Faculty of Science, Engineering and Technology

Computer Systems

Week 3

Student Name: Lau Ngoc Quyen

Student ID: 104198996

Overview

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

Purpose: To consolidate your knowledge of Flip Flops, and how they can be used.

Task:

Time: This lab is due by the start of your week 3 lab.

Assessment: This lab is worth 1% (up to a maximum of 5%) of your assessment for this unit, and only if demonstrated to your lab demonstrator in the week it is due.

Resources:

- Flip Flop tutorials
  - Intro to Flip Flops

Submission Details

You must submit the following files to Canvas:

- A document containing all required work as described below.

Instructions

1. Start Logisim and open a new canvas

Part 1: 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. In lectures we discussed Flip Flops, which are simple block circuits designed to maintain a particular binary state, and transition between binary states depending on the inputs given.

2. Review this week's lecture slides, and if needed, also take a look at the quick video tutorials linked under resources at the beginning of this lab sheet.
3. Create a clear canvas.
4. Using the lecture slides as a guide, 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'* .
5. When you've finished wiring it up, set both input pins to 1. The LEDs should both be dark (assuming you've wired it correctly).

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

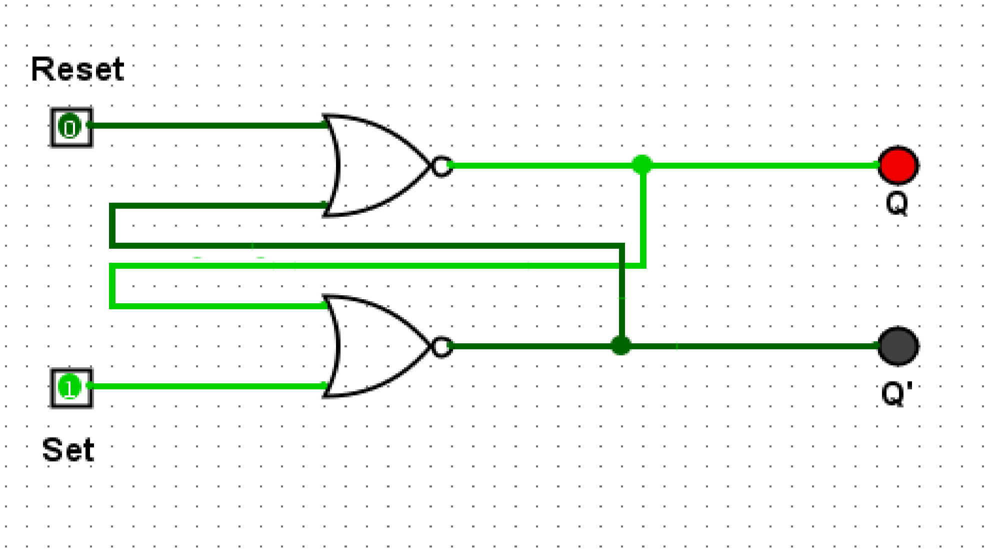

***Image 1:* Reset = 1; Set = 0**

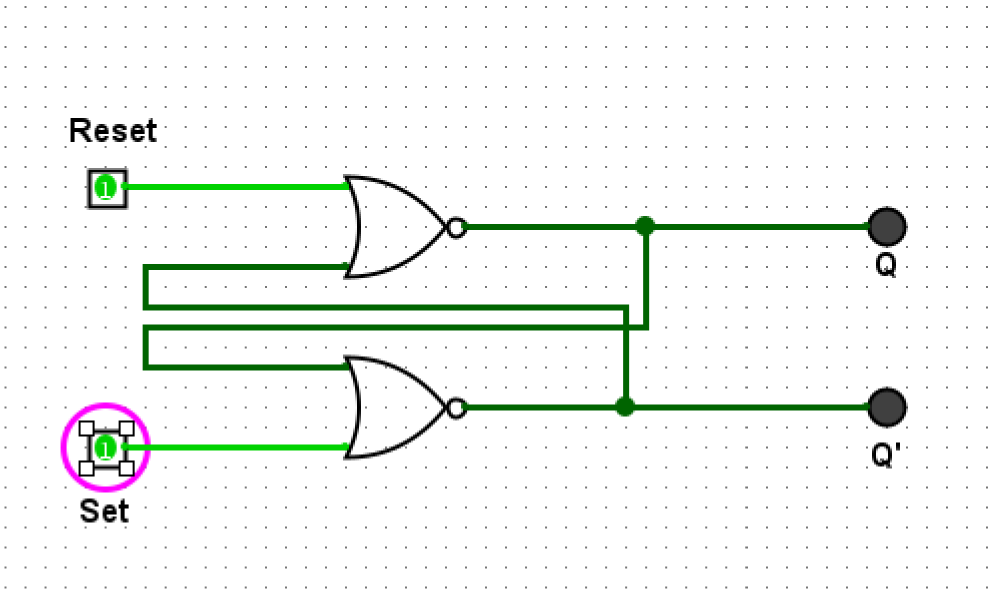

