</tbody>
</table>
Choosing the right memory requires balancing many tradeoffs

- **Volatility**: need to retain state during power down?
- **Cost**: wide range of absolute $ and $/bit costs
- **Organization**: 64Kbx1 or 8Kbx8?
- **Interface**
  - Serial or parallel?
  - Synchronous or asynchronous?
- **Access times**: critical for high-performance
- **Modify times**: critical for write-intensive workloads
- **Erase process**: at wire-line speed or 5 minutes in UV?
- **Erase granularity**: word, page, sector, chip?
Internal organization of memory is usually an array

Different memory types (e.g. SRAM vs DRAM) are distinguished by the technology used to implement the memory cell, e.g.:
- SRAM: 6T
- DRAM: 1T/1C

What should be the aspect ratio (# rows vs #cols)?
Physical (on-chip) memory configuration

- Physical configurations are typically square
- Square minimizes length of (word line + bit line)
- Shorter length means
  - Shorter propagation time
  - Faster data access
  - Smaller $t_{rc}$ (read cycle time)

- Exercise: Assume $n^2$ memory cells configured as
  - n-by-n square array. What is the worst case delay?
  - $n^2$-by-1 rectangular. What is the worst case delay?

- Exercise: Does wire length dominate access time?
  - Assume propagation speed on chip is $2/3$ c ($2 \times 10^8$ m/s)
  - Assume 2Mbit array is 1.5 cm x 1.5 cm
• External configurations are tall and narrow
  - More address lines (12 to 20+, typically)
  - Fewer data lines (8 or 16, typically)

• The narrower the configuration
  - The greater the pin efficiency
  - Adding one address pin cuts data pins in half
  - The easier the data bus routing

• Many external configurations for given capacity
  - 64 Kb = 64K
  - 64 Kb = 32K
  - 64 Kb = 16K
  - 64 Kb = 8K
  - 64 Kb = 4K
  - 64 Kb = 2K
Supporting circuitry is needed to address memory cell and enable reads and writes.

Control signals:
- Select chip
- Select memory cell
- Control read/write
- Map internal array to external configuration

(4x4 $\rightarrow$ 16x1)
Figure 2. System Configuration with SPI port
Refresher on the memory-bus interface

• Chip Select (CS#)
  - Enables device
  - Ignores all other inputs if CS# is not asserted

• Write Enable (WE#)
  - Enables write tri-state buffer
  - Store D0 at specified address

• Output Enable (OE#)
  - Enable read tri-state buffer
  - Drive D0 with value at specified address
Mask ROM

- The “simplest” memory technology
- Presence/absence of diode at each cell denote value
- Pattern of diodes defined by mask used in fab process
- Contents are fixed when chip is made; cannot be changed
- High upfront setup costs (mask costs)
- Small recurring marginal costs
- Good for applications where
  - Cost sensitivity drives design
  - Upgrading contents not an issue
  - e.g. boot ROM, CPU microcode
- Exercise:
  - What “value” does a diode encode?
  - What are the contents:
    - Where A<2:0> = 101?
    - Where A<2:0> = 110?
EPROM

- **Erasable Programmable Read-Only Memory**
- Constructed from floating gate FETs
  - Charge trapped on the FG erases cell
  - High voltage (13V +) applied to the control gate
    - “Writes” the cell with a 0
    - Allows FG charge to be dissipated
- Erasing means changing form 0 \(\rightarrow\) 1
  - Uses UV light (not electrically!)
  - Electrons are trapped on a floating gate
- Writing means changing from 1 \(\rightarrow\) 0
- Erase unit is the whole device
- Retains data for 10-20 years
- Not used much these days
- Costly because
  - Use of quartz window (UV transparent)
  - Use of ceramic package
- PROM (or OTP) is same, just w/o window
Flash Memory

- Electrically erasable (like EEPROM, unlike EPROM)
- Used in many reprogrammable systems these days
- Erase size is block (not word); can’t do byte modifications
- Erase circuitry moved out of cells to periphery
  - Smaller size
  - Better density
  - Lower cost
- Reads are like standard RAM
- Can “write” bits/words (actually, change from 1 → 0)
  - Write cycle is O(microseconds)
  - Slower then RAM but faster than EEPROM
  - To (re)write from 0 → 1, must explicitly erase entire block
    - Erase is time consuming O(milliseconds to seconds)
- Floating gate technology
  - Erase/write cycles are limited (10K to 100K, typically)
Flash Memory Wear

- If a block was erased and written to every minute how long would it take to fail?

- Using a 16Gb flash with 8192 blocks how long before failure if you distribute the erase and writes among the whole chip?
Cost-per-bit

Standby Power

File Storage Use

Active Power (*)

Code Execution

Read Speed

Write Speed

Capacity

(*): Dependant on how memory is used.
NOR is typically slow on writes and consumes more power than NAND.
NOR is typically fast on reads, which consume less power.
<table>
<thead>
<tr>
<th></th>
<th>NAND</th>
<th>NOR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cell Array</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Unit Cell</strong></td>
<td><strong>Unit Cell</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Bit line</strong></td>
<td><strong>Bit line</strong></td>
</tr>
<tr>
<td><strong>Word line</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Source line</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Layout</strong></td>
<td>2F</td>
<td>5F</td>
</tr>
<tr>
<td></td>
<td>2F</td>
<td>2F</td>
</tr>
<tr>
<td><strong>Cross Section</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cell Size</strong></td>
<td>4F²</td>
<td>10F²</td>
</tr>
</tbody>
</table>
Serial input (x8 or x16): 25ns (MAX CLK)

NAND Flash Memory Array

PROGRAM: ≈ 220μs/page

NAND Flash Page 2112 bytes

READ (page load): ≈ 25μs

NAND Flash Block

BLOCK ERASE: ≈ 500μs

2048 blocks (2Gb SLC device)

Data area: 2048 bytes

Spare area (ECC, etc.) 64 bytes

8-bit byte or 16-bit word
The diagram illustrates the processes of CHE Injection and FN Tunneling in a semiconductor device.

**CHE Injection**

- Source
- +12V
- Gate
- Floating gate
- Drain
- GND
- +~6V

**FN Tunneling**

- Source
- +12V
- Gate
- Floating gate
- Drain
Static RAM

- SRAMs are volatile
- Basic cell
  - Bistable core
    - 4T: uses pullup resistors for M2, M4
    - 6T: uses P-FET for M2, M4
  - Access transistors
    - BL, BL# are provided to improve noise margin
- 6T is typically used (but has poor density)
- Fast access times $O(10 \text{ ns})$
- Read/write speeds are symmetric
- Read/write granularity is word
Static RAM

Diagram of a static RAM cell with transistors labeled $M_1$ to $M_6$, voltage levels $VL$ and $V_{DD}$, and input/output signals $Q$, $\overline{Q}$, $BL$, and $\overline{BL}$. The circuit includes diodes and capacitors associated with each transistor to store and maintain the state of the RAM cell.
Static RAM
Dynamic RAM

- Requires only 1T and 1C per cell
- Outstanding density and low cost
- Compare to the 6T’s per SRAM cell
- Cost advantage to DRAM technology

- Small charges involved $\rightarrow$ relatively slow
  - Bit lines must be pre-charged to detect bits
  - Reads are destructive; internal writebacks needed

- Values must be refreshed periodically
  - Prevents charge from leaking away
  - Complicates control circuitry slightly