Dr. Nicola Nicolici **COE2DI4 Midterm Test #2** *Nov 19, 2007*

Instructions: This examination paper includes 12 pages and 20 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 20) – indicate your answer by filling the corresponding circle on the OMR answer sheet

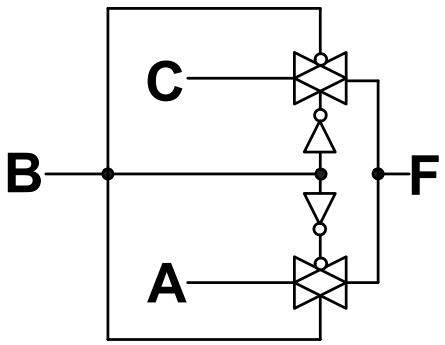


Figure 1 - CMOS circuit for question 1.

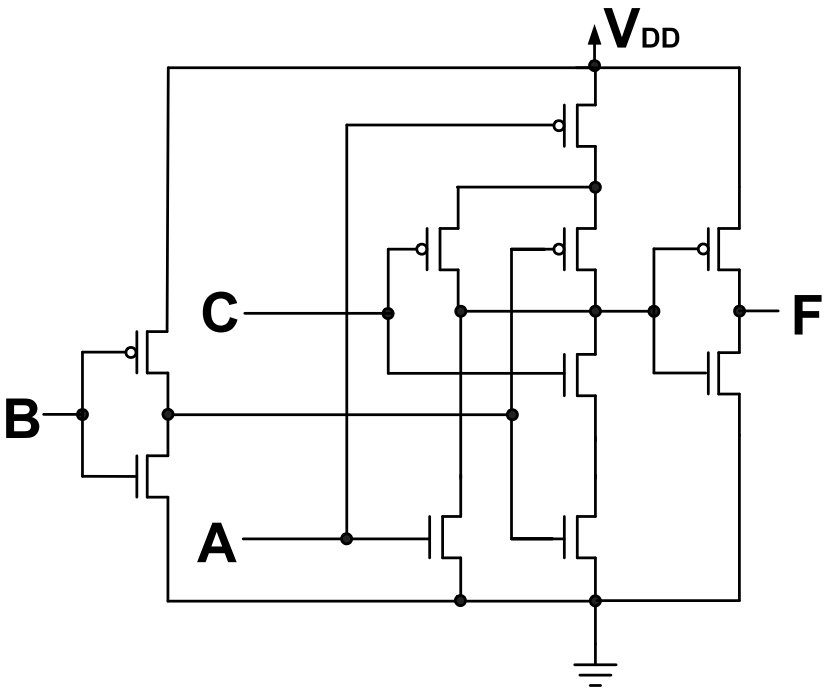


Figure 2 - CMOS circuit for question 2.

1. The logic function **F(A,B,C)** implemented by the complementary metal-oxide semiconductor (CMOS) circuit shown in Figure 1:

1. AB' + B'C
2. AB' + BC
3. AB + B'C
4. AB + BC
5. none of the above

2. The logic function **F(A,B,C)** implemented by the CMOS circuit shown in Figure 2 is:

1. A + BC
2. A + B'C
3. AB + AC
4. A'B + A'C'
5. none of the above

Continued on page 4

3. The function implemented using three 2-input look-up tables (LUTs) as shown in Figure 3 is:

1. **F(A,B,C)** = **Σ m** (0,1,2,5,6)
2. **F(A,B,C)** = **Σ m** (0,2,5,6)
3. **F(A,B,C)** = **Σ m** (2,5)
4. **F(A,B,C)** = **Σ m** (5)
5. none of the above

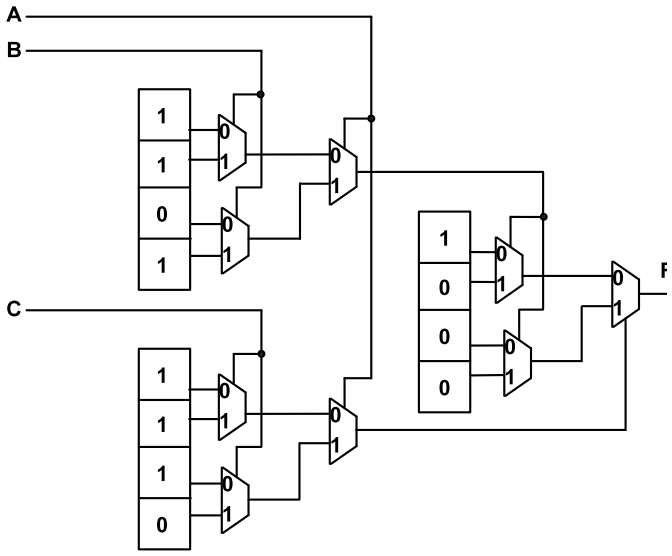


Figure 3 – Circuit for question 3.

