UNIT-IV

Fundamentals of semiconductor devices and digital circuits

Lecture 23

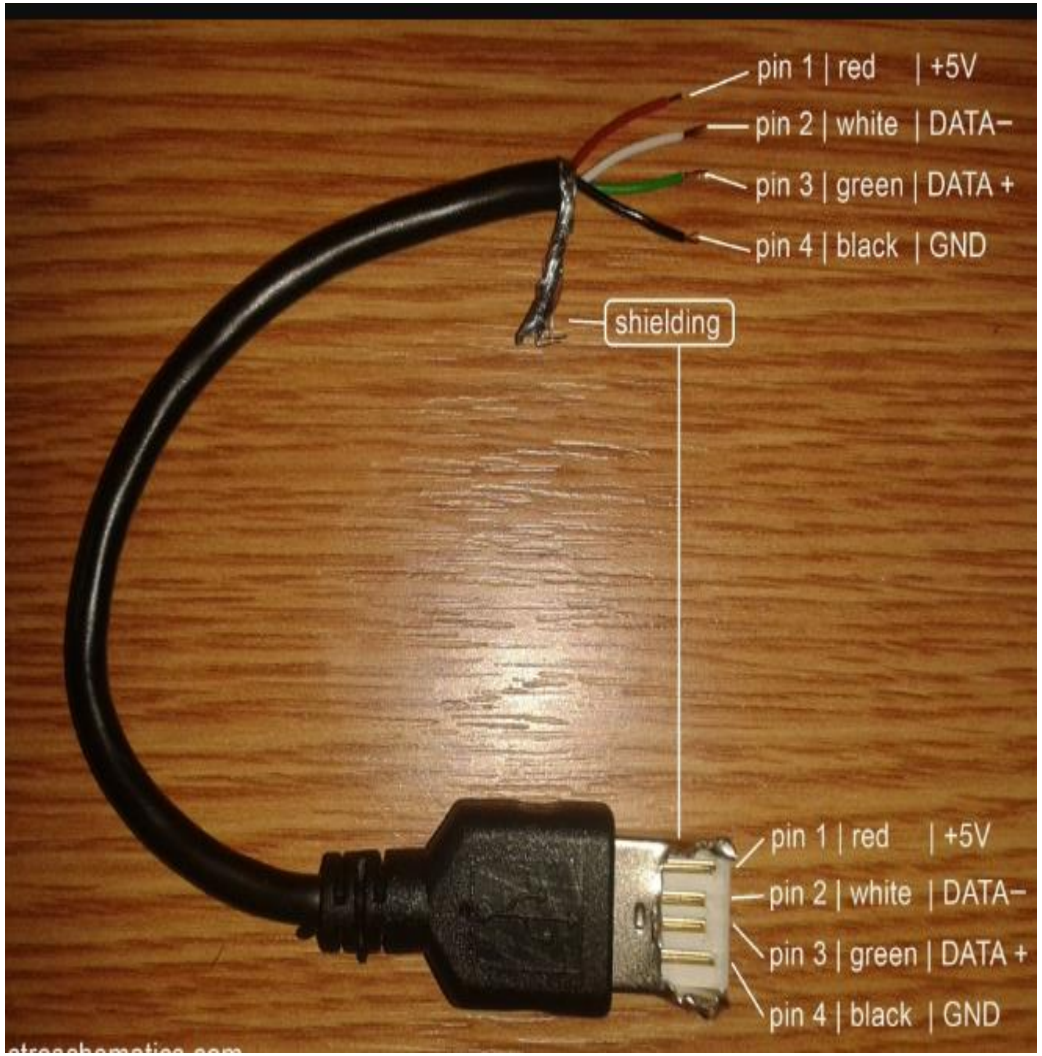
Prepared By:
Pawandeep Kaur
Assistant Professor and Head-ECE

Unit-4 content

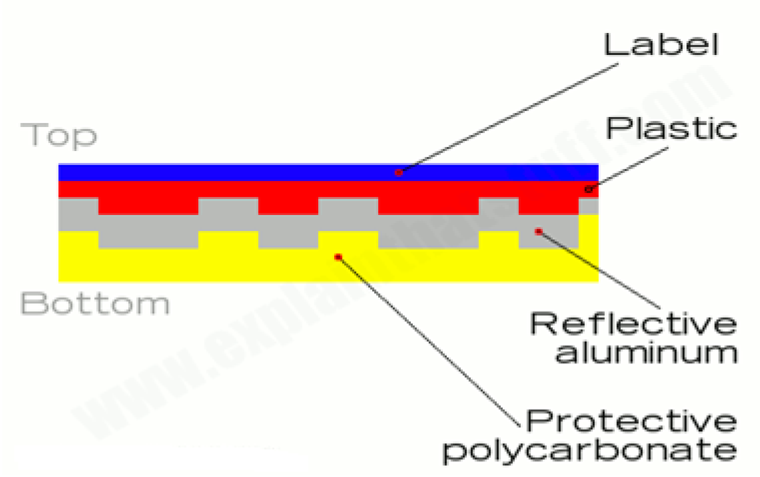
Fundamentals of semiconductor devices and digital circuits : digital abstraction- voltage levels and the static discipline, boolean logic, combinational gates, fan-in and fan-out of gates, noise margin in details, pn junction and zener diode characteristics and analysis, testing of diodes and its applications, basic operation and testing of BJT, MOSFET representation and its characteristics, handling of integrated circuits-ESD phenomena

Digital System and Signal

- A digital **system** processes digital signals
- Examples: computer, cellphone, DVD, digital camera, etc.
- **What is a signal then?**
- A signal is a physical quantity (sound, light, voltage, current) that carries **information**
  - The power/charging cable supplies power but no information (not data)
  - A USB cable carries information (files or data)
- Examples of quantities used as digital information signals
  - Voltage: 5V (logic 1), 0V (logic 0) in digital circuits
  - Pits and lands on the CD surface reflect the light from the laser differently, and that difference is encoded as binary data

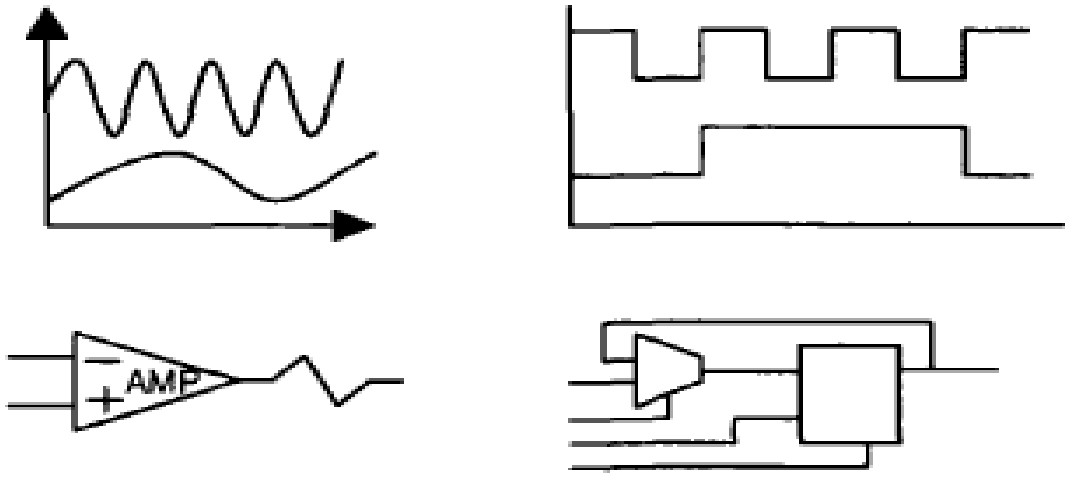


USB Cable Connections



ANALOG VS DIGITAL

- Analog devices and systems: Process analog signals (time-varying signals that can take any value across a continuous range known as dynamic range)
- Digital devices and systems: Process digital signals (analog signals that are modeled as having at any time one of two discrete values)



a. Analog Circuit. b. Digital Circuit.

Example of analog vs digital system

Digital advantages:

Battery life

Programmability

Accuracy

The two levels in the binary representation are variously called
(a) TRUE or FALSE,
(b) ON or OFF,
(c) 1 or 0,
(d) HIGH or LOW.

Quick Quiz (Poll)

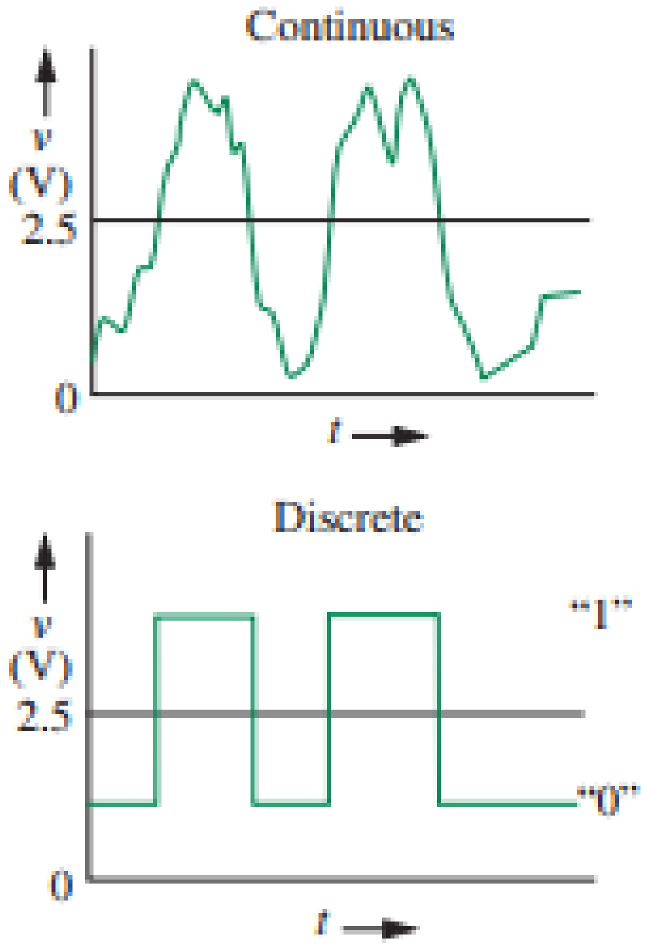
A signal is a way to transmit ________________ by means of gesture, action, sound, or other means.

a) money
b) information
c) time
d) light

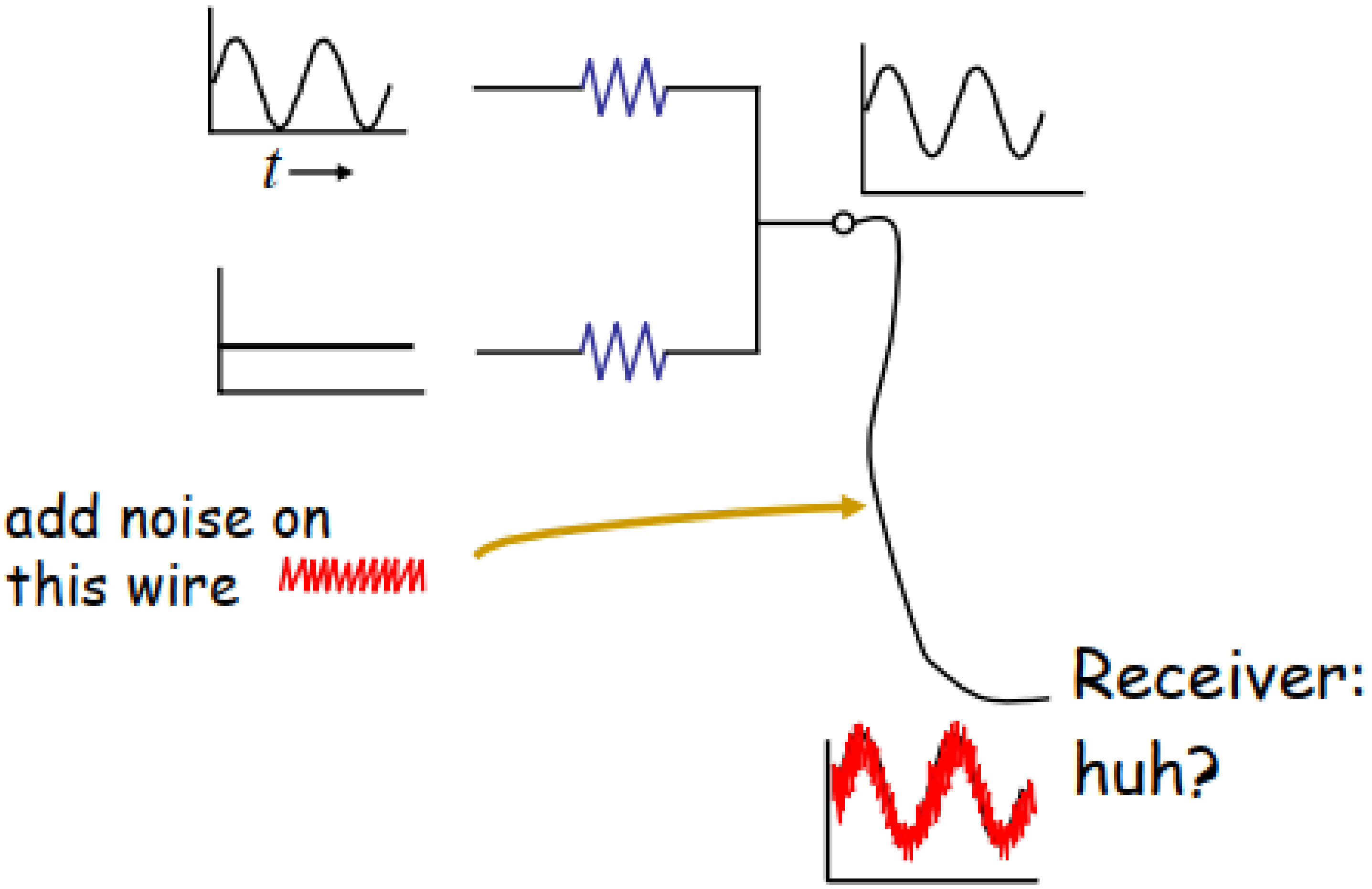
The Digital Abstraction

Value Discretization

- Breaking the value into chunks
- Value discretization forms the basis of the digital abstraction.
- The idea is to lump signal values that fall within some interval into a single value.
- An observed voltage value between 0 volts and 2.5 volts is treated as a "0," and a value between 2.5 volts and 5 volts as a "1."



Noise Problem



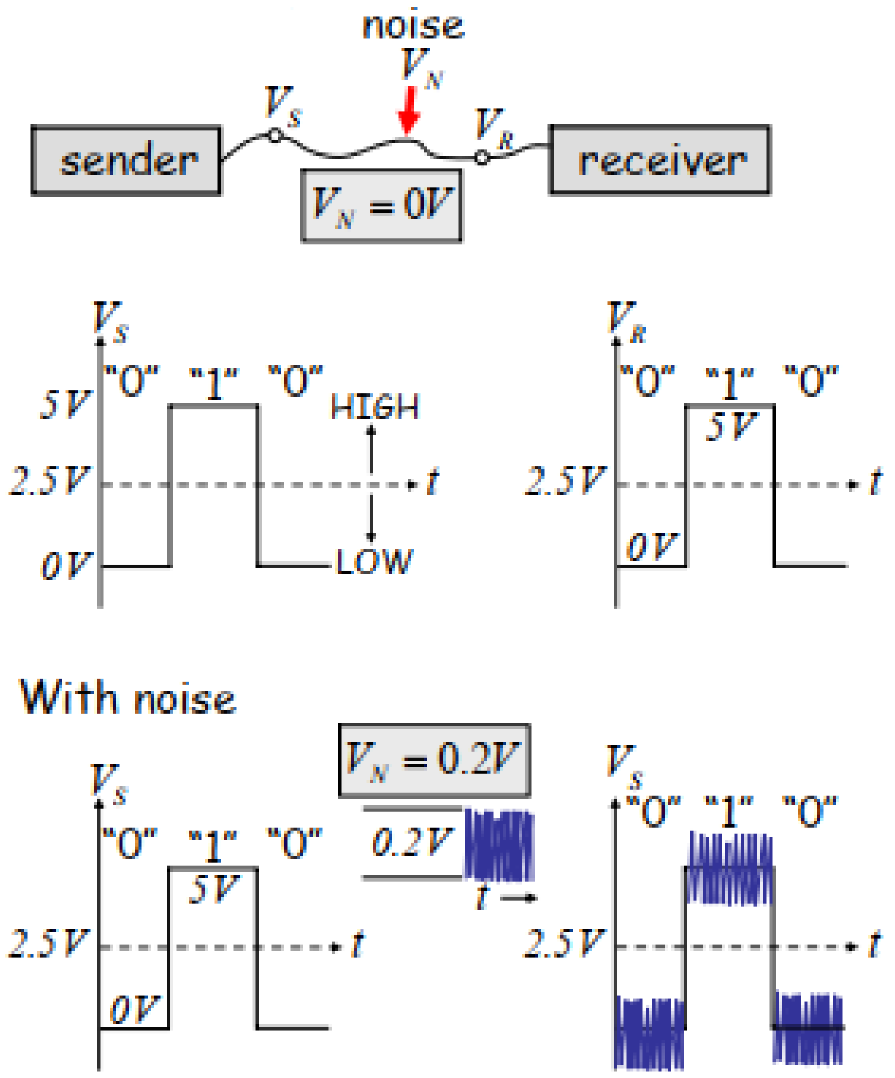
Why is this discretization useful?