***Image 2:* Reset = 1; Set = 1**

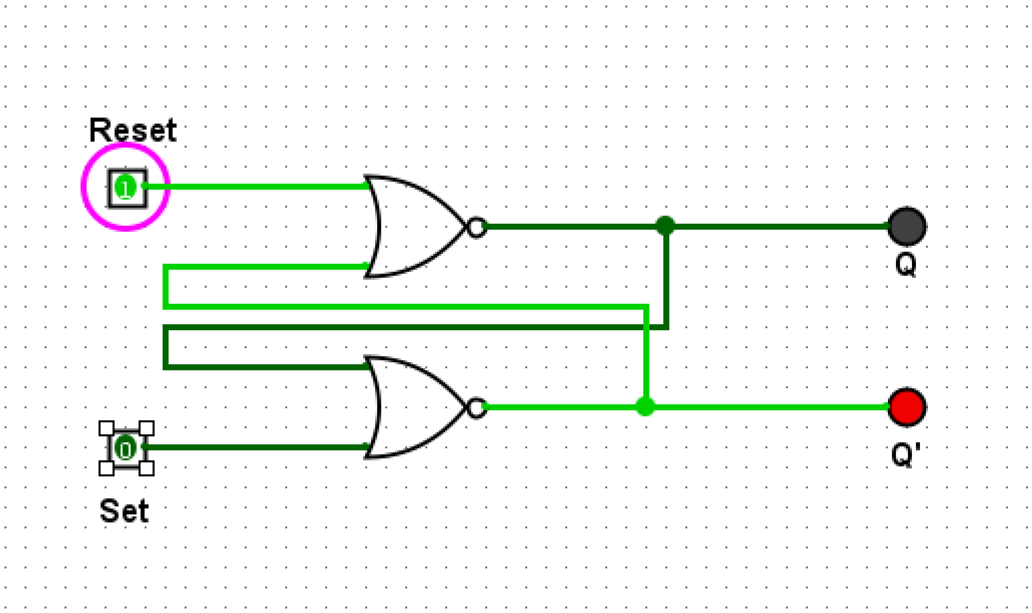

***Image 3:* Reset = 1; Set = 0**

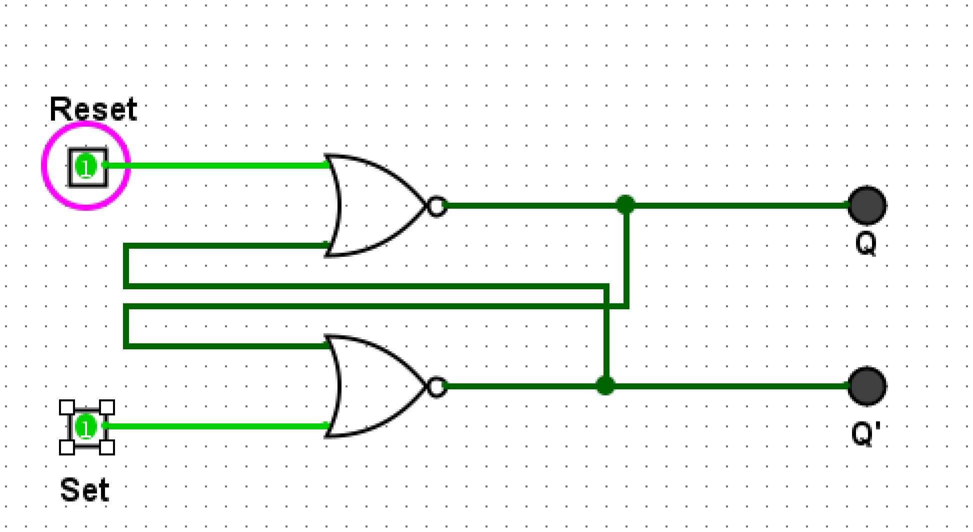

***Image 4:* Reset = 1; Set = 1**

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

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

7. Describe in a sentence, the behaviour of the circuit when one of the inputs is 1 (but not both) and why this is useful for digital circuit design.

   - ***When one input is 1 (Set = 1, Reset = 0), the output Q is set to 1 (Q' = 0). When Set = 0 and Reset = 1, Q is reset to 0 (Q' = 1). This is useful because it stores a stable state (0 or 1), essential for memory in digital circuits.***

8. 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 are 1, the RS flip-flop enters an invalid state where both Q and Q' could be 0, which breaks the rule that they should be opposites. This creates confusion, leading to unreliable circuit behavior.***

Discuss 7 and 8 with your lab demonstrator and provide your answer in your submission document, along with the truth table in Step 6.

9. So the unclocked R-S flip flop has issues. Lets talk about the D Flip-Flop then. Review the lectures on the D Flip-Flop, and when you feel comfortable, wire up a D Flip Flip using AND gates and NOR gates, with output LEDS labeled Q and Q’.

- For this you will have only 1 input pin, as well as a clock input. The clock can be pulsed on and off by clicking it with the operation pointer (the finger in the top left of screen), or you can simple enable clock ticking from the menu (under “Simulate”).

