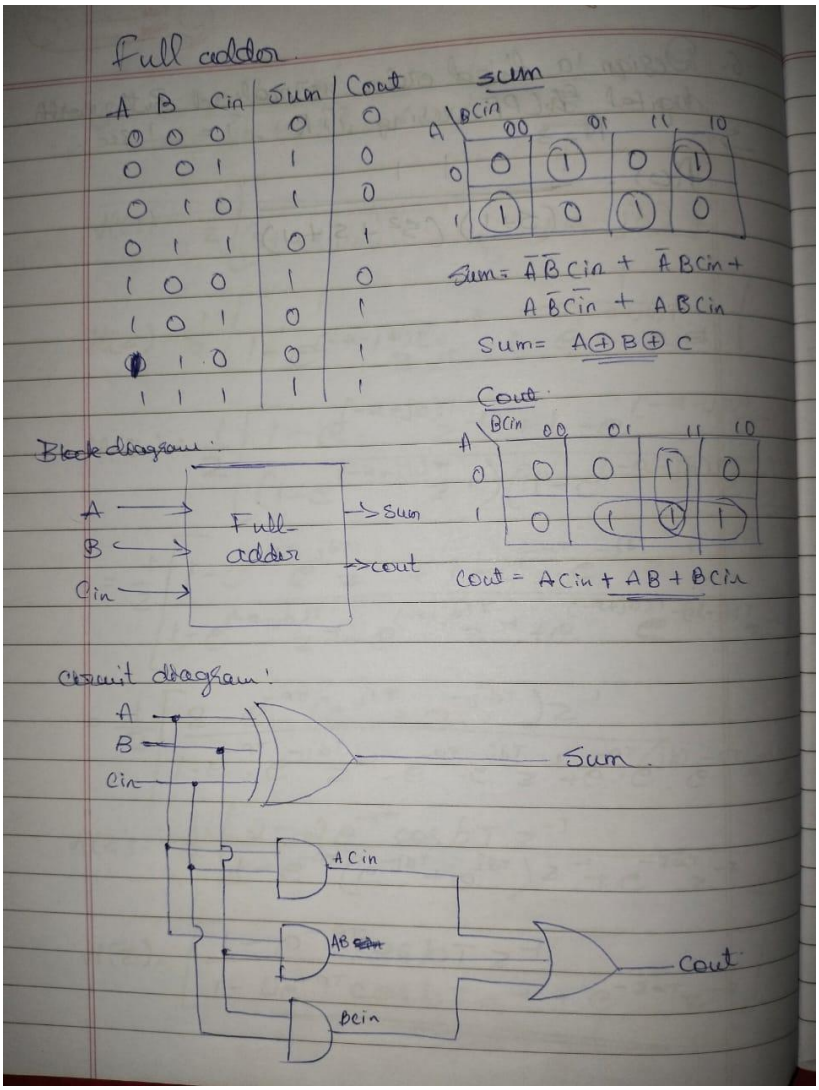
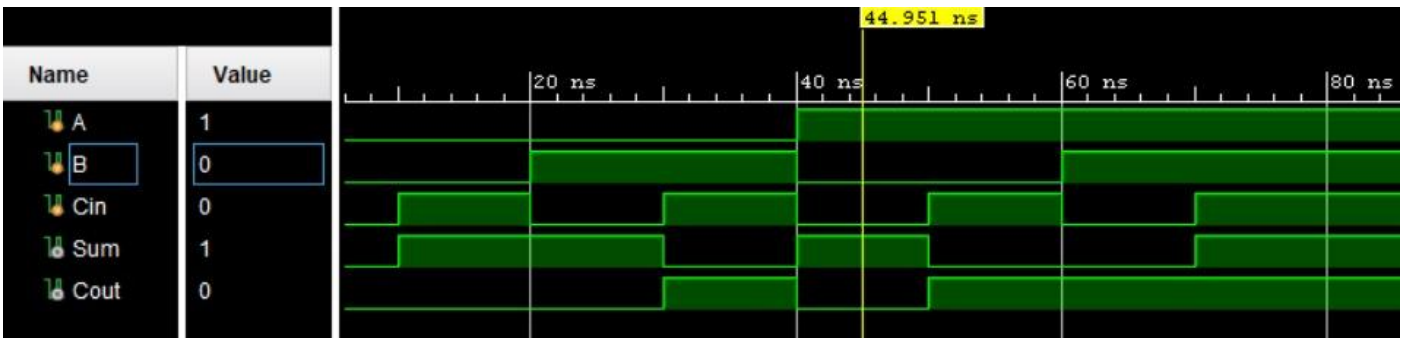
Full Adder

