Digital Logic & Design (Lab)

Examination: Lab

Instructor: Muhammad Amin
Programs: BS(SE)/BS(CS)
Course Codes: SEC-201/CSC-201
EDP Codes: 102007017
Semester: Summer 2020
Date: Oct. 5, 2020
Timing: 9:00am to 1:00pm

Name: Muhammad Hamza

ID: 13136

Note: Use MultiSim to design the following circuits. Use truth tables where necessary.

Q.1Design and verify the logic circuit for the following:

(a) Half adder using logic gates

Half Adder

With the help of half adder, we can design circuits that are capable of performing simple addition with the help of logic gates.

Let us first take a look at the addition of single bits.

0+0 = 0

0+1 = 1

1+0 = 1

1+1 = 10

These are the least possible single-bit combinations. But the result for 1+1 is 10. Though this problem can be solved with the help of an EXOR Gate, if you do care about the output, the sum result must be re-written as a 2-bit output.

Thus, the above equations can be written as

0+0 = 00

0+1 = 01

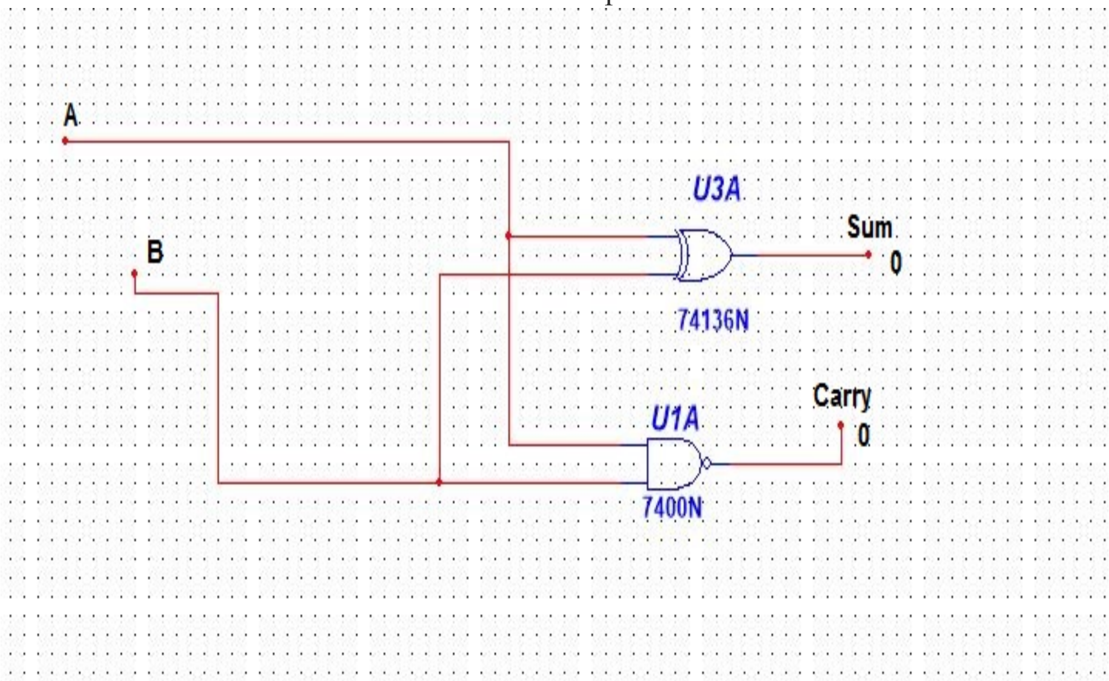
1+0 = 01

1+1 = 10

Here the output '1'of '10' becomes the carry-out. The result is shown in a truth-table below. 'SUM' is the normal output and 'CARRY' is the carry-out.

