Presentation on

"DIGITAL TECHNIQUES"

Author

Manoj A. Jethwa

(HOD, Department of Computer Technology)

Sanjay P. Pande

(HOD, Department of Information Technology)

Shri Datta Meghe Polytechnic
Wanadongri , Hingna Road
Nagpur-441 110

PREFACE

As educators, we all have the same common goal "to guide our students" so that they gain the maximum possible in a positive environment that promotes their success and inculcates in them desire to learn. One of the best tools available to us in this pursuit is PPT instruction that is systematic and self Learning. The goal of this PPT is to help teachers in the use of eLearning that it is both *effective* and *efficient method* for teaching our students. It has been developed for purely non-commercial academic purpose.

Our desire in preparing this PPT is to support the teachers, who have the very demanding task of Teaching-Plan to deliver instruction on a lecture/period basis. The PPT is therefore prepared lecture wise. Further at the end of each chapter Questions have also been included for practice.

We begin in Chapter 1 with basic Understanding of the Digital Systems and Logic Families. In Chapters 2 we learn in details the Boolean Laws and concept of Logic Gates. Chapter 3 focuses on Digital design using K-map. Chapter 4 concentrates on Understanding the concept one bit memory cell – Flip-flop ,counters and their Applications. In Chapter 5, we focus on understanding different Analog to Digital and Digital to Analog Conversion Techniques.

With deep regards and humility, we thank our Management for motivating and strong follow-ups to prepare PPTs . We dedicate this PPT to students and our shared profession.

M.A.Jethwa

Sanjay Pande

CONTENT: DIGITAL TECHNIQUES

CHAPTER NO.1

Introduction to Digital Techniques

CHAPTER 1:- INTRODUCTION TO DIGITAL TECHNIQUES

1. Digital signal , Digital systems.
2. Logic families- Characteristics, Classification.
3. Number System- Classification.

CHAPTER-1 SPECIFIC OBJECTIVE

The student will be able to:

1. Understand the Digital Systems and Logic Families.
2. Convert different number systems..

Analog vs. Digital Signals

- An analog signal assumes a continuous range of values.
- They can have infinite number of different values.
- The example of analog signals are temperature, pressure, distance, sound, brightness, voltage, current, power.
- A digital signal assumes discrete values.
- Digital signals are not continuous signal.

Generation of Digital Signals

- Digital signals consist of patterns of *bits* of information.
- These patterns can be generated in many ways, each producing a specific code.
- Modern digital computers store and process all kinds of information as binary patterns.
- All the pictures, text. sound and video stored in this computer are held and manipulated as patterns of binary values.

Advantages of Digital Circuits

- They have higher accuracy.
- They are less affected by ageing and variation in temperature.
- Less affected by noise.
- Digital systems have memory.
- Design is much easier.
- Highly reliable systems.

Comparison chart

Parameters	Analog	Digital
Signal:	Analog signal is a continuous signal which represents physical measurements.	Digital signals are discrete time signals generated by digital modulation.
Waves:	Denoted by sine waves	Denoted by square waves
Representation:	Uses continuous range of values to represent information	Uses discrete or discontinuous values to represent information
Example:	Human voice in air, analog electronic devices.	Computers, CDs, DVDs, and other digital electronic devices.
Data transmissions:	Subjected to deterioration by noise during transmission and write/read cycle.	Can be noise-immune without deterioration during transmission and write/read cycle.

