Name: Daniyar Zhakyp

ID: 201774605

Lab Section: 1

Lab report

1.1.1. Testing a circuit

Figure 1. The circuit with an AND gate in off state.

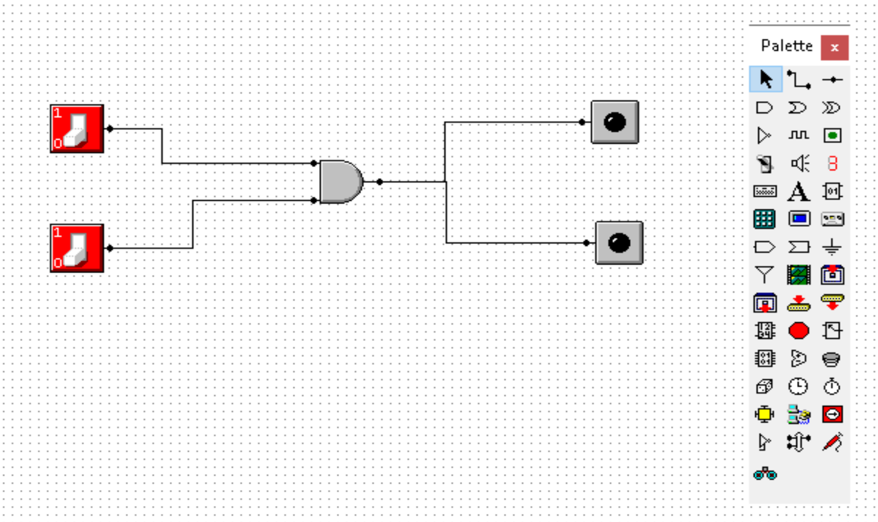
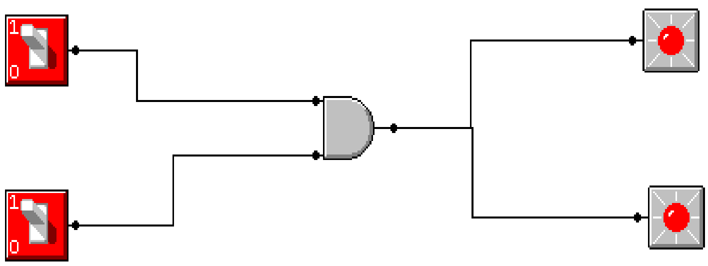


Figure 2. The circuit with an AND gate in on state.



Results: The circuit is working, now let's deduce the truth table for it:

Table 1. LEDs working conditions.

Upper switch	Lower switch	LEDs
Off	Off	Off
Off	On	Off
On	Off	Off
On	On	On

The LEDs are working when both switchs are on, the basic logic of an AND gate.

Table 2. The carry in the addition of two binary numbers.

Bit of #1 binary number	Bit of #2 binary number	Carry
0	0	0
0	1	0
1	0	0
1	1	1

So we see here the same logic: the carry is appearing only when the two summing digits are both equal to 1.

1.1.2. Changing a Circuit

Figure 3. The circuit with a NAND gate.

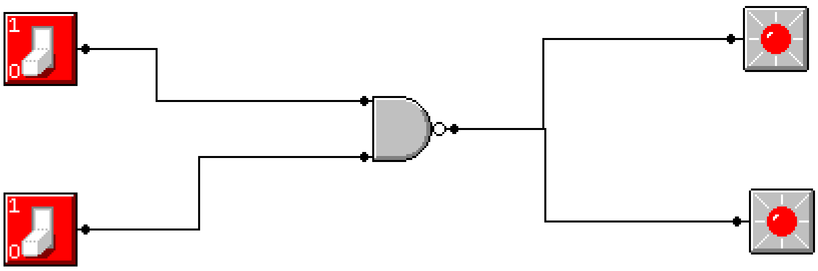


Table 3. The truth table for the NAND gate.