4. The programmable array logic (PAL) implementation of function F shown in Figure 4 is:

1. **F(A,B,C)** = **Σ m** (0,1,2,6)
2. **F(A,B,C)** = **Σ m** (0,1,6,7)
3. **F(A,B,C)** = **Σ m** (0,2,6,7)
4. **F(A,B,C)** = **Σ m** (0,1,2,5,6,7)
5. none of the above

5. The PAL implementation of function G shown in Figure 4 is:

1. **G(A,B,C)** = **Σ m** (2,3,5,7)
2. **G(A,B,C)** = **Σ m** (2,3,4,7)
3. **G(A,B,C)** = **Σ m** (2,4,5,7)
4. **G(A,B,C)** = **Σ m** (2,3,4,5,7)
5. none of the above

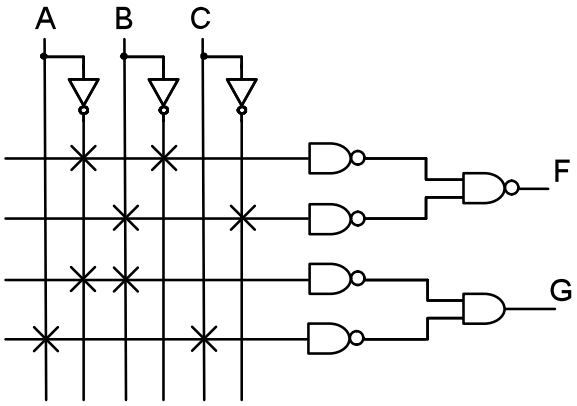


Figure 4 – Circuit for questions 4 and 5. An X indicates that the wires are connected, i.e., the vertical signal is an input to the corresponding horizontal NAND gate.

Continued on page 5

6. Consider a read-only memory (ROM) that performs binary to 3-digit binary-coded decimal (BCD) conversion as follows. The binary number is applied on the 10 input address lines, $A_9A_8A_7A_6A_5A_4A_3A_2A_1A_0$, and the equivalent 3-digit BCD number is read on the 12 output data lines, $D_{11}D_{10}D_9D_8D_7D_6D_5D_4D_3D_2D_1D_0$, after the memory access time (note, the most significant BCD digit will appear on the four most significant bits on the output and the least significant BCD digit will appear on the four lest significant bits on the output). Consequently, when applying the input address $A_9A_8A_7A_6A_5A_4A_3A_2A_1A_0$=1100000000, after the memory access time the output data will become:

1. $D_{11}D_{10}D_9D_8D_7D_6D_5D_4D_3D_2D_1D_0$=0001 0001 0010
2. $D_{11}D_{10}D_9D_8D_7D_6D_5D_4D_3D_2D_1D_0$=0010 0101 0110
3. $D_{11}D_{10}D_9D_8D_7D_6D_5D_4D_3D_2D_1D_0$=0011 1010 1110
4. $D_{11}D_{10}D_9D_8D_7D_6D_5D_4D_3D_2D_1D_0$=0111 0110 1000
5. none of the above

7.

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

ENTITY question7 IS
       PORT(w : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
            y : OUT STD_LOGIC_VECTOR(2 DOWNTO 0));
END question7;

