Hardware Desing and Lab

Lab_5 individual Report

Name: 林奕為

Student's Id: 110062271

Department: CS

Contents

What I have learned from this Lab:

In this lab, I have learned about using FPGA board to produce sound and receive input from keyboard, making it a more versatile way of implementing a Verilog project.

Additionally, I have spend most of the time doing group project, so I only completed GCD module in this lab for the individual part.

Greatest Common Divisor:

How Does My Module Work:

My module consists of 2 parts, one being the *Greatest_Common_Divisor* which equals the top module in the design, is responsible for the state transition.

The other module is *Calculate_submodule*, which is the one actually implemented the calculation of the GCD function.

According to the specification, the state would be WAIT upon reset, after the start signal beign 1’b1, the state would enter CAL.

In the CAL state, it would set the en signal to 1’b1, which is passed into *Calculate_submodule*, and the module would start calculation. After completing the calculation, the result would be buffed into *gcd_result*,
and cal_finished would be 1’b1, and eventually enter FINISH state.

In the FINISH state, *done* would be set to 1’b1, and the *gcd* signal would be the *gcd_result,* which is the calculation result which is previously buffed in the CAL state.

After 2 cycle, the state would enter WAIT again, and enter next cycle.

Source Code:

```
`timescale 1ns/1ps

module Calculate_submodule(clk, en, a, b, gcd, done);
input clk, en;
input [15:0] a, b;
output reg done;
output reg [15:0] gcd;

reg [15:0] A, B;

always @(en) begin
    A = a;
    B = b;
end
```

```
always @(posedge clk) begin
    if(~en) done <= 1'b0;
    else begin
        if(~done) begin
            if(A == 0) begin
                gcd <= B;
                done <= 1'b1;
            end
            else begin
                if(B != 16'd0) begin
                    if(A > B) A <= A - B;
                    else B <= B - A;
                end
                else begin
                    gcd <= A;
                    done <= 1'b1;
                end
            end
        end
    end
end

endmodule

module Greatest_Common_Divisor (clk, rst_n, start, a, b, done, gcd);
input clk, rst_n;
input start;
input [15:0] a;
input [15:0] b;
output done;
output [15:0] gcd;

parameter WAIT = 2'b00;
parameter CAL = 2'b01;
parameter FINISH = 2'b10;

wire cal_finished;
wire [15:0] gcd_result;

reg en, done;
reg [15:0] A, B;
reg [1:0] state, next_state;
reg [15:0] gcd;

reg [1:0] counter;

Calculate_submodule c1(clk, en, A, B, gcd_result, cal_finished);

always @(posedge clk) begin
    if(~rst_n) begin
        state <= WAIT;
        en <= 1'b0;
        gcd <= 16'd0;
        done <= 1'b0;
    end
    else begin
        state <= next_state;
        if(state == WAIT) begin
            en <= 1'b0;
            gcd <= 16'd0;
            done <= 1'b0;
            counter <= 2'd0;
        end
        if(state == FINISH) begin
            counter <= counter + 2'd1;
            if(counter == 2'd2) begin
                state <= WAIT;
                en <= 1'b0;
                gcd <= 16'd0;
                done <= 2'd0;
            end
            else begin
                gcd <= gcd_result;
                done <= cal_finished;
            end
        end
    end
end

always @(*) begin
    if(state == WAIT) begin
        A = a;
        B = b;
    end
end

always @(*) begin
    case(state)
        WAIT: begin
            if(start == 1'b1) next_state = CAL;
            else next_state = WAIT;
        end

        CAL: begin
            en <= 1'b1;
            if(cal_finished == 1'b1) next_state = FINISH;
        end

    endcase

end

endmodule
```

How Do I Test My Design:

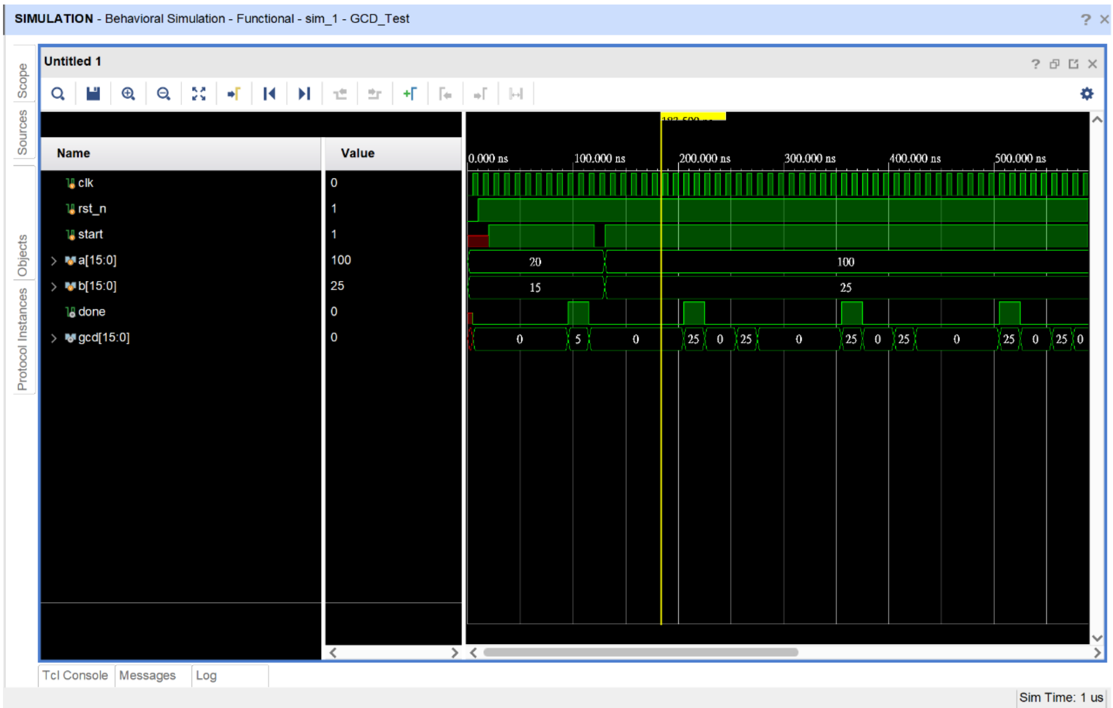
I use two testcases, one being (a, b) = (20, 15) and the other one is (a, b) = (25, 100). In the wave form result, the calculation result is correct correspondingly. (5, and 25).