For "1": noise margin $5V$ to $2.5V = 2.5V$
For "0": noise margin $0V$ to $2.5V = 2.5V$

when noise is introduced in the medium and digital signal and analog signal both passes through it. The analog signal will be affected more because it varies continuous with time so it is difficult to identify that noise has destroyed which value of voltage

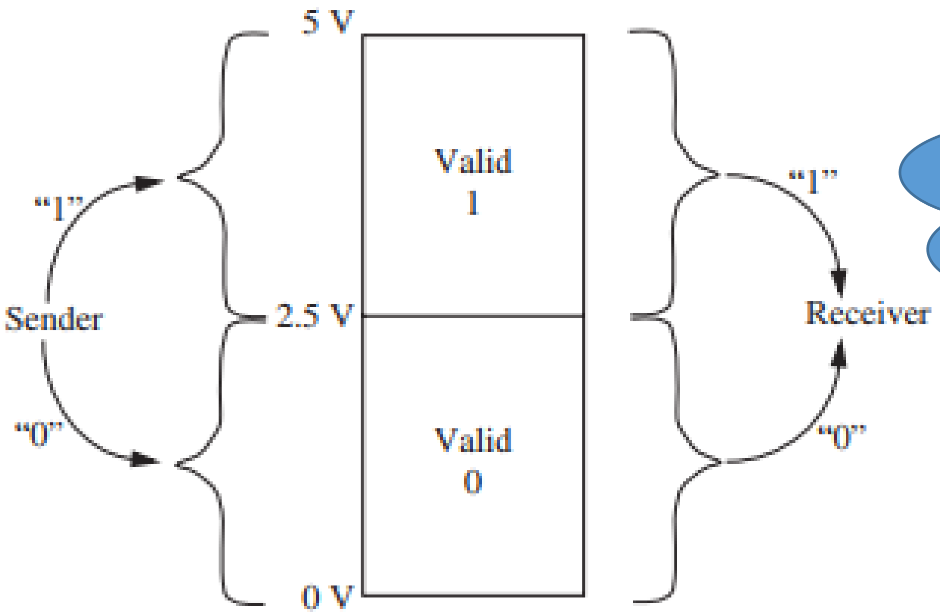
Discretization Provides

- ❑ Better noise immunity.
- ❑ Higher Noise Margin.



Problem -1

- The static discipline requires devices to adhere to a common representation, and to guarantee that they interpret correctly inputs that are valid logical signals according to the common representation, and to produce outputs that are valid logical signals provided they receive valid logical inputs.



Logic 0 : 0.0 V $\leq V <$ 2.5 V.

Logic 1 : 2.5 V $\leq V \leq$ 5.0 V.

- What does the receiver do if it sees a voltage level of **2.5 V on the wire**?
- Does it interpret this signal value as a logical 0 or as a logical 1?

Explanation Slide

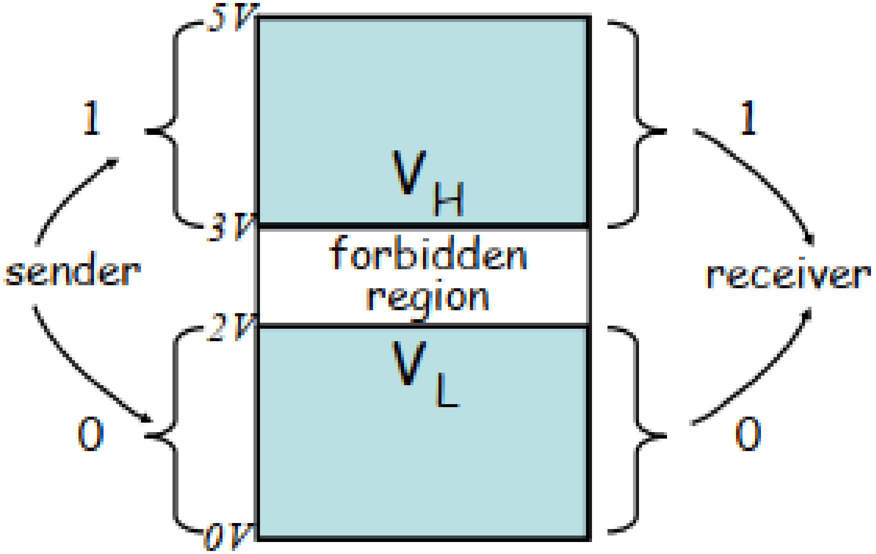
Problem1-Solution

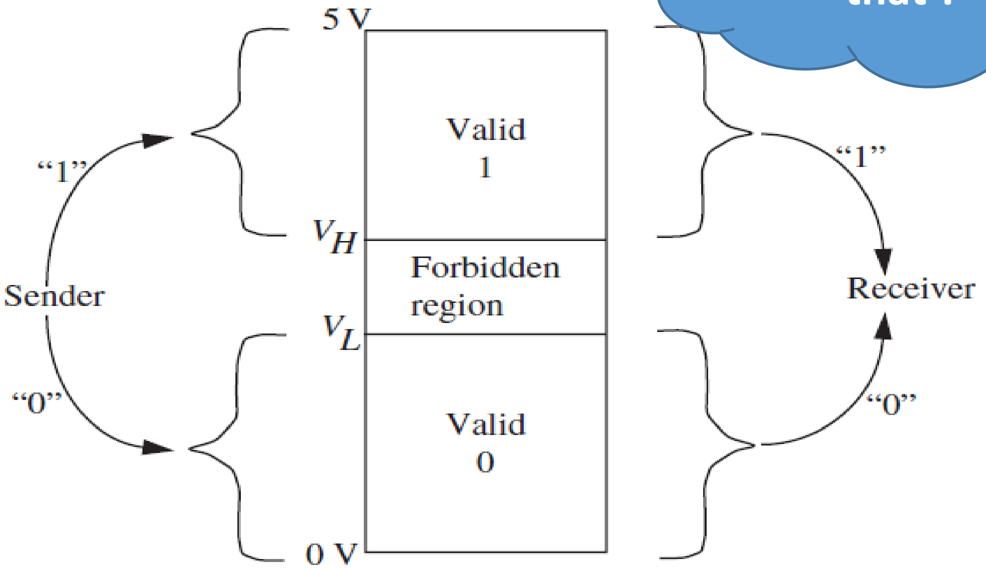
- What does the receiver do if it sees a voltage level of **2.5 V on the wire**?
- Does it interpret this signal value as a logical 0 or as a logical 1?

To eliminate such confusion, we further prescribe a *forbidden region that separates the two valid regions.*

Create a “no-man’s land” or a forbidden Region.