ARCHITECTURE Behavior OF question7 IS
BEGIN
       y <= "111" WHEN w(3) = '1' ELSE
            "100" WHEN w(2) = '1' ELSE
            "010" WHEN w(1) = '1' ELSE
            "001";
END Behavior;
```

In the above VHDL description, the assignment to the output signal `y(0)` is equivalent to:

1. `y(0) <= w(3) or ((not w(2)) and w(1));`
2. `y(0) <= w(3) or (w(2) and (not w(1)));`
3. `y(0) <= w(3) or ((not w(2)) and (not w(1)));`
4. `y(0) <= (not w(3)) or ((not w(2)) and (not w(1)));`
5. none of the above

Continued on page 6

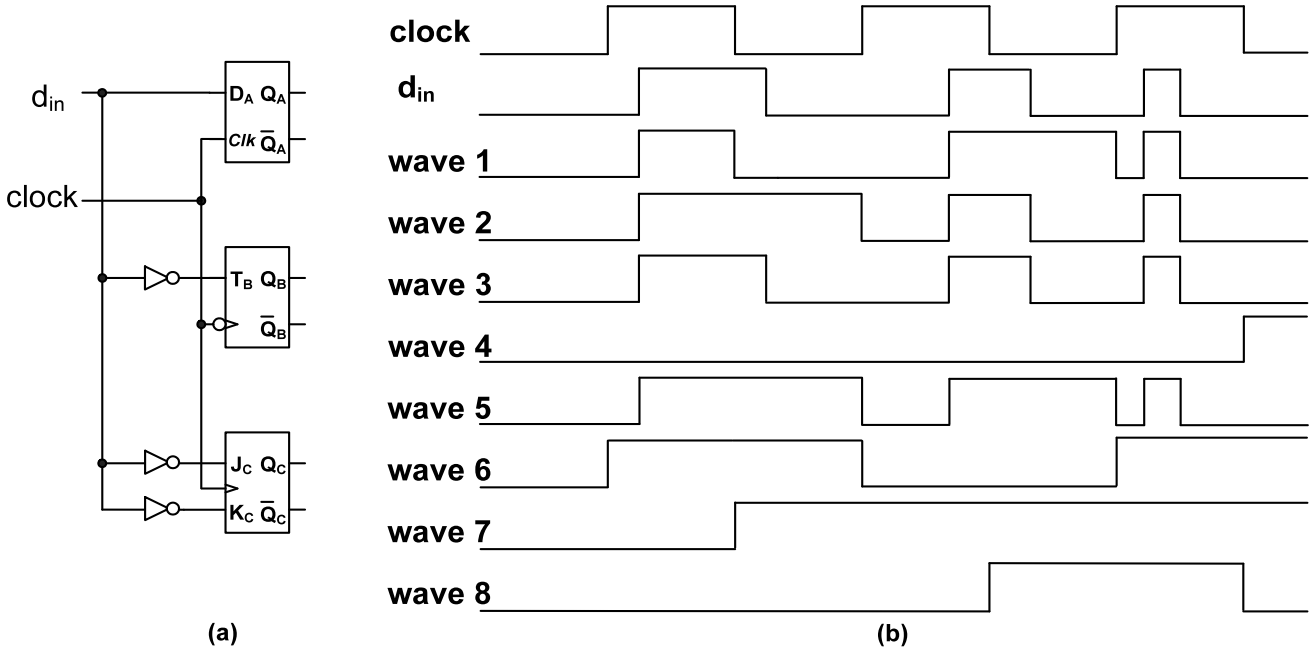


Figure 5 - Sequential (or storage) elements and signal waveforms for questions 8, 9 and 10. Note, the initial value of all the storage elements is zero and they are assumed to be ideal, i.e., the propagation delays and setup/hold times are considered to be zero. The propagation delays on the inverters placed between d_{in} and flip-flops' inputs are considered also to be zero.

8. Given the **clock** and $\mathbf{d_{in}}$ waveforms in Figure 5(b) the waveform for $\mathbf{Q_A}$ from Figure 5(a) is:

1. wave 1 from Figure 5(b)
2. wave 2 from Figure 5(b)
3. wave 3 from Figure 5(b)
4. wave 4 from Figure 5(b)
5. wave 5 from Figure 5(b)

9. Given the **clock** and $\mathbf{d_{in}}$ waveforms in Figure 5(b) the waveform for $\mathbf{Q_B}$ from Figure 5(a) is:

1. wave 4 from Figure 5(b)
