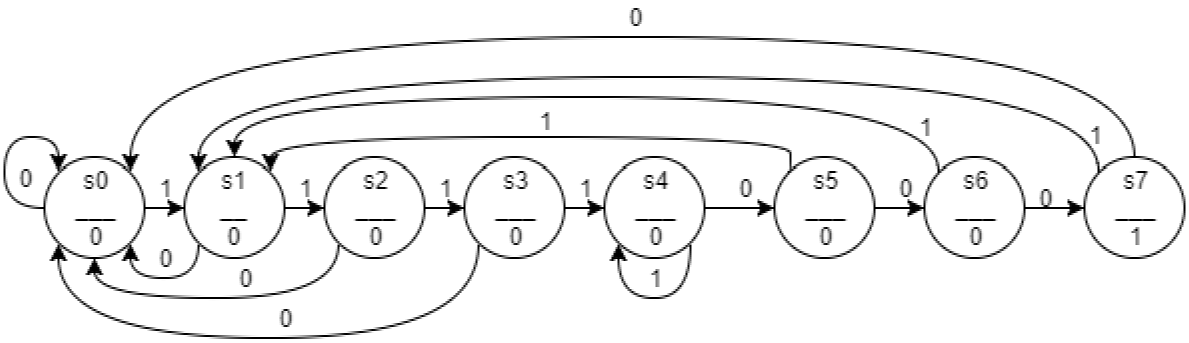
CECS 225: LAB 6

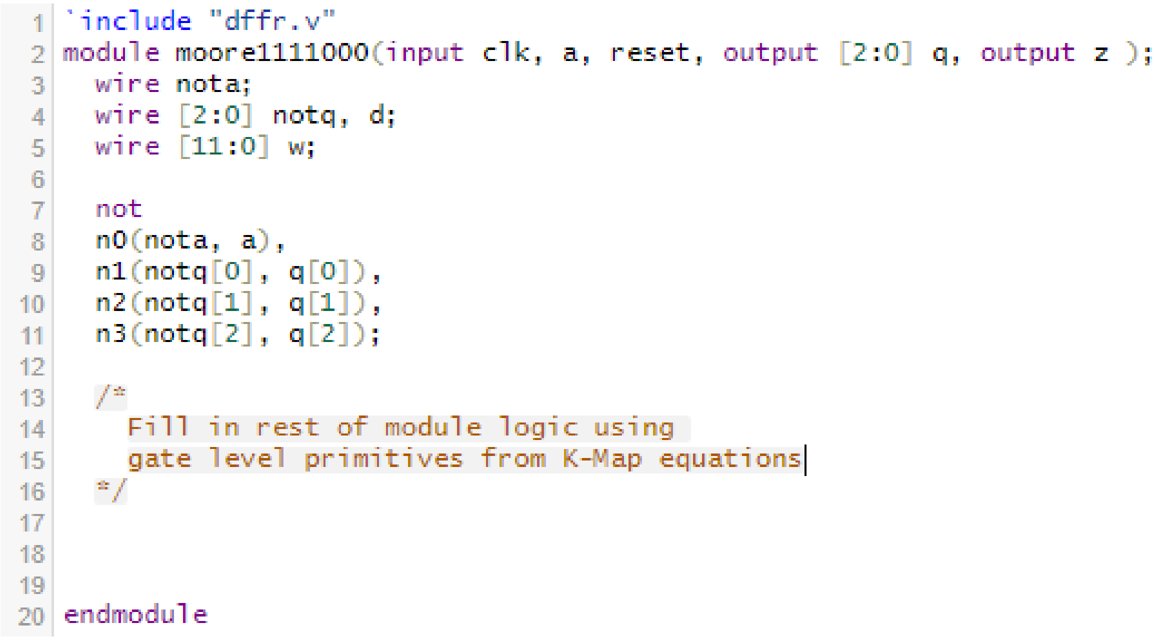
OBJECTIVE: Model a “1111000” detector Moore Finite State Machine.

