EE486A Introduction to Computer Architecture
Lab 2
Ripple Carry Adder/Subtracter in VHDL

CDR Charles B. Cameron

January 25, 2007

1 Introduction

The purpose of this lab is to design and simulate a ripple-carry adder/subtracter using VHDL. At the same time you will practice using the `GENERIC` keyword in your VHDL designs as well as placing sub-designs into a package for use in higher-level designs.

2 Equipment

Altera Quartus II software

3 Description

Start by designing a full adder capable of handling both addition and subtraction of two 1-bit 2's complement numbers. If the input signal $\overline{\mathsf{Add}}/\mathsf{Subtract}$ is low, the circuit should perform addition; otherwise it should perform subtraction.

The full adder should implement the following two functions:

$$C_{\text{out}} = A \cdot B + A \cdot C_{\text{in}} + B \cdot C_{\text{in}}$$
$$S = A \oplus B \oplus C_{\text{in}}$$

where A and B are one-bit addends, C_{in} is a one-bit input carry, and C_{out} is a one-bit output carry.

In your initial implementation of the circuit, use a `GENERIC` parameter `N` to represent the number of bits in the adder and let it have the default value 4.[1] Your design should easily extend to more bits simply by modifying the value of `N`.

[1] Details on how to create parameterized functions can be found in the Quartus II software help file.

When you have fully simulated and debugged your design, place it into a package [2] so that it will be available for incorporation in a larger design.

Then design an `N`-bit adder/subtracter with a high-order output carry bit C_{N}. Use the `GENERIC` parameter with a default value 4 at first so that you implement a 4-bit adder. To make the circuit perform addition, set the low-order input carry $C_0 = 0$. To make it perform subtraction complement one input and set the input carry bit $C_0 = 1$.

Ensure your design extends easily to more bits simply by changing the value of `N`. Your design should instantiate `N` copies of the full adder you created in the first part of the lab. Once the `N`-bit adder has been simulated and debugged for `N`= 4, modify parameter `N` to implement a 32-bit adder and simulate it, too.

4 Lab Report

Your lab report must contain all of your fully commented VHDL source files along with simulations for each of the following circuits:

1. the one-bit full adder.
2. the `N`-bit adder with `N`= 4.
3. the `N`-bit adder with `N`= 32.

The report itself should provide commentary on the VHDL code and the simulations so that they all can be easily understood.

[2] How to place a design in a package also is shown in the Quartus II software help file.