FUNDAMENTALS OF VLSI DESIGN

(19EC6DCFOV)

Module-4

Advanced CMOS Logic Structures: Mirror circuits, Pseudo-nMOS Logic, Tristate circuits, Clocked CMOS Logic, Dynamic CMOS Logic circuits (Text book-2)

Array Subsystems: Introduction, Static Random-Access Memory (SRAM), Dynamic Random-Access Memory (DRAM), Read only Memory, Serial Access Memories, Content addressable memory. (Text 3)

Text books:

2. John P.Uyemura, “Introduction to VLSI Circuits and Systems”, Wiley India Edition, 3rd print, 2007.
3. Neil H.E. Weste, Harris, Banerjee, “CMOS VLSI design”, *Pearson*, Third Edition, 2007.

Advanced CMOS Logic Structures

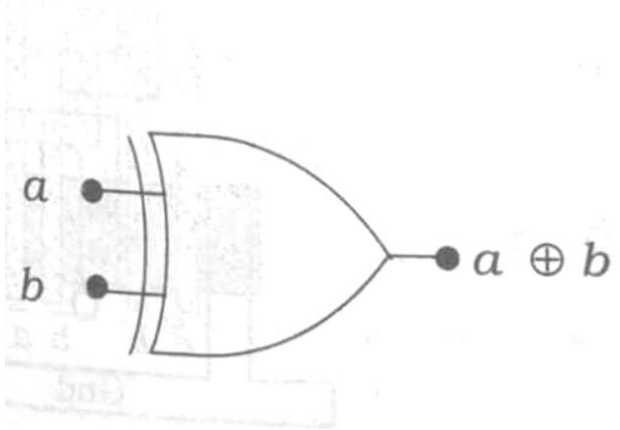
Variety of CMOS circuit design styles

- useful in the design of high-speed VLSI networks.
- operate in distinct ways.
- To overcome one or more problems that have arisen as VLSI applications
- Some are very general, while others are used only for special cases.

1. Mirror circuits
2. Pseudo-nMOS Logic
3. Tristate circuits
4. Clocked CMOS Logic
5. Dynamic CMOS Logic circuits

Mirror Circuits

- Mirror circuits are based on series-parallel logic gates, but are usually faster and have more uniform layout.
- The basic idea of a mirror is seen from XOR truth table.



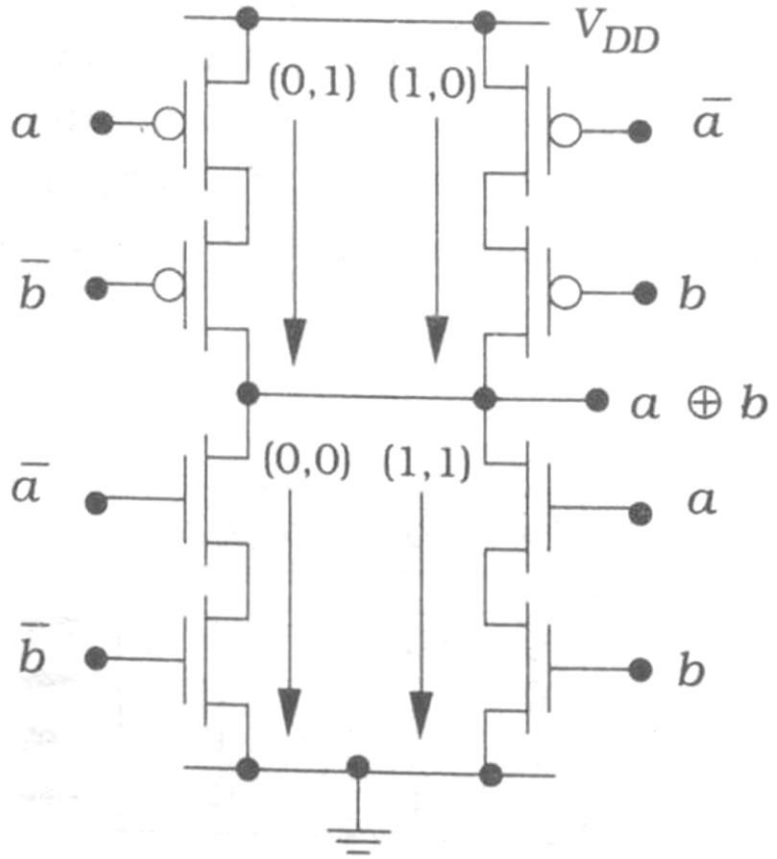
a	b	$a \oplus b$	On devices
0	0	0	← nFET
0	1	1	← pFET
1	0	1	← pFET
1	1	0	← nFET

Important aspect:
There are equal numbers of input combinations that produce 0's and 1's.

- Output 0's imply that an nFET chain is conducting to ground.
- Output 1's means that a pFET group provides support from the power supply.

- Mirror effect can be understood by placing a mirror along the output line, facing either up or down.
- The mirror image seen in the mirror will be the other side of the circuit.

a	b	$a \oplus b$	On devices
0	0	0	← nFET
0	1	1	← pFET
1	0	1	← pFET
1	1	0	← nFET

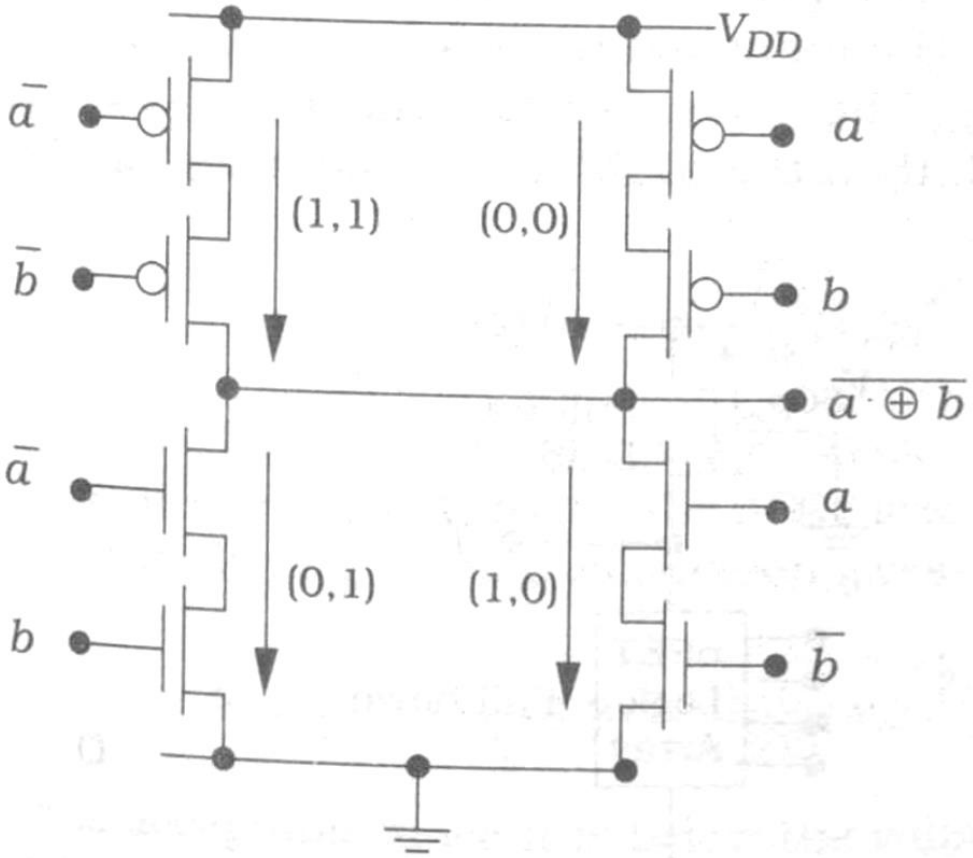


XOR mirror circuit

Advantages:

1. circuit are more symmetric layout
2. shorter rise and fall times.

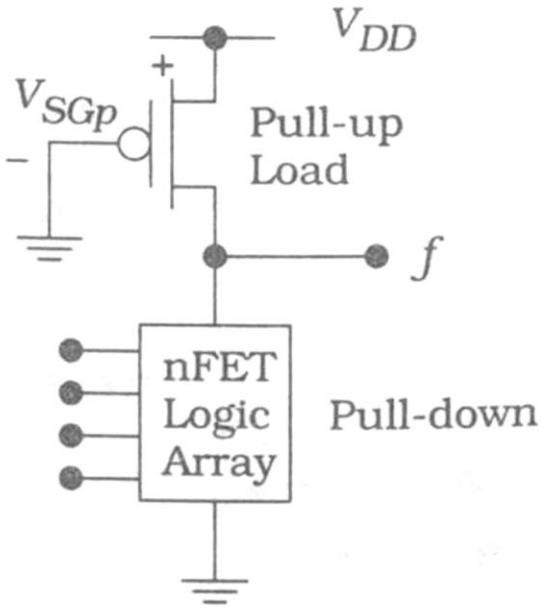
XNOR Mirror circuit



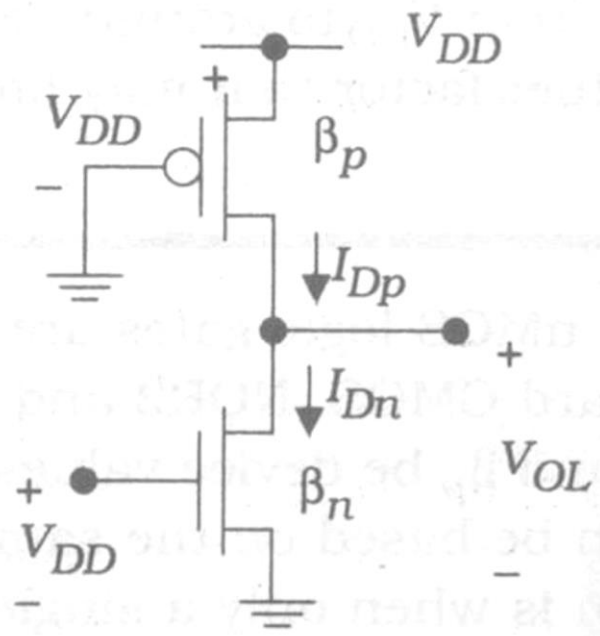
a	b	$\overline{a \oplus b}$
0	0	1
0	1	0
1	0	0
1	1	1

Pseudo-nMOS Logic

- Adding a single pFET to nFET arrays - called pseudo-nMOS.
- For N inputs, a pseudo-nMOS logic gate requires (N + 1) FETs.



- To illustrate the sizing problem,



$$I_{Dn} = I_{DP}$$

$$\beta\left[(V_{gs} - V_t)V_{ds} - \frac{V_{ds}^2}{2}\right] = \frac{\beta}{2}(V_{gs} - V_t)^2$$

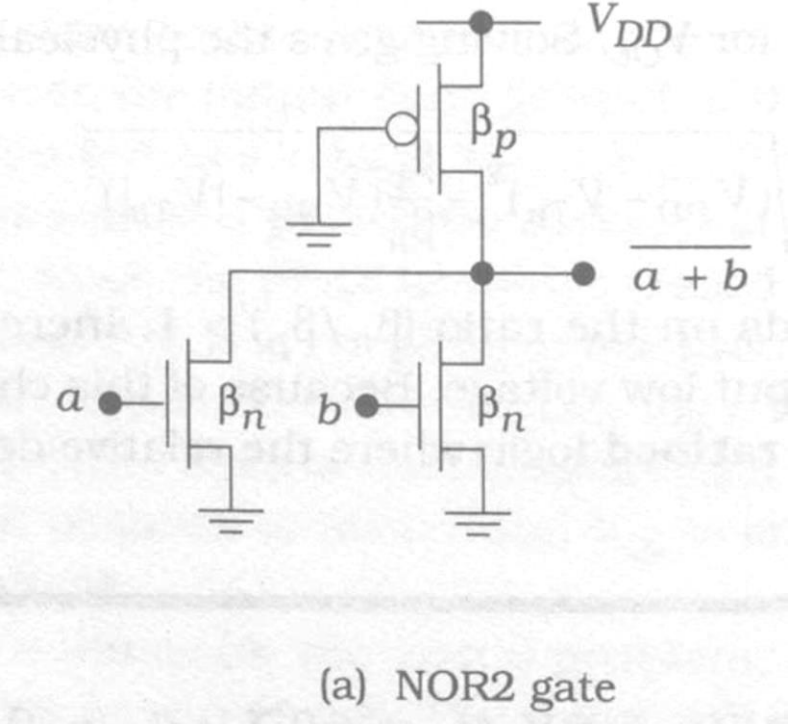
$$\frac{\beta_n}{2}[2(V_{DD} - V_{Tn})V_{OL} - V_{OL}^2] = \frac{\beta_p}{2}(V_{DD} - |V_{Tp}|)^2$$

$$V_{OL} = (V_{DD} - V_{Tn}) - \sqrt{(V_{DD} - V_{Tn})^2 - \frac{\beta_p}{\beta_n}(V_{DD} - |V_{Tp}|)^2}$$

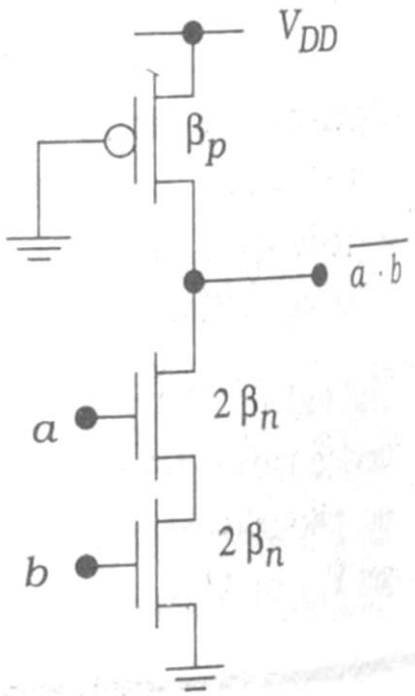
V_{OL} thus depends on the ratio $(\beta_n/\beta_p) > 1$.

- Increasing the device ratio decreases the output low voltage.
- pseudo-nMOS is a type of ratioed logic

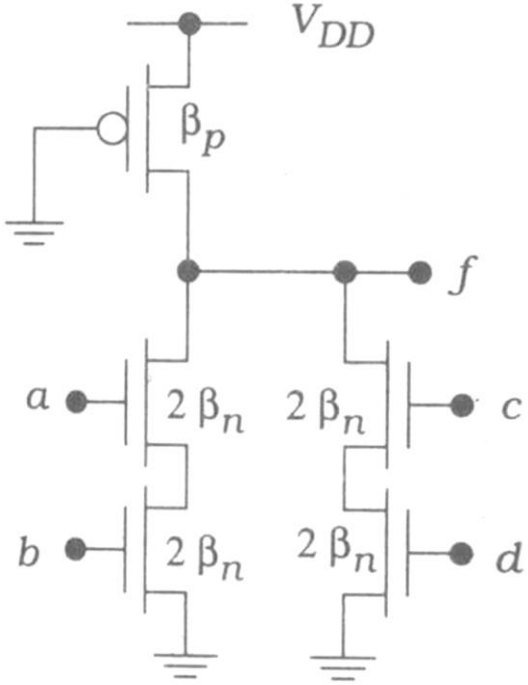
- NOR and NAND



(a) NOR2 gate

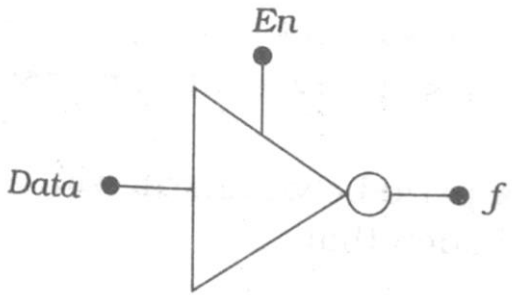


(b) NAND2 gate



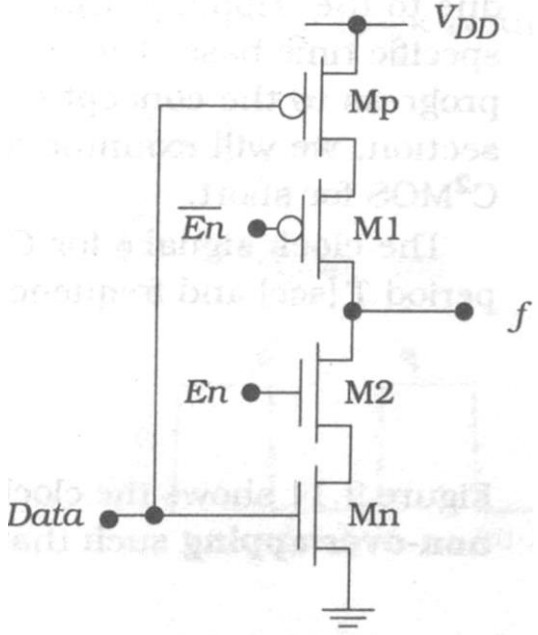
Tristate circuits

- A tri-state circuit produces the usual 0 and 1 voltages, but also gives third high-impedance Z (or Hi-Z) state
- Tri-state circuits are useful for isolating circuits from common bus lines.



En	f
0	Z
1	$\overline{Data}$

(a) Symbol and operation



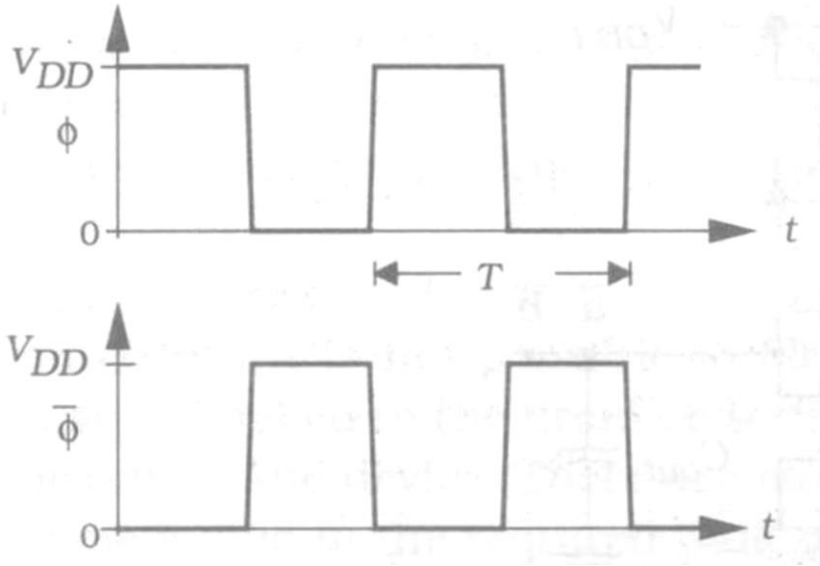
(b) CMOS circuit

- Enable signal En controls the operation.
- With En = 0, the output is "tri-stated" which means that f= Z.
- Normal operation occurs with En = 1.
- A non-inverting circuit (a buffer) can be obtained by neg static inverter to the input.

