EXPERIMENT NO.6

Title of experiment: Design of half adder and full adder in proteus.

Equipments required: Proteus 8

Theory:

1) The half adder circuit has two inputs: A and B, which add two input digits and generates a carry and a sum. The full adder circuit has three inputs: A and C, which add three input numbers and generates a carry and sum. This article gives detailed information about what is the purpose of a half adder and full adder in tabular forms and even in circuit diagrams too. It is already mentioned that the main and crucial purpose of adders is addition. Below are the detailed half adder and full adder theory.
2)

---Schematic representation of half adder----

A B S(sum) C(carry)

$$\begin{array}{cccc} 0 & 0 & 1 & 1 \\ \underline{+0} & \underline{+1} & \underline{+0} & \underline{+1} \\ 0 & 1 & 1 & (\text{carry})\ 1\ 0 \end{array}$$

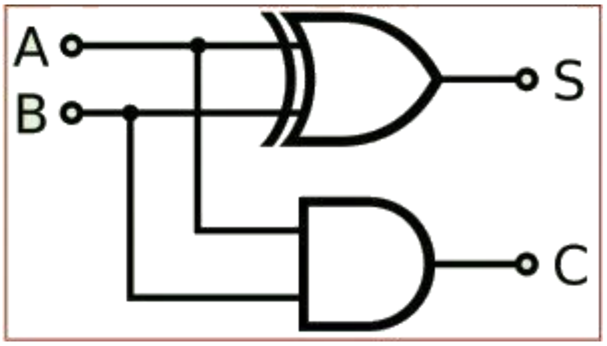
2) Full adder is a digital circuit used to calculate the sum of three binary bits. Full adders are complex and difficult to implement when compared to half adders. Two of the three bits are same as before which are A, the augend bit and B, the addend bit. The additional third bit is carry bit from the previous stage and is called 'Carry' – in generally represented by CIN. It calculates the sum of three bits along with the carry. The output carry is called Carry – out and is represented by Carry OUT.
The block diagram of a full adder with A, B and CIN as inputs and S, Carry OUT as outputs is shown below.

4) ---Schematic representation of full adder----

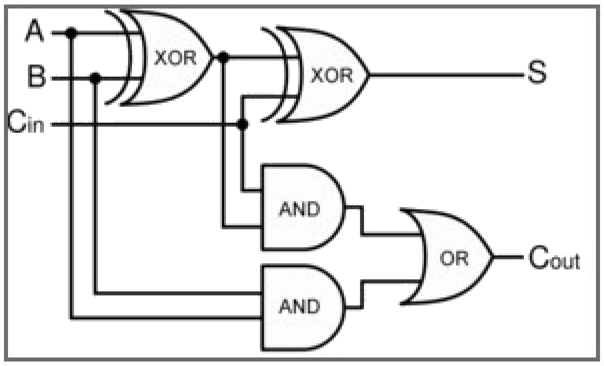
Input			Output	
A	B	Cin	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Implementation:

1) If A and B are binary inputs to the half adder, then the logic function to calculate sum S is Ex – OR of A and B and logic function to calculate carry C is AND of A and B. Combining these two, the logical circuit to implement the combinational circuit of half adder is shown below.



2) full adder diagram:



Full Adder Logic Diagram

3) Full Adder

The full adder is a little more difficult to implement than a half adder. The main difference between a half adder and a full adder is that the full-adder has three inputs and two outputs. The two inputs are A and B, and the third input is a carry input C_{IN}. The output carry is designated as C_{OUT}, and the normal output is designated as S.

The **truth table** of the Full Adder Circuit is shown below.

A	B	CIN	COUT	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

The output S is an EX-OR between the input A and the half-adder SUM output B. The COUT will be true only if any of the two inputs out of the three are HIGH or at logic 1.
Thus, a full adder circuit can be implemented with the help of two half adder circuits. The first half adder circuit will be used to add A and B to produce a partial sum. The second half adder logic can be used to add CIN to the sum produced by the first half adder circuit. Finally, the output S is obtained.
If any of the half adder logic produces a carry, there will be an output carry. Thus, COUT will be an OR function of the half adder CARRY outputs.

