FWC RTL Assignment 1

Question:

Design an 8-bit down counter. The count value is loaded from 'in' input on a positive clock edge when 'latch' input is high. Count value is decremented by 1 on a positive clock edge while 'dec' input is high. Stop decrementing at 0 and raise 'zero' flag output active high whenever count value is 0. 'latch' input has priority over 'dec' input. Whenever 'divide-by-two' input is high and 'latch' input is low and 'dec' input is low (both dec and divide-by-two will never be high at the same time), divide the current contents of the counter by 2.

module code:

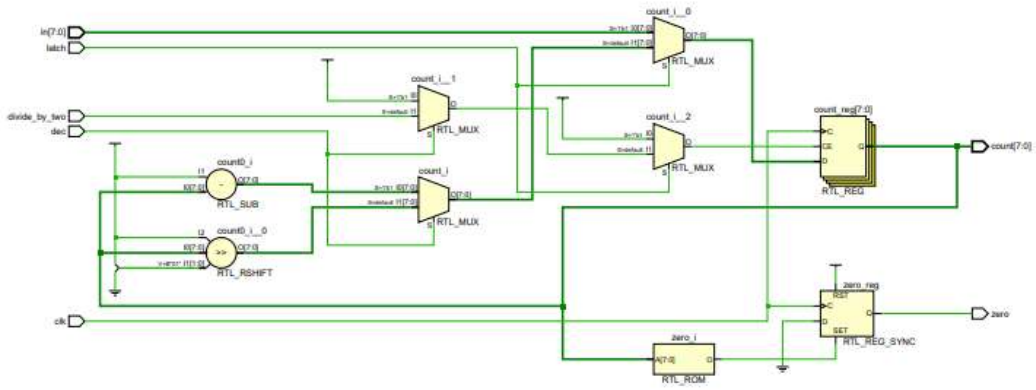
```
module counter(clk,in,latch,dec,divide_by_two,count,zero);
 input clk,latch,dec,divide_by_two;
 input [7:0] in;
 output reg [7:0] count;
 output reg zero;
 always@(posedge clk )
 begin
  if(latch)    //load the counter with in value
    count <= in;
  else if(dec)       //count down
     count <= count - 1;
  else if (divide_by_two)     //count/2
    count <= count/2;
  if (count==0)
     zero=1;
  else
    zero=0;
 end
endmodule
```

test bench code

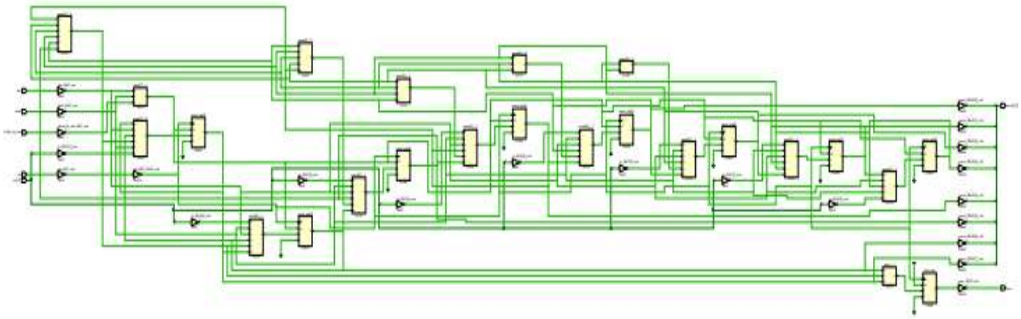
```
module counter8_tb();
 reg clk,latch,dec,divide_by_two;
 reg [7:0] in;
 wire [7:0] count;
 wire zero;
 counter ct_1(clk,in,latch,dec,divide_by_two,count,zero);
 //clock generator
 initial begin clk = 1'b0;
  forever #1 clk= ~clk;
 end
 //insert all the input signal
 initial begin #1 latch=1'b1;#1 latch=1'b0; end
 initial begin #1 dec=1'b1;#5 dec=1'b0;#5 dec=1'b1; end
 initial begin
  #1 divide_by_two=1'b0;
   #5 divide_by_two=1'b1;
```

```
  #5 divide_by_two=1'b0;end
 initial begin in=8'b00010010;end
 initial begin #30 $finish ; end
endmodule
```

design diagram



Netlist



Timing diagram: