Points missed: _____ Student's Name: _________________________________

Total score: _____/100 points

East Tennessee State University
Department of Computer and Information Sciences
CSCI 2150 (Tarnoff) – Computer Organization
TEST 2 for Spring Semester, 2009

Read this before starting!

- The total possible score for this test is 100 points.
- This test is ***closed book and closed notes.***
- ***Please turn off all cell phones & pagers during the test.***
- **All** answers **must** be placed in space provided. Failure to do so may result in loss of points.
- **1 point** will be deducted per answer for missing or incorrect units when required. **No** assumptions will be made for hexadecimal versus decimal, so you should always include the base in your answer.
- If you perform work on the back of a page in this test, indicate that you have done so in case the need arises for partial credit to be determined.
- ***Calculators are not allowed.*** Use the tables below for any conversions you may need. Leaving numeric equations is fine too.

Binary	Hex
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binary	Hex
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

Power of 2	Equals
2^3	8
2^4	16
2^5	32
2^6	64
2^7	128
2^8	256
2^9	512
2^{10}	1K
2^{20}	1M
2^{30}	1G

"Fine print"

Academic Misconduct:

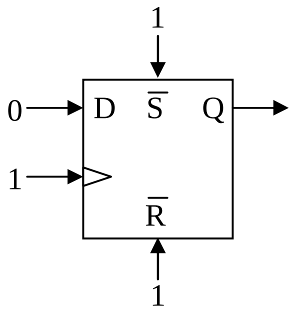
Section 5.7 "Academic Misconduct" of the East Tennessee State University Faculty Handbook, June 1, 2001:

"Academic misconduct will be subject to disciplinary action. Any act of dishonesty in academic work constitutes academic misconduct. This includes plagiarism, the changing of falsifying of any academic documents or materials, cheating, and the giving or receiving of unauthorized aid in tests, examinations, or other assigned school work. Penalties for academic misconduct will vary with the seriousness of the offense and may include, but are not limited to: a grade of 'F' on the work in question, a grade of 'F' of the course, reprimand, probation, suspension, and expulsion. For a second academic offense the penalty is permanent expulsion."

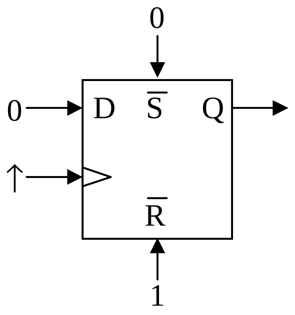
Short answers – 2 points each unless otherwise noted

For each of the following ***three*** circuits, identify the value of the output Q from the following choices. Consider the D-latch a ***rising edge triggered latch***.

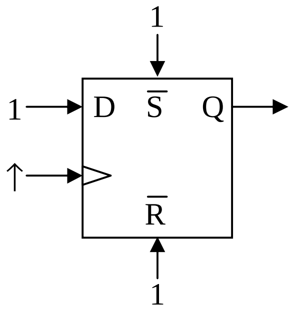
a.) 1 b.) 0 c.) Q_0 (stored value of Q) d.) undefined/illegal e.) can't tell



Answer: c
First, neither the R-bar or S-bar lines are 0. Therefore, we can look at D and the clock to see what Q might be. Since the D-latch is rising edge triggered, and the input the clock is a stable 1, i.e., it is not transitioning, then it doesn't matter that D equals 0 - it has no effect on Q and Q is simply outputting the stored value of Q, i.e., Q_0.



Answer: a
Although the clock is transitioning in this case, it doesn't matter because S-bar is low. This overrides the inputs clock and D. When S-bar goes low, the stored value (Q) is forced to equal 1.



Answer: a
This is the basic "store data" case. The clock is transitioning from 0 to 1 (a rising edge), and S-bar and R-bar are both high (idle). This means that the 1 at the input D is being stored to Q.

4. Assume you have a truth table representing a circuit with four inputs, A, B, C, and D. How many rows in this truth table would have ones as a result of the expression $A \cdot \overline{B} \cdot \overline{D}$?

   [✓ a.)] 2 b.) 4 c.) 6 d.) 8 e.) 12 f.) Cannot be determined

   The output from this product is equal to 1 when A=1, B=0, and D=0. C can be either 1 or 0. Therefore, the truth table will have two rows with ones in it from this product, one for A=1, B=0, C=0, and D=0 and one for A=1, B=0, C=1, and D=0.

