



**Course Code - CS223**

**Course Name - Digital Design**

**Section – 001**

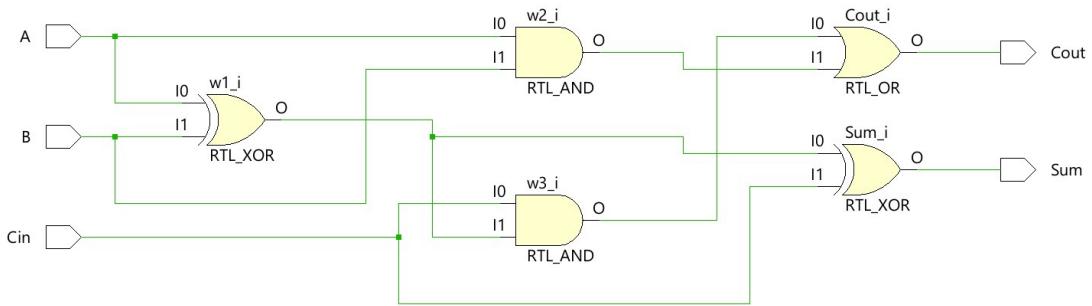
**Lab Number – 02**

**Name Surname – Moin Khan**

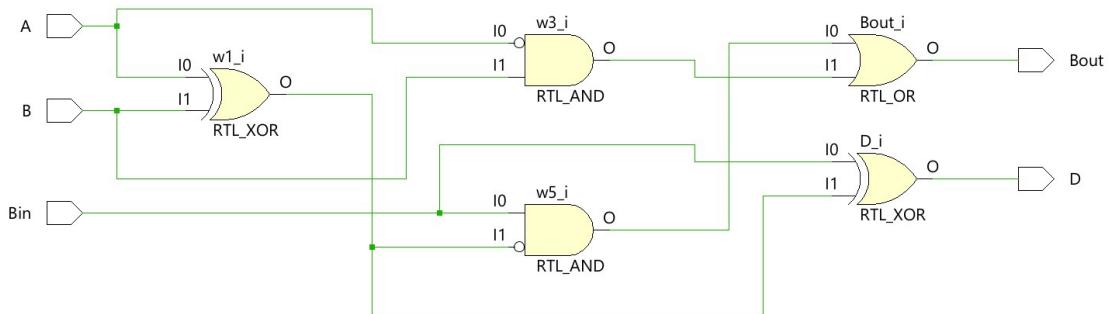
**Student Id – 22101287**

**Date – 08/10/2023**

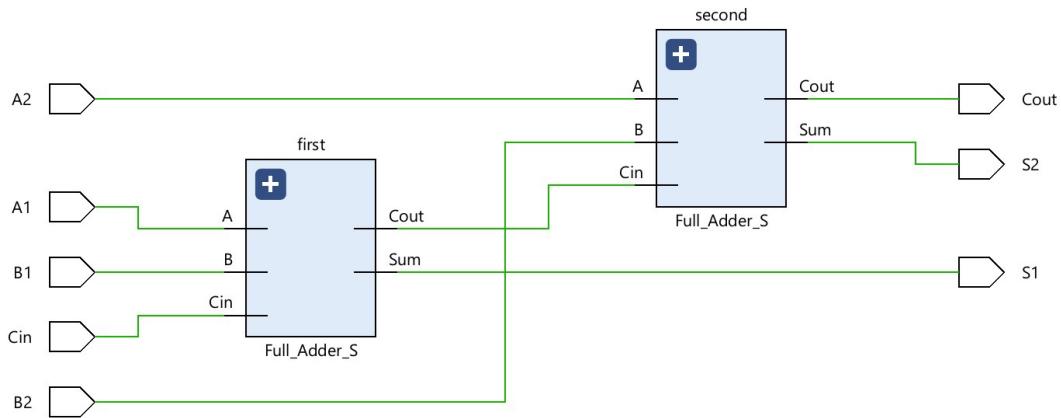
# 1. Full Adder Schematics



# 2. Full Subtractor Schematics



# 3. 2-Bit Full Adder



# 4. Behavioural Full Adder

## 4.1 Module

```
'timescale 1ns / 1ps
```

```
module Full_Adder_B( input wire A,B,Cin,  
output wire Sum,Cout);  
assign Cout = (A&B)+(B&Cin)+(Cin&A);  
assign Sum = A^B^Cin;  
  
endmodule
```

```
Full_Adder_B.v *  x Full_Adder_B_TB.v  x Full_Subtractor.v  x Full_Adder_S.v  x Full_Adder_S_TB.v  x Full_Subtractor_TB.v
C:/Users/moink/OneDrive/Desktop/CS223/Labs/Lab_2/Lab_2.srcc/sources_1/new/Full_Adder_B.v

Q | F | ← | → | X | D | C | X | // | E | Q |

1 `timescale 1ns / 1ps
2
3
4 module Full_Adder_B( input wire A,B,Cin,
5 output wire Sum,Cout);
6 assign Cout = (A&B)+(B&Cin)+(Cin&A);
7 assign Sum = A^B^Cin;
8
9
10 endmodule
11
```

## 4.2 Test Bench

```
module Full_Adder_B_TB();
reg A,B,Cin;
wire Sum,Cout;
Full_Adder_B tb(A,B,Cin,Sum,Cout);
initial begin
assign A = 0;
assign B = 0;
assign Cin = 0;
#5;
assign A = 0;
assign B = 0;
assign Cin = 1;
#5;
assign A = 0;
assign B = 1;
assign Cin = 0;
#5;
```

```
assign A = 0;
assign B = 1;
assign Cin = 1;
#5;
assign A = 1;
assign B = 0;
assign Cin = 0;
#5;
assign A = 1;
assign B = 0;
assign Cin = 1;
#5;
assign A = 1;
assign B = 1;
assign Cin = 0;
#5;
assign A = 1;
assign B = 1;
assign Cin = 1;
#5;
end
endmodule
```

```

11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 @ ///////////////////////////////////////////////////////////////////
21
22
23 module Full_Adder_B_TB();
24 reg A,B,Cin;
25 wire Sum,Cout;
26 Full_Adder_B tb(A,B,Cin,Sum,Cout);
27 initial begin
28 assign A = 0;
29 assign B = 0;
30 assign Cin = 0;
31 #5;
32 assign A = 0;
33 assign B = 0;
34 assign Cin = 1;
35 #5;
36 assign A = 0;
37 assign B = 1;
38 assign Cin = 0;
39 #5;
40 assign A = 0;
41 assign B = 1;
42 assign Cin = 1;
43 #5;
44 assign A = 1;
45 assign B = 0;
46 assign Cin = 0;
47 #5;
48 assign A = 1;
49 assign B = 0;
50 assign Cin = 1;
51 #5;
52 assign A = 1;
53 assign B = 1;
54 assign Cin = 0;
55 #5;
56 assign A = 1;
57 assign B = 1;
58 assign Cin = 1;
59 #5;
60
61 end
62 endmodule
63

```

## 5. Structural Full Adder

### 5.1 Module

```

module Full_Adder_S( input wire A,B,Cin, output wire Sum , Cout);
wire w1,w2,w3;
assign w1 = A^B;

```

```

assign w2 = A&B;
assign Sum = w1^Cin;
assign w3 = Cin&w1;
assign Cout = w3|w2;
endmodule

19 ' //
20 ' //////////////////////////////////////////////////////////////////
21 '
22 '
23 module Full_Adder_S( input wire A,B,Cin, output wire Sum , Cout);
24   wire w1,w2,w3;
25   assign w1 = A^B;
26   assign w2 = A&B;
27   assign Sum = w1^Cin;
28   assign w3 = Cin&w1;
29   assign Cout = w3|w2;
30 endmodule
31 '

```

## 5.2 Test Bench

```

module Full_Adder_S_TB();
reg A,B,Cin;
wire Sum,Cout;
Full_Adder_S tb(A,B,Cin,Sum,Cout);
initial begin
  assign A = 0;
  assign B = 0;
  assign Cin = 0;
  #5;
  assign A = 0;
  assign B = 0;
  assign Cin = 1;
  #5;

```

```
assign A = 0;  
assign B = 1;  
assign Cin = 0;  
#5;  
assign A = 0;  
assign B = 1;  
assign Cin = 1;  
#5;  
assign A = 1;  
assign B = 0;  
assign Cin = 0;  
#5;  
assign A = 1;  
assign B = 0;  
assign Cin = 1;  
#5;  
assign A = 1;  
assign B = 1;  
assign Cin = 0;  
#5;  
assign A = 1;  
assign B = 1;  
assign Cin = 1;  
#5;  
  
end  
endmodule
```

```

module Full_Adder_S_TB();
reg A,B,Cin;
wire Sum,Cout;
Full_Adder_S tb(A,B,Cin,Sum,Cout);
initial begin
    assign A = 0;
    assign B = 0;
    assign Cin = 0;
#5;
    assign A = 0;
    assign B = 0;
    assign Cin = 1;
#5;
    assign A = 0;
    assign B = 1;
    assign Cin = 0;
#5;
    assign A = 0;
    assign B = 1;
    assign Cin = 1;
#5;
    assign A = 1;
    assign B = 0;
    assign Cin = 0;
#5;
    assign A = 1;
    assign B = 0;
    assign Cin = 1;
#5;
    assign A = 1;
    assign B = 1;
    assign Cin = 0;
#5;
    assign A = 1;
    assign B = 1;
    assign Cin = 1;
#5;
end
endmodule

