Introduction

Unit: 1

COMPUTER SYSTEM ORGANIZATION (AMCA 0104)

MCA-I SEM

Mr. Lakshman Singh

Syllabus

Course Contents / Syllabus		
UNIT-I	**Introduction**	**8 hours**
Introduction: Digital Computers and Number System, Logic Gates, Boolean Algebra, Map Simplification upto five variables, Combinational Circuits, Sequential Circuits, Look ahead carry adders, Data types, Complements, Fixed point representation, Fixed Point Addition & Subtraction, floating point Representation, Booth's Multiplication, IEEE754 Floating point standards.		
UNIT-II	**Register Transfer &Microoperations**	**8 hours**
Register Transfer Language, Register Transfer, Bus and Memory Transfers, Common Bus System, Two Bus Organization, Three Bus Organization, Arithmetic Microoperations, Logic Microoperations, Shift Microoperations, Arithmetic &Logic unit design.		
UNIT-III	**Central Processing Unit**	**8 hours**
Microprogrammed Control Unit, Hardwired Control Unit, General register Organization, Stack Organization, Instruction types, formats, instruction cycles and sub cycles (Fetch, decode, execute etc.), execution of a complete instruction, Addressing Modes, Reduced Instruction set computer, Complex Instruction set Computer		

Syllabus

UNIT-IV	Memory Management	8 hours

Memory Hierarchy, Main Memory (RAM and ROM chips), Auxiliary Memory, and Associative memory, Cache Memory, Memory Mapping: Associative mapping, Direct mapping, Set associative mapping. 2D and 2.5D memory organization

UNIT-V	Input/output	8 hours

I/O interface, I/O ports, Interrupts, Modes of data Transfer: Programmed I/O, Interrupt Initiated I/O, and Direct memory access (DMA), I/O channels and processors, Serial Communication, Standard communication interfaces.
Case Study : Multicore processing, Multithreading architecture

Contents

Introduction

- Digital computers and Number system
- Logic gates
- Boolean Algebra
- Map simplification
- Combinational circuits
- Sequential circuits
- Look ahead carry adders
- Data types and complements

Contents Cont..

- Fixed point representation
- Fixed point addition and Subtraction
- Floating Point representation
- Booth’s multiplication algorithm
- IEEE 754 floating point standards

Course Objective

- Discuss the basic concepts and structure of computers.
- Understand concepts of register transfer logic and arithmetic operations.
- Explain different types of addressing modes and memory organization.
- To discuss different types of number systems and their significance and usage

Prerequisite

- Before learning the concepts of **Basic Electronics** and **Computer Organization**, you should have a basic knowledge of functional units of a **computer** system.
- The basic knowledge of mathematics.
- The knowledge of English language upto a satisfactory level.

Topic

- Digital computer and Number systems

Topic objective

- To discuss about digital computer, different number systems, bases and the conversion of one number system to another.

- Representation of number systems in unsigned forms.

Introduction (CO1)

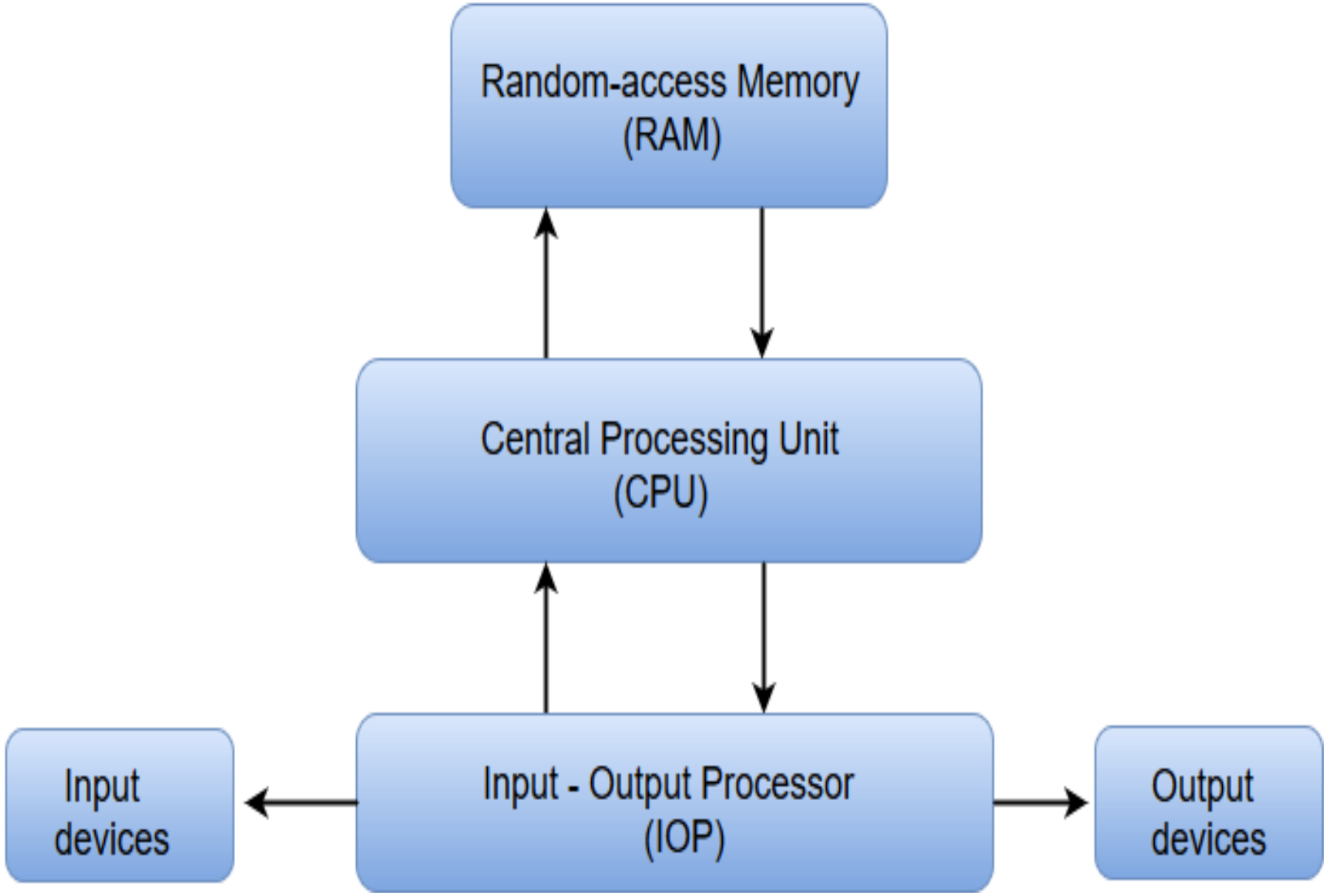
Digital Computer (Definition):

- An electronic computer in which the input is discrete rather than continuous, consisting of combinations of numbers, letters, and other characters written in an appropriate programming language and represented internally in binary notation.