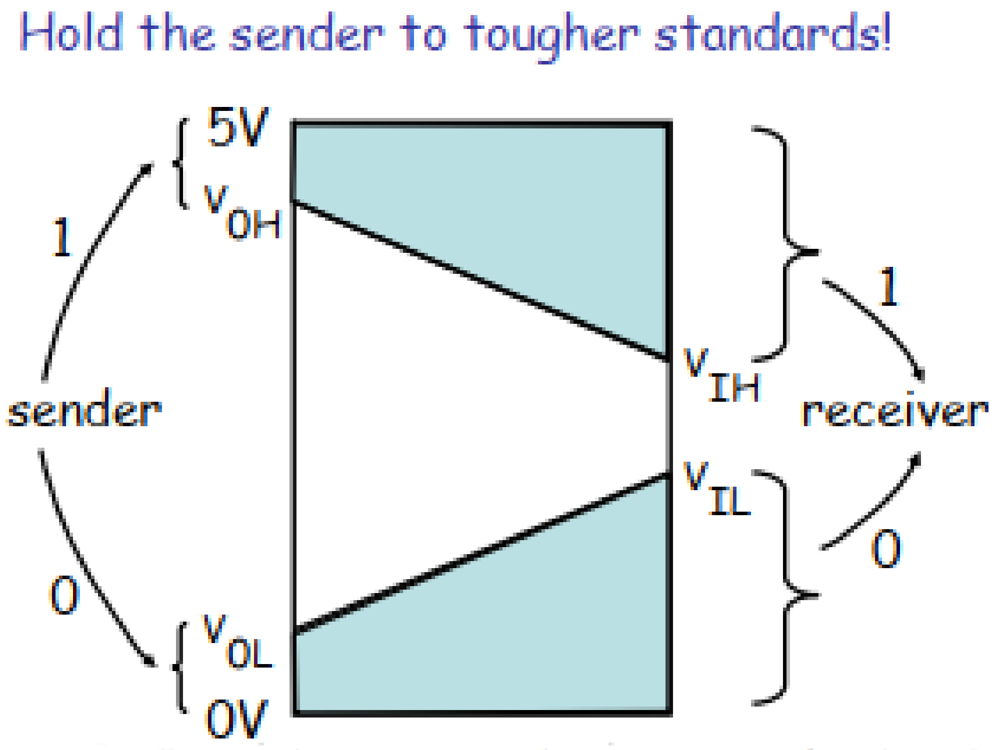
Still Problem-2 exist , What is that ?





Final Solution

- By making tough standards towards sender side

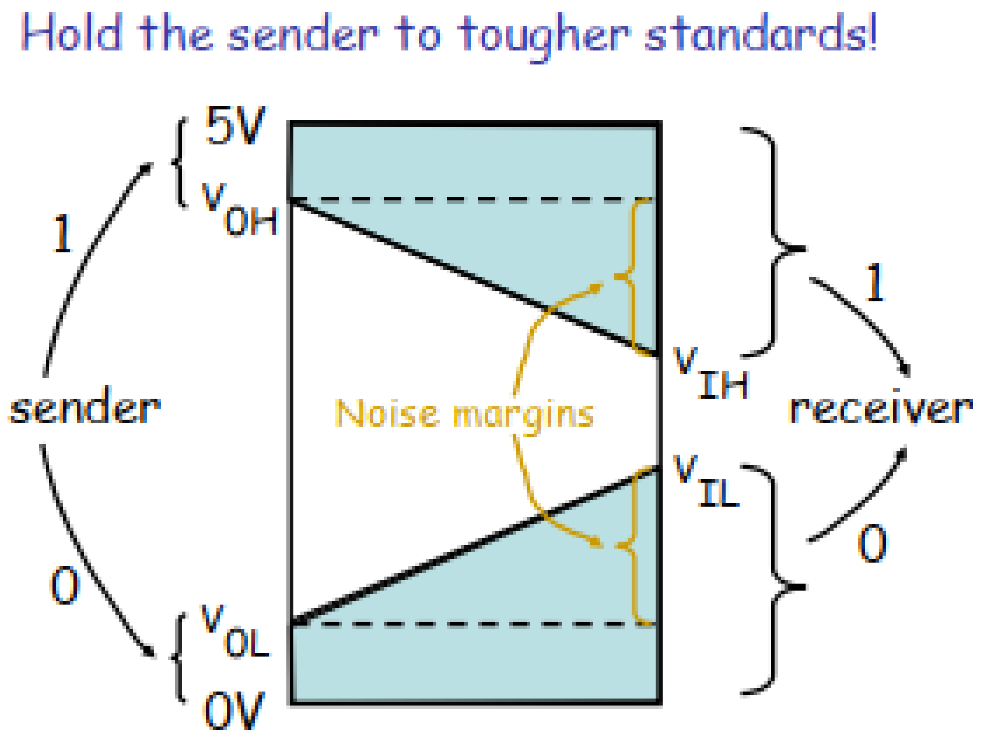


As illustrated in Figure the *noise margin for a logical 0* is given by

$$NM_0 = V_{IL} - V_{OL}$$

and the *noise margin for a logical 1* is given by

$$NM_1 = V_{OH} - V_{IH}.$$



"1" noise margin: V_{IH} - V_{OH}

"0" noise margin: V_{IL} - V_{OL}

Voltage Levels and the Static Discipline

To send a logical 0, the sender must produce an *output* voltage value that is less than V_{OL}. Correspondingly, the receiver must interpret *input* voltages below V_{IL} as a logical 0.

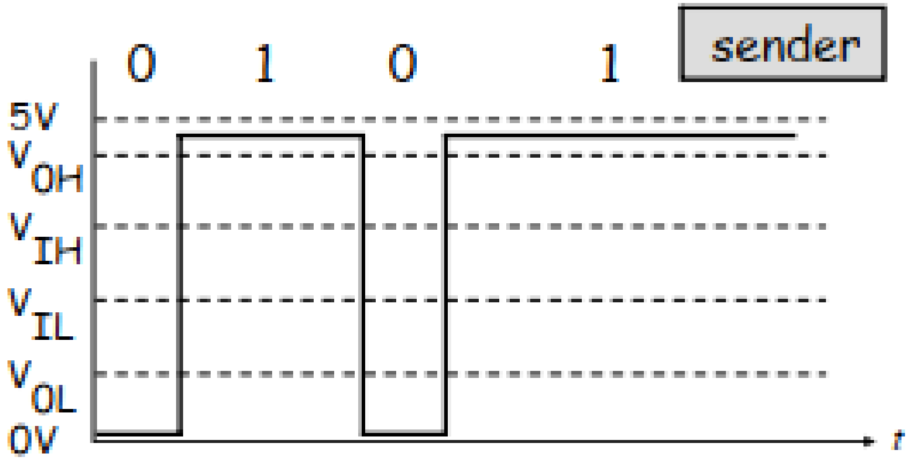
To allow for a reasonable noise margin, V_{IL} must be greater than V_{OL}.

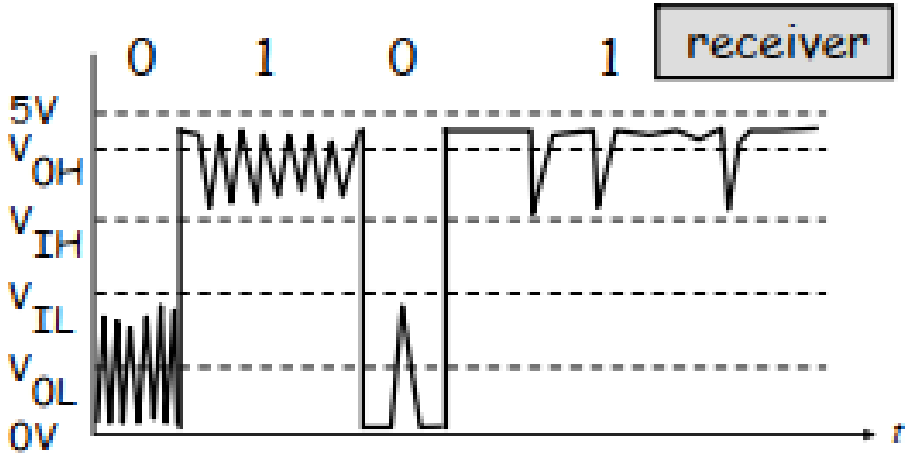
Similarly, to send a logical 1, the sender must produce an *output* voltage value that is greater than V_{OH}. Further, the receiver must interpret voltages above V_{IH} as a logical 1.

