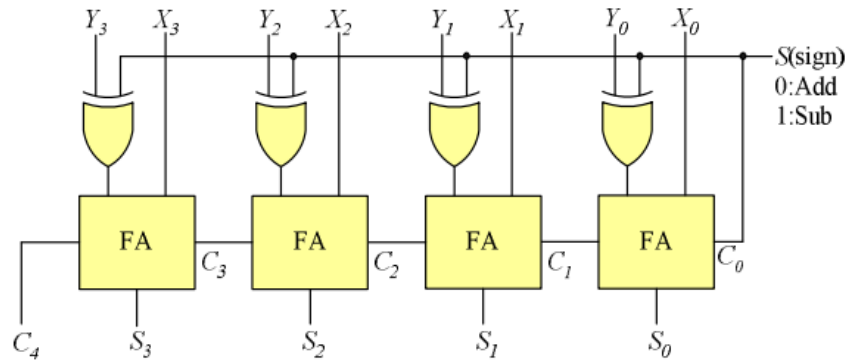


## 문제 1. CLA-4bit Parallel Adder/Subtractor

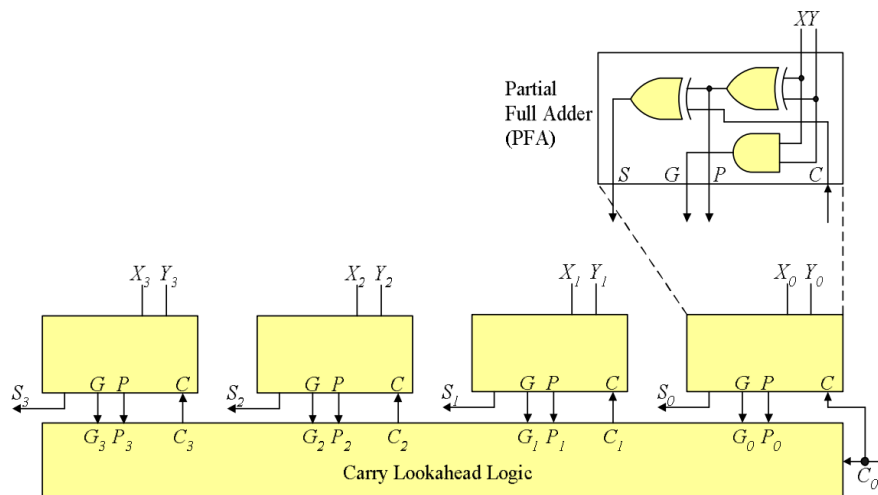
### 1-1. RCA 가감산기의 속도 문제



<RCA 4 비트 병렬 가감산기>

전가산기 4 개를 사용하여 만든 RCA 4 비트 병렬 가감산기를 고려하자, 이 가감산기는  $X_1 \sim X_4$ ,  $Y_1 \sim Y_4$ 의 입력을 통해 가감산처리를 할 때 올림(Carry) 부분을 순차적으로 계산한다. 즉 각 단계의 전가산기에서  $S_n$ 을 구하기 위해서는,  $C_{n-1}$ 의 연산까지가 모두 완료되어야  $S_n$ 의 연산 결과를 얻을 수 있다. 전가산기  $n$  개를 지날 때마다 지연 시간이 점점 늘어나게 되며, 4 비트 이상의 가산기에서는 지연이 더욱 심해질 것이다. 이것은 빠른 연산이 요구되는 시스템에서 심각한 문제로 작용할 수 있다.

### 1-2. CLA 가감산기



<CLA 4 비트 병렬 가산기와 부분가산기>

CLA 가감산기는 올림(Carry) 부분을 예측 처리하여 RCA 가감산기의 가산 부분을 처리하는 전가산기 부분을 병렬 연산이 가능한 부분전가산기(Partial Full Adder)로 대체하여 만든 고속 가감산기이다.