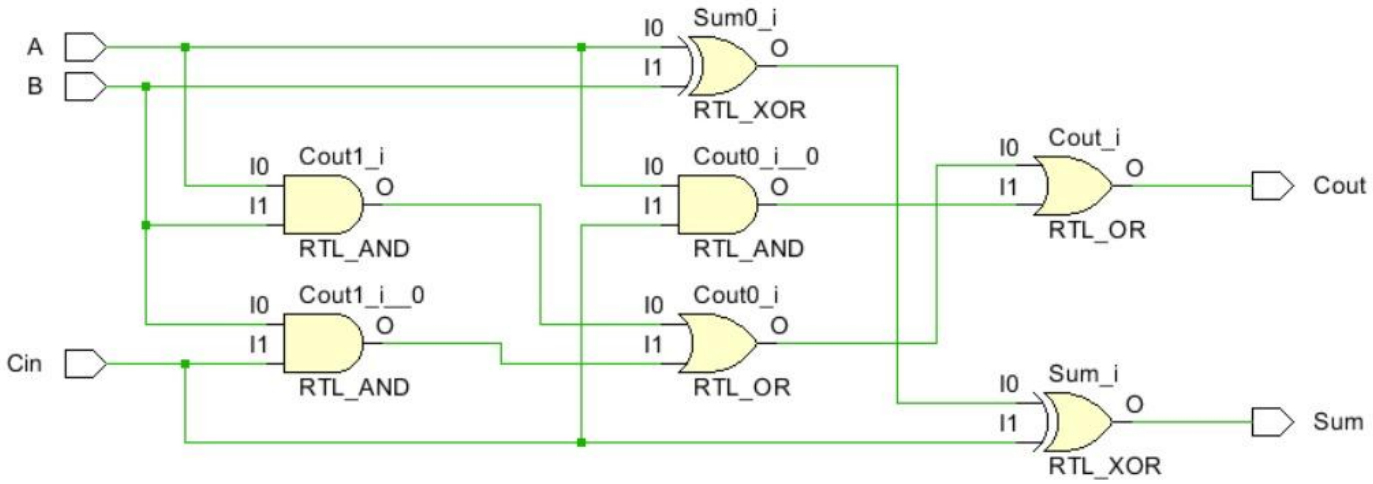



```
# run 1000ns
A   B  Cin | Sum  Cout
0 0  0   |  0    0
0 0  1   |  1    0
0 1  0   |  1    0
0 1  1   |  0    1
1 0  0   |  1    0
1 0  1   |  0    1
1 1  0   |  0    1
1 1  1   |  1    1
$finish called at time : 80 ns
```

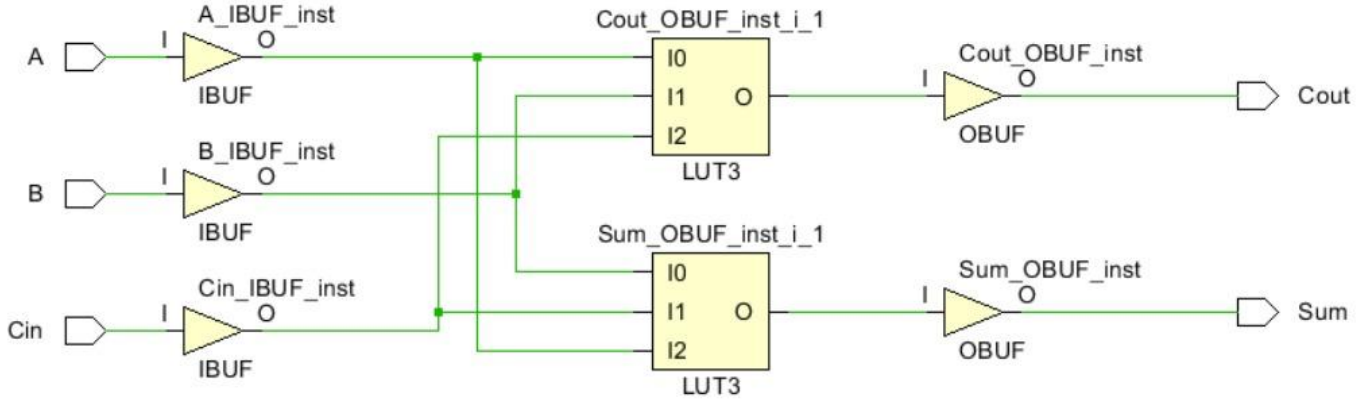
```
//Verilog code
module fulladder(A, B, Cin, Sum, Cout);
 input wire A, B, Cin;
 output wire Sum, Cout;
 assign Sum = A ^ B ^ Cin; assign Cout = (A & B) | (B & Cin) | (A & Cin);
endmodule
```



RTL analysis schematic:



Synthesis schematic:



Look Up Table:

Cell Properties

Sum_OBUF_inst_i_1

I2	I1	I0	O=I0 & !I1 & !I2 + !I0 & I1 & !I2 + !I0 & !I1 & I2 + I0 & I1 & I2
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

LUT for Sum

Cell Properties

Cout_OBUF_inst_i_1

I2	I1	I0	O=I0 & I1 + I0 & I2 + I1 & I2
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

LUT for Cout