LEARNING OUTCOME #3: "an ability to analyze and design sequential logic circuits."

Multiple Choice – select the <u>single</u> most appropriate response for each question.

Note that "none of the above" MAY be a VALID ANSWER.

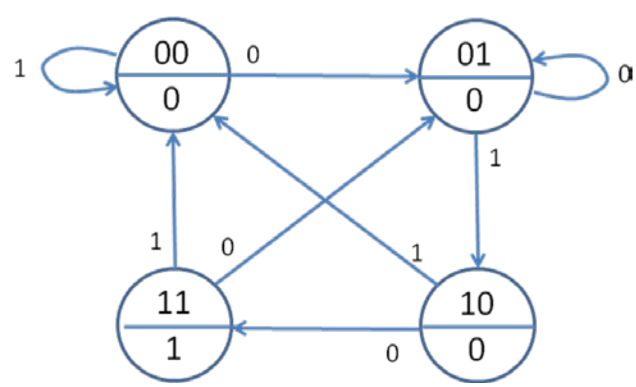
PU Flip-Flop PS-NS Table for questions 1 and 2

P	U	Q	Q*
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

1. A "new" type of flip-flop, the "PU", is described by the given PS-NS table. The characteristic equation for this flip-flop is:
   (A) **Q* = P•Q′ + U′•Q**
   (B) **Q* = P•Q + U′•Q′**
   (C) **Q* = P′•Q + U•Q′**
   (D) **Q* = P•U + Q′**
   (E) none of the above

2. The excitation required to effect a state transition of the PU flip-flop from "0" to "1" is:
   (A) **P=d, U=0**
   (B) **P=d, U=1**
   (C) **P=0, U=d**
   (D) **P=1, U=d**
   (E) none of the above

State Transition Diagram for questions 3 and 4



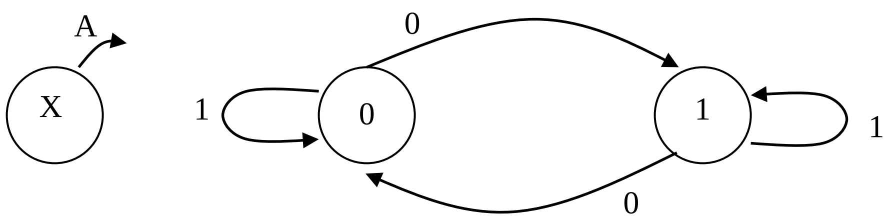
3. Assuming the state machine depicted in the given state transition diagram is initialized to state **00**, the input sequence **1101000100** will cause the following output sequence to be generated:
   (A) **0000100000**
   (B) **0100001000**
   (C) **0010000100**
   (D) **0000100010**
   (E) none of the above

4. The **embedded binary sequence** recognized by this state machine is the pattern:
   (A) **001**
   (B) **010**
   (C) **011**
   (D) **0101**
   (E) none of the above

5. A **D latch** is called a **transparent** because:
   (A) the output "follows" the input when the latch is closed
   (B) the output "follows" the input when the latch is open
   (C) the output "freezes" when the latch is closed
   (D) the output "freezes" when the latch is open
   (E) none of the above

6. **Metastable** behavior of an edge-triggered D flip-flop can be caused by:
   (A) violating its minimum setup time requirement
   (B) violating its minimum hold time requirement
   (C) violating its minimum clock pulse width requirement
   (D) all of the above
   (E) none of the above

7. The next state equation represented by the following state transition diagram is:



   (A) **X* = A′·X′ + A·X**
   (B) **X* = A′·X + A·X′**
   (C) **X* = A + X**
   (D) **X* = A·X**
   (E) none of the above

