ENCS2110

DIGITAL ELECTRONICS AND COMPUTER ORGANIZATION LABORATORY

Experiment 5: Sequential Logic Circuits

Name: Sara Totah

Student ID: 1181779

Instructor: Khader Mohammed

T.A: Enas Jawabreh

Date: 20/03/2021

Table of Contents

Table of Figures

Abstract

This experiment aims to understand the difference between the sequential and combinational logic circuit in addition to understanding how to flip flops to work (D flip-flop & JK flip-flop).

Equipment used:

We were supposed to use a Basic Electricity Circuit Lab, JK Flip-flop circuits and Flip-flop circuits, but since the experiment was implemented online due to the pandemic, we used proteus software.

Pre-Lab

The SR (set-Reset) Latch

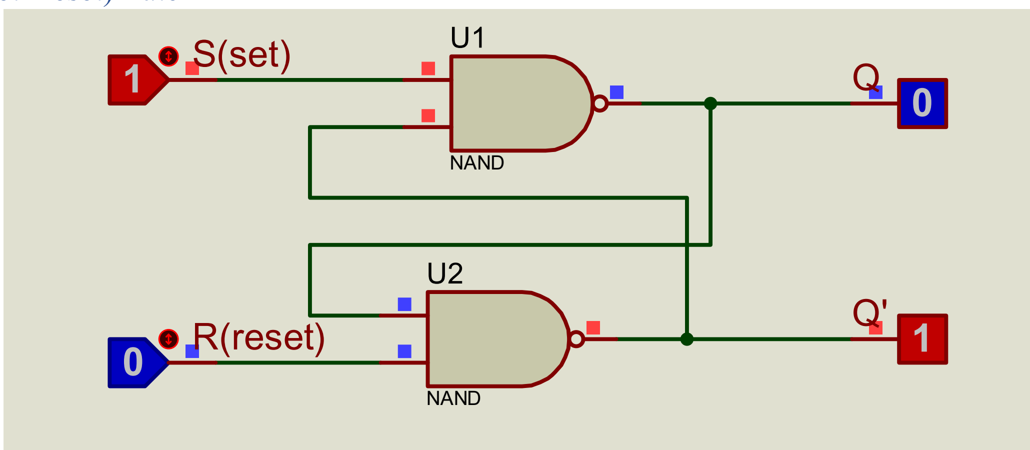


Figure 1: SR Latch with NAND gate

S	R	Q	Q'	
1	0	0	1	
1	1	0	1	After S = 1, R = 0
0	1	1	0	
1	1	1	0	After S = 0, R = 1
0	0	1	1	

RS Latch with Control input

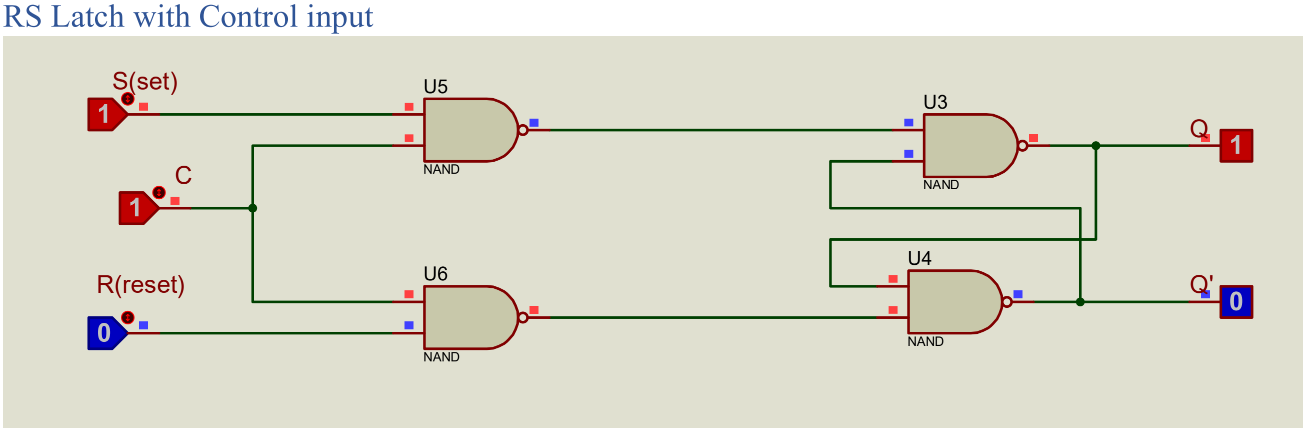


Figure 2: RS Latch with control input

C	S	R	NEXT STATE OF Q
0	X	X	No change
1	0	0	No change
1	0	1	Q = 0 Reset state
1	1	0	Q= 1 set state
1	1	1	Indeterminate

