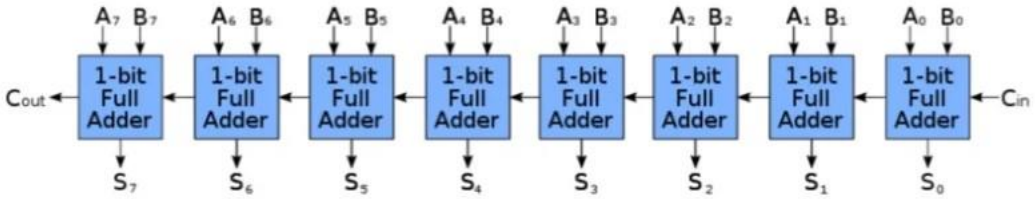
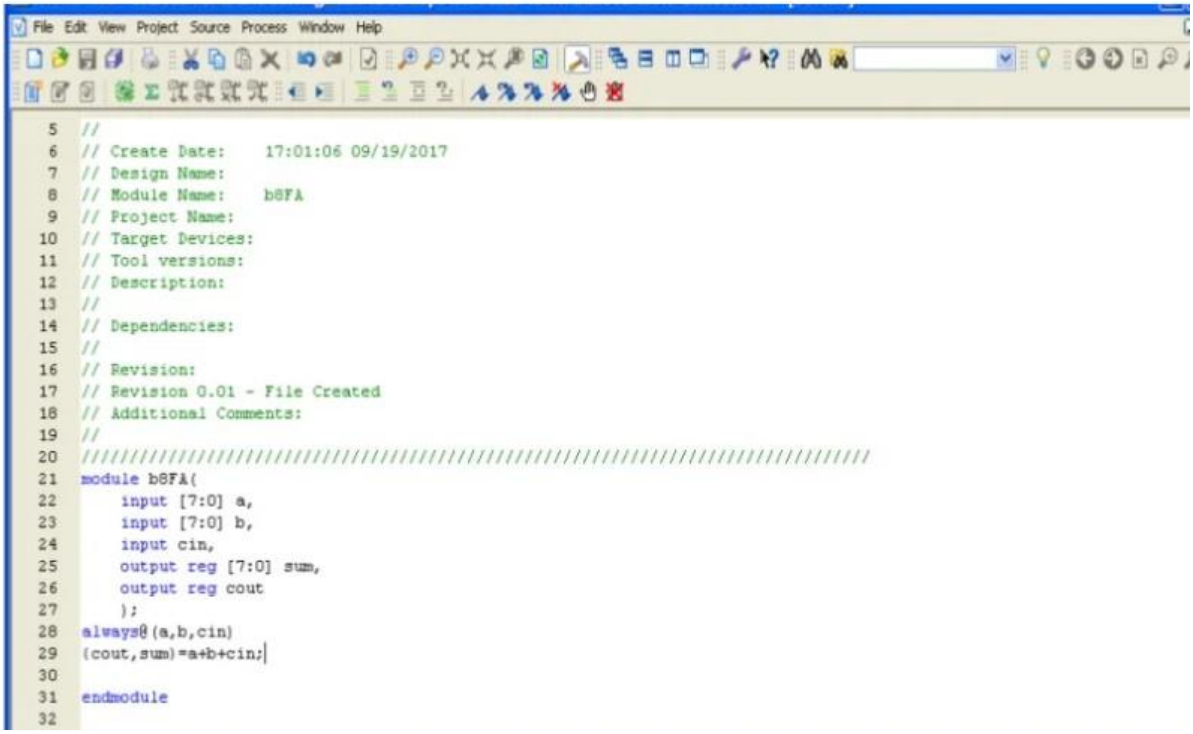
BLOCK DIAGRAM OF 8 BIT ADDER:



CODE FILE:

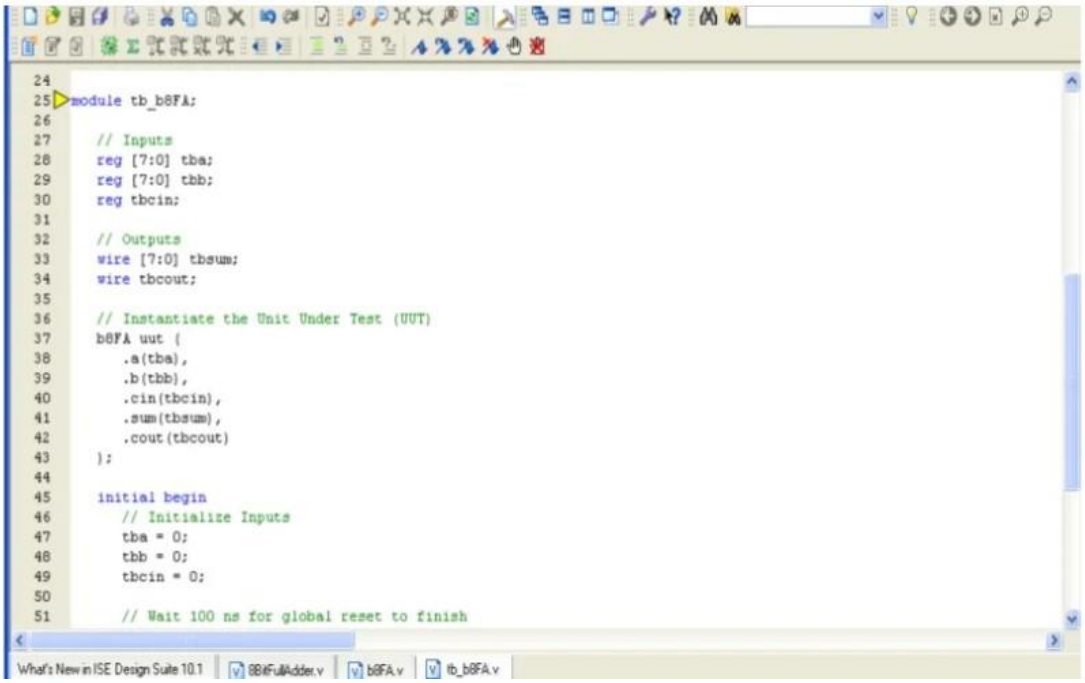


```
File Edit View Project Source Process Window Help

 5  //
 6  // Create Date:    17:01:06 09/19/2017
 7  // Design Name:
 8  // Module Name:    b8FA
 9  // Project Name:
10  // Target Devices:
11  // Tool versions:
12  // Description:
13  //
14  // Dependencies:
15  //
16  // Revision:
17  // Revision 0.01 - File Created
18  // Additional Comments:
19  //
20  //////////////////////////////////////////////////////////////////////////////////
21  module b8FA(
22      input [7:0] a,
23      input [7:0] b,
24      input cin,
25      output reg [7:0] sum,
26      output reg cout
27      );
28  always@(a,b,cin)
29  {cout,sum}=a+b+cin;
30
31  endmodule
32
```

TEST BENCH CODE:

We design a Test Bench code to check our code's simulated functional behavior and also we put our initial value in test bench and also we put a value which we want to add.