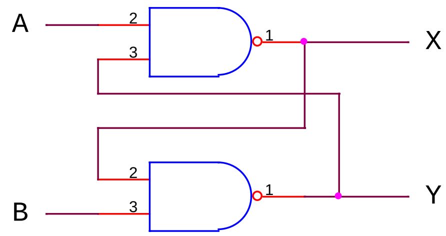
Verilog program for questions 8 and 9:

```
module mystery_seq(CLK, START, Q);
  input wire CLK;
  input wire START;
  output reg[2:0] Q;

always @ (posedge CLK, posedge START)
 begin
   if (START == 1'b1)
     Q <= 3'b111;
   else
     Q <= next_Q;
  end

always @ (Q) begin
   case(Q)
     3'b000: next_Q = 3'b'000;
     3'b001: next_Q = 3'b'010;
     3'b010: next_Q = 3'b'101;
     3'b011: next_Q = 3'b'110;
     3'b100: next_Q = 3'b'001;
     3'b101: next_Q = 3'b'011;
     3'b110: next_Q = 3'b'100;
     3'b111: next_Q = 3'b'110;
   endcase
endmodule
```

8. The initial state (after **START** is asserted) is:
   (A) **000**
   (B) **010**
   (C) **110**
   (D) **111**
   (E) none of the above

9. The number of states in the **periodic sequence** (after **START** is asserted) is:
   (A) **1**
   (B) **2**
   (C) **6**
   (D) **7**
   (E) none of the above

The following circuit applies to questions 10 through 14:



10. If the **input** combination **A=0, B=1** is applied to this circuit, the (steady state) output will be:
    (A) X=0, Y=0
    (B) X=0, Y=1
    (C) X=1, Y=0
    (D) X=1, Y=1
    (E) unpredictable

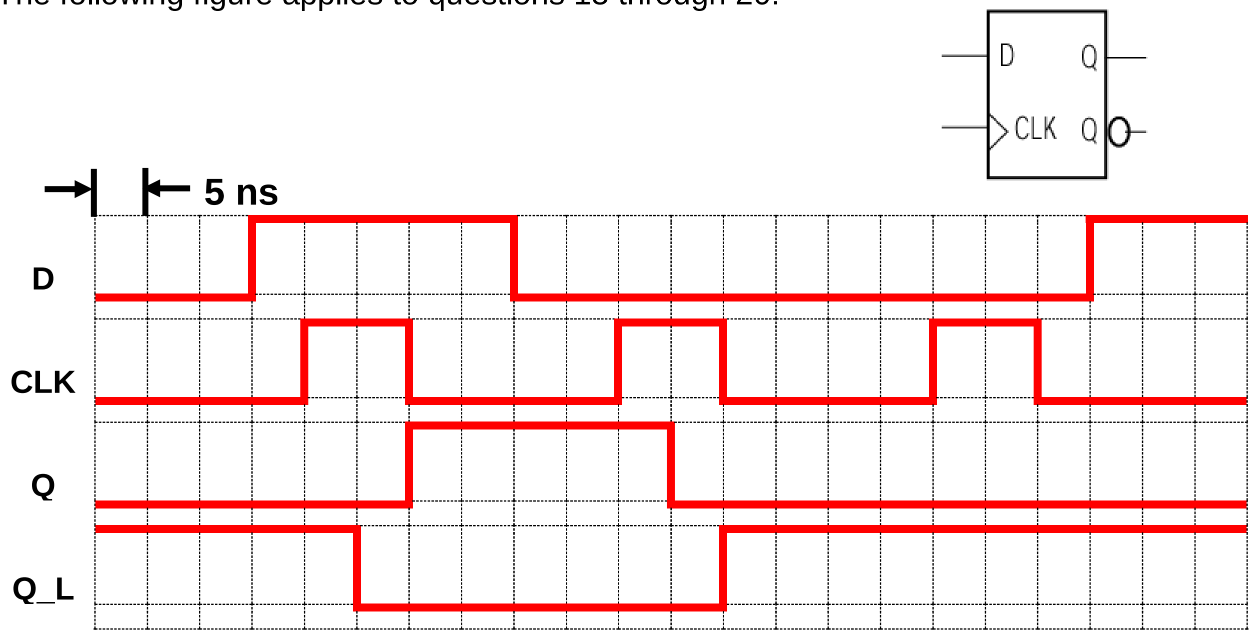
11. If the **input** combination **A=0, B=0** is applied to this circuit, **followed immediately** by the **input** combination **A=1, B=0,** the (steady state) output will be:
    (A) X=0, Y=0
    (B) X=0, Y=1
    (C) X=1, Y=0
    (D) X=1, Y=1
    (E) unpredictable