5. How many cells would a three-input Karnaugh Map have?

   a.) 2 b.) 4 c.) 6 [✓ d.)] 8 e.) 12 f.) 16 g.) 32

   Like rows in a truth table, Karnaugh maps must have a cell for every pattern of 1's and 0's for its inputs. Since a 3-input circuit has 2^3 = 8 patterns of 1's and 0's, then the answer is 8 cells.

6. [✓ True] or False: The expression $(A+\overline{B})\cdot(\overline{A}+\overline{B}+C)$ is in proper product-of-sums format.

Nothing is wrong with this POS expression. Basically what you're looking for when you examine whether a Boolean expression is in valid POS form is whether or not it can be realized in the standard two-level logic format of inputs (either inverted or non-inverted) going directly to OR gates and all of the outputs of the OR gates going into the inputs of a single AND gate (no inverters between the ORs and the ANDs).

7. Create the sum that represents the truth table shown to the right. (3 points)

A	B	C	X
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Remember that a ***sum*** creates a single zero in the truth table and a ***product*** creates a single 1 in the truth table. Since the truth table to the right contains a single zero, it is best realized using a single sum. The goal is to figure out how to invert the appropriate inputs of the sum (A + B + C) in order to force a zero at the row where A=0, B=1, and C=1. The sum generates a zero ONLY when you have 0 + 0 + 0. Therefore, B and C need to be inverted. This gives us the answer:

$$\boxed{A+\overline{B}+\overline{C}}$$

8. Create the product that represents the truth table shown to the right.

A	B	X
0	0	0
0	1	0
1	0	1
1	1	0

Once again, a ***sum*** creates a single zero in the truth table and a ***product*** creates a single 1 in the truth table. Since the truth table to the right contains a single one, it is best realized using a single product, i.e., A·B. Which inputs get inverted? Well, the product generates a one ONLY when you have 1·1. Since the row we want to have a 1 output in has A=1 and B=0, then we need to invert B. This gives us the answer:

$$\boxed{A\cdot\overline{B}}$$

9. In a 4-variable Karnaugh map, how many input variables (A, B, C, and/or D) does a single product have if its corresponding rectangle of 1's contains 2 cells?

a.) 1 b.) 2 c.) 3 d.) 4 e.) Cannot be determined

The simplest way to answer this question is to create a 4-variable/input Karnaugh map that has a rectangle with two 1's, then figure out what the product is. The number of inputs in the product will give us our answer.

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	0	1	1
10	0	0	0	0

A	B	C	D
1	1	1	1
1	1	1	0

} A·B·C

This rectangle has three inputs, A, B, and C, that remain constant. Therefore, the resulting product consists of A, B, and C. You'll find that regardless of how the 2-cell rectangle is arranged, the resulting product will always have 3 inputs.

Another way of doing this is to realize that one variable drops out every time the size of a rectangle is doubled. For a two-cell rectangle, the rectangle doubled only once. This means only 1 variable drops out of product from the original four.

10. An falling-edge latch copies data from the D input to the Q output when the clock is:

a.) a logic 0 [✓ b.)] changing from a 1 to a 0 c.) a logic 1 d.) changing from a 0 to a 1

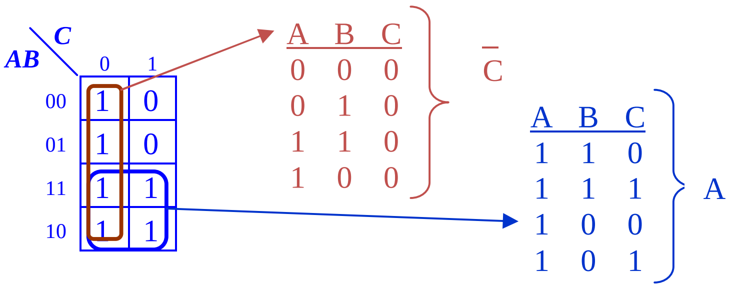
11. Which of the following expressions produces the truth table to the right?

a.) $A\cdot\overline{B}+\overline{C}$ b.) $A+\overline{C}$ c.) $A\cdot B+\overline{C}$

d.) $\overline{B}+A\cdot C$ e.) $A+\overline{B}\cdot C$ f.) $A+B\cdot\overline{C}$

A	B	C	X
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

There are a number of ways to solve this problem. The easiest way to do this I think is to put together the Karnaugh map for the truth table.