- Capable of solving problems by processing information in discrete form.
- The first electronic digital computer was developed in the late 1940s and was used primarily for numerical computations. By convention, the digital computers use the binary number system, which has two digits: 0 and 1. A binary digit is called a bit.

- A computer system is subdivided into two functional entities: Hardware and Software.
- The hardware consists of all the electronic components and electromechanical devices that comprise the physical entity of the device.
- The software of the computer consists of the instructions and data that the computer manipulates to perform various data-processing tasks.

Block diagram of a digital computer:



- The Central Processing Unit (CPU) contains an arithmetic and logic unit for manipulating data, a number of registers for storing data, and a control circuit for fetching and executing instructions.
- The memory unit of a digital computer contains storage for instructions and data.
- The Random Access Memory (RAM) for real-time processing of the data.
- The Input-Output devices for generating inputs from the user and displaying the final results to the user.
- The Input-Output devices connected to the computer include the keyboard, mouse, terminals, magnetic disk drives, and other communication devices.

Introduction (CO1)

Number System (Definition):

- A number system of base, or radix, r is a system that uses distinct symbols for r digits.
- Numbers are represented by a string of digit symbols.
- For example- The decimal number system in everyday use employs the radix 10 system. The 10 symbols are 0,1,2,3,4,5,6,7,8, and 9.
- The string of digits 724.5 is interpreted to represent the quantity

$$7 \times 10^2 + 2 \times 10^1 + 4 \times 10^0 + 5 \times 10^{-1}$$

Unsigned Numbers (CO1)

- Unsigned numbers don’t have any sign, these can contain only magnitude of the number.
- So, representation of unsigned binary numbers are all positive numbers only.
- For example, representation of positive decimal numbers are positive by default. We always assume that there is a positive sign symbol in front of every number.

Unsigned Numbers (CO1)

- **Example-1:** Represent decimal number 92 in unsigned binary number.
- Simply convert it into Binary number, it contains only magnitude of the given number.
  $= (92)_{10} = (1x2^6+0x2^5+1x2^4+1x2^3+1x2^2+0x2^1+0x2^0)_{10} = (1011100)_2$
- It's 7 bit binary magnitude of the decimal number 92.

Converting between bases (CO1)

Other Base System to Decimal System

Steps

Step 1 − Determine the column (positional) value of each digit (this depends on the position of the digit and the base of the number system).

Step 2 − Multiply the obtained column values (in Step 1) by the digits in the corresponding columns.

Step 3 − Sum the products calculated in Step 2. The total is the equivalent value in decimal.

Converting between bases (CO1)

Example

Binary Number :11101_2

Calculating Decimal Equivalent :

Step Binary Number Decimal Number: 1

$11101_2=((1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0))_{10}$

Step 2

$11101_2 =(16 + 8 + 4 + 0 + 1)_{10}$

Step 3

$11101_2=29_{10}$

Binary Number 11101_2 = Decimal Number 29_{10}

Converting between bases Cont.. (CO1)

Other Base System to Non-Decimal System

Step 1 − Convert the original number to a decimal number (base 10).

Step 2 − Convert the decimal number so obtained to the new base number.

Example: Octal Number : 25_8

Calculating Binary Equivalent

Step 1 Convert to Decimal

Octal Number 25_8 = Decimal Number 21_{10}

Step	Octal Number	Decimal Number
Step 1	25_8	$((2 \times 8^1) + (5 \times 8^0))_{10}$
Step 2	25_8	$(16 + 5)_{10}$
Step 3	25_8	21_{10}

Daily Quiz

- Write counting upto 100 in Octal and Hexadecimal?
- Write counting upto 20 in base = 3 and base =5?
- Convert the following numbers with the indicated bases to decimal: $(12121)_3$; $(4310)_5$; $(50)_7$; and $(198)_{12}$.
- Convert the following decimal numbers to binary: 1231; 673; and 1998.

Recap

In the last lecture we have discussed about

- Number systems, base of number systems and the conversion of one number system to another.
- Representation of number systems in unsigned forms.

Topic

- Signed integer representation
- Fixed and floating point representation

Topic objective

- To discuss about representation of negative numbers In signed magnitude, 1’s complement and 2’s complement representation schemes.
- How to represent the numbers in fixed and floating point forms.
- What are different character codes used to represent characters.

Signed Integer Representation (CO1)

- To represent negative integers, we need a notation for negative values.
- It is customary to represent the sign with a bit placed in the leftmost position of the number since binary digits.
- The convention is to make the sign bit 0 for positive and 1 for negative.
- Example: there are 3 different ways to represent -9 with eight bits

Signed-magnitude representation:	10001001
Signed-1's-complement representation:	11110110
Signed-2's-complement representation:	11110111

Fixed and Floating Point Representations (CO1)

- The floating-point representation of a number has two parts.
- The first part represents a signed, fixed-point number called the mantissa.
- The second part designates the position of the decimal (or binary) point and is called the exponent.
- The fixed-point mantissa may be a fraction or an integer. For example, the decimal number + 6132.789 is represented in floating-point with a fraction and an exponent as follows:
  - Fraction Exponent

  + 0 .6132789 + 04

Character Codes (CO1)

1. Gray Code:-The advantage of the Gray code over straight binary numbers is that the Gray code changes by only one bit as it sequences from one number to the next. (Refer Table- 3.5)

2. Other Decimal Codes:-

- BCD, EBCDIC
- Excess-3, Excess-127
- 8421, 2421
- Error Detection codes (Parity bit)

Daily Quiz

1. Represent decimal number 8620 in
(a) BCD;
(b) excess-3 code;
(c) 2421 code;
(d) as a binary number

2. What is the radix of the numbers if the solution to the quadratic equation $x^2 - 10x + 31 = 0$ is $x = 5$ and $x = 8$

Recap

In the last lecture we have discussed about

- How to represent negative numbers In signed magnitude, 1's complement and 2's complement representation schemes.
- How to represent the numbers in fixed and floating point forms.
- What are different character codes used to represent characters.