12. If the **input** combination **A=0, B=0** is applied to this circuit, the (steady state) output will be:
    (A) X=0, Y=0
    (B) X=0, Y=1
    (C) X=1, Y=0
    (D) X=1, Y=1
    (E) unpredictable

13. If the **input** combination **A=0, B=0** is applied to this circuit, **followed immediately** by the **input** combination **A=1, B=1,** the (steady state) output will be:
    (A) X=0, Y=0
    (B) X=0, Y=1
    (C) X=1, Y=0
    (D) X=1, Y=1
    (E) unpredictable

14. If the **propagation delay** of each gate is **10 ns,** the **minimum length of time** that (valid) input combinations need to be asserted **in order to prevent metastable behavior** is**:**
    (A) 10 ns
    (B) 20 ns
    (C) 30 ns
    (D) 40 ns
    (E) none of the above

The following figure applies to questions 15 through 20:



15. The **duty cycle** of the clocking signal is:
    (A) 20%
    (B) 33%
    (C) 40%
    (D) 67%
    (E) none of the above

16. The **nominal setup time** provided for the D flip-flop, based on the excitation signals (D and CLK) depicted in the timing chart, is:

    (A) 5 ns (B) 10 ns (C) 15 ns (D) 20 ns (E) none of the above

17. The **nominal hold time** provided for the D flip-flop, based on the excitation signals (D and CLK) depicted in the timing chart, is:

    (A) 5 ns (B) 10 ns (C) 15 ns (D) 20 ns (E) none of the above

18. The **nominal clock pulse width** provided for the D flip-flop, based on the excitation signals (D and CLK) depicted in the timing chart, is:

    (A) 5 ns (B) 10 ns (C) 15 ns (D) 20 ns (E) none of the above

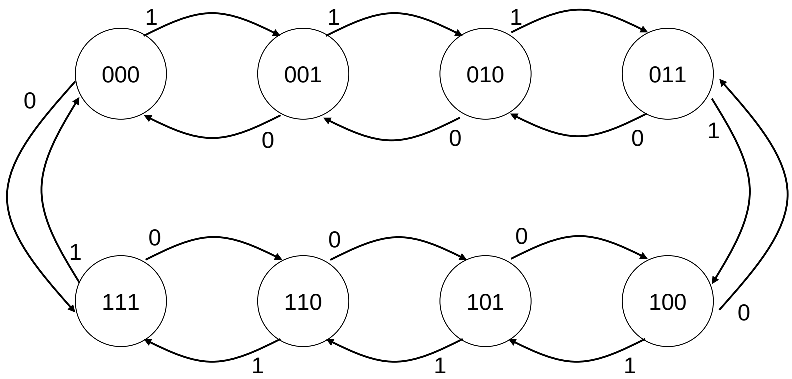
19. The $\mathbf{t}_{\mathbf{PLH(C \to Q)}}$ of the D flip-flop is:

    (A) 5 ns (B) 10 ns (C) 15 ns (D) 20 ns (E) none of the above

20. The $\mathbf{t}_{\mathbf{PHL(C \to Q)}}$ of the D flip-flop is:

    (A) 5 ns (B) 10 ns (C) 15 ns (D) 20 ns (E) none of the above

21. The following Verilog program implements the state transition diagram below:



(A)

```
/* Program (A) */
module CQ(CLK, M, Q);
  input wire CLK, M;
  output reg [2:0] Q;
  reg [2:0] next_Q;
  always @ (posedge CLK) begin
    Q <= next_Q;
  end
  always @ (Q, M) begin
   next_Q[0] = ~Q[0];
   next_Q[1] = ~Q[1] ^ (~M&~Q[0] | M&Q[0]);
   next_Q[2] = ~Q[2] ^ (~M&~Q[1]&~Q[0] | M& Q[1]& Q[0]);
  end
endmodule
```