Clocked CMOS

- The real power of digital logic is realized only when we progress to the concept of clock control and sequential circuits.
- Clocked CMOS (C^2MOS)
- The clock signal φ(or Clk), period T, Frequency f

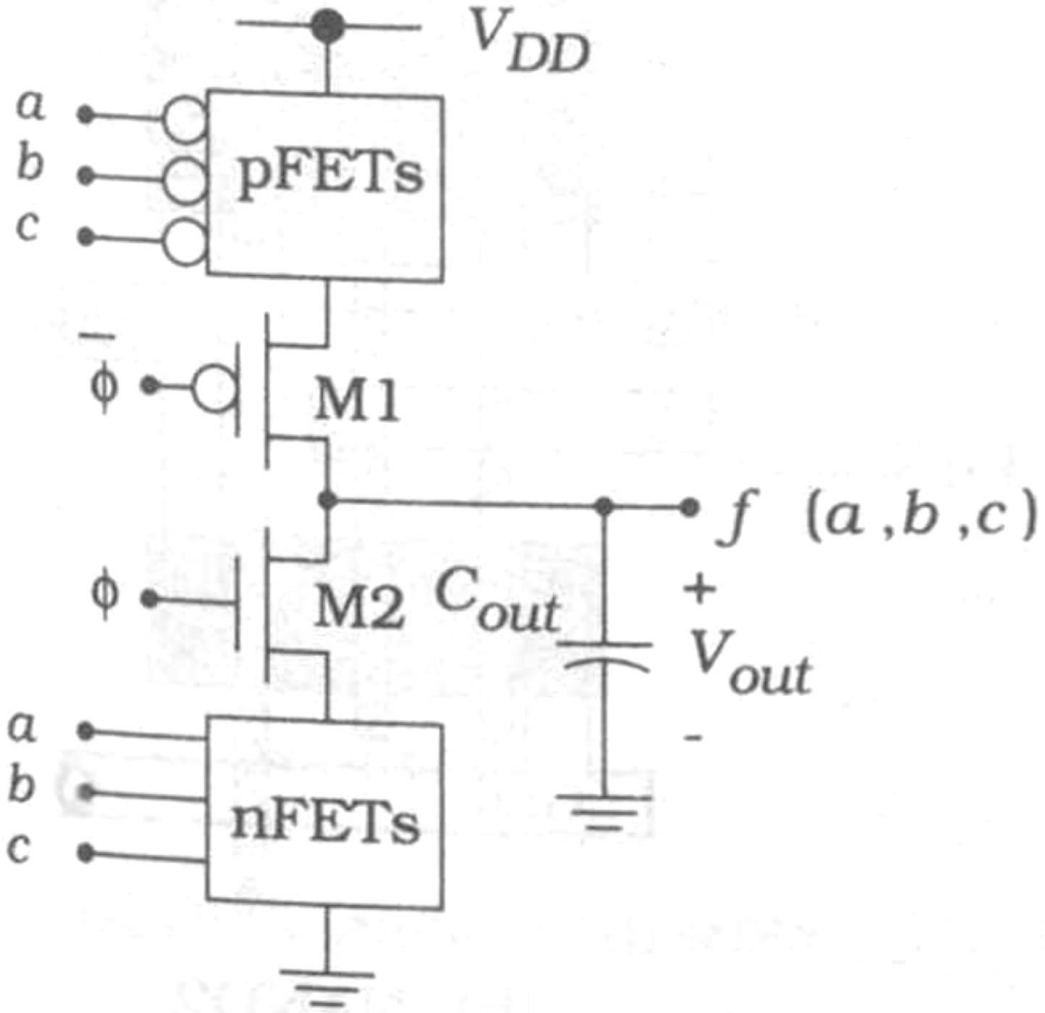
$$f = \frac{1}{T}$$

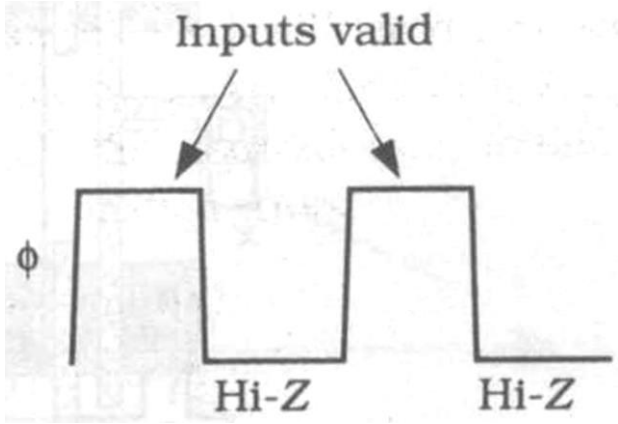


- Clk, φ and its complement $\bar{\varphi}$
- non-overlapping $\phi(t) \cdot \bar{\phi}(t) = 0$
- φ(t) minimum value of 0 and a maximum value of Vdd,

$$\bar{\phi}(t) = V_{DD} - \phi(t)$$

General structure of C²MOS





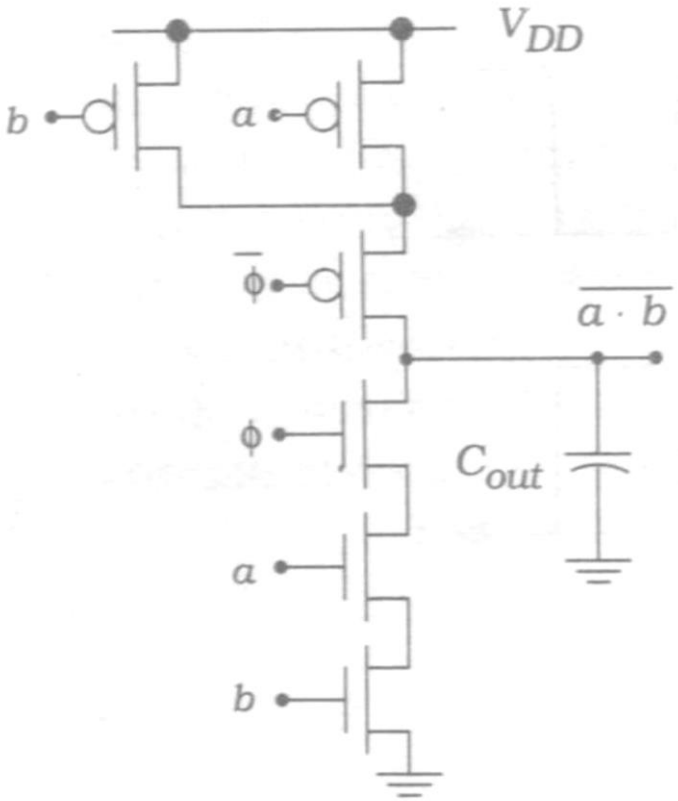
$\emptyset = 1$,

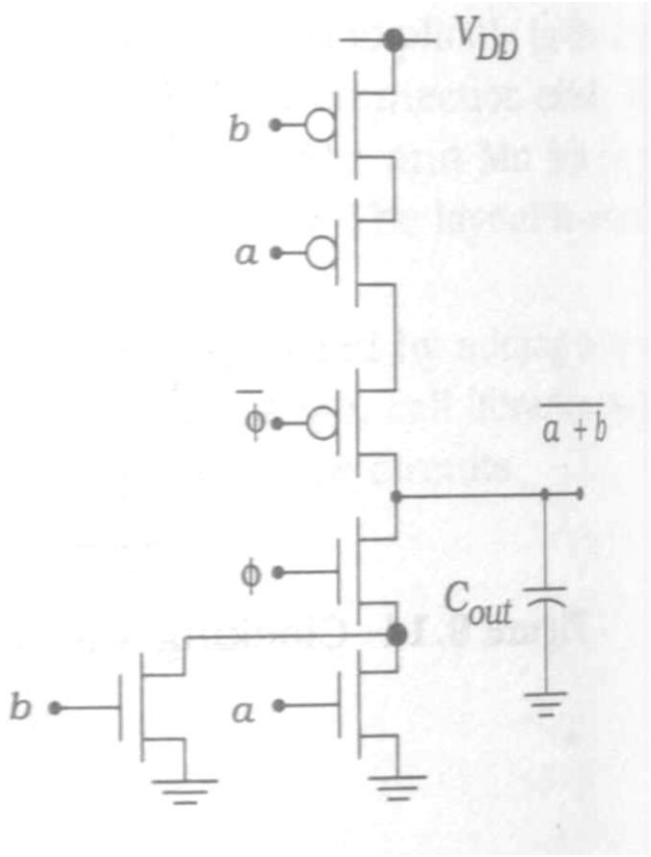
- M1 and M2 → Active
- Standard output

$\emptyset = 0$,

- M1 and M2 → Cut-off
- High impedance state

- NAND and NOR

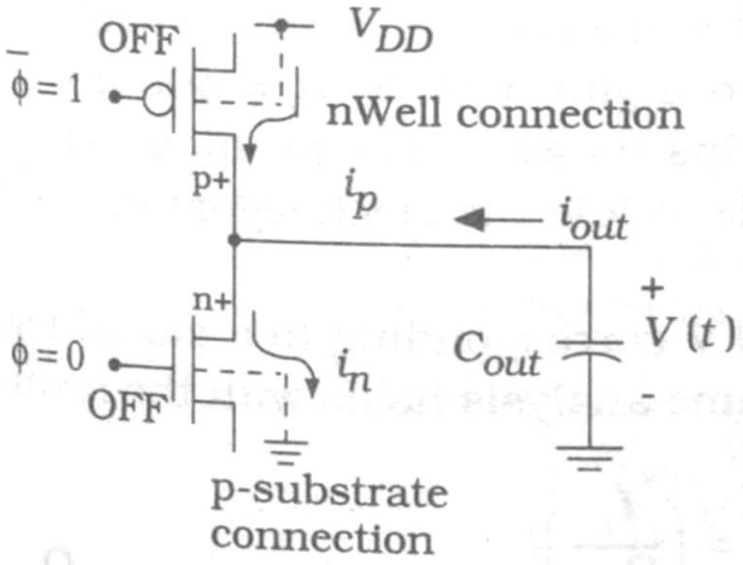

V_{DD}
b
a
$\overline{\phi}$
$\overline{a \cdot b}$
ϕ
C_{out}
a
b


V_{DD}
b
a
$\overline{\phi}$
$\overline{a+b}$
ϕ
C_{out}
b
a

Draw backs:

1. Charge leakage
2. Subthreshold leakage

- output node cannot hold the charge on Vout very long due to a phenomenon called **charge leakage.**
- If a voltage is applied to the drain or source, a small leakage current flows into, or out of, the device.



$$i_{out} = i_n - i_p$$
$$= -C_{out}\frac{dV}{dt}$$

To see the effect of leakage currents,

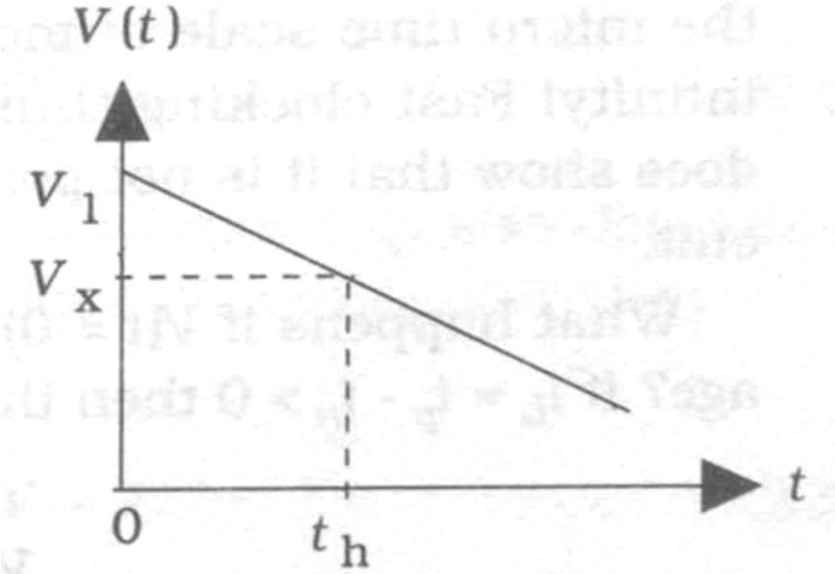
- Initial voltage $V(t = 0) = V_1$ stored on the capacitor.
- If $i_n > i_p$, then $i_{out} = I_L$, is a Positive number, indicating current flow off of the capacitor.

$$I_L = -C_{out}\frac{dV}{dt}$$

$$\int_{V_1}^{V(t)} dV = -\int_0^t \left(\frac{I_L}{C_{out}}\right) dt$$

$$V(t) = V_1 - \left(\frac{I_L}{C_{out}}\right) t$$

To calculate the hold time,



$$V(t_h) = V_1 - \left(\frac{I_L}{C_{out}}\right) t_h$$

$$t_h = \left(\frac{C_{out}}{I_L}\right)(V_1 - V_x)$$

Draw backs:

1. Charge leakage
2. **Subthreshold leakage, I_{sub}**

- Drain-source current that flows even through the gate voltage is less than V_T.

$$I = I_{D0}\left(\frac{W}{L}\right)e^{-(V_{GS}-V_T)/(nV_{th})}$$

I_{D0} varies with V_{ds}

V_{th}, is the thermal voltage $\frac{kT}{q} = 26mV = 0.026V$

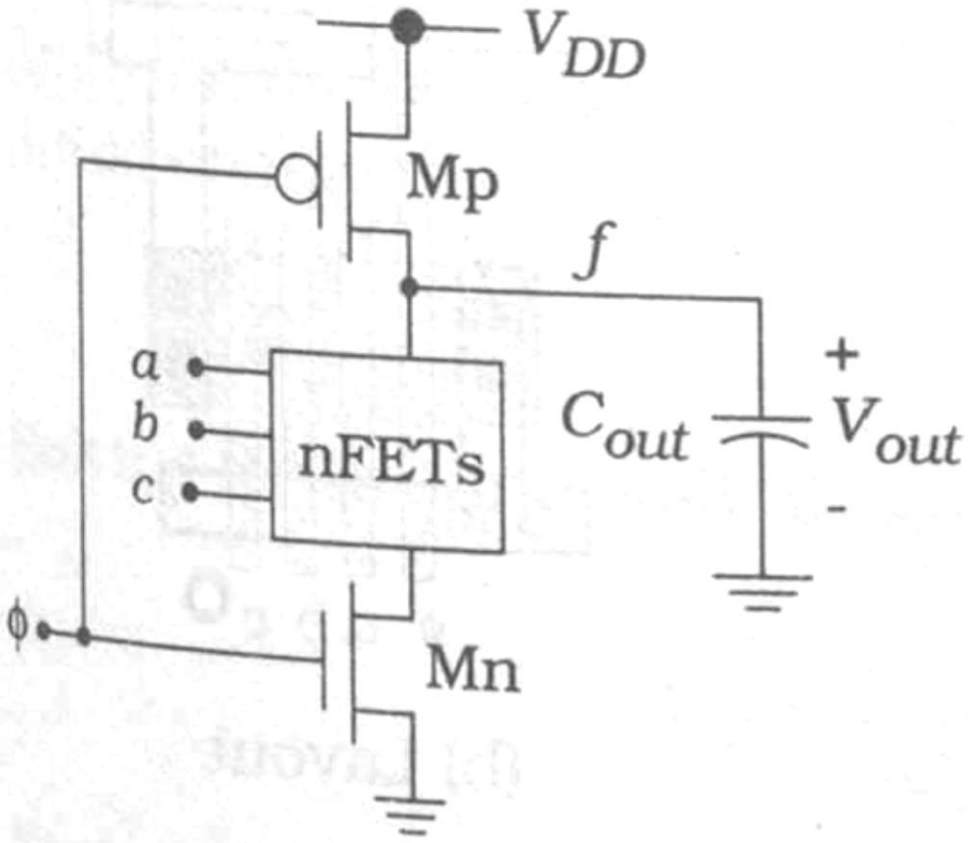
n is a parameter that varies with capacitance.

- Other contributions to create leakage current
    1. Originates from the physical structure
    2. Materials used to create the silicon circuit

- Motivation for Research:
    1. refining the fabrication process using different materials and variations in the FET structures.
    2. to develop new types of transistors to replace the standard MOSFET

Dynamic CMOS

- uses clocking and charge storage properties of MOSFETs to implement logic operations.



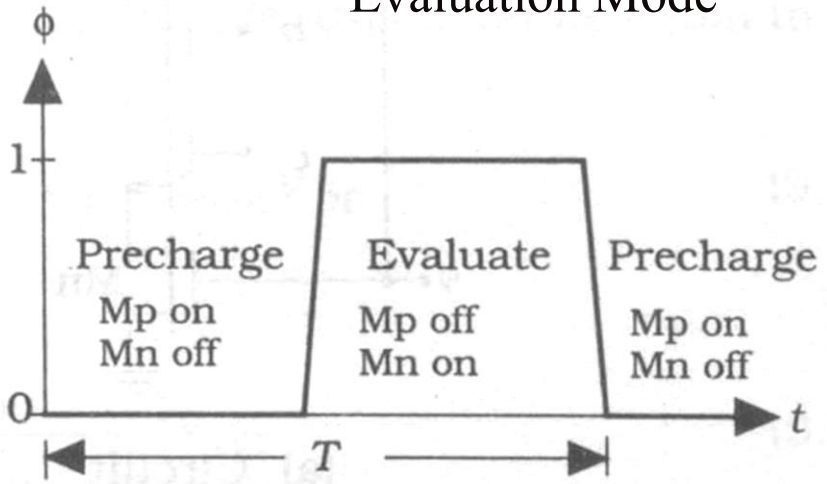
Basic Dynamic Logic circuit

$\emptyset = 0$,

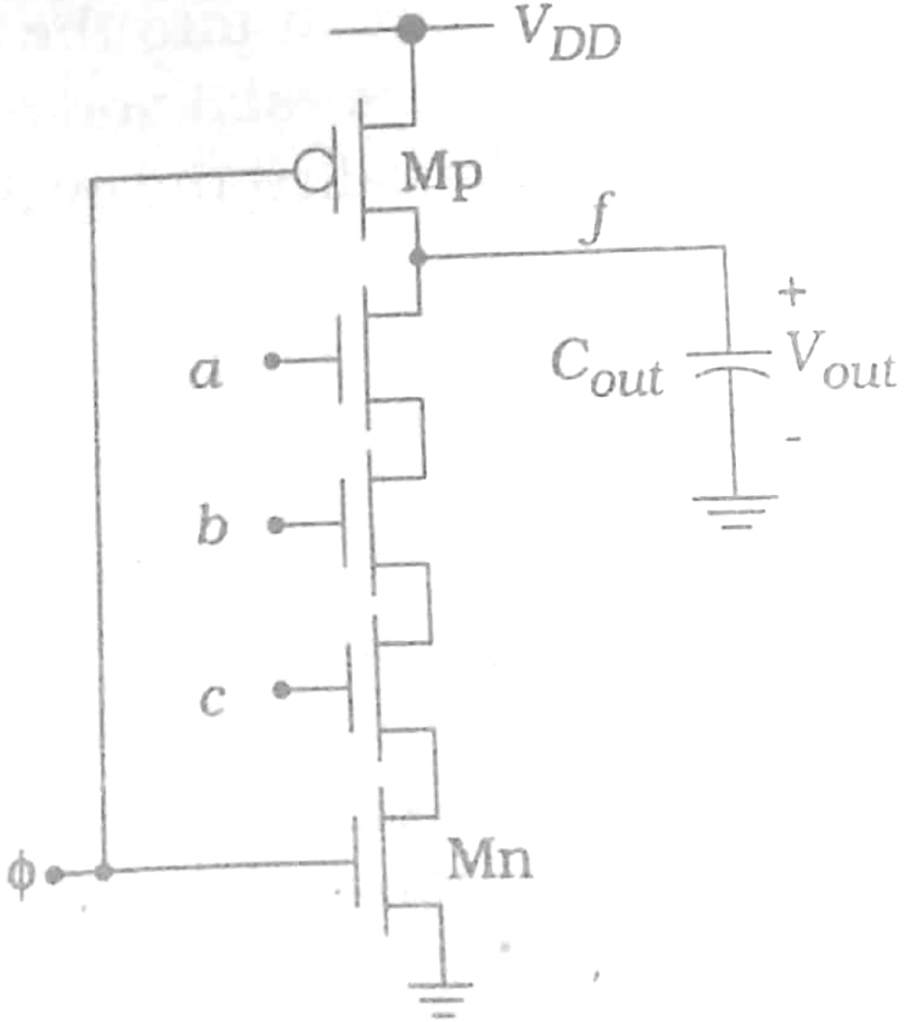
- Mp→ ON (Precharge FET)
- Mn →OFF
- Precharge Mode

$\emptyset = 1$,

- Mp→ OFF
- Mn →ON (Evaluation FET)
- Evaluation Mode



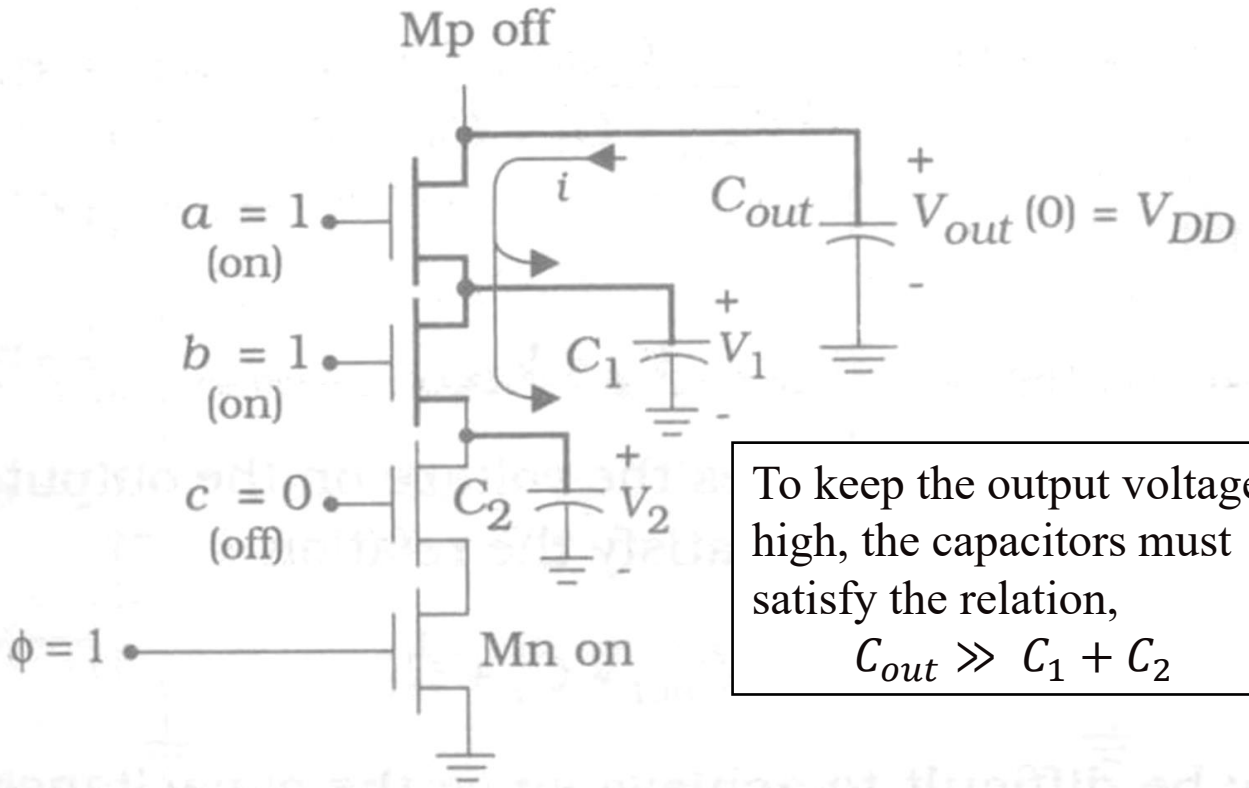
- NAND3



(a) Circuit

$$f = \overline{a \cdot b \cdot c}$$

Draw back → Charge sharing



To keep the output voltage high, the capacitors must satisfy the relation,

$$C_{out} \gg C_1 + C_2$$

- Occur when the clock makes the transition to $\emptyset = 1$
- Effect of reducing the output voltage
- origin of the charge sharing problem is the parasitic node capacitance C1 and C2 between FETs.

When $\emptyset = 1$,

- Mp → off, isolating the output node from power supply.
- At the start of evaluation Vout = VDD
- Assume, V1 and V2 are 0V
- Total charge in the circuit, $Q = C_{out}V_{DD}$
- Worst case charge sharing condition for this circuit is when (a,b,c) = (1,1,0)
- With, C=0, there is no discharge path to ground.

$$V_{out} = V_2 = V_1 = V_f$$

$$Q = C_{out}V_f + C_1V_f + C_2V_f$$

$$= (C_{out} + C_1 + C_2)V_f$$

$$Q = (C_{out} + C_1 + C_2)V_f = C_{out}V_{DD}$$

$$V_f = \left(\frac{C_{out}}{C_{out} + C_1 + C_2}\right)V_{DD}$$

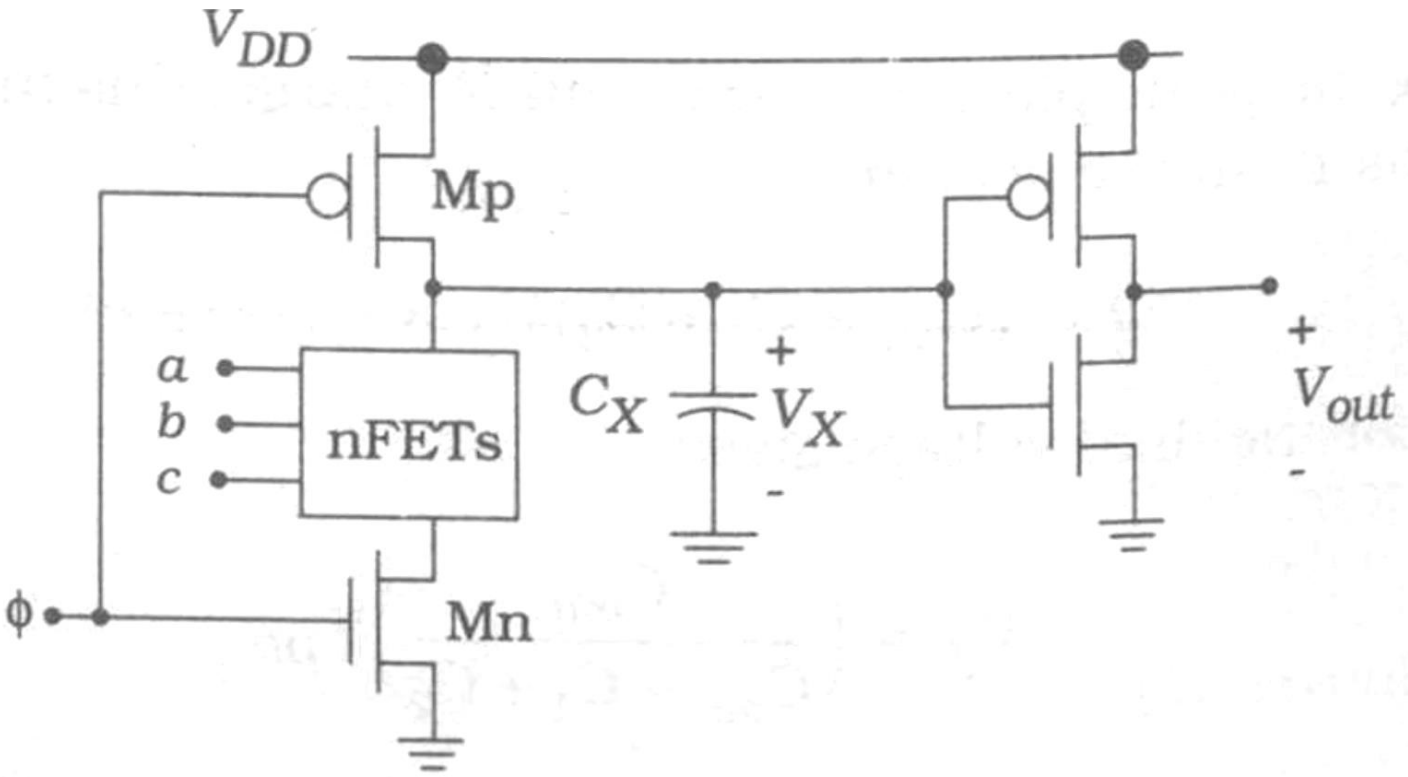
<1

$$V_f < V_{DD}$$

Charge sharing thus reduces the voltage on the output node.

Domino Logic

- adding static inverter to the output of the basic dynamic gate circuit.



- Domino logic gates are **non-inverting** because of the output inverter.

The precharge and evaluate events still occur, but the capacitor Cx is affected.

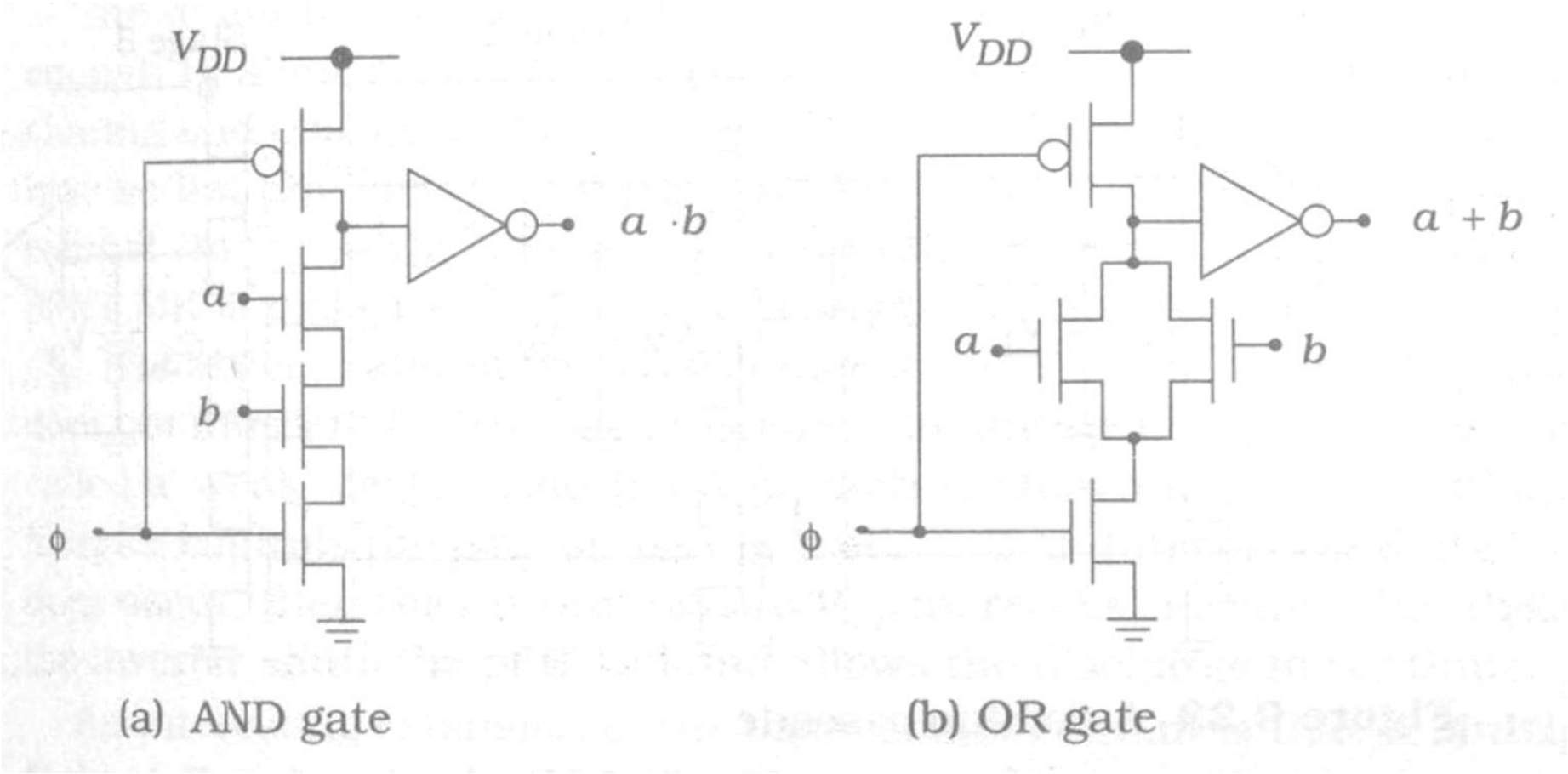
$\emptyset = 0$,

- defines the precharge
- Cx is charged to a voltage Vx = V_{DD}
- Output voltage, Vout= 0 V

$\emptyset = 1$,

- Inputs are valid during the evaluation interval
- If Cx holds its charge, Vx→high and Vout= 0 V indicates a **"logic 0"**
- If Cx discharges, then Vx → 0 V and Vout= V_{DD} corresponds a **"logic 1"**

- AND and OR



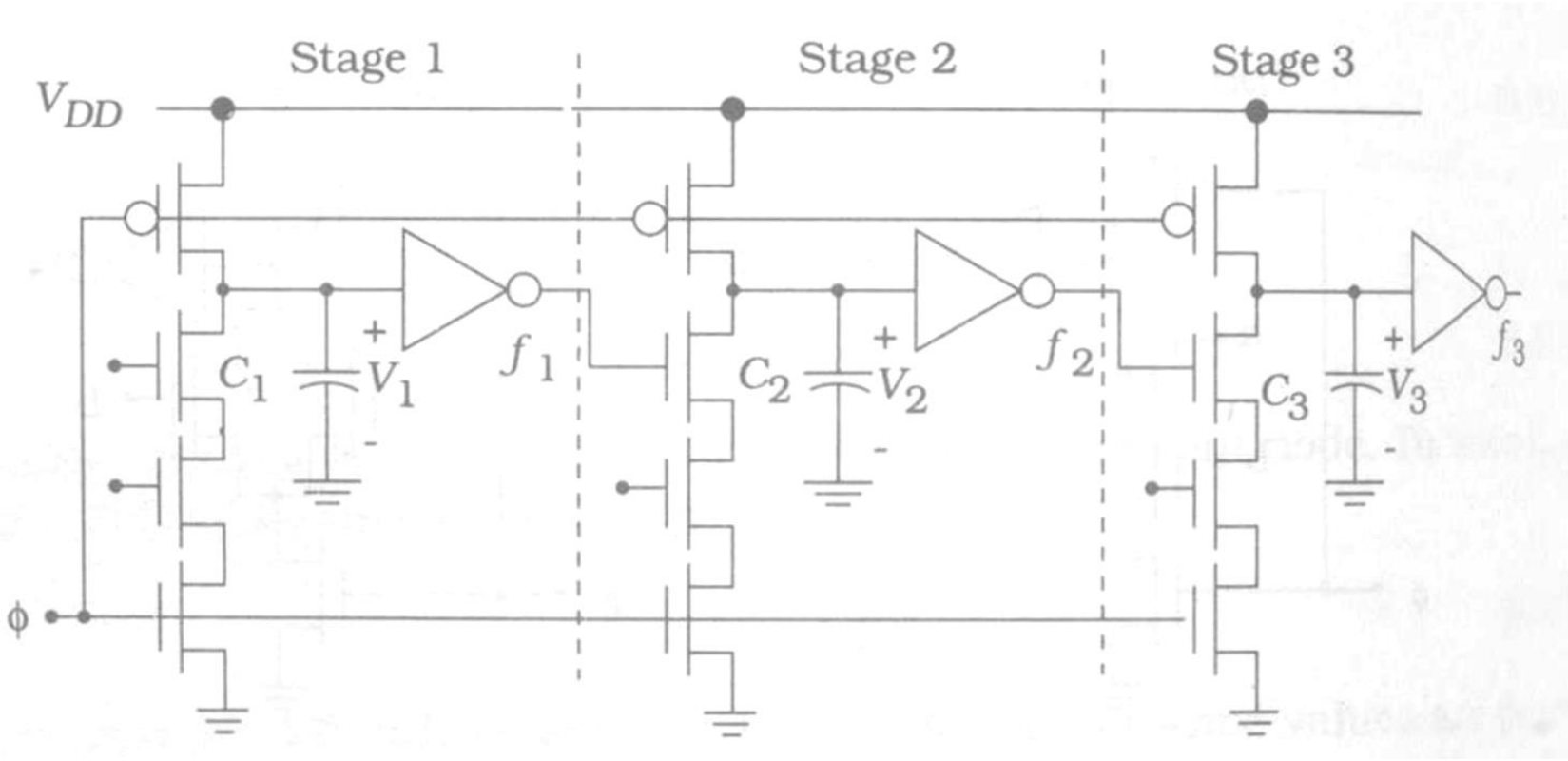
Domino Effect

- a cumulative effect produced when one event initiates a **succession of similar events**

- everyone will fall over like a row of dominoes

Domino logic derives its name from the manner in which a **cascade operates.**

Domino cascade



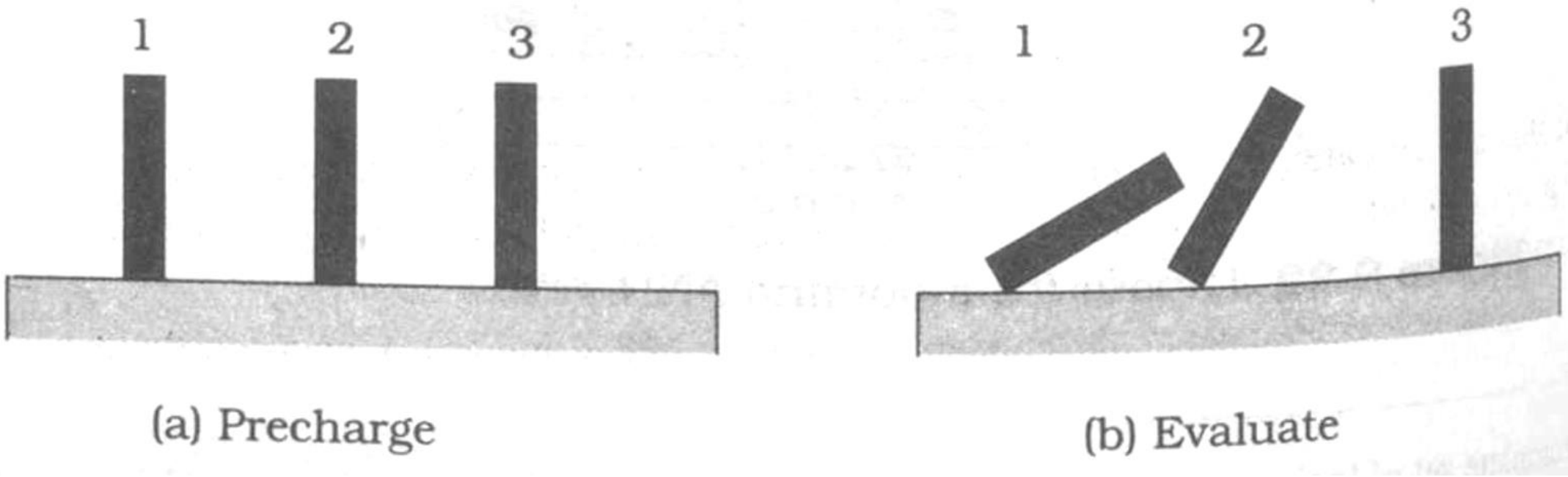
$\emptyset = 0$,

- Precharge event
- capacitors C1, C2, and C3 are simultaneously charged to VDD
- outputs f1, f2, and f3 to all be 0.

$\emptyset = 1$,

- Evaluation mode
- ***"domino chain reaction"*** that must start at the first stage and then propagate stage by stage to the output.

Visualization of the domino effect

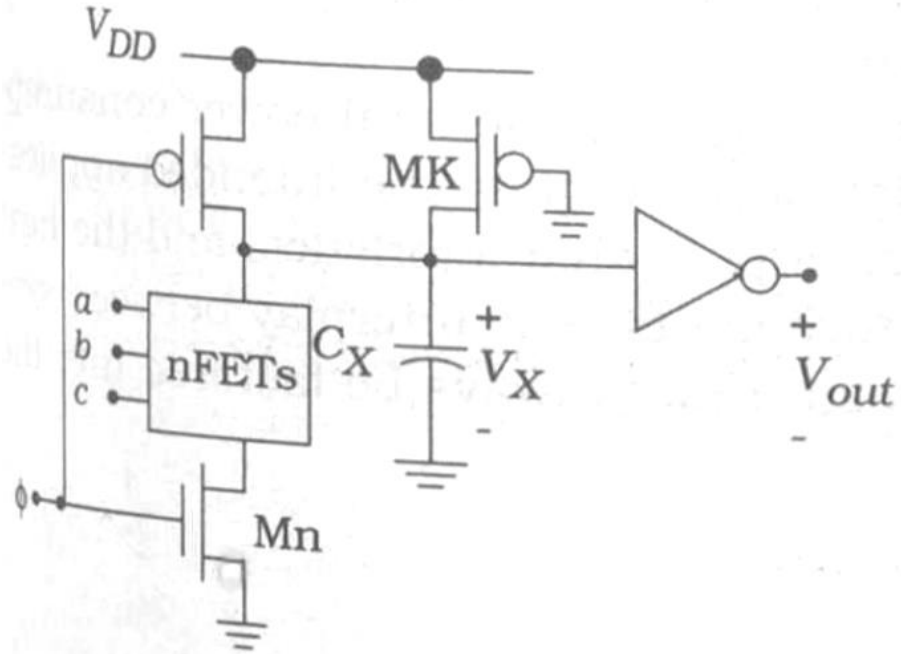


dominos standing on end.

indicated by a falling domino.