Topic

- Logic gates

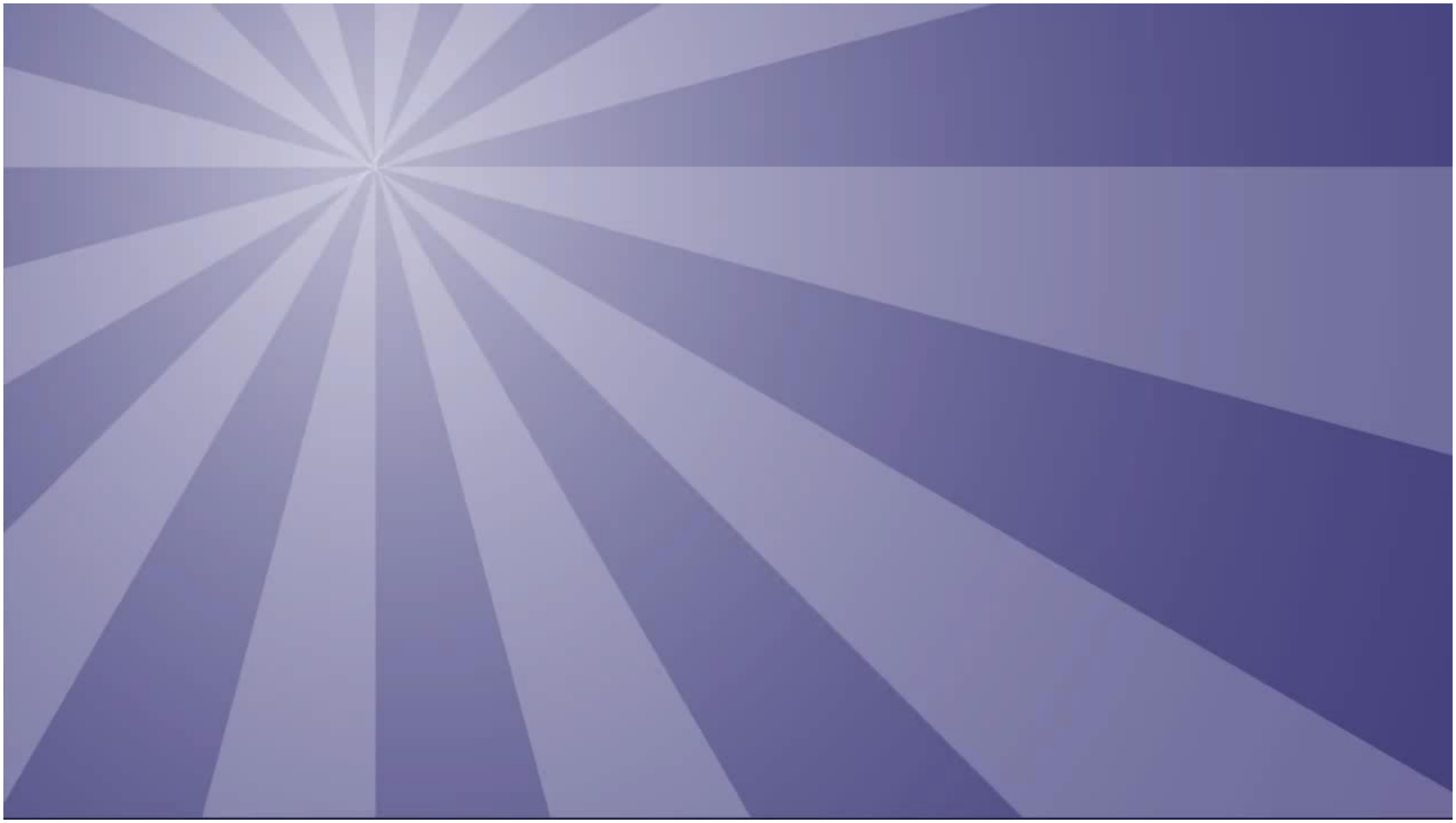
Topic objective

- To discuss about the function of logic gates using different representation schemes.

Logic Gates

- The gates are blocks of hardware that produce a binary output.
- There are three elementary logic gates and a range of other simple gates.
- Each gate has its own logic symbol which allows complex functions to be represented by a logic diagram.
- The function of each gate can be represented by a truth table or using Boolean notation.

Digital Logic (CO1)



Logic Gates (CO1)

The AND gate

(a) Circuit symbol

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

(b) Truth table

$C = A \cdot B$

(c) Boolean expression

Logic Gates Continued (CO1)

The OR gate

(a) Circuit symbol

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

(b) Truth table

$C = A + B$

(c) Boolean expression

Logic Gates Continued (CO1)

The NAND gate

(a) Circuit symbol

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

(b) Truth table

$$C = \overline{A \cdot B}$$

(c) Boolean expression

Logic Gates Continued (CO1)

The NOR gate

(a) Circuit symbol

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

(b) Truth table

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

(c) Boolean expression

Logic Gates Continued (CO1)

The Exclusive OR (XOR) gate

(a) Circuit symbol

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

(b) Truth table

$$C = A \oplus B$$

(c) Boolean expression

Logic Gates Continued (CO1)

The Exclusive NOR (X-NOR) gate

(a) Circuit symbol

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

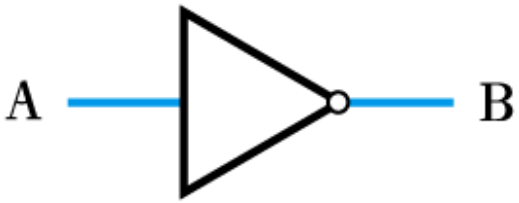
(b) Truth table

$$C = \overline{A \oplus B}$$

(c) Boolean expression

Logic Gates Continued (CO1)

- **The NOT gate (or inverter)**



(a) Circuit symbol

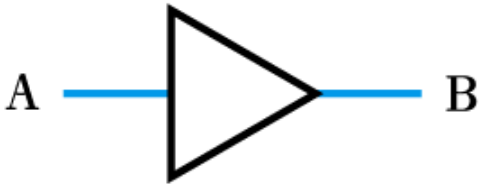
A	B
0	1
1	0

(b) Truth table

$B = \overline{A}$

(c) Boolean expression

- **A logic buffer gate**



(a) Circuit symbol

A	B
0	0
1	1

(b) Truth table

$B = A$

(c) Boolean expression

Daily Quiz

1. Which of these are called universal gate, And why ?

2. Differentiate between ex-or and ex-nor gate.

Weekly Assignment

1. Simplify the Boolean function F together with the don't-care conditions d in

   sum-of-products form.