To allow for a reasonable noise margin, V_{OH} must greater than V_{IH}.[4] We can define both a noise margin for transmitting logical 1's and for transmitting logical 0's.

Noise Margin: The absolute value of the difference between the prescribed output voltage for a given logical value and the corresponding forbidden region voltage threshold for the receiver is called the *noise margin* for that logical value.

Standardization

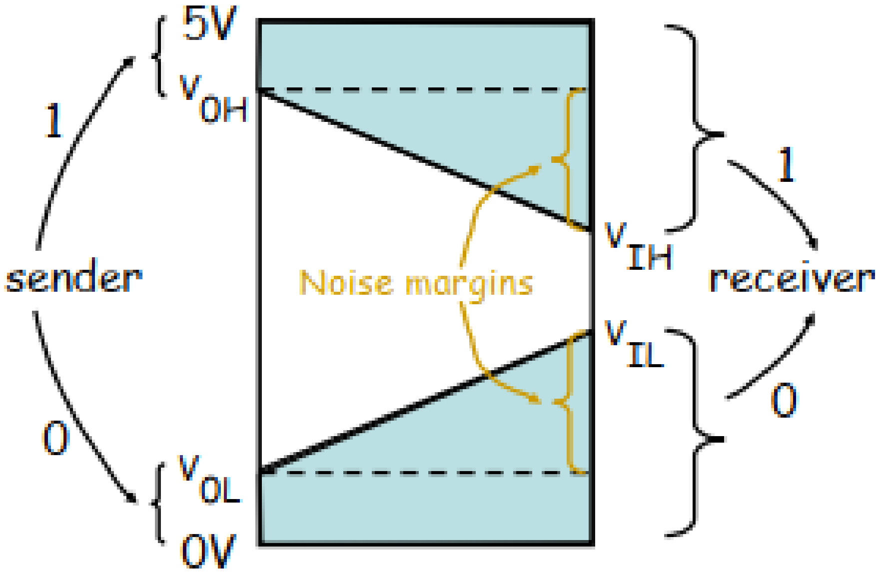




Digital systems follow static discipline: if inputs to the digital system meet valid input thresholds, then the system guarantees its outputs will meet valid output thresholds.

Noise Margin Formula

Hold the sender to tougher standards!



"1" noise margin: V_{IH} - V_{OH}

"0" noise margin: V_{IL} - V_{OL}

As illustrated in Figure the *noise margin for a logical 0* is given by

$$NM_0 = V_{IL} - V_{OL}$$

and the *noise margin for a logical 1* is given by

$$NM_1 = V_{OH} - V_{IH}.$$

QUICK QUIZ (POLL)

Noise Margin is:

a) Amount of noise the logic circuit can withstand

b) Difference between VOH and VIH

c) Difference between VIL and VOL

d) All of the Mentioned

QUICK QUIZ (POLL)

The Lower Noise Margin is given by:

a) VOL – VIL

b) VIL – VOL

c) VIL ~ VOL(Difference between VIL and VOL, depends on which one is greater)

d) All of the Mentioned

Exercise (POLL)

COMPARING NOISE MARGINS Which of the two static disciplines shown below offers better noise margins?

Static discipline A has the voltage thresholds given by:

$$V_{IL} = 1.5\ \mathrm{V},\ \ V_{IH} = 3.5\ \mathrm{V},\ \ V_{OL} = 1\ \mathrm{V},\ \text{ and }\ V_{OH} = 4\ \mathrm{V}.$$

Static discipline B has the voltage thresholds given by:

$$V_{IL} = 1.5\ \mathrm{V},\ \ V_{IH} = 3.5\ \mathrm{V},\ \ V_{OL} = 0.5\ \mathrm{V},\ \text{ and }\ V_{OH} = 4.5\ \mathrm{V}.$$

A. A is better
B. B is better
C. Both are same
D. None of these

Solution

For static discipline A:

$$NM_0 = 1.5 \text{ V} - 1 \text{ V} = 0.5 \text{ V}$$

and

$$NM_1 = 4 \text{ V} - 3.5 \text{ V} = 0.5 \text{ V}.$$

For static discipline B:

$$NM_0 = 1.5 \text{ V} - 0.5 \text{ V} = 1 \text{ V}$$

and

$$NM_1 = 4.5 \text{ V} - 3.5 \text{ V} = 1 \text{ V}.$$

Thus, the voltage thresholds of static discipline B offer a better noise margin.

QUICK QUIZ (POLL)

The noise immunity _____________ with noise margin.

a) Decreases
b) Increases
c) Constant
d) None of the Mentioned

Quick Quiz (Poll)

The Higher Noise Margin is given by:

a) VOH – VIH
b) VIH – VOH
c) VIH ~ VOH(Difference between VIH and VOH, depends on which one is greater)
d) All of the mentioned

Quick Quiz (Poll)

Input Voltage between VIL and VOL is considered as:

a) Logic Input 1
b) Logic Input 0
c) Uncertain
d) None of the mentioned

Advantages of Going Digital!

1. Better Noise Margin.
2. Programmability.
3. Reduction in hardware (compact).
4. Less cost.
5. High Reliability.
6. High Speed.
7. **Easy to design**: These signals do not require exact value at a particular time but it consists of range of particular values of voltage. Thus, it comprises of basically two values 0 and 1 i.e high or low.
8. **Information Storage is Simpler:** The storage of information in digital systems is easy. It can be stored by latching thus, the it can be stored for a long period of time.

UNIT-IV

Fundamentals of semiconductor devices and digital circuits

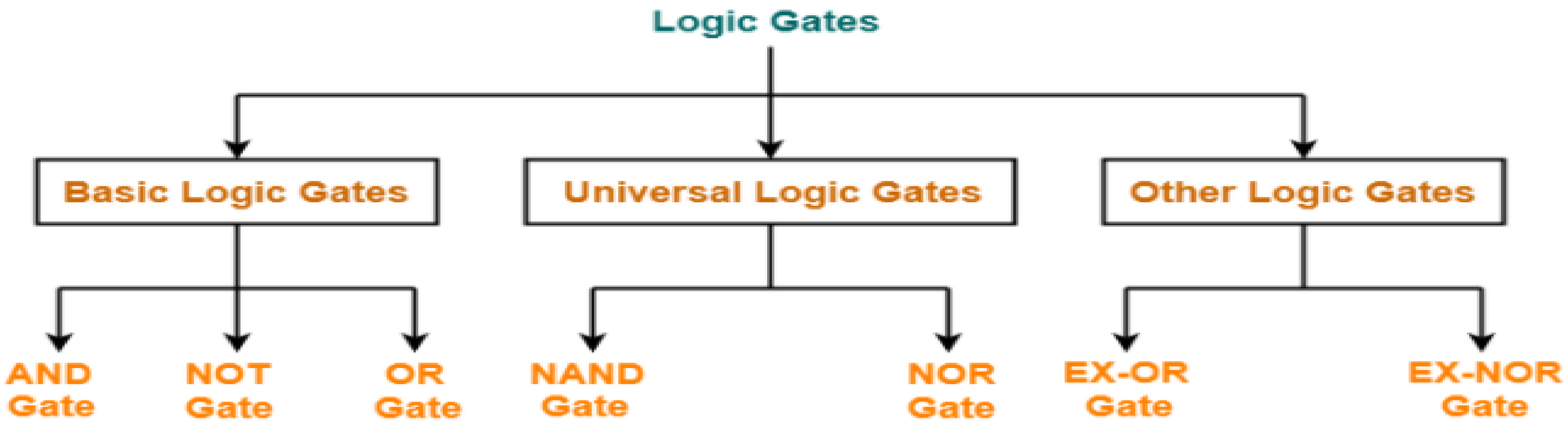
Lecture 24

Prepared By:

Pawandeep Kaur

Assistant Professor and Head-ECE

BOOLEAN LOGIC and COMBINATIONAL GATES



BOLEAN LOGIC and COMBINATIONAL GATES

- The binary representation has a natural correspondence to logic, and therefore digital circuits are commonly used to implement logic procedures.

