Exercise 4

Monday, October 17, 2022 6:09 PM

Problem I, FF, Register, Counters:

a) On the following graph, inputs clk and D are shown. They are inputs to both a D latch and a positive edge D flip-flop. clk goes into the Clock input. Write the output of the D latch as Q_L on the graph. Then write the output of the D flip-flop as Q_{FF} on the graph. Both outputs are initially 0 at the start of the graph, as shown. Do the two outputs differ, and if so, why?

D latch:

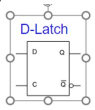
C = 0 keep value

= 1 "transparant"

Q = D

C = 1: D = 1 → Q = 1

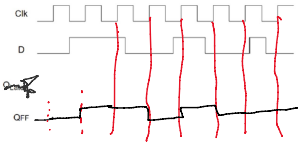
D = 0 → Q = 0



D FF

DFF

↑ D = Q⁺



1 b)

b) Provide the circuit diagram of an asynchronous modulo 8 counter using JK-FFs.

c) Draw the circuit diagram of an asynchronous modulo 10 counter using a D-Fs.

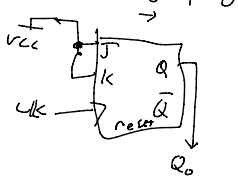
d) Write the VHDL code of a rising edge D-flip flop with synchronous reset.

0 → 8

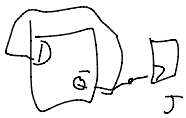
000 → 111

Q₀ Q₁ Q₂

0 1 0

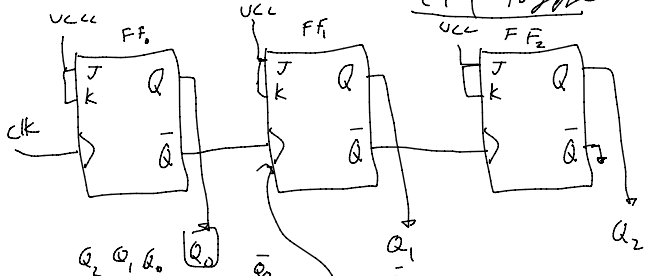


D ff



JK toggle

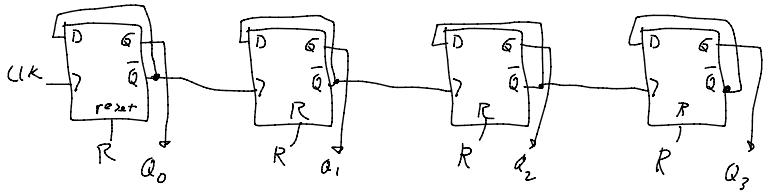
J K	Action
0 0	hold
0 1	reset
1 0	set
1 1	toggle



Q₂	Q₁	Q₀	Q̄₀	Q₁	Q̄₁
0	0	0	1		1
0	0	1	0		1
0	1	0	1		0
0	1	1	0		0
1	0	0			

1c) 0 → 10 DFF's 0 → 16

0000 → 1010 ✗ 16



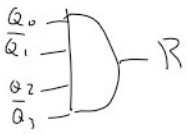
R = 1 reset

R = 0 not reset

Q₀ Q₁ Q₂ Q₃

1 0 1 0 → Reset

R = Q₀ · Q̄₁ · Q₂ · Q̄₃



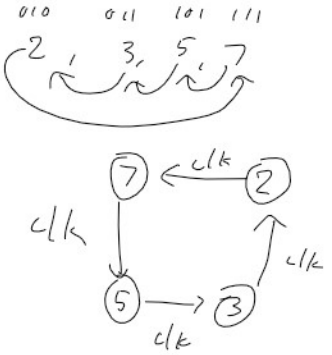
1)d)

```
Library IEEE;
USE IEEE.Std_logic_1164.all;

Entity DFF is
port(
      Q: out std_logic;
      Clk :in std_logic;
      Syn_reset: in std_logic;
      D: in std_logic);
End Dff;

Architecture Behav of DFF is
Begin

Process(Clk)
Begin
If (rising_edge(Clk)) then

If(sync_reset = '1') then
      Q <= '0';
      Else
      Q <= D;
End if;
End if;
End process;
End behav;
```

Problem II, FSM:

a) Design a finite state machine (FSM) for a counter that counts through the 3-bit prime numbers downwards. Assume the counter starts with initial prime value set to 010 as its first 3-bit prime number. You need to provide the state transition table and the state transition diagram. Assume that the state is stored in three D-FFs. Hint: The set of all 3-bit prime numbers includes 2, 3, 5 and7.

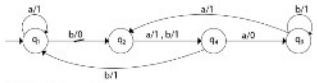
	Q_2 Q_1 Q_0	Q_2^+ Q_1^+ Q_0^+	
0	0 0 0	X X X	X
1	0 0 1	X X X	X
2	0 1 0	1 1 1	7
3	0 1 1	0 1 0	2
4	1 0 0	X X X	X
5	1 0 1	0 1 1	3
6	1 1 0	X X X	X
7	1 1 1	1 0 1	5



mealy: immediate response → imealy

moore: synchronous outputs

b) Convert the following Mealy machine into equivalent Moore machine.