D-Latch

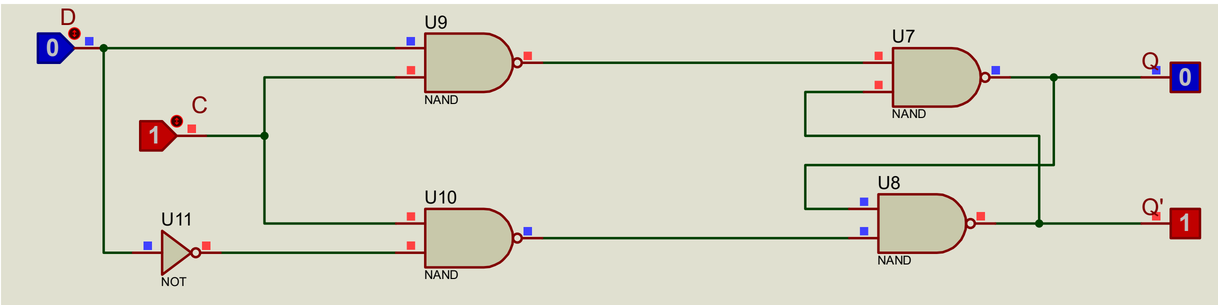


Figure 3: D-Latch

C	D	Next state of Q
0	X	No Change
1	0	Q = 0 Reset state
1	1	Q = 1 Set state

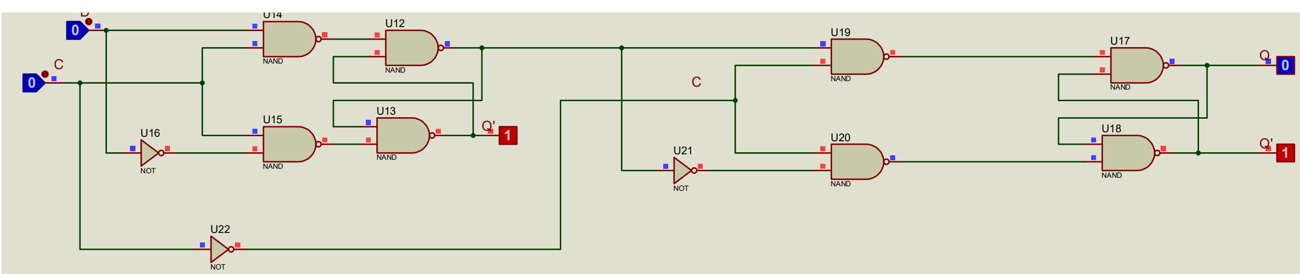


Figure 4: D flip flop with two D latches

Theory

Sequential Circuits

A sequential circuit has a memory so output can vary depending on the input and previous output which was saved in the memory.

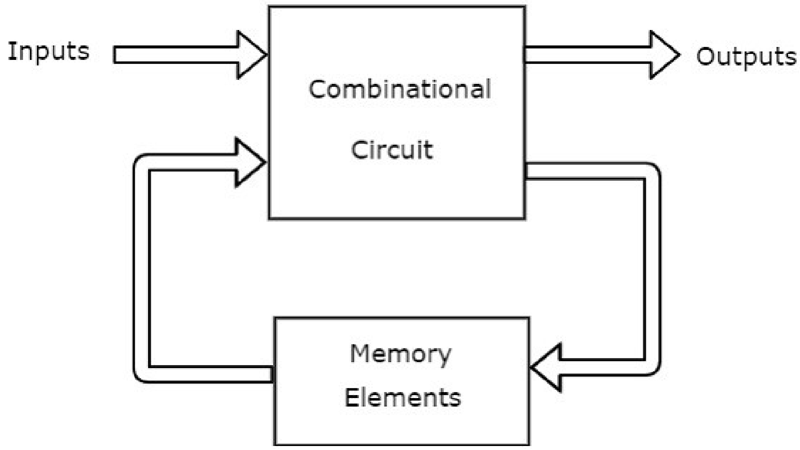


Figure 5: Sequential Circuits

There are two types of memory elements depending on the type of triggering that is suitable to operate

Latches

Latches are basic storage elements that are level-sensitive devices, they are controlled by a clock, there are a couple of types for the latches, which are:

1. The SR (Set-Reset) Latch

This latch needs two inputs (Set and Reset) and two outputs, it can be implemented using 2-cross-coupled NOR gates or two cross-coupled NAND gates.

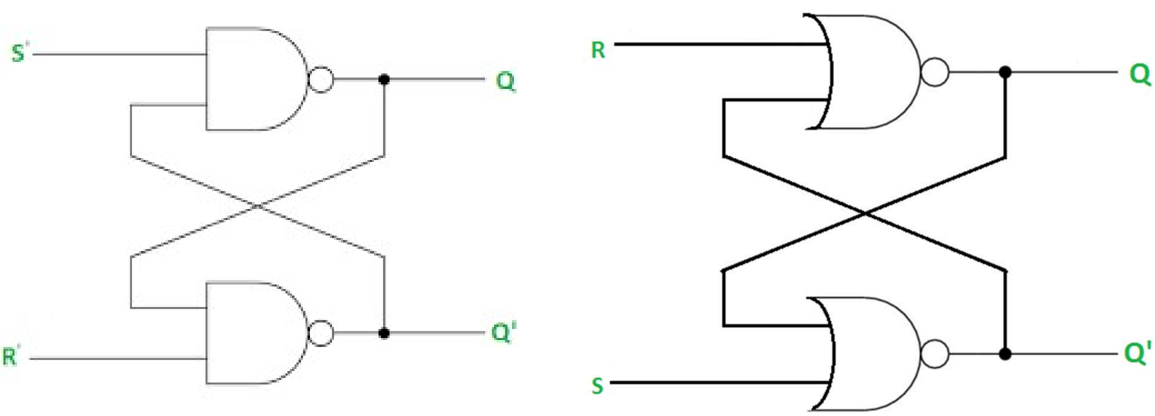


Figure 6: RS latch with NAND gates and RS latch with NOR gates

Q	Q’	STATE
1	0	Set
0	1	Reset

S	R	Q	Q'	
1	0	0	1	
1	1	0	1	After S = 1, R = 0
0	1	1	0	
1	1	1	0	After S = 0, R = 1
0	0	1	1	Indeterminate

2. The D Latch

The input in this type is more complicated than the ones in SR latches

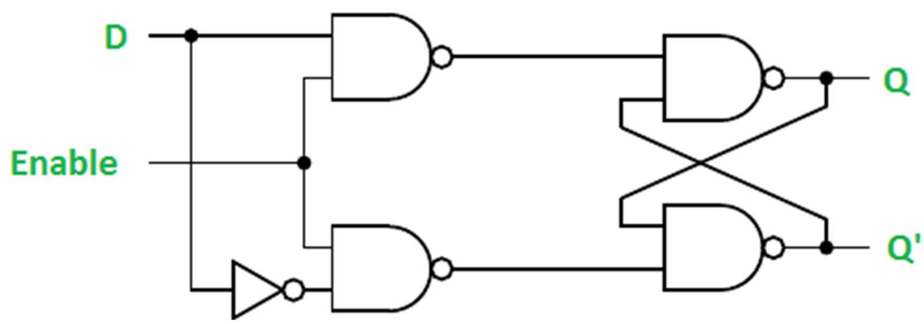


Figure 7: D Latch

It was developed to eliminate the undefined condition of the indeterminate in the RS latch

