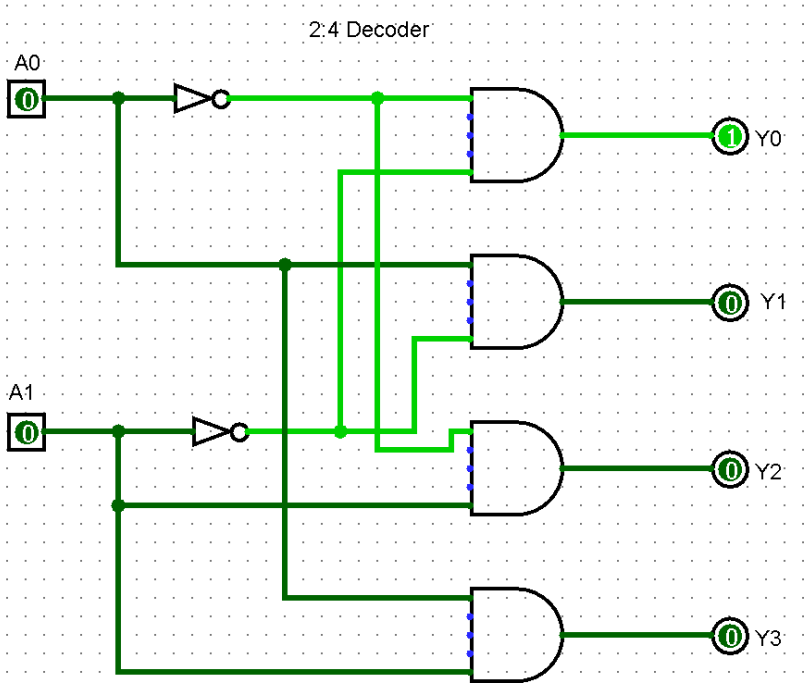
LD Lab Report

Farhan Syed
IS1500
2 November 2022

Assignment 1: Decoder

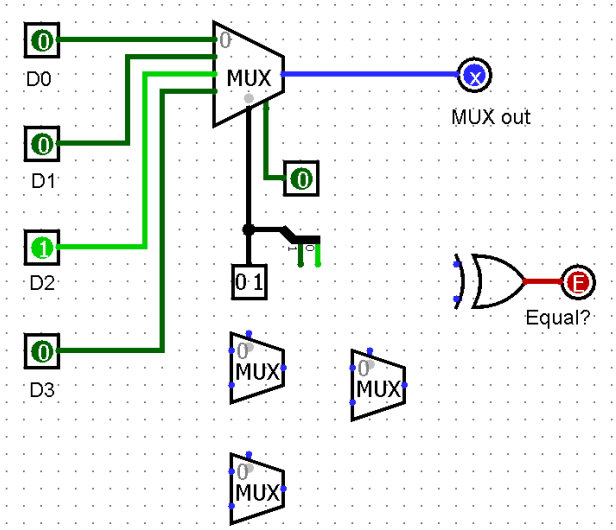
Task 1.1



I verified the design by poking the inputs to test all 4 combinations. This proved that the correct output was turned on and that the circuit is one-hot.

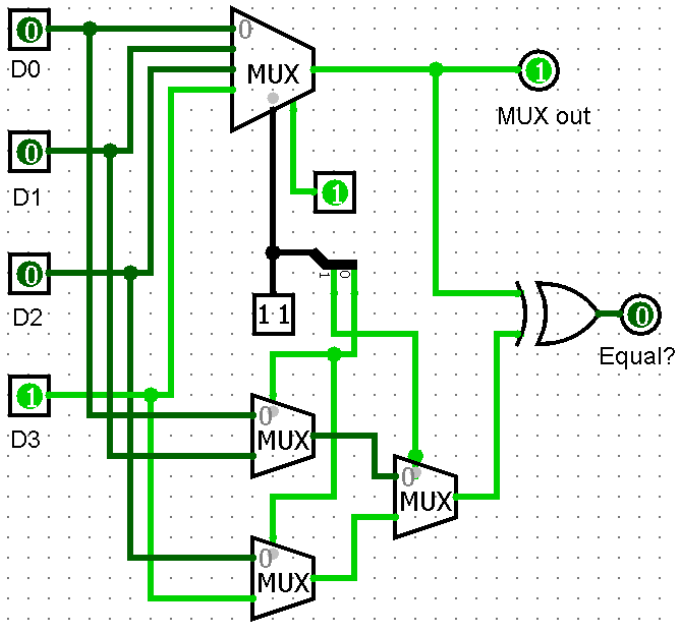
Assignment 2: Multiplexer

Task 2.1



- "MUX out" is blue because it is carrying a floating one-bit value, due to the enabler having the value 0.
- "Equal?" is red because it is carrying an error value because there are not any inputs.
- The select signal is black because it is carrying a multi-bit value, due to the 2 bit select port.

