Introduction to Computing Systems Homework 2

PB20000051 Fu Shen(fushen@mail.ustc.edu.com)

October 26, 2021

Question 1.

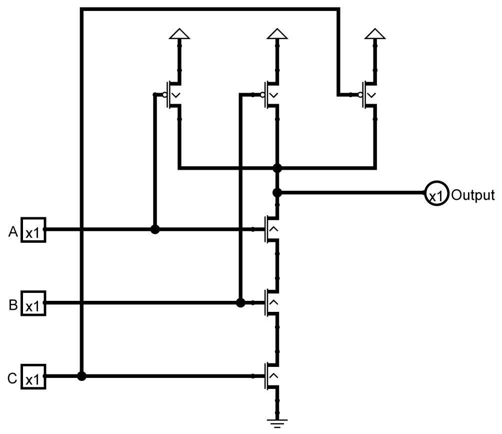
It's 2^{-126}, as `0000 0000 1000 0000 0000 0000 0000 0000` represents.

Question 2.

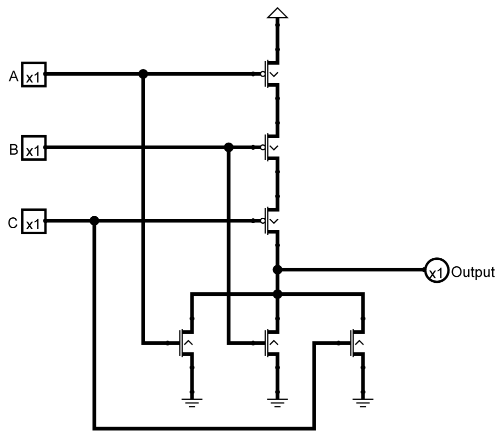
It's $2^{31} - 1 = 2147483647$, as `0111 1111 1111 1111 1111 1111 1111 1111` represents.

Question 3.

a. See Figure 1.



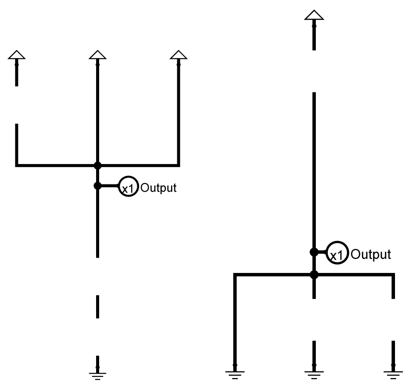
(a) NAND gate



(b) NOR gate

Figure 1: 3-input gate (Made by Logisim)

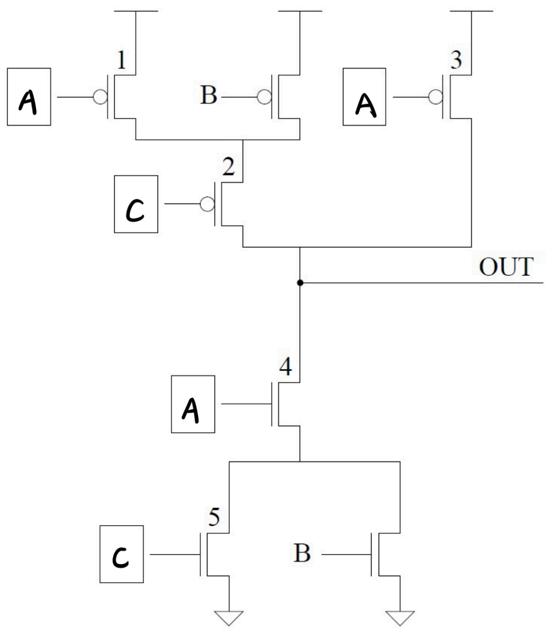
b. See Figure 2. The output of NAND gate is 1 and the output of NOR gate is 0;

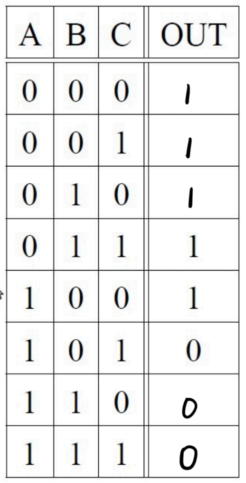


(a) NAND gate (b) NOR gate

Figure 2: Circuit's operation (Made by Logisim)

c. See Figure 3.





A	B	C	OUT
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Figure 3: The answer to question 3.c

Question 4.

a. X

b. 1

c. 0

d. X

e. 0

f. 0

Question 5.

