Computer Architecture Fall 2016

Lab 0 Report
Full Adder on FPGA

Ian Paul, Jee Kim, Shruti Iyer

Waveforms

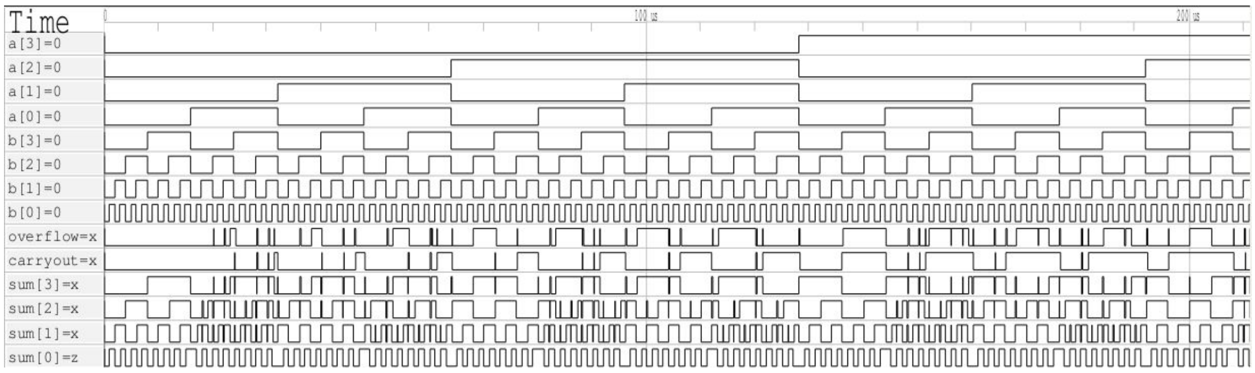


Figure 1. Full waveform of all 256 cases

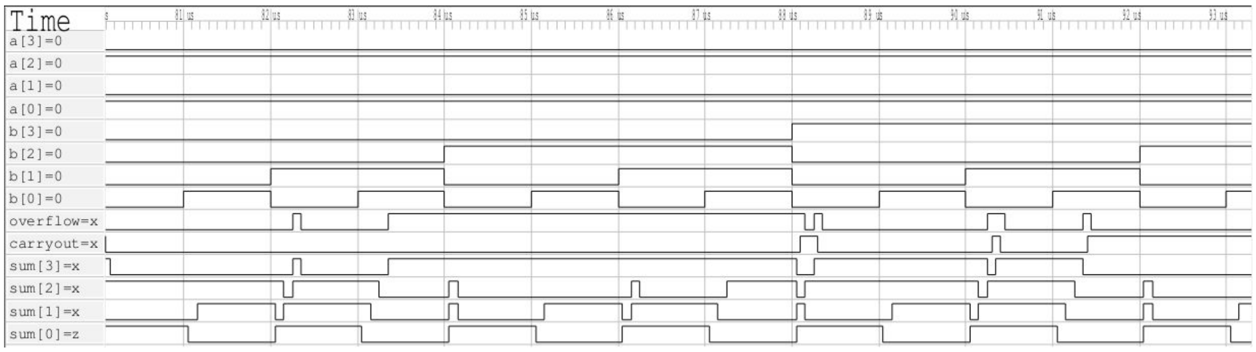


Figure 2. Zoomed in version of waveforms from 80us to 96us

Worst Case Delay

Our worst case delay on the test bench is 450 nanoseconds (with the delay on each gate set to 50 nanoseconds). This delay happens when everything in the sum changes from 1 to 0 or 0 to 1.

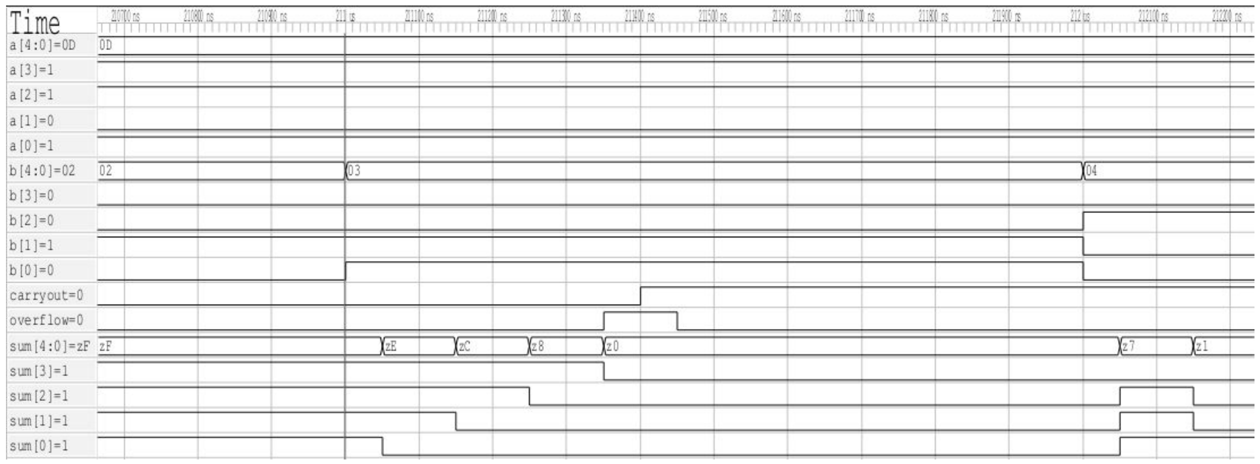


Figure 3. Showing one of the worst case delays that happened at 211us. The carry out took 400ns to switch and there is a glitch at overflow happened from 350ns to 450ns since the b changed at 211ns.

Tests

List of Test Cases

a	b	sum	cout	overflow
0000	0000	0000	0	0
0111	0111	1110	0	1
1001	1001	0100	1	1
0111	1001	0001	0	0
0011	0011	0110	0	0
0110	0001	0111	0	0
1111	0001	0000	1	0
1101	1101	1010	1	0
1011	1101	1000	1	0
0001	0001	0010	0	0
0101	1010	1011	0	0
0110	0011	1001	0	1
1111	1111	1110	1	0
1110	1101	1011	1	0
1100	1100	1000	1	0
1000	1111	0100	1	1

Test Case Strategy

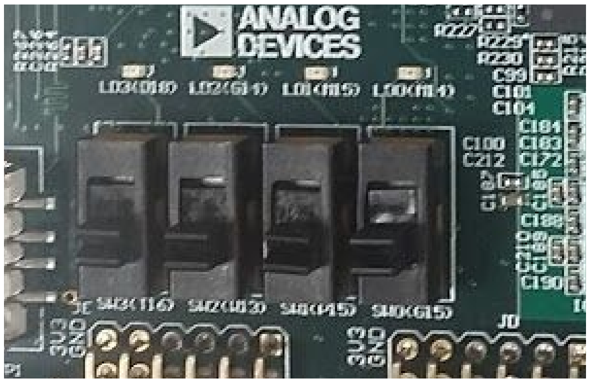
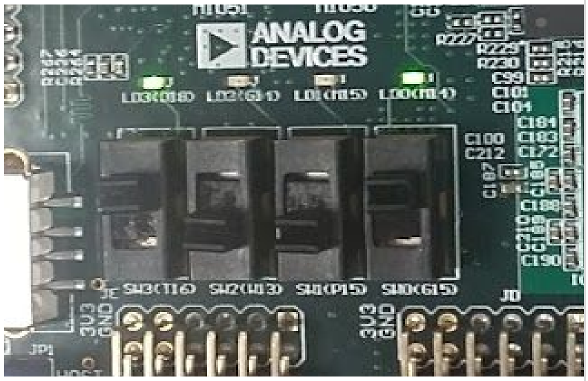
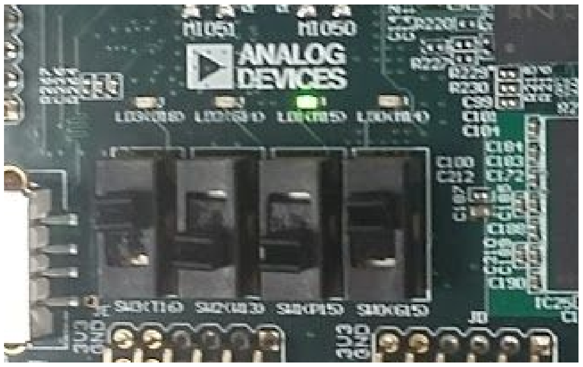
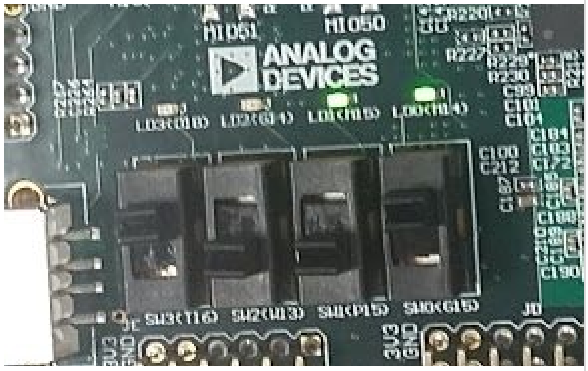
We wanted our test cases to capture both the areas we expect failure and success. We looked for cases where we add two positives (with and without overflow), two negatives (with and without overflow), a positive and a negative, and zero to values. We also checked to make sure the carry out was behaving as expected.

Test Bench Failures

Initially, we detected overflow by XORing the most significant sum (the last sum computed) with the last carryout. However, when we added -1_{10}(b1111) with 2_{10}(b0011), we got 1_{10}(b0001) with carryout 1. XOR of the most significant sum, b0, and the carryout, b1, gave b1 although there was no overflow.
From this test bench failure, we learned that we should calculate overflow by XORing the last carryout with the last carryin and corrected out 4 bit full adder code accordingly.

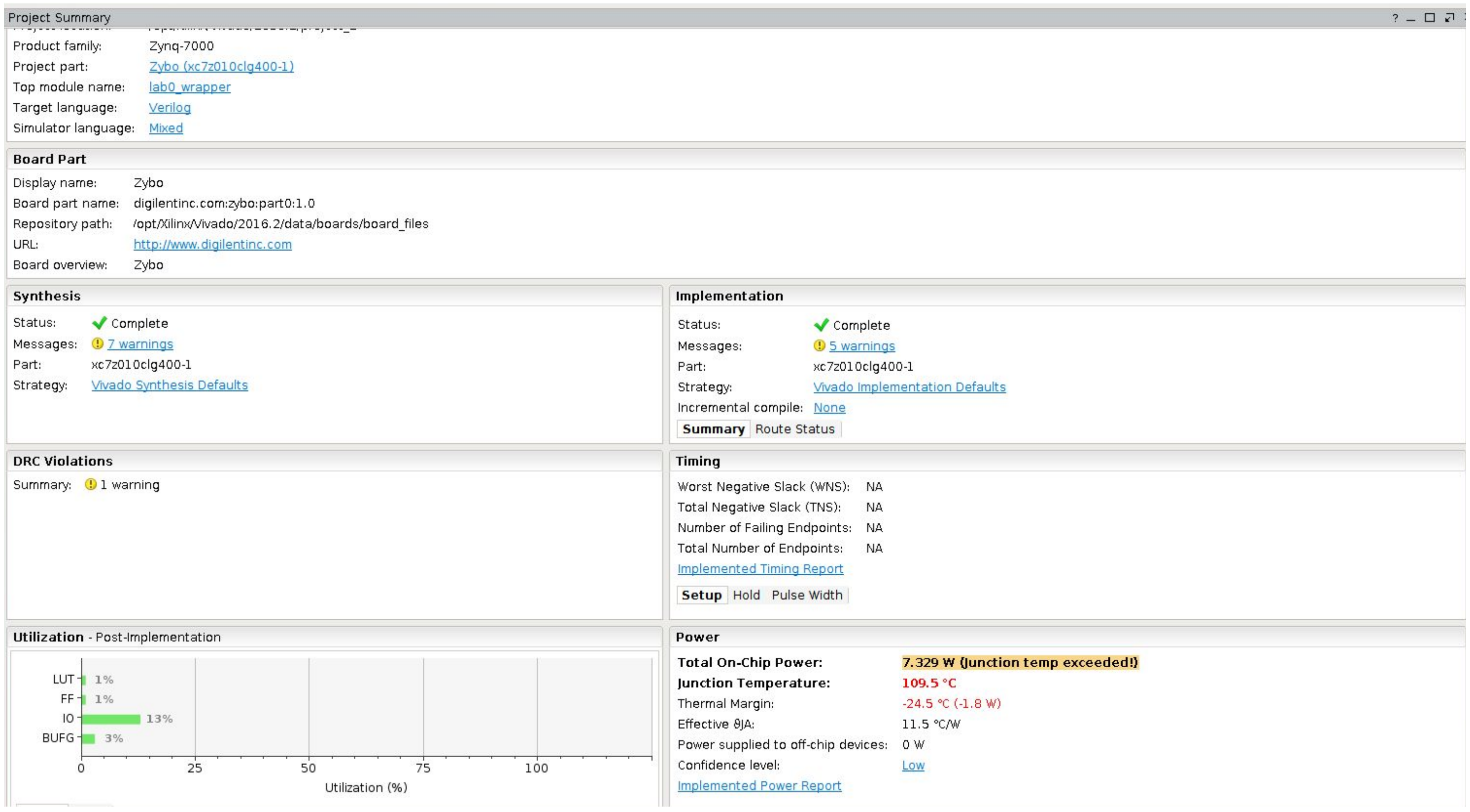
FPGA Testing

We tested all of the test cases in the table above on the FPGA. Below are pictures of one of the test cases.

	
1. We started out with a, b, and the sum all at 0	2. First we set a to 1001
	
3. We set b to 1001, and the sum showed as 0010	4. Both the overflow and carry out showed as 1

Summary statistics

Test Cases: 16
Tests passed: 16
Percent of tests passes: 100%


Project Summary

Product family:	Zynq-7000
Project part:	Zybo (xc7z010clg400-1)
Top module name:	lab0_wrapper
Target language:	Verilog
Simulator language:	Mixed

Board Part

Display name:	Zybo
