RTL CODING

#DAY1 Yash Ekhande

Aim - To make a Full Adder using gate primitives

Code

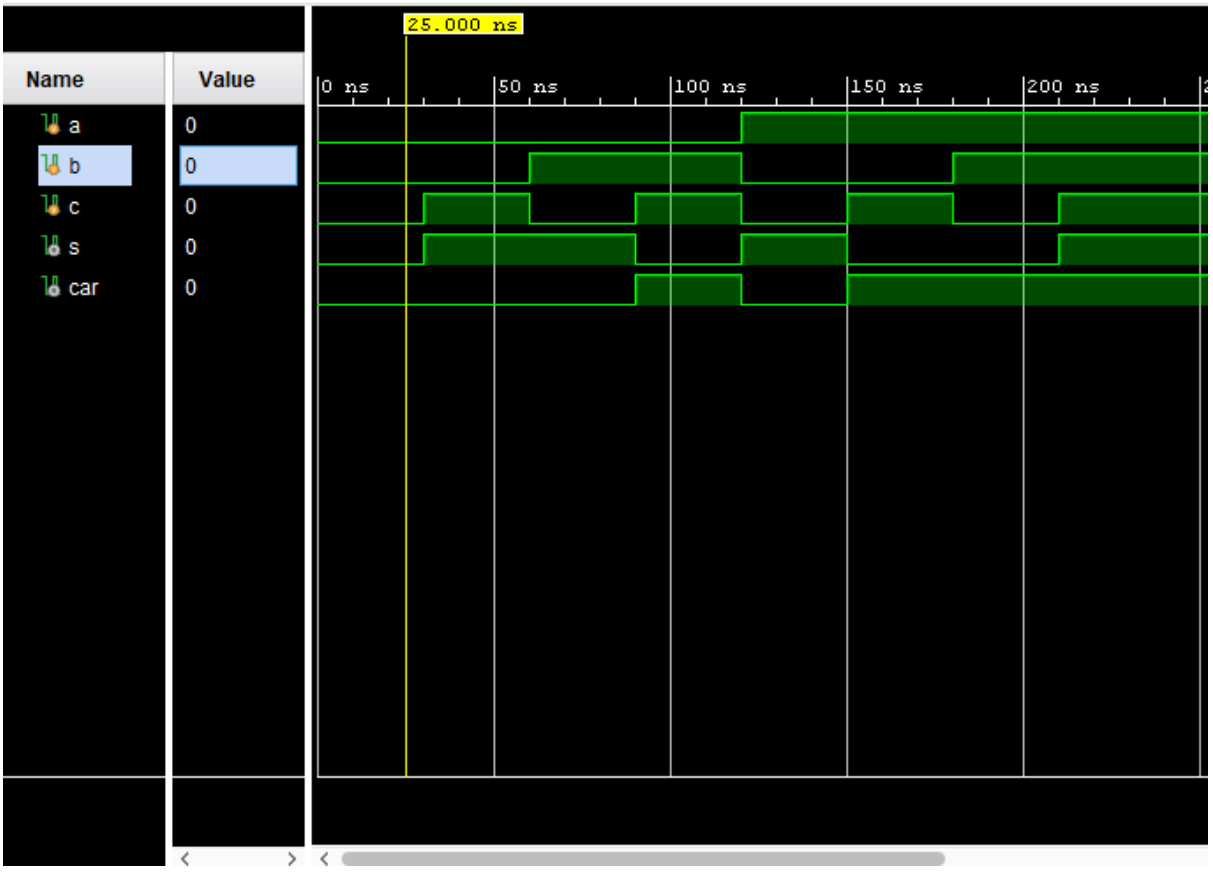
```
module FA_gate(
    input a,
    input b,
    input c,
    output sum,
    output  cout
    );
wire s1,s2,s3;
xor x1(s1,a,b);
and a1(s2,a,b);
xor x2(sum,s1,c);
and a2(s3,s1,c);
or y1(cout,s3,s2);

endmodule
```

Testbench

```
module FAgatetb;
reg a,b,c;
wire s,car;
FA_gate f0(a,b,c,s,car);
initial begin
a=1'b0; //1st case
b=1'b0;
c=1'b0;
#30;
a=1'b0; //2d case
b=1'b0;
c=1'b1;
#30;
a=1'b0; //3rd case
b=1'b1;
c=1'b0;
#30;
a=1'b0; // 4t case
b=1'b1;
c=1'b1;
#30;
a=1'b1; //5t case
b=1'b0;
c=1'b0;
#30;
```

```
a=1'b1; //6th case
b=1'b0;
c=1'b1;
#30;
a=1'b1; //7t case
b=1'b1;
c=1'b0;
#30;
a=1'b1; //8t case
b=1'b1;
c=1'b1;
#30;
end
endmodule
```

Waveform



Schematic