***King Fahd University of Petroleum and Minerals***

***College of Computer Science and Engineering***

***Computer Engineering Department***

**COE 202: Digital Logic Design (3-0-3)**

**Term 151 (Fall 2015)**

**Final Exam**

**Sunday Dec. 20, 2015**

**7:00 p.m. – 9:30 p.m.**

**Time: 150 minutes, Total Pages: 12**

**Name: \_KEY\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ID: \_\_\_\_\_\_\_\_\_\_\_\_\_ Section: \_\_\_\_\_\_**

**Notes:**

* Do not open the exam book until instructed
* Calculators are not allowed (basic, advanced, cell phones, etc.)
* Answer all questions
* All steps must be shown
* Any assumptions made must be clearly stated

|  |  |  |
| --- | --- | --- |
| **Question** | **Maximum Points** | **Your Points** |
| **1** | **12** |  |
| **2** | **14** |  |
| **3** | **10** |  |
| **4** | **8** |  |
| **5** | **9** |  |
| **6** | **7** |  |
| **Total** | **60** |  |

**Question 1. (12 Points)**

The shown synchronous sequential circuit has a single input X and a single output Y. Answer the following questions:



## The circuit is \_\_MOORE\_\_\_ (Moore / Mealy) **(1 point)**

## Derive Boolean expressions for the flip-flop D-inputs, i.e. DA and DB and the output Y.

**(3 points)**

DA= XA + XB=X.(A+B)

DB= XA + XB’=X.(A+B’)

Y = AB

## Fill in the state table of this circuit and draw the corresponding state diagram **(6 Points)**

0

00

|  |  |  |
| --- | --- | --- |
| **PS**  **AB** | **NS (A+ B+)**  **X=0 X=1** | **Y**  **X=0 X=1** |
| 0 0 | 0 0 0 1 | 0 0 |
| 0 1 | 0 0 1 0 | 0 0 |
| 1 1 | 0 0 1 1 | 1 1 |
| 1 0 | 0 0 1 1 | 0 0 |

1

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

0

0

0

1

1

1

1

0

0

10

11

01

0

## For a starting state of **(10)**, what is the resulting output sequence (**Y**) for an applied input sequence (**X**) of **{1🡪 1🡪0}? (2 points)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PS | (10) | (11) | (11) | (00) |
| X | 1 | 1 | 0 | Don’t care |
| Y | 0 | 1 | 1 | 0 |

## 

**Question 2. (14 Points)**

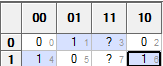
Consider the following state transition table for a synchronous sequential circuit that increments a binary number by two, i.e. Z=X+2. The circuit has a single input **X**, a single output **Z**, and two state variables **Y1**, and **Y0**. The states are encoded using binary codes **00**, **01**, **10.** The **reset state** is **10.**

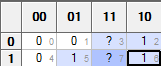
|  |  |  |
| --- | --- | --- |
| **PS** (**Y1 Y0**)t | **NS (Y1 Y0)t+1** | **Z** |
| **X = 0 X = 1** | **X = 0 X = 1** |
| 0 0 | 0 0 0 0 | 0 1 |
| 0 1 | 0 0 0 1 | 1 0 |
| 1 0 | 0 1 0 1 | 0 1 |

## Using D-FFs and **minimal** combinational logic, determine the equations for the D-FF inputs and the output Z for this circuit.

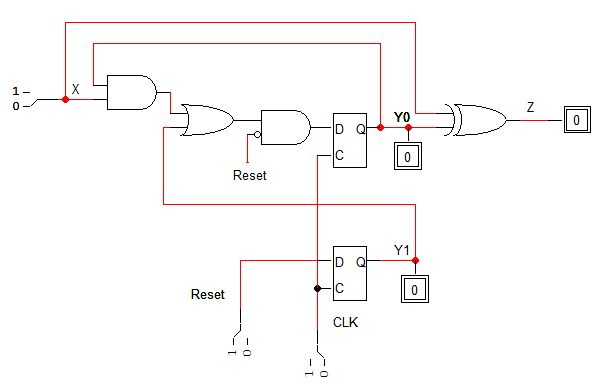
## Draw the resulting circuit and add the required logic to achieve **Synchronous Reset**, to reset the machine to state Y1Y0=**10** when a Reset input is asserted.

## **(7 points)**

 Z = X' Y0 + X Y0' = X ⊕ Y0

 D0 = Y1 + X Y0

D1 = 0



## You are required to implement the above circuit using a **ROM** and a **register** with minimum sizes.

1. What is the minimum size of the ROM (number of memory locations × number of memory bits per location)? **(2 points)**

Since D1=0, we do not need to store it. Thus, the ROM size is 23 x 2 bits

1. Draw the block diagram for such implementation. Add **Asynchronous Reset** to the register to reset the machine to state Y1Y0=**10**. (**Label all components inputs and outputs together with various signals**) **(3 points)**

X

A0

A1

A2

ROM

Z

O0

O1

2-bit Register

Q0 D0

0

Q1 D1

CLK

Reset

Note that the Reset will be connect to Set input for Q1 and to CLR inputs for Q0.

|  |  |
| --- | --- |
| **Binary Address** | **Binary Stored Data** |
| 000 | 00 |
| 001 | 01 |
| 010 | 01 |
| 011 | 10 |

1. Starting from address **0,** fill in the following table to show the data stored in the first four memory locations in the ROM device. **(2 points)**

**Question 3. (10 Points)**

## Using minimum number of states, show the state diagram of a circuit that counts the number of 1's in an input stream (not necessarily consecutive). The circuit has one input X and one output Y. Y remains 0 until 4 ones are received on X, then Y becomes 1 and the circuit starts counting the number of 1s in the input stream again and Y return to 0. The asynchronous Reset input resets the circuit to an initial state where no 1 has been received yet. An example of an input sequence and the corresponding output sequence is shown below: **(5 points)**

t = 0

time

|  |  |
| --- | --- |
| X | 0 0 1 0 1 1 0 0 1 0 0 1 1 0 1 0 1 |
| Y | 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 |

(**NOTE**: You are *only* required to draw the state diagram **Nothing MORE**)

1/0

Reset

0/0

0/0

1/0

0/0

1/0

0/0

1/1

## Show the **Moore-Model** state diagram of a sequential circuit that has two inputs X1, X2 and one output Z. The asynchronous Reset input resets the circuit to an initial state with Z=0. If X1=X2 (i.e. X2X1 = 00 or 11), the circuit remains at the initial state. While in the initial state, if X2X1 become 01, the circuit produces an output of 1 then goes back to the initial state. If X2X1 become 10, the circuit produces the output sequence {1,1} then goes back to the initial state. **If the inputs change while the circuit is producing an output sequence, these changes are ignored until the circuit goes back to the initial state.** An example of an input sequence and the corresponding output sequence is shown below: **(5 points)**

t = 0

time

## 

|  |  |
| --- | --- |
| X2 | 0 1 1 1 1 0 0 0 1 … |
| X1 | 0 1 0 0 0 0 1 1 0 … |
| Z | 0 0 0 1 1 0 0 1 0 1 1 0 … |

(**NOTE**: You are *only* required to draw the state diagram **Nothing MORE**)

10

Reset

00,11

xx

01

xx

xx

10

Reset

00,11

xx

01

xx

**Another solution—S1 and S3 are the same**

**Question 4. (8 Points)**

## Using minimum number of D-FFs and other needed components, show the design of a 3-bit register Q that has two control inputs: Load and Shift. The register has a 3-bit external input I2I1I0 and a serial inputSin. The table below shows the functionality of the register. **(4 points)**

|  |  |  |
| --- | --- | --- |
| Load | Shift | Action |
| 0 | 0 | No change in Q |
| 1 | X | Load parallel input (i.e. Q2Q1Q0🡨 I2I1I0) |
| 0 | 1 | Shift Q to the right with Sin inserted from the left (i.e., most significant bit) |