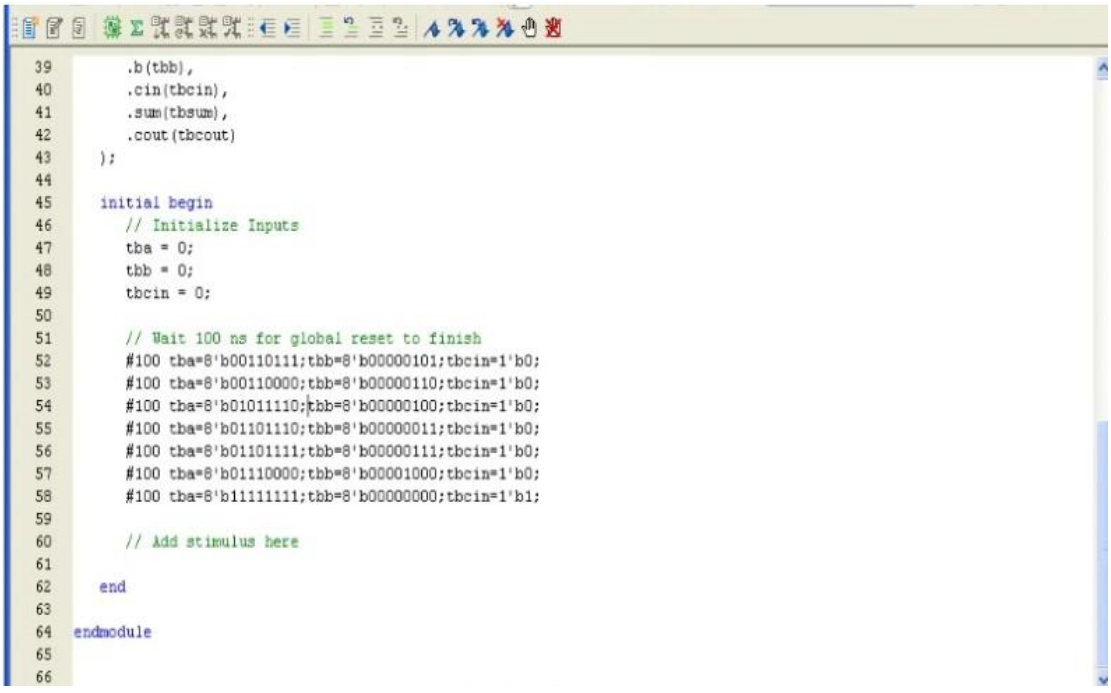


```
24
25  module tb_b8FA;
26
27      // Inputs
28      reg [7:0] tba;
29      reg [7:0] tbb;
30      reg tbcin;
31
32      // Outputs
33      wire [7:0] tbsum;
34      wire tbcout;
35
36      // Instantiate the Unit Under Test (UUT)
37      b8FA uut (
38          .a(tba),
39          .b(tbb),
40          .cin(tbcin),
41          .sum(tbsum),
42          .cout(tbcout)
43      );
44
45      initial begin
46          // Initialize Inputs
47          tba = 0;
48          tbb = 0;
49          tbcin = 0;
50
51          // Wait 100 ns for global reset to finish
```

What's New in ISE Design Suite 10.1 | 8BitFullAdder.v | b8FA.v | tb_b8FA.v

MORE SCREEN SHOT (tb_8bitFA):

Here we give input with **100ns** delay and also check the simulated result in the next screen shots.

