EGR 400 A
Advanced Digital
System Design Using FPGAs

Lab 2: Barrel Shifter Design

Prepared for: Dr. Foist

Christopher Parisi

College of Engineering
California Baptist University

10/05/12

Introduction

The objective for this lab was to design, simulate, synthesize, and verify a multifunctional barrel shifter. A multifunctional barrel shifter is a shifter that can shift right or left depending on an external signal. The multifunctional barrel shifter can also shift by different amounts. The purpose of the lab was to teach students about the barrel shifter and how to design one. This lab also teaches students more about the ISE IDE, Spartan 3E-Starter board, and Xilinx Design Flow.

Procedure

Preparation

Before starting the lab 2 assignment, I began by downloading Dr. Chu's code for a basic rotate-right barrel circuit. I downloaded the code onto the Spartan 3E-Starter board and verified the design. This design utilized 8 LEDs as outputs, and 3 switches for the shift amount. Once I had confirmed the design, I saved it in my lab 2 directory for future use. I then modified the code so that the module became a rotate-left circuit. I confirmed this modification and saved the module. The VHDL code for the rotate left and rotate right circuits can be found in Appendix A.

Step 1

"Design the circuit using one rotate-right circuit, one rotate-left circuit, and one 2-to-1 multiplexer to select the desired result. Derive the code."

I began this step by drawing out the hardware diagram for this circuit. I then used this schematic to write the VHDL code to implement my design. My code utilized the previously mentioned rotate right and left shifter modules. The right shifted result or left shifted result is determined by the rl input used in the MUX. My schematic and code can be found in Appendix B. A summary of the chip resources used can also be found in Appendix B.

Step 2

"Derive a testbench and use simulation to verify operation of the code."

After writing the code, I coded a testbench to verify my code before downloading it to the board. I programmed in the important inputs and confirmed my design. The simulation was accurate and verified my design. My testbench VHDL and Output Waveform can be found in Appendix C.

Step 3
"Synthesize the circuit, program the FPGA, and verify its operation."

After testing my code with the testbench, I generated the programming files and downloaded my design to the board. Flipping the four switched turned on the 4 LEDs, then pressing the 3 buttons shifted the amount left. I pressed the button for shifting right and again pressed the 3 buttons for shifting which verified my design.

Conclusion

In summary, this lab has taught me the functionality and implementation of a barrel shifter. This lab has also given me more experience "thinking hardware" as I write VHDL code. Everything during this lab worked correctly and there were no major hindrances. I learned more about the naming convention of files and how to keep projects organized in order to be efficient. I now feel very comfortable deriving testbenches and using ISE for programming and synthesizing Digital System Designs using FPGA.

Appendix A

Rotate-Right VHDL Code

```
module barrel_shifter_right
  (
   input wire [7:0] a,
   input wire [2:0] amt,
   output wire [7:0] y
  );

  // signal declaration
  wire [7:0] s0, s1;

  // body
  // stage 0, shift 0 or 1 bit
  assign s0 = amt[0] ? {a[0], a[7:1]} : a;
  // stage 1, shift 0 or 2 bits
  assign s1 = amt[1] ? {s0[1:0], s0[7:2]} : s0;
  // stage 2, shift 0 or 4 bits
  assign y = amt[2] ? {s1[3:0], s1[7:4]} : s1;

endmodule
```

Rotate-Left VHDL Code

```
module barrel_shifter_left
  (
   input wire [7:0] a,
   input wire [2:0] amt,
   output wire [7:0] y
  );

  // signal declaration
  wire [7:0] s0, s1;

  // body
  // stage 0, shift 0 or 1 bit
  assign s0 = amt[0] ? {a[6:0], a[7]} : a;
  // stage 1, shift 0 or 2 bits
  assign s1 = amt[1] ? {s0[5:0], s0[7:6]} : s0;
  // stage 2, shift 0 or 4 bits
  assign y = amt[2] ? {s1[3:0], s1[7:4]} : s1;

endmodule
```

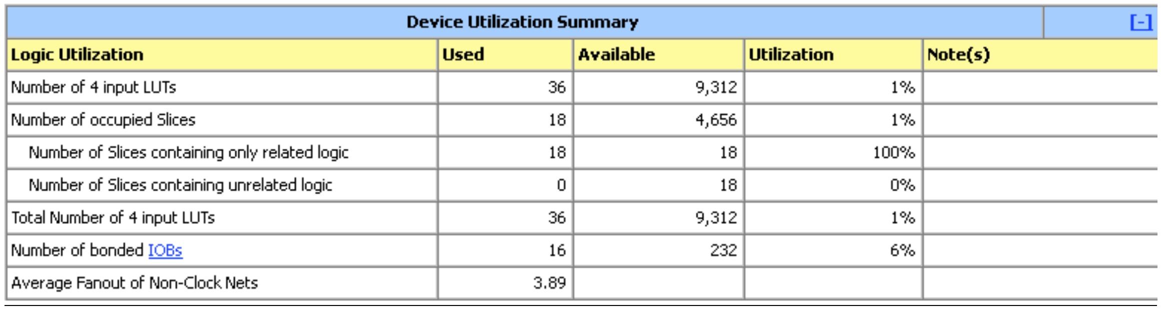
Appendix B

Multifunction Barrel Shifter Schematic

Multifunction Barrel Shifter VHDL Code

```
module multifunctional_shifter_fpga
  (
   input wire [2:0] btn,
   input wire [3:0] sw,
   input lr,
   output wire [7:0] led
  );

  wire [7:0] barShft_in;
          wire [7:0] s0, s1;

  // Assign 4 0's + 4 switches as BarS input (we have only 4 switches)
  assign barShft_in = {4'b0000, sw};

  // Rotate-left module connections
  barrel_shifter_left u1
          (.a(barShft_in), .amt(btn), .y(s0));

  // Rotate-right module connections
  barrel_shifter_right u2
          (.a(barShft_in), .amt(btn), .y(s1));

 // If North switch is pressed, shift right. If not, shift left
  assign led = lr ? s1 : s0;

endmodule
```

Summary of Chip Resources



Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of 4 input LUTs	36	9,312	1%	
Number of occupied Slices	18	4,656	1%	
Number of Slices containing only related logic	18	18	100%	
Number of Slices containing unrelated logic	0	18	0%	
Total Number of 4 input LUTs	36	9,312	1%	
Number of bonded IOBs	16	232	6%	
Average Fanout of Non-Clock Nets	3.89			

Appendix C

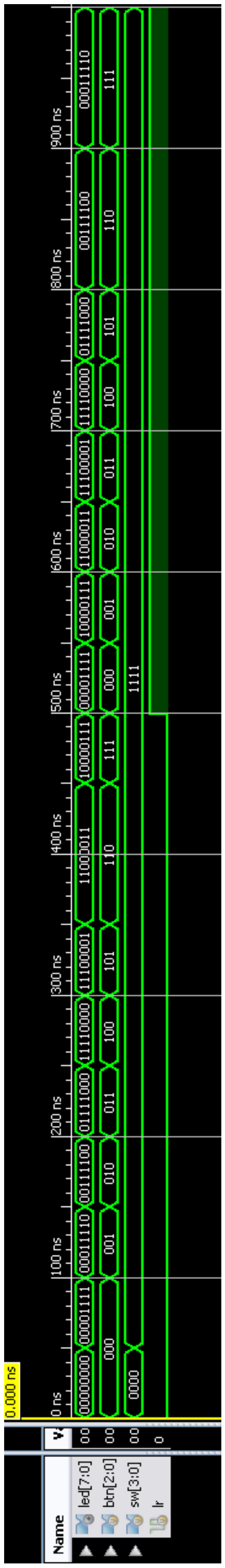
Multifunction Barrel Shifter Testbench VHDL

```
`timescale 1ns / 1ps

module tb;

	// Inputs
	reg [2:0] btn;
	reg [3:0] sw;
	reg lr;

	// Outputs
	wire [7:0] led;

	// Instantiate the Unit Under Test (UUT)
	multifunctional_shifter_fpga uut (
		.btn(btn),
		.sw(sw),
		.lr(lr),
		.led(led)
	);

	initial begin
		// Initialize Inputs
		btn = 0;
		sw = 0;
		lr = 0;

		// Wait 100 ns for global reset to finish
		#50;

		// Test Vector 2
		btn = 0;
		sw = 4'b1111;
		lr = 0;
		#50;

		// Test Vector 3
		btn = 3'b001;
		sw = 4'b1111;
		lr = 0;
		#50;
```

```
// Test Vector 4
btn = 3'b010;
sw = 4'b1111;
lr = 0;
#50;

// Test Vector 5
btn = 3'b011;
sw = 4'b1111;
lr = 0;
#50;

// Test Vector 6
btn = 3'b100;
sw = 4'b1111;
lr = 0;
#50;

// Test Vector 7
btn = 3'b101;
sw = 4'b1111;
lr = 0;
#50;

// Test Vector 8
btn = 3'b110;
sw = 4'b1111;
lr = 0;
#100;

// Test Vector 9
btn = 3'b111;
sw = 4'b1111;
lr = 0;
#50;

//Right Shift
// Test Vector 10
btn = 0;
sw = 4'b1111;
lr = 1;
#50;

// Test Vector 11
btn = 3'b001;
sw = 4'b1111;
```

```
		lr = 1;
		#50;

		// Test Vector 12
		btn = 3'b010;
		sw = 4'b1111;
		lr = 1;
		#50;

		// Test Vector 13
		btn = 3'b011;
		sw = 4'b1111;
		lr = 1;
		#50;

		// Test Vector 14
		btn = 3'b100;
		sw = 4'b1111;
		lr = 1;
		#50;

		// Test Vector 15
		btn = 3'b101;
		sw = 4'b1111;
		lr = 1;
		#50;

		// Test Vector 16
		btn = 3'b110;
		sw = 4'b1111;
		lr = 1;
		#100;

		// Test Vector 17
		btn = 3'b111;
		sw = 4'b1111;
		lr = 1;
		#50;


	end

endmodule
```

Multifunctional Barrel Shifter Simulation Waveform


Name
led[7:0]
btn[2:0]
sw[3:0]
lr
0.000 ns
0 ns
100 ns
200 ns
300 ns
400 ns
500 ns
600 ns
700 ns
800 ns
900 ns