OPERATOR	SYMBOL
AND	·
OR	+
NOT	~

Some logic operations and their symbols

OR function is represented using “+”
AND function is represented using the “·” symbol
NOT function using the bar symbol as in “*X* ” or the ~ symbol as in ~ *X*.
For example,
we represent the condition “*B* is FALSE” as *B* or ~ *B* or the complement of *B*.
For convenience, we will use 1, TRUE, and high interchangeably.
Similarly, we will use 0, FALSE,
and low interchangeably.

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

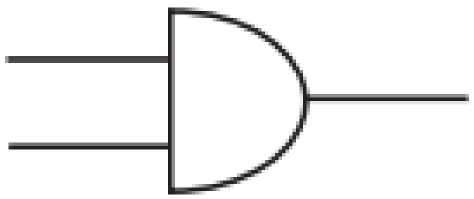
Truth table for
$Z = X \cdot Y.$

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

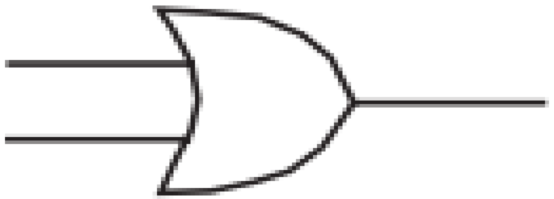
Truth table for
$Z = X + Y$

X	Z
0	1
1	0

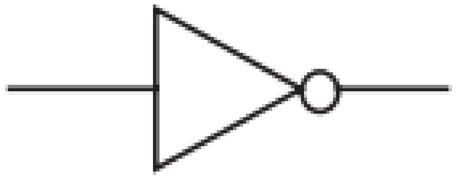
Truth table for $Z = \overline{X}$



AND

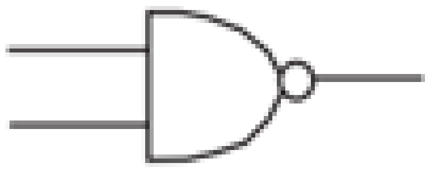


OR



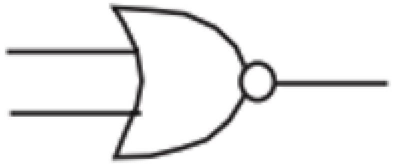
NOT

- The *NAND* function is equivalent to the *AND* operation followed by the *NOT* operation.



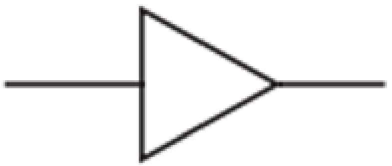
NAND

- The *NOR* operation is equivalent to the *OR* operation followed by the *NOT* operation



NOR

- The buffer gate or identity gate simply copies the input value to its output.



BUFFER

Truth table for several two-input functions.

INPUTS		AND	OR	NAND	NOR
B	C	$B \cdot C$	$B + C$	$\overline{B \cdot C}$	$\overline{B + C}$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	1	0
1	1	1	1	0	0

Representation of the Boolean functions:-

a) ***Truth table*** *:-* It is convenient to use a *truth table* representation of boolean functions.

- A truth table enumerates all possible input value combinations and the corresponding output values.

b) ***Combinational gate abstraction :-*** A combinational gate is an abstract representation of a circuit that satisfies two properties:

1. Its outputs are a function of its inputs alone.
2. It satisfies the static discipline.

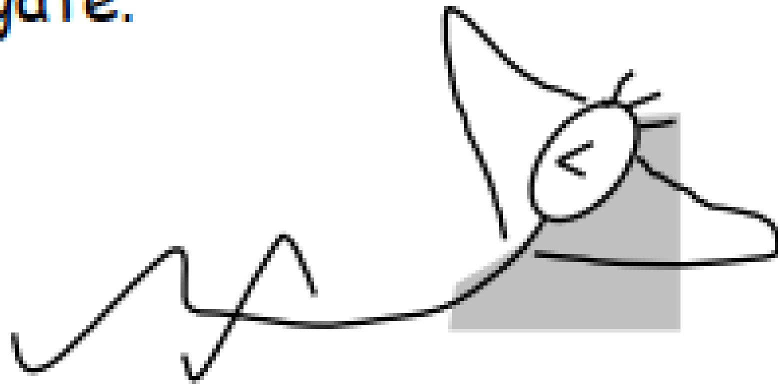
- Combinational gates follow the static discipline.
- Gates can have multiple inputs.

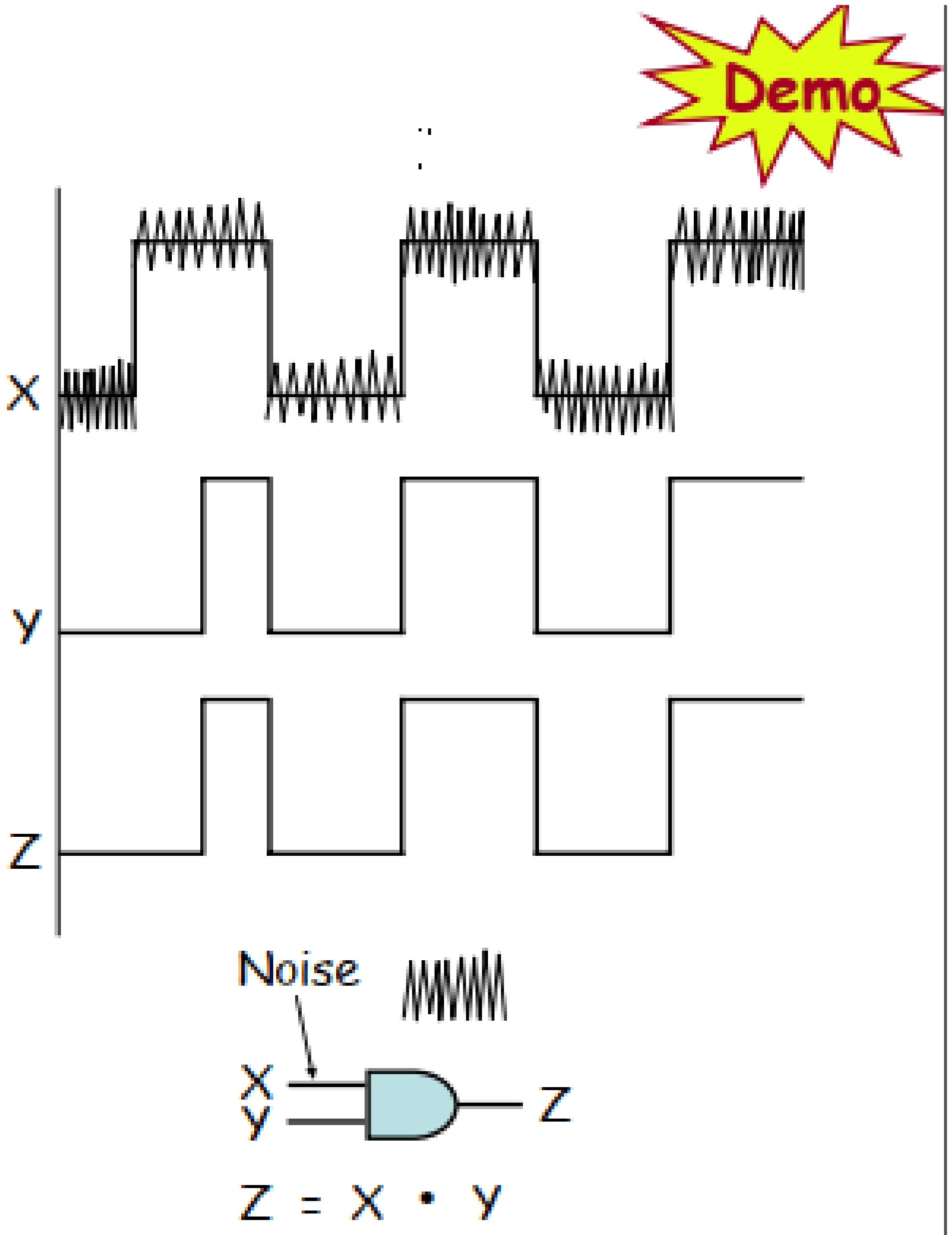
Explanation

Combinational gate abstraction

- Adheres to static discipline
- Outputs are a function of inputs alone.

