Memories and SRAM
Silicon Memories

- Why store things in silicon?
  - It’s fast!!!
  - Compatible with logic devices (mostly)

- The main goal is to be cheap
  - Dense -- The smaller the bits, the less area you need, and the more bits you can fit on a chip/wafer/through your fab.
  - Bit sizes are measured in $F^2$ -- the smallest feature you can create.
  - $F^2$ is a function of the memory technology, not the manufacturing technology.
  - i.e. an SRAM in today’s technology will take the same number of $F^2$ in tomorrow’s technology
Questions

• What physical quantity should represent the bit?
  • Voltage/charge -- SRAMs, DRAMs, Flash memories
  • Magnetic orientation -- MRAMs
  • Crystal structure -- phase change memories
  • The orientation of organic molecules -- various exotic technologies
  • All that’s required is that we can sense it and turn it into a logic one or zero.

• How do we achieve maximum density?
• How do we make them fast?
Anatomy of a Memory

- **Dense:** Build a big array
  - bigger the better
  - less other stuff
  - Bigger -> slower
- **Row decoder**
  - Select the row by raising a “word line”
- **Column decoder**
  - Select a slice of the row
- **Decoders are pretty big.**
The Storage Array

• Density is king.
  • Highly engineered, carefully tuned, automatically generated.
  • The smaller the devices, the better.

• Making them big makes them slow.
  • Bit/word lines are long (millimeters)
  • They have large capacitance, so their RC delay is long
  • For the row decoder, use large transistors to drive them hard.

• For the bit cells...
  • There are lots of these, so they need to be as small as possible (but not smaller)
Sense Amps

- Sense amplifiers take a difference between two signals and amplify it
- Two scenarios
  - Inputs are initially equal ("precharged") -- they each move in opposite directions
  - One input is a reference -- so only one signal moves
- Frequently used in memories
  - Sense amps can detect small analog signals from the storage cell, and convert it into a logic one or logic zero.
Static Random Access Memory (SRAM)

- **Storage**
  - Voltage on a pair of cross-coupled inverters
  - Durable in presence of power

- **To read**
  - Pre-charge two bit lines to Vcc/2
  - Turn on the “word line”
  - Read the output of the sense-amp
Static Random Access Memory (SRAM)

- **Storage**
  - Voltage on a pair of cross-coupled inverters
  - Durable in presence of power

- **To read**
  - Pre-charge two bit lines to Vcc/2
  - Turn on the “word line”
  - Read the output of the sense-amp
Static Random Access Memory (SRAM)

- **Storage**
  - Voltage on a pair of cross-coupled inverters
  - Durable in presence of power

- **To read**
  - Pre-charge two bit lines to Vcc/2
  - Turn on the “word line”
  - Read the output of the sense-amp
Static Random Access Memory (SRAM)

- **Storage**
  - Voltage on a pair of cross-coupled inverters
  - Durable in presence of power

- **To read**
  - Pre-charge two bit lines to Vcc/2
  - Turn on the “word line”
  - Read the output of the sense-amp
Static Random Access Memory (SRAM)

- **Storage**
  - Voltage on a pair of cross-coupled inverters
  - Durable in presence of power
- **To read**
  - Pre-charge two bit lines to Vcc/2
  - Turn on the “word line”
  - Read the output of the sense-amp
SRAM Writes

• To write
  • Turn off the sense-amp
  • Turn on the wordline
  • Drive the bitlines to the correct state
  • Turn off the wordline
SRAM Writes

• To write
  • Turn off the sense-amp
  • Turn on the wordline
  • Drive the bitlines to the correct state
  • Turn off the wordline
SRAM Writes

• To write
  • Turn off the sense-amp
  • Turn on the wordline
  • Drive the bitlines to the correct state
  • Turn off the wordline
SRAM Writes

- To write
  - Turn off the sense-amp
  - Turn on the wordline
  - Drive the bitlines to the correct state
  - Turn off the wordline

![SRAM Write Diagram]
SRAM Writes

- To write
  - Turn off the sense-amp
  - Turn on the wordline
  - Drive the bitlines to the correct state
  - Turn off the wordline
Building SRAM

- This is “6T SRAM”
- 6 transistors is pretty big
- SRAMs are not dense
SRAM Density

- At 65nm: 0.52um²
- 123-140 F²
- 1 F² is one “square feature”
- [ITRS 2008]
SRAM Ports

- Add word and bit lines
- Read/write multiple things at once
- Density decreases quadratically
- Bandwidth increase linearly
SRAM Performance

- Read and write times
  - 10s-100s of ps
- Bandwidth
  - Registers -- 324GB/s
  - L1 cache -- 128GB/s
SRAM’s future

- SRAM is a mature technology. No new, big breakthroughs or advances are expected beyond CMOS scaling.