C	D	Next state of Q
0	X	No Change
1	0	Q = 0 Reset state
1	1	Q = 1 Set state

Flip-Flops

This also is a sequential circuit, it's a 1-bit memory cell that can be used to store digital data, and can be said that the flip flop is an edge-triggered device, it also has some types:

1. D Flip flop

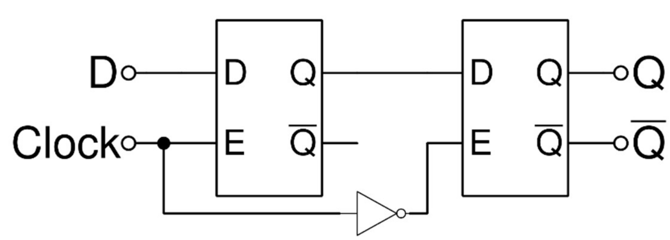
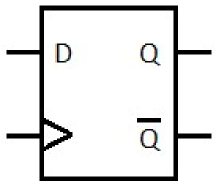


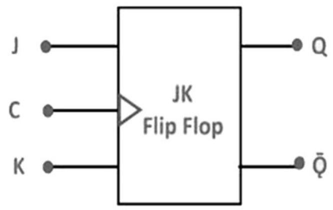
Figure 8: D Flip flop with two D latches



D	Q (t +1)
0	0 Reset
1	1 Set

Figure 9: D flip flop symbol

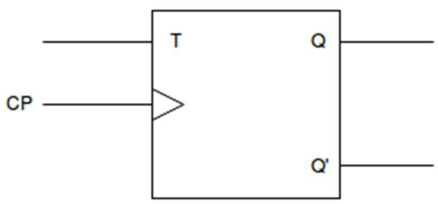
2. JK Flip flop



J	K	Q (t +1)
0	0	Q(t) No change
0	1	0 Reset
1	0	1 Set
1	1	Q'(t) Complement

Figure 10: JK flip flop symbol

3. T flip flop



T	Q (t +1)
0	Q(t) No Change
1	Q'(t) Complement

Figure 11: T flip flop symbol

Registers

As it was mention before the flip-flop is a 1bit memory cell, to increase the storage capacity (number of bits), we can use several flip-flops, and this group is called **Register**, N flip-flops together make an N-bit register, all the flip-flops are driven by the same clock, as an example of the registers is the shift-right register.

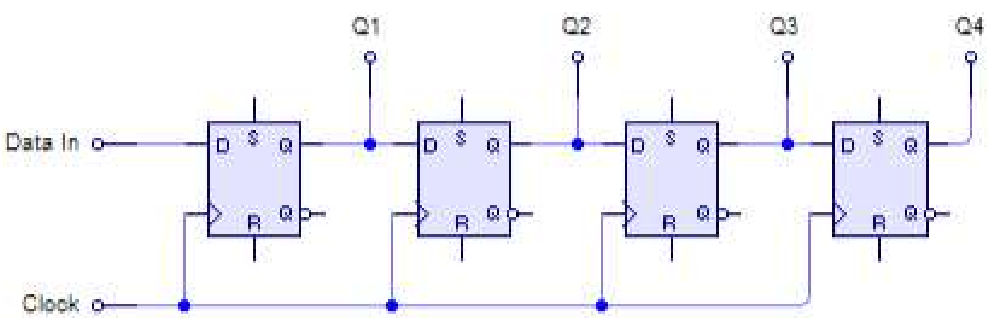


Figure 12:4-bit shift-right register

Shift register

Group of flip-flops connected so that the output of the first flip-flop is the input for the second one and so on.

Counters

The counter is a special type register that goes through a prescribed sequence of states, there are two types of counters: Ripples and synchronous

1. Ripple Counters (Asynchronous)

It's an asynchronous counter, it counts up to 2^n states, it is known by that name due to the way the clock pulse ripples its way through the flip-flops.