2. wave 5 from Figure 5(b)
3. wave 6 from Figure 5(b)
4. wave 7 from Figure 5(b)
5. wave 8 from Figure 5(b)

10. Given the **clock** and $\mathbf{d_{in}}$ waveforms in Figure 5(b) the waveform for $\mathbf{Q_C}$ from Figure 5(a) is:

1. wave 4 from Figure 5(b)
2. wave 5 from Figure 5(b)
3. wave 6 from Figure 5(b)
4. wave 7 from Figure 5(b)
5. wave 8 from Figure 5(b)

Continued on page 7

11. In Figure 6, if the current state is $Q_3Q_2Q_1Q_0$=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

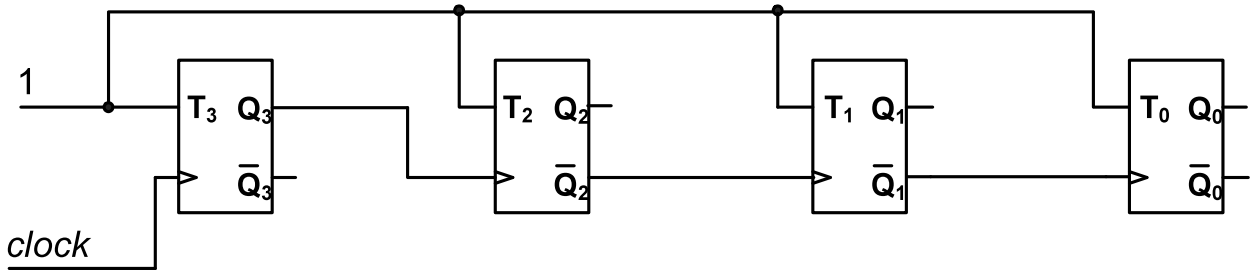


Figure 6 - Circuit for question 11 (note, the flip-flops are *positive edge* triggered).

Note for question 11: 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.

12. The periodic sequence $Q_2Q_1Q_0$ generated by the circuit shown in Figure 7 is:

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

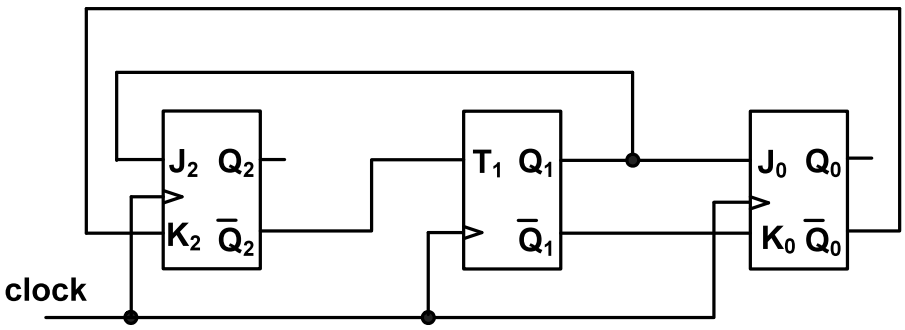


Figure 7 - Circuit for question 12 (it is assumed that after power up the state is 000).

Continued on page 8

13. The periodic sequence Q_1Q_0 generated by the circuit shown in Figure 8 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

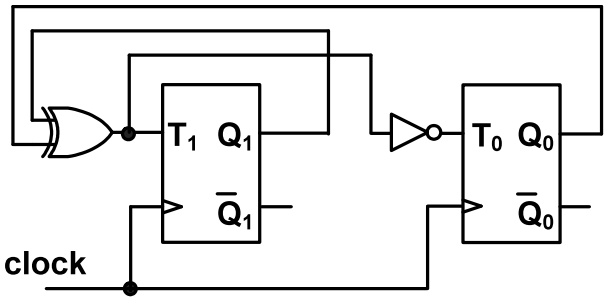


Figure 8 - Circuit for question 13 (it is assumed that after power up the state is 00).