Upper switch	Lower switch	LEDs
Off	Off	On
Off	On	On
On	Off	On
On	On	Off

1.1.3 Making an adder

Figure 4. An adder circuit

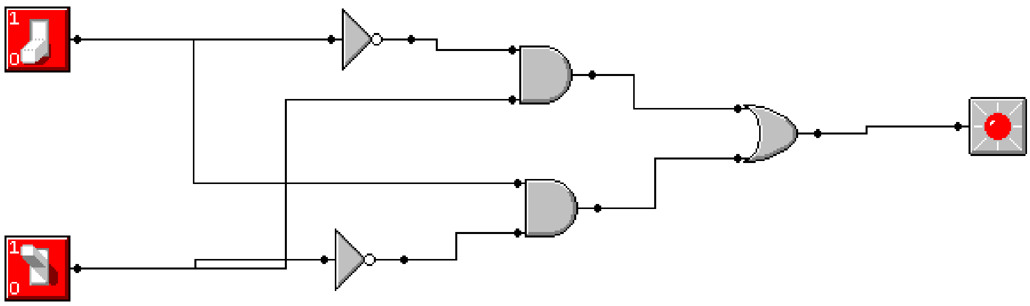


Table 4. The truth table for the adder circuit.

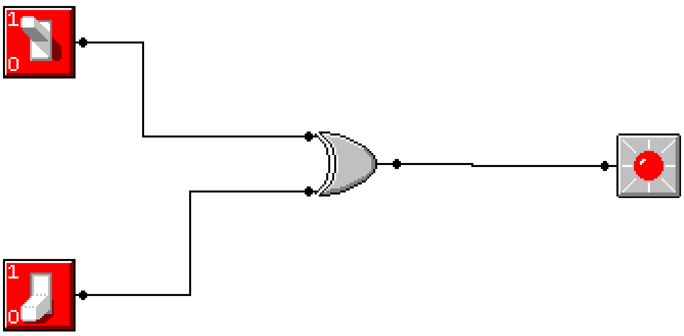
Upper switch	Lower switch	LEDs
Off	Off	Off
Off	On	On
On	Off	On
On	On	Off

Table 5. The truth table of a binary adder.

A	B	Z (LSB)
0	0	0
0	1	1
1	0	1
1	1	0

Result: They are the same.

Figure 5. The XOR circuit.



1.1.4 Adding with carry

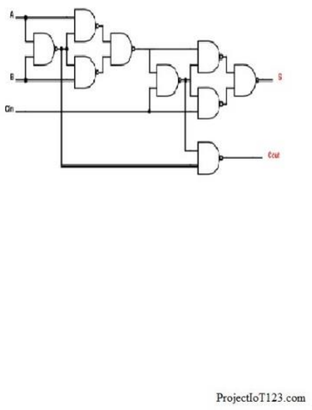
Table 6. The circuit of adding with carry

Left upper switch	Left lower switch	Bottom switch	Upper LED	Lower LED
Off	Off	Off	Off	Off
Off	Off	On	Off	On
Off	On	Off	Off	On
Off	On	On	On	Off
On	Off	Off	Off	On
On	Off	On	On	Off
On	On	Off	On	Off
On	On	On	On	On

Figure 6. Truth table for a full adder

Full Adder Truth Table

Inputs			Outputs	
A	B	C_{in}	C_{out}	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



Result: Our circuit actually works as a full adder. It shows the MSB – Cout.

1.1.5. Adding two digits

Figure 7. The circuit that adds two digits.

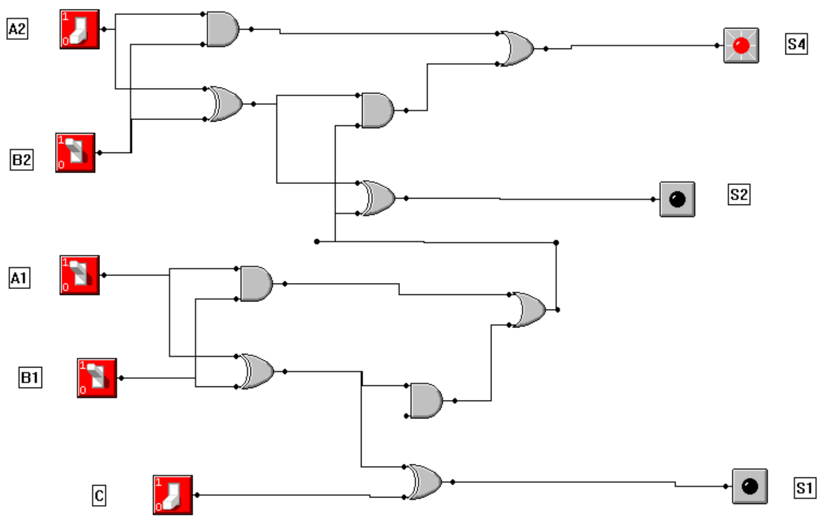


Table 7. Some values for adding two digits.

