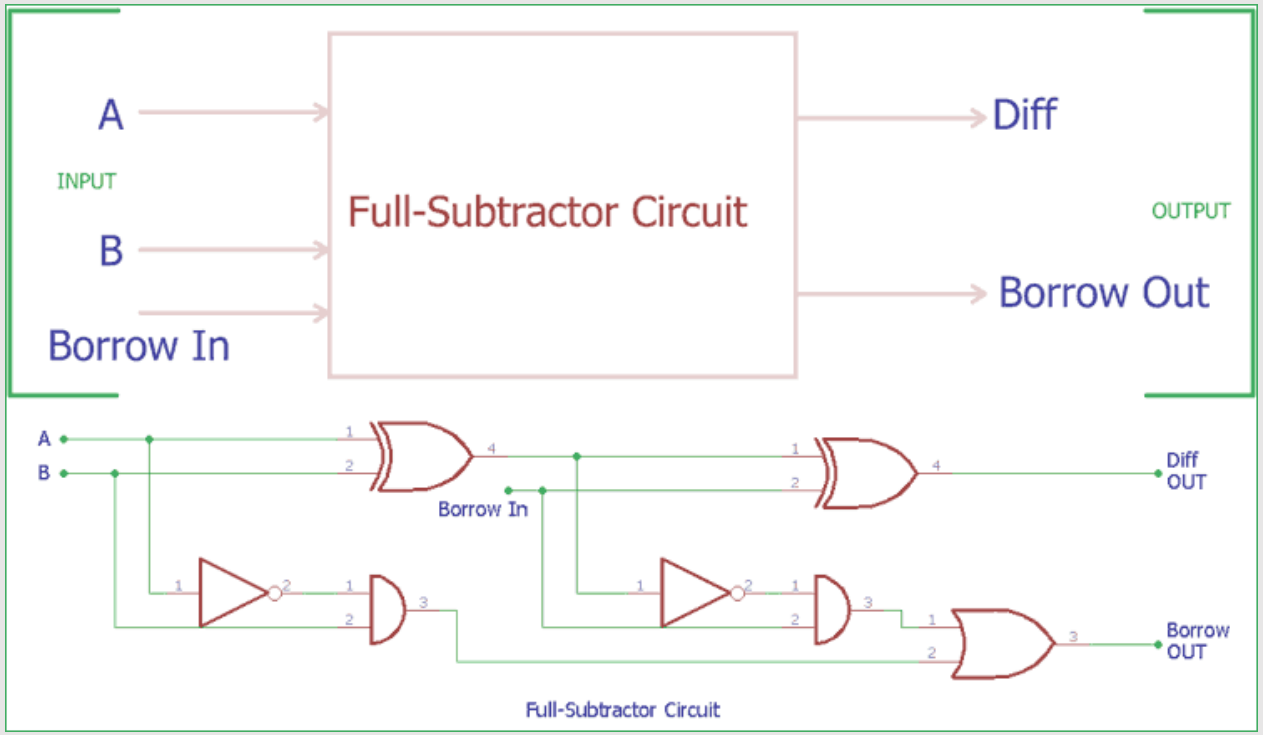
VLSI DESIGN LAB

1. Design of Full Subtractor
2. Design of 4x1 MUX

Report made and submitted by:-
Amlan Sahoo, Roll No. 117EC0322

FULL SUBTRACTOR

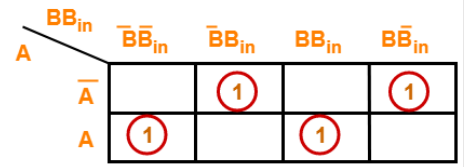


CIRCUIT DIAGRAM

Inputs			Outputs	
A	B	$Borrow_{in}$	Diff	Borrow
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

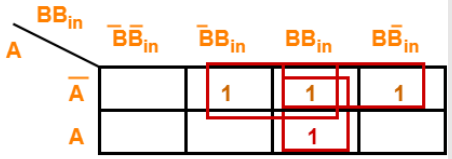
TRUTH TABLE

For D:



$D = A \oplus B \oplus B_{in}$

For B_{in}:



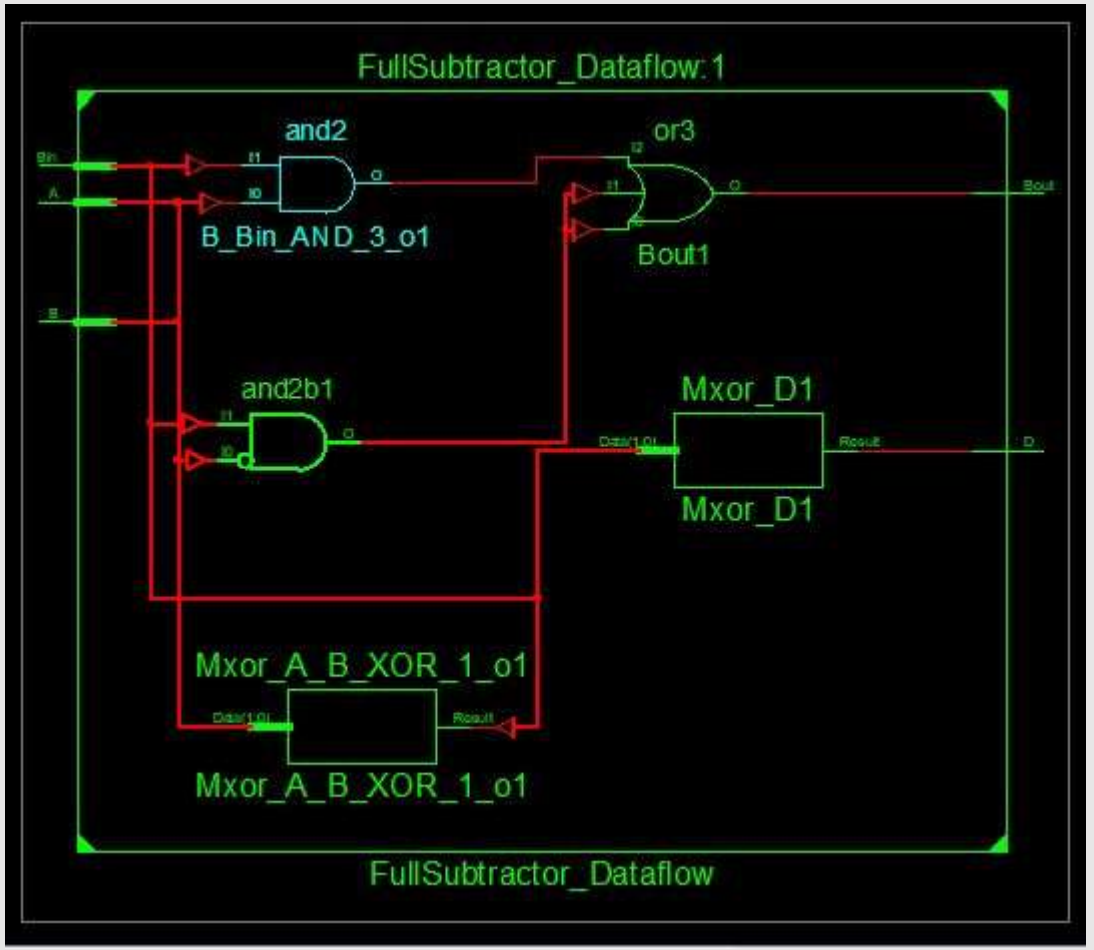
$B_{out} = \bar{A}B + (\bar{A} + B)B_{in}$