14. Figure 9 shows a frequency division circuit implemented using a synchronous 3-bit *down-counter* with parallel load capability. If the frequency of the ***clock*** signal is 60 MHz, then signal ***z***, which is a periodic signal, has its frequency equal to:

1. 30 MHz
2. 20 MHz
3. 15 MHz
4. 12 MHz
5. 10 MHz

<u>Tip for question 14:</u> To compute the frequency of signal *z*, you must derive the periodic sequence generated by the circuit from Figure 9.

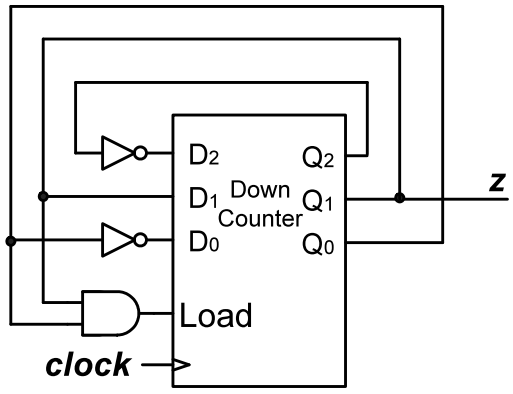


Figure 9 – Circuit for question 14
(it is assumed that after power up the state is 000).

15. Which of the circuits shown in Figure 10 (see next page) are equivalent?

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

Continued on page 9

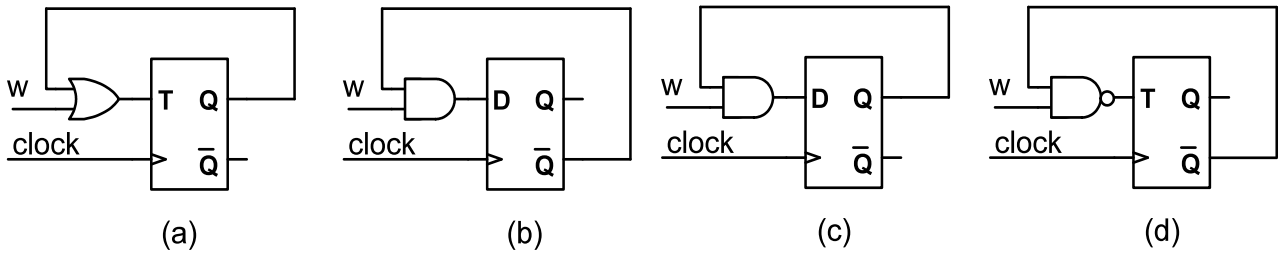
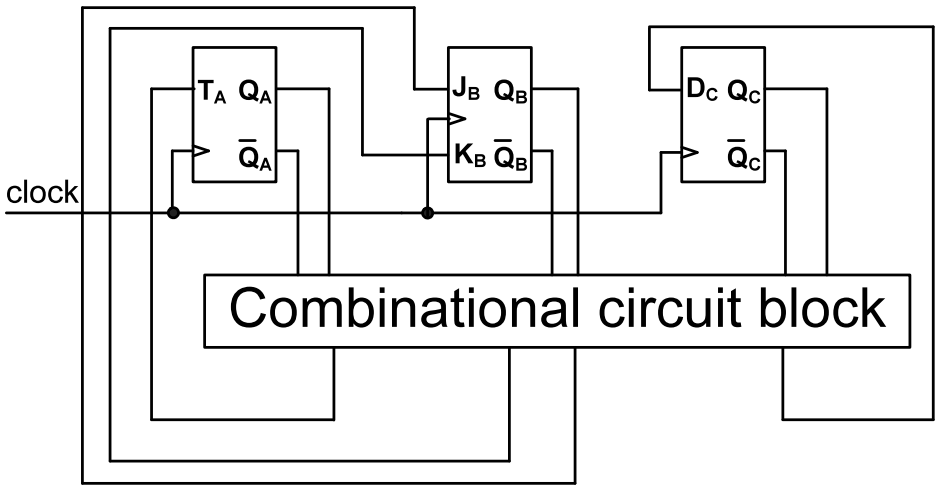


Figure 10 - Circuits for question 15. Note: For subfigures (b) and (d), Q' is used as an input to the AND and the NAND gate respectively.

Present State	Next State	Use this area for the truth tables of the flip-flops' inputs
000	100	
001	010	
010	011	
011	111	
100	101	
101	001	
110	000	
111	110	

(a)



(b)

Figure 11 - State table and circuit for questions 16, 17 and 18.
Note: Q_A is the most significant bit.
Tip: fill in the truth tables and use K-maps for logic minimization.

Continued on page 10

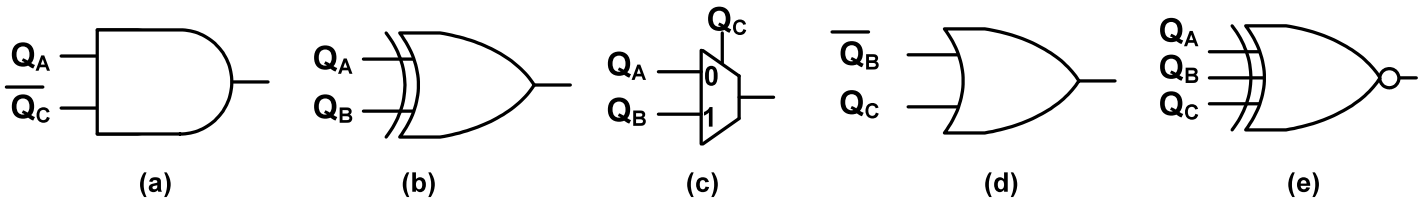


Figure 12 - Circuit implementations for questions 16, 17 and 18.

