ELD LAB HOMEWORK-1

Aditya Gautam
2023043

- Note 1: I am on Arch Linux operating system and hence, the date and time on the top right do not appear like the one on Windows operating system
- Note 2: $ cat [filename] is a bash command which shows the contents of a file. I am using that to show that I intend to show that file's contents here
- Note 3: This line has been added by Vivado 2019.1 by default : `` `timescale 1ns / 1ps ``

$ cat top_adder.v

```
`timescale 1ns / 1ps
module top_adder (
       input [3:0] InA,
       input [3:0] InB,
       output [3:0] OutSum,
       output overflow
);

  wire carry1, carry2, carry3;

  full_adder_1bit in0 (
       .FA1_InA(InA[0]),
       .FA1_InB(InB[0]),
       .FA1_InC(1'b0),
       .FA1_OutSum(OutSum[0]),
       .FA1_OutC(carry1)
  );
  full_adder_1bit in1 (
       .FA1_InA(InA[1]),
       .FA1_InB(InB[1]),
       .FA1_InC(carry1),
       .FA1_OutSum(OutSum[1]),
       .FA1_OutC(carry2)
  );
  full_adder_1bit in2 (
       .FA1_InA(InA[2]),
       .FA1_InB(InB[2]),
       .FA1_InC(carry2),
       .FA1_OutSum(OutSum[2]),
       .FA1_OutC(carry3)
  );
  full_adder_1bit in3 (
       .FA1_InA(InA[3]),
       .FA1_InB(InB[3]),
       .FA1_InC(carry3),
       .FA1_OutSum(OutSum[3]),
       .FA1_OutC(overflow)
  );

endmodule
```

$ cat full_adder_1_bit.v

```
`timescale 1ns / 1ps
module full_adder_1bit (
       input  FA1_InA,
       input  FA1_InB,
       input  FA1_InC,
       output FA1_OutSum,
       output FA1_OutC
);

  assign FA1_OutSum = FA1_InA ^ FA1_InB ^ FA1_InC;
  assign FA1_OutC   = ((FA1_InA ^ FA1_InB) & FA1_InC) | (FA1_InA & FA1_InB);

endmodule
```

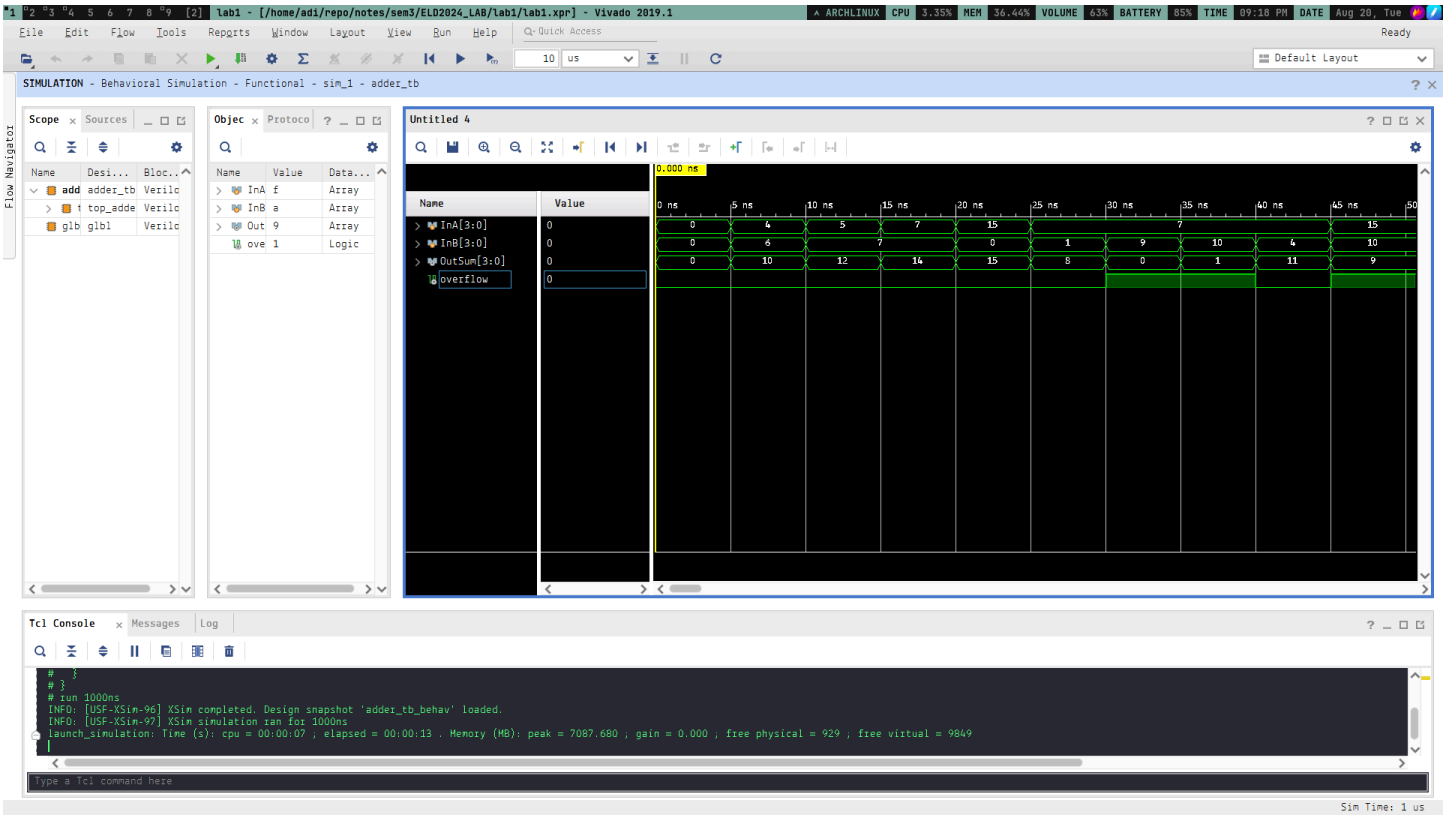
$ cat adder_tb.v

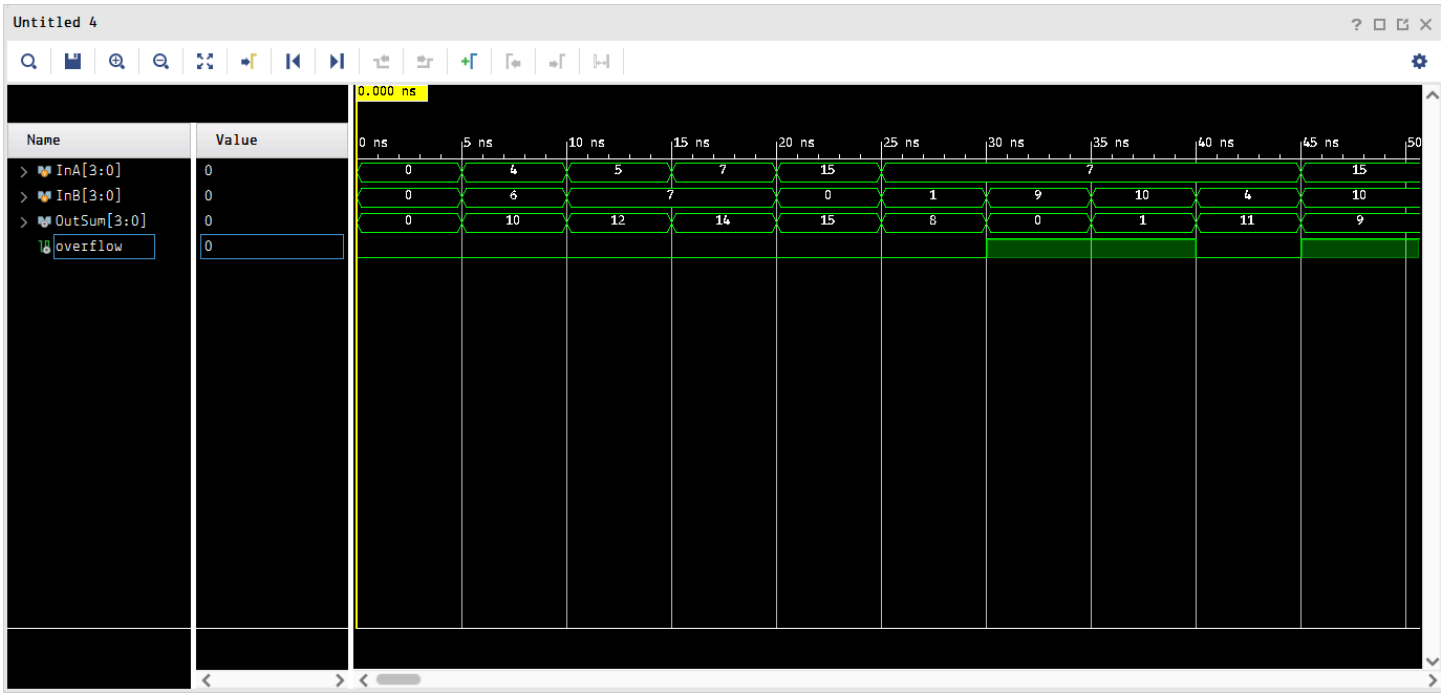
```
module adder_tb ();

  reg [3:0] InA, InB;
  wire [3:0] OutSum;
  wire overflow;

  top_adder tb0 (
       .InA(InA),
       .InB(InB),
       .OutSum(OutSum),
       .overflow(overflow)
  );


  initial begin
       InA = 4'b0000;
       InB = 4'b0000;
       #5 InA = 4'b0100;
       InB = 4'b0110;
       #5 InA = 4'b0101;
       InB = 4'b0111;
       #5 InA = 4'b0111;
       InB = 4'b0111;
       #5 InA = 4'b1111;
       InB = 4'b0000;
       #5 InA = 4'b0111;
       InB = 4'b0001;
       #5 InA = 4'b0111;
       InB = 4'b1001;
       #5 InA = 4'b0111;
       InB = 4'b1010;
       #5 InA = 4'b0111;
       InB = 4'b0100;
       #5 InA = 4'b1111;
       InB = 4'b1010;
  end
endmodule
```





Changes Made

- Changed the dimension of OutSum from 5 to 4
- Declared a new variable overflow as an output
- Set the variable which gets the value returned by Carry out to the new overflow variable in the 4th full_adder_1_bit module call