Assignment 13

Adder/Subtractor

Introduction

Beginning with our 32-bit adder, we will modify it to be an adder/subtractor. The carry-in bit c_0 will double as the carry-in bit as well as the select signal to select between inverting or not inverting the subtrahend.

The Circuit

Create a new circuit named `Add-Sub.LogicCircuit`. Import into it your 32-bit adder from Assignment 11. Note that when you import the 32-bit adder, LogicCircuit also imports all the circuits on which the 32-bit adder depends. So you will suddenly have a lot of circuits in your project.

Create a circuit named `Inverter 2` (label `INV 2`), which will be a 2-bit inverter. (A 1-bit converter is simply a NOT-gate.) Split the 2-bit input into two 1-bit channels, run them through a pair of NOT-gates, and then join the outputs into a 2-bit output from the inverter. Use this idea to build recursively a 4-bit inverter, an 8-bit inverter, a 16-bit inverter, and a 32-bit inverter.

The final step is to send the input from the subtrahend (b) into the 0-input of a 2:1 32-bit multiplexor (`MUX 2:1 (32)`). Also send the input from b through the 32-bit inverter and then into the 1-input of the multiplexor. Use the carry-in bit c_0 as the select signal. The output from the multiplexor should be sent into the 32-bit adder. You will need to create a 2:1 32-bit multiplexor to complete this stage.

The output from the adder/subtractor will be the sum if $c_0 = 0$ and it will be the difference if $c_0 = 1$.

Due Date

Place this file in a folder named `Assignment 13` and drop it in the dropbox. It is due by midnight Wednesday, November 6.