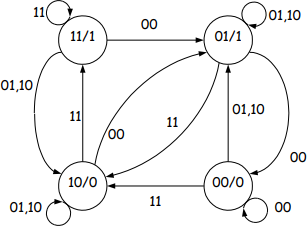
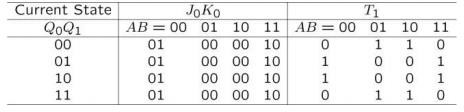
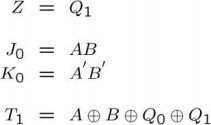
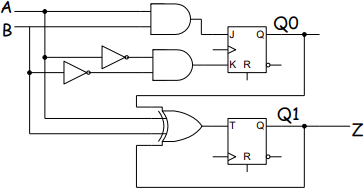
## DAILY ASSESSMENT

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 29-5-2020 | **Name:** | Rasika Patil |
| **Course:** | Logic design | **USN:** | 4AL16EC057 |
| **Topic:** | Analysis of clocked sequential circuits, Digital clock design | **Semester &**  **Section:** | 8th B |
| **Github**  **Repository:** | Rasika B Patil |  |  |

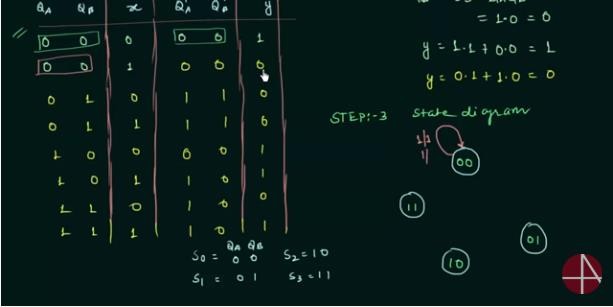
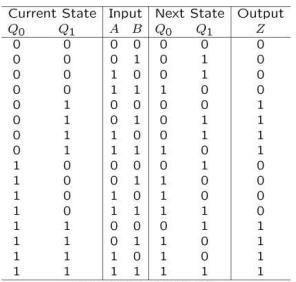
|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **REPORT:**   **Analysis of Clocked Sequential Circuits (with D Flip Flop)**   * The behavior of a clocked sequential circuit is determined from its inputs, outputs and state of the flip-flops (i.e., the output of the flip-flops). * The analysis of a clocked sequential circuit consists of obtaining a table of a diagram of the time sequences of inputs, outputs and states.   Consider the following circuit |



The circuit has two inputs A and B, one output Z, The circuit has two flip-flops (different types) with outputs Q0 and Q1 (This implies that there are as many as 4 different states in the circuit, namely Q0Q1 = 00, 01, 11, 10). The circuit output depends on the current state (flip-flop outputs) only.

Another way to illustrate the behavior of a clocked sequential circuit is with a state diagram.

Each “bubble” (state bubble) in the state diagram represents one state of the system. The flip-flop outputs that correspond to that state are labeled inside of the bubble. Each edge leaving a bubble represents a possible transition to another state once the active clock edge arrives. The edges are labeled with the input values that cause the transition to occur. In this state diagram, the output values are labeled inside of the state bubbles.



* **Digital clock design:**
* There are many stuff on the internet about this project, but we are gonna add something or two.

|  |  |
| --- | --- |
| * This circuit was a school project and it was a 24h clock, but i decided to extend it to 12h and 24h with the transferring between them using a switch so you can choose whatever mode you want. * Anyway, i made this circuit just for fun so i just simulated it on proteus, i.e we're not going into pcb design.     The main parts of the circuit are as follows:   1. **Timer 555**: Responsible for generating the clock pulses for the counters, the frequency of the output shoul be 1 hz which means 1 second for each pulse. 2. **Counters**: Responsible for generating the time in BCD (Binary Coded decimal). 3. **Decoders** : Takes the BCD of the counter as input and produces 7 segment output . 4- 7 **segments** : Displays the time, of course   MSB---LSB | |
| 0: | 0000 |
| 1: | 0001 |
| 2: | 0010 |
| 3: | 0011 |
| 4: | 0100 |
| 5: | 0101 |

|  |  |
| --- | --- |
| 6: | 0110 |
| 7: | 0111 |
| 8: | 1000 |
| 9: | 1001 |
| 0: | 0000 |
| Remember that 7490 decade counters respond only to the pulses that go from 1 to 0 and notice that this case only happens in the BCD code above when the output changes from 9 to 0 (the Most significant bit changes from 1 to 0). So, we'll just connect the clock input of the 2nd counter to the most significant bit of the output of the first counter.  The 4th counter will be the same as the second one so we are clocking it using the Most Significant Bit of the output of the previous one.  Again, the 5th counter is the same as the 3rd one and takes its clock from the AND gate.  The 5th and the 6th counters are responsible for hours so they are limited to 23, and resets themselves to 00 when the 5th counter is 4 and the last one is 2 (24).  This is done using and gate with Q2 (3rd bit) of the 5th counter as one input and Q1 (second bit) of the last counter as the other input, and the output of this AND gate will be connected to both resets of the last 2 counters.  When the last counter is 0(0000) or 1(0001), Q1 which is one of the inputs to the AND gate will be 0 so the output of the AND gate will be zero. when it counts to 2 this bit will be 1 so the output of the and gate will depend on the the other input which is Q2 of the previous counter, and this bit will be zero until it reaches 4 (0100),So, the output of the and gate will be 1 (0--->1) resetting both counters to 00, | |