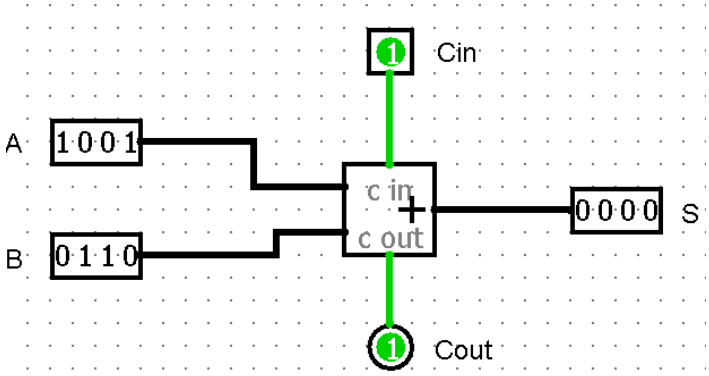
Task 2.2



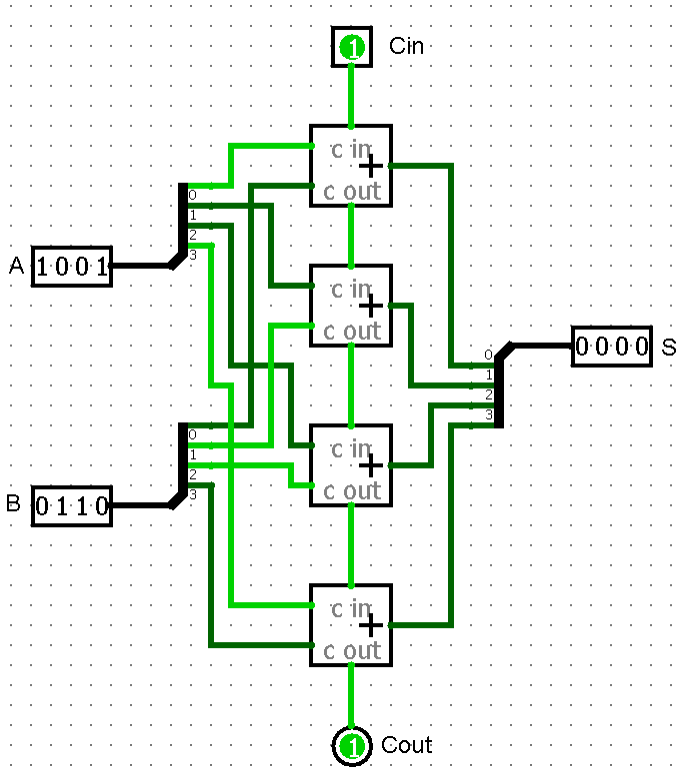
If the "Equal?" output is 0, both implementations work. This is because the XOR gate will make the output 1 if only one of the designs work.

Assignment 3: Adder

Task 3.1

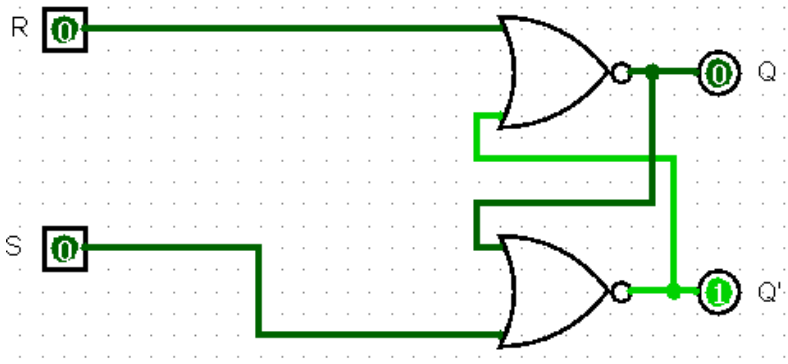


Task 3.2

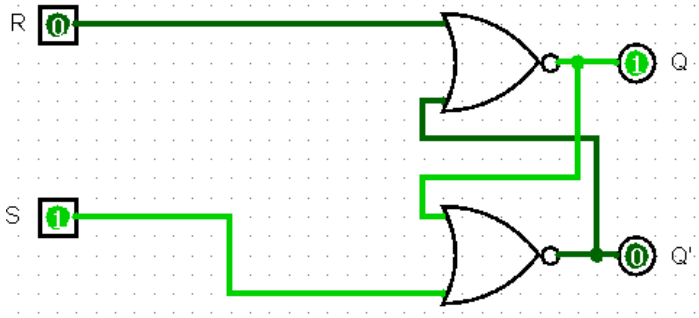


Assignment 4: Latches

Task 4.1



1. Because S and R are 0, Q and Q' will have their stored values. The possible values of Q and Q':
   - Q = 0 and Q' = 1.
   - Q = 1 and Q' = 0.

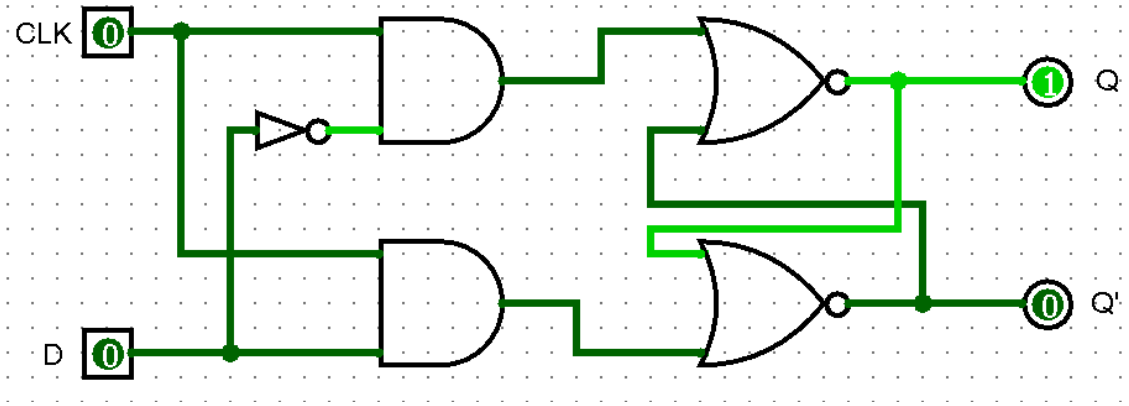


2. S is 1 and R is 0. The only possible values of Q and Q' are:
   - Q = 1 and Q' = 0.

   If we toggle S, the circuit will not change. This is because when both S and R become 0, Q and Q' will have their previously stored values.

3. If S is 0 and we toggle R, then Q will always be 0 and Q' will be 1.

Task 4.2



1. If D is 0 and we toggle CLK, Q will always be 0 and Q' will be 1. They will not change because CLK determines when Q and Q' can be accessed and changed.

   If CLK is 0 and we toggle D, Q and Q' will have their previously saved values. No AND gates will become 1 so nothing changes.

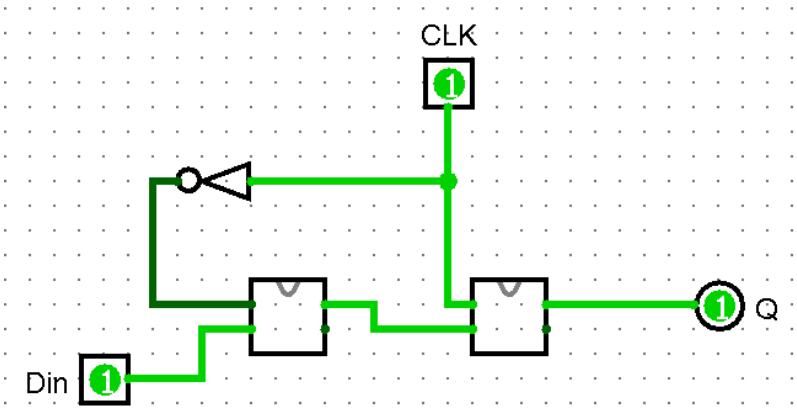
2. If CLK is 1 and we toggle D, then Q and Q' will change values because these are determined by the D input (while CLK is 1)

3. Firstly, the D latch avoids the case where S=1 and R=1 that results in that both Q and Q' are 0. Secondly, the D latch clearly defines *what* and *when* updates are made.

