Faculty of Science, Engineering and Technology

Computer Systems

Week 3

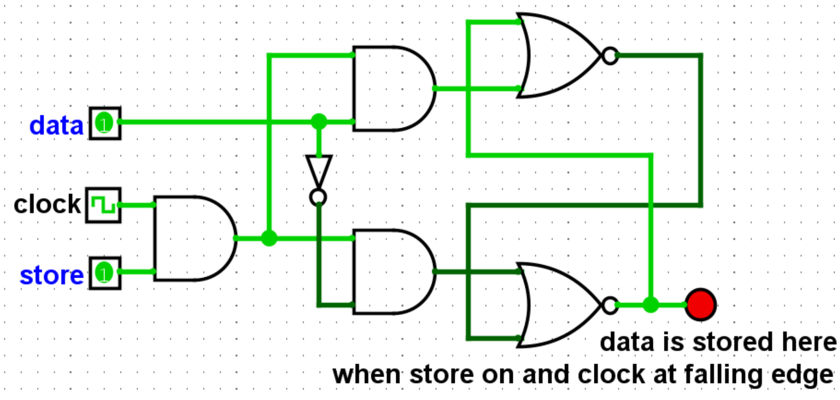
Overview

In this laboratory session we start using Flip Flops to build useful things like registers, counters and shift registers.

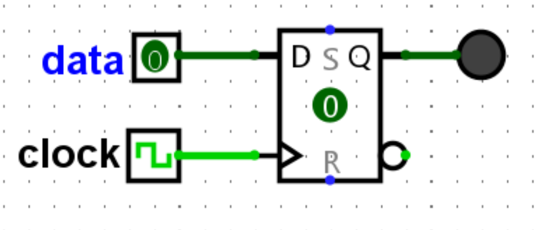
Name: Le Quang Hai

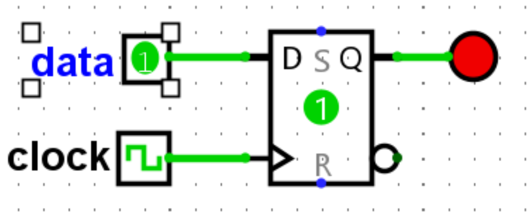
Student ID: 104175779

1. Wire up a register in Logisim.

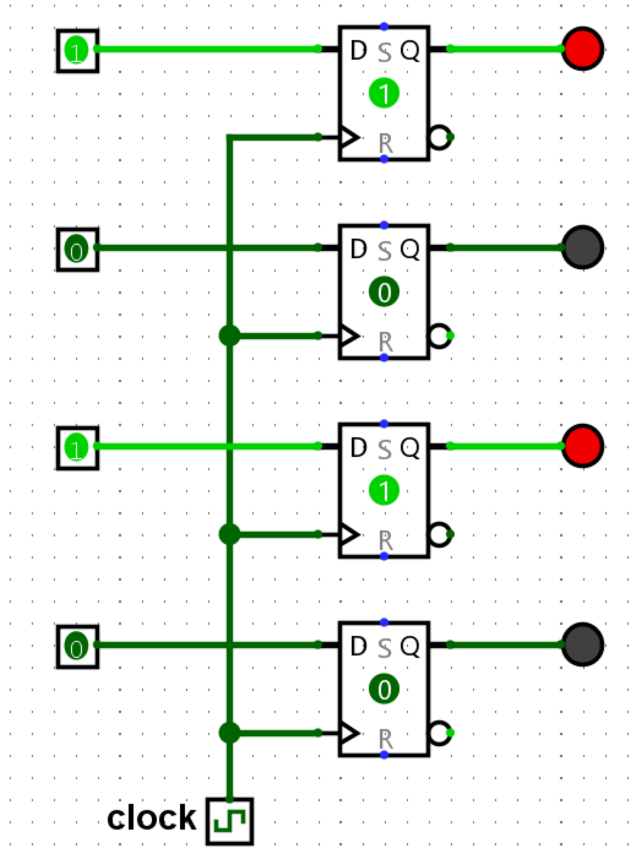


2. We're not going to wire our own Flip Flops anymore. Familiarize yourself with Logisim's D Flip Flop. Bring one into your canvas, and connect up an input pin, and a clock, and connect an LED to the output "Q". Verify it works as you expect