A_2	A_1	A	B_2	B_1	B	C	S_4	S_2	S_1
0	1	1	0	1	1	0	0	1	0
0	1	1	0	1	1	1	0	1	1
1	0	1	1	0	1	0	1	0	0
1	0	1	1	0	1	1	1	0	1

Answers:

a) Actually, this circuit can be perceived as one of the stages of an adder, where A2,A1 are the digits of one binary number, B2B1 of the other one, and C is the carry from the previous stage of the adder. Thus, S4 is the final carry which then can be used in another stage.

b) I think that the logic of the circuitry is the same, we just need to add some new switches and LEDs.

1.2 Electronic Dice

Part 1:

We fill out the table:

Figure 8. A table for an electronic dice.

Required pattern for each die roll	4 possible light patterns available: A	B	C	D	Count $Q_2Q_1Q_0$
1	1				1 001_2
2				1	2 010_2
3	1			1	3 011_2
4			1	1	4 100_2
5	1		1	1	5 101_2
6		1	1	1	6 110_2

Figure 9. K-maps

Q_2Q_1 \ Q_0	0	1
00	(0)	1 (1)
01	(2)	1 (3)
11	(6)	(7)
10	(4)	1 (5)

A = $\overline{Q_2}Q_0 + \overline{Q_1}Q_0$

Q_2Q_1 \ Q_0	0	1
00	(0)	(1)
01	(2)	(3)
11	1 (6)	(7)
10	(4)	(5)

B = $Q_2Q_1\overline{Q_0}$

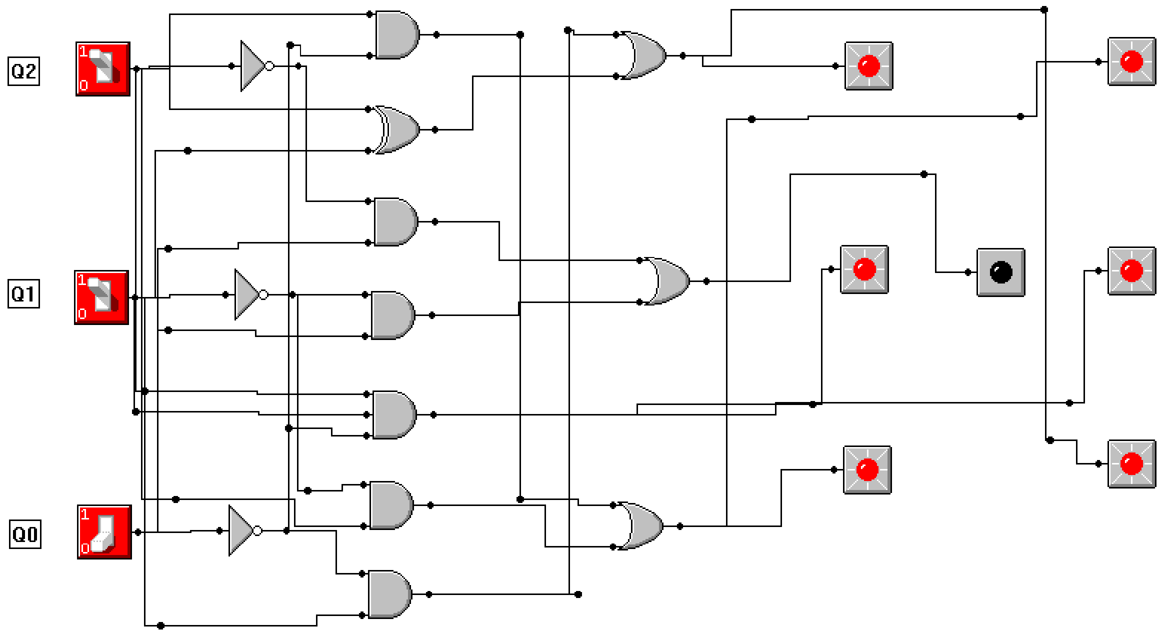
Q_2Q_1 \ Q_0	0	1
00	(0)	(1)
01	(2)	(3)
11	1 (6)	(7)
10	1 (4)	1 (5)