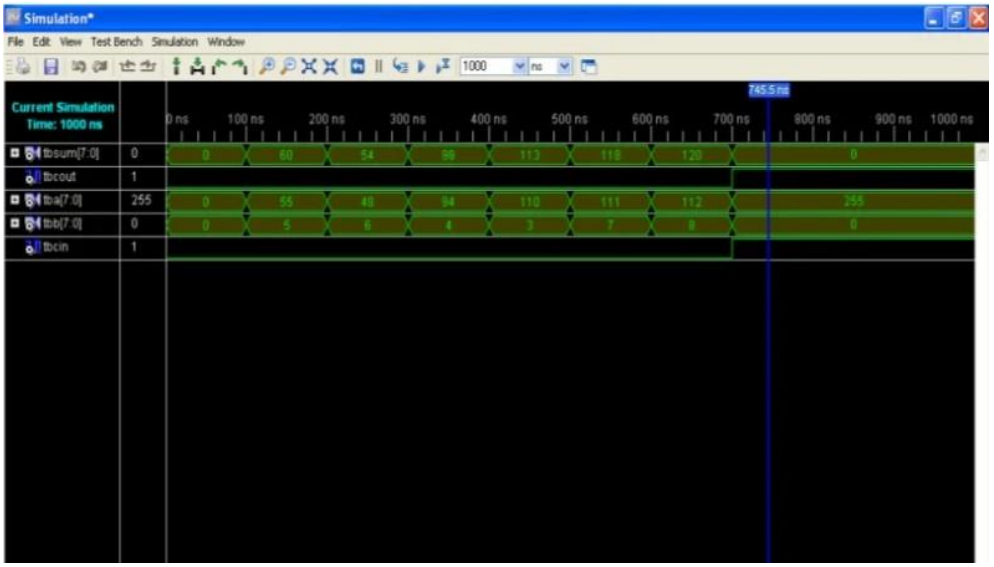


```
39          .b(tbb),
40          .cin(tbcin),
41          .sum(tbsum),
42          .cout(tbcout)
43      );
44
45      initial begin
46          // Initialize Inputs
47          tba = 0;
48          tbb = 0;
49          tbcin = 0;
50
51          // Wait 100 ns for global reset to finish
52          #100 tba=8'b00110111;tbb=8'b00000101;tbcin=1'b0;
53          #100 tba=8'b00110000;tbb=8'b00000110;tbcin=1'b0;
54          #100 tba=8'b01011110;tbb=8'b00000100;tbcin=1'b0;
55          #100 tba=8'b01101110;tbb=8'b00000011;tbcin=1'b0;
56          #100 tba=8'b01101111;tbb=8'b00000111;tbcin=1'b0;
57          #100 tba=8'b01110000;tbb=8'b00001000;tbcin=1'b0;
58          #100 tba=8'b11111111;tbb=8'b00000000;tbcin=1'b1;
59
60          // Add stimulus here
61
62      end
63
64  endmodule
65
66
```

SIMULATED RESULT:

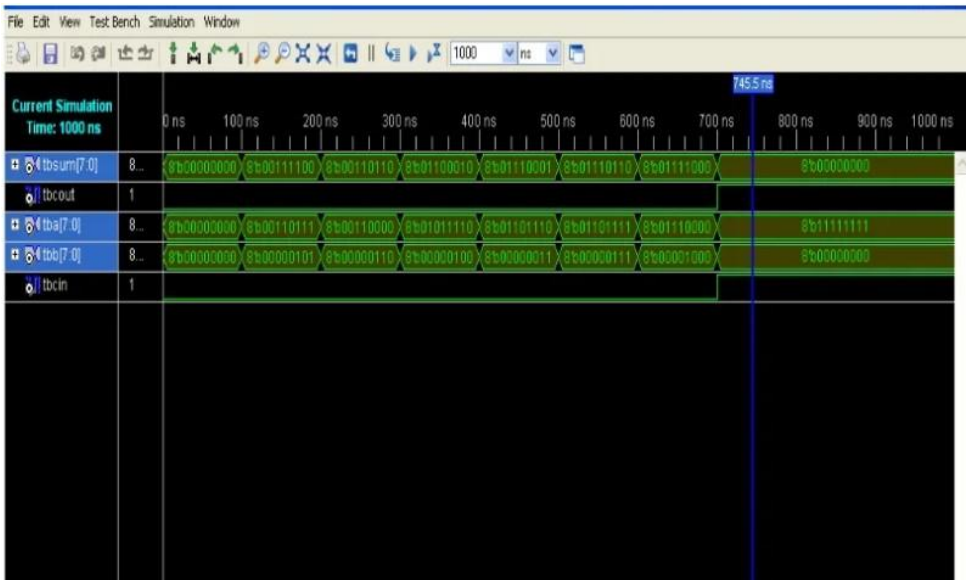
- DECIMAL RESULT:

Here we check the simulated result which is in decimal and also we can see clearly a carry result and also with **100ns** delay.



- BINARY RESULT:

We also check the result in binary by right click and selecting a binary instead of decimal. So, we get a simulated result in binary form.



Q: Why sum is zero when we add 1 in 255?

REASON:

As maximum bits show in 8'bit system **is 255=$(11111111)_2$** as with increment of 1 bit it become 9'bit number **256=$(100000000)_2$** so we get MSB as carry **1** and remaining 8'bits as zeros.