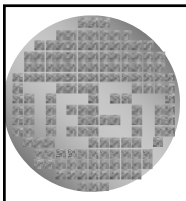


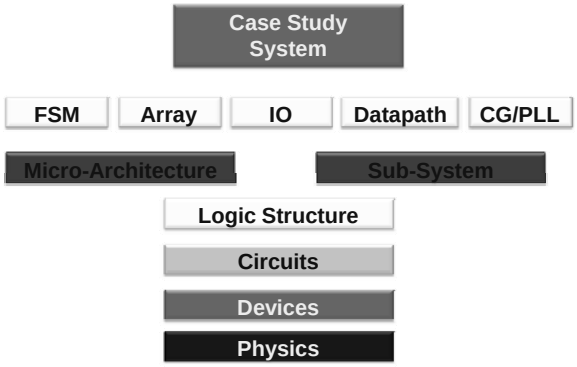
VLSI Design

Tsung-Chu Huang

Department of Electronic Engineering
National Changhua University of Education
Email: tch@cc.ncue.edu.tw

2011/12/19

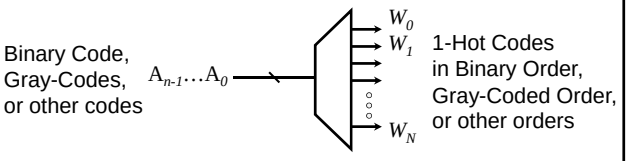
Reviews on Syllabus



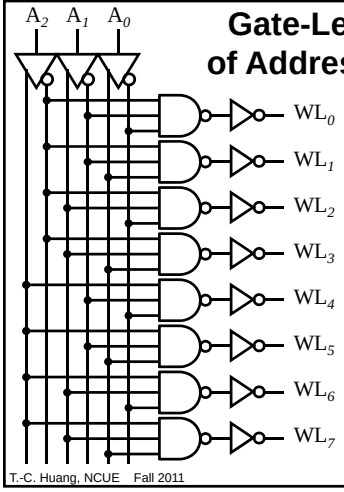
Classification of Arrays

- **In terms of LEVELs:**
  1. **Semi-customed transistor gates**
     - **"Gate Array", Sea-of-Gates (SoG)**
  2. **Logic Switches of X-Y cross-interconnections**
     - **Programmable Logic Array (PLA)**
  3. **Logic Gates**
     - **PAL, GAL, FPGA**
  4. **Memory Cells**
     - **Memory Array**
- **In terms of Location:**
  1. **Addressable**
  2. **Non-Addressable**

Function of *n-to-N* Address Decoders

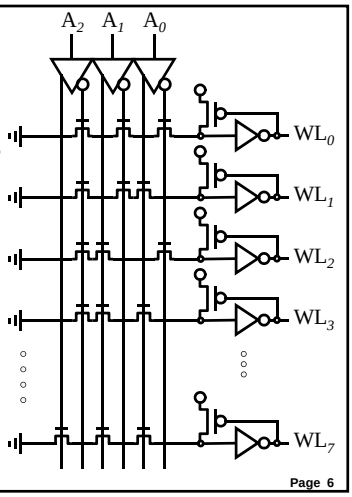


Gate-Level Design of Address Decoders



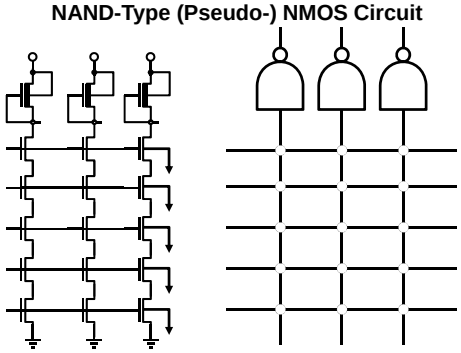
- The NAND can be substituted by NOR gates
- The gates can be CMOS cells in cell-based design, or diffusion stripes in transistor-level design.

Pseudo-NMOS Transistor-Level Single-Diffusion Design of Address Decoders



5. NMOS-like Structures

NAND-Type (Pseudo-) NMOS Circuit

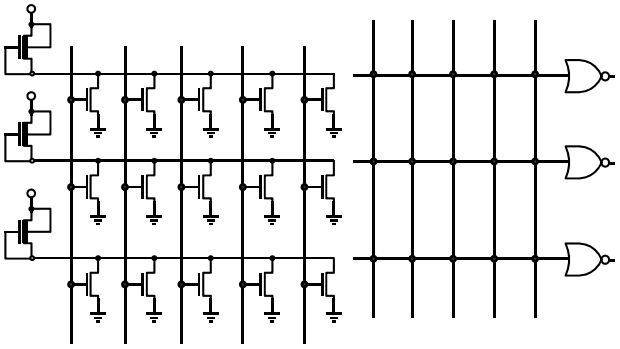


Body Effect!

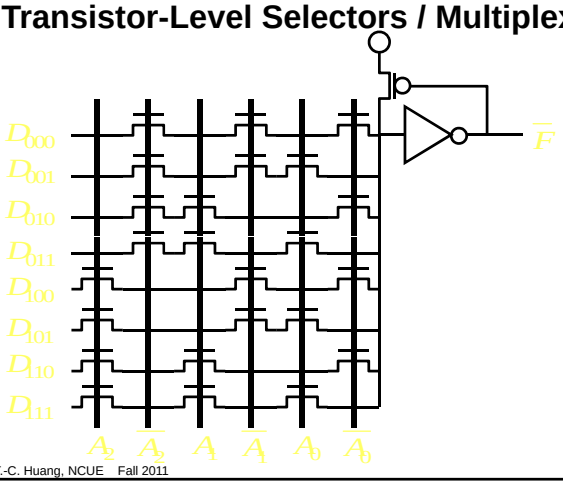
Compact, low-power but slower.

