Dr. Alexandru Patriciu **COE2DI4 Midterm Test #2** *Nov 12, 2008*

Instructions: This examination paper includes 12 pages and 18 multiple-choice questions starting on page 3. You are responsible for ensuring that your copy of the paper is complete. Bring any discrepancy to the attention of your invigilator. The answers for all the questions must be indicated by filling the corresponding circle on the optical scanning (OMR) examination sheet. **This OMR examination sheet is the only page to be handed in.** The instructions for completing the OMR examination sheet are provided on page 2. Read and follow these instructions with care! There is one mark for each question. Answer all questions. There is no penalty for guessing. This is a closed book exam. No reference material of any kind is permitted. No calculators of any kind are permitted. **Time allowed is 50 minutes**.

Note: A' and $\overline{A}$ are used interchangeably.

Continued on page 2

OMR examination instructions

Continued on page 3

Multiple choice questions (numbered 1 to 18) – indicate your answer by filling the corresponding circle on the OMR answer sheet

1. The correct implementation of a three input XOR gate **f(x,y,z) = x ⊕ y ⊕ z** is:

1. shown in Figure 1(a)
2. shown in Figure 1(b)
3. shown in Figure 1(c)
4. shown in Figure 1(d)

[✓ 5.] none of the above

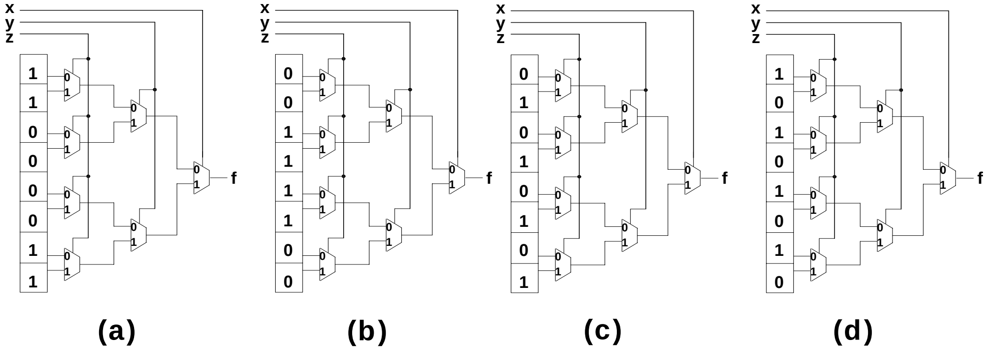


Figure 1 - Look-up table (LUT) and MUX implementations for function f(x,y,z) for question 1.

2.

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY question_11 IS
      PORT (w0, w1, w2, w3    : IN  STD_LOGIC ;
            s                 : IN  STD_LOGIC_VECTOR(1 DOWNTO 0) ;
            f                 : OUT STD_LOGIC ) ;
END question_11 ;

