DLD LAB WIN_2021

LAB-2

Aim: To simulate different types of logic gates of full adder, half subtractor, full subtractor.

Tools used: vivado software

Truth tables:

- **Full adder**:

A	B	C	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

- Half subtractor:

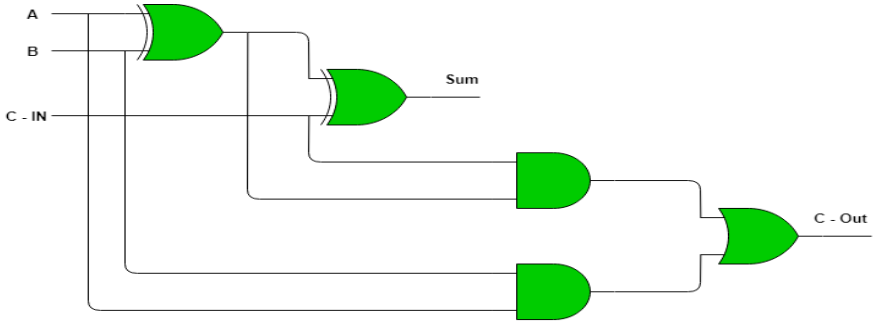
A	B	SUM	CARRY
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

- **Full subtractor:**

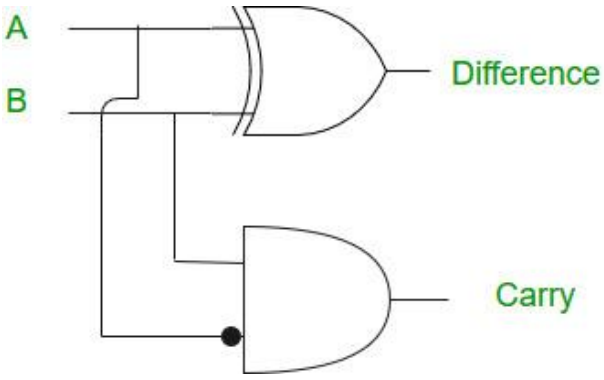
A	B	C	borrow	difference
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Circuit diagrams:

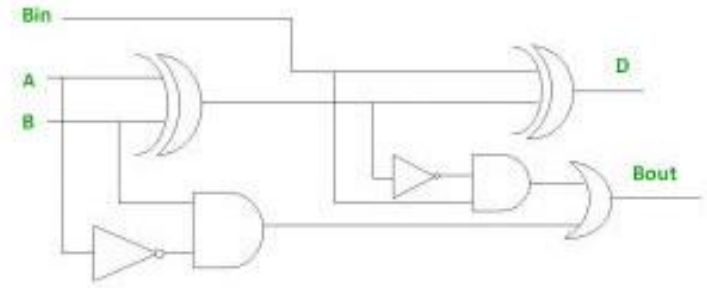
Full adder:



Half subtractor:



Full subtractor:



Codes:

Full adder:

C:/Users/91995/full_adder_14mar2021/full_adder_14mar2021.srcs/sources_1/new/fulladder.v

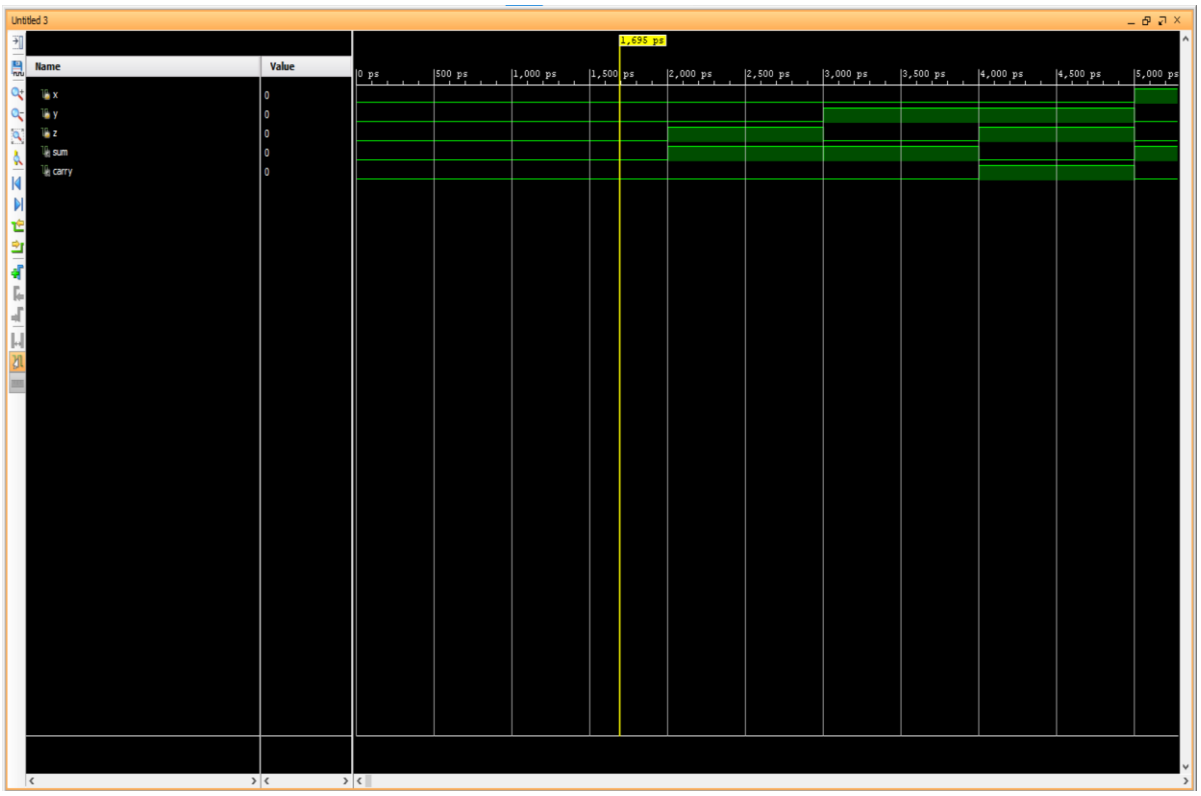
```
1 `timescale 1ns / 1ps
2 //////////////////////////////////////////////////////////////////////////////////
3 // Company:
4 // Engineer:
5 //
6 // Create Date: 19.03.2021 22:51:10
7 // Design Name:
8 // Module Name: fulladder
9 // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 //////////////////////////////////////////////////////////////////////////////////
21
22
23 module fulladder(x,y,z,sum,carry);
24 input x,y,z;
25 output sum,carry;
26 wire sum1 , carry1,carry2;
27
28 halfadder x1(x,y,sum1,carry1);
29 halfadder a1(sum1,z,sum,carry2);
30 xorgate_behavioral b1(carry1,carry2,carry);
31 endmodule
32
```

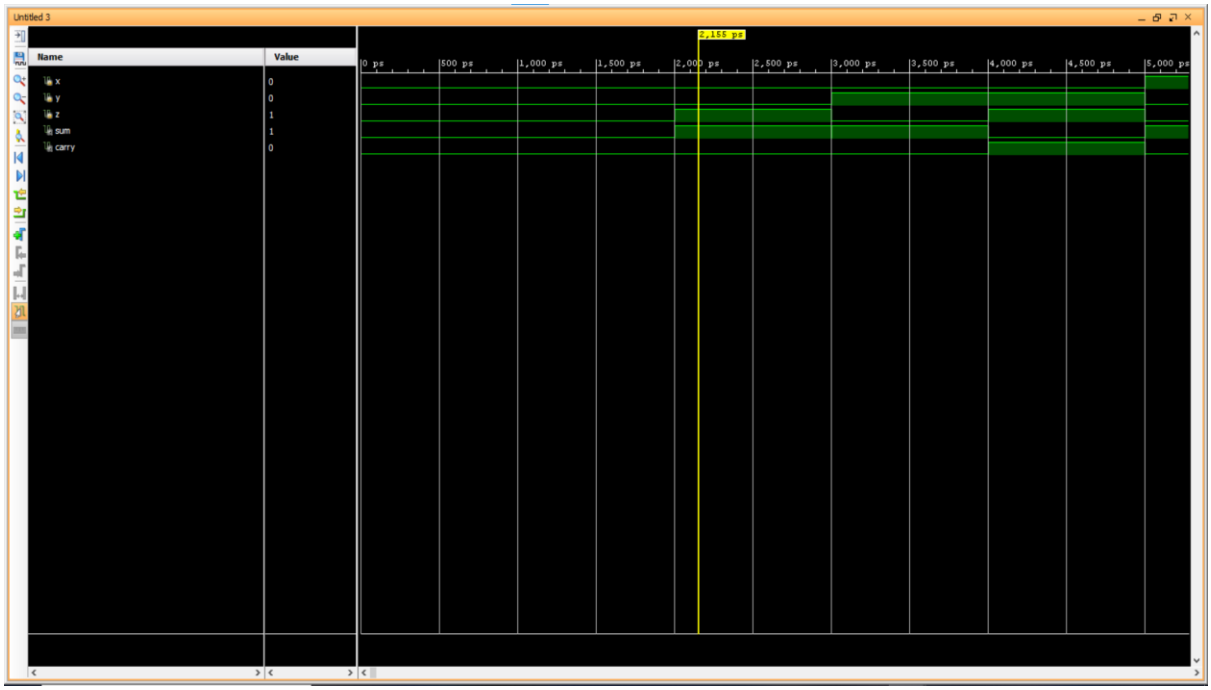
Test bench code:

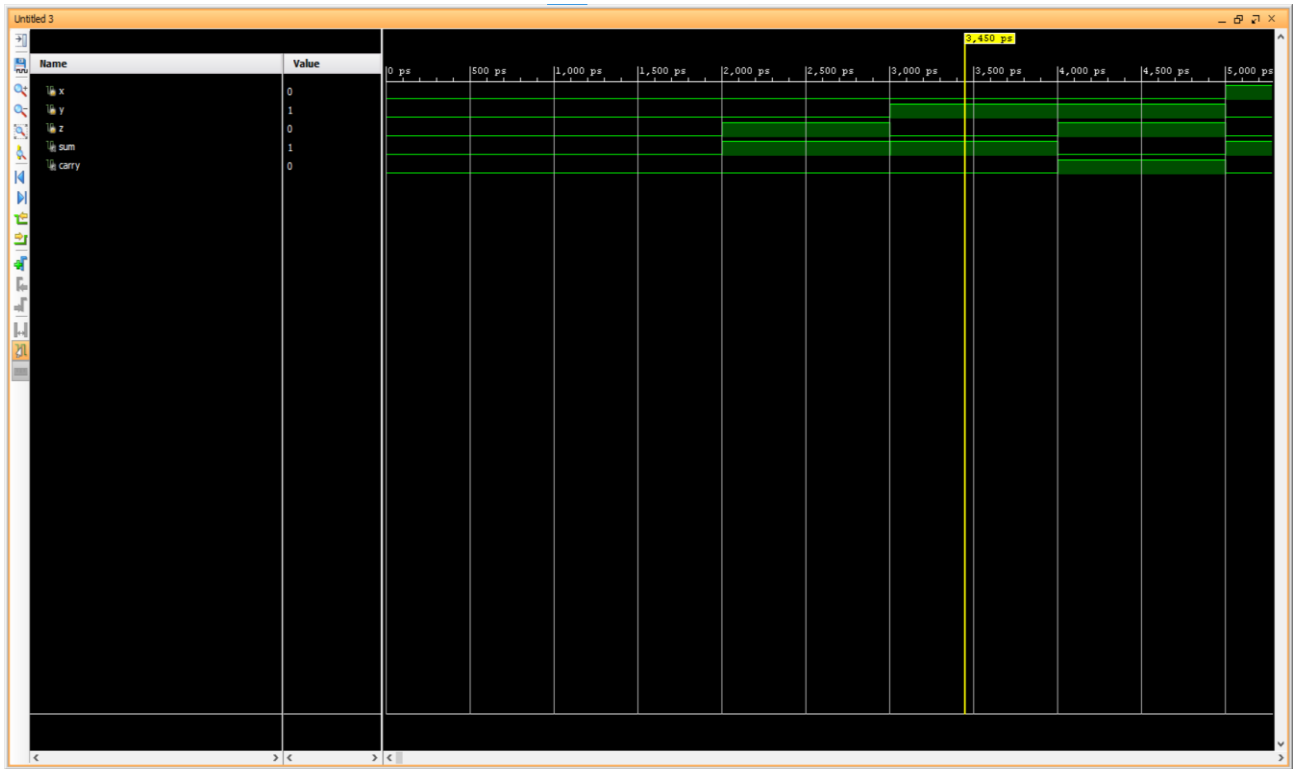
C:/Users/91995/full_adder_14mar2021/full_adder_14mar2021.srcs/sim_1/new/tb_fulladder.v

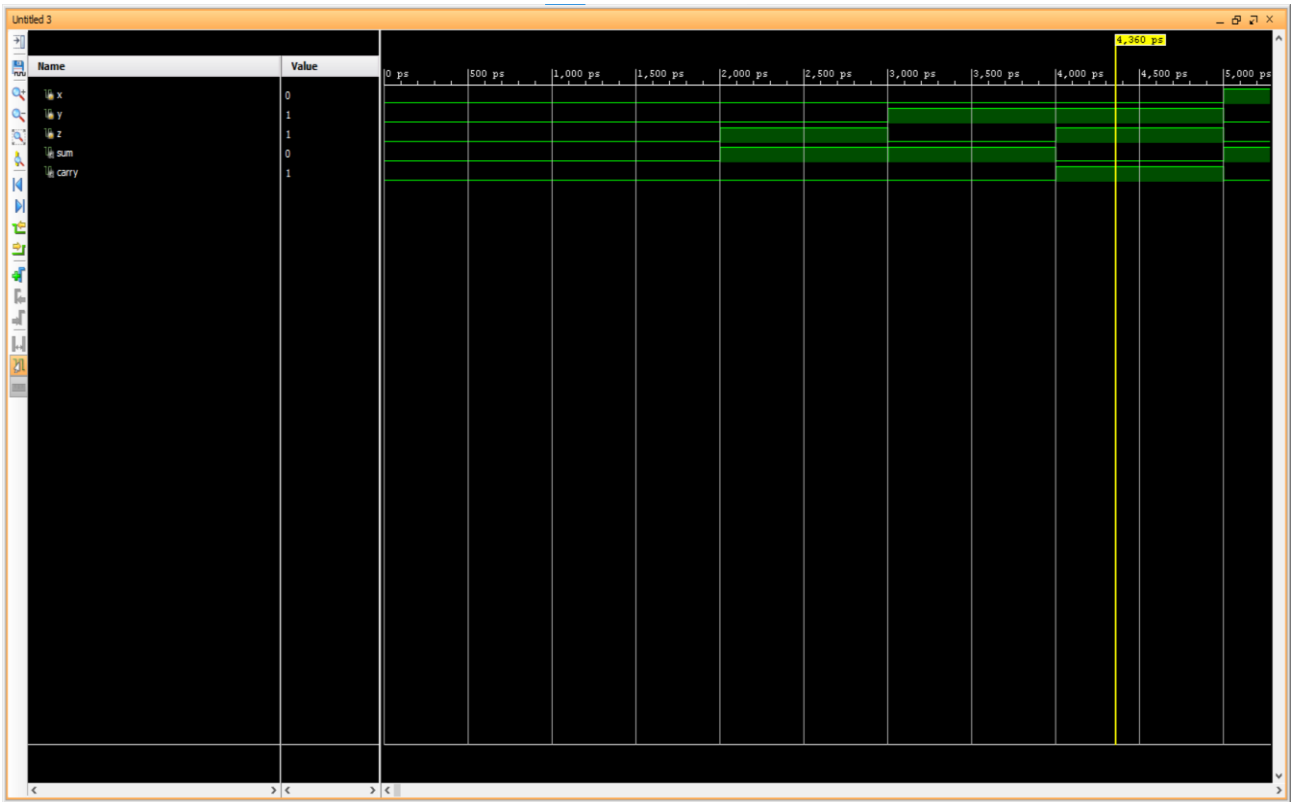
```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date: 19.03.2021 22:53:37
7  // Design Name:
8  // Module Name: tb_fulladder
9  // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 //////////////////////////////////////////////////////////////////////////////////
21
22
23 module tb_fulladder;
24 reg x,y,z;
25 wire sum,carry;
26
27 fulladder b1(x,y,z,sum,carry);
28
29 initial
30 begin
31 x = 0;y=0;z=0;
32 #1 x = 0;y = 0;z = 0;
33 #1 x = 0;y = 0;z = 1;
34 #1 x = 0;y = 1;z = 0;
35 #1 x = 0;y = 1;z = 1;
36 #1 x = 1;y = 0;z = 0;
37 #1 x = 1;y = 0;z = 1;
38 #1 x = 1;y = 1;z = 0;
39 #1 x = 1;y = 1;z = 1;
40
41 end
42 endmodule
```

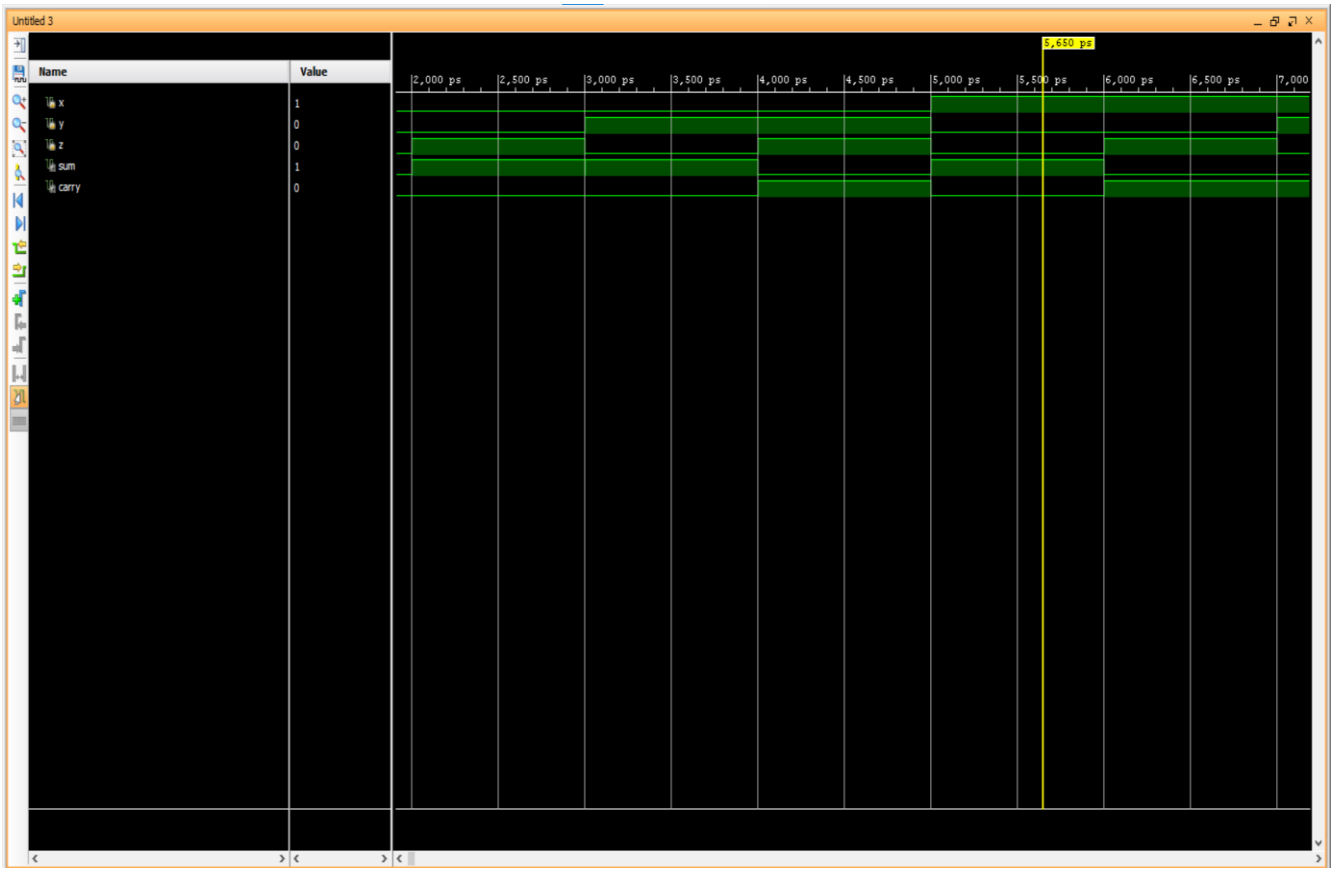
Outputs:

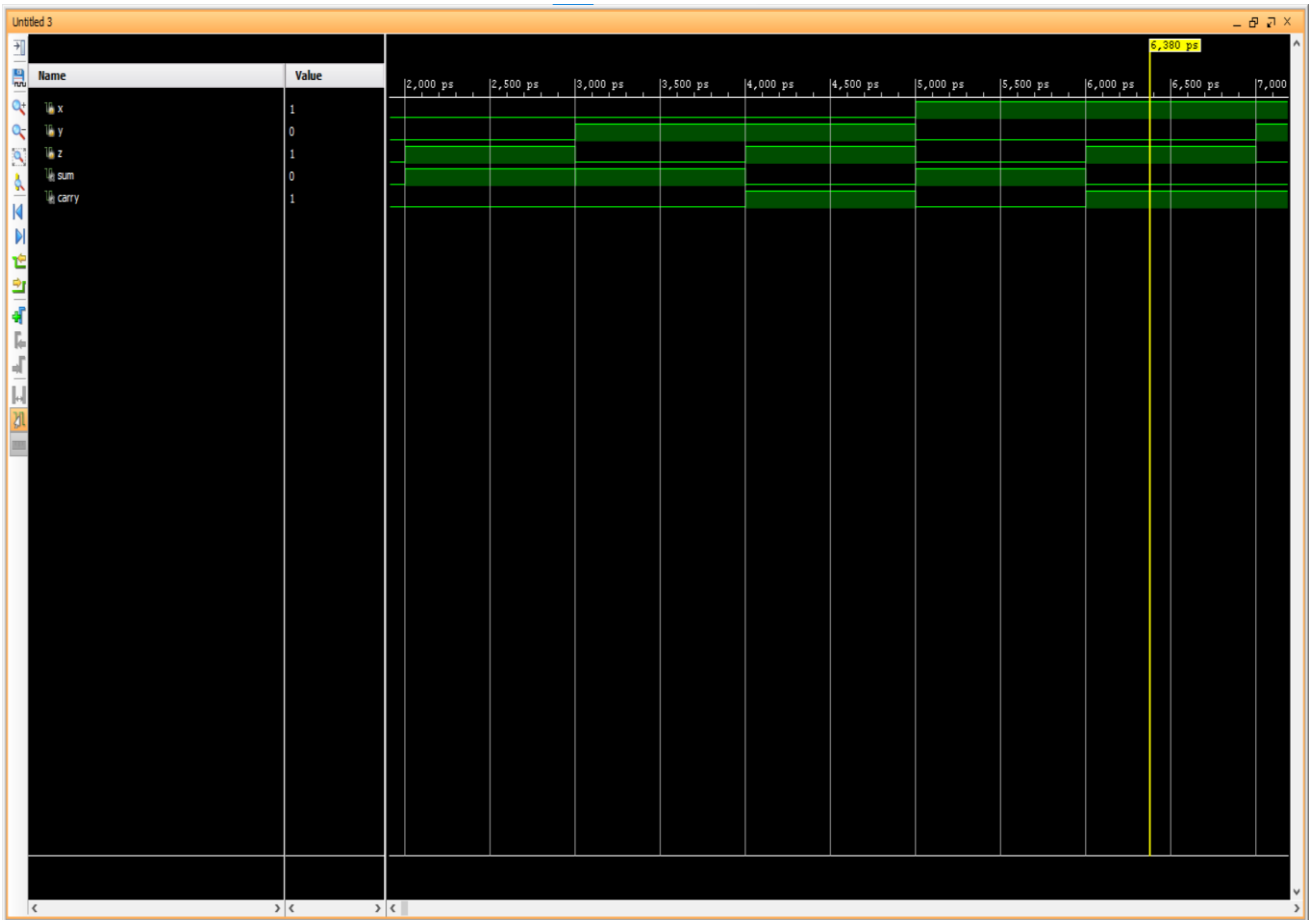

Untitled 3
Name
Value
x
y
z
sum
carry
0
0
0
0
0
1,695 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps

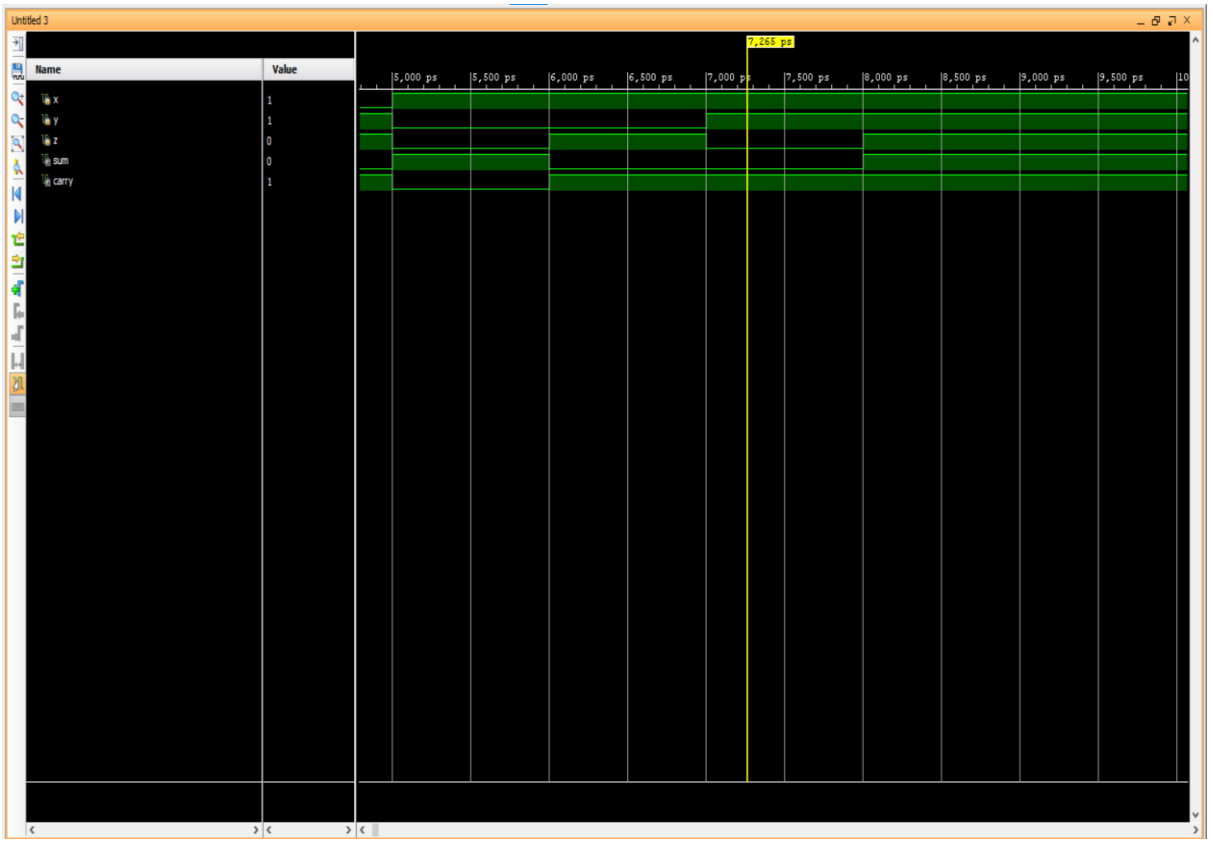

Untitled 3
Name
Value
x
y
z
sum
carry
0
0
1
1
0
2,155 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps

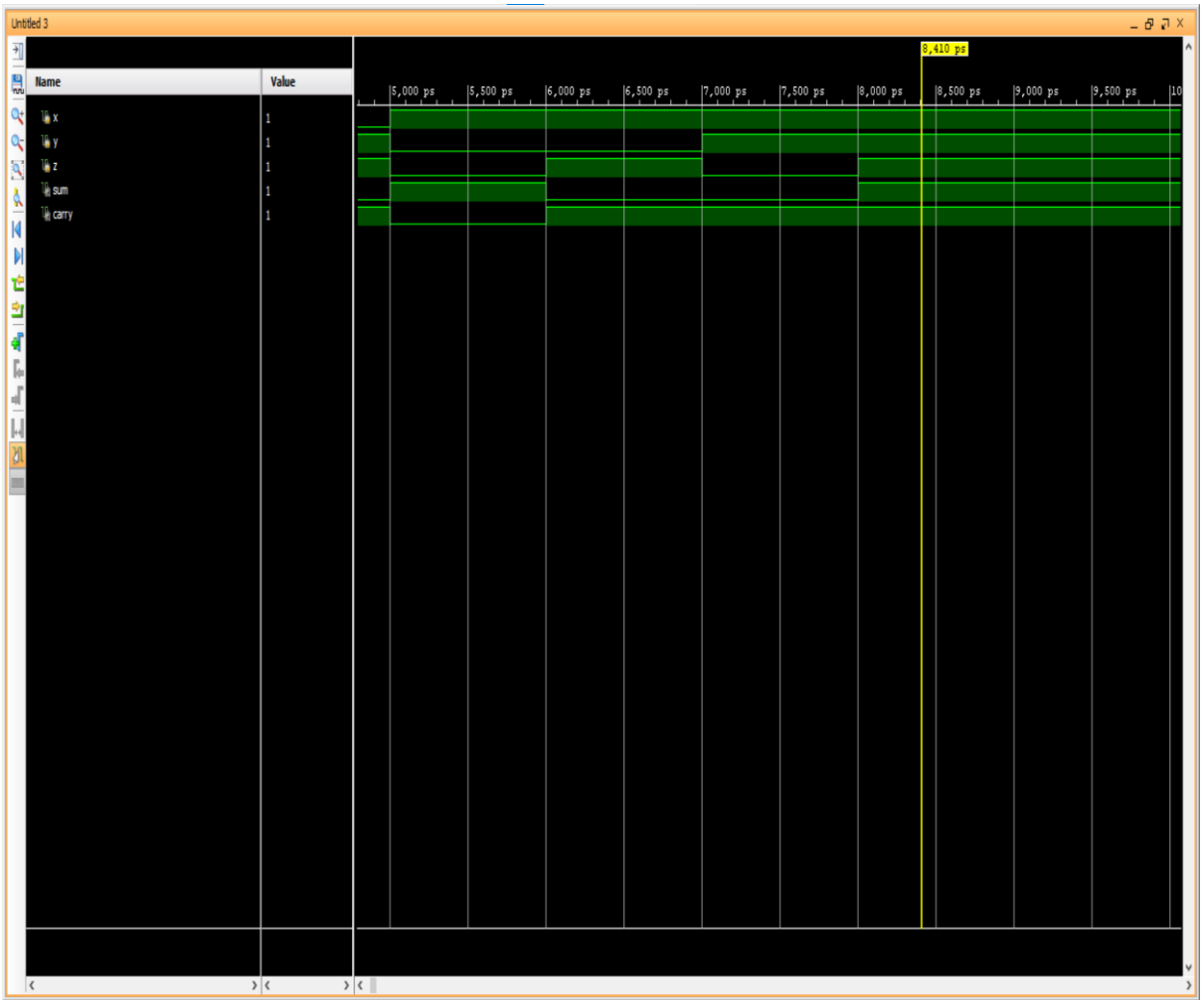

Untitled 3
Name
Value
x
y
z
sum
carry
0
1
0
1
0
3,450 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps


Untitled 3
Name
Value
x
y
z
sum
carry
0
1
1
0
1
4,360 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps


Untitled 3
Name
Value
x
y
z
sum
carry
1
0
0
1
0
5,650 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps
5,500 ps
6,000 ps
6,500 ps
7,000


Untitled 3
Name
Value
x
y
z
sum
carry
1
0
1
0
1
6,380 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps
5,500 ps
6,000 ps
6,500 ps
7,000


Untitled 3
Name
Value
x
y
z
sum
carry
1
1
0
0
1
7,265 ps
5,000 ps
5,500 ps
6,000 ps
6,500 ps
7,000 ps
7,500 ps
8,000 ps
8,500 ps
9,000 ps
9,500 ps



Half subtractor:

C:/Users/91995/halfsubractor/halfsubractor.srcs/sources_1/new/halfsubractor.v

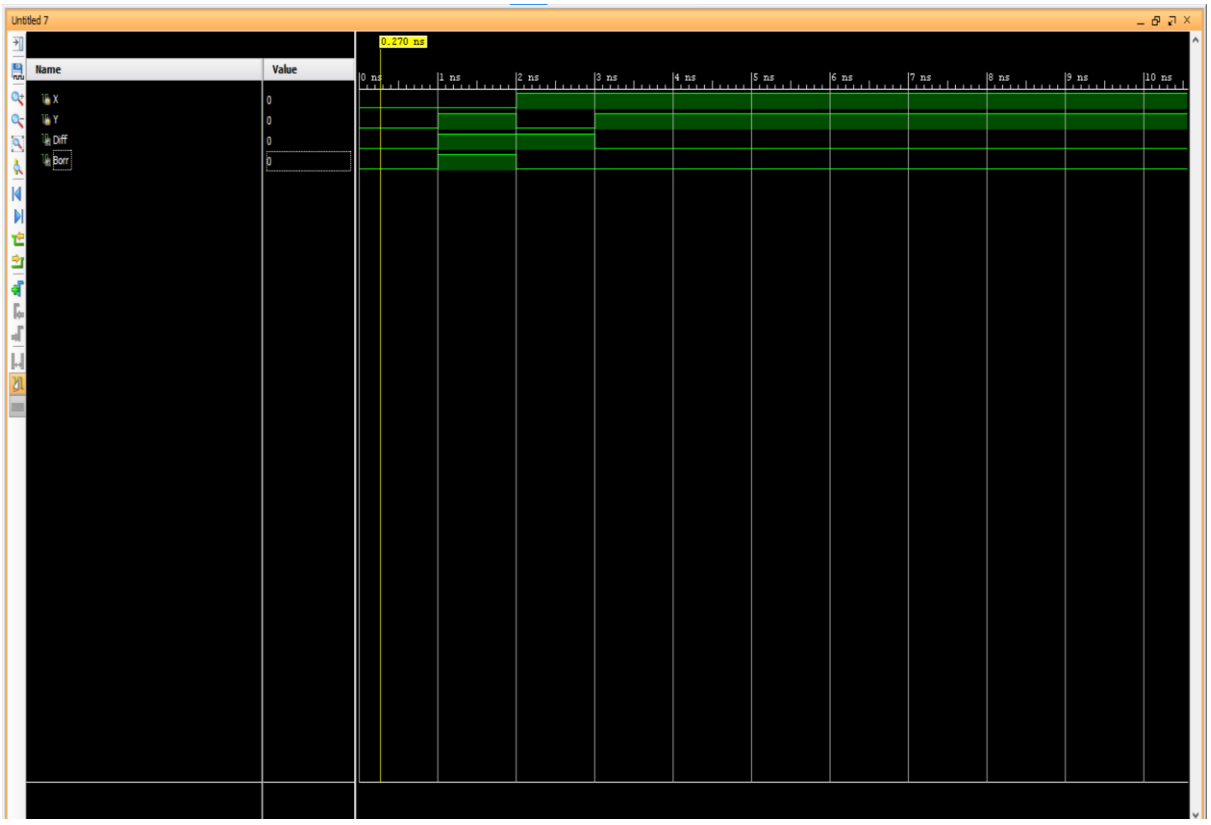
```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date: 19.03.2021 22:38:06
7  // Design Name:
8  // Module Name: halfsubractor
9  // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 //////////////////////////////////////////////////////////////////////////////////
21
22
23
24 module halfSubractor(X,Y,Diff,Borr);
25 input X, Y;
26 output Diff, Borr;
27 assign Diff = X ^ Y;
28 assign Borr = ~X & Y;
29 endmodule
30
31
```