5. NMOS-like Logic Structures

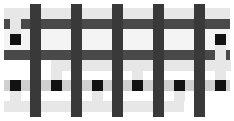
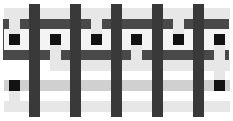
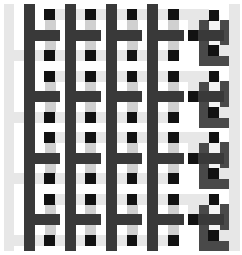
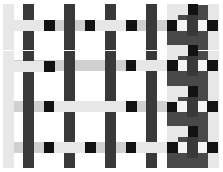
NOR-Type (Pseudo-) NMOS Circuit



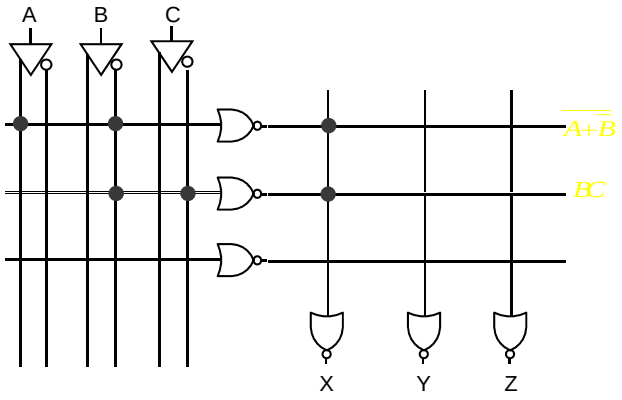
Transistor-Level Selectors / Multiplexers



5. Compaction, Leakages & Speed

	NOR Type	NAND Type
CMOS		
NMOS-like		 + Low-area + Low-power + Low-cost - Low-speed

Programmable Logic Array (PLA)



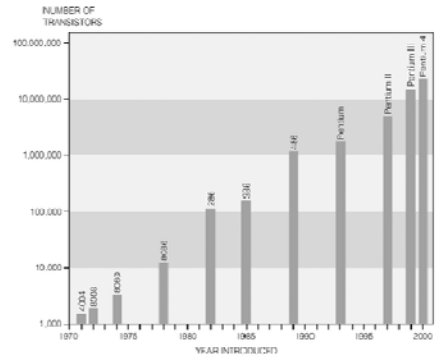
Classification of Memory Array

1. Development of Memory Industry
2. Taxonomy of Memory
3. Standardization
4. CCD
5. SRAM
6. DRAM
7. ROM
8. CAM
9. Memory Hierarchy
10. Timing Diagram
11. Factors Lowing Down a Memory System

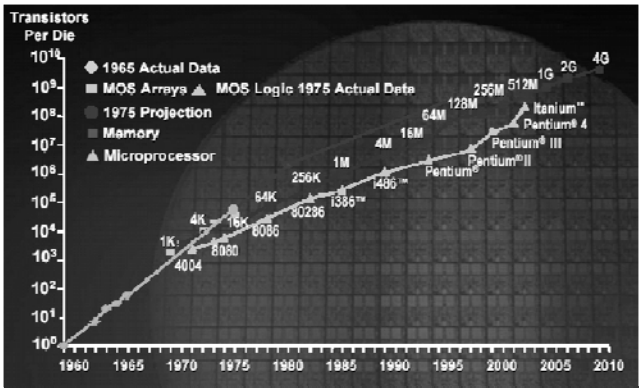
Classification of Memory Array

1. Development of Memory Industry
2. Taxonomy of Memory
3. Standardization
4. CCD
5. SRAM
6. DRAM
7. ROM
8. CAM
9. Memory Hierarchy
10. Timing Diagram
11. Factors Lowing Down a Memory System

Moore's Law on Transistor Counts



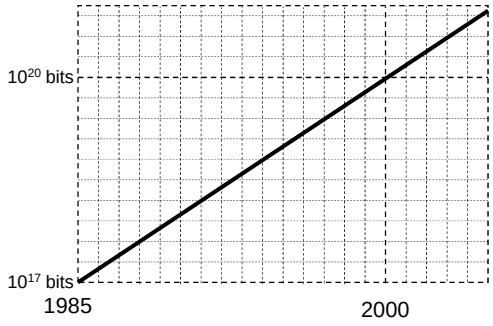
Moore's Law on Transistor Counts



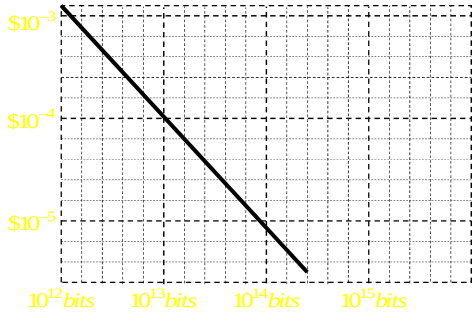
Units of Memory

Name	Abbreviation	Exact Number of Bytes	Approximate Number of Bytes
Byte	B	1	1
Kilobyte	kB	1,024 bytes	1 thousand
Megabyte	MB	1,024 kilobytes	1 million
Gigabyte	GB	1,024 megabytes	1 billion
Terabyte	TB	1,024 gigabytes	1 trillion
Petabyte	PB	1,024 terabytes	1 quadrillion

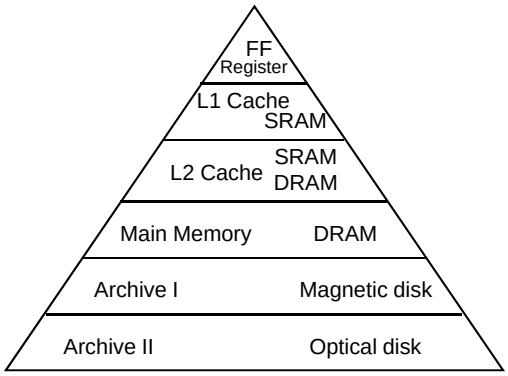
Moore's Law on Memory



MOS Memory Learning Curve



Typical Storage Hierarchy

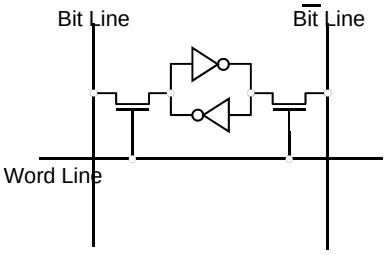


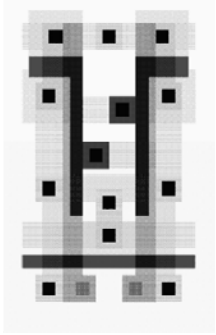
Classification

1. **Random Access Memory (RAM)**
   a. **R/W (RAM, Volatile)**
   b. **ROM (Nonvolatile)**
      ① Mask ROM (MROM)
      ② Programmable ROM (PROM)
      ③ Erasable PROM (EPROM)
      ④ Electronically EPROM (EEPROM)
      ⑤ Flash ROM
2. **Serial Access Memory (SAM)**
   a. **Shift Register**
      ① Serial-In Parallel-Out (SIPO)
      ② Parallel-In Serial-Out (PISO)
   b. **Stack (LIFO) and Queue (FIFO)**
   c. **(Tape)**
3. **Content-Addressable Memory (CAM)**

