

DIGITAL ELECTRONICS: ECE 213

Topic: Computer Memory || RAM vs
ROM || SRAM vs DRAM

**UNIT VI: MEMORY AND
CONVERTERS**

Lecture No.: 40

Prepared By: Irfan Ahmad Pindoo

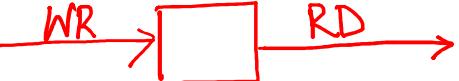
Assistant Professor

VLSI Design, ECE

School of Computer Science and Engineering



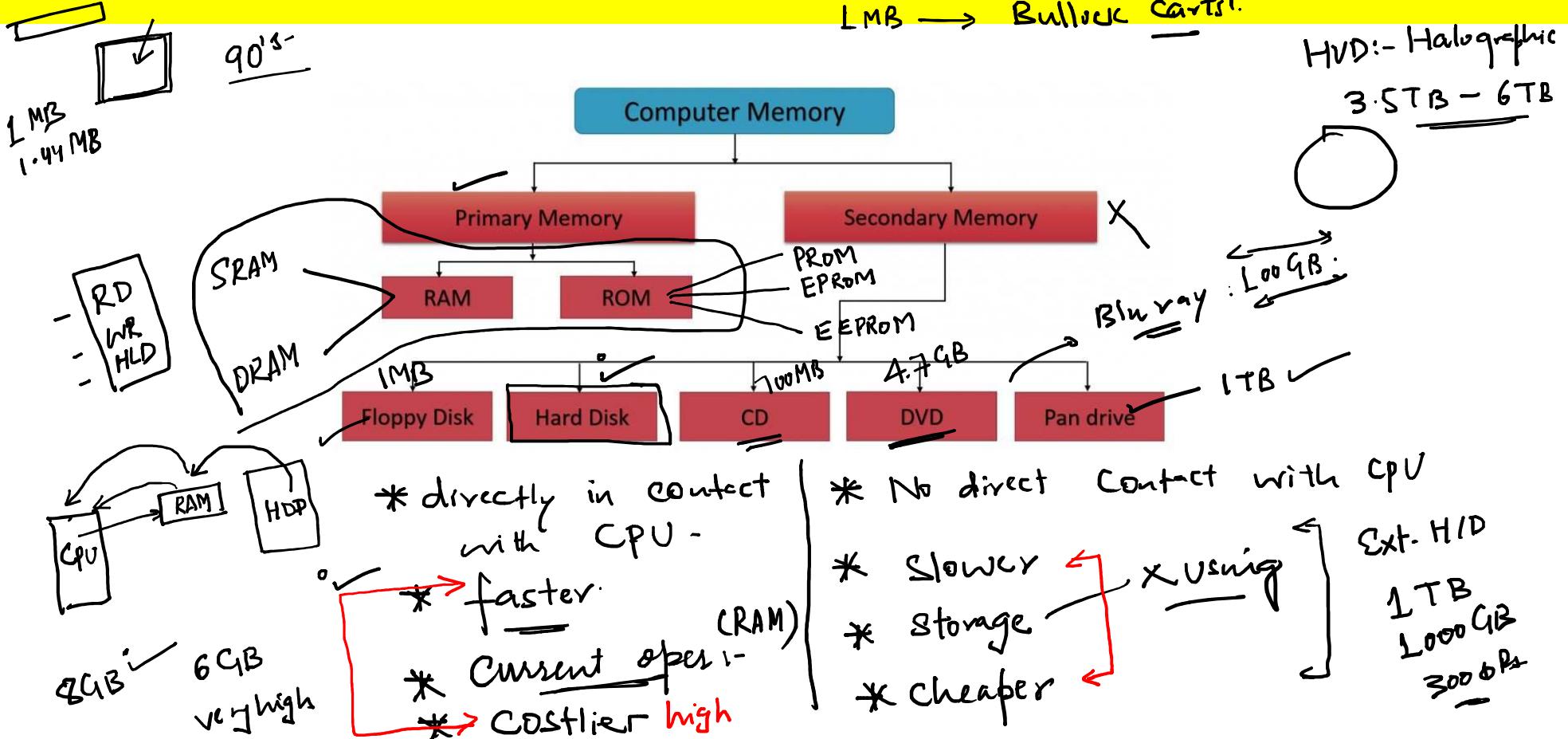
Introduction

- A memory unit is a device to which binary information is transferred for storage and from which information is retrieved when needed for processing.
- Binary information received from an input device is stored in memory, and information transferred to an output device is taken from memory.
- A memory unit is a collection of cells capable of storing a large quantity of binary information

Types of Memories

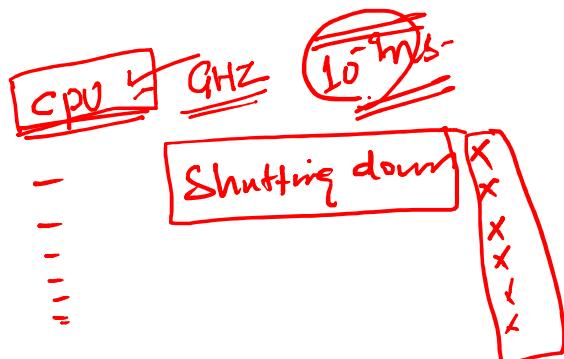
- RAM (Random Access Memory)
Performs read as well as write operation.
- ROM (Read Only Memory)
Performs only read operation.

Types of Memories



Differences between RAM and ROM

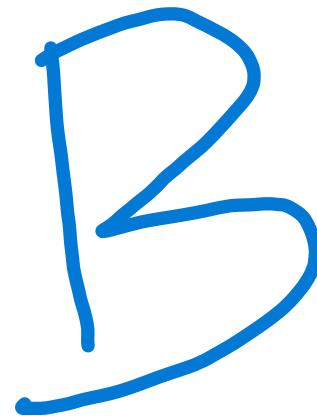
RAM (Random Access Memory)	ROM (Read Only Memory)
Read-Write	Read Only \leftrightarrow Write
Volatile \rightarrow Power dependent	Non Volatile : BIOS
Data can be modified \rightarrow Write	Data cannot be modified
Higher Speed \rightarrow	Slow \times
Costlier \rightarrow	Cheaper \rightarrow
Types of RAM: <u>SRAM</u> and <u>DRAM</u>	Types of ROM: <u>PROM</u> , <u>EPROM</u> , <u>EEPROM</u>



QUICK QUIZ (POLL)

Identify the wrong statement?

- a) RAM is volatile ✓
- b) RAM is cheaper ✗
- c) Write operation can also be performed by RAM. ✓
- d) High speed available in RAM ✓



Types of RAM

1. SRAM (Static Random Access Memory)
2. DRAM (Dynamic Random Access Memory)

Differences between SRAM and DRAM

Parameter	SRAM Static	DRAM Dynamic.
→ Construction	Complex	Simple
→ Number of Transistors Required	6 ✓	$1 Q_r + \text{Capacitor}$
→ Charge Leakage	No	Yes
→ Refreshing	Not Required	Needed periodically
→ Speed	Faster ✓	Slower ✓
→ Size	Smaller $Q_r \rightarrow nm \rightarrow 14 nm^2$	Larger
→ Cost	Expensive ✓	Cheap ✓
→ Application	Cache Memory SRAM	Main Memory
→ Power Consumption	Higher 6	Lower ✓
→ Density	Low	High
→ Capacity	Low $256 KB$ $512 KB$	High $MB CR$ $1000 Chrom PPT$

Diagram illustrating the differences:

- SRAM (Static RAM):** Shows a capacitor symbol with a 'time constant' of $37 f$. It is labeled 'Static'.
- DRAM (Dynamic RAM):** Shows a capacitor symbol with 'Charging & Discharging' indicated by arrows. It is labeled 'Dynamic.'
- SRAM Applications:** Cache Memory (labeled 'Larger').
- DRAM Applications:** Main Memory, Cache memory, RAM, ROM, and Hard Disk (HDD).
- Power Consumption:** SRAM has higher power consumption (6), while DRAM has lower power consumption.
- Capacity:** SRAM has low capacity (e.g., 256 KB, 512 KB), while DRAM has high capacity (e.g., MB CR, 1000 Chrom PPT).
- Speed:** SRAM is faster than DRAM.
- Cost:** SRAM is expensive, while DRAM is cheap.
- Size:** SRAM is smaller (e.g., 14 nm²), while DRAM is larger.
- Construction:** SRAM is complex, while DRAM is simple.
- Transistors:** SRAM requires 6 transistors, while DRAM requires one transistor plus a capacitor.
- Refresh:** DRAM needs periodic refresh, while SRAM does not require it.
- Leakage:** SRAM has no charge leakage, while DRAM has charge leakage.

QUICK QUIZ (POLL)

Number of transistors required in SRAM are:

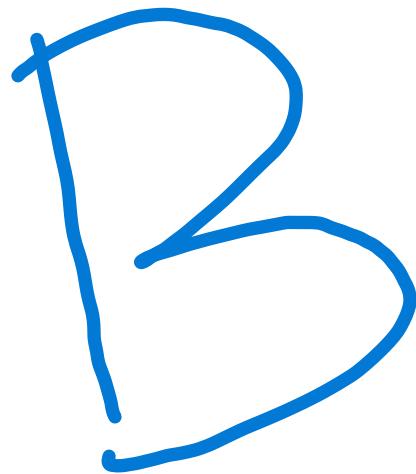
- a) 1
- b) 3
- c) 6
- d) 1 with 1 capacitor



QUICK QUIZ (POLL)

Periodic refreshing is required in:

- a) SRAM
- b) DRAM
- c) EEPROM
- d) PLA



DIGITAL ELECTRONICS: ECE 213

Topic: SRAM: READ, WRITE and HOLD OPERATION

UNIT VI: MEMORY AND CONVERTERS

Lecture No.: 41

Prepared By: Irfan Ahmad Pindoo

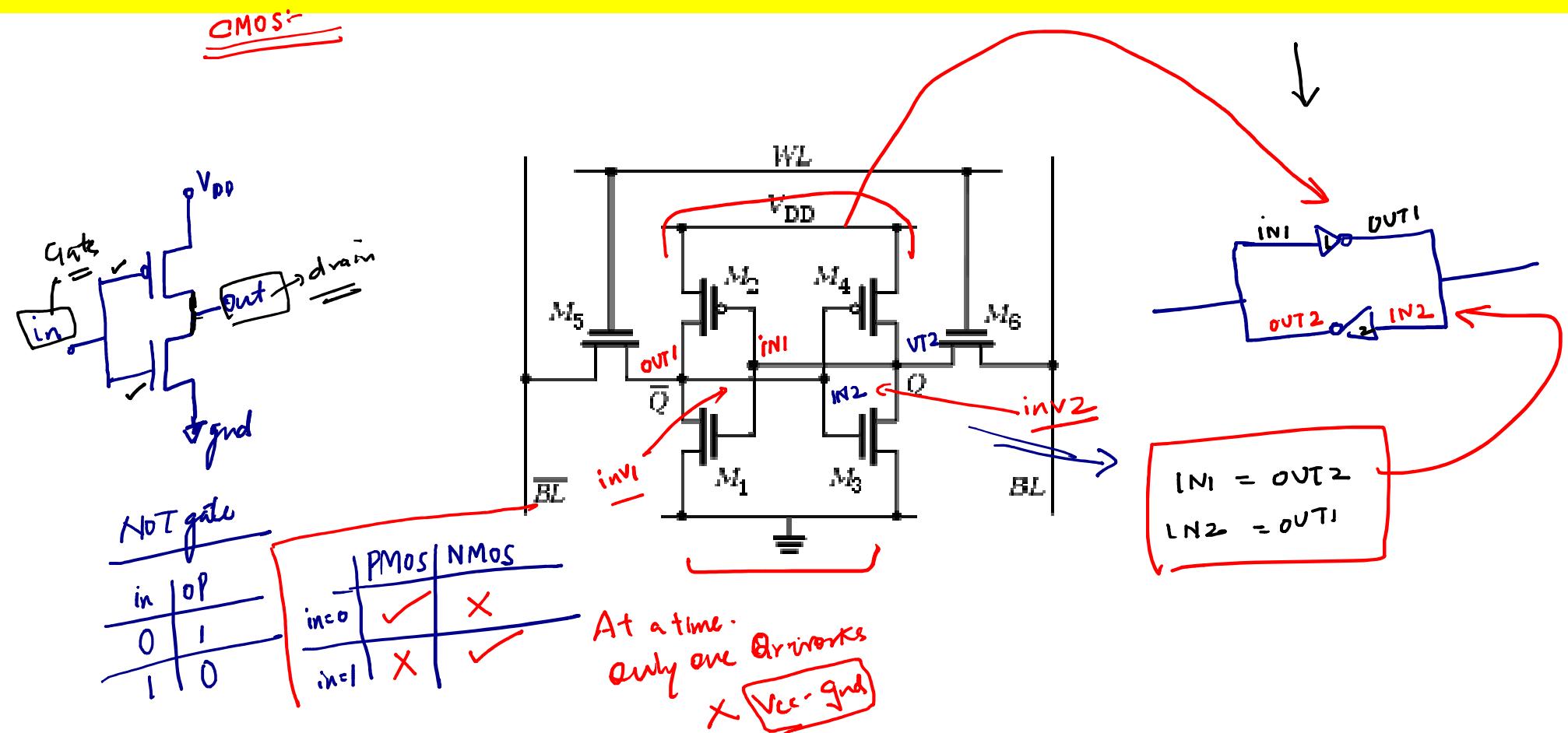
Assistant Professor

VLSI Design, ECE

School of Computer Science and Engineering



Working of SRAM



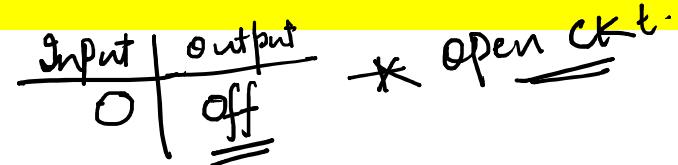
Delivered By: Irfan Ahmad Pindoo

1/0 Bits

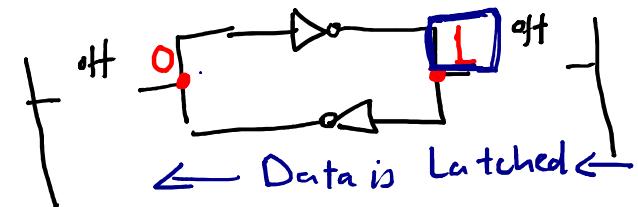
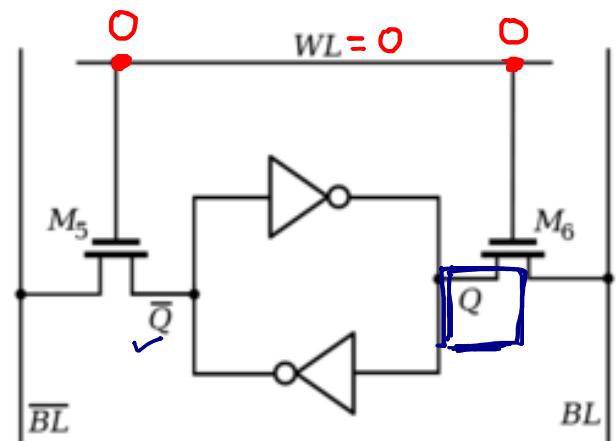
Working of SRAM

- Hold \Rightarrow data restored
- WL: Word Line
- BL: Bit Line.

WL \swarrow M₅ (NMOS Q_{rs})
WL \swarrow M₆ (NMOS Q_{rs})



- ✓ 1. WL = 0
- ✓ 2. Data is held in the latch mode
 - Read [available \longrightarrow retrieve]
- ✓ 1. WL = 1
2. Access transistors are turned ON
3. BL and \overline{BL} values are read by Sense Amplifier



Working of SRAM

allowing

- 1. Read
- 2. $WL = 1$
- 3. Access transistors are turned ON
- 4. BL and \overline{BL} values are read by Sense Amplifier

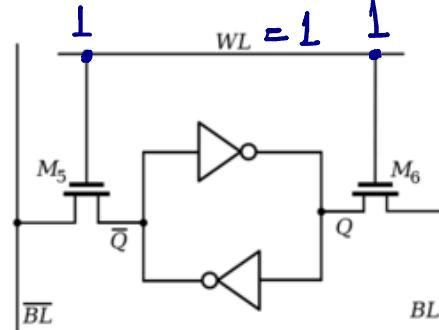
M_5, M_6 : NMOS
Input 1 | Output ON \Rightarrow short circuit

$$SV \frac{1}{T} i = \frac{V}{R} \frac{1}{T} SV$$

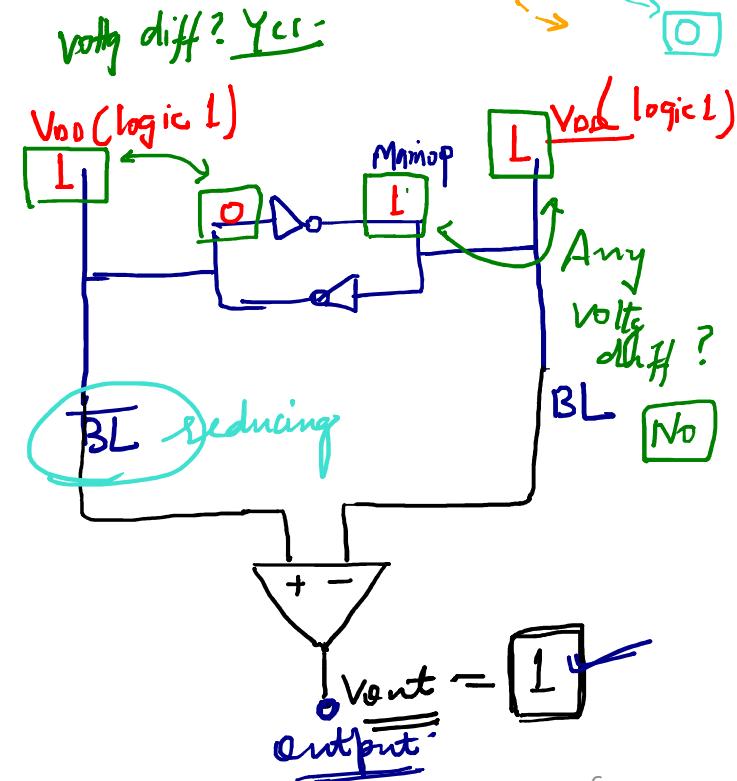
trend

BL & \overline{BL} :-

- 1) Purpose : operation faster.
- 2) Precharge BL and \overline{BL} with certain voltage
- 3) Value of $BL = \overline{BL}$ \Rightarrow same decrease.



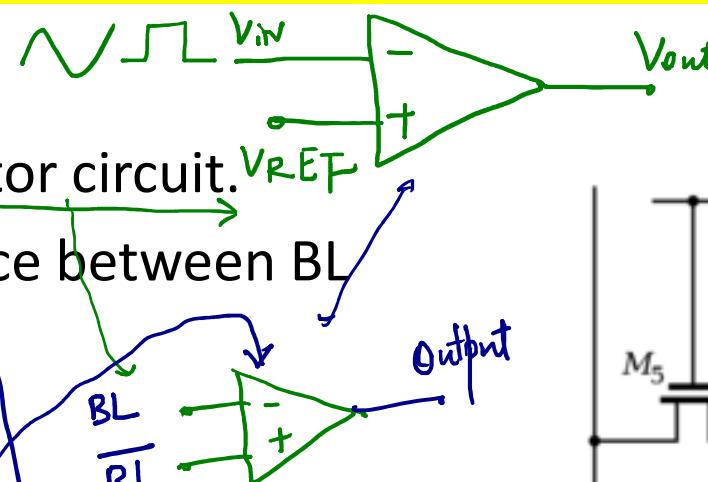
- 4) If $BL > \overline{BL}$: out = 1
- 5) $BL > \overline{BL}$



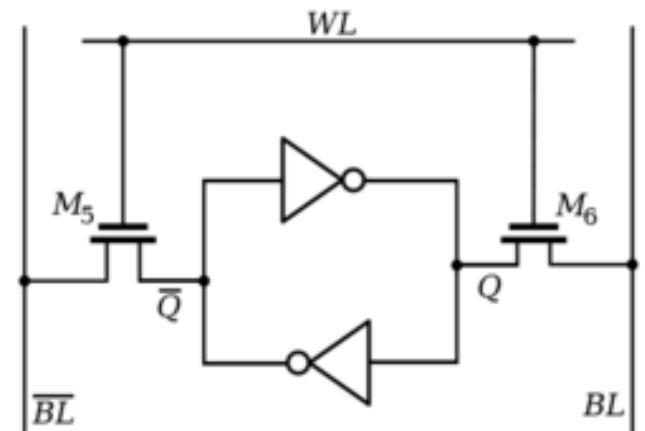
Working of SRAM

Sense Amplifier

- It is an op amp comparator circuit.
- It compares the difference between BL and \overline{BL}
- If $BL > \overline{BL}$, output is 1.
- If $BL < \overline{BL}$, output is 0.
- Advantage: It sets the output quickly, without fully charging/discharging.



$V_{in} > V_{ref}$:
 $V_{in} < V_{ref}$:

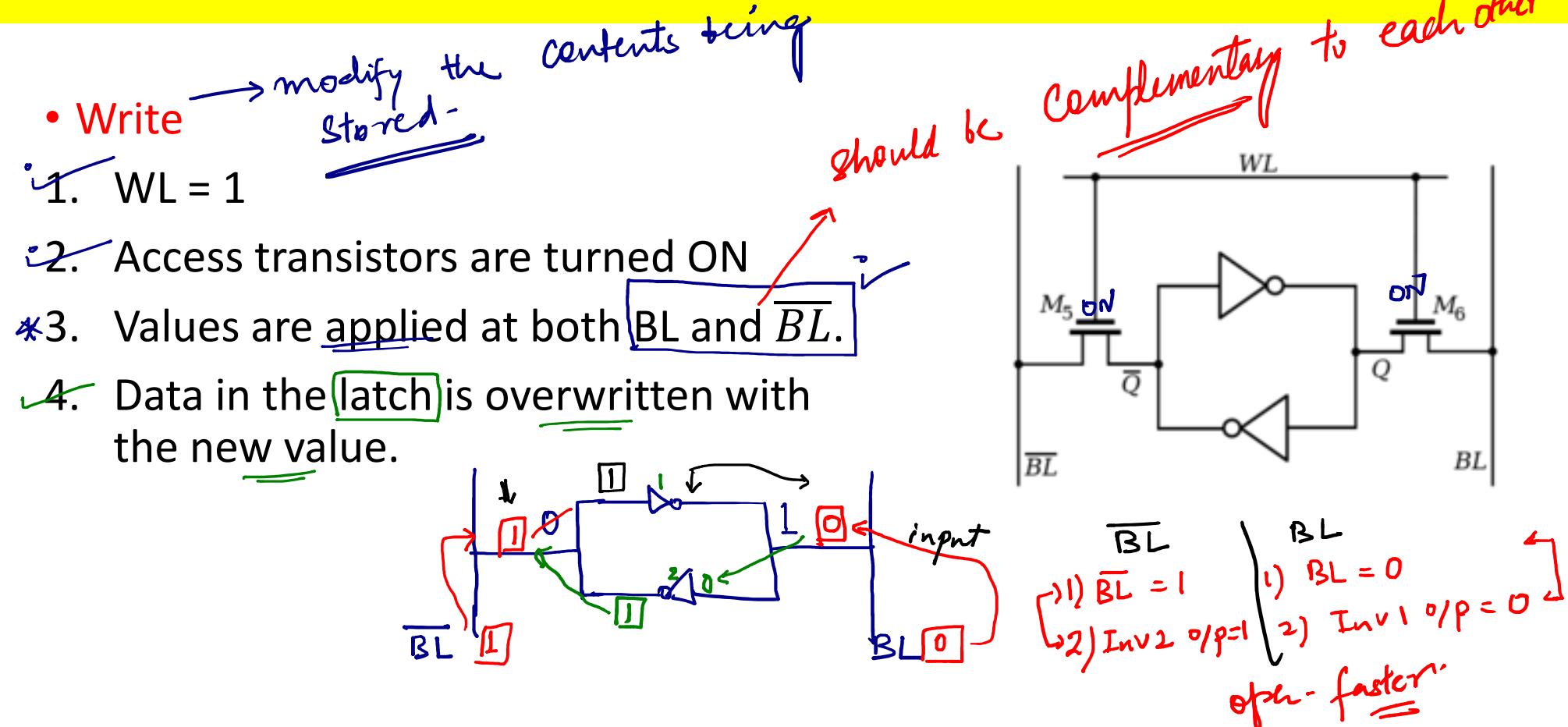


Quick Quiz POLL

For a read operation WL must be

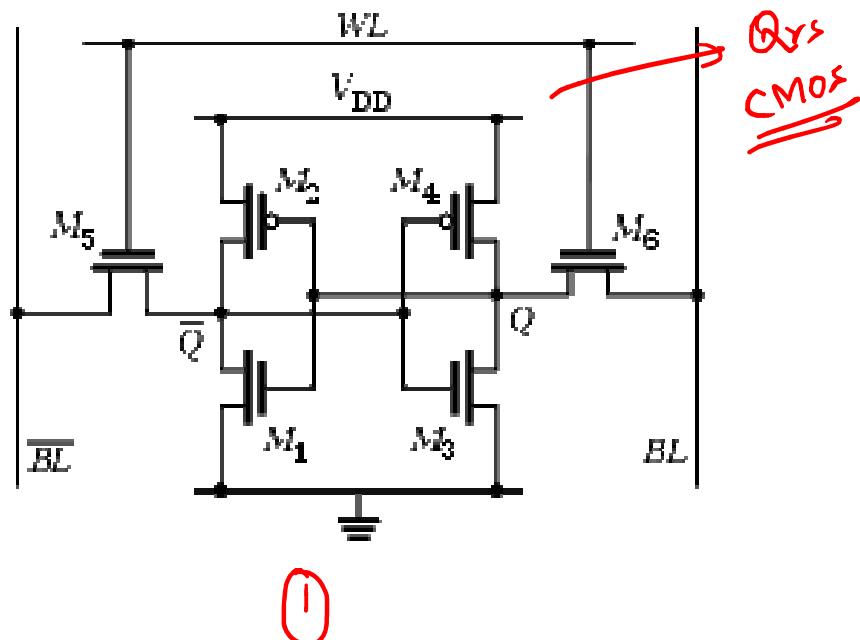
- a) 0
- b) 1 ✓
- c) Applied with clock
- d) None of these