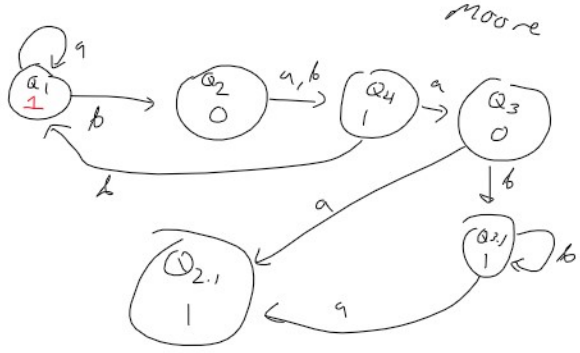


current	next a Q^+	next a z	next b Q^+	next b z
Q_1	Q_1	1	Q_2	0
Q_2	Q_4	1	Q_4	1
Q_3	Q_2	1	Q_3	1
Q_4	Q_3	0	Q_1	1

input / output

a b z = 0 or 1 z = 1

Moore



c) Consider a sequence detector that receives a bit-serial input X and asserts an output Z (i.e. Z = 1) when it detects a binary string 0110 in sequence of 0s and 1s. Use symbolic states with letters such as A, B, etc. You can also assume that 'A' is start state, in which the machine can start out or reset.

i. Draw a Moore Machine state diagram for this sequence detector.

ii. Draw a Mealy Machine state diagram for this sequence detector.

0 1 1 0

X input Z output = 1 when 0110

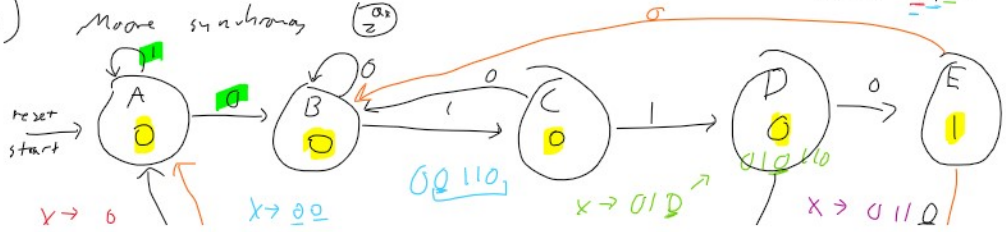
0 0 1 1 0, 1 1 0

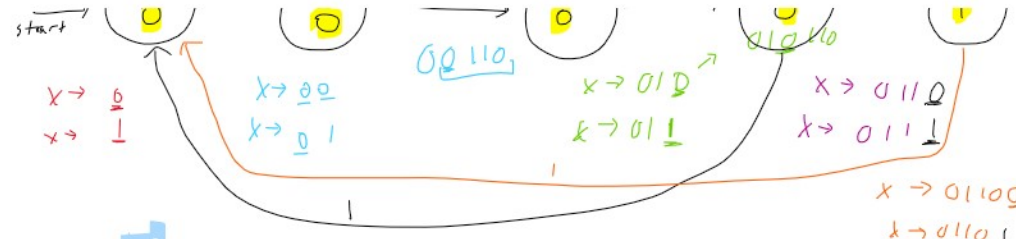
Overlapping

0 0 1 1 0 0 1 1 0

2 C i)

Moore synchronous

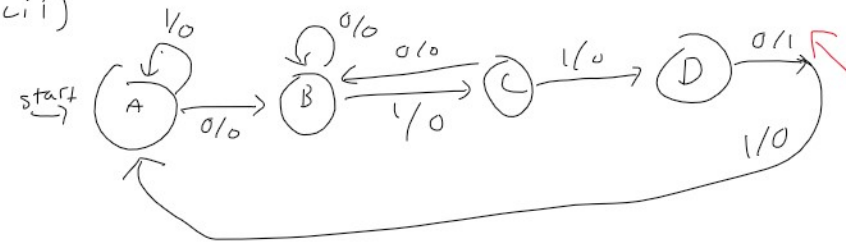




X → 0 0 1 0 1 1 1 0 1 0 0 1

input/output
X / Z

2cii)



X → 0 1 1 0 → A
X → 0 1 1 1 → A

- string ✓
- reset

0 1 1 0 A

Problem II, FSM:

a) Design a finite state machine (FSM) for a counter that counts through the 3-bit prime numbers downwards. Assume the counter starts with initial prime value set to 010 as its first 3-bit prime number. You need to provide the state transition table and the state transition diagram. Assume that the state is stored in three D-FFs. Hint: The set of all 3-bit prime numbers includes 2, 3, 5 and7.

