5) Design of an FSM (finite state machine) in Verilog
(Implement Using minimum resources.)

Coffee/Tea vending machine using FSM

- Modes: 1)Coffee 2)Tea
- Take $5, $3, $2, $1 coins as inputs.
- Display 1) the amount already inside the machine and 2) the mode selected.
- Deliver a coffee at $10 (light an LED) if mode selected is coffee
- Deliver a tea at $ 15 (light another LED) if mode selected is tea
- Return the balance (display the balance)

6) Design of a sequence detector
(Implement Using minimum resources)

Detect an 8 bit binary sequence when entered one bit at a time. For each matching bit, the bit pattern entered should be displayed on seven-segment display. The 8 bit pattern should be user-defined.

For example, let us consider the input sequence to be detected as **01100100**. Initially, the user should be able to define/set this sequence to be detected using switches. Afterwards, the user should be able to enter continuous input sequences using push buttons. (i.e., user should be able to give either a 0 or a 1 as input bit at any instant.)

If the user enters in the order 011 then 011 should be displayed on the seven-segment display. If the fourth input is 1, then the red led should glow, rejecting a match (display should be cleared). Once the whole pattern 01100100 is entered, a green LED should indicate the match of 8 bit sequence.

On the other hand, if the user enters in the order 011001, 011001 should be displayed. If the seventh input is 1, then the display should change from 011001 to 011 (because the last three bits entered match) instead of being cleared.