Working of SRAM

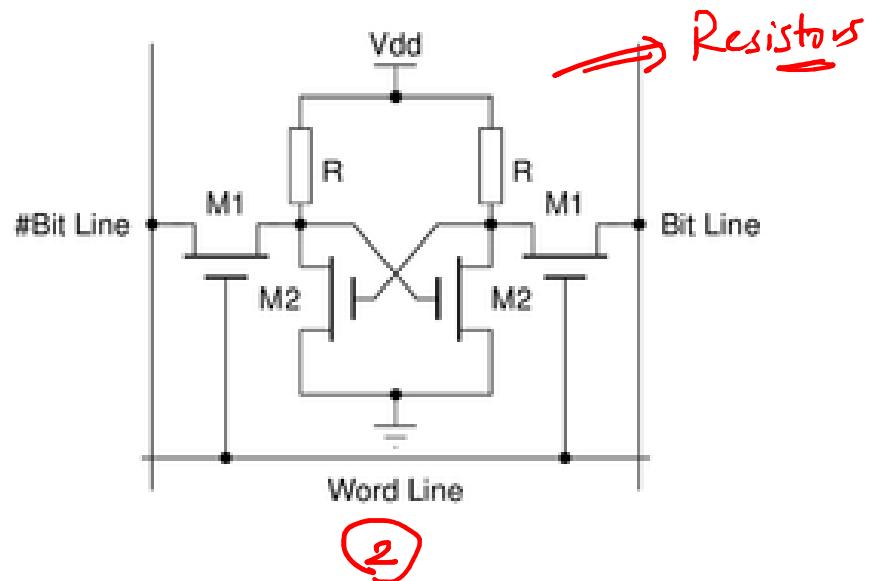


6T vs 4T Cell SRAM

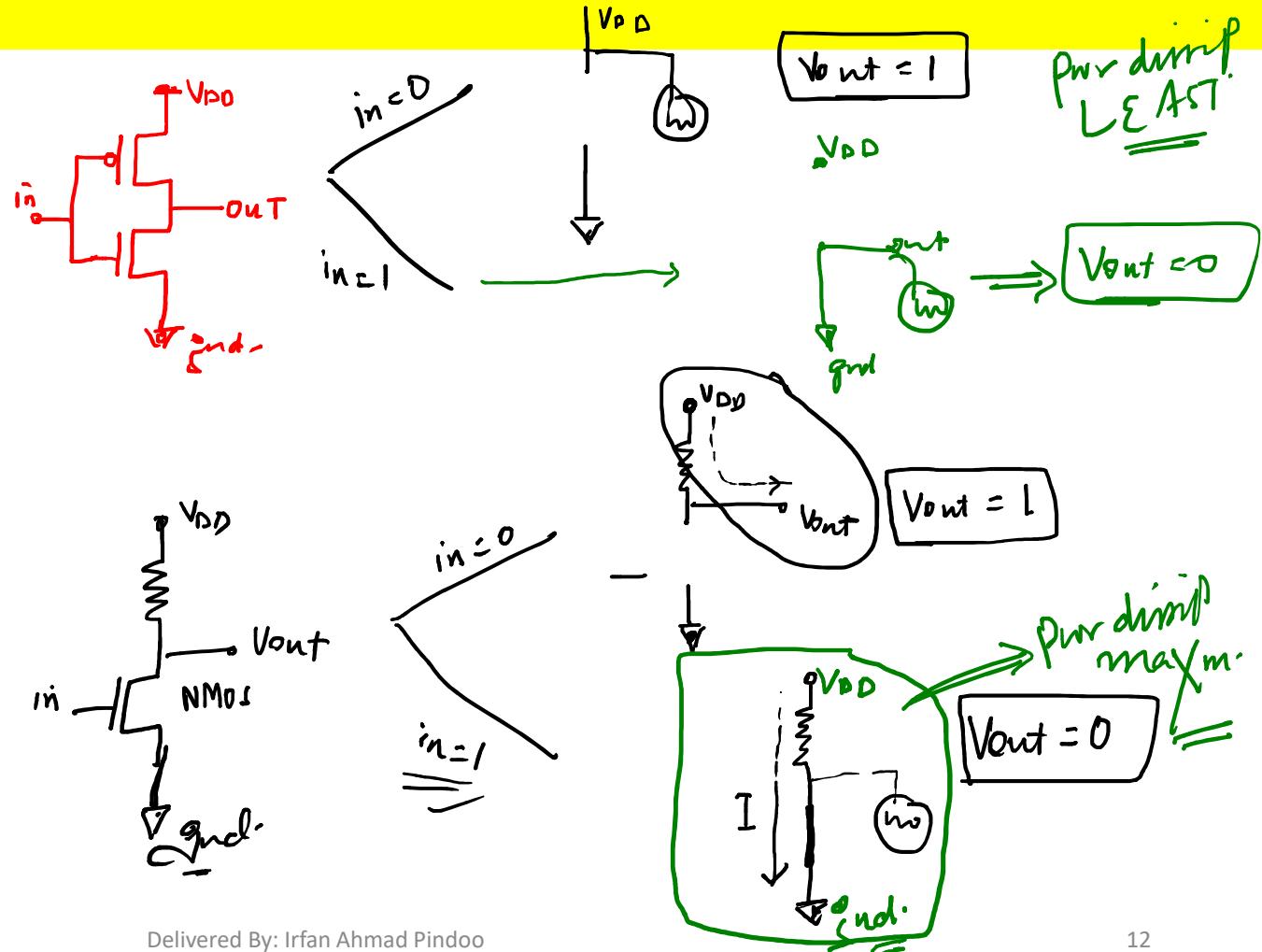
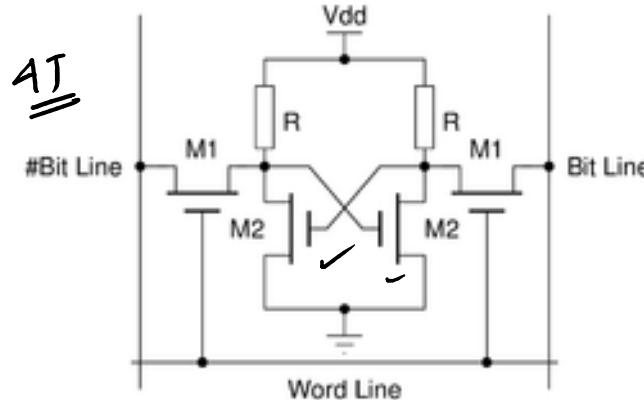
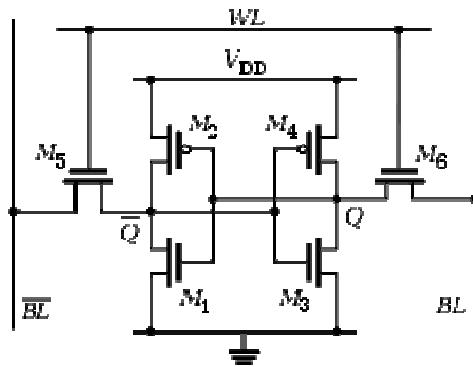
- All CMOS, least Power dissipation, better noise margin, larger size



- High value resistors used, higher Power dissipation, lesser noise margin, smaller size



6T vs 4T Cell SRAM



Delivered By: Irfan Ahmad Pindoo

DIGITAL ELECTRONICS: ECE 213

Topic: DRAM: READ, WRITE and HOLD operation

UNIT VI: MEMORY AND CONVERTERS

Lecture No.: 42

Prepared By: Irfan Ahmad Pindoo

Assistant Professor

VLSI Design, ECE

School of Computer Science and Engineering



Working of DRAM

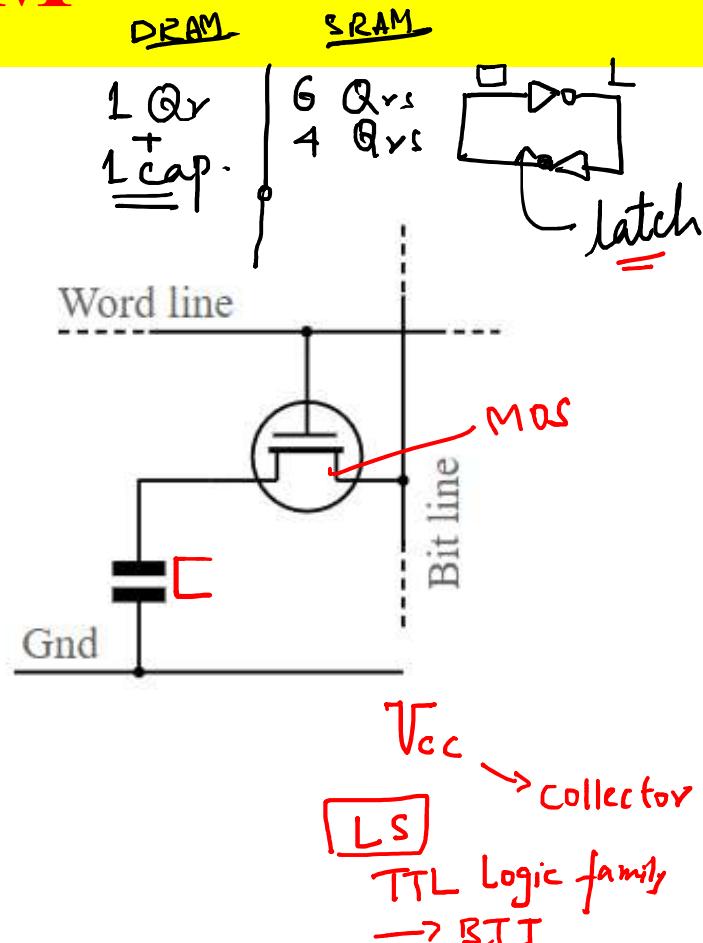
- Basics

- Data is held by the storage capacitor.
- ~~Dynamic~~: Needs periodic refreshing.
- Volatile: loses data when powered OFF
- Single transistor. Therefore smaller, high density and cheaper.
- Charge storage, $Q = CV_{DD}$

$$Q = CV$$

drain
→ MOS Logic family-

data: charge



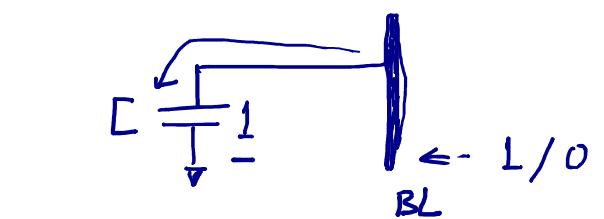
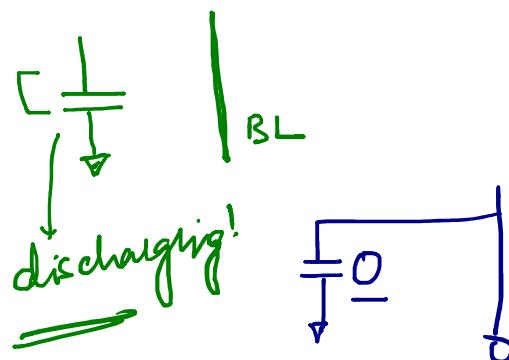
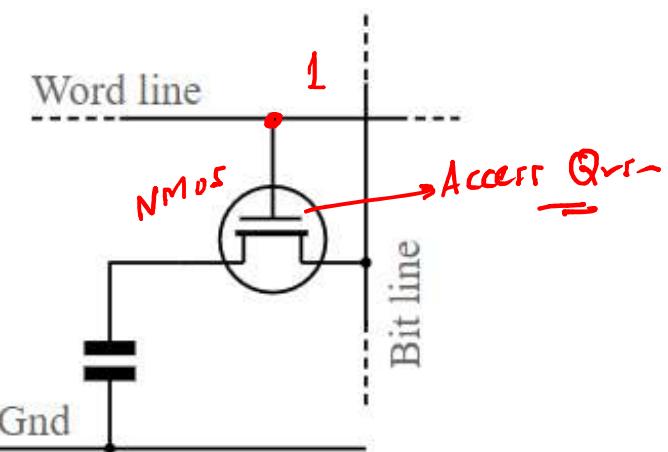
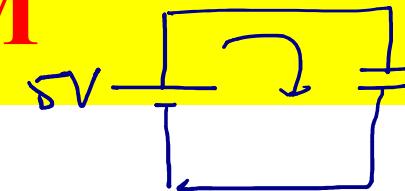
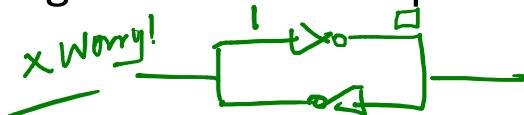
Working of DRAM

- Write Operation

- ✓ 1. WL = 1
- ✓ 2. Access transistors turned ON.
- ✓ 3. Apply voltage (Logic 1 = V_{DD} and Logic 0 = GND) to Bit Line.
- ✓ 4. Accordingly, capacitor will be charged to V_{DD} or discharged to GND.

- Hold Operation

- ✓ 1. WL = 0 → NMOS: OFF
- ✓ 2. Access transistors turned OFF.
- ✓ 3. Charge is held on a capacitor.



Working of DRAM

• Hold Operation

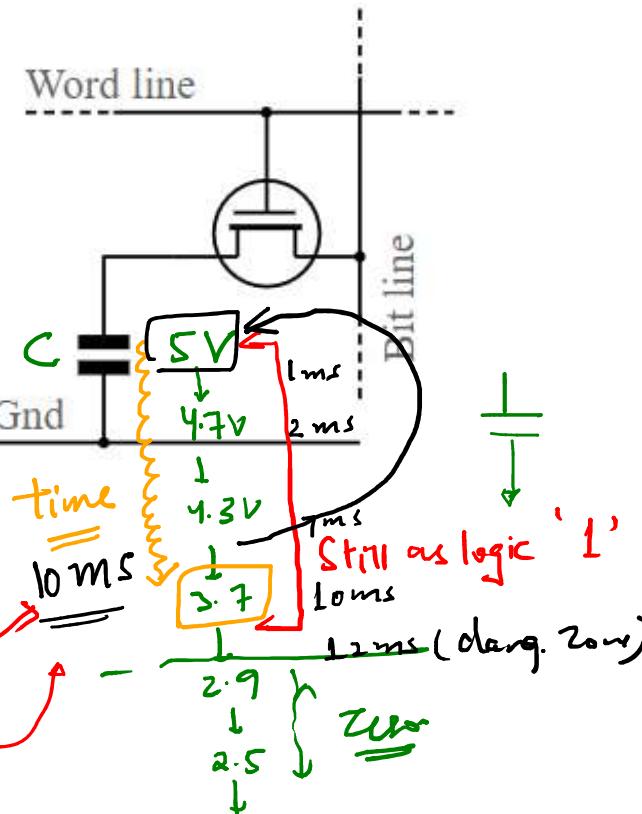
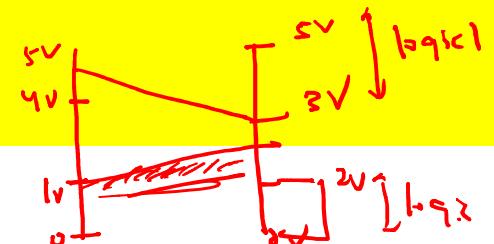
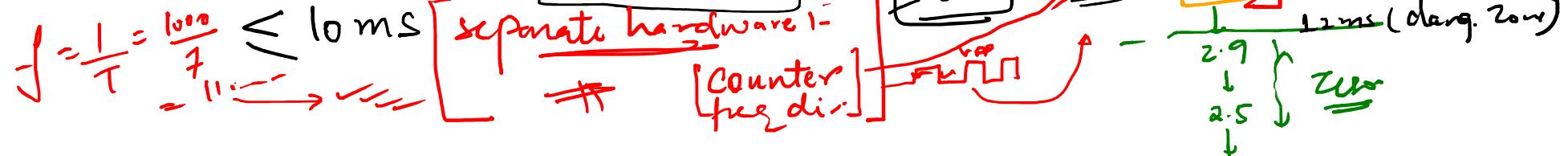
✓ 4. Leakage currents due to discharging of a capacitor.

$$I = Q/T = CV/T$$

5. Determine T to find out the hold time:
(Maximum time upto which the voltage of the capacitor remains high enough to be at logic 1)

$$T = CV/I$$

6. Accordingly, determine the refresh rate.



Working of DRAM

- Read Operation

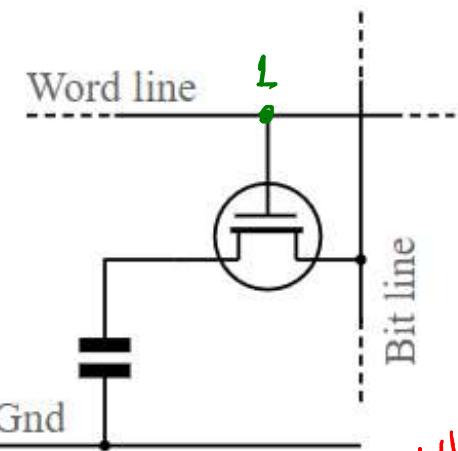
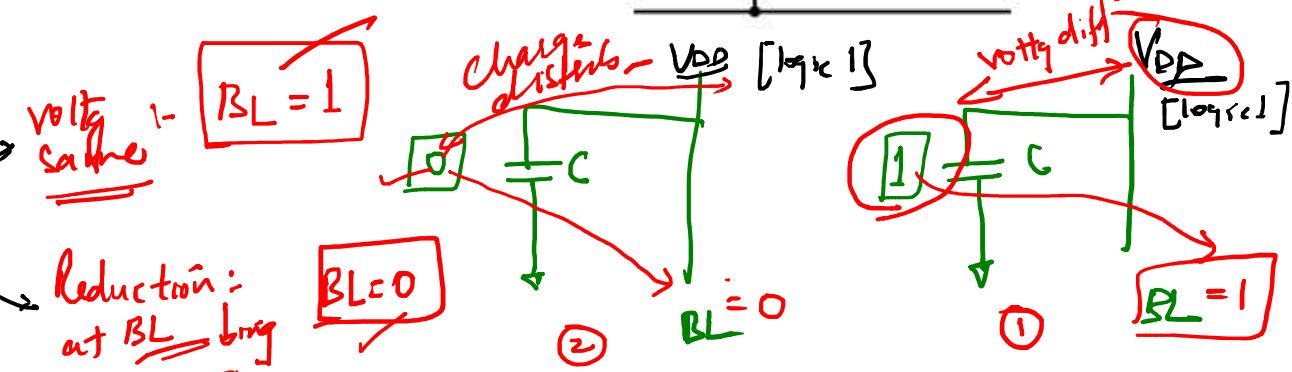
- ✓ 1. $WL = 1$
- ✓ 2. Access transistors turned ON.
- ✓ 3. Charge of the capacitor would get distributed with bit line capacitance.
- ✗ 4. This will change the bit line voltage as 1 or 0

Read-oper (SRAM) :-

(1) Precharging the



Prech. BL



No.

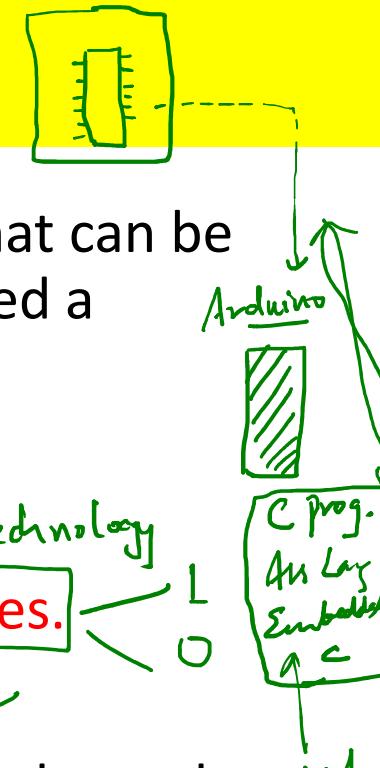
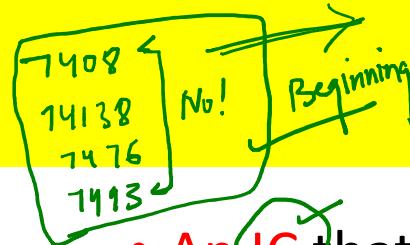
Quick Quiz (POLL)

Precharging is required in _____ operation.

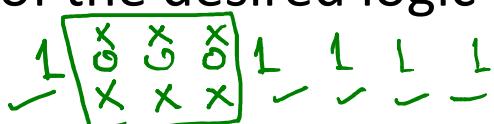
- a) READ
- b) WRITE
- c) HOLD
- d) All of these



Programmable Logic Devices



- An IC that contains large numbers of gates, flip-flops, etc. that can be configured by the user to perform different functions is called a Programmable Logic Device (PLD).
- The internal logic gates and/or connections of PLDs can be changed/configured by a programming process.
- One of the simplest programming technologies is to use fuses.
- In the original state of the device, all the fuses are intact.
- Programming the device involves blowing those fuses along the paths that must be removed in order to obtain the particular configuration of the desired logic function.

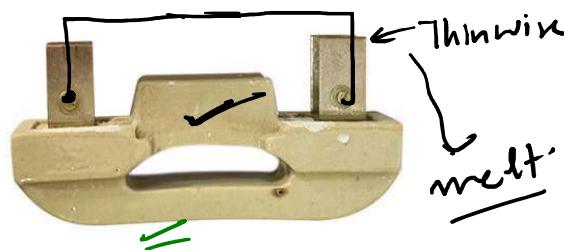
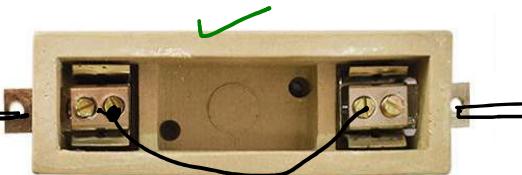


⇒ High voltage ←

High
level
lang



Kit-Fuse
Kat



current passing thru
High voltage \Rightarrow logic 0



-Xsfun
inverters



Wire Connected : logic 1
X Wire " " : logic 0

PLDS-

Types of PLD

MCQ: [PLD, FPGA]

Simple

- ✓ 1. SPLDs (Simple Programmable Logic Devices)

- ROM (Read-Only Memory)
- PLA (Programmable Logic Array)
- PAL (Programmable Array Logic)
- GAL (Generic Array Logic)

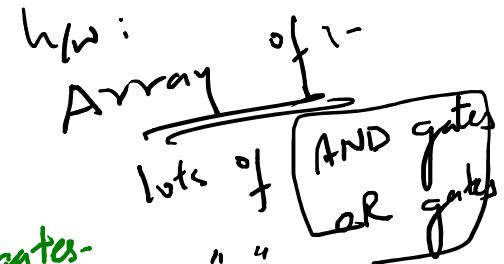
Complex

- ✓ 2. CPLD (Complex Programmable Logic Device)

- ✓ 3. FPGA (Field-Programmable Gate Array)

- C Programming
- Verilog, VHDL
- H/W

MCQ: [PLD, FPGA]

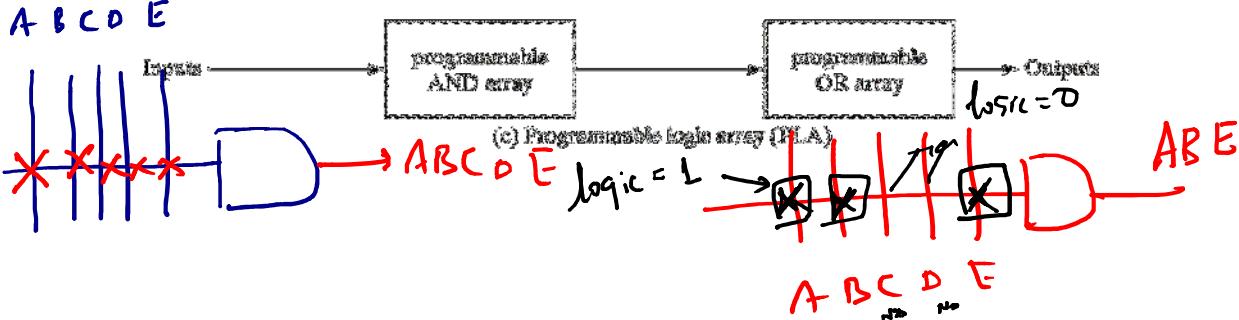
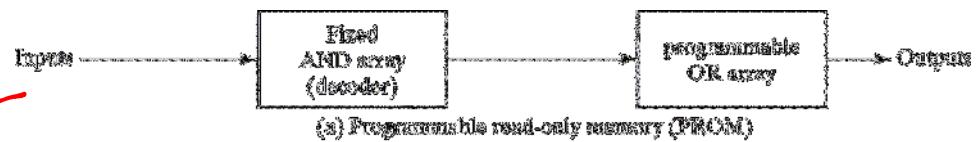


10,000 gates

Types of PLD

↑
Programmable

Device	AND-array	OR-array
PROM	Fixed	Programmable
PLA	Programmable	Programmable
PAL	Programmable	Fixed
GAL	Programmable	Fixed



(a) Conventional Symbol

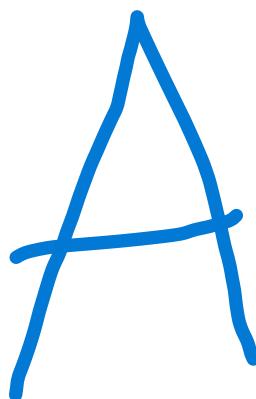
(b) Array Logic Symbol

Source: Morris Mano

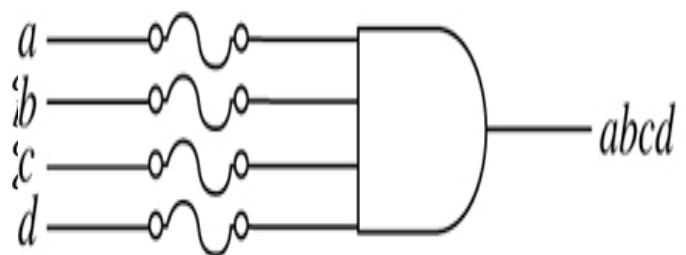
Quick Quiz (POLL)

PAL consists of:

- a) Programmable AND and fixed OR array of gates.
- b) Programmable OR and fixed AND array of gates.
- c) Programmable AND and Programmable OR array of gates.
- d) fixed AND and fixed OR array of gates.