OR-ing these two products together gives us $A+\overline{C}$ which is answer b.

We can also put together the truth table for each of the options a through f to see if any of them match. For example, $A\cdot\overline{B}+\overline{C}$ is equal to 1 when $A\cdot\overline{B}$ equals 1 (A=1 and B=0) or when $\overline{C}$ equals 1 (C=0). This gives us a truth table with a 1 in the following rows:

A=1, B=0, C=0 A=1, B=0, C=1

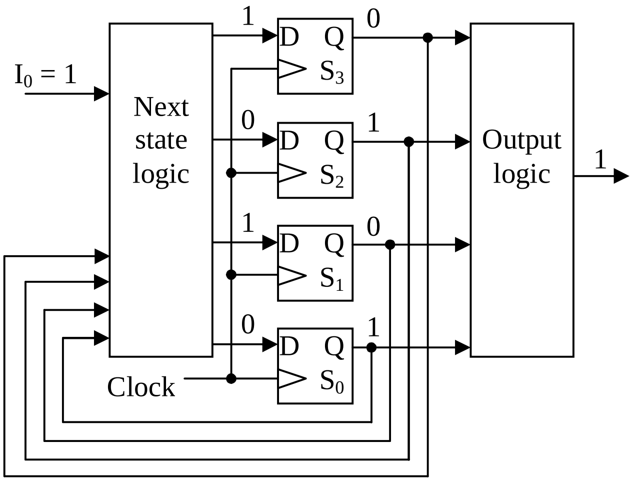
A=0, B=0, C=0 A=0, B=1, C=0 A=1, B=0, C=0 A=1, B=1, C=0

The first two are for A=1 and B=0 while the last 4 represent when C=0. All other rows will be 0.

If we do this for all options a through f, we get the following truth table(s). The only column that matches the original truth table exactly is column b. Therefore, the answer is b.

A	B	C	a	b	c	d	e	f
0	0	0	1	1	1	1	0	0
0	0	1	0	0	0	1	1	0
0	1	0	1	1	1	0	0	1
0	1	1	0	0	0	0	0	0
1	0	0	1	1	1	1	1	1
1	0	1	1	1	0	1	1	1
1	1	0	1	1	1	0	1	1
1	1	1	0	1	1	1	1	1

The next six problems use the state machine circuit to the right. Assume that the states are numbered so that bit S_3 is the most significant bit and bit S_0 is the least significant bit.



12. What is the maximum number of states that this system can handle?

Since there are four latches, then the internal memory of this state machine can remember $2^4 = 16$ states: 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, and 1111. Therefore, the answer is 16.

13. What is the current state of this system? Keep your answer in binary.

The current state is the value stored in the latches and found at the Q outputs. In the case of the diagram above that is 0-1-0-1 with S_3 being the most significant bit.

14. If the clock were to pulse right now, what would the next state be? Keep your answer in binary.

The next state is the value present at the D inputs, i.e., the value that would be stored if we had a clock pulse occur. In the case of the diagram above that is 1-0-1-0 with S_3 being the most significant bit.

S_3	S_2	S_1	S_0	X
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

15. The truth table to the right represents the output logic truth table for the above state machine. Circle the row that identifies the current output condition of the system, i.e., which row is represented by the current state of the logic in the diagram above?

The output is determined by the current state, i.e., 0-1-0-1. Therefore, the sixth row identifies the output, which equals 1 just like in the figure.

16. If the clock were to pulse right now, what would the new output be? Use the truth table from the previous problem to answer the question.

[✓ a.)] 0 b.) 1 c.) Not enough information given

The next state is 1-0-1-0. From the truth table we see that the output for state 1-0-1-0 is 0.