Board part name:	digilentinc.com:zybo:part0:1.0
Repository path:	/opt/Xilinx/Vivado/2016.2/data/boards/board_files
URL:	http://www.digilentinc.com
Board overview:	Zybo

Synthesis

Status:	✓ Complete
Messages:	7 warnings
Part:	xc7z010clg400-1
Strategy:	Vivado Synthesis Defaults

Implementation

Status:	✓ Complete
Messages:	5 warnings
Part:	xc7z010clg400-1
Strategy:	Vivado Implementation Defaults
Incremental compile:	None

DRC Violations

Summary: 1 warning

Timing

Worst Negative Slack (WNS):	NA
Total Negative Slack (TNS):	NA
Number of Failing Endpoints:	NA
Total Number of Endpoints:	NA

Implemented Timing Report

Utilization - Post-Implementation

Power

Total On-Chip Power:	**7.329 W (Junction temp exceeded!)**
Junction Temperature:	**109.5 °C**
Thermal Margin:	-24.5 °C (-1.8 W)
Effective θJA:	11.5 °C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Low

Implemented Power Report

Resources Used

[1] https://sites.google.com/site/ca16fall/resources/fpga
[2] https://github.com/CompArchFA16/Lab0
[3] http://sandbox.mc.edu/~bennet/cs110/tc/orules.html
[4] http://teaching.idallen.com/cst8214/08w/notes/overflow.txt