Basic CCD

Basic Static RAM (SRAM)





SRAM

–High Speed

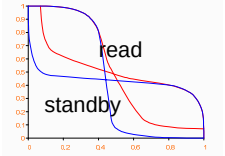
–High Cost

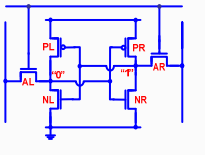
–Low Density

Six-transistor full CMOS

Hold Stability during READ

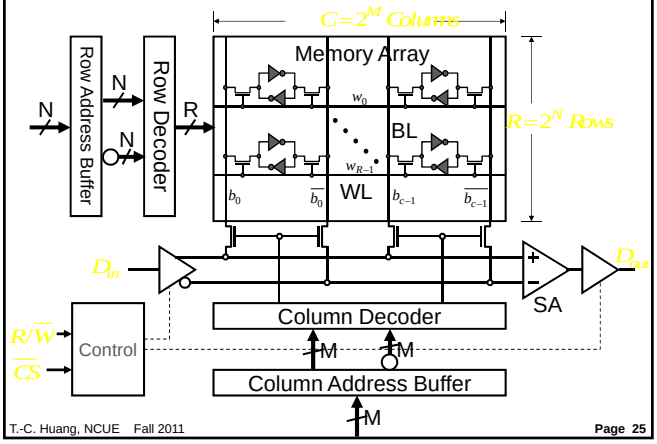
- Similar to Read Stability analysis without access transistor.
- PR must provide enough leakage to compensate for leakage in NMOS pull-down and access transistors.



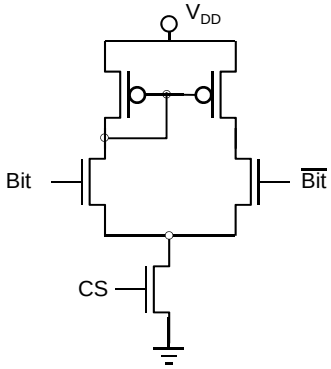


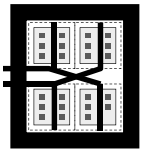
(butterfly curves)

Basic SRAM Architecture



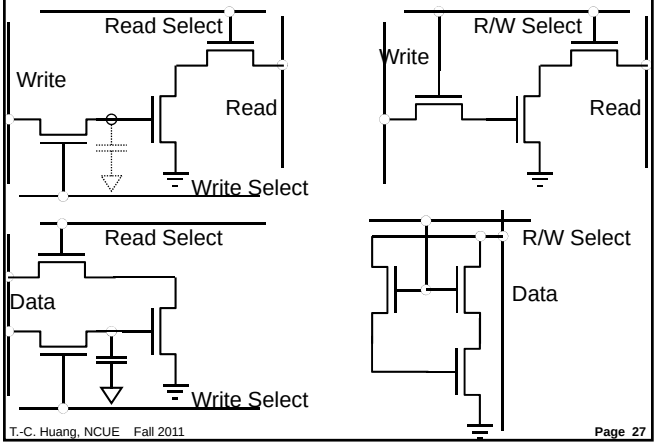
A Simple Sense Amplifier (SA)





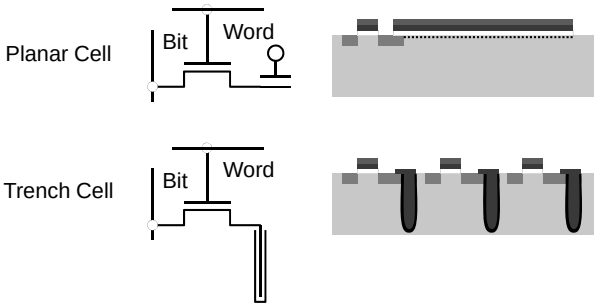
Typically, the SA must be sensitive enough to read about 10mV.

Historical Evolution of DRAM

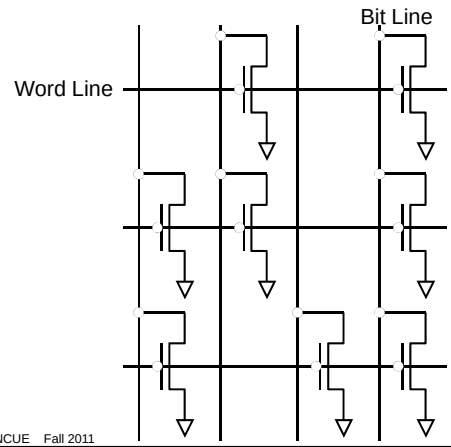


Historical Evolution of DRAM

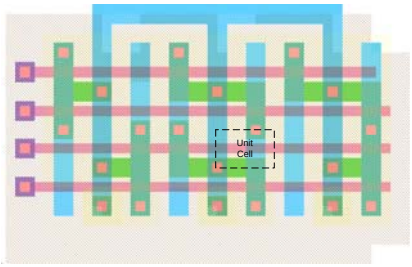
Basic Planar and Trench DRAM Cells



Basic ROM Architecture



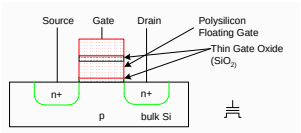
Basic ROM Architecture



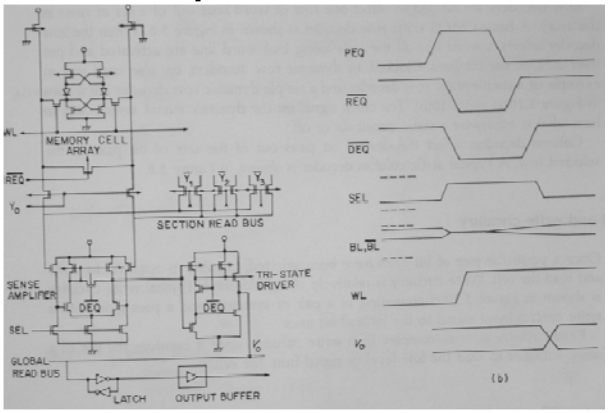
Flip cells to share the wires and contacts.

