Memory Structures

Ramon Canal
NCD - Master MIRI

Slides based on: Introduction to CMOS VLSI Design. D. Harris
Outline

• Memory Arrays
• SRAM Architecture
  – SRAM Cell
  – Decoders
  – Column Circuitry
  – Multiple Ports
• Serial Access Memories
Memory Arrays

Random Access Memory
- Read/Write Memory (RAM) (Volatile)
  - Static RAM (SRAM)
  - Dynamic RAM (DRAM)
- Read Only Memory (ROM) (Nonvolatile)
  - Serial In Parallel Out (SIPO)
  - Parallel In Serial Out (PISO)

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

Content Addressable Memory (CAM)
- Shift Registers
- Queues

Non Volatile Memories
- Mask ROM
- Programmable ROM (PROM)
- Erasable Programmable ROM (EPROM)
- Electrically Erasable Programmable ROM (EEPROM)
- Flash ROM

NCD - Master MIRI
Array Architecture

• $2^n$ words of $2^m$ bits each
• If $n >> m$, fold by $2^k$ into fewer rows of more columns

- Good regularity – easy to design
- Very high density if good cells are used
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 λ unit cell
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
SRAM Read

- 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

![Diagram of SRAM Read](image)
SRAM Read

- 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
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
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
  - $N2 \gg P1$
SRAM Sizing

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

```
bit bit_b

A

A_b

strong

weak

med

word
```
SRAM Column Example

Read

Write
SRAM Layout

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

- Decoders
- Sense Amplifiers
- Input/Output Buffers
- Control / Timing Circuitry
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
Decoder Layout

- 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
Predecoding

- Many of these gates are redundant
  - Factor out common gates into predecoder
  - Saves area
  - Same path effort
Periphery

- Decoders
- Sense Amplifiers
- Input/Output Buffers
- Control / Timing Circuitry
Sense Amplifiers

\[ t_p = \frac{C \cdot \Delta V}{I_{av}} \]

make \( \Delta V \) as small as possible

Idea: Use Sense Amplifier

small transition

input

output
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 sense_clk after enough bitline swing
- Isolation transistors cut off large bitline capacitance
Periphery

- Decoders
- Sense Amplifiers
- Input/Output Buffers
- Control / Timing Circuitry
Column Circuitry

• Some circuitry is required for each column
  – Bitline conditioning
  – Column multiplexing
Bitline Conditioning

• Precharge bitlines high before reads

• Equalize bitlines to minimize voltage difference when using sense amplifiers
Twisted Bitlines

- Sense amplifiers also amplify noise
  - Coupling noise is severe in modern processes
  - Try to couple equally onto bit and bit_b
  - Done by *twisting* bitlines

![Diagram of twisted bitlines]
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 $2^k:1$ mux
  - No external decoder logic needed
Single Pass-Gate Mux

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

- Multiported memories
- CAM Memories
- Serial Access, Queues
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 processor must read two sources or write a result on some cycles
  – Pipelined processor must read two sources and write a third result each cycle
  – Superscalar processor 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
- Multiported SRAM isolates reads from state node
- Single-ended design minimizes number of bitlines
Memory configurations

- Multiported memories
- CAM Memories
- Serial Access, Queues
Contents-Addressable Memory

- I/O Buffers
- Address Decoder
- Data (64 bits)
- Comparand
- Mask
- CAM Array
  - $2^9$ words 3 64 bits
- R/W Address (9 bits)
- Control Logic
- Commands
- Priority Encoder
- $2^9$ Validity Bits
Memory configurations

- Multiported memories
- CAM Memories
- Serial Access, Queues
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)
  – Queues (FIFO, LIFO)
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 read/write pointers to RAM rather than data
  - Initialize read address to first entry, write to last
  - Increment address on each cycle
Tapped Delay Line

• 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
Queues

- **Queues** allow data to be read and written at different rates.
- Read and write each use their own clock, data
- Queue indicates whether it is full or empty
- Build with SRAM and read/write counters (pointers)
FIFO, LIFO Queues

• *First In First Out* (FIFO)
  – Initialize read and write pointers to first element
  – Queue is EMPTY
  – On write, increment write pointer
  – If write almost catches read, Queue is FULL
  – On read, increment read pointer

• *Last In First Out* (LIFO)
  – Also called a *stack*
  – Use a single *stack pointer* for read and write
Other considerations

- Leakage control
- Redundancy
- Flash Memories
Suppressing Leakage in SRAM

- Inserting Extra Resistance
- Reducing the supply voltage
Other considerations

- Leakage control
- Redundancy
- Flash Memories
Redundancy

Redundant rows
Redundant columns

Memory Array

Row Decoder
Column Decoder

Row Address
Column Address

Fuse Bank

NCD - Master MIRI
Error-Correcting Codes

Example: Hamming Codes

$P_1 P_2 B_3 P_4 B_5 B_6 B_7$

with

$P_1 \oplus B_3 \oplus B_5 \oplus B_7 = 0$

$P_2 \oplus B_3 \oplus B_6 \oplus B_7 = 0$

$P_4 \oplus B_5 \oplus B_6 \oplus B_7 = 0$

e.g. $B_3$ Wrong

1

1 = 3

0
Redundancy and Error Correction

![Graph showing the relationship between percent yield and average number of failing cells per chip for ECC only, redundancy only, and redundancy and ECC.]
Other considerations

- Leakage control
- Redundancy
- Flash Memories
Flash EEPROM

Many other options …
Cross-sections of NVM cells

Flash

EPROM

Courtesy Intel
Basic Operations in a NOR Flash Memory—Erase
Basic Operations in a NOR Flash Memory—Write
Basic Operations in a NOR Flash Memory—Read
Conclusions

Memory Structure:

- **Row Decoder**
  - \( A_K \)
  - \( A_{K11} \)
  - \( A_{L21} \)

- **Bit line**
  - \( 2^L2^K \)

- **Storage cell**

- **Word line**
  - \( M.2^K \)

- **Sense amplifiers / Drivers**

- **Column decoder**
  - \( A_0 \)
  - \( A_{K21} \)

- **Input-Output**
  - \( (M \text{ bits}) \)

- **Amplify swing to rail-to-rail amplitude**

- **Selects appropriate word**