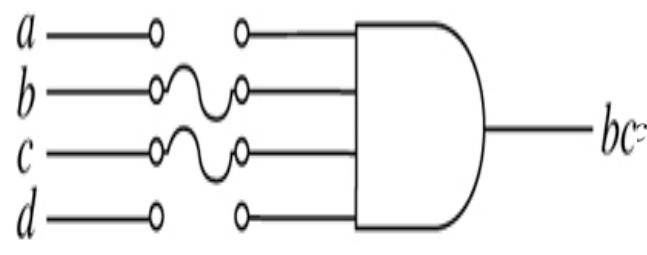


Programming by Blowing the fuses



(a)

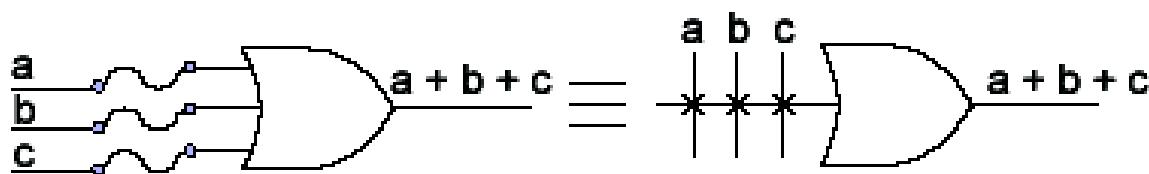
(a) Before programming.



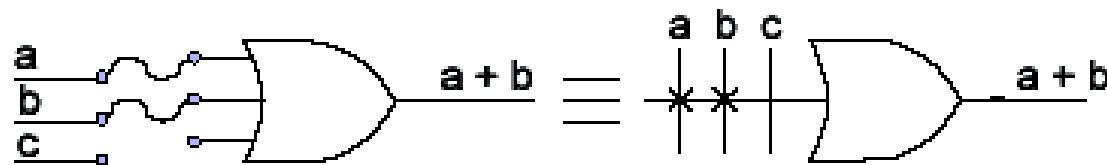
(b)

(b) After programming.

OR PLD Notation

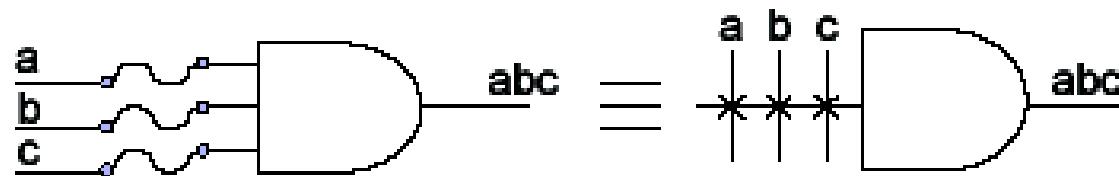


OR gate before programming

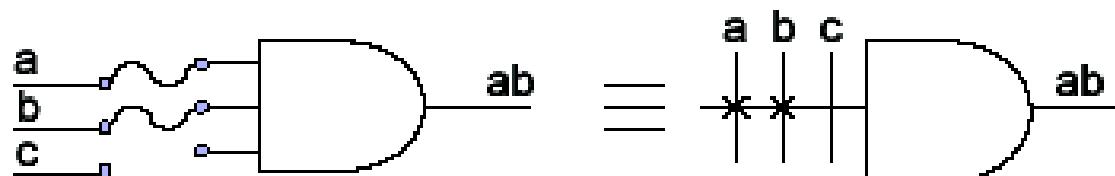


OR gate after programming

AND PLD Notation



AND gate before programming



AND gate after programming

DIGITAL ELECTRONICS: ECE 213

Topic: Programmable Logic Devices
(PLDs)

**UNIT VI: MEMORY AND
CONVERTERS**

Lecture No.: 43 (Tutorial)

Prepared By: Irfan Ahmad Pindoo

Assistant Professor

VLSI Design, ECE

School of Computer Science and Engineering



Read Only Memory



- A block diagram of a ROM is shown below. It consists of k address inputs and n data outputs.
- The number of words in a ROM is determined from the fact that k address input lines are needed to specify 2^k words.

Decoder:

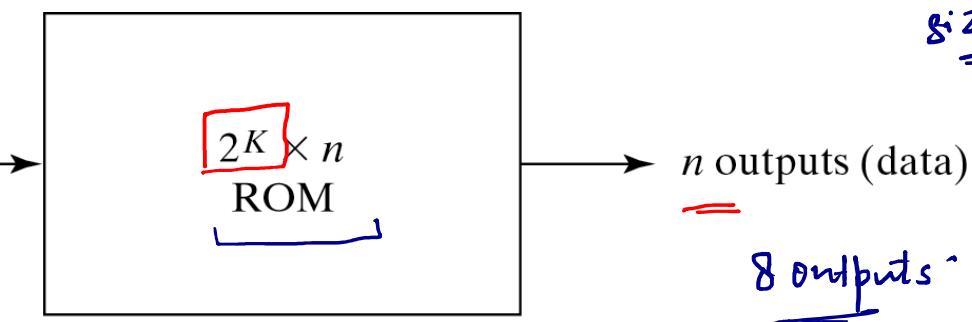
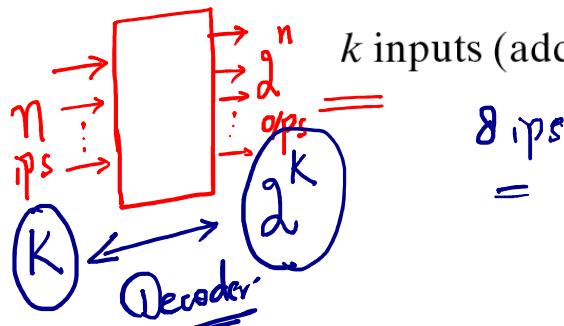


Fig. 7-9 ROM Block Diagram

$$2^k \times n$$

$$2^8 \times 8$$

size: 256×8
2048 bits

Quick Quiz (POLL)

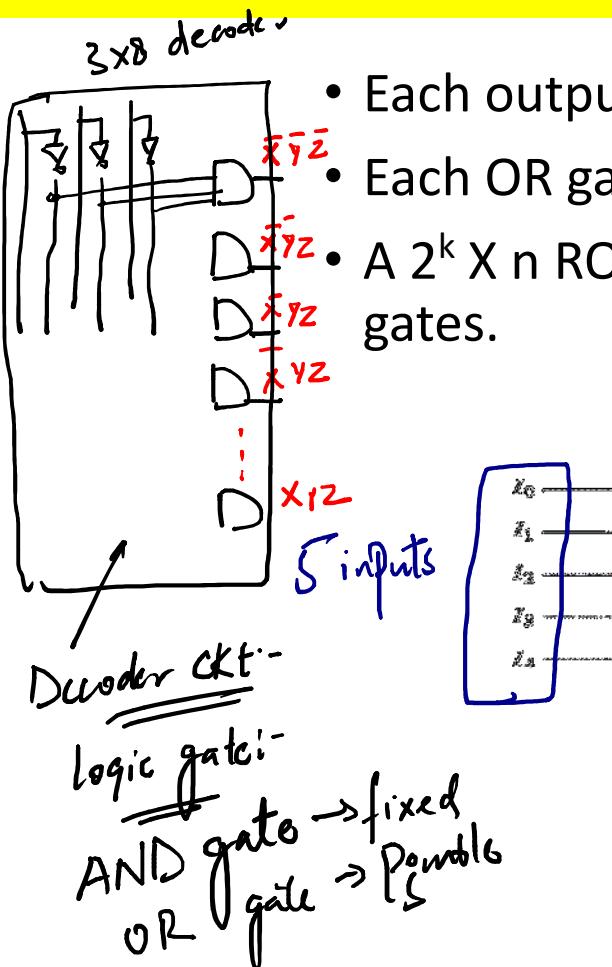
For a $2^k \times n$ ROM, the number of input lines would be :

- a) 2^k
- b) n
- c) k
- d) 2^{k-1}

(k)
Ans



Construction of ROM



- Each output of the decoder represents a memory address.
- Each OR gate must be considered as having 32 inputs.
- A $2^k \times n$ ROM will have an internal $k \times 2^k$ decoder and n OR gates.

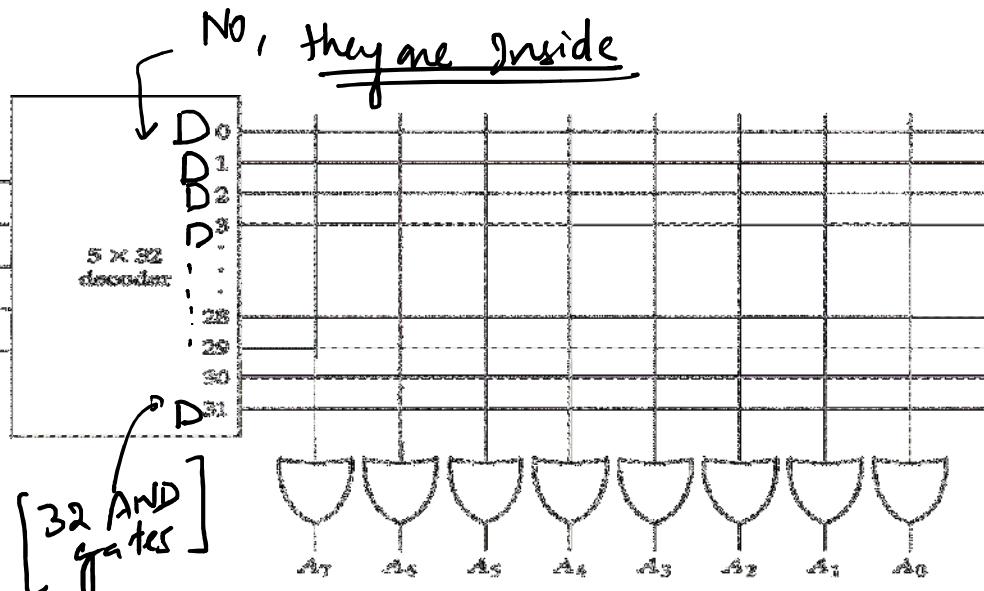


Fig. 7-10 Internal Logic of a 32 \times 8 ROM

Truth Table of ROM

- A programmable connection between two lines is logically equivalent to a switch that can be altered to either be close or open.
- Intersection between two lines is sometimes called a cross-point.

Table 7-3
ROM Truth Table (Partial)

Inputs					Outputs							
I4	I3	I2	I1	I0	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	0	1	0	1	1	0	1	1	0
0	0	0	0	1	0	0	0	1	1	1	0	1
0	0	0	1	0	1	1	0	0	0	1	0	1
0	0	0	1	1	1	0	1	1	0	0	1	0
⋮					⋮							
1	1	1	0	0	0	0	0	0	1	0	0	1
1	1	1	0	1	1	1	0	0	0	0	1	0
1	1	1	1	0	0	1	0	0	1	0	1	0
1	1	1	1	1	0	0	1	1	0	0	1	1

Programming the ROM

✓ K-map
 ✓ Simplif.
 ✓ Canonical format

In Table 7-3, 0 → no connection
1 → connection

Address 3 = 10110010 is permanent storage using fuse link

$32 \times 8 = 256$ Connections

1/0

Pgmg is done as per O/P T/T.

Table 7-3
ROM Truth Table (Partial)

Inputs					Outputs							
I ₄	I ₃	I ₂	I ₁	I ₀	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
0	0	0	0	0	1	0	1	1	0	1	1	0
0	0	0	0	1	0	0	0	1	1	1	0	1
0	0	0	1	0	1	1	0	0	1	0	1	1
0	0	0	1	1	1	0	0	0	1	0	1	0
⋮												
1	1	1	1	0	0	0	0	1	0	0	1	0
1	1	1	1	0	1	1	0	0	0	0	1	0
1	1	1	1	1	0	1	0	0	1	0	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1

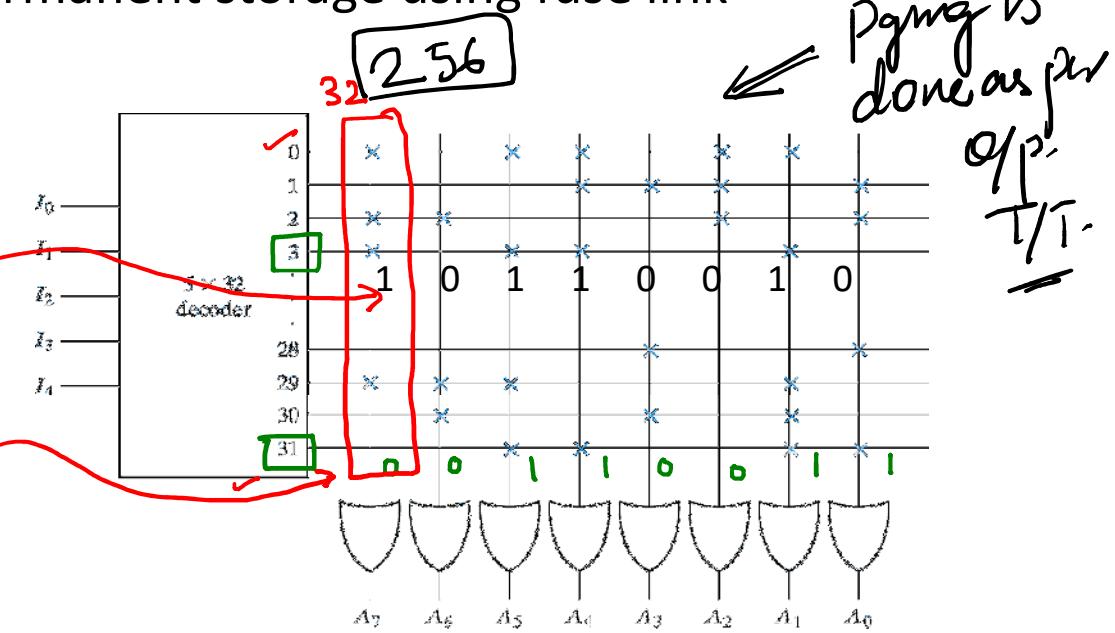


Fig. 7-11 Programming the ROM According to Table 7-3

X : means connection

Using PROM for Logic Design

Univ. ckt

Example 2:

Truth Table - 2 ops

x_2	x_1	x_0	f_1	f_2
0	0	0	1	0
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	0
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

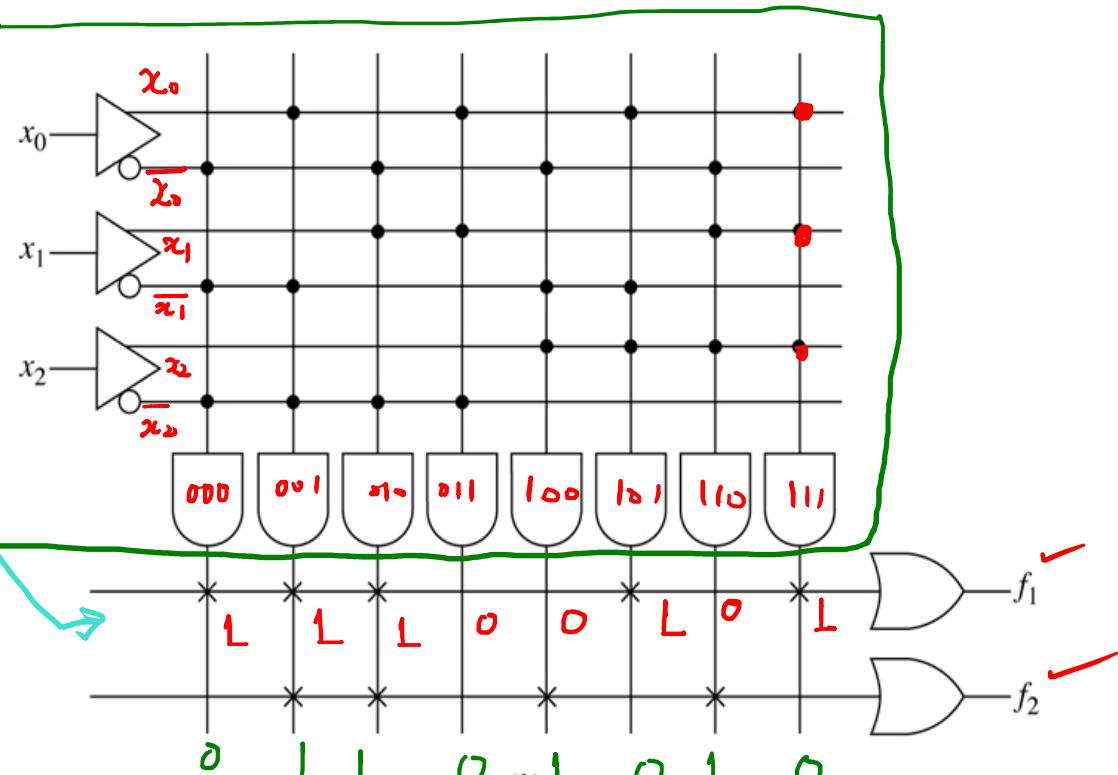
→ Connection
+ → No Gr.

fixed
AND Array

(a)

| No. of OR gates =
| No. of Outputs

(a) Truth table.

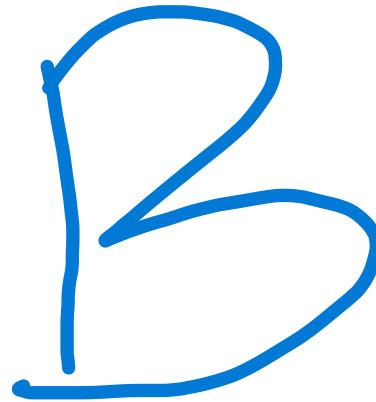


(b) PROM realization.

Quick Quiz (POLL)

Which of the following combinational circuit is the part of ROM design?

- a) Encoder
- b) Decoder B
- c) Multiplexer
- d) Demultiplexer



Types of ROM

1. PROM (Programmable Read Only Memory)

- It can be programmed by user. Once programmed, the data and instructions in it cannot be changed.

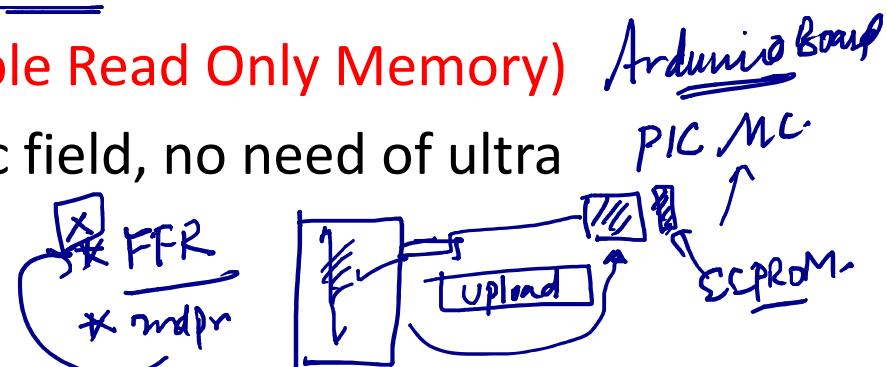
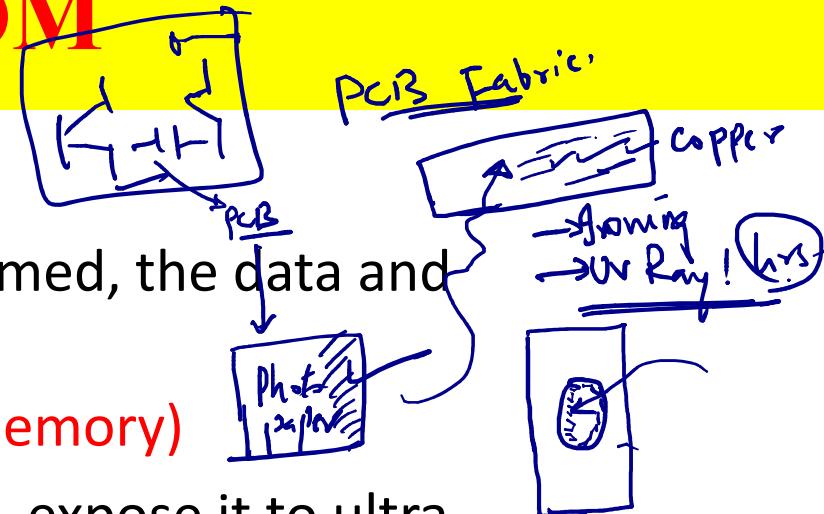
2. EPROM (Erasable Programmable Read Only Memory)

- It can be reprogrammed. To erase data from it, expose it to ultra violet light. To reprogram it, erase all the previous data.

3. EEPROM (Electrically Erasable Programmable Read Only Memory)

- The data can be erased by applying electric field, no need of ultra violet light. reprogrammed

160
300

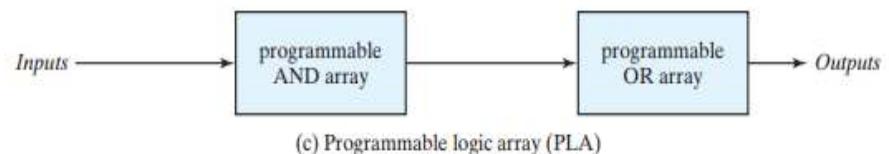
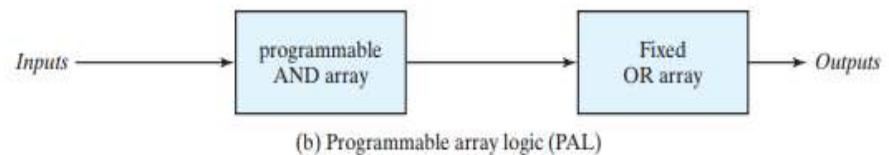
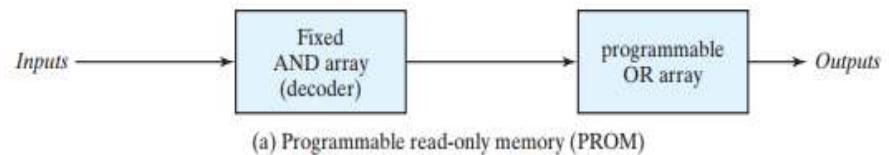


Programmable Logic Array (PLA)

x minimization
x addl. h/w

- It has programmable AND array and programmable OR array.
- Because **both arrays are programmable**, it is flexible.
- The PLA is similar in concept to the PROM, except that the PLA does not provide full decoding of the variables and does not generate all the minterms.

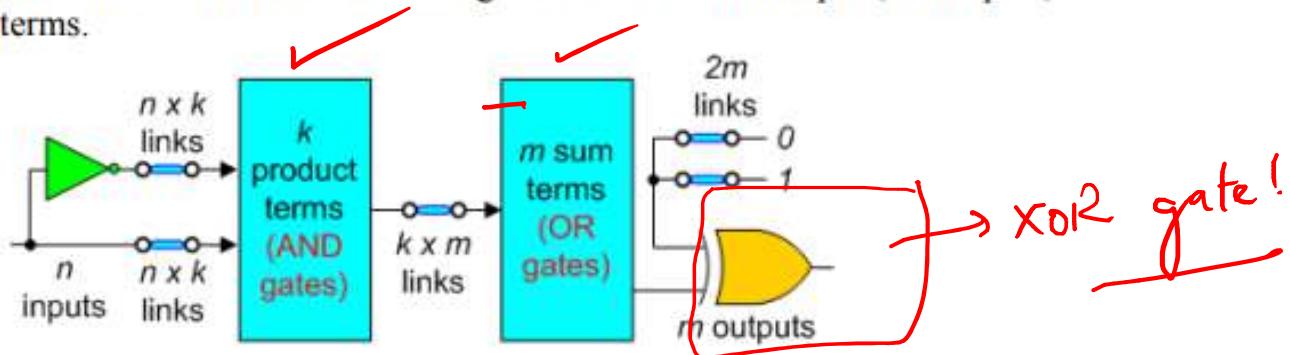
✓ Kmaps
✓ reduced expressions
— find output!



Programmable Logic Array (PLA)

- In PLAs, instead of using a decoder as in PROMs, a number (k) of AND gates is used where $k < 2^n$, (n is the number of inputs).

A block diagram of the PLA is shown in the figure. It consists of n inputs, m outputs, and k product terms.



The product terms constitute a group of k AND gates each of $2n$ inputs.

Links are inserted between all n inputs and their complement values to each of the AND gates.

Programmable Logic Array (PLA)

least
logic
gates

minm amt of logic gates

complement \bar{f} . Both are function f . generated.

- The output of the OR gate goes to an XOR gate, where the other input can be programmed to receive a signal equal to either logic 1 or logic 0.
- The output is inverted when the XOR input is connected to 1 (since $x \text{ XOR } 1 = \bar{x}$).
- The output does not change when the XOR input is connected to 0 (since $x \text{ XOR } 0 = x$).

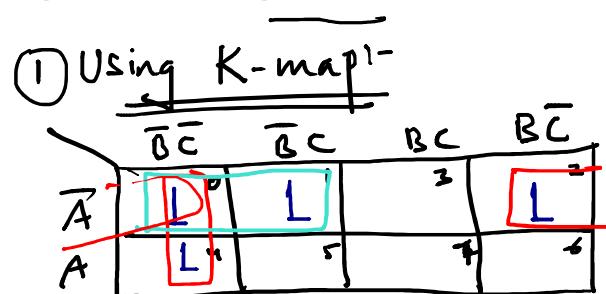
Programmable Logic Array (PLA)

?

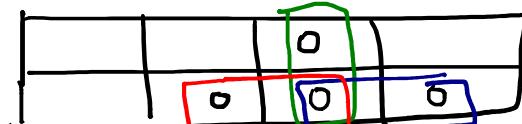
$$\overline{B}C + BC + \overline{B}\overline{C}$$

Example 1: Implement the following table using PLA.

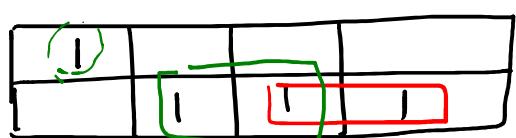
A	B	C	f_1	f_2
0	0	0	1	1
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	0	1



$$f_1 \xrightarrow{\text{from L's}} \overline{B}\overline{C} + \overline{A}\overline{B} + \overline{A}\overline{C}$$

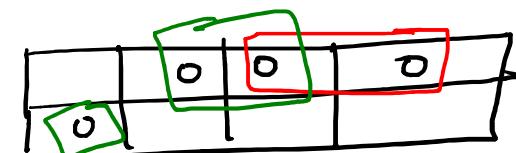


$$f_1 = \overline{B}C + AB + AC$$



$$f_2 = AB + AC + \overline{A}\overline{B}\overline{C}$$

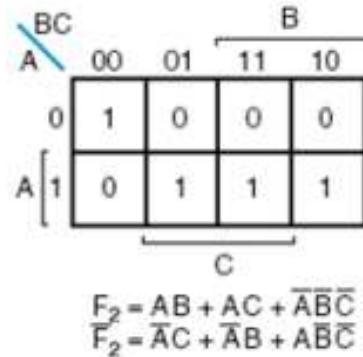
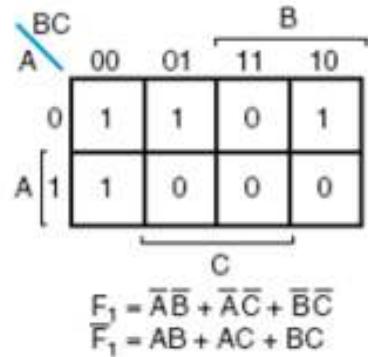
$f_1, f_2 \Rightarrow$ x good choice (6)
 $f_1, \overline{f}_2 \Rightarrow$ // (6)
 $\overline{f}_2, f_1 \Rightarrow$ // (6)
 $f_1, f_2 \Rightarrow$ (4)



$$\overline{f}_2 = \overline{AB} + \overline{AC} + A\overline{B}\overline{C}$$

Programmable Logic Array (PLA)

Step 1: K-maps for simplification:



- Designing using a PLA, a careful investigation must be taken in order to reduce the distinct product terms.
- Both the true and complement forms of each function should be simplified to see which one can be expressed with fewer product terms and which one provides product terms that are common to other functions.