4) There are two inputs and two outputs in a Half Adder. Inputs are named as A and B, and the outputs are named as Sum (S) and Carry (C). The Sum is X-OR of the input A and B. Carry is AND of the input A and B. With the help of half adder, one can design a circuit that is 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 =10. This problem can be solved with the help of an EX-OR gate. The sum results can be re-written as a 2-bit output.
Thus the above combination 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. SUM is the normal output and the CARRY is the carry-out.
The truth table of the half adder :-

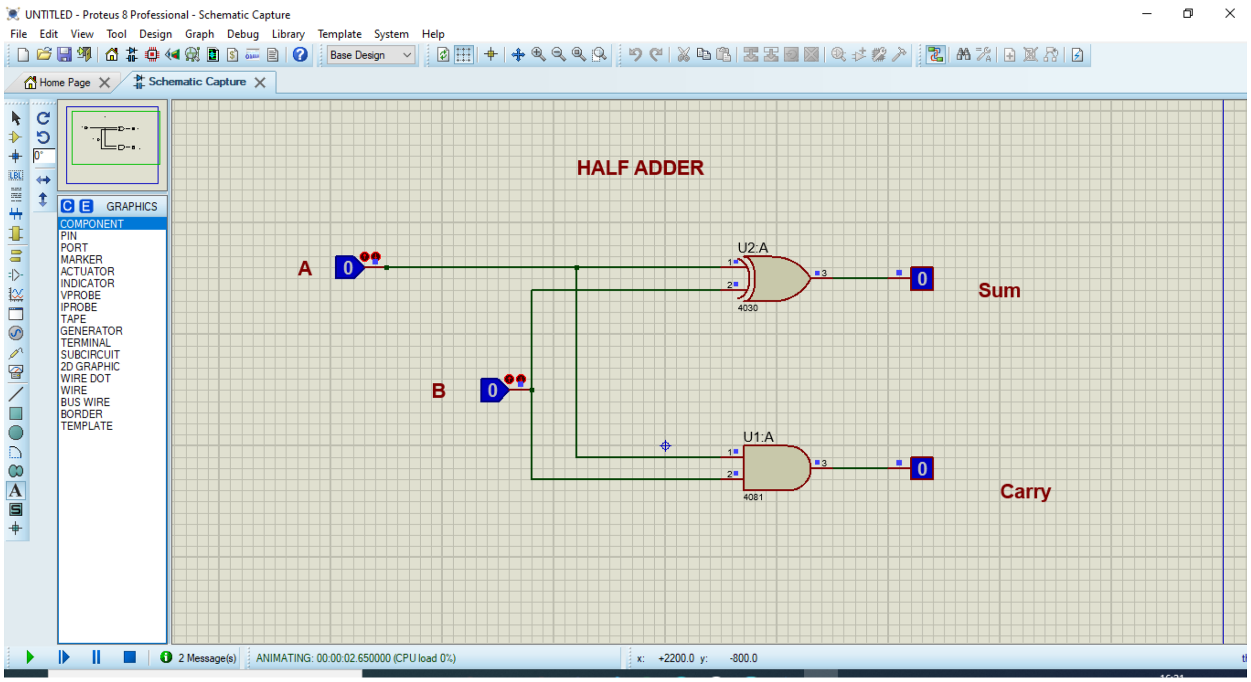
Inputs		Outputs	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