(B)

```
/* Program (B) */
module CQ(CLK, M, Q);
  input wire CLK, M;
  output reg [2:0] Q;
  reg [2:0] next_Q;
  always @ (posedge CLK) begin
    Q <= next_Q;
  end
  always @ (Q, M) begin
   next_Q[0] = ~Q[0];
   next_Q[1] =  Q[1] ^ (~M&Q[0] | M&~Q[0]);
   next_Q[2] =  Q[2] ^ (~M& Q[1]& Q[0] | M&~Q[1]&~Q[0]);
  end
endmodule
```

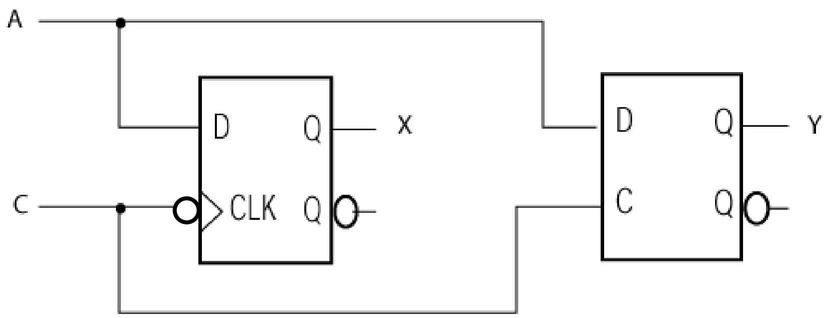
(C)