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

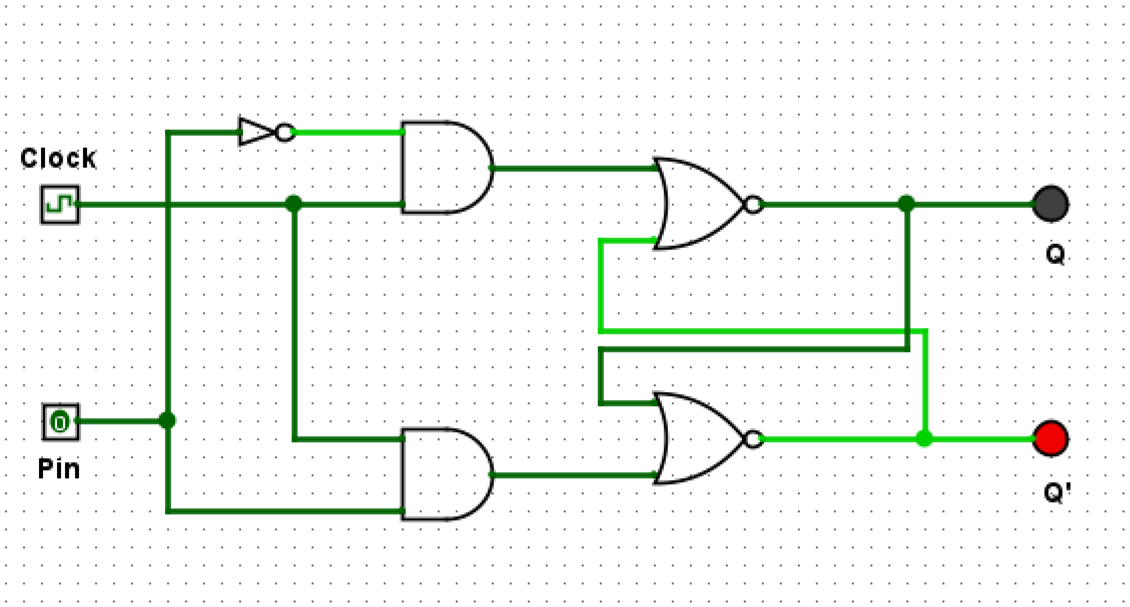

***Image 4:* Clock = 0; Pin = 0**

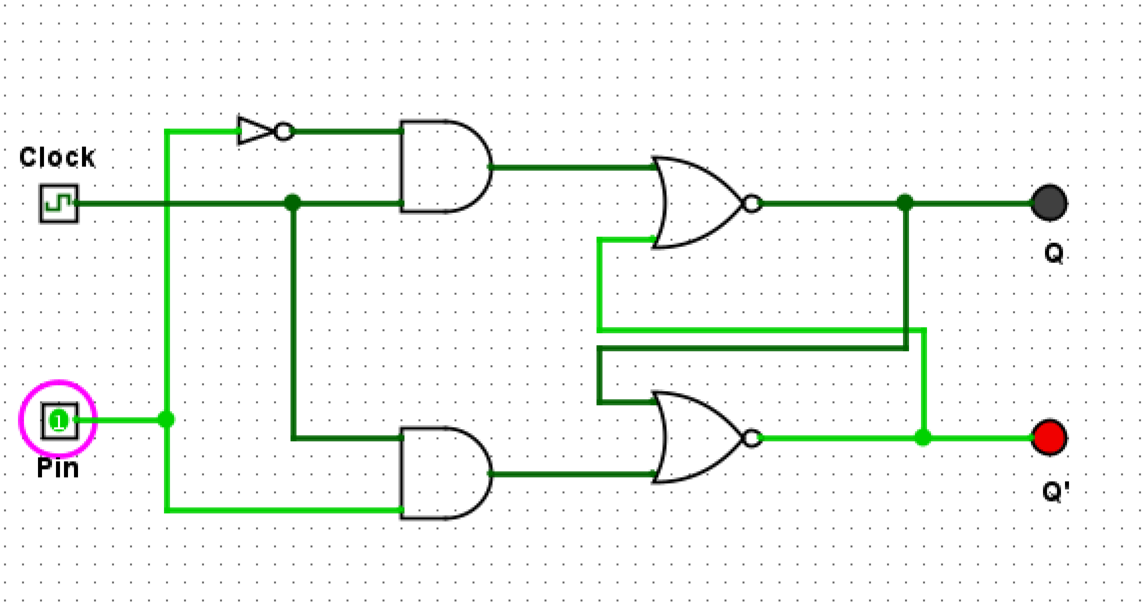

***Image 5:* Clock = 0; Pin = 1**

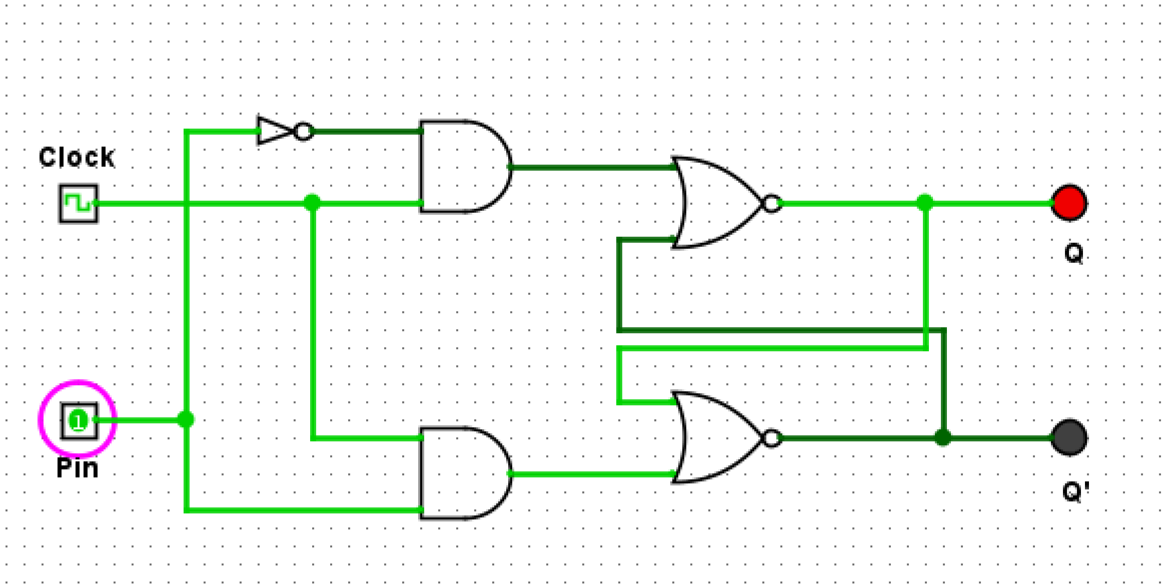

Image 6: ***Clock = 1; Pin = 1***

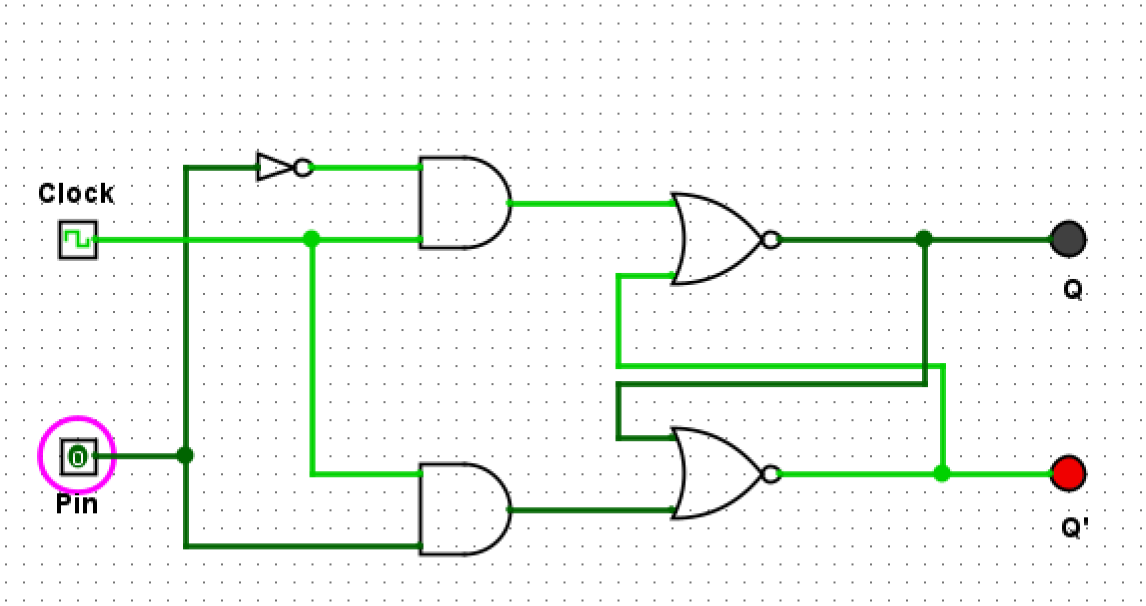

Image 7: ***Clock = 1; Pin = 0***

10. Explore the behaviour 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	0	1

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

    - ***A D Flip-Flop captures the value of the input (Pin) on the rising edge of the clock and holds it until the next clock pulse. It ensures stable data storage, making it ideal for synchronization in digital circuits.***

12. What is the role of the clock ? How does it impact the changing of state of Q and Q’ ?

    - ***The clock controls when the Flip-Flop can change its state. Changes in Q and Q' only occur when the clock signal is active (rising edge or specific clock pulse), making the system more predictable and reliable.***

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

    - ***The D Flip-Flop avoids the invalid state issue of the R-S Flip-Flop (when both inputs are 1), providing a more stable and simple design, as it has only one data input (Pin).***

Discuss 11 -13 with your lab demonstrator and provide your answer in your submission document, along with the truth table above.

14. J-K Flip Flops are like your general purpose Flip Flop because they are programmable. Review the video on JK Flip Flops, and when you’re feeling comfortable, 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.

