Lecture 14: Memory Elements

Mark McDermott
Electrical and Computer Engineering
The University of Texas at Austin
Memory Elements

- Memory arrays
- SRAMs
- Serial Memories
- Dynamic memories

Pentium-4 (Willamette)
Yellow boxes are memory arrays
Memory Arrays

Random Access Memory
- Read/Write Memory (RAM) (Volatile)
  - Static RAM (SRAM)
  - Dynamic RAM (DRAM)
- Read Only Memory (ROM) (Nonvolatile)
  - Programmable ROM (PROM)
  - Erasable Programmable ROM (EPROM)
  - Electrically Erasable Programmable ROM (EEPROM)

Serial Access Memory
- Shift Registers
  - Serial In Parallel Out (SIPO)
  - Parallel In Serial Out (PISO)
- Queues
  - First In First Out (FIFO)
  - Last In First Out (LIFO)

Content Addressable Memory (CAM)

Memory Arrays
1D Memory Architecture

n words → n select signals

Decoder reduces # of inputs

$k = \log_2 n$
2D Memory Architecture

Row Address: $A_j, A_{j+1}, A_{k-1}$

Column Address: $A_0, A_1, A_{j-1}$

Row Decoder: $2^{k-j}$

Column Decoder: $m2^j$

Storage (RAM) cell

Sense Amplifiers

Read/Write Circuits

Input/Output (m bits)

bit line

word line

selects appropriate word from memory row

amplifies bit line swing
Array Architecture

- $2^n$ words of $2^m$ bits each
- If $n \gg m$, fold by $2^k$ into fewer rows of more columns

- Good regularity – easy to design
- Very high density if good cells are used
SRAM Simulation Cross Section
6T SRAM Cell

- Cell size accounts for most of array size
  - Reduce cell size at expense of complexity

- 6T SRAM Cell
  - Used in most commercial chips
  - Data stored in cross-coupled inverters

- Read:
  - Precharge bit, bit_b
  - Raise wordline

- Write:
  - Drive data onto bit, bit_b
  - Raise wordline
Moore’s Law Scaling for Memory

Source: Kelin Kuhn, Intel
SRAM Memory Cell Improvements

22 nm Process

.108 \text{um}^2
(Used on CPU products)

14 nm Process

.0588 \text{um}^2
(0.54x area scaling)
12T SRAM Cell

- Basic building block: SRAM Cell
  - Holds one bit of information, like a latch
  - Must be read and written

- 12-transistor (12T) SRAM cell
  - Use a simple latch connected to bitline
  - 46 x 75 \( \lambda \) unit cell
SRAM Read

- Improve performance when bit-line capacitance is high
- Precharge both bitlines high
- Then turn on wordline
- One of the two bitlines will be pulled down by the cell

- Ex: $A = 0, A\_b = 1$
  - bit discharges, bit\_b stays high
  - But A bumps up slightly

- Read stability
  - A must not flip
  - $N1 >> N2$

![Diagram of SRAM read circuit](image-url)
Read Timing

CLK, WL, BL/BL#, SAE, YSEL#, BLPC#, SAPC#, SAOUT, DATA_OUT

\( \phi_1 \) and \( \phi_2 \)

BL equalized, DV differential, BL precharge, SA precharge

Evaluate/latch

t

10/23/18
Read Requirements

- Pre-charge & equalize bit-lines from previous cycle
- Minimum “Design Margin” before next READ begins
- Delay requirement to allow sufficient bit-line voltage development

**NOTE:** The $\Delta t$ delay can be generated by a chain of inverter delays or by replica “dummy” row and column composed of bitcells.
SRAM Write

- Drive one bitline high, the other low
- Then turn on wordline
- Bitlines overpower cell with new value

- Ex: $A = 0$, $A_b = 1$, $\text{bit} = 1$, $\text{bit}_b = 0$
  - Force $A_b$ low, then $A$ rises high

- **Writability**
  - Must overpower feedback inverter
  - $N4 \gg P2$
  - $N2 \gg P1$ (symmetry)
Write Timing

- **WREN**
- **CLK**
- **WL**
- **WSEL**
- **BL/BL#**
- **BLPC#**
- **DATA**
- **DATA_OUT**

- Data arrives early from previous cycle
- Holds state from previous READ
- Write pulsewidth
- Write mode from latch
- Read mode
- Time
- BL equalized
- BL precharge

Note: The image captures the timing diagram for write operations in a memory circuit, highlighting the timing relationships between various signals and the timing parameters involved.
Pre-charge & equalize bit-lines from previous cycle
WRITE can begin as soon as word-line is available
Must guarantee minimum write pulse-width, data valid time and write recovery; internal “high node” reaches say 90% of VDD

NOTE: Write pulse-width margin increases with lower frequency
SRAM Sizing

- High bitlines must not overpower inverters during reads
- But low bitlines must write new value into cell

\[
\text{Wdn} > \text{Wpass} \quad \text{for read stability} \\
\text{Wpass} > \text{Wup} \quad \text{to enable writes}
\]
In 45nm CMOS, a typical 6T bit-cell area = 0.38 μm²
Decoders

- \( n:2^n \) decoder consists of \( 2^n \) \( n \)-input AND gates
  - One needed for each row of memory
  - Build AND from NAND or NOR gates

Static CMOS

Pseudo-nMOS
Decoders must be pitch-matched to SRAM cell
  - Requires very skinny gates
Large Decoders

- For $n > 4$, NAND gates become slow
  - Break large gates into multiple smaller gates
Many of these gates are redundant
- Factor out common gates into pre-decoder
- Saves area
- Same path effort
Some circuitry is required for each column
  – Bitline conditioning
  – Sense amplifiers
  – Column multiplexing
Bitline Conditioning

- Precharge bitlines high before reads

- Equalize bitlines to minimize voltage difference when using sense amplifiers
Sense Amplifiers

- **Bitlines have many cells attached**
  - Ex: 32-kbit SRAM has 256 rows x 128 cols
  - 128 cells on each bitline

- **$t_{pd} \propto (C/I) \Delta V$**
  - Even with shared diffusion contacts, 64C of diffusion capacitance (big C)
  - Discharged slowly through small transistors (small I)

- **Sense amplifiers are triggered on small voltage swing (reduce $\Delta V$)**
Differential Pair Amp

- Differential pair requires no clock
- But always dissipates static power
Clocked Sense Amp

- Clocked sense amp saves power
- Requires timing the sense_clk signal to arrive after enough bitline swing
- Isolation transistors cut off large bitline capacitance

```
bit    bit_b
sense_clk

isolation transistors

regenerative feedback

sense sense_b
```
De-coupled Sense Amplifier

- With SAE low; Data and Data# are pre-charge high
- When SAE goes high; source-coupled pair acts as differential amplifier
- Cross coupled inverters amplify and latch any voltage difference
Twisted Bitlines

- Sense amplifiers also amplify noise
  - Coupling noise is severe in modern processes
  - Try to couple equally onto bit and bit\_b (common mode)
  - Done by twisting bitlines

```
  b0  b0\_b  b1  b1\_b  b2  b2\_b  b3  b3\_b
```

![Diagram of twisted bitlines](image)
Column Multiplexing

- Recall that array may be folded for good aspect ratio
- Ex: 2 kword x 16 folded into 256 rows x 128 columns
  - Must select 16 output bits from the 128 columns
  - Requires 16 8:1 column multiplexers
Tree Decoder Mux

- Column mux can use pass transistors
  - Use nMOS only, precharge outputs
- One design is to use k series transistors for 2k:1 mux
  - No external decoder logic needed
Single Pass-Gate Mux

- Or eliminate series transistors with separate decoder
Ex: 2-way Muxed SRAM

VLSI-1 Class Notes
Multiple Ports

- **We have considered single-ported SRAM**
  - One read or one write on each cycle
- **Multiported SRAM are needed for register files**
- **Examples:**
  - Multicycle MIPS must read two sources or write a result on some cycles
  - Pipelined MIPS must read two sources and write a third result each cycle
  - Superscalar MIPS must read and write many sources and results each cycle
Dual-Ported SRAM

- Simple dual-ported SRAM
  - Two independent single-ended reads
  - Or one differential write

- Do two reads and one write by time multiplexing
  - Read during ph1, write during ph2
Multi-Ported SRAM

