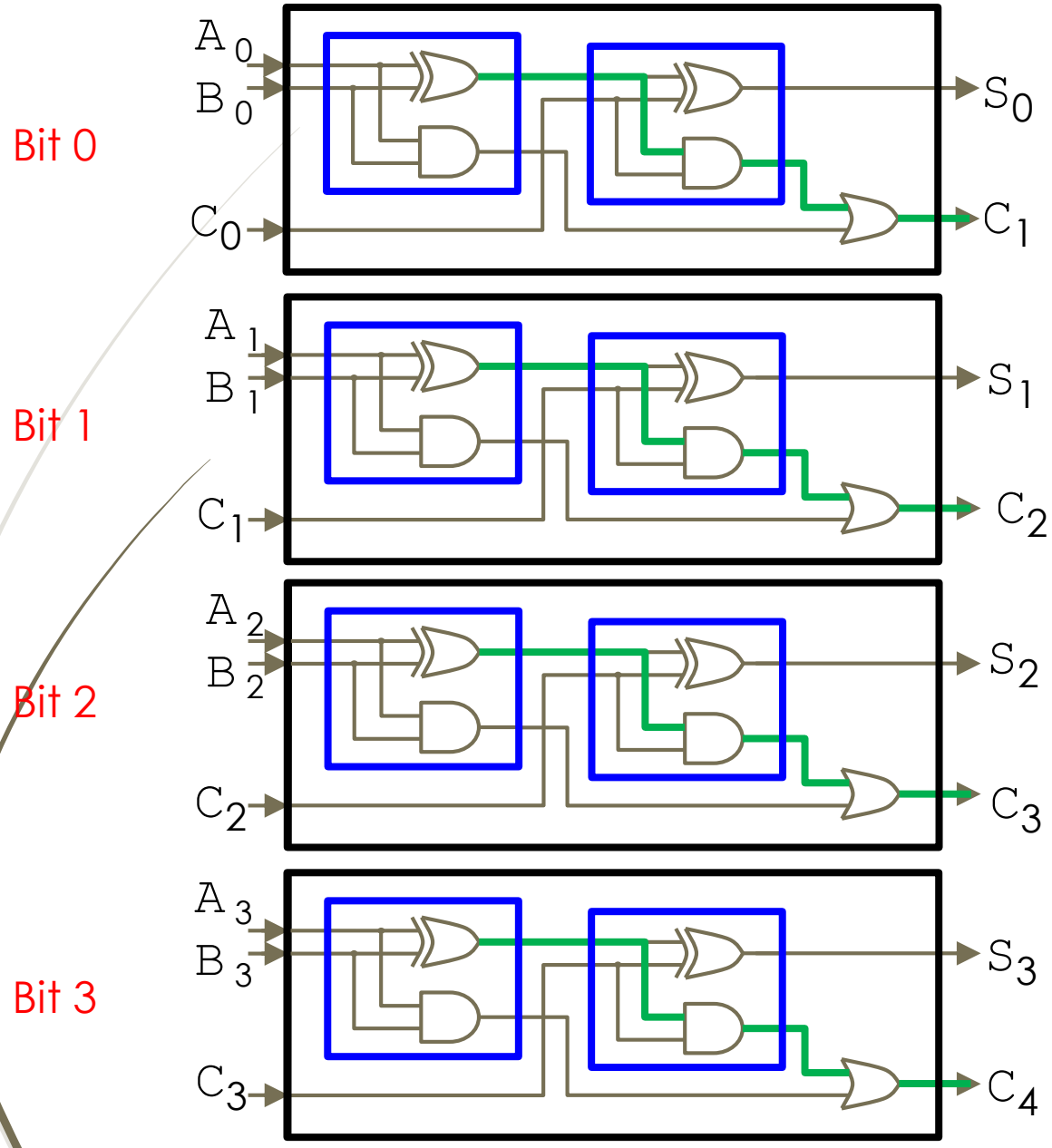
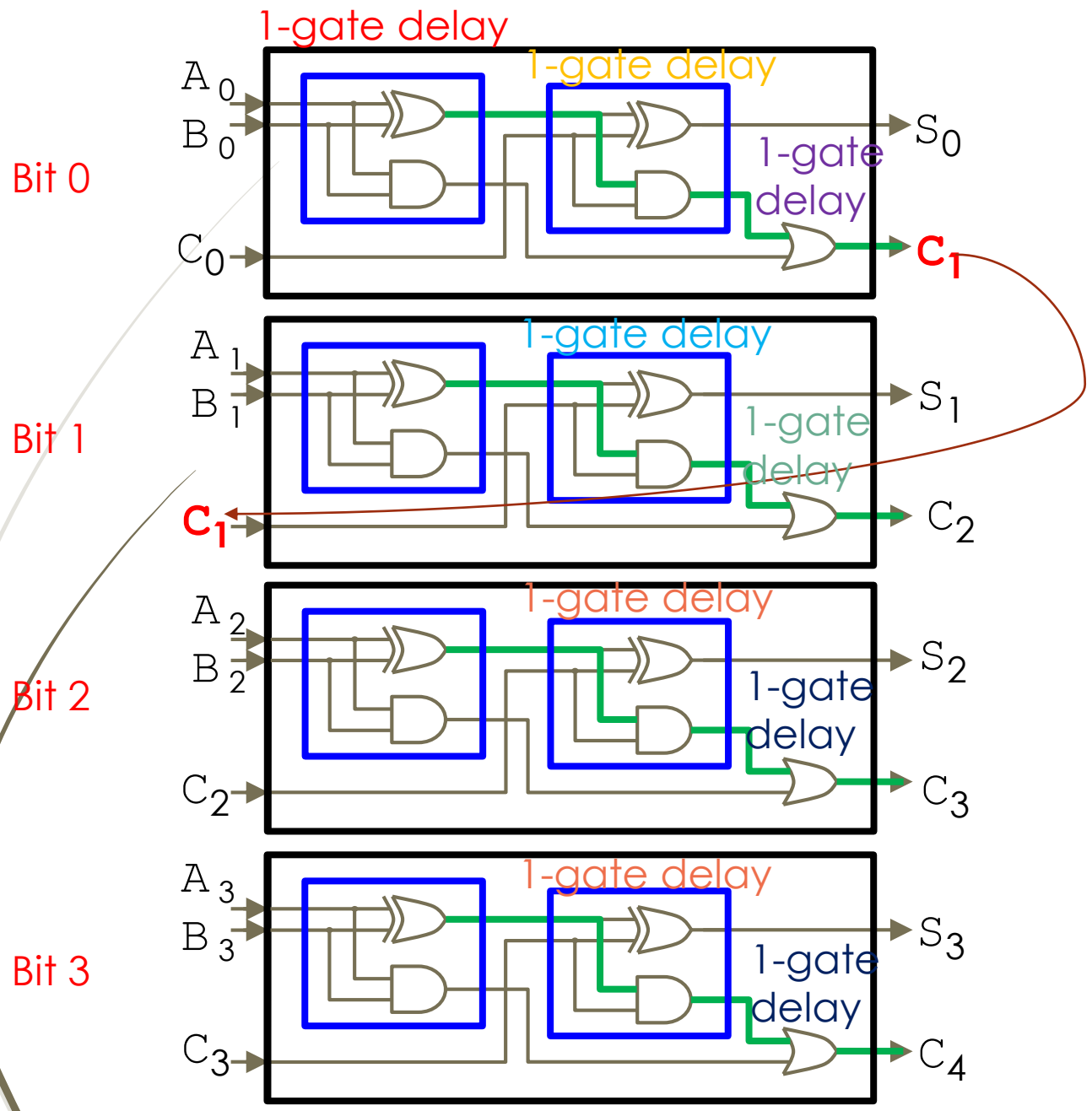
Homework



- Why is the propagation delay of this Full Adder 4 (also known as a 4-bit (w-bit) ripple carry adder) 9 gates (2w + 1 gates) where w is the number of bits?

Homework solution



Answer:
4 (n) * 2-gate delay +1 = 9-gate delay

Explanation:
Imagine you are adding 1000 (8)
+ 1001 (9)
10001

So, $A_3 = 1$, $A_2 = 0$, $A_1 = 0$, $A_0 = 0$,
$B_3 = 1$, $B_2 = 0$, $B_1 = 0$, $B_0 = 1$,
$S_3 = 0$, $S_2 = 0$, $S_1 = 0$, $S_0 = 1$ and
$C_4 = 1$, $C_3 = 0$, $A_2 = 0$, $A_1 = 0$, $C_0 = 0$.

This signifies that the 4 bits of A and B are coming into the first half-adder at the same time and producing the output in 1-gate delay for all 4 full-adders at the same time. This is the "+1".

Looking at the **Bit 0 full-adder**, this output (from first half-adder) becomes an input, along with C_0, to the second half-adder, which produces S_0 in 1-gate delay. One of its output becomes the input to the OR gate and produces C_1 in 1 gate delay. So, it takes 1-gate delay + 1-gate delay + 1-gate delay to produces C_1.

In the meantime, the second half-adder of **Bit 1 full-adder** is waiting for C_1 in order to produce S_1 and C_2. So, it takes 1-gate delay + 1-gate delay + 1-gate delay + 1-gate delay + 1-gate delay to produces C_2.

In the meantime, the second half-adder of **Bit 2 full-adder** is waiting for C_2 in order to produce S_2 and C_3. So, it takes 1-gate delay + 1-gate delay + 1-gate delay + 1-gate delay + 1-gate delay + 1-gate delay + 1-gate delay to produces C_3.

Finally, the second half-adder of **Bit 3 full-adder** is waiting for C_3 in order to produce S_3 and C_4 and this will also take 2-gate delay.

To conclude, there is the initial 1-gate delay associated with the first half-adder which occurs at the same time for each full-adder, then, for each full-adder, there is 2-gate delay, serially (one after the other). Since we have n = 4 full-adders, the total delay will be 1 + 4 * 2-gate delay = 9-gate delay.

Homework solution