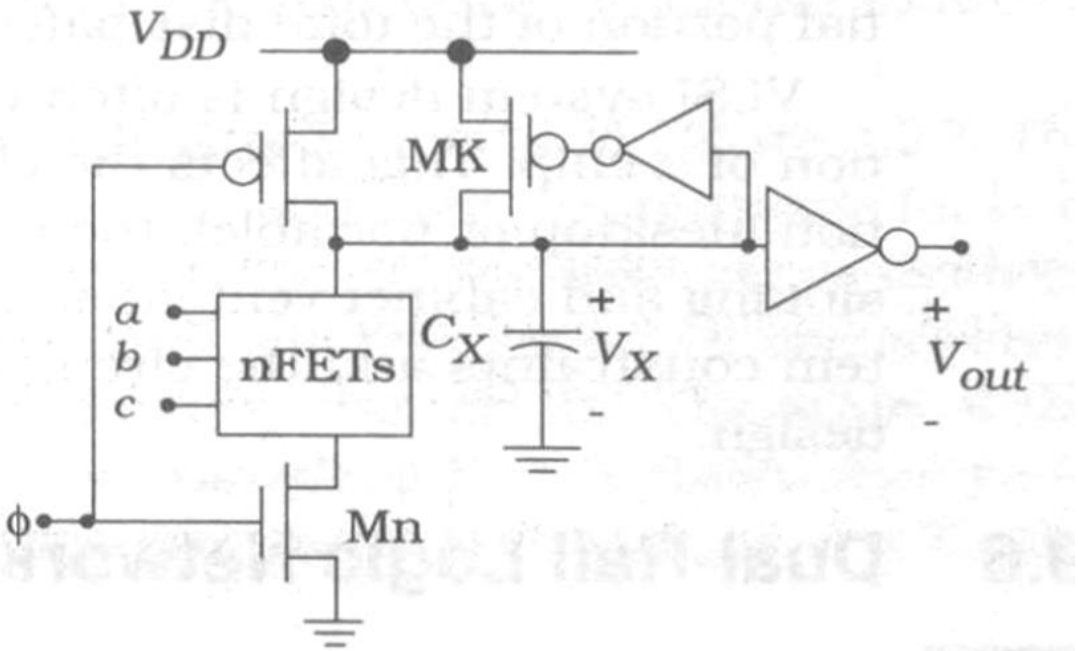
- Stages 1 and 2 have undergone a discharge, but Stage 3 remain high(in its precharge state).
- The domino cascade must have an **evaluation interval** that is long enough to allow every stage time to discharge.
- means that **charge sharing** and **charge leakage processes** that **reduce the interval voltage Vx** may be limiting factors.

Charge Keeper



(a) Single-FET charge keeper

- **pFET MK** is biased active to **allow** a small current **to restore charge on Cx**.
- The **aspect ratio** of the charge-keeper FET **must be small** so that it does not interfere with a discharge event in any significant manner; this is called a ***'weak' device.***

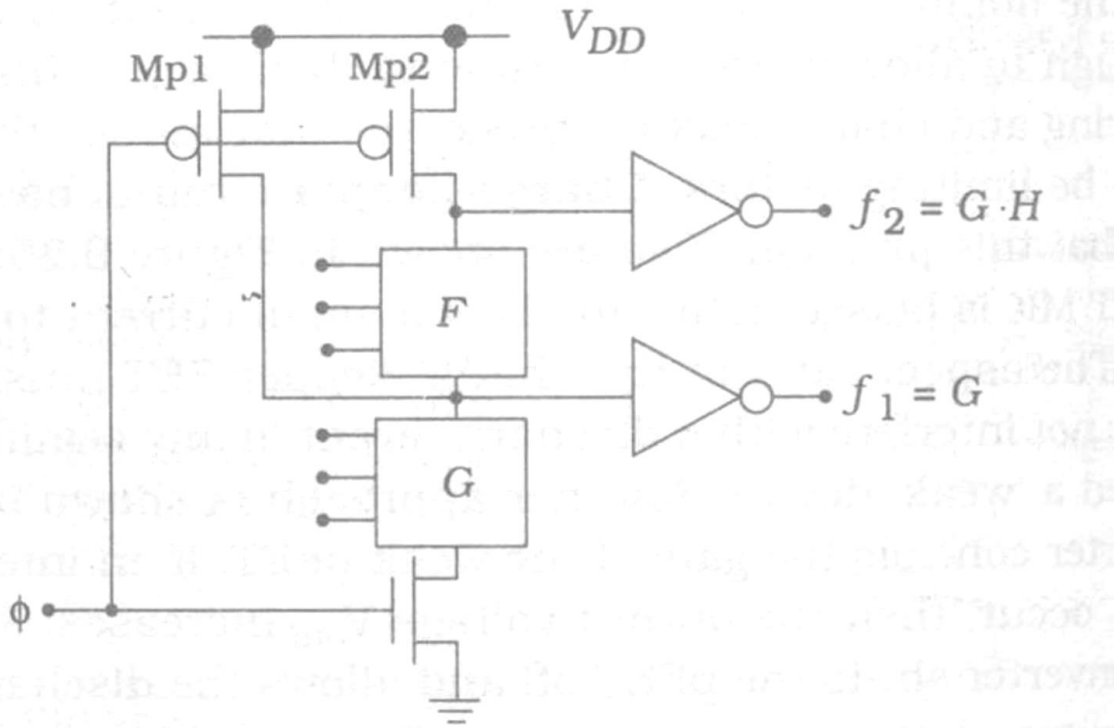


(b) Feedback controlled keeper

- An **inverter controls** the gate of the **weak pFET**.
- If an **internal discharge of Cx does occur**, then the output voltage **Vout increases**.
- **Feeding** this through the **inverter shuts the pFET** off and allows the discharge to continue.

Extension of domino circuit : **Multiple output Domino Logic (MODL)**

- allows two or more outputs from a single logic gate, making it quite unique.



Structure of a 2 output MODL circuit

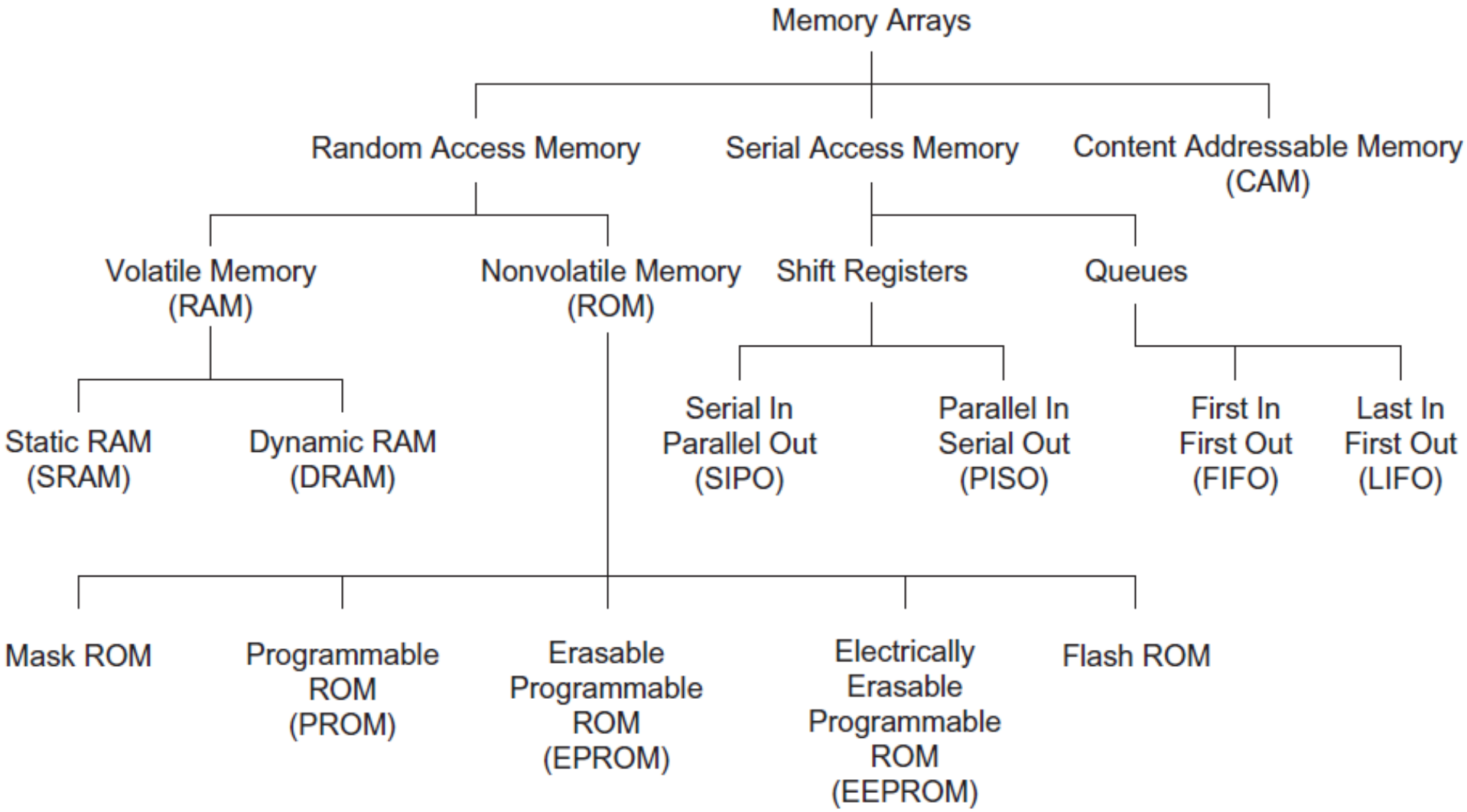
- The logic array has been **split into two separate blocks** denoted as F and G, which **creates an additional output node**.
- If the **G-logic block** acts like a **closed switch**, then it produces an output $\mathbf{f_1 = G}$.
- If this occurs, then it is possible for the **second logic block F** to **induce a discharge** by also **acting as a closed switch**.

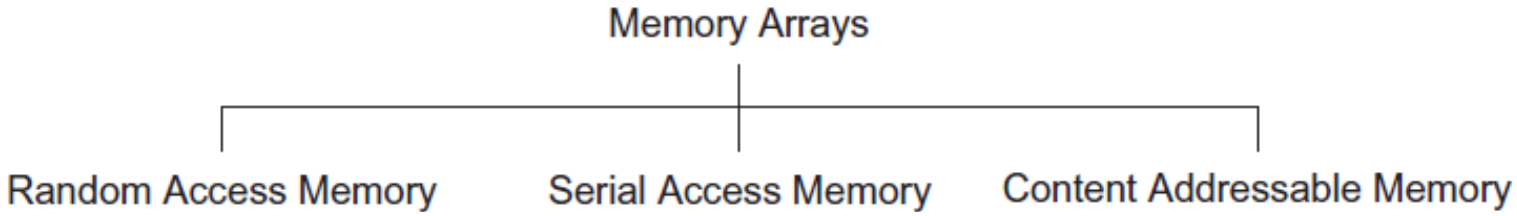
Power Dissipation of Dynamic Logic Circuits

- Dynamic logic circuits can be designed **to provide very fast switching**
- they can be **quite power hungry**
- charge cannot be held on a **capacitive node**, every precharge cycle will pull current from the voltage source, **adding to the overall power dissipation of the circuit.**
- clock circuits them selves require dynamic power to drive the FETs.
- every stage presents a capacitance of C_L to the clock drivers corresponding to the precharge and evaluate transistors.

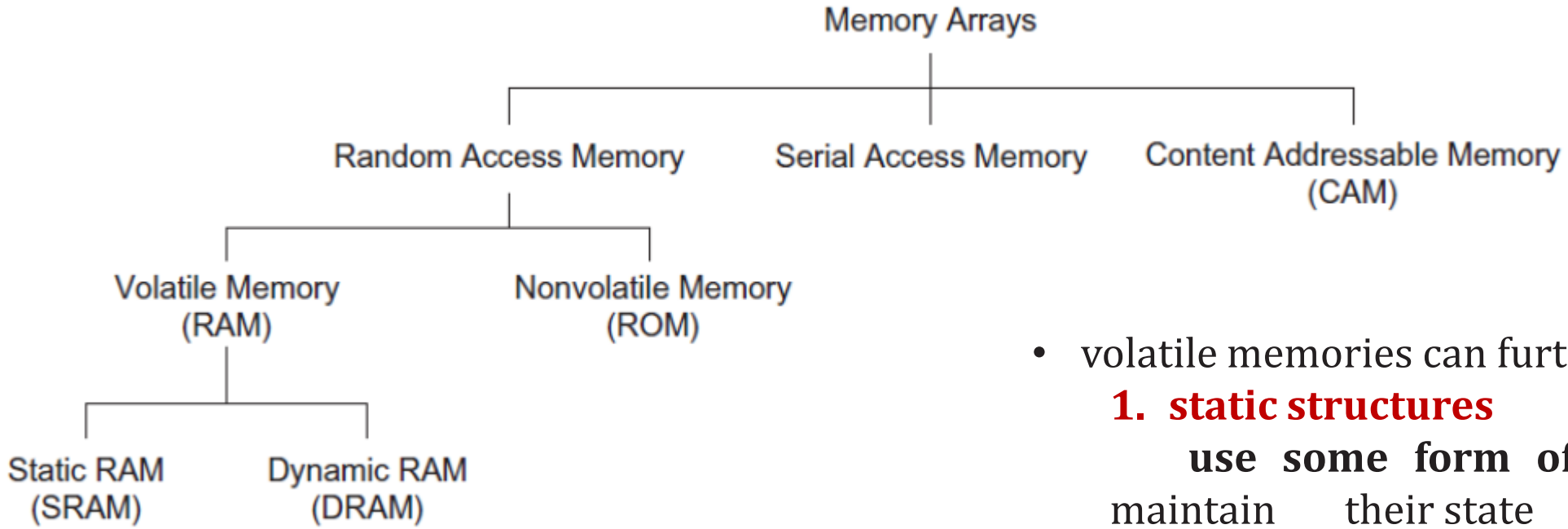
$$C_L = C_{Gp} + C_{Gn}$$

Introduction





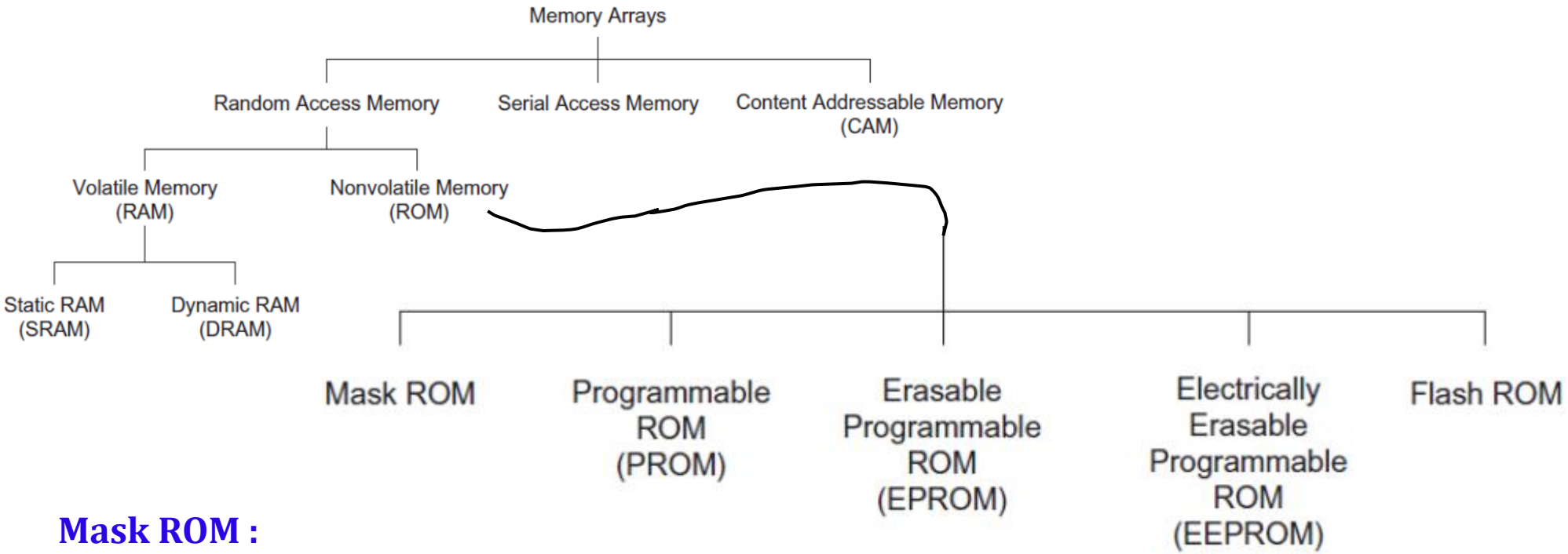
- **Random access memory** is **accessed with an address** and **has a latency independent of the address.**
- **Serial access memories** are **accessed sequentially** so **no address is necessary.**
- **Content addressable memories (CAM) determine which address(es) contain data** that matches a specified key.



- Random access memory is commonly classified
  - ***1) read-only memory*** (ROM)
  - ***2) read/write memory*** (confusingly called RAM).
- classification is *volatile* vs. *nonvolatile* memory.
  1. RAM is synonymous with **volatile memory.**
  2. ROM is synonymous with **nonvolatile memory.**

- volatile memories can further be divided
  1. **static structures**
     use some form of feedback to maintain their state
  2. **dynamic structures**
     use charge stored on a floating capacitor through an access transistor.

- Memory cells can have one or more ports for access.
- On a read/write memory, each port can be read-only, write-only, or capable of both read and write.



Mask ROM :

Hardwired during fabrication and cannot be changed.

***Programmable* ROM(PROM) :**

Programmed once after fabrication by blowing on-chip fuses with a special high programming voltage.

***Erasable programmable* ROM (EPROM) :**

Programmed by storing charge on a floating gate. It can be erased by exposure to ultraviolet (UV) light for several minutes to knock the charge off the gate. Then the EPROM can be reprogrammed.

***Electrically erasable programmable* ROMs (EEPROMs) :**

EEPROMs are similar to EPROM, but can be erased in microseconds with on-chip circuitry.

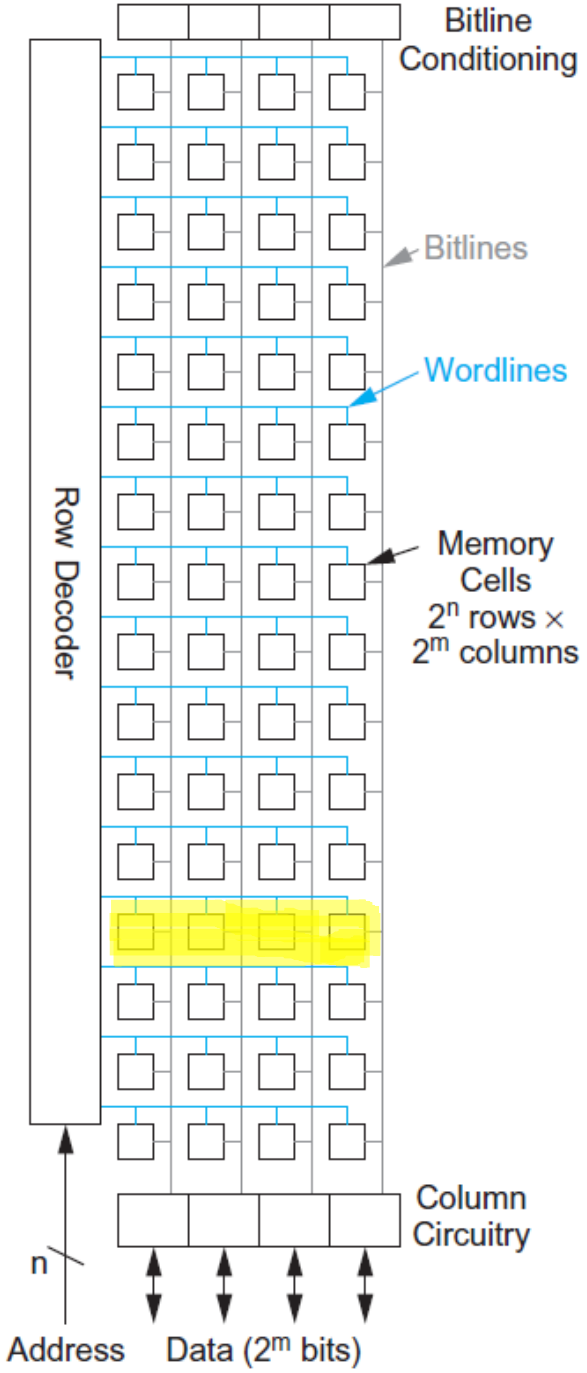
***Flash* memories :**

Variant of EEPROM that erases entire blocks rather than individual bits.

