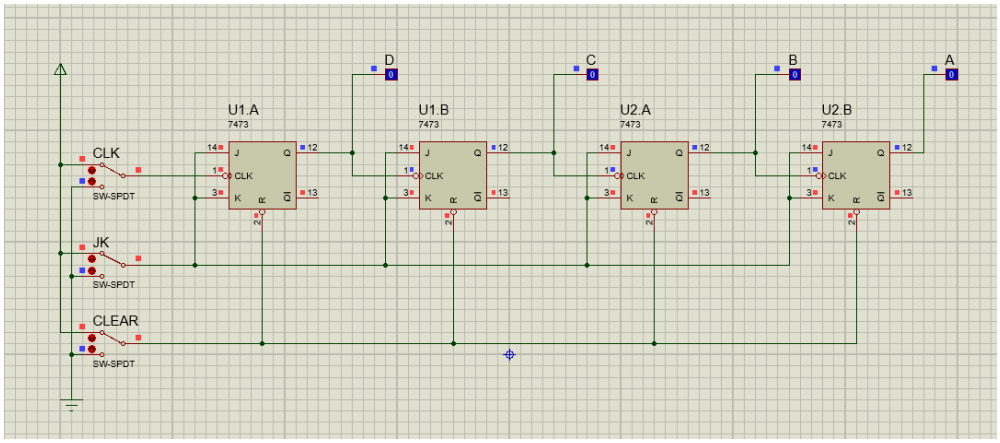
NAMA : MOTWKEL MHMOUD MOHMOED ADAM
NIM : L200184220
CLASS : X

Experiment 1

1. Make JK the flip-flop combination!



2. Simulate your circuit!

	Input			Output			
	Clear	**JK**	**CLK**	**A**	**B**	**C**	**D**
1	1	1	0	0	0	0	0
2	1	1	1	0	0	0	0
3	1	1	0	0	0	0	1
4	1	1	1	0	0	0	1
5	1	1	0	0	0	1	0
6	1	1	1	0	0	1	0
7	1	1	0	0	0	1	1
8	1	1	1	0	0	1	1
9	1	1	0	0	1	0	0
10	1	1	1	0	1	0	0
11	1	1	0	0	1	0	1
12	1	1	1	0	1	0	1
13	1	1	0	0	1	1	0
14	1	1	1	0	1	1	0
15	1	0	0	0	1	1	1
16	1	0	1	0	1	1	1
17	1	1	0	0	1	1	1
18	1	1	1	0	1	1	1
19	0	1	0	0	0	0	0
20	0	1	1	0	0	0	0

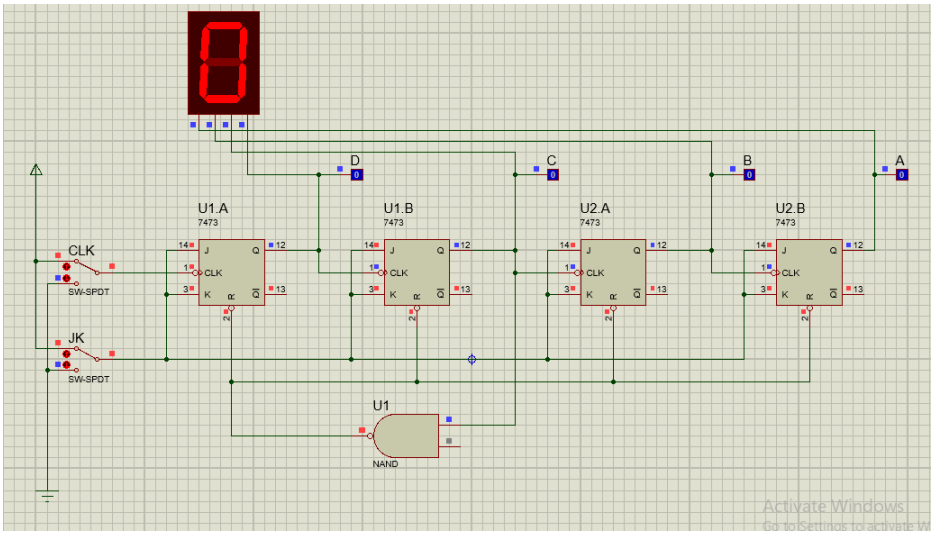
3. What's the function of:
    a. Switch CLK
        - Add the output value and as long as they keep in their CLOCK place and changing it from 0 to 1 will change the result of the output value too. So when the CLK repeatedly switched from 0 to 1 it will add some binary digit.
    b. Switch JK
        - Store the output value (Set condition), so the output condition will display the previous condition
    c. Switch Clear
        - Reset the memory.