C = $Q_2\overline{Q_0} + Q_2\overline{Q_1}$

Q_2Q_1 \ Q_0	0	1
00	(0)	(1)
01	1 (2)	1 (3)
11	1 (6)	(7)
10	1 (4)	1 (5)

D = $\overline{Q_2}Q_1 + Q_1\overline{Q_0} + Q_2\overline{Q_1}$

Figure 10. The circuit when 110_2 or 6 is activated.



Part 2:

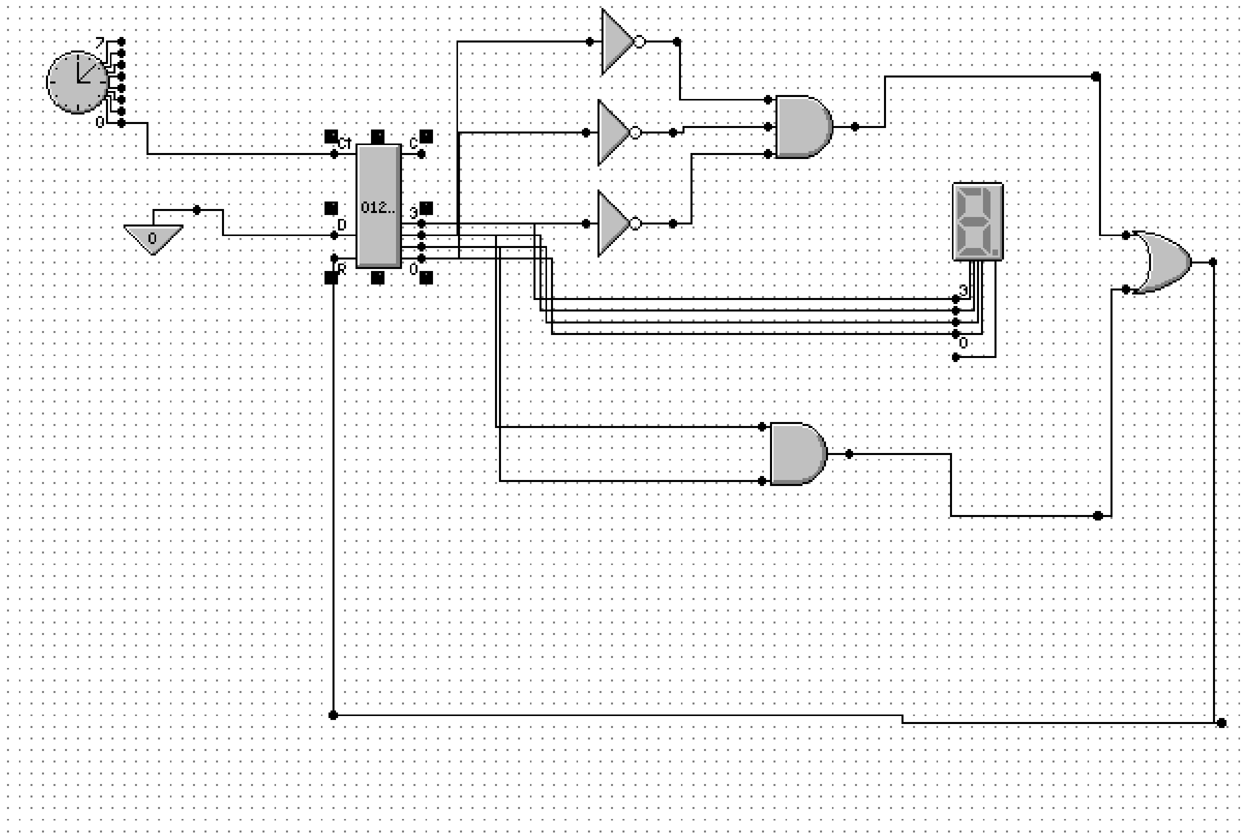
Figure 11. K-map for an asynchronous 4-bit counter.

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00	1 (0)	0 (1)	0 (3)	0 (2)
01	0 (4)	0 (5)	X (7)	1 (6)
11	X (12)	X (13)	X (15)	X (14)
10	X (8)	X (9)	X (11)	X (10)

Reset circuit = $Q_2Q_1 + \bar{Q}_2\bar{Q}_1\bar{Q}_0$

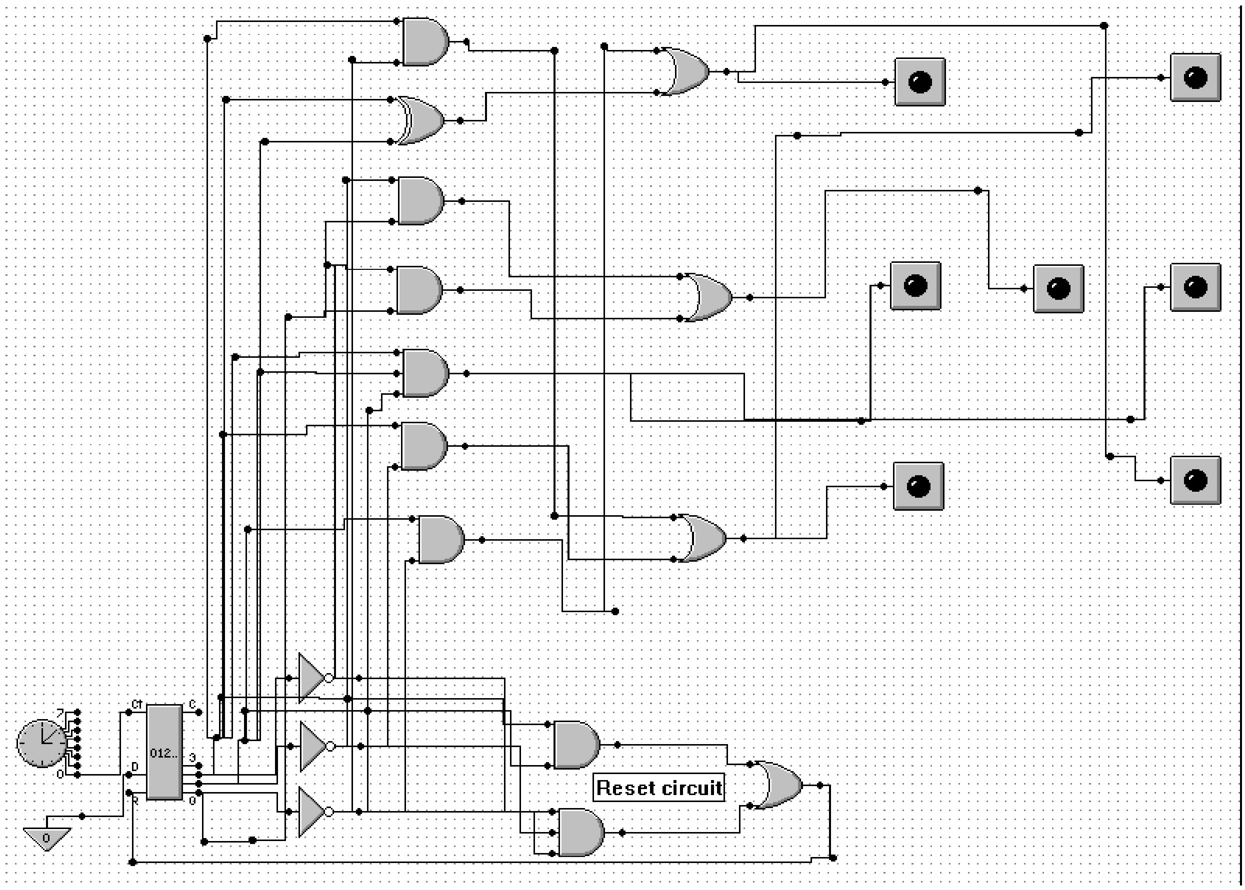
We have to ensure that $reset = 1$ when 0 and 6 are present. Also the initial value of the counter is 1.

Figure 12. The reset circuit connected to the counter.



Part 3:

Figure 13. The first circuit combined with the counter.



Note: The initial value of the counter is 1. It starts from 1, ends with 5, and then resets to 1. I will attach the files so you can run it by yourself.