### 1-3. CLA 가감산기의 원리와 구현

CLA 의 원리는  $X_i$  와  $Y_i$  가 모두 1 일 때, 또는  $C_i$  가 1 이고  $X_i, Y_i$  의 배타적 논리합이 1 일때 올림이 발생하는 점을 이용한 것이다.

$$C_{out} = C_{i+1} = X_i Y_i + (X_i \oplus Y_i) C_i$$

위 식에서  $X_i, Y_i$  를  $G_i, X_i \text{ xor } Y_i$  를  $P_i$  라고 하면

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

위와 같이 표현할 수 있다.

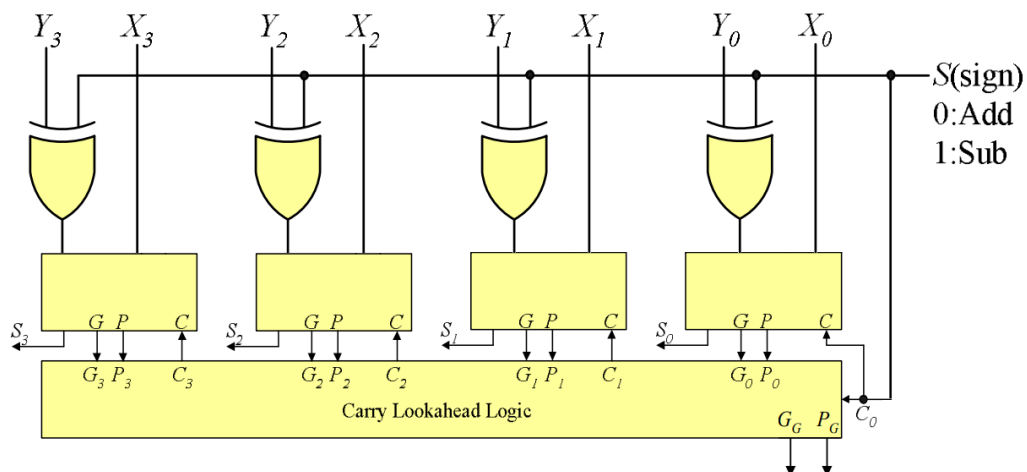
이 원리를 이용하여 4 비트 가감산기에서 사용될  $C_1 \sim C_4$  (Carry, 올림),  $S$  (Sum, 합)을 구해보면 다음과 같다.

$$\begin{aligned} C_1 &= G_0 + P_0 C_0 \\ C_2 &= G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0) = G_1 + P_1 G_0 + P_1 P_0 C_0 \\ C_3 &= G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0) = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0 \\ C_4 &= G_3 + P_3 C_3 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0) = G_3 + P_3 G_2 + P_3 P_2 G_1 + \\ &\quad + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0 \\ S_i &= X_i \oplus Y_i \oplus C_i = P_i \oplus C_i \end{aligned}$$

또한 최종 가감산기 올림수를 구하기 위해  $P_G, G_G$  를 정리하면 다음과 같다.

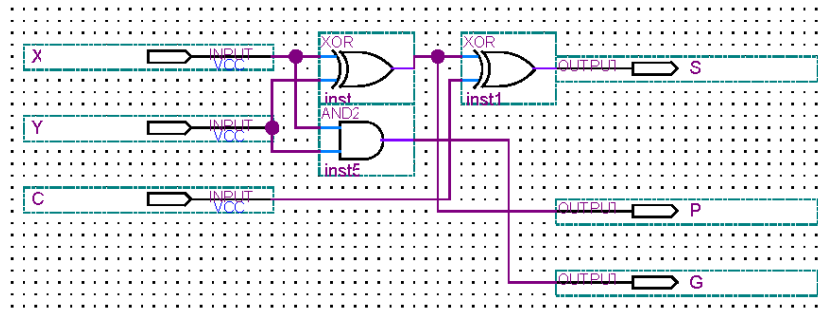
$$P_G = P_3 P_2 P_1 P_0, \quad G_G = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0$$

감산 기능을 구현하기 위해  $Y$  입력에 대해  $S$  (Sign, 부호) 입력과 XOR 연산한 값을 부분 가산기에 연결하고, 위의 식을 이용하여 CLA 4 비트 가감산기를 도식화하면 다음과 같다.