   $F(w, x, y, z) = \sum(0, 1, 2, 3, 7, 8, 10)$

   $d(w, x, y, z) = \sum(5, 6, 11, 15)$

2. Which of the gates are called universal gate, And why ?

Recap

In the last lecture we have discussed about

- Function of logic gates and their representation

Topic

- Boolean Algebra

Topic objective

In this lecture we will discuss about-

- Boolean algebra, boolean identities and show how we can minimize a boolean expression by using the identities of boolean algebra.

Boolean algebra (CO1)

- Boolean algebra is an algebra that deals with binary variables and logic operations.
- The variables are designated by letters such as A, B, x, and y.
- The three basic logic operations are AND, OR, and complement.
- A Boolean function can be expressed algebraically with binary variables, the logic operation symbols, parentheses, and equal sign. For a given value of the variables, the Boolean function can be either 1 or 0.

Boolean algebra continued.. (CO1)

Basic identities of Boolean algebra

(1) x+0=x
(2) x.0=0
(3) X+1=1
(4) x.1=x
(5) X+X=X
(6) X.X=X
(7) X+X'=1
(8) X.X'=0
(9) X+Y =Y+X
(10) X.Y=Y.X
(11) X+(Y+Z) = (X+Y) +Z
(12) X(YZ) = (XY)Z
(13) X(Y+Z) =XY+XZ
(14) X+YZ= (X+Y)(X+Z)
(15) (X+Y)' = X'Y'
(16) (XY)'=X'+Y'
(17) X''=X

Boolean algebra continued.. (CO1)

Example: The expression can be simplified using Boolean algebra.

F = ABC+A BC'+ A 'C = AB(C +C ')+ A 'C = AB +A 'C

Note that(C +C)' = 1 by identity 7 and AB ·1 = AB by identity 4.

we can make logic diagram for both (original and minimized) expressions and show how the hardware requirement is minimized by minimizing the boolean expressions using boolean identities.

Daily Quiz

1. Simplify the following expressions using Boolean algebra.

a. A + AB b. AB + AB‘ c. A' BC + AC d. A'B +ABC' + ABC

2. Given the Boolean function

F = xy’z + x’y’z + xyz

a. List the truth table of the function.

b. Draw the logic diagram using the original Boolean expression.

c. Simplify the algebraic expression using Boolean algebra.

Recap

In the last lecture we have discussed about

Boolean algebra, boolean identities and show how we can minimize a boolean expression by using the identities of boolean algebra.

Topic

- Combinational and sequential circuits.
- Minimization of boolean expressions using K-map method.

Topic objective

In this lecture we will discuss about

- Combinational and sequential circuits.
- Minimization of boolean expressions using K-map method.

Combinational and sequential circuits (CO1)

Combinational Circuits

- A combinational circuit is a connected arrangement of logic gates with a set of inputs and outputs.
- At any given time, the binary values of the outputs are a function of the binary combination of the inputs.

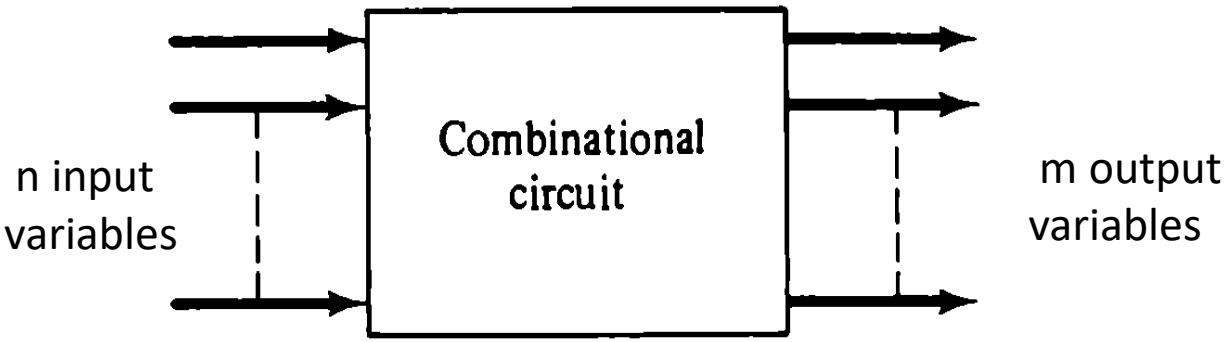


Figure 1.1 Block diagram of a combinational circuit

Combinational and sequential circuits continued.. (CO1)

Sequential Circuits

- A sequential circuit is an interconnection of flip-flops and gates.

- The gates by themselves constitute a combinational circuit, but when included with the flip-flops, the overall circuit is classified as a sequential circuit.