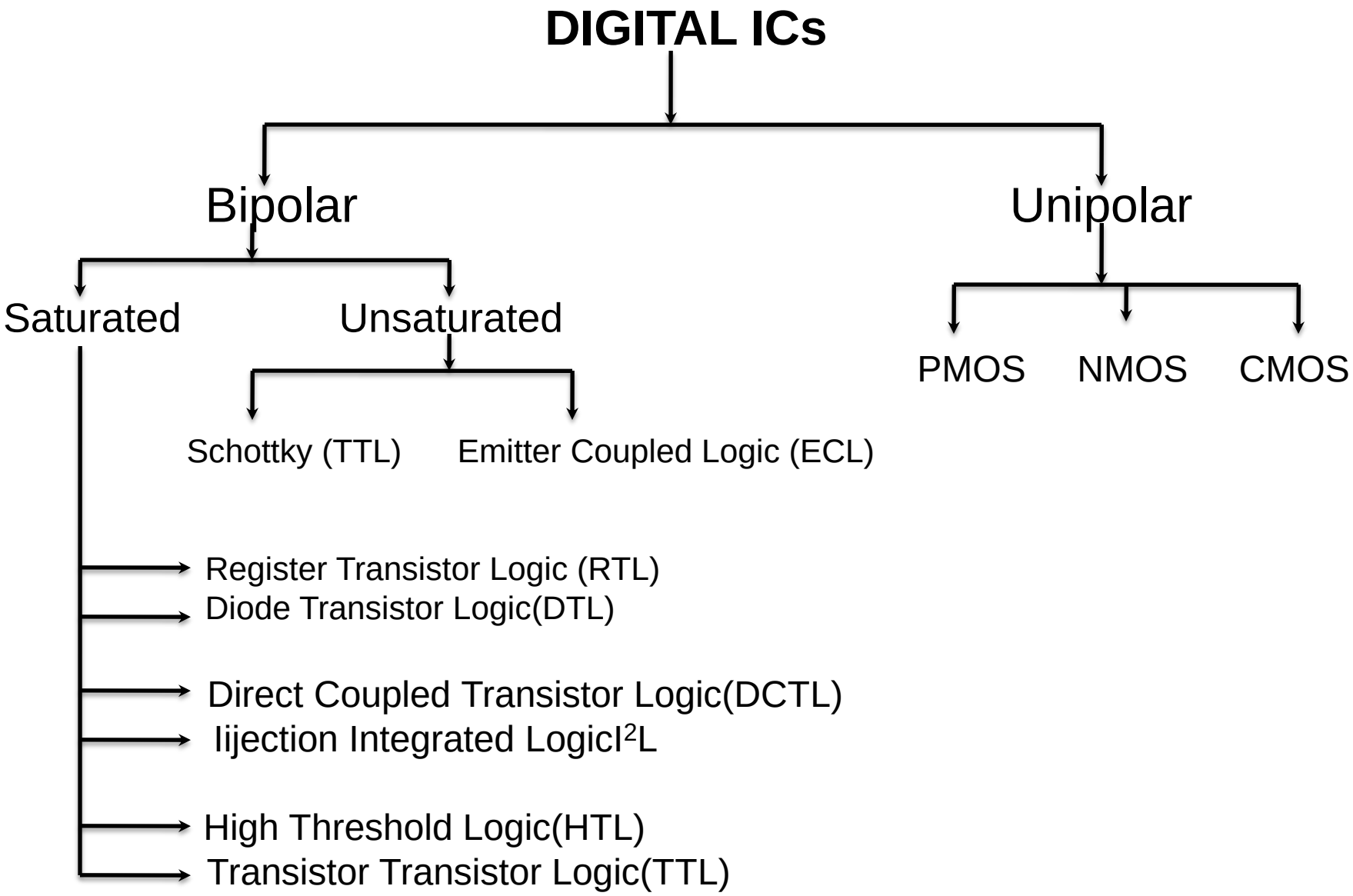
Comparison chart

Parameters	Analog	Digital
Technology:	Analog technology records waveforms as they are.	Samples analog waveforms into a limited set of numbers and records them.
Response to Noise:	More likely to get affected reducing accuracy	Less affected since noise response are analog in nature
Flexibility:	Analog hardware is not flexible.	Digital hardware is flexible in implementation.
Uses:	Can be used in analog devices only. Best suited for audio and video transmission.	Best suited for Computing and digital electronics.
Applications:	Thermometer	PCs, PDAs

Introduction to ICs

- An integrated circuit (IC) is a small semiconductor-based electronic device consisting of fabricated transistors, resistors and capacitors.
- Integrated circuits are the building blocks of most electronic devices and equipment.
- An integrated circuit is also known as a chip or microchip.

Classification of ICs



Generation of ICs

- **Small Scale Integration or (SSI)** –
  Contain up to 10 transistors or a few gates within a single package such as AND, OR, NOT gates.

- **Medium Scale Integration or (MSI)** –
  Between 10 and 100 transistors or tens of gates within a single package and perform digital operations such as adders, decoders, counters, flip-flops and multiplexers.

- **Large Scale Integration or (LSI)** –
  Between 100 and 1,000 transistors or hundreds of gates and perform specific digital operations such as I/O chips, memory, arithmetic and logic units.

Generation of ICs

- **Very-Large Scale Integration or (VLSI)** –
  Between 1,000 and 10,000 transistors or thousands of gates and perform computational operations such as processors, large memory arrays and programmable logic devices.

- **Super-Large Scale Integration or (SLSI)** –
  Between 10,000 and 100,000 transistors within a single package and perform computational operations such as microprocessor chips, micro-controllers, basic PICs and calculators.

- **Ultra-Large Scale Integration or (ULSI)** –
  More than 1 million transistors - the big boys that are used in computers CPUs, GPUs, video processors, micro-controllers, FPGAs and complex PICs.

Specifications :

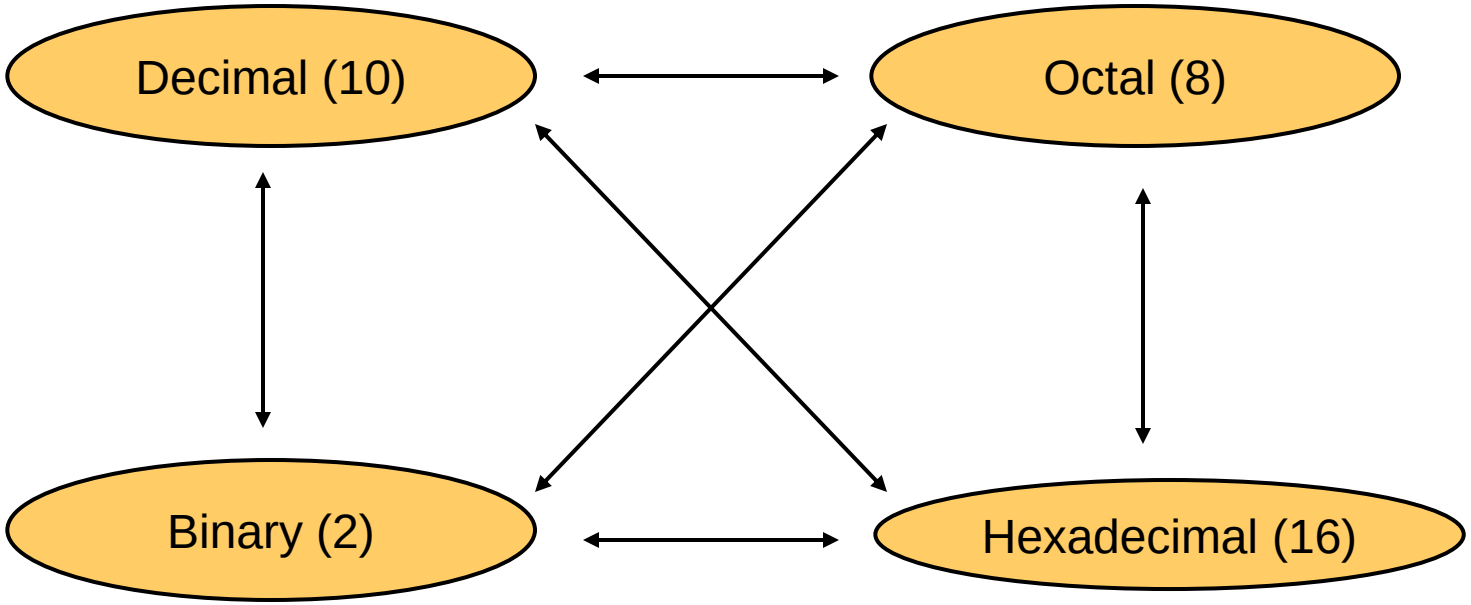
- Fan-out
- Noise immunity
- Noise margin
- Propagation delay
- Power dissipation
- Threshold voltage
- Operating temperature