- Adding more access transistors hurts read stability
- Multi-ported SRAM isolates reads from state node
- Single-ended design minimizes number of bitlines
BASIC ARRAY LAYOUT

**Columns**

**WordLine**

**BitLine**

**Word Line Decoder**

**Column Decoder**

**Bit-line Sense Amplifiers**

**Write Buffers**

**Pre-Decode**

**Address**

**Write Data**

**Read Data**

**Rows**
SRAM Layout Using a Memory Compiler
CAMs

- Extension of ordinary memory (e.g. SRAM)
  - Read and write memory as usual
  - Also *match* to see which words contain a *key*
10T CAM Cell

- Add four match transistors to 6T SRAM
  - 56 x 43 unit cell
CAM Cell Operation

- Read and write like ordinary SRAM

For matching:
- Leave wordline low
- Precharge matchlines
- Place key on bitlines
- Matchlines evaluate

Miss line
- Pseudo-nMOS NOR of match lines
- Goes high if no words match
4x4 DRAM Memory

2 bit words

Row Decoder

read
precharge

enable

A₁

A₂

Column Decoder


bit line precharge
WL[0]
WL[1]
WL[2]
WL[3]
sense amplifiers
write circuitry

clocking, control, and refresh

A₀
3-Transistor DRAM Cell

No constraints on device sizes (ratioless)
Reads are non-destructive
Value stored at node X when writing a “1” is $V_{WWL} - V_{tn}$
1-Transistor DRAM Cell

Write: $C_s$ is charged (or discharged) by asserting WL and BL
Read: Charge redistribution occurs between $C_{BL}$ and $C_s$

Read is destructive, so must refresh after read
1-T DRAM Cell

(a) Cross-section

(b) Layout

Used Polysilicon-Diffusion Capacitance

Expensive in Area
Dense 1T DRAM Cell

- Cell Plate Si
- Capacitor Insulator
- Storage Node Poly
- 2nd Field Oxide
- Refilling Poly
- Si Substrate

Trench Cell
DRAM Cell Observations

- DRAM memory cells are single ended (complicates the design of the sense amp)
- 1T cell requires a sense amp for each bit line due to charge redistribution read
- 1T cell read is destructive; refresh must follow to restore data
- 1T cell requires an extra capacitor that must be explicitly included in the design
- A threshold voltage is lost when writing a 1 (can be circumvented by bootstrapping the word lines to a higher value than Vdd)
Serial Access Memories

- **Serial access memories do not use an address**
  - Shift Registers
  - Tapped Delay Lines
  - Serial In Parallel Out (SIPO)
  - Parallel In Serial Out (PISO)
Shift Register

- Shift registers store and delay data
- Simple design: cascade of registers
  - Watch your hold times!
Denser Shift Registers

- Flip-flops aren’t very area-efficient
- For large shift registers, keep data in SRAM instead
- Move R/W pointers to RAM rather than data
  - Initialize read address to first entry, write to last
  - Increment address on each cycle
A tapped delay line is a shift register with a programmable number of stages.

Set number of stages with delay controls to mux
- Ex: 0 – 63 stages of delay
Serial In Parallel Out

- 1-bit shift register reads in serial data
  - After N steps, presents N-bit parallel output
Parallel In Serial Out

- Load all $N$ bits in parallel when shift = 0
  - Then shift one bit out per cycle
SOFT ERROR RATE (SER)
BACKGROUND

- There are 2 categories of system failure:
  - hard failure (permanent failures that require replacement)
  - soft failure (non-permanent random system failures)
- Cause of failures could be noise, power glitches, design margins, etc
- In large memory systems, soft errors are mostly due to radiation
- In 1978, May & Woods[5] (Intel) found radioactive materials in memory packages emitting alpha particles which can generate sufficient charge to switch the state of stored charge in DRAMs
- Minute traces of radioactive elements can be found in alumina-based ceramics, zirconia & silica fillers used in packaging
- Another potential source of alpha particles is from cosmic radiation
  - High energy particles from cosmic rays can have energies greater than 1GeV
  - Alpha particle energies typically range from 0.1 to 10 MeV
ALPHA PARTICLES

- An alpha-particle is a doubly charged helium nucleus (2 protons, 2 neutrons) that is generated during radioactive decay of high-Z atoms.

