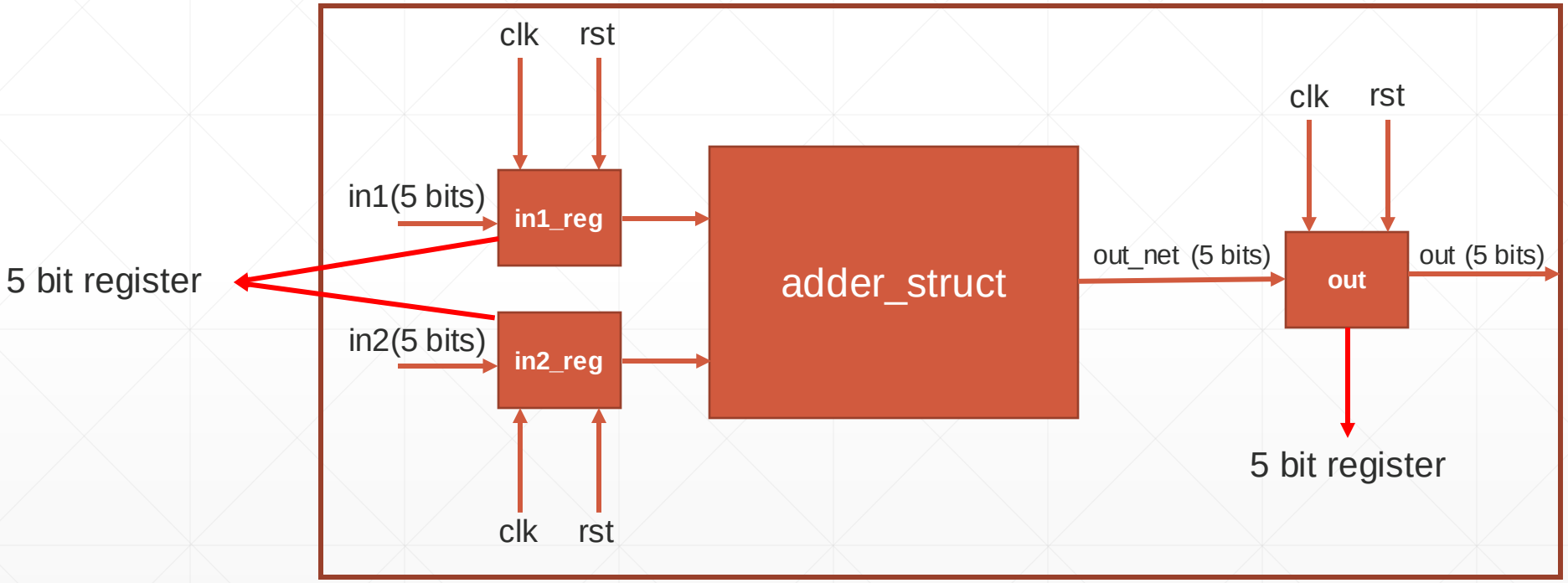
Timing Constraints Demo

Block Diagram



Description of the Verilog Modules

```
module adder_struct(input[4:0] in1,input[4:0] in2,output [4:0] out);
wire c_out1,c_out2,c_out3,c_out4;
half_addr addr1(.in1(in1[0]),.in2(in2[0]),.out(out[0]),.cout(c_out1));
full_addr addr2(.in1(in1[1]),.in2(in2[1]),.cin(c_out1),.out(out[1]),.cout(c_out2));
full_addr addr3(.in1(in1[2]),.in2(in2[2]),.cin(c_out2),.out(out[2]),.cout(c_out3));
full_addr addr4(.in1(in1[3]),.in2(in2[3]),.cin(c_out3),.out(out[3]),.cout(c_out4));
full_addr addr5(.in1(in1[4]),.in2(in2[4]),.cin(c_out4),.out(out[4]),.cout());
endmodule
```

Combinational 5 bit adder module

```
module wrapper(input clk, input rst, input [4:0] in1, input [4:0] in2, output reg [4:0] out);

reg [4:0] in1_reg;
reg [4:0] in2_reg;
wire [4:0] out_net;
always @(posedge clk)
    begin
        if(rst)
            begin
                in1_reg<=5'd0;
                in2_reg<=5'd0;
                out<=5'd0;
            end
        else
            begin
                in1_reg<=in1;
                in2_reg<=in2;
                out<=out_net;
            end
    end
```

Register initialization

```
adder_struct add(in1_reg,in2_reg,out_net);
```

adder_struct module instantiation

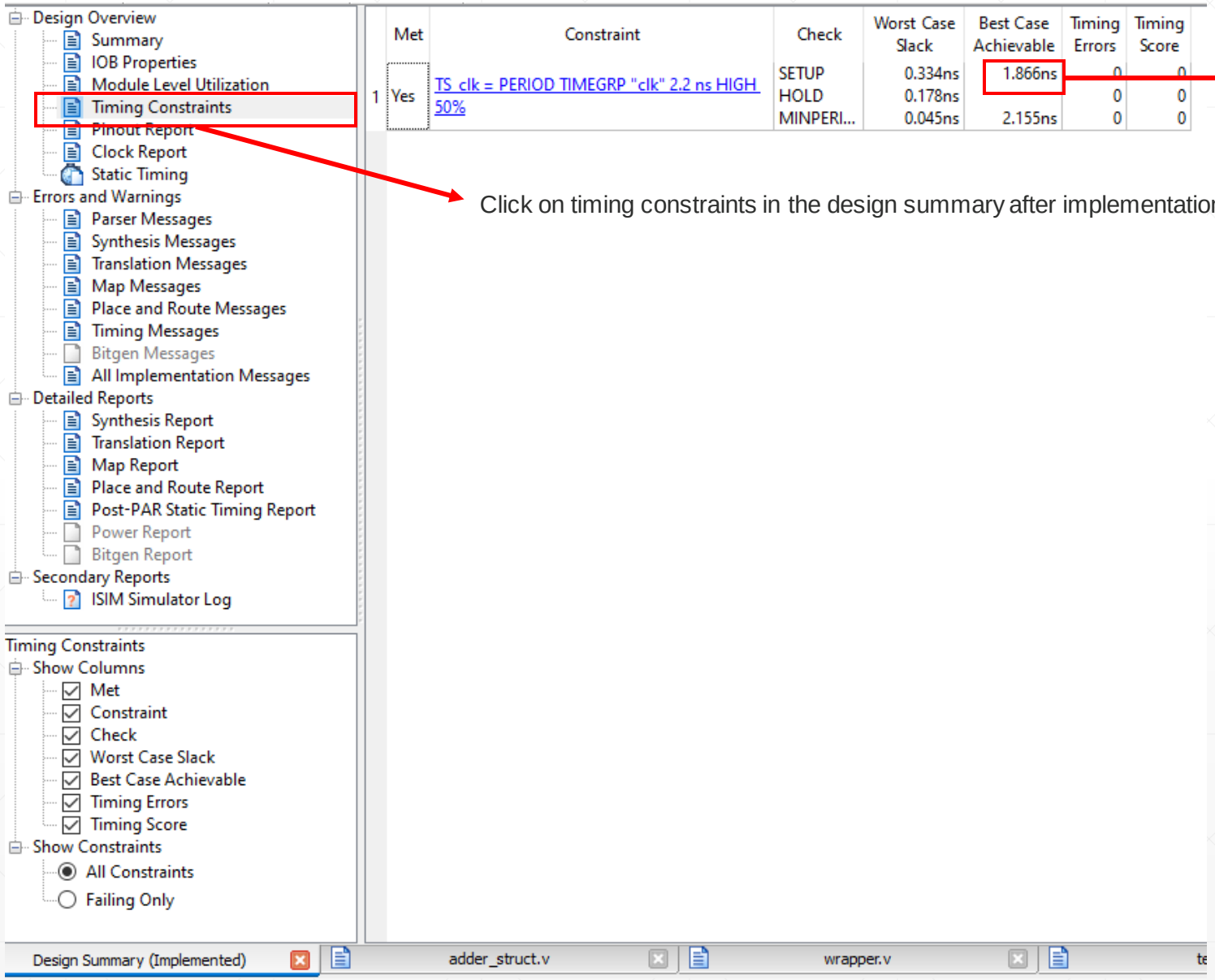
```
endmodule
```

Timing UCF

```
NET "clk" TNM_NET = clk;
TIMESPEC TS_clk = PERIOD "clk" 2.2 ns HIGH 50%;
```

Same clock port as defined in wrapper module

Timing Report



This is best achievable critical path (clock period) of the circuit

Click on timing constraints in the design summary after implementation