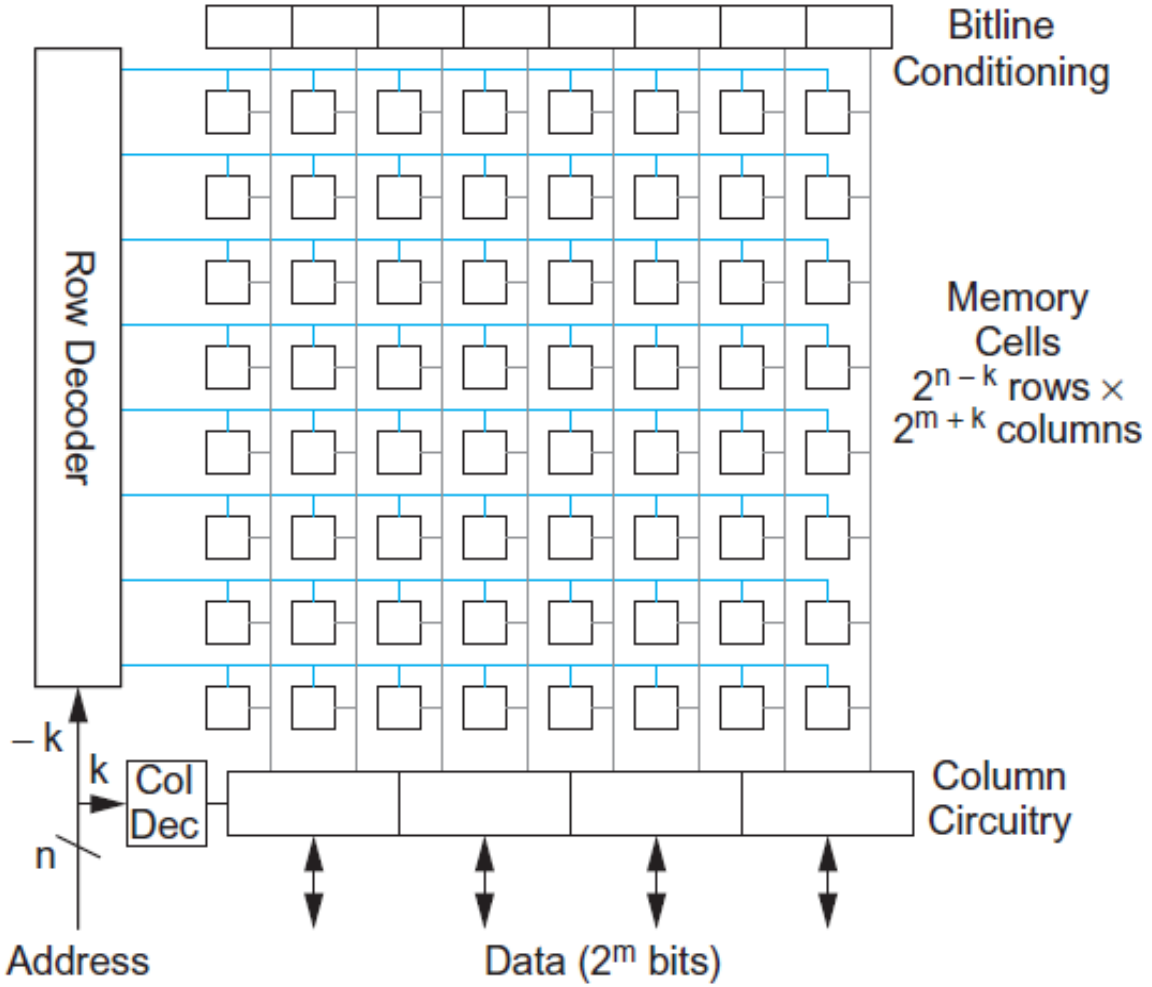
Small memory array architecture

16 → 4 bits

Word line: 1100



- A memory array contains 2^n *words* of 2^m bits each.
- Each bit is stored in a memory cell.
- The row decoder uses the address to activate one of the rows by asserting the wordline.
- During a read operation, the cells on this wordline drive the *bitlines*, which may have been conditioned to a known value in advance of the memory access.
- The column circuitry may contain amplifiers or buffers to sense the data.
- A typical memory array may have **thousands or millions of words**, which would lead to a tall, skinny layout that **is hard to fit in the chip** floorplan and slow because of the long vertical wires. Therefore, **the array is often folded into fewer rows of more columns.**

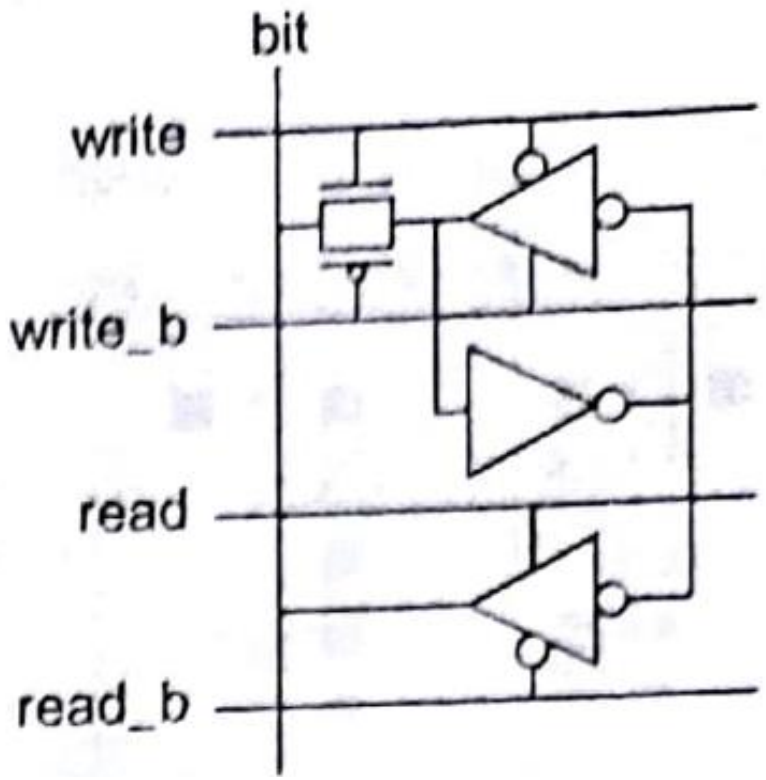


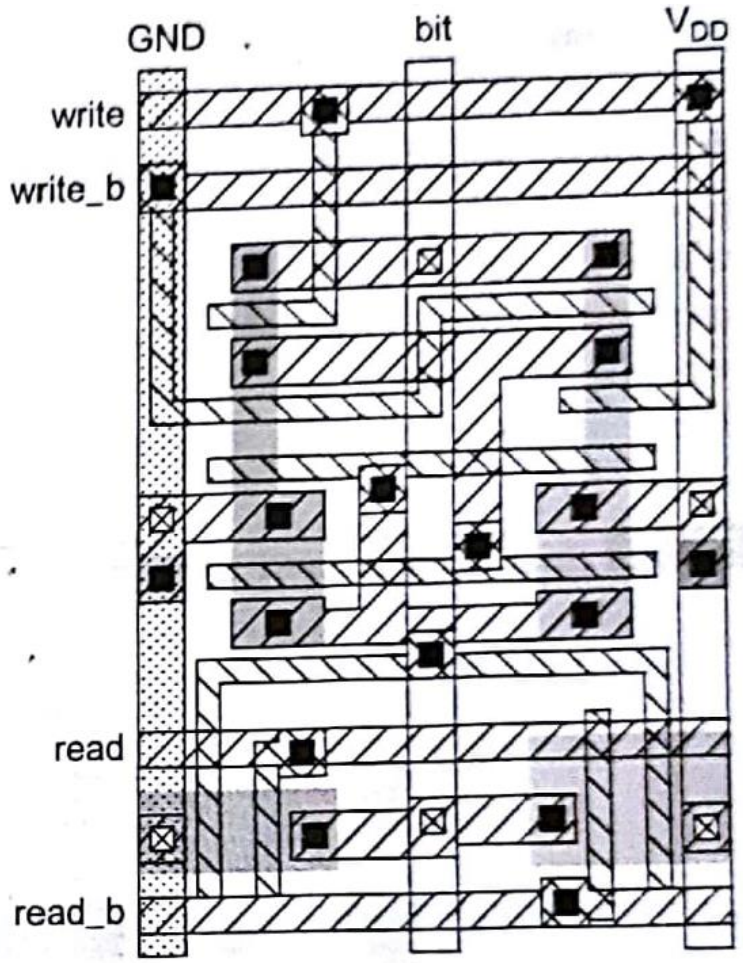
- each row of the memory contains 2^k words, so the array is physically organized as 2^{n-k} rows of 2^{m+k} columns or bits.
- Figure shows a two-way fold (k = 1) with eight rows and eight columns.
- The column decoder controls a multiplexer in the column circuitry to select 2^m bits from the row as the data to access.
- Larger memories are generally built from multiple smaller subarrays so that the wordlines and bitlines remain reasonably short, fast, and low in power dissipation.

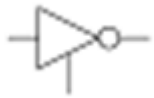
SRAM

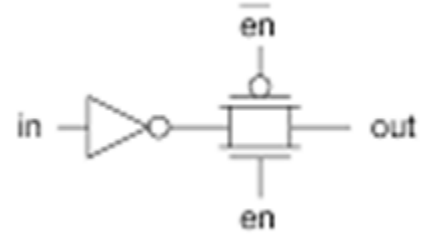
- Static RAMs **use a memory cell** with internal feedback that **retains its value as long as power is applied**.
- Attractive properties:
  1. Denser than flip-flops
  2. Compatible with standard CMOS processes
  3. Faster than DRAM
  4. Easier to use than DRAM
- SRAM cell needs to be **able to read and write data and to hold the data** as long as the power is applied.
- SRAM cell is **activated** by raising the **wordline** and is **read or written through bitline**.

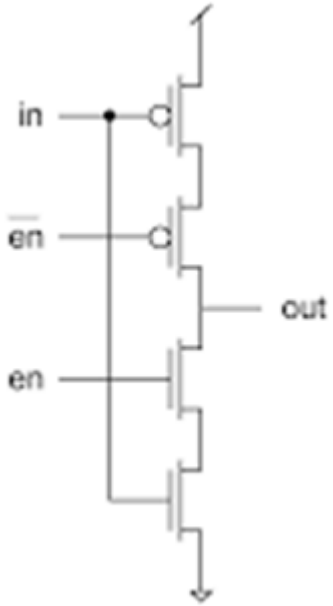
SRAM memory cell



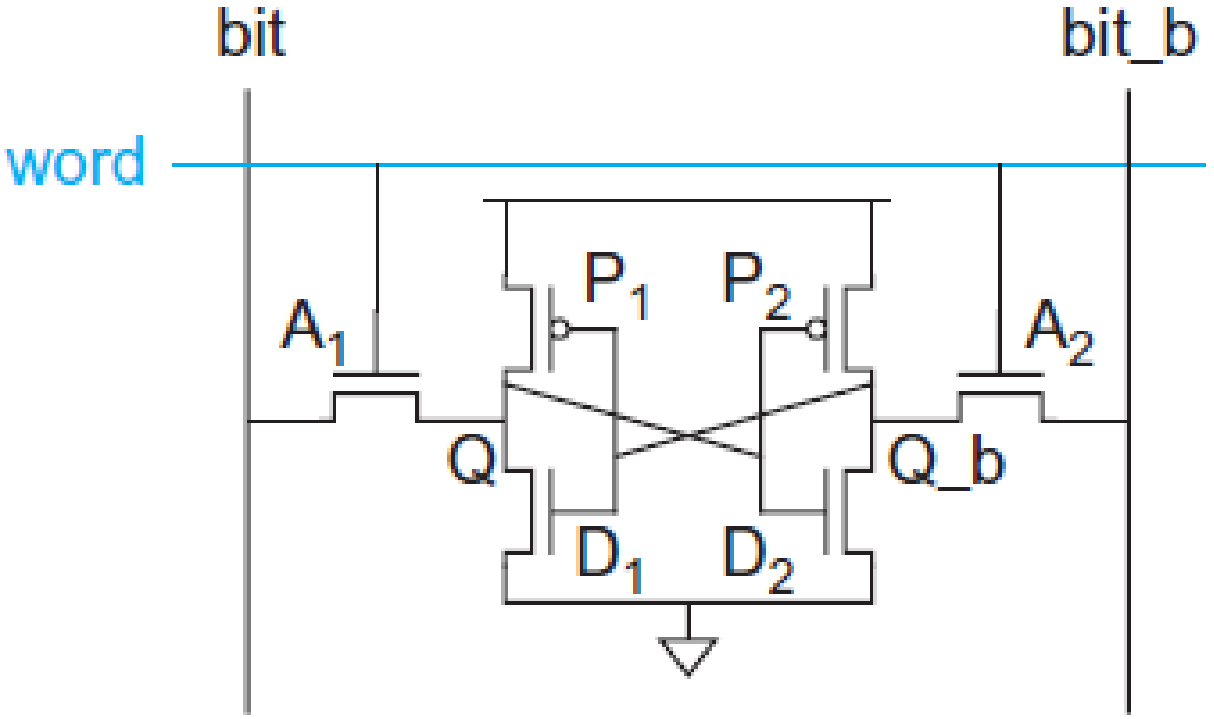








12 Transistor SRAM cell



6 Transistor SRAM cell

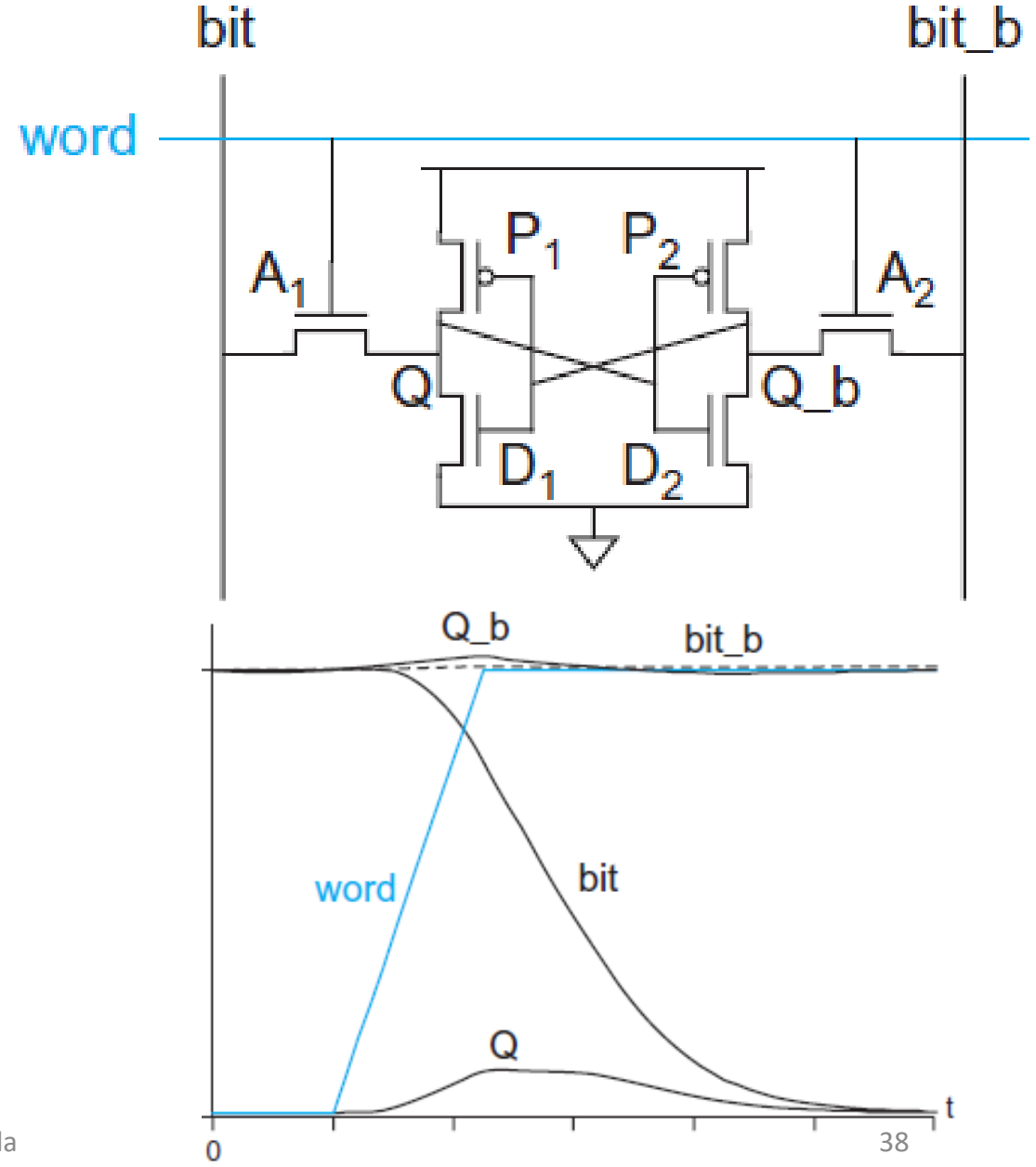
- SRAM operation is divided into two phases, generated from clk and its complement clkb.
- In phase 2, the SRAM is precharged/hold. (wordline→0)
- In phase 1, the SRAM is read or written. (wordline→1)

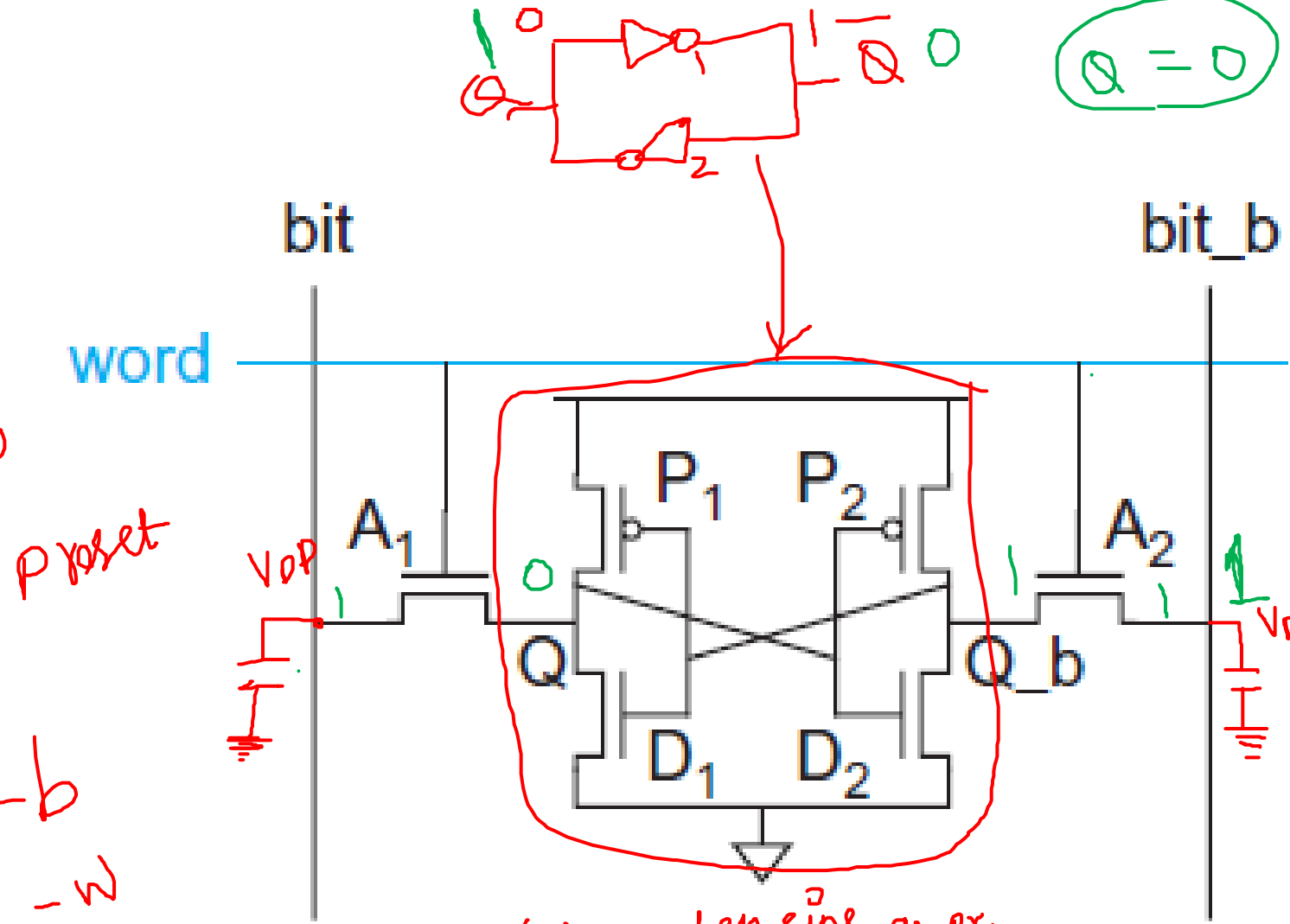
- The 6T SRAM cell contains a **pair of cross-coupled inverters** holding the state and a **pair of access transistors** to read or write the state.
- **Read:**
  - Bitlines are initially precharged high
  - Then one is pulldown by the SRAM cell through access transistors.
- **Write:**
  - Bitlines are actively driven low
  - This low values over powers the cell to write the new value