Programmable Logic Array (PLA)

- K-maps for simplification:

BC		B			
A		00	01	11	10
0	1	1	0	1	
1	1	0	0	0	

$$\begin{aligned}F_1 &= \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C} \\F_1 &= AB + AC + BC\end{aligned}$$

BC		B			
A		00	01	11	10
0	1	0	0	0	
1	0	1	1	1	

$$\begin{aligned}F_2 &= AB + AC + \bar{A}\bar{B}\bar{C} \\F_2 &= \bar{A}C + \bar{A}B + A\bar{B}C\end{aligned}$$

The combination that gives a minimum number of product terms is:

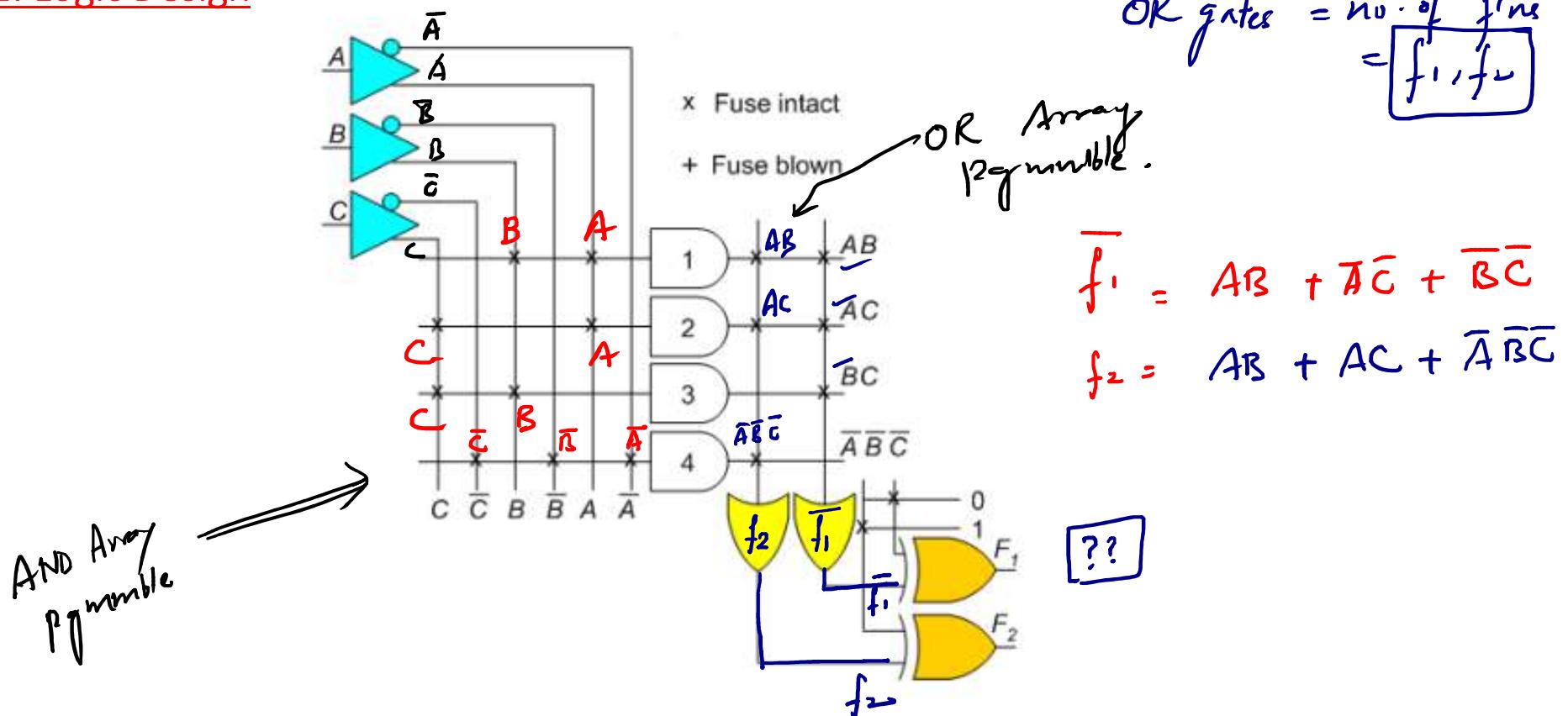
$$F_1 = AB + AC + BC \text{ or } F_1 = (AB + AC + BC)'$$

$$F_2 = \bar{A}C + \bar{A}B + A\bar{B}C'$$

This gives only 4 distinct product terms: AB , AC , BC , and $A'B'C'$.

Programmable Logic Array (PLA)

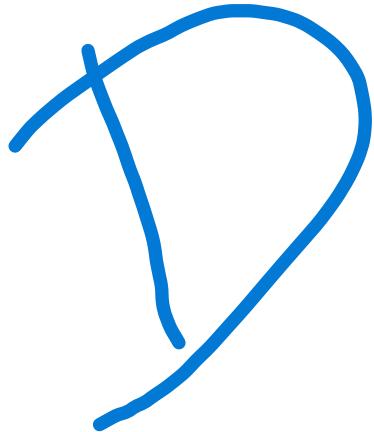
Step 2: Logic Design



Quick Quiz (POLL)

Which of the following statement is wrong?

- a) $x \text{ XOR } 0 = x$
- b) $x \text{ XOR } x = 0$
- c) $x \text{ XOR } 1 = \bar{x}$
- d) $x \text{ XOR } x = \bar{x}$



DIGITAL ELECTRONICS: ECE 213

Topic: Programmable Logic Devices
(PLDs) an FPGA

**UNIT VI: MEMORY AND
CONVERTERS**

Lecture No.: 44

Prepared By: Irfan Ahmad Pindoo

Assistant Professor

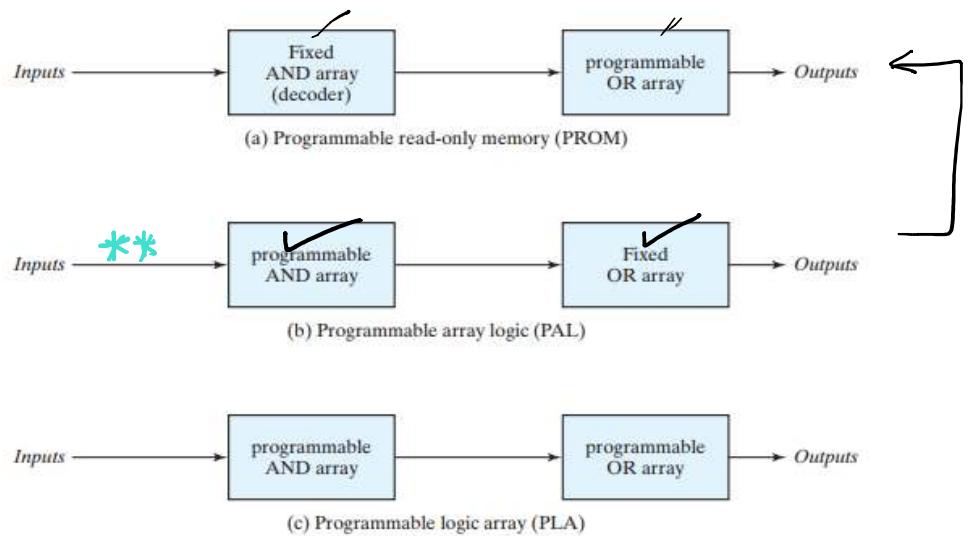
VLSI Design, ECE

School of Computer Science and Engineering



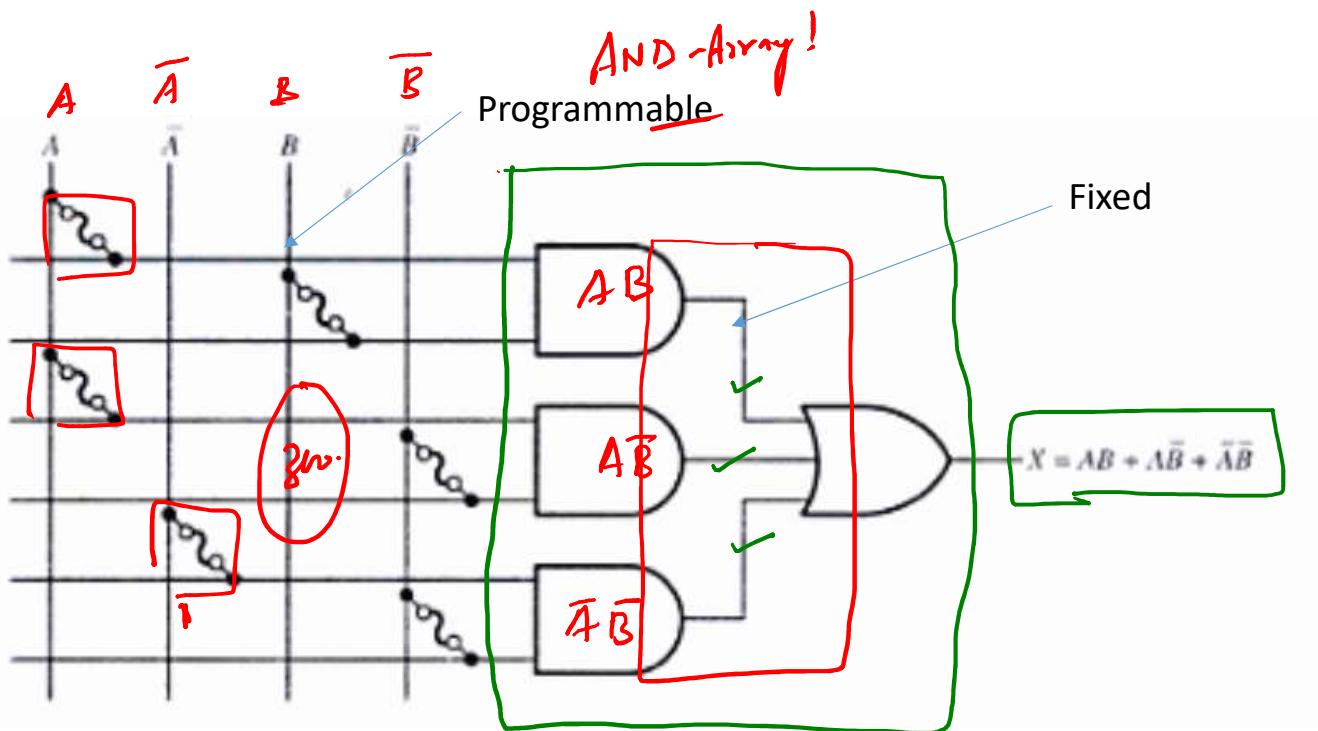
Programmable Array Logic (PAL)

- It has programmable AND array and **fixed OR array**.
- Because **only the AND array is programmable**, it is **easier to use**
- However, it is not flexible as compared to Programmable Logic Array (PLA).

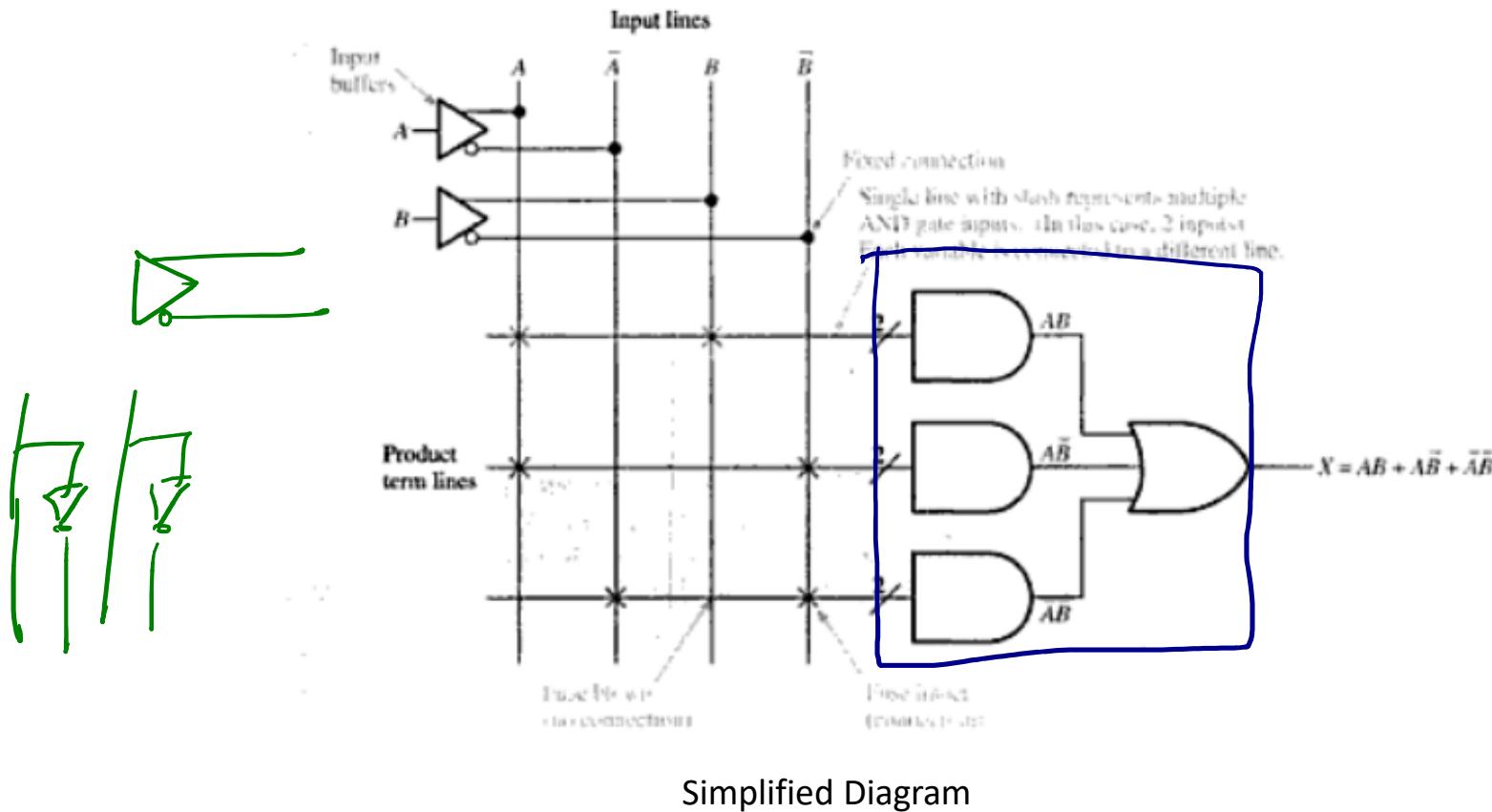


Programmable Array Logic (PAL)

Example 1: Implement $AB + A\bar{B} + \bar{A}\bar{B}$ using PAL?



Programmable Array Logic (PAL)



Programmable Array Logic (PAL)

Example 2: Implement following function using PAL?

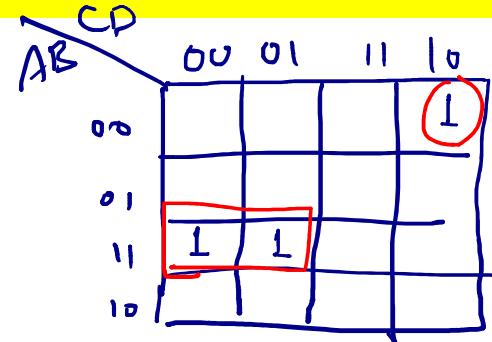
$$w(A, B, C, D) = \Sigma(2, 12, 13) \quad \checkmark$$

$$x(A, B, C, D) = \Sigma(7, 8, 9, 10, 11, 12, 13, 14, 15) \quad *$$

$$y(A, B, C, D) = \Sigma(0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 15) \quad *$$

$$z(A, B, C, D) = \Sigma(1, 2, 8, 12, 13) \quad \checkmark$$

* Kmap
* Eqns
* designing



$$\begin{aligned} w &= \bar{A}\bar{B}C\bar{D} + A\bar{B}\bar{C} \\ z &= \bar{A}\bar{B}\bar{C}D + \boxed{\bar{A}\bar{B}C\bar{D} + A\bar{B}\bar{C}} + A\bar{C}\bar{D} \\ &= \bar{A}\bar{B}\bar{C}D + w + A\bar{C}\bar{D} \end{aligned}$$

Programmable Array Logic (PAL)

$$w(A, B, C, D) = \Sigma(2, 12, 13)$$

$$x(A, B, C, D) = \Sigma(7, 8, 9, 10, 11, 12, 13, 14, 15)$$

$$y(A, B, C, D) = \Sigma(0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 15)$$

$$z(A, B, C, D) = \Sigma(1, 2, 8, 12, 13)$$

PAL

- \nearrow Pgm: AND Array
- \searrow fix: OR Array

Find out the gp with highest no.
of AND gates.

Simplifying the four functions to a minimum number of terms results in the following
Boolean functions:

$$w = ABC' + A'B'CD'$$

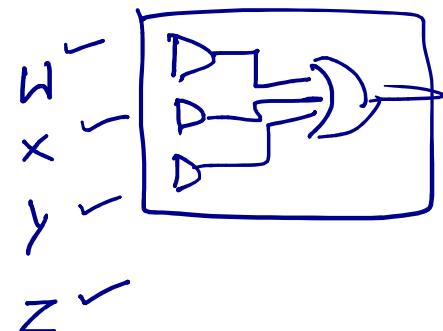
$$x = A + BCD$$

$$y = A'B + CD + B'D'$$

$$\begin{aligned} z &= ABC' + A'B'CD' + AC'D' + A'B'C'D \\ &= w + AC'D' + A'B'C'D \end{aligned}$$

w	2
x	2 / 1
y	3
z	3

highest: $\boxed{3}$



LOGICAL DIAGRAM for PAL:

*AND Array
possible*

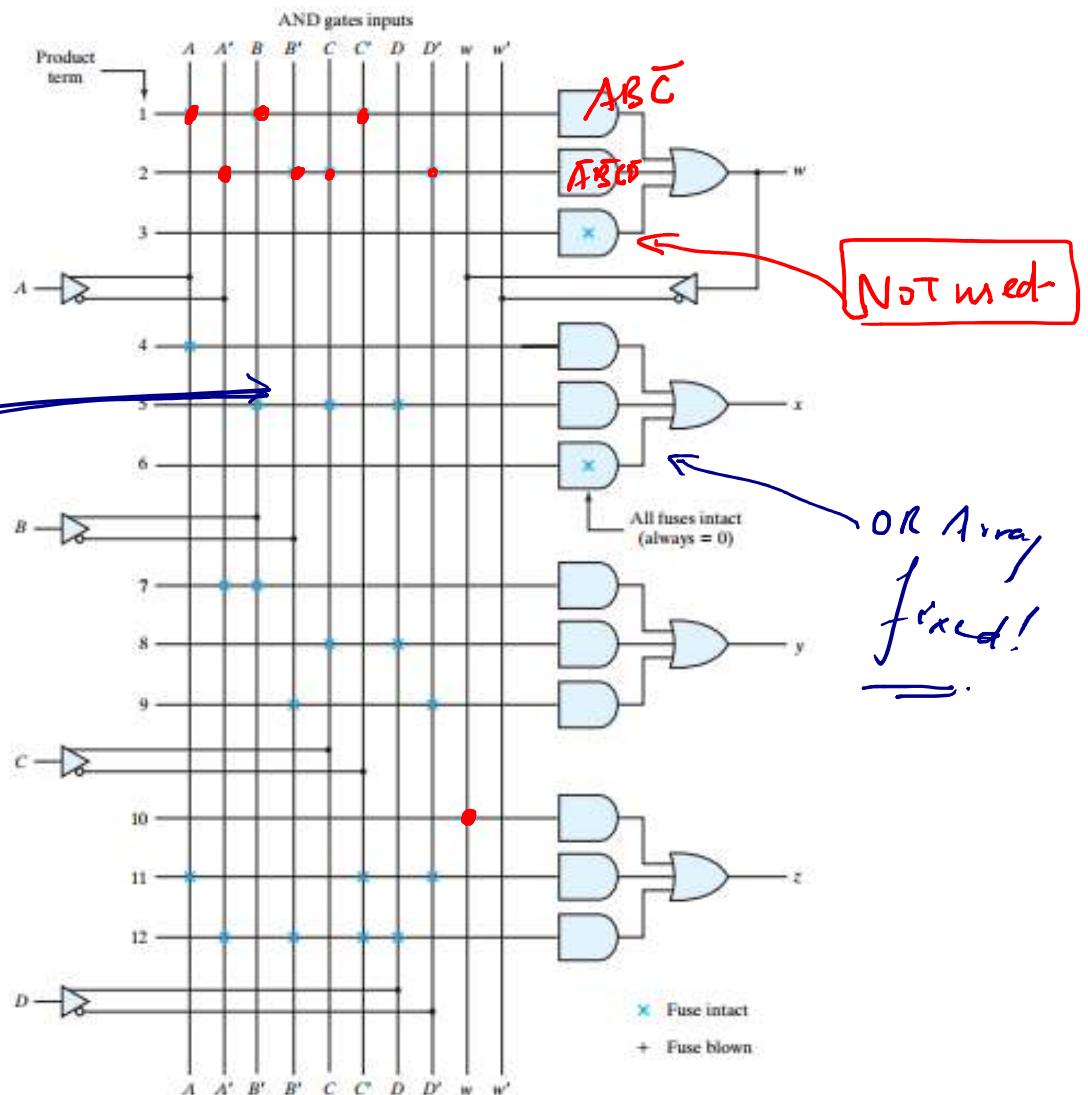
Simplifying the four functions to a minimum number of terms results in the following Boolean functions:

$$w = \underline{ABC'} + \underline{A'B'CD'}$$

$$x = A + BCD$$

$$y = A'B + CD + B'D'$$

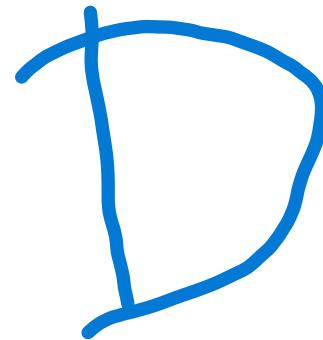
$$\begin{aligned} z &= ABC' + A'B'CD' + AC'D' + A'B'C'D \\ &= w + AC'D' + A'B'C'D \end{aligned}$$



Quick Quiz (POLL)

Which of the following statement is right for PAL?