```

## 6.0 Full Subtractor Structural

### 6.1 Module

```

module Full_Subtractor( input wire A,B,Bin, output wire D ,Bout);
wire w1,w2,w3,w4,w5;
assign w1 = A^B;
assign w2 = ~A;
assign w3 = w2&B;

```

```

assign w4 = ~w1;
assign D = Bin^w1;
assign w5 = Bin&w4;
assign Bout = w5|w3;

endmodule

```

---

```

21
22 module Full_Subtractor( input wire A,B,Bin, output wire D ,Bout);
23   wire w1,w2,w3,w4,w5;
24   assign w1 = A^B;
25   assign w2 = ~A;
26   assign w3 = w2&B;
27   assign w4 = ~w1;
28   assign D = Bin^w1;
29   assign w5 = Bin&w4;
30   assign Bout = w5|w3;
31 endmodule
32

```

## 6.2 Test Bench

```

module Full_Subtractor_TB();
reg A,B,Bin;
wire D,Bout;
Full_Subtractor tb(A,B,Bin,D,Bout);
initial begin
  assign A = 0;
  assign B = 0;
  assign Bin = 0;
  #5;
  assign A = 0;
  assign B = 0;
  assign Bin = 1;
  #5;

```

```
assign A = 0;  
assign B = 1;  
assign Bin = 0;  
#5;  
assign A = 0;  
assign B = 1;  
assign Bin = 1;  
#5;  
assign A = 1;  
assign B = 0;  
assign Bin = 0;  
#5;  
assign A = 1;  
assign B = 0;  
assign Bin = 1;  
#5;  
assign A = 1;  
assign B = 1;  
assign Bin = 0;  
#5;  
assign A = 1;  
assign B = 1;  
assign Bin = 1;  
#5;  
  
end  
endmodule
```

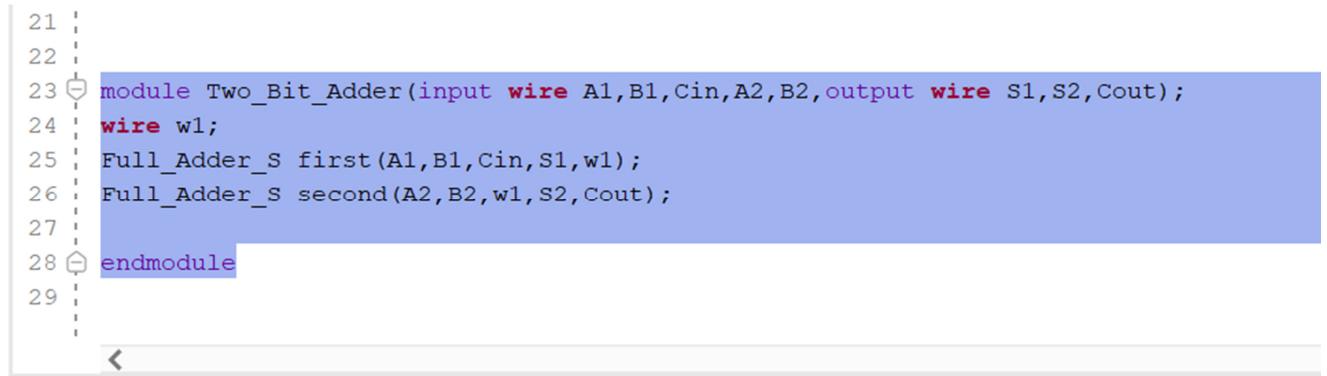
```
21 //////////////////////////////////////////////////////////////////
21 module Full_Subtractor_TB();
22   reg A,B,Bin;
23   wire D,Bout;
24   Full_Subtractor tb(A,B,Bin,D,Bout);
25 initial begin
26   assign A = 0;
27   assign B = 0;
28   assign Bin = 0;
29   #5;
30   assign A = 0;
31   assign B = 0;
32   assign Bin = 1;
33   #5;
34   assign A = 0;
35   assign B = 1;
36   assign Bin = 0;
37   #5;
38   assign A = 0;
39   assign B = 1;
40   assign Bin = 1;
41   #5;
42   assign A = 1;
43   assign B = 0;
44   assign Bin = 0;
45   #5;
46   assign A = 1;
47   assign B = 0;
48   assign Bin = 1;
49   #5;
50   assign A = 1;
51   assign B = 1;
52   assign Bin = 0;
53   #5;
54   assign A = 1;
55   assign B = 1;
56   assign Bin = 1;
57   #5;
58
59 end
60 endmodule
```