4. Conclusion

    JK flip-flop as the input controller that determine what will the FF do. Whenever it received the increased clock condition, the circuit Counter Asynchrony Model 16 used for expanding data.

Experiment 2

1. Make the flip-flop combination!



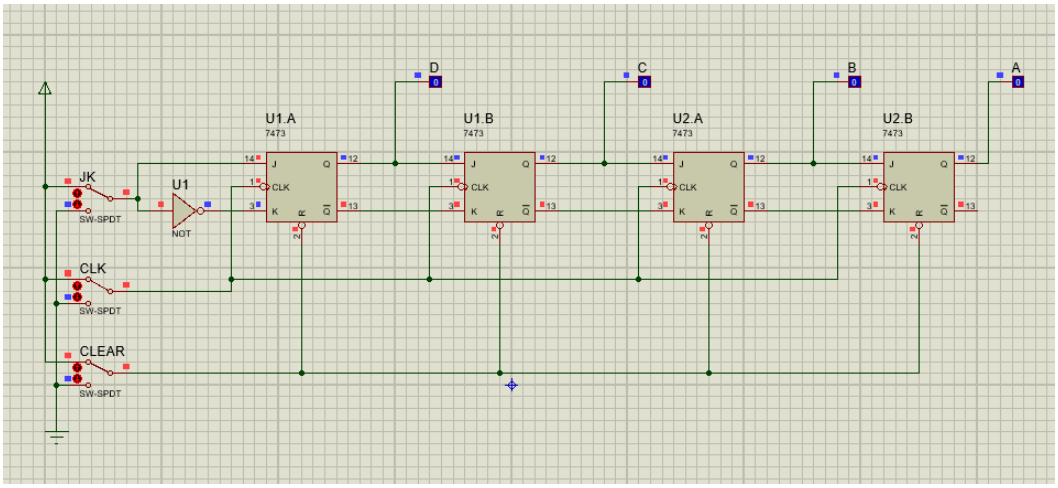
2. Simulate your circuit!

	Input		Output			
	JK	**CLK**	**A**	**B**	**C**	**D**
1	1	0	0	0	0	0
2	1	1	0	0	0	0
3	1	0	0	0	0	1
4	1	1	0	0	0	1
5	1	0	0	0	1	0
6	1	1	0	0	1	0
7	1	0	0	0	1	1
8	1	1	0	0	1	1
9	1	0	0	1	0	0
10	1	1	0	1	0	0
11	1	0	0	1	0	1
12	1	1	0	1	0	1
13	1	0	0	1	1	0
14	1	1	0	1	1	0
15	1	0	1	0	0	0
16	1	1	1	0	0	0
17	1	0	1	0	0	1
18	1	1	1	0	0	1
19	1	0	0	0	0	0
20	1	1	0	0	0	0
21	0	0	0	0	0	0
22	0	1	0	0	0	0
23	1	0	0	0	0	0
24	1	1	0	0	0	0

3. Conclusion:
   This flip-flop function is the same as the previous flip-flop. The difference is it used the 7SEG-BCD to translate the BCD binary into the decimal form. When the JK-switch is in condition 1, use the Clock to change the decimal. However, it wouldn’t work if the JK-switch is in condition 0.

Experiment 3

1. Make the flop-flop combination!



2. Simulate your circuit!

	Input			Output			
	Clear	**JK**	**CLK**	**A**	**B**	**C**	**D**
1	0	*x*	-	0	0	0	0
2	1	1	-	0	0	0	0
3	1	1	*1*	0	0	0	1
4	1	1	*2*	0	0	1	1
5	1	1	*3*	0	1	1	1
6	1	0	*4*	1	1	1	0
7	1	0	*5*	1	1	0	0
8	1	0	*6*	1	0	0	0
9	1	0	*7*	0	0	0	0
10	1	0	*8*	0	0	0	0
11	1	1	*9*	0	0	0	1
12	0	0	*10*	0	0	1	0
13	0	0	*11*	0	1	0	0
14	0	0	*12*	1	0	0	0
15	0	0	*13*	0	0	0	0

3. Conclusion:
   The binary changed if the clock's condition changed from 1 to 0. And the Clear-switch reset the output values as if the JK is in condition 1.