16. Consider the state transition table of a sequential circuit shown in Figure 11(a) (see previous page). If in our technology library we have only one **T** flip-flop, one **JK** flip-flop and one **D** flip-flop, then a generic implementation is shown Figure 11(b) (see previous page). Based on the state transition table and circuit topology shown in Figure 11, the logic function, which drives the *input* $\mathbf{T_A}$ *of the leftmost flip-flop from Figure 11(b)* is shown in:

1. Figure 12(a)
2. Figure 12(b)
3. Figure 12(c)
4. Figure 12(d)
5. Figure 12(e)

17. Consider the state transition table of a sequential circuit shown in Figure 11(a) (see previous page). If in our technology library we have only one **T** flip-flop, one **JK** flip-flop and one **D** flip-flop, then a generic implementation is shown Figure 11(b) (see previous page). Based on the state transition table and circuit topology shown in Figure 11, the logic function, which drives the *input* $\mathbf{K_B}$ *of the center flip-flop from Figure 11(b)* is shown in:

1. Figure 12(a)
2. Figure 12(b)
3. Figure 12(c)
4. Figure 12(d)
5. Figure 12(e)

18. Consider the state transition table of a sequential circuit shown in Figure 11(a) (see previous page). If in our technology library we have only one **T** flip-flop, one **JK** flip-flop and one **D** flip-flop, then a generic implementation is shown Figure 11(b) (see previous page). Based on the state transition table and circuit topology shown in Figure 11, the logic function, which drives the *input* $\mathbf{D_C}$ *of the rightmost flip-flop from Figure 11(b)* is shown in:

1. Figure 12(a)
2. Figure 12(b)
3. Figure 12(c)
4. Figure 12(d)
5. Figure 12(e)

Continued on page 11

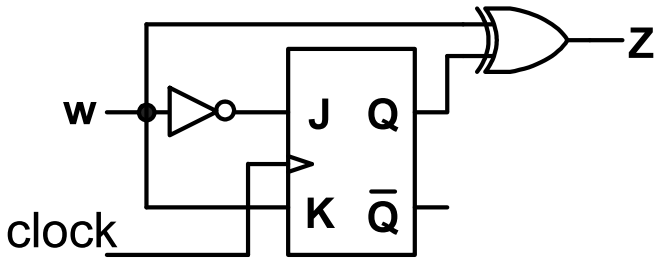


Figure 13 - Circuit for question 19.

19. Consider the sequential circuit shown in Figure 13. Consider the following state assignment: **A** stands for **Q**=0, **B** stands for **Q**=1. The state transition diagram for the circuit from Figure 13 is shown in:

1. Figure 14(a)
2. Figure 14(b)
3. Figure 14(c)
4. Figure 14(d)
5. none of the above

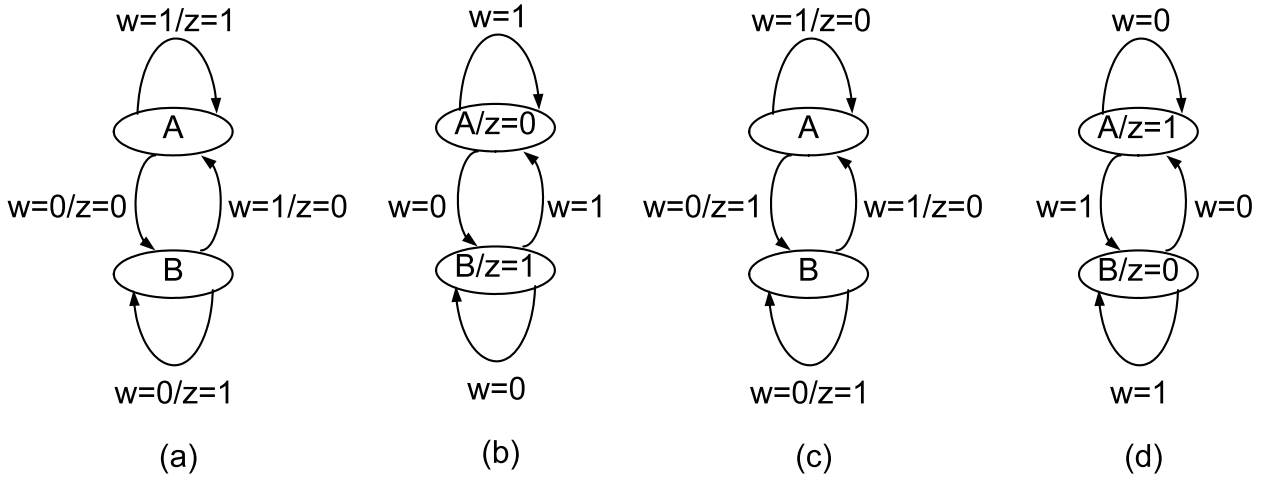


Figure 14 - State transition diagrams for question 19.

Continued on page 12

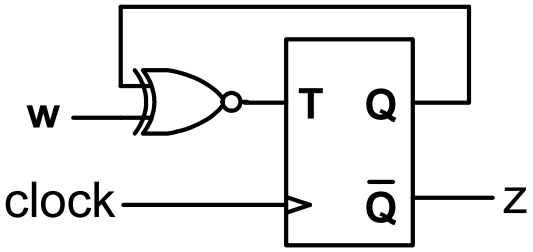


Figure 15 - Circuit for question 20.

20. Consider the sequential circuit shown in Figure 15. Consider the following state assignment: **A** stands for **Q**=0, **B** stands for **Q**=1. The state transition diagram for the circuit from Figure 15 is shown in:

1. Figure 16(a)
2. Figure 16(b)
3. Figure 16(c)
4. Figure 16(d)
5. none of the above

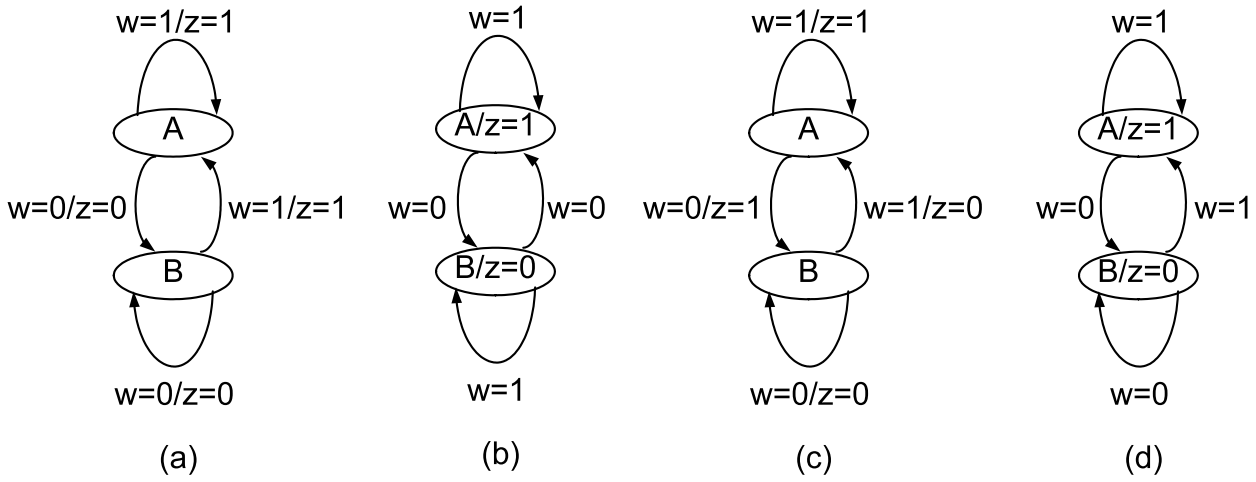


Figure 16 - State transition diagrams for question 20.

- THE END -