ARCHITECTURE Behavior OF question_11 IS
BEGIN
      WITH s SELECT
            f <=  w0 WHEN "00",
                  w1 WHEN "01",
                  w2 WHEN "10",
                  w3 WHEN OTHERS ;
END Behavior ;
```

The above VHDL code describes:

1. an 1-to-2 decoder
2. a 2-to-4 decoder
3. a 4-to-2 priority encoder

Continued on page 4

4. a 2-to-1 multiplexer
[✓ 5.] a 4-to-1 multiplexer

3. Consider the shift register circuit shown in Figure 2. Assume that **$I_3I_2I_1I_0$** =0101 has been loaded in the 4-bit register using the parallel load mechanism (i.e., **shift**=0). After the data has been stored in the register, for how many consecutive positive edges of the clock signal we need to keep **shift**=1 such that **zero_detect** is activated to a 1?

1. 1
2. 2
[✓ 3.] 3
4. 4
5. none of the above

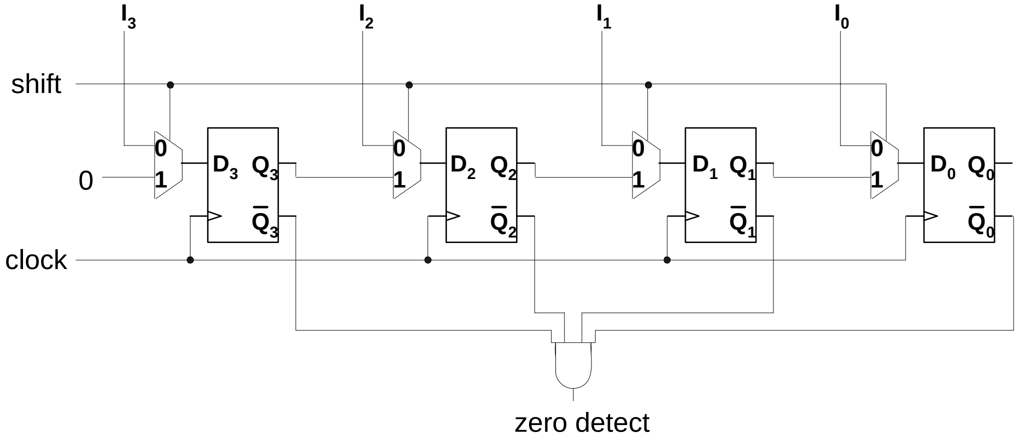
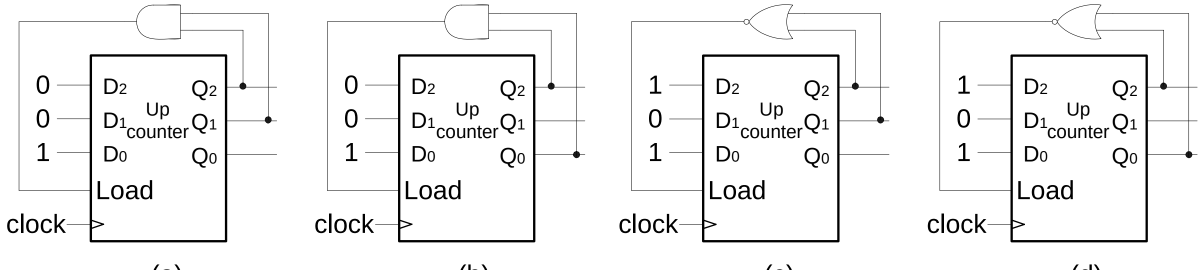


Figure 2 - Shift register circuit for question 3.



(a) (b) (c) (d)

Figure 3 - Circuits for question 4(it is assumed that after power up the state is 001).

4. Using an *up counter* with parallel load capability (**Q_2** and **D_2** are the most significant bits) we can generate a counting sequence 1, 2, 3, 4, 5 using the circuit shown in:

1. Figure 3(a)
[✓ 2.] Figure 3(b)