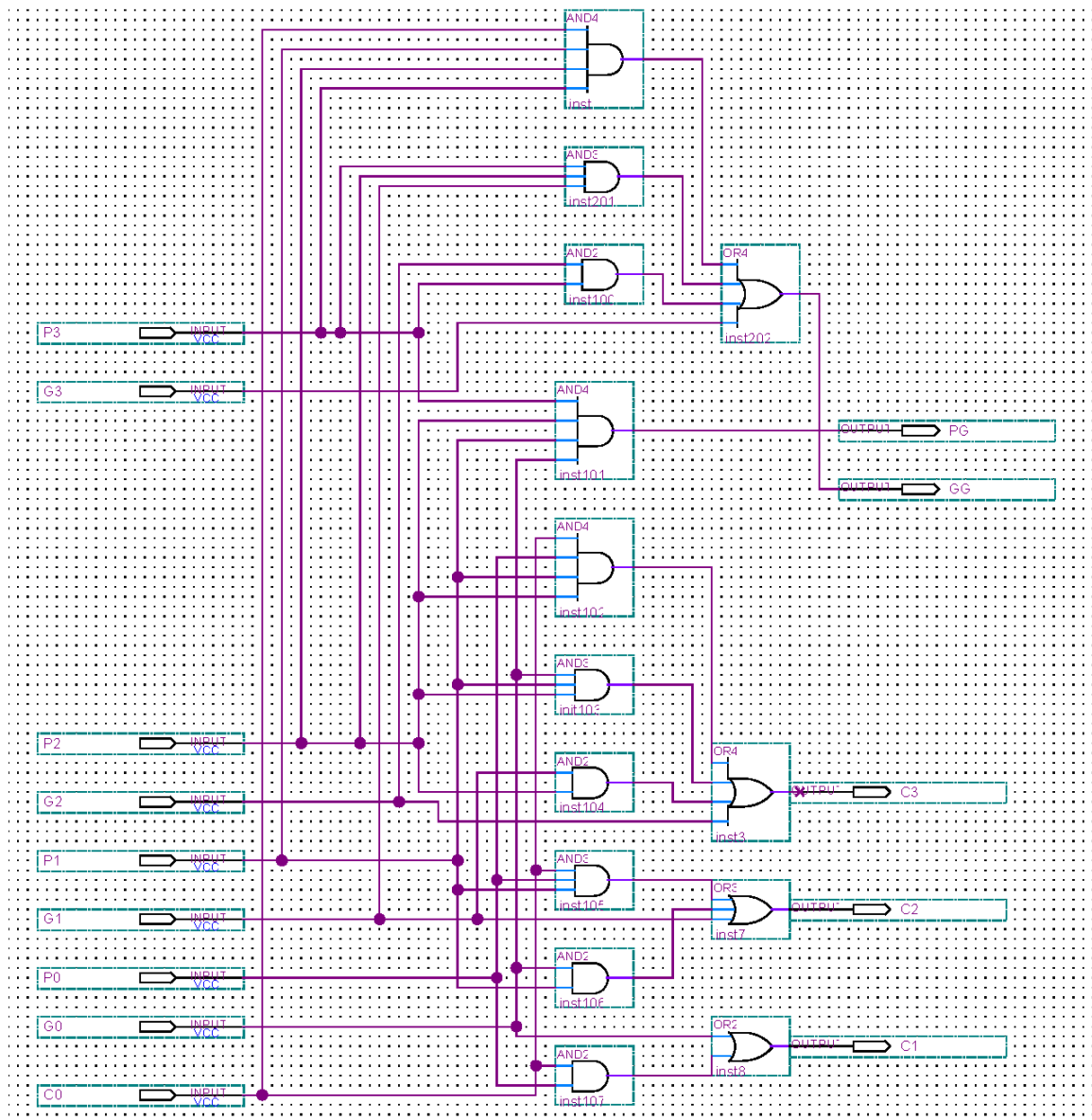


<CLA 4 비트 가감산기>