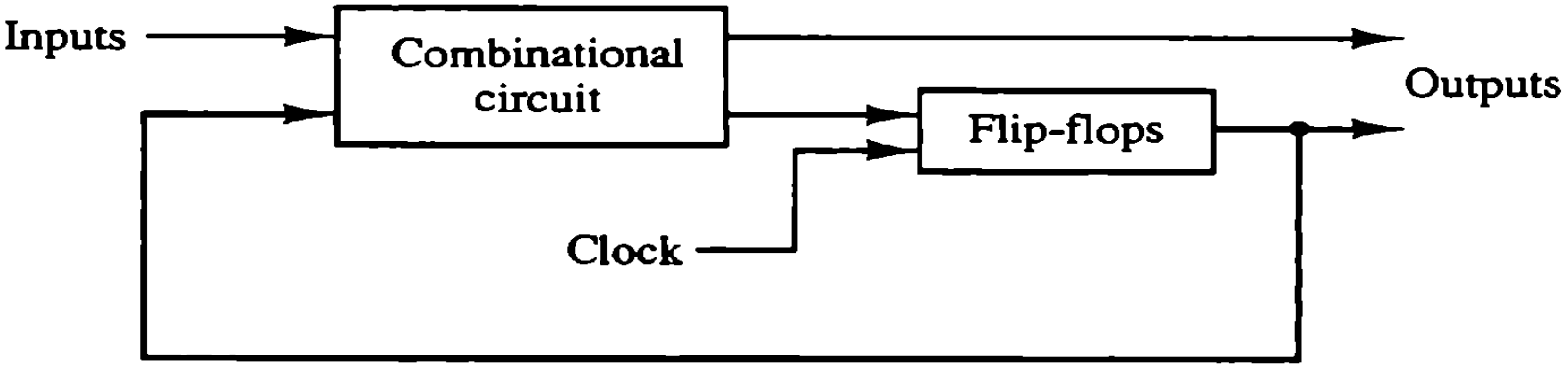


Figure 1.2 Block diagram of a clocked synchronous sequential circuit

Gate Minimization (CO1)

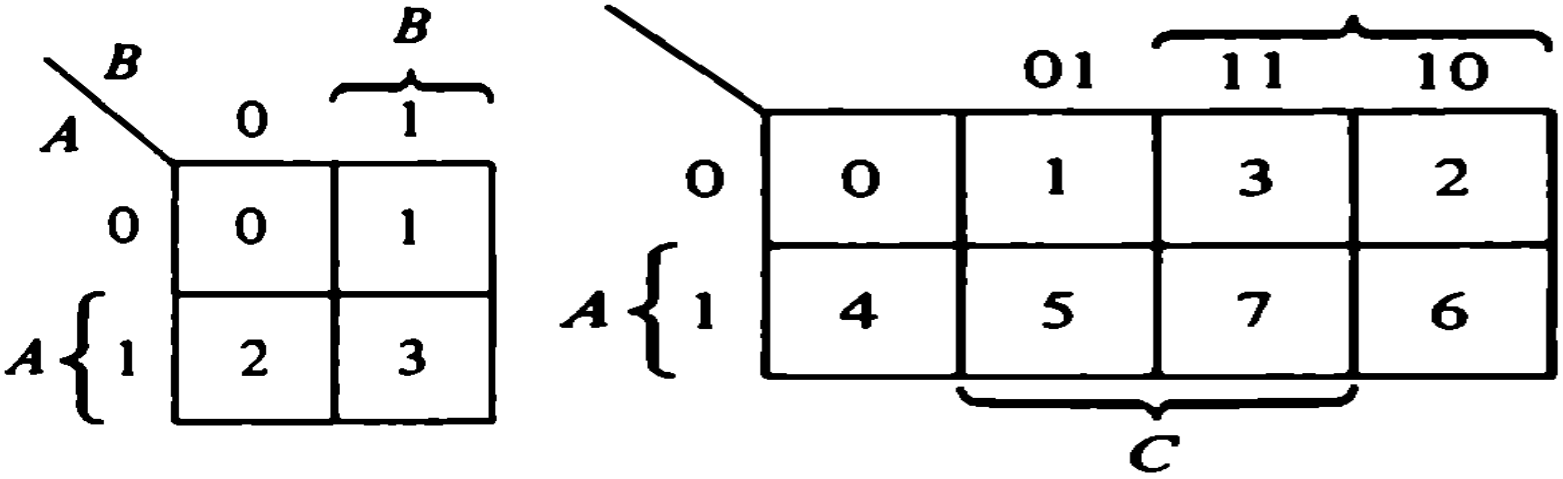
Map Simplification

- The expression may be simplified using the basic relations of Boolean algebra.
- However, this procedure is sometimes difficult because it lacks specific rules for predicting each succeeding step in the manipulative process.
- The map method provides a simple, straightforward procedure for simplifying Boolean expressions.
- The map method is also known as the Karnaugh map or K-map

Gate Minimization Cont.. (CO1)

- This method may be regarded as a pictorial arrangement of the truth table which allows an easy interpretation for choosing the minimum number of terms needed to express the function algebraically.

- Each combination of the variables in a truth table is called a ***minterm***. 2 variables and 3 variables maps are shown below-



Gate Minimization Cont.. (CO1)

Example: simplifies the following Boolean function using K-map method

$F(A, B, C) = \sum (3, 4, 6, 7)$

	B'C'	B'C	BC	BC'
A'			1	
A	1		1	1

The simplified function is

$F = AC' + BC$

Consider another example with don't care minterms-

$F(A,B,C) = \sum(0,2,6)$ $d(A,B,C) = \sum(1,3,5)$

The map with don't care conditions is

	B'C'	B'C	BC	BC'
A'	1	X	X	1
A		X		1

The simplified function is

$F = A' + BC'$

Daily Quiz

1. Simplify the following Boolean functions using three and four-variable maps.

- F(A,B,C) = ∑(0, 1,5, 7)
- F(A,B,C) = ∑(1,2,3,6,7)
- F(A,B,C,D) = ∑(0, 2,4,5,6, 7,11,15)
- F(A,B,C,D) = ∑(0, 2,4,5,6, 7, 8, 10, 13, 15)

2. Simplify the Boolean function F together with the don't-care conditions d in

sum-of-products form.

- F(w, x, y, z) = ∑(0, 1, 2 , 3, 7, 8, 10)
- d(w, x, y, z) = ∑(5 , 6, 11, 15)

Recap

In the last lecture we have discussed about

- Combinational and sequential circuits.
- Minimization of boolean expressions using K-map method.