LOGIC EQUATIONS

FULL SUBTRACTOR - DATAFLOW

```
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date:    21:21:03 09/08/2020
7  // Design Name:
8  // Module Name:    FullSubtractor_Dataflow
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 FullSubtractor_Dataflow(
22     input A, B, Bin,
23     output D, Bout
24     );
25 assign D = A^B^Bin;
26 assign Bout = (~A&Bin) | (~A&B) | (B&Bin);
27
28 endmodule
29
```

Code for Dataflow Model

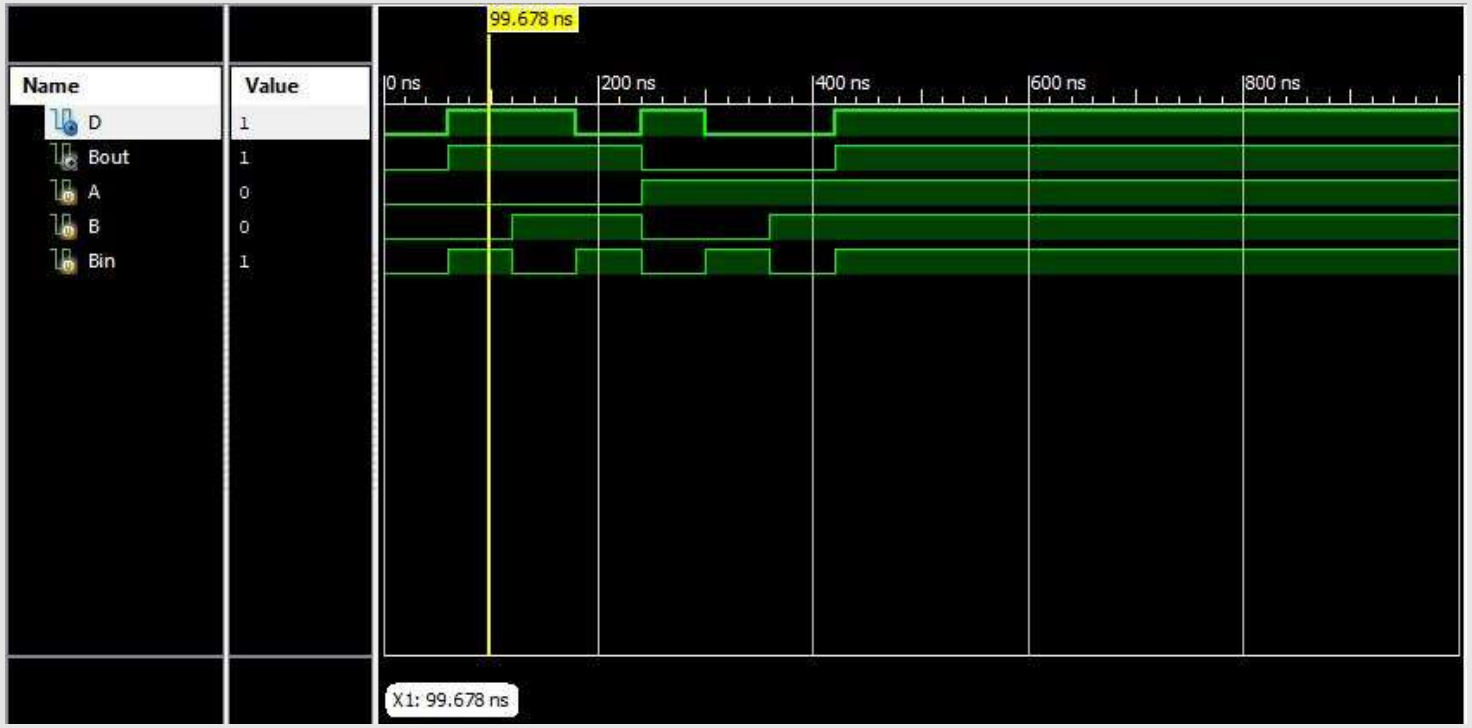


Schematic for Dataflow Model

FULL SUBTRACTOR - DATAFLOW

```
36      // Instantiate the Unit Under Test (UUT)
37      FullSubtractor_Dataflow uut (
38          .A(A),
39          .B(B),
40          .Bin(Bin),
41          .D(D),
42          .Bout(Bout)
43      );
44
45      initial begin
46          // Initialize Inputs
47          A = 0; B = 0; Bin = 0;
48          #60 A = 0; B = 0; Bin = 1;
49          #60 A = 0; B = 1; Bin = 0;
50          #60 A = 0; B = 1; Bin = 1;
51          #60 A = 1; B = 0; Bin = 0;
52          #60 A = 1; B = 0; Bin = 1;
53          #60 A = 1; B = 1; Bin = 0;
54          #60 A = 1; B = 1; Bin = 1;
55          // Wait 100 ns for global reset to finish
56          #100;
57
58          // Add stimulus here
59
60      end
61
62  endmodule
63
```