- More than 300 known alpha-emitting nuclides:
  - Uranium(238), Thorium(232) can be found in package materials for semiconductors.
  - Radioactive decay of U238 → Th234 + He4 until it decays to a stable Pb206 (8 alphas are generated).
  - Thorium generates 6 alphas as it decays from Th232 to stable Pb208.

- Alpha particles interact with silicon to generate an ionization trail of electron-hole pairs.

- The amount of electron-hole pairs generated depends on the particle’s initial energy (~3.6eV per e-h pair).

10/23/18
VLSI-1 Class Notes
Page 58
α Particle Energy for Silicon

Electron-Hole Pair Generation versus alpha energy for Silicon

Alpha particles generated from Th\textsuperscript{232} and U\textsuperscript{238} will have energies ranging from 3.95MeV to 8.78MeV
FUNNEL EFFECT

Depletion region distortion results in enhanced charge collection by the hit node.

Drift current causes the storage node to discharge.

If the total charge exceeds $Q_{\text{crit}}$, then state of bitcell or latch will flip.

Funnel of charge due to E-H pair generation.

$Q_{\text{crit}}$ = number of electrons which differentiates between a “1” and “0”  
(May & Woods) \cite{1}

He$^{++}$

Alpha track

$J = \text{Angle of Incidence}$

$\text{SiO}_2$

$\text{N}$

$\text{P}$
SOFT ERROR RATE

6-transistor SRAM cell

α particle hit

a-particle “hit” can be modeled as a sub-nanosecond current pulse as described by Chenming Hu
Memory Array Redundancy
BASIC ARRAY LAYOUT

- Cells
- Address
- WordLine
- BitLine
- Bitline Receivers
- Write Buffers
- Precharge
- Decoder
- Read Data
- Write Data

Columns:
- Precharge
- WordLine
- Bitline

Rows:
- Decoder
- Pre-Dec
- Address
- Write Data
- Read Data
ARRAY REDUNDANT ELEMENTS

- Redundant Wordline & Driver
- Redundant Address & enable
- Redundant Column & Bitslice

Account for area overhead if redundancy is used for repair.
SRAM Cell Stability Analysis
SRAM Cell Stability Analysis

- Bitlines are precharged to VCC → this is the critical situation because the nmos access device “shunts” the pmos load device; thereby reducing the gain of the inverters
- Static noise voltage sources are inserted into cross-couple path between inverters

![Diagram of SRAM cell with noise voltage source](image)

- SNM is defined by the maximum value of Vn that can be tolerated before changing state; sweep noise voltage from 0V to the point where differential collapses to zero
- Include temperature, voltage and process variation using a Monte Carlo simulator
SRAM Cell Stability Analysis

Retention Curve

Large diagonal due to access devices being “OFF”
Standby or “Data Retention” state when wordline is not selected → Cell is stable

The “maximum square” is approximately $V_{CC}/2$
SRAM Cell Stability Analysis

The "maximum square" represents SNM of the bitcell.

Transform coordinates 45°:

\[
\begin{align*}
x &= \frac{1}{\sqrt{2}} u + \frac{1}{\sqrt{2}} v \\
y &= \frac{1}{\sqrt{2}} u + \frac{1}{\sqrt{2}} v
\end{align*}
\]
Backup
4-Transistor Dynamic RAM Cell

Remove the two p-channel transistors from static RAM cell, to get a four-transistor dynamic RAM cell.

Data stored as charge on gate capacitors (complementary nodes)

Data must be refreshed regularly

Dynamic cells must be designed very carefully
3-Transistor Dynamic RAM Cell

Data stored on the gate of a transistor
Need two additional transistors, one for write and the other for read control
1-Transistor Dynamic RAM Cell

Value of $C_B$ must be chosen very carefully; otherwise, voltage on bit-line will be affected by charge sharing.

Cannot get any smaller than this: data stored on a (trench) capacitor $C$, need a transistor to control data. Bit line normally precharged to $\frac{1}{2} V_{DD}$ (need a sense amplifier).
SRAM Layout

- Cell size is critical: $26 \times 45 \lambda$ (even smaller in industry)
- Tile cells sharing $V_{DD}$, GND, bitline contacts