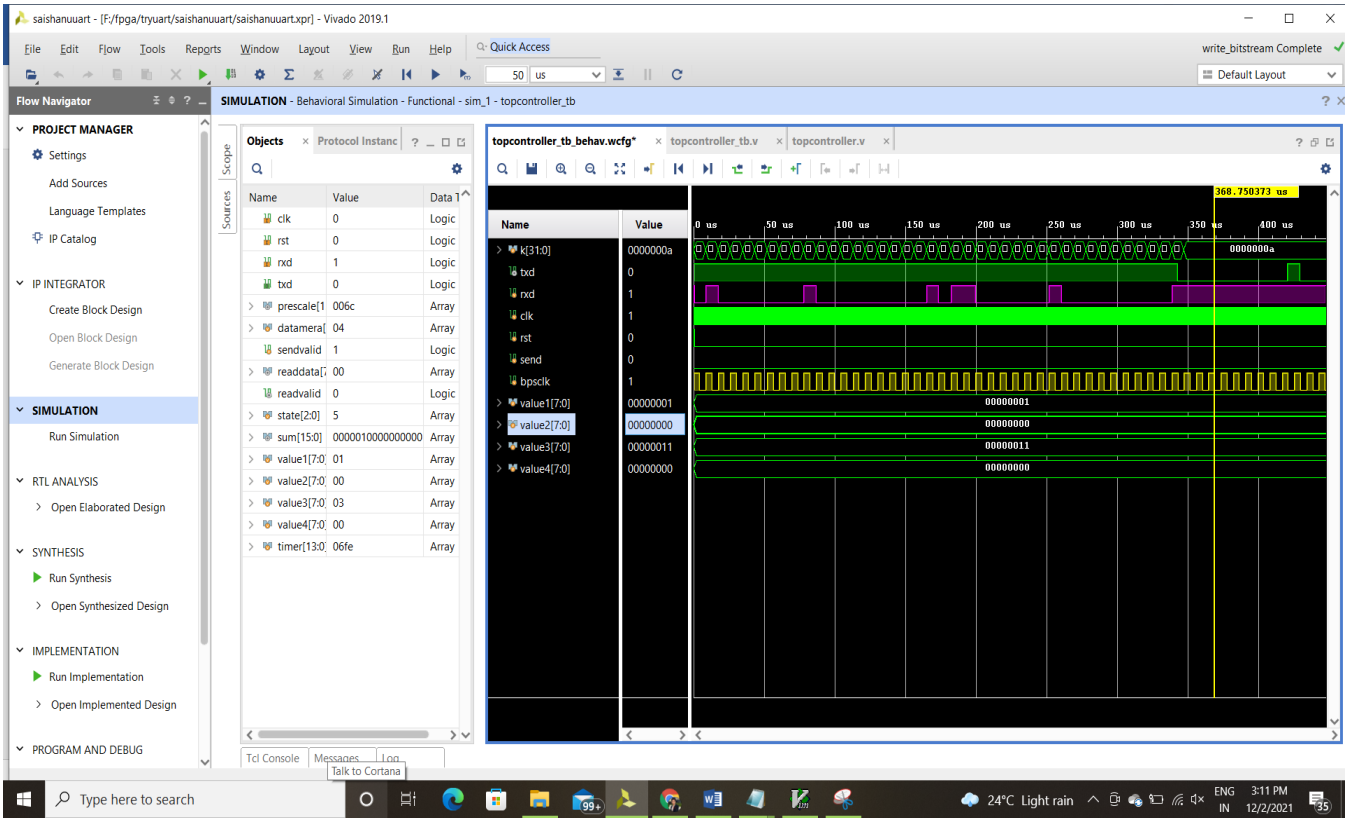
ASSIGNMENT 3(UART)

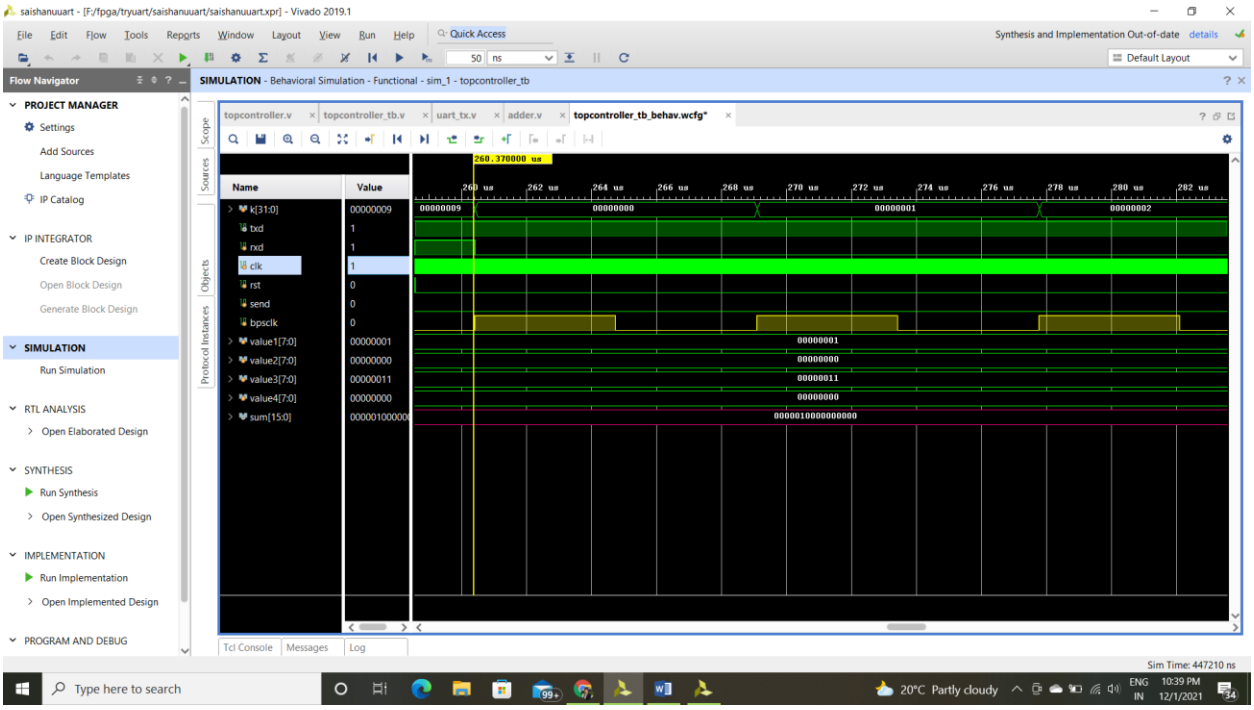
NAME: PRAFFUL CHOUDHARY

ROLL NUMBER: MT2021523

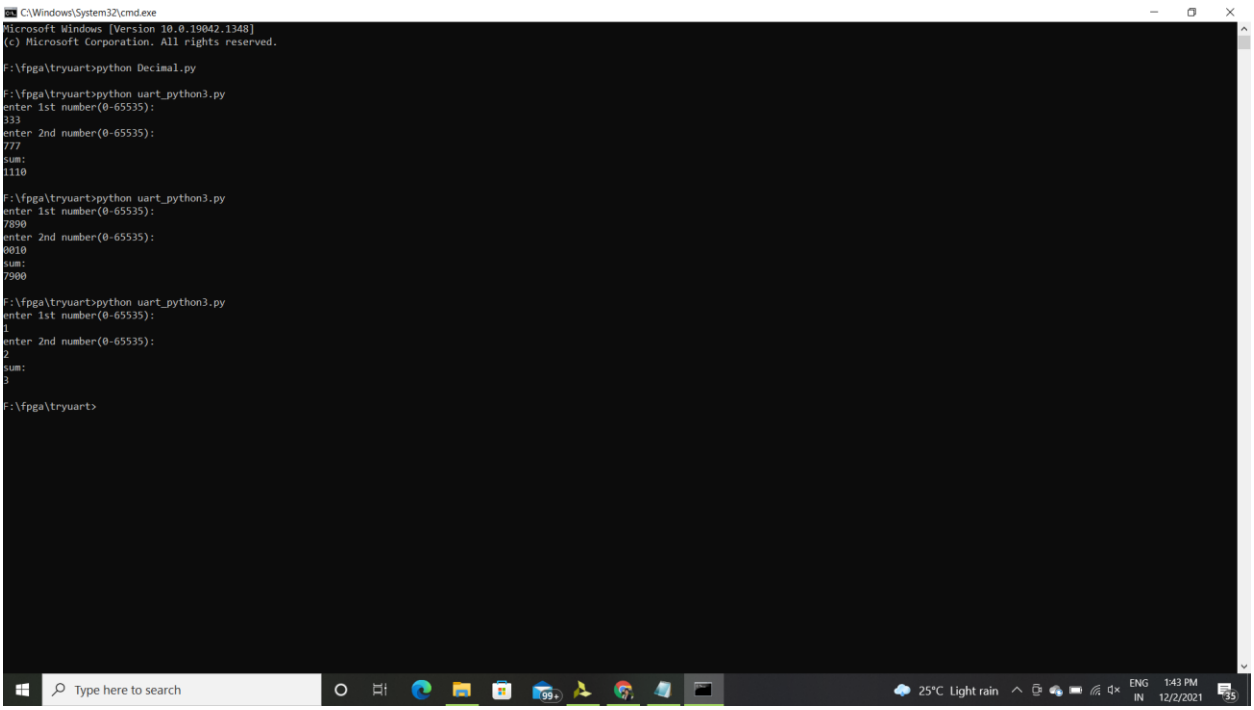
QUESTION: ADD TWO 16BIT NUMBER TOGETHER BY USING UART.

Simulation with the help of TestBench



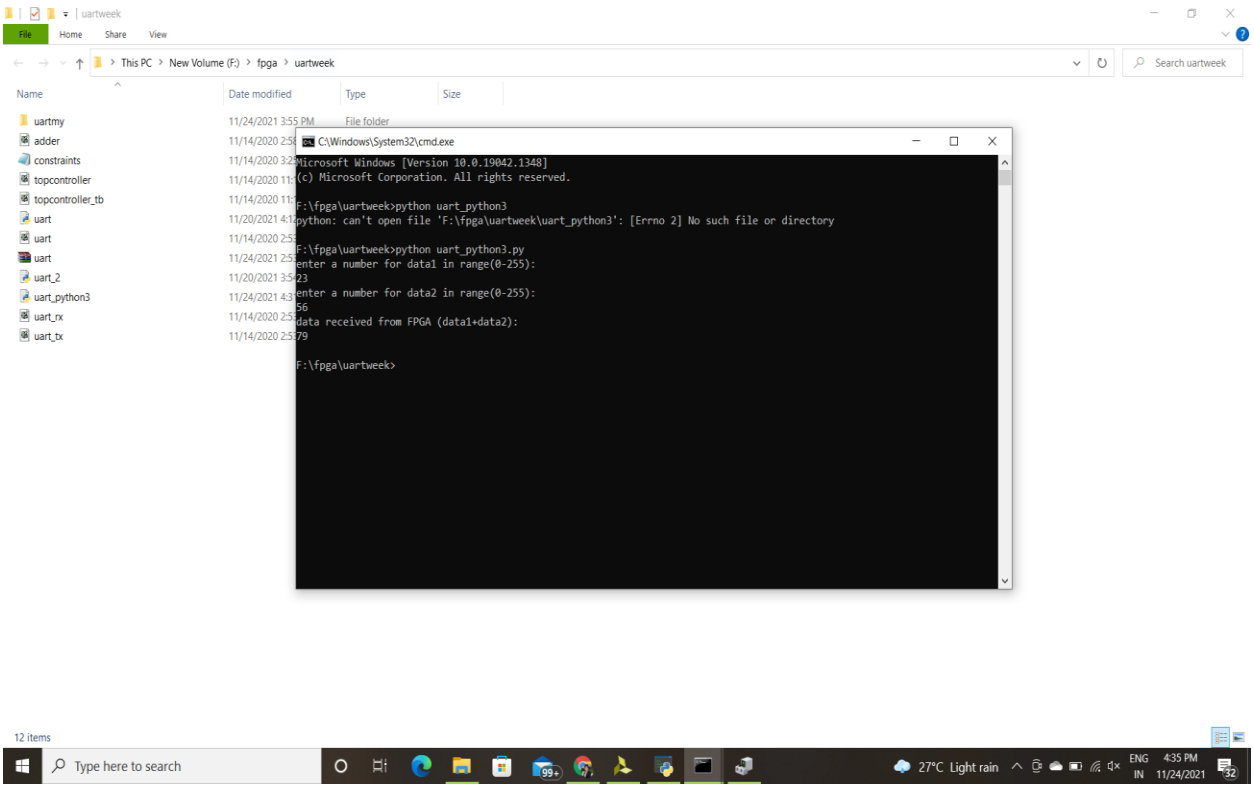


Command Prompt Output of 2 Byte Addition



OUTPUT OF HOMEWORK

Class Work (Addition of 1Byte)



THIS IS THE OUTPUT OF CLASS ASSIGNMENT FOR ONLY 8BIT ADDITION

APPENDIX 1.A

CODE(ADDER)

```
module adder(   input clk, input [7:0] value1,input [7:0] value2, input [7:0] value3, input [7:0] value4,
                output reg [15:0] dataout);
    always @(posedge clk) begin
        dataout = {value1,value2} + {value3,value4};
    end

endmodule
```

APPENDIX 1.B

CODE(TOP CONTROLLER)

```
module topcontroller( input clk, input rst, input rxd, output txd );
    reg[2:0]state;
    reg [2:0] S0,S1,S2,S3,S4,S5; // SIX STATES
    wire [15:0] prescale=100000000/(115200*8);
    reg [7:0] datamera;
    wire [7:0] readdata;
    reg sendvalid=0;
    wire readvalid;
    wire [15:0]sum;
    reg [7:0] value1=0,value2=0,value3=0,value4=0;
    reg [14:0]x=0;


adder adderuut (.clk(clk),
                .value1(value1),
                .value2(value2),
                .value3(value3),
                .value4(value4),
                .dataout(sum));

uart uartuut(
    .clk(clk),
    .rst(rst),
    .s_axis_tdata(datamera),
    .s_axis_tvalid(sendvalid),
    .m_axis_tdata(readdata),
    .m_axis_tvalid(readvalid),
    .m_axis_tready(1),
    .rxd(rxd),
    .txd(txd),
    .prescale(prescale)

);
```

```
parameter S0=3'b000,S1=3'b001,S2=3'b010,S3=3'b011,S4=3'b100,S5=3'b101;

always @(posedge clk) begin
    if(rst) begin state=0; sendvalid=0; value1=0; value2=0; value3=0; value4=0;
    end

 case(state)

    S0: begin sendvalid=0;if(readvalid) begin
        value1=readdata;
        state=S1;
        end
        end
    S1: begin if(readvalid) begin
        value2=readdata;
        state=S2;
        end
        end
     S2: begin if(readvalid) begin
        value3=readdata;
        state=S3;
        end
        end

     S3: begin if(readvalid) begin
         value4=readdata;
         state=S4;
         end
         end
     S4: begin sendvalid=1; datamera=sum[7:0]; x=x+1;
         if(x==8680) begin
         state=S5;
         x=0;
         end
         end

      S5: begin datamera=sum[15:8];x=x+1;
         if(x==8680) begin
         state=S0;
         x=0;
         end
         end

    default: begin state<=S0; x=0; end
    endcase

    end
    endmodule
```

APPENDIX 1.C TEST BENCH

```
`timescale 1ns / 1ps
module topcontroller_tb;
reg clk=1;
reg rst;
integer i;
integer k;
reg [7:0] sw=0;
reg send=0;
wire txd;
reg  rxd;
topcontroller top_instance(
    .clk(clk),
    .rst(rst),
    .rxd(rxd),
    .txd(txd) );
reg bpsclk=1;
always #(4340) bpsclk=!bpsclk;
always #(5) clk=!clk;
reg [7:0] value1=8'b00000001;
reg [7:0] value2=8'b00000000;
reg [7:0] value3=8'b00000011;
reg [7:0] value4=8'b00000000;
initial begin
    rst=1;
    #(10)rst=0;
    for(k=0;k<10;k=k+1) begin
        if(k==0)
            rxd=0;
        else if(k==9)
            rxd=1;
        else begin
            rxd=value1[k-1];
        end
        #(8680);
    end
     for(k=0;k<10;k=k+1) begin
        if(k==0)
            rxd=0;
        else if(k==9)
            rxd=1;
        else begin
            rxd=value2[k-1];
        end
```

```
        #(8680);
    end
    for(k=0;k<10;k=k+1) begin
        if(k==0)
            rxd=0;
        else if(k==9)
            rxd=1;
        else begin
            rxd=value3[k-1];
        end
        #(8680);
    end
    for(k=0;k<10;k=k+1) begin
        if(k==0)
            rxd=0;
        else if(k==9)
            rxd=1;
        else begin
            rxd=value4[k-1];
        end
        #(8680);
    end
    #(100000)
    $finish;
    end
endmodule
```

----------THANKYOU----------