Faculty of Science, Engineering and Technology

Computer Systems

Week 2

Overview

Building a 4 bit adder and working with Flips Flops.

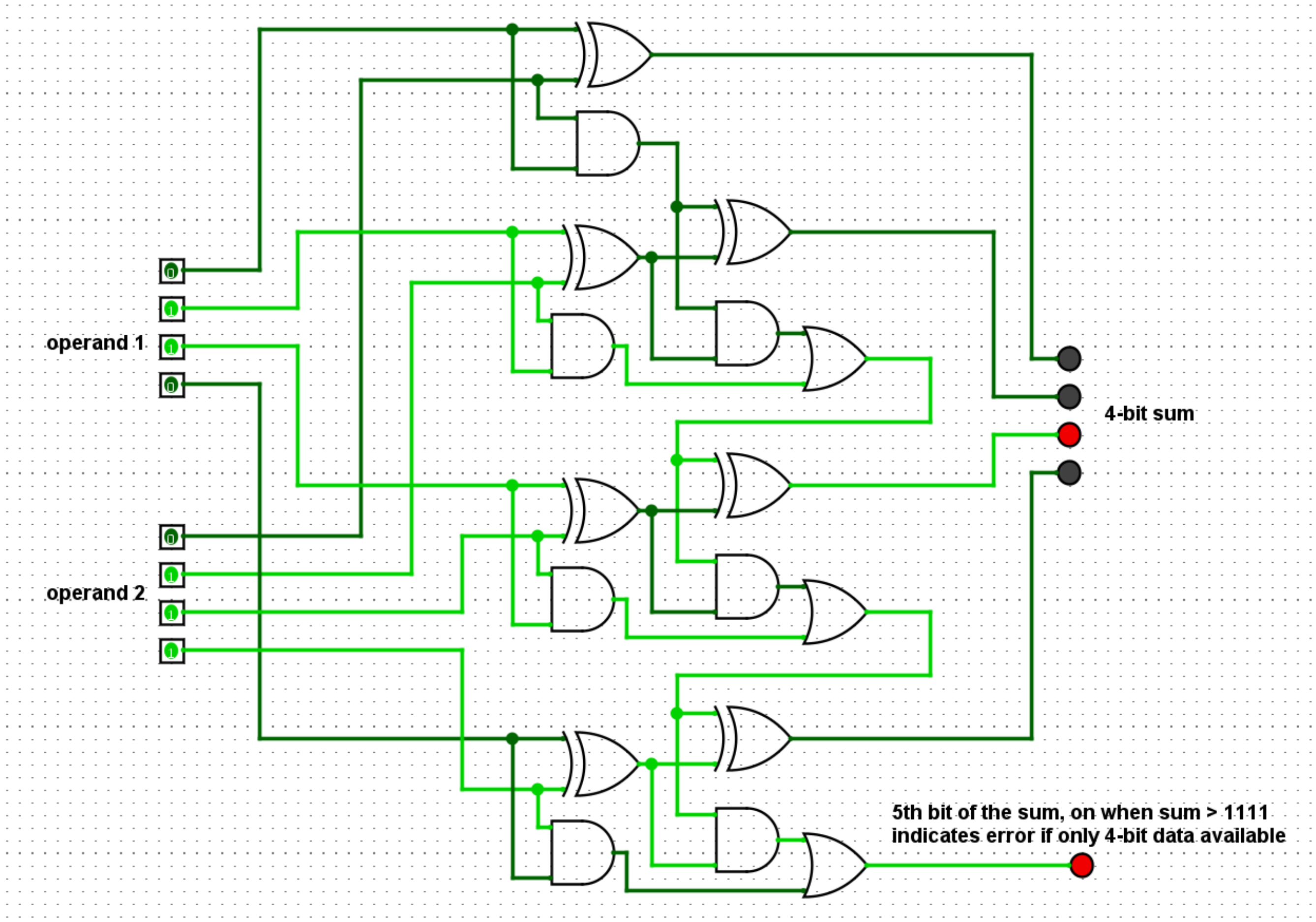
Name: Le Quang Hai

Student ID: 104175779

Part 1: Building a 4-bit adder

A 4-bit adder sums together two 4-bit binary numbers. Each bit of each number is represented by a binary on/off pin.