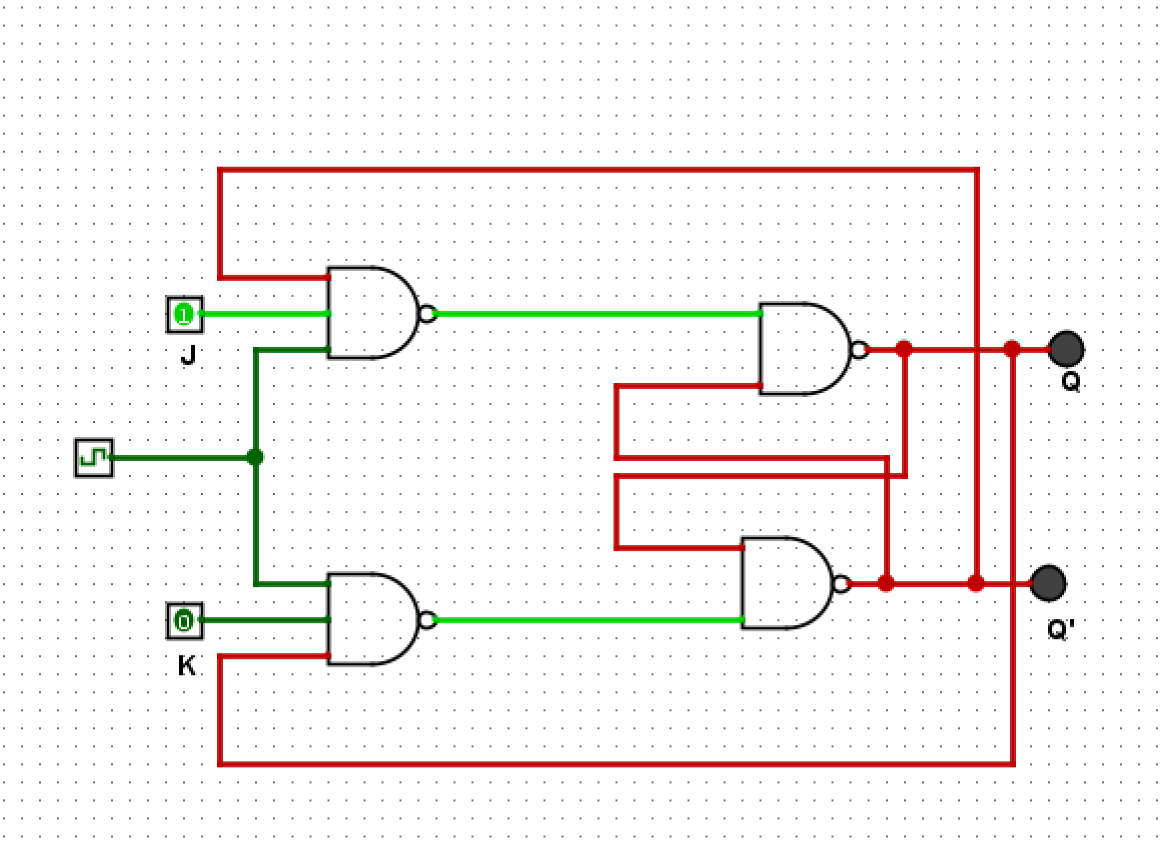

Image 8: JK Flip Flops

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

J	K	Q(When Clocked)	Q'(When Clocked)
0	**0**	Q	Q'
1	0	1	0
0	**1**	0	1
1	**1**	Toggle	Toggle

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

- ***Either J or K is On***

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

- ***Both J or K is On***

Discuss these questions with your lab demonstrator and provide your answer in your submission document, along with the truth table in Step 15.

Part 2 - Register this !

18. Registers are just adjacent Flip-Flops that store collections of bits. You're about to wire up a register in Logisim, but first review the lecture slides, and if needed, take a look at the resources above to remind yourself how Flip Flops work. We're going to work with D Flip Flops here.
19. We're not going to wire our own Flip Flops anymore. We're going to use Logisism's. Familiarise 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". Have a play and verify it works as you expect (ask your lab demonstrator for assistance if needed).
20. Now 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.
21. When complete, 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.**