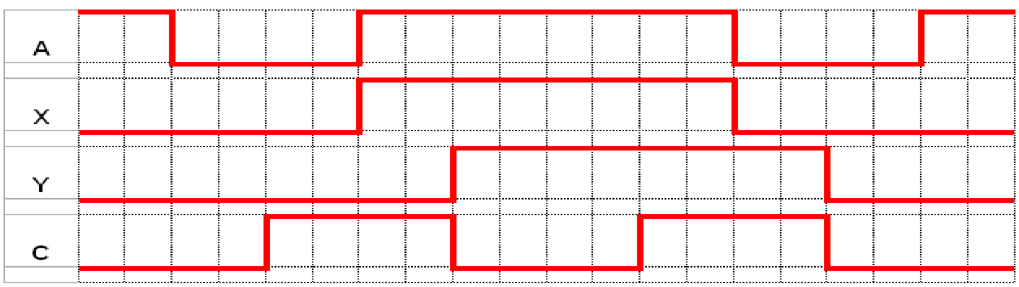
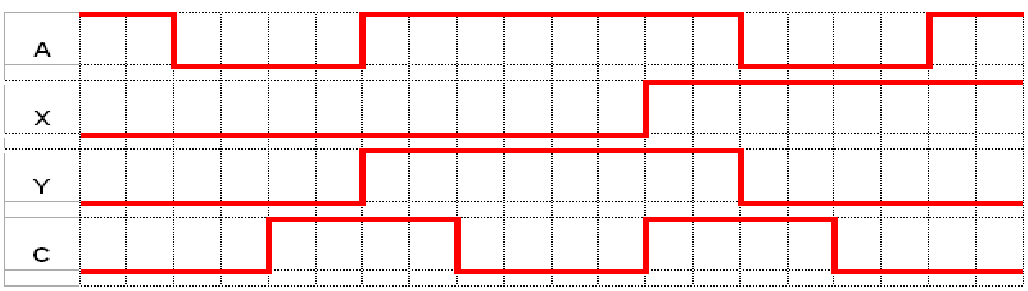
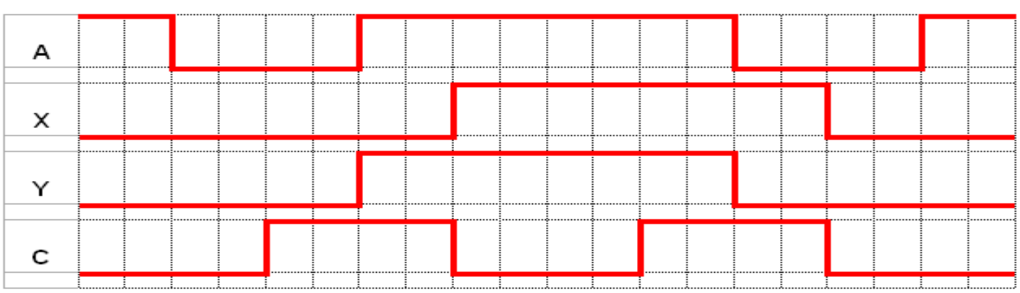
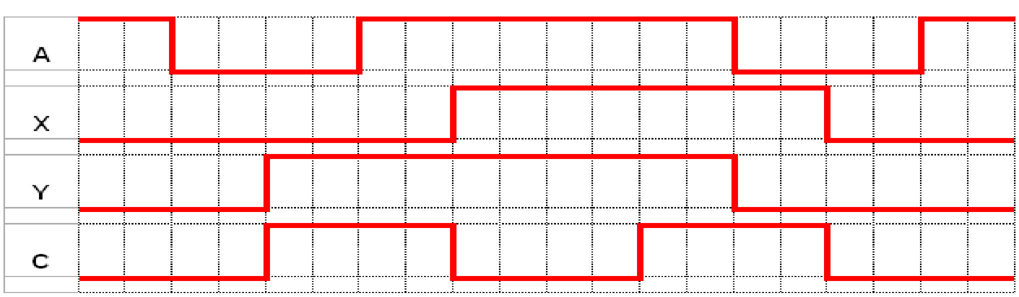
```
/* Program (C) */
module CQ(CLK, M, Q);
  input wire CLK, M;
  output reg [2:0] Q;
  reg [2:0] next_Q;
  always @ (posedge CLK) begin
    Q <= next_Q;
  end
  always @ (Q, M) begin
   next_Q[0] = ~Q[0];
   next_Q[1] =  Q[1] ^ (~M&~Q[0] | M& Q[0]);
   next_Q[2] =  Q[2] ^ (~M&~Q[1]&~Q[0] | M& Q[1]& Q[0])
  end
endmodule
```

(D) all of the above

(E) none of the above

22. The following timing diagram depicts the behavior of the circuit shown below:



(A)	
(B)	
(C)	
(D)	
(E)	none of the above

23. As a contestant on the soon-to-be-cancelled TV series ***Digital Moment of Truth*, **you have been asked to identify which of the following statements concerning **state machine models** is **<u>true</u>:**
    (A) Mealy and Moore models that represent equivalent state machines will ***always*** have the **same** number of states
    (B) Mealy and Moore models that represent equivalent state machines will ***always*** have a **different** number of states
    (C) **any Mealy model** can be transformed into **an equivalent Moore model,** and *vice-versa*
    (D) Mealy and Moore models that represent equivalent state machines, when realized, will exhibit the **same observable behavior** (i.e., if placed in a "black box", their **observable behavior** would be **indistinguishable**)
    (E) none of the above

24. As a contestant on the TV series ***Are You Smarter Than a Website Contractor?***, you have been asked to explain why a **"D" latch** is called **transparent.** Hoping to forgo an admission before a national television audience to the contrary, you calmly answer that **a "D" latch is called transparent because its output:**
    (A) is equal to its input when the latch enable is high-impedance
    (B) is equal to its input when the latch enable is asserted
    (C) is equal to its input when the latch enable is negated
    (D) changes state as soon as the latch is clocked
    (E) none of the above