Topic

- Digital components

Topic objective

To discuss about

- Various digital components used in the building of digital equipments

Digital Components (CO1)

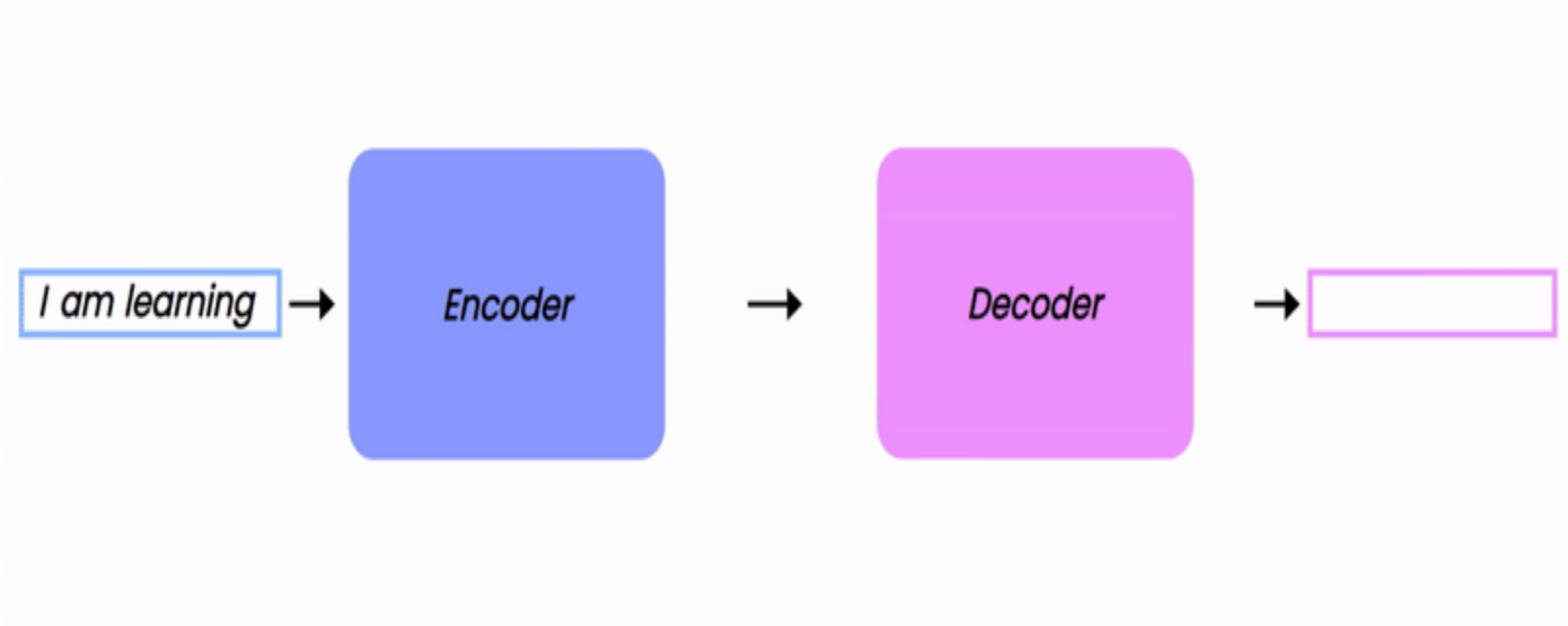
Integrated Circuits:

- Digital circuits are constructed with integrated circuits.
- Integrated circuit (abbreviated IC) is a small silicon semiconductor crystal. called a chip, containing the electronic components for the digital gates.
- The various gates are interconnected inside the chip to form the required circuit.
- The chip is mounted In a ceramic or plastic container, and connections are welded by thin gold wires.

Examples of Digital Components

- Half Adder/Full Adder
-Multiplexer/ Demultiplexer
-Decoder/Encoder
-Registers/Counters

Decoder/Encoder (CO1)

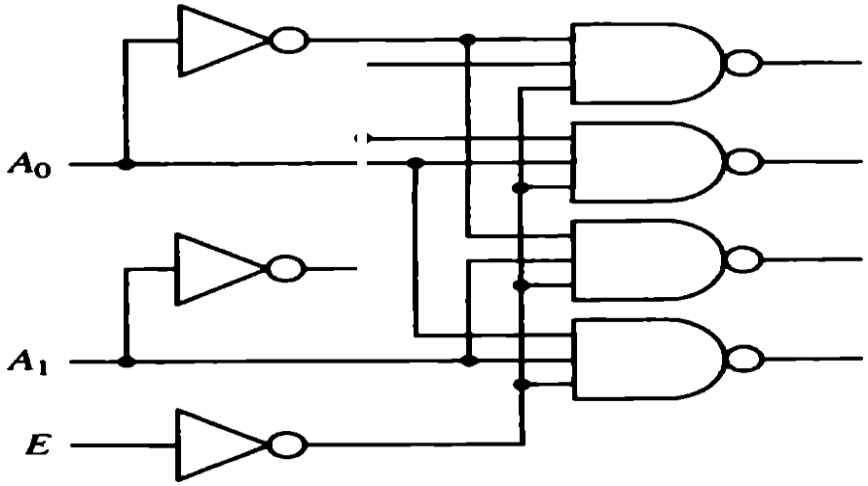


Decoder/Encoder (CO1)

A decoder is a combinational circuit that converts binary information from the n coded inputs to a maximum of 2^n unique outputs. If the n-bit coded information has unused bit combinations, the decoder may have less than 2^n outputs.

An encoder is a digital circuit that performs the inverse operation of a decoder.

An encoder has 2^n(or less) input lines and n output lines.



Multiplexer (CO1)

- A multiplexer is a combinational circuit that receives binary information from one of 2^n input data lines and directs it to a single output line.
- The selection of a particular input data line for the output is determined by a set of selection inputs.
- A 2^n-to-1 multiplexer has 2^n input data lines and n input selection lines whose bit combinations determine which input data are selected for the output.

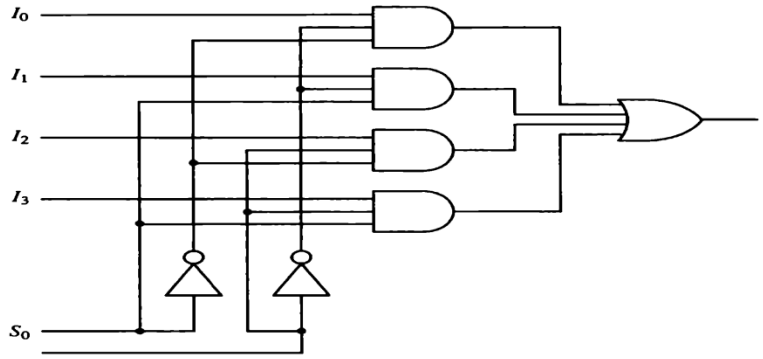


Fig-1.4 4 to 1 line multiplexer.

Table-1.1 Function Table for 4 to 1 Line Multiplexer

SELECT		OUTPUT
S1	S0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Daily Quiz

1. Construct a 5-to-32 line decoder with four 3-to-8 line decoders with enable and one 2-to 1 line decoder. Use block diagrams similar to Fig. 2·3 (M.Mano).

