EXPERIMENT NO: 11

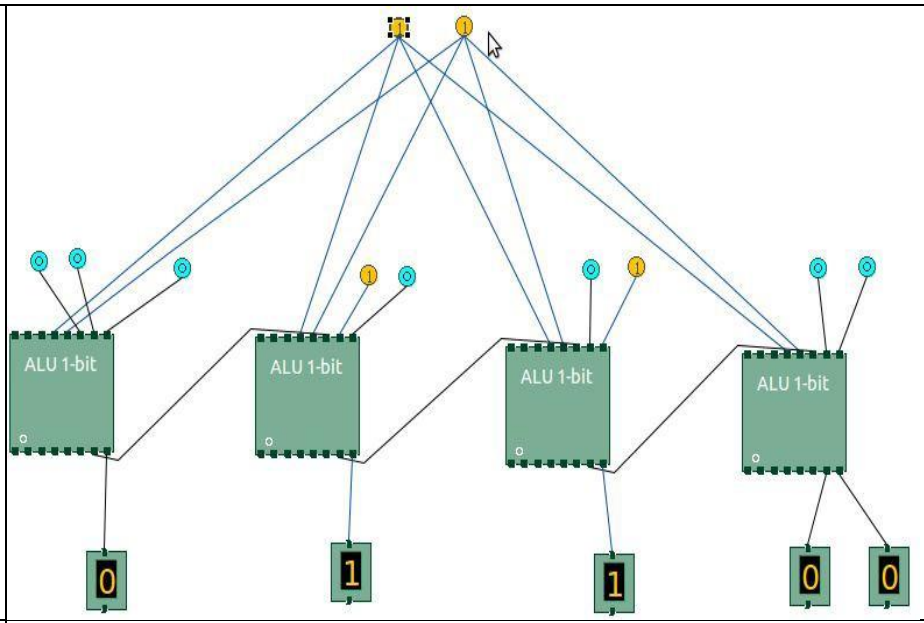
Title: To implement an arithmetic logic unit