- a) Programmable OR and fixed AND array of logic gates
- b) Programmable OR and Programmable AND array of logic gates
- c) Fixed OR and fixed AND array of logic gates
- d) Programmable AND and fixed OR array of logic gates



Programmable Array Logic (PAL)

Example 3: Implement the given function using PAL Logic.

$$f_1(A, B, C) = \sum(2, 3, 6, 7), f_2(A, B, C) = \sum(1, 3, 7), f_3(A, B, C) = \sum(3, 5, 6, 7),$$

Step 1: Truth Table

A	B	C	F1	F2	F3
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	0	0	1
1	1	0	1	0	1
1	1	1	1	1	1

Programmable Array Logic (PAL)

Step 2: K-Maps for simplification

	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}			1	1
A			1	1

$$f_1 = B$$

	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}		1	1	
A			1	

$$f_2 = \bar{A}C + BC$$

	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}				1
A			1	1

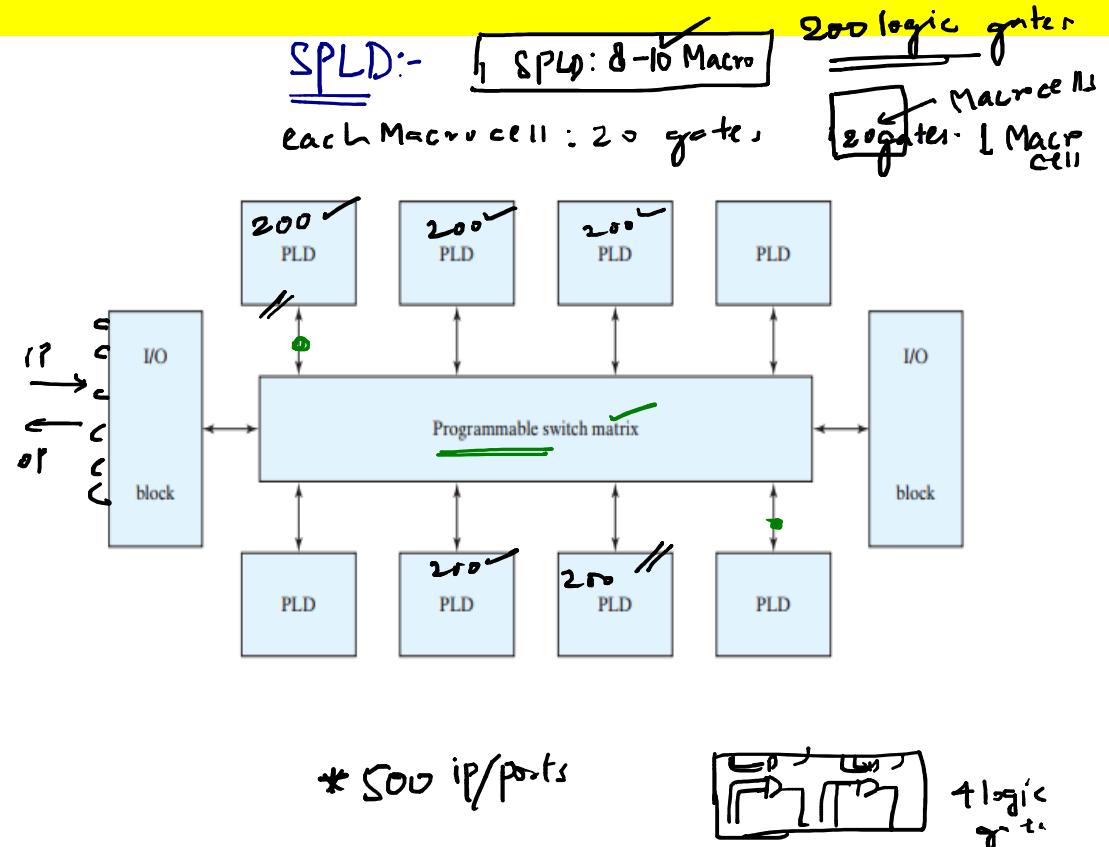
$$f_3 = AB + BC + AC$$

Step 3: Logic Diagram



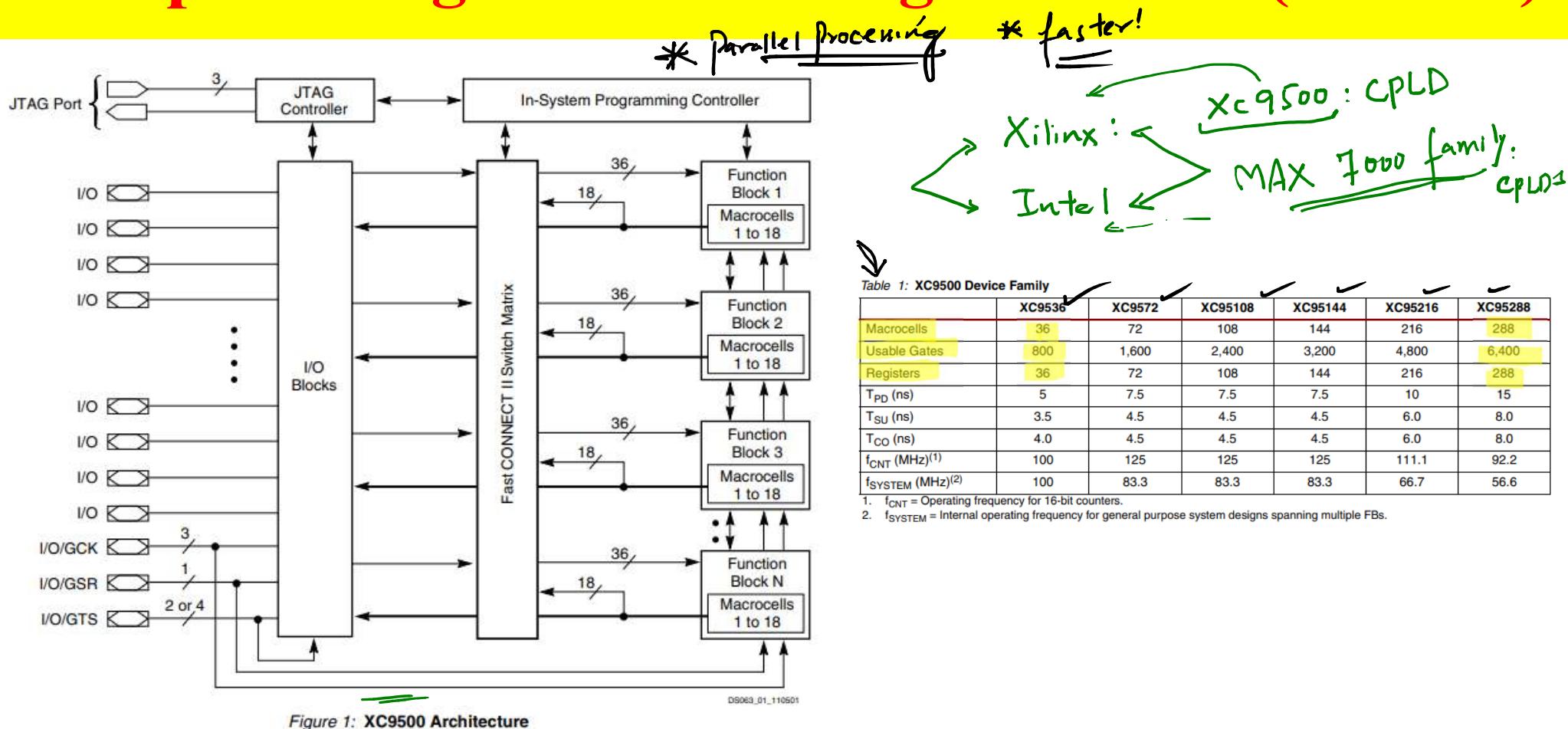
Complex Programmable Logic Devices (CPLDs)

- A CPLD contains a **bunch of PLD blocks** whose inputs and outputs are connected together by a global interconnection matrix.
- Thus a CPLD has **two levels of programmability**: each PLD block can be programmed, and then **the interconnections** between the PLDs can be programmed.



Source of Figure: Morris Mano

Complex Programmable Logic Devices (CPLDs)



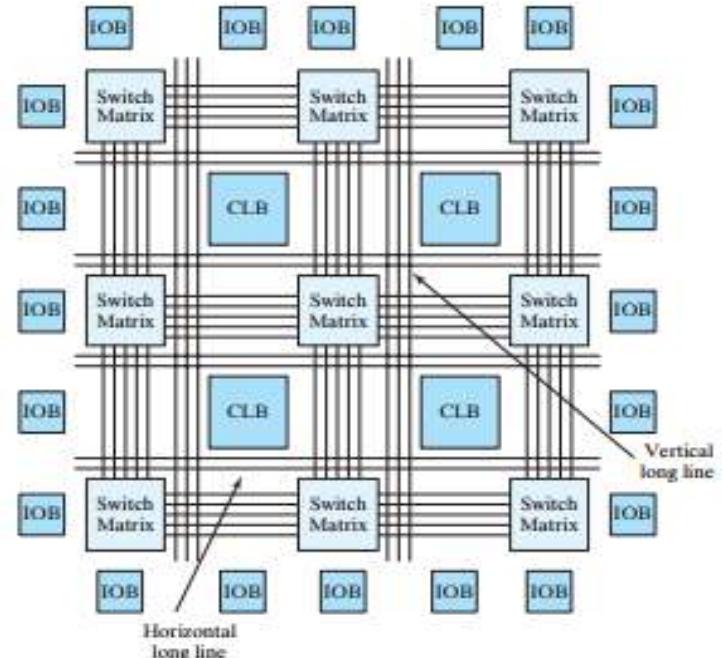
Source of Figure: Morris Mano

Field Programmable Gate Arrays (FPGAs)

'User' = Programmable at any field → applic

- A field-programmable gate array (FPGA) is a VLSI circuit that can be programmed at **the user's location**.
- A typical FPGA consists of an array of **millions** of logic blocks, surrounded by programmable input and output blocks and connected together via programmable interconnections.

FPGA: "sea of gates"



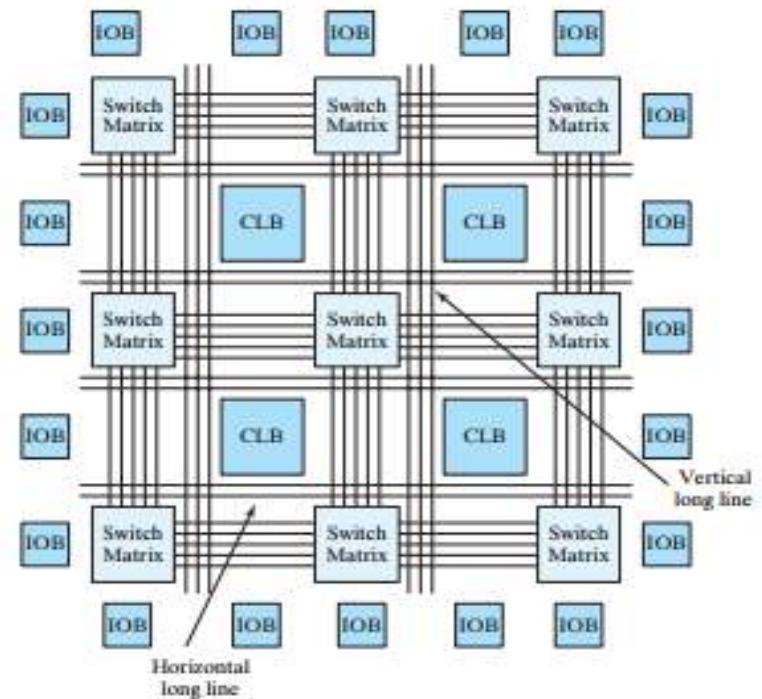
Field Programmable Gate Arrays (FPGAs)

FPGA consists of:

1. Configurable Logic Blocks. **CLB**
2. Switch Matrix
3. Input Output Blocks **IOB**

FPGA is programmable at all 3 levels.

* faster processing
Parallel
— very high pin count —



Source of Figure: Morris Mano

Quick Quiz (POLL)

Which of the following is user configurable in the “field”?

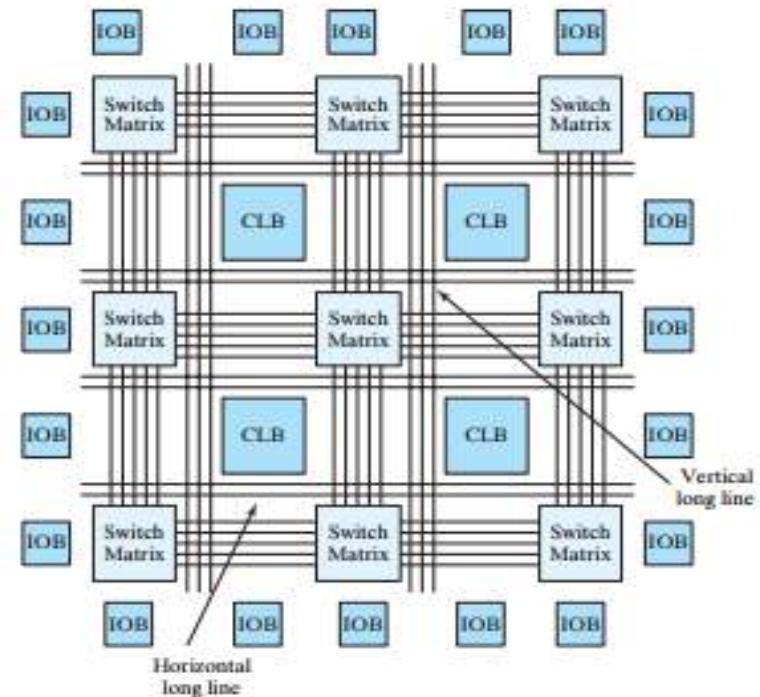
- a) SPLD
- b) CPLD
- c) FPGA
- d) All of these



Field Programmable Gate Arrays (FPGAs)

1. Configurable Logic Blocks (CLBs).

- The programmable logic structure FPGA consists of a 2-dimensional array of **configurable logic blocks (CLBs)**.
- Each CLB can be configured (programmed) to implement *any* Boolean function of its input variables.
- Typically CLBs have between 4-6 input variables.
- Functions of larger number of variables are implemented using more than one CLB.
- In addition, each CLB typically contains 1 or 2 FFs to allow implementation of sequential logic.
- Large designs are partitioned and mapped to a number of CLBs with each CLB configured (programmed) to perform a particular function. These CLBs are then connected together to fully implement the target design.
- Connecting the CLBs is done using the FPGA programmable routing structure.

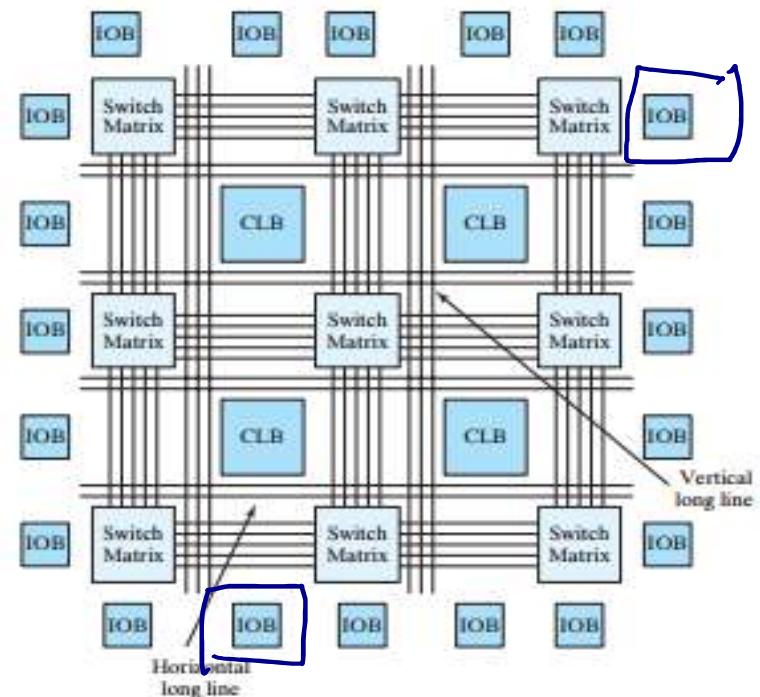


Source of Figure: Morris Mano

Field Programmable Gate Arrays (FPGAs)

2. Switch Matrix

- To allow for flexible interconnection of CLBs, FPGAs have *programmable* routing resources.



Quick Quiz (POLL)

In FPGA, vertical and horizontal directions are separated by

-
- a) A line
 - b) A channel
 - c) A strobe
 - d) A flip-flop

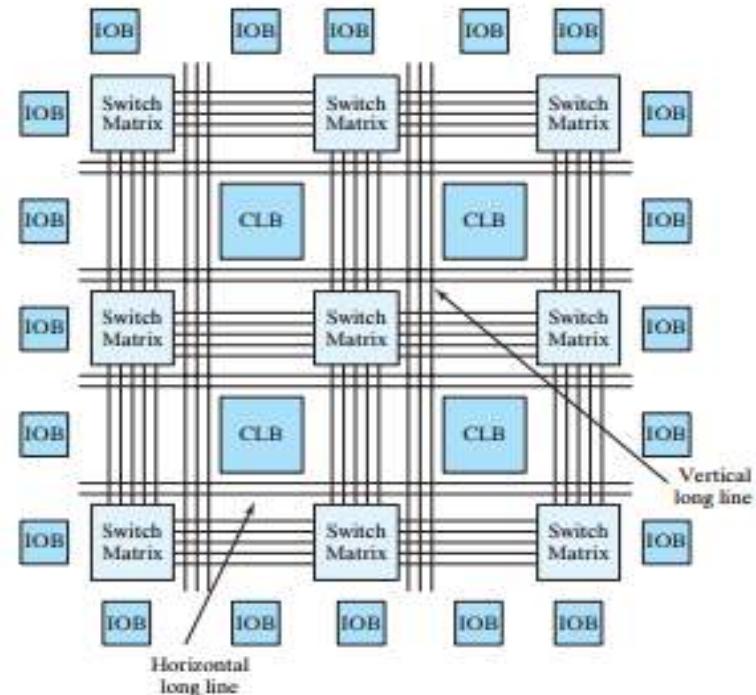
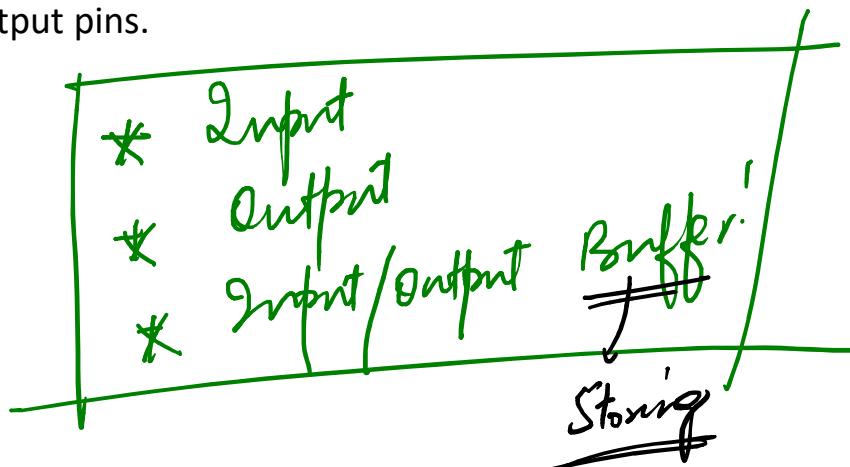


Field Programmable Gate Arrays (FPGAs)

6 months

3. Programmable I/O

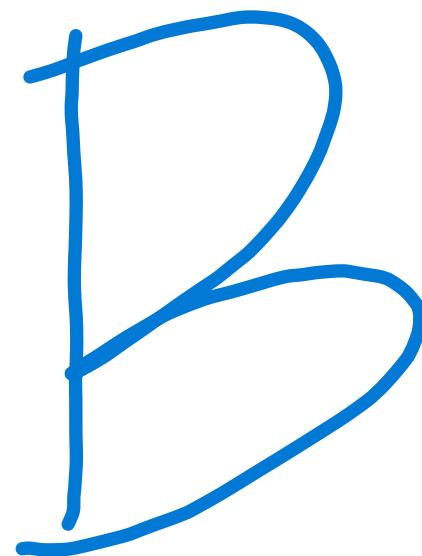
1. These are mainly buffers that can be configured either as input buffers, output buffers or input/output buffers.
2. They allow the pins of the FPGA chip to function either as input pins, output pins or input/output pins.



Quick Quiz (POLL)

The FPGA refers to _____

- a) First programmable Gate Array
- b) Field Programmable Gate Array
- c) First Program Gate Array
- d) Field Program Gate Array



CPLD vs FPGA

CPLD	FPGA
Number of gates <u>Thousands</u>	<u>Millions</u> of logic gates
Coarse grained 	<u>Fine Grained</u> 
Simple ✓	Complex ✓
<u>EEPROM</u> based	RAM Based
Cheap ✓	Expensive ! <i>faster operation</i>

Applications of FPGA

1. DSP
2. Defense System
3. ASIC Prototyping
4. Medical Imaging
5. Healthcare
6. Computer Vision
7. Speech Recognition
8. Cryptography
9. Bioinformatics, etc.

<https://www.xilinx.com/products/silicon-devices/fpga/what-is-an-fpga.html>

DIGITAL ELECTRONICS: ECE 213

**Topic: CONVERTERS: Analog to
Digital and Digital to Analog**

**UNIT VI: MEMORY AND
CONVERTERS**

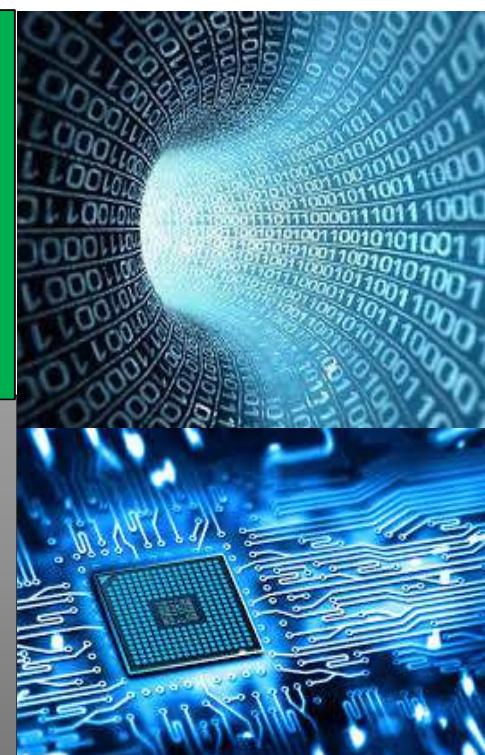
Lecture No.: 45

Prepared By: Irfan Ahmad Pindoo

Assistant Professor

VLSI Design, ECE

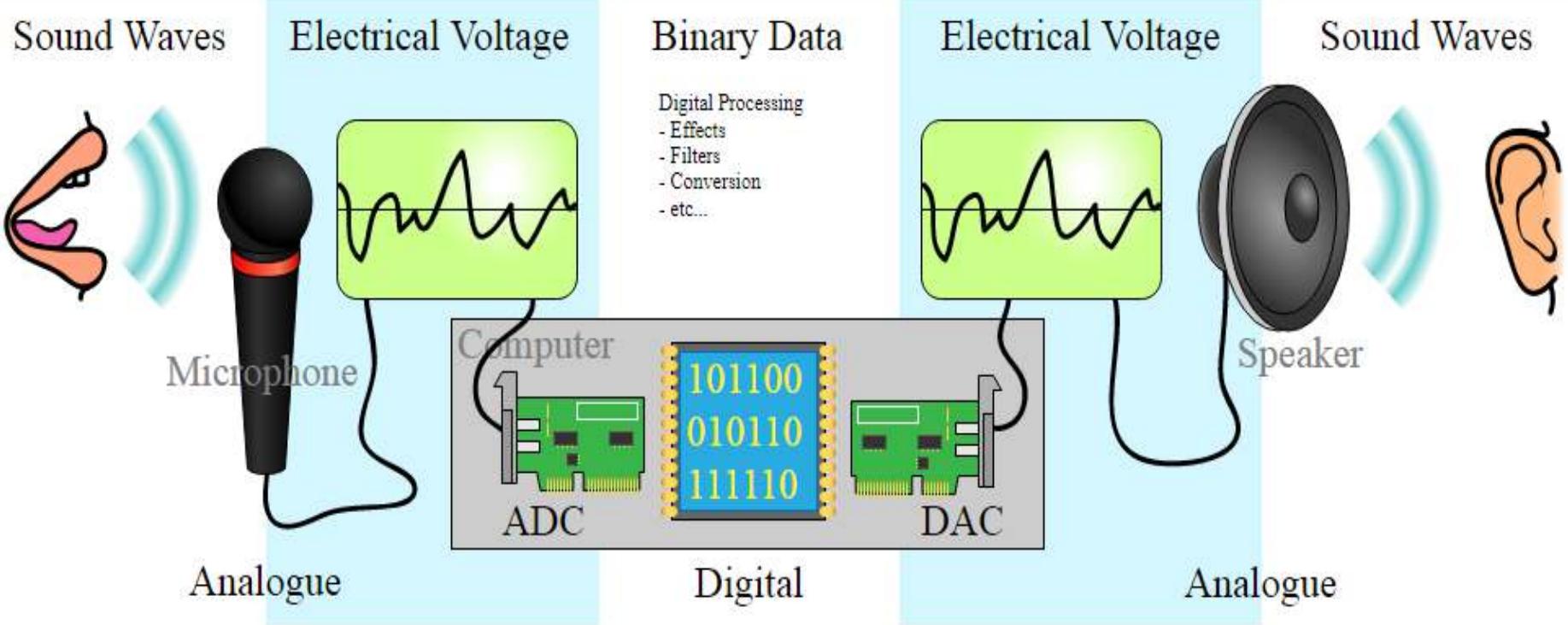
School of Computer Science and Engineering



Introduction

- Most physical variables are **analog** in nature and can take on any value within **a continuous range** of values.
- Examples include **temperature, pressure, light intensity, audio signals, position, rotational speed, and flow rate**.
- Digital systems perform all of their internal operations using digital circuitry and digital operations.

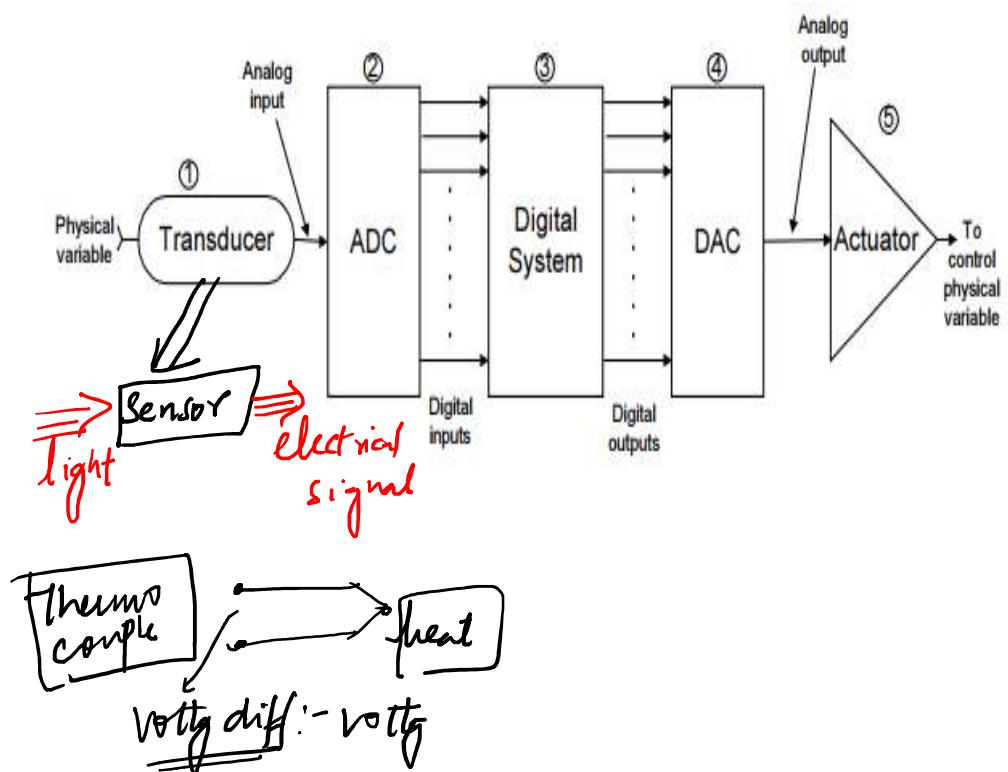
Introduction



ADC – Analog to Digital Convertor
DAC – Digital to analog Converter

Transducer

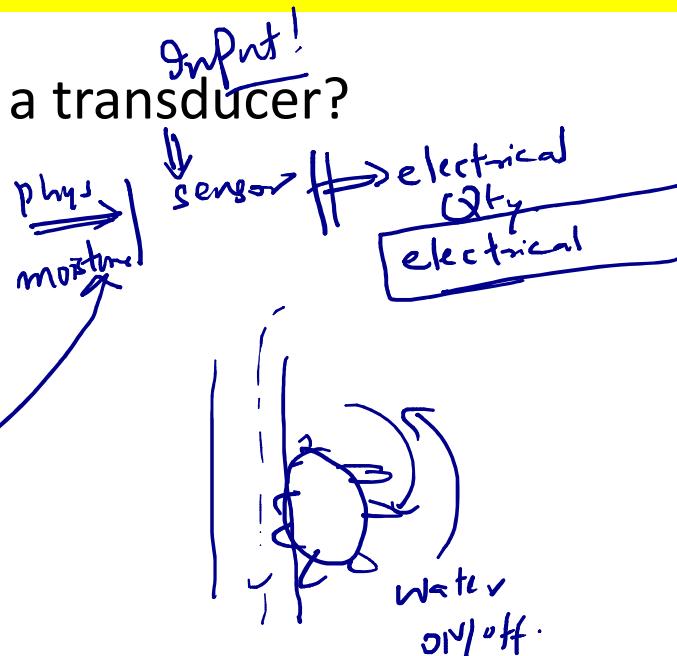
- The physical variable is normally a **nonelectrical** quantity.
- A transducer is a device that **converts** the **physical variable** to an **electrical variable**.
- Some common transducers include **thermistors**, **photocells**, **photodiodes**, **flow meters**, **pressure transducers**, and **tachometers**.
speed!



