Introduction to Combinational Circuit Simulation Lab: 2

Half Adder

Student : Chandani Lapasia

User id : fpga0522-chan50

Code snippet for Half adder DATA flow modelling.

Design:

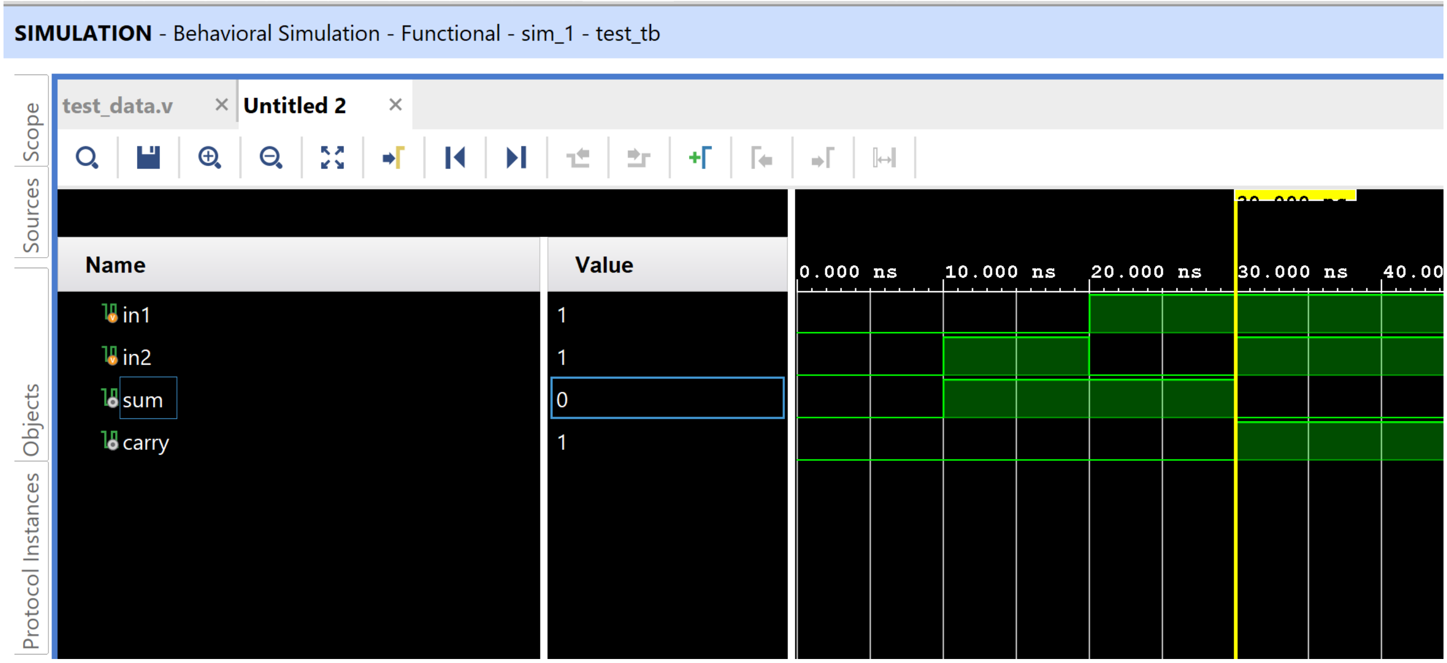
```
module half_adder(sum,carry,in1,in2);
input in1, in2;
output sum, carry;

assign sum=(~in1 && in2) || (in1 && ~in2);
assign carry=(in1 && in2);

endmodule
```

Testbench:

```
module test_tb();
reg in1,in2;
wire sum, carry;

half_adder u0 (sum,carry,in1,in2);

initial begin
in1=0; in2=0;
#10 in1=0; in2=1;
#10 in1=1; in2=0;
#10 in1=1; in2=1;
end
```

```
initial begin

$monitor($time, "in1=%d in2=%d carry=%d sum=%d", in1, in2,carry, sum);

end


endmodule
```

Output waveform:



SIMULATION - Behavioral Simulation - Functional - sim_1 - test_tb

Name	Value
in1	1
in2	1
sum	0
carry	1

Output console :

```
 0 in1=0 in2=0 carry=0 sum=0

10 in1=0 in2=1 carry=0 sum=1

20 in1=1 in2=0 carry=0 sum=1

30 in1=1 in2=1 carry=1 sum=0
```

Code snippet for Half adder BEHAVORAL flow modelling.

Design code :

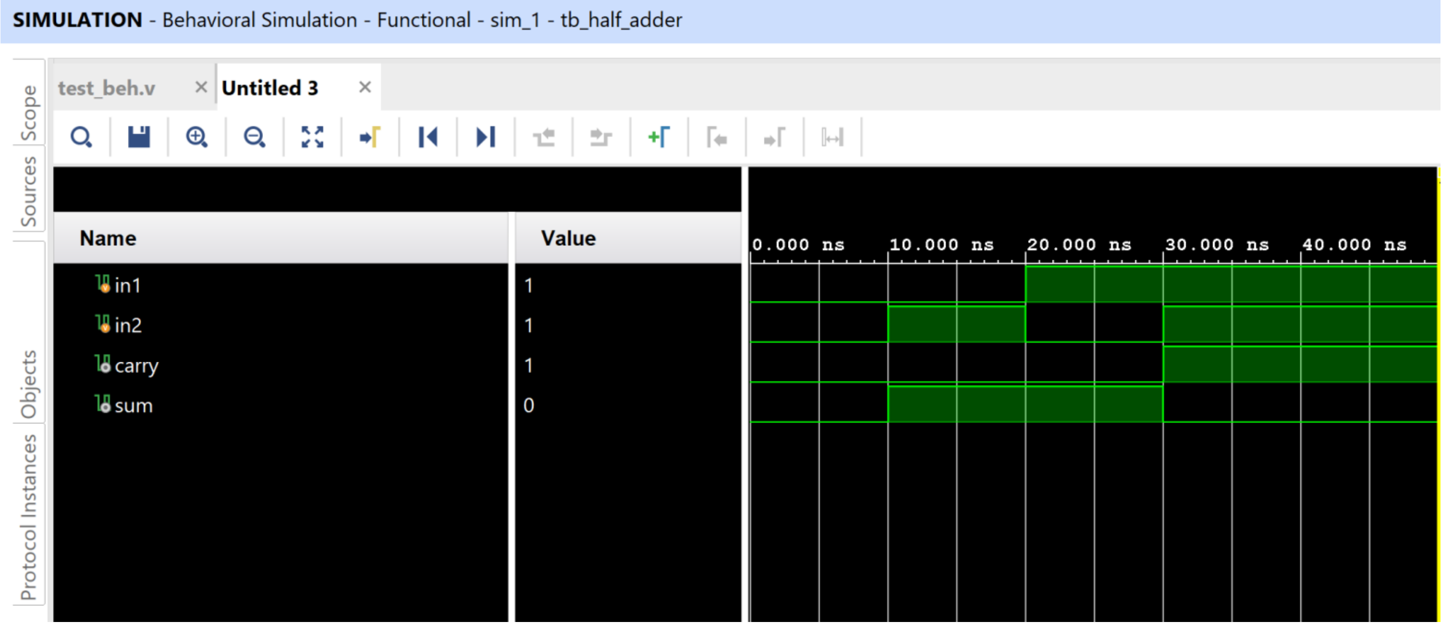
```
module half_adder_beh(carry,sum,in1,in2);

input in1, in2;
output sum, carry;
reg sum,carry;
always @(in1 or in2)
begin
{carry,sum}= in1+in2;
end
endmodule
```

Testbench code :

```
module tb_half_adder();

reg in1, in2;
wire carry, sum;

half_adder_beh u0(carry,sum,in1,in2);

initial begin
in1=0; in2=0;
#10 in1=0; in2=1;
#10 in1=1; in2=0;
#10 in1=1; in2=1;
end

initial begin
```

```
$monitor($time, "in1=%d in2=%d carry=%d sum=%d", in1, in2,carry, sum);
end


endmodule
```

Output waveform :



SIMULATION - Behavioral Simulation - Functional - sim_1 - tb_half_adder

Name	Value
in1	1
in2	1
carry	1
sum	0

Output console :

```
 0 in1=0 in2=0 carry=0 sum=0

10 in1=0 in2=1 carry=0 sum=1

20 in1=1 in2=0 carry=0 sum=1

30 in1=1 in2=1 carry=1 sum=0
```

Code snippet for Half adder STRUCTURAL flow modelling.

Design code :

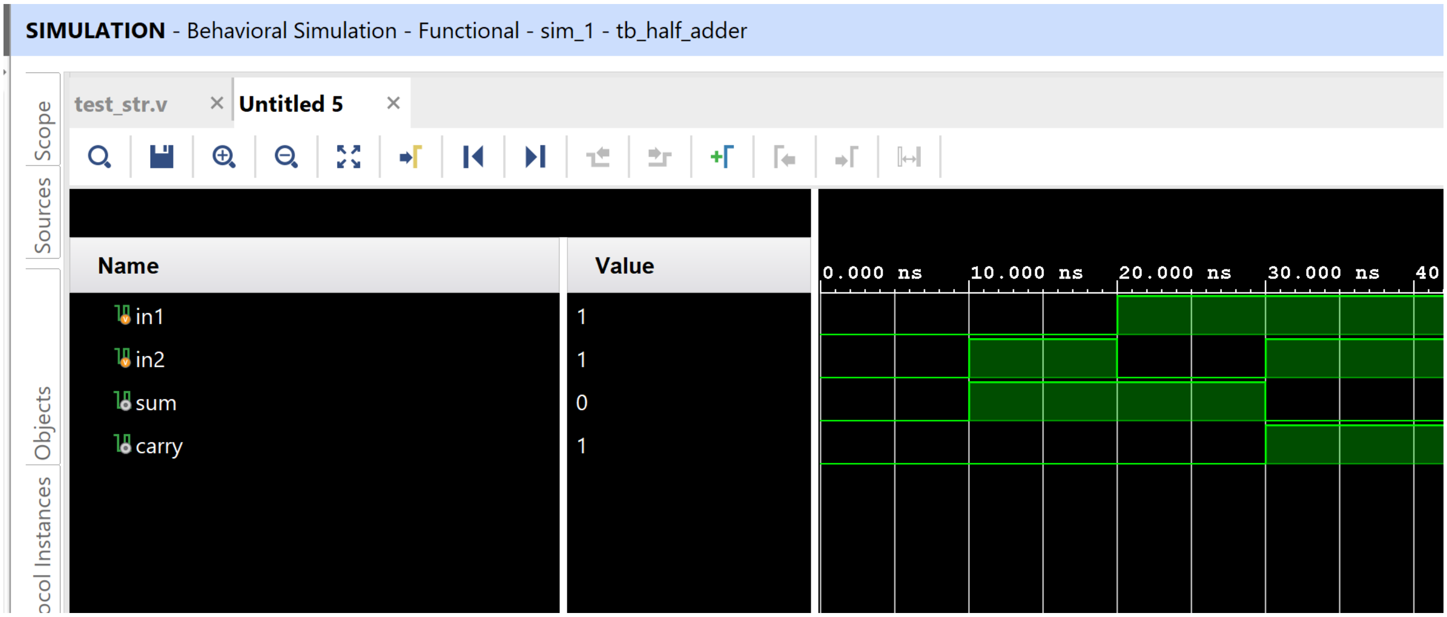
```
module test_half_adder_str(sum,carry,in1,in2);


input in1, in2;
output sum,carry;


wire temp1, temp2, temp3, temp4;


and(carry, in1, in2);


not(temp1, in1);
not(temp2,in2);
and(temp3,temp1,in2);
and(temp4,in1,temp2);
or(sum,temp3,temp4);


endmodule
```

Testbench code :

```
module tb_half_adder();


reg in1,in2;
wire sum,carry;
test_half_adder_str u0(sum,carry,in1,in2);
```

```
initial begin

in1=0; in2=0;

#10 in1=0; in2=1;

#10 in1=1; in2=0;

#10 in1=1; in2=1;

end


initial begin

$monitor($time, "in1=%d in2=%d carry=%d sum=%d", in1, in2,carry, sum);

end


endmodule
```

Output waveform :



SIMULATION - Behavioral Simulation - Functional - sim_1 - tb_half_adder

Name	Value
in1	1
in2	1
sum	0
carry	1

Output console :

```
         0 in1=0 in2=0 carry=0 sum=0
        10 in1=0 in2=1 carry=0 sum=1
        20 in1=1 in2=0 carry=0 sum=1
        30 in1=1 in2=1 carry=1 sum=0
```

Question and Answers for Lab2 are answered in second PDF along with Full Adder design.

=============================== END===