Introduction to CMOS VLSI Design

Semiconductor Memory

Harris and Weste, Chapter 12
25 October 2018

J. J. Nahas and P. M. Kogge

Modified from slides by Jay Brockman 2008

[Including slides from Harris & Weste, Ed 4,

Adapted from Mary Jane Irwin and Vijay Narananan, CSE Penn State adaptation of Rabaey's *Digital Integrated Circuits*, ©2002, J. Rabaey et al.]

Outline

- ❑ Memory features and comparisons
- ❑ Generic Memory Architecture
  - – Architecture Overview
  - – Row and Column Decoders
  - – Redundancy and Error Correction
- ❑ Static Random Access Memory (SRAM)
- ❑ Dynamic Random Access Memory (DRAM)
- ❑ Flash (EEPROM) Memory
- ❑ Other Memory Types

Memory Features and Comparisons

Memory Characteristics

- Read/Write Attributes
  - **Read-Only Memory** (ROM): Programmed at manufacture
    - Being phased out of use in favor of Flash
  - **Read-Write Memory**: Can change value dynamically
    - SRAM, DRAM
  - **Read-Mostly**: Can write, but much more slowly than read
    - EEPROM (Electrically Eraseable, Programable Read Only Memory) (pronounced “Double E Prom”)
    - Flash (A form of EEPROM)
- Volatility: sensitivity to losing power
  - **Volatile**: loses contents when power turned off
  - **Non-volatile**: does not lose contents

Memory Characteristics

- ❑ Addressability
  - – **Random-Access**: provide address to access a "word" of data
    - • No correlation between successive addresses
  - – **Block oriented**: read and write large blocks of data at a time
  - – **Content-addressable**: search memory for match against partial data
  - – **Serial Access:** e.g.. FIFO. Queue, stack
- ❑ **Wearout**
  - – Some memories have a limited number of write cycles before changing characteristics renders them unusable
- ❑ **Cell Size**
  - – Primary determinent of the physical size of the memory array

Lecture Focus

- ❑ The most used technologies
  - – **SRAM**
    - • Local memory
    - • Registers, Cache
  - – **DRAM**
    - • Main memory for computers and processors
  - – **Flash** and other EEPROMs
    - • Program and data storage
    - • Operational parameter storage
      - – Used in controllers for everything from a key fob to an automobile engine/transmission.

Memory Features

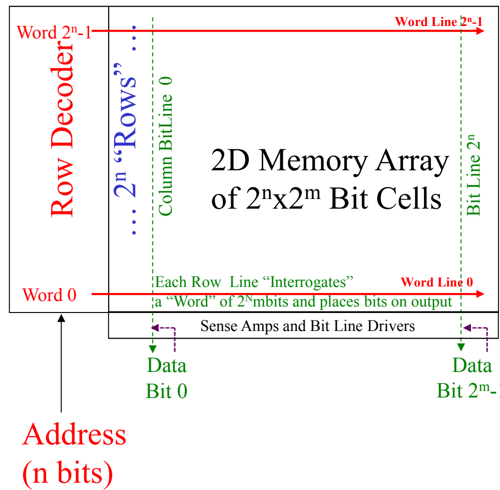
	SRAM	DRAM	Flash
Non-volitile	No	No	Yes
Cell Size (F^2)	25 to 40	6 to 8	4 to 5
Read Access Time	~ 0.1 ns	~ 10 ns	~ 10 ns
Write Time	~ 0.1 ns	~ 10 ns	N.A.
Erase Time	N.A.	N.A.	~ 10 ms
Program Time	N.A.	N.A.	~ 10 us
Wearout	N.A.	N.A.	10k to 100k erase/program cycles
Use in Computer/Processor Systems	Registers Cache (L1, L2, L3)	Main Memory	Program and Data Storage

- Note: Flash is process compatible with SRAM but DRAM is not.

Generic Memory Architecture

Westre and Harris, Chapter 12

Notional 2 Dimensional Memory (2^n Words of 2^m bits each)



Architecture

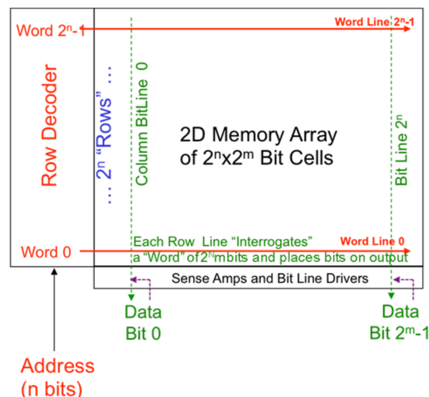
- Array of 2^n horizontal **word lines**
- and 2^m vertical **Bit lines**
- With a "memory bit" at each intersection

Key Logic Cells:

- **Bit Cell**: when its word line is active, place its current value on bit line
- **Row Decoder**: Convert 2^n bit address into 1 out of 2^n-1 lines
- **Sense Amps**: convert minute signal on a bit line into full-fledged digital signal
- **Bit Line Driver**: Drives input data onto bit line for storage in cell

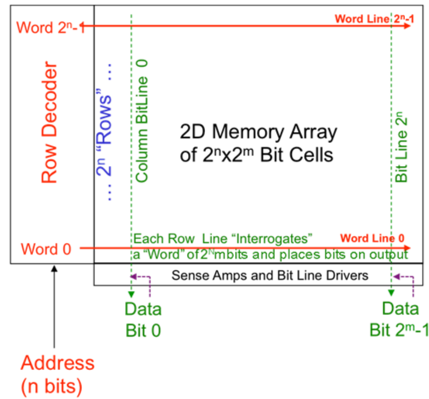
Read Access Signal Flow

- Transmit Row Address "UP" thru row decoder
- Decoder at each row determines if it matches transmitted row address
- If so, it raises its word line
- As word line signal goes from left to right, each cell it reaches places its data onto its bitline
- Only one cell per column places its data on bit line
- Signal travels down the bit line
- When signal arrives at Sense Amp, it is amplified and converted into full digital value



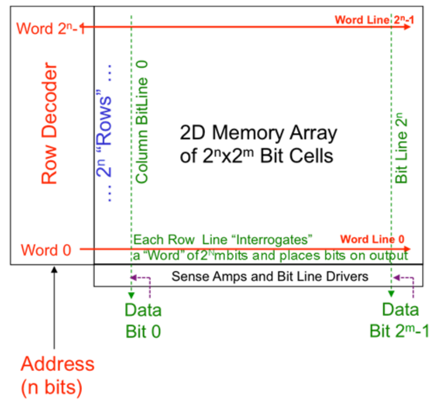
Write Access Signal Flow

- Transmit Row Address "UP" thru row decoder
- Decoder at each row determines if it matches transmitted row address
- If so, it raises its word line
- As word line signal goes from left to right, each cell it reaches receives its data from its bitline
- Only one cell per column receives data from its bit line



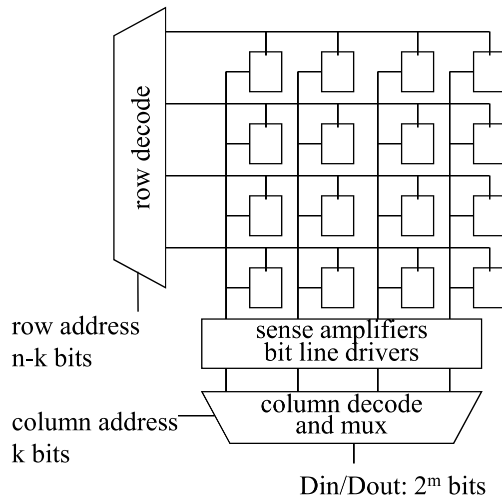
Questions

1. How many bits of storage?
2. What is logic function at each row decoder?
3. What memory parameters have major effect on access time, and why?
4. What memory parameters affect power?
5. What happens if memory is "tall and skinny"? i.e. n>>m
6. What happens if memory is "short and fat"? i.e. n<<m



More Accurate Array-Structured Memory Architecture

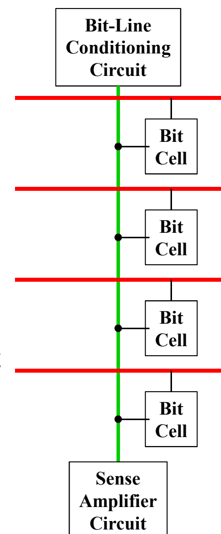
Assume we still want 2^n words of 2^m bits each,
but implement 2D array of (2^{n-k}) rows of $(2^{(k+m)})$ bits



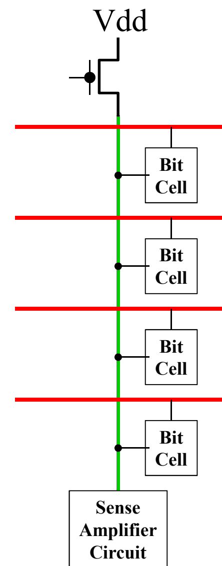
- Address of n bits, split into two parts
  - "Row" (n-k bits)
  - "Column" (k bits)
- n-k row address bits used to decode 1 of 2^{n-k} rows
- All cells on selected row sensed simultaneously
- Array reads out $(2^k)*(2^m)$ bits – called an ***open row***
- C Column address bits select one of 2^k words from open row

What About the Bit Lines

- Each bit line has multiple (2^{n-k}) cells on it
- Only one cell is "active" at a time
  - Remember: its "activated" by row line
- How do we drive one line from so many sources?
  - Key parameter: what is "area" per cell
    - # of transistors, size of transistors, …
  - Do we treat column as a "wire" or as part of a circuit?
  - If so, can we simplify by placing part of circuit "at top?"
- Option 1: Tri-state drivers?
- Option 2: Multi-legged NAND?
- Option 3: Multiplexor-like pass transistors?



A Common Solution: Active Pull-up



- Use “High Resistance” P type at top
- Turn on for access
- Bit line pulled high
- Bit cell must pull down if “1” not desired value

Questions:

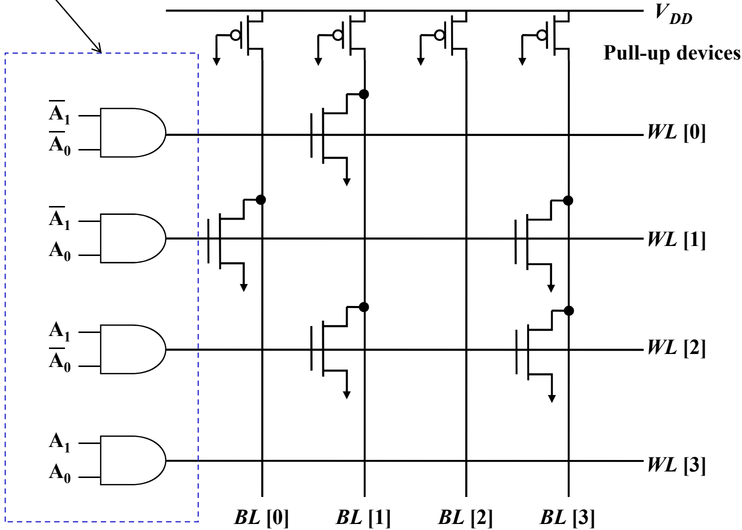
1. What width transistors do we want for the bit cell pull downs?
2. Many early designs have P-type always on. Why is that bad?
3. Why is P type “high resistance”?

Dynamic Sensing

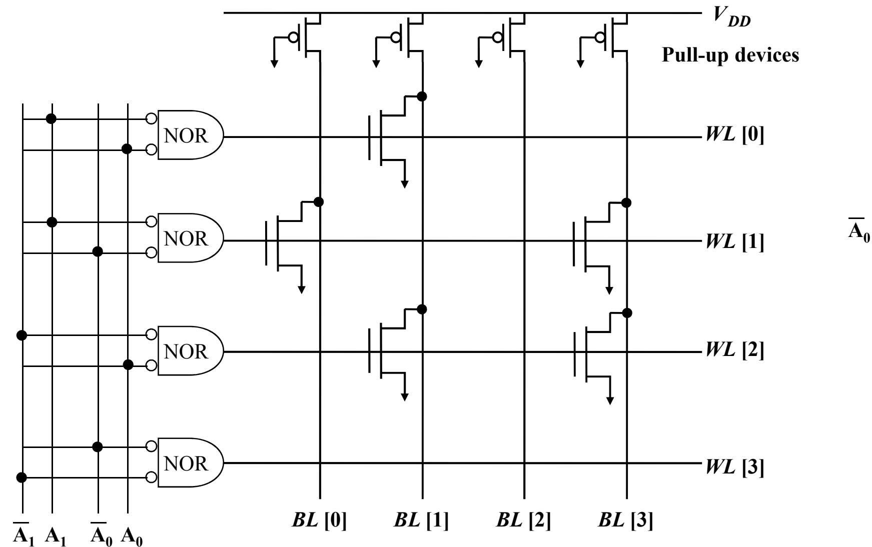
- With all P types always on, when n-types are on you have “short circuit” current – wastes power
- Observation: a bit line is a long wire
  - With significant capacitance
- What if we “pulse” p –types just before reading
  - Called **precharging** the bit lines
  - Bit lines all “charged” to a high voltage
- Now when we activate n-types
  - Only bits that represent “0” are pulled low
  - NO SHORT CIRCUIT CURRENT
- “Sensing:” no longer static voltage level
  - Charge on bit line decays with time
  - Need sampling **sense amplifier** to sense charge

Row and Column Decoders

Row Decoder: n bit address in, 2^n rows out



Standardizing Row Decoder



Row Decoders

Collection of 2^n complex logic gates
Organized in regular and dense fashion

(N)AND Decoder

$$WL_0 = A_0A_1A_2A_3A_4A_5A_6A_7A_8A_9$$

$$WL_{511} = \overline{A}_0\overline{A}_1\overline{A}_2\overline{A}_3\overline{A}_4\overline{A}_5\overline{A}_6\overline{A}_7\overline{A}_8A_9$$

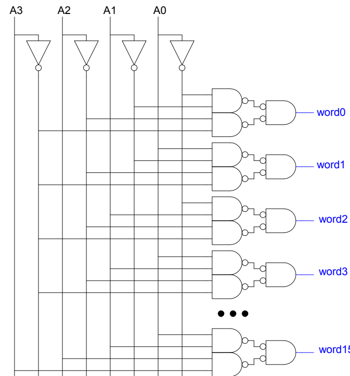
NOR Decoder

$$WL_0 = \overline{A_0 + A_1 + A_2 + A_3 + A_4 + A_5 + A_6 + A_7 + A_8 + A_9}$$

$$WL_{511} = \overline{\overline{A}_0 + \overline{A}_1 + \overline{A}_2 + \overline{A}_3 + \overline{A}_4 + \overline{A}_5 + \overline{A}_6 + \overline{A}_7 + \overline{A}_8 + A_9}$$

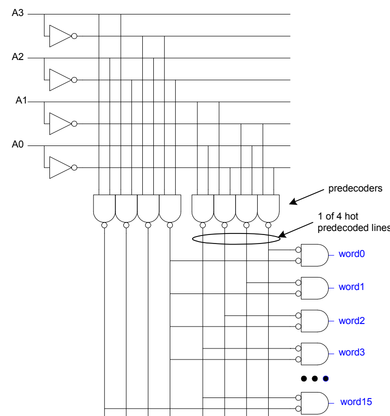
Large Decoders

- For n > 4, NAND gates become slow
  - Break large gates into multiple smaller gates

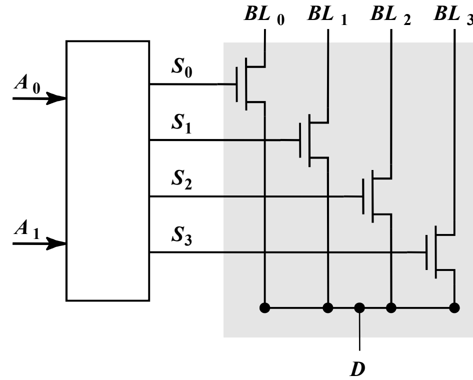


Predecoding

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



4-input pass-transistor based column decoder/multiplexor



Advantages: speed (t_{pd} does not add to overall memory access time

Only one extra transistor in signal path

Disadvantage: Large transistor count

Redundancy and Error Correction

Weste and Harris Section 12.8 and 11.7.2

Redundancy

- To improve yield, large memory arrays typically have redundant rows and columns.
- During testing, defective bits, rows, and columns are identified.
- Algorithm then determines which rows and/or columns to replace to avoid the defective bits.
- Laser programming or fuses are used to program the replacements into the chip

Error Correction

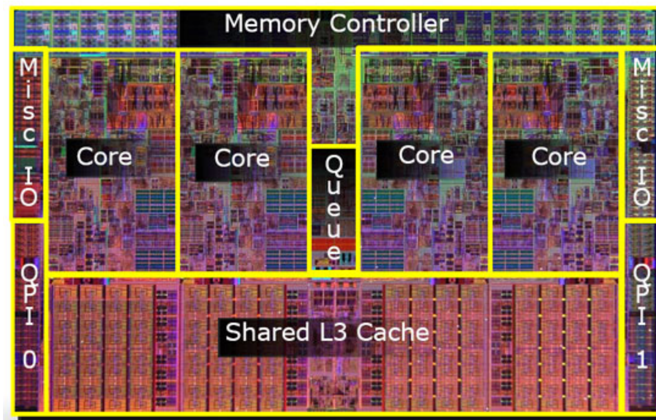
- Large memory arrays can also have soft errors due to signals being marginal.
  - The larger the number of bits, the larger the distribution of signals.
  - What is the error rate for a memory?
    - 1 error in 10^{12} reads is not acceptable
    - 1 error in 10^{16} reads is acceptable for most applications
      - A few errors per year at 1 GHz read rate.
    - 1 error in 10^{24} reads is needed for applications in financial institutions
- A 64 bit word plus nine parity bits (73 bits total) can be used to correct one bit error in the 64 bit word and detect two bit errors.
  - The use of 64 bit single error correction would decrease the error rate from 1 error in 10^{12} to 1 error in 10^{16} reads.

SRAM: Static Random Access Memory

Weste and Harris

Section 12.2

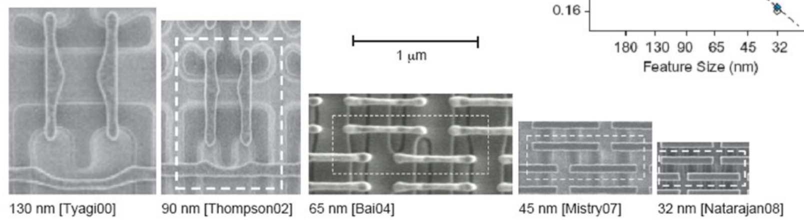
Intel Nehalem I7 Processor



- 2008, 45 nm CMOS
- SRAM used in registers, L1 and L2 Caches in Cores and Shared L3 Cache
- 90+% of chip is SRAM

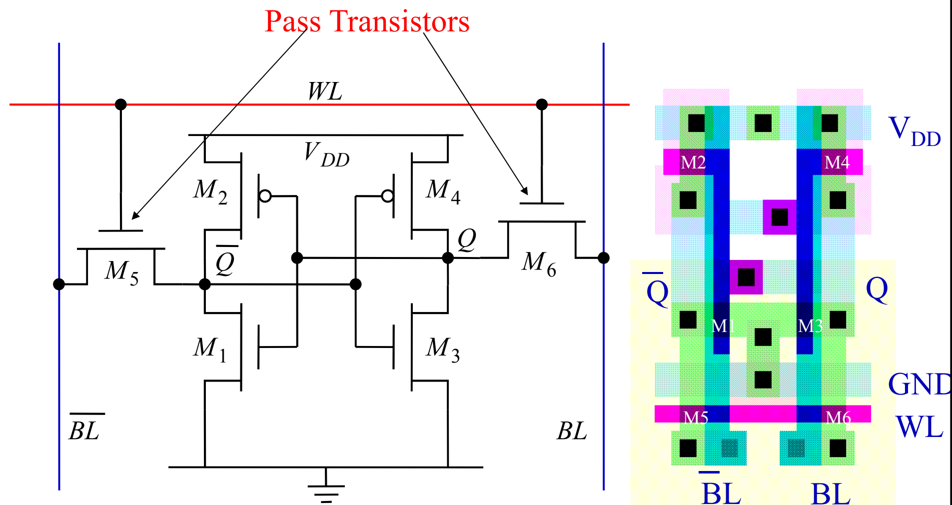
Commercial SRAMs

- Five generations of Intel SRAM cell micrographs
  - Transition to thin cell at 65 nm
  - Steady scaling of cell area



130 nm [Tyagi00] 90 nm [Thompson02] 65 nm [Bai04] 45 nm [Mistry07] 32 nm [Natarajan08]

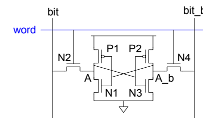
6-transistor CMOS SRAM Cell

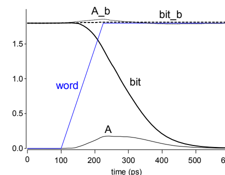


There are **2 bit lines** per column: True and Complement
Sense Amp looks for "1-0" or "0-1"

SRAM Read

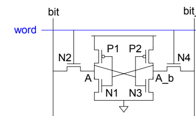
- Precharge both bitlines high
- Then turn on wordline
- One of the two bitlines will be pulled down by the cell
- Ex: A = 0, A_b = 1
  - bit discharges, bit_b stays high
  - But A bumps up slightly
- *Read stability*
  - A must not flip
  - N1 >> N2

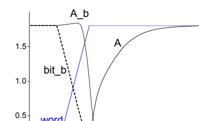




SRAM Write

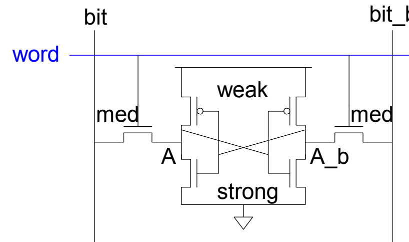
- Drive one bitline high, the other low
- Then turn on wordline
- Bitlines overpower cell with new value
- Ex: A = 0, A_b = 1, bit = 1, bit_b = 0
  - Force A_b low, then A rises high
- *Writability*
  - Must overpower feedback inverter
  - N2 >> P1





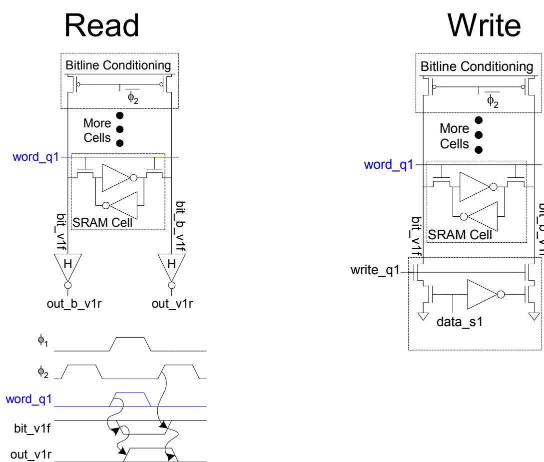
SRAM Sizing

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



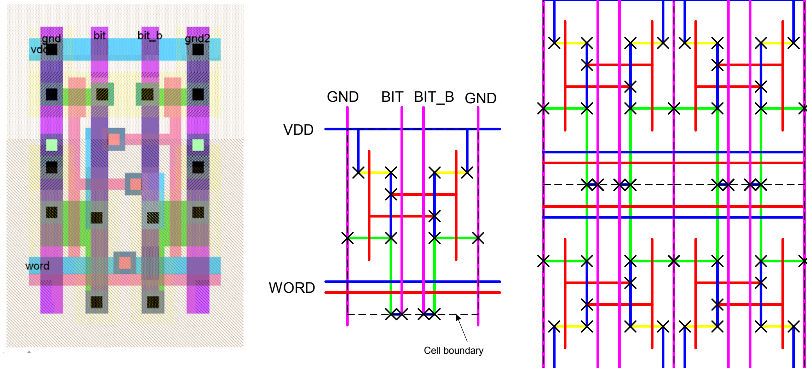
SRAM Column Example

Read Write



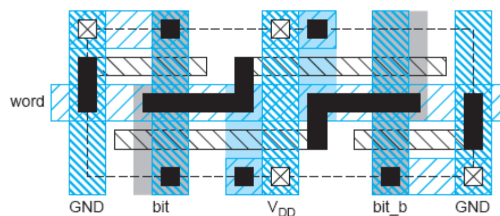
SRAM Layout

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



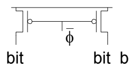
Thin Cell

- In nanometer CMOS
  - Avoid bends in polysilicon and diffusion
  - Orient all transistors in one direction
- *Lithographically friendly* or *thin cell* layout fixes this
  - Also reduces length and capacitance of bitlines

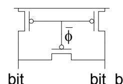


Bitline Conditioning

- 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 128 rows x 256 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 ΔV)

Differential Pair Amp

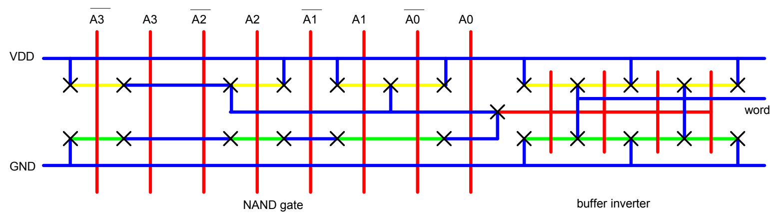
- Differential pair requires no clock
- But always dissipates static power

Clocked Sense Amp

- Clocked sense amp saves power
- Requires sense_clk after enough bitline swing
- Isolation transistors cut off large bitline capacitance

Decoder Layout

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

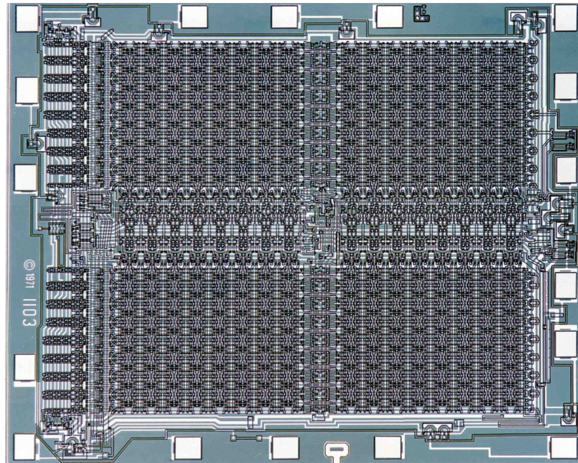


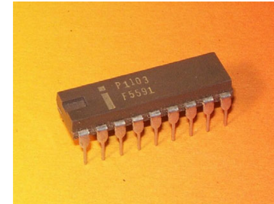
Pre-decoders help reduce size of decoders.

Dynamic RAM (DRAM)

Weste and Harris
Section 12.3

Intel 1103 1Kbit 1024 x 1 DRAM





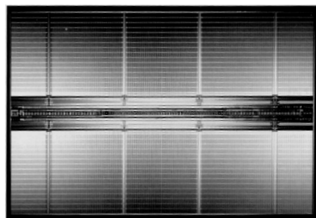
http://www.cpu-museum.com/Thumbs/Intel-P1103_t.jpg

- **Four arrays of 256 bits**
- **Each array is 16 X 16**
- **4 cell symmetry**
- **On-pitch word line driver and sense amplifiers.**

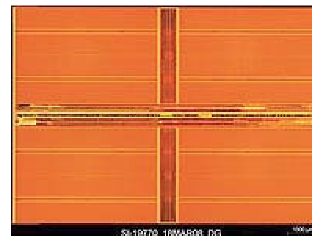
- First commercial DRAM. Introduced in 1970
- Replaced core memory on mainframe computers

http://cdn.dipity.com/uploads/events/8d74b8890633acbbdff01e3508c7bfe0_1M.png

Some More Modern DRAMs

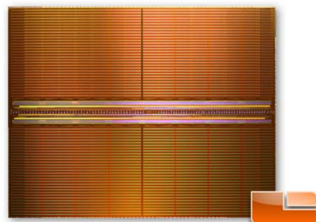


256Mb DRAM



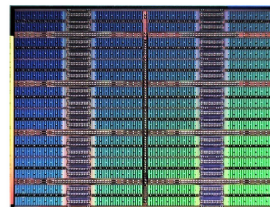
2008 DDR3 DRAM

http://www.eetasia.com/IMAGES/EEOL_2008APR24_STOR_NP_01a.jpg



42 nm 2Gb DRAM die

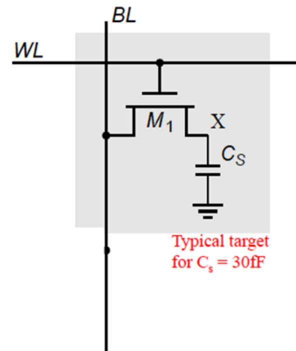
http://legitreviews.com/images/news/2010/42-nm-DRAM-die.jpg



Prototype ½ Gb DRAM for 3D Stacks

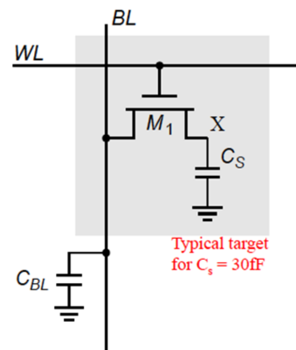
http://eda360insider.files.wordpress.com/2011/08/micron-hmc-memory-die-photo.jpg?w=571&h=442

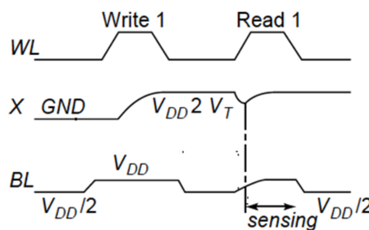
1-Transistor DRAM Cell



- C_s stores either a charge or no charge ("1" or "0")
- When word line active, transistor connects cap to bit line
- Charge on C_s distributed onto bit line
- This changes voltage on bit line

DRAM Operation





Write: C_S is charged or discharged by asserting WL and BL.

Read: First precharge BL to Vdd/2

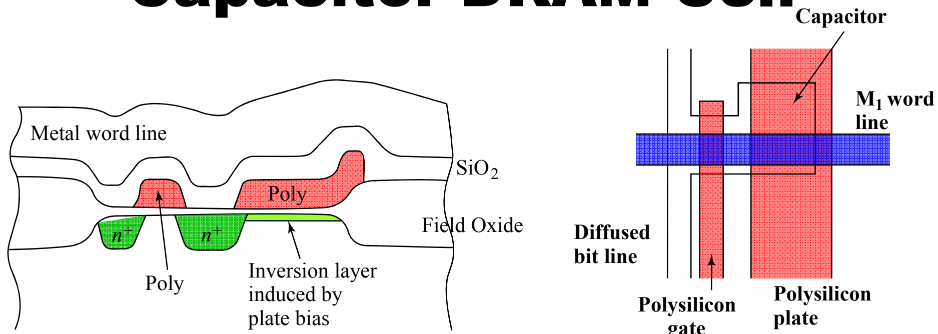
During Read, charge redistribution takes places between C_{BL} & C_S
Sense difference in voltage from Vdd/2
If C_S "charged" to >Vdd/2, BL pulled "up" a bit
If CS discharged, BL pulled "down" a bit

ΔV proportional to $C_S/(C_S + C_{BL})$

Ratio tends to be static across multiple generations

Voltage swing is small; typically around 250 mV.

Early Poly Diffused Capacitor DRAM Cell



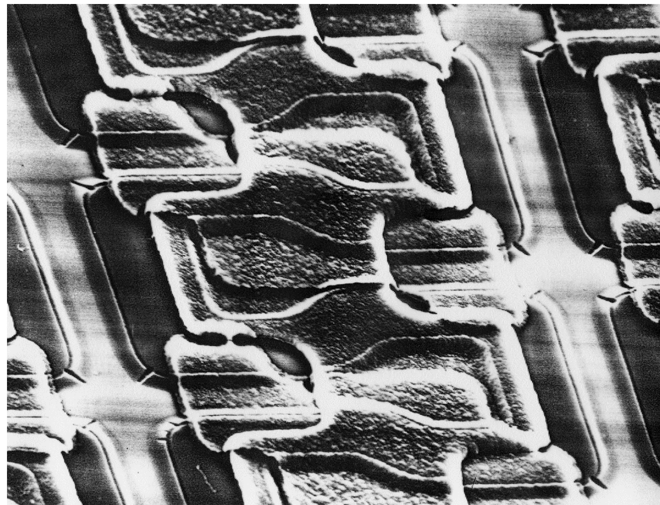
Cross-section

Layout

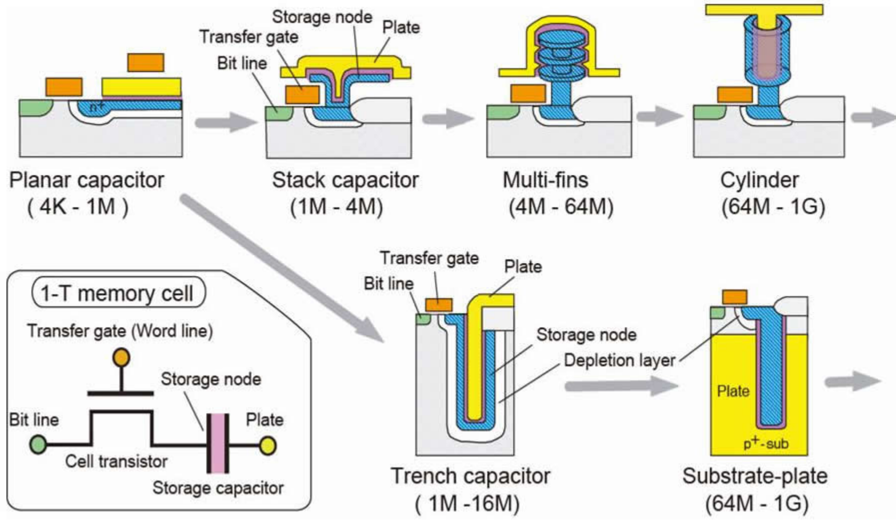
Uses Polysilicon-Diffusion Capacitance

Expensive in Area

SEM of poly-diffusion capacitor 1T-DRAM

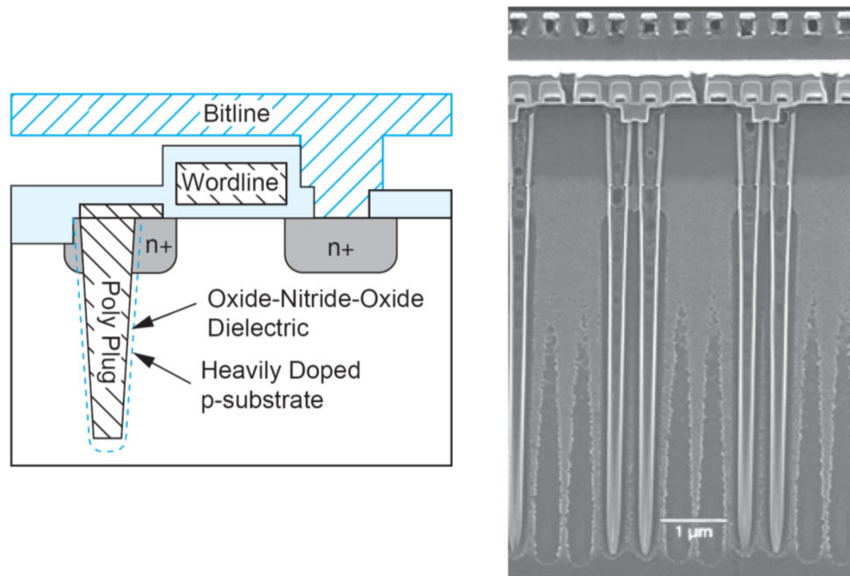


Capacitor Implementations

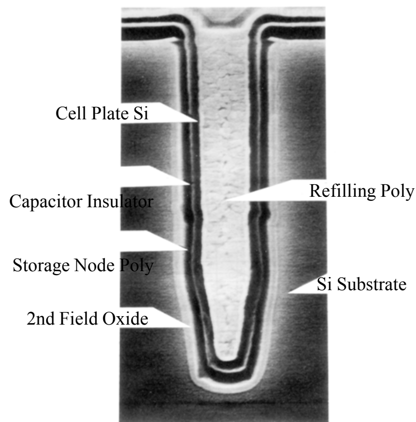


http://www.ieee.org/portal/cms_docs_sscs/sscs/08Winter/sunami-fig5.jpg

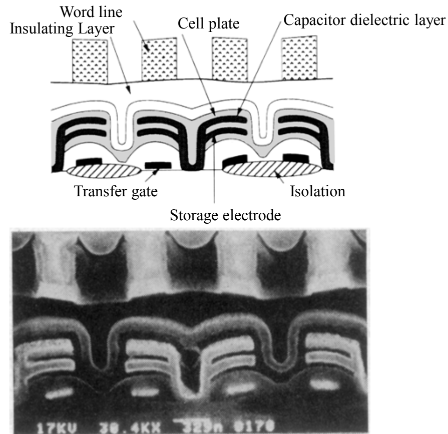
Trench Capacitor (Fig. 12.42)



Making the DRAM Capacitors



Trench Cell



Stacked-capacitor Cell

DRAM Folded Bitline Subarray

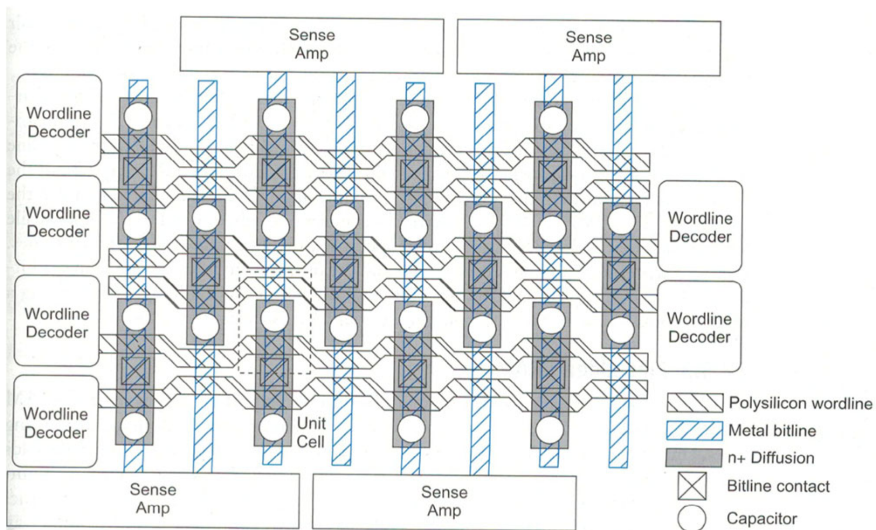


FIGURE 12.46 Layout of folded bitline subarray

DRAM Sense Amplifier

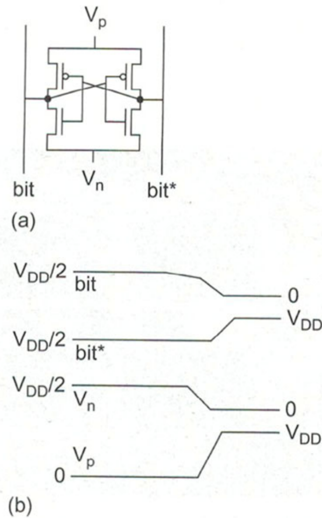


FIGURE 12.47 Sense amplifier

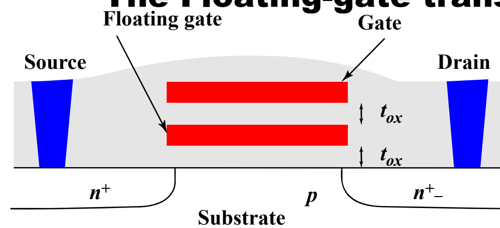
DRAM Refresh

- Memory capacitor can discharge by themselves in ~10-100 ms.
- A read operation senses the capacitor voltage and, using positive feedback, recharges the capacitor.
- All bit cells in a DRAM must be read periodically to refresh the capacitor voltage.
  - This periodic reading is called a refresh cycle.

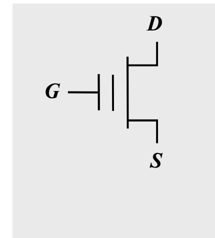
Flash Memory

Weste and Harris
Section 12.4.3

Non-Volatile Memories
The Floating-gate transistor (FAMOS)



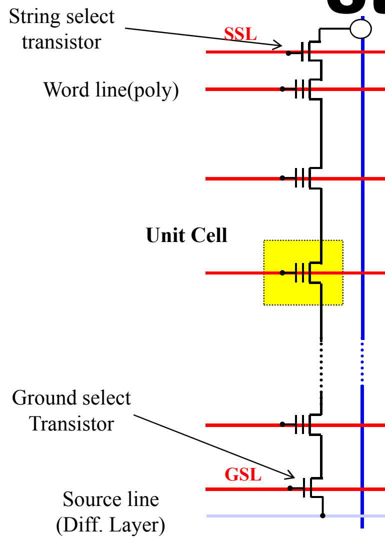
Device cross-section

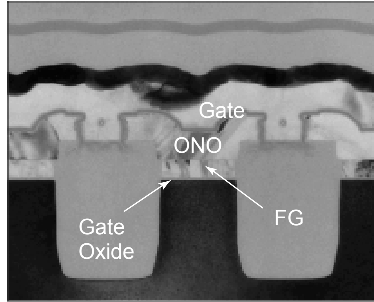


Schematic symbol

- Storage determined by charge on the floating gate
  - "0" = negative charge (extra electrons)
  - "1" = no charge
- Negative charge on floating gate "screens" normal gate, raising threshold
- Charge can take years to "leak off" once placed there
- **Multi Level** flash: different charge levels represent different values
  - We are "programming" Vt of the transistor

NAND Flash Memory "String"





- All wordlines other than the one to be read are >> "0" threshold, so they turn on.
- Wordline to be read has lower voltage:
  - If cell has "0" – no current
  - If cell has "1" – device is on

Picture courtesy of Toshiba

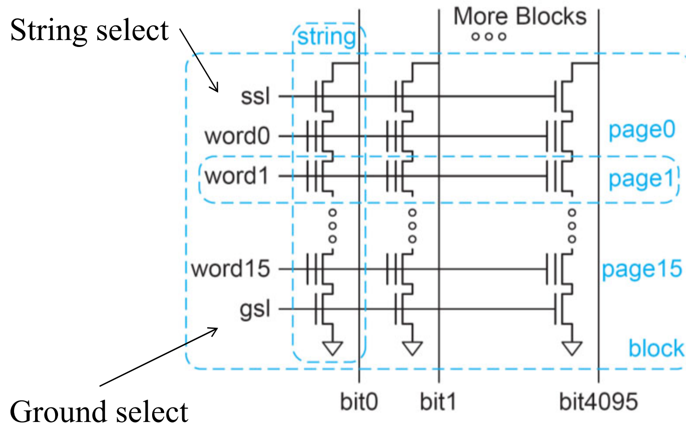


FIGURE 12.60 NAND Flash string

NAND Flash Memory

Select transistor

Word lines

Active area

STI

Bit line contact

Source line contact

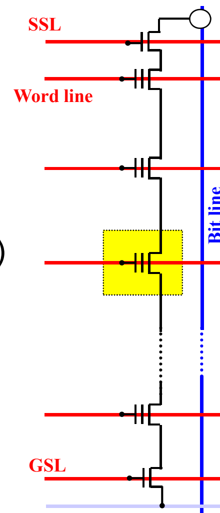
Sense Amp
Row Decoder
32 Gb Plane 0
Row Decoder
32 Gb Plane 1
Sense Amp
Peripheral Circuits

64 Gb (8GB) flash
- 2 independent panes
- 64K columns/pane
- Thus 64kbit page
- Each cell holds 4 bits
- Each string = 64 cells
- Each block has 256 pages
- Each pane has 2K blocks

Courtesy Toshiba

Reading Data

- ❑ Precharge bit lines
- ❑ SSL & GSL set high
- ❑ Set all word lines **but** desired page to high enough to turn transistors on, regardless of state
- ❑ Set **word line for desired page** high enough to turn on IF NO CHARGE (“1”) is present on floating gate
- ❑ Result depends on floating gate:
  - – If no charge (1), all transistors on & bit line discharged
  - – If negatively charged (0), bit line charge not disturbed



Question: Why do we have ***string select*** and ***ground select*** transistors?

Programming (Writing) Data

- Cell "programmed" by placing electrons on floating gate
- Charge moved to/from via tunneling to substrate
- Writing is done a page at a time
  - Substrate held to ground
  - Word line for *selected page* raised very high (e.g. 20V) *to trigger tunneling*
  - Word line for *all other pages* at intermediate level (10V) guaranteed to turn transistors on, *but not tunnel*
  - Desired bit values placed on bit lines
  - If a "0" on bit line, then electrons tunnel to floating gate occurs

SSL
Word line
Selected page's Word line
GSL
Bit line

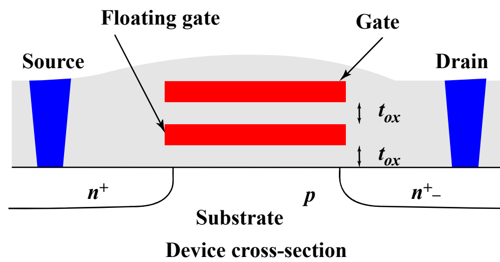
Note: we can ONLY PROGRAM ZEROs!!!

Block Erasure

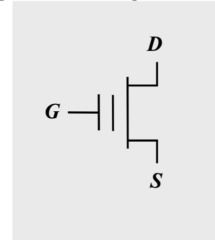
Floating gate
Gate
Source
Gnd
Drain
t_{ox}
t_{ox}
Electron tunneling
n^+
20V
p
n^+
Substrate

- If we can only write "0"s, how do we store "1"s?
- Answer: we "erase" all cells to 1 before writing and write only 0s
- **Erasing process**:
  - Set substrate very high (e.g. 20V)
  - Set all control gates to ground
  - Over time (ms), electrons on floating gates tunnel to substrate
- Cannot control substrate voltage of single transistors, so erase ***all cells in a block*** at the same time

Non-Volatile Memories
The Floating-gate transistor (FAMOS)



Device cross-section

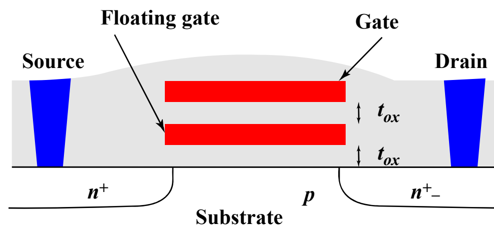


Schematic symbol

- **ERASE**: Raising substrate to high + voltage (e.g. 20V) with Gate at Ground, causes tunneling from floating gate to substrate, clearing floating gate of all charge
- **PROGRAM**: High + voltage on Control Gate, with substrate at ground, causes electrons to tunnel from substrate to floating gate, raising effective threshold of device, and representing a "0"
  - Different voltages can store different amounts of charge, changing threshold
- **READ** state by applying voltage (< "0" threshold) to control gate and seeing if current flows

Erase	Inhibit Erase	Program 0	Do Not Program	Inhibit Program
Gate 0 V, Substrate 20 V	Gate 20 V, Substrate 20 V	Gate 20 V, Source/Drain 0 V / 0 V, Substrate 0 V	Gate 20 V, Source/Drain 8 V / 8 V, Substrate 0 V	Gate 10 V, Source/Drain ? / ?, Substrate 0 V

Summary

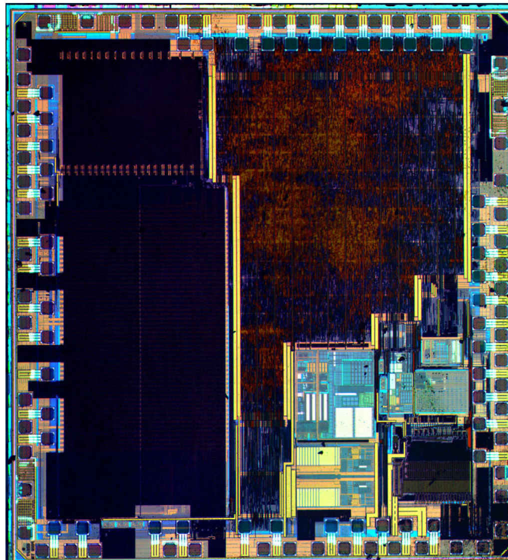


Erase	Inhibit Erase	Program 0	Do Not Program	Inhibit Program
Gate 0 V, Substrate 20 V	Gate 20 V, Substrate 20 V	Gate 20 V, Source/Drain 0 V / 0 V, Substrate 0 V	Gate 20 V, Source/Drain 8 V / 8 V, Substrate 0 V	Gate 10 V, Source/Drain ? / ?, Substrate 0 V

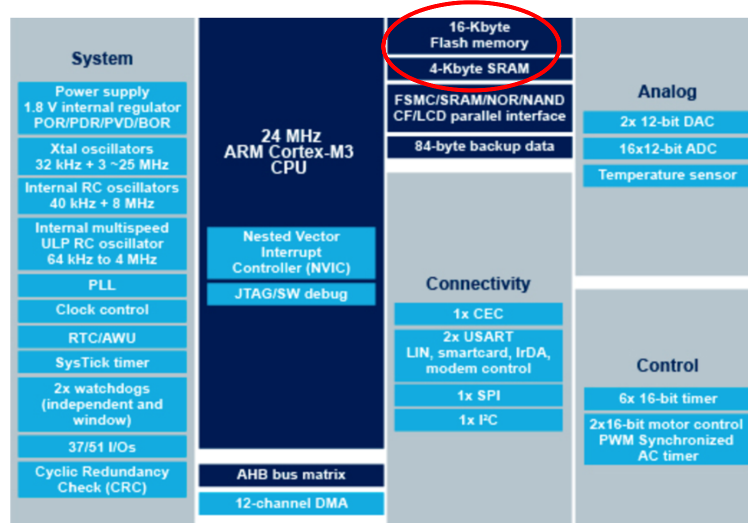
Microcontrollers

- An example of SRAM combined with Flash Memory
- Usually include many other functions

STM ARM Cortex-M3 MCU



STM ARM Cortex-M3 MCU



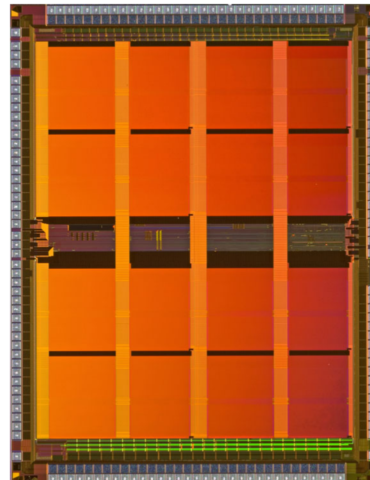
STM ARM Cortex MCUs

- Clock speed up to 400 MHz
- Flash memory up to 3 MB
- SRAM up to 1 MB

Other Semiconductor Memory Technologies

First Commercial MRAM

► 4 Mb Toggle MRAM
- 35ns symmetrical read/write
- Unlimited endurance
- Date retention >>10 Years
- 256Kx16bit organization
- 3.3V single power supply
- Fast SRAM pinout
- Consumer temperature range



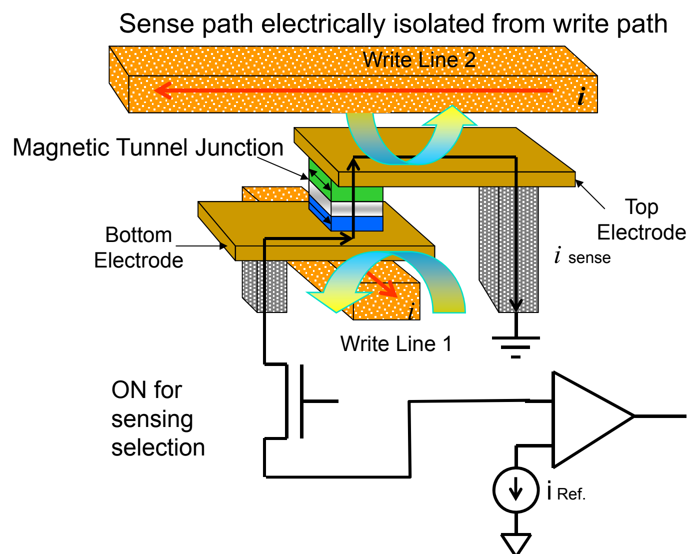
Developed by Motorola and Freescale Semiconductor
Spun-off as Everspin Technologies, Chandler, AZ

MRAM Attributes

- Non-Volatility
  - fast writing
  - no write endurance limitation
- Random Access
  - no refresh
  - no erase/program write sequence.
- Non-destructive read

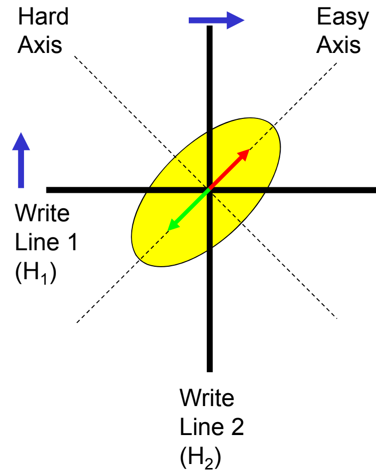
Toggle MRAM Bit Cell

Sense path electrically isolated from write path



Elements of Toggle Bit

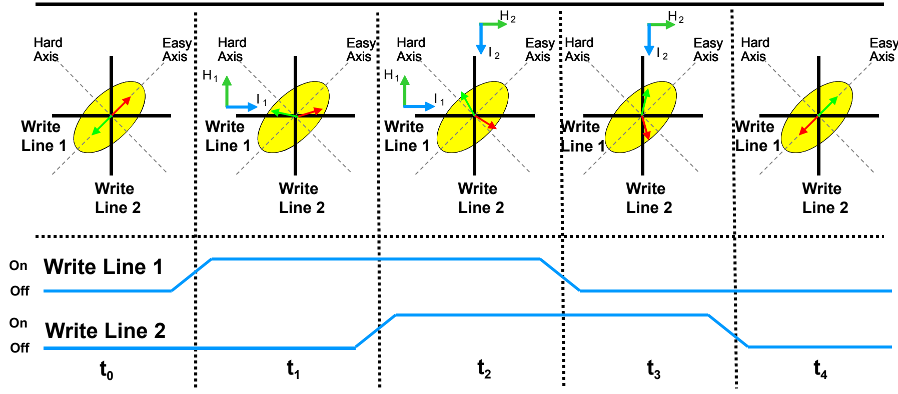
- Balanced SAF free-layer
- Bit oriented 45º to lines
- Unipolar currents
- Overlapping pulse sequence
- Pre-read / decision write



Joe Nahas 28 Feb 2007

CMOS VLSI Design
Slide 73

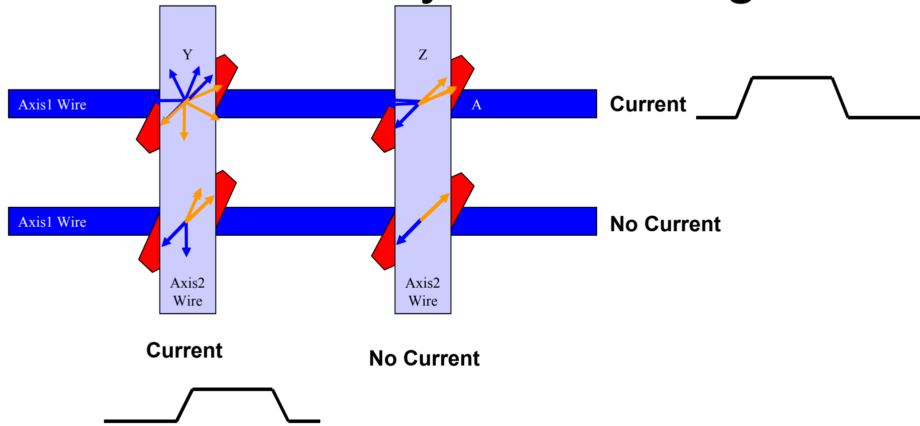
Toggle MRAM Switching Sequence



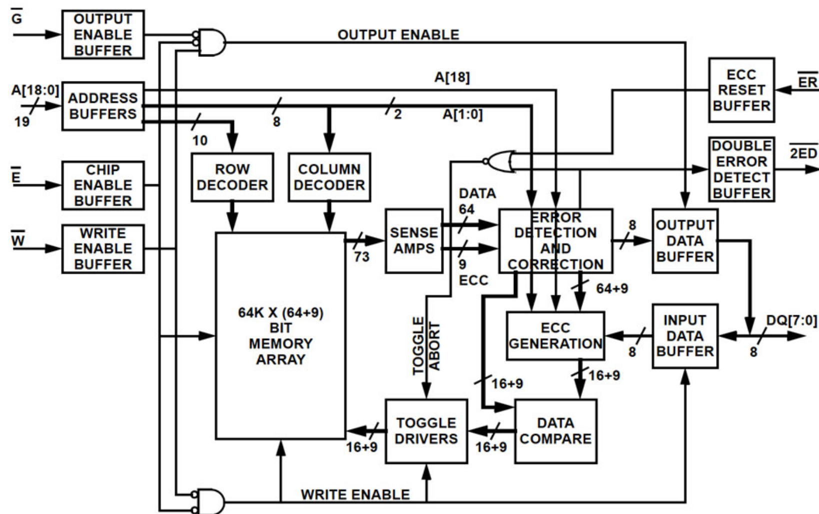
Joe Nahas 28 Feb 2007

CMOS VLSI Design
Slide 74

2 X 2 Array Addressing

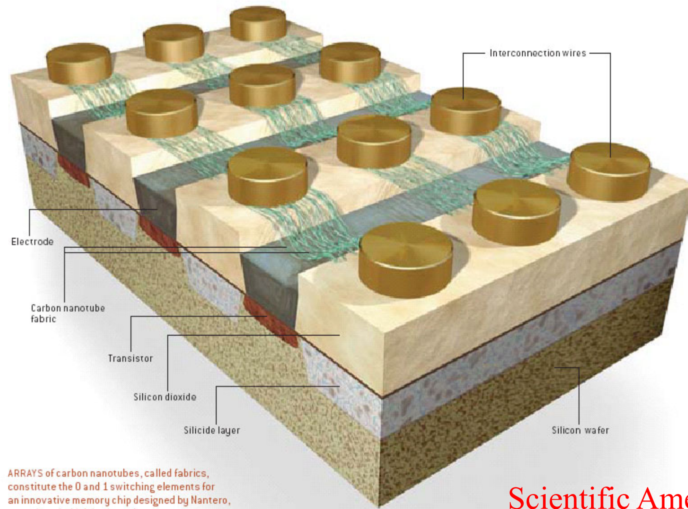


Toggle MRAM Block Diagram



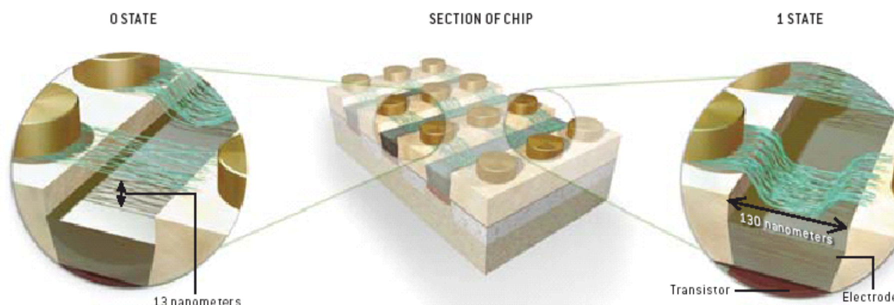
Sold by Everspin Technologies, Chandler, AZ

The Radical Fringe: Carbon Nanotubes



Scientific American, Feb. 2005

The Radical Fringe: Carbon Nanotubes



SAGGING AND STRAIGHTENING represent the 1 and 0 states for a random-access memory made up of groupings of nanotubes. In its 0 state, the fabric remains suspended above the electrode (*left*). When a transistor turns on, the electrode produces an electric field that causes a nanotube fabric to bend and touch an electrode, a configuration that denotes a 1 state (*right*).

Developed by Nantera, being commercialized by LSI Logic