Screenshot:


0
1
operand 1
1
0
0
1
operand 2
1
1
4-bit sum
5th bit of the sum, on when sum > 1111
indicates error if only 4-bit data available

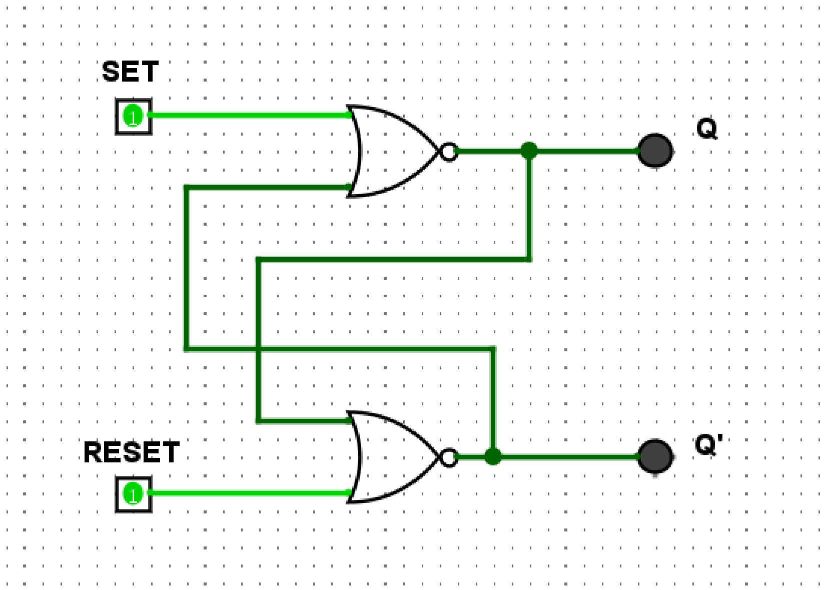
Input A	Input B	Output
0101	0000	0101
0101	0001	0110
0101	0010	0111
0101	0011	1000
0101	0100	1001
0101	0101	1010
0101	0110	1011
0101	0111	1100
0101	1000	1101
0101	1001	1110
0101	1010	1111
0101	1011	(1)0000
0101	1100	(1)0001
0101	1101	(1)0010
0101	1110	(1)0011
0101	1111	(1)0100

Part 2: Storing bits with Flip Flops

Any computing hardware that seeks to perform meaningful calculations using bits requires circuitry to store them - that is, circuits that can maintain a given state.

Wire up your own R-S Flip Flop using a pair of 2-input NOR gates (**do not use Logisim's S-R Flip-Flop!**). You should have 2 input pins, one for the "*Set*" pin, and one for the "*Reset*", and two output LEDs: *Q* and *Q'* .

When you've finished wiring it up, set both input pins to 1. The LEDs should both be dark (**Export your circuit as an image and include it in your submission document.)**



Set the pins *in the following order* and record the states for Q and Q’

Set	**Reset**	**Q**	**Q’**
1	**0**	**0**	**1**
1	1	0	0
0	1	1	0
1	1	0	0

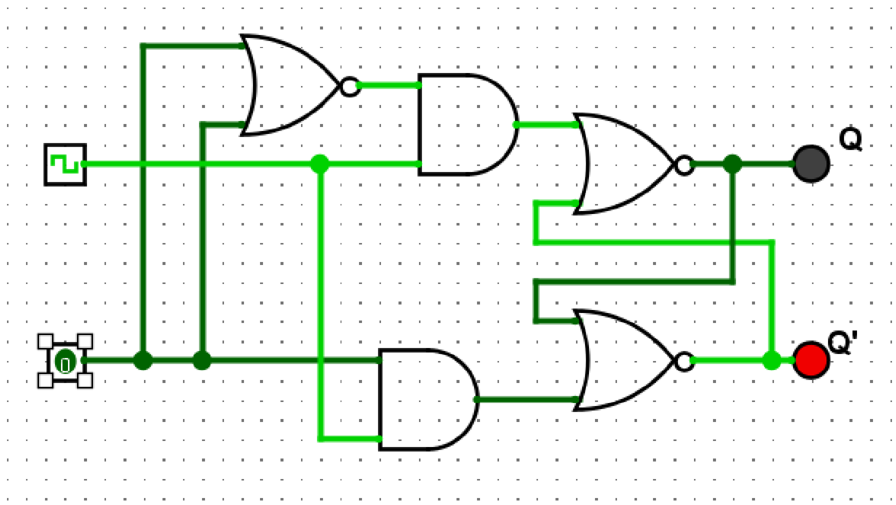
Describe in a sentence, the behavior of the circuit when one of the inputs is 1 (but not both) and why this is useful for digital circuit design.
→ When one of the inputs of an RS flip-flop is 1 (but not both), the circuit will change its output state to the corresponding value of the input and remain in that state until the other input changes, which is useful for storing a binary value.

