**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **27/05/2020** | **Name:** | **PRIYA P RAO** |
| **Course:** | **Logic Design** | **USN:** | **4AL18EC041** |
| **Topic:** | * **Boolean equation for digital circuits** * **Combinational circuits – Conversion of MUX and Decoders to logic gates** * **Design of 7 segment decoder with common anode display** | **Semester & Section:** | **4TH sem ‘A’ section** |
| **Github Repository:** | **Priya-Rao** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session**  **C:\Users\Pawan\Desktop\today3.PNG**  **C:\Users\Pawan\Desktop\today1.PNG**  **C:\Users\Pawan\Desktop\today2.PNG** |
| **Chapter 1: Boolean equation for digital circuits:**  **Boolean Algebra is used to analyze and simplify the digital (logic) circuits. It uses only the binary numbers i.e. 0 and 1. It is also called as Binary Algebra or logical Algebra.**  **Rule in Boolean Algebra:**   * **Variable used can have only two values. Binary 1 for HIGH and Binary 0 for LOW.** * **Complement of a variable is represented by a bar over the variable (-). Thus, complement of variable B is represented as B Bar. Thus if B = 0 then B Bar = 1 and B = 1 then B Bar = 0.** * **OR operation of the variables is represented by a plus (+) sign between them. For example OR operation of A, B, C is represented as A + B + C.** * **Logical AND operation of the two or more variable is represented by writing a dot between them such as A.B.C. Sometime the dot may be omitted like ABC.**   **Boolean Laws:**  **There are six types of Boolean Laws.**   * **Commutative law**   **Any binary operation which satisfies the following expression is referred to as commutative operation.**  Commutative Law  **Commutative law states that changing the sequence of the variables does not have any effect on the output of a logic circuit.**   * **Associative law**   **This law states that the order in which the logic operations are performed is irrelevant as their effect is the same.**  Associative Law   * **Distributive law:**   **Distributive law states the following condition.**  **Distributive Law**   * **AND law:**   **These laws use the AND operation. Therefore they are called as AND laws.**  **AND Law**   * **OR law**   **These laws use the OR operation. Therefore they are called as OR laws.**  OR Law   * **INVERSION law**   **This law uses the NOT operation. The inversion law states that double inversion of a variable results in the original variable itself.**  NOT Law  **Chapter 2: Combinational circuits – Conversion of MUX and Decoders to logic gates.**  **It is a Combinational circuit which have many data inputs and single output depending on control or select inputs.​ For N input lines, log n (base2) selection lines, or we can say that for 2n input lines, n selection lines are required. Multiplexers are also known as** **“Data n selector, parallel to serial convertor, many to one circuit, universal logic circuit​”. Multiplexers are mainly used to increase amount of the data that can be sent over the network within certain amount of time and bandwidth.**  **A Decoder is a combinational circuit that converts binary information from input lines to unique output lines. Apart from the Input lines, a decoder may also have an Enable input line.**  **Chapter 3: Design of 7 segment decoder with common anode display**   * **The basic idea involves driving a common cathode 7 - segment LED display using combinational logic circuit.** * **The logic circuit is designed with 4 inputs and 7 outputs, each representing an input to the display IC.** * **Using Karnough’s map, logic circuitry for each input to the display is designed.**   **C:\Users\Pawan\Downloads\New Doc 2020-05-27 18.14.11_1.jpg** |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **27/05/2020** | **Name:** | **PRIYA P RAO** |
| **Course:** | **Python** | **USN:** | **4AL18EC041** |
| **Topic:** | **Application 5 – How the output will look like.** | **Semester & Section:** | **4TH sem ‘A’ section.** |
| **Github Repository:** | **Priya-Rao** |  |  |

|  |
| --- |
| **AFTERNOON SESSION DETAILS** |
| **Image of session**  **C:\Users\Pawan\Desktop\today4.PNG** |
| **Chapter 1: Application 5 – How the output will look like.**   * **User interface design.** * **Frontend and Backend interface.** * **Connecting the Frontend to the Backend fixing the bug.** * **Creating the Standalone Executable Version of the Program.** |