17. How many rows would the next state logic truth table have for this circuit?

a.) $2^2 = 4$ b.) $2^3 = 8$ c.) $2^4 = 16$ [✓ d.)] $2^5 = 32$ e.) $2^6 = 64$

The next state logic is the logic to the right of the latches that is used to determine the next state of the system. In the case of this circuit, the next state logic truth table depends on the current state (derived from S_3, S_2, S_1, and S_0) and the system's input (I_0). This gives us 5 inputs. Since there are $2^5 = 32$ combinations of 1's and 0's for 5 inputs, the answer is d. (Counting the arrows into the block labeled “Next state logic” would have also shown that there are 5 inputs to this logic.)

18. True or [✓ False:] Re-numbering the states of a state machine has no effect on the ***output logic*** for the digital hardware implementation.

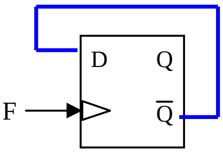
The numbering of the states directly affects the how both the output truth table and next state truth table are assigned their different outputs. Therefore, renumbering changes the logic that is derived from them. This makes our answer **False**.

19. How many latches will a state machine with 24 states require?

a.) 3 b.) 4 c.) 5 d.) 6 e.) 7 f.) 8 g.) 9

With 24 states, they would be numbered 0, 1, … , 22, and 23. Since $23_{10} = 10111_2$, a binary value with 5 bits, we will need 5 bits to represent state 23. **Therefore, the answer is c.** Another way of looking at it is to see how many states it is possible to represent with n bits, and to figure out what value of 2^n is greater than or equal to 24 different values. $2^4 = 16$ which is not enough, $2^5=32$ which is still not enough, but $2^6=64$ which is exactly what is needed. Therefore, 6 bits will do the trick.

20. Make the connections to the latch in the figure to the right that makes a divide-by-two circuit, i.e., divides the frequency F in half at the output Q.

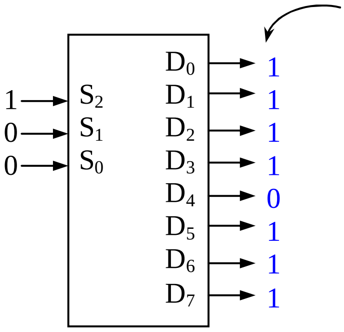


21. Two examples were given in class representing reasons why you would want to cascade multiple divide-by-two circuits. Name one of them.

Cascaded divide-by-two circuits successively divide the incoming frequency by 2. In other words, if four divide-by-two circuits are cascaded, each dividing the frequency of the previous one in half, then the output of the first one is the input frequency divided by 2, the output of the second is the input frequency divided by 4, the output of the third is the input frequency divided by 8, and the output of the fourth is the input frequency divided by 16. There are two reasons to do this. First, it allows us to slow the frequency down even further than just dividing it by 2. Second, the values stored in the latches are a count of the number of clock pulses of the original frequency. Therefore, these cascaded divide-by-two circuits could also be used as a timer or clock pulse counter.

22. For the ***active-low*** output decoder shown to the right, fill in the values for all of the outputs D_0 through D_7. Assume S_2 is most significant bit of the inputs. (3 points)

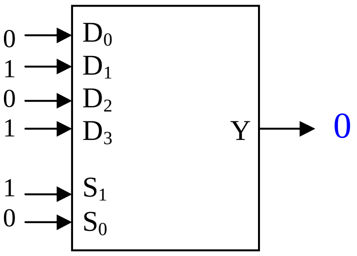
Remember that the bits S_2, S_1, and S_0 make up a digital selector with S_2 being the most significant bit. The digital value of these three bits is 100_2 which equals 4_{10}. That means that output D_4 is the selected output. Since it's active low, there will be a zero output on D_6 and 1's everywhere else.



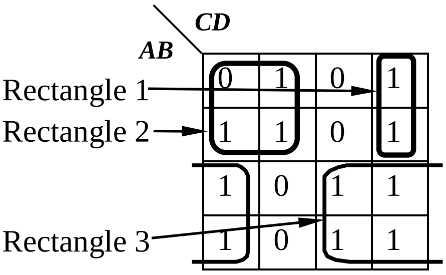
Fill in the ones and zeros for all of these outputs.

23. For the multiplexer/selector shown to the right, what is the output Y?

The input that is being routed to the output is “selected” by the selector inputs S_1 and S_0 with S_1 being the most significant bit. The digital value of these two bits is 10_2 which equals a decimal 2. Therefore, the input D_2 is going to be routed to the output Y making Y have an output of 0.



