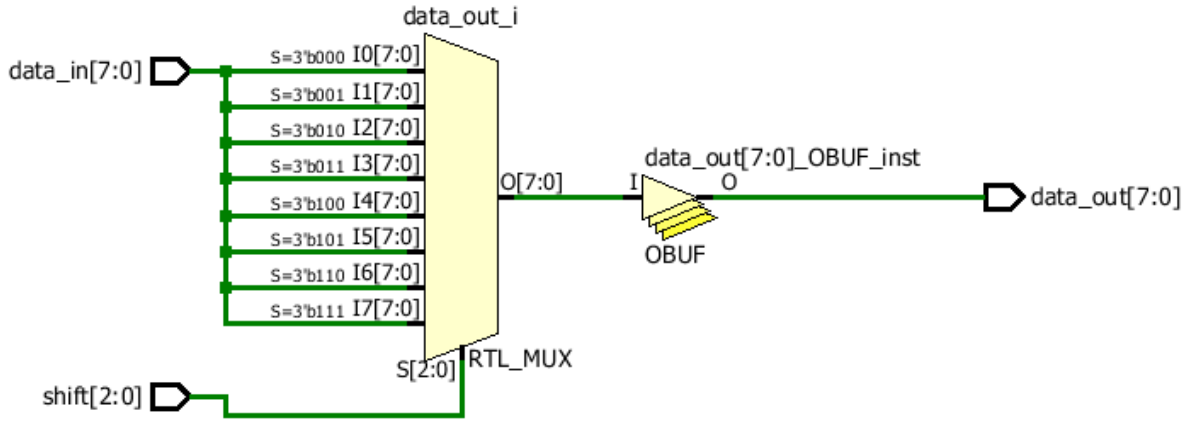
Q3) A barrel shifter is a combinational shifter that rotates an arbitrary number of bits. A barrel shifter is often used to shift and rotate n-bits in modern microprocessors, typically within a single clock cycle. Design a barrel shifter that takes in 8-bit data input and a 3 bit control signal, shift which specifies the amount to be rotated.

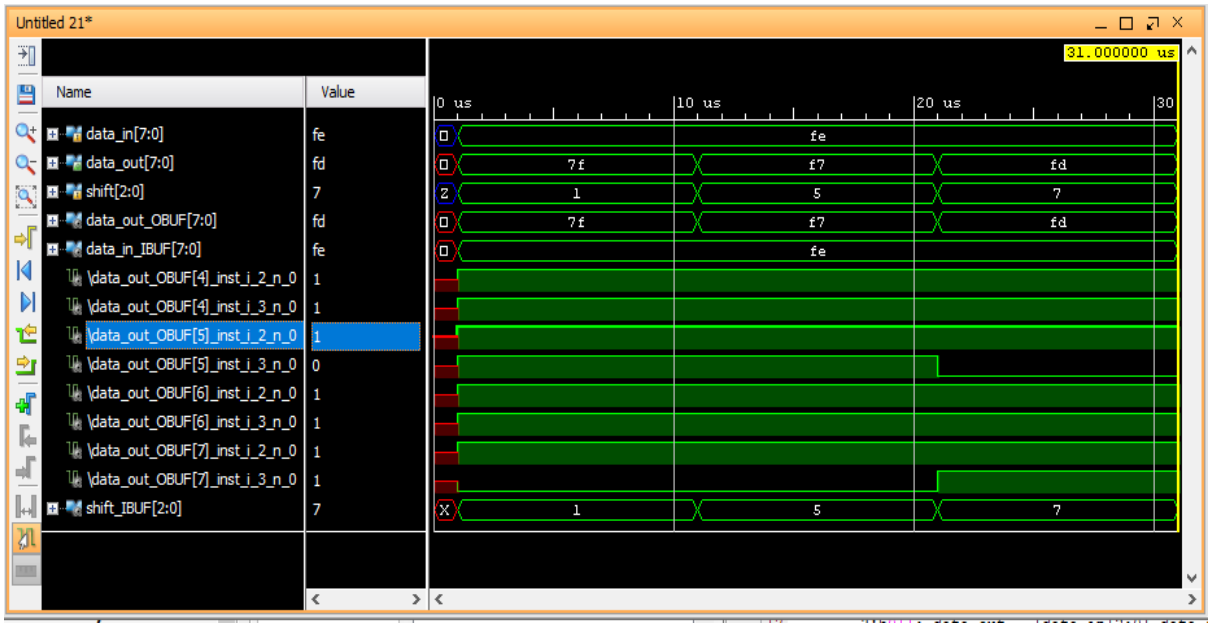
(a) Write a verilog description using case statement