Memory cell Read/Write Operation

Read Stability:

- The bitlines are both initially floating high. Without loss of generality, assume Q is initially 0 and thus Q_b is initially 1. Q_b and bit_b both should remain 1.
- When the wordline is raised, bit should be pulled down through driver and access transistors D1 and A1.
- At the same time bit is being pulled down, node Q tends to rise. Q is held low by D1, but raised by current flowing in from A1. Hence, the driver D1 must be stronger than the access transistor A1.
- Specifically, the transistors must be ratioed such that node Q remains below the switching threshold of the P2/D2 inverter.
- This constraint is called **read stability.**
- Waveforms for the read operation are shown in Figure as a 0 is read onto bit.





word = 1
R & W

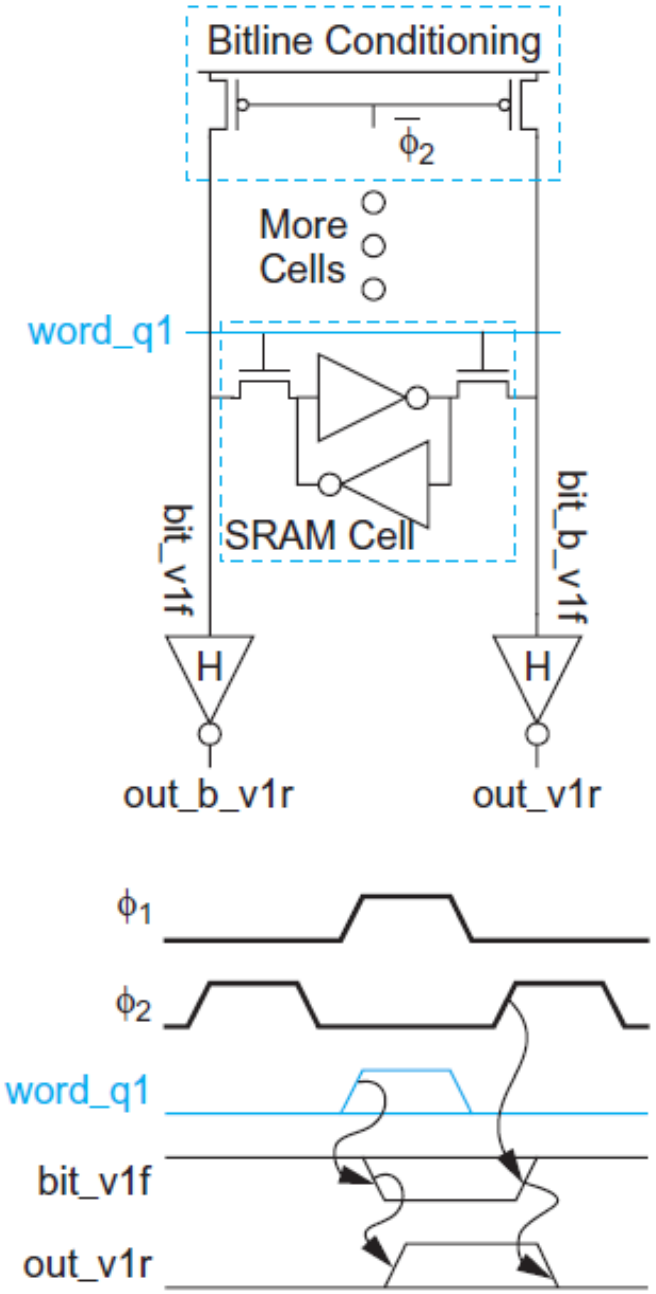
word = 0
hold / preset

bit bit_b
o/p lines - W
o/p lines - R

Q = 0 $\bar{Q}$ = 1 bit ↓ es o/p = 0

Read
1) Q = 1 $\bar{Q}$ = 0
2) word = 1
3) bit & bit_b o/p
4) Precharged to VDD
5) bit Vt → no change
bit_b Vt → cap discharge

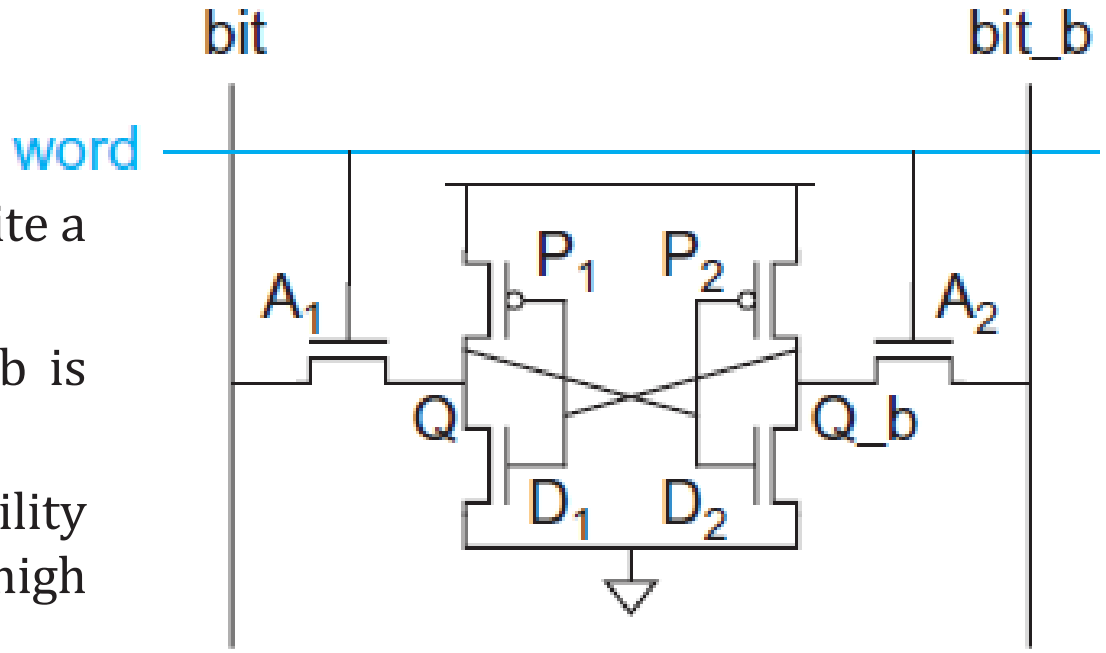
6) → sensing ampr. comp.
bit_b Vt ↓ es
o/p = 1 ✓
o/p =

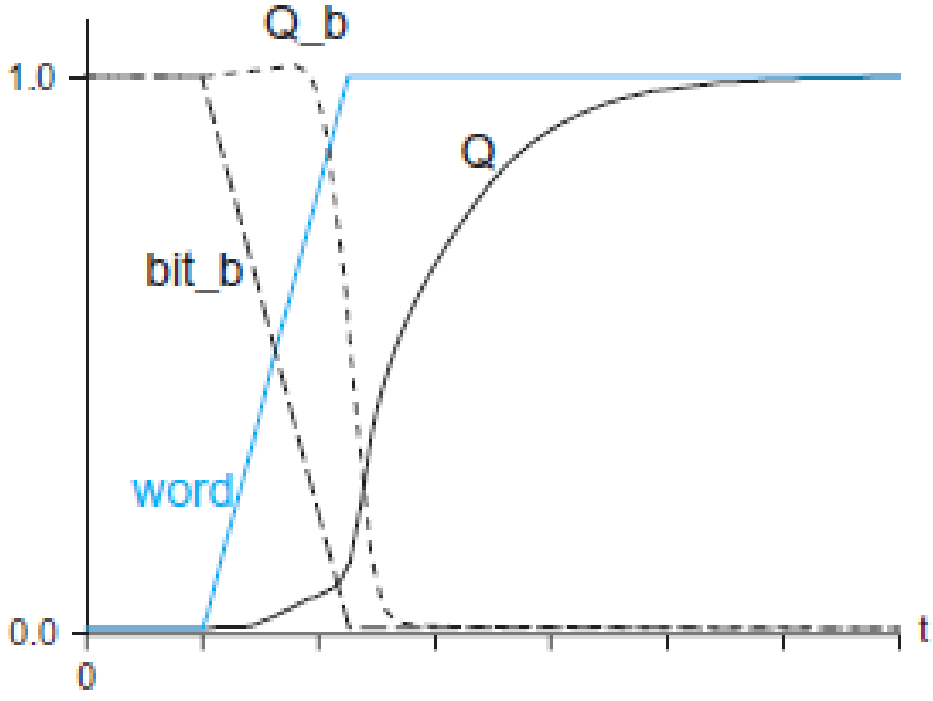


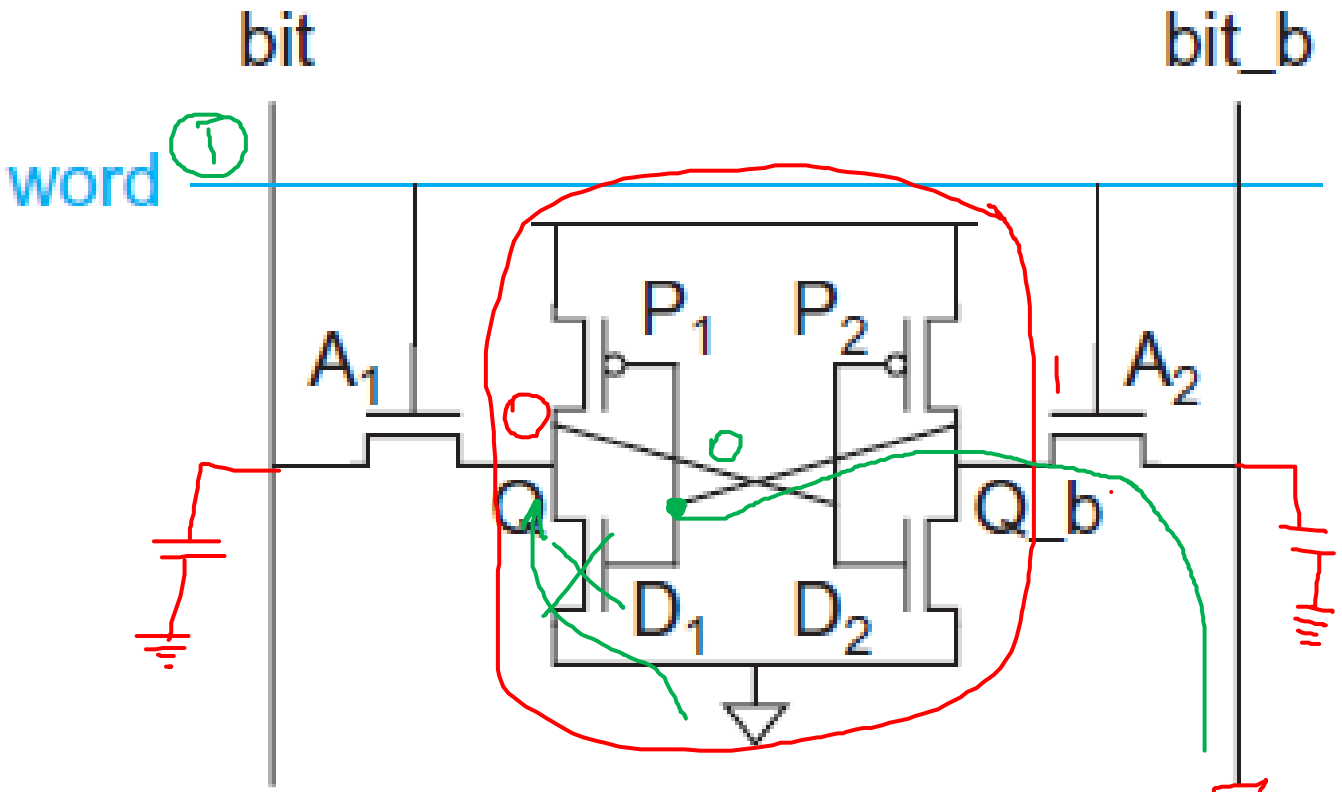
- same cell in the context of a full column from the SRAM.
- During phase 2, the bitlines are precharged high.
- The wordline only rises during phase 1; hence, it can be viewed as a _q1 qualified clock.
- Many SRAM cells share the same bitline pair, which acts as a distributed dual-rail footless dynamic multiplexer.
- The capacitance of the entire bitline must be discharged through the access transistor.
- The output can be sensed by a pair of HI-skew inverters.
- By raising the switching threshold of the sense inverters, delay can be reduced at the expense of noise margin.
- The outputs are dual-rail monotonically rising signals, just as in a domino gate.

Write Operation:

- assume Q is initially 0 and that we wish to write a 1 into the cell.
- bit is precharged high and left floating. bit_b is pulled low by a write driver.
- We know on account of the read stability constraint that bit will be unable to force Q high through A1.
- Hence, the cell must be written by forcing Q_b low through A2.
- P2 opposes this operation; thus, P2 must be weaker than A2 so that Q_b can be pulled low enough.
- This constraint is called **writability.**
- Once Q_b falls low, D1 turns OFF and P1 turns ON, pulling Q high as desired.







Q = 0
Q = 1

write

1) Q = 0 Q_b = 1

2) word = 1

3) bit & bit_b ⇒ i/p lines

4) bit_b ⇒ gnd write driver

5) V < Vth of D1

D1 → OFF P1 → ON

Q = 1

Cell Stability:

- To ensure both read stability and writability, the transistors must satisfy ratio constraints.
- The ***nMOS pulldown transistor*** in the cross-coupled inverters **must be strongest**.
- The ***access transistors*** are of **intermediate strength**
- The ***pMOS pullup transistors*** must be **weak.**
- To achieve good layout density, all of the transistors must be relatively small.
- For example,
    - pulldowns could be 8/2
    - access transistors 4/2
    - pullups 3/3.

DRAM

- Dynamic RAMs (DRAMs) ***store their contents as charge on a capacitor*** rather than in a feedback loop.
- Thus, the **basic cell** is substantially **smaller than SRAM**, but the cell must be periodically read and refreshed so that its contents do not leak away.
- **DRAM offer a factor of 10–20 greater density** (bits/cm2) than high-performance SRAM built in a standard logic process, **but they also have much higher latency.**

Read and write Operation

word = 1

Read

bit VL = $\frac{V_{dd}}{2}$

bit → o/p line / data line

M → ON

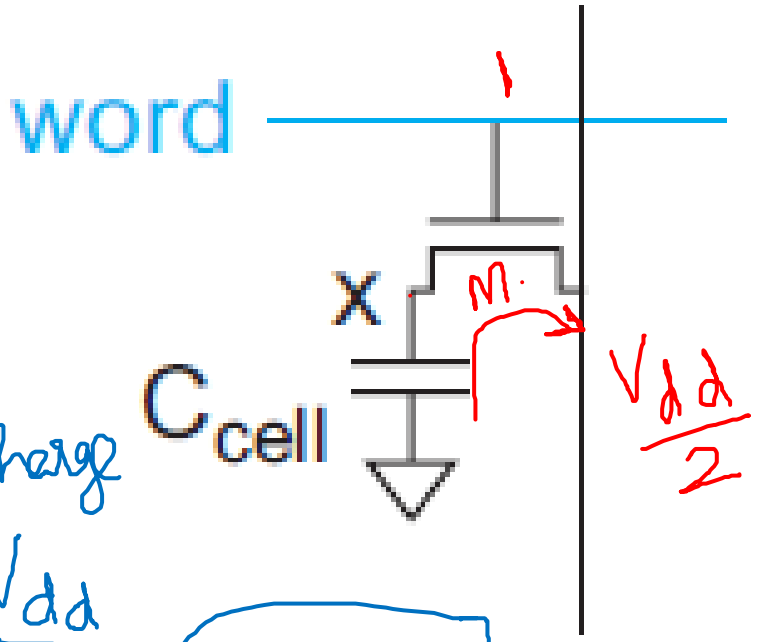
if C_{cell} = 1V = charged

bit VL = $V_{dd} + \frac{V_{dd}}{2}$ (V) > $\frac{V_{dd}}{2}$ → Sense amp → V > $\frac{V_{dd}}{2}$ → o/p 1