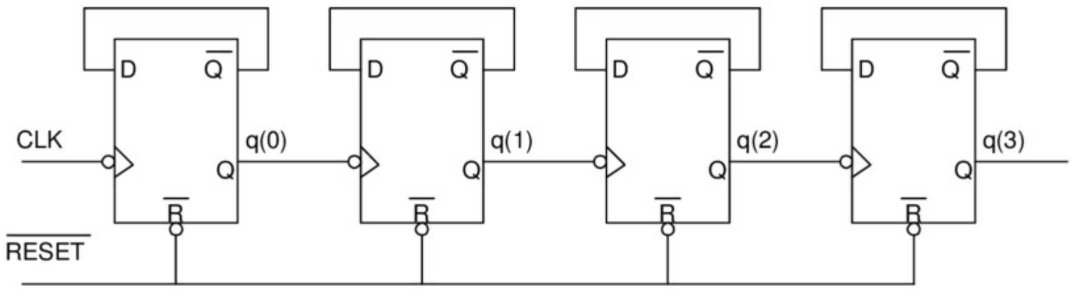


Figure 13:4-bit ripple counter

2. Synchronous Counters

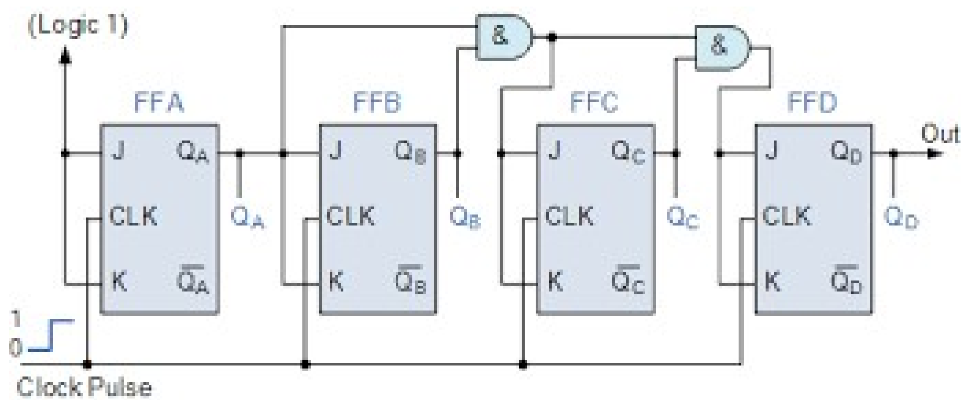


Figure 14: Synchronous Counter

The procedure, Data and Results

Latches and Flip flops

A. Constructing RS latch with Basic Logic Gates

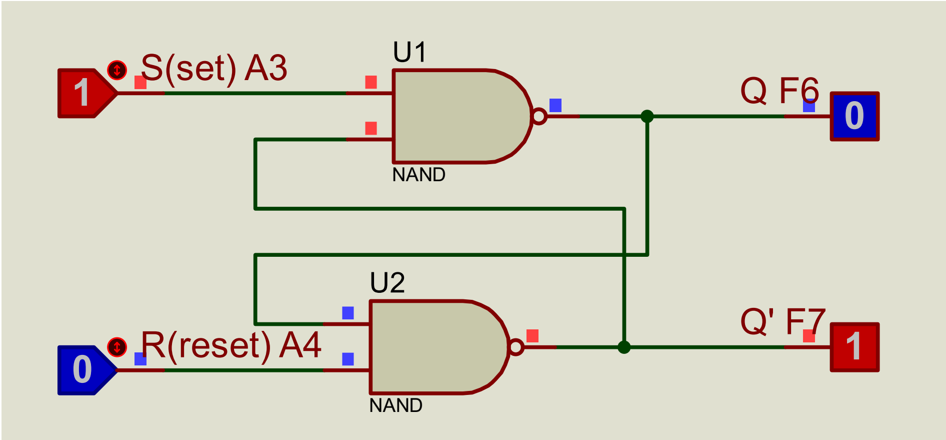


Figure 15: RS latch

A3	A4	F6	F7
0	0	1	1
0	1	1	0
1	0	0	1
1	1	0	1

We noticed that when A3 and A4 were set to equal zero, both Q and Q' became 1 which is not allowed, so this case is prohibited, if A3 = 0 and A4 = 1, then Q would equal 1 and be at the SET state when reversing both inputs (A3 = 1, A4 = 0) Q will be 0 (at the RESET state), if both inputs were set to equal 1, no change will happen to the output it will remain the same as before.

B. Constructing an RS latch with control input

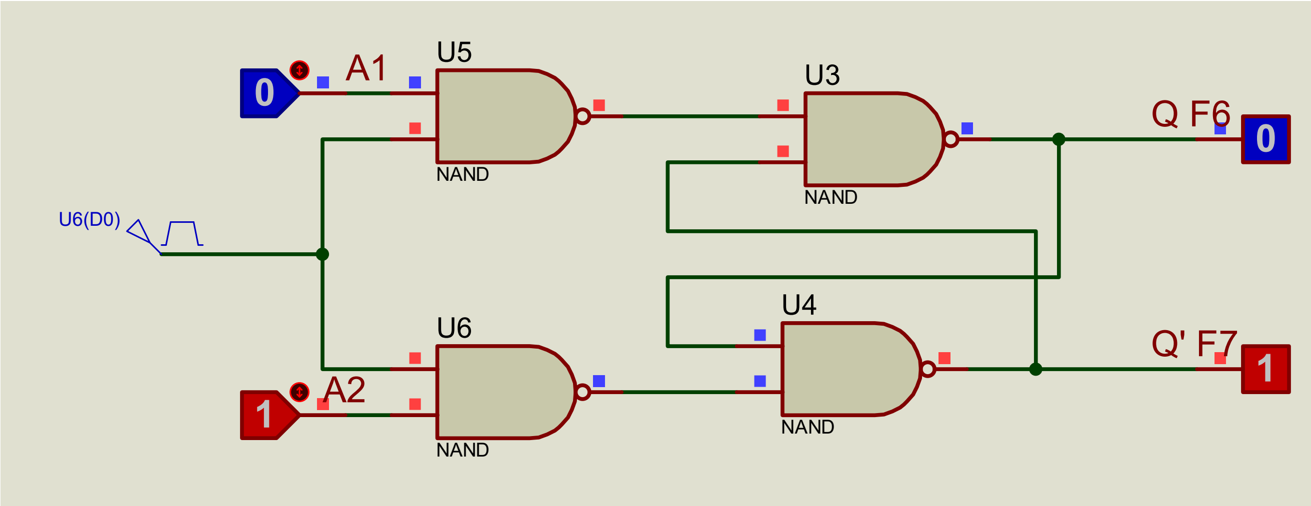


Figure 16: RS Latch with control input

A1	A2	F6	F7
0	0	No change	No change
0	1	0	1
1	0	1	0
1	1	1	1

If A1 = 0 and A2 = 1then the output Q would equal 0 and Q' = 1, if we changed A2 to zero the outputs won't change, and we noticed too that the case when both inputs = 1 is a prohibited case since both Q and its complement = 1 which is not allowed.

