

Kogge-Stone Adder (1A)

-
-

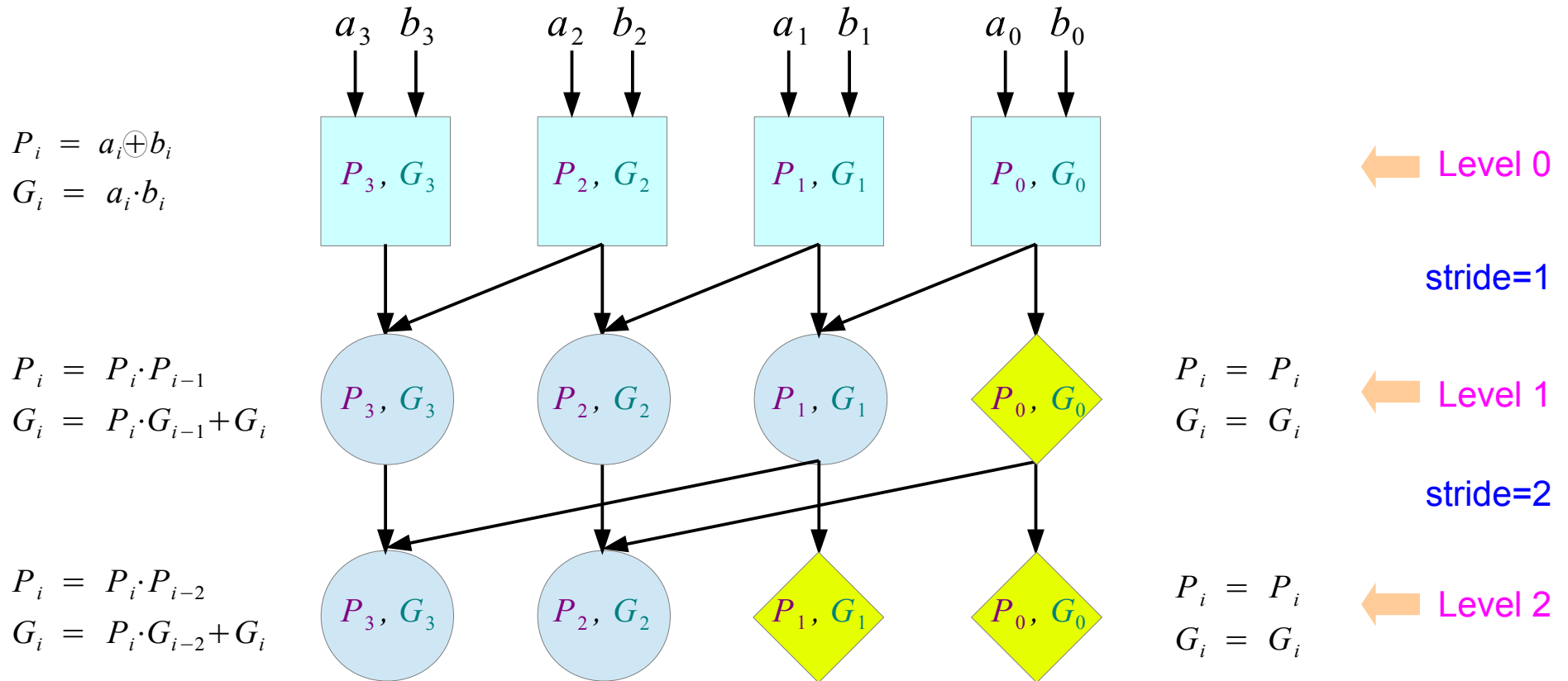
Copyright (c) 2013 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

4-bit Kogge-Stone Adder

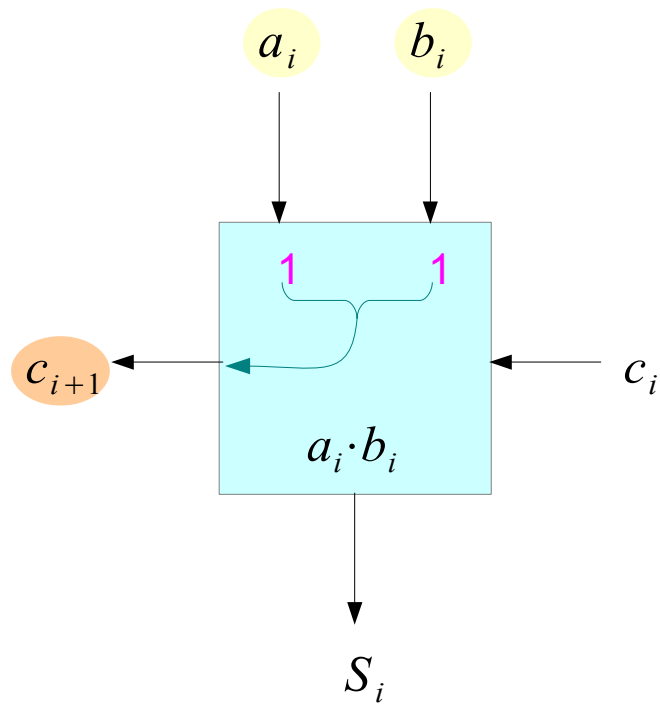


G & P at the leaf level

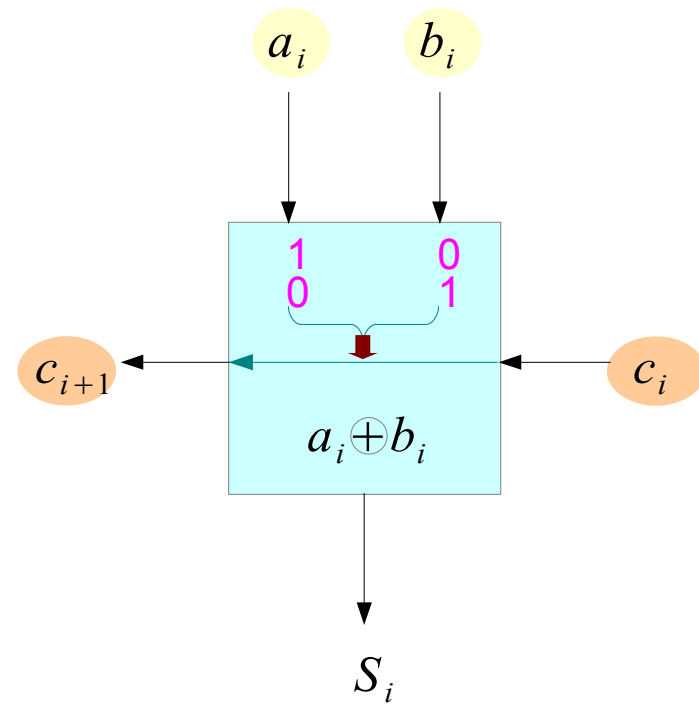
Generate $G_i = a_i \cdot b_i$

Propagate $P_i = a_i \oplus b_i$

$$c_{out} = G_i + P_i c_i$$

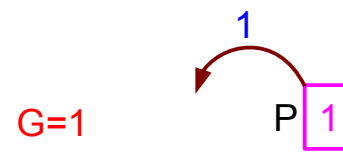
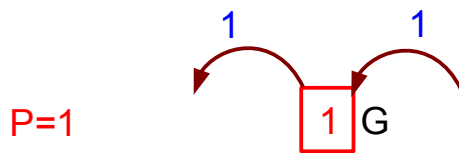
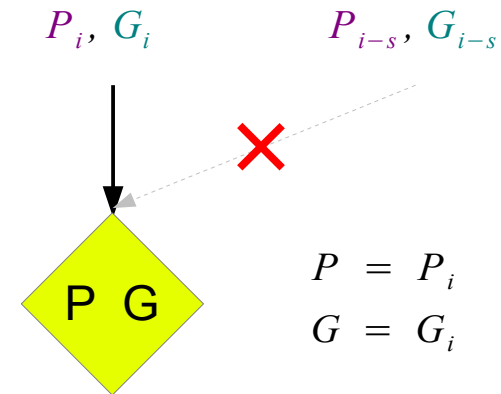
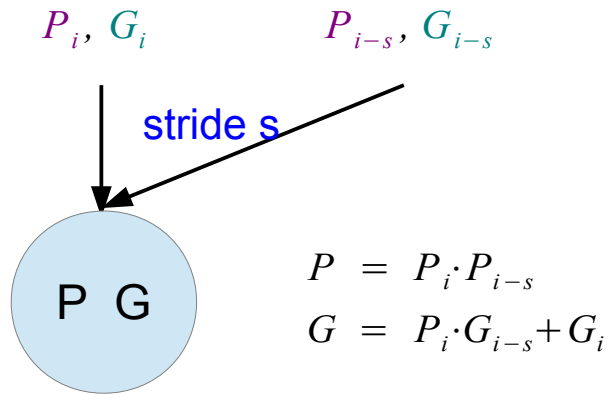


Generate c_{i+1}

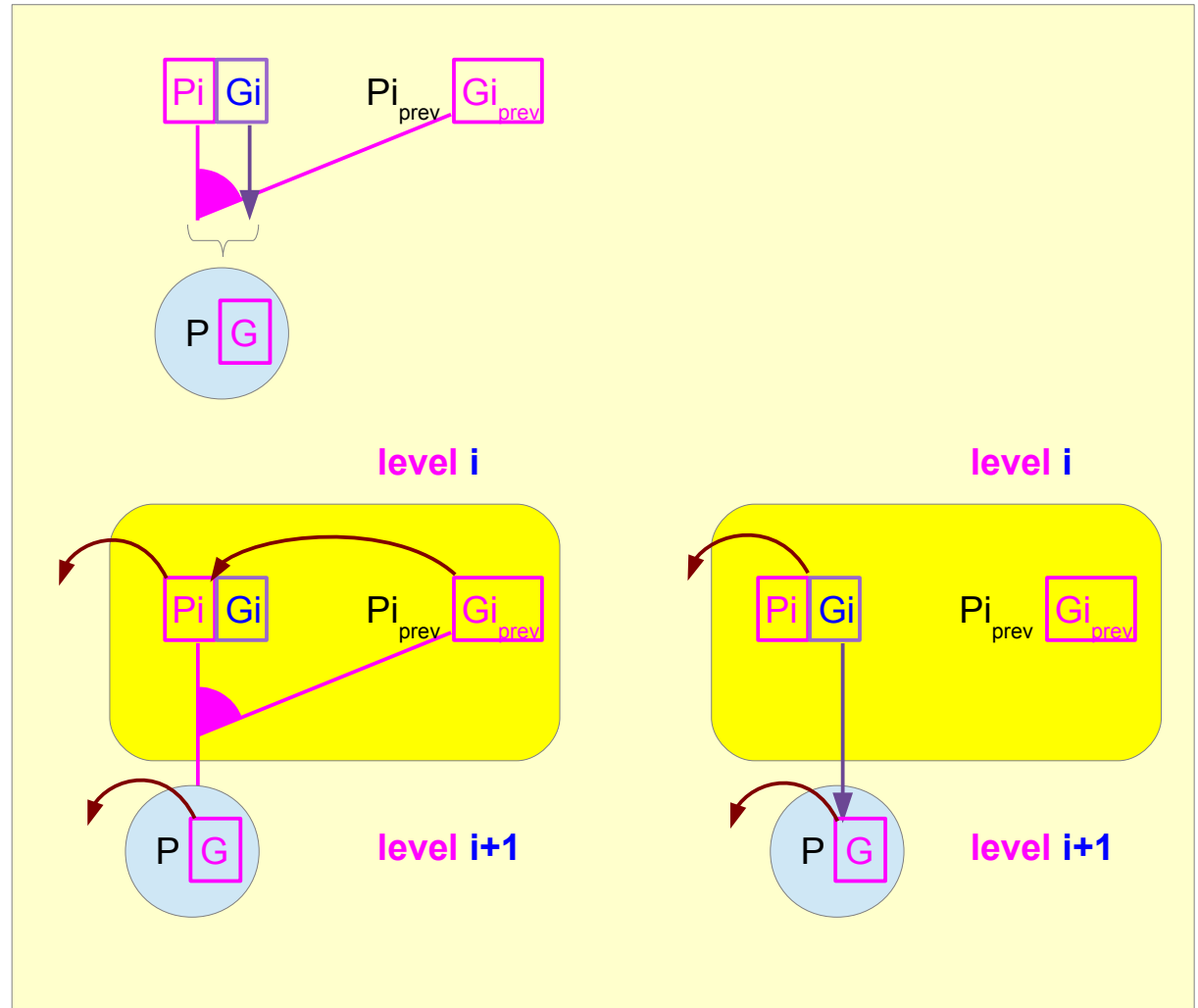
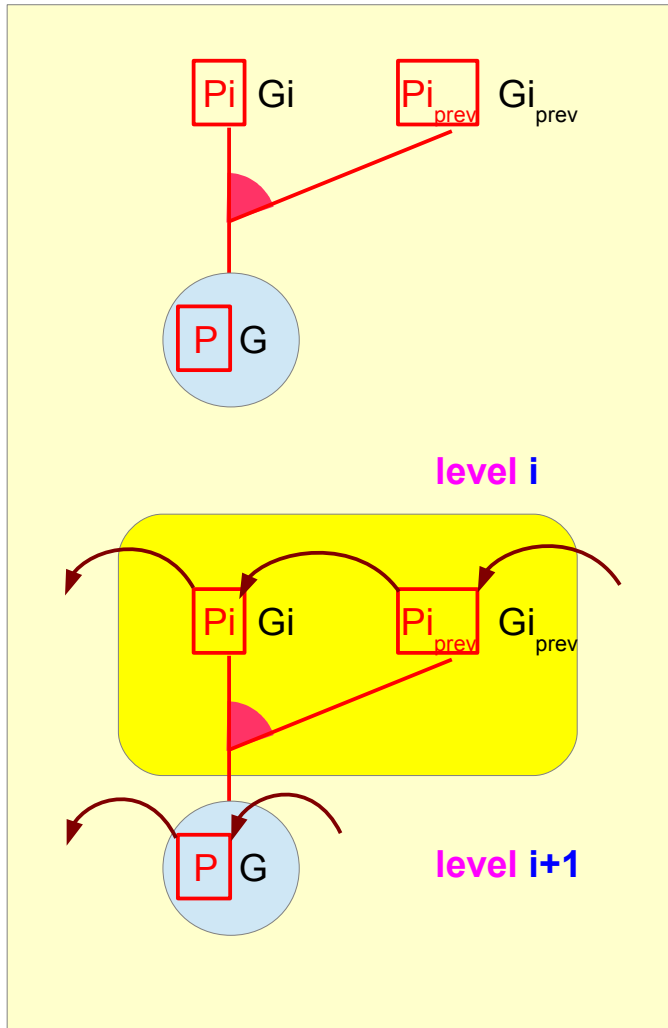


Propagate c_i

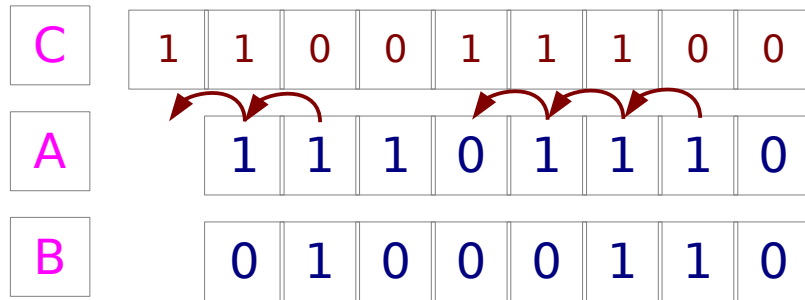
G & P at the non-leaf nodes



Computing G & P at the non-leaf nodes

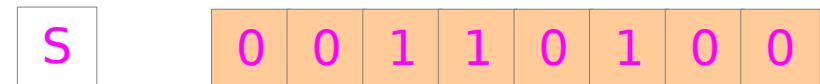
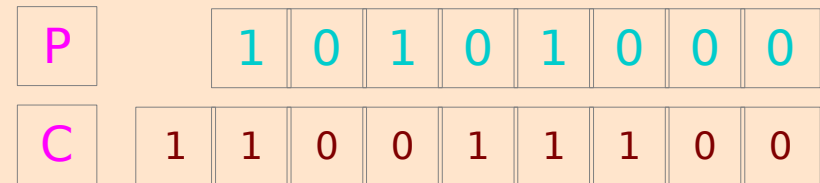
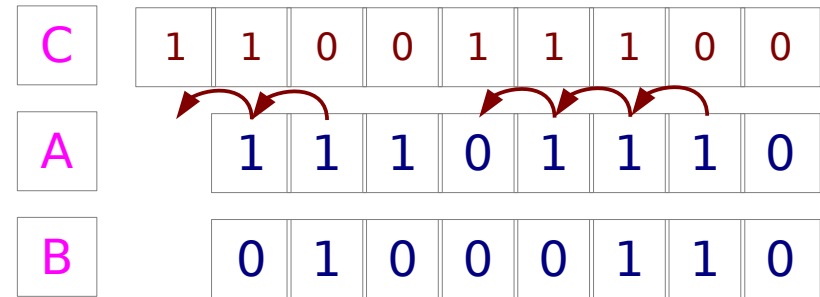
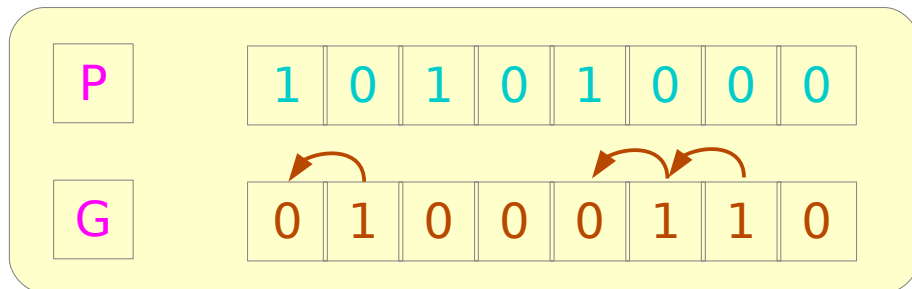


Example 1: G & P at the leaf level



$$P_i = a_i \oplus b_i$$

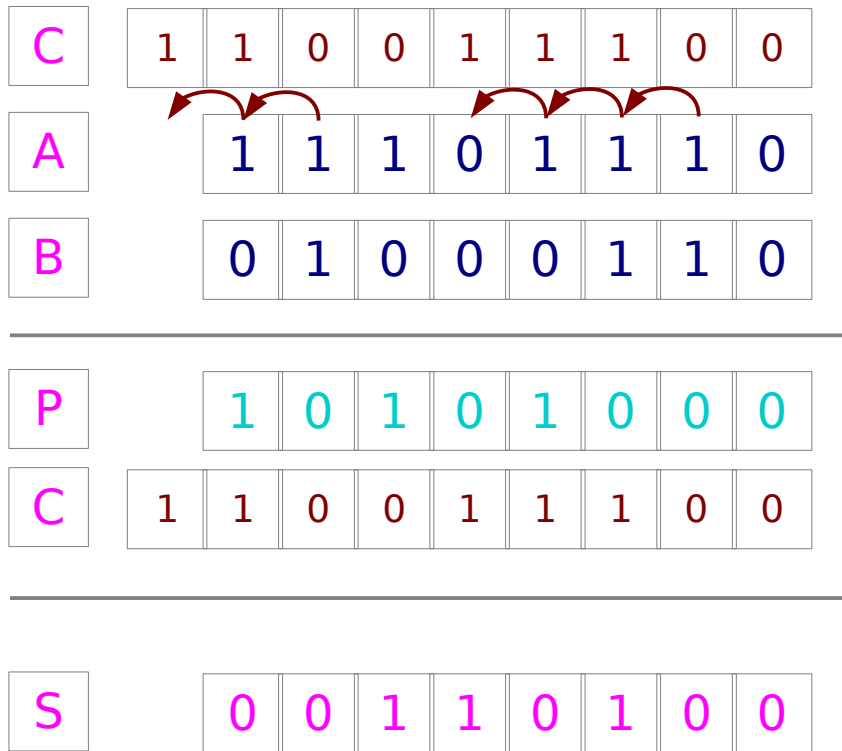
$$G_i = a_i \cdot b_i$$



$$c_{out} = G_i + P_i c_i$$

$$S_i = P_i \oplus c_i$$

Example 1: Computing (C_8, S_7)