# 7.0 Two-Bit Full Adder Structural

## 7.1 Module

```
module Two_Bit_Adder(input wire A1,B1,Cin,A2,B2,output wire S1,S2,Cout);
wire w1;
Full_Adder_S first(A1,B1,Cin,S1,w1);
Full_Adder_S second(A2,B2,w1,S2,Cout);

endmodule
```



```
21
22
23 module Two_Bit_Adder(input wire A1,B1,Cin,A2,B2,output wire S1,S2,Cout);
24   wire w1;
25   Full_Adder_S first(A1,B1,Cin,S1,w1);
26   Full_Adder_S second(A2,B2,w1,S2,Cout);
27
28 endmodule
29
```

## 7.2 Test Bench

```
module Two_Bit_Adder_TB();
reg A1,A2,B1,B2,Cin;
wire S1,S2,Cout;
Two_Bit_Adder test(A1,B1,Cin,A2,B2,S1,S2,Cout);
initial begin
assign A1 = 0; assign B1 = 0 ; assign Cin = 0; assign A2 = 0; assign B2 = 0 ; #5;
assign A1 = 0; assign B1 = 0 ; assign Cin = 0; assign A2 = 0; assign B2 = 1 ; #5;
```



```
$finish;  
end  
  
endmodule
```

```
21 :  
22 :  
23 : module Two_Bit_Adder_TB();  
24 :   reg A1,A2,B1,B2,Cin;  
25 :   wire S1,S2,Cout;  
26 :   Two_Bit_Adder test(A1,B1,Cin,A2,B2,S1,S2,Cout);  
27 : initial begin  
28 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 0; assign A2 = 0; assign B2 = 0 ; #5;  
29 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 0; assign A2 = 0; assign B2 = 1 ; #5;  
30 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 0; assign A2 = 1; assign B2 = 0 ; #5;  
31 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 0; assign A2 = 1; assign B2 = 1 ; #5;  
32 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 1; assign A2 = 0; assign B2 = 0 ; #5;  
33 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 1; assign A2 = 0; assign B2 = 1 ; #5;  
34 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 1; assign A2 = 1; assign B2 = 0 ; #5;  
35 :   assign A1 = 0; assign B1 = 0 ; assign Cin = 1; assign A2 = 1; assign B2 = 1 ; #5;  
36 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 0; assign A2 = 0; assign B2 = 0 ; #5;  
37 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 0; assign A2 = 0; assign B2 = 1 ; #5;  
38 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 0; assign A2 = 1; assign B2 = 0 ; #5;  
39 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 0; assign A2 = 1; assign B2 = 1 ; #5;  
40 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 1; assign A2 = 0; assign B2 = 0 ; #5;  
41 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 1; assign A2 = 0; assign B2 = 1 ; #5;  
42 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 1; assign A2 = 1; assign B2 = 0 ; #5;  
43 :   assign A1 = 0; assign B1 = 1 ; assign Cin = 1; assign A2 = 1; assign B2 = 1 ; #5;  
44 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 0; assign A2 = 0; assign B2 = 0 ; #5;  
45 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 0; assign A2 = 0; assign B2 = 1 ; #5;  
46 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 0; assign A2 = 1; assign B2 = 0 ; #5;  
47 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 0; assign A2 = 1; assign B2 = 1 ; #5;  
48 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 1; assign A2 = 0; assign B2 = 0 ; #5;  
49 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 1; assign A2 = 0; assign B2 = 1 ; #5;  
50 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 1; assign A2 = 1; assign B2 = 0 ; #5;  
51 :   assign A1 = 1; assign B1 = 0 ; assign Cin = 1; assign A2 = 1; assign B2 = 1 ; #5;  
52 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 0; assign A2 = 0; assign B2 = 0 ; #5;  
53 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 0; assign A2 = 0; assign B2 = 1 ; #5;  
54 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 0; assign A2 = 1; assign B2 = 0 ; #5;  
55 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 0; assign A2 = 1; assign B2 = 1 ; #5;  
56 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 1; assign A2 = 0; assign B2 = 0 ; #5;  
57 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 1; assign A2 = 0; assign B2 = 1 ; #5;  
58 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 1; assign A2 = 1; assign B2 = 0 ; #5;  
59 :   assign A1 = 1; assign B1 = 1 ; assign Cin = 1; assign A2 = 1; assign B2 = 1 ; #5;  
60 : $finish;  
61 : end  
62 :  
63 :  
64 : endmodule  
--
```