PROCEDURE: First reference and understand the implementation of the “110” detector discussed in class. The Verilog implementation can be found at: https://www.edaplayground.com/x/3a2C

1. Use the given **state diagram** below as a reference for modeling this detector.

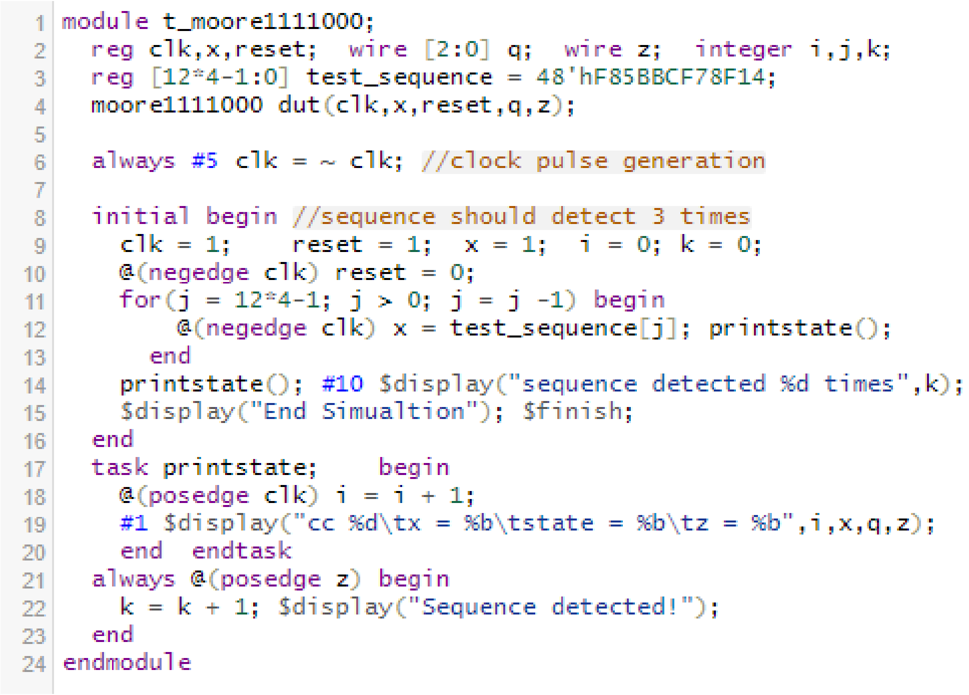


2. “*Encode*” each state by using a 3-bit binary equivalent of its index i.e. *s4 is encoded as 100*
3. *Create* a **state table** for this system.
4. Use **K-Maps** to find **next state equations** for the DFF inputss and *derive* the **output logic equation for variable Z**.
5. *Convert* the equations to logic gates and create a Verilog module using gate level primitives to model your design. (Use the DFF module named **dffr.v** which can be found in the project link at the beginning of this lab description.) Use the given Verilog skeleton module:



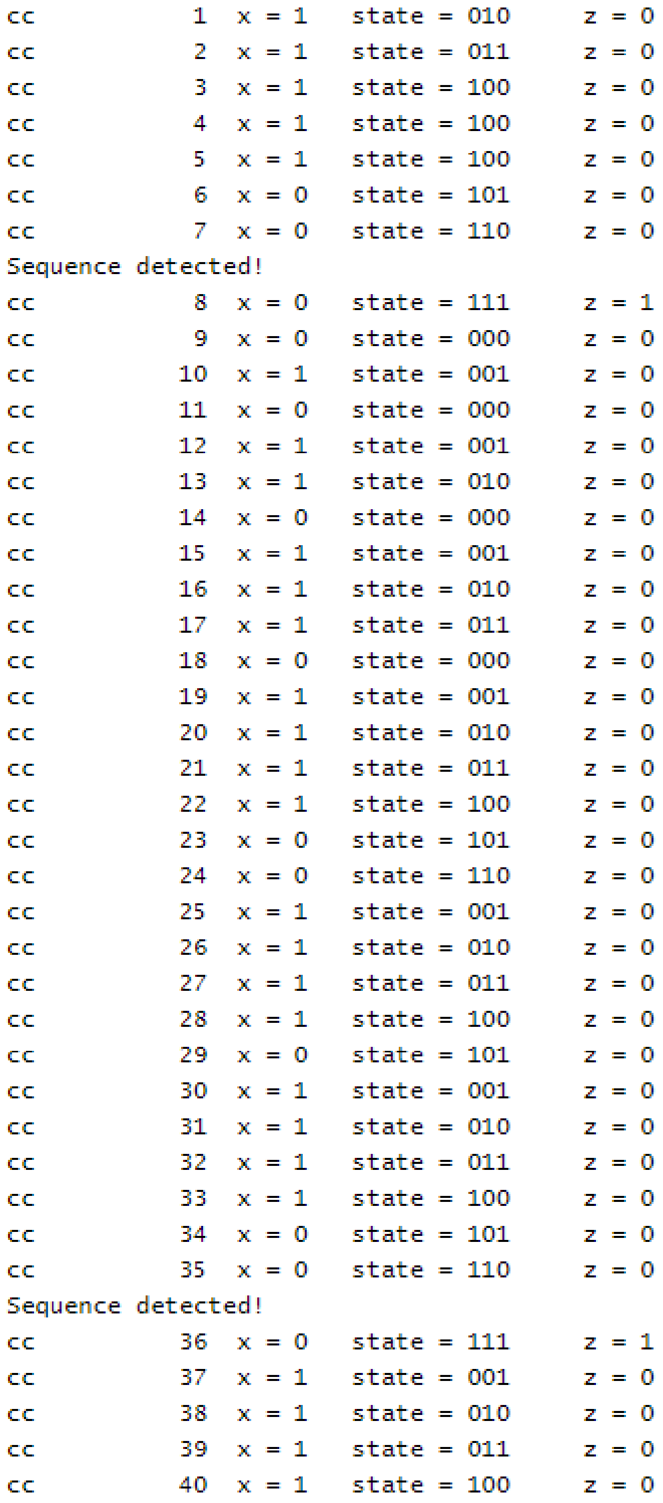
```
`include "dffr.v"
module moore1111000(input clk, a, reset, output [2:0] q, output z );
  wire nota;
  wire [2:0] notq, d;
  wire [11:0] w;

  not
  n0(nota, a),
  n1(notq[0], q[0]),
  n2(notq[1], q[1]),
  n3(notq[2], q[2]);

  /*
    Fill in rest of module logic using
    gate level primitives from K-Map equations
  */



endmodule
```