Digital logic designers do not have to care about what is inside a gate.





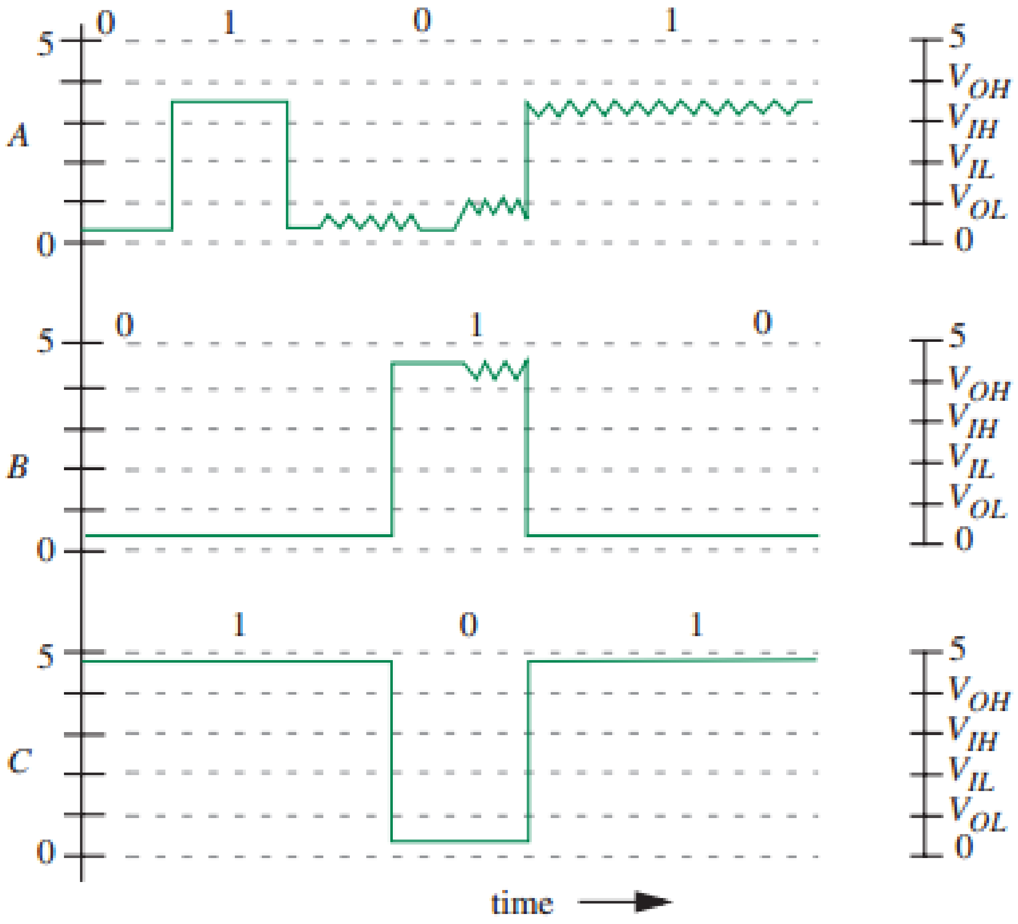


FIGURE 5.17 A noisy signal input to a digital circuit.

Practical demonstration of Noise Margin (Anant Aggarwal, MIT):
https://www.youtube.com/watch?v=sIgtZ4HUJ00

Boolean Algebra

- A Boolean function has:
  - At least one Boolean variable,
  - At least one Boolean operator, and
  - At least one input from the set {0,1}.
- It produces an output that is also a member of the set {0,1}.

Now you know why the binary numbering system is so handy in digital systems.

Quick Quiz (Poll)

- Boolean algebra can be used _____________

  a) For designing of the digital computers
  b) In building logic symbols
  c) Circuit theory
  d) Building algebraic functions

Rules of Boolean Algebra

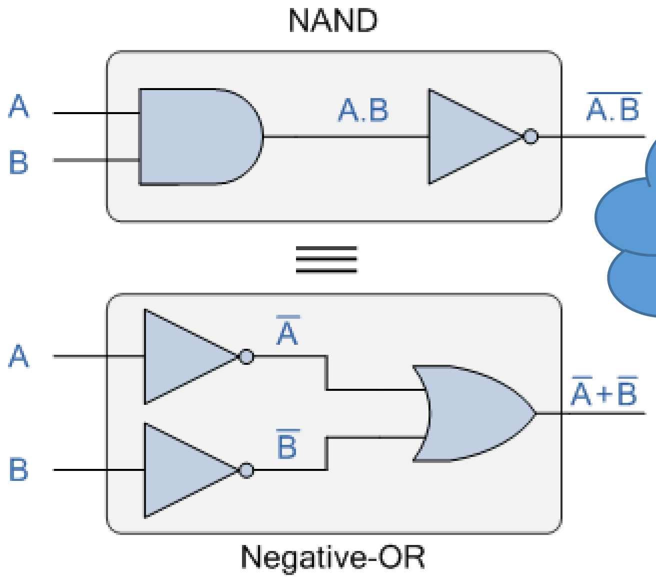
Also known as switching algebra

S.no	Rule Name	OR law	AND law
1	Null element	A+1=1	A.0=0
2	Identity law	A+0=A	A.1=A
3	Idempotent Law	A+A=A	A.A=A
4	Complement Law	A+A’=1 [controlling pin]	A.A’=0 [controlling pin]
5	Commutative law	A+B=B+A [order does not matter]	A.B=B.A [order does not matter]
6	Associative law	A+(B+C)=(A+B)+C	A.(B.C)=(A.B).C
7	Distributive law	A+(B.C)=(A+B).(A+C)	A.(B+C)=(A.B)+(A.C)
8	**Absorption law**	**A+AB=A**	**A.(A+B)=A**

Rules of Boolean Algebra

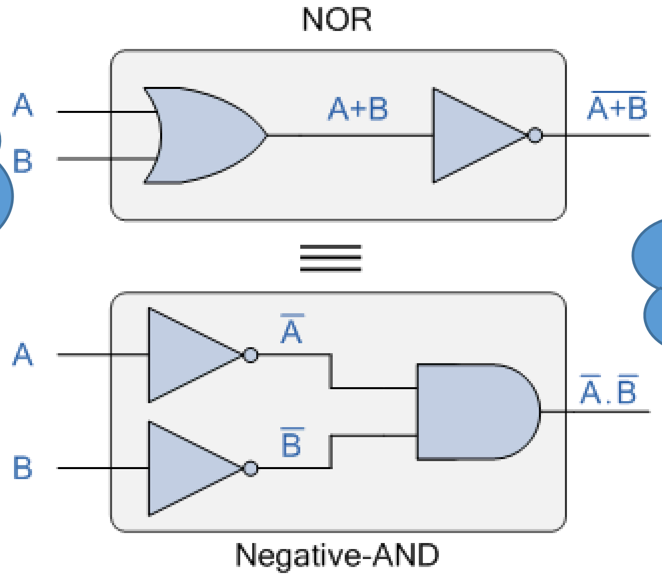
9	Double Inversion law	A’’=A	
10	Redundant Literal Rule (RLR)	A+A’B=A+B or A’+AB=A’+B or A’+AB’=A’+B’ or A +A’B’=A+B’	
11	Consensus Theorem	AB+A’C+BC=AB+A’C	

DeMorgan’s First Law Implementation using Logic Gates



NAND = Bubbled OR gate

DeMorgan’s Second Law Implementation using Logic Gates



NOR=bubbled AND

EXPLANATION

Exercise

Simplify the expression $Z = \overline{X}\,Y + X\,\overline{Y} + X\,Y.$

Quick Quiz (Poll)

Applying DeMorgan's theorem to the expression $\overline{ABC}$, we get ________.