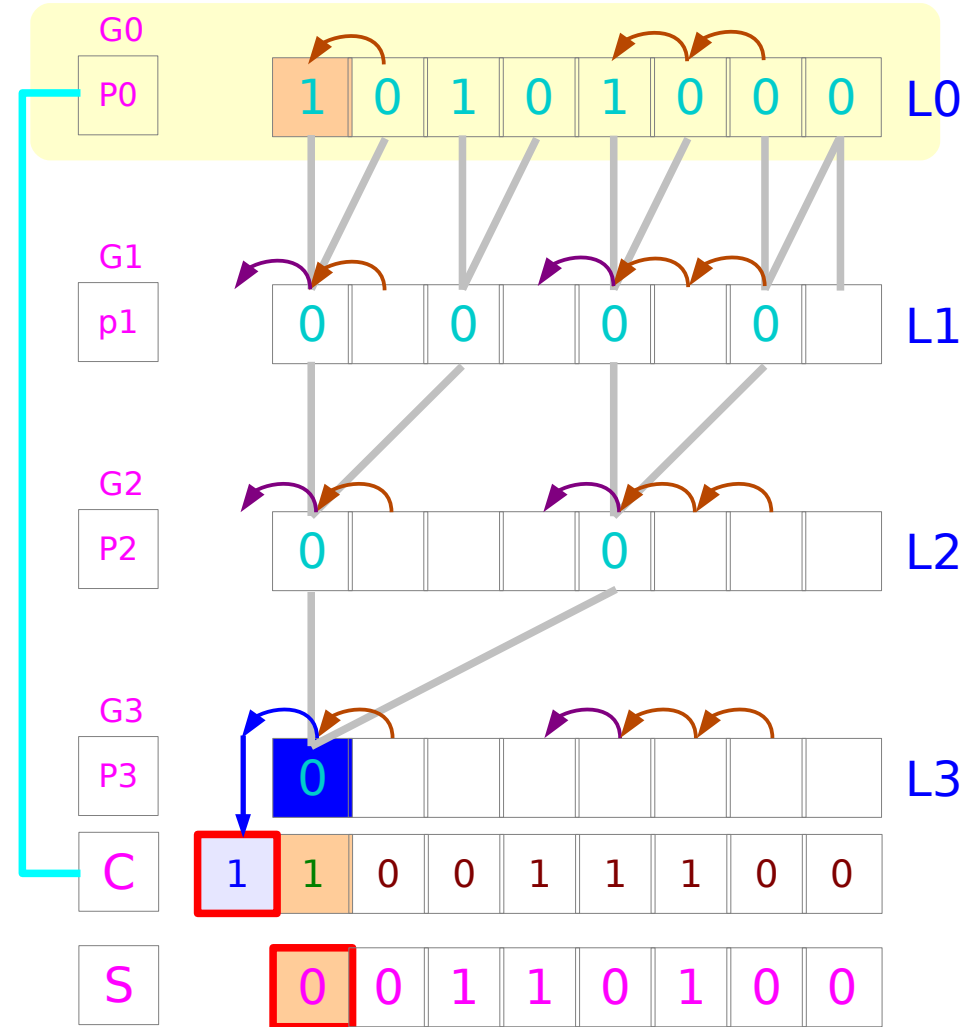


$$P_i = P_i \cdot P_{i-s}$$

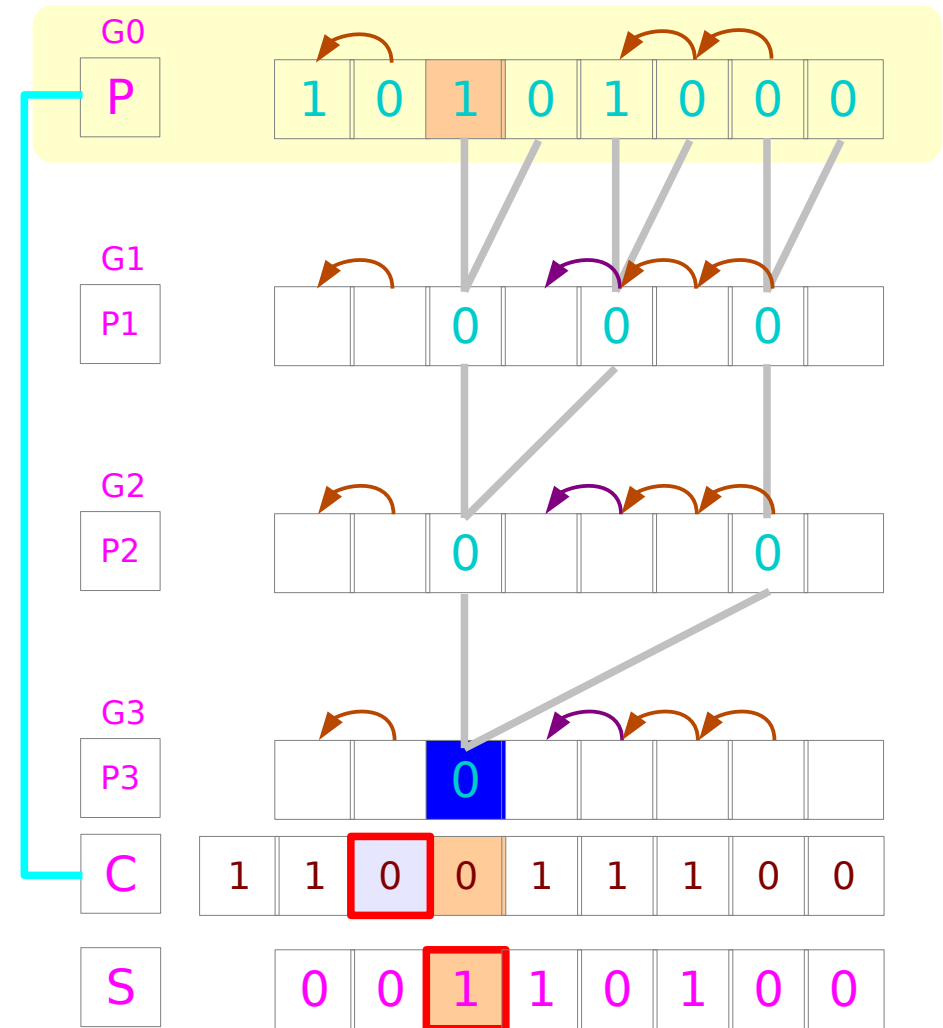
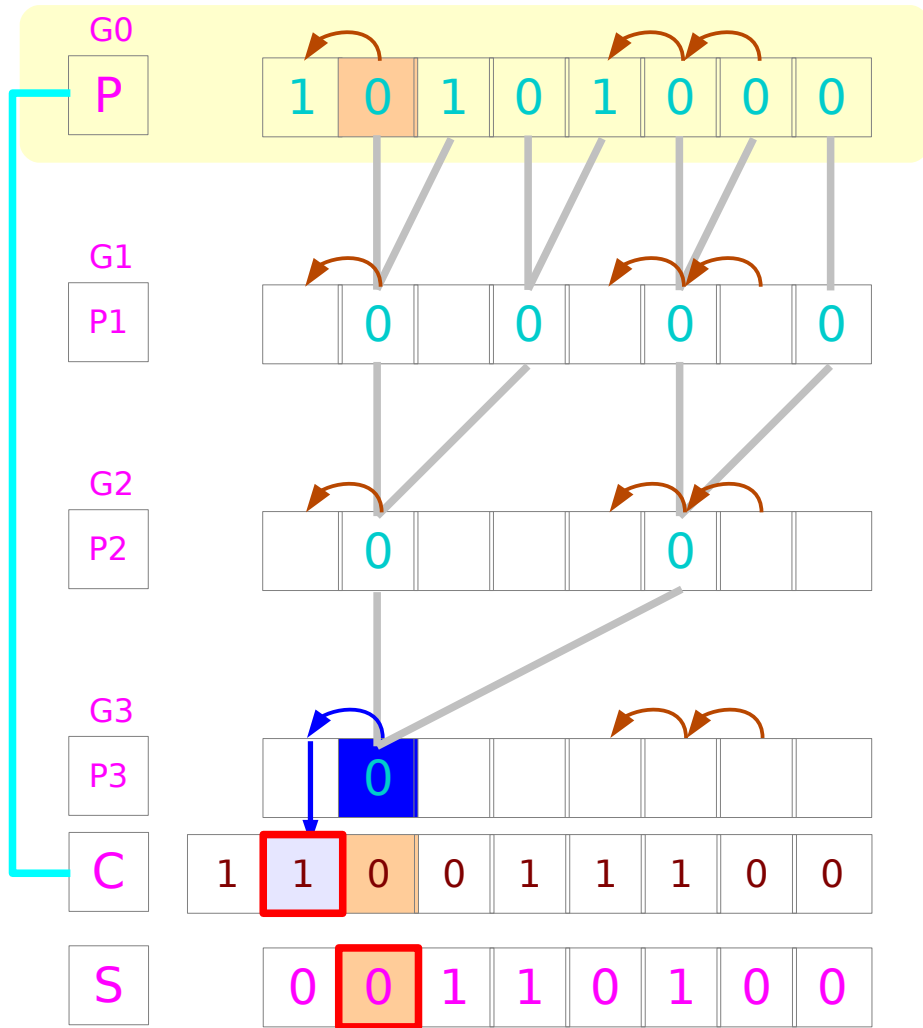
$$G_i = P_i \cdot G_{i-s} + G_i$$

$$C_{i+1} = G_i$$

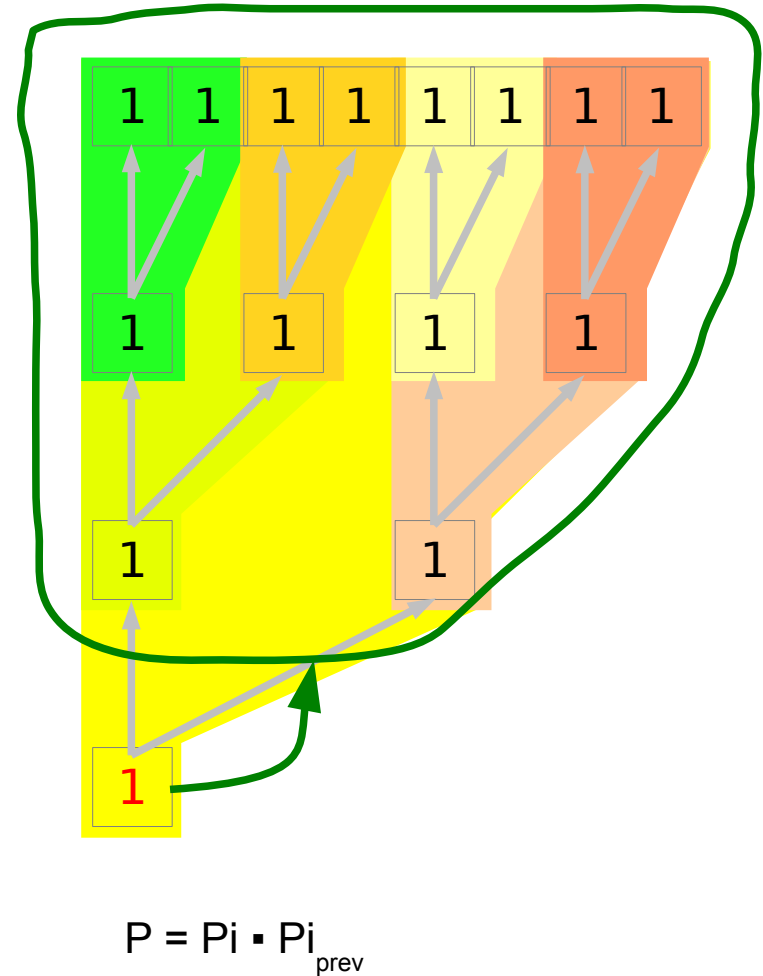
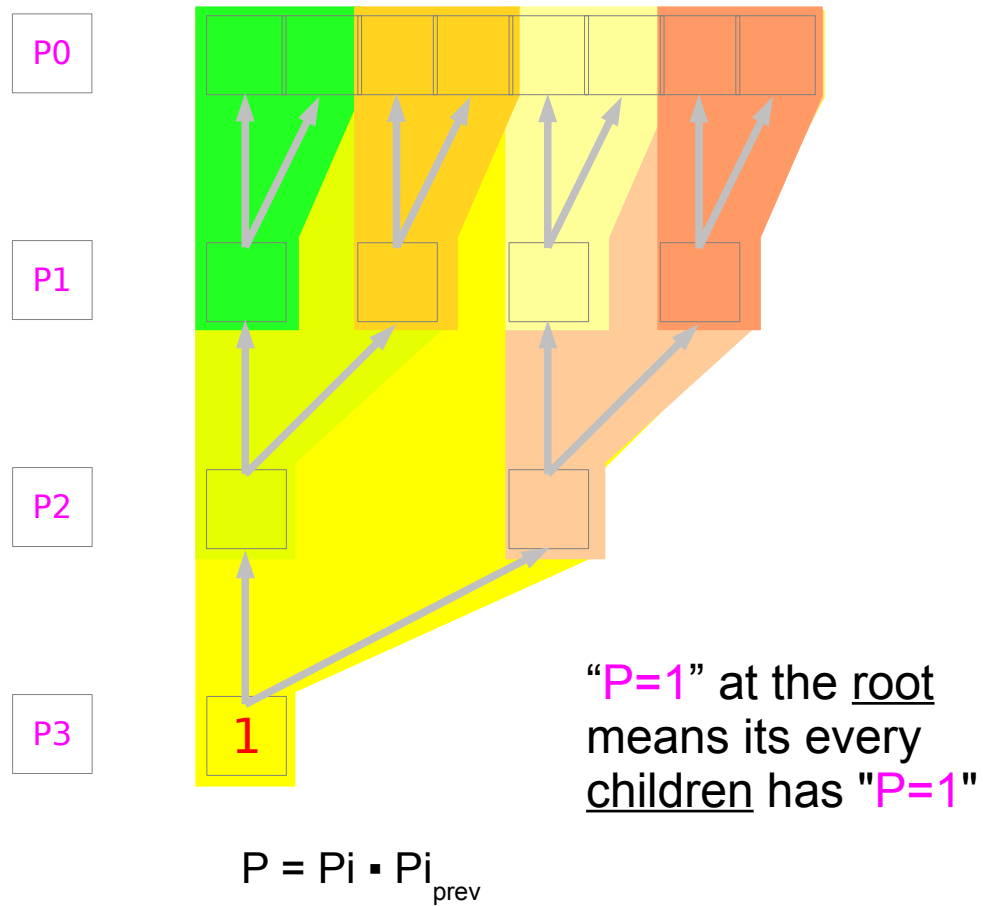
$$S_i = P0_i \cdot C_i$$



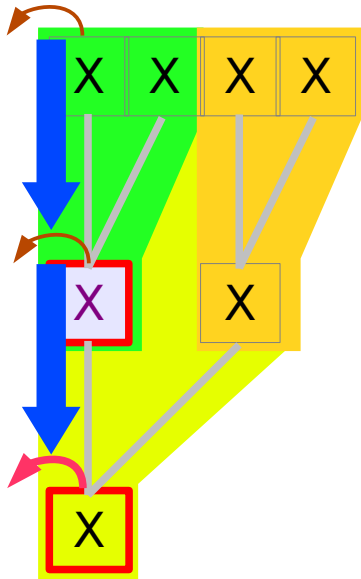
Example 1: Computing (C_7, S_6) & (C_6, S_5)



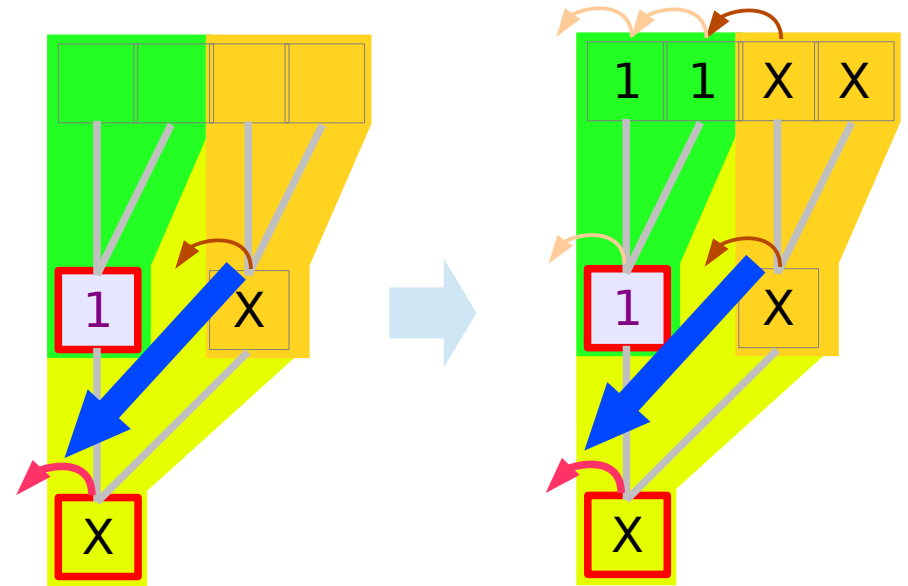
P=1 at the root



G=1 at the root

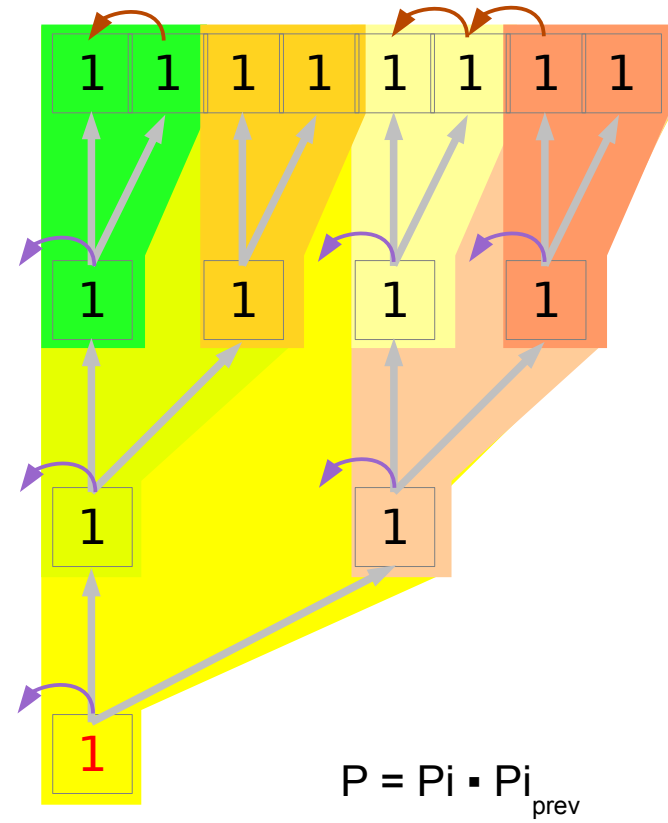
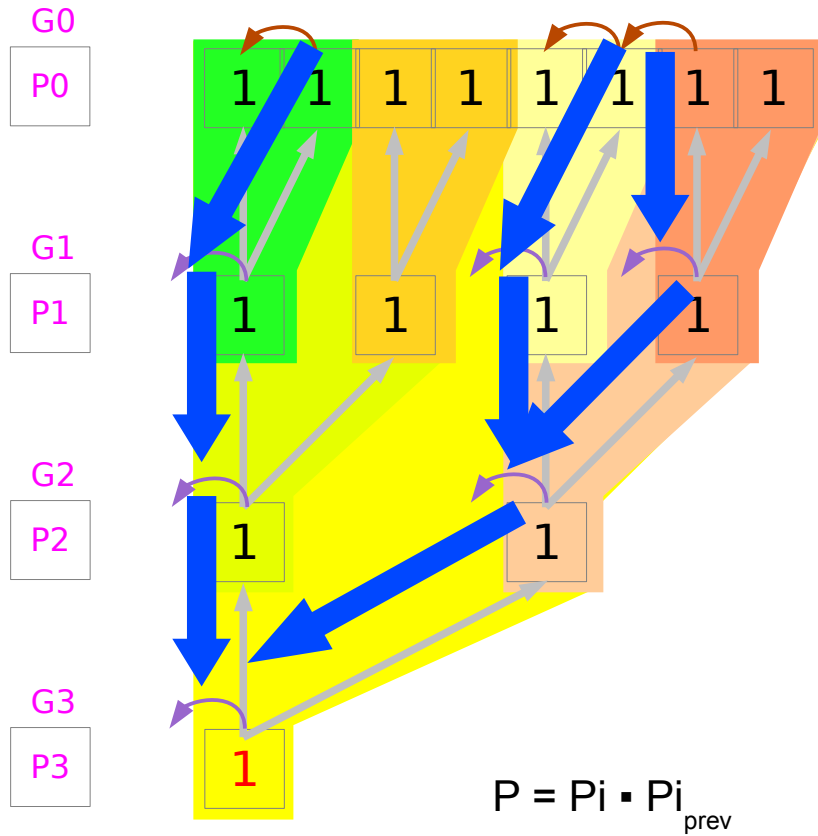


