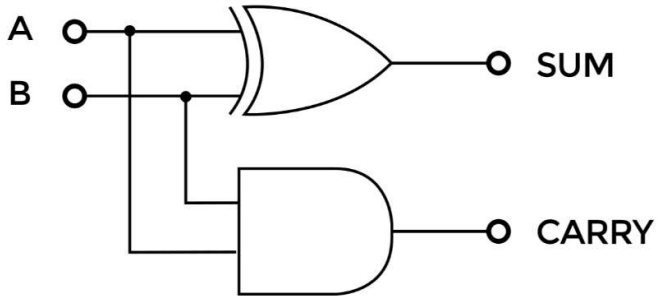
Digital Design Lab Report

Name	Dev Goel
Roll No.	B20CS090
Experiment Number	**I**

Objective

(i) Create a Half Adder. Write its test bench and simulate.
(ii) Use Half Adder to create a Full Adder.
(iii) Use Full Adder to create a Four Bit Adder. Write its test bench and simulate.

Work 1: Create a Half Adder. Write its test bench and simulate.



A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

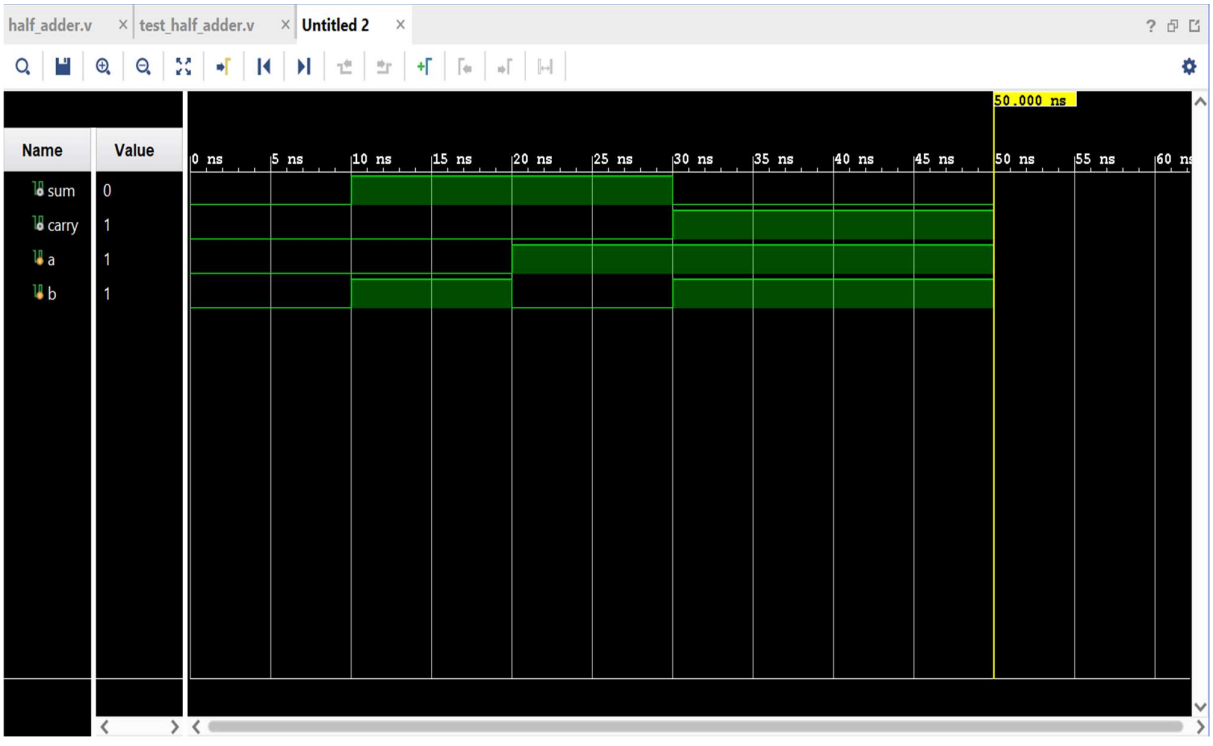
- **Half Adder**

```
`timescale 1ns / 1ps
module half_adder(a, b, sum, carry);
input a, b;
output sum, carry;
assign sum = a^b;
assign carry = a&b;
endmodule
```

- **Test Bench**

```
`timescale 1ns / 1ps
module test_half_adder;
wire sum, carry;
reg a, b;
half_adder H1(a, b, sum, carry);
initial
  begin
      a = 1'b0;
      b = 1'b0;
      #10 b= 1'b1;
      #10 b = 1'b0; a = 1'b1;
      #10 b = 1'b1;
  end
initial #50 $finish;
endmodule
```

- **Waveform**



Work 2: Use Half Adder to create a Full Adder.

- **Full Adder**

```
`timescale 1ns / 1ps
module full_adder(a, b, c, sum, carry);
input a, b, c;
output sum, carry;
wire ha1sum, ha1carry, ha2sum, ha2carry;
half_adder ha1(b, c, ha1sum, ha1carry);
half_adder ha2(a, ha1sum, ha2sum, ha2carry);
assign sum = ha2sum;
assign carry = ha1carry|ha2carry;
endmodule
```

Work 3: Use Full Adder to create a Four Bit Adder. Write its test bench and simulate.

- **Four Bit Adder**

```
`timescale 1ns / 1ps
module four_bit_adder(a0, b0, a1, b1, a2, b2, a3, b3, s0, s1, s2, s3, carry);
input a0, b0, a1, b1, a2, b2, a3, b3;
output s0, s1, s2, s3, carry;
wire fa0carry, fa1carry, fa2carry;
full_adder f1(a0, b0, 0, s0, fa0carry);
full_adder f2(a1, b1, fa0carry, s1, fa1carry);
full_adder f3(a2, b2, fa1carry, s2, fa2carry);
full_adder f4(a3, b3, fa2carry, s3, carry);
endmodule
```

- **Test Bench**

```
`timescale 1ns / 1ps
module test_four_bit_adder;
wire s0, s1, s2, s3, carry;
reg a0, b0, a1, b1, a2, b2, a3, b3;
four_bit_adder fba(a0, b0, a1, b1, a2, b2, a3, b3, s0, s1, s2, s3, carry);
initial
 begin
   // 0000 + 0000
   a0 = 1'b0; b0 = 1'b0; a1 = 1'b0; b1 = 1'b0; a2 = 1'b0; b2 = 1'b0; a3 =
1'b0; b3 = 1'b0;
   // 0001 + 0001
   #10 a0 = 1'b1; b0 = 1'b1;
   // 0101 + 0111
   #10 a2 = 1'b1; b2 = 1'b1; b1 = 1'b1;
   // 1000 + 0100
   #10 a3 = 1'b1; a2 = 1'b0; a0 = 1'b0; b1 = 1'b0; b0 = 1'b0;
   // 1111 + 0001
   #10 a2 = 1'b1; a1 = 1'b1; a0 = 1'b1; b2 = 1'b0; b0 = 1'b1;
   // 1111 + 1111
   #10 b3 = 1'b1; b2 = 1'b1; b1 = 1'b1;
 end
initial #60 $finish;
endmodule
```

- **Waveform**