ASIAN INSTITUTE OF COMPUTER STUDIES
Bachelor of Science in Computer Science
Course Modules
CS316 – Architecture and Organization
3rd Year – 1st Semester

MODULE 1:DIGITAL LOGIC
WEEK 1

Learning Outcomes:

After completing this course you are expected to demonstrate the following:

1. *Familiarize the overview and history of Computer Architecture*
2. *Explain Computer Architecture*
3. *Understand flip flops*

A. Engage

GAME

The content contain short game or trivia related to Digital Logic. It will give you the ideas about terms and concept in the said topic.

Follow the link: https://quizizz.com/join/quiz/56eaa8bfd3acdda523c3117e/start

B. Explore

Video title: Intro to Computer Architecture
YouTube Link:https://www.youtube.com/watch?v=HEjPop-aK_w
Module VideoFilename:**Week 1 – Intro to Computer Architecture**

C. Explain

1. Overview and History of Computer Architecture

Computer Architecture is the field of study of selecting and interconnecting hardware components to create computers that satisfy functional performance and cost goals. It refers to those attributes of the computer system that are visible to a programmer and have a direct effect on the execution of a program. Computer Architecture concerns Machine Organization, interfaces, application, technology, measurement & simulation that Includes:

- **Instruction set**
- **Data Formats**
- **Principle of operation (**formal description of every operation)
- **Features (**organization of programmable storage, registers used, interrupts mechanism, etc.)

2. Brief History of Computer Architecture

FIRST GENERATION (1940-1950) : Vacuum Tube

- **ENIAC [1945]:** Designed by Mauchly & Echert, built by US Army to calculate trajectories for ballistic shells during World War II. Around 18000 vacuum tubes and 1500 relays were used to build ENIAC and it was programmed by manually setting switches
- **UNIVAC [1950]:** the first commercial computer
- **JOHN VON NEUMANN Architecture:** Goldstine and Von Neumann took the idea of ENIAC and developed concept of storing a program in the memory. Known as the Von

Neumann's architecture and has been the basis for virtually every machine designed since then.

- **Features:**
  - Electron emitting devices
  - Data and programs are stored in a single read-write memory
  - Memory contents are addressable by location, regardless of the content itself
  - Machine language / Assemble language
  - Sequential execution

SECOND GENERATION (1950-1964): Transistors

- William Shockley, John Bardeen and Walter Brattain invent the transistor that reduce the size of computers and improve reliability. Vacuum tubes has been replaced by transistors
- First operating system: handled one program at a time
- On-off switches controlled by electronically
- High-level languages
- Floating point arithmetic

THIRD GENERATION (1964 -1974): Integrated Circuits (IC)

- Microprocessor chips combines thousands of transistors, entire circuit on one computer chip
- Semiconductor memory
- Multiple computer models with different performance characteristics
- The size of computers has been reduced drastically

FOURTH GENERATION(1974-Present): Very Large-Scale Integration (VLSI) /Ultra Large Scale Integration (ULSI)

- Combines million of transistors
- Single-chip processor and the single-board computer emerged
- Creation of the Personal Computer (PC)
- Use of data communications
- Massively parallel machine

D. Elaborate

3. Combinational Circuits

A combinational circuit comprises of logic gates whose outputs at any time are determined directly from the present combination of inputs without any regard to previous inputs.

It performs a specific information-processing operation fully specified logically by a set of Boolean functions.

The basic components of a combinational circuit are:

a. input variables,
b. logic gates, and;
c. output variables.

Design procedure of a Combinational Circuit

The design procedure of a combinational circuit involves the following steps:

1. The problem is stated.
2. The total number of available input variables and required output variables is determined.
3. The input and output variables are allocated with letter symbols.
4. The exact truth table that defines the required relationships between inputs and outputs is derived.
5. The simplified Boolean function is obtained from each output.
6. The logic diagram is drawn.

The combinational circuit that performs the addition of two bits is called a **half adder** and the one that performs the addition of three bits (two significant bits and a previous carry) is a **full adde**r.

3.1Half - Adder

A Half-adder circuit needs two binary inputs and two binary outputs. The input variable shows the augend and addend bits whereas the output variable produces the sum and carry. We can understand the function of a half-adder by formulating a truth table.

The truth table for a half-adder is:

1. 'x' and 'y' are the two inputs, and S (Sum) and C (Carry) are the two outputs.
2. The Carry output is '0' unless both the inputs are 1.
3. 'S' represents the least significant bit of the sum.

The simplified sum of products (SOP) expressions is:

S = x'y+xy', C = xy

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

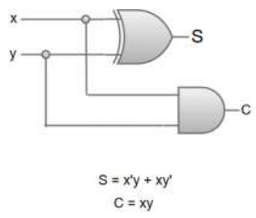

Figure 1.1Logic diagram for a Half-Adder circuit

3.2 Full - Adder

This circuit needs three binary inputs and two binary outputs. The truth table for a full-adder is:

1. Two of the input variable 'x' and 'y', represent the two significant bits to be added.
2. The third input variable 'z', represents the carry from the previous lower significant position.
3. The outputs are designated by the symbol 'S' for sum and 'C' for carry.
4. The eight rows under the input variables designate all possible combinations of 0's, and 1's that these variables may have.
5. The input-output logical relationship of the full-adder circuit may be expressed in two Boolean functions, one for each output variable.

6. Each output Boolean function can be simplified by using a unique map method.

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

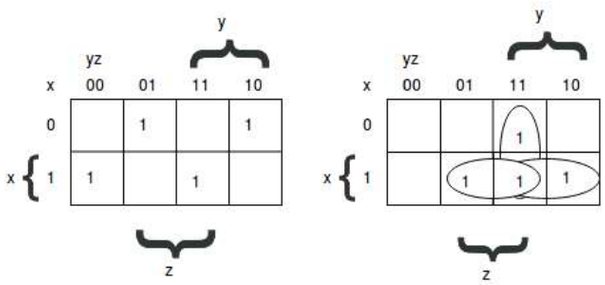

Figure 4.2 Maps for Full Adder

Figure 3.2 Maps for Full Adder

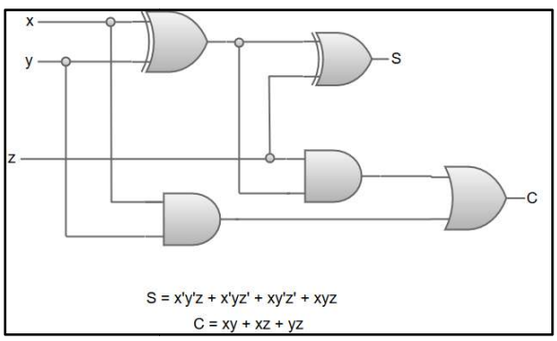

Figure 2.3 Maps for Full Adder

4. S-R Flip-flop/Basic Flip-Flop

Flip flops are an application of logic gates. A flip-flop circuit can remain in a binary state indefinitely (as long as power is delivered to the circuit) until directed by an input signal to switch states.

S-R flip-flop stands for SET-RESET flip-flops. It consist of two NOR gates and two NAND gates (S-R Latch)

5. Clocked S-R Flip-Flop

The operation of a basic flip-flop can be modified by providing an additional control input that determines when the state of the circuit is to be changed.

The limitation with a S-R flip-flop using NOR and NAND gate is the invalid state. This problem can be overcome by using a stable SR flip-flop that can change outputs when certain invalid states are met, regardless of the condition of either the Set or the Reset inputs.

A clock pulse is given to the inputs of the AND Gate. If the value of the clock pulse is '0', the outputs of both the AND Gates remain '0'.

6. D Flip-Flop

D flip-flop is a slight modification of clocked SR flip-flop.

When the value of CP is '1' (HIGH), the flip-flop moves to the SET state if it is '0' (LOW), the flip-flop switches to the CLEAR state.

7. J-K Flip-Flop

J-K flip-flop can be considered as a modification of the S-R flip-flop.
The main difference is that the intermediate state is more refined and precise than that of an S-R flip-flop.

8. **T Flip-Flop**

T flip-flop is a much simpler version of the J-K flip-flop.

E. Evaluate

ASSESSMENT:

Instructions: You may write your answer on the Answer Sheet (AS) provided in this module.

CONTENT FOR ASSESSMENT:

1. What do you think is the imporatance of Computer Architecture? (5-points)
2. Discuss the difference of using half-adder from full-adder circuit? (5-points)
3. What is the significance of studying different flip-flops? (5-point)

References:

1. *Introduction to Computer Architecture:* https://www.youtube.com/watch?v=HEjPop-aK_w
2. **Architecture & Organization:**https://www.javatpoint.com/computer-organization-and-architecture-tutorial
3. **Combinational Circuits:** https://www.javatpoint.com/combinational-circuits

Facilitated By:		
Name	:	
MS Teams Account (email)	:	
Smart Phone Number	:	