Quick Quiz (POLL)

Which of the following is NOT a transducer?

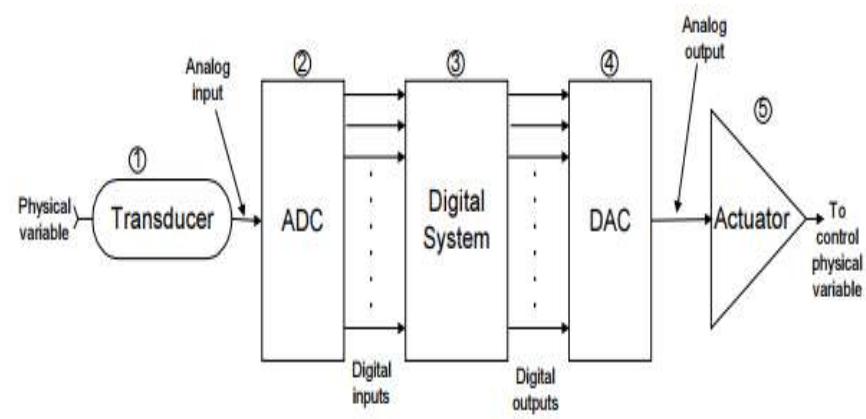
- a) thermistor *
- b) valve output
- c) photodiode *
- d) Moisture sensor*



B

ADC

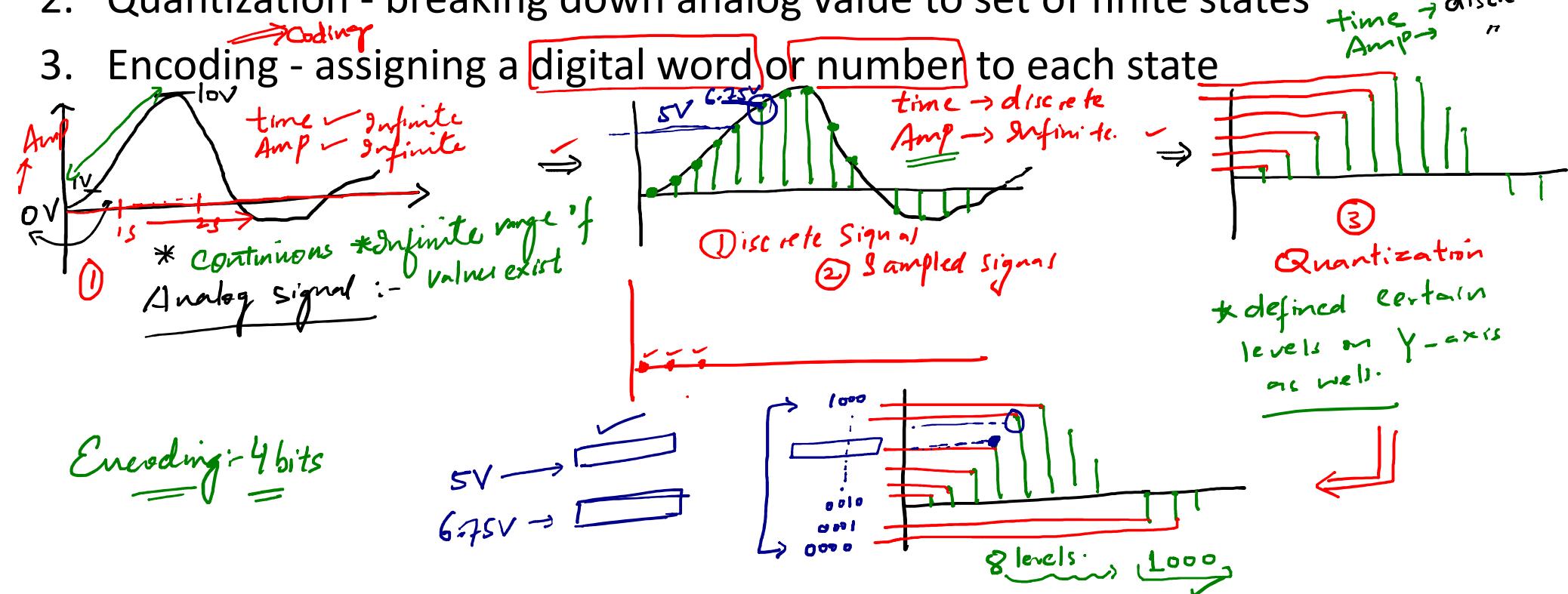
- The transducer's electrical analog output serves as the **analog input to the ADC**.
- The ADC converts this analog input to a digital output. This digital output consists of a **number of bits** that represent the value of the analog input.
- The digital representation of the analog values is transmitted from the ADC to the **digital computer**, which **stores** the digital value and **processes** it according to a program of instructions that it is executing.



Analog to Digital Conversion: Three step process

No. of Books :- 1, 2, 3, ..., L. 1...L - - -

1. Sampling - Taking samples
2. Quantization - breaking down analog value to set of finite states
3. Encoding - assigning a **digital word** or number to each state

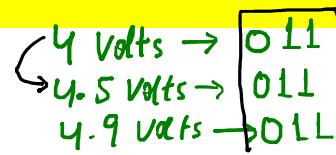


Example

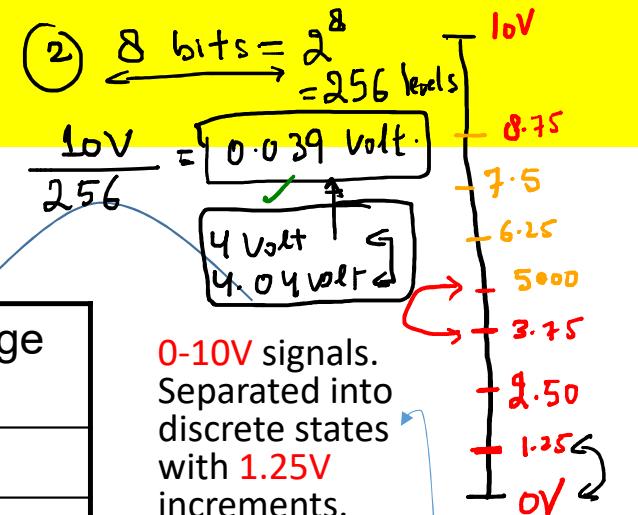
Encoding: 3 bits = 8 samples
 (3) 12 bits → $\frac{10V}{2^{12}} = 0.0024$
only 3

000 0000

111



Inaccurate!



Output States	Output Binary Equivalent
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Sampling!
 64 Kbps
 128 Kbps
 320 Kbps

Bits

Accuracy!
 sound → better

Output States	Discrete Voltage Ranges (V)
0	0.00-1.25
1	1.25-2.50
2	2.50-3.75
3	3.75-5.00
4	5.00-6.25
5	6.25-7.50
6	7.50-8.75
7	8.75-10.0

Analog quantization size:

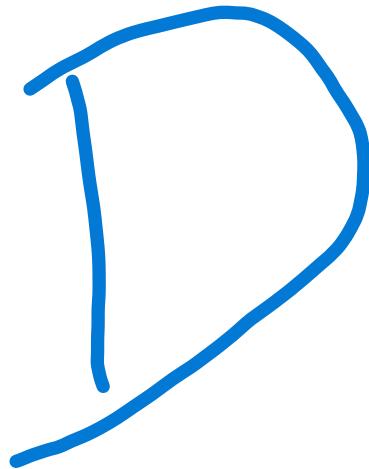
$$Q = (V_{max} - V_{min})/N = (10V - 0V)/8 = 1.25V$$

$$\frac{10 \text{ Volts}}{8 \text{ levels}} = \frac{\text{Full Scale Voltage}}{\text{No. of levels}} = 1.25 \text{ Volt.}$$

Quick Quiz (POLL)

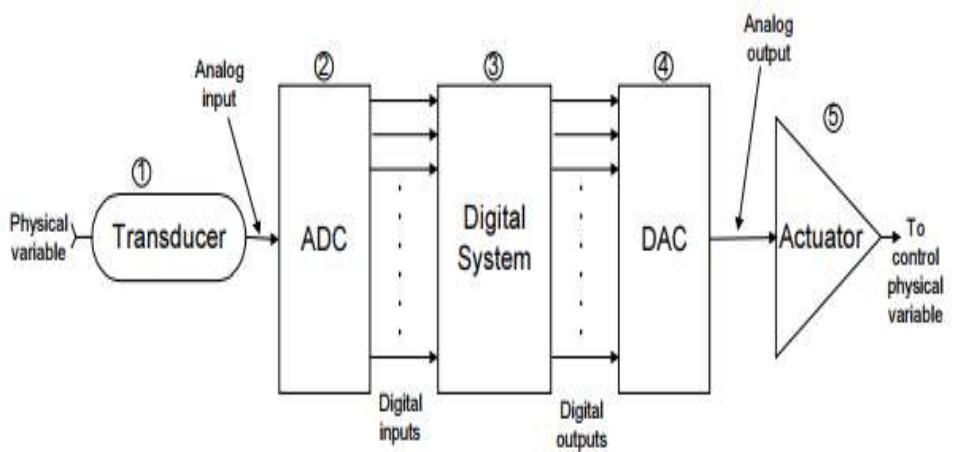
Identify the right sequence for analog to digital conversion?

- a) Encoding, Sampling and Quantization
- b) Quantization, Sampling and Encoding
- c) Sampling, Encoding and Quantization
- d) Sampling, Quantization and Encoding



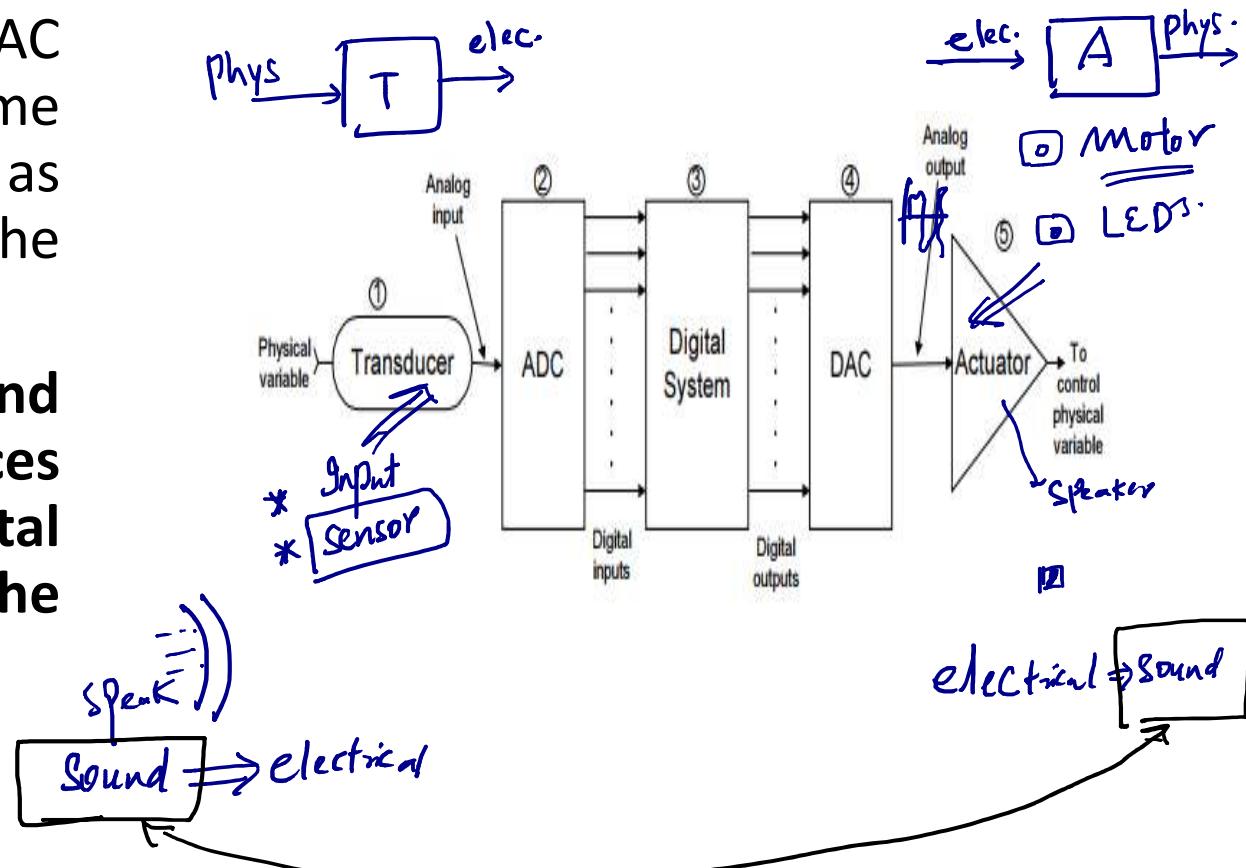
DAC

- This **digital output from the computer** is connected to a DAC, which converts it to a **proportional analog** voltage or current.
- Basically, D/A conversion is the process of taking a value represented in digital code (such as straight binary or BCD) and converting it to a voltage or current which is proportional to the digital value.



Actuator

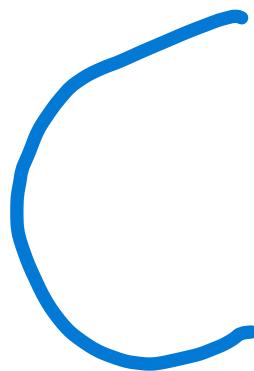
- The analog signal from the DAC is often connected to some device or circuit that serves as an actuator to control the physical variable.
- Thus we see that ADCs and DACs function as interfaces between a completely digital system, like a computer, and the analog world.



Quick Quiz (POLL)

Which of the following component is used at the output terminal?

- a) ADC
- b) Transducer
- c) Actuator
- d) DAC



DIGITAL ELECTRONICS: ECE 213

**Topic: CONVERTERS: Analog to
Digital and Digital to Analog**

**UNIT VI: MEMORY AND
CONVERTERS**

Lecture No.: 46

Prepared By: Irfan Ahmad Pindoo

Assistant Professor

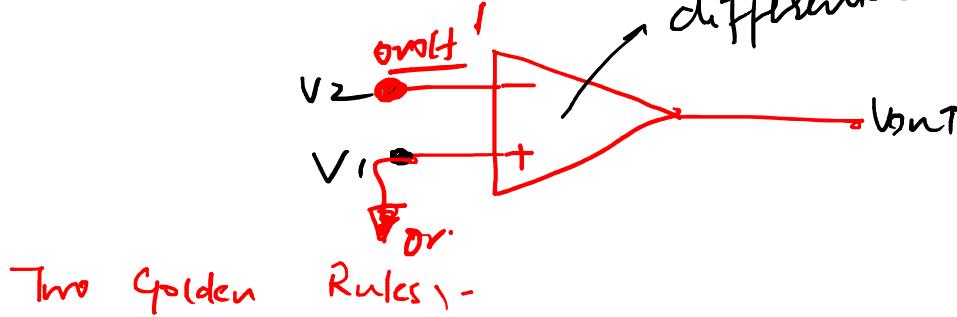
VLSI Design, ECE

School of Computer Science and Engineering



Digital to Analog Converters

1. Binary Weighted Resistor
2. R-2R Ladder



- ① No current enters inside an op amp: $[Z_{in} = \infty]$ $\left[I = \frac{V}{Z_{in}} = 0 \right]$
- ② Virtual Short / virtual ground: $[Gain = \infty] \therefore$ $V_1 = V_2 = 0$

$$V_{out} = \check{A} (V_1 - V_2)$$

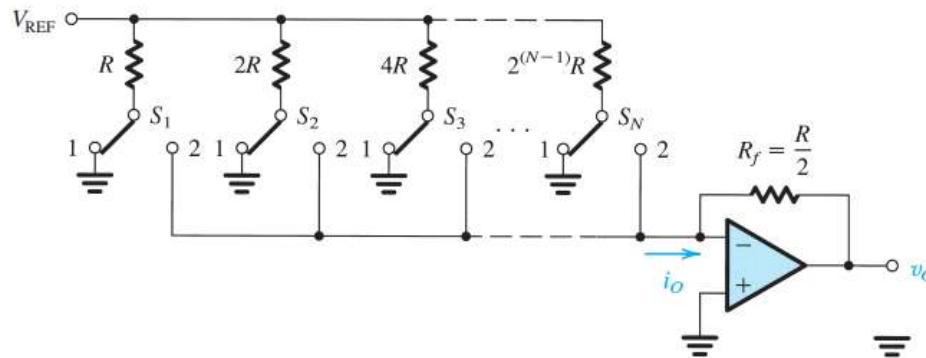
$$\Rightarrow V_1 - V_2 = \frac{V_{out}}{A \rightarrow \infty} = 0$$

$$\boxed{V_1 = V_2}$$
 $V = \text{Short}$

$$\nexists V_1 \text{ or } V_2 = 0$$

$$\boxed{V_1 = V_2 = 0}$$

Binary Weighted Resistor



$$\begin{aligned}
 i_O &= \frac{V_{\text{REF}}}{R}b_1 + \frac{V_{\text{REF}}}{2R}b_2 + \cdots + \frac{V_{\text{REF}}}{2^{N-1}R}b_N \\
 &= \frac{2V_{\text{REF}}}{R} \left(\frac{b_1}{2^1} + \frac{b_2}{2^2} + \cdots + \frac{b_N}{2^N} \right)
 \end{aligned}$$

Thus,

$$i_O = \frac{2V_{\text{REF}}D}{R}$$

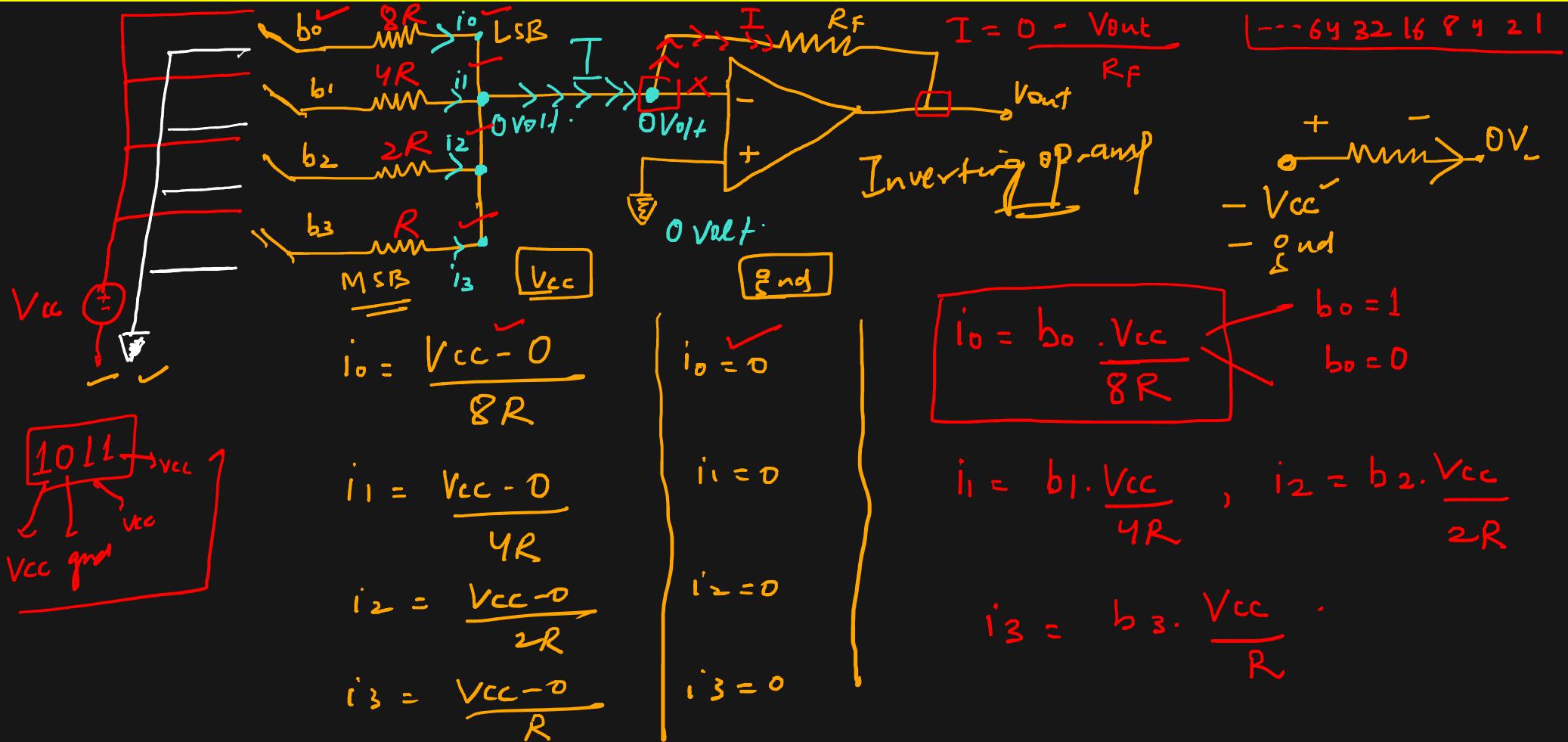
and the output voltage v_O is given by

$$v_O = -i_O R_f = -V_{\text{REF}} D$$

which is directly proportional to the digital word D , as desired.

\Rightarrow DAC

Binary Weighted Resistor



Binary Weighted Resistor

Use KCL :

UBits

$$I = i_0 + i_1 + i_2 + i_3$$

$$\frac{0 - V_{out}}{R_F} = b_0 \frac{V_{cc}}{8R} + b_1 \frac{V_{cc}}{4R} + b_2 \frac{V_{cc}}{2R} + b_3 \frac{V_{cc}}{R} .$$

$$\Rightarrow V_{out} = - \frac{R_F V_{cc}}{8R} [b_0 + 2b_1 + 4b_2 + 8b_3] .$$

$$\Rightarrow V_{out} = - \frac{R_F}{8R} V_{cc} [2^0 b_0 + 2^1 b_1 + 2^2 b_2 + 2^3 b_3]$$

Generalize —————

$R_F = R$

$$V_{out} = - \frac{V_{REF}}{2^{N-1}} [2^0 b_0 + 2^1 b_1 + \dots + 2^{N-1} b_{N-1}]$$

Binary Weighted Resistor

Ques:

4 bit data $\xrightarrow{\text{dig.}}$ b₃b₂b₁b₀ \Rightarrow V_{out}? (BWR)

$V_{REF} = 15 \text{ V}$ $\xrightarrow{\square}$ decimal $\xleftarrow{\square}$

$R_F = \frac{R}{2}$ $\xleftarrow{\square}$

$$V_{out} = - \frac{V_{REF}}{2^{N-1}} \left[2^0 b_0 + 2^1 b_1 + \dots + 2^{N-1} b_{N-1} \right].$$

b ₀	b ₁	b ₂	b ₃
L	1	0	1

$$V_{out} = - \frac{15 \text{ V}}{8} \left[1 \cdot 1 + 2 \cdot L + 0 + 2^3 \times 1 \right]$$

$$1 + 2 + 8 = \boxed{11}$$

$$= \frac{15}{8} \times 11 = \frac{165}{8} = 20.6 \text{ volt}$$

$= 10.3 \text{ volt}$

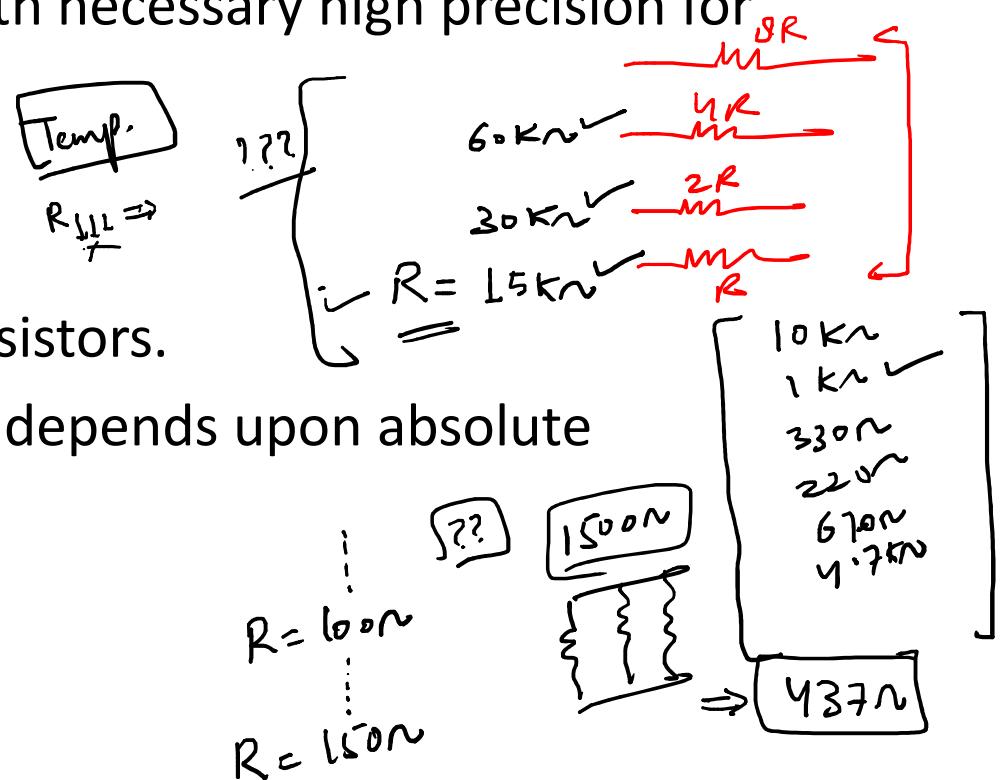
$\xrightarrow{\square}$ Analog output

Advantages of Binary Weighted Resistor DAC

1. Simple in construction
2. Provides faster conversion

Disadvantages of Binary Weighted Resistor DAC

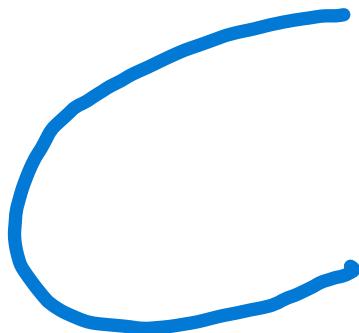
- 1. Requires larger range of resistors with necessary high precision for low resistors.
- 2. It is not easy to maintain the ratio.
- 3. Therefore, Limited to 4 to 8-bits.
- 4. Different current flowing through resistors.
- 5. Accuracy and stability of conversion depends upon absolute accuracy of resistors being used.
- 6. Can be expensive.