Logic circuit 2 involve the storage of information but logic circuit 1 do not. For example, when the voltage at input A goes from 0 to 1, the output of logic circuit 1 will go from C to B. But if the input B in logic circuit 2 is 1, the output D won't change, remaining to be 1, which is described the *quiescent* state, so the previous information is stored. And if the input B is 0, the output D will go from a uncertain state to 0. The name of logic circuit is the 2-to-1 mux, and the name of the other is the R-S latch.

Question 6.

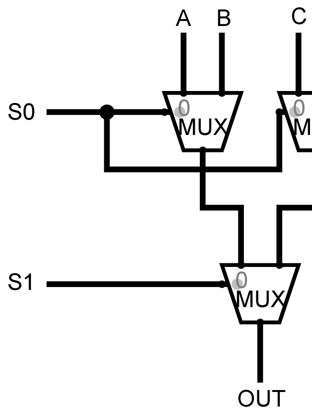
6.1) See Table 1.

Table 1: Truth table of Question 6.(1)

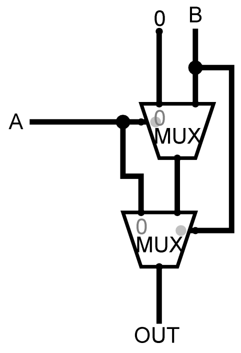
S1	S0	A	B	C	D	OUT	S1	S0	A	B	C	D	OUT
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0	1	0
0	0	0	0	1	0	0	1	0	0	0	1	0	1
0	0	0	0	1	1	0	1	0	0	0	1	1	1
0	0	0	1	0	0	0	1	0	0	1	0	0	0
0	0	0	1	0	1	0	1	0	0	1	0	1	0
0	0	0	1	1	0	0	1	0	0	1	1	0	1
0	0	0	1	1	1	0	1	0	0	1	1	1	1
0	0	1	0	0	0	1	1	0	1	0	0	0	0
0	0	1	0	0	1	1	1	0	1	0	0	1	0
0	0	1	0	1	0	1	1	0	1	0	1	0	1
0	0	1	0	1	1	1	1	0	1	0	1	1	1
0	0	1	1	0	0	1	1	0	1	1	0	0	0
0	0	1	1	0	1	1	1	0	1	1	0	1	0
0	0	1	1	1	0	1	1	0	1	1	1	0	1
0	0	1	1	1	1	1	1	0	1	1	1	1	1
0	1	0	0	0	0	0	1	1	0	0	0	0	0
0	1	0	0	0	1	0	1	1	0	0	0	1	1
0	1	0	0	1	0	0	1	1	0	0	1	0	0
0	1	0	0	1	1	0	1	1	0	0	1	1	1
0	1	0	1	0	0	1	1	1	0	1	0	0	0
0	1	0	1	0	1	1	1	1	0	1	0	1	1
0	1	0	1	1	0	1	1	1	0	1	1	0	0
0	1	0	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	0	0	0	1	1	1	0	0	0	0
0	1	1	0	0	1	0	1	1	1	0	0	1	1
0	1	1	0	1	0	0	1	1	1	0	1	0	0
0	1	1	0	1	1	0	1	1	1	0	1	1	1
0	1	1	1	0	0	1	1	1	1	1	0	0	0
0	1	1	1	0	1	1	1	1	1	1	0	1	1
0	1	1	1	1	0	1	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1

6.2) See Figure 4(a).

6.3) See Figure 4(b).



(a) 4-to-1 mux



(b) XOR gate

Figure 4: The implementation using 2-to-1 muxes.(Made by Logisim)

Question 7.

a. It's 3.

b. It's 12.

c. See Figure 5. Its propagation delay is 3, instead of 4 in the question figure.

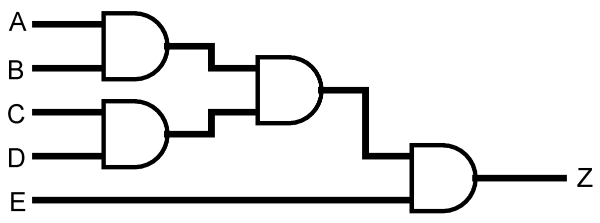


Figure 5: A different implementation. (Made by Logisim)

Question 8.

Just connect every condition that Si/Ci+1 is 1 with the corresponding OR gate. See Figure 6.