$$G = P_i \cdot G_{i_{\text{prev}}} + G_i$$

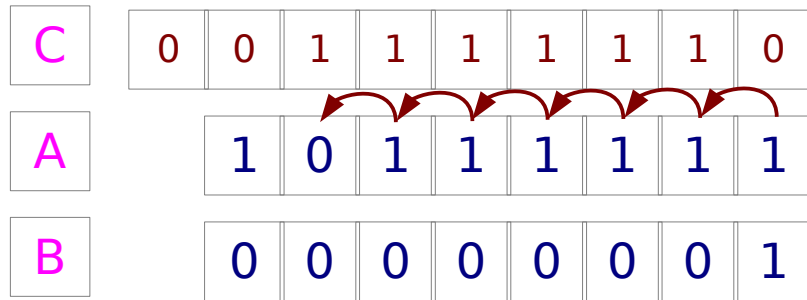


$$G = P_i \cdot G_{i_{\text{prev}}} + G_i$$

G, P=1 at the root example

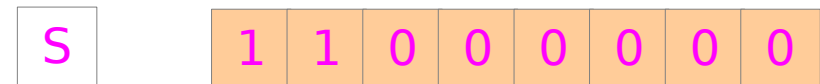
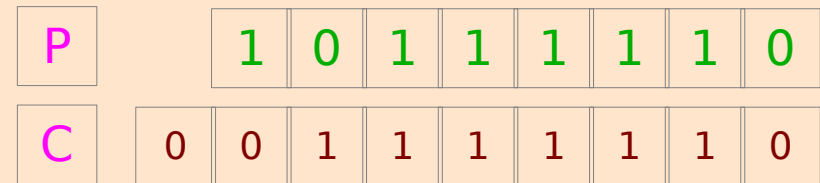
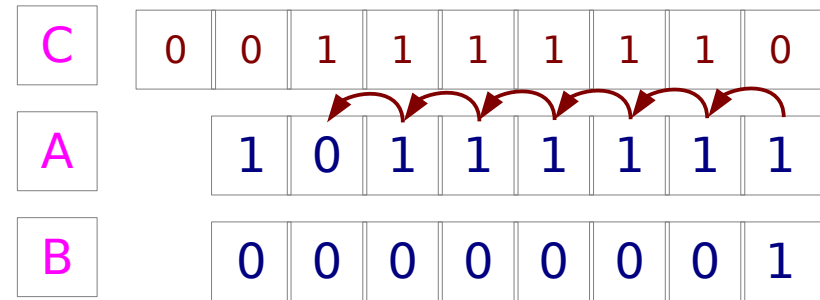
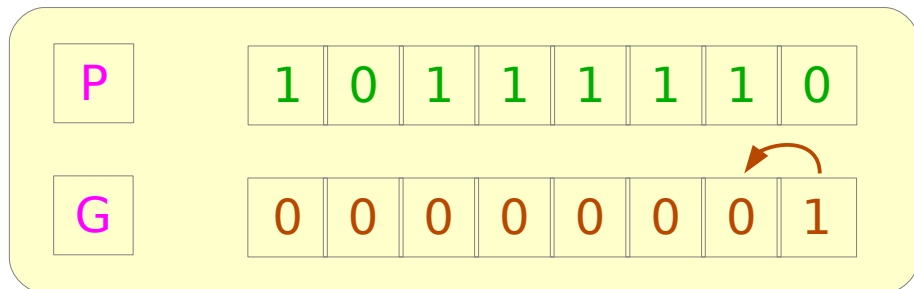


Example 2: G & P at the leaf level



$$P_i = a_i \oplus b_i$$

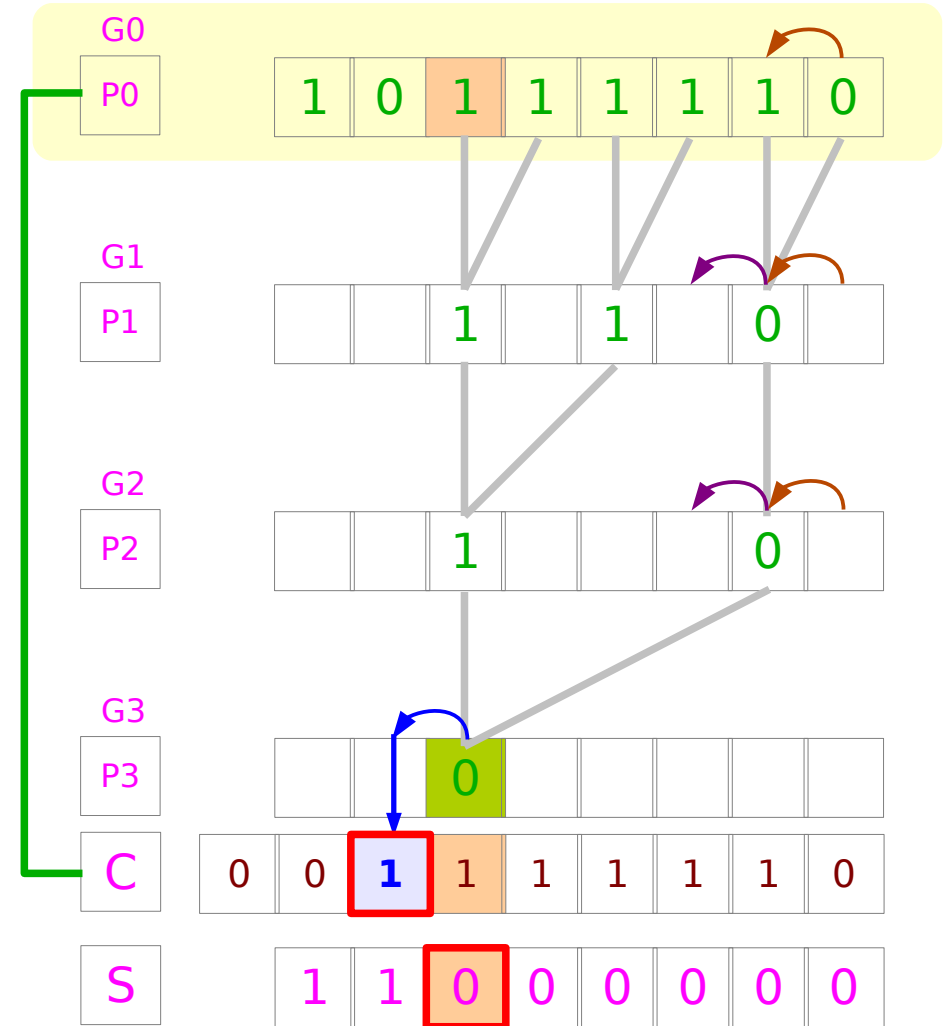
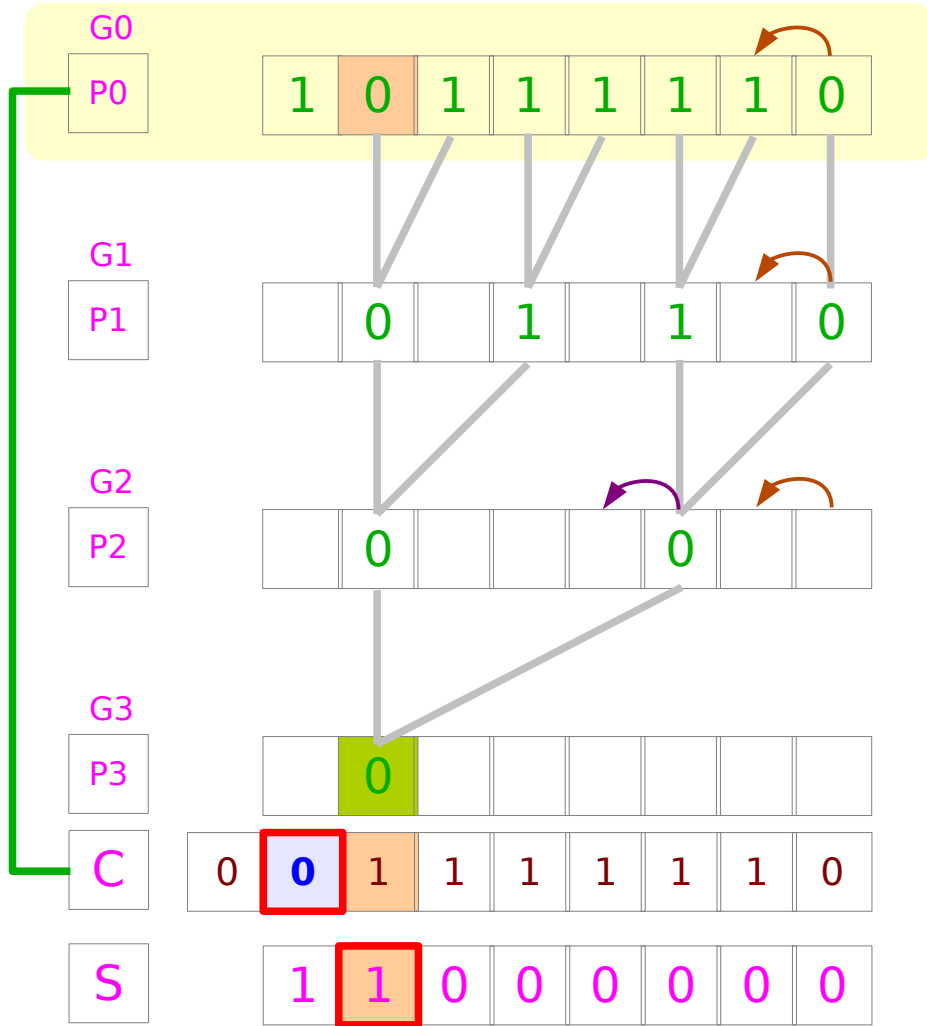
$$G_i = a_i \cdot b_i$$



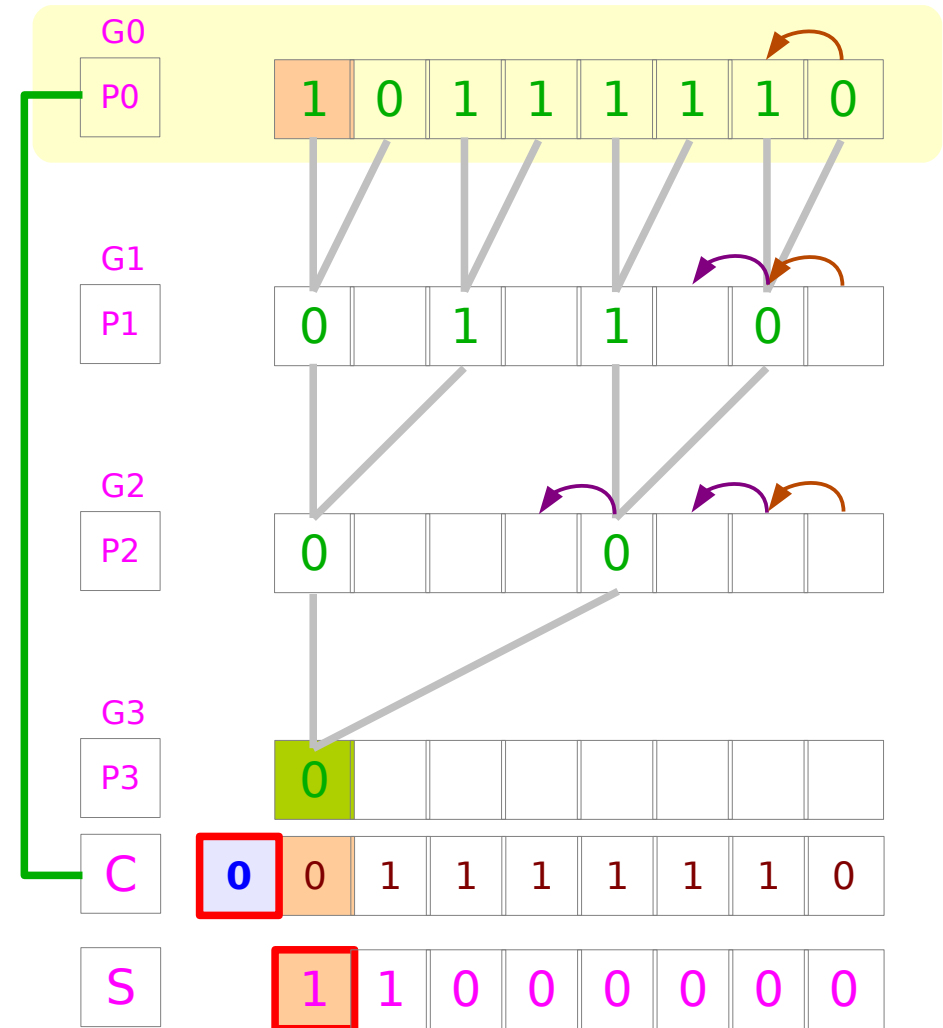
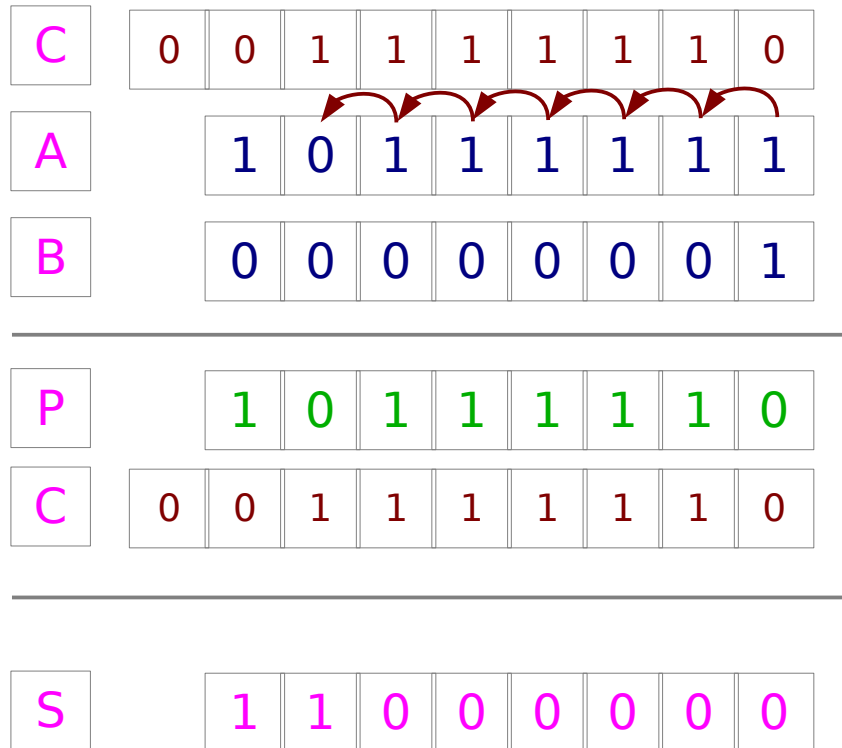
$$c_{out} = G_i + P_i c_i$$

$$S_i = P_i \oplus c_i$$

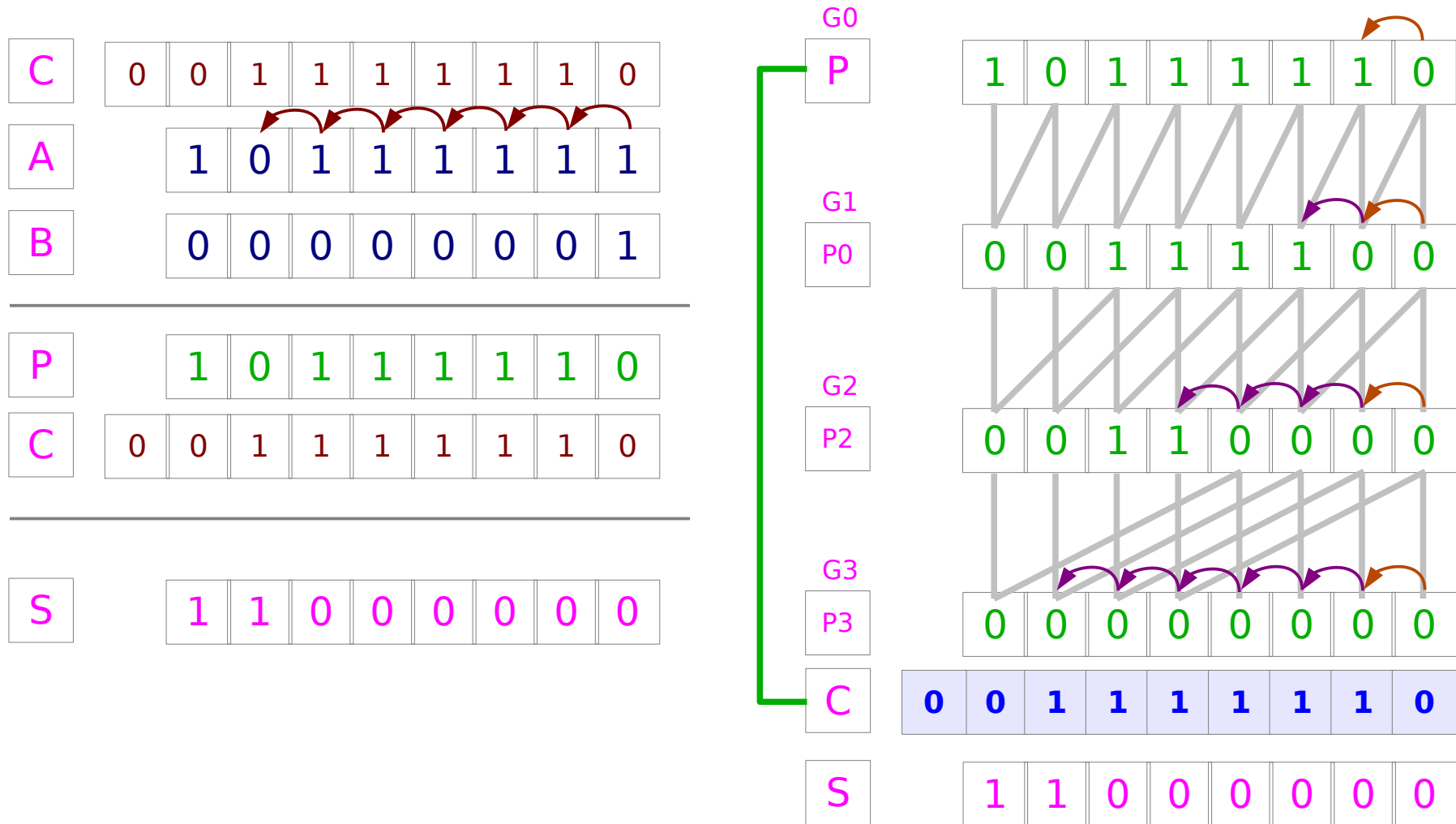
Example 2: Computing (C_7, S_6) & (C_6, S_5)



Example 2: Computing (C_8, S_7)



Example 2: Computed Carries ($C_8 \sim C_0$)



Leaf Level G & P

a	1	1	0	0
b	1	0	1	0
G				
P	0	1	1	0

$$P = P_i \cdot P_{i_{prev}}$$

$$0 = 1 \cdot 1$$

$$1 = 1 \cdot 0$$

$$1 = 1 \cdot 1$$

$$0 = 0 \cdot 0$$

$$G = P_i \cdot G_{i_{prev}} + G_i$$

$$1 = 1 \cdot 0 + 1$$

$$0 = 1 \cdot 0 + 0$$

$$0 = 1 \cdot 0 + 0$$

$$0 = 0 \cdot 0 + 0$$

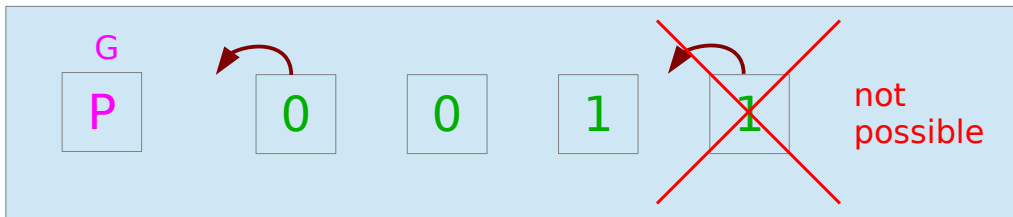
a	1	1	0	0
b	1	0	1	0
G				
P	0	1	1	0

$$1 = 1 \cdot 1 + 1$$

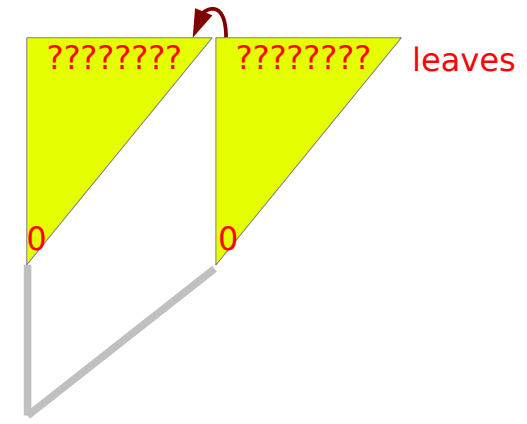
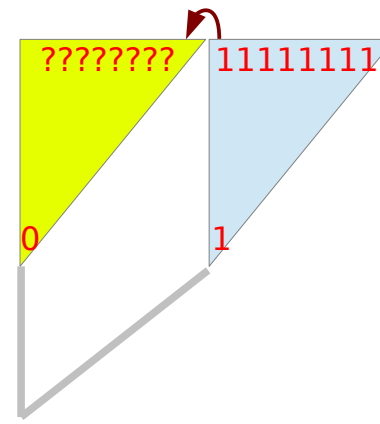
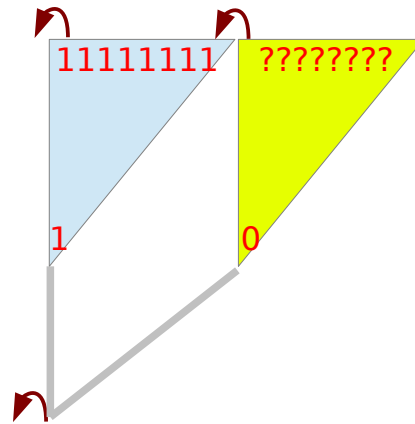
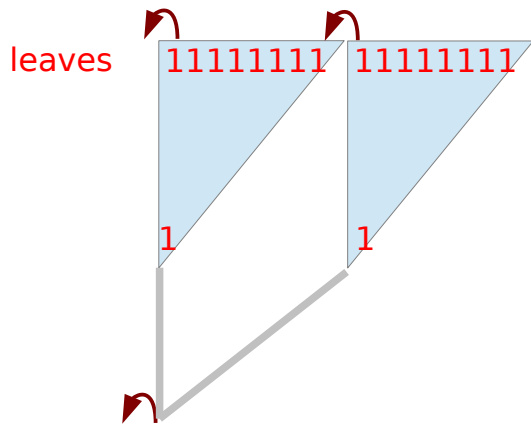
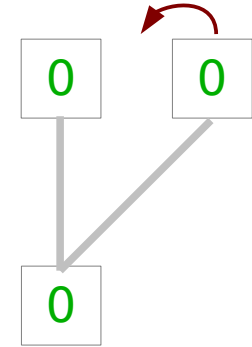
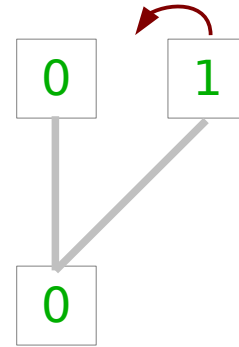
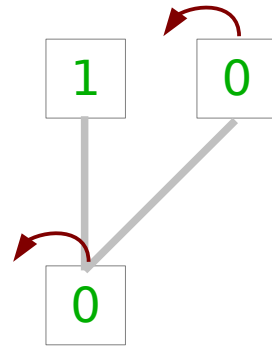
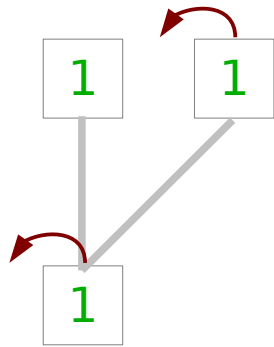
$$1 = 1 \cdot 1 + 0$$

$$1 = 1 \cdot 1 + 0$$

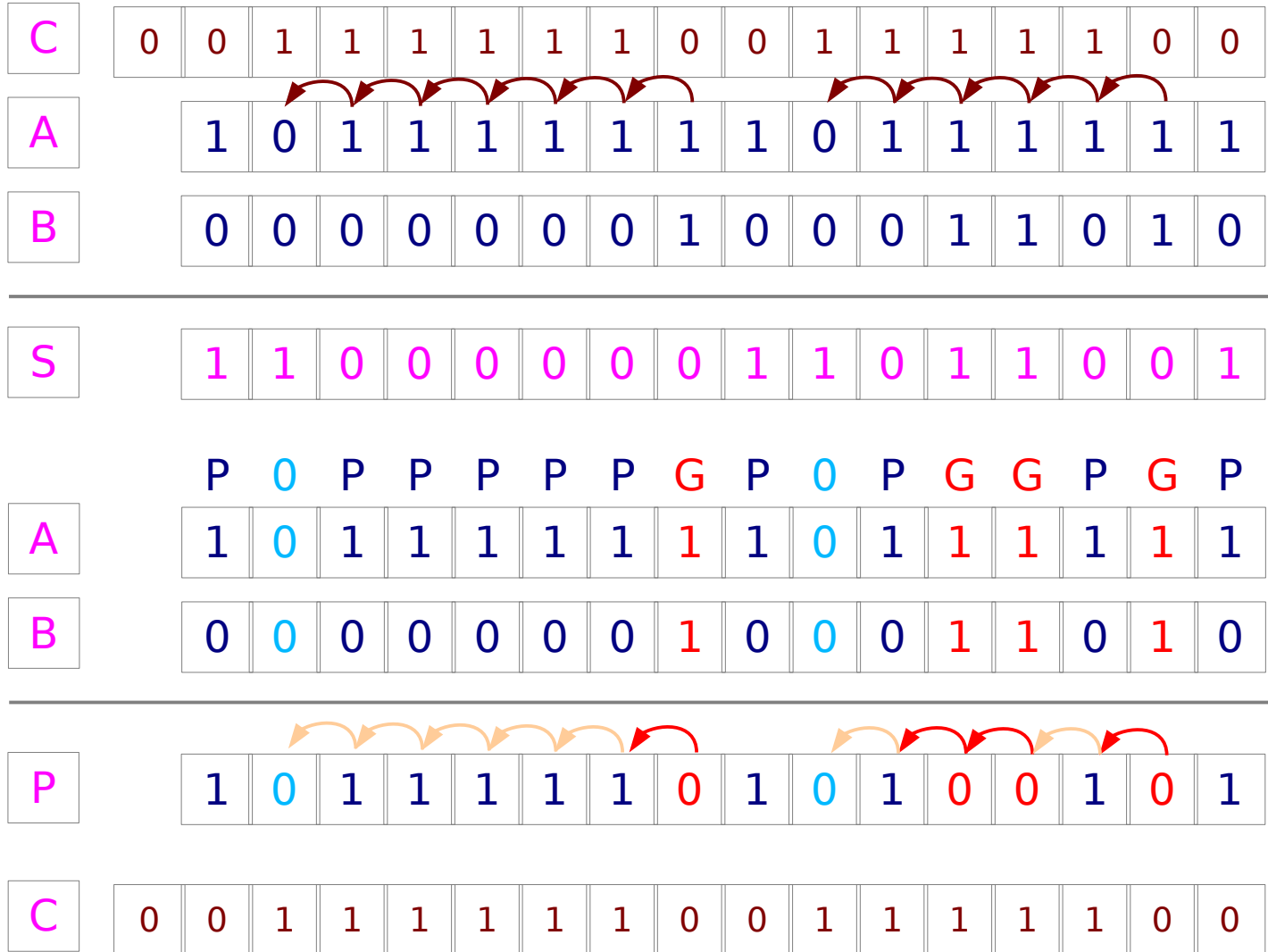
$$0 = 0 \cdot 1 + 0$$



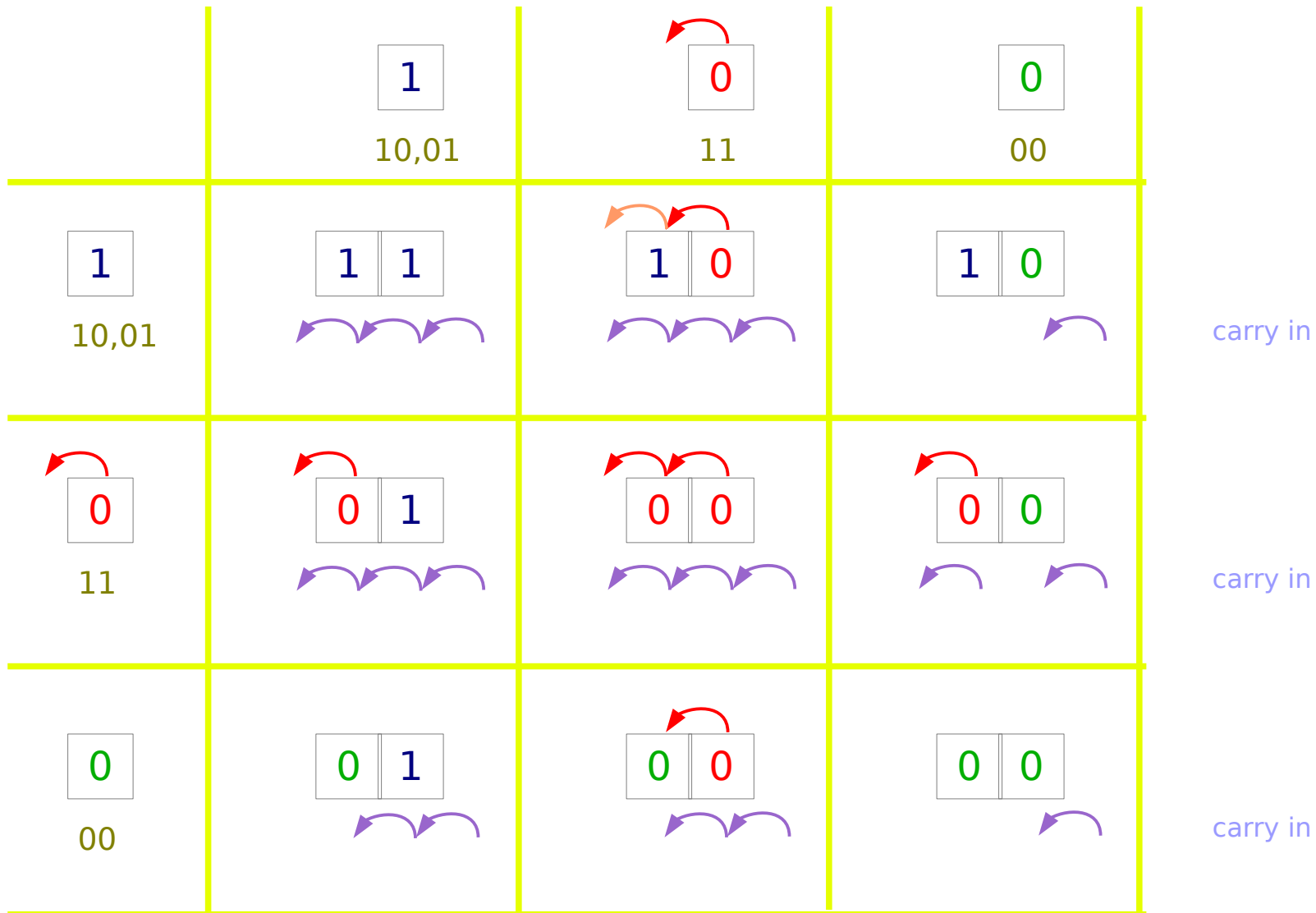
Non-Leaf Level G & P



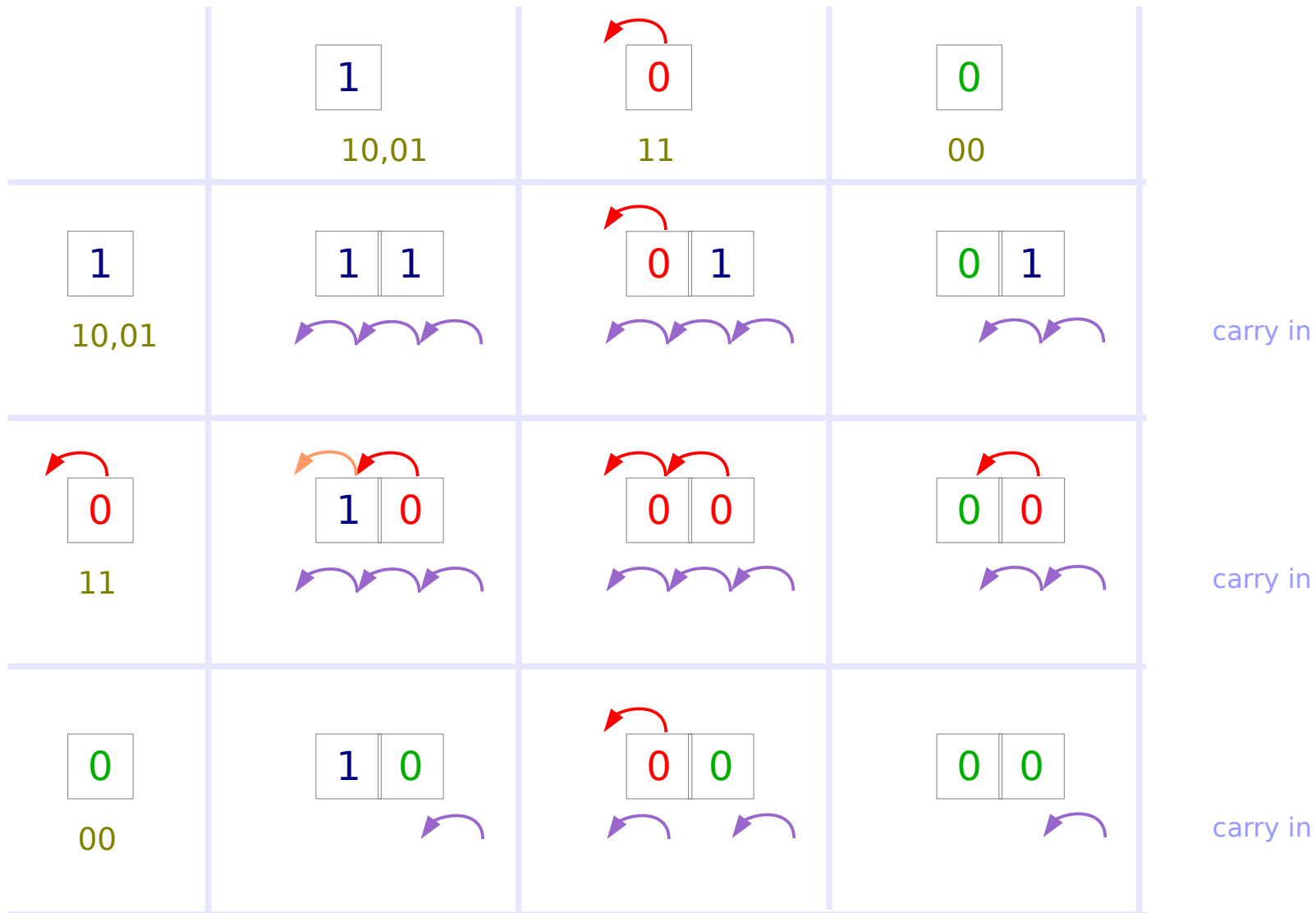
Leaf Level Carry Propagation



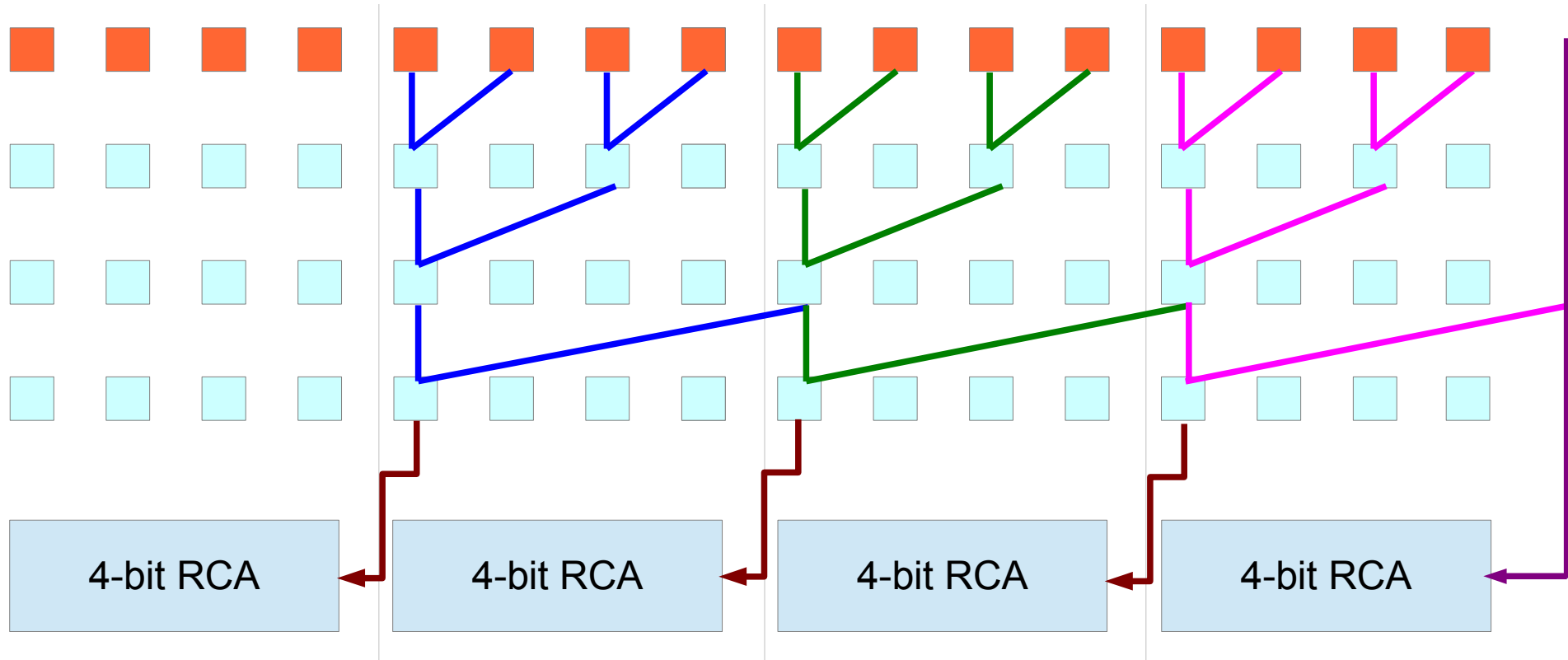
Leaf Level G and P – (1)



Leaf Level G and P – (2)



Kogge-Stone Adder with Sparsity 4



Kogge-Stone Adder with Sparsity 4

Kogge-Stone Adder with Sparsity 4



References

- [1] en.wikipedia.org
- [2] D.M. Harris, S. L. Harris, “Digital Design and Computer Architecture”
- [3] <http://www.aoki.ecei.tohoku.ac.jp/arith/mg/algorithm.html>