3. Figure 3(c)
4. Figure 3(d)
5. none of the above

Continued on page 5

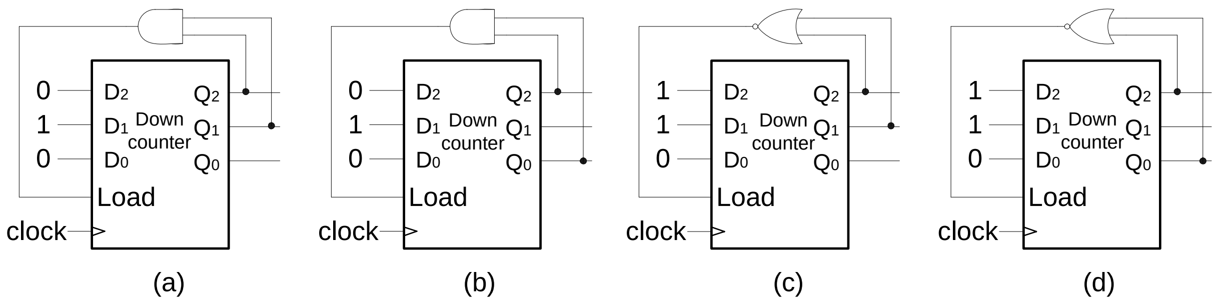


Figure 4 - Circuits for question 5 (it is assumed that after power up the state is 110).

5. Using a *down counter* with parallel load capability ($\mathbf{Q_2}$ and $\mathbf{D_2}$ are the most significant bits) we can generate a counting sequence 6, 5, 4, 3, 2 using the circuit shown in:

1. Figure 4(a)
2. Figure 4(b)
3. Figure 4(c)

[✓ 4.] Figure 4(d)

5. none of the above

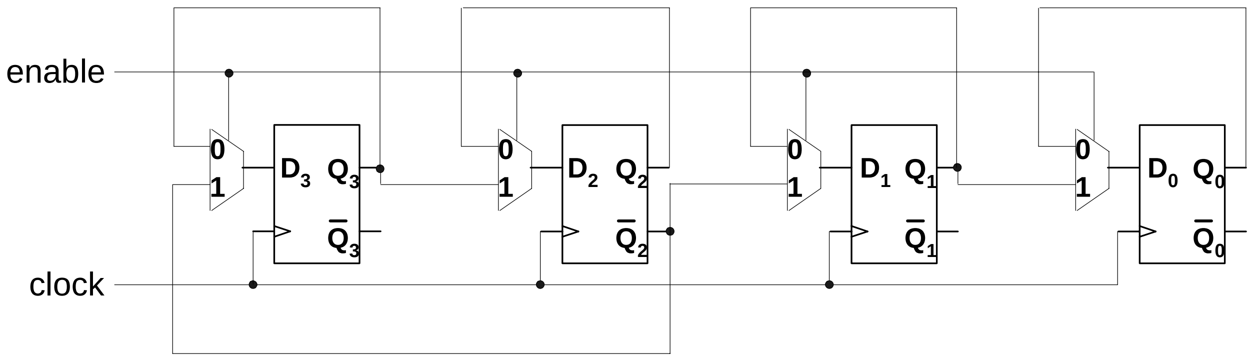


Figure 5 - Circuit for questions 6 and 7

6. In Figure 5, if enable=1 and the current (or present) state is $\mathbf{Q_3Q_2Q_1Q_0}$=0101, then the next state will be:

1. 1111
2. 1010
3. 1100
4. 0011

[✓ 5.] 0000

7. In Figure 5, let's assume that that t_{CQ} = 5ns, t_{su} = 2ns, and the propagation time through the MUX is t_{MUX} = 2ns. Which of the following values can be used as a clock time period?

1. 1nS
2. 5nS
3. 2nS
4. 4nS

Continued on page 6

5. 10nS

8. Figure 6 uses a 3-bit synchronous up counter with parallel load capability (**Q_2** and **D_2** are the most significant bits) to can generate a periodic signal COut. If the period of the input Clock is t_{CLK} what is the period of COut:

1. t_{CLK}