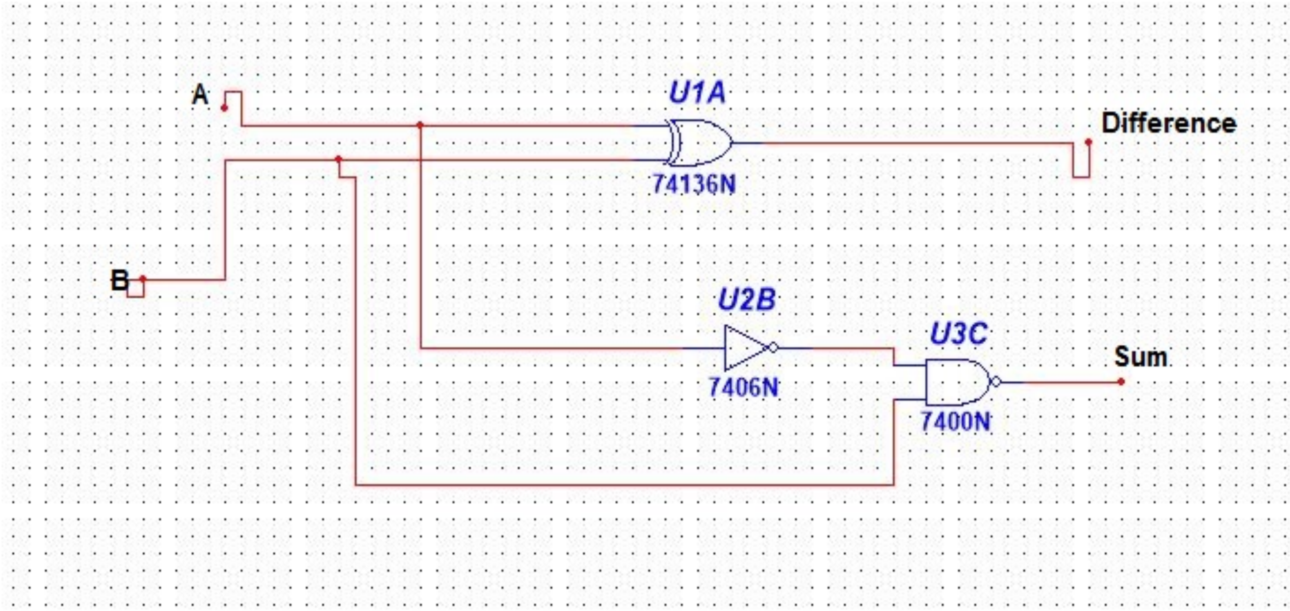
INPUTS		**OUTPUTS**	
A	**B**	**SUM**	**CARRY**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

From the equation, it is clear that this 1-bit adder can be easily implemented with the help of EXOR Gate for the output 'SUM' and an AND Gate for the carry. Take a look at the implementation below



(b) Half-subtractor using logic gate

The designing of half subtractor can be done by using logic gates like NAND gate & Ex-OR gate. In order to design this half subtractor circuit, we have to know the two concepts namely difference and borrow.



If we monitor cautiously, it is fairly clear that the variety of operation executed by this circuit which is accurately related to the EX-OR gate operation. Therefore, we can simply use the EX-OR gate for making difference. In the same way, the borrow produced by half adder circuit can be simply attained by using the blend of logic gates like AND- gate and NOT-gate.