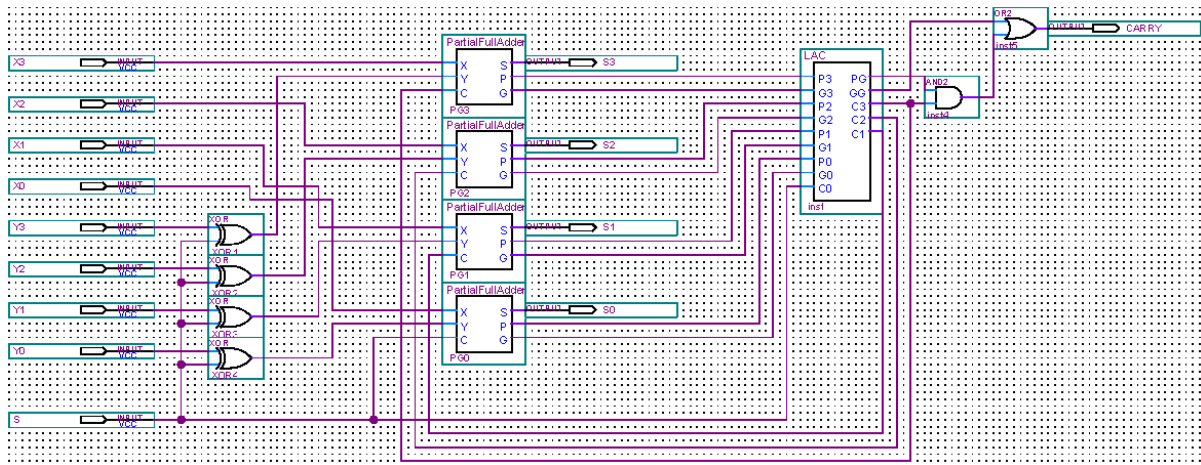
## 문제 2. ALU Logic Symbol & CAD with Simulation