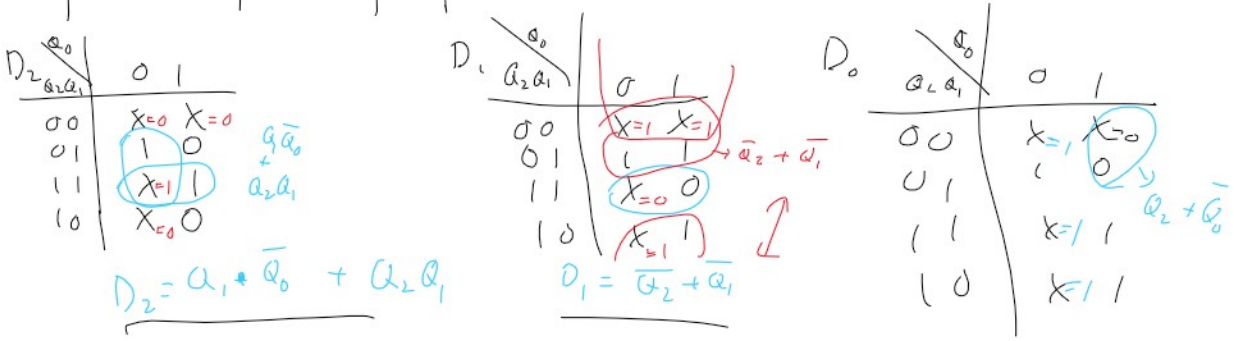
2 → 7 → 5 → 3

3 FF's
DFF

D = Q+

create eq. for $D_{2\to0}$ using $Q_{2\to0}$ as our inputs

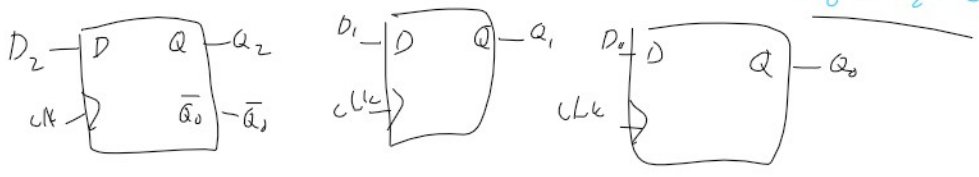
	Q_2 Q_1 Q_0	Q_2^+ Q_1^+ Q_0^+		D_2 D_1 D_0
0	0 0 0	x x x	x	x x x
1	0 0 1	x x x	x	x x x
2	0 1 0	1 1 1	7	1 1 1
3	0 1 1	0 1 0	2	0 1 0
4	1 0 0	x x x	x	x x x
5	1 0 1	0 1 1	3	0 1 1
6	1 1 0	x x x	x	x x x
7	1 1 1	1 0 1	5	1 0 1

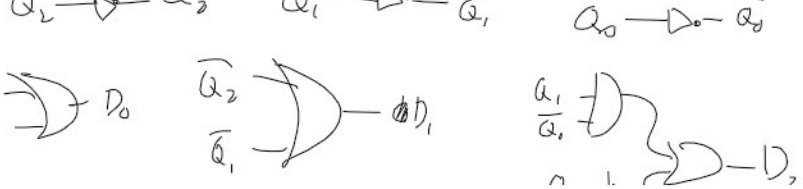


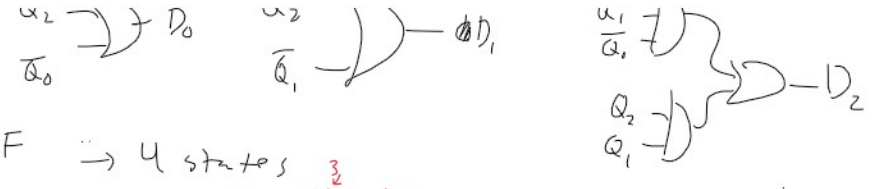
$D_2 = Q_1 \cdot \bar{Q}_0 + Q_2 Q_1$

$D_1 = \bar{Q}_2 + \bar{Q}_1$

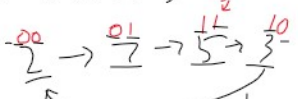
$D_0 = Q_2 + \bar{Q}_0$







2 FF → 4 states



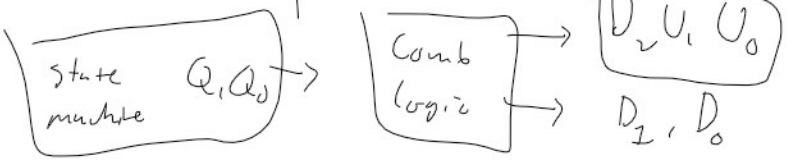
Q_1 Q_0	Q_1^+ Q_0^+	D_1 D_0	U_2 U_1 U_0	
0 0	0 1	0 1	0 1 0	$D_1 = Q_0$
0 1	1 1	1 1	1 1 1	$D_0 = \overline{Q_1}$
1 0	0 0	0 0	0 1 1	
1 1	1 0	1 0	1 0 1	2, 3, 5, 7

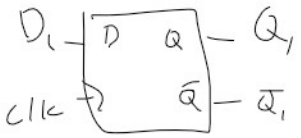
Countdown

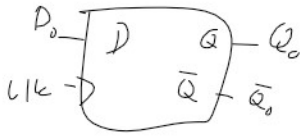
$U_2 = Q_0$

$U_1 = \overline{Q_1} + \overline{Q_0}$

$U_0 = Q_1 + Q_0$







$D_1 <= Q_0$

$D_0 <= \overline{Q_1}$

$U_2 <= Q_0$