A. $\overline{A} + \overline{B} + \overline{C}$

B. $\overline{A + B + C}$

C. $A + \overline{B} + C\overline{C}$

D. $A(B + C)$

Quick Quiz (Poll)

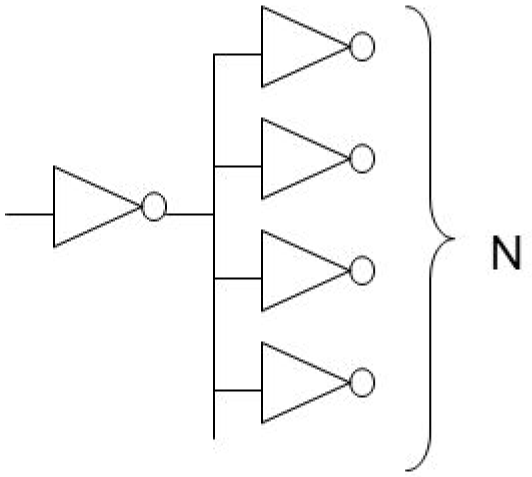
- $AC + ABC = AC$

A True

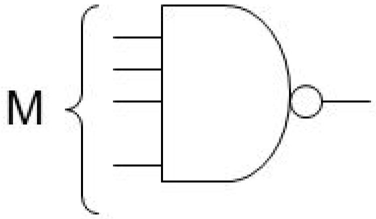
B False

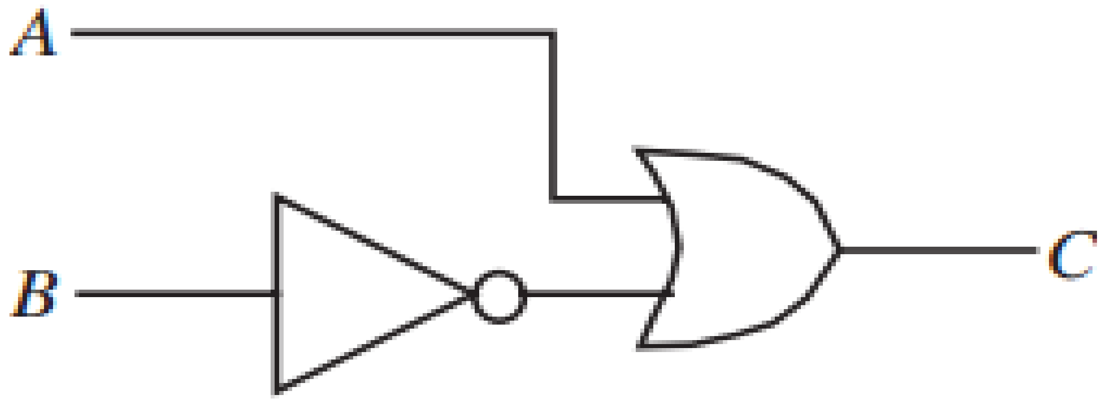
Fan-In and Fan-Out

- Fan-out – number of load gates connected to the output of the driving gate
  - gates with large fan-out are slower



- Fan-in – the number of inputs to the gate
  - gates with large fan-in are bigger and slower





The gate-level digital circuit for $C = A + \overline{B}$.

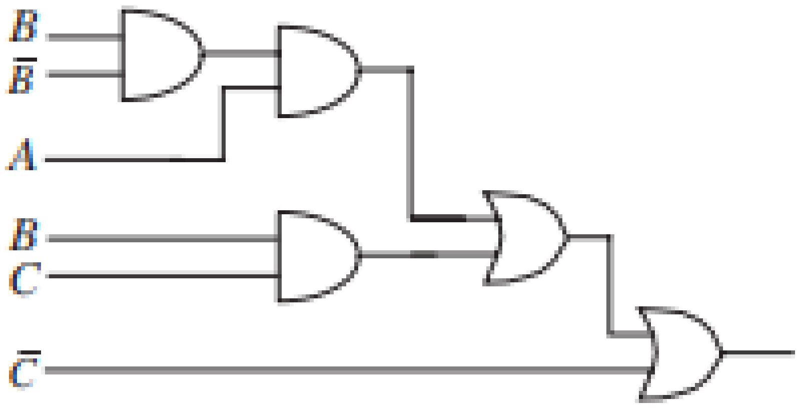
Question :- Design a circuit for the expression using the gates.

1) $Output = \overline{AB + C + D}$

2) $\overline{(A + B)CD}.$

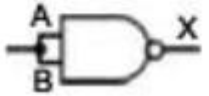
Exercise

Find the output?



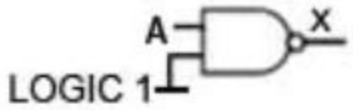
A. A + BC’
B. A + C’
C. B’ + C’
D. B + C’

Universal Gates (NAND and NOR)



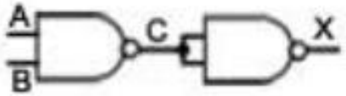
NOT

A	B	X
0	0	1
1	1	0



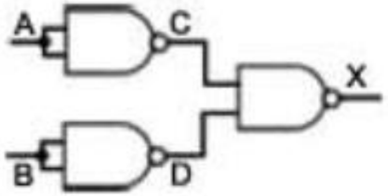
NOT

A	B	X
0	1	1
0	1	1
1	1	0
1	1	0



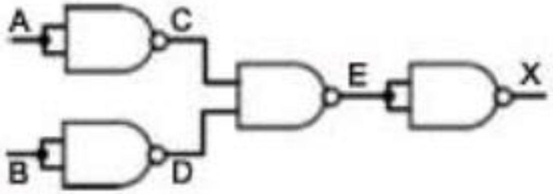
AND

A	B	C	X
0	0	1	0
0	1	1	0
1	0	1	0
1	1	0	1



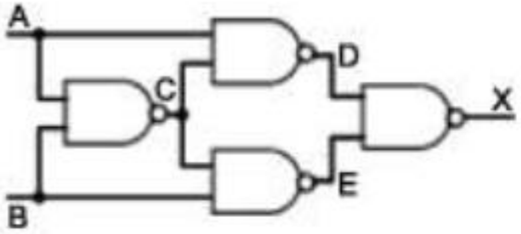
OR

A	B	C	D	X
0	0	1	1	0
0	1	1	0	1
1	0	0	1	1
1	1	0	0	1



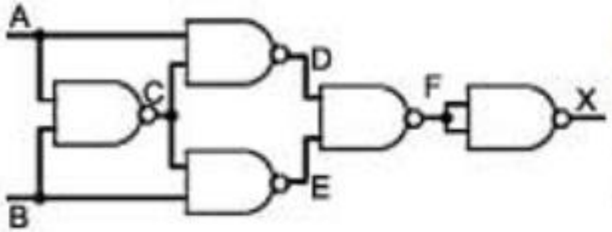
NOR

A	B	C	D	E	X
0	0	1	1	0	1
0	1	1	0	1	0
1	0	0	1	1	0
1	1	0	0	1	0



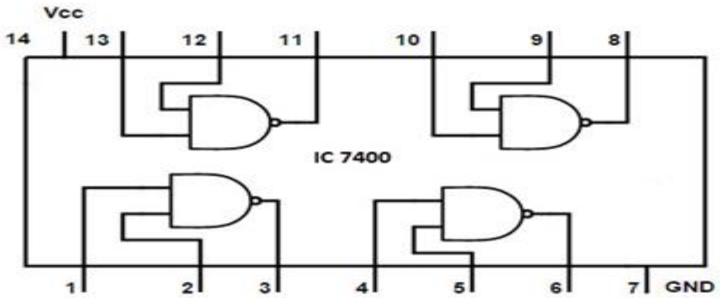
XOR

A	B	C	D	E	X
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	0



XNOR

A	B	C	D	E	F	X
0	0	1	1	1	0	1
0	1	1	1	0	1	0
1	0	1	0	1	1	0
1	1	0	1	1	0	1



They are called as "**Universal Gates**" because-

- They can realize all the binary operations.
- All the basic logic gates can be derived from them.

They have the following properties-

- Universal gates are not associative in nature.
- Universal gates are commutative in nature.

Explanation

Useful Links:

- Digital Abstraction Concept (Anant Aggarwal, MIT): https://www.youtube.com/watch?v=4TCnYYpZxEc&list=PL9F74AFA03AA06A11&index=4&t=2129s

- Practical demonstration of Noise Margin (Anant Aggarwal, MIT): https://www.youtube.com/watch?v=sIgtZ4HUJ00