TestBench source code:

```
`timescale 1ns/1ps

module GCD_Test;

reg clk, rst_n, start;
reg [15:0] a, b;
wire done;
wire [15:0] gcd;

Greatest_Common_Divisor g1(clk, rst_n, start, a, b, done, gcd);

always #5 clk = ~clk;

initial begin
    clk = 1'b0;
    rst_n = 1'b0;
    a = 16'd20;
    b = 16'd15;

    #10
    rst_n = 1'b1;

    #10
    start = 1'b1;

    #100;
    start = 1'b0;
    #10
    a = 16'd100;
    b = 16'd25;
    start = 1'b1;
    #200;
end
```

```
endmodule
```

Wave from result:



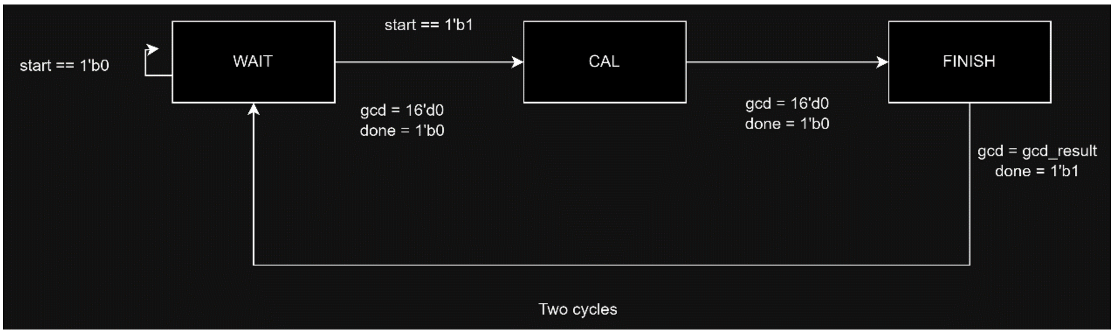
SIMULATION - Behavioral Simulation - Functional - sim_1 - GCD_Test

Untitled 1

Name	Value
clk	0
rst_n	1
start	1
a[15:0]	100
b[15:0]	25
done	0
gcd[15:0]	0

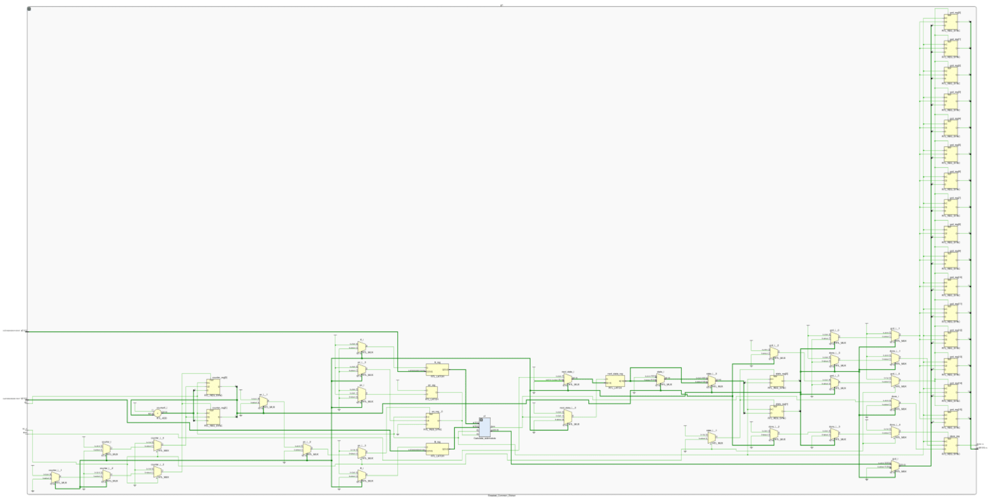
Sim Time: 1 us

State Diagram:



Block Diagram:

Greatest Common Divisor:



Calculation_submodule: