Carry Lookahead and Signed-Digit Addition

Tom Kelliher, CS 240

Feb. 27, 2012

1 Administrivia

Announcements

Pay attention!!! A future homework assignment will cover this material, which isn't in the textbook.

Assignment

Read 4.6–4.7.

From Last Time

Addition limits.

Outline

1. Carry lookahead addition.

2. Signed digit representations.

Coming Up

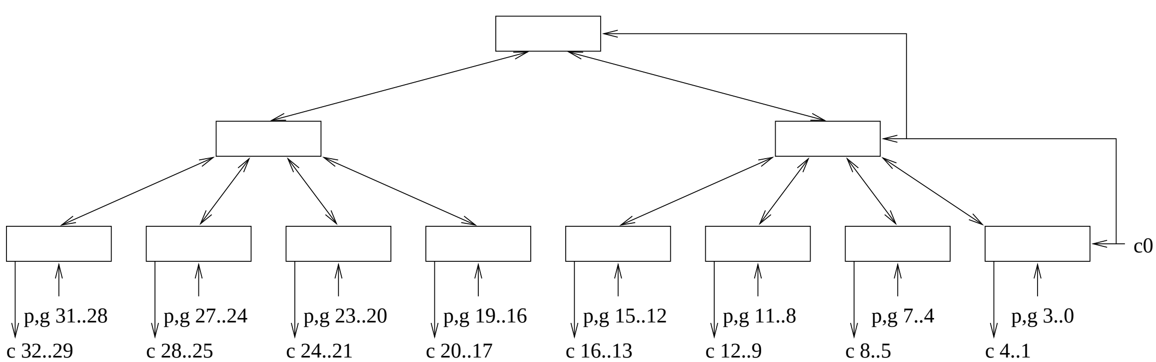
Introduction to VHDL.

2 Carry Lookahead Addition

1. Now, we demonstrate a feasible O($\log n$) adder.
2. Recall:
   (a) Carry generate: $g_i = a_i b_i$.
   (b) Carry propagate: $p_i = a_i \oplus b_i$.

2.1 Carry Lookahead: The Big Picture

Restricting the carry computation circuitry to a tree structure:

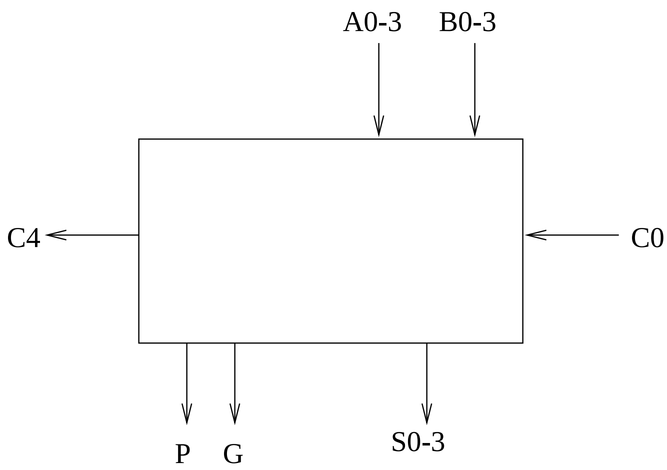


- Leaves: Four-bit carry lookahead adders.
- Non-Leaves: Four-bit carry lookahead group units.

2.2 Four-Bit Carry Lookahead Adder

1. Design a four-bit full carry lookahead adder.

Block diagram:



Block generate, propagate.

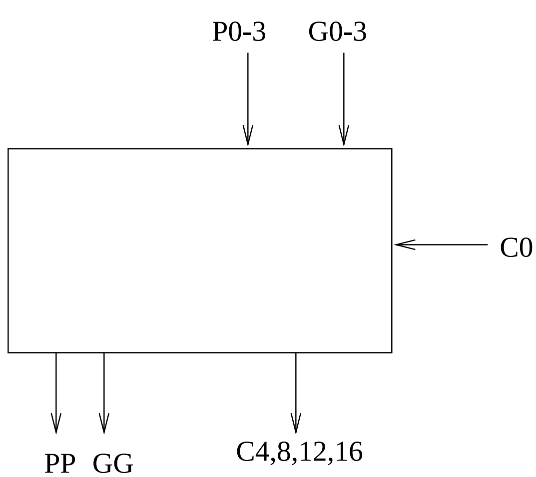
2. What is the fan-in?

3. What is the delay model from inputs to outputs?

2.3 4-Bit Group Carry Lookahead Unit

1. Design a 4-Group carry lookahead unit.

   Block diagram:



Use of block generates, propagates.

2. What is the fan-in?

3. What is the delay model from inputs to outputs?

2.4 16-Bit Carry Lookahead Adders

Total gate delays for ripple-carry adder.

Gate delays for cascaded and full carry lookahead adders.

3 Signed Digit Representations

1. Consider the digit set of the maximally redundant signed digit representation for radix r: $\{\overline{r-1}, \overline{r-2}, \ldots, \overline{1}, 0, 1, \ldots, r-1\}$

2. For radix 2 we have: $\{\overline{1}, 0, 1\}$.

   Radix 4: $\{\overline{3}, \overline{2}, \ldots, 3\}$.

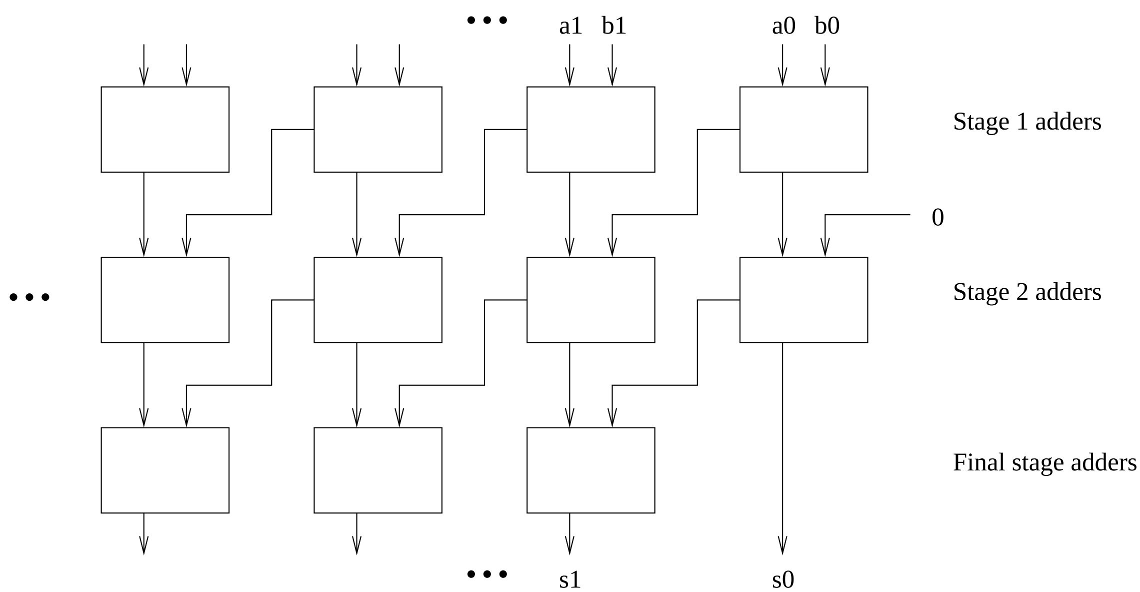
3. For some values, there are multiple representations. For example: $3 = 011 = 10\overline{1}$ (radix 2).

4. This redundancy can be exploited so that we can design constant time signed digit adders.

3.1 Constant Time Radix 2 Signed Digit Adder

1. Idea: Ensure that a carry propagates no further than two bit positions.

2. Circuit sketch:



3. Stage 1 adder addition table:

Addend + Augend	Carry	Sum
$\overline{2}$	$\overline{1}$	0
$\overline{1}$	$\overline{1}$	1
0	0	0
1	0	1
2	1	0

Goal: Ensure sums are ≥ 0 to eliminate -2 as a possible starting sum in the next stage.

4. Stage 2 adder addition table:

Addend + Augend	Carry	Sum
$\overline{1}$	0	$\overline{1}$
0	0	0
1	1	$\overline{1}$
2	1	0

Goal: Ensure sums are ≤ 0 to eliminate 2 and -2 as possible starting sums in the next stage.

5. Final stage addition table:

Addend + Augend	Carry	Sum
$\overline{1}$	0	$\overline{1}$
0	0	0
1	0	1