

**DIGITAL SYSTEM DESIGN APPLICATION**

**EHB436E CRN: 11280**

**Salih Ömer Ongün**

**040220780**

**Experiment 3**

**REALIZATION WITH SSI Library**

**Karnaugh Map of Outputs**

metin, el yazısı, çizim, taslak içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, el yazısı, kağıt, mürekkep içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, el yazısı, çizim, mürekkep içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Gate Level Circuit Schematic**

metin, çizim, taslak, el yazısı içeren bir resim

Açıklama otomatik olarak oluşturuldu

**TCL Console Output**

**metin, ekran görüntüsü, yazı tipi, siyah beyaz içeren bir resim

Açıklama otomatik olarak oluşturuldu**

As you can see from the TCL console output, the design made with the SSI Library works correctly.

If there are 2-bit 2 numbers at the input of the function, 2-bit parallel operation can be performed in the function. These inputs can be used for two-bit parallel multipliers.

**NO TIMING AND LOC CONSTRAINTS**

**RTL Schematic**

**diyagram, metin, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Technology Schematic**

**metin, diyagram, sayı, numara, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Pad to Pad Delays Resource Usage metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**TIMING CONSTRAINED DESIGN WITHOUT LOC CONSTRAINS**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

I set constraints to adjust maximum delay to 9 ns. However, the delays from port "a" to port "f1" exceed 9 ns by 0.048 ns. Although the delays cannot achieve the 9ns target, there is a significant reduction in delays.

**Resource Usage**

**metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

There are not any differences in resource usage.

**LOC CONSTRAINED DESIGN WITHOUT TIMING CONSTRAINS**

**Placement**

metin, yazılım, multimedya yazılımı, grafik yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

**f0 placement**

ekran görüntüsü, yazılım, multimedya yazılımı, grafik yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

**f1 Placement**

f1 output is was saved as xor1 not f1\_OBUF\_inst\_i\_1.

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

I placed f2 and f3 at X13Y67 and X14Y64 respectively. However, f2 and f3 are placed in the same "Slice". They cannot be separated.

**Pad to Pad Path Delays**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Delays are more than no constraints and timing constraints.

**TIMING AND LOC CONSTRAINED DESIGN**

**Pad to Pad Path Delays**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

I set constraints to adjust maximum delay to 9 ns. However, some delays exceed 9 ns. Although the delays cannot achieve the 9ns target, there is a significant reduction in delays.

The design where timing is constrained has the least delay design. However, it has not reached the 9ns goal. Placement increases pad to pad delays. I set constraints to adjust maximum delay to 9 ns, but design has not reached this goal. Vivado adjusts placement according to optimum design for time and power consumption. Therefore, different placements increase delays.

**Simulation Wave**

Post-implementation timing simulation waveform

ekran görüntüsü, multimedya yazılımı, grafik yazılımı, renklilik içeren bir resim

Açıklama otomatik olarak oluşturuldu

Behavioral Simulation Waveform

ekran görüntüsü, multimedya yazılımı, grafik yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

Behavioral simulation waveforms have clear waveforms. However, post-implementation timing simulation waveform shifted waveforms. Redline delays may have caused this shift.

**REALIZATION WITH DECODER**

**4X16 Decoder**

**metin, el yazısı, taslak, çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Behavioral Simulation Waveform**

**ekran görüntüsü, multimedya yazılımı, grafik yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu**

According to a,b,c,d inputs from m0 to m15 outputs have logic 1 value. Our f0,f1,f2,f3 outputs have SOP forms. In this form outputs have or structure. For a=1,b=1,c=0, d=1 values m13 have logic 1 value. F0 and f1 have m13 outputs. Therefore, f0 and f1 have logic 1 value.

**RTL Schematic**

**diyagram, metin, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Technology Schematic**

**metin, diyagram, plan, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Pad to Pad Delays**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

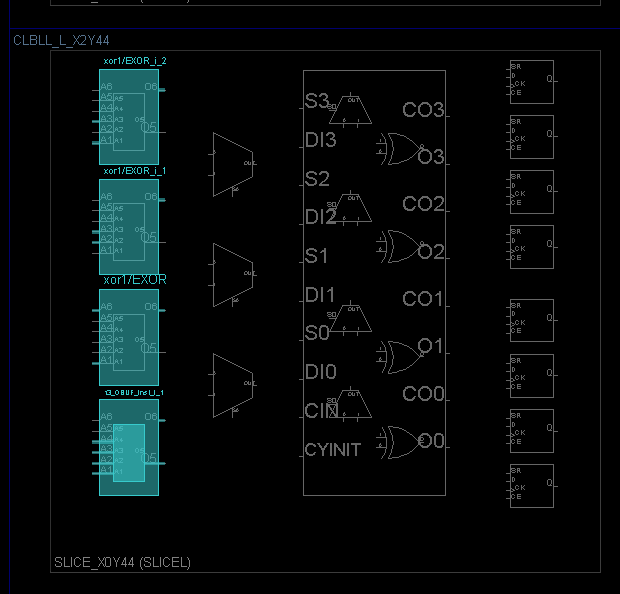
**Resource Usage**

**metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Resource usage is the same as the SSI library design.

**Placement**

****

**f2, f 3, xor1(f1) placement**

**metin, ekran görüntüsü, diyagram, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**f0 placement**

The cell used is the same as the SSI library design.

**DESIGN WITH TIMING CONSTRAINTS**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

I set constraints to adjust maximum delay to 6 ns. However, all paths exceed 6 ns. Although the delays cannot achieve the 6ns target, there is a significant reduction in delays.

**REALIZATION WITH MUX**

**Handwriting**

metin, el yazısı, kağıt, mürekkep içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Behavioral Simulation Waveform**

**ekran görüntüsü, multimedya yazılımı, grafik yazılımı, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu**

For f1 output, a=0 and c=0, f1= 0. a=0 and c=1, f1= b. c=1 and a=0, f1= b. c=1 and a=1, f1= (b xor d). For all outputs waveform provide exact true value.

**RTL Schematic**

metin, diyagram, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Technology Schematic**

metin, diyagram, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Resource Usage**

**metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Resource usage is the same as the SSI library design and Decoder design.

**Pad to Pad Delays**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Placement**

**metin, ekran görüntüsü, diyagram, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**f0 placement**

**metin, ekran görüntüsü, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**f2, f 3, xor1(f1) placement**

The cell used is the same as the SSI library design.

**DESIGN WITH TIMING CONSTRAINTS**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

I set constraints to adjust maximum delay to 6 ns. However, all paths exceed 6 ns. Although the delays cannot achieve the 6ns target, there is a significant reduction in delays.

All realizations have almost the same no timing constraints design delays. SSI\_Library design cannot achieve the 9ns target but there is a significant reduction in delays. Likewise, decoder and mux design cannot achieve the 6ns target, but there is a significant reduction in these delays. Decoder and mux have less delay than SSI\_Library design with timing constraints. Decoder and Mux designs have less logic gates. Therefore, it is natural that their delays are less than SSI\_Library.

All three realizations have same resource usage.

**RESEARCH**

1. There are different types of simulations in Vivado. There are fundamental differences for these simulations. There are different types of simulations in Vivado. There are fundamental differences for these simulations. Behavioral Simulation simulates design by ignoring the physical structure. They show the basic result that shows whether the design is correct or not. RTL analysis represents a digital system with logic gates. It does not show the structure of Fpga. Synthesis represents the digital system with FPGA resources. Therefore, Post-Synthesis Functional Simulation has more reliable simulation results compared to Behavioral Simulation. The synthesis phase does not include device placement. Therefore, it does not show absolute simulation and timing results. The implementation phase computes the simulation and timing results according to the placement on the FPGA board. Therefore, Post-Implementation Functional Simulation has more reliable results than Post-Synthesis Functional Simulation. There are different types of constraints. Timing constraint is one of these types. The design is put into practice according to the designer's timing constraint. Therefore, Post-Implementation Timing Simulation gives the most reliable design if there are any timing constraints.
2. Vivado implements the design onto the FPGA board based on optimal conditions. The design provides optimal results for timing, power consumption, placement, and other constraints. However, designers may want a design with less delay. The system may sacrifice less power consumption to meet the timing constraint. For such reasons, designers create different constraints to fulfill their demands.

Synthesis constraints are constraints that include how the synthesis of HDL code into RTL occurs. They have different synthesis constraints that have different format and using.

I/O constraints are used assign input/ outputs pins to specific locations. I/O constraints is important especially ASIC and fast Fpga applications.

Timing constraints are used for specific timing assignments. Designers can assign specific time value for specific paths or all paths. Vivado implements design according to these constraints.

Placement constraints are used for assignment Fpga resources to specific locations. Designers can want place LUTs or other resources to specific locations.

1. Synthesis attributes allow the design to be shaped as the designer wishes. Vivado wants to optimize the design for the most favorable conditions, sometimes optimizations are not the best solutions for specific designs. Therefore, synthesis attributes help designers for these implementations.

"DONT\_TOUCH" prevents Vivado from changing or deleting a signal for optimization purposes.

(\* dont\_touch = "yes" \*) wire sig1; assign sig1 = in1 & in2;

assign out1 = sig1 & in2;

**Verilog example of DONT\_TOUCH**

There are different memory RAM types in Vivado. RAM\_STYLE allows the designer to use any type of RAM they want.

(\* ram\_style = "distributed" \*) reg [data\_size-1:0] myram [2\*\*addr\_size-1:0];

**Verilog example of RAM\_STYLE**