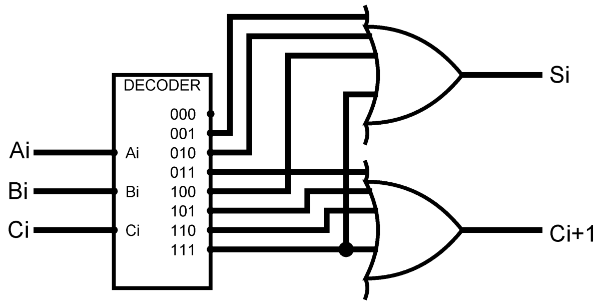


Figure 6: A full-adder. (Made by Logisim)

Question 9.

a. See Figure 7

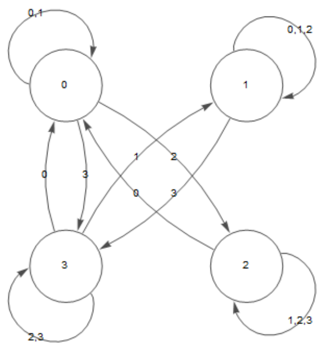


Figure 7: The state diagram of the elevator scheduling

b. See Table 2. The representation of State is S, and the representation of input & output is In & Out.

Table 2: Truth table of elevator controller.

S_1	S_0	In_1	In_0	S_1'/Out_1'	S_0'/Out_0'
0	0	0	x	0	0
0	0	1	0	1	0
0	0	1	1	1	1
0	1	0	x	0	1
0	1	1	0	0	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	1	0
1	0	1	x	1	0
1	1	0	0	0	0
1	1	0	1	0	1
1	1	1	x	1	1

Question 10.

The state after 50 cycles is 111000. It takes 6 cycles for a specific state to show up again, that is

$$100000 \rightarrow 111000 \rightarrow 111110 \rightarrow 011111 \rightarrow 000111 \rightarrow 000001 \rightarrow 100000$$

Question 11.

See Figure 8.

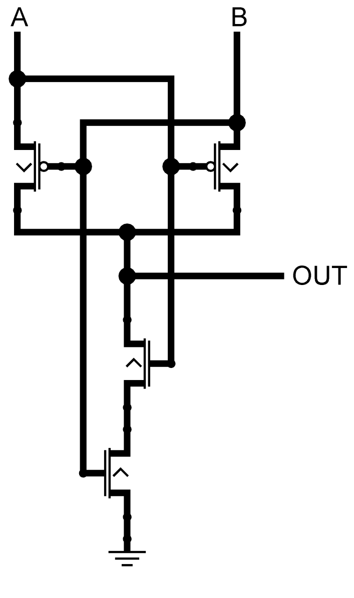


Figure 8: 2 input XOR gate

Question 12.

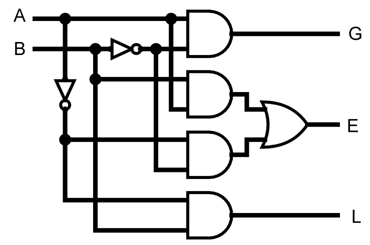
a. See Table 3.

Table 3: The truth table for a 1-bit comparator.

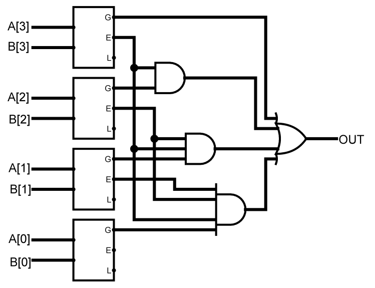
A	B	G	E	L
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

b. See Figure 9(a).

c. See Figure 9(b).



(a) Inplementation of 1-bit comparator.



(b) Inplementation of 4-bit comparator.

Figure 9: Comparator

Question 13.

See Table 4 and Figure 10.

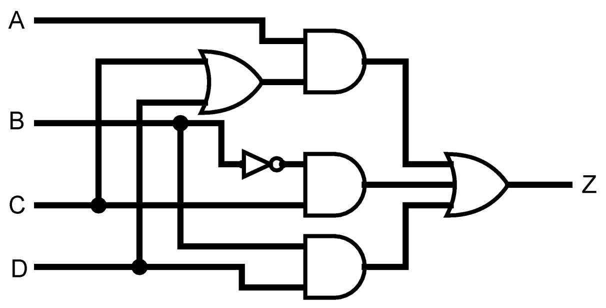


Figure 10: The logic circuit for the alarm.

Table 4: The truth table for the alarm.

A	B	C	D	Z
0	x	0	0	0
0	0	1	0	0
0	0	x	1	1
0	1	0	1	0
0	1	1	x	1
1	x	0	0	0
1	x	0	1	1
1	x	1	x	1

Question 14.

It's obvious that

$$\begin{gathered}\text{NOT } X = X \text{ NAND } X \\ X \text{ AND } Y = \text{NOT } (X \text{ NAND } Y) = (X \text{ NAND } Y) \text{ NAND } (X \text{ NAND } Y) \\ X \text{ OR } Y = (\text{NOT } X) \text{ NAND } (\text{NOT } Y) = (X \text{ NAND } X) \text{ NAND } (Y \text{ NAND } Y)\end{gathered}$$

And because the set of gates {AND, OR, NOT} is logically complete , NAND is logically complete.