Conversion Among Bases

- The number of values that a digit can assume is equal to the base of the system.
- This is called as radix of the system.
- The largest value of a digit is always one less than the base.
- Each digit position represents a different multiple of base.

Conversion Among Bases

- Types of Number Systems :



Numeral systems conversion table

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Complements

- Two numbers in a complement number system can be added/subtracted directly without the sign and magnitude checks.

- There are two types of Complements:
  - 1’s Complement
  - 2’s Complement

CHAPTER NO.2
LOGIC GATES

CHAPTER 2 :- LOGIC GATES

1. Basic gates
2. Boolean laws
3. Application of Boolean laws to simplify the Boolean expressions
4. Construction of logical circuits by simplifying the Boolean Expression.

CHAPTER-2 SPECIFIC OBJECTIVE

The student will be able to:

1. Understand Boolean Laws and concept of Logic Gates.

Types of GATES

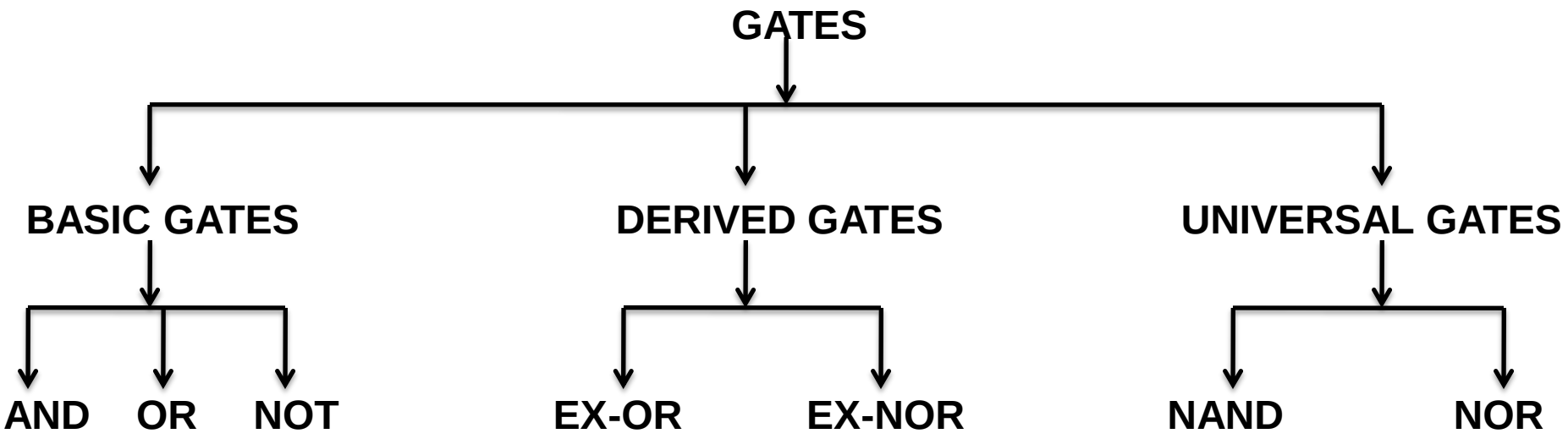
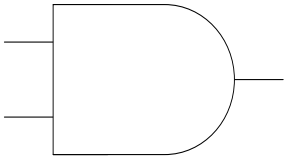