Code for Testbench



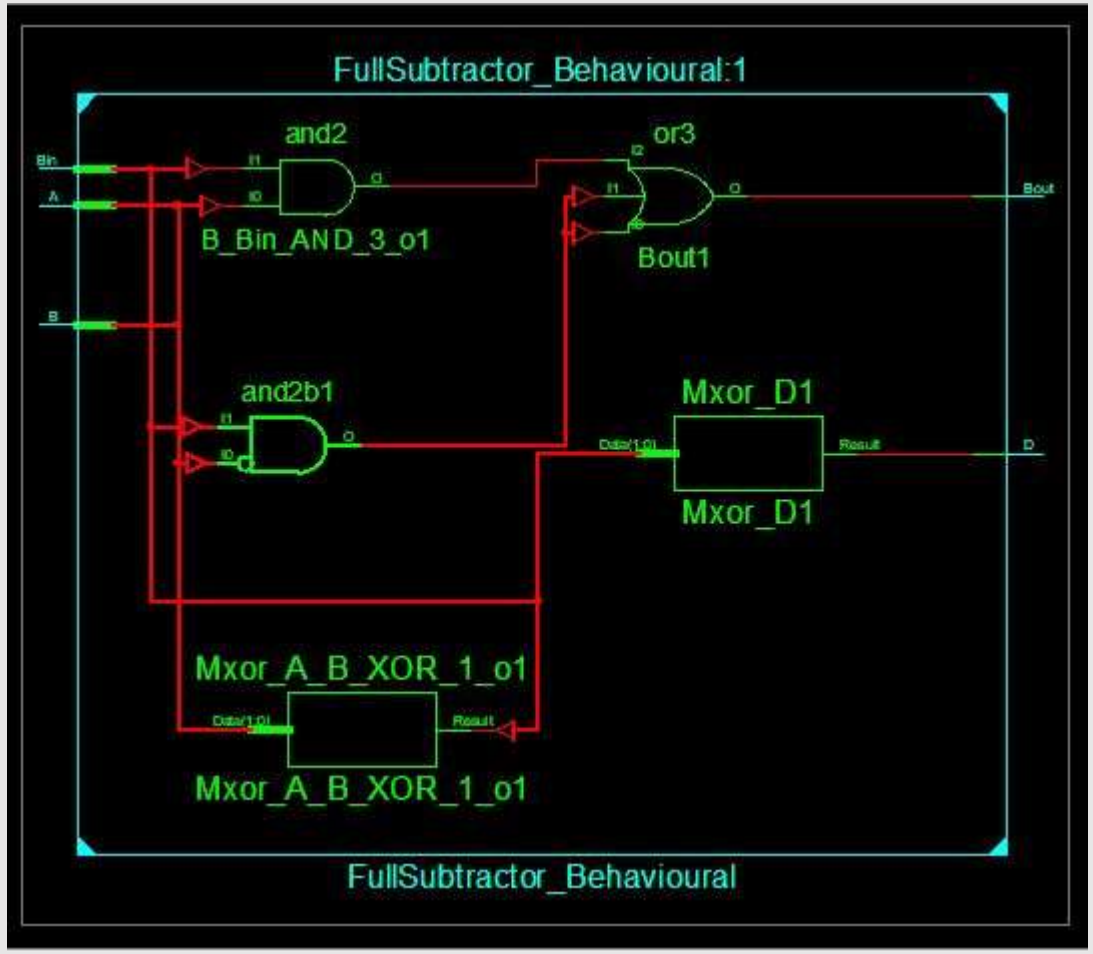
Waveform for Dataflow Model:

The input value of 0,0,1 for A,B,Bin gives output value of 1,1 for D and Bout resp. which confirms with the TruthTable.

FULL SUBTRACTOR - BEHAVIOURAL

```
5  //
6  // Create Date:    22:05:58 09/08/2020
7  // Design Name:
8  // Module Name:    FullSubtractor_Behavioural
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 FullSubtractor_Behavioural(
22     input A, B, Bin,
23     output reg D, Bout
24     );
25 always @(A, B, Bin)
26    begin
27    D = A^B^Bin;
28    Bout = (~A&Bin) | (~A&B) | (B&Bin);
29    end
30
31 endmodule
32
```

Code for Behavioural Model



Schematic for Behavioural Model

FULL SUBTRACTOR - BEHAVIOURAL

```
36      // Instantiate the Unit Under Test (UUT)
37      FullSubtractor_Behavioural uut (
38          .A(A),
39          .B(B),
40          .Bin(Bin),
41          .D(D),
42          .Bout(Bout)
43      );
44
45      initial begin
46          // Initialize Inputs
47          A = 0; B = 0; Bin = 0;
48          #60 A = 0; B = 0; Bin = 1;
49          #60 A = 0; B = 1; Bin = 0;
50          #60 A = 0; B = 1; Bin = 1;
51          #60 A = 1; B = 0; Bin = 0;
52          #60 A = 1; B = 0; Bin = 1;
53          #60 A = 1; B = 1; Bin = 0;
54          #60 A = 1; B = 1; Bin = 1;
55
56          // Wait 100 ns for global reset to finish
57          #100;
58
59          // Add stimulus here
60
61      end
62
63  endmodule
```