C. Constructing D latch with RS latch

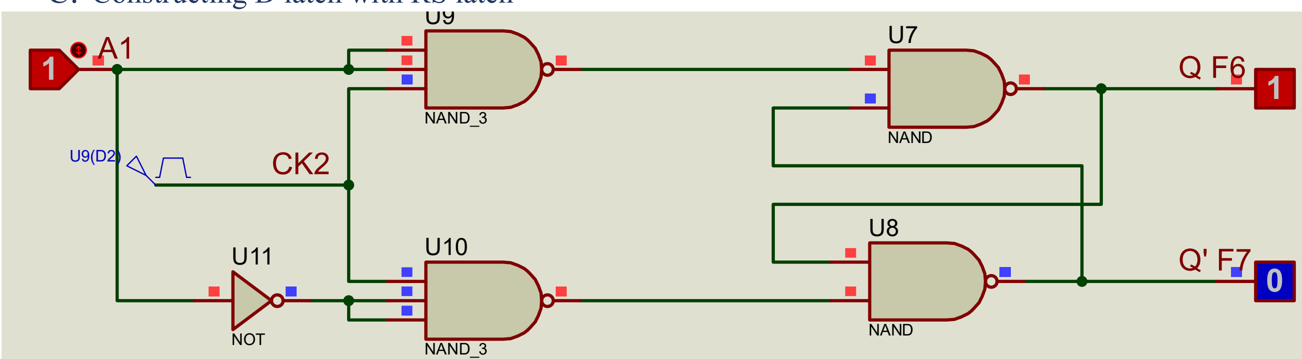


Figure 17: D Latch

CK2	A1	F6
0	0	memory
0	1	memory
⎍	0	0
⎍	1	1

First, if we set the clock to zero, there will be no output no matter what the input A1 was, but after changing the clock into 1 and setting the input to 0 the output will equal zero, and after setting the input A1 to 1 the output will equal one, after any case of the last two cases, if we changed the clock back to zero the output will keep holding the result it already got from the last time.

D. Constructing JK latch with RS latch

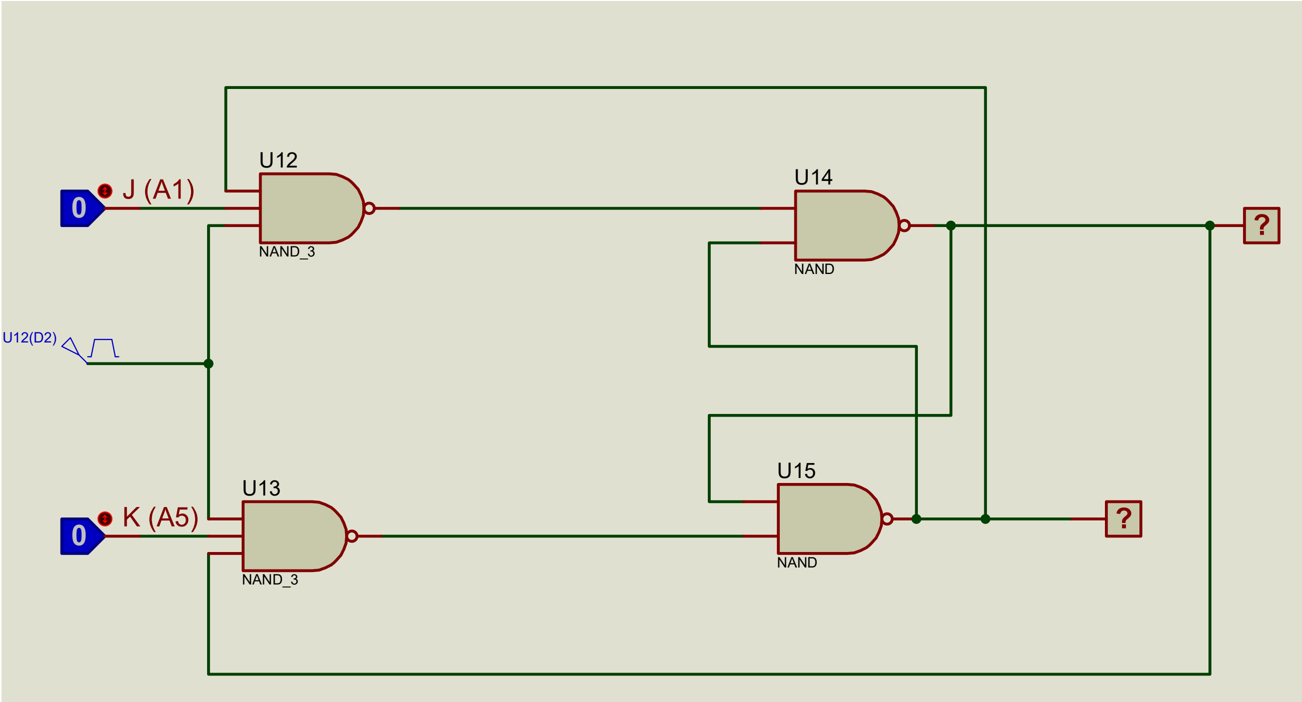


Figure 18: JK Latch

CK	A1	A5	F6
⎍	0	0	memory
⎍	0	1	0 Reset
⎍	1	0	1 Set
⎍	1	1	Complement

The JK flip-flop is an SR flip-flop with feedback if both inputs (A1 & A5) are set at logic 0 (J = K = 0), no matter what the clock state was the flip-flop will keep containing the data it already has (as a memory), but when both inputs are set to 1 (J = K = 1) the output will switch and change his state to its complement for example if the J = 1 and K = 0 the output state will be SET the output F6 = 0, so when changing the K value to 1 the output will be the complement of the previous output.