25. The next topic over which you've been asked to "Digitally Digress" with the stars of ***Dual Dynasty*** is the phenomenon of metastability. You confidently explain that **the <u>next state</u> of an edge-triggered D flip-flop will most likely be <u>random</u> if:**
    (A) its minimum setup time requirement is not met
    (B) its minimum hold time requirement is not met
    (C) its minimum clock pulse width requirement is not met
    (D) all of the above
    (E) none of the above

26. As a contestant on the hit TV series ***Digital Survivor – Flips vs. Flops,*** you have been asked to implement a **<u>negative</u> edge-triggered D flip-flop** using **<u>only</u> 2-input NAND** gates. The **<u>minimum</u>** number of gates you will need to complete this task is:
    (A) 9 (B) 10 (C) 11 (D) 12 (E) none of these

27. Your next task on ***Digital Survivor*** is to build a circuit that **divides** the frequency of a clocking signal **by two.** Provided you have successfully completed Problem 26, above, and have a working negative edge-triggered D flip-flop, the number of **<u>additional</u> 2-input NAND gates** you will need to complete this task is:
    (A) 0 (B) 1 (C) 2 (D) 3 (E) none of these

28. Your final task on ***Digital Survivor*** is to implement a finite state machine that has **212 states** with as few flip-flops as possible. To **reduce** the **number of flip-flops** required in this design **by one** (using either "obvious" or "formal" state minimization procedures)**, you would have to identify and eliminate ____ redundant state(s).**
    (A) 1 (B) 2 (C) 84 (D) 128 (E) none of these

The following Verilog program applies to questions 29 and 30:

```
/* Multi-Color LED Light Machine */
module mcleds(CLK, M, R, G, Y, B);
  input wire CLK;
  input wire M;
  output wire R, G, B, Y;
  reg [1:0] Q, next_Q;
  reg [5:0] nQRGYB;

  always @ (posedge CLK) begin
    Q <= next_Q;
  end

  assign next_Q    = nQRGYB[5:4]
  assign {R,G,Y,B} = nQRGYB[3:0];

  always @ (Q, M) begin
    case ({Q,M})
      3'b000: nQRGYB = {2'b10,4'b1000};
      3'b001: nQRGYB = {2'b11,4'b1000};
      3'b010: nQRGYB = {2'b11,4'b0010};
      3'b011: nQRGYB = {2'b00,4'b1111};
      3'b100: nQRGYB = {2'b01,4'b0100};
      3'b101: nQRGYB = {2'b01,4'b1110};
      3'b110: nQRGYB = {2'b00,4'b0001};
      3'b111: nQRGYB = {2'b10,4'b1100};
    endcase
  end
endmodule
```

29. When **M=0,** the (repeating) colored LED sequence produced will be:
    - (A) **R**→**G**→**Y**→**B**→…
    - (B) **R**→**Y**→**G**→**B**→…
    - (C) **B**→**Y**→**G**→**R**→…
    - (D) **B**→**G**→**Y**→**R**→…
    - (E) none of the above

30. When **M=1,** the (repeating) colored LED sequence produced will be:
    - (A) **R**→**RGYB**→**RGY**→**RG**→…
    - (B) **R**→**RG**→**RGY**→**RGYB**→…
    - (C) **RGYB**→**RGY**→**RG**→**R**→…
    - (D) **R**→**RGY**→**RG**→**RGYB**→…
    - (E) none of the above

Answer key:
1-B, 2-A, 3-D, 4-B, 5-B, 6-D, 7-A, 8-D, 9-C, 10-C, 11-B, 12-D, 13-E, 14-B, 15-B, 16-A,
17-C, 18-B, 19-B, 20-A, 21-C, 22-C, 23-C, 24-B, 25-D, 26-B, 27-A, 28-C, 29-A, 30-B

Answer key:
1-B, 2-A, 3-D, 4-B, 5-B, 6-D, 7-A, 8-D, 9-C, 10-C, 11-B, 12-D, 13-E, 14-B, 15-B, 16-A, 17-C, 18-B, 19-B, 20-A, 21-C, 22-C, 23-C, 24-B, 25-D, 26-B, 27-A, 28-C, 29-A, 30-B