24. For the Karnaugh map to the right, identify three ***problems*** with how the rectangles have been made. Note that not all of the problems may be with a specific rectangle, but if there is a problem with a rectangle, be sure to identify it using the names given. (2 points each)



0	1	0	1
1	1	0	1
1	0	1	1
1	0	1	1

Problem 1: Rectangle one is not large enough. It can be extended to overlap the two ones below it resulting in a simpler product.
Problem 2: Rectangle two contains a zero – a big no-no.

Problem 2: Rectangle 3 encircles 6 cells which is not a power of two – another big no-no.

Medium answers – 4 points each

25. Complete the truth table to the right with the values for the following sum-of-products expression:

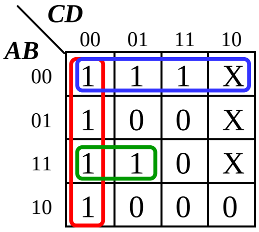
$$\overline{A}\cdot B + A\cdot\overline{C}$$

A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Remember that each product generates a 1 when all of its inputs are one, e.g., $1\cdot1\cdot1 = 1$. This means that if we can figure out where each product equals 1, we know where the 1's are in the truth table. The remaining positions are filled with zeros.

The first product, $\overline{A}\cdot B$, equals one when A=0 and B=1. Therefore, the two rows where A=0 and B=1 should be set to 1. They are the rows A=0, B=1, and C=0 and A=0, B=1, and C=1. The second product, $A\cdot\overline{C}$, equals 1 when A=1 and when C=0. This happens in the rows A=1, B=0, and C=0 and A=1, B=1, and C=0. This gives us the pattern of 1's and 0's you see filled in above.

26. In the Karnaugh map to the right, draw the best pattern of rectangles you can. ***Do not derive the SOP expression.***

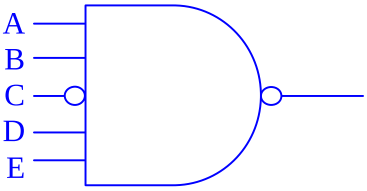


	00	01	11	10
00	1	1	1	X
01	1	0	0	X
11	1	1	0	X
10	1	0	0	0

Remember to only include X's in rectangles if they make the rectangle bigger. Do not include an X if it adds an additional rectangle.

The top rectangle includes the X because if we didn't, we'd need to have two smaller 2-cell rectangles to cover the three ones on the top row. The other rectangles with X's in them are not included in any rectangles because to do so would have required us to add new rectangles.

27. In the space to the right, draw the decoding logic circuit with an active-low output that identifies when A = 1, B = 1, C = 0, D = 1, and E = 1.



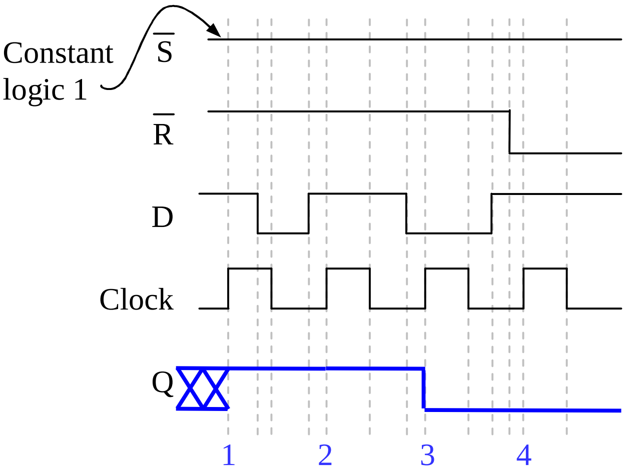
Remember than a decoder is always an AND function, i.e., the output goes "active" for exactly one pattern of 1's and 0's at the inputs. An active low decoder inverts the output making it a NAND. Other than that, simply invert the inputs that are supposed to equal 0.

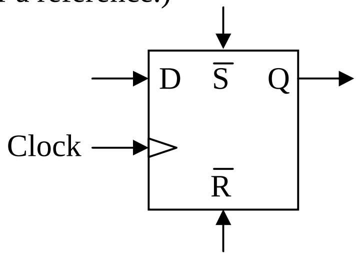
28. Create a Karnaugh map from the truth table below. ***Do not worry about making the rectangles.***

A	B	C	X
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	X

AB \ ***C***	0	1
00	1	1
01	0	0
11	1	X
10	1	0

29. Show the *D* latch output waveform *Q* based on the inputs *D*, $\overline{S}$, $\overline{R}$, and clock indicated in the graph to the right. Assume the latch captures on the rising edge. (The figure below is just for a reference.)