2. $2*t_{CLK}$
[✓ 3.] $6*t_{CLK}$
4. $4*t_{CLK}$
5. $8*t_{CLK}$

9. Figure 6 uses a 3-bit synchronous up counter with parallel load capability (**Q_2** and **D_2** are the most significant bits) to can generate a periodic signal COut. Let's assume that the duty cycle of the input Clock is 50% (Clock = 1 from 0 to $t_{CLK}/2$ and 0 from $t_{CLK}/2$ to t_{CLK}). What is the duty cycle of COut.

1. 50%
2. 20%
3. 40%
4. 60%
[✓ 5.] none of the above

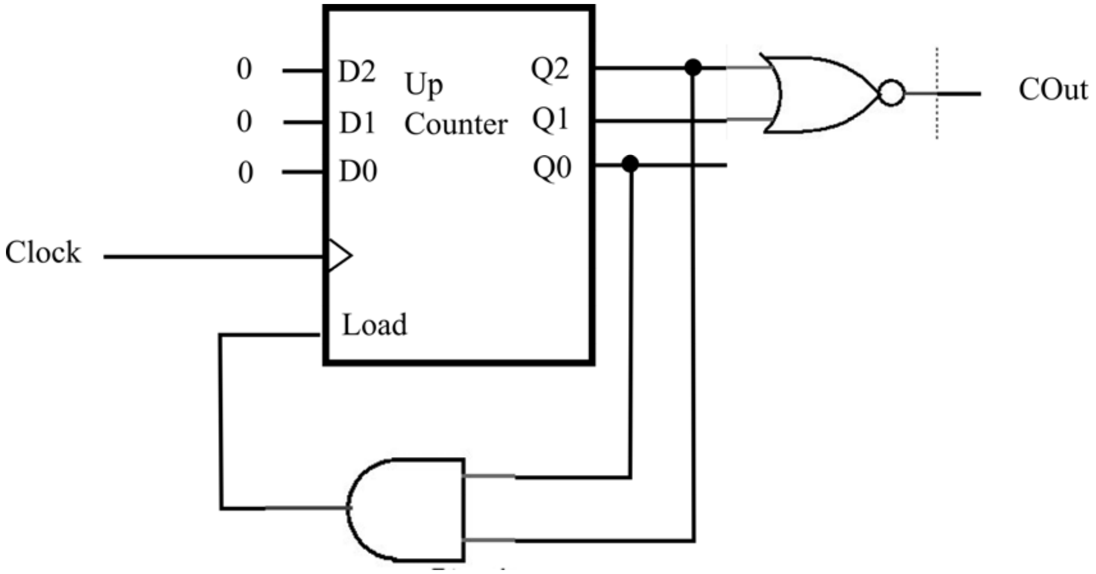


Figure 6: Circuit for questions 8 and 9

10. The periodic sequence **Q_1Q_0** generated by the circuit shown in Figure 7 is:

1. 00, 01, 10, 11
[✓ 2.] 00, 01, 11, 10
3. 00, 10, 11, 01
4. 00, 10, 01, 11
5. none of the above

Continued on page 7

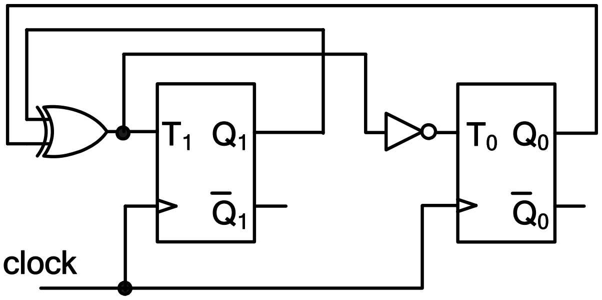


Figure 7 - Circuit for question 10 (it is assumed that after power up the state is 00).

11. The periodic sequence $\mathbf{Q_2Q_1Q_0}$ generated by the circuit shown in Figure 8 is:

1. 000, 011, 010, 101, 100
[✓ 2.] 000, 010, 101, 100
3. 000, 010, 101
4. 000, 011
5. none of the above

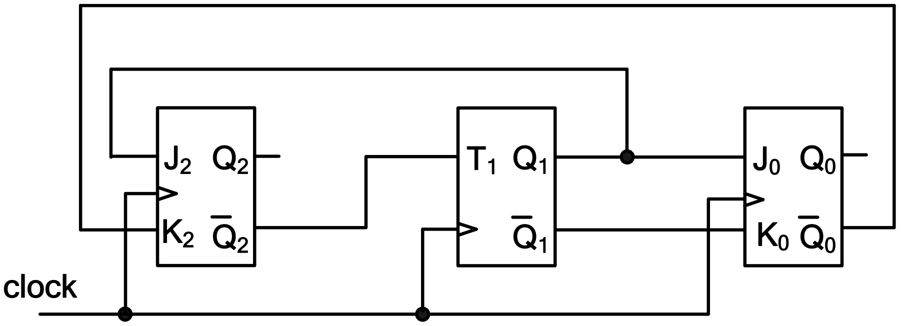


Figure 8 - Circuit for question 11 (it is assumed that after power up the state is 000).