2. Construct a 16-to 1 line multiplexer with two 8 to 1 line multiplexers and one 2 to 1 line multiplexer. Use block diagrams for the three multiplexers.

Topic

- Booth’s multiplication algorithm
- IEEE 754 floating point standard.

Topic objective

- To discuss about arithmetic algorithms for addition and subtraction, Booth multiplication algorithm and representation of floating point numbers into IEEE 754 floating point standard.

Arithmetic Algorithms (CO1)

- In ordinary arithmetic, a negative number is indicated by a minus sign and a positive number by a plus sign.
- Because of hardware limitations, computers must represent everything with 1's and 0's, including the sign of a number.
- The convention is to make the sign bit equal to 0 for positive and to 1 for negative in the leftmost position of the number.

Addition, Subtraction (CO1)

• Addition and Subtraction of two signed binary numbers when negative numbers are in 2' s complement form is very simple and can be stated as follows

• Take the 2's complement of the subtrahend (including the sign bit) and add it to the minuend (including the sign bit). A carry out of the sign bit position is discarded.

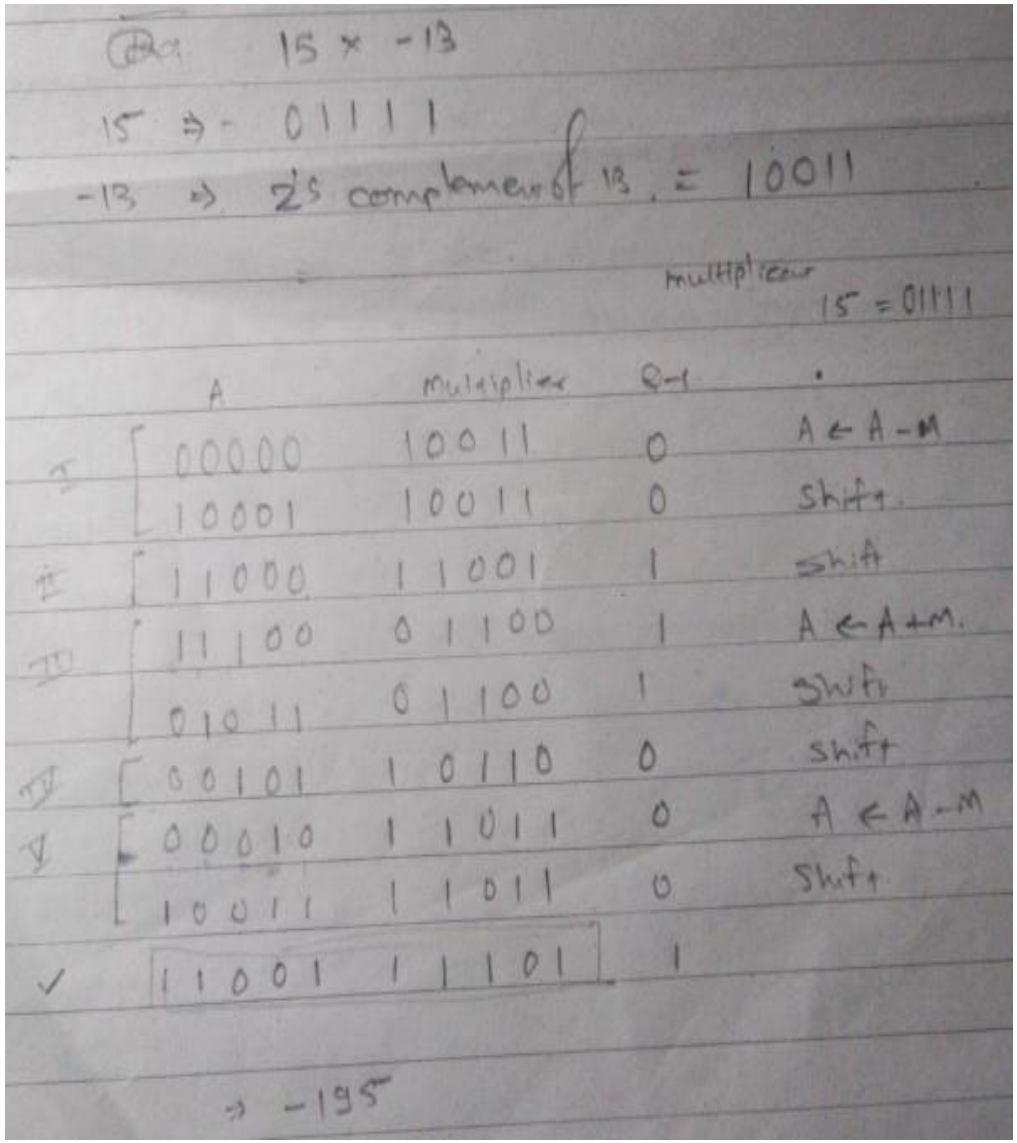
$$(\pm A) - (+ B) = (\pm A) + (- B)$$

$$(\pm A) - (- B) = (\pm A) + (+ B)$$

Booth Multiplication (CO1)

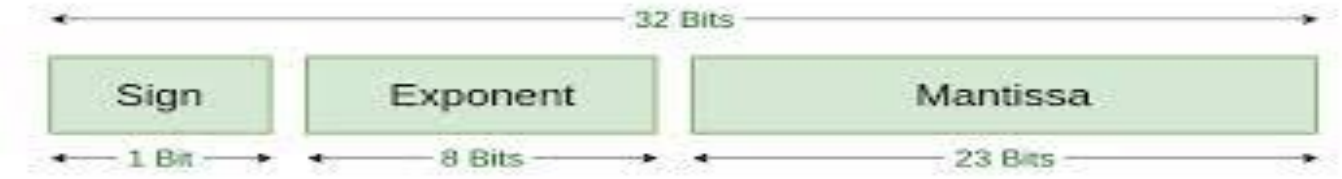
1. Booth algorithm gives a procedure for multiplying binary integers in signed-2's complement representation.