Note that since S-bar is 1 for the entire time, so it has no effect on the stored value. R-bar, however, is 0 for the last little bit of the time shown on the graph. As soon as R-bar goes to zero, the output Q is forced to 0.

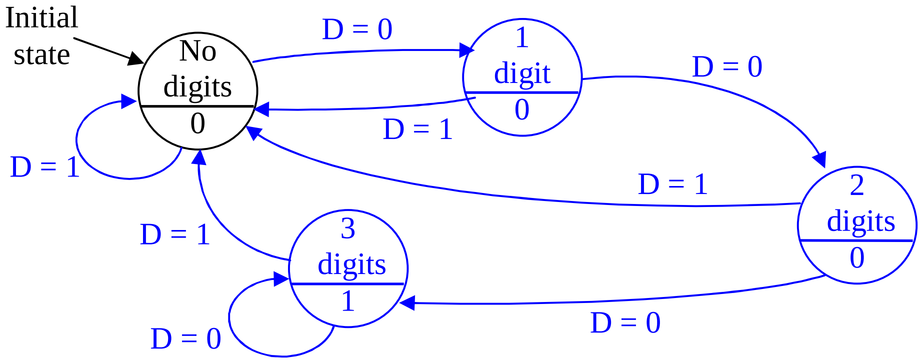
For the remaining time shown on the graph, the output Q is driven by when the clock stores values of D to Q. This happens exactly at the moments when the clock goes from 0 to 1. By the way, before the first clock pulse, we do not know what Q is, so leave it blank or add the cross hatches. When the first clock pulse at '1' occurs, the value 1 that is on D is stored to Q. When the second clock pulse at '2' occurs, the value is once again 1 on D, so another 1 is stored to Q. At clock pulse '3', a 0 is on D and it is stored to Q. The fourth clock pulse at '4' is ignored because R-bar equals zero overriding both clock and D.

Longer answers – Points vary per problem

30. Make the state diagram that will output a ‘1’ when the sequence ‘000’ is detected in a serial stream of bits. For example, if the following binary stream is received:

```
1 0 1 0 1 1 1 0 1 0 0 0 0 1 1 0 0 0 1 1 0 1 0 0 0 1 1 0
```

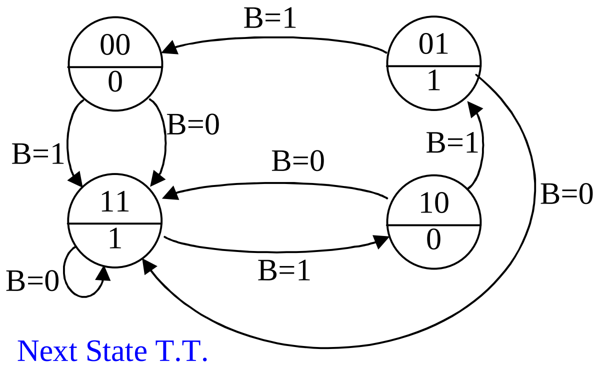
then 1’s will be output at these points. At all other times, the system will output zeros. Label the input D. (7 points)



The basic operation of this state machine is this. Any time a one is received, we know it can't be part of the pattern we're looking for since no ones exist in the pattern. Therefore, all inputs equal to 1 send us back to the "no digits received" state. Inputs of zero simply move us to the next number of bits received. For example, if we have no digits received, we move to the 1 digit received. If we have one digit received, we move to the three digits received. If we have two digits received, we move to the three digits received. At the three digits received state, we output a 1 to say that we've found the pattern. As long as we're in the three digits received state and continue to receive zeros, we stay in that state indicated that the last three bits have all been zeros.