12. In Figure 9, if the current state is $\mathbf{Q_3Q_2Q_1Q_0}$=0110, then after the next *positive edge* of the ***clock*** signal the new state will be:

1. 0110
2. 0100
3. 1000
[✓ 4.] 1001
5. 1110

Continued on page 8

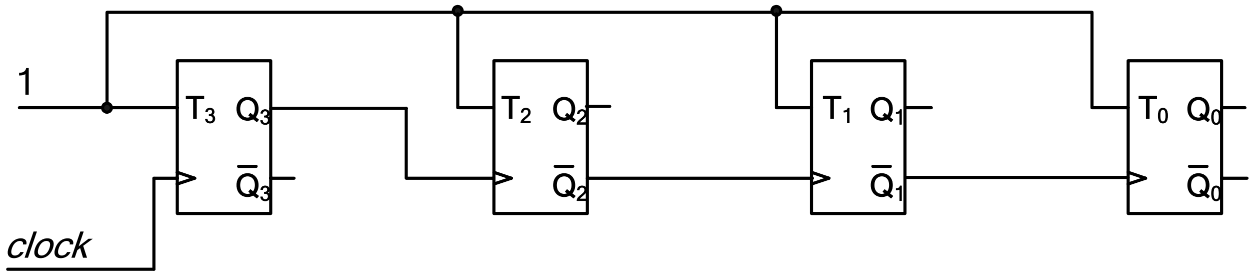
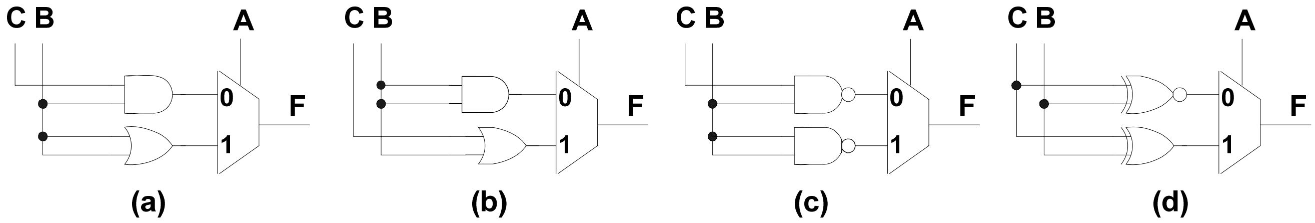


Figure 9 - Circuit for question 12 (note, the flip-flops are *positive edge* triggered).

<u>Note for question 12:</u> Because NOT all the flip-flops' clock inputs share the same *clock* signal, it is considered that the new state value will be observed only when all the flip-flops' outputs have become stable, i.e., after *all* the events triggered by the *positive edge* of the *clock* signal have been completed.

13. Consider function **G(A,B,C)=AB+BC**. Let **F(A,B,C)** be the dual of **G(A,B,C)**. Then **F(A,B,C)** can be implemented using **A** as the select (control) input to a 2-to-1 multiplexer. The correct implementation of **F(A,B,C)** is shown in:

1. Figure 10(a)
[✓ 2.] Figure 10(b)
3. Figure 10(c)
4. Figure 10(d)
5. none of the above



(a) (b) (c) (d)

Figure 10 - Implementations for function F(A,B,C) for question 13.

14. The periodic sequence $\mathbf{Q_2Q_1Q_0}$ generated by the circuit shown in Figure 11 is:

1. 000, 111, 011, 010, 101
[✓ 2.] 000, 111, 011, 101
3. 000, 111, 101
4. 000, 111
5. none of the above