E. Constructing JK Flip-flop with master-slave RS latches

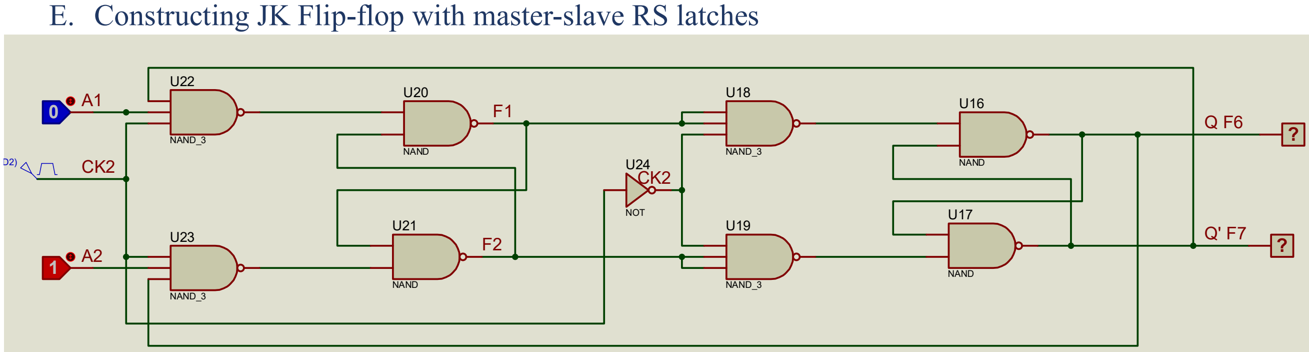


Figure 19: JK Flip-Flop

CK	K	J	F1	F2	F6	F7
⎍	0	0	1	0	1	0
⎍	0	1	0	1	1	0
⎍	1	0	1	0	0	1
⎍	1	1	0	1	1	0
⎍	1	1	1	0	0	1

Registers

A. Constructing Shift Register with D Flip-Flops

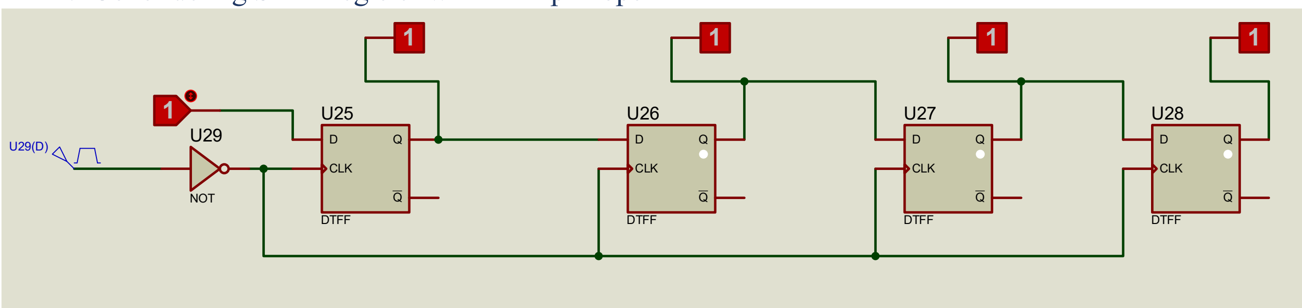


Figure 20: Shift Right Register

When we set the input to equal 1, on each clock pulse the register shifts the input which equals 1, in this case, 1-bit to the right.