Code for Testbench

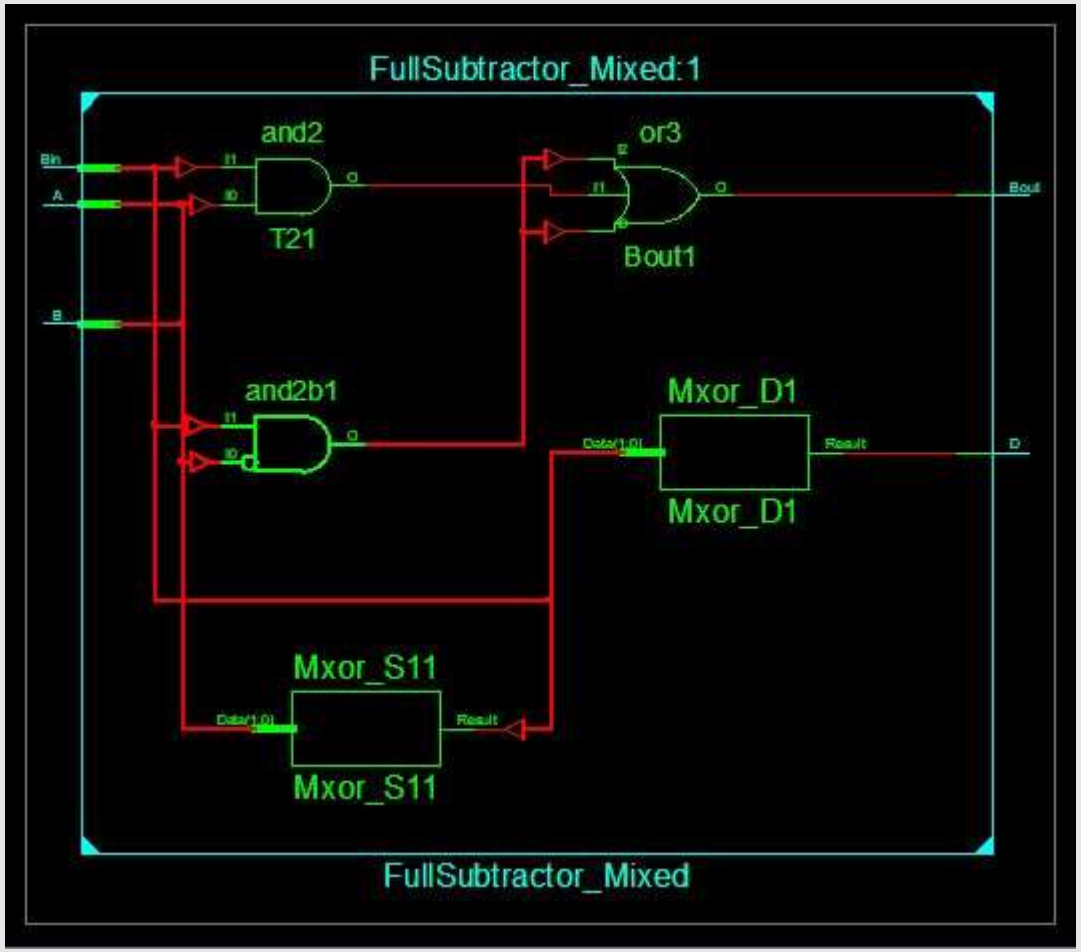
Waveform for Behavioural Model:

The input value of 0,1,0 for A,B,Bin gives output value of 1,1 for D and Bout resp. which confirms with the TruthTable.

FULL SUBTRACTOR – MIXED MODELLING

```
17  // Revision 0.01 - File Created
18  // Additional Comments:
19  //
20  //////////////////////////////////////////////////////////////////////////////////
21  module FullSubtractor_Mixed(
22      input A, B, Bin,
23      output D, reg Bout
24      );
25     wire S1;
26     reg T1, T2, T3;
27
28     // Structural
29     xor X1(S1, A, B);
30
31     // Dataflow
32     assign D = S1^Bin;
33
34     // Behavioural
35     always @(A, B, Bin)
36        begin
37        T3 = ~A&B;
38        T1 = ~A&Bin;
39        T2 = B&Bin;
40        Bout = T1 | T2 | T3;
41        end
42
43
44  endmodule
```

Code for Mixed Model



Schematic for Mixed Model

FULL SUBTRACTOR – MIXED MODELLING

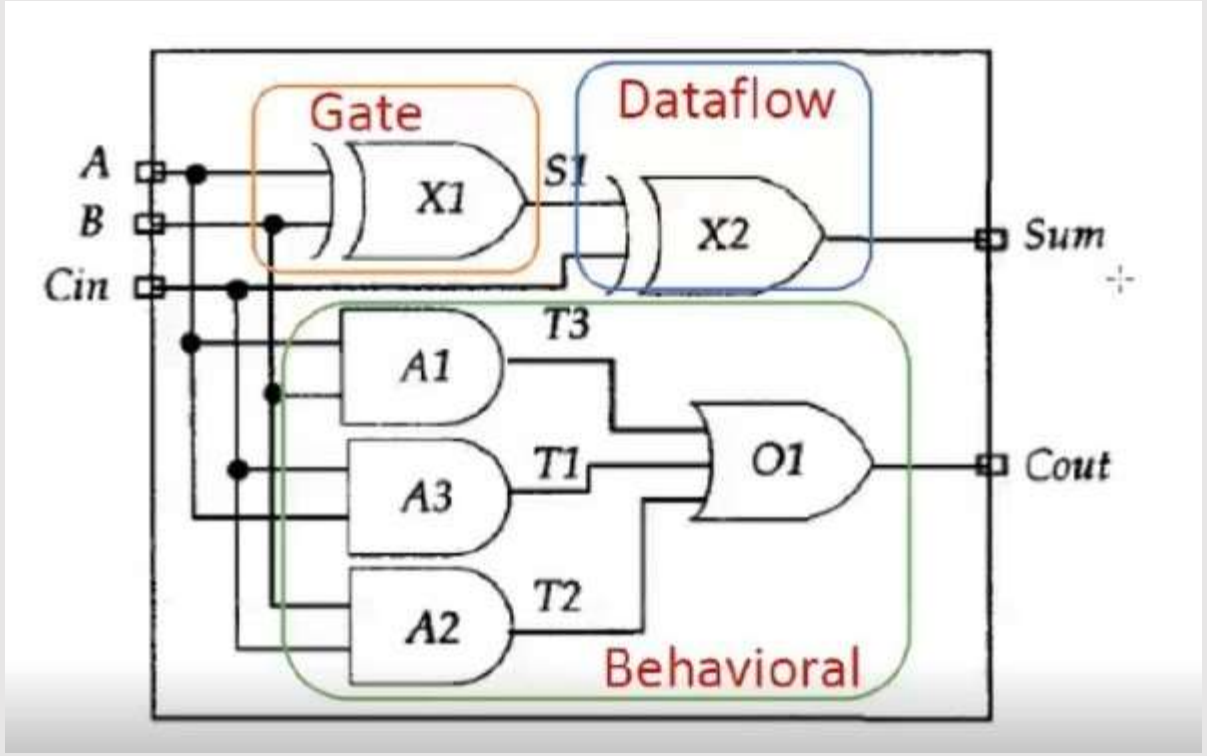
```
36      // Instantiate the Unit Under Test (UUT)
37      FullSubtractor_Mixed uut (
38          .A(A),
39          .B(B),
40          .Bin(Bin),
41          .D(D),
42          .Bout(Bout)
43      );
44
45      initial begin
46          // Initialize Inputs
47          A = 0; B = 0; Bin = 0;
48          #60 A = 0; B = 0; Bin = 1;
49          #60 A = 0; B = 1; Bin = 0;
50          #60 A = 0; B = 1; Bin = 1;
51          #60 A = 1; B = 0; Bin = 0;
52          #60 A = 1; B = 0; Bin = 1;
53          #60 A = 1; B = 1; Bin = 0;
54          #60 A = 1; B = 1; Bin = 1;
55
56          // Wait 100 ns for global reset to finish
57          #100;
58
59          // Add stimulus here
60
61      end
62
63  endmodule
```