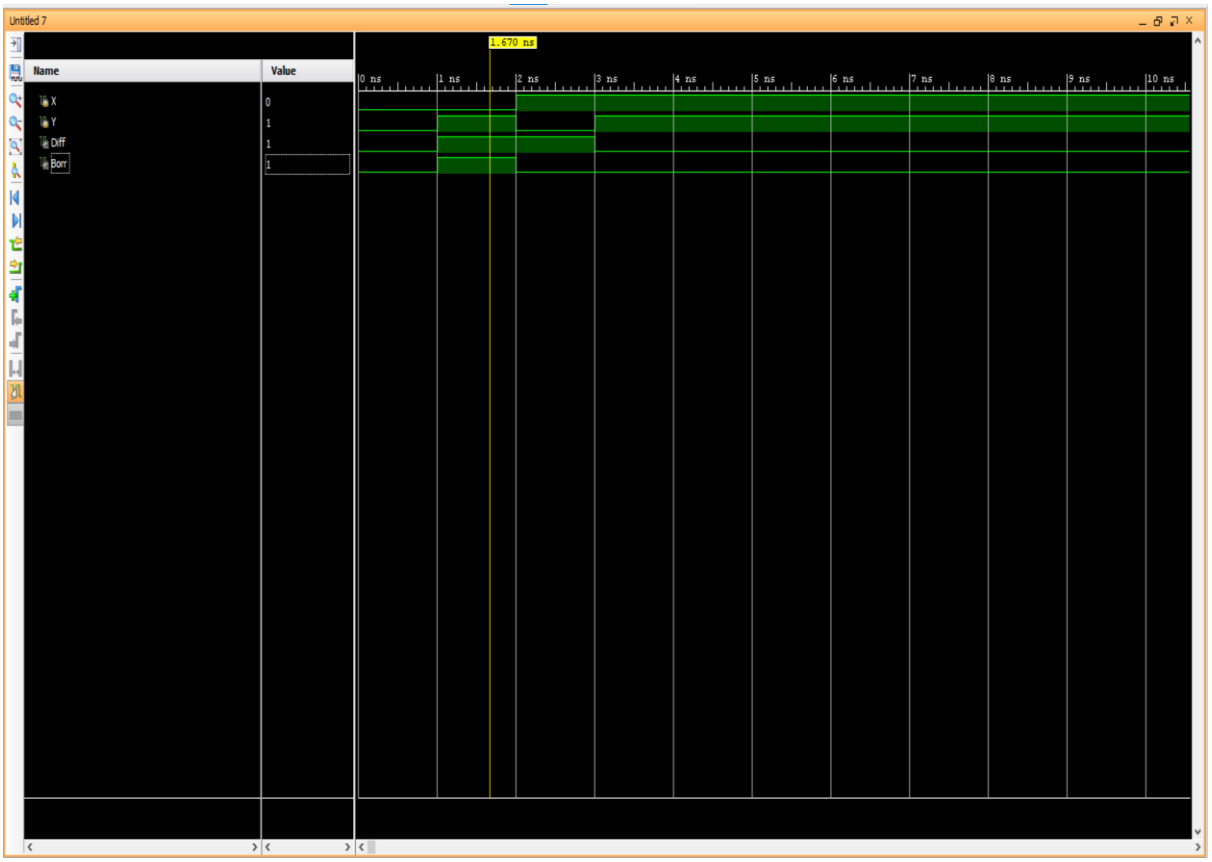
Test bench code:

C:/Users/91995/halfsubractor/halfsubractor.srcs/sim_1/new/tb_halfsubractor.v

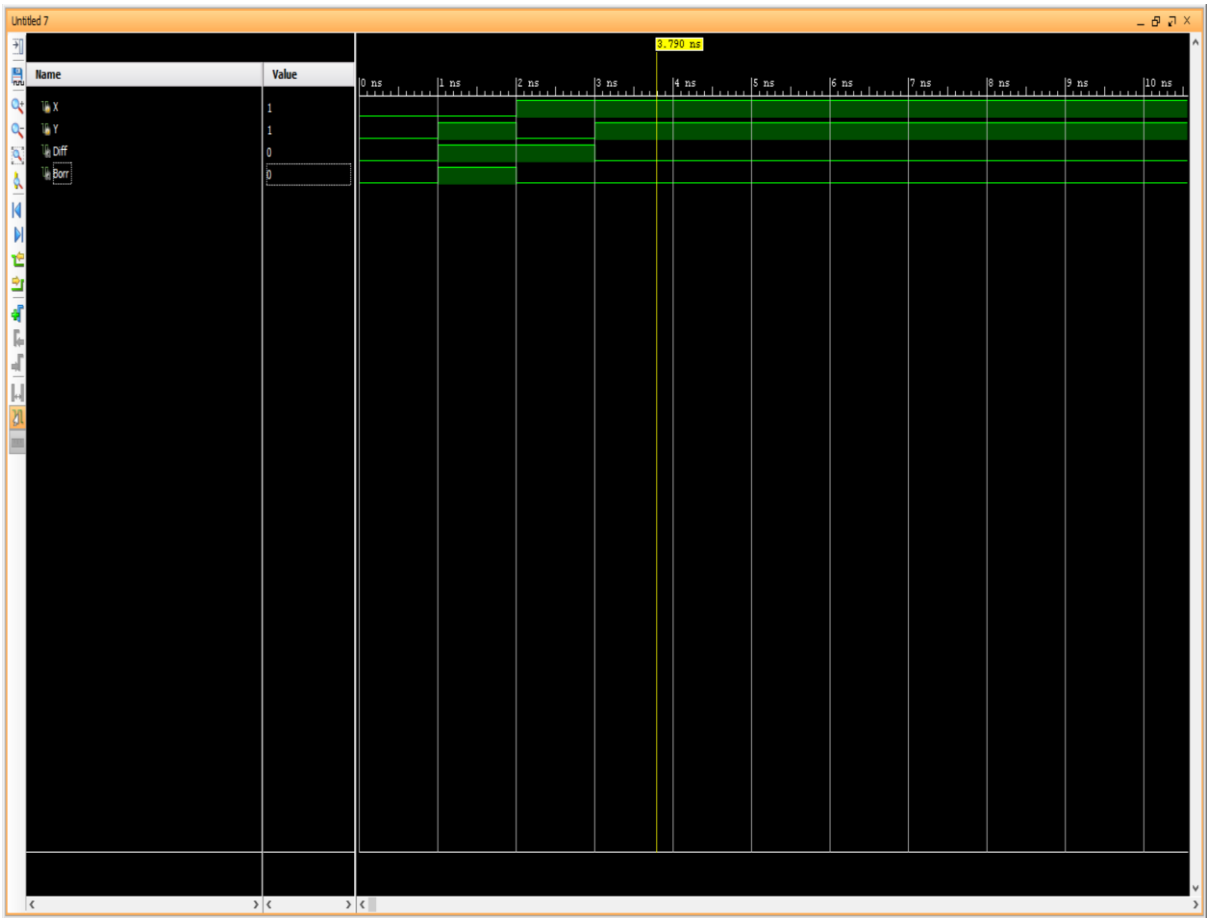
```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date: 19.03.2021 22:38:29
7  // Design Name:
8  // Module Name: tb_halfsubractor
9  // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 //////////////////////////////////////////////////////////////////////////////////
21
22
23 module tb_halfSubractor;
24 reg X, Y;
25 wire Diff, Borr;
26
27 halfSubractor h1(X,Y,Diff,Borr);
28 initial begin
29     X = 0; Y = 0;
30 #1  X = 0; Y = 1;
31 #1  X = 1; Y = 0;
32 #1  X = 1; Y = 1;
33 end
34 endmodule
```

Outputs:


Untitled 7
Name
Value
X
Y
Diff
Borr
0
0
0
0
0.270 ns
0 ns
1 ns
2 ns
3 ns
4 ns
5 ns
6 ns
7 ns
8 ns
9 ns
10 ns


Untitled 7
Name
Value
X
Y
Diff
Borr
0
1
1
1
1.670 ns
0 ns
1 ns
2 ns
3 ns
4 ns
5 ns
6 ns
7 ns
8 ns
9 ns
10 ns


Untitled 7
Name
Value
X
Y
Diff
Borr
1
0
1
0
2.400 ns
0 ns
1 ns
2 ns
3 ns
4 ns
5 ns
6 ns
7 ns
8 ns
9 ns
10 ns


Untitled 7
Name
Value
X
Y
Diff
Borr
1
1
0
0
3.790 ns
0 ns
1 ns
2 ns
3 ns
4 ns
5 ns
6 ns
7 ns
8 ns
9 ns
10 ns

Full subtractor:

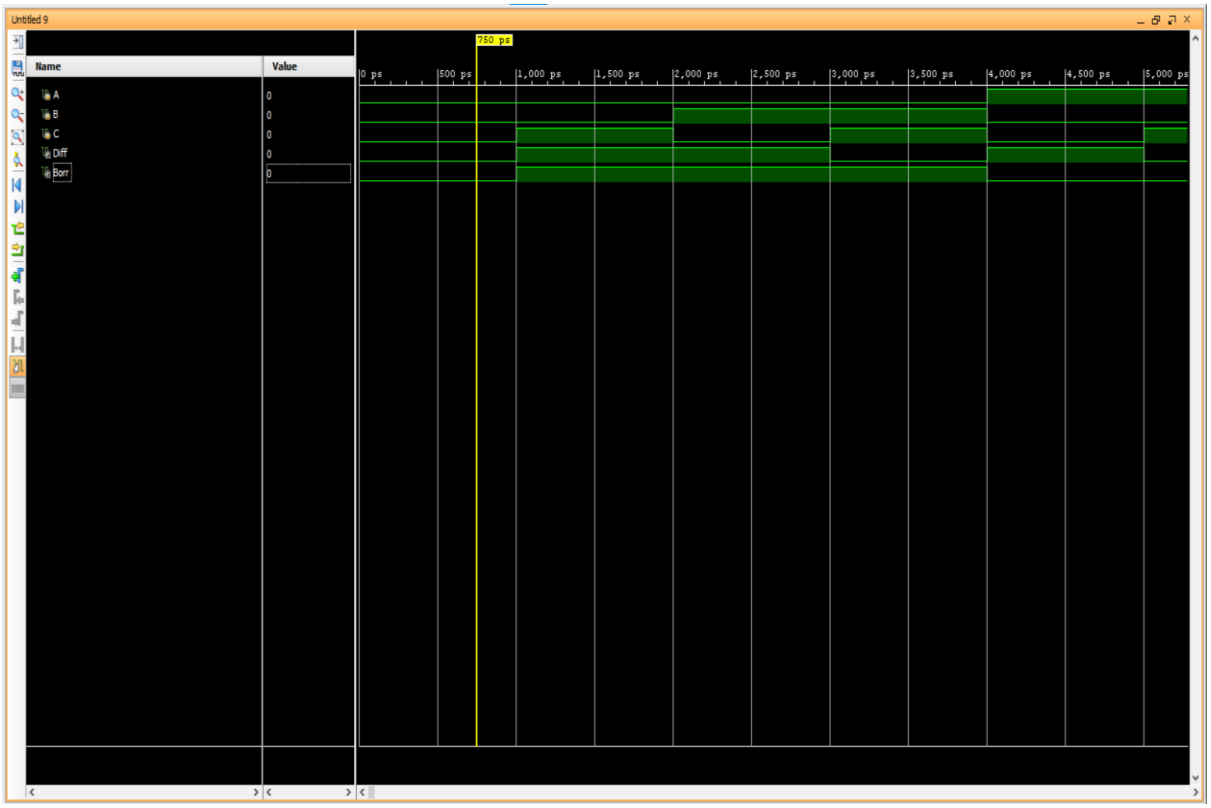
```
module fullsubtractor(A,B,C,Diff,Borr);
input wire A, B, C;
output reg Diff,Borr;
always @(A or B or C)
begin
 if(A==0 && B==0 && C==0)
  begin
   Diff=0;
   Borr=0;
  end
 else if(A==0 && B==0 && C==1)
  begin
     Diff=1;
     Borr=1;
  end
 else if(A==0 && B==1 && C==0)
  begin
   Diff=1;
     Borr=1;
  end
 else if(A==0 && B==1 && C==1)
  begin
   Diff=0;
     Borr=1;
  end
 else if(A==1 && B==0 && C==0)
```

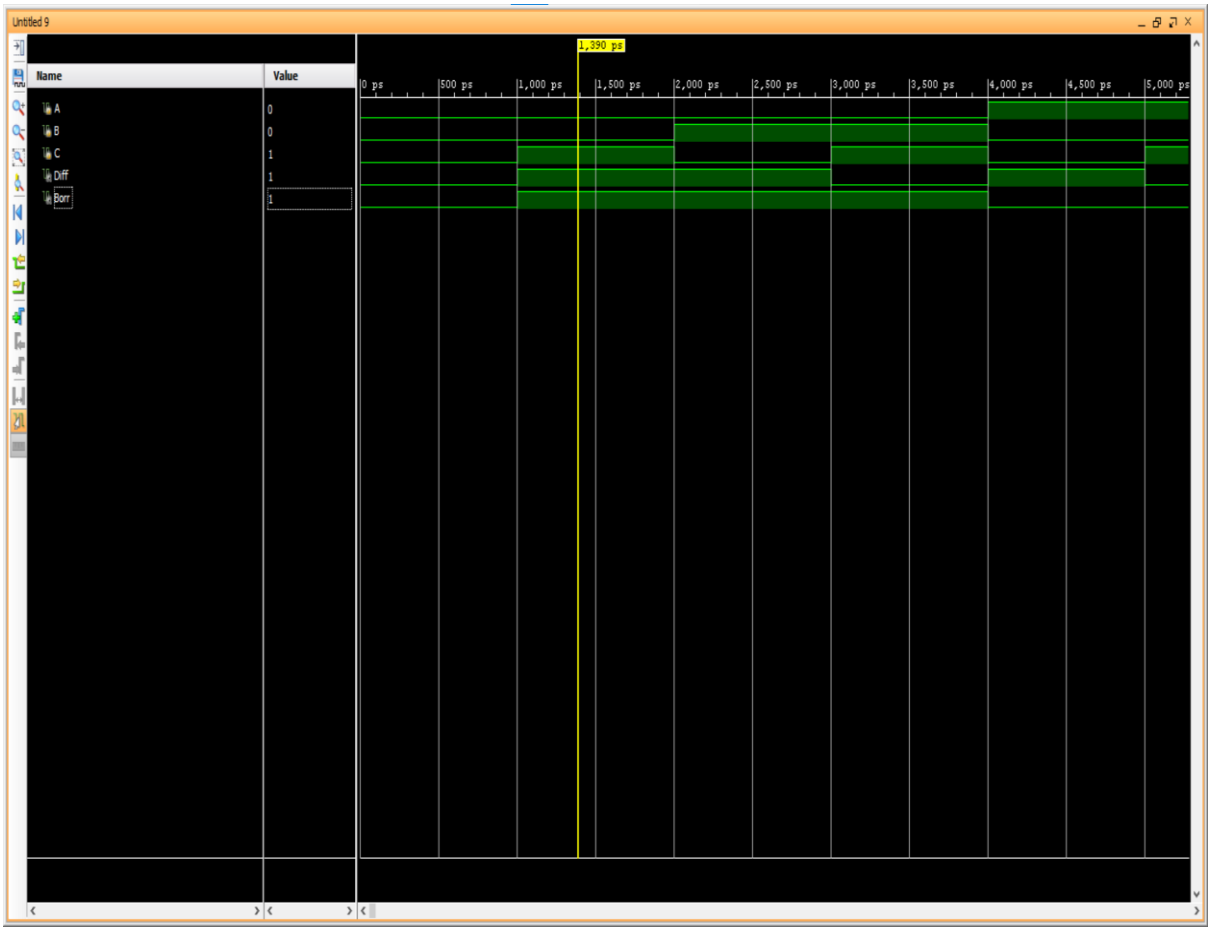
```
  begin
   Diff=1;
     Borr=0;
  end
 else if(A==1 && B==0 && C==1)
  begin
   Diff=0;
     Borr=0;

  end
 else if(A==1 && B==1 && C==0)
  begin
 Diff=0;
     Borr=0;
  end
 else if(A==1 && B==1 && C==1)
  begin
   Diff=1;
     Borr=1;
  end
end
endmodule
```

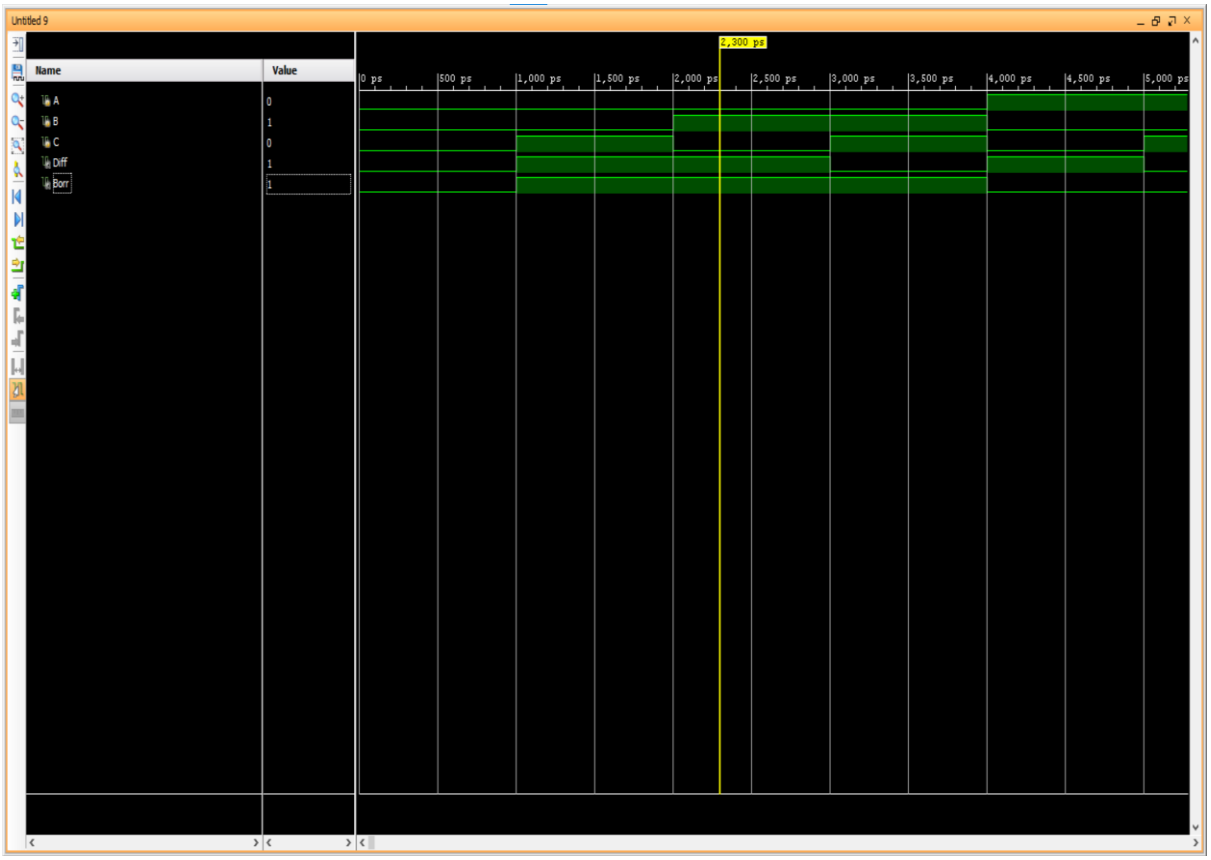
Test bench code:

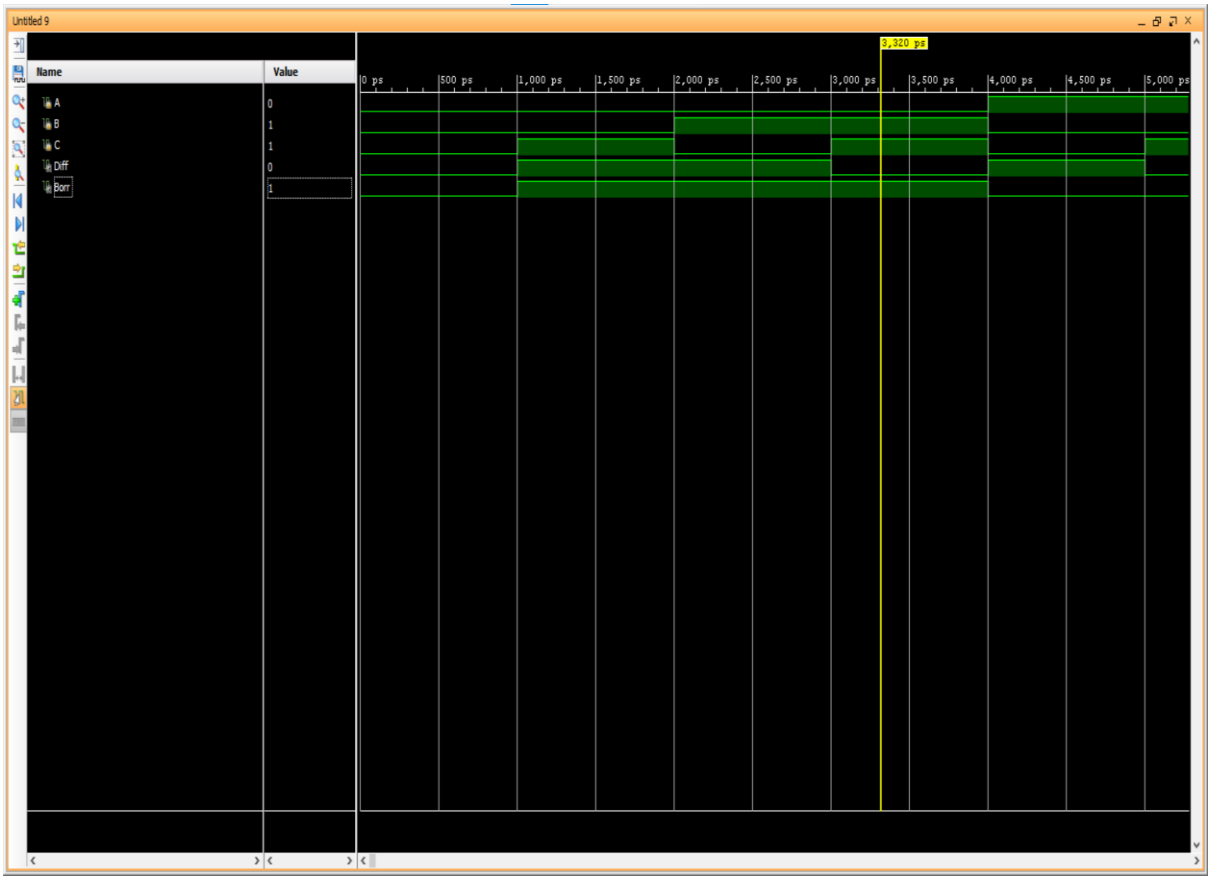
```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 19.03.2021 23:31:37
// Design Name:
// Module Name: tb_fullsubtractor
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////



module tb_fullsubtractor;
reg A,B,C;
wire Diff, Borr;
fullsubtractor f1(A,B,C,Diff,Borr);
initial
 begin
    A = 0; B = 0; C = 0;
#1  A = 0; B = 0; C = 1;
#1  A = 0; B = 1; C = 0;
#1  A = 0; B = 1; C = 1;
#1  A = 1; B = 0; C = 0;
#1  A = 1; B = 0; C = 1;
#1  A = 1; B = 1; C = 0;
#1  A = 1; B = 1; C = 1;
end
endmodule
```

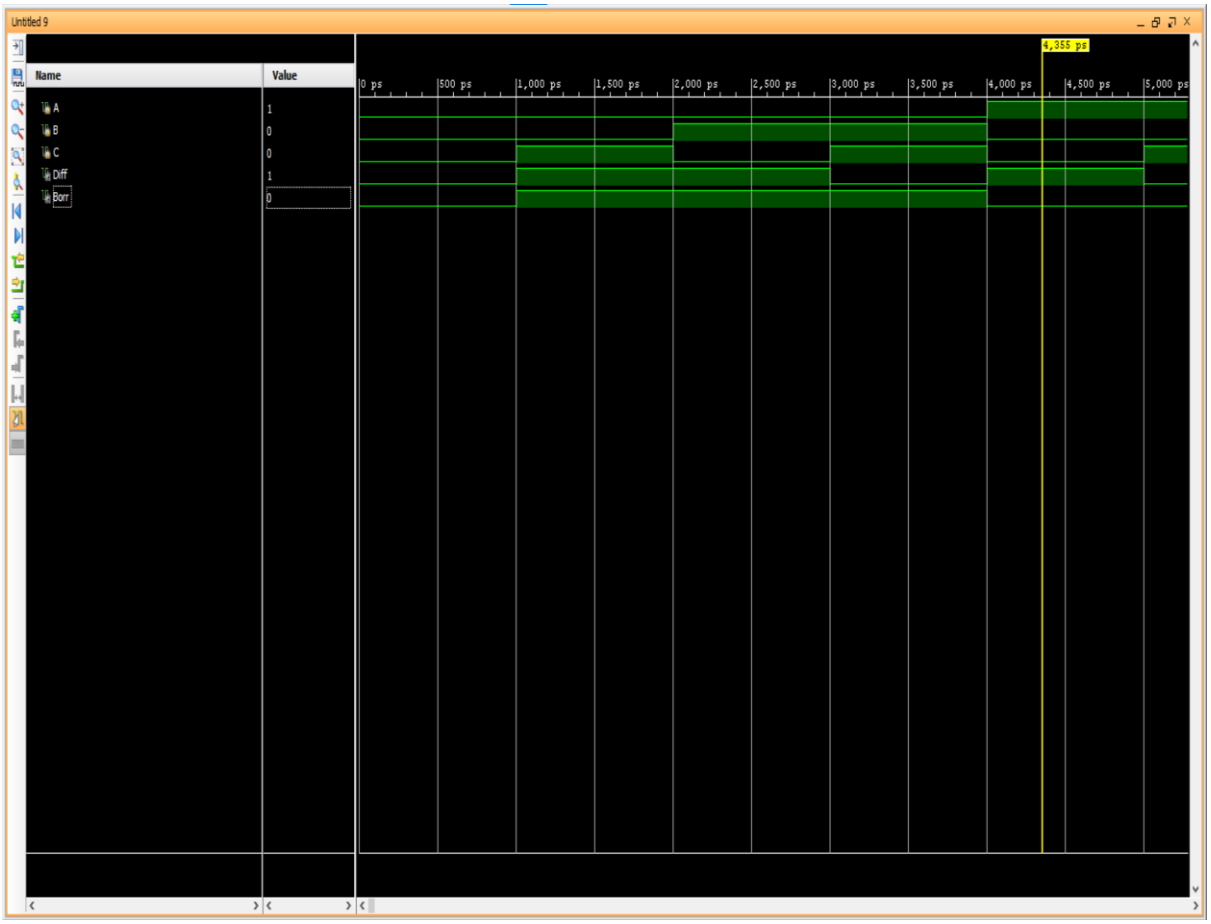
Outputs:

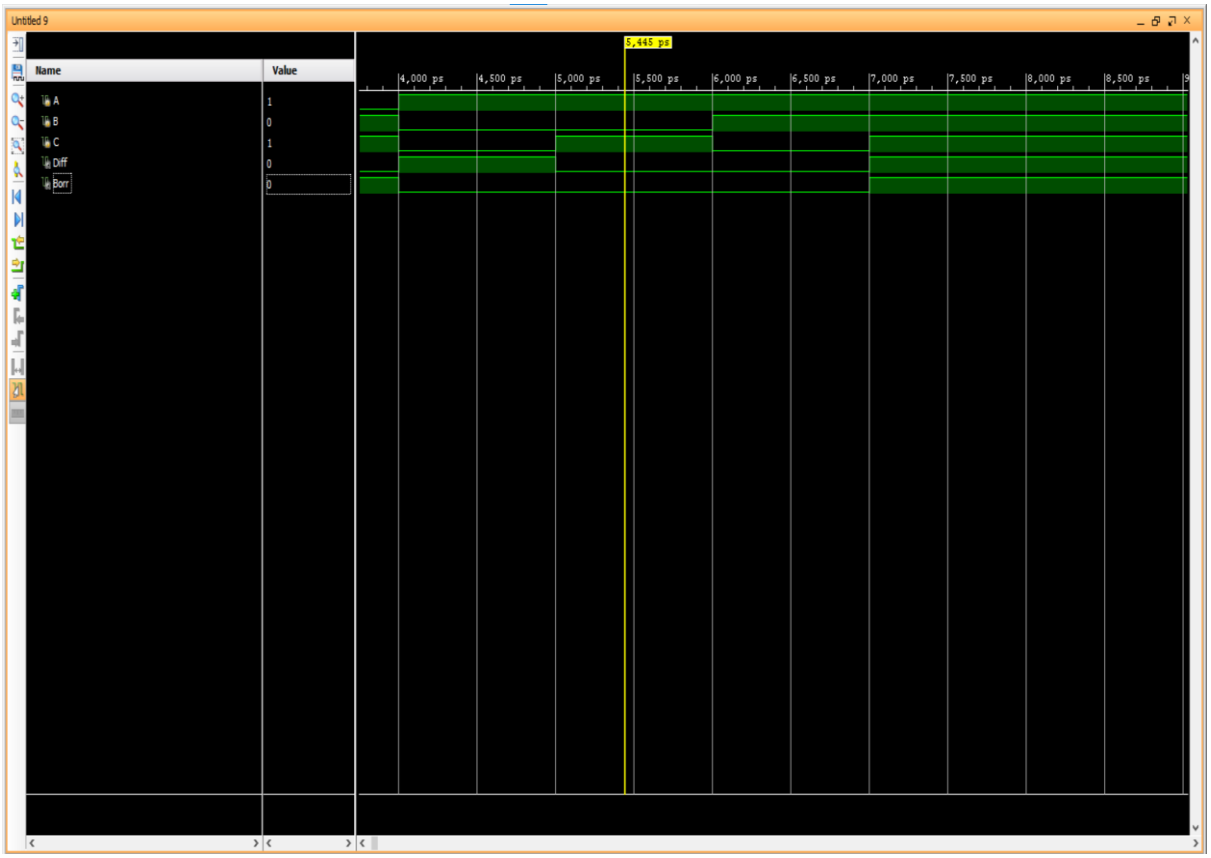

Untitled 9
750 ps
Name
Value
A
B
C
Diff
Borr
0
0
0
0
0
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps

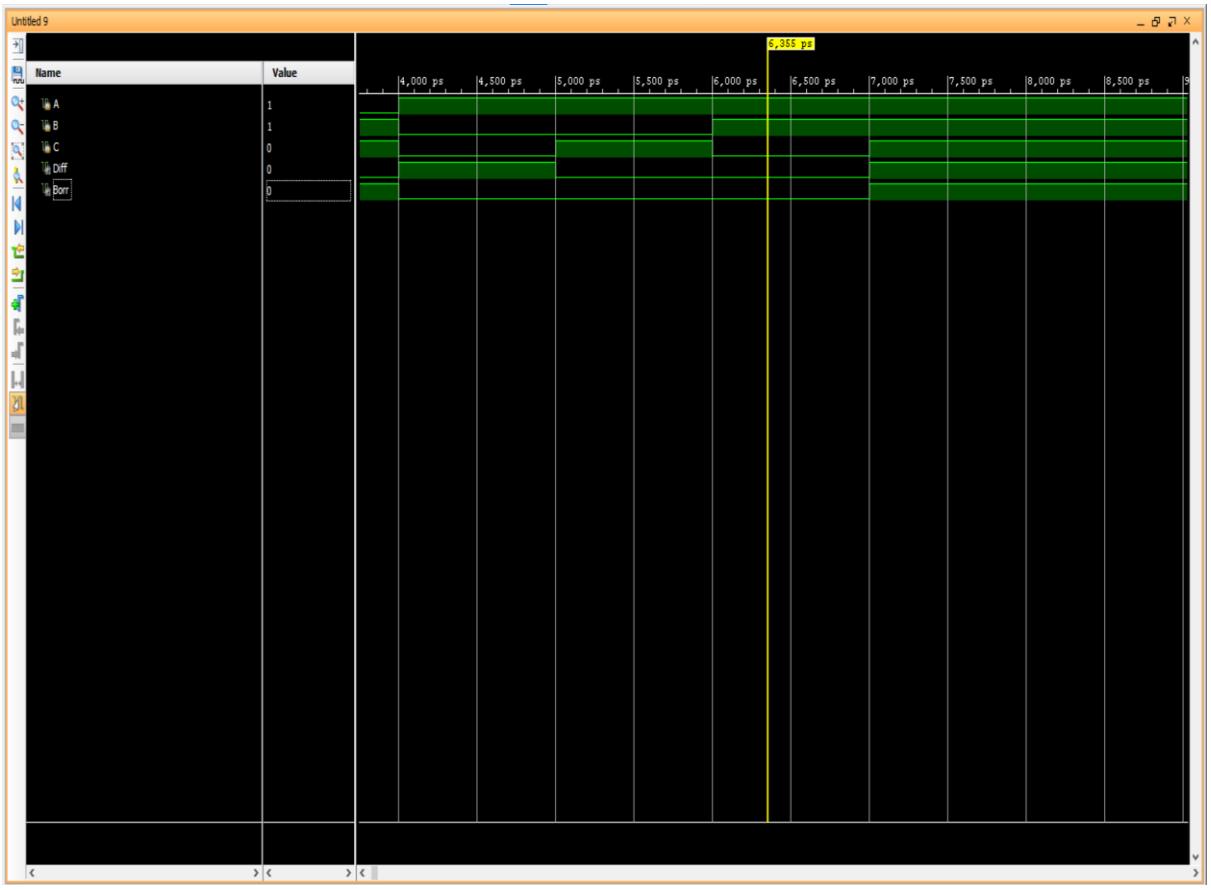

Untitled 9
1,390 ps
Name
Value
A
B
C
Diff
Borr
0
0
1
1
1
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps

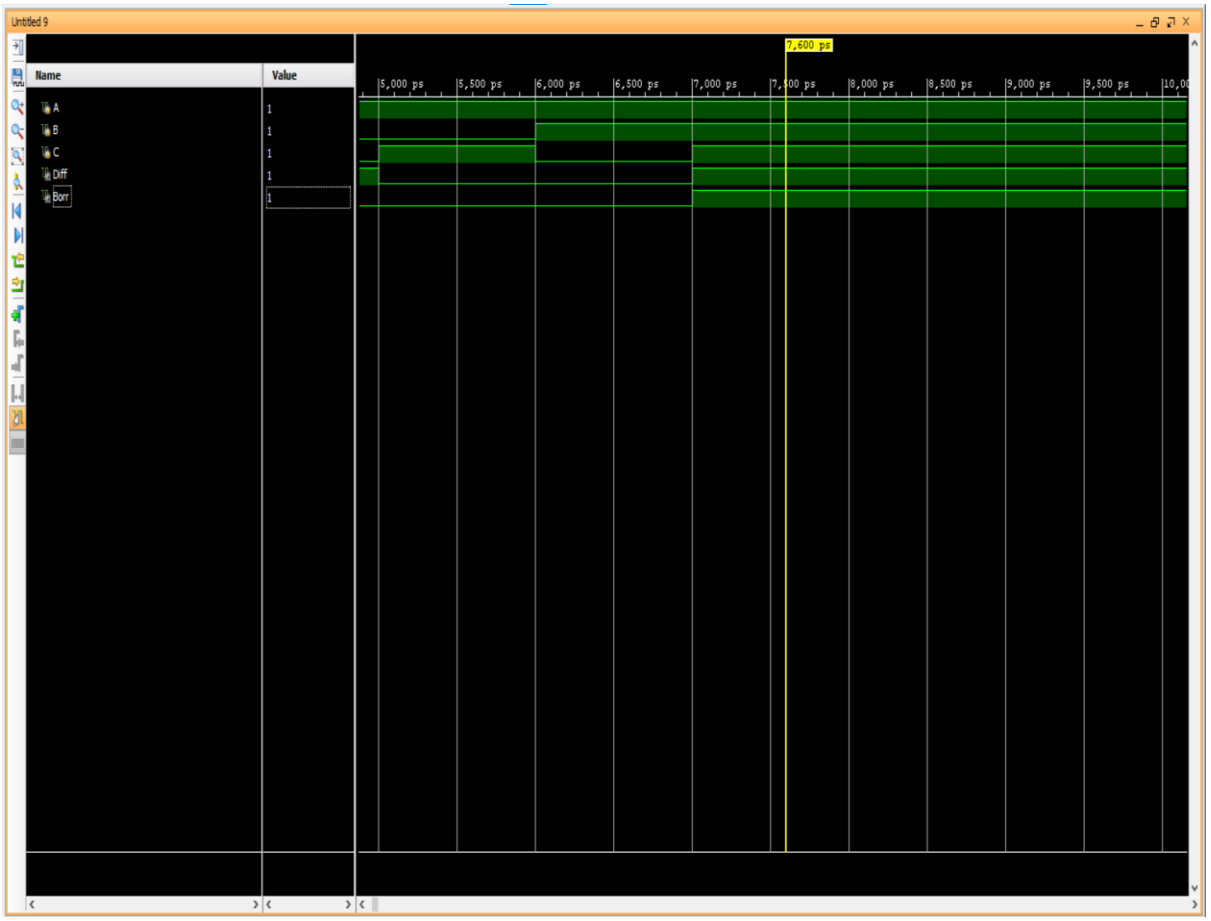

Untitled 9
Name
Value
A
B
C
Diff
Borr
0
1
0
1
1
2,300 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps


Untitled 9
Name
Value
A
B
C
Diff
Borr
0
1
1
0
1
3,320 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps


Untitled 9
Name
Value
A
B
C
Diff
Borr
1
0
0
1
0
4,355 ps
0 ps
500 ps
1,000 ps
1,500 ps
2,000 ps
2,500 ps
3,000 ps
3,500 ps
4,000 ps
4,500 ps
5,000 ps


Untitled 9
Name
Value
A
B
C
Diff
Borr
1
0
1
0
0
5,445 ps
4,000 ps
4,500 ps
5,000 ps
5,500 ps
6,000 ps
6,500 ps
7,000 ps
7,500 ps
8,000 ps
8,500 ps


Untitled 9
Name
Value
A
B
C
Diff
Borr
1
1
0
0
0
6,355 ps
4,000 ps
4,500 ps
5,000 ps
5,500 ps
6,000 ps
6,500 ps
7,000 ps
7,500 ps
8,000 ps
8,500 ps


Untitled 9
Name
Value
A
B
C
Diff
Borr
1
1
1
1
1
7,600 ps
5,000 ps
5,500 ps
6,000 ps
6,500 ps
7,000 ps
8,000 ps
8,500 ps
9,000 ps
9,500 ps

Conclusion:

There are three different types of modelling styles

1. Dataflow
2. Structural
3. Behavioral

With these types of modelling styles I have done the simulation of logic gates (full adder, half subtractor, full subtractor)

Now I'm aware of full adder, half subtractor, full subtractor logic gates

Finally, in this lab I came to know about

• Truth tables

• Logic gate diagrams

• Boolean expressions for logic gates

• Different types of modelling

• Output analysis

Reference links:

https://www.geeksforgeeks.org/

https://theorycircuit.com/