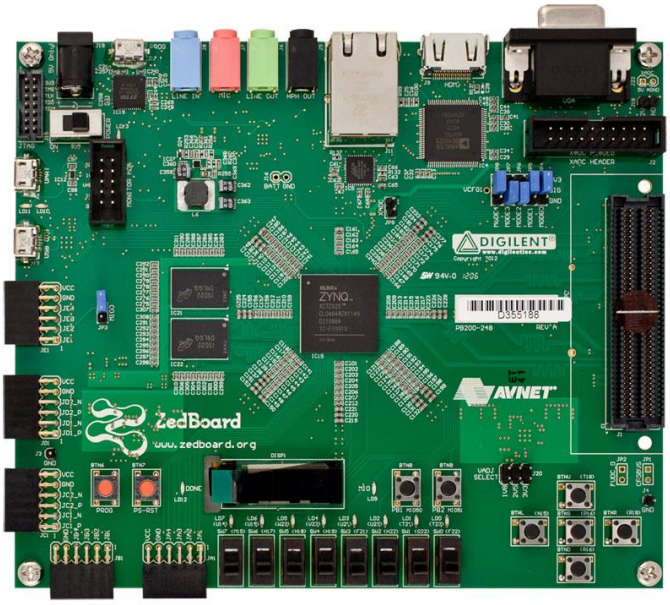
ELECTRONIC ROULETTE WHEEL

COMPONENTS USED:

- MOD 8 COUNTER
- 3:8 DECORDER
- Zed-BOARD (Zynq 7000 Soc)



ZedBoard™ is a development board for the Xilinx Zynq®-7000 SoC. This board contains everything necessary to create a Linux, Android, Windows® or other OS/RTOS-based design. Additionally, several expansion connectors expose the processing system and programmable logic I/Os for easy user access. Take advantage of the Zynq-7000 SoC's tightly coupled ARM® processing system and 7 series programmable logic to create unique and powerful designs with the ZedBoard.

SOFTWARE USED :

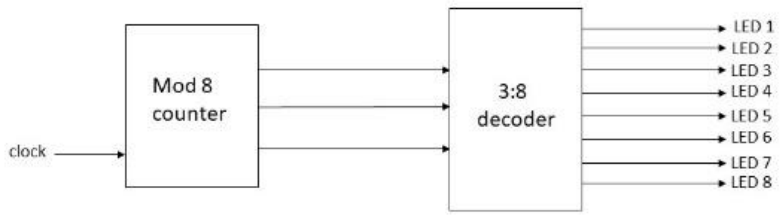
- Vivado xilinx (Older version)

WORKING

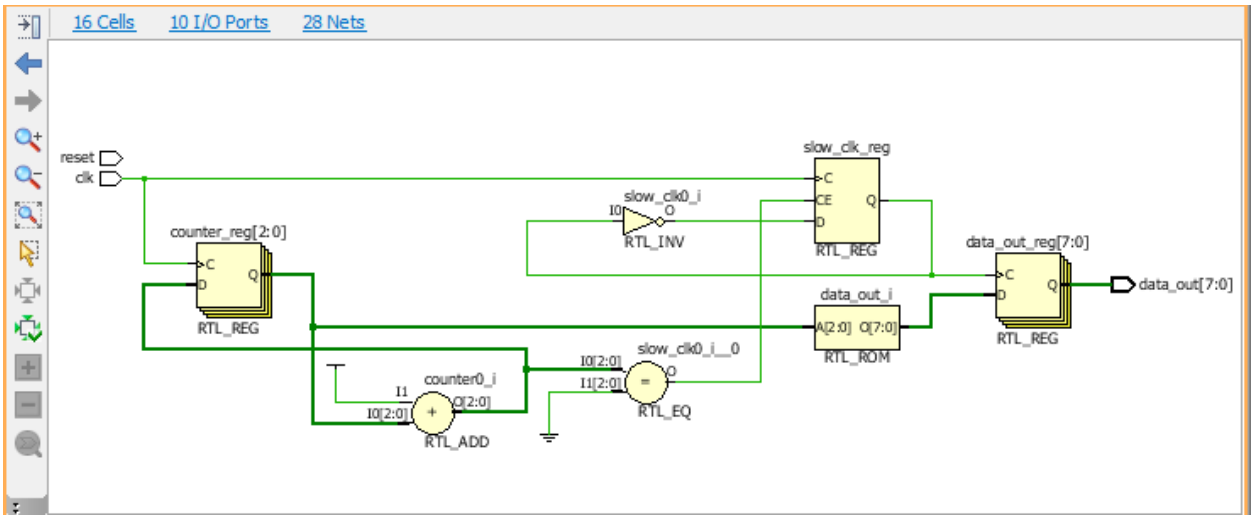
A push button is used to give clock signal to all the three D flipflops of a Mod-8 synchronous up counter. Initially the clear is zero and it is connected to the all the flip flops of the counter. When the push button is pressed , a clock signal is given to the counter. For every clockpulse ,the counter is going to count from zero to seven in a loop.

The output of this counter is connected to the 3:8 decoder. Initially when the input is 000 , the first LED (S-0) lights up . As the input is changes sequentially, the next LED will light up and the previous LED is cut off .This is going to work till the input of the clock is 111 and after that it is going to get reverted back to 000 that is (S-0) . This process will continue when the push button is released and a corresponding LED will glow according to the output of the counter.

The code is dumbed to Z-BOARD .The reset pin of the counter is given to the 22th pin of FPGA . Clock is connected to Y9 pin of FPGA and Output LEDs are connected to the correspond BANK 33 of FPGA



VIVADO DESIGN

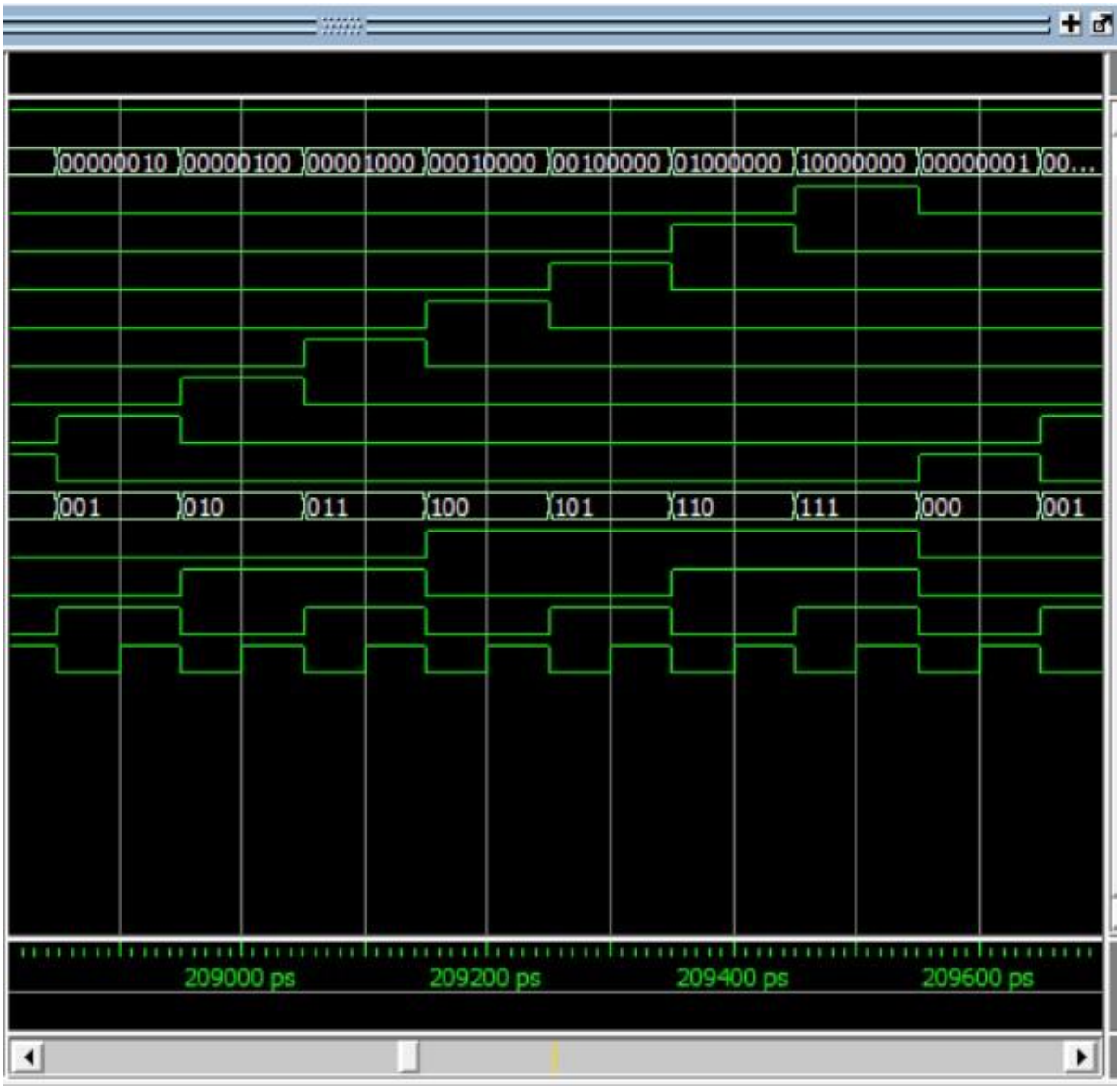


VERILOG CODE

```
module rowlette(
 input clk,reset,
 output reg[7:0] data_out
 );
  reg slow_clk;
 reg [2:0] counter;
 initial begin
   slow_clk = 1'b0;
   counter = 0;
   data_out=8'b00000000;
 end
 always @ (posedge clk)
 begin
   counter = counter + 1;
   if (counter == 0)
     slow_clk = ~slow_clk;
   end
     always @ (posedge slow_clk)
 begin
```

```
    case(counter)
       3'b000:data_out=8'b00000001;
       3'b001:data_out=8'b00000010;
       3'b010:data_out=8'b00000100;
       3'b011:data_out=8'b00001000;
       3'b100:data_out=8'b00010000;
       3'b101:data_out=8'b00100000;
       3'b110:data_out=8'b01000000;
       3'b111:data_out=8'b10000000;
      endcase
   if (counter == 8)
    begin
      counter = 0;
    end
   end
endmodule
```

WAVEFORM


00000010
00000100
00001000
00010000
00100000
01000000
10000000
00000001
001
010
011
100
101
110
111
000
001
209000 ps
209200 ps
209400 ps
209600 ps

CONCLUSION

The Electronic Roulette Wheel has been successfully implemented . One person have to press the push switch which is used to give a input .Then the LED's rotate for some time and come to halt at a particular led according to the time of push button pressed .

The working of MOD 8 Synchronous up counter , 3:8 Decoder and FPGA is summarized

BIBLIOGRAPHY

- www.youtube.com
- www.wikkipedia.com