Code for Testbench

Waveform for Mixed Model:

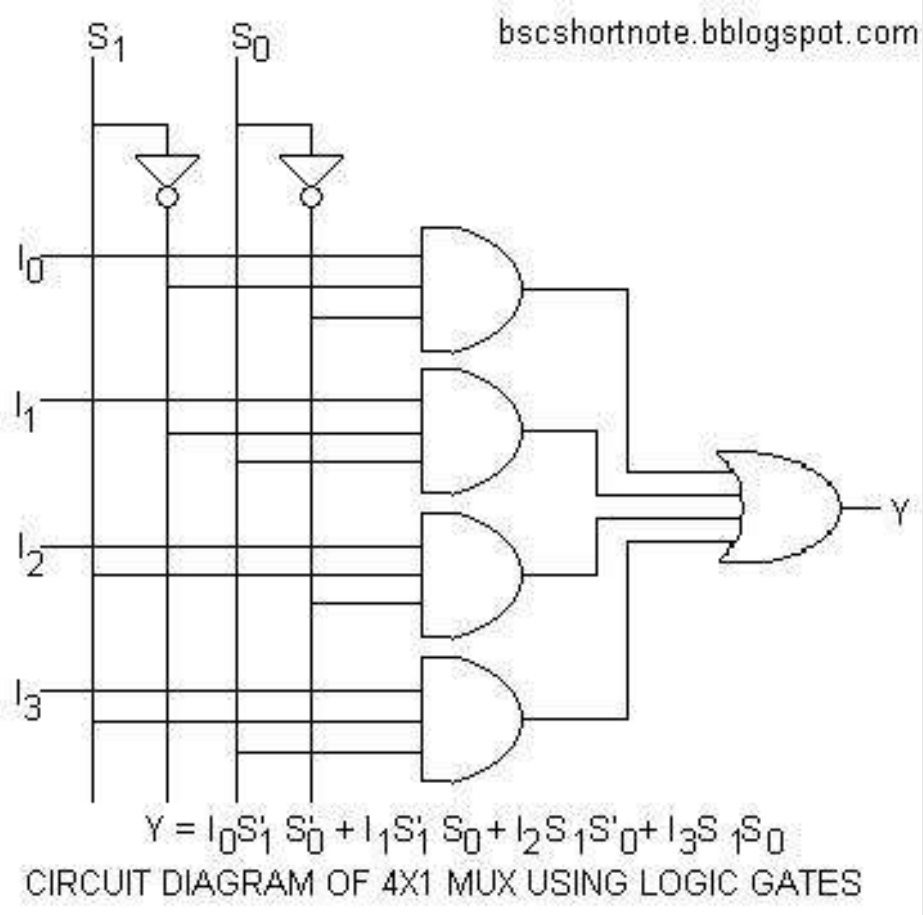
The input value of 1,0,1 for A,B,Bin gives output value of 0,0 for D and Bout resp. which confirms with the TruthTable.

FULL SUBTRACTOR – MIXED MODELLING

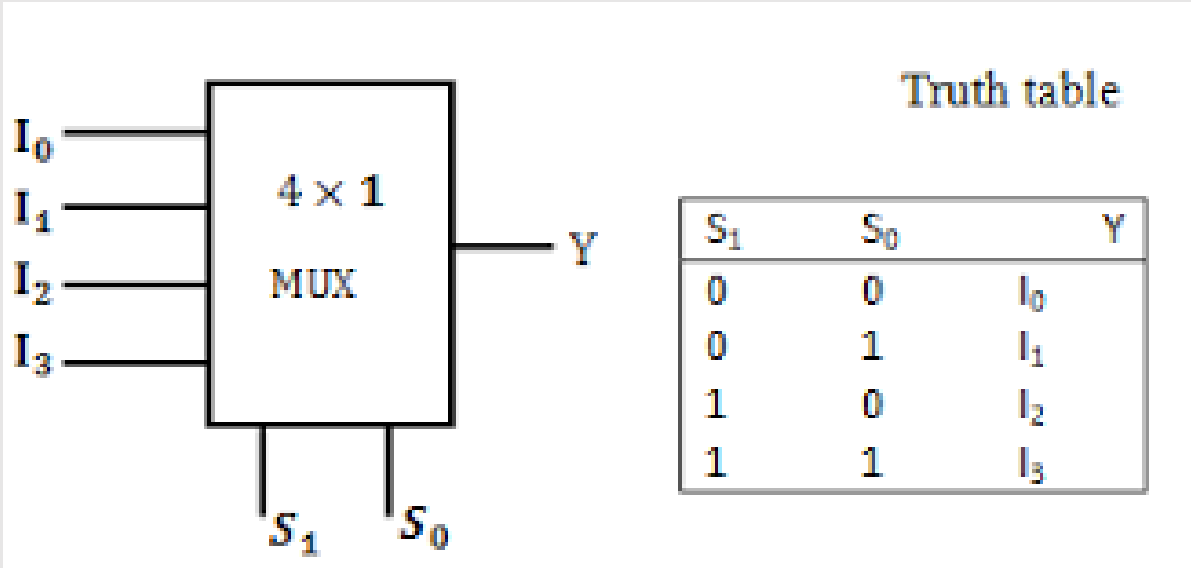


The mixed style of modelling was based upon this particular style discussed in class for a Full Adder.

4x1 MULTIPLEXER



CIRCUIT DIAGRAM AND LOGIC EQUATION

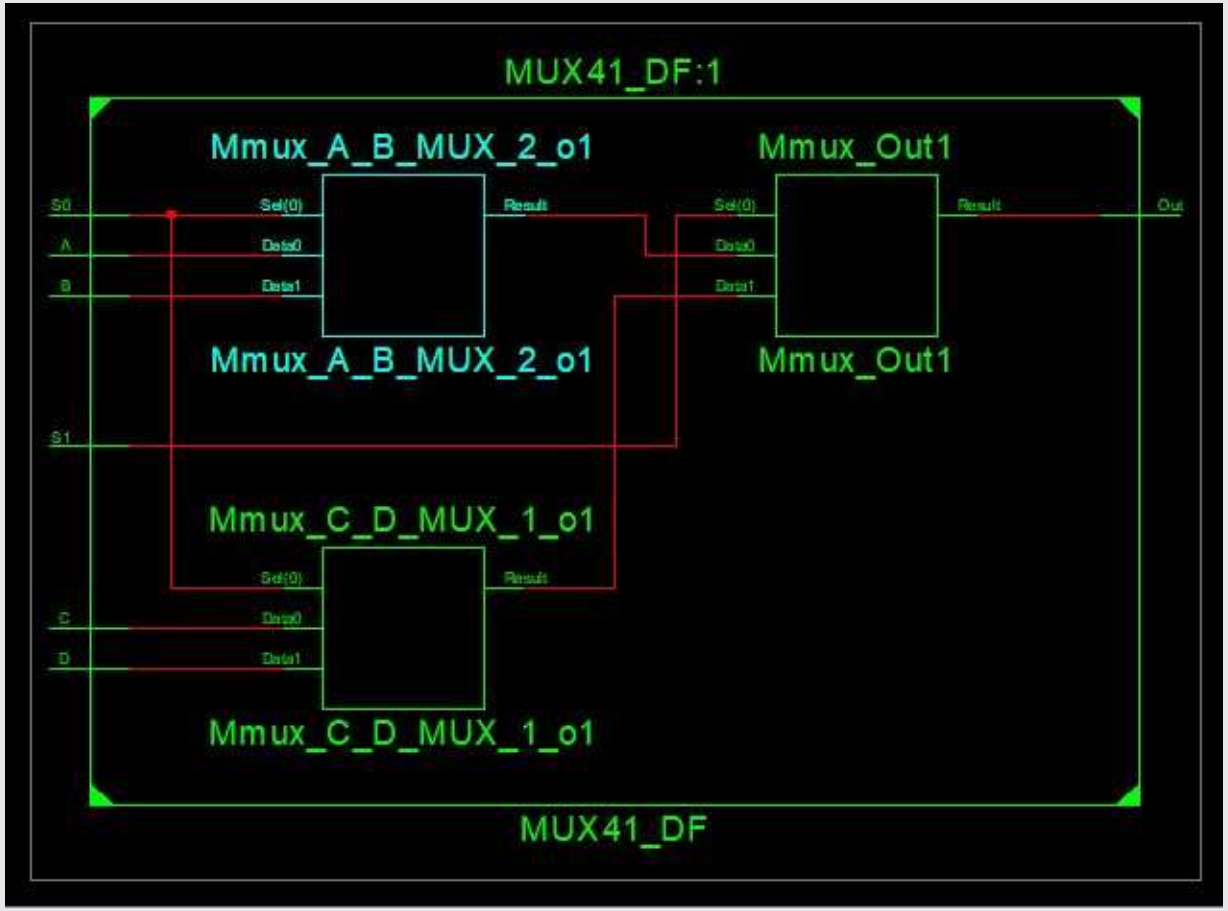


TRUTH TABLE

4x1 MUX – DATAFLOW

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date:    00:10:01 09/09/2020
7  // Design Name:
8  // Module Name:    MUX41_DF
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 MUX41_DF(
22     input A, B, C, D, S0, S1,
23     output Out
24     );
25 assign Out = S1?(S0?D:C):(S0?B:A);
26
27 endmodule
28
```

Code for Dataflow Model

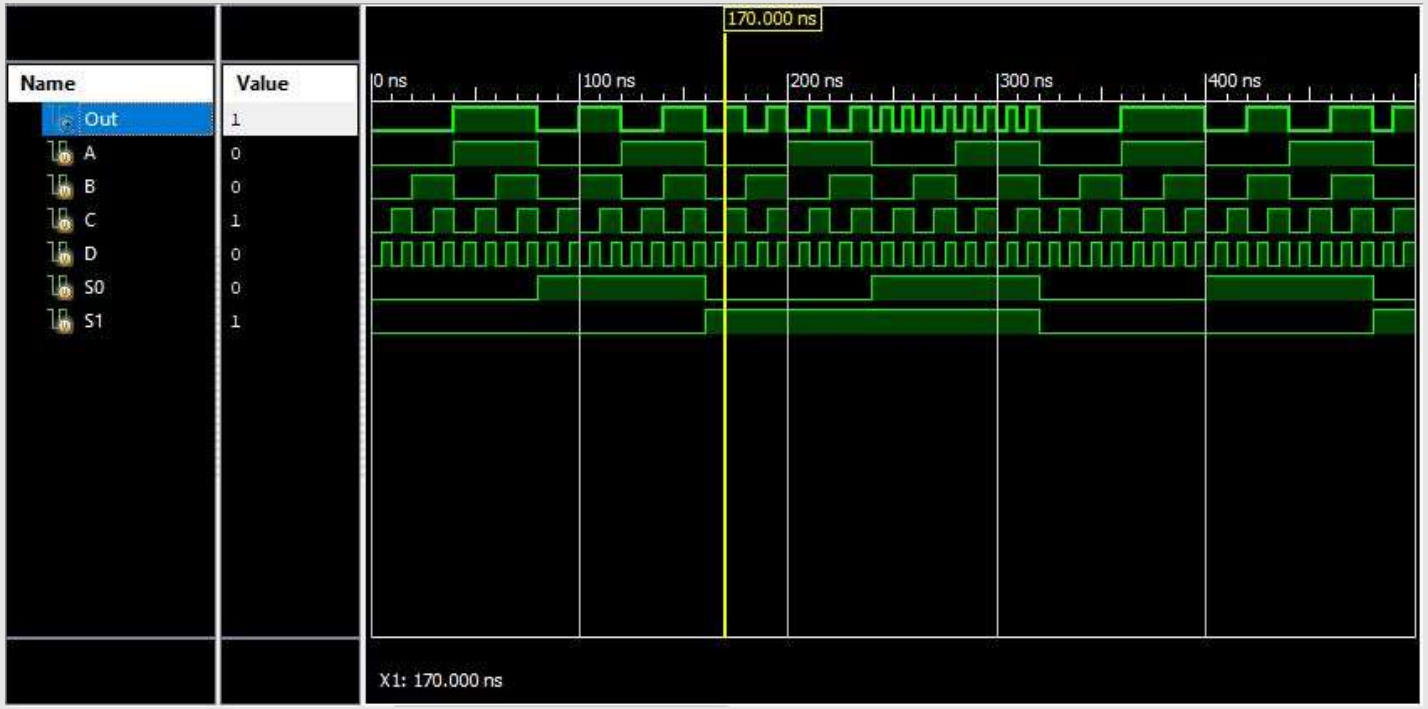


Schematic for Dataflow Model

4x1 MUX – DATAFLOW

```
39      MUX41_DF uut (
40          .A(A),
41          .B(B),
42          .C(C),
43          .D(D),
44          .S0(S0),
45          .S1(S1),
46          .Out(Out)
47      );
48
49      initial begin
50          // Initialize Inputs
51          A=1'b0; B=1'b0; C=1'b0; D=1'b0; S0=1'b0; S1=1'b0;
52          #500 $finish;
53
54      end
55
56      always #40 A=~A;
57      always #20 B=~B;
58      always #10 C=~C;
59      always #5 D=~D;
60      always #80 S0=~S0;
61      always #160 S1=~S1;
62
63      always@(A or B or C or D or S0 or S1)
64      $monitor("At time = %t, Output = %d", $time, Out);
65
66  endmodule
```

Code for Testbench



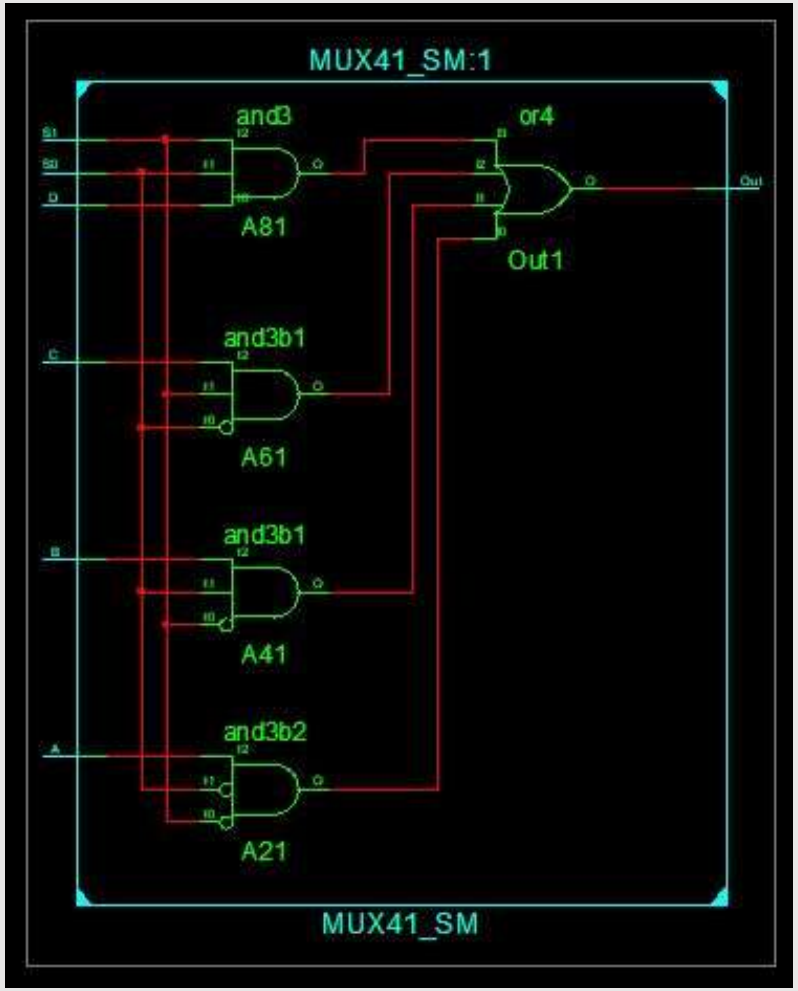
Waveform for Dataflow Model:

The input value of 1,0 for S1,S0 indicates that the output value should be equal to the third data input, i.e., C which is exactly the case here

4x1 MUX – STRUCTURAL

```
13  //
14  // Dependencies:
15  //
16  // Revision:
17  // Revision 0.01 - File Created
18  // Additional Comments:
19  //
20  //////////////////////////////////////////////////////////////////////////////////
21  module MUX41_SM(
22      input A, B, C, D, S0, S1,
23      output Out
24      );
25  wire A1, A2, A3, A4, A5, A6, A7, A8, O1, O2;
26
27  and a1(A1,~S1,~S0);
28  and a2(A2,A1,A);
29  and a3(A3,~S1,S0);
30  and a4(A4,A3,B);
31  and a5(A5,S1,~S0);
32  and a6(A6,A5,C);
33  and a7(A7,S1,S0);
34  and a8(A8,A7,D);
35  or o1(O1,A2,A4);
36  or o2(O2,O1,A6);
37  or o3(Out,O2,A8);
38
39
40  endmodule
```

Code for Structural Model

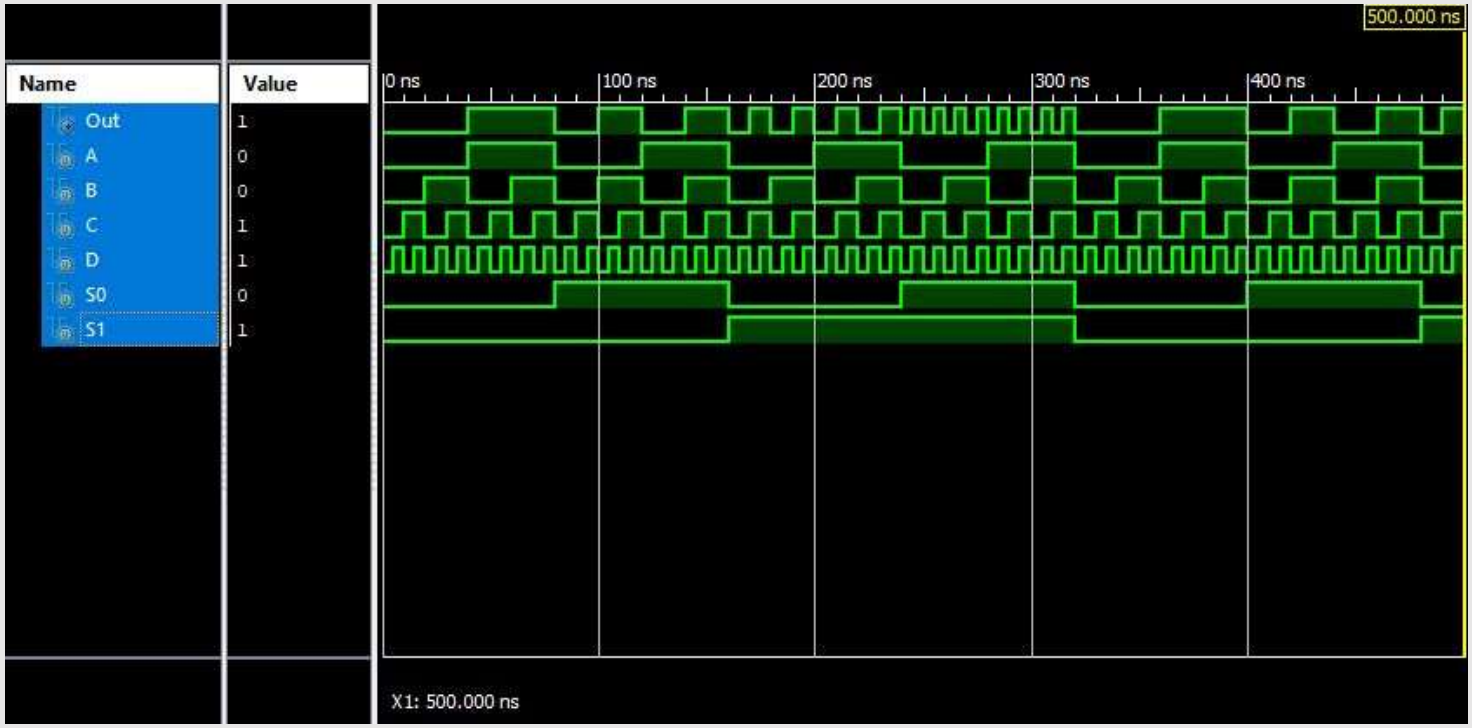


Schematic for Structural Model

4x1 MUX – STRUCTURAL

```
39      MUX41_DF uut (
40          .A(A),
41          .B(B),
42          .C(C),
43          .D(D),
44          .S0(S0),
45          .S1(S1),
46          .Out(Out)
47      );
48
49      initial begin
50          // Initialize Inputs
51          A=1'b0; B=1'b0; C=1'b0; D=1'b0; S0=1'b0; S1=1'b0;
52          #500 $finish;
53
54      end
55
56      always #40 A=~A;
57      always #20 B=~B;
58      always #10 C=~C;
59      always #5 D=~D;
60      always #80 S0=~S0;
61      always #160 S1=~S1;
62
63      always@(A or B or C or D or S0 or S1)
64      $monitor("At time = %t, Output = %d", $time, Out);
65
66  endmodule
```

Code for Testbench



Waveform for Structural Model:

The input value of 1,0 for S1,S0 indicates that the output value should be equal to the third data input, i.e., C which is exactly the case here

FINAL DISCUSSIONS

1. For Full Subtractor circuit, all the three types of modelling gave the same RTL schematic diagrams because they were mostly based on algorithms and data flow rather than gate level logics.
2. For 4x1 MUX, I could see the difference in RTL Schematics for Dataflow and Structural(Gate Level) Models.
3. I was unable to search for a method to implement a single test bench for all type of models but the resulting waveforms were same and matching with the truth tables confirming the correctness of the designing and simulations.
4. By the help of this assignment I learnt about Verilog coding and HDL abstraction levels. I covered behavioural, dataflow and structural modelling since it was in the scope of this assignment and read about the switch level of modelling too.
5. Overally, I was able to get familiar with Verilog code, Test benches and the Xilinx environment with the help of this assignment.