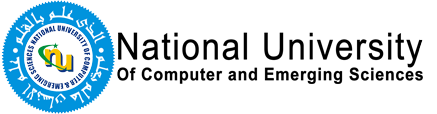
|  |  |
| --- | --- |
|  | **2021** |
|  | Armghan Ahmad  BSCS 2B |

|  |
| --- |
| **[Design 101 sequence detector]** |
| Roll no. :20P-0183 |



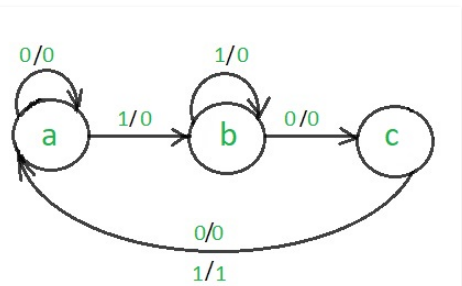
Design 101 sequence detector

A **sequence detector** is a sequential state machine that takes an input string of bits and generates an output 1 whenever the target sequence has been detected. In a Mealy machine, output depends on the present state and the external input (x). Hence, in the diagram, the output is written outside the states, along with inputs.

In an overlapping sequence detector, the last bit of one sequence becomes the first bit of the next sequence. However, in a non-overlapping sequence detector, the last bit of one sequence does not become the first bit of the next sequence.

In this project, I am going discuss the design procedure for non-overlapping 101 Mealy sequence detectors.

**Step 1: Develop the state diagram –**   
The state diagram of a Mealy machine for a 101 sequence detector is:



**Step 2: Code Assignment –**

**Rule 1**: States having the same next states for a given input condition should have adjacent assignments.

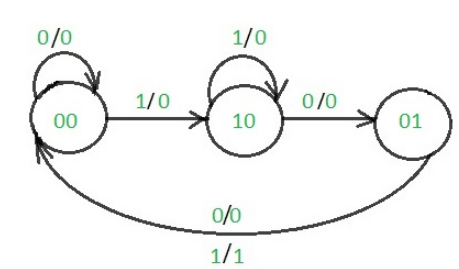
**Rule 2**: States that are the next states to a single state must be given adjacent assignments.   
Rule 1 given preference over Rule 2.

|  |  |  |
| --- | --- | --- |
| Previous  state | state | Next state |
| a , c  b, a  b | a  b  c | a, b  b, c  a |

K-map

|  |  |  |
| --- | --- | --- |
| y\x | 0 | 1 |
| 0 | a | b |
| 1 | c |  |

The state diagram after the code assignment is:



**Step 3: Make Present State/Next State table –**   
We’ll use D-Flip Flops for design purposes.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Previous state | | I/p | Next state | | Flip Flop Executions | | o/p |
| X | Y | I | X’ | Y’ | Dx | Dy | output |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | X | X | X | X | X |
| 1 | 1 | 1 | x | X | x | X | X |

**Step 4: Draw K-maps for Dx, Dy and output (Z) –**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| I \ xy | 00 | 01 | 11 | 10 |
| 0 | 0 | 0 | X | 0 |
| 1 | 1 | 0 | x | 1 |

Dx= Y’.I

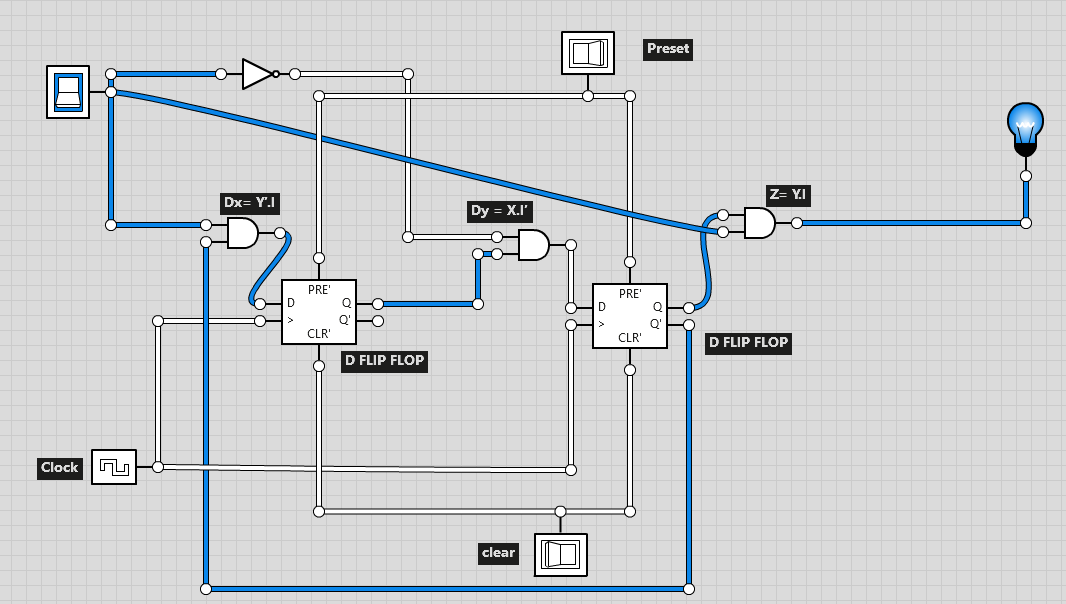
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| I \ xy | 00 | 01 | 11 | 10 |
| 0 | 0 | 0 | X | 1 |
| 0 | 0 | 0 | x | 0 |

Dy = X.I’

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| I \ xy | 00 | 01 | 11 | 10 |
| 0 | 0 | 0 | X | 0 |
| 1 | 0 | 1 | x | 0 |

Z= Y.I

**Step 5: Finally implement the circuit –**



This is the final circuit for a Mealy 101 non-overlapping sequence detector.