31. Create the next state truth table and the output truth table for the state diagram to the right. The states have already been numbered. Use the variable names S_1 and S_0 to represent the most significant and least significant bits respectively of the binary number identifying the state. Label the output 'X'. (7 points)

Remember that the next state truth table represents a one-to-one mapping of each of the transitions (arrows) from one state to the next state. This means it depends on the current state defined by S_0 and S_1 and the input, B. The output truth table is a one-to-one mapping of the current state to the output value contained in the circle, i.e., the output is the value below the line while the value of S_1 and S_0 is shown above the line.



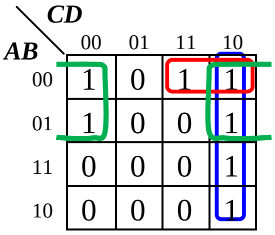
Next State T.T.

S_1	S_0	P	S_1'	S_0'
0	0	0	1	1
0	0	1	1	1
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	1	1
1	1	1	1	0

Output T.T.

S_1	S_0	X
0	0	0
0	1	1
1	0	0
1	1	1

32. Derive the minimum SOP expression from the Karnaugh map below. (6 points)



AB \ CD	00	01	11	10
00	1	0	1	1
01	1	0	0	1
11	0	0	0	1
10	0	0	0	1

Red rectangle

A	B	C	D
0	0	1	1
0	0	1	0

D drops out. Since A=0 and B=0, they are inverted. C=1, so it isn't inverted.

$\overline{A}\cdot\overline{B}\cdot C$

Blue rectangle

A	B	C	D
0	0	1	0
0	1	1	0
1	1	1	0
1	0	1	0

A and B drop out. Since C=1, it isn't inverted. Since D=0, it is inverted.

$C\cdot\overline{D}$

Green rectangle

A	B	C	D
0	0	0	0
0	1	0	0
0	0	1	0
0	1	1	0

B and C drop out. Since both A=0 and D=0, they are inverted in the product.

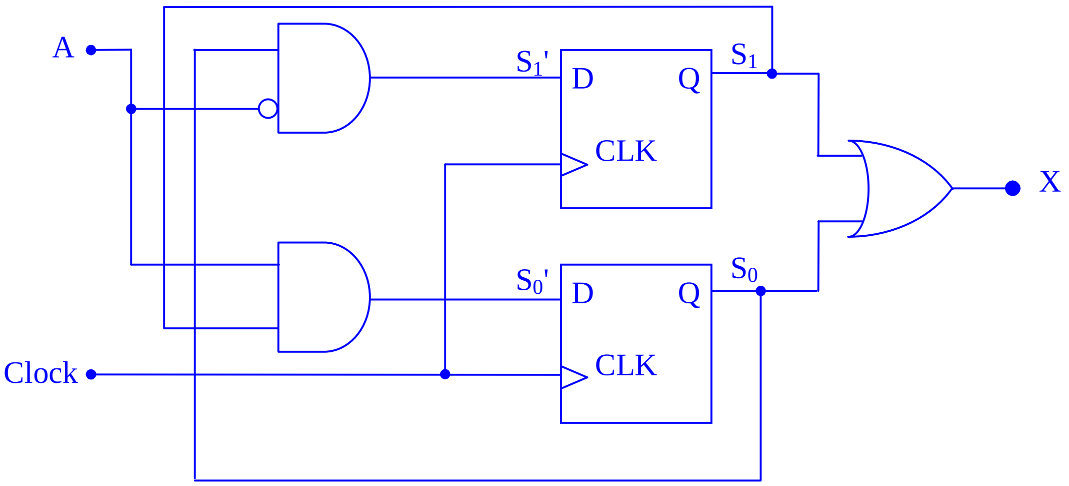
$\overline{A}\cdot\overline{D}$

The final answer is:

$\overline{A}\cdot\overline{B}\cdot C + C\cdot\overline{D} + \overline{A}\cdot\overline{D}$

33. The three Boolean expressions below represent the ***next state bits (S_0' and S_1')*** and the ***output bit X*** based on the ***current state (S_0 and S_1)*** and the ***input A***. Draw the logic circuit for the state machine including the latches and output circuitry. ***Be sure to label the latch inputs and other signals.*** (6 points)

$S_0' = A\cdot S_1$ $\quad\quad$ $S_1' = \overline{A}\cdot S_0$ $\quad\quad$ $X = S_1 + S_0$


A
Clock
S_1'
D
Q
CLK
S_1
S_0'
D
Q
CLK
S_0
X