

RTL CODING

#DAY3

Yash Ekhande

AIM - Implementation of Controlled Adder and Subtractor(4-bit)

Function- if K=0 the circuit will perform addition , k=1 the circuit will perform Subtraction

CODE

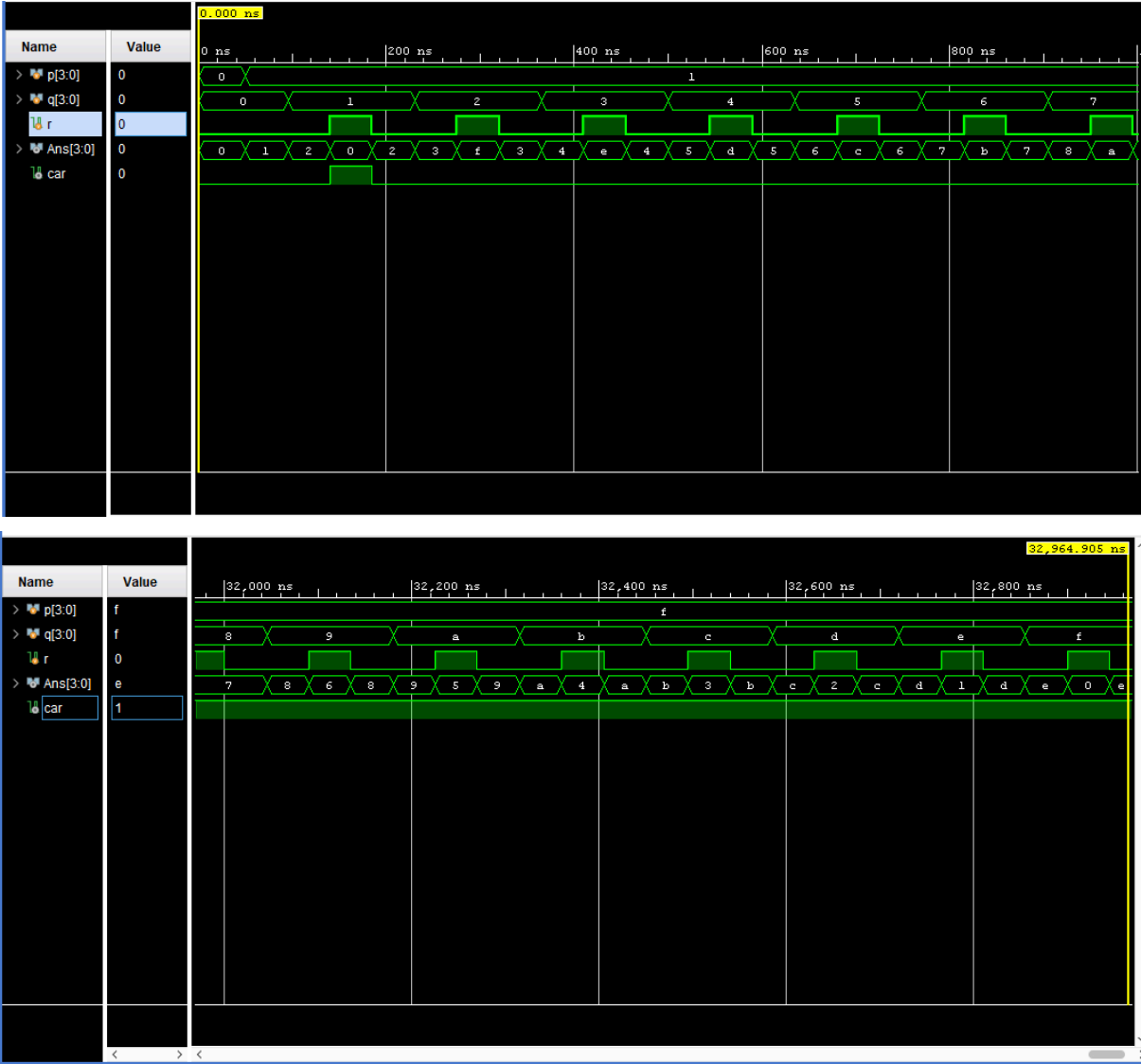
```
1  `timescale 1ns / 1ps
2  module Ctrl_adrsub(
3      input [3:0] a,
4      input [3:0] b,
5      input k,
6      output [3:0] Ans,
7      output cout
8  );
9      wire c1,c2,c3,s0,s1,s2,s3;
10     assign s0=k^b[0];
11     assign s1=k^b[1];
12     assign s2=k^b[2];
13     assign s3=k^b[3];
14     FA_gate z0(a[0],s0,k,Ans[0],c1);
15     FA_gate z1(a[1],s1,c1,Ans[1],c2);
16     FA_gate z2(a[2],s2,c2,Ans[2],c3);
17     FA_gate z3(a[3],s3,c3,Ans[3],cout);
18 endmodule
```

TESTBENCH

```
22 module ctrltb;
23     reg [3:0] p;
24     reg [3:0] q;
25     reg r;
26     wire [3:0] Ans;
27     wire car;
28     Ctrl_adrsub s1(p,q,r,Ans,car);
29     integer i,j,k; //data type declaration for loops
30     initial begin
31         p=4'b0000; //initial value declration(p,a)
32         q=4'b0000; //initial value declration(q,b)
33         r=1'b0; //initial value declaration of c
34         #50;
35         for(i=0;i<16;i=i+1) //changing values from 0 to 16 of a/p
36         begin
37             p=p+1;
38             #45 //acts as counter
39             for(j=0;j<16;j=j+1) // changing values for 0 to 16 of q/b
40             begin
41                 q=q+1;
42                 #45;
43                 for(k=0;k<2;k=k+1)
44                 begin
45                     r=r+1;
46                     #45;
47                 end
48             end
49         end
```

```
48 end
49 end
50 end
51 endmodule
52
53
```

WAVEFORM



SCHEMATIC

