Experiment 3

Aim: Write Verilog codes for all given condition having structural, data flow and behavioural model and perform synthesis by generating different synthesis and timing waveforms.

a) 2 x 1 mux
b) 4 x 1 mux using 2 x 1 mux
c) 16 x 1 mux using 4 x 1 mux

Theory:

- A Multiplexer (mux) is a combinational logic circuit that selects one of the multiple input data lines and routes it to a single output line based on the select signal.
- The 2x1 mux has two data inputs (a and b), one select input (sel), and one output (y). It selects either a or b based on the value of sel.
- The 4x1 mux uses four 2x1 muxes to select one of the four data inputs (a, b, c, d) based on the 2-bit select input (sel).
- The 16x1 mux uses four 4x1 muxes to select one of the sixteen data inputs (data) based on the 4-bit select input (sel).
- In the structural models, lower-level muxes (2x1 and 4x1) are instantiated to build higher-level muxes (4x1 and 16x1).

Code for 2x1 mux:

1) Structural

```
module mux_2x1_s(
   input I0, I1, s,
   output out
   );
   wire w1, w2, w3;
   not(w1, s);
   and(w2,I0,s);
   and(w3,I1,w1);
   or(out,w2, w3);
endmodule
```

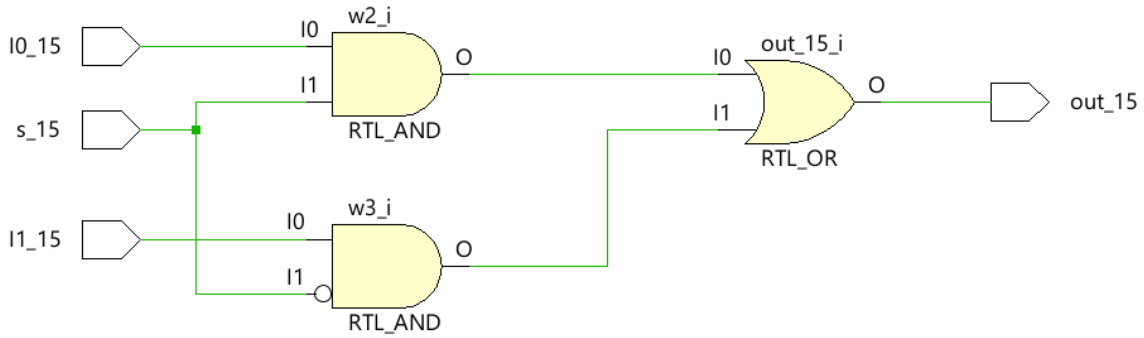
2) Data Flow

```
module mux_2x1_d(
    input I0, I1, s,
    output out
    );
    wire w1, w2, w3;
    assign w1 = ~(s);
    assign w2 = I0 & s;
    assign w3 = I1 & w1;
    assign out = w2 | w3;
endmodule
```

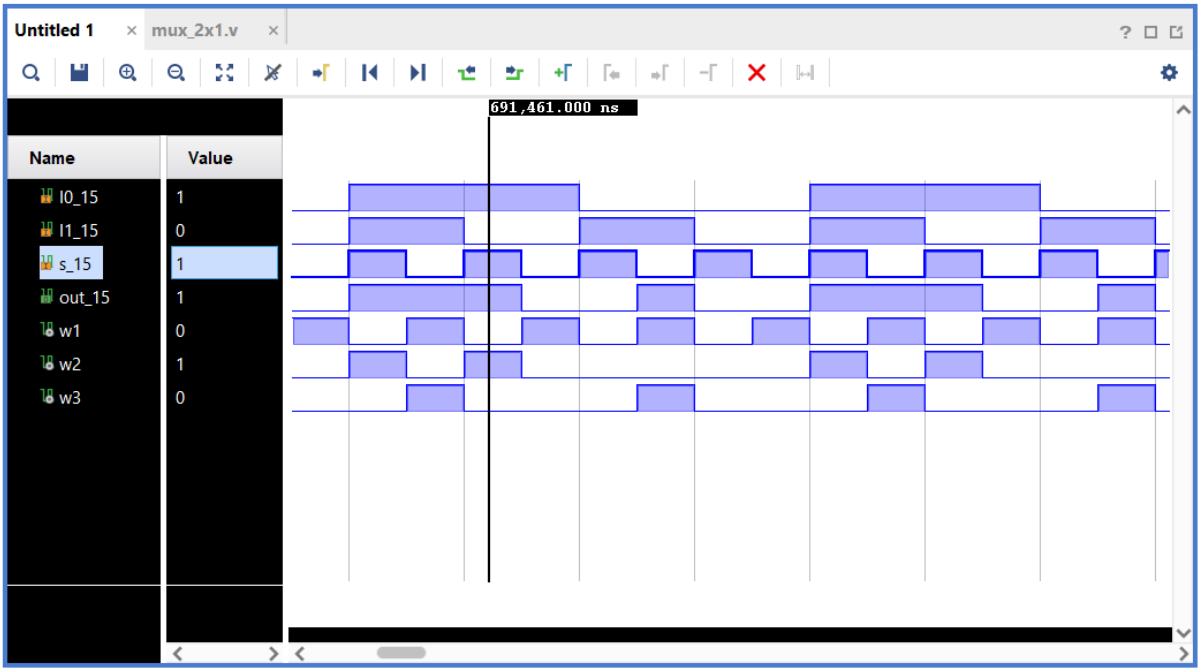
3) Behavioural

```
module mux_2x1_b(
    input I0, I1, s,
    output out
    );
    always @(I0, I1, s)
    begin
        if (s == 1'b0)
            out = I0;
        else
            out = I1;
    end
endmodule
```