2. It operates on the fact that strings of O's in the multiplier require no addition but just shifting, and a string of 1's in the multiplier from bit weight 2' to weight 2m can be treated as 2'+1 - 2m

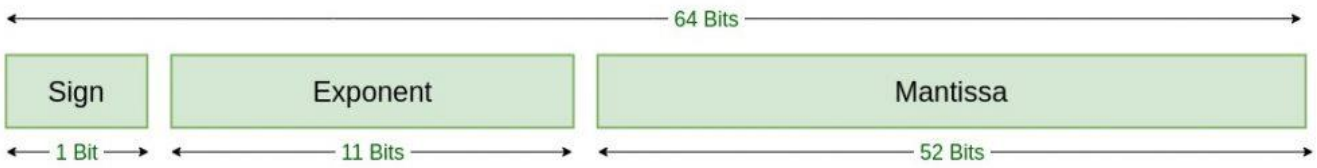


IEEE standard for Floating point numbers (CO1)

The **IEEE** Standard for **Floating-Point** Arithmetic (**IEEE 754**) is a technical standard for **floating-point** arithmetic established in 1985 by the Institute of Electrical and Electronics Engineers (**IEEE**). ... rounding rules: properties to be satisfied when rounding numbers during arithmetic and conversions.



Single Precision
IEEE 754 Floating-Point Standard



Double Precision
IEEE 754 Floating-Point Standard

Daily Quiz

1. Show the step-by-step multiplication process using Booth algorithm when the following binary numbers are multiplied. Assume 5-bit registers that hold signed numbers. The multiplicand in both cases is +15.

   a. (+ 15) x (+ 13)
   b. (+ 15) X (- 13)

2. Represent – (0.75) into IEEE 754 standard for single precision and double precision.

Weekly Assignment

1. A sequential circuit has two D flip-flops A and B, two inputs x and y, and one output z. The flip-flop input equations and the circuit output are as follows:

$D_A = x'y + xA$

$D_B = x\ 'B + xA$

$z = B$

a. Draw the logic diagram of the circuit.

b. Tabulate the state table.

2. Show the contents of registers E, A, Q, and SC (as in Table 10-2) during the process of multiplication of two binary numbers, 11111 multiplicand) and 10101 (multiplier). The signs are not included.

3. Show the contents of registers E, A, Q, and SC (as in Fig. 10-12) during the process of division of (a) 10100011 by 1011; (b) 00001111 by 0011 . (Use a dividend of eight bits.)

Glossary

1. Single choice questions (Glossary)

(**Transistor, Cache, D, 1)**

a. 1 + x = --------.

b. ---------- type of materials used in second generation.

c. Hexa decimal of 13 is ----------.

d. ---------- memory is placed in between main memory and CPU.

2. (Bus, Secondary, Invertor, Universal gate)

Magnetic is ---------- type memory.

a. The communication line between the CPU, memory and other devices is called ----------.

b. NAND and NOR are called ----------.

c. A NOT gate is also known as ----------.

Glossary

3. (word size,1, Instruction Set, Accumulator)

a. ---------- is a general purpose processing register.

b. The length of a register is called _____.

c. The number of sign bits in a 32-bit IEEE format _____.

d. CPU has built-in ability to execute a particular set of machine instructions, called as _____.

4.

Faculty Video Links, YouTube & NPTEL Video Links and Online Courses Details

- YouTube/other Video Links

- https://www.youtube.com/watch?v=yKPD_UkbgXo
- https://www.youtube.com/watch?v=CDO28Esqmcg
- https://www.youtube.com/watch?v=SZu2pchW54Q
- https://www.youtube.com/watch?v=L9X7XXfHYdU&list=PLxCzCOWd7aiHMonh3G6QNKq53C6oNXGrX

Multiple Choice Questions

1. The _____ format is usually used to store data.
   a) BCD b) Decimal c) Hexadecimal d) Octal
2. The 8-bit encoding format used to store data in a computer is _____
   a) ASCII b) EBCDIC c) ANCI d) USCII
3. Which memory device is generally made of semiconductors?
   a) RAM b) Hard-disk c) Floppy disk d) Cd disk
4. The small extremely fast, RAM's are called as _____
   a) Cache b) Heaps c) Accumulators d) Stacks
5. The I/O interface required to connect the I/O device to the bus consists of _____
   a) Address decoder and registers
   b) Control circuits
   c) Address decoder, registers and Control circuits
   d) Only Control circuits

Old Question Papers

- http://www.aktuonline.com/papers/mca-1-sem-computer-organization-and-architecture-rca-104-2018-19.html
- http://www.aktuonline.com/papers/mca-1-sem-computer-organization-and-architecture-rca-104-2017-18.html
- http://www.aktuonline.com/papers/mca-1-sem-computer-organization-and-architecture-rca-104-2016-17.html

Expected Questions for University Exam

Q-1. Perform the arithmetic operation (+42) + (-77) using 2,s complement representation for negative numbers.

Q-2. Define (r-1)'s complement with example.

Q-3. Represent decimal numbers from 0 to 9 in Excess-127 binary code.

Q-4. Determine by means of a truth table the validity of DeMorgan's theorem for three variables: (A+B+C)' =A'.B'. C'

Q-5.Simplify the following Boolean functions using three variable maps.

$F(x,y,z) = \sum(0,1,5,7)$

$F(x,y,z) = \sum(0,2,3,4,6)$

Q-6. Explain all logic gates with symbol, algebraic expression and truth table.

Summary

1. As the technology of ICs has improved, the number of gates that can be put in a single chip has increased considerably.
2. Very large scale integration (VLSI) devices contain thousands of gates within a single package. Examples are large memory arrays and complex microcomputer chips.
3. Various types of registers are available commercially, the simplest register is one that consists only of flip flop, with no external gates.
4. A flip-flop is a binary cell capable of storing one bit of information.
5. The algebraic operation symbol of the AND function is the same as the multiplication symbol of ordinary arithmetic.
6. OR operation is like arithmetic addition, but the output must be 0 or 1.
7. A buffer does not produce any particular logic function. This circuit is merely used for power amplification.

References

1. Logic and Digital Design, ***Morris mano and Kimicharels 4th Edition, Prentice Hall.***

2.Computer System Architecture, M.Mano(PHI)

3.Computer Organization, Vravice, Zaky&Hamacher (TMH Publication)

4. Structured Computer Organization, Tannenbaum(PHI)

5. Computer Organization, Stallings(PHI)

6. Computer Organization, John P.Hayes (McGraw Hill)

Thank You