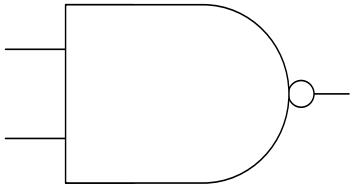
Figure: Types of Logic Gates

SYMBOLS

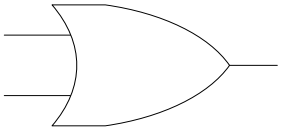
- AND GATE



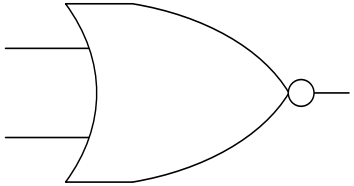
- NAND GATE



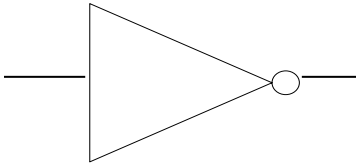
- OR GATE



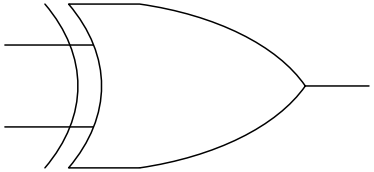
- NOR GATE



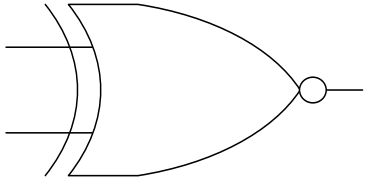
- NOT GATE



- EX-OR GATE



- EX-NOR GATE



TRUTH TABLE

- A truth table shows how a logic circuit's output responds to various combinations of the inputs, using logic 1 for true and logic 0 for false.
- Formula for truth table:

  $2^n = m$

Where, n → number of inputs

m → combination of inputs

UNIVERSAL GATES

- NAND and NOR as Universal Logic Gates

- Any logic circuit can be built using only NAND gates, or only NOR gates. They are the only logic gate needed.

UNIVERSAL GATES

- NAND and NOR as Universal Logic Gates

- Any logic circuit can be built using only NAND gates, or only NOR gates. They are the only logic gate needed.

Laws of Boolean.

- Following are the some laws of Boolean:
  - Commutative Law: A.B = B.A; A+B = B+A
  - Associative Law: (A.B).C = A. (B.C); (A+B)+C= A+(B+C)
  - Distributive Law: A. (B+C) = AB + AC
  - AND Law: A. 0 = 0; A.1 = A; A.A = A; A. A = 0
  - OR Laws: A + 0 = A; A + 1 = 1; A + A = A; A + A = 1;
  - Inversion Laws: $\overline{\overline{A}} = A$
  - Duality theorem

- **De-Morgans Theorem**:

First theorem states that complement of sum is equal to the product of product of their individual complements.

$$\overline{A + B} = \overline{A} * \overline{B}$$

- **De-Morgans theorem:**

Second theorem states that complement of products is equal to the sum of their complements.

$$\overline{A * B} = \overline{A} + \overline{B}$$

Redundancy laws

The following laws will be proved with the basic laws.

Absorption

$x + x \cdot y = x$

Proof:

$x + x \cdot y$

$= x \cdot 1 + x \cdot y$

$= x \cdot (1 + y)$

$= x \cdot 1$

$= x$

$x \cdot (x + y) = x$

Proof:

$x \cdot (x + y)$

$= (x + 0) \cdot (x + y)$

$= x + (0 \cdot y)$

$= x + 0$

$= x$

Redundancy laws

The following laws will be proved with the basic laws.

$$\boldsymbol{x \cdot y + x \cdot z + y \cdot z = x \cdot y + x \cdot z}$$

Proof:

$$
\begin{aligned}
& x \cdot y + x \cdot z + y \cdot z \\
&= x \cdot y + x \cdot z + 1 \cdot y \cdot z \\
&= x \cdot y + x \cdot z + (x + x) \cdot y \cdot z \\
&= x \cdot y + x \cdot z + x \cdot y \cdot z + x \cdot y \cdot z \\
&= x \cdot y + x \cdot y \cdot z + x \cdot z + x \cdot y \cdot z \\
&= x \cdot y \cdot 1 + x \cdot y \cdot z + x \cdot 1 \cdot z + x \cdot y \cdot z \\
&= x \cdot y \cdot (1 + z) + x \cdot z \cdot (1 + y) \\
&= x \cdot y \cdot 1 + x \cdot z \cdot 1 \\
&= x \cdot y + x \cdot z
\end{aligned}
$$

$(x + y) \cdot (x + z) \cdot (y + z) = (x + y) \cdot (x + z)$

Proof:

$(x + y) \cdot (x + z) \cdot (y + z)$

$= (x + y) \cdot (x + z) \cdot (0 + y + z)$

$= (x + y) \cdot (x + z) \cdot (x \cdot x + y + z)$

$= (x + y) \cdot (x + z) \cdot (x + y + z) \cdot (x + y + z)$

$= (x + y) \cdot (x + y + z) \cdot (x + z) \cdot (x + y + z)$

$= (x + y + 0) \cdot (x + y + z) \cdot (x + 0 + z) \cdot (x + y + z)$

$= (x + y + 0 \cdot z) \cdot (x + z + 0 \cdot y)$

$= (x + y + 0) \cdot (x + z + 0)$

$= (x + y) \cdot (x + z)$

CHAPTER NO. 3
COMBINATIONAL LOGIC CUIRCUTS

CHAPTER 3 : Combinational Logic Circuits

1. SOP & POS
2. K-map representation of logical functions.
3. Designing of Adder and subtractor.
4. Multiplexer
5. Demultiplexer
6. Priority Encoders and Decoder

CHAPTER-3 SPECIFIC OBJECTIVE

The student will be able to:

1. Reduce the Boolean expression using K-map.
2. Understand and design Multiplexer, Demultiplexer, Encoder, and Decoder.

Combinational Circuits

- Combinational Logic Circuits are made up from basic logic NAND, NOR or NOT gates that are "combined" or connected together to produce more complicated switching circuits. These logic gates are the building blocks of combinational logic circuits.