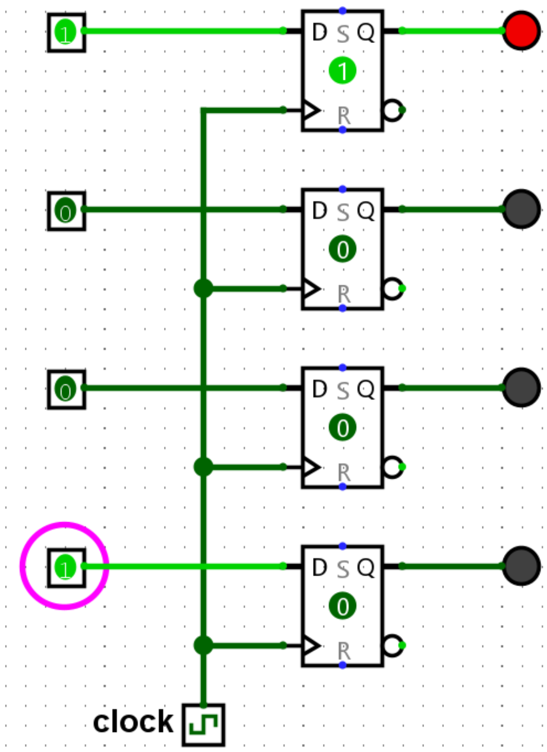
3. Wire-up a 4-bit big-endian register with D Flip Flops in Logisim. Do this by using 4 pins for each input, and connect 4 LEDS to the output.

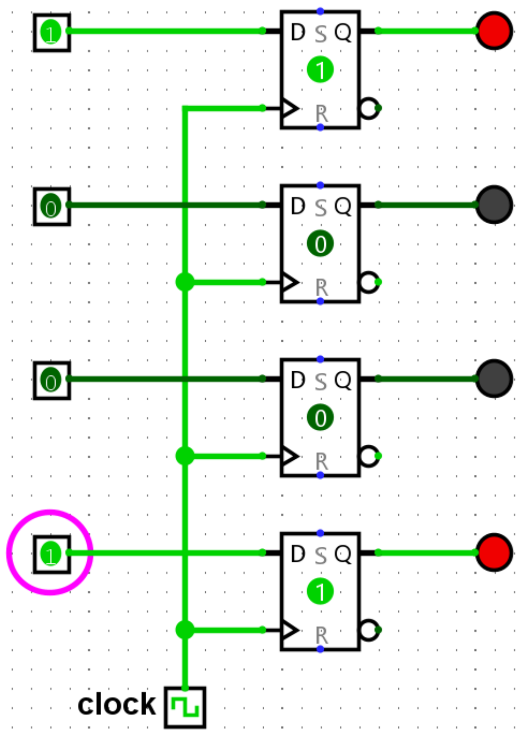


In this scenario, big or little endian does not really matter since the register store bit by bit so the circuit can either be big or little endian

4. Demonstrate your register to your lab demonstrator by showing them different combinations of input bits, and how this changes the output when the clock pulses.

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





5. Use your register to fill out the following test schedule:

Ox	Input Binary	Output Binary
0	**0000**	0000
1	0001	**0001**
2	0010	**0010**
3	**0011**	**0011**
5	**0101**	**0101**
A	**1010**	**1010**
B	**1011**	**1011**
C	**1100**	**1100**
D	**1101**	**1101**
E	**1110**	**1110**
F	**1111**	**1111**

6. Answer the following questions:

6.1. Name one crucial role (hardware) counters play in modern computing architectures?

→ In modern computing architecture, counters can play a role as real-time monitors (time calculator)

6.2. Describe in a few sentences how a ripple counter works. How does the "ripple" occur?

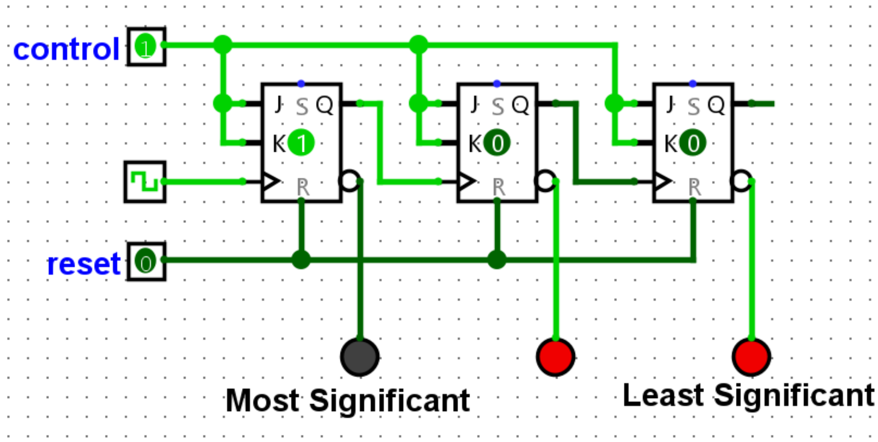
→ In a ripple counter, the output of one (JK)flip-flop is the input of the next (JK)flip-flop. This construction makes the next flip-flop change state at the pace of half the previous flip-flop.

→ **Explain:** 1 period involves 2 changes in state therefore:

1 period of previous FF = 1 change in following FF = ½ period of following FF

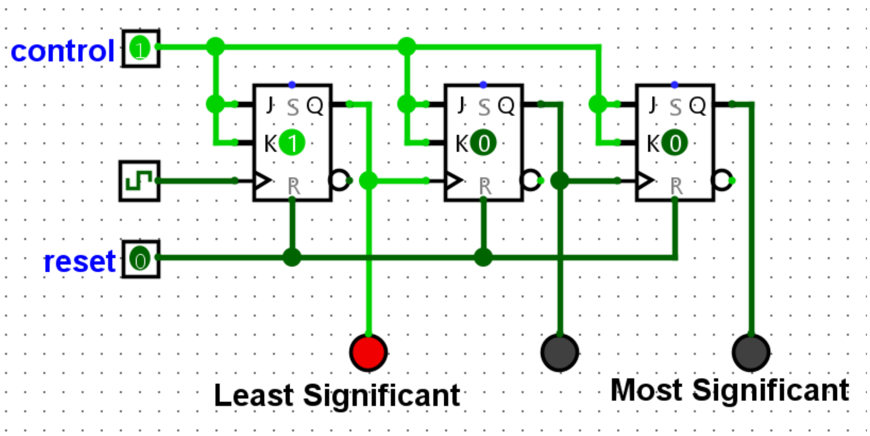
7. Build a big-endian 3-bit ripple counter out of JK Flip-Flops. Your counter should count from 000 to 111. Use LEDs to show the output "Q" from each Flip Flop. For this to work, you will also need to set your JK Flip Flops to Trigger with the clock's Falling Edge instead of its Rising Edge

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



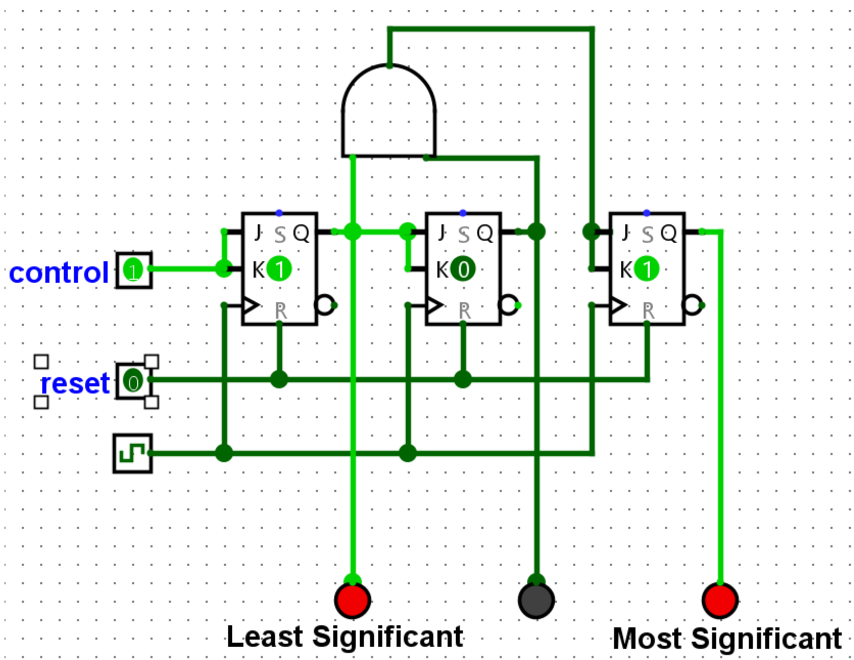
8. Now build a big-endian 3-bit “count down” counter, that counts from 111 to 000.

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

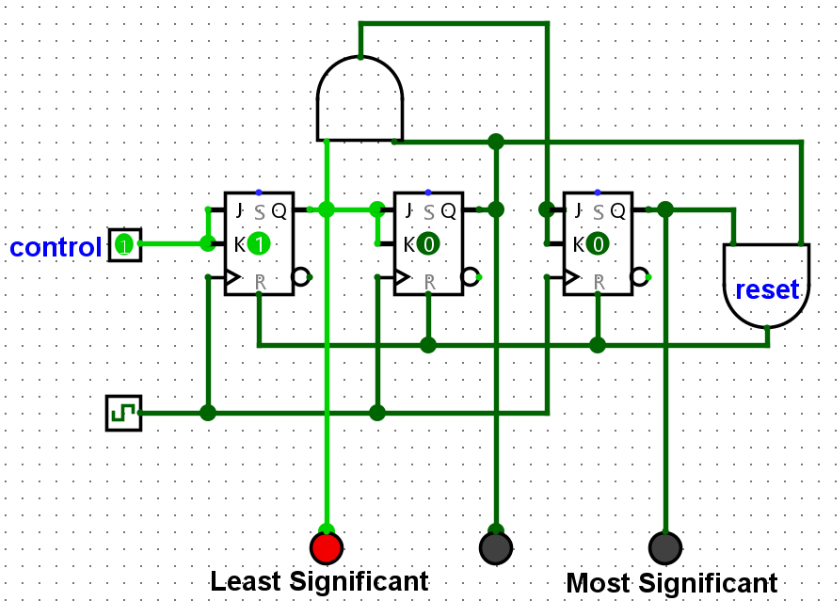


9. Take your original counter and modify it so it now counts from 0 to 111 using a common clock. That is, each flip flop receives a clock pulse at the same time

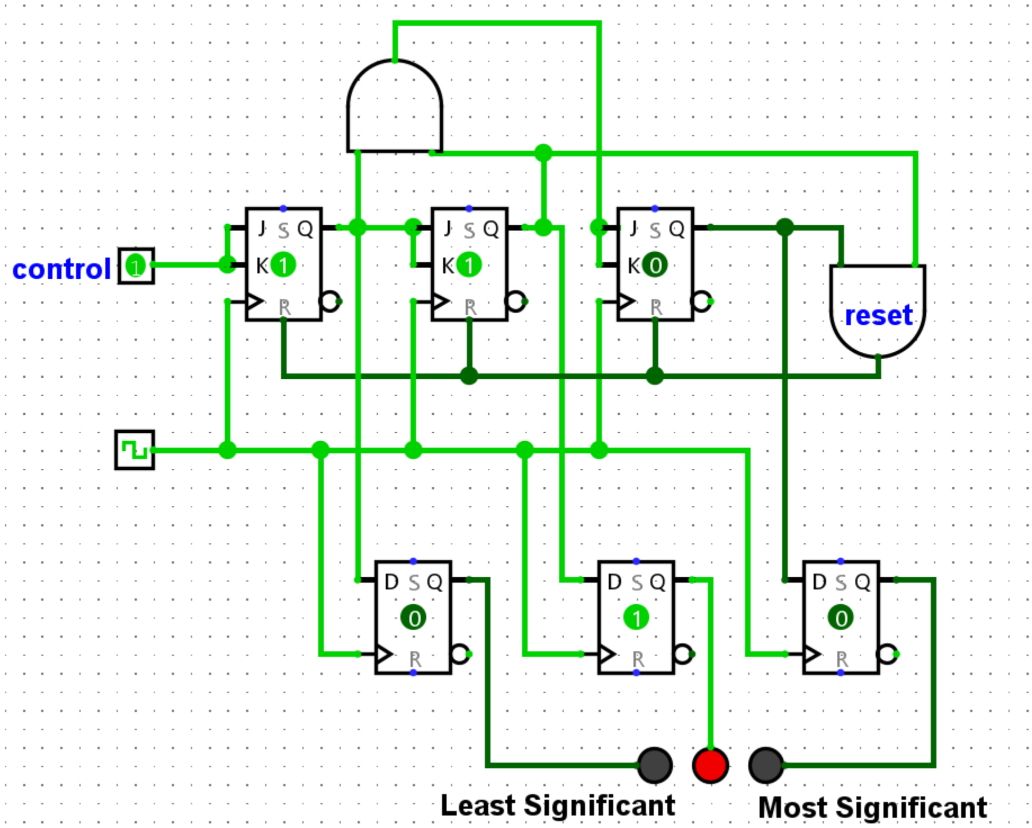
Export your circuit as an image and include it in your submission document.



10. Now modify your clock from Step 9 so it counts from 0 to 5 (i.e, MOD 6), and then wraps around back to 0.



11. The circuit in Step 10 goes into a momentary illegal state (i.e, it displays the binary string 110 due to the delay between detection of the limit, and the eventual reset back to 0. In the lecture we discussed using D Flip Flops as a buffer, to hold the output state one extra clock pulse before showing (allowing time for any resetting to occur first).



Modify your counter so that it resets after 5 (101) back to 0 (000) without the momentary illegal state. Why is handling such things important ?

→ Because we want to prevent unexpected behavior to make sure no error occurs when running (by forcing the output to arrive synchronously)

12. Display your counter output using the HEX Digit Display. Note that the Logisim HEX Display uses a single pin input with a 4 bit width. That means a 4 bit integer is expected along a single wire. Because our wires are carrying only 1 bit at a time, we will require a “Splitter” (in reverse) to combine multiple bit streams into a 4-bit “wire bundle” that is fed into the HEX display.