

**EE 231**  
**Homework 10**  
**Due November 5, 2010**

1. Design a synchronous sequential circuit which generates the following sequence. (The sequence should repeat itself.)

```
00000001
00000010
00000001
00000100
00000001
00001000
00000001
00010000
00000001
00100000
00000001
01000000
00000001
10000000
00000001
00000010
```

- (a) Draw a state transition diagram for the circuit.  
 (b) Write a Verilog program to implement the circuit.
2. Design a synchronous sequential circuit which detects the occurrence of at least three 1's arriving at the input. The 1's do not need to arrive in consecutive clock periods. The output will go high after it sees three 1's at the input. The output will stay high until the system sees three **consecutive** 0's at the input. When it sees three consecutive 0's, the circuit should return to the reset state and start looking for three 1's.

Here is what the output should look like for typical input:

Input	0 0 1 1 0 0 0 1 0 1 0 0 1 1 1 0 1 0 0 0 1 0 1 0 1 1 0
Output	0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1

- (a) Draw a state transition diagram for the circuit.  
 (b) Write down a state transition table for the circuit.  
 (c) Write a Verilog program to implement the circuit.
3. Design a serial 8-bit two's complementer. Eight bits are fed serially into the circuit, least significant bit first. The serial output should be the 8-bit two's complement of the input. The circuit will transition back to the reset state after each eight-bit packet is processed. Page 11 of the text discusses how you can find the 2's complement of a number by looking at the bits, starting with the least significant bit.

- (a) Draw a state transition diagram for the circuit.
  - (b) Write a Verilog program to implement the circuit.
4. Design a synchronous sequential circuit with two inputs  $x_1$  and  $x_0$  and a single output  $z$ . The circuit detects any violation of the rule *i before e except after c*. The letter *i* is represented by  $x = 01$ , the letter *e* is represented by  $x = 10$ , the letter *c* is represented by  $x = 11$ , and all other letters are represented by  $x = 00$ . The output  $z$  will go high for one clock cycle if the circuit sees either an *e* followed by an *i* which was not preceded by a *c*, or if the circuit sees *c i e*.
- (a) Draw a state transition diagram for the circuit.
  - (b) Write a Verilog program to implement the circuit.