Continued on page 9

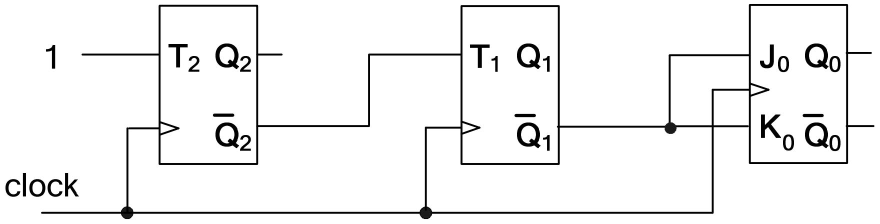


Figure 11 - Circuit for question 14 (it is assumed that after power up the state is 000).

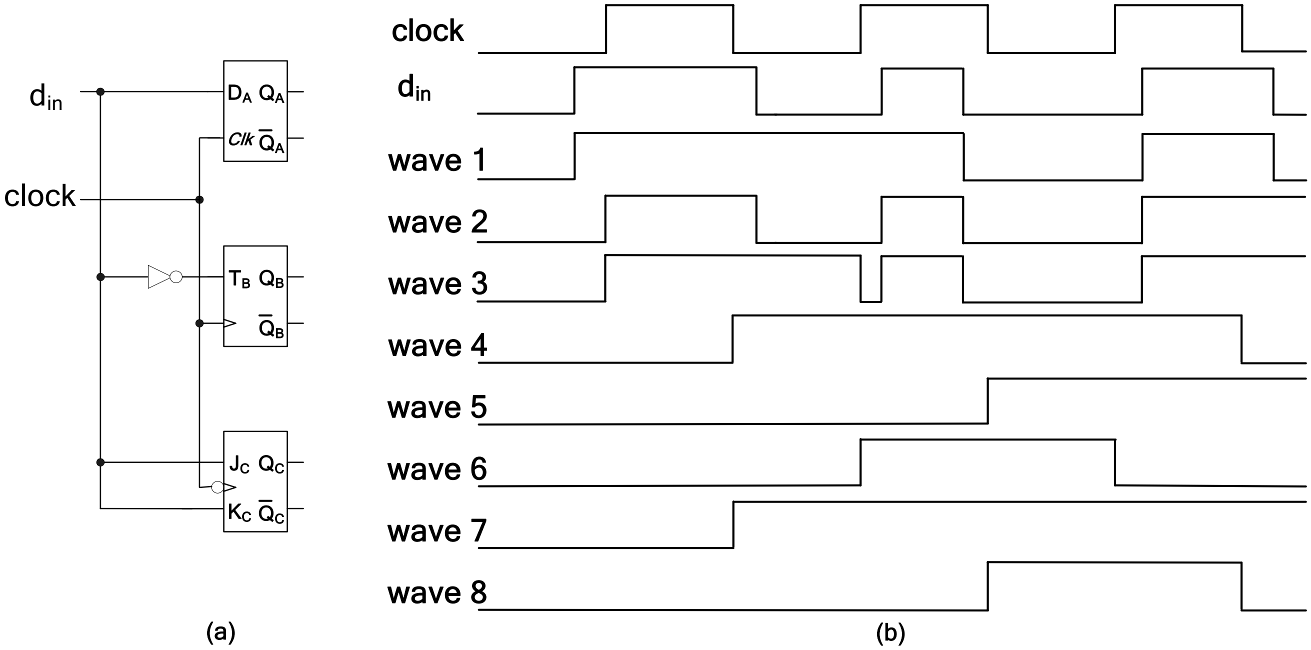


Figure 12 - Sequential (or storage) elements and signal waveforms for questions 15, 16 and 17. Note, the initial value of all the storage elements is zero and they are assumed to be ideal, i.e., the propagation delays and setup/hold times are considered to be zero.

15. Given the **clock** and **d_{in}** waveforms in Figure 12(b) the waveform for **Q_A** from Figure 12(a) is:

1. wave 1 from Figure 12(b)
2. wave 2 from Figure 12(b)
[✓ 3.] wave 3 from Figure 12(b)
4. wave 4 from Figure 12(b)
5. wave 5 from Figure 12(b)

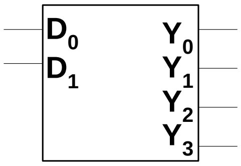
16. Given the **clock** and **d_{in}** waveforms in Figure 12(b) the waveform for **Q_B** from Figure 12(a) is:

1. wave 4 from Figure 12(b)
2. wave 5 from Figure 12(b)
[✓ 3.] wave 6 from Figure 12(b)
4. wave 7 from Figure 12(b)
5. wave 8 from Figure 12(b)

Continued on page 10

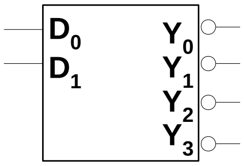
17. Given the **clock** and $\mathbf{d_{in}}$ waveforms in Figure 12(b) the waveform for $\mathbf{Q_C}$ from Figure 12(a) is:

[✓ 1.] wave 4 from Figure 12(b)
2. wave 5 from Figure 12(b)