if C_{cell} = 0 = discharge

bit VL ≤ $\frac{V_{dd}}{2}$

o/p → 0



Read and write Operation

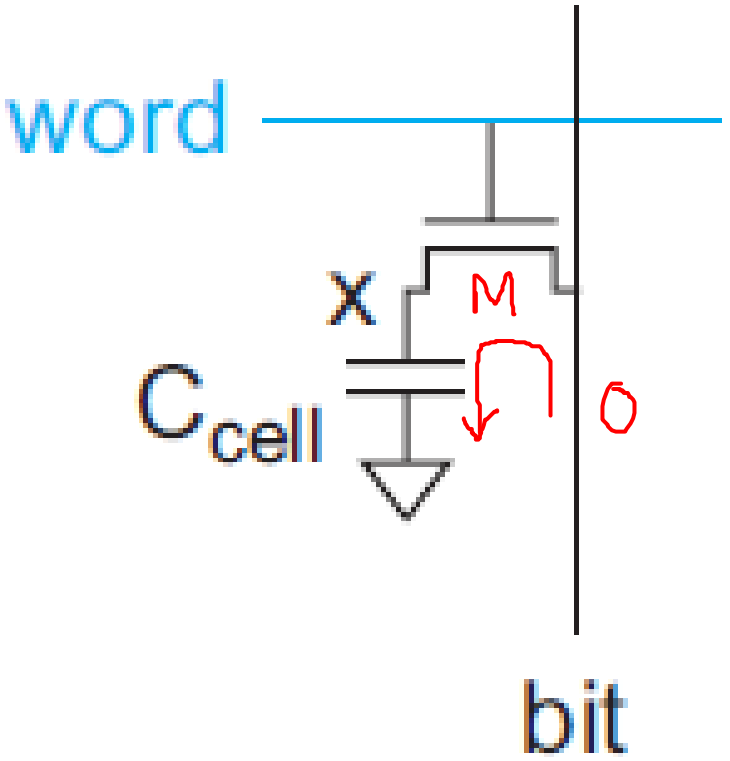
write

word = 1 M → ON

bit = i/p line

bit = 0 → C_{cell} = 0 discharging

bit = 1 → C_{cell} = V_{dd} charge



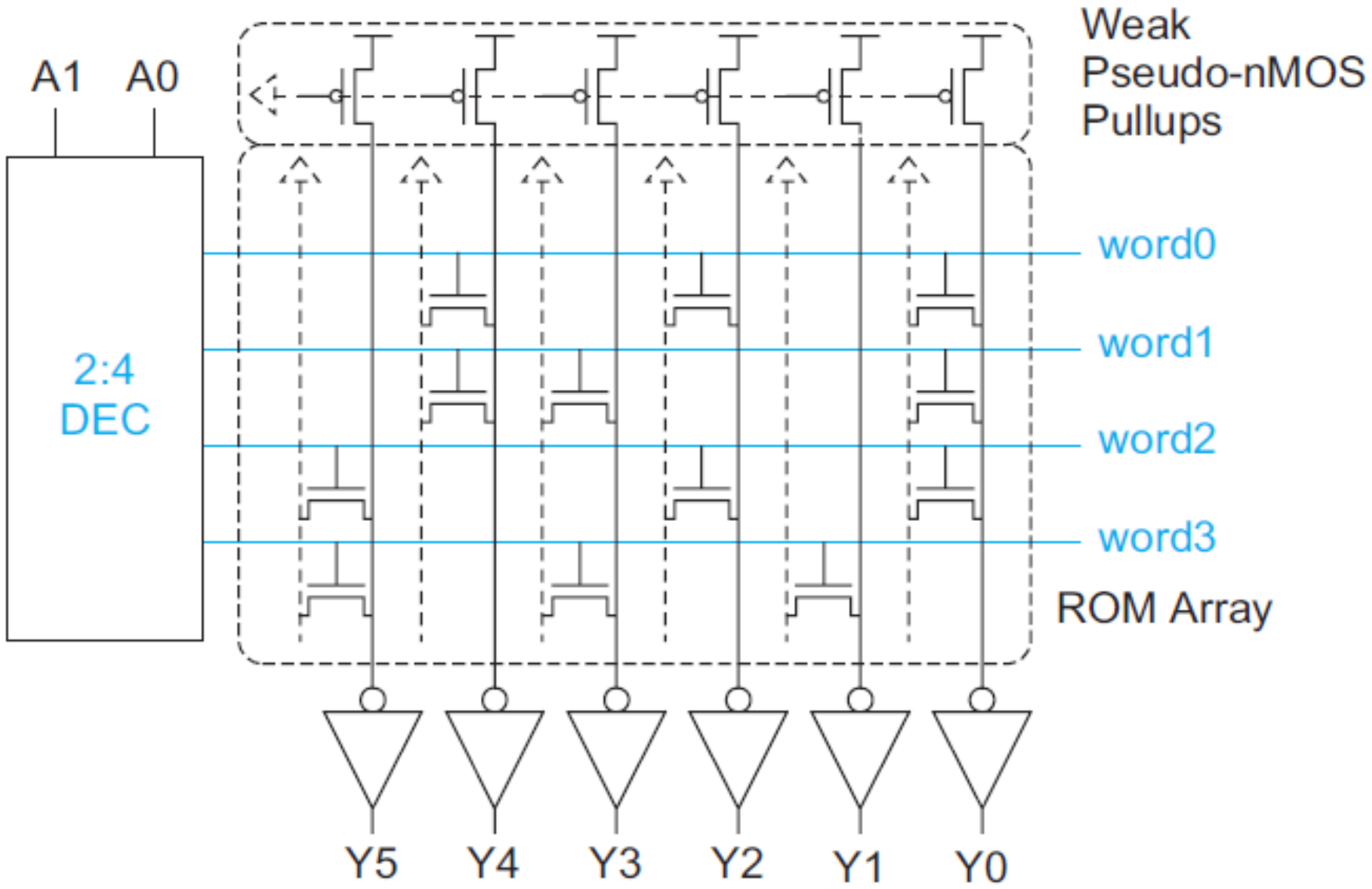
- The DRAM capacitor **Ccell must be as physically small** as possible to achieve good density.
- However, the **bitline is contacted to many DRAM cells** and has a relatively **large capacitance Cbit**.
- Therefore, the **cell capacitance is typically much smaller than the bitline capacitance.**
- According to the charge-sharing equation, the voltage swing on the bitline during readout is

$$\Delta V = \frac{V_{DD}}{2} \frac{C_{\text{cell}}}{C_{\text{cell}} + C_{\text{bit}}}$$

Read-Only Memory

- Read-Only Memory (ROM) cells can be **built with only one transistor per bit of storage.**
- A ROM is a **non-volatile memory structure** in that the state is retained indefinitely—even without power.
- A ROM array is commonly implemented as a **single-ended NOR array**.
    - Commercial ROMs are normally pseudo-nMOS and dynamic NOR
- **Advantages:**
    - DC power dissipation is acceptable
    - the speed is sufficient
    - easiest to design
    - requiring no timing
- The DC power dissipation can be significantly reduced in multiplexed ROMs by **placing the pullup transistors** after the column multiplexer.

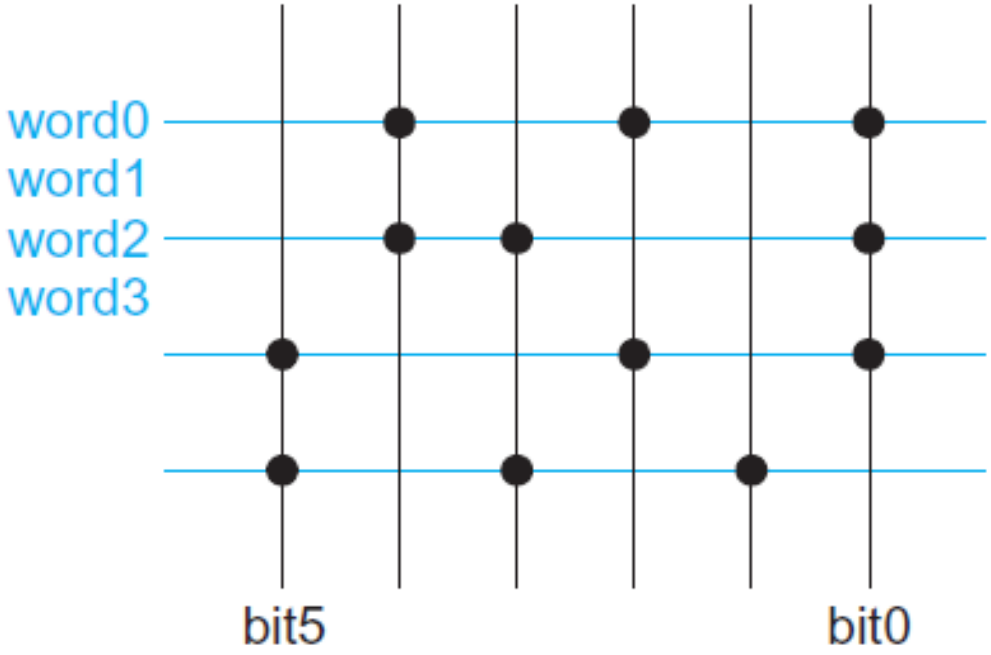
Pseudo-nMOS ROM



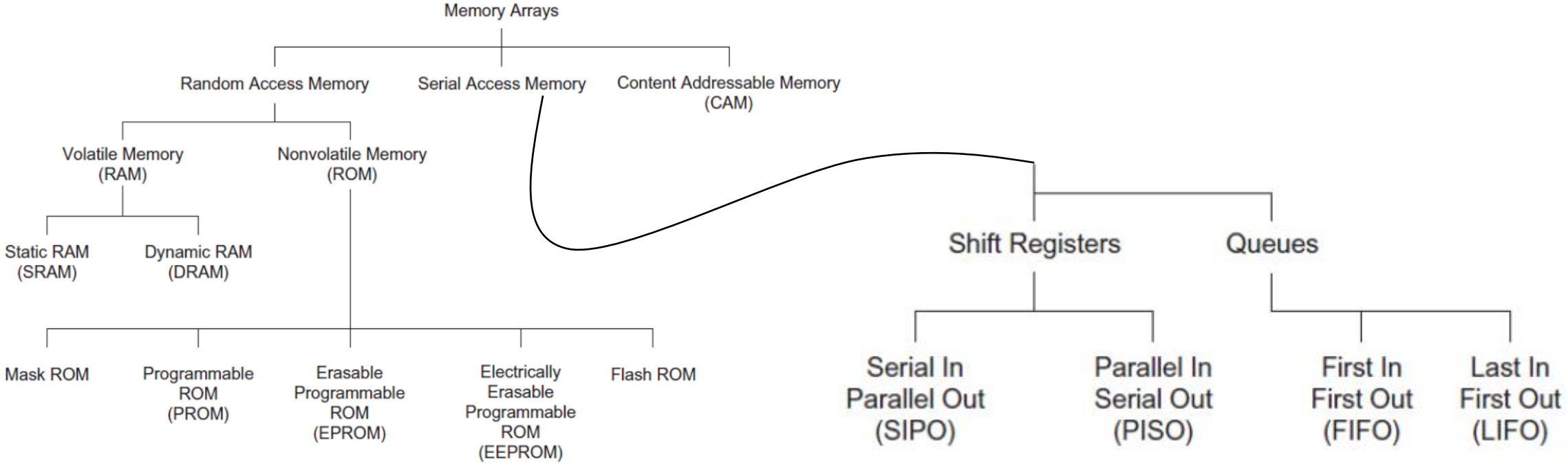
4 –word by 6 bit ROM

```
Word 0: 010101
Word 1: 011001
Word 2: 100101
Word 3: 101010
```

Dot diagram representation of ROM



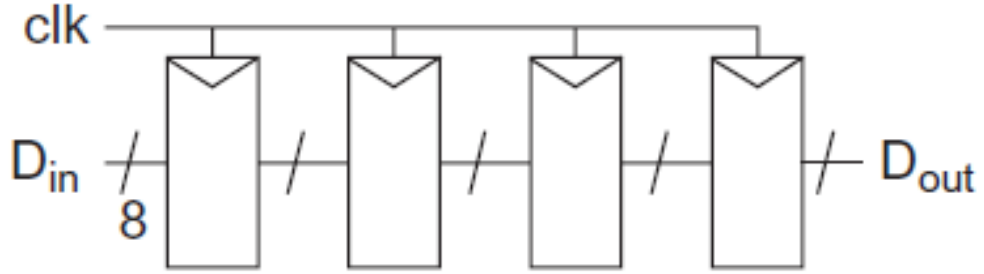
Serial Access Memories



- Using the basic **SRAM cell and/or registers**, we can **construct a variety of serial access memories** including shift registers and queues.
- These memories **avoid the need for external logic to track addresses for reading or writing.**

Shift Registers

- Commonly used in signal-processing applications to store and delay data



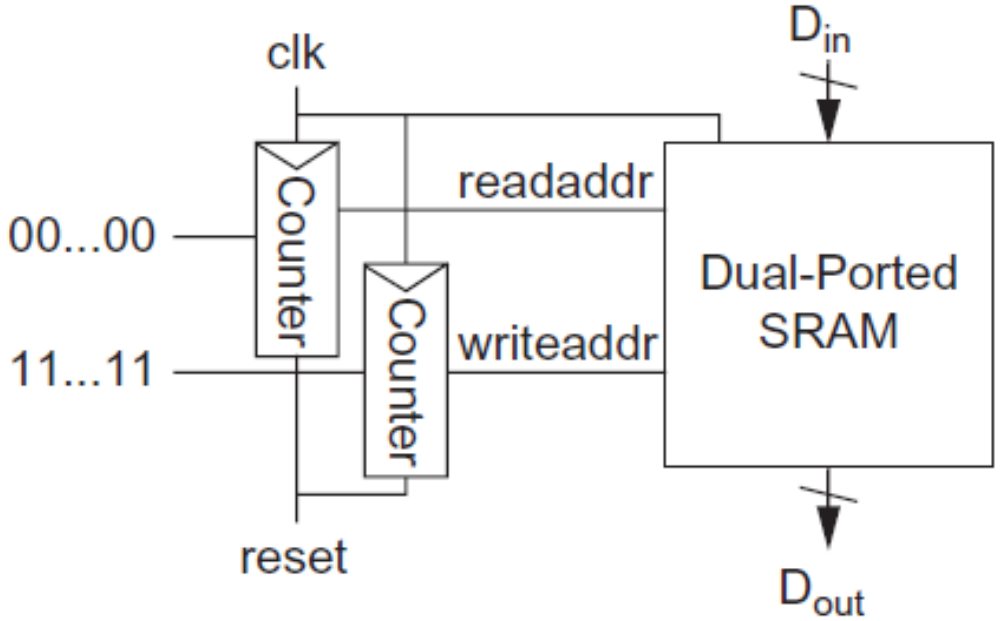
4-stage 8-bit shift register

constructed from 32 f.f's.

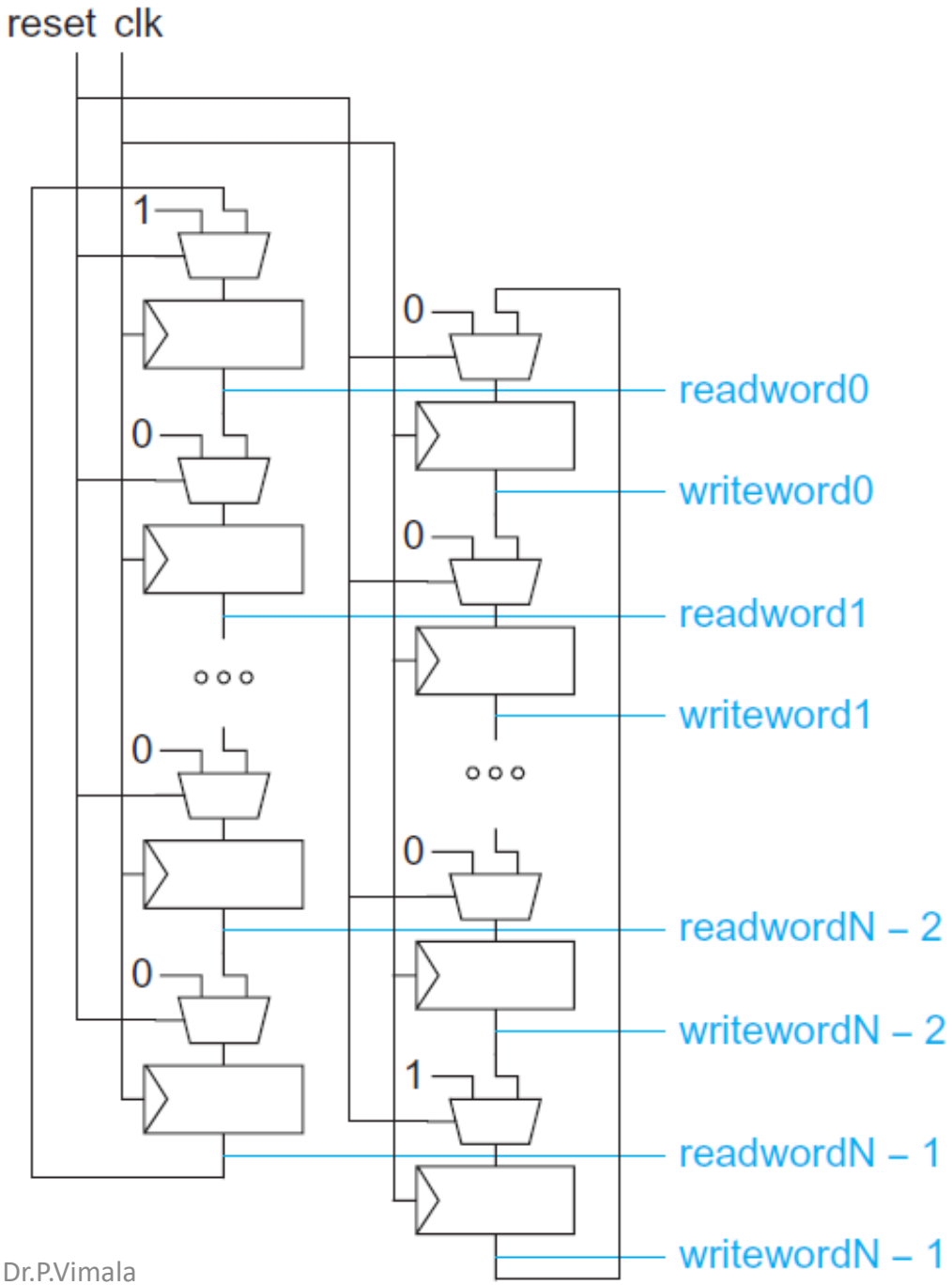
no logic between the registers, particular care must be taken that hold times are satisfied

- FF's are so large,big and dense **shift registers use dual-port RAMs**

The read counter is initialized to the first entry and the write counter to the last entry on reset



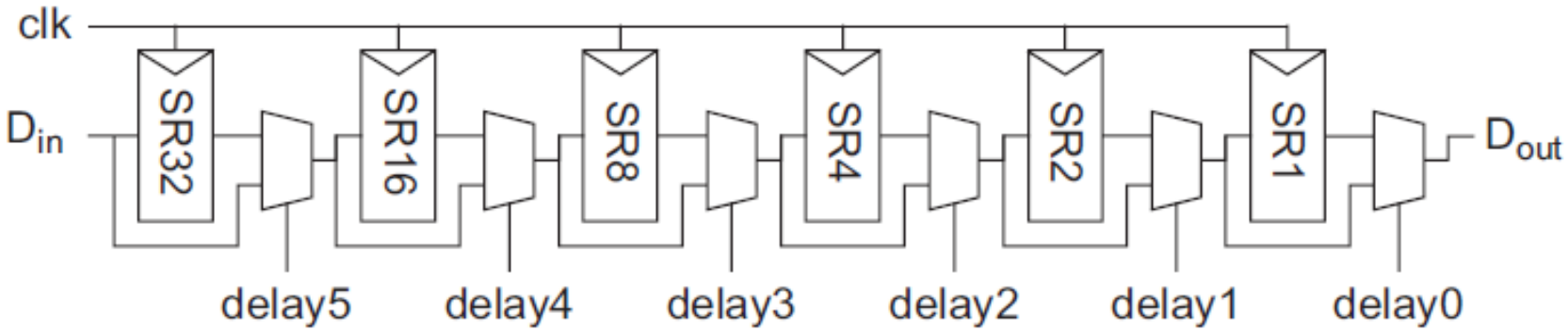
- Counters in an N-stage shift register can use **two 1-of-N hot registers to track which entries should be read and written.**
- Again, **one is initialized to point to the first entry** and the **other to the last entry.**
- These **registers can drive the wordlines** directly **without the need for a separate decoder**



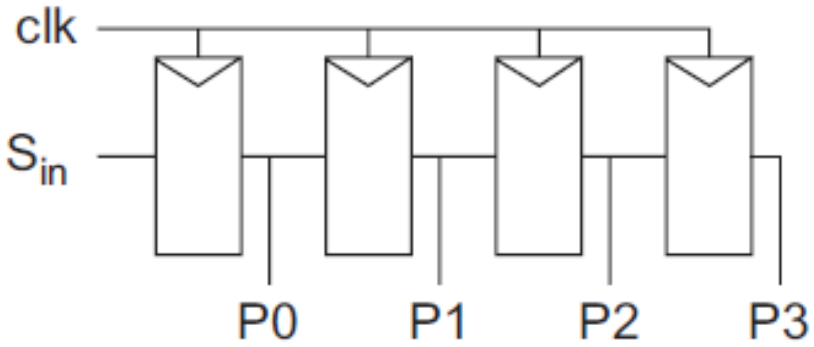
shift register variants

1) tapped delay line

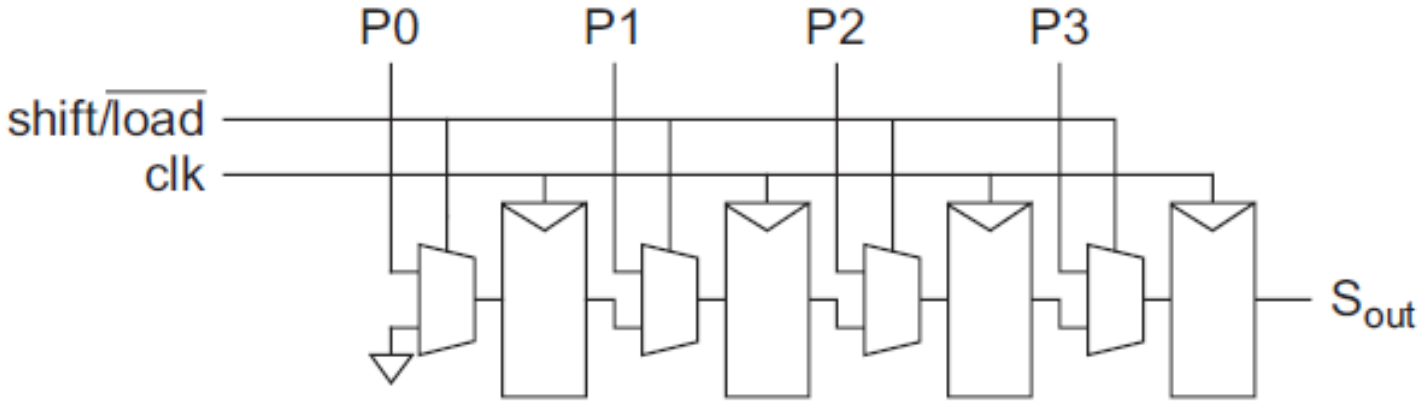
offers a variable number of stages of delay.



2) serial/parallel memory



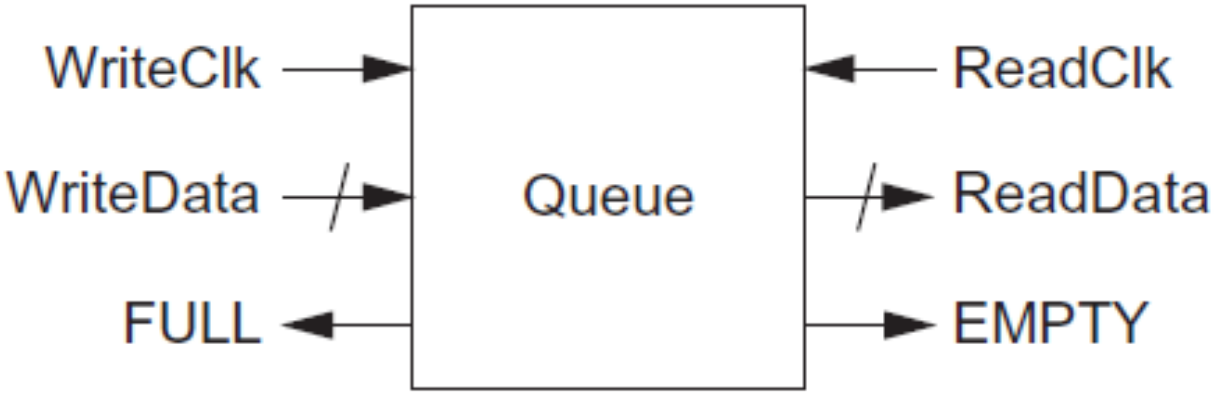
4-stage SIPO



4-stage PISO

Queues (FIFO, LIFO)

- Queues allow **data to be read and written at different rates.**
- The queue **internally maintains read and write pointers** indicating which data should be accessed next.
- As with a shift register, **the pointers can be counters** or **1-of-N hot registers.**



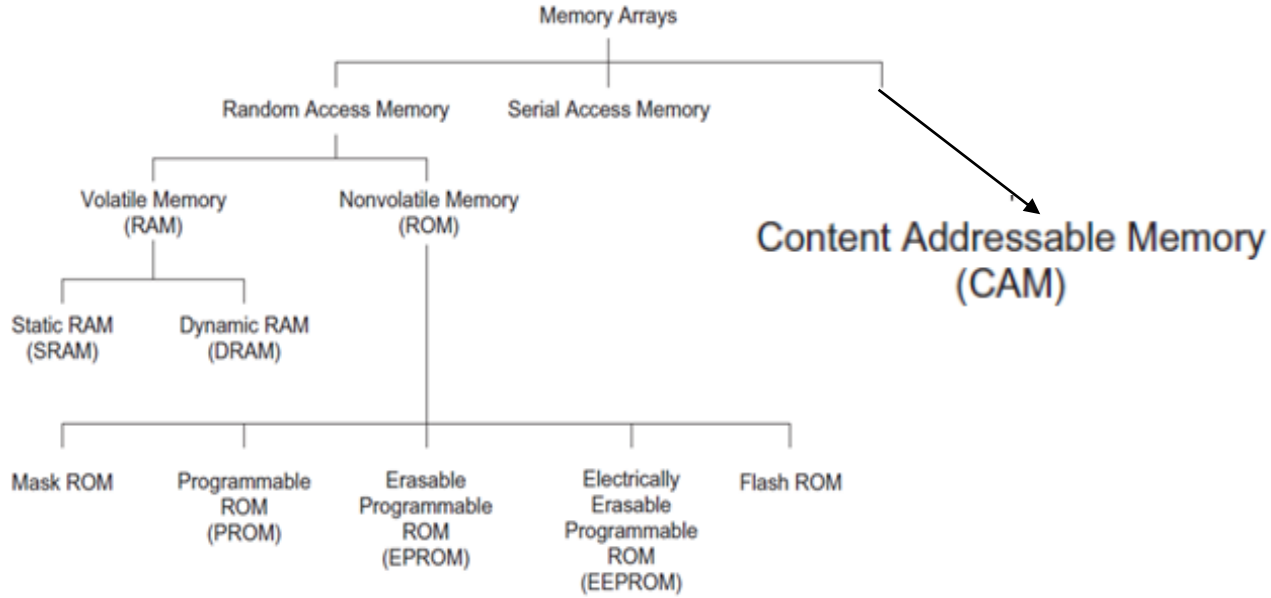
FIFO

- ***First In First Out (FIFO)*** queues are commonly used to buffer data between two asynchronous streams.
- Like a shift register, the FIFO is **organized as a circular buffer.**
- **On reset,** the read and write pointers are both initialized to the first element and the FIFO is EMPTY.
- **On a write**, the write pointer advances to the next element. If it is about to catch the read pointer, the FIFO is FULL.
- **On a read,** the read pointer advances to the next element. If it catches the write pointer, the FIFO is EMPTY again.

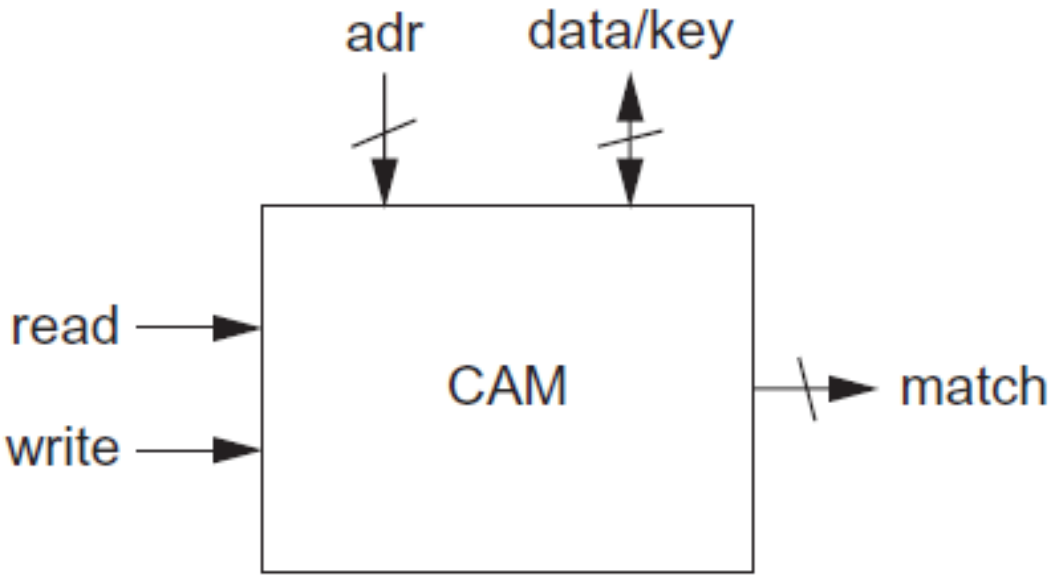
LIFO

- ***Last In First Out (LIFO)*** queues, also known as stacks, are used in applications such as subroutine or interrupt stacks in microcontrollers.
- The LIFO uses a **single pointer for both read and write.**
- **On reset,** the pointer is initialized to the first element and the LIFO is EMPTY.
- **On a write,** the pointer is incremented. If it reaches the last element, the LIFO is FULL.
- **On a read,** the pointer is decremented. If it reaches the first element, the LIFO is EMPTY again.

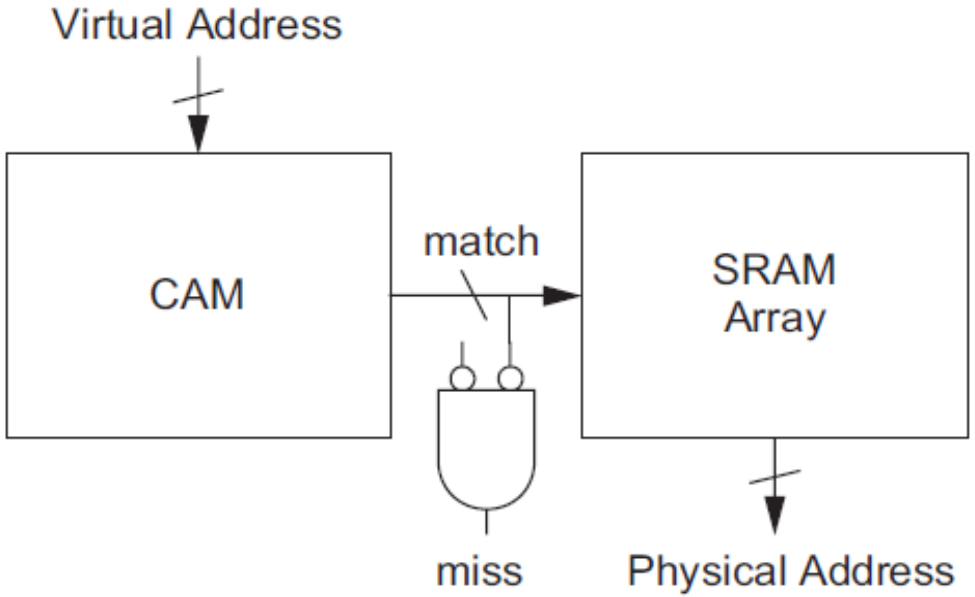
Content-Addressable Memory



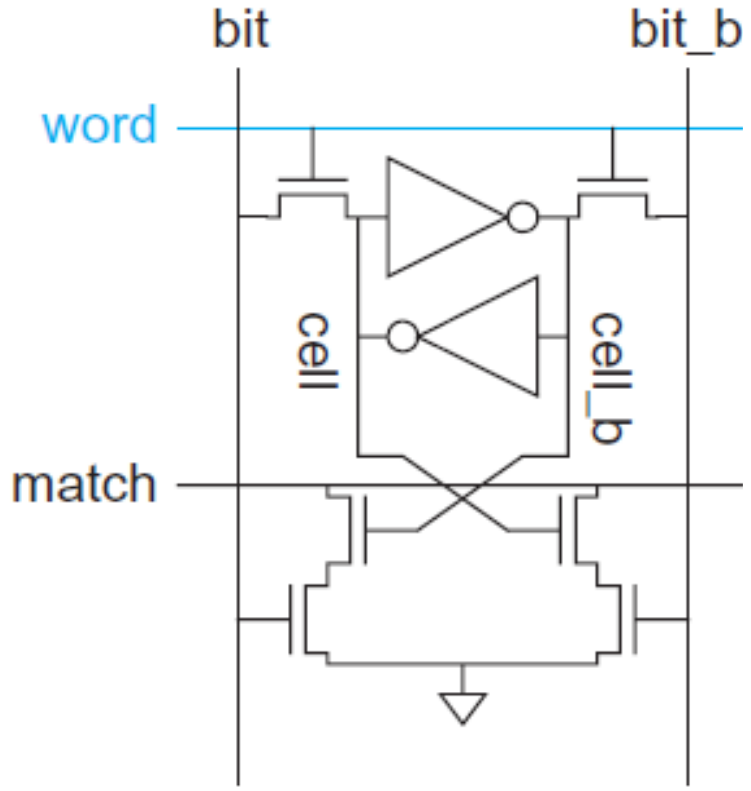
- The CAM acts as an ordinary SRAM that can be read or written given adr and data, but also performs ***matching operations.***
- Matching asserts a ***matchline*** output for each word of the CAM that contains a specified ***key.***



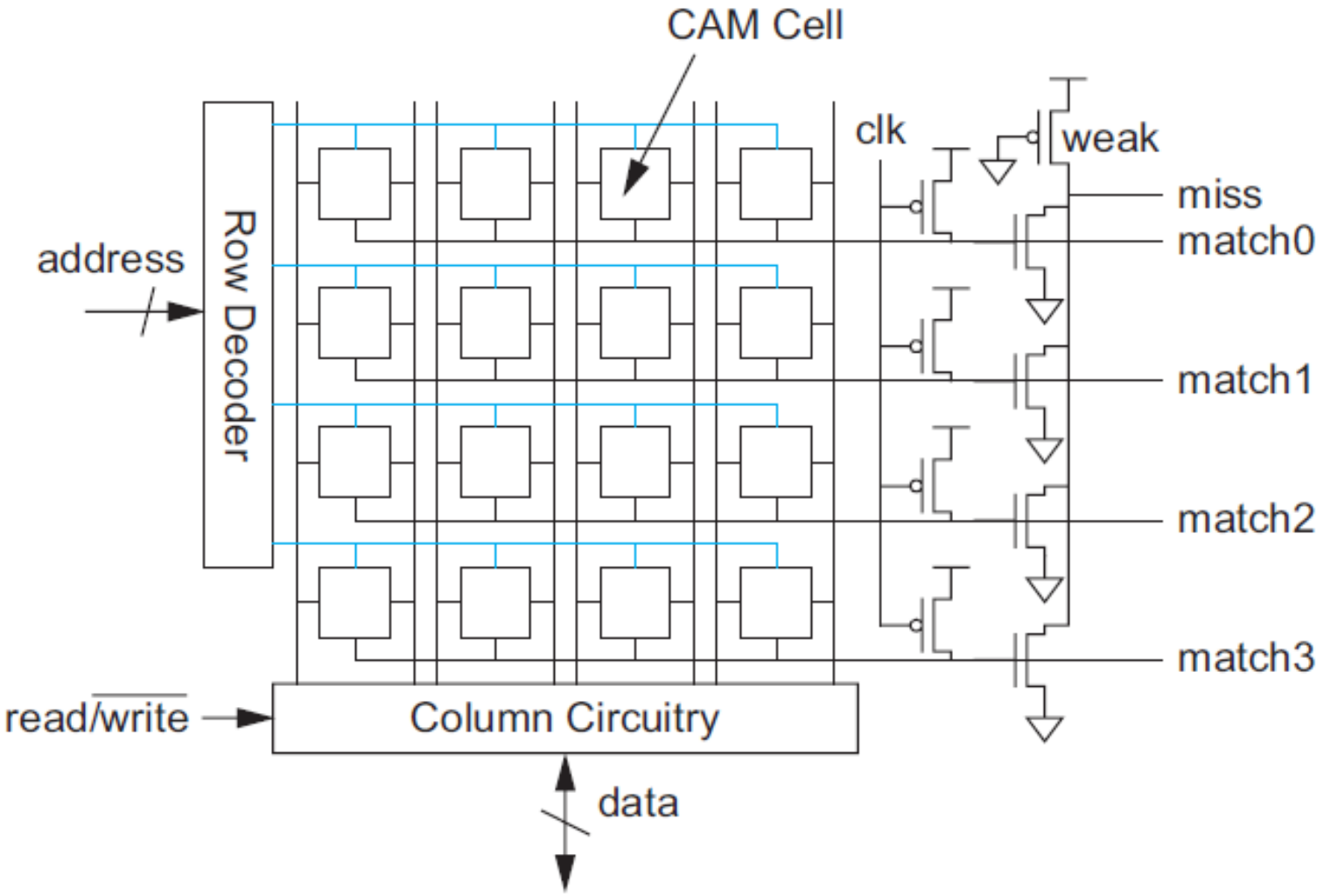
- Application of CAMs is **translation lookaside buffers (TLBs)** in microprocessors
- The **virtual address is given as the key** to the TLB CAM.
- **If this address is in the CAM**, the corresponding **matchline is asserted**.
- This **matchline can serve as the wordline** to access a RAM containing the associated physical address.
- A **NOR gate** processing all of the match lines **generates a miss signal** for the CAM.

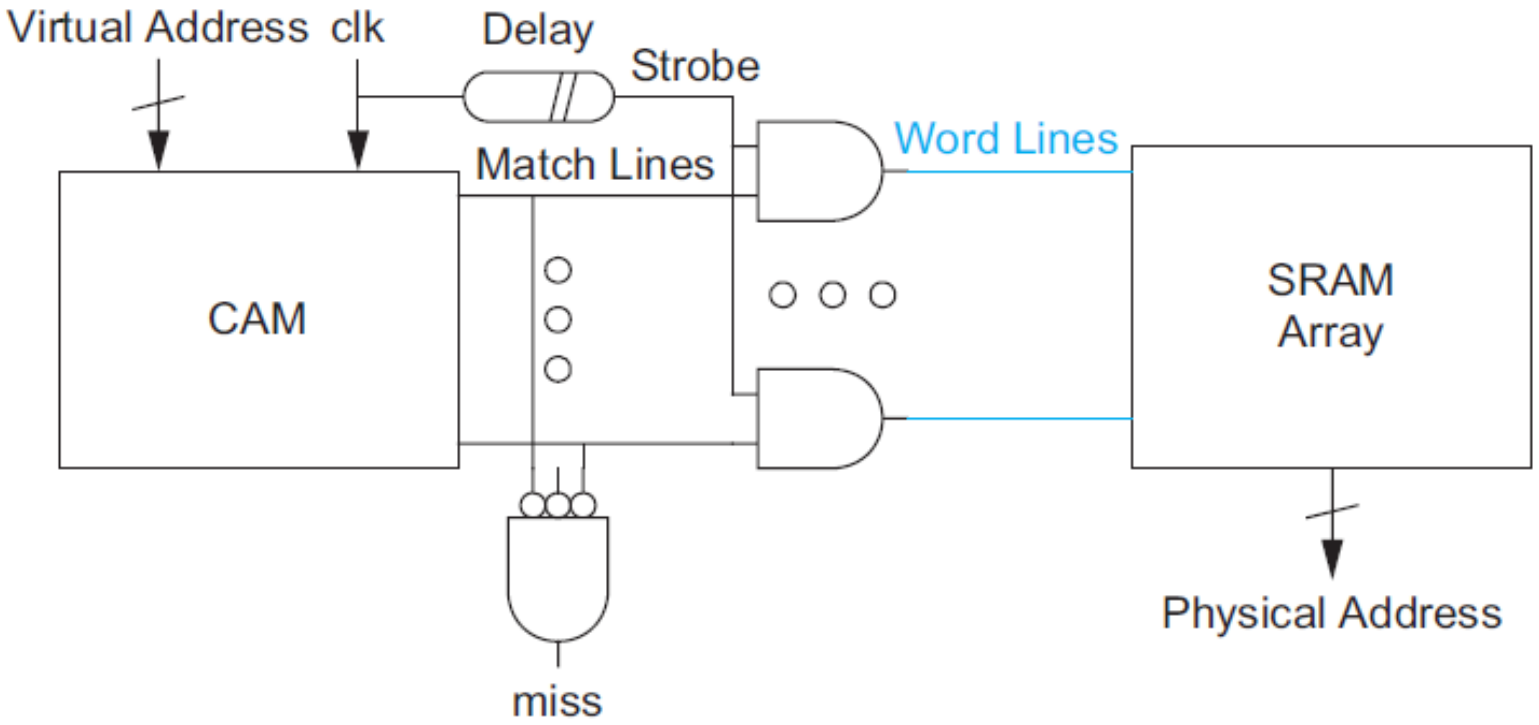


- **10T CAM cell** consisting of a **normal SRAM cell with additional transistors to perform the match.**
- Multiple CAM cells in the same word are tied to the same matchline.
- The matchline is either precharged or pulled high as a distributed pseudo-nMOS gate.
- The key is placed on the bitlines. If the key and the value stored in the cell differ, the matchline will be pulled down.
- Only if all of the key bits match all of the bits stored in the word of memory will the matchline for that word remain high.
- The key can contain a “don’t care” by setting both bit and bit_b low.



4x4 CAM array





- Refined TLB path with monotonic wordlines
- Strobe can be timed with in inverter chain or replica delay
- Consume relatively large amounts of power because the bitlines and matchlines may all transistors during parallel search.