6. Verify proper functionality of your lab using the given Verilog testbench:

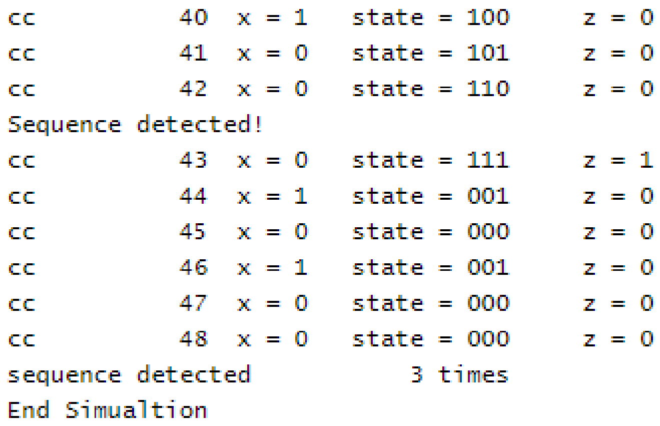


```
module t_moore1111000;
  reg clk,x,reset;  wire [2:0] q;  wire z;  integer i,j,k;
  reg [12*4-1:0] test_sequence = 48'hF85BBCF78F14;
  moore1111000 dut(clk,x,reset,q,z);

  always #5 clk = ~ clk; //clock pulse generation

  initial begin //sequence should detect 3 times
    clk = 1;     reset = 1;  x = 1;  i = 0; k = 0;
    @(negedge clk) reset = 0;
    for(j = 12*4-1; j > 0; j = j -1) begin
        @(negedge clk) x = test_sequence[j]; printstate();
      end
    printstate(); #10 $display("sequence detected %d times",k);
    $display("End Simualtion"); $finish;
  end
  task printstate;    begin
    @(posedge clk) i = i + 1;
    #1 $display("cc %d\tx = %b\tstate = %b\tz = %b",i,x,q,z);
    end  endtask
  always @(posedge z) begin
    k = k + 1; $display("Sequence detected!");
  end
endmodule
```

7. A properly working module should produce the following Console output:



```
cc           1  x = 1   state = 010     z = 0
cc           2  x = 1   state = 011     z = 0
cc           3  x = 1   state = 100     z = 0
cc           4  x = 1   state = 100     z = 0
cc           5  x = 1   state = 100     z = 0
cc           6  x = 0   state = 101     z = 0
cc           7  x = 0   state = 110     z = 0
Sequence detected!
cc           8  x = 0   state = 111     z = 1
cc           9  x = 0   state = 000     z = 0
cc          10  x = 1   state = 001     z = 0
cc          11  x = 0   state = 000     z = 0
cc          12  x = 1   state = 001     z = 0
cc          13  x = 1   state = 010     z = 0
cc          14  x = 0   state = 000     z = 0
cc          15  x = 1   state = 001     z = 0
cc          16  x = 1   state = 010     z = 0
cc          17  x = 1   state = 011     z = 0
cc          18  x = 0   state = 000     z = 0
cc          19  x = 1   state = 001     z = 0
cc          20  x = 1   state = 010     z = 0
cc          21  x = 1   state = 011     z = 0
cc          22  x = 1   state = 100     z = 0
cc          23  x = 0   state = 101     z = 0
cc          24  x = 0   state = 110     z = 0
cc          25  x = 1   state = 001     z = 0
cc          26  x = 1   state = 010     z = 0
cc          27  x = 1   state = 011     z = 0
cc          28  x = 1   state = 100     z = 0
cc          29  x = 0   state = 101     z = 0
cc          30  x = 1   state = 001     z = 0
cc          31  x = 1   state = 010     z = 0
cc          32  x = 1   state = 011     z = 0
cc          33  x = 1   state = 100     z = 0
cc          34  x = 0   state = 101     z = 0
cc          35  x = 0   state = 110     z = 0
Sequence detected!
cc          36  x = 0   state = 111     z = 1
cc          37  x = 1   state = 001     z = 0
cc          38  x = 1   state = 010     z = 0
cc          39  x = 1   state = 011     z = 0
cc          40  x = 1   state = 100     z = 0
```



```
cc           40  x = 1   state = 100     z = 0
cc           41  x = 0   state = 101     z = 0
cc           42  x = 0   state = 110     z = 0
Sequence detected!
cc           43  x = 0   state = 111     z = 1
cc           44  x = 1   state = 001     z = 0
cc           45  x = 0   state = 000     z = 0
cc           46  x = 1   state = 001     z = 0
cc           47  x = 0   state = 000     z = 0
cc           48  x = 0   state = 000     z = 0
sequence detected           3 times
End Simualtion
```

WHAT TO TURN IN: Once your **moore1111000** module is working correctly:

- Copy the contents of your **moore1111000** module to a file named **moore.txt**
- Upload **moore.txt** to the beachboard dropbox named moore1111000