Sum of Product

- Here is how to get the sum-of-products solution, given a truth table, you have to locate each output 1 in the truth table and write down the fundamental product. For instance, the first output 1 appears for an input of A=0,B=1 and C=1. The corresponding fundamental product is ABC. To get the sum of products equation, all you have to is *OR the fundamental products.*

$$Y = \bar{A}\,B\,C + A\,\bar{B}\,C + A\,B\,C$$

Product-of-sum Method

- Everything is the opposite of SOP form.
- Several terms are formed by OR ing variables together, then AND ing the terms together.
- Example:

  $X = (A+B)(B+C)(A+C)$

- This is a “product” of 3 “sums.”

- Given a truth table, you identify the fundamental sums needed for a logic design.

- Then by AND ing these sums, you get the product-of-sum equation corresponding to the truth table.
-
- With SOP the fundamental sum produces an output 0 for the corresponding input condition.

- How to convert truth table to Karnaugh Map.

The vertical column has A followed by A, and horizontal row has B followed by B.

Now, look for output 1s in table. for A=1 and B=0.

The fundamental product for this input condition is AB.

Enter this fundamental product on the Karnaugh map as shown in figure b.

This 1 represent the product AB because the 1 is in row A and column B.

ADDER

- An **adder** or **summer** is a digital circuit that performs addition of numbers.
- In many computers and other kinds of processors, adders are used not only in the arithmetic logic unit(s), but also in other parts of the processor, where they are used to calculate addresses, table indices.

- A **half adder** adds two one-bit binary numbers *A* and *B*.
- It has two outputs, *Sum* (S)and *Carry(C).*
- The final sum is 2*C* + *S*.
- The simplest half-adder design, pictured on the right, incorporates an XOR gate for *S* and an AND gate for *C*.
- Half adders cannot be used compositely, given their incapacity for a carry-in bit.

- EQUATION FOR SUM AND CARRY

SUM = X'Y + XY'

CARRY = XY

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

- A **full adder** adds binary numbers and accounts for values carried in as well as out.
- A one-bit full adder adds three one-bit numbers, often written as *A*, *B*, and C_{in}; *A* and *B* are the operands, and C_{in} is a bit carried in.
- The circuit produces a two-bit output sum typically represented by the signals C_{out} and *S.*

Truth Table

A	B	Cin	SUM	CARRY
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

SUM = X'Y'Z + XY'Z' + X'YZ'

SUM = X $\oplus$ Y $\oplus$ Z

CARRY = XY + XZ + YZ

Half subtractor

- The half-subtractor is a combinational circuit which is used to perform subtraction of two bits.

- It has two inputs, X (minuend) and Y (subtrahend) and two outputs D (difference) and B (borrow)

Truth Table

X	Y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

$D = X \oplus Y$

$B = X'.Y$

Full subtractor

- The full-subtractor is a combinational circuit which is used to perform subtraction of three bits.
- It has three inputs, X (minuend) and Y (subtrahend) and Z (subtrahend) and two outputs D (difference) and B (borrow).

- **Types of Multiplexer**
  - **2:1 Multiplexer**
  - **4:1 Multiplexer**
  - **8:1 Multiplexer**
  - **16:1 Multiplexer**

DECODER

- **Binary n-to-2^n Decoders**

A binary decoder has n inputs and 2^n outputs. Only one output is active at any one time, corresponding to the input value. Figure below shows a representation of Binary n-to-2^n decoder

Encoders

An encoder is a combinational circuit that performs the inverse operation of a decoder. If a device output code has fewer bits than the input code has, the device is usually called an encoder. e.g. 2^n-to-n, priority encoders.

Priority Encoder/ Priority generator

The truth table of a 4-input priority encoder is as shown below. The input D3 has the highest priority, D2 has next highest priority, D0 has the lowest priority. This means output Y2 and Y1 are 0 only when none of the inputs D1, D2, D3 are high and only D0 is high.

A 4 to 3 encoder consists of four inputs and three outputs, truth table and symbols of which is shown below.

CHAPTER NO. 4
Sequential Logic Circuit

CHAPTER 4 : Sequential Logic Circuit

1. Introduction to Sequential Logic Circuit
2. One-bit memory cell, clock signal.
3. Flip Flops.
4. Applications of flip flops
5. Memories

CHAPTER-4 SPECIFIC OBJECTIVE

The student will be able to:

1. Understanding the concept one bit memory cell Flip-flop and their Applications.

2. To understand the concept of memories.

TYPES OF DIGITAL SYSTEM

- The digital system are classified into two categories

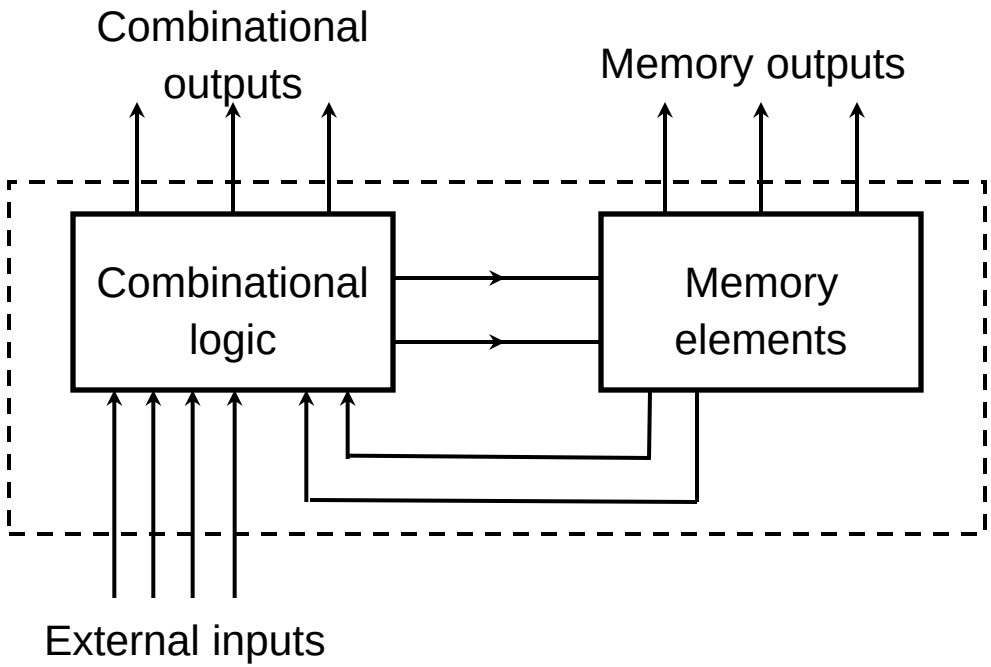
✓ Combinational logic circuit

✓ Sequential logic circuit

Comparison

Sr. No.	Combinational Circuit	Sequential Circuit
1.	It contains no memory elements	It contains memory elements
2.	The present value of it's outputs are determined solely by the present values of it's inputs	The present value of it's outputs are determined by the present value of it's inputs and it's past state
3.	It's behavior is described by the set of output functions	It's behavior is described by the set of next-state(memory) functions and the set of output functions

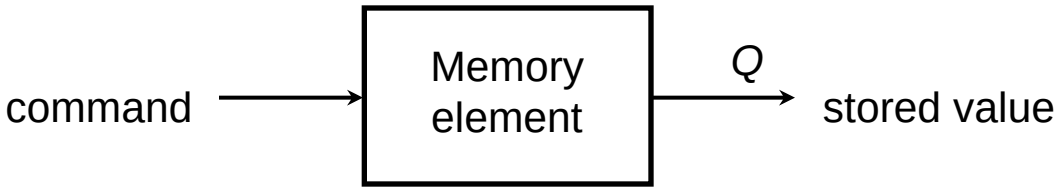
A sequential circuit consists of a feedback path, and employs some memory elements.



Sequential circuit = Combinational logic + Memory Elements

- There are two types of sequential circuits:
  - Synchronous: Outputs change only at specific time.
  - Asynchronous: Outputs change at any time.

- **Memory element**
  A device which can remember value indefinitely, or change value on command from its inputs.



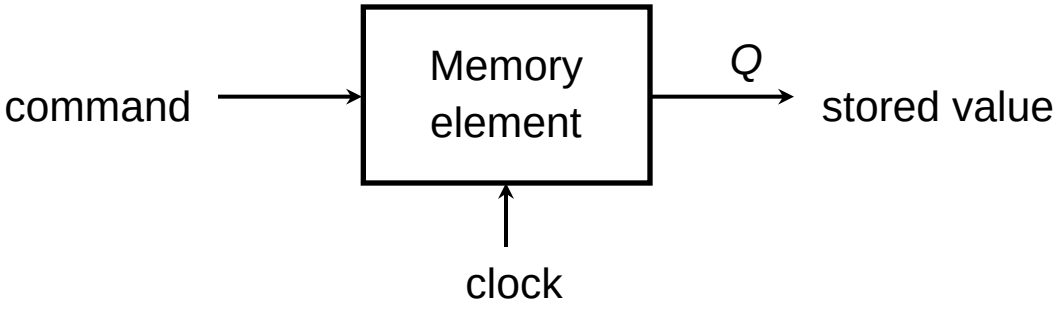
- Characteristic table:

Command (at time *t*)	*Q(t)*	*Q(t+1)*
Set	X	1
Reset	X	0
Memorise / No Change	0	0
	1	1

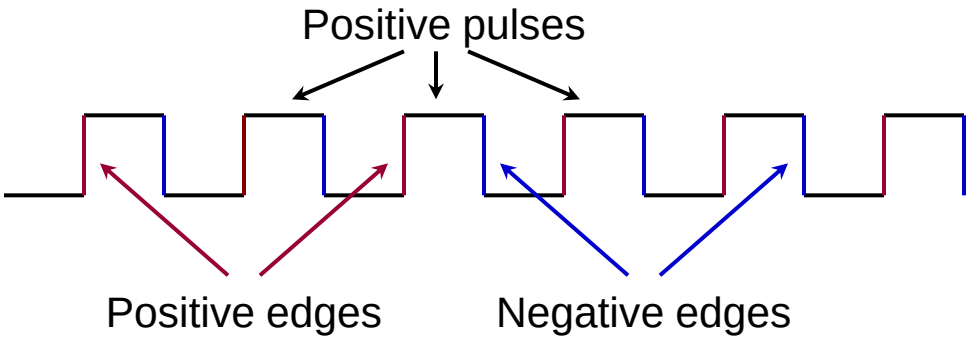
Q(t): current state
Q(t+1) or Q^+: next state

Memory Elements

- Memory element with clock. Flip-flops are memory elements that change state on clock signals.



- Clock is usually a square wave.