3. wave 6 from Figure 12(b)
4. wave 7 from Figure 12(b)
5. wave 8 from Figure 12(b)



D_1 D_0	Y_0 Y_1 Y_2 Y_3
0 0	1 0 0 0
0 1	0 1 0 0
1 0	0 0 1 0
1 1	0 0 0 1

(a) Decoder with un-inverted outputs



D_1 D_0	Y_0 Y_1 Y_2 Y_3
0 0	0 1 1 1
0 1	1 0 1 1
1 0	1 1 0 1
1 1	1 1 1 0

(b) Decoder with inverted outputs

Figure 13 - Block diagrams and truth tables for decoders without an enable signal for questions 18 and 19

18. The circuit shown in Figure 14 is:

1. **F(A,B,C,D)** = Σ **m**(0,1,2,3)
2. **F(A,B,C,D)** = Σ **m**(0,3,12,15)
3. **F(A,B,C,D)** = Σ **m**(0,4,8,12)

[✓ 4.] **F(A,B,C,D)** = Σ **m**(0,5,10,15)
5. none of the above

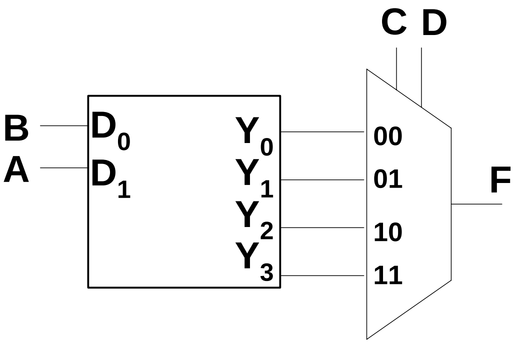


Figure 14 – Circuit for question 18. Use the truth table of the decoder with un-inverted outputs shown in Figure 13(a).

19. The circuit shown in Figure 15 is:

1. **F(A,B,C,D)** = $\prod$ **M**(0,2,13,15)
2. **F(A,B,C,D)** = $\prod$ **M**(0,4,11,15)

[✓ 3.] **F(A,B,C,D)** = $\prod$ **M** (0,5,10,15)
4. **F(A,B,C,D)** = $\prod$ **M** (0,7,8,15)
5. none of the above

Continued on page 11

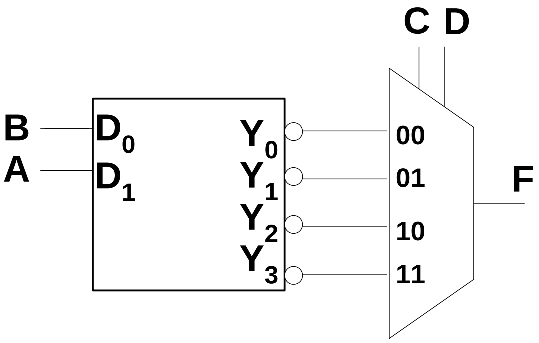


Figure 15 – Circuit for question 19. Use the truth table of the decoder with inverted outputs shown in Figure 13(b).

- THE END -

Continued on page 12