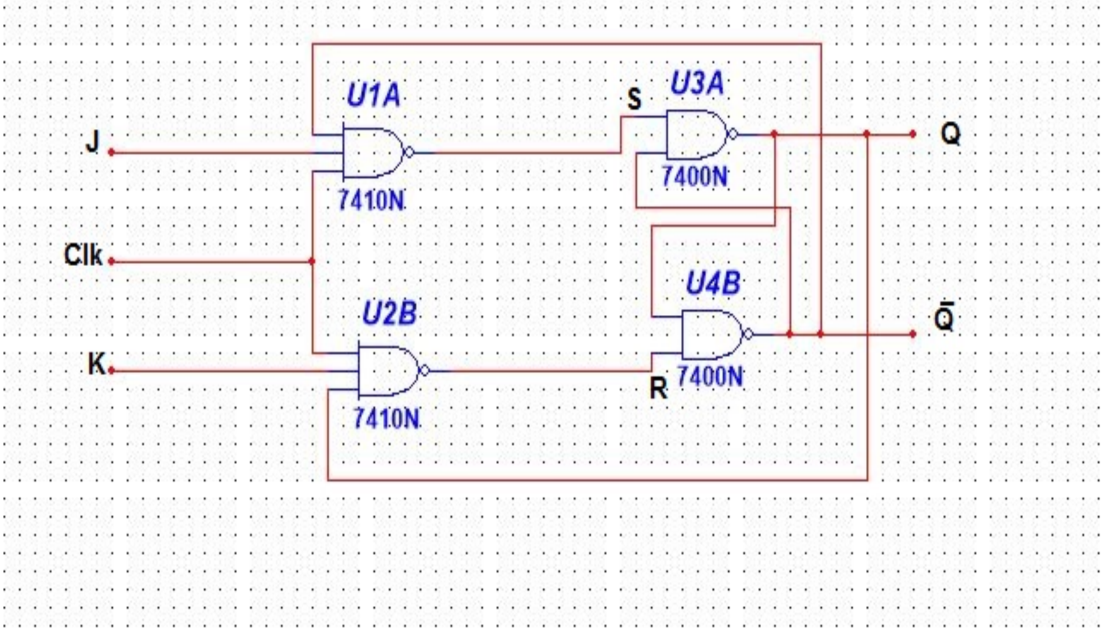
Truth Table

The half subtractor truth table explanation can be done by using the logic gates like EX-OR logic gate and AND gate operation followed by NOT gate.

First Bit	Second Bit	Difference (EX-OR Out)	Borrow (NAND Out)
0	0	0	0
1	0	1	0
0	1	1	1
1	1	0	0

(c) J K Flip flop

The JK flip flop is basically a gated SR flip-flop with the addition of a clock input circuitry that prevents the illegal or invalid output condition that can occur when both inputs S and R are equal to logic level “1”. Due to this additional clocked input, a JK flip-flop has four possible input combinations, “logic 1”, “logic 0”, “no change” and “toggle”. The symbol for a JK flip flop is similar to that of an *SR Bistable Latch* as seen in the previous tutorial except for the addition of a clock input.