Memory Elements

- Two types of triggering:
  - ❖pulse-triggered
  - ❖edge-triggered
- **Pulse-triggered**
  - ❖latches
  - ❖ON = 1, OFF = 0
- **Edge-triggered**
  - ❖flip-flops
  - ❖positive edge-triggered
  - ❖negative edge-triggered

Triggering

- Clocks transitions are used in different ways
  - Level triggering
    - When clock is in a given state
  - Edge triggering
    - Raising edge triggered
      - When the clock is in transition from 0 → 1
    - Falling edge triggered
      - When the clock is in transition from 1 → 0

S-R Latch

- SR latch based on **NOR gates**.
- The S input sets the Q output to 1 while R reset it to 0.
- When R=S=0 then the output keeps the previous value.
- When R=S=1 then Q=Q'=0, and the latch may go to an unpredictable next state.

S-R Latch

- S-R latch based on **NAND gates**.
- The S’ input sets the Q output to 1 while R’ reset it to 0.
- When R’=S’=1 then the output keeps the previous value.
- When R’=S’=1 then Q=Q’=1, and the latch may go to an unpredictable next state.

D Latch

- This latch eliminates the problem that occurs in the S'R' latch when R=S=0.
- C is an enable input:
  - When C=1 then the output follows the input D and the latch is said to be open.
  - When C=0 then the output retains its last value and the latch is said to be closed.

J-K Flip-flop

- J-K flip-flop: Q and Q' are fed back to the pulse-steering NAND gates.
- No invalid state.
- Include a *toggle* state.
  - *J*=HIGH (and *K*=LOW) a SET state
  - *K*=HIGH (and *J*=LOW) a RESET state
  - both inputs LOW a no change
  - both inputs HIGH a toggle

J-K Flip-flop

- J-K flip-flop.
- Characteristic table.

J	K	CLK	Q(t+1)	Comments
0	0	↑	Q(t)	No change
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	Q(t)'	Toggle

Q	J	K	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

$$Q(t+1) = J.Q' + K'.Q$$

Edge Triggered J-K Flip-Flop

- The operation of inputs J and K in the J-K flip-flop is similar to the operation of inputs S and R in the S-R flip-flop. The difference arises when J and K are asserted simultaneously. In this situation the output of the J-K flip-flop inverts its current state.

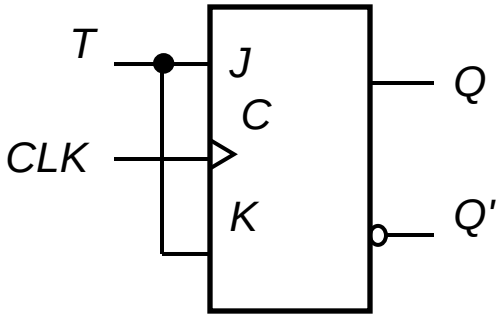
MASTER SLAVE J-K FLIP-FLOP

- The **Master-Slave Flip-Flop** is basically two gated SR flip-flops connected together in a series configuration with the slave having an inverted clock pulse.

- The outputs from Q and Q from the "Slave" flip-flop are fed back to the inputs of the "Master" with the outputs of the "Master" flip-flop being connected to the two inputs of the "Slave" flip-flop.

T Flip-flop

- T flip-flop: single-input version of the J-K flip flop, formed by tying both inputs together.



- Characteristic table.

T	CLK	Q(t+1)	Comments
0	↑	Q(t)	No change
1	↑	Q(t)'	Toggle

Q	T	Q(t+1)
0	0	0
0	1	1
1	0	1
1	1	0

$$Q(t+1) = T.Q' + T'.Q$$

Asynchronous Inputs

- S-R, D and J-K inputs are synchronous inputs, as data on these inputs are transferred to the flip-flop's output only on the triggered edge of the clock pulse.
- **Asynchronous inputs** affect the state of the flip-flop independent of the clock; example: *preset* (*PRE*) and *clear* (*CLR*).
- When *PRE*=HIGH, *Q* is immediately set to HIGH.
- When *CLR*=HIGH, *Q* is immediately cleared to LOW.
- Flip-flop in normal operation mode when both *PRE* and *CLR* are LOW.

Applications of flip flops

- Flip flops are used in Counters/timers.
- In various types of Registers.
- It is used as a delay element.
- It is used as a memory element.

Introduction to Registers

- A **register** is a digital device used for storage. —
- Register is a group of Flip flop to store a data (0 or 1). —
- To store ‘n’ no. of data , ‘n’ bit register is required which consist of ‘n’ no. of flip flop.
- **Types of registers:**
  - Buffer register
  - Shift registers

Types of Registers

Buffer register

- An ‘n’ bit registers has group of ‘n’ flip flop and capable to store any binary information,— which contains ‘n’ numbers of bits.
- This type of register is also called storage registers.
- These are used for temporary storage of data

Types of Registers

Shift Registers

•—The binary data in register can be moved within the register from one flip-flop to the other or out side it with application of clock pulse.

•This register allow such data transferred are called as Shift register.

Mode of operation of a Shift Registers:

- Serial Input Serial Output(SISO)
- Serial Input Parallel Output(SIPO)
- Parallel Input Serial Output(PISO)
- parallel Input parallel Output(PIPO)

Serial IN-Serial OUT

- The flip-flops used to construct registers are usually edge-triggered JK, SR or D types

To shift a 0 into the flip-flop,
J=0 and K=1

To shift a 1 into the flip-flop,
J=1 and K=0

Counters

- Counter is used to count the number of clock pulses.
- A counter is a device which stores the number of times a particular event or process has occurred, often in relationship to a clock signal.

