EECS 150 - Components and Design Techniques for Digital Systems

Lec 17 – Addition, Subtraction, and Negative Numbers

David Culler
Electrical Engineering and Computer Sciences
University of California, Berkeley

http://www.eecs.berkeley.edu/~culler
http://www-inst.eecs.berkeley.edu/~cs150

Overview

- *Binary Addition*
  - Full Adder Revisited
- Ripple Carry
- Carry-select adder
- Carry lookahead adder
- *Binary Number Representation*
  - Sign & Magnitude, Ones Complement, Twos Complement

Computer Number Systems

- We all take positional notation for granted
  - $D_{k-1}\ D_{k-2}\ \ldots D_0$ represents $D_{k-1}B^{k-1} + D_{k-2}B^{k-2} + \ldots + D_0\ B^0$ where $B \in \{ 0, \ldots, B\text{-}1 \}$
  - Example: 2004_{10}, $1101_2 = 13_{10} = 0D_{16}$
- We all understand how to compare, add, subtract these numbers
  - Add each position, write down the position bit and possibly carry to the next position
- Computers represent finite number systems
- How do they efficiently compare, add, sub?
  - How do we reduce it to networks of gates and FFs?
- Where does it break down?
  - Manipulation of finite representations doesn't behave like same operation on conceptual numbers