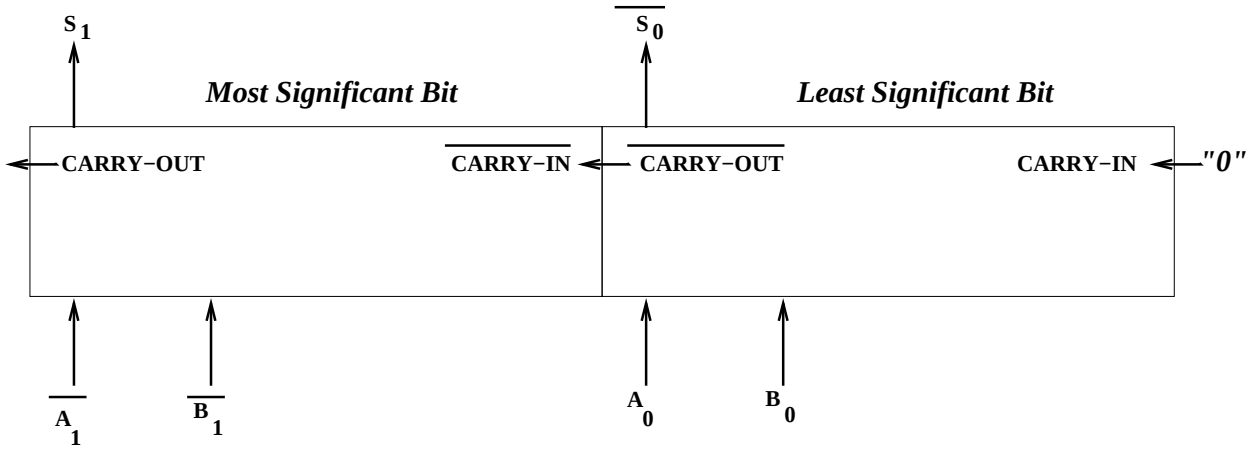
RUTGERS UNIVERSITY
Department of Electrical and Computer Engineering
14:332:479 Concepts in VLSI Design
Assignment II
Assigned: September 18, 2006
Due September 27, 2006

Reading Assignment: Chapters 1 and 2 and Section 10.2 up to the end of Section.10.2.2.2 of Weste and Eshraghian.

No collaboration is permitted on this assignment. Your work must be your own.

1. (**CMOS Arithmetic Circuit Wiring by Abutment.**)

   You are to design an adder module that takes two input n-bit unsigned binary numbers $\overline{A}$ and $\overline{B}$ and produces an output n-bit sum S, which is a logic 1 if the output sum bit is 1, and a logic 0 if the output sum bit is 0. The module also produces an output $CARRY\_OUT$ signal that is logic 1 if a carry was generated in that bit position, and a logic 0 otherwise. This type of adder is described in Figure 10.12(b) in Section 10.2.2.1 of the VLSI book. Its inputs are in negative logic, and its outputs are in positive logic. For ease of design, we arrange the adder for two n-bit numbers as an array of n one-bit *cells*. Thus, we only need to do a sticks layout for one cell, as follows. Each cell has three inputs (corresponding bits of $\overline{A}$ and $\overline{B}$,



   and input $CARRY\_IN$ signals) and two outputs (an output sum bit S and $CARRY\_OUT$). The cells are designed to be rectangular and designed to fit conveniently next to one another. We wire the cells by making sure that the outputs of one cell touch the inputs of the next cell, and are routed on the same wiring layer. This allows us to build an adder of arbitrary

size (any n) by simply juxtaposing a sufficient number of cells. You only have to design a single bit of the adder cell. However, the signals $\overline{A}$ and $\overline{B}$ should be on the bottom of the cell, and the output sum signal S should be on the top of the cell, directly across from input $\overline{A}$. Also, the $\overline{CARRY\_IN}$ signal should be on the left side of the cell and the $CARRY\_OUT$ signal should be on the right side. $\overline{CARRY\_IN}$ and $CARRY\_OUT$ signals for neighboring cells should line up and appear on the same routing layer. Otherwise, we cannot create an n-bit adder by lining up a row of your one-bit cells. For the least-significant bit of the adder, we wire the incoming signal $\overline{CARRY\_IN}$ to logic "0" to indicate to the hardware that the two numbers start out with no input carry. Turn in a logic schematic, a transistor schematic, and a sticks diagram for a single cell. Concentrate on making the cell rectangular and very compact. Don't forget to minimize your cell logic, using Karnaugh maps. For extra credit, explain what simple additions to the cell would turn it into a subtracter.

Turn in a sticks diagram with two copies of your cell wired by abutment. Each copy must be identical. Since this adder is its own self-dual, it can be viewed as adding negative logic inputs and producing positive logic outputs, or as adding positive logic inputs and producing negative logic outputs by adding $A + B$ with $CARRY\_IN$ and producing $\overline{S}$ and $\overline{CARRY\_OUT}$ (see Figure 10.12 (b) in the book).