Name: Hayden Cordeiro
Class: SE Comps
Roll No: 5

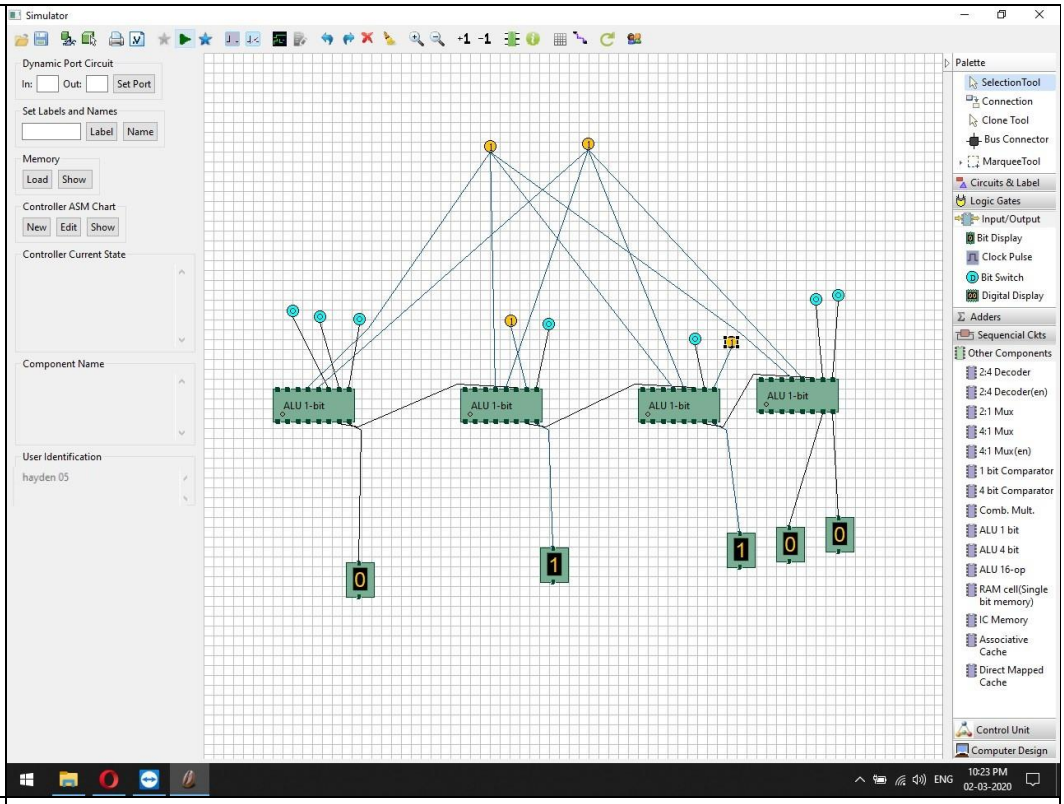
EXPERIMENT NO: 11

Simulate an arithmetic logic unit

AIM	To implement an arithmetic logic unit using simulator
LEARNING OBJECTIVE	To explore a simulation tool for computer organization components.
LEARNING OUTCOME	Students can simulate the operation of ALU unit .
LAB OUTCOME	CSL 403.2: Ability to estimate the output of computer hardware operations using simulator.
PROGRAM OUTCOME	PO1-1, PO3-2, PO4-1, PO5-2, PO8-3, PO9-3, PO12-2, PSO1-1
BLOOM'S TAXONOMY LEVEL	Evaluate
THEORY	An arithmetic logic unit (ALU) is a digital circuit used to perform arithmetic and logic operations. It represents the fundamental building block of the central processing unit (CPU) of a computer. Modern CPUs contain very powerful and complex ALUs. In addition to ALUs, modern CPUs contain a control unit (CU). Most of the operations of a CPU are performed by one or more ALUs, which load data from input registers. A register is a small amount of storage available as part of a CPU. The control unit tells the ALU what operation to perform on that data, and the ALU stores the result in an output register. The control unit moves the data between these registers, the ALU, and memory. An ALU performs basic arithmetic and logic operations. Examples of arithmetic operations are addition, subtraction, multiplication, and division. Examples of logic operations are comparisons of values such as NOT, AND, and OR. All information in a computer is stored and manipulated in the form of binary numbers, i.e. 0 and 1. Transistor switches are used to manipulate binary numbers since there are only two possible states of a switch: open or closed. An open transistor, through which there is no current, represents a 0. A closed transistor, through which there is a current, represents a 1. Operations can be accomplished by connecting multiple transistors. One transistor can be used to control a second one - in effect, turning the transistor switch on or off depending on the state of the second transistor. This is referred to as a gate because the arrangement can be used to allow or stop a current.

	
COMPONENTS USED	The components needed to create 4 bit carry lookahead adder is listed here - • 7 half-adders: 4 to create the look adder circuit, and 3 to evaluate Si and Pi · Ci • 3 OR gates to generate the next level carry Ci+1 • wires to connect • LED display to obtain the output

STEPS TO DESIGN THE CIRCUIT	Start the simulator as directed.This simulator supports 5-valued logic. 1. To design the circuit we need 4 1-bit ALU, 11 Bit switch (to give input,which will toggle its value with a double click), 5 Bit displays (for seeing output), wires. 2. The pin configuration of a component is shown whenever the mouse is hovered on any canned component of the palette. Pin numbering starts from 1 and from the bottom left corner (indicating with the circle) and increases anticlockwise. 3. For 1-bit ALU input A0 is in pin-9,B0 is in pin-10, C0 is in pin-11 (this is input carry), for selection of operation, S0 is in pin-12, S1 is in pin-13, output F is in pin-8 and output carry is pin-7 4. Click on the 1-bit ALU component (in the Other Component drawer in the pallet) and then click on the position of the editor window where you want to add the component (no drag and drop, simple click will serve the purpose), likewise add 3 more 1-bit ALU (from the Other Component drawer in the pallet), 11 Bit switches and 5 Bit Displays (from Display and Input drawer of the pallet,if it is not seen scroll down in the drawer), 3 digital display and 1 bit Displays (from Display and Input drawer of the pallet,if it is not seen scroll down in the drawer) 5. To connect any two components select the Connection menu of Palette, and then click on the Source terminal and click on the target terminal. According to the circuit diagram connect all the components. Connect the Bit switches with the inputs and Bit displays component with the outputs. After the connection is over click the selection tool in the pallete. 6. See the output, in the screenshot diagram we have given the value of S1 S0=11 which will perform add operation and two number input as A0 A1 A2 A3=0010 and B0 B1 B2 B3=0100 so get output F0 F1 F2 F3=0110 as sum and 0 as carry which is indeed an add operation.you can also use many other combination of different values and check the result. The operations are implemented using the truth table for 4 bit ALU given in the theory.
SIMULATED	

	
CONCLUSION	Hence, we implemented the circuit for arithmetic logic unit using the simulator. The outputs were verified with the respective theoretical values and the simulation was completed successfully.
REFERENCES	**http://sit.iitkgp.ernet.in/~coavl/Simulator.html**