PROMs and EPROMs

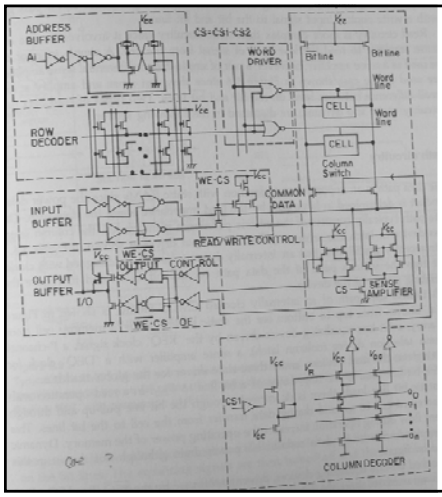
- Programmable ROMs
  - Build array with transistors at every site
  - Burn out fuses to disable unwanted transistors
- Electrically Programmable ROMs
  - Use floating gate to turn off unwanted transistors
  - EPROM, EEPROM, Flash



Amplification of SA

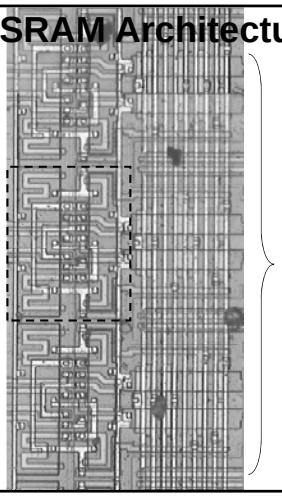


SRAM Circuitry

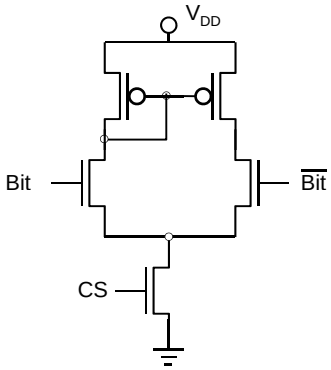


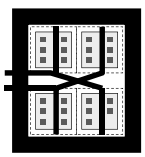
Basic SRAM Architecture

Row/Column Multiplexer And Buffer



A Simple Sense Amplifier (SA)

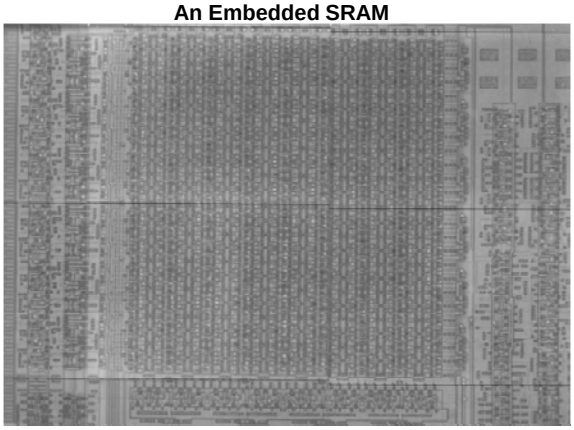




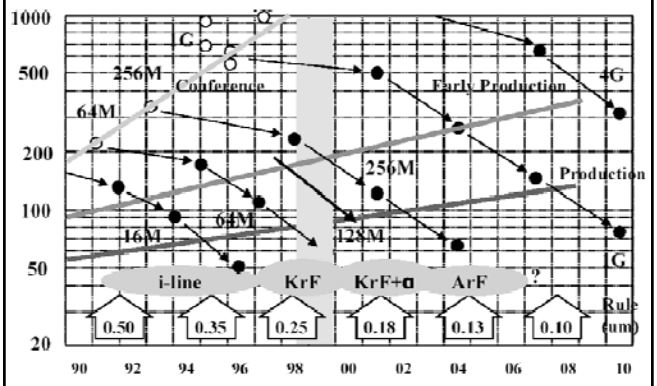
Typically, the SA must be sensitive enough to read about 10mV.

Basic SRAM Architecture

An Embedded SRAM



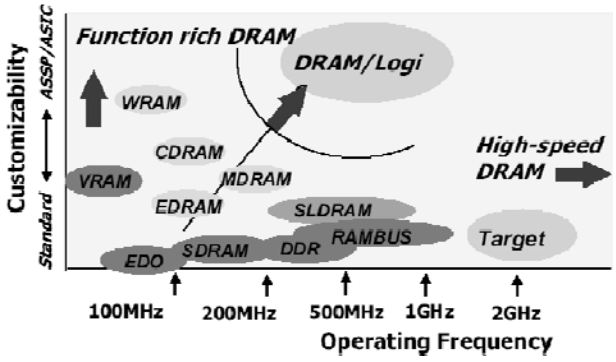
Standard DRAM Development



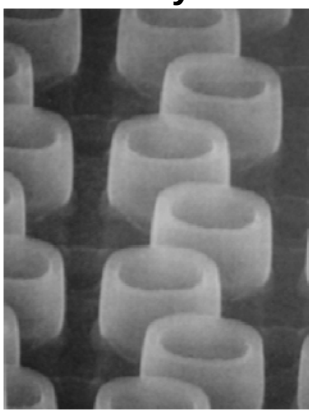
Related Techniques of DRAM

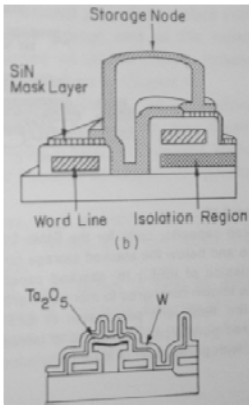
1. **EDO: Extended Data Output**
   + Fast Page Mode
2. **BEDO: Burst EDO (4 addr. per burst)**
   + Burst Mode: 4 Addresses per burst
3. **DDR: Double Data Rate Technique**
4. **SDRAM: Synchronous DRAM**
5. **RDRAM: RAMBus DRAM**
6. **VDRAM: Video RAM**

Op Frequency vs. Customizability



DRAM Cylindrical Stacked Cells





Basic Standards of Timing Diagrams

Basic I/O Signals

1. **CS: Chip Select**
2. **Adr.: Address**
3. **WE: Write Enable**
4. **OE: Output Enable**
5. **Di: Data-input**
6. **Do: Data-output**
7. **RAS: Row Address Strobe**
8. **CAS: Column Address Strobe**

Basic Standards of Timing Diagrams

Approaches

1. **JEDEC (Joint Electronics Device Engineering)**
2. **EIAJ**
3. **IEC**

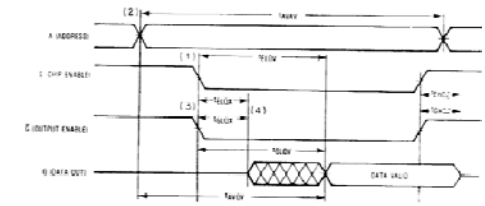
1. **JEDEC Standard Timing Symbol**
2. **Order notes in parentheses**
3. **Causal Arrows**
4. **Description**

Basic Standards of Timing Diagrams

Example

READ CYCLE

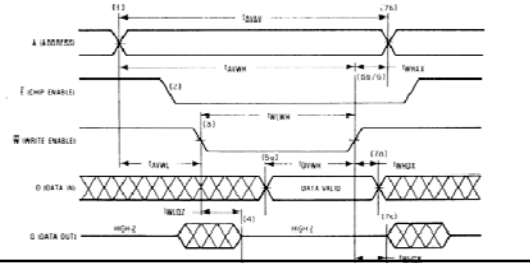
Parameter	Symbol		Min	Max	Min	Max	Min	Max
	Standard	Alternate						
Read Cycle Time	t_{AVAV}	t_{RC}	25	–	30	–	35	
Address Access Time	t_{AVQV}	t_{AA}	–	25	–	30	–	35
Chip Enable Access Time	t_{ELQV}	t_{ACS}	–	25	–	30	–	35
Output Enable Access Time	t_{GLQV}	t_{OE}	–	12	–	15	–	15
Output Hold from Address Change	t_{AXQX}	t_{OH}	5	–	5	–	5	
Chip Enable Low to Output Active	t_{ELQX}	t_{LZ}	5	–	7	–	10	–
Chip Enable High to Output High Z	t_{EHQZ}	t_{HZ}	0	10	0	12	0	15
Output Enable Low to Output Active	t_{GLQX}	t_{LZ}	5	–	5	–	10	–
Output Enable High to Output High Z	t_{GHQZ}	t_{HZ}	0	10	0	12	0	15



Write Cycle

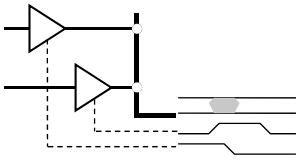
WRITE CYCLE

Parameter	Symbol		Min	Max	Min	Max	Min	Max	Unit
	Standard	Alternate							
Write Cycle Time	t_{AVAV}	t_{WC}	25		30		35		ns
Address Setup Time	t_{AVWL}	t_{AS}	0		0		0		ns
Address Valid to End of Write	t_{AVWH}	t_{AW}	20		25		30		ns
Write Pulse Width	t_{WLWH}	t_{WP}	20		25		30		ns
Data Valid to End of Write	t_{DVWH}	t_{DW}	10		12		15		ns
Data Hold Time	t_{WHDX}	t_{DH}	0		0		0		ns
Write Low to Output High-Z	t_{WLQZ}	t_{WZ}	0	10	0	12	0	15	ns
Write High to Output Active	t_{WHQX}	t_{OW}	5		8		10		ns
Write Recovery Time	t_{WHAX}	t_{WR}	0		0		0		ns



Factors Slowing Down Memory Sys.

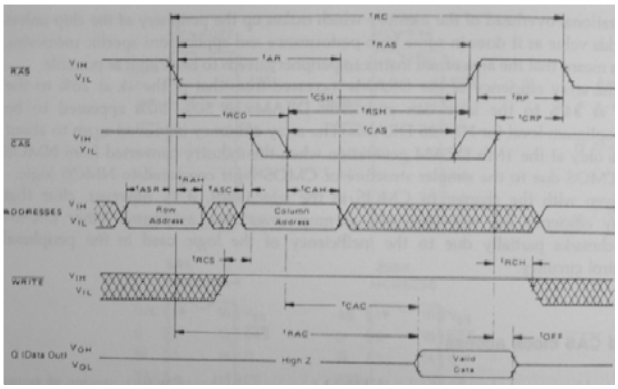
Bus Contention



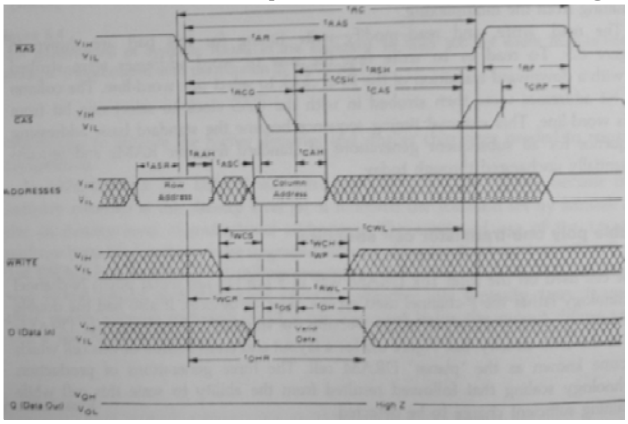
Ground Bounce

System Bandwidth

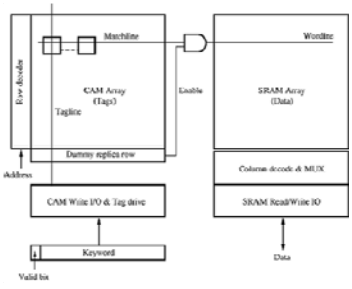
Address-Multiplexed Read Timing



Address-Multiplexed Write Timing

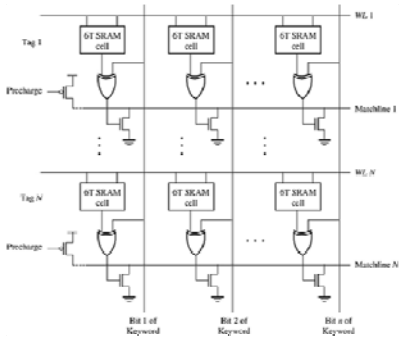


SRAM Array in CAM



- Invalid rows if not fully used
- Timing is critical since Matchlines are initially all prechaged
- Write data same as SRAM operations except Tagelines start low