B. 4-Bit Shift Register with serial and parallel load

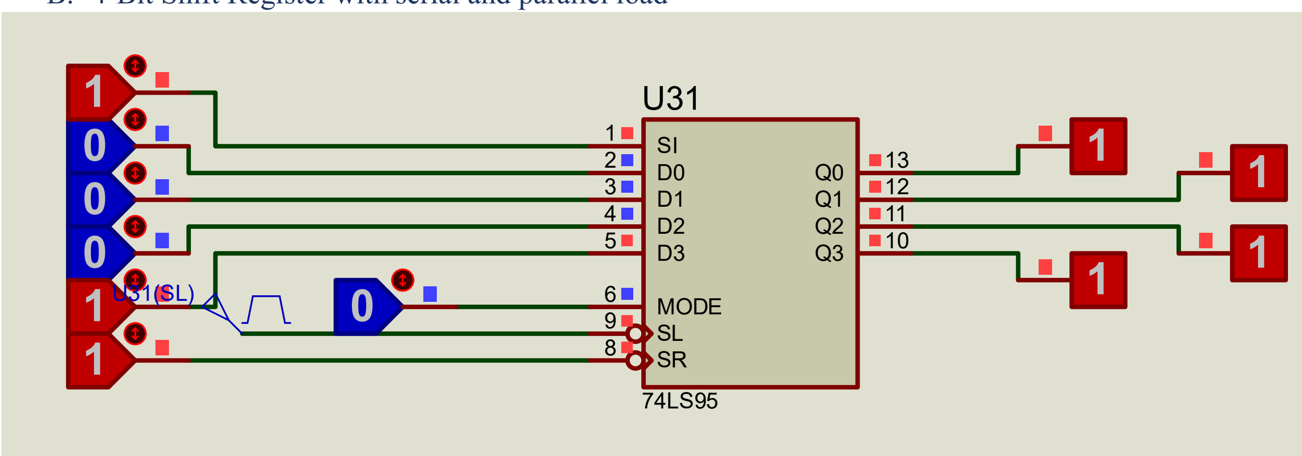


Figure 21: Shift register with serial and parallel load

MODE	SL	Q3	Q2	Q1	Q0
0	⎍	1	0	0	1
0	⎍	0	0	1	1
0	⎍	0	1	1	0
1	⎍	1	1	0	0

Counters

A. 2-bit Synchronous Counter

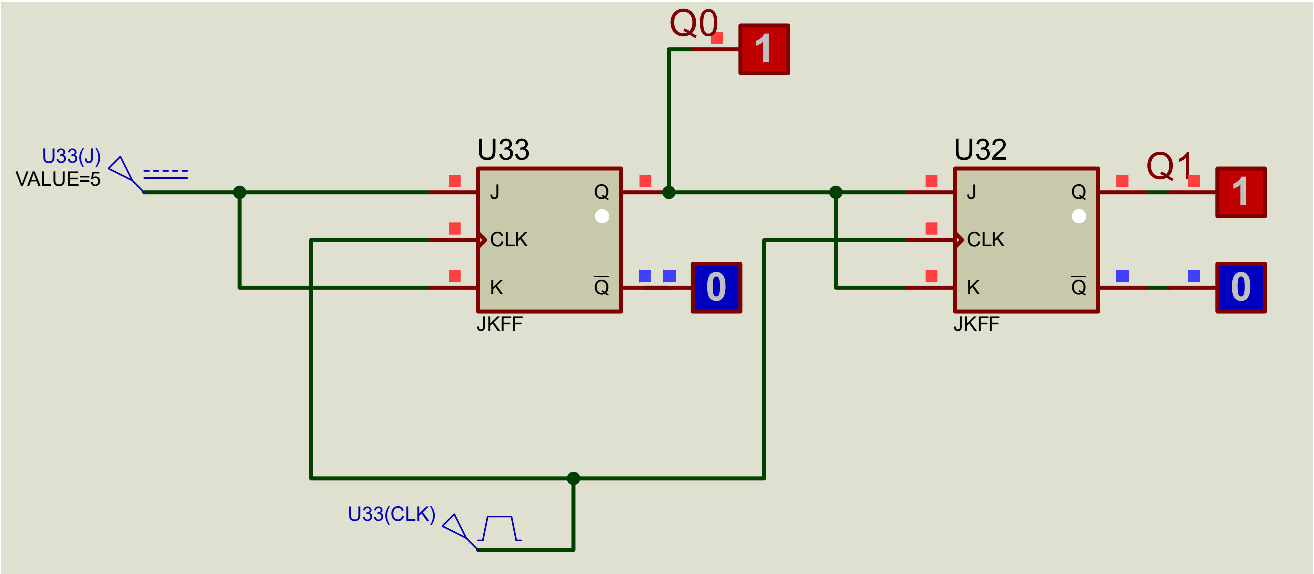


Figure 22: 2-bit Synchronous Counter

CLK	Q1	Q0
⎍	0	0
⎍	0	1
⎍	1	0
⎍	1	1
⎍	0	0
⎍	0	1
⎍	1	0
⎍	1	1

First, both Q2 and Q1 were in the reset state, after the second clock edge, Q0 should change from 0 to 1, but both J1 and K1 = 0 so the second flip-flop won't change its state so Q1 will stay 0.

After the third clock edge, Q0 changes from 1 to 0 again, but J2 and K2 now = 1 so Q1 changes from 0 to 1

After the fourth clock edge, Q0 will change to 1 but the Q1 will remain the same.

After the fifth clock edge, Q0 will change to 0 and Q1 will change to 0 too, now they are back to the reset state.

B. 3-bit (divide-by-eight) Ripple Counter

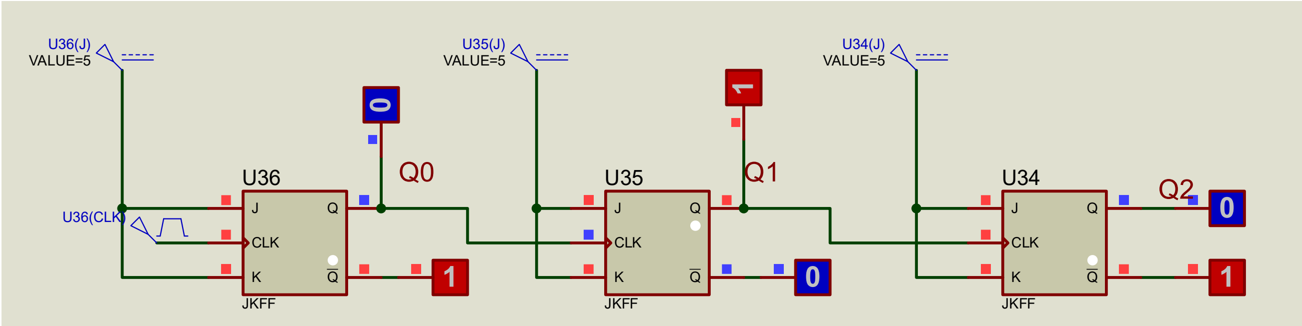


Figure 23: 3-bit Ripple Counter

CLK	Q2	Q1	Q0
⎍	0	0	0
⎍	0	0	1
⎍	0	1	0
⎍	0	1	1
⎍	1	0	0
⎍	1	0	1
⎍	1	1	0
⎍	1	1	1

Here there are three flip-flops so the counter can count up to $2^3 = 8$ values.

C. BCD Counter

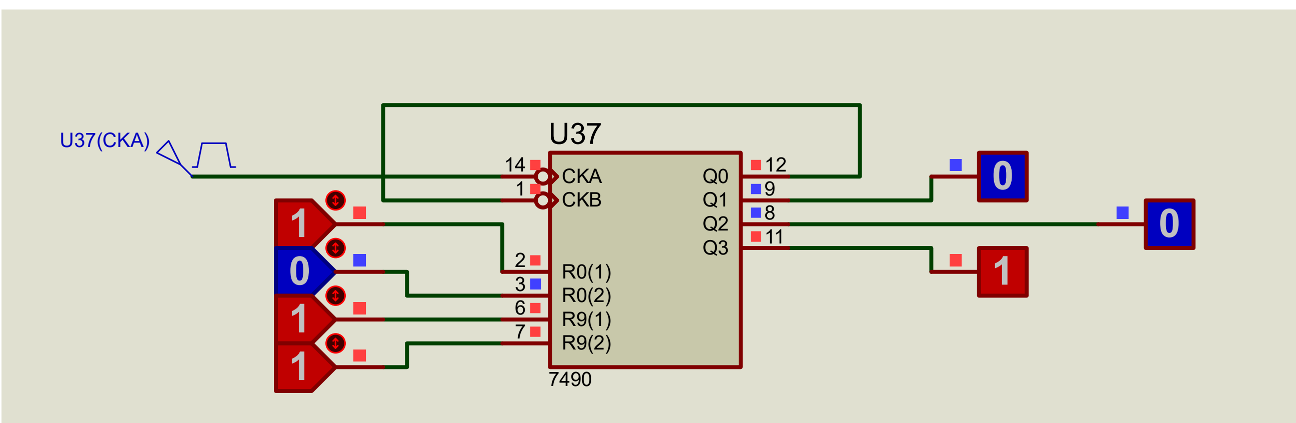


Figure 24: IC 7490 BCD Counter

Post-Lab

Task 1

Modify the circuit in Figure 22 to be a 3-bit Synchronous Counter. Attach the design with this experiment report.