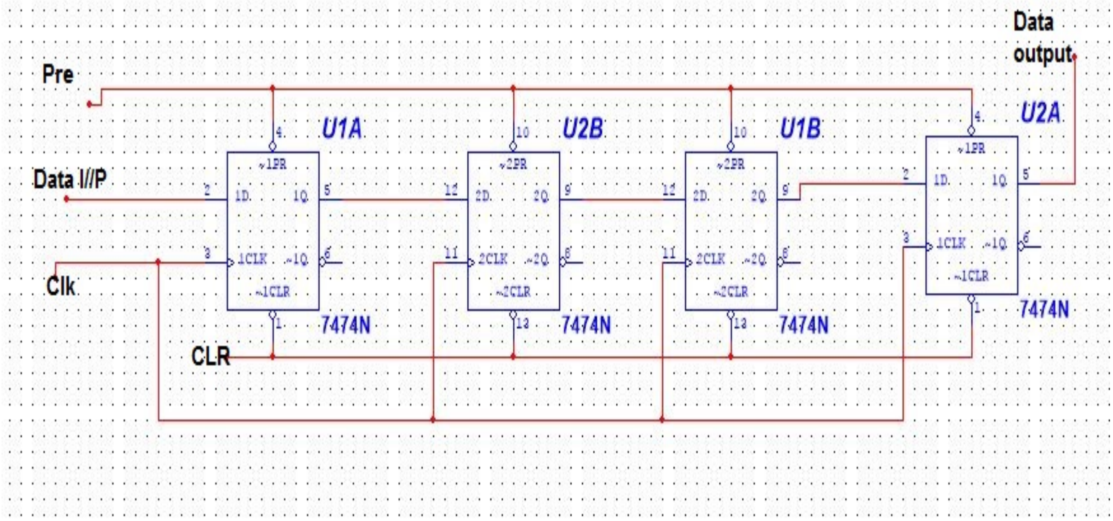


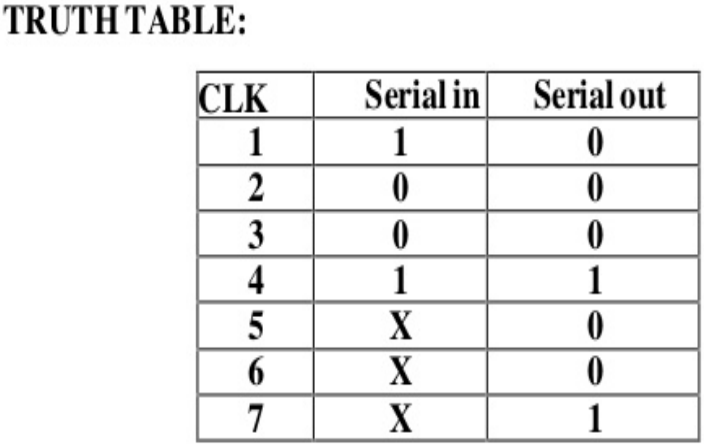
The Truth Table for the JK Function

	Clock	Input		Output		Description
	Clk	J	K	Q	Q	
same as for the Jk Latch	X	0	0	1	0	Memory no change
	X	0	0	0	1	
	¯↓_	0	1	1	0	Reset Q » 0
	X	0	1	0	1	
	¯↓_	1	0	0	1	Set Q » 1
	X	1	0	1	0	
toggle action	¯↓_	1	1	0	1	Toggle
	¯↓_	1	1	1	0	

(D) Serial in-serial Out shift register

The serial in/serial out shift register accepts data serially – that is, one bit at a time on a single line. It produces the stored information on its output also in serial form.

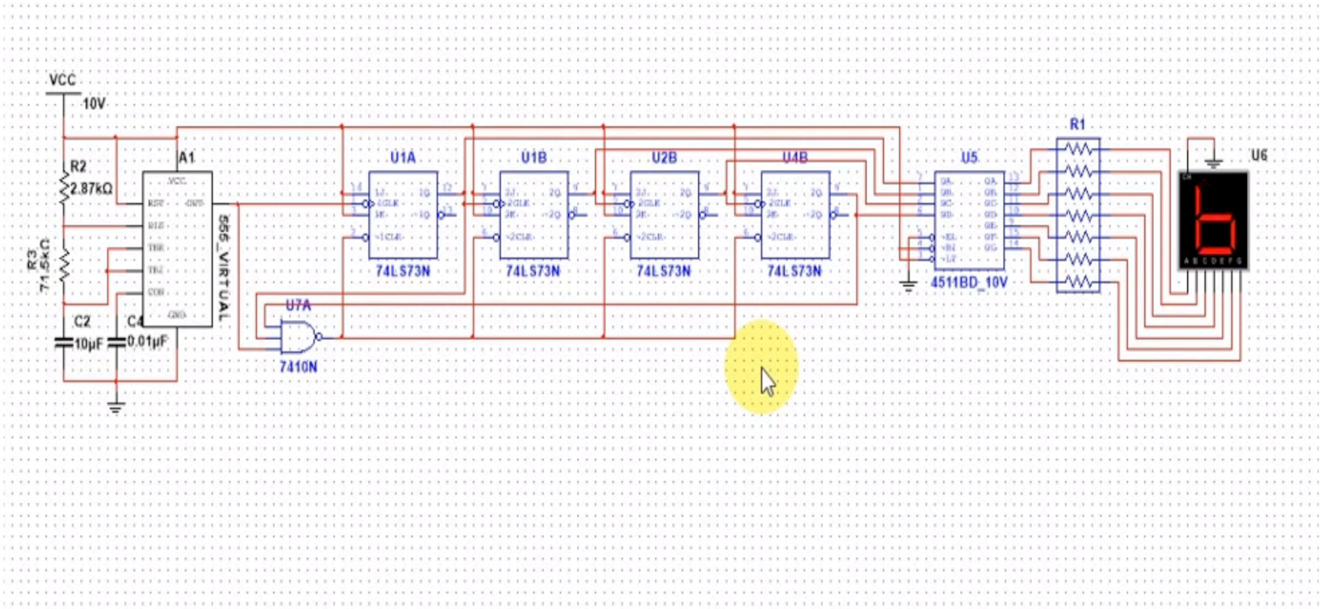




TRUTH TABLE:

CLK	Serial in	Serial out
1	1	0
2	0	0
3	0	0
4	1	1
5	X	0
6	X	0
7	X	1

(E) Synchronous BCD Counter


VCC
10V
R2
2.87kΩ
R3
71.5kΩ
C2
10µF
C4
0.01µF
A1
555_VIRTUAL
U1A
74LS73N
U1B
74LS73N
U2B
74LS73N
U4B
74LS73N
U7A
7410N
U5
4511BD_10V
R1
U6