Verilog Code:

```
`timescale 1ns / 1ps
//Q3a - Barrel Shifter using case statement.
//Barrel shifter is a combinational circuit that performs ROR operation.
module barrelshifter(
    output reg [7:0] data_out, //Data Output (8bits)
    input [7:0] data_in,       //Data Input (8bits)
    input [2:0] shift          //Shift control
    );

    always @(*) //Combinational Circuit
    begin
        case(shift)
/*Using case statement to infer a MUX that selects the corresponding bits
from input*/
        3'b000: data_out = data_in;                    //No rotate
        3'b001: data_out = {data_in[0],data_in[7:1]}; //Right Rotate by 1
        3'b010: data_out = {data_in[1:0],data_in[7:2]};//Right Rotate by 2
        3'b011: data_out = {data_in[2:0],data_in[7:3]};//Right Rotate by 3
        3'b100: data_out = {data_in[3:0],data_in[7:4]};//Right Rotate by 4
        3'b101: data_out = {data_in[4:0],data_in[7:5]};//Right Rotate by 5
        3'b110: data_out = {data_in[5:0],data_in[7:6]};//Right Rotate by 6
        3'b111: data_out = {data_in[6:0],data_in[7]}; //Right Rotate by 7
        endcase
    end
endmodule
```





Name	Slice LUTs (63400)	Slice (15850)	LUT as Logic (63400)	LUT Flip Flop Pairs (63400)	Bonded IOB (210)
barrelshifter	12	3	12	12	19

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	inf	Worst Hold Slack (WHS):	inf	Worst Pulse Width Slack (WPWS):	NA
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	NA
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	NA
Total Number of Endpoints:	8	Total Number of Endpoints:	8	Total Number of Endpoints:	NA

There are no user specified timing constraints.

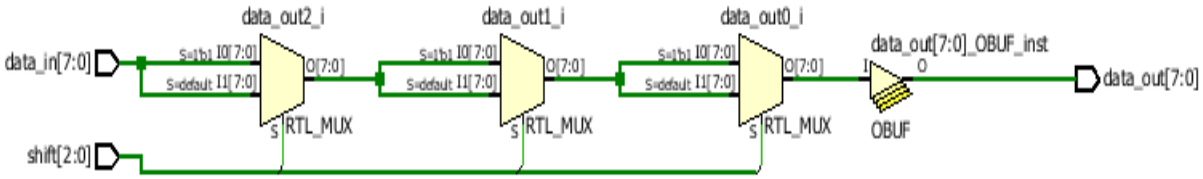
(b) Alternatively, you may construct a barrel shifter circuit by stages. In the n-th stage, the input signal is either passed directly to output or rotated by 2n positions. The n-th stage is controlled by the n-th bit of the shift signal. Write a verilog code using this logic.

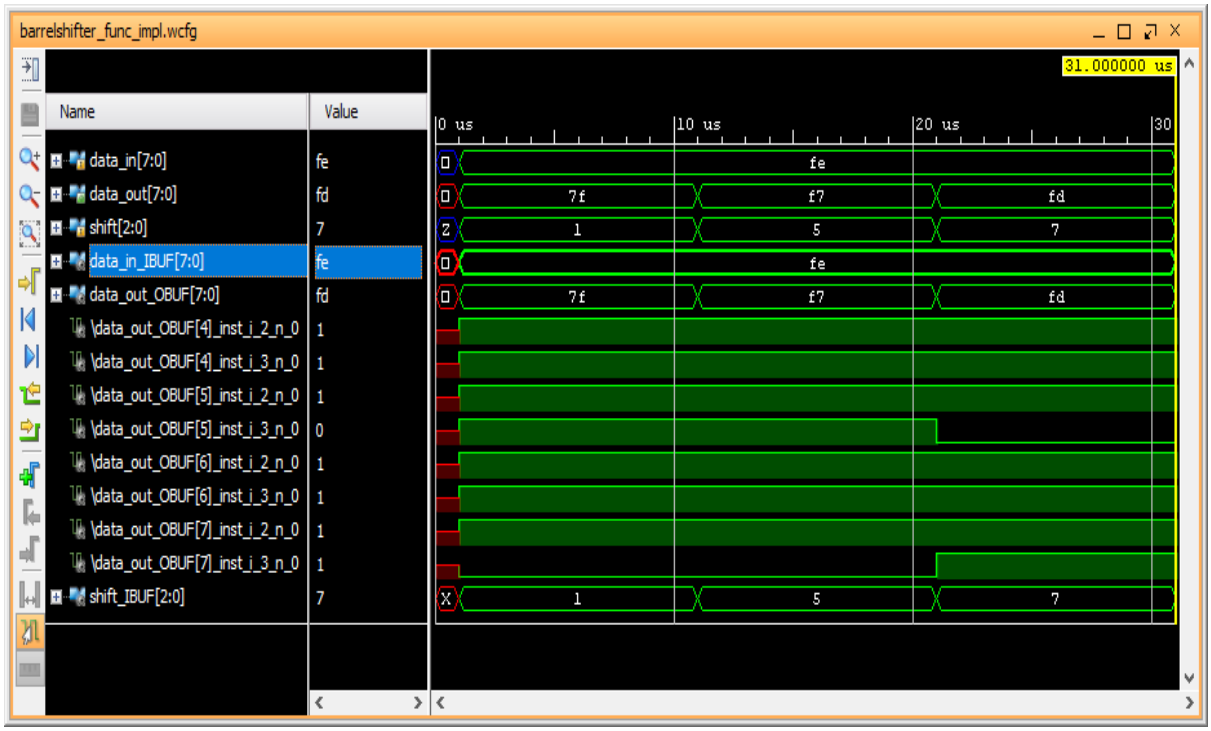
Verilog Code:

```
`timescale 1ns / 1ps
//Q3b Barrel Shifter using Stage wise Rotation
//It rotates in stages, wherein Nth stage right rotates by 2^N.
module barrelshifter(
    output reg [7:0] data_out,  //Data Output (8bits)
    input [7:0] data_in,    //Data Input (8bits)
    input [2:0] shift      //Shift control
    );
    always @(*)
            begin
            data_out = shift[0] ? {data_in[0],data_in[7:1]} : data_in;
          //Loads a right rotated by 1  version of input if LSB of shift is
1, else loads in the input directly

            data_out = shift[1] ? {data_out[1:0],data_out[7:2]}: data_out;
          //Right rotates by 2 if middle bit of shift is 1, else retains
the previous value
```

```
            data_out = shift[2] ? {data_out[3:0],data_out[7:4]} : data_out;
          //Right rotates by 4 if MSB of shift is 1, else retains previous
output
            end
    endmodule
```





Name	Slice LUTs (63400)	Slice (15850)	LUT as Logic (63400)	LUT Flip Flop Pairs (63400)	Bonded IOB (210)
barrelshifter	12	3	12	12	19

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	inf	Worst Hold Slack (WHS):	inf	Worst Pulse Width Slack (WPWS):	NA
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	NA
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	NA
Total Number of Endpoints:	8	Total Number of Endpoints:	8	Total Number of Endpoints:	NA

There are no user specified timing constraints.

(c) List applications of barrel shifter

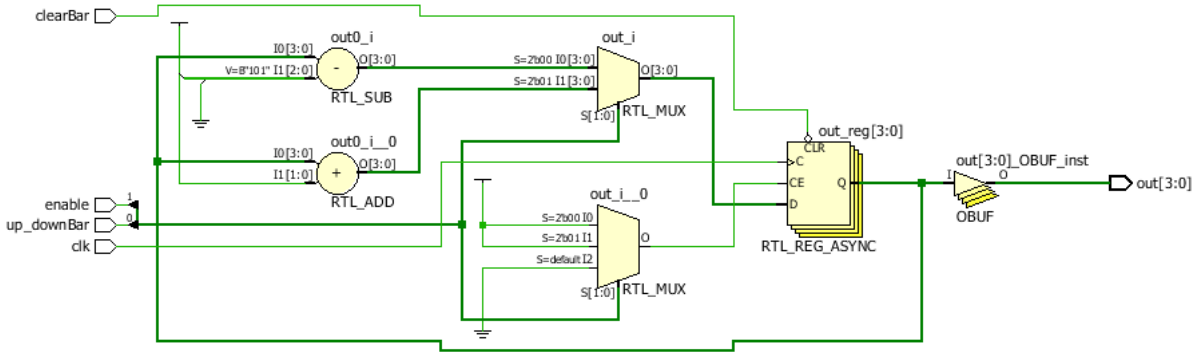
Barrel Shifter is used in:

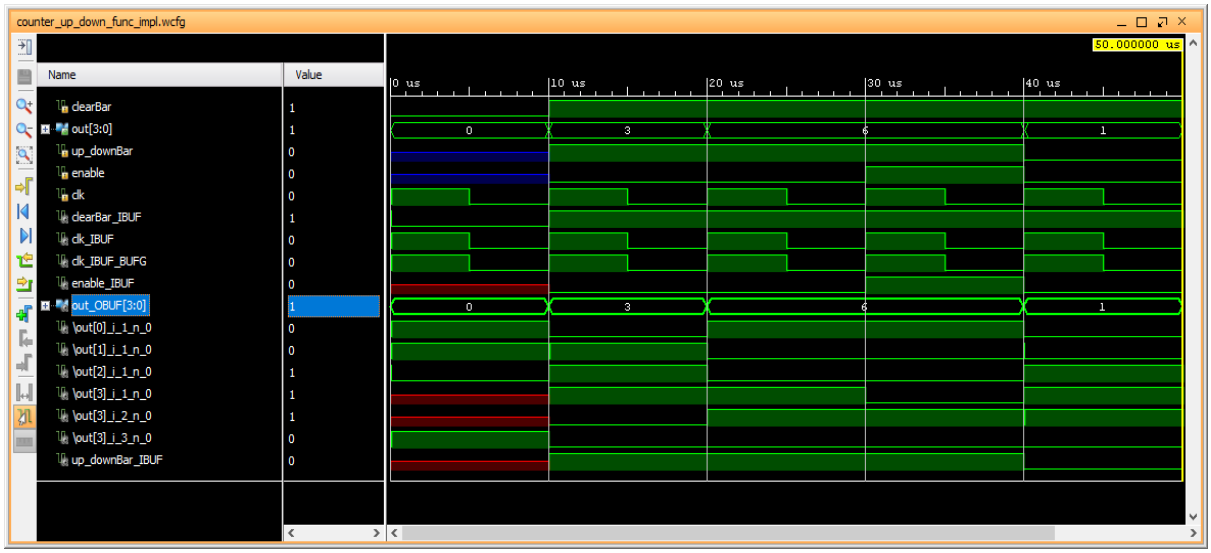
1. Barrel shifter is used extensively in RISC architecture's ALU datapath to shift numbes. The

Q5

Verilog Code:

```
`timescale 1ns / 1ps
// Q5 Up Down Counter
// Counts up by 3 when enable=0 and up_downBar=1
// Counts down by 5 when enable=0 and up_downBar=0
// Count value is retained when enable=1
// clearbar is fed as asynchronous clear
module counter_up_down(
    output reg [3:0] out,                  //Output of the counter
    input enable, clearBar, up_downBar, clk
    );
    always @(posedge clk, negedge clearBar)
    begin
    if(clearBar == 0) out<= 4'b0000;            //To clear the output
    else begin
       case({enable,up_downBar})
       2'b00: out <= out-5;    //Down Count
       2'b01: out <= out+3;    //Up Count
       default: out <= out;    //Retain value
       endcase
       end
    end
endmodule
```





Name	Slice LUTs (63400)	Slice Registers (126800)	Slice (15850)	LUT as Logic (63400)	LUT Flip Flop Pairs (63400)	Bonded IOB (210)	BUFGCTRL (32)
counter_up_down	4	4	1	4	4	8	1

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	inf	Worst Hold Slack (WHS):	inf	Worst Pulse Width Slack (WPWS):	NA
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	NA
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	NA
Total Number of Endpoints:	16	Total Number of Endpoints:	16	Total Number of Endpoints:	NA

There are no user specified timing constraints.

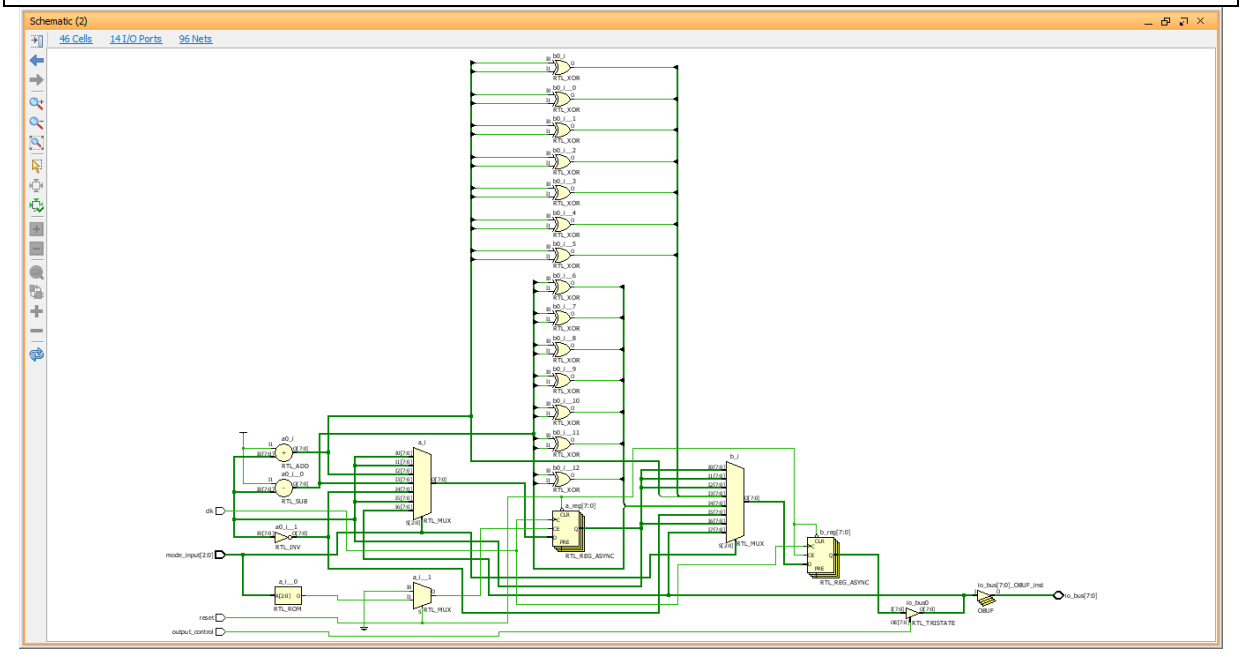
Q7

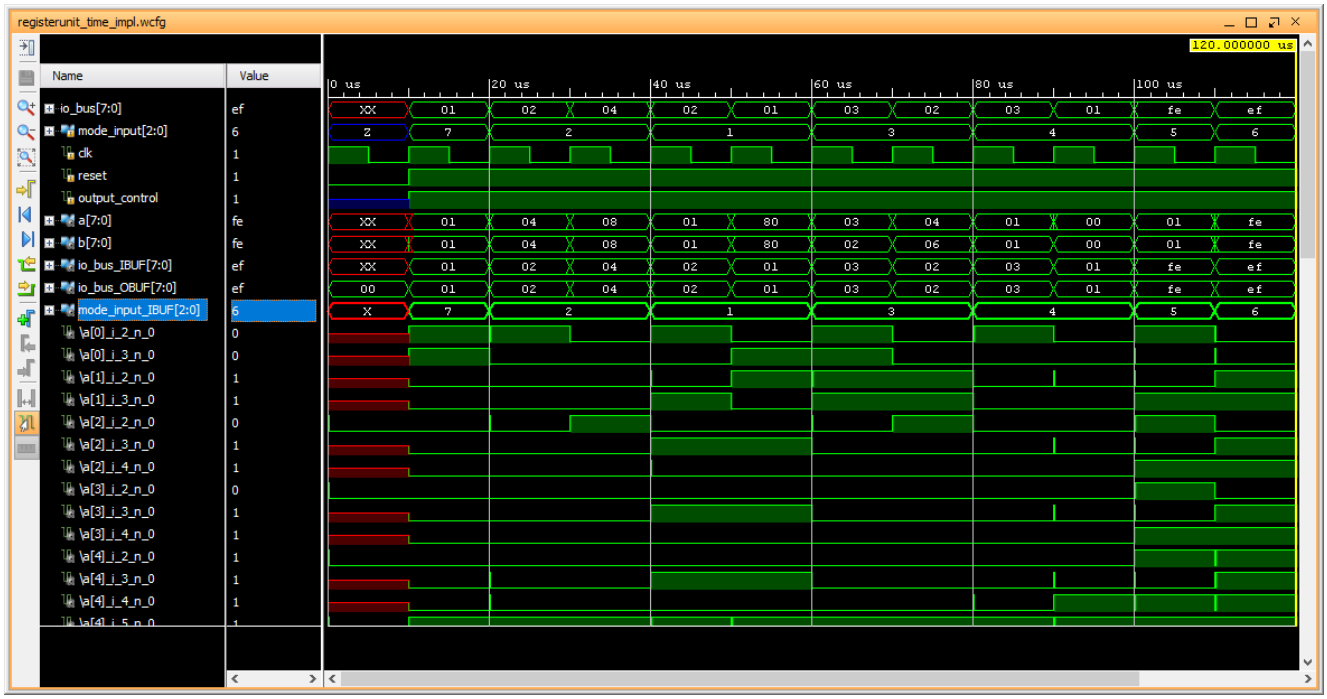
Verilog Code:

```
`timescale 1ns / 1ps
//Q7 Register Unit. Performs operations on bidirectional 8-bit bus.
module registerunit(
    inout [7:0] io_bus, //Bidirectional bus
    input [2:0] mode_input, //Mode Input
    input clk,
    input reset,        //Active low asynchronus reset
    input output_control //1 bit output control
    );

    reg[7:0] a,b; //Internal registers used for
    assign io_bus = output_control?b:8'bZZZZZZZZ; //assigns the output
depending on output_control value
    always @(posedge clk, negedge reset)
    begin
    if(reset==0) //used to reset a and b
        begin
        a <= 8'b00000000;
        b <= 8'b00000000;
        end
    else
        case(mode_input)
        3'b000: b = a; //Do nothing. Holding the previous value
```

```
        3'b001: begin //Right shift by 1 value
                a = {a[0],a[7:1]};
                b = a;
                end
        3'b010: begin //Left shift by 1 value
                a = {a[6:0],a[7]};
                b = a;
                end
        3'b011:  //8 bit Gray code upcount by 1
                begin
                a = a+1;
                b =
{a[7],a[7]^a[6],a[6]^a[5],a[5]^a[4],a[4]^a[3],a[3]^a[2],a[2]^a[1],a[1]^a[0]
};
                end
        3'b100: //8 bit Gray code downcount by 1
                begin
                a = a-1;
                b =
{a[7],a[7]^a[6],a[6]^a[5],a[5]^a[4],a[4]^a[3],a[3]^a[2],a[2]^a[1],a[1]^a[0]
};
                end
        3'b101: begin //Bitwise NOT the register contents
                a = ~a;
                b = a;
                end
        3'b110: begin // Swapping right and left 4 bits
                a = {a[3:0],a[7:4]};
                b = a;
                end
        3'b111: begin //parallel load to register
                a = io_bus;
                b = a;
                end
        endcase
    end
endmodule
```





Name	Slice LUTs (63400)	Slice Registers (126800)	F7 Muxes (31700)	Slice (15850)	LUT as Logic (63400)	LUT Flip Flop Pairs (63400)	Bonded IOB (210)	BUFGCTRL (32)
registerunit	49	16	7	16	49	49	14	1

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	inf	Worst Hold Slack (WHS):	inf	Worst Pulse Width Slack (WPWS):	NA
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	NA
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	NA
Total Number of Endpoints:	56	Total Number of Endpoints:	56	Total Number of Endpoints:	NA

There are no user specified timing constraints.