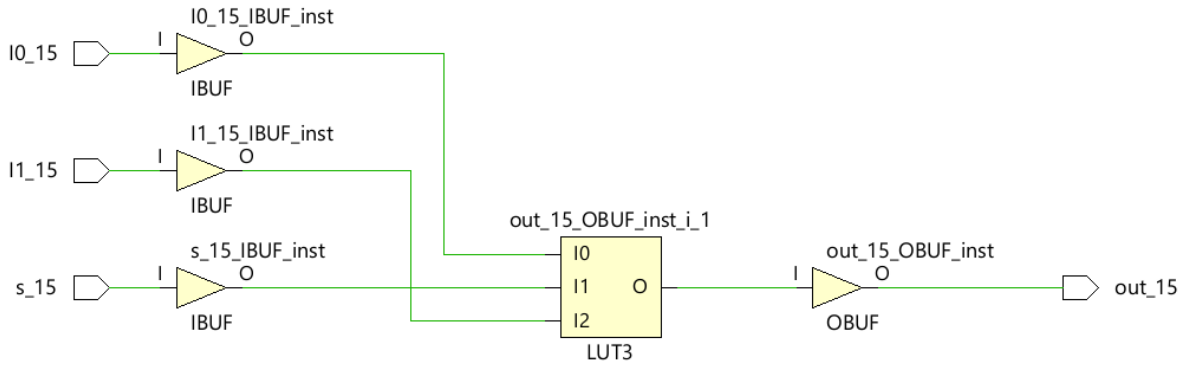
RTL Schematic:



Function Verification:



Post synthesis schematic:



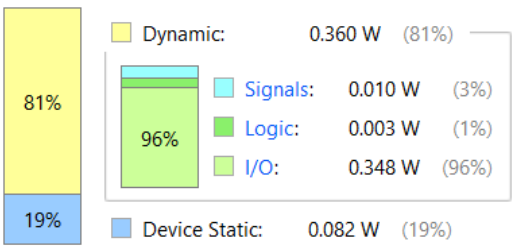
Post power report:

Summary

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**0.442 W**
Design Power Budget:	**Not Specified**
Process:	typical
Power Budget Margin:	**N/A**
Junction Temperature:	**25.8°C**
Thermal Margin:	59.2°C (31.2 W)
Ambient Temperature:	25.0 °C
Effective ϑJA:	1.9°C/W
Power supplied to off-chip devices:	0 W

On-Chip Power



Post synthesis timing summary:

Design Runs | Timing | Power

Unconstrained Paths - NONE - NONE - Setup

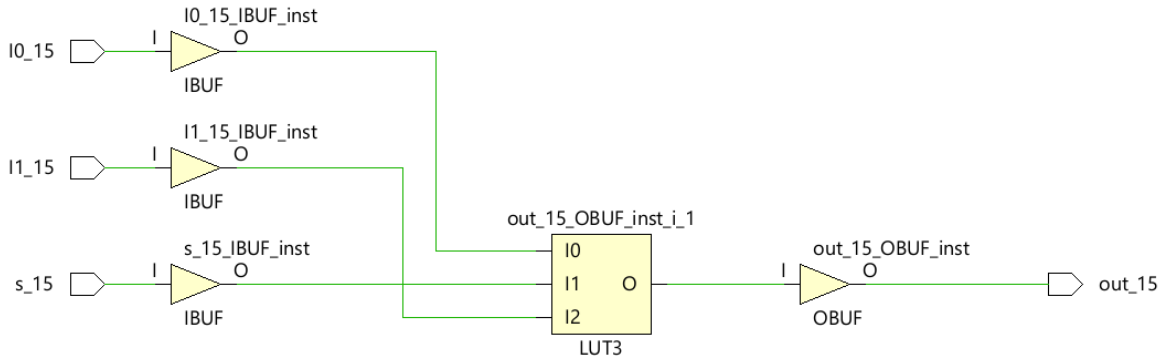
Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock
Path 1	∞	3	4	1	I1_15	out_15	4.512	3.345	1.167	∞	input port clock

Post utilization/ area summary:

Hierarchy

Name	Slice LUTs (41000)	Bonded IOB (300)
mux_2x1_s	1	4

Post implementation schematic:



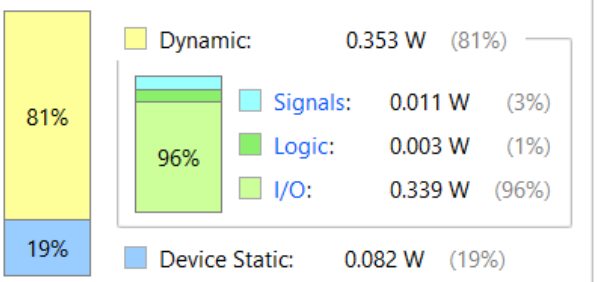
Post implementation power report:

Summary

Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power:	**0.435 W**
Design Power Budget:	**Not Specified**
Process:	typical
Power Budget Margin:	**N/A**
Junction Temperature:	**25.8°C**
Thermal Margin:	59.2°C (31.2 W)
Ambient Temperature:	25.0 °C
Effective ϑJA:	1.9°C/W
Power supplied to off-chip devices:	0 W

On-Chip Power



Post implementation timing summary:

Unconstrained Paths - NONE - NONE - Setup

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock
Path 1	∞	3	2	1	I0_15	out_15	5.615	3.274	2.341	∞	input port clock

Post implementation utilization/ area summary:

Hierarchy

Name	Slice LUTs (41000)	Slice (10250)	LUT as Logic (41000)	Bonded IOB (300)
mux_2x1_s	1	1	1	4

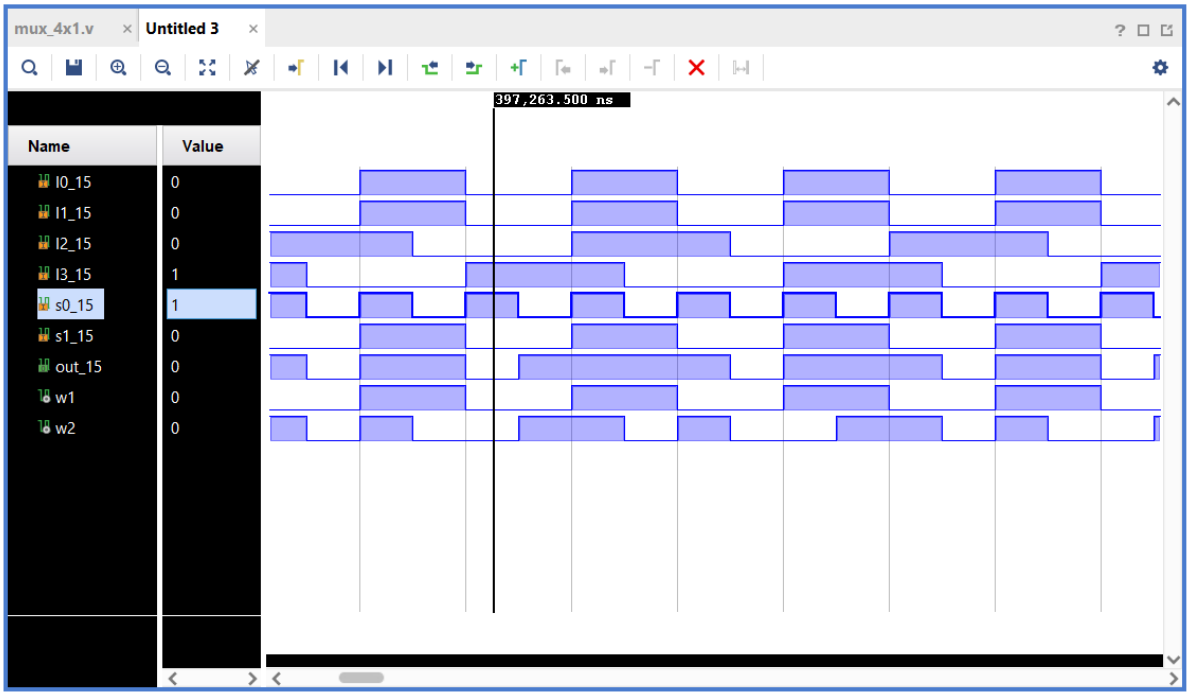
Code for 4x1 mux using 2x1:

Structural

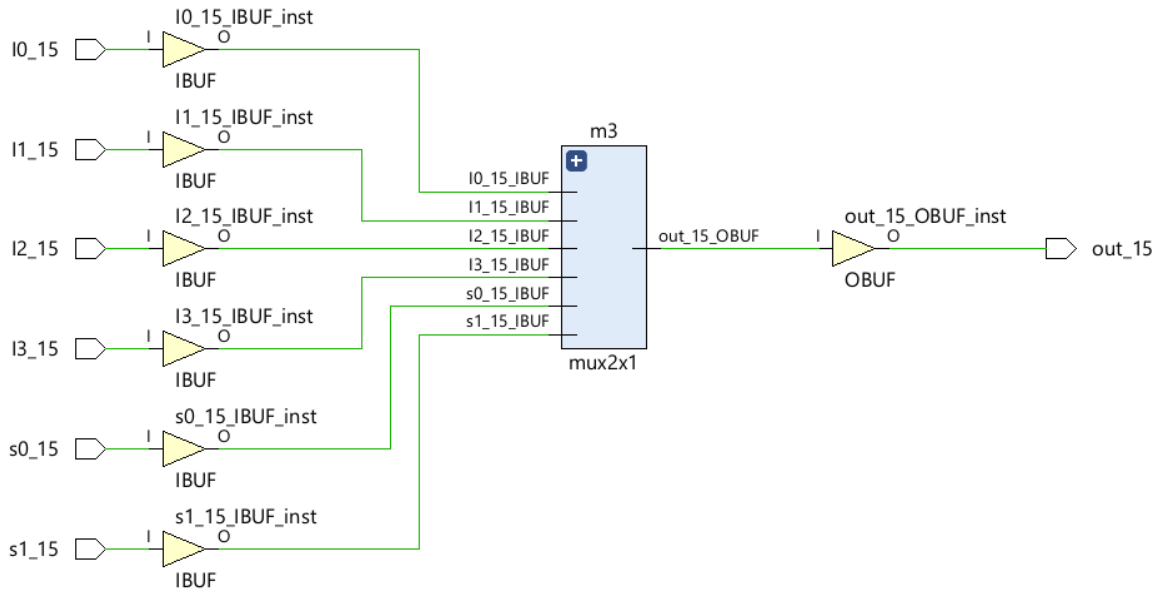
```
module mux2x1(
   input I0, I1, s,
   output out
   );
   wire w1, w2, w3;
   not(w1, s);
   and(w2,I0,s);
   and(w3,I1,w1);
   or(out,w2, w3);
endmodule

module mux_4x1(
   input I0, I1,I2,I3, s0,s1,
   output out
   );
   wire w1,w2;
   mux2x1 m1(I0, I1,s0,w1);
   mux2x1 m2(I2, I3,s0,w2);
   mux2x1 m3(w1, w2,s1,out);
endmodule
```

RTL Schematic:

Function Verification:



Post synthesis schematic:

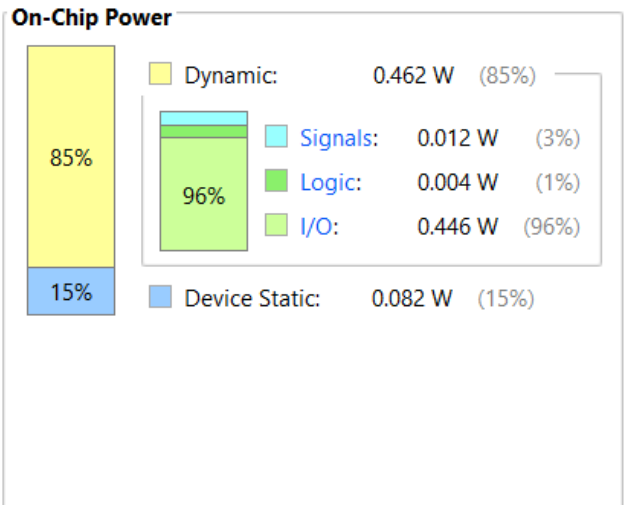


Post power report:

Summary

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**0.544 W**
Design Power Budget:	**Not Specified**
Process:	typical
Power Budget Margin:	**N/A**
Junction Temperature:	**26.0°C**
Thermal Margin:	59.0°C (31.1 W)
Ambient Temperature:	25.0 °C
Effective ϑJA:	1.9°C/W



Post synthesis timing summary:

Unconstrained Paths - NONE - NONE - Setup

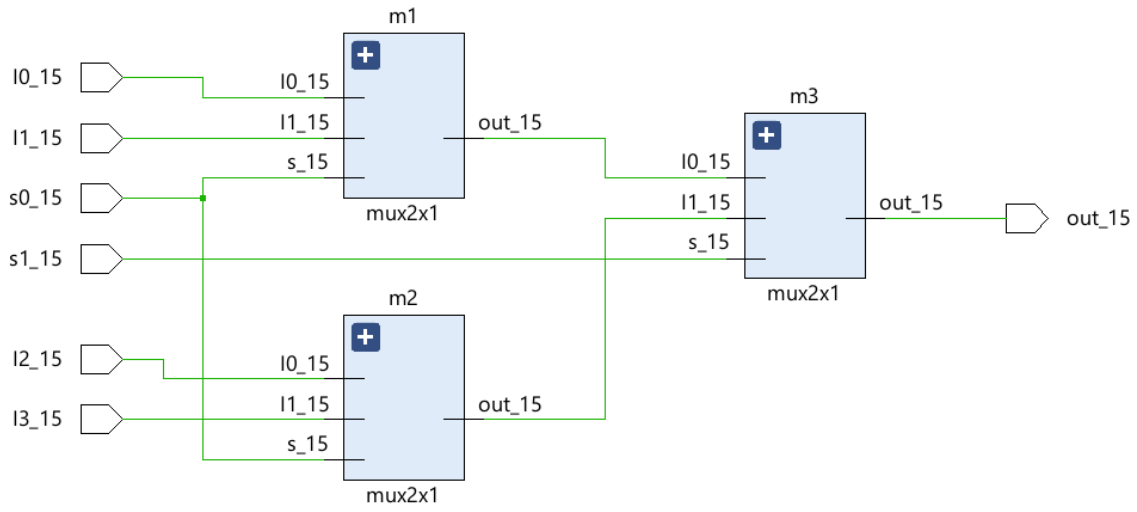
Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock
Path 1	∞	3	4	1	I3_15	out_15	4.512	3.345	1.167	∞	input port clock

Post utilization/ area summary:

Hierarchy

Name	Slice LUTs (41000)	Bonded IOB (300)
mux_4x1	1	7
m3 (mux2x1)	1	0

Post implementation schematic:

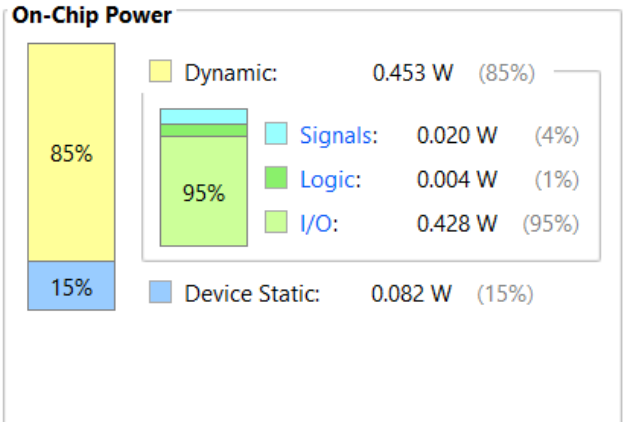


Post implementation power report:

Summary

Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power:	**0.535 W**
Design Power Budget:	**Not Specified**
Process:	typical
Power Budget Margin:	**N/A**
Junction Temperature:	**26.0°C**
Thermal Margin:	59.0°C (31.1 W)
Ambient Temperature:	25.0 °C



Post implementation timing summary:

Unconstrained Paths - NONE - NONE - Setup

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock
Path 1	∞	3	2	1	I0_15	out_15	5.696	3.315	2.381	∞	input port clock

Post implementation utilization/ area summary:

Hierarchy

Name	Slice LUTs (41000)	Slice (10250)	LUT as Logic (41000)	Bonded IOB (300)
mux_4x1	1	1	1	7
m3 (mux2x1)	1	1	1	0

Code for 16x1 mux using 4x1 mux:

Structural

```
module m_2x1(
   input I0, I1,
   input sel,
   output out
   );
   assign out = (sel) ? I1 : I0;
endmodule

module m_4x1(
   input [3:0] I0, I1, I2, I3,
   input [1:0] sel,
   output out
```

```
    );
    wire w0, w1, w2;
    m_2x1 mux1(I0, I1, sel[0], w0);
    m_2x1 mux2(I2, I3, sel[0], w1);
    m_2x1 mux3(w0, w1, sel[1], w2);
    assign out = w2;
endmodule

module mux_16x1(data, sel, y);
     input [15:0] data;
     input [3:0] sel;
     output y;
     wire [0:3] ma;
     m_4x1 mux1(data[3:0],sel[1:0],ma[0]);
     m_4x1 mux2(data[7:4],sel[1:0],ma[1]);
     m_4x1 mux3(data[11:8],sel[1:0],ma[2]);
     m_4x1 mux4(data[15:12],sel[1:0],ma[3]);
     m_4x1 mux5(ma,sel[3:2],y);
endmodule
```

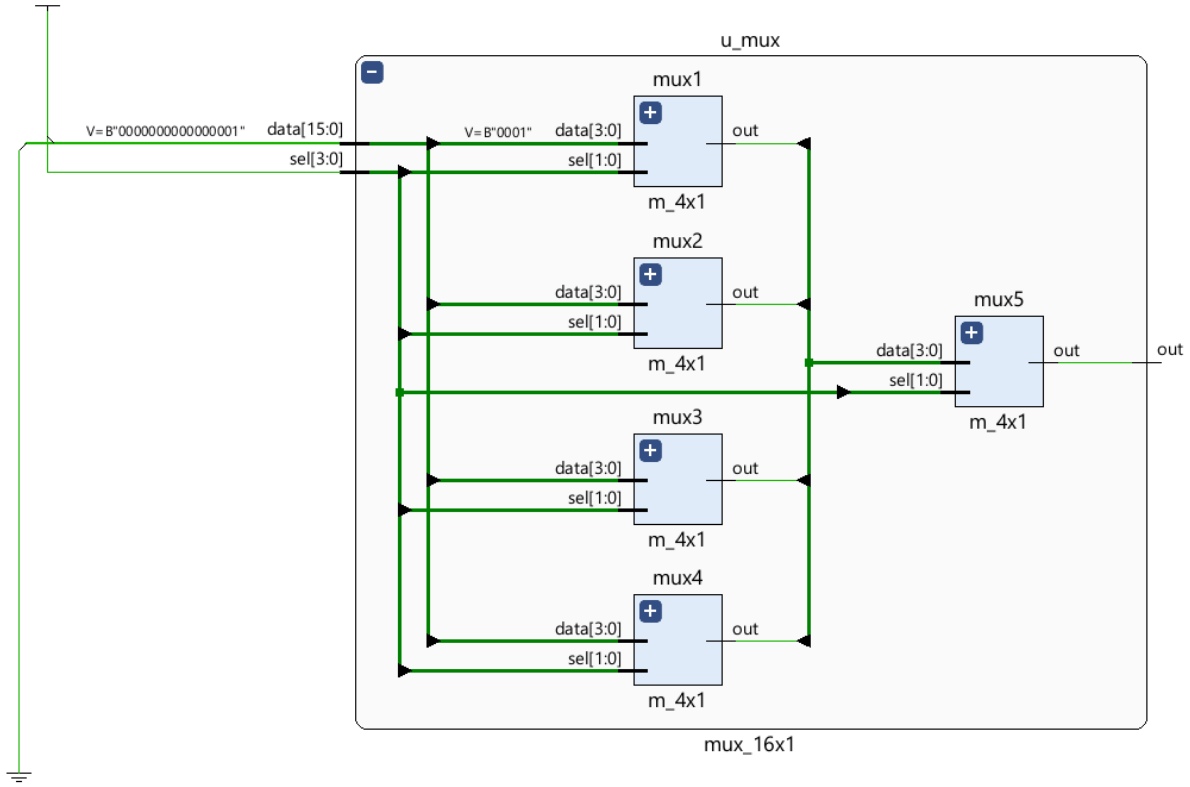
Test-bench

```
module tb_mux_16x1;

  // Inputs and outputs of the mux_16x1 module
  reg [15:0] data;
  reg [3:0] sel;
  wire y;

  // Instantiate the mux_16x1 module
  mux_16x1 u_mux (
    .data(data),
    .sel(sel),
    .y(y)
  );

  // Stimulus generation without a clock
  initial begin
    // Initialize inputs
    data = 16'b0000000000000000;
    sel = 4'b0000;

    // Test Case 1: Select input 0
    data = 16'b0000000000000001;
    sel = 4'b0000;
    #10; // Delay for simulation purposes
```

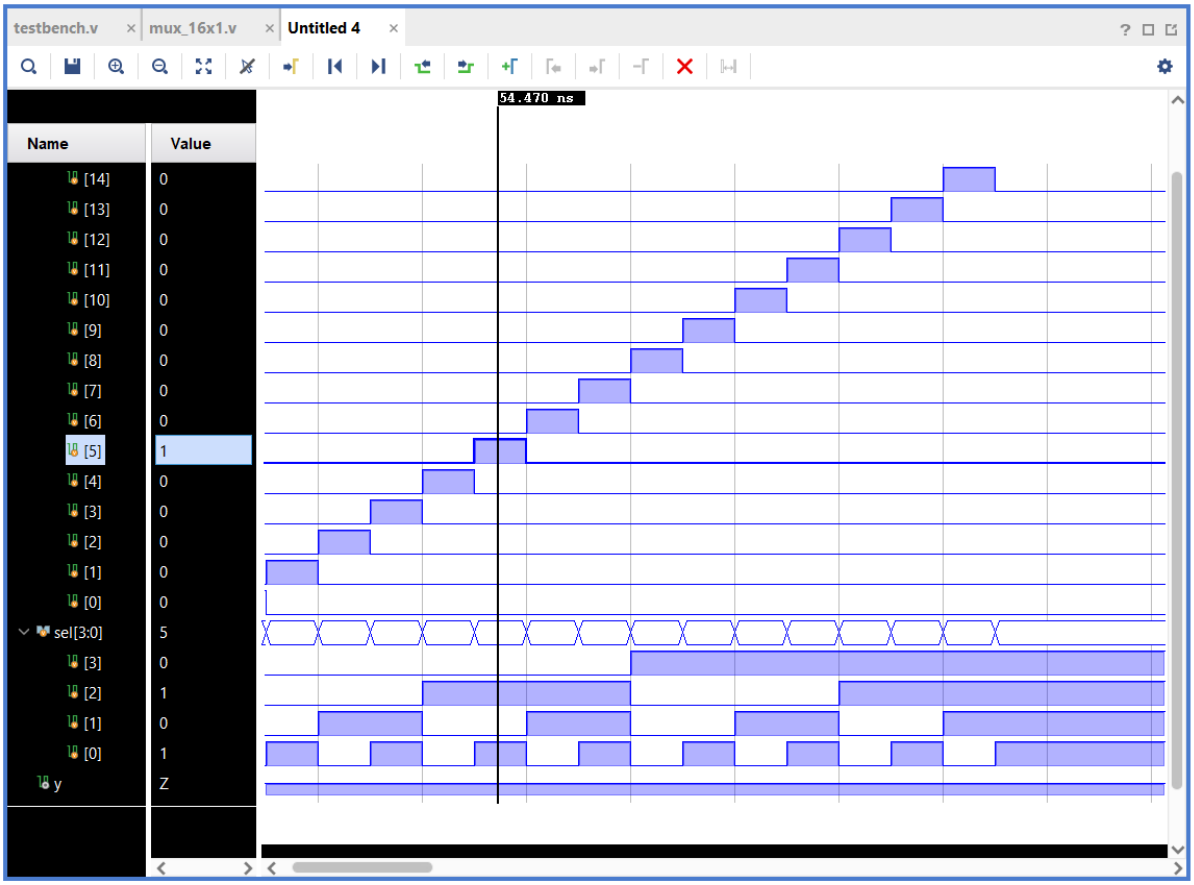
```
// Test Case 2: Select input 1
data = 16'b0000000000000010;
sel = 4'b0001;
#10; // Delay for simulation purposes

// Test Case 3: Select input 2
data = 16'b0000000000000100;
sel = 4'b0010;
#10; // Delay for simulation purposes

// Test Case 4: Select input 3
data = 16'b0000000000001000;
sel = 4'b0011;
#10; // Delay for simulation purposes

// Test Case 5: Select input 4
data = 16'b0000000000010000;
sel = 4'b0100;
#10; // Delay for simulation purposes

// Test Case 6: Select input 5
data = 16'b0000000000100000;
sel = 4'b0101;
#10; // Delay for simulation purposes

// Test Case 7: Select input 6
data = 16'b0000000001000000;
sel = 4'b0110;
#10; // Delay for simulation purposes

// Test Case 8: Select input 7
data = 16'b0000000010000000;
sel = 4'b0111;
#10; // Delay for simulation purposes

// Test Case 9: Select input 8
data = 16'b0000000100000000;
sel = 4'b1000;
#10; // Delay for simulation purposes

// Test Case 10: Select input 9
data = 16'b0000001000000000;
sel = 4'b1001;
#10; // Delay for simulation purposes

// Test Case 11: Select input 10
```

```
    data = 16'b0000010000000000;
    sel = 4'b1010;
    #10; // Delay for simulation purposes

    // Test Case 12: Select input 11
    data = 16'b0000100000000000;
    sel = 4'b1011;
    #10; // Delay for simulation purposes

    // Test Case 13: Select input 12
    data = 16'b0001000000000000;
    sel = 4'b1100;
    #10; // Delay for simulation purposes

    // Test Case 14: Select input 13
    data = 16'b0010000000000000;
    sel = 4'b1101;
    #10; // Delay for simulation purposes

    // Test Case 15: Select input 14
    data = 16'b0100000000000000;
    sel = 4'b1110;
    #10; // Delay for simulation purposes

    // Test Case 16: Select input 15
    data = 16'b1000000000000000;
    sel = 4'b1111;
    #10; // Delay for simulation purposes

    // End simulation
    $finish;
  end

  // Display output
  always @(y) begin
    $display("Output y = %b", y);
  end

endmodule
```

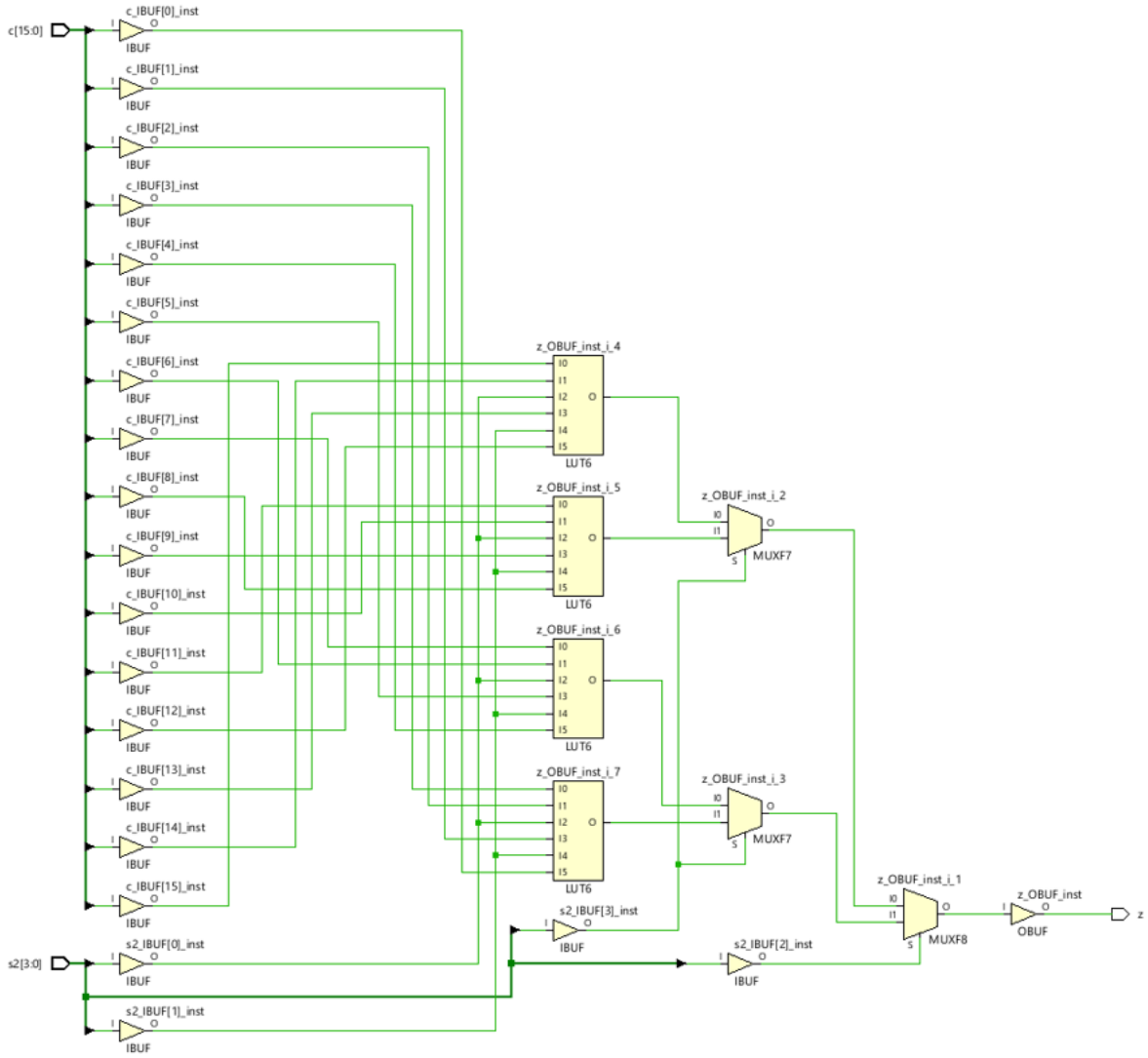
RTL Schematic:



Function Verification:



Post synthesis schematic:



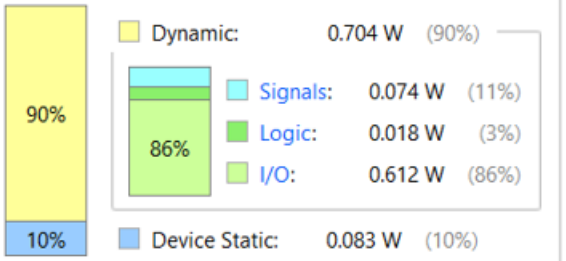
Post power report:

Summary

Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power:	**0.787 W**
Design Power Budget:	**Not Specified**
Process:	typical
Power Budget Margin:	**N/A**
Junction Temperature:	**26.5°C**
Thermal Margin:	58.5°C (30.9 W)

On-Chip Power



Post synthesis 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:	1	Total Number of Endpoints:	1	Total Number of Endpoints:	NA

There are no user specified timing constraints.

Post utilization/ area summary:

Hierarchy

Name	Slice LUTs (41000)	F7 Muxes (20500)	F8 Muxes (10250)	Slice (10250)	LUT as Logic (41000)	Bonded IOB (300)
mux161	4	2	1	1	4	21