CAM Lookup Array

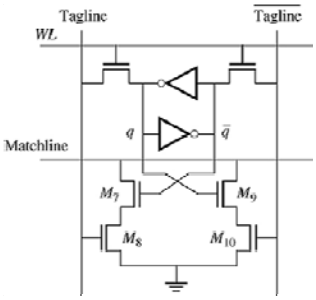


• э Diff: XOR=1
⇒NMOS on
Machi= 0

•∀ Same: XOR=0
⇒NMOS off
Machii=1
(precharged)

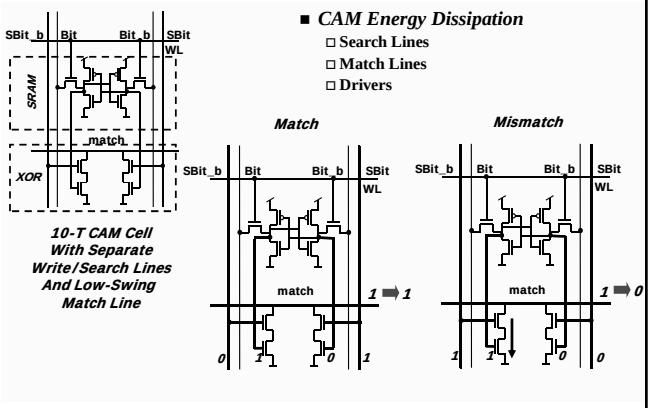
• Select only one or none row

CAM Cell



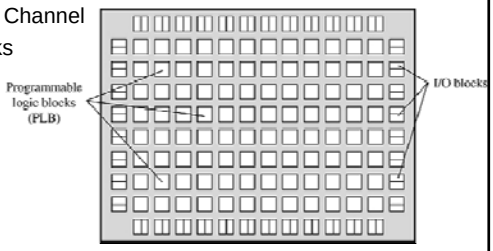
• M_7, M_8, M_9, M_{10}
⇒XOR
• Compare (q, q') with (Tagline, Tagline')
• 20% to 30% > Area of SRAM cell
• Taglines
  •Must start low
  •To avoid discharge in precharge phase

CAM Functionality

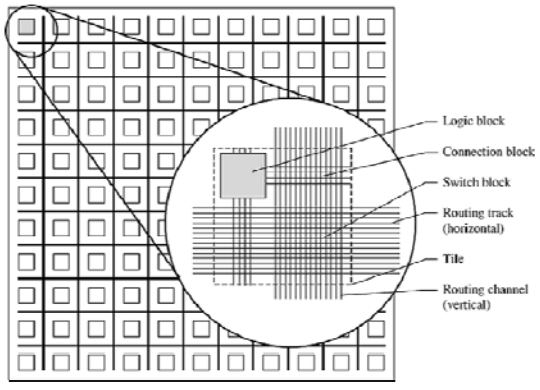


Field-Programmable Gate Array

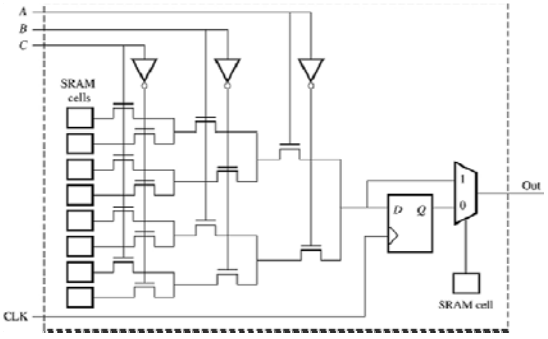
- Rapid Implementation
- Verify function but not timing
- Overall architecture
  - Routing Channel
  - I/O blocks
  - PLBs



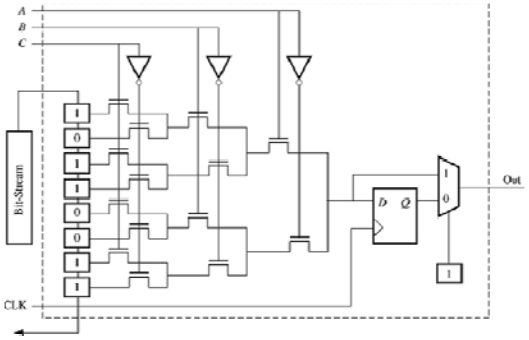
FPGA Architecture



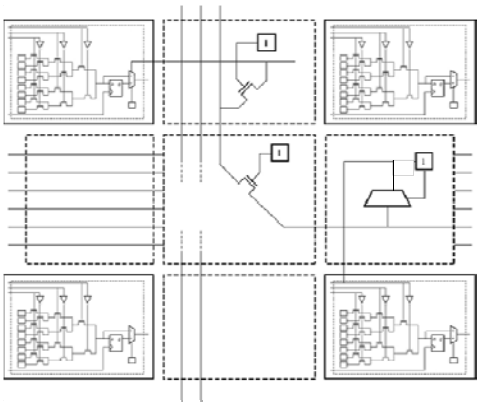
PLB Structure



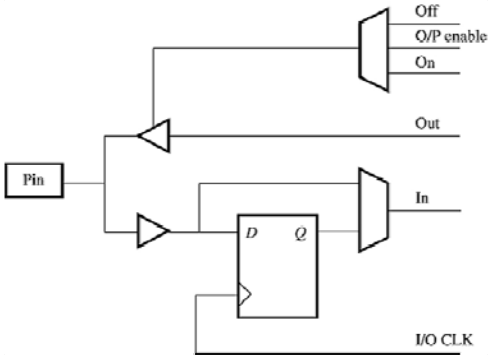
PLB Structure Example



Programmable Connections

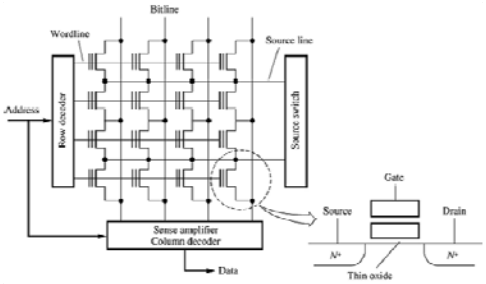


I/O Block

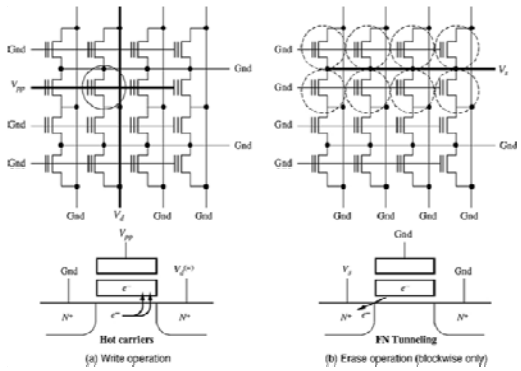


Flash Memory

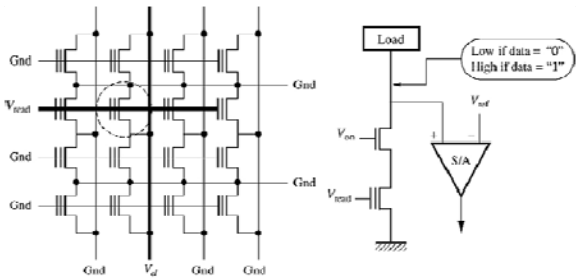
- NOR Structure
  - Write: Hot Carriers Injection
  - Erase: Fowler-Nordheim Tunneling



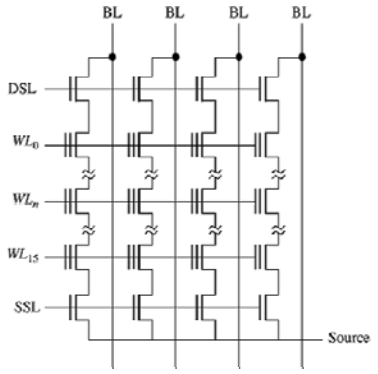
Write and Erase



Flash Read Operation



NAND Flash

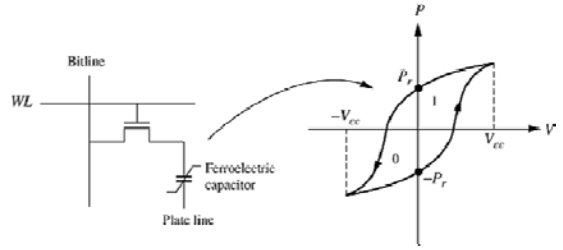


FRAM

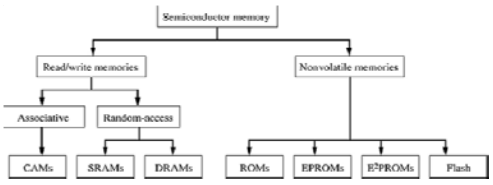
- Ferroelectric material
- Nonvolatile
- Destructive read-out
- Need write back similar to DRAM
- Drawback: (not semiconductor memory)
  - Cost, Speed, Size
- Advantages:
  - Less power
  - high density/reliability

FRAM

- Cap: Perovskite Crystal be polarized in 2 directions
  - Loigc-1: Vcc
  - Logic-0: –Vcc

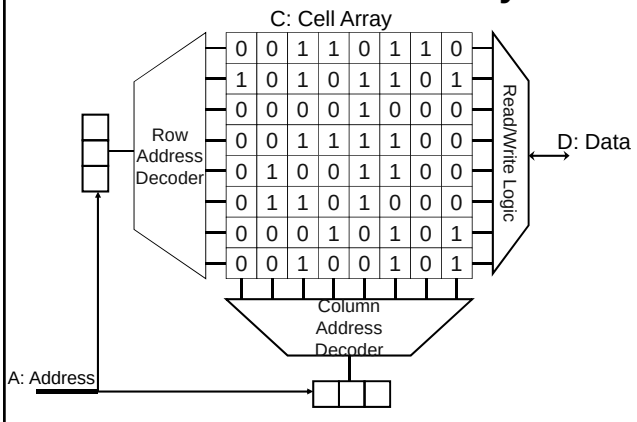


Summary of Semiconductor Memories



Memory	Prog type	Erase typ	Erase Res.	Endurance Cycle	Cell Size	Speed	Power
EPROM	HCI	UV	Full Mem	10^2	S	Fast	High
EEPROM	FN	FN	Bit/Byte	10^5 -10^6	L	Med	Med
Flash	HCI/FN	FN	Block	10^6	S	Fast	Low
FRAM	Polarize	Polarize	Bit	10^{10}-10^{12}	S	Fast	Low

Reduced Functional Memory Model



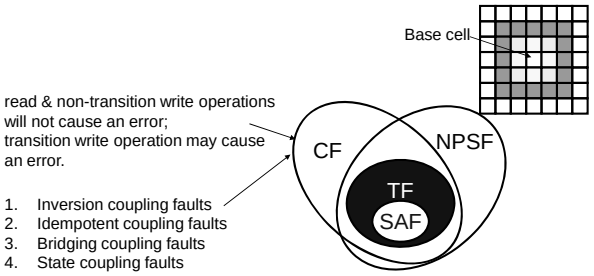
Reduction of Functional Faults

1. Stuck-At Faults
   - Cell stuck
   - Driver stuck
   - Read/write line stuck
   - Chip-select line stuck
   - Data line stuck
   - Open in data line
2. Transition Faults
   - Cell can be set to 0 but not to 1 or vice versa.
3. Coupling Faults
   - Short between data lines
   - Crosstalk between data lines
4. Neighborhood Pattern Sensitive Faults
   - Pattern sensitive interaction between cells
5. Address-decoder Faults
   - Address line stuck
   - Open in address line
   - Shorts between address lines
   - Open decoder
   - Wrong access
   - Multiple access