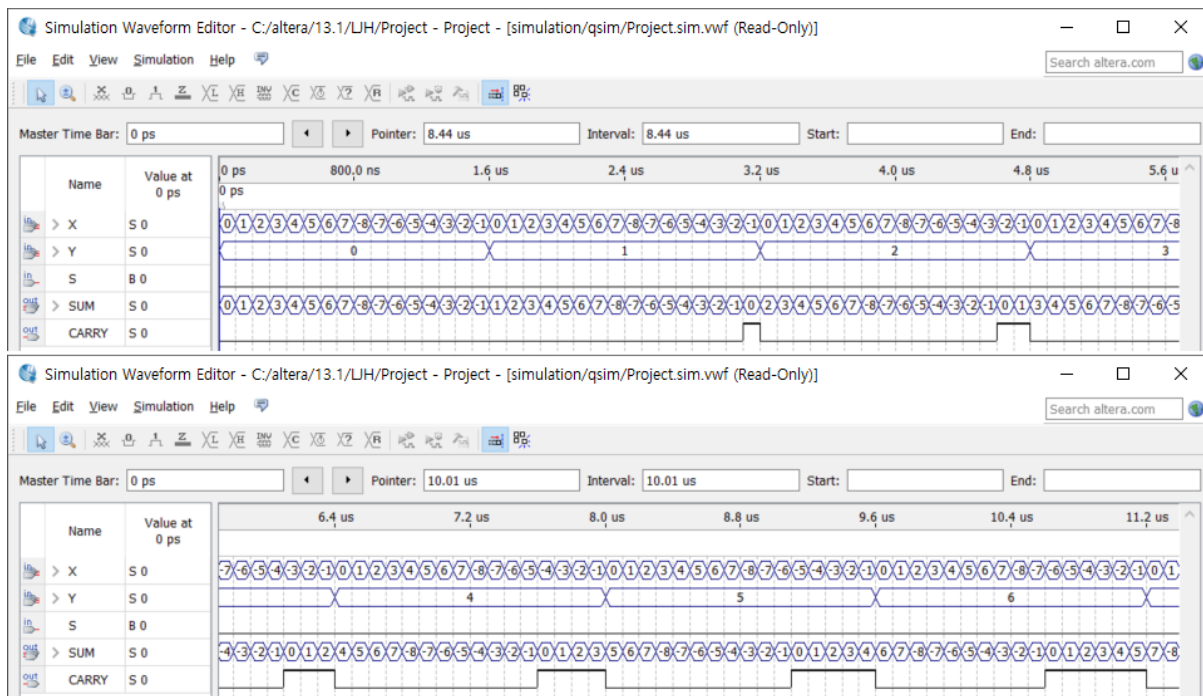
<Partial Full Adder Circuit>



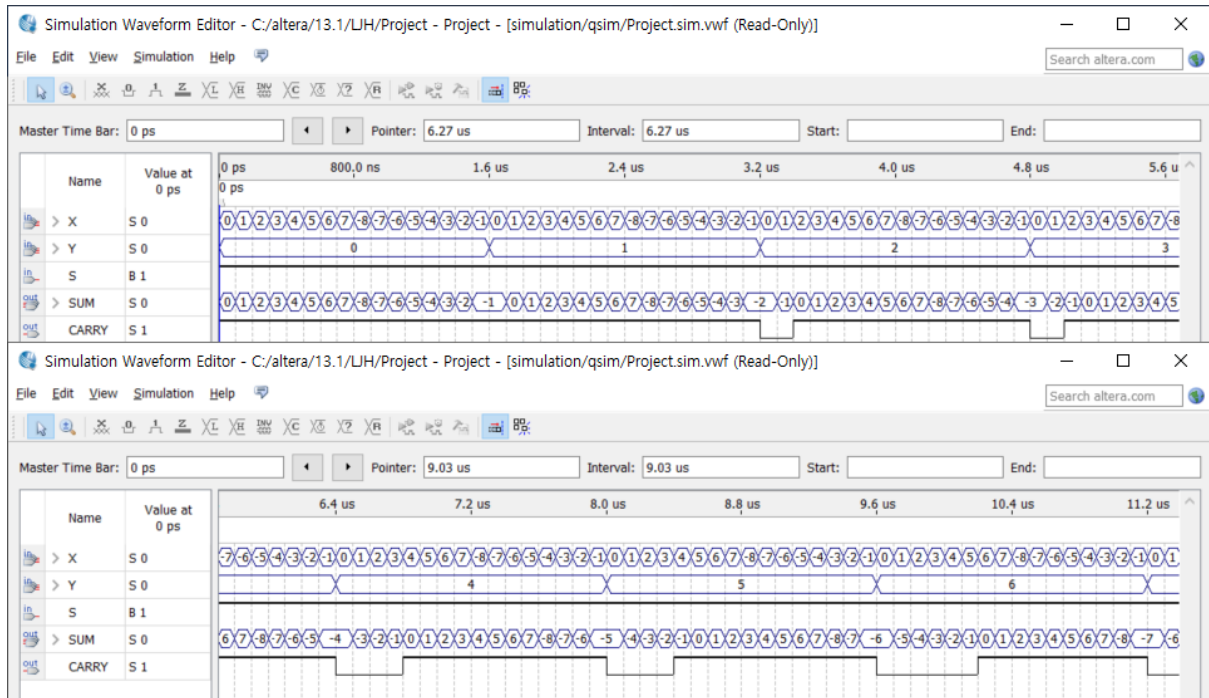
<LAC(Look Ahead Carry) Circuit>



<CLA 4bit Parallel Adder/Subtractor>



