Carry Look-Ahead Adder

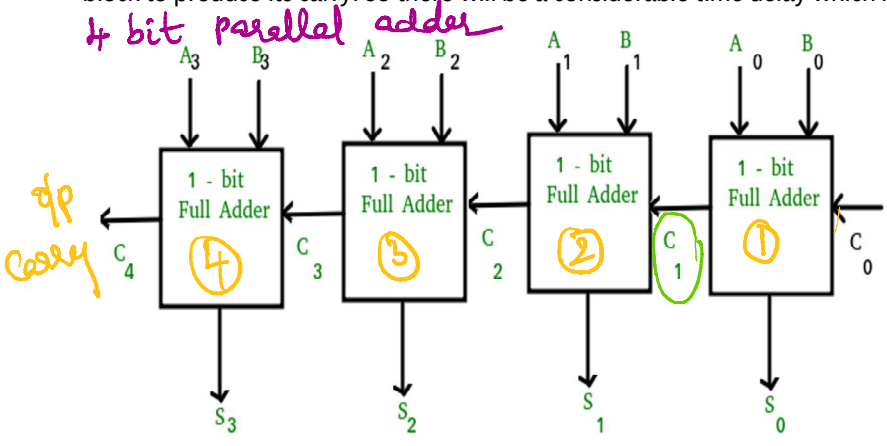
Saturday, October 10, 2020 8:56 AM

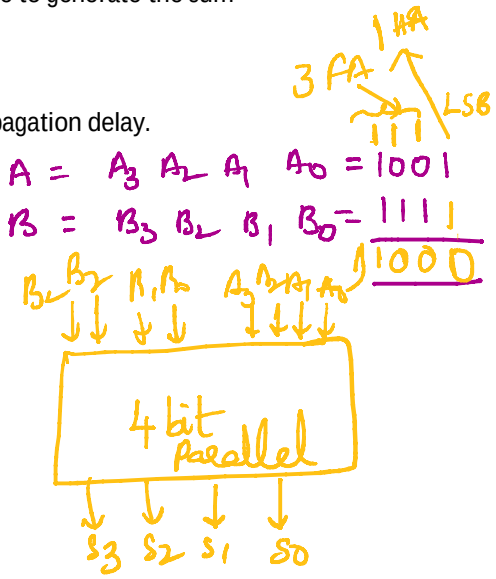
Motivation behind Carry Look-Ahead Adder :

In ripple carry adders, for each adder block, the two bits that are to be added are available instantly. However, each adder block waits for the carry to arrive from its previous block. So, it is not possible to generate the sum and carry of any block until the input carry is known. The

block waits for the

block to produce its carry. So there will be a considerable time delay which is carry propagation delay.



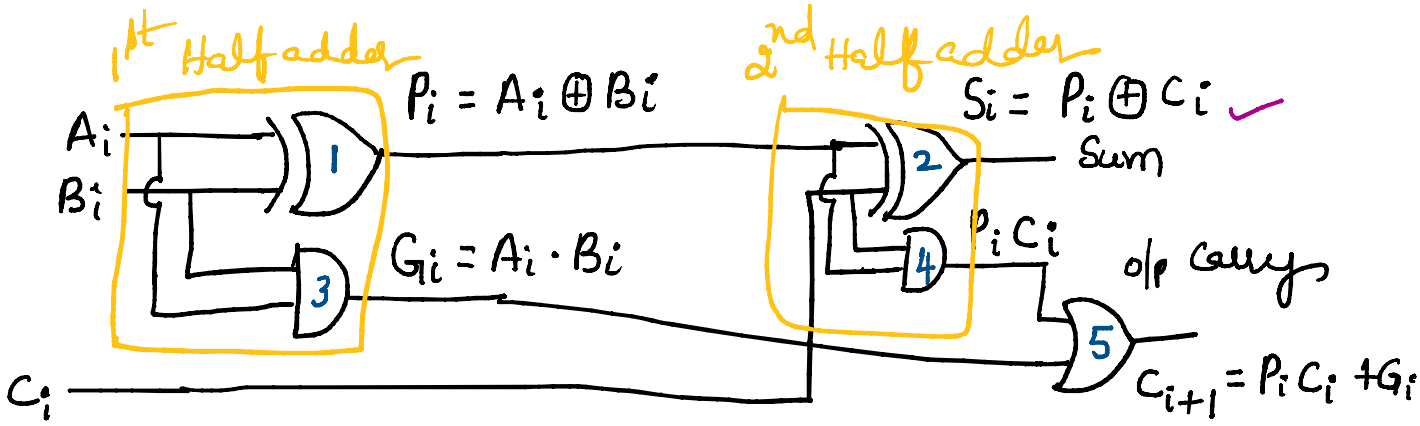


Carry Look-ahead Adder :

A carry look-ahead adder reduces the propagation delay by introducing more complex hardware. In this design, the ripple carry design is suitably transformed such that the carry logic over fixed groups of bits of the adder is reduced to two-level logic.

Carry look ahead generator

Full adder using two half adders



Carry look ahead generator

Carry propagation $(P_i) = A_i \oplus B_i$

Carry generation $(G_i) = A_i \cdot B_i$

Sum o/p $= P_i \oplus C_i$

carry o/p $C_{i+1} = P_i C_i + G_i$

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

Let $i = 1 \Rightarrow$ $C_2 = P_1 C_1 + G_1$

$i = 2 \Rightarrow$ $C_3 = P_2 C_2 + G_2$

$= P_2 (P_1 C_1 + G_1) + G_2$

$C_3 = P_2 P_1 C_1 + P_2 G_1 + G_2$

$i = 3 \Rightarrow$ $C_4 = P_3 C_3 + G_3$

$= P_3 (P_2 P_1 C_1 + P_2 G_1 + G_2) + G_3$

$C_4 = P_3 P_2 P_1 C_1 + P_3 P_2 G_1 + P_3 G_2 + G_3$

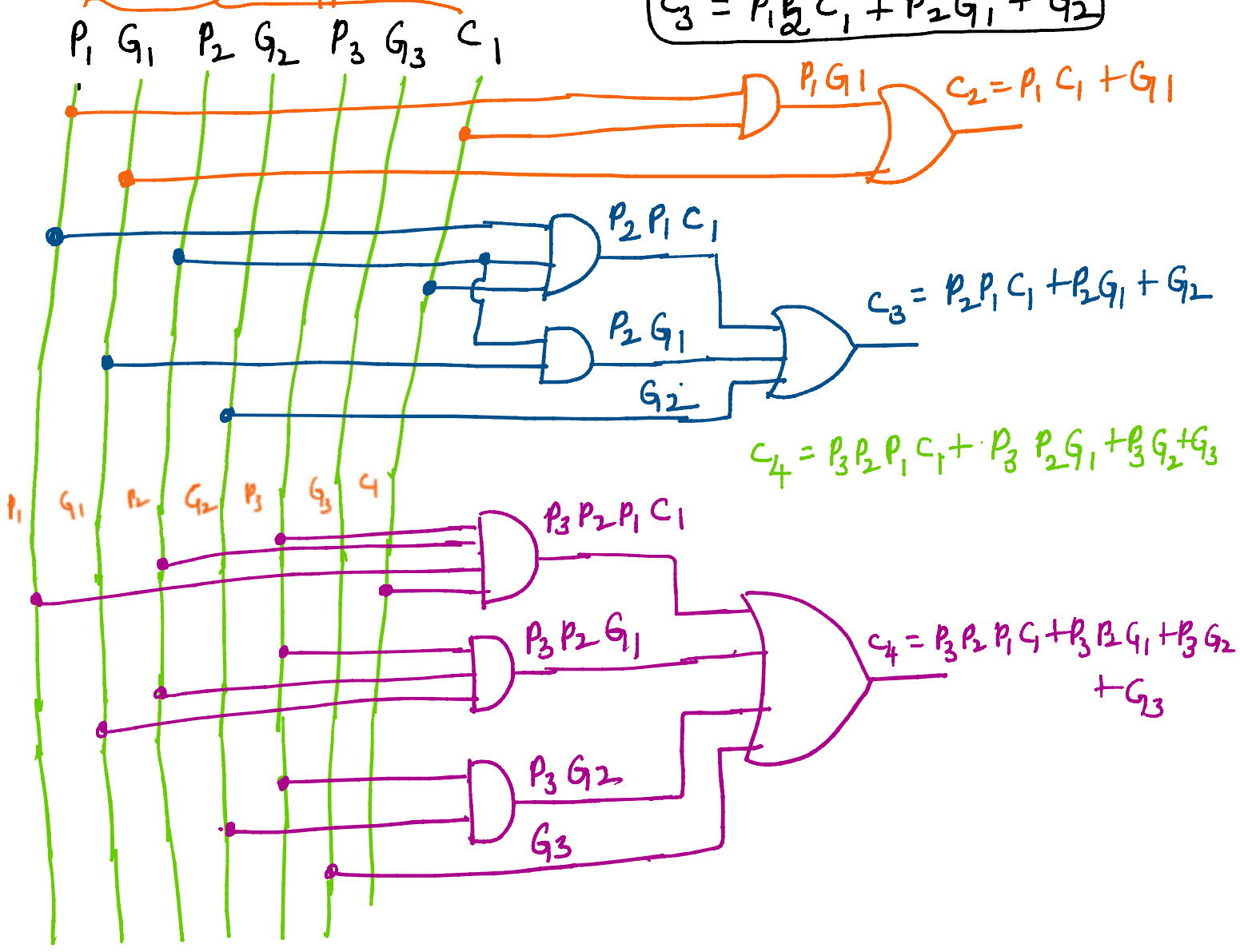
$$C_4 = P_3 P_2 P_1 C_1 + P_3 P_2 G_1 + P_3 G_2 + G_3$$

Here C_2, C_3 & C_4 depends only on C_1 thereby we can eliminate interstage propagation delay time

o/p Variables = C_2, C_3 & C_4

i/p Variables = $P_1 G_1 P_2 G_2 P_3 G_3$ & C_1

$$C_3 = P_1 P_2 C_1 + P_2 G_1 + G_2$$



4 bit parallel adder using lookahead generator

4 bit parallel adder using lookahead generator

(or)

(Look ahead adder)