4. The problem with D latches is that Q can change more than once during a single clock pulse. Therefore, D latches are not used in synchronous sequential logic because a requirement for synchronous sequential logic is that states are only updated on clock edges.

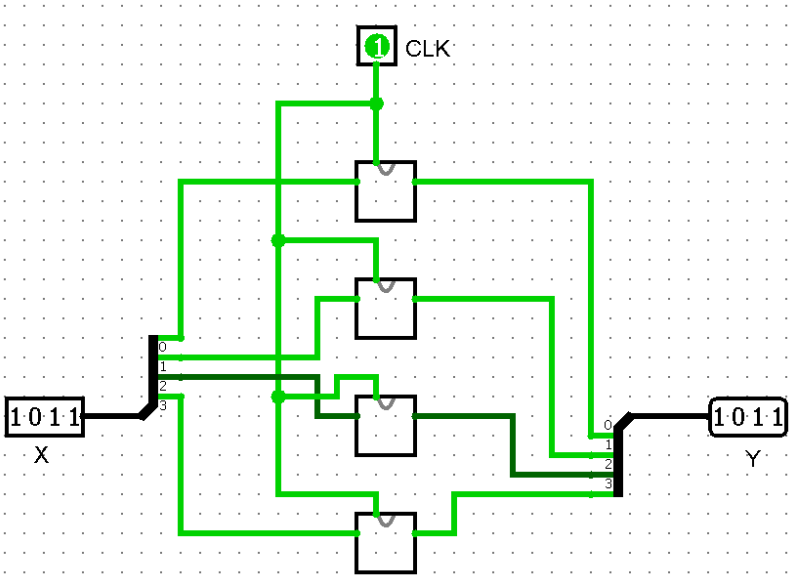
Assignment 5: Flip-Flops

Task 5.1



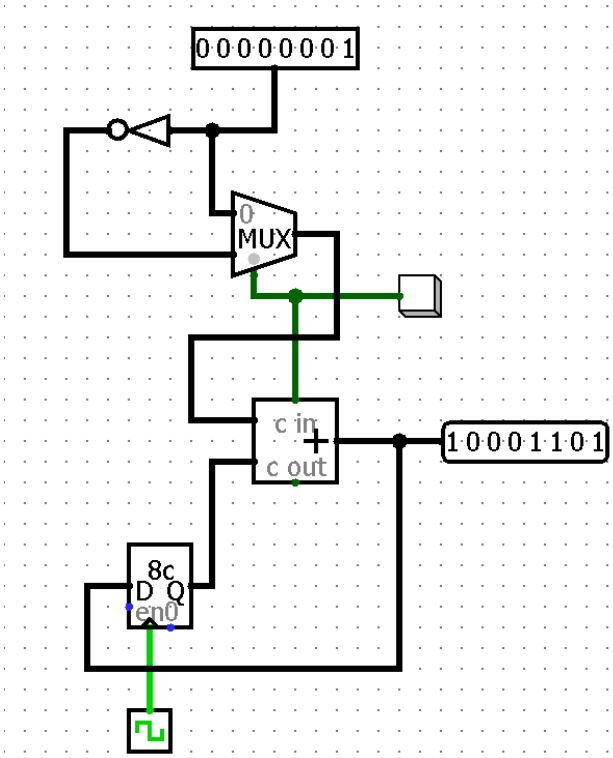
Q will only change during the rising edge phase of the CLK i.e. when the clock goes from 0 to 1. During this phase, Q will get the current value of D. For example, if CLK is 1 and we toggle D, Q will not change. But if D is 1 and we change CLK from 0 to 1, Q will become 1.

Task 5.2



Assignment 6: Design of a Synchronous Sequential Circuit

Task 6.1



An input enters the multiplexer.

- If the button is not pressed, the input number will then enter the adder. This input is added to the current output (sum).
- If the button is pressed, the inverse of the number will enter the adder. There will also be a carry in of 1. As a result, the adder will act as a subtract instead (because we have the two's complement of the input), and subtract the input from the current sum.
- The CLK controls when these addition and subtraction operations occur - they only occur during the rising edge of the CLK

For a circuit to be considered synchronous sequential, the circuit must contain registers and states must only be updated on clock edges. Task 5.2 is a register that only updates during rising edges, and is thus a synchronous sequential circuit. Task 5.1 is also synchronous sequential because a D-Flip-Flop can be considered a register with only 1 flip-flop, and because it updates during the clock's rising edge.