What do you notice about the two times you set both inputs to 1. Briefly explain what is happening here and why this is an issue for digital circuit design ?

→ When both inputs of an RS flip-flop are set to 1, the output of the flip-flop can be set randomly before settling to a stable state, leading to potential data loss or corruption.

So the unclocked R-S flip flop has issues. Let's talk about the D Flip-Flop then. Wire up a D Flip Flop using AND gates and NOR gates, with output LEDS labeled Q and Q’.

Export your circuit as an image and include it in your submission document.



Explore the behavior of the D Flip Flop by filling out the following truth table

Clock	Pin	Q	Q'
0	0	0	1
0	1	0	1
1	1	1	0
1	0	1	0

Briefly explain the behavior of a D Flip Flop and how it is useful for digital circuit design.

→ Whenever the clock reaches the '1' state, the value of the pin(data) is stored and will not change until the next time the clock reaches the '1' state. This is useful for storing data and avoid asynchronous misbehavior

What is the role of the clock ? How does it impact the changing of state of Q and Q' ?

The clock make sure data can only be stored after a specific amount of time

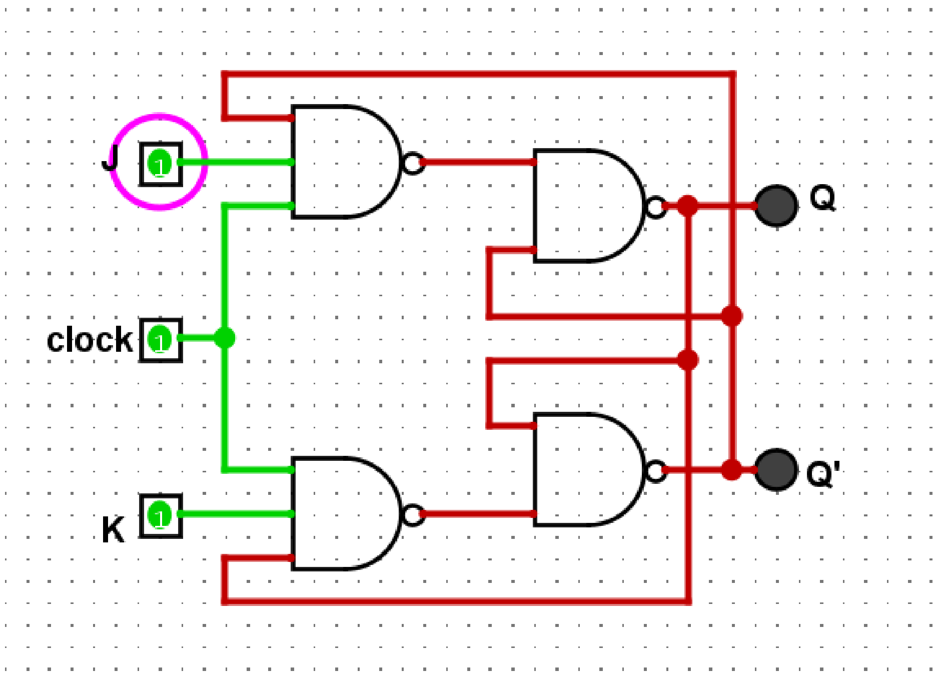
→ Q and Q' can only be changed when clock reaches '1'

Why is it generally preferred over the R-S Flip Flop ?

→ D Flip Flop does not have the indeterminate state and allows synchronous operation

Wire up a J-K FF using NAND gates. Two of your NAND gates will need to deal with three inputs.

Logisim will not be able to simulate this circuit, but export your completed circuit as an image and include it in your submission document.



19. Complete and include this truth table for JK Flip Flops in your submission document.

J	K	Q (when clocked)	Q’ (when clocked)
0	0	Qn	Q’n
1	0	1	0
0	1	0	1
1	1	NOT(Qn)	NOT(Q’n)

How can a J-K Flip Flop be made to behave like a D Flip Flop ?

→ To do that we connect the one input to J, that input also go through a NOT gate and connects to K

How can a J-K Flip Flop be made to behave like a toggle (T Flip Flop) ?

→ When J and K are both set high