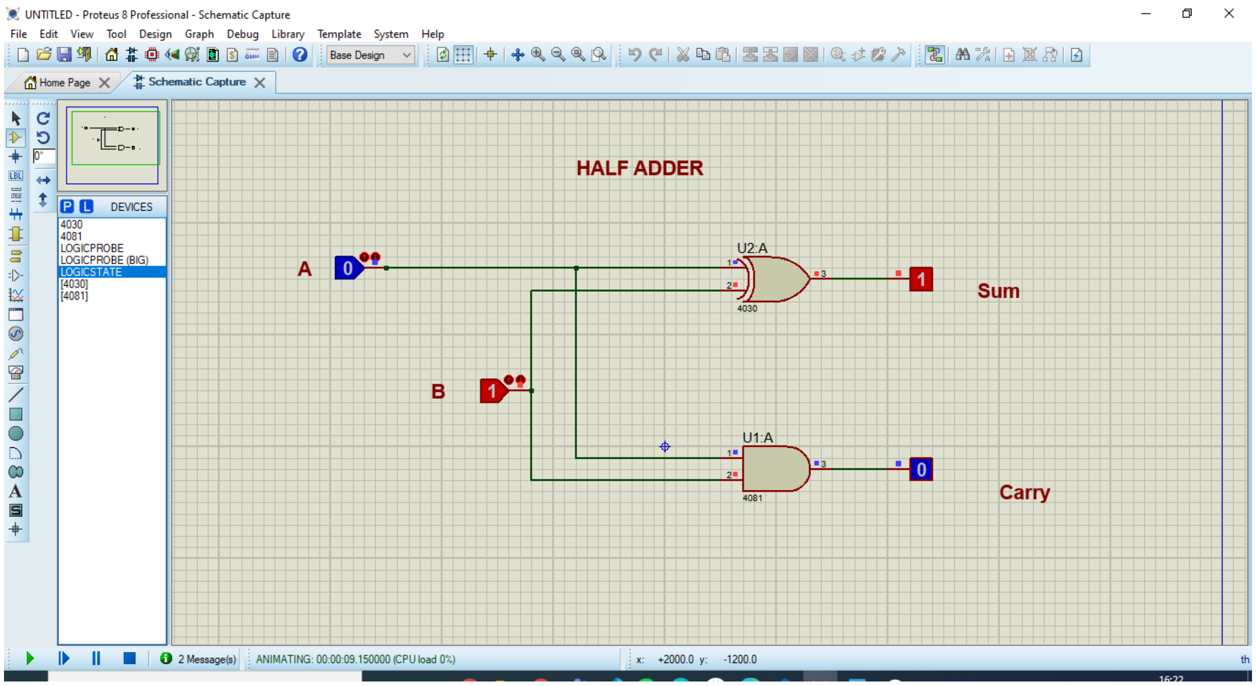
The half adder circuit :

The main disadvantage of this circuit is that it can only add two inputs and if there is any carry, it is neglected. Thus, the process is incomplete.

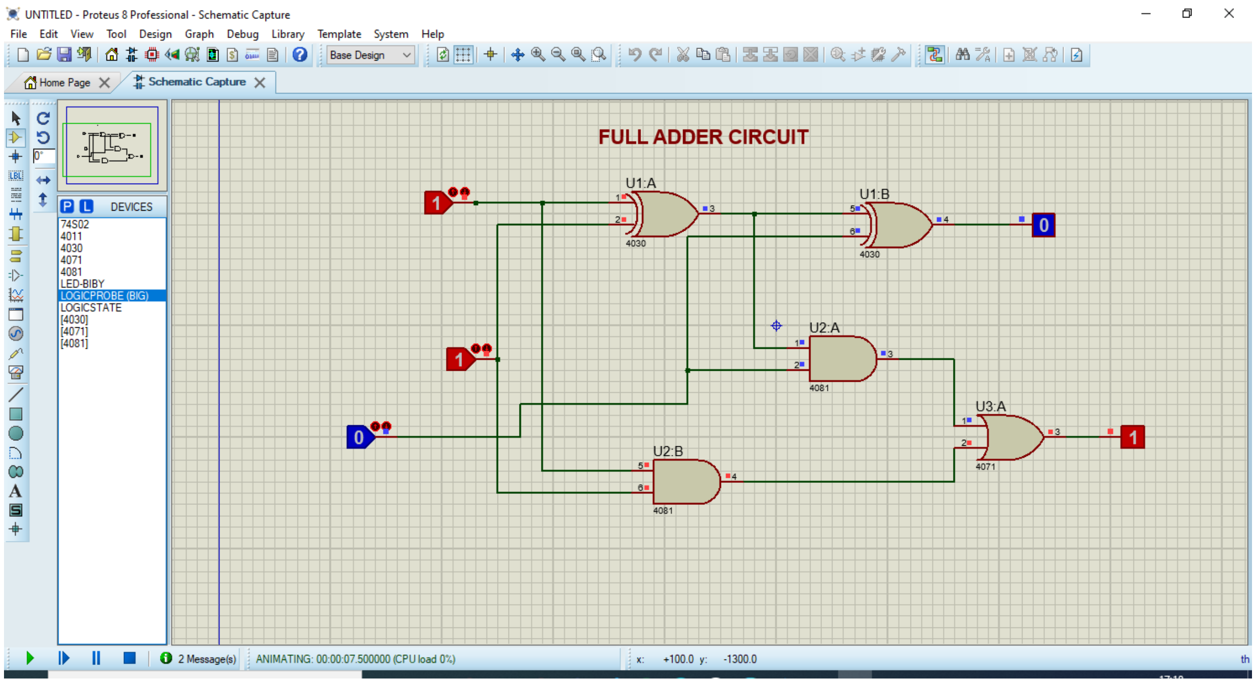
Snapshots:

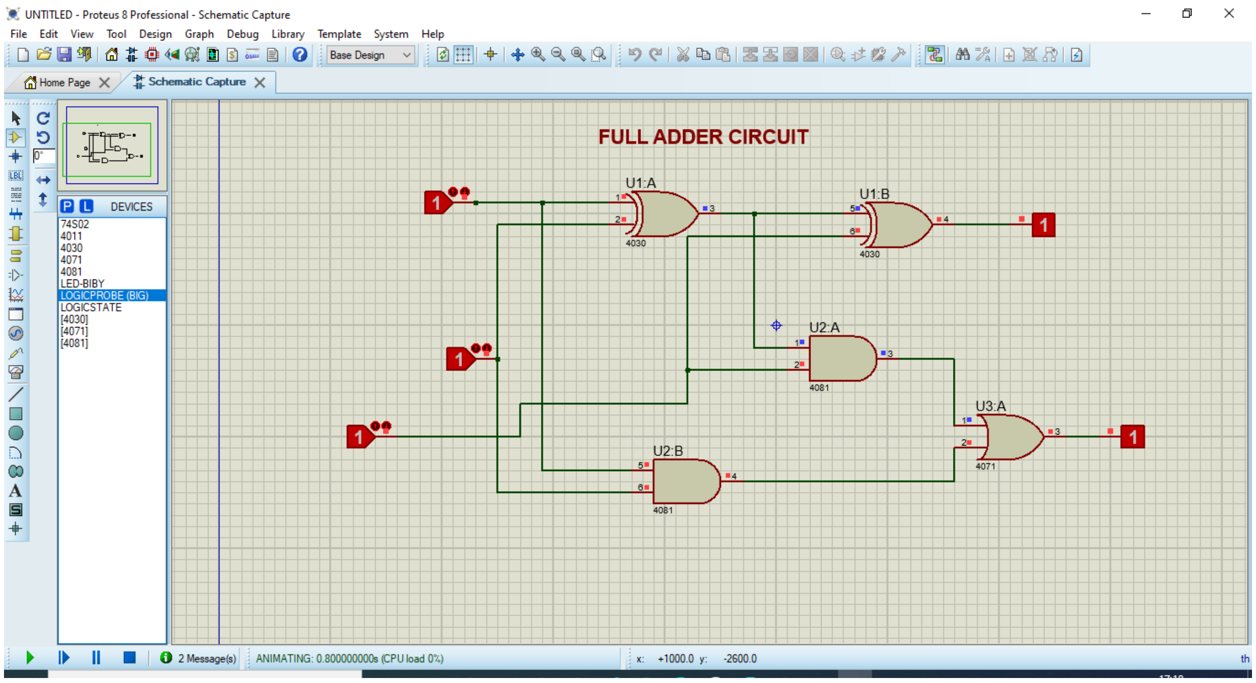
Half adders:

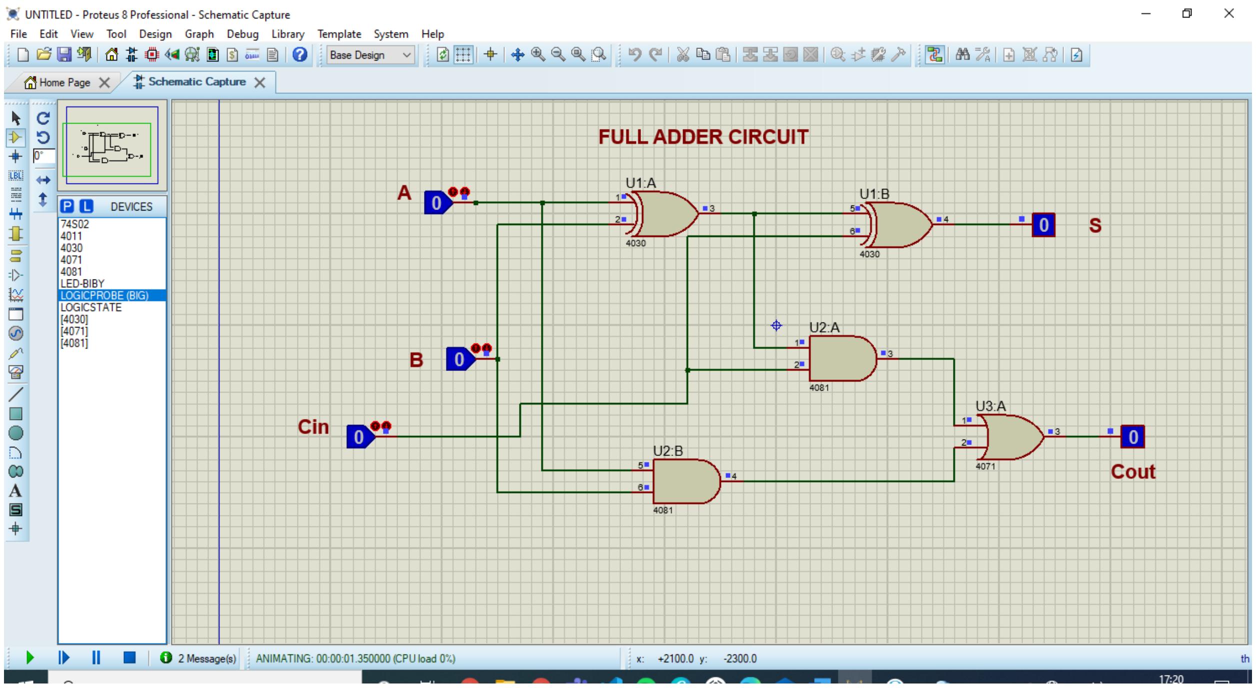



UNTITLED - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Home Page
Schematic Capture
DEVICES
4030
4081
LOGICPROBE
LOGICPROBE (BIG)
LOGICSTATE
[4030]
[4081]
HALF ADDER
A
0
B
1
U2:A
4030
1
Sum
U1:A
4081
0
Carry
2 Message(s)
ANIMATING: 00:00:09.150000 (CPU load 0%)
x: +2000.0 y: -1200.0

Full adders:


UNTITLED - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Home Page
Schematic Capture
DEVICES
74S02
4011
4030
4071
4081
LED-BIBY
LOGICPROBE (BIG)
LOGICSTATE
[4030]
[4071]
[4081]
FULL ADDER CIRCUIT
U1:A
4030
U1:B
4030
U2:A
4081
U2:B
4081
U3:A
4071
1
1
0
0
1
2 Message(s)
ANIMATING: 00:00:07.500000 (CPU load 0%)
x: +100.0 y: -1300.0


UNTITLED - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Home Page
Schematic Capture
DEVICES
74S02
4011
4030
4071
4081
LED-BIBY
LOGICPROBE (BIG)
LOGICSTATE
[4030]
[4071]
[4081]
FULL ADDER CIRCUIT
U1:A
U1:B
U2:A
U2:B
U3:A
4030
4081
4071
1
1
1
1
1
2 Message(s)
ANIMATING: 0.800000000s (CPU load 0%)
x: +1000.0 y: -2600.0


UNTITLED - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Home Page
Schematic Capture
DEVICES
74S02
4011
4030
4071
4081
LED-BIBY
LOGICPROBE (BIG)
LOGICSTATE
[4030]
[4071]
[4081]
FULL ADDER CIRCUIT
A
B
Cin
S
Cout
U1:A
U1:B
U2:A
U2:B
U3:A
4030
4081
4071
0
0
0
0
0
2 Message(s)
ANIMATING: 00:00:01.350000 (CPU load 0%)
x: +2100.0 y: -2300.0

Conclusions:

This is the designing of half adders and full adders using various gates in Proteus.