Types of counters

- Asynchronous (ripple) counter – changing state bits are used as clocks to
  subsequent state flip-flops

- Synchronous counter – all state bits change under control of a single clock

Types of counters

- Decade counter – counts through ten states per stage
- Up/down counter – counts both up and down, under command of a control input
- Ring counter – formed by a shift register with feedback connection in a ring
- Johnson counter – a twisted ring counter
- Cascaded counter

Asynchronous Counters

- Ripple Counter
  - When the output of a flip-flop is used as the clock input for the next flip-flop, we call the counter a ripple counter, or asynchronous counter.

Mod -6 counter

- Let us try to design a modulo-6 counter, the counting states (memory values) of which are shown in state transition diagram. We need three memory element or flip-flops for this as with *n* flip-flop we can get at most 2*n* number of different counting states.

Johnson counter

STATES					COUNT
A	B	C	D	E	
0	0	0	0	0	0
1	0	0	0	0	1
1	1	0	0	0	2
1	1	1	0	0	3
1	1	1	1	0	4
1	1	1	1	1	5
0	1	1	1	1	6
0	0	1	1	1	7
0	0	0	1	1	8
0	0	0	0	1	9

Memories

- Main memory consists of a number of storage locations, each of which is identified by a **unique address.**
- The ability of the CPU to identify each location is known as its addressability.
- Each location stores a **word** i.e. the number of bits that can be processed by the CPU in a single operation. **Word length** may be typically 16, 24, 32 or as many as 64 bits

Classification of Memories

Random Access Memory (RAM)

- Random access memory is called as read write memory.
- Described as being volatile
- Its purpose is to temporarily hold programs and data for processing.
- In modern computers it also holds the operating system

Types of RAM:

- ✓ Static RAM
- ✓ Dynamic RAM

Classification of Memories

Read only memory (ROM)

- ROM holds programs and data **permanently** even when computer is switched off
- Data can be read by the CPU in any order so ROM is also **direct access**
- The contents of ROM are fixed at the time of manufacture

Types of ROM

- ✓ PROM
- ✓ EPROM
- ✓ EEPROM

Classification of Memories

➢Programmable Read Only Memory (PROM)

- A PROM can be programmed only once after its fabrication. After programming the contents will become permanently fixed in the ROM.

➢Erasable Programmable Read Only Memory (EPROM)

- Can be programmed, erased and reprogrammed
- The EPROM chip has a small window on top allowing it to be erased by shining ultra-violet light on it

Classification of Memories

➢ **Electrically Erasable Programmable Read Only Memory (EEPROM)**

- Reprogrammed electrically **without** using ultraviolet light
- Must be removed from the computer and placed in a special machine to do this

➢ **Flash ROM**

- Similar to EEPROM. However, can be reprogrammed while still in the computer
- Easier to upgrade programs stored in Flash ROM

Classification of Memories

Sequential Memories:

- The example of sequential memory are magnetic tape audio/video cassette.
- In the sequential memories the memory locations are organized in a sequence.

Types of sequential memories:

- ✓ Shift registers
- ✓Charge coupled devices(CCD)

Types of Sequential memory

Shift registers

- The shift registers can be of two types i.e. static or dynamic.
- In the static memory , the memory contents do not change with time as long as power is ON.
- In the dynamic memory, The memory contents can change with time. So it is necessary to refresh such memories at regular intervals.

Charge Coupled Devices (CCD)

- The CCD are manufactured using MOS technology.
- The advantages of CCD are high density and low cost.

Content Addressable Memory(CAM)

- CAM can be used as a search engine.
- We want to find matching contents in a database or Table.

Type of CAMs

Binary CAM (BCAM) only stores 0s and 1s

Ternary CAM (TCAM) stores 0s, 1s and don’t cares.

CHAPTER NO. 5
A-D AND D-A CONVERTERS

CHAPTER 5 : A-D AND D-A CONVERTERS

1. DAC - Weighted resistor and R-2R Ladder - Circuit diagram, working, Advantages and Disadvantages- DAC specifications

2. ADC - Ramp, Dual slope and Successive approximation Circuit diagram, working, Advantages and Disadvantages - ADC Specifications.

CHAPTER-5 SPECIFIC OBJECTIVE

The student will be able to:

1. Understand different Analog to Digital and Digital to Analog Conversion Techniques

- When data is in binary form, the 0's and 1's may be of several forms such as the TTL form where the logic zero may be a value up to 0.8 volts and the 1 may be a voltage from 2 to 5 volts. The data can be converted to clean digital form using gates which are designed to be on or off depending on the value of the incoming signal.

- Data in clean binary digital form can be converted to an analog form by using a summing amplifier. For example, a simple 4-bit D/A converter can be made with a four-input summing amplifier. More practical is the R-2R Network DAC.

DAC Specifications

- Resolution
- Speed
- Linearity
- Settling Time
- Reference Voltages
- Errors

DAC Errors

- Gain Error
- Offset Error
- Full Scale Error
- Non-monotonic Output Error
- Diff- Non-linearity Error (A.K.A. Linearity Error)
- Settling Time and Overshoot Error
- Resolution Error

- This is a sample of the large number of analog-to-digital conversion methods. The basic principle of operation is to use the comparator principle to determine whether or not to turn on a particular bit of the binary number output. It is typical for an ADC to use a digital-to-analog converter (DAC) to determine one of the inputs to the comparator.

References Books:

- Modern Digital Electronics - R. P. Jain
- Digital Techniques - A. P. Godse , D.A. Godse
- Digital Principles - Malvino Leach
- Digital Fundamentals - Thomas Floyd