Quick Review over the Last Lecture

DRAM:
- Data stored in a capacitor.
- Electric charge needs to be refreshed.
- DRAM requires large power consumption.

Read-out operation of 1C1T:
- "1"-data:
  \[ 1 \text{ V} + \Delta V = 2 \text{ V} \]
- "0"-data:
  \[ 0 \text{ V} = 1 \text{ V} - \Delta V \]

Refresh operation of DRAM:
- When data is "1":
  \[ 3.6 \text{ V} \]
- When data is "0":
  \[ 3.6 \text{ V} \]

* http://www.wikipedia.org/
* http://users.cis.fiu.edu/~prabakar/cda4101/Common/notes/lecture09.html
10 Static Random Access Memory

- Volatile memory development
- 6T-SRAM architecture
- Read / write operation
  - 1T-SRAM
  - Various ROMs

Memory Types

- Rewritable
- Volatile
- Dynamic
  - SRAM
  - DRAM
- Static
- Non-volatile
- Static

- Read only
- Non-volatile
- Static
  - PROM
  - Mask ROM

- Read majority (Writable)
- Non-volatile
- Static
  - Flash
  - EPROM

* http://www.semiconductorjapan.net/serial/lesson/12.html
In 1949, Frederic C. Williams and Tom Kilburn developed Manchester Mark 1:

One of the earliest computers

Williams-Kilburn Tube

Cathode-ray tube to store data:

Utilise a small change of electron charges at a fluorescent screen when an electron hit it.
In 1947, John P. Eckart invented a mercury delay line memory:

Utilise an ultrasonic wave generated by a transducer to store a data.

In 1953, Jan A. Rajchman (RCA) invented a selectron tube:

An array of cathode-ray tubes is used to store data electrostatically.
Static Random Access Memory (SRAM)

Static random access memory (SRAM):

No need to dynamically refresh data.

→ Even so, the data is lost once the power is off.

Flip flop is used to store data.

→

6T-SRAM Read Operation

A standard SRAM cell:

READ OPERATION: i) Precharge BL, BL' to HIGH
ii) Turn on WL
iii) BL or BL' will be pulled down to LOW depending on Q, Q'
iv) Eg. If Q = 0, Q' = 1, BL discharges through N2 - N1 - GND and BL' stays high. But Q bumps up slightly
v) In order for Q to not flip N1 should be stronger than N2, i.e N1 >> N2

* http://allthingsvlsi.wordpress.com/tag/6t-sram-operation/
6T-SRAM Write Operation

Write operation:

1. Drive BL, BL' with necessary values
2. Turn on WL
3. Bit lines overpower cell with new value
4. Eq. Q = 0, Q' = 1 and BL = 1, BL' = 0. This forces Q' to low and Q to high
5. To overpower feedback inverter loop, N2 should be stronger than P1, i.e. N2 >> P1

As Q starts to charge up to 1, output of inverter
P2-N3 starts to discharge which in turn, makes P1
turn on. Thus the feedback inverters lock on to
the values to be written

* http://allthingsvlsi.wordpress.com/tag/6t-sram-operation/

6T-SRAM Operation

* https://www.youtube.com/watch?v=kU2SsUUsftA
1T-SRAM

Pseudo SRAM developed by MoSys:

By comparing with the conventional 6T-SRAM,

- < 1/3 area
- lower power consumption
- to be embedded
- simple interface
- similar to SRAM performance
- latency as compared with DRAM
- fidelity (< 1 FIT / Mbit, FIT : failure in time of $10^9$ hours)

![Diagram of 1T-SRAM structure]

* http://www.wikipedia.org/
** http://www.mosys.com/high-density-memory.php

Advantages of 1T-SRAM

Comparison between 1T-SRAM, embedded DRAM (eDRAM) and 6T-SRAM:

<table>
<thead>
<tr>
<th></th>
<th>eDRAM</th>
<th>1T-SRAM</th>
<th>6T-SRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Technology</strong></td>
<td>High</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Complexity</td>
<td>Low</td>
<td>Early</td>
<td>Early</td>
</tr>
<tr>
<td>Availability</td>
<td>Limited</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>Very High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Macro Density</td>
<td>Slow</td>
<td>Fast</td>
<td>High</td>
</tr>
<tr>
<td>Speed</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Active Power</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Manufacturing</strong></td>
<td>Very Long</td>
<td>Short</td>
<td>Short</td>
</tr>
<tr>
<td>Cycle Time</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Yield</td>
<td>Limited</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Capacity</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* http://media.corporate-ir.net/media_files/nsd/mosy/presentations/corp_pres_1103/sld012.htm
Mask ROM

Read-only memory made by a photo-mask:

- Cheap
- Simple structure
- Ideal for integration
- Initial mask fabrication cost
- Lead time for mask fabrication
- No design change without mask replacement

Programmable ROM (PROM)

PROM bipolar cell:

* http://www.smspower.org/Development/MaskROMs

* http://www.electronics.dit.ie/staff/tscarff/memory/rom.htm
PROM Architecture

PROM architecture:

* http://www.electronics.dit.ie/staff/tscarff/memory/rom.htm

Erasable PROM

UV-light can erase stored data:

* http://www.wikipedia.org/
* http://www.electronics.dit.ie/staff/tscarff/memory/rom.htm
Electrically EPROM (EEPROM)

In 1978, George Perlegos (Intel) developed electrical erasing mechanism:

- To flash memory
- Individual bits are erasable.
- Rewritable by simply writing a new data
- Rewritability > 100k times
- × capacity (~ bytes)
- × More complicated architecture as compared with flash memory
- × Higher cost for fabrication as compared with flash memory

EEPROM Usages

EEPROM is used in various applications:

<table>
<thead>
<tr>
<th>Memory capacity</th>
<th>Setting Value</th>
<th>ID Code</th>
<th>Compensation Data</th>
<th>Maintenance Data</th>
<th>Authentication Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Home Appliance Fine-Tuned Settings</td>
<td>TV Channel Setting Memory</td>
<td>AV Equipment User Setting Values</td>
<td>Communication and Industrial Equipment Maintenance Data</td>
<td>Mobile Devices Battery Authentication</td>
</tr>
<tr>
<td></td>
<td>Set-Top Box IDs and Security Codes</td>
<td>Mobile phone ID Codes</td>
<td>Image Sensor Stabilization (Compensation for Inconsistency)</td>
<td>Counter Records</td>
<td>Printers Cartridge Authentication</td>
</tr>
</tbody>
</table>

Specific Examples of Applications

Connections between the Components

Connections between CPU, in/outputs and storages:

* http://testbench.in/introduction_to_pci_express.html;