Reduced Functional Faults
Fault Models

1. Address Decoder Faults
2. Memory Cell Faults
   1. Single Cell Faults
      - Single-Cell Stuck-At Faults (SCSF)
      - Single-Cell Transition Faults (SCTF)
   2. Dual-Cell Faults
      - Coupling Faults (CF)
   3. Multiple-Cell Faults
      - Neighbor Cell Faults
      - Single Line Faults
      - Neighbor Line Faults

Fault Levels and Assumptions



Coupling Functional Faults
Coupling Relations

- Assume w(x, y) denotes a write y operation to a cell containing an x.
- <I/F> denotes a fault in a single cell where I describes the sensitizing input and F describes the fault value.
- <I_1, I_2, …, I_{n-1}; I_n/F> denotes a fault involving n cells where I_1, I_2, …, I_{n-1} describes conditions on the n-1 cells to sensitize the fault in cell n and In describes the condition for the fault to be sensitized in cell n.

Coupling Functional Faults
A. J. van de Goor, 1991

- Classified by Cell Count *k* and Affected Position Count *p*.
- *r*: the corresponding row; *c*: the corresponding column
- *n* = *R* x *C*, the total number of cells.

k	*p*			
	1	*row*	*column*	*n*
1	*k1p1*	*k1pr*	*k1pc*	*k1pn*
2	*k2p1*	*k2pr*	*k2pc*	*k2pn*
I	*kip1*	*kipr*	*kipc*	*kipn*
n	*knp1*	*knpr*	*knpc*	*knpn*

Brief Introduction to Memory Test

Fault Model:

0	0	1	1	0
1	0	1	0	1
0	1	0	0	0
0	0	1	1	0
0	1	0	0	0

Basics: $\Downarrow w0 \Downarrow r0 \Downarrow w1 \Downarrow r1 \Downarrow (w1r1w0r0)$

Detailed in the *Introduction to IC Test*.

March Test
Suk, 1981

- A march test consists of a finite sequence of march element that is a finite sequence of operations applied to every cell in memory before proceeding to the next cell.
- Notation of March Tests:

$\Uparrow(operations) \equiv$
```
for(a=0;a<n;a++){
    operations;
}
```
$\Downarrow(operations) \equiv$
```
for(a=n-1;a>=0;a--){
    operations;
}
```

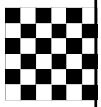
March Test

Suk, 1981

- Operations of March Tests:
  - *w0*: write zero to the cell,
  - *w1*: write one to the cell,
  - *r0*: read and detect whether the result is 0,
  - *r1*: read and detect whether the result is 1.
- Example: the simplest model (non-coupling SAF)
- For the non-coupling SAF, *2n wr*-operations are needed.

Traditional RAM Test

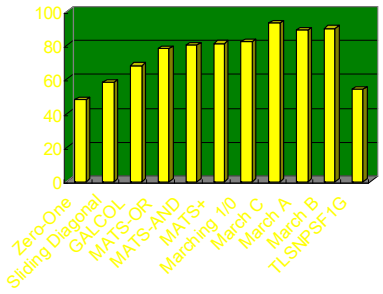
- Zero-One:
  - Not all TF, CF are detected, 4x2a length (a-bit address)
- Checkboard:
  - additionally detects shorts btw adjacent cells.
- GALPAT (Galloping pattern) and Walking 1/0
- Sliding Diagonal
- Butterfly



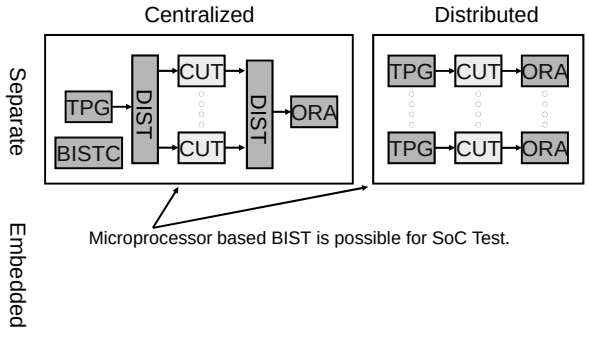
Multiple RAM Fault March Tests

- Test-US: MATS, MATS+
  - Modified Algo. Test Seq. for unlinked SAFs.
- Test-UT: Marching 1/0, MATS++
- Test-UCin: March X
- Test-UCid: March C- (C)
- Test-LCid: March A
- Test-LTin: March Y
- Test-LTCid: March B

Comparison on Fault Coverage



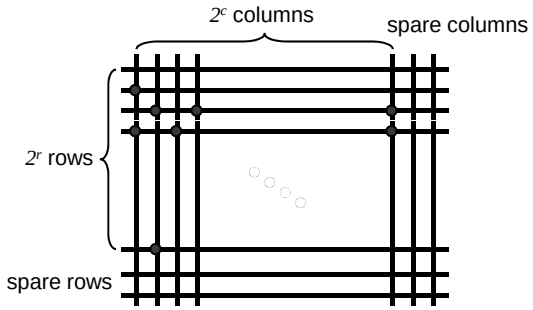
Memroy BIST Architecture



TPG: Test Pattern Generator, ORA: Output Result Analyzer
CUT: Circuit under Test, BISTC: BIST Controller

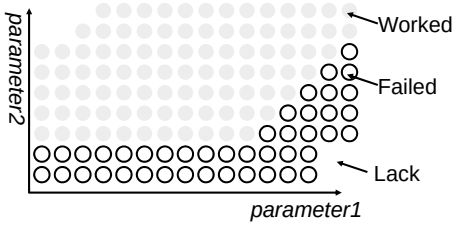
RAM Self-Repair

- To promote the product yield.



Schmoo Plot

• To show the parametric relations during parameter test.



Concurrent Test and Partitioning

• Partitioning is frequently used to reduce the power dissipation, time response and to provide possible concurrency for most circuits.

• Generic Memory Partitioning: