Template Week 2 – Digital Circuits

Student number: 552517

Assignment 2.1: Parking lot

Which gates do you need?

To make this circuit, we need 2 AND logic gates, as the red sign on the street will show up when: **Parking lot 1 AND Parking lot 2 AND Parking lot 3** are occupied. The table is completed with 0 as false (no red light) and 1 as true (red light).

Parking lot 1	Parking lot 2	Parking lot 3	Result (full)
0	0	0	0
0	0	1	0
0	1	0	0
1	0	0	0
1	1	0	0
1	0	1	0
0	1	1	0
1	**1**	**1**	**1**

Assignment 2.2: Android/iPhone

Which gates do you need?

In this circuit we are going to need an XOR (exclusive or) logic gate. The XOR gate will output a high signal only when the employee has made an exclusive choice, i.e., chosen either an Android phone or an iPhone, but not both.

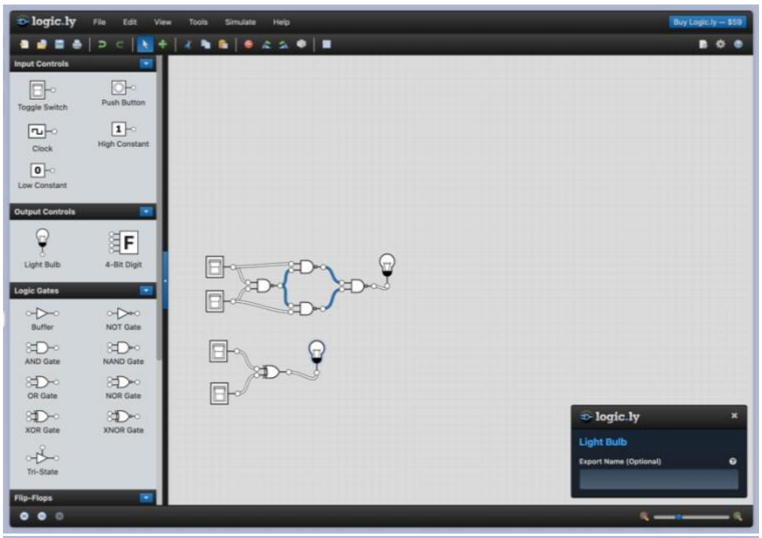
Android phone	iPhone	Result (Phone in possession)
0	0	0
0	**1**	**1**
1	**0**	**1**
1	1	0

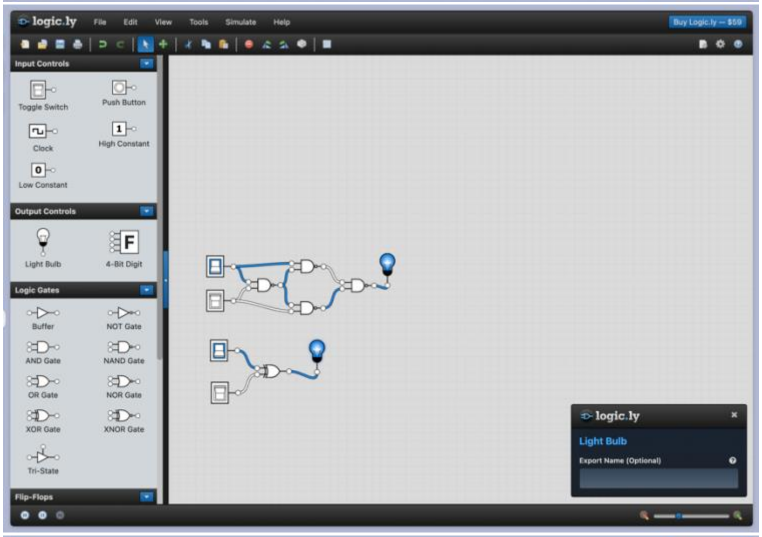
Assignment 2.3: Four NAND gates

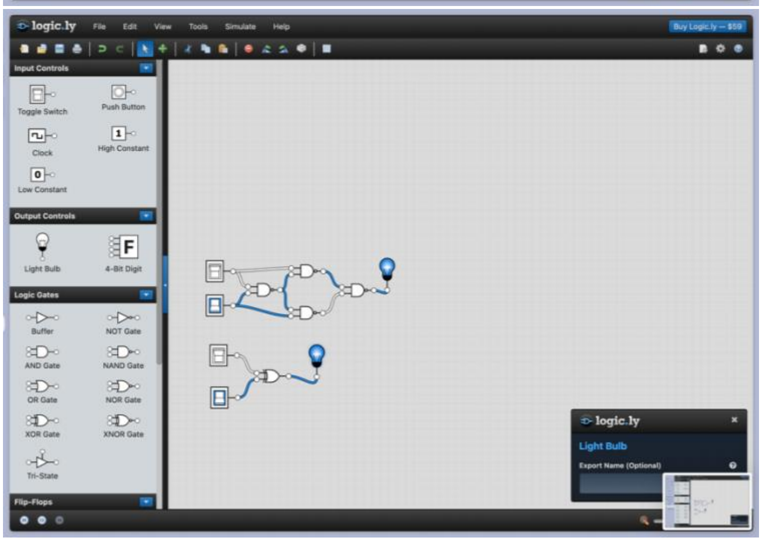
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

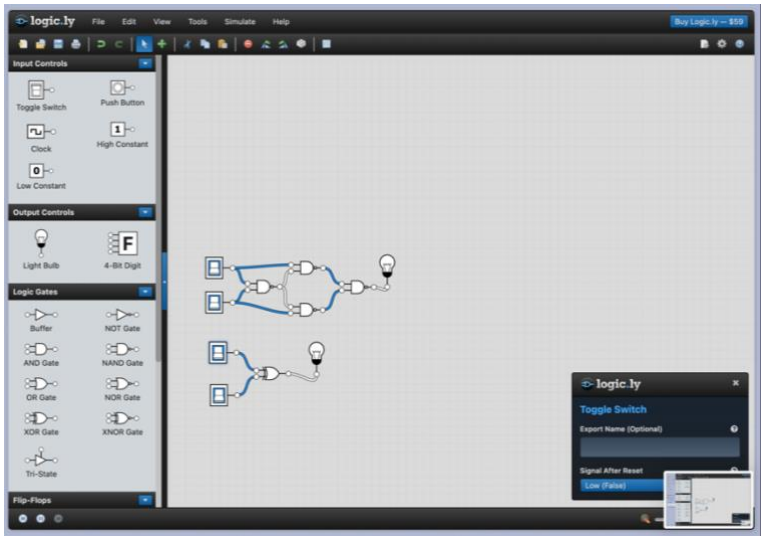
Instead of this circuit we could create 1 XOR circuit with 2 inputs. It would produce the same output: output is high (1) when only 1 of the 2 inputs is high (1).

On the following screenshots you can see that the 4 NAND gate circuit, that was mentioned before, produces the same output as a simple XOR gate. This was done using an online tool logic.ly.



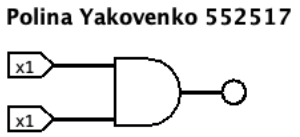






Assignment 2.4: Getting to know Logisim evolution

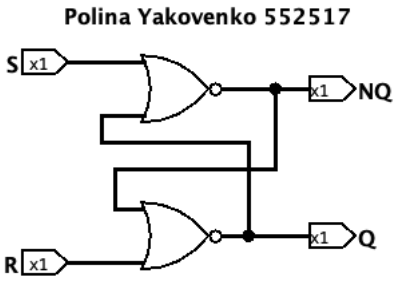
Screenshot of the design with your name and student number in it:



In this example we have created a simple AND gate, which works the following way: the output value is high (1, the lightbulb lights) only when both of the input values are high (1). If any of input values, or both, are low (0), the output value is low (0, no light) as well.

Assignment 2.5: SR Latch

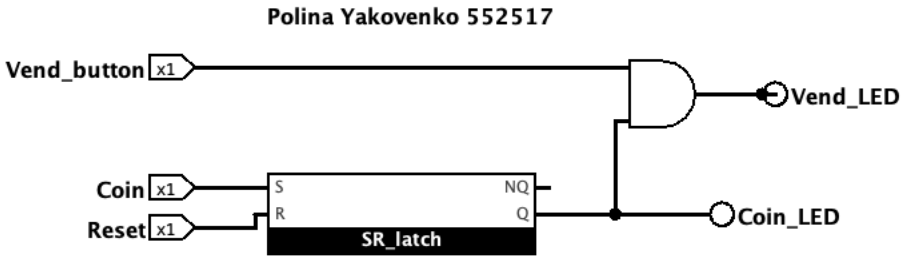
Screenshot SR Latch in Logisim with your name and student number:



In this example we have created a Set-Reset (SR) latch, a digital circuit that can store one bit of information, which consists of two cross-coupled NOR gates. When the Set input is activated, the output Q becomes '1,' and when the Reset input is activated, it resets the output to '0.' The purpose of SR latch is to "remember" its state.

Assignment 2.6: Vending Machine

Screenshot Vending Machine in Logisim with your name and student number:



In our vending machine circuit with an SR latch, the SR latch is controlled by two inputs: "coin" and "reset." When a coin is inserted, the SR latch may be set, causing its output (Q) to go high, and activating the Coin LED. If the "reset" input is triggered, it resets the latch, turning off the Coin LED. The vend button, when pressed and in conjunction with a high Q output from the SR latch, activates an AND gate, lighting up the vend lightbulb, indicating that the vending process can proceed.

In this design the vending action occurs when both the latch is set and the vend button is pressed.

One simpler approach to achieve a vending machine circuit is to use a single D flip-flop or a T flip flop, because of the following reasons: D flip-flop or a T flip flop require only a single input, which leads to a more straightforward design; SR latch can lead to potential issues when S and R are activated simultaneously, which does not occur with T and D flip flops.