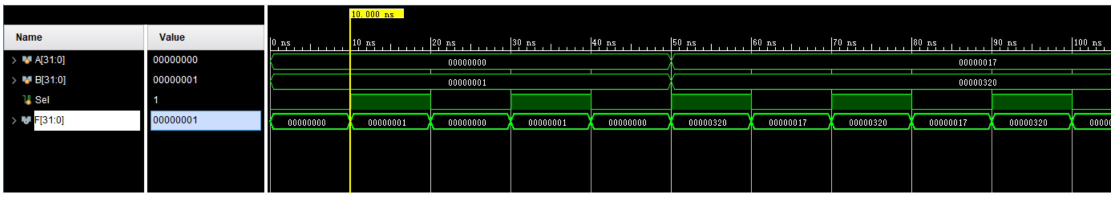
HW5 Report

Q1.

Verilog Code:

```
module Q1(A,B,Sel,F);
    input [31:0]A,B;
    input Sel;
    output [31:0] F;
    reg [31:0] F;
    always @(A,B,Sel)begin
        case(Sel)
            0'b0: F=A;
            1'b1: F=B;
            default F=0;
        endcase
    end
endmodule
```

Stimulation Result:

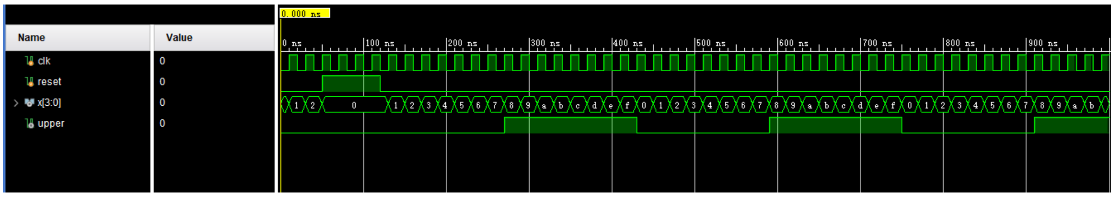


Q2.

Verilog Code:

```
module Q2(Data_in, clk, R ,Data_out);
    input [3:0] Data_in;
    input clk,R;
    output [3:0] Data_out;
    reg [3:0] Data_out;
    always @(posedge clk or negedge clk)
    if(R==1'b0) Data_out <= 4'b0;
    else Data_out <= Data_in;
endmodule
```

Stimulation Result:

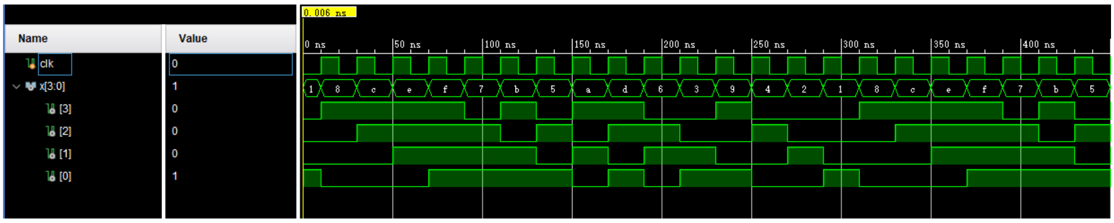


Q3.

Verilog Code:

```
module Q3(clk,x);
input clk;
output [3:0]x;
reg [3:0]x=4'b0001;
always @(posedge clk)
    begin
    x[0]<=x[1];
    x[1]<=x[2];
    x[2]<=x[3];
    x[3]<=x[0]+x[3];
    end
endmodule
```

Stimulation Result:

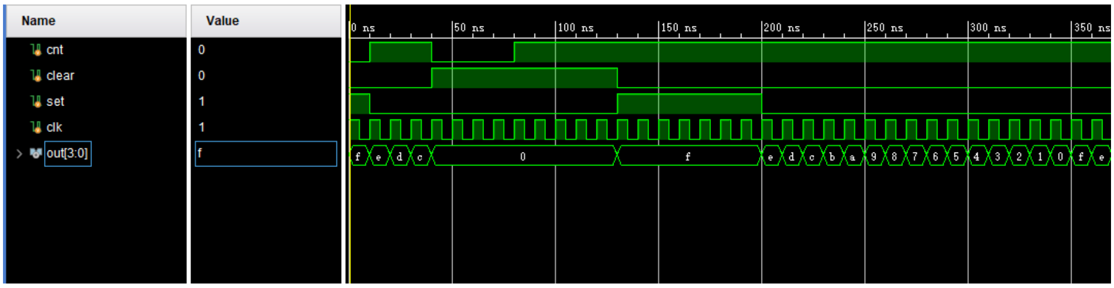


Q4.

Verilog Code:

```
module Q4(cnt,clear,set,clk,out);
input cnt,clear,set,clk;
output [3:0]out;
reg [3:0]out=4'b1111;
always @(posedge clk or posedge cnt or posedge clear or posedge set)begin
        if(set==1)out<=4'b1111;
        else if(clear==1)out<=4'b0000;
        else if(cnt==1)out<=out-4'b0001;
end
endmodule
```

Stimulation Result:



Name	Value
cnt	0
clear	0
set	1
clk	1
out[3:0]	f

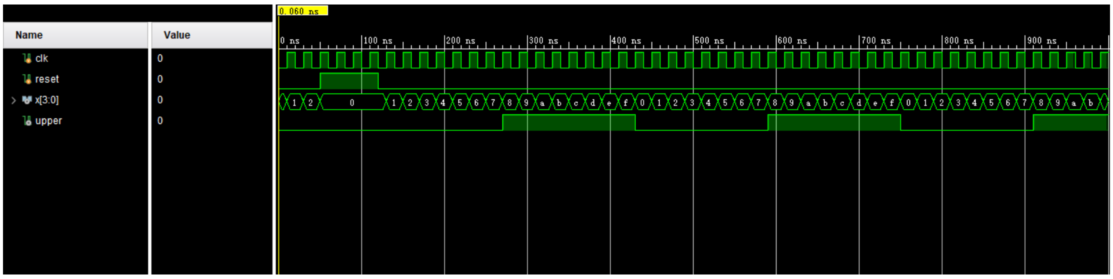
Q5.

Verilog Code:

```
module Q5(clk,x,upper,reset);
input clk,reset;
output upper;
output [3:0]x;

reg [3:0]x=4'b0000;
reg upper=1'b0;
always @(posedge clk or posedge reset)begin
    if(reset==1'b0001)x=4'b0000;
    else x=x+4'b0001;
    if(x>=8)upper=1'b1;
    else upper=1'b0;
end
endmodule
```

Stimulation Result:



Name	Value
clk	0
reset	0
x[3:0]	0
upper	0

0.060 ns
0 ns
100 ns
200 ns
300 ns
400 ns
500 ns
600 ns
700 ns
800 ns
900 ns
1 2 0 1 2 3 4 5 6 7 8 9 a b c d e f 0 1 2 3 4 5 6 7 8 9 a b c d e f 0 1 2 3 4 5 6 7 8 9 a b