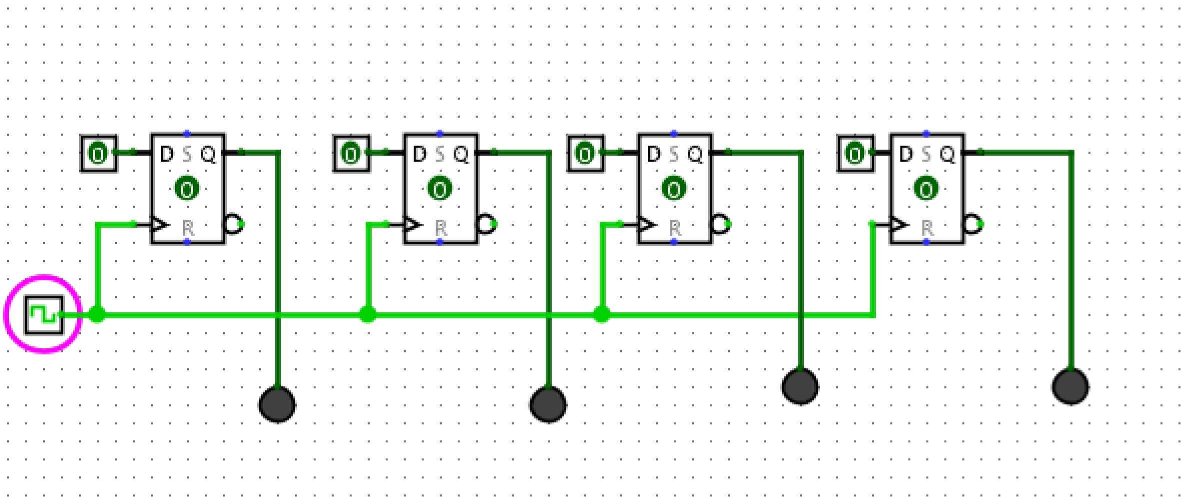

***Image 9:* 0000**

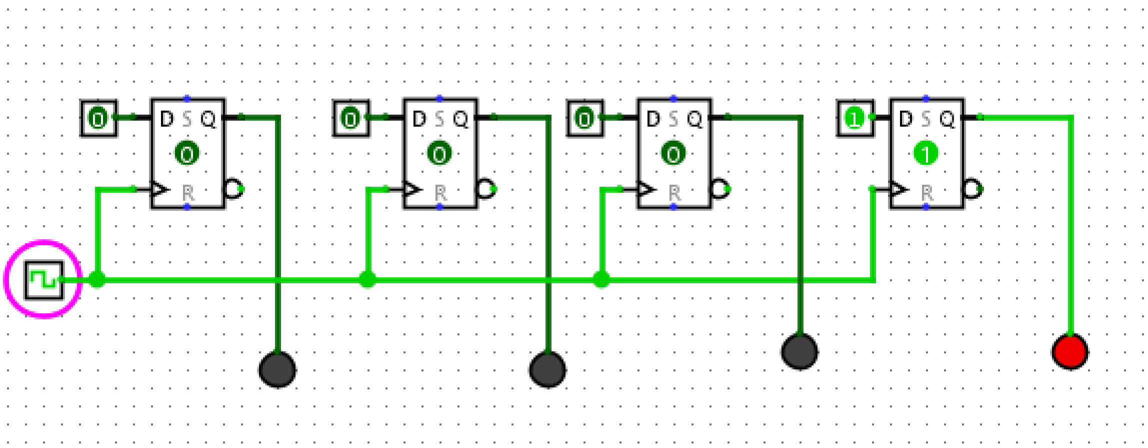

***Image 10:** 0001*

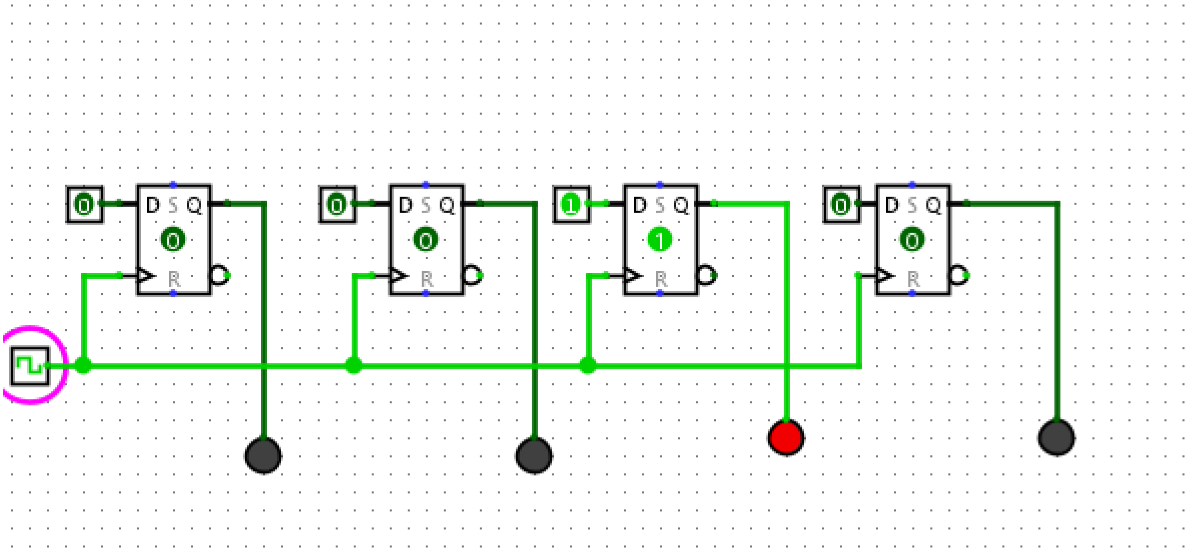

***Image 11:** 0010*

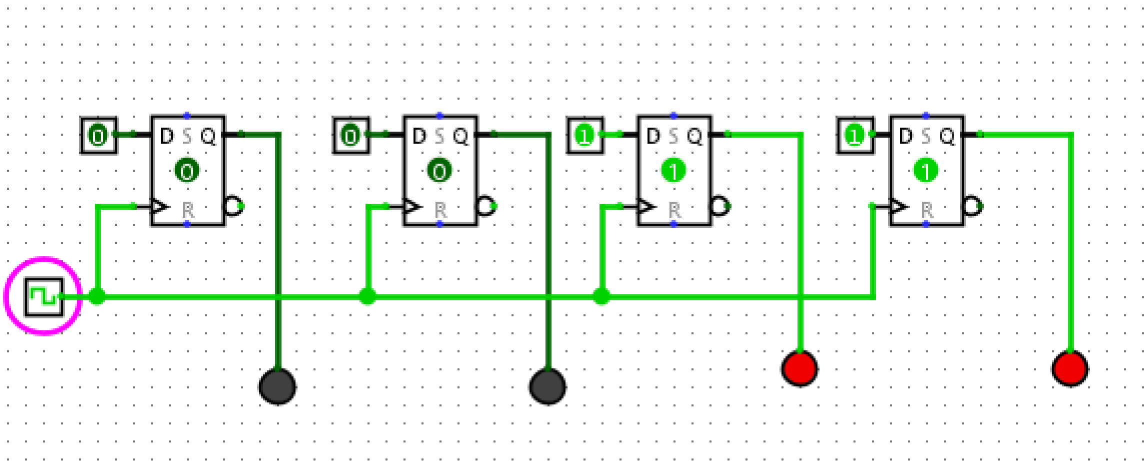

***Image 12:* 0011**

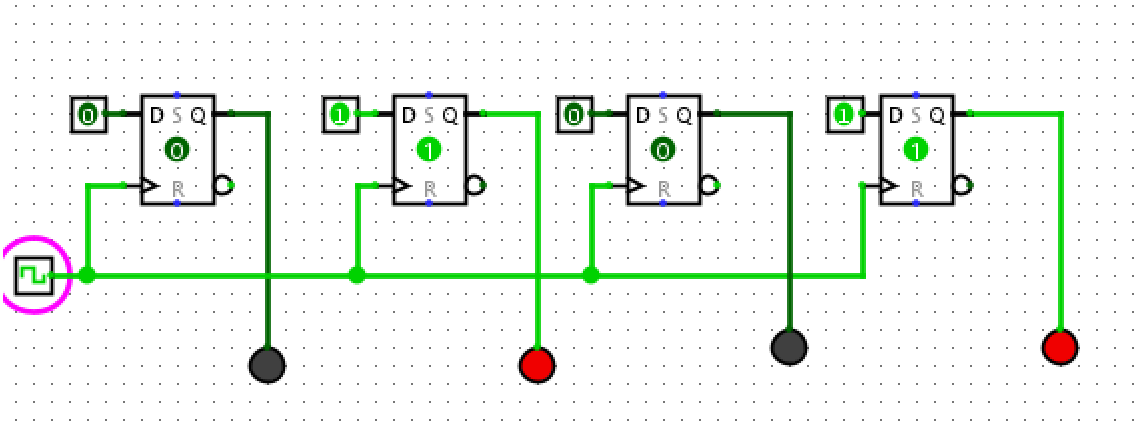

***Image 13:* 0101**

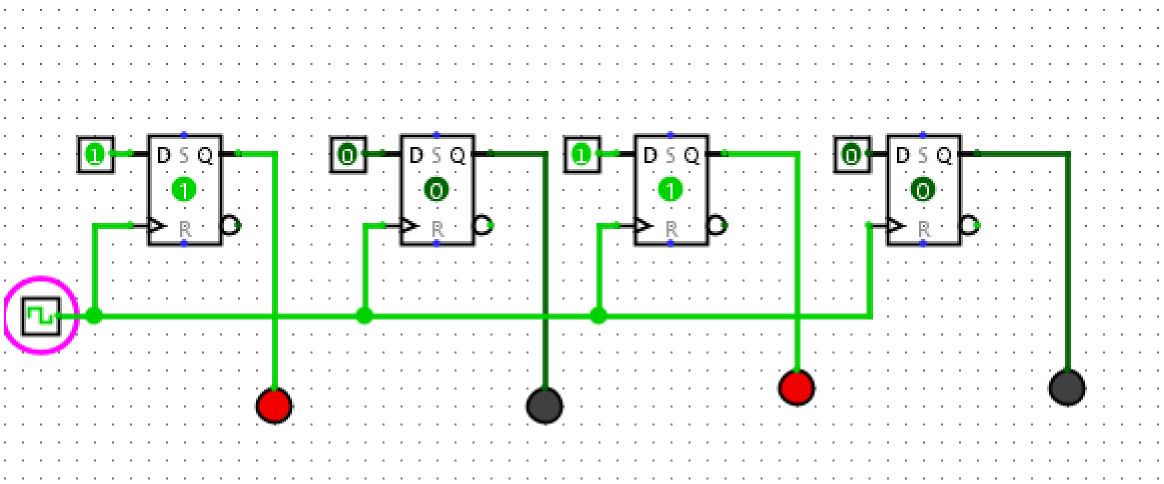

***Image 14:* 1010**

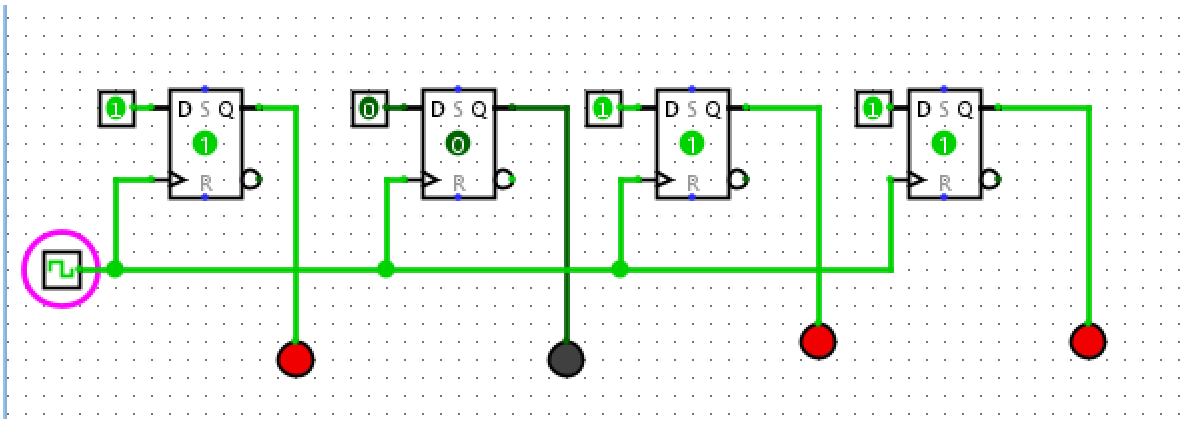

***Image 15:* 1011**

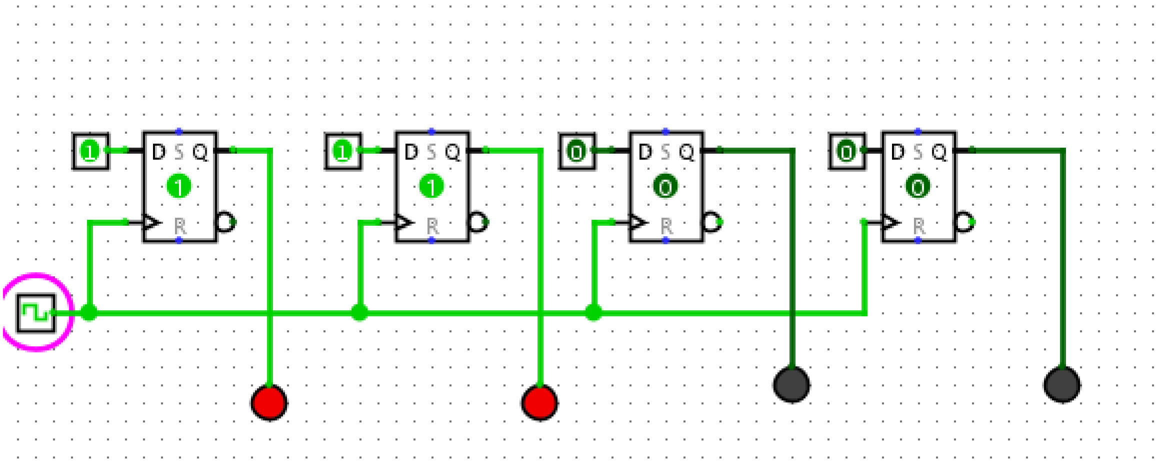

***Image 16:* 1100**

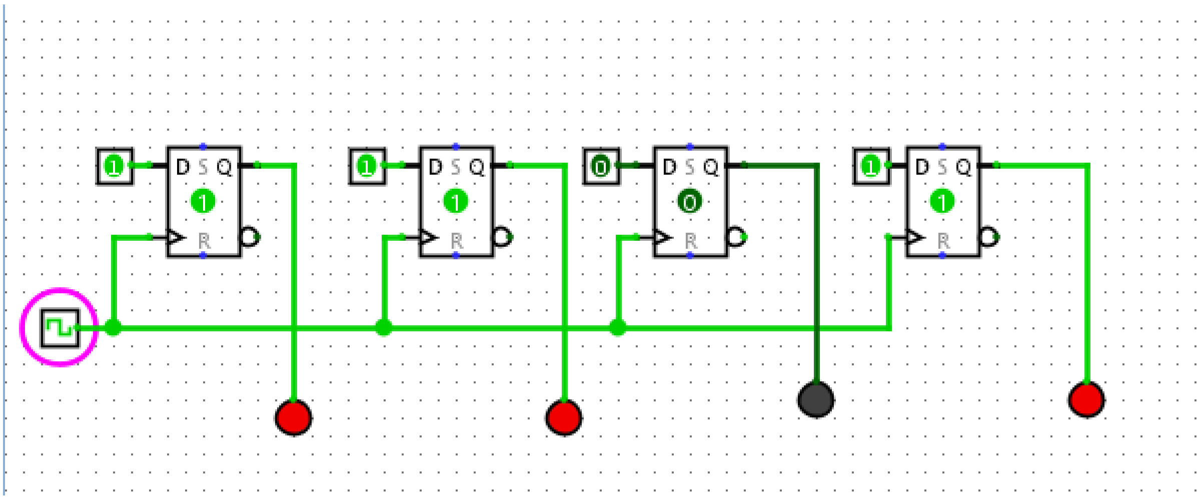

***Image 17:* 1101**

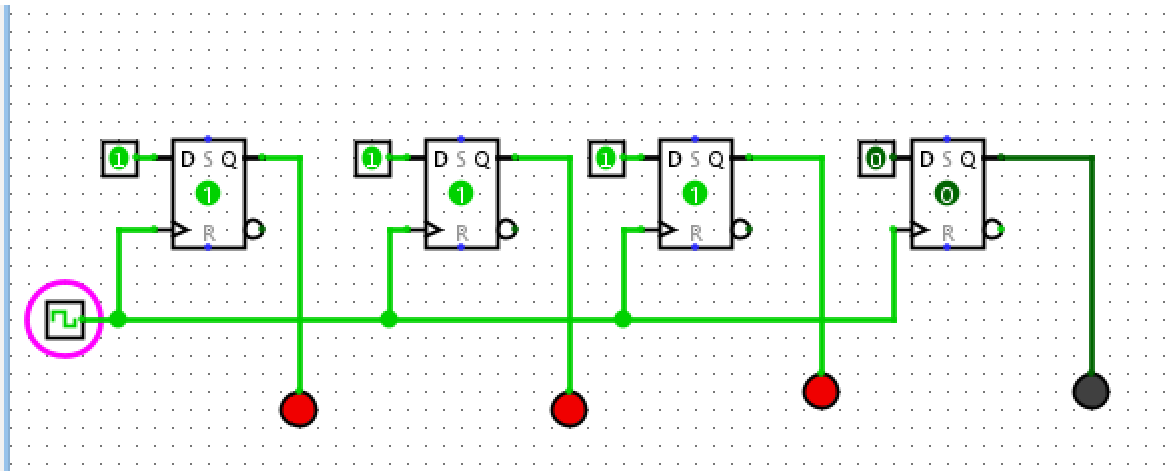

***Image 18:* 1110**

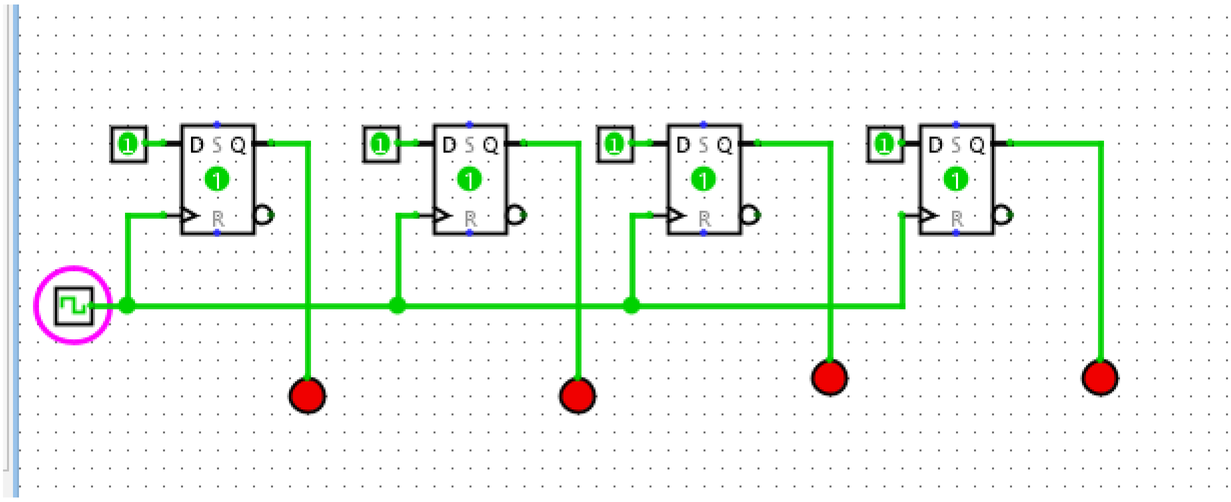

***Image 19:* 1111**

22. 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

Complete this table and place a copy of it in your submission document

Provide your answers in your submission document

When complete:

■ Submit your answers (screen shots, etc) in a single document using **Canvas**

■ Show your lab demonstrator your working circuits in class (you must do this to get the credit). Your lab demonstrator may request you to resubmit if issues exist.