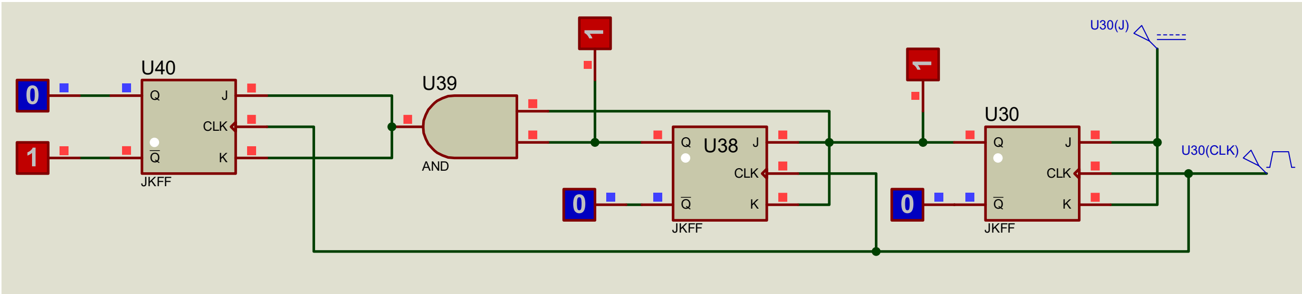


Figure 25: 3-bit Synchronous Counter

Task 2

Change the connection of counter in Figure 24 to count from:

- 0-to-5

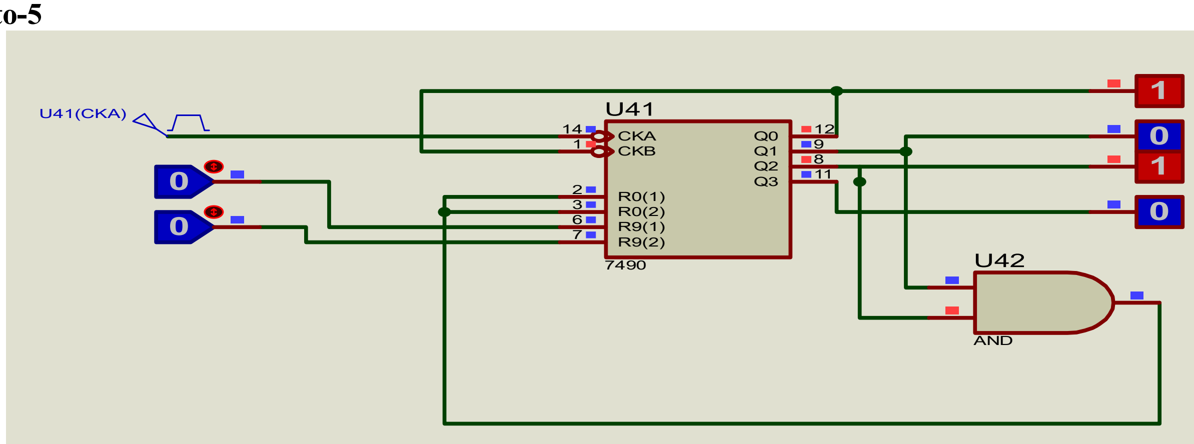


Figure 26: 0-5 Counter

- 0-to4

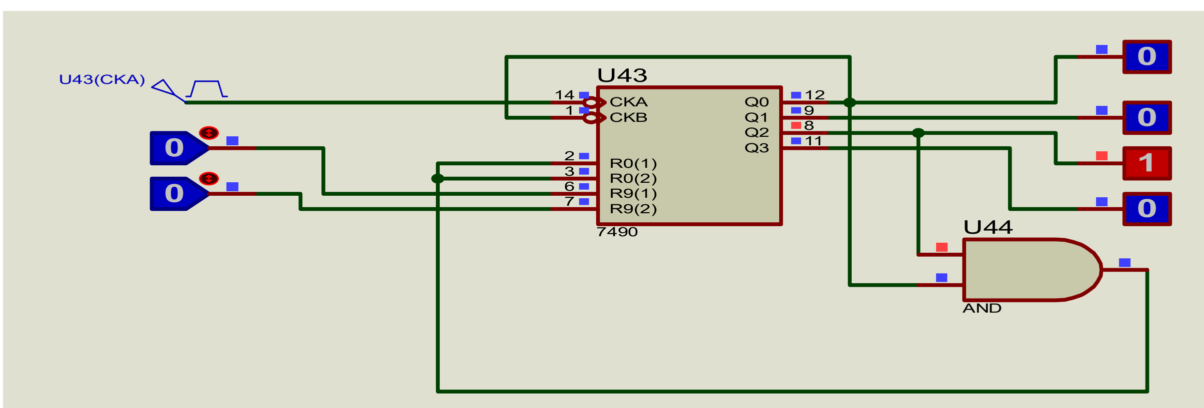


Figure 27: 0-4 Counter

Discussion

Although latches are useful for storing binary information, they are rarely used in sequential circuit design, why?

It's right that the latches are faster than the flip-flops since the flip-flops consist of several latches, and they consume less power but the latches tend to make glitches which are not appreciated in the designing, also the latches are level triggered as mentioned before while the flip-flops are edge-triggered, so the change in the flip-flop will happen only at the triggering edge.

What is the disadvantage of the RS flip flops?

The major disadvantage is when both inputs are 1 (S=R=1) in this case the output and its complement will be 1 which is not allowed.

What is the difference between “synchronous” and “ripple” counters?

At the ripple-counters, each flip-flop is triggered with its clock, while at the synchronous counter all flip-flops are triggered with the same clock so the synchronous counter should be faster than the ripple counter.

Conclusion

In this experiment the meaning of sequential circuits and the concept of memory units such as latches and flip-flop was covered, the experiment went smoothly with no troubles.

References

- https://www.tutorialspoint.com/computer_logical_organization/sequential_circuits.htm
- https://www.geeksforgeeks.org/latches-in-digital-logic/
- https://www.tutorialspoint.com/digital_circuits/digital_circuits_latches.htm
- https://www.tutorialspoint.com/computer_logical_organization/digital_counters.htm
- Digital Lab manual.