**Load**

D Q

R

**Q2**

**reset**

**Shift**

**S1 S0**

**Sin**

**I2**

**Load**

D Q

R

**Q1**

**reset**

**Shift**

**S1 S0**

**I1**

**Load**

D Q

R

**Q1**

**reset**

**Shift**

**S1 S0**

**I1**

## Complete the following table by showing the content of register Q after each clock cycle:

## **(4 points)**

Inputs in a cycle affect the register in the next cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clock # | Load | Shift | Sin | D2 D1 D0 | Q2 Q1 Q0 |
| 1 | 0 | 0 | 0 | 0 0 0 | 0 0 0 |
| 2 | 1 | 0 | 0 | 1 0 1 | 0 0 0 |
| 3 | 0 | 1 | 1 | 1 1 1 | 1 0 1 |
| 4 | 0 | 0 | 1 | 0 0 1 | **1 1 0** |
| 5 | 0 | 1 | 0 | 0 0 0 | **1 1 0** |
| 6 | 1 | 1 | 0 | 1 0 0 | **0 1 1** |
| 7 | 0 | 0 | 1 | 1 1 0 | **1 0 0** |

**Question 5 (9 Points)**

***In this question, you are to use only a 4-bit register and MSI parts, and minimum number of gates.***

## Design a modulo-16 counter that can increment by 3, i.e. (0 🡪 3 🡪 6 …13 🡪 0). **(3 points)**

## Show how to modify the above counter such that it can count either up (by 3) or down (by 3) based on the value of an additional control input **U (**if U=1 counting is up otherwise it is down**)**

**(2 points)**

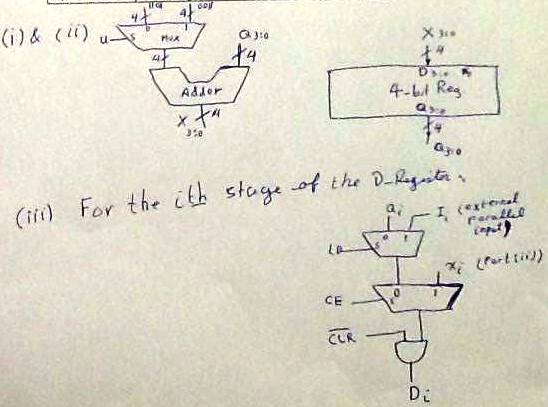
## Show how to modify the above counter to have synchronous clear, parallel load and count enable capabilities as follows: **(4 points)**

|  |  |
| --- | --- |
| **CLR CE LD** | **Operation** |
| 1 X X | Counter is Cleared |
| 0 1 X | Counting (Up or down depending on the value of U) |
| 0 0 1 | Parallel Load of external input Data |
| 0 0 0 | No Change |

**Note that you have the option to show the final design to implement all the requirements without showing a step-by-step design**.

1101

0011



**Question 6 (7 Points)**

## Write a Verilog module for modeling the behavior of the sequential circuit represented by the state diagram given below which has a single input X and a single output Z. Assume the state assignment: S0=00, S1=01, and S2=10. Assume the availability of **Synchronous Reset** input that resets the machine to state S0.

## 

1/1

Reset

0/1

0/0

S0

S1

1/0

0/0

S2

1/1

module FSM(output reg Z, input X, Reset, CLK);

parameter S0 = 2'b00, S1=2'b01, S2=2'b10;

reg [1:0] state, next\_state;

always @(posedge CLK)

if (Reset) state <= S0; else state <= next\_state;

always @(state, X) begin

Z=0;

case (state)

S0: if (X) begin Z=1; next\_state=S1; end else next\_state=S0;

S1: if (X) next\_state=S2; else begin Z=1; next\_state=S0; end

S2: if (X) begin Z=1; next\_state=S2; end else next\_state=S1;

default: begin Z='bx; next\_state='bx; end

endcase

end   
endmodule