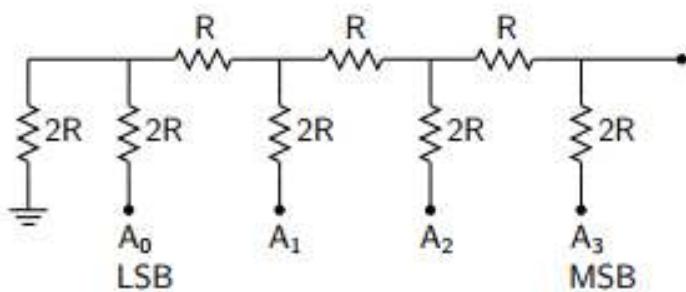
Quick Quiz (POLL)

Identify the incorrect statement about Binary Weighted Resistor?

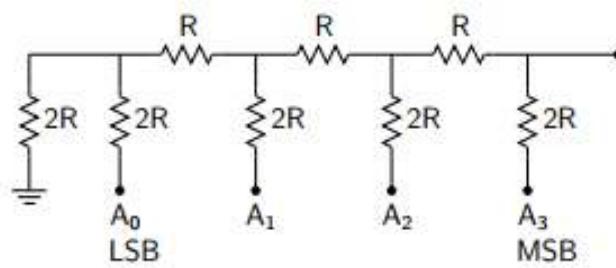
- a) Requires larger range of resistors ✓
- b) Accuracy and stability of conversion depends upon absolute accuracy of resistors ✓
- c) Provides slower conversion ✗
- d) Limited to 4 to 8-bits. ✓



R-2R Ladder DAC

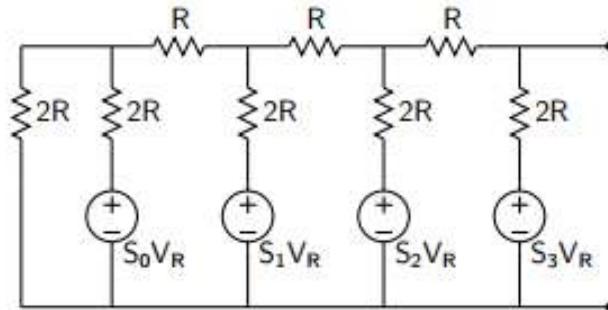


Node A_k is connected to V_R if input bit S_k is 1; else, it is connected to ground.

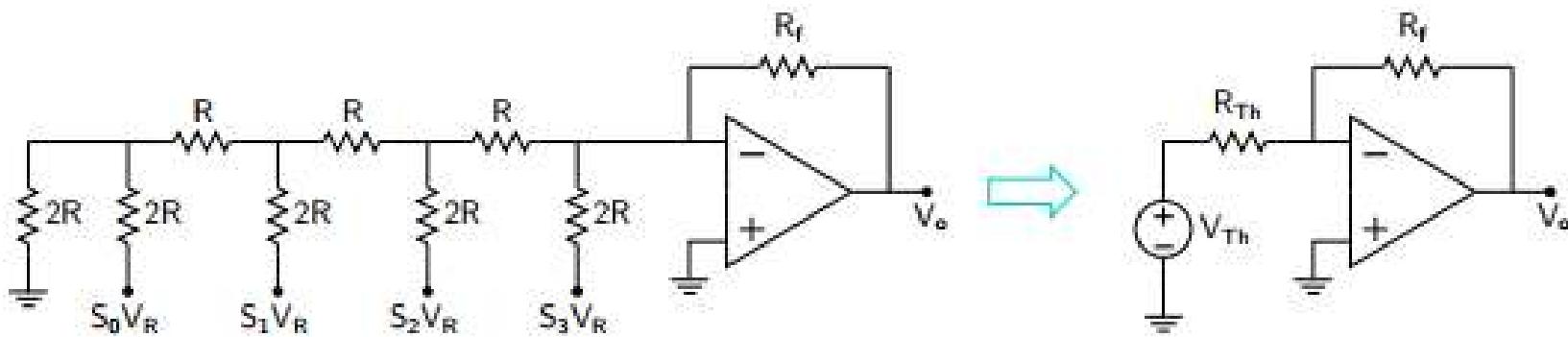


Node A_k is connected to V_R if input bit S_k is 1; else, it is connected to ground.

The original network is equivalent to

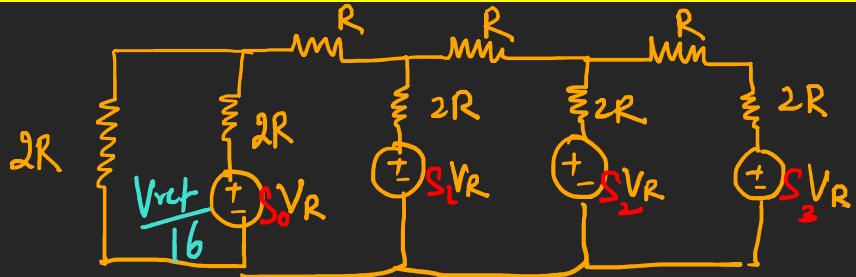


R-2R Ladder DAC



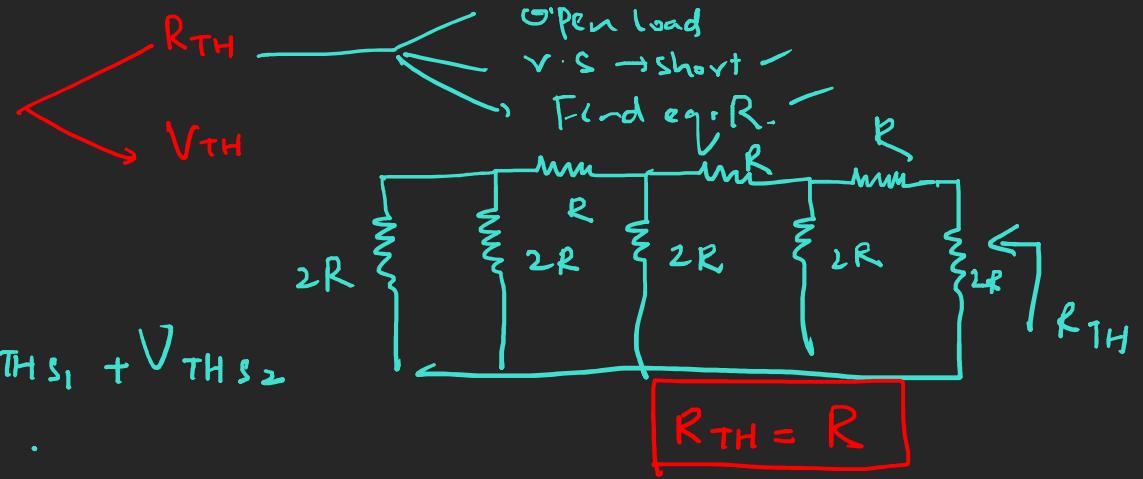
- * $V_o = -\frac{R_f}{R_{Th}} V_{Th} = -\frac{R_f}{R_{Th}} \frac{V_R}{16} [S_0 \cancel{2^0} + S_1 \cancel{2^1} + S_2 \cancel{2^2} + S_3 \cancel{2^3}]$ R/2 ℓ *
- * For an N-bit DAC, $V_o = -\frac{R_f}{R_{Th}} V_{Th} = -\frac{R_f}{R_{Th}} \frac{V_R}{2^N} \sum_{k=0}^{N-1} S_k 2^k$. *
- * 6- to 20-bit DACs based on the R-2R ladder network are commercially available in monolithic form (single chip).

R-2R Ladder DAC



$V_{TH}:$ Superposition

$$V_{TH} = V_{THs_0} + V_{THs_1} + V_{THs_2} + V_{THs_3}$$

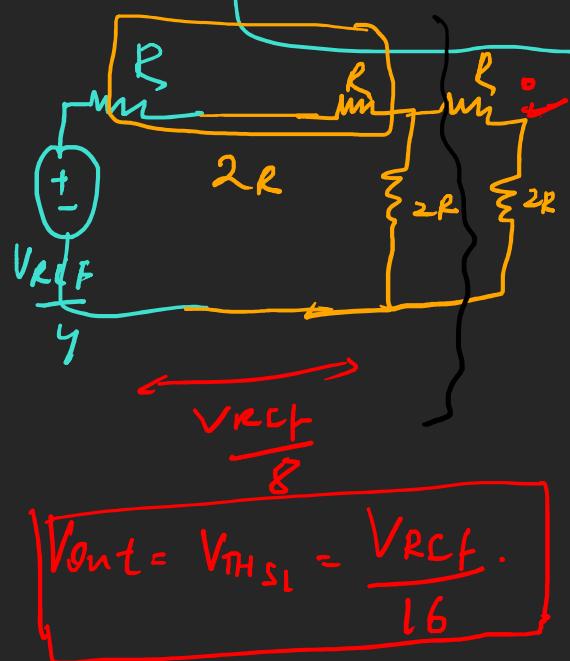
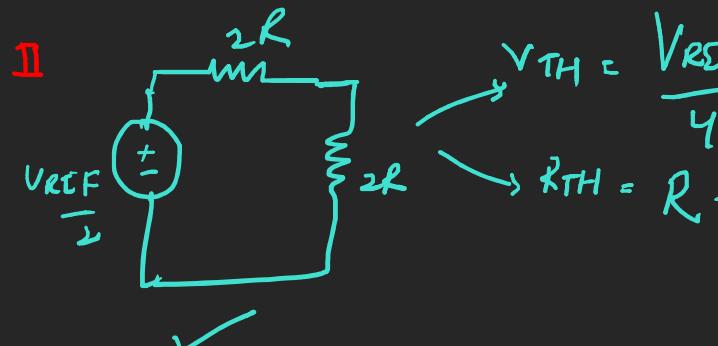
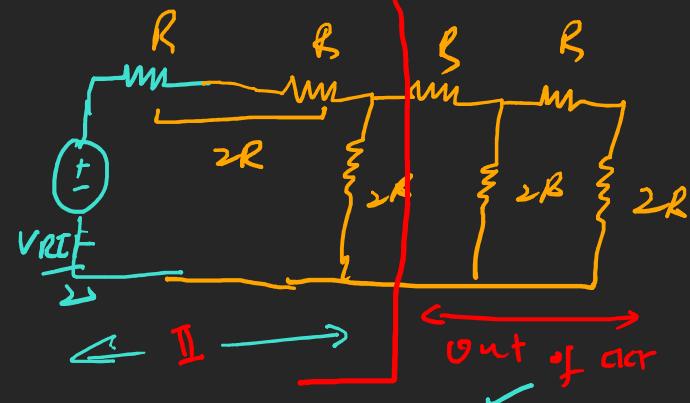
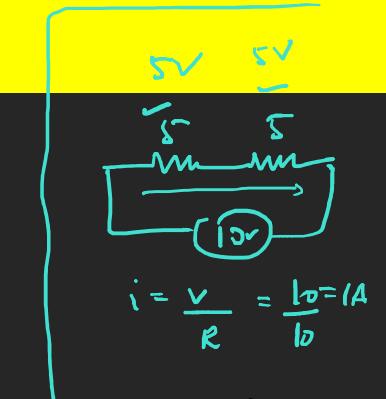
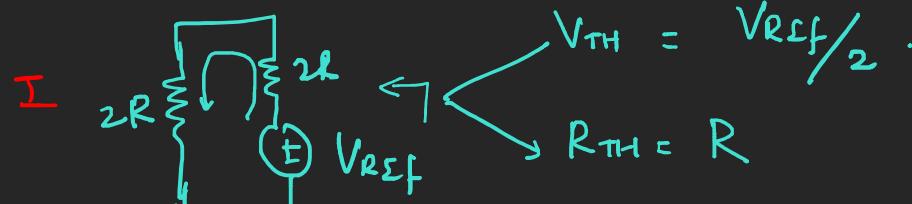
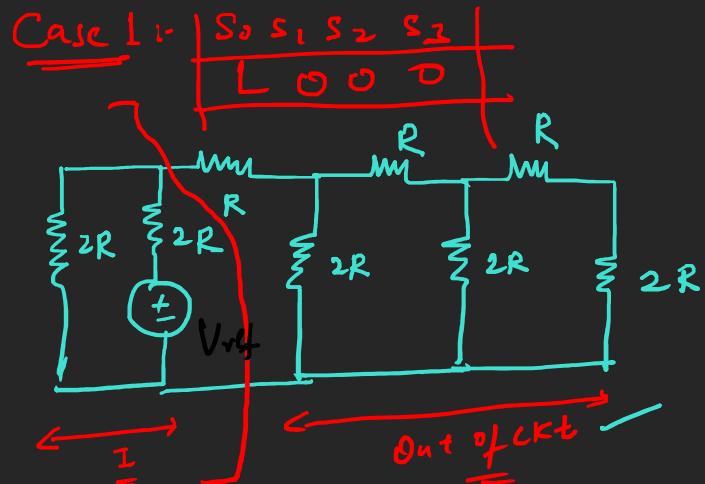


Case 1:-

S_0	S_1	S_2	S_3
1	0	0	0

:-

R-2R Ladder DAC



DIGITAL ELECTRONICS: ECE 213

**Topic: CONVERTERS: Analog to
Digital and Digital to Analog**

**UNIT VI: MEMORY AND
CONVERTERS**

Lecture No.: 47 (TUT)

Prepared By: Irfan Ahmad Pindoo

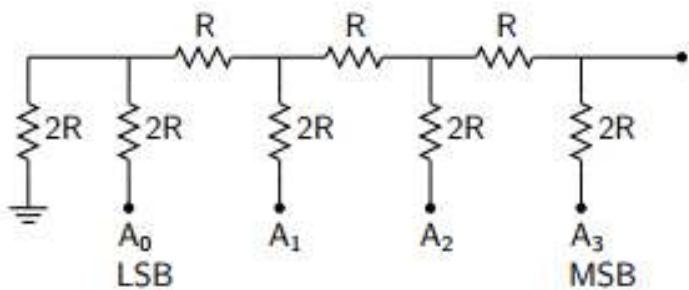
Assistant Professor

VLSI Design, ECE

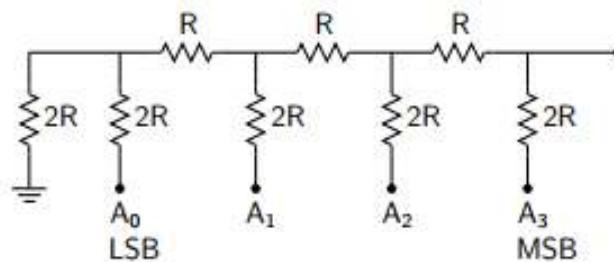
School of Computer Science and Engineering



R-2R Ladder DAC



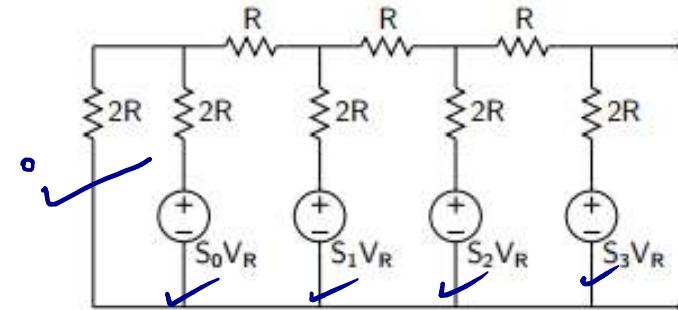
Node A_k is connected to V_R if input bit S_k is 1; else, it is connected to ground.



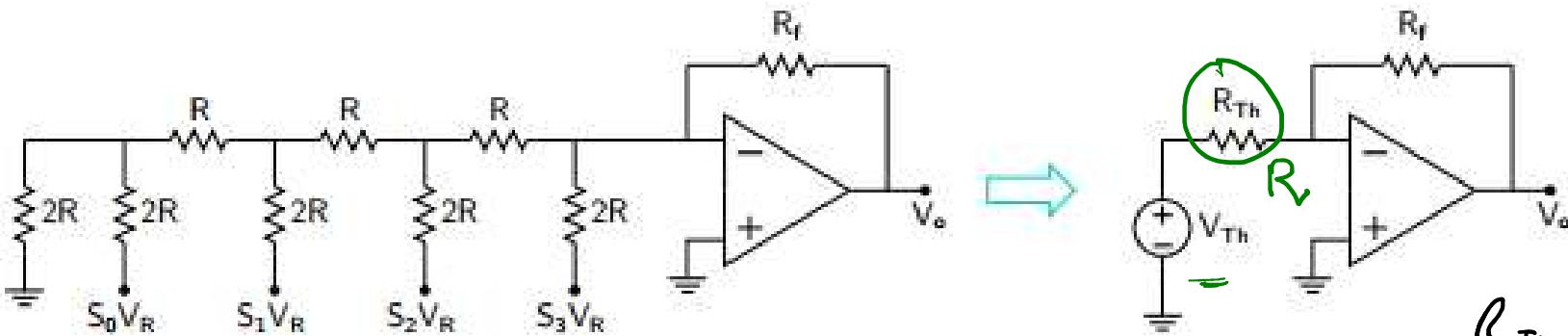
Node A_k is connected to V_R if input bit S_k is 1; else, it is connected to ground.

The original network is equivalent to

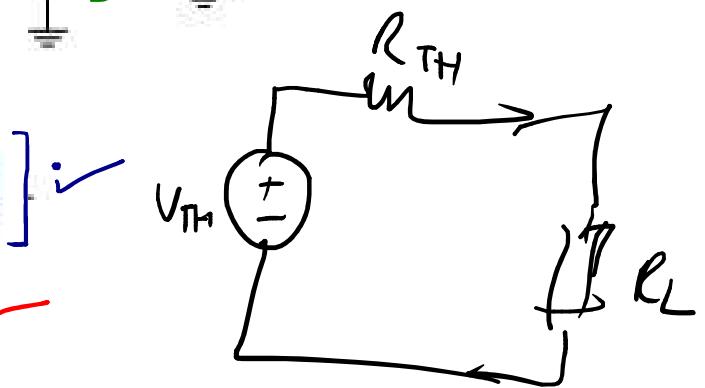
\rightarrow Indep sources
 \rightarrow Dep. sources -



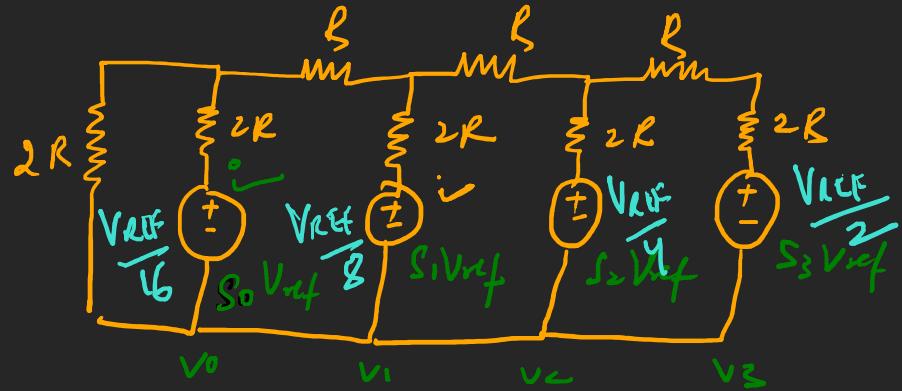
R-2R Ladder DAC



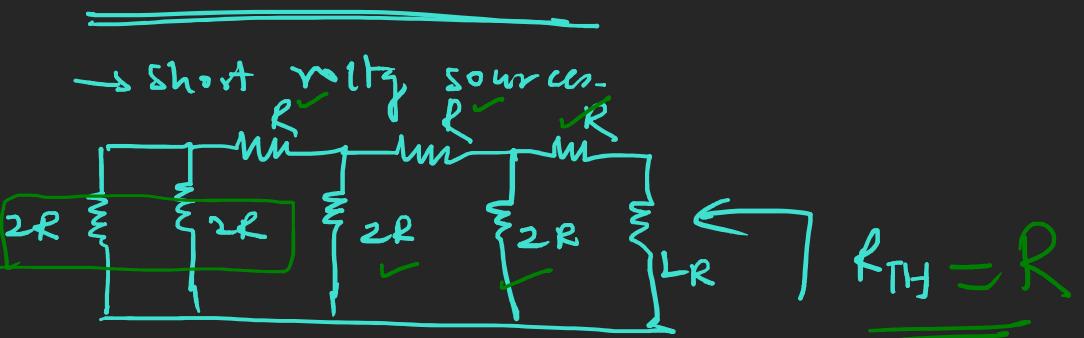
- * $V_o = -\frac{R_f}{R_{Th}} V_{Th} = -\frac{R_f}{R_{Th}} \frac{V_R}{16} [S_0 2^0 + S_1 2^1 + S_2 2^2 + S_3 2^3]$
- * For an N-bit DAC, $V_o = -\frac{R_f}{R_{Th}} V_{Th} = -\frac{R_f}{R_{Th}} \frac{V_R}{2^N} \sum_{k=0}^{N-1} S_k 2^k$.
- * 6- to 20-bit DACs based on the R-2R ladder network are commercially available in monolithic form (single chip).



R-2R Ladder DAC



R_{TH} Calculation :



~~Superposition~~

$$V_{TH} = V_{0_{TH}} + V_{1_{TH}} + V_{2_{TH}} + V_{3_{TH}}$$

At a time, take effect of only one voltage source

$S_0 \rightarrow$ voltage source

$S_1, S_2, S_3 \Rightarrow$ shorted out

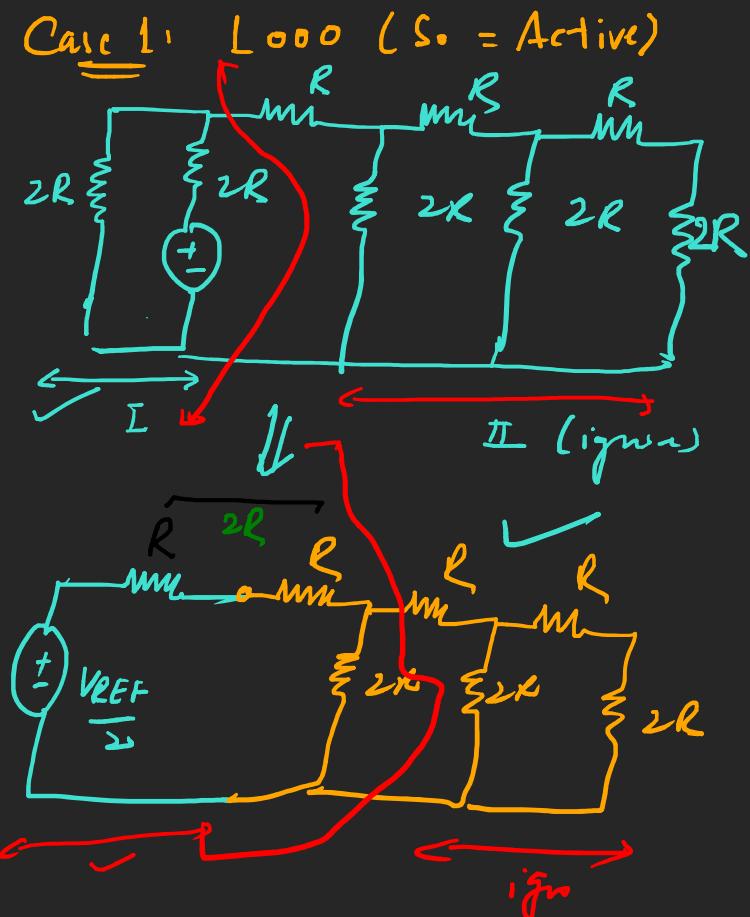
Case 3: $0010 \vdash V_2$

Case 4: $0001 \vdash V_3$

Case 1: $\underline{\underline{1000}} : V_0 \underline{\underline{\underline{\underline{x}}}}$

Case 2: $\underline{\underline{0100}} : V_1 \underline{\underline{\underline{\underline{x}}}}$

R-2R Ladder DAC



$$R_{TH} = \frac{2R}{2^0 + 2^1 + 2^2} = \frac{2R}{7}$$

$$V_{out} = \frac{V_{REF}}{7} \cdot I$$



$$R_{TH} = \frac{2R}{2^0 + 2^1 + 2^2} = \frac{2R}{7}$$

$$V_{out} = \frac{V_{REF}}{7} \cdot I$$

$\frac{V_{REF}}{8} \rightarrow \text{next steps}$

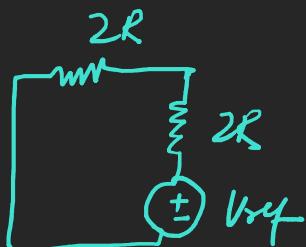
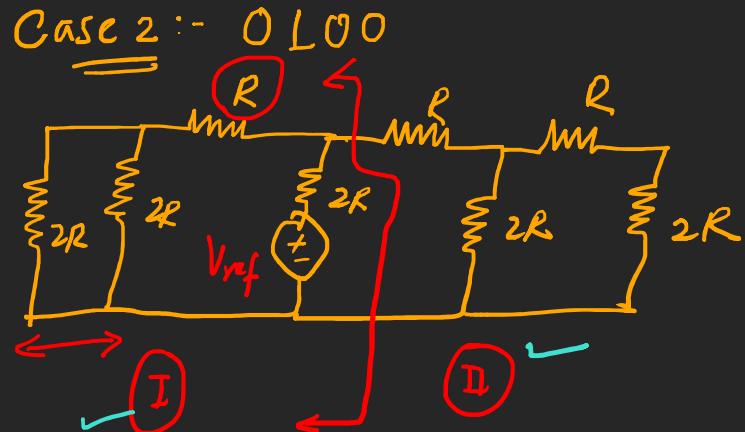
Region I (Top): $V_{out} = V_{REF} \cdot \frac{I}{2^0}$

Region II (Bottom): $V_{out} = V_{REF} \cdot \frac{I}{2^0}$

$$V_{out} = \frac{V_{REF}}{16} \cdot I, \quad R_{TH} = R$$

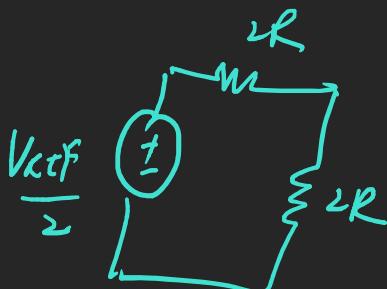
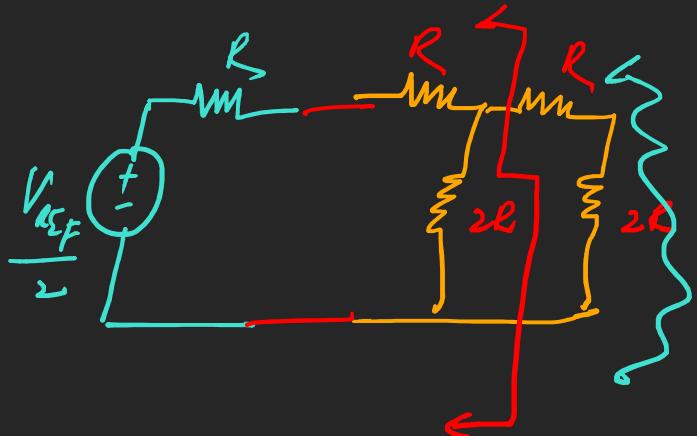
R-2R Ladder DAC

Case 2 :- 0100



$$R_{TH} = R$$

$$V_{TH1} = \frac{V_{ref}}{2}$$

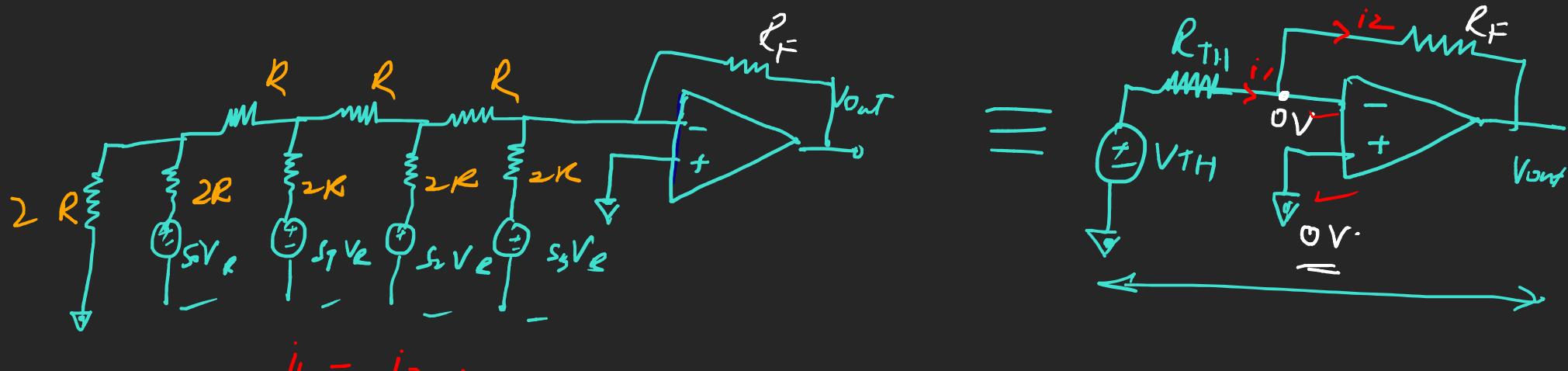


$$R_{TH} = R$$

$$V_{TH2} = \frac{V_{ref}}{8}$$

$$V_{out} = V_{ref} \cdot \frac{1}{8}$$

R-2R Ladder DAC



$$i_1 = i_2.$$

$$\frac{V_{TH} - 0}{R_{TH}} = \frac{0 - V_{out}}{R_f} \Rightarrow V_{out} = -\frac{R_f}{R_{TH}} V_{TH}$$

$$V_{TH} = V_{TH,S_0} + V_{TH,S_1} + V_{TH,S_2} + V_{TH,S_3}$$

$$\left[S_0 \frac{V_{ref}}{16} + S_1 \frac{V_{ref}}{8} + S_2 \frac{V_{ref}}{4} + S_3 \frac{V_{ref}}{2} \right]$$

(1)

(2)

R-2R Ladder DAC

$$V_{out} = \frac{-R_f}{R_{TH}} \left[S_0 \frac{V_{ref}}{16} + S_1 \frac{V_{ref}}{8} + S_2 \frac{V_{ref}}{4} + S_3 \frac{V_{ref}}{2} \right]$$

$$V_{out} = -\frac{R_f}{R_{TH}} \cdot \frac{V_{ref}}{16} \left[S_0 2^0 + S_1 2^1 + S_2 2^2 + S_3 2^3 \right]$$

Generalized eqn:

$$\begin{aligned} V_{out} &= \frac{-R_f}{R_{TH}} \cdot \frac{V_{ref}}{2^N} \left[S_0 2^0 + S_1 2^1 + \dots + S_{N-1} 2^{N-1} \right] \\ &= \frac{-R_f}{R_{TH}} \cdot \frac{V_{ref}}{2^N} \sum_{k=0}^{N-1} S_k 2^k \end{aligned}$$

Advantages of R-2R Ladder

1. Only two resistor values are required, so it is easy to select and design more accurate resistors.
2. If more number of bits are present in the digital input, then we have to include required number of R-2R sections additionally.
3. It is cheap and easy to manufacture.

Due to the above advantages, R-2R Ladder DAC is preferable over binary weighted resistor DAC.

no. of bits $\uparrow\uparrow$: Accuracy Higher

Quick Quiz (POLL)

What is the major advantage of the R/2R ladder digital-to-analog (DAC), as compared to a binary-weighted digital-to-analog DAC converter??

- a) It only uses two different resistor values. ✓
- b) It has fewer parts for the same number of inputs. ✗
- c) Its operation is much easier to analyze. ✗
- d) The virtual ground is eliminated and the circuit is therefore easier to understand and troubleshoot. ✗



Problem 1

In a Binary weighted Resistor DAC, if $V_{REF} = 10V$,
 $R_F = R$. Find ?

- a) Resolution → minm amt. of change that would be detected by o/p.
 b) Full scale output (for 4 bits)

input: Inverting terminal.

Sol: $V_{out} = -\frac{V_{REF}}{2^{N-L}} [2^0 b_0 + 2^1 b_1 + 2^2 b_2 + 2^3 b_3 \dots]$

Resolution!

a) $Resol = \frac{10V}{2^{4-1}} = \frac{10V}{8} = 1.25$

b) $\frac{b_0 b_1 b_2 b_3}{1 1 1 1} = \text{full scale}$

$V_{out} = -\frac{10V}{2^{4-1}} [1 + 2 + 4 + 8] = \frac{-10 \times 15}{8}$

$V_{out} = -18.75 \text{ Volt}$



Problem 2

In a Binary weighted resistor DAC,
with $V_{ref} = -5V$ and $R_f = R$, find the
analog output voltage, if data provided
at input is $D_3 D_2 D_1 D_0 = \underline{1} \underline{0} \underline{1} \underline{L}$.

$$\begin{aligned} \text{Given: } V_{out} &= -\frac{V_{ref}}{2^{N-1}} [2^0 b_0 + 2^1 b_1 + \dots + 2^{N-1} b^{N-1}] \\ &= -\frac{-5V}{2^{4-1}} [2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 0 + 2^3 \cdot 1] \\ &= -\frac{5}{8} [1 + 2 + 8] \\ &= +\frac{5}{8} \times 11 = \frac{55}{8} \Rightarrow V_{out} = 6.875 \text{ Volts} \end{aligned}$$

Problem 3

For the 4-bit weighted-resistor DAC:

(a) Determine the **full scale** output if

$$R_f = R = 1 \text{ k}\Omega, \text{ & } V_{REF} = 5V.$$

(b) Find **full scale** output if R_f is changed to 500Ω .

$$\textcircled{a} \quad V_{out} = - \frac{5V}{2^3} [1+2+4+8]$$

$$= - \frac{5}{8} \times 15 = - \frac{75}{8}$$

$$\Rightarrow V_{out} = -9.375 \text{ Volt}$$

$$V_{out} = - \frac{V_{REF} R_f}{2^{N-1} R} [2^0 b_0 + 2^1 b_1 + \dots + 2^{N-1} b_N]$$

\textcircled{b}

$$V_{out} = - \frac{5V}{2^3} \frac{500\Omega}{1000\Omega} [1+2+4+8]$$

$$= -9.375 \times \frac{1}{2}$$

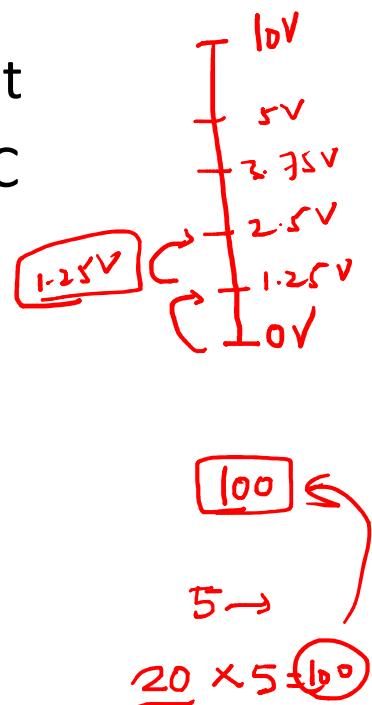
$$\Rightarrow V_{out} = -4.6875 \text{ Volt}$$

Parameters of DAC

1. Resolution or Step size
2. Accuracy
3. Settling Time
4. Off Set Voltage
5. Monotonicity

Resolution or Step size

- Smallest change in the analog output for a change in digital input
- Reciprocal of number of discrete steps in full scale output of DAC
- It is equivalent to the size of LSB
- $\% \text{Resolution} = \frac{\text{Step size}}{\text{full scale}} \times 100\%$
- * • Since, full scale = number of steps \times step size, therefore
- $\% \text{Resolution} = \frac{1}{\text{Total number of steps}} \times 100\%$



Quick Quiz (POLL)

MCQ, ETE

Determine the resolution of 6-bit DAC in terms of percentage?

- a) 5.1%
- b) 15.7%
- c) ~~1.587%~~
- d) 2%

$$\boxed{\text{Resol} = \frac{1}{2^N - 1}}$$

$$= \frac{1}{63} \times 100\%$$

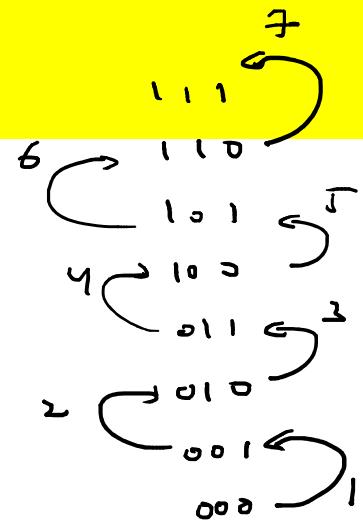
=

N = 6

$$\text{Resol} = \frac{\text{Step size}}{\text{Full scale}}$$

$$\text{Resol} = \frac{1}{\text{Total no. of steps}}$$

$$\begin{array}{ccc} N \rightarrow \text{Bits} & \longrightarrow & 3 \text{ bits} \\ 2^N \rightarrow \text{Levels} & \longrightarrow & 8 \text{ levels} \\ 2^N - 1 \leftarrow \text{steps} & \longrightarrow & 7 \text{ steps} \end{array}$$



Problem

A 6 bit DAC has a step size of 50 mV. Determine the full scale output voltage and percentage resolution?

$$\text{Sol: } \text{Resolution} = \frac{1}{2^N - 1} = \frac{1}{2^6 - 1} = \frac{1}{64 - 1} = \frac{1}{63} = 1.587\%.$$

Full scale o/p = Total no. of steps \times step size.

$$= (2^N - 1) \times 50 \text{ mV} = 63 \times 50 \text{ mV}$$
$$\Rightarrow V_{\text{out}} = 3.15 \text{ Volts}$$

Problem

An 8 bit DAC produces $V_{out} = 0.05 V$ for a digital input of 00000001. lowest no. (minimum change)
Determine the full scale output voltage and percentage resolution?
Also find V_{out} for an input of 00101010?

(a) Resolution = $\frac{1}{2^N - 1} = \frac{1}{255} = 0.00396 \times 100\% =$

(b) Full scale o/p = Step size \times Total no. of steps
= 0.05×255
= 12.75 Volts.

(c) $\begin{array}{r} 00101010 \\ \hline 32 \quad 8 \quad 2 \end{array} \Rightarrow \text{decimal equiv.} = \underline{\underline{42}}$

$$\boxed{V_{out} \rightarrow 0.05 \times 42 = 2.1 \text{ Volt.}}$$

DIGITAL ELECTRONICS: ECE 213

**Topic: CONVERTERS: Analog to
Digital and Digital to Analog**
**UNIT VI: MEMORY AND
CONVERTERS**

Lecture No.: 48

Prepared By: Irfan Ahmad Pindoo

Assistant Professor

VLSI Design, ECE

School of Computer Science and Engineering



Parameters of DAC

1. Resolution or Step size
2. Accuracy
3. Settling Time
4. Off Set Voltage
5. Monotonicity

Resolution or Step size

- Smallest change in the analog output for a change in digital input
- Reciprocal of number of discrete steps in full scale output of DAC
- It is equivalent to the size of LSB
- $\%Resolution = \frac{Step\ size}{full\ scale} \times 100\%$
- Since, full scale = number of steps \times step size, therefore
- $\%Resolution = \frac{1}{Total\ number\ of\ steps} \times 100\%$

Accuracy

- Specified in terms of DAC's full scale error and linearity error which are normally expressed as the percentage of the converter's full scale output.
- Full Scale Error: Maximum Deviation of the DAC's output from the expected value
- Linearity Error: Maximum deviation of analog output from the ideal output.
- Expressed as percentage of full scale.

Settling Time

- The time required for the output to reach the final value and remain **within $\pm \frac{1}{2}$ LSB** after overshoot.
- Gives operating speed.

Offset Voltage

- Ideally, the output of DAC should be zero, when the binary input is Zero.
- However, there is a very small output voltage under this situation called the off set voltage.
- This off set voltage, if not corrected, will be added to the expected DAC output for all input cases

Monotonicity

- A DAC is said to be monotonous, if its **output increases as the binary input is incremented** from one value to the next.
- This means that the stair-case output will have no downward steps as the binary input is incremented from 0 to full scale.

Analog to Digital Converters

Types:

1. Counter Type ADC
2. Successive Approximation Register (SAR) ADC
3. Flash Type ADC

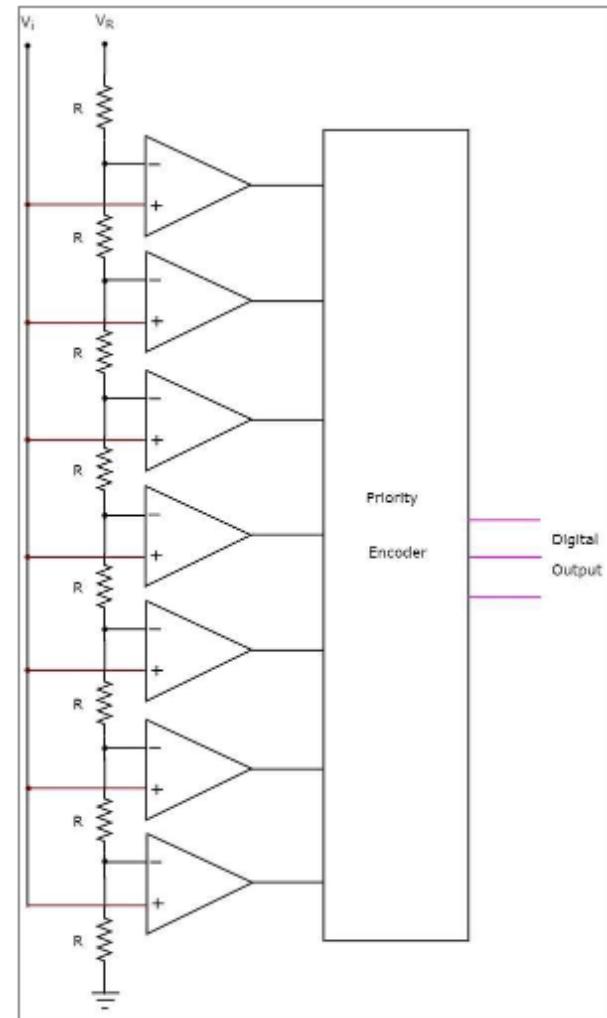
Flash Type ADC

- A **flash type ADC** produces an equivalent digital output for a corresponding analog input in no time. Hence, flash type ADC is the fastest ADC.
- The 3-bit flash type ADC consists of a voltage divider network, 7 comparators and a priority encoder.
- The **circuit diagram** of a 3-bit flash type ADC is shown in the following figure

Flash Type ADC

The working of a 3-bit flash type ADC is as follows.

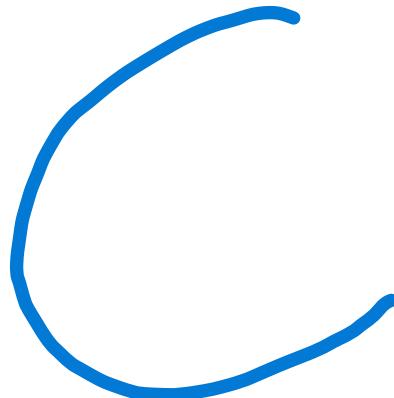
- The voltage divider network contains 8 equal resistors. A reference voltage V_R is applied across that entire network with respect to the ground. The voltage drop across each resistor from bottom to top with respect to ground will be the integer multiples (from 1 to 8) of $\frac{V_R}{8}$.
- The external input voltage V_i is applied to the non-inverting terminal of all comparators. The voltage drop across each resistor from bottom to top with respect to ground is applied to the inverting terminal of comparators from bottom to top.
- At a time, all the comparators compare the external input voltage with the voltage drops present at the respective other input terminal. That means, the comparison operations take place by each comparator parallelly.
- The output of the comparator will be '1' as long as V_i is greater than the voltage drop present at the respective other input terminal. Similarly, the output of comparator will be '0', when, V_i is less than or equal to the voltage drop present at the respective other input terminal.
- All the outputs of comparators are connected as the inputs of priority encoder. This priority encoder produces a binary code (digital output), which is corresponding to the high priority input that has '1'.
- Therefore, the output of priority encoder is nothing but the binary equivalent (digital output) of external analog input voltage, V_i .



QUICK QUIZ (POLL)

Which A/D converter is considered to be simplest, fastest and most expensive?

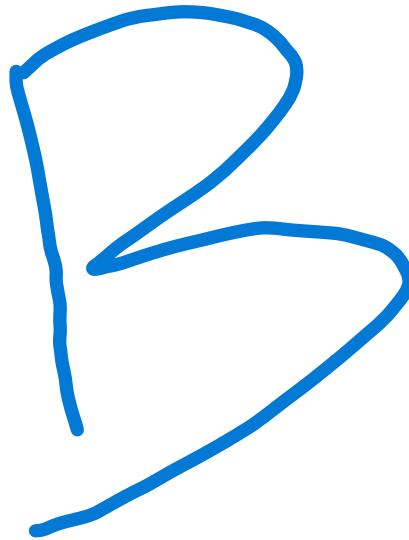
- a) Servo converter
- b) Counter type ADC
- c) Flash type ADC
- d) All of the mentioned



QUICK QUIZ (POLL)

Number of comparators required in 8 bit flash ADC is?

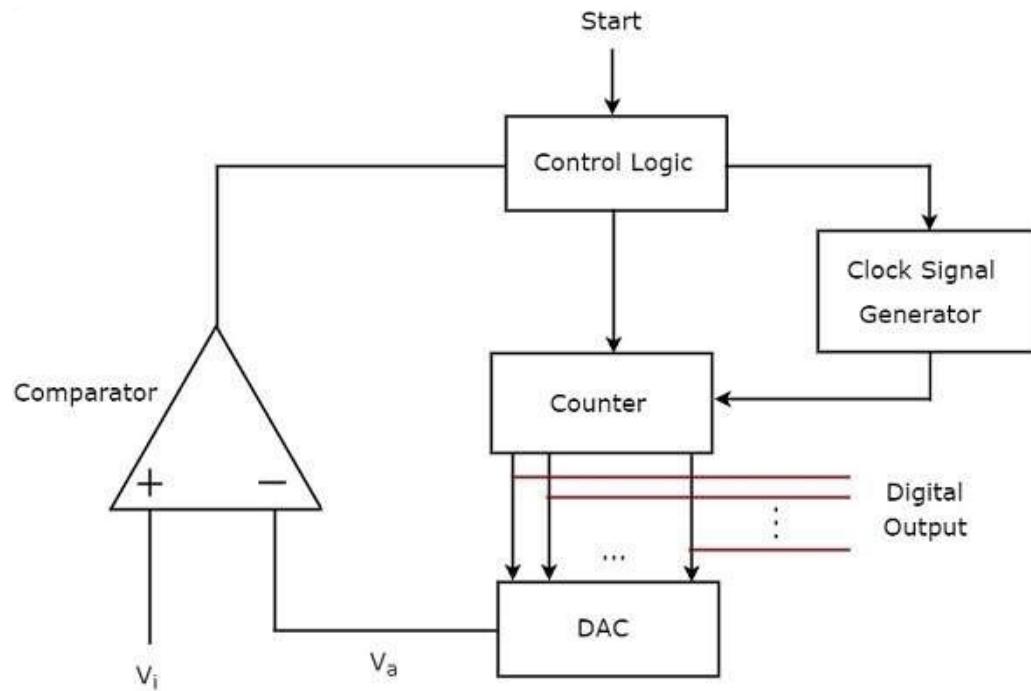
- a) 15
- b) 255
- c) 256
- d) 8

A large, hand-drawn blue question mark is centered on the page, serving as a visual cue for the poll question.

Counter Type DAC

The working of a counter type ADC is as follows –

- The **control logic** resets the counter and enables the clock signal generator in order to send the clock pulses to the counter, when it received the start commanding signal.
- The **counter** gets incremented by one for every clock pulse and its value will be in binary (digital) format. This output of the counter is applied as an input of DAC.
- **DAC** converts the received binary (digital) input, which is the output of counter, into an analog output. Comparator compares this analog value, V_a with the external analog input value V_i .
- The **output of comparator** will be ‘1’ as long as V_i is greater than. The operations mentioned in above two steps will be continued as long as the control logic receives ‘1’ from the output of comparator.
- The **output of comparator** will be ‘0’ when V_i is less than or equal to V_a . So, the control logic receives ‘0’ from the output of comparator. Then, the control logic disables the clock signal generator so that it doesn’t send any clock pulse to the counter.
- At this instant, the output of the counter will be displayed as the **digital output**. It is almost equivalent to the corresponding external analog input value V_i .



QUICK QUIZ (POLL)

Drawback of counter type A/D converter

- a) Counter clears automatically
- b) More complex
- c) High conversion time
- d) Low speed

D

QUICK QUIZ (POLL)

Calculate the conversion time of a 12-bit counter type ADC with 1MHz clock frequent to convert a full scale input?

- a) 4.095 μ s
- b) 4.095ms
- c) 4.095s
- d) None of the mentioned

B

Mat^{m conversion time =}

$$\frac{(2^n - 1)}{f} T_c$$
$$= \frac{(2^{12} - 1)}{1} = 4095 \mu\text{s}$$
$$= 4.095 \text{ms}$$

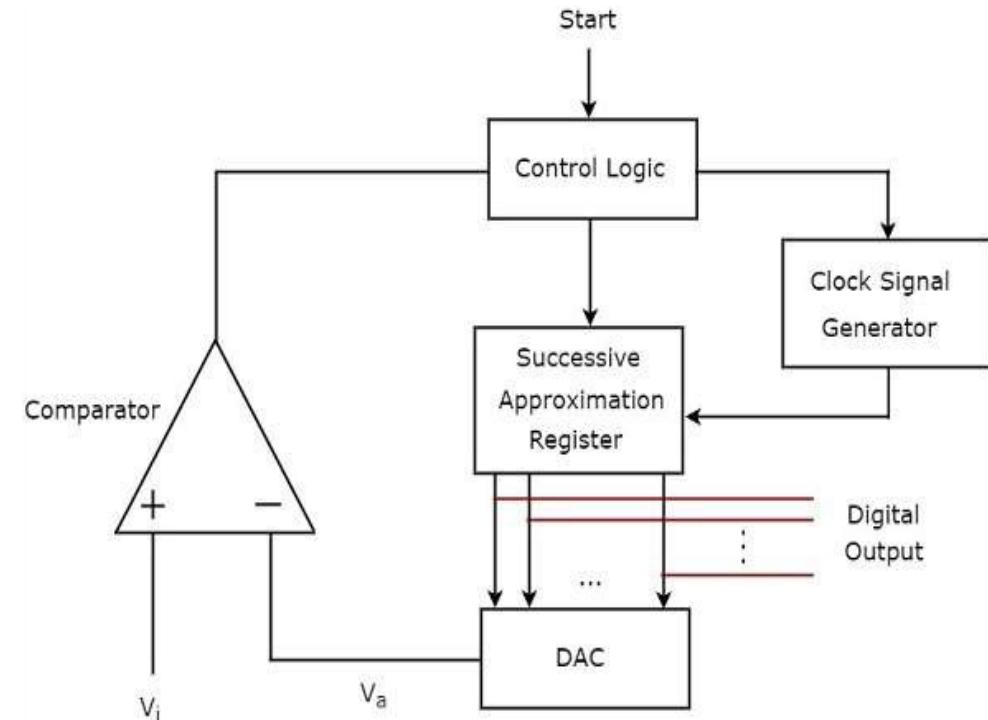
Successive Approximation Register (SAR) ADC

The successive approximation ADC mainly consists of 5 blocks- Clock signal generator, Successive Approximation Register (SAR), DAC, comparator and Control logic.

The **working** of a successive approximation ADC is as follows -

- The **control logic** resets all the bits of SAR and enables the **clock signal generator** in order to send the clock pulses to SAR, when it received the start commanding signal.
- The binary (digital) data present in **SAR** will be updated for every clock pulse based on the output of comparator. The output of SAR is applied as an input of DAC.
- **DAC** converts the received digital input, which is the output of SAR, into an analog output. The comparator compares this analog value V_a with the external analog input value V_i .
- The **output of a comparator** will be '1' as long as V_i is greater than V_a . Similarly, the output of comparator will be '0', when V_i is less than or equal to V_a .
- The operations mentioned in above steps will be continued until the digital output is a valid one.

The digital output will be a valid one, when it is almost equivalent to the corresponding external analog input value V_i .



End of UNIT 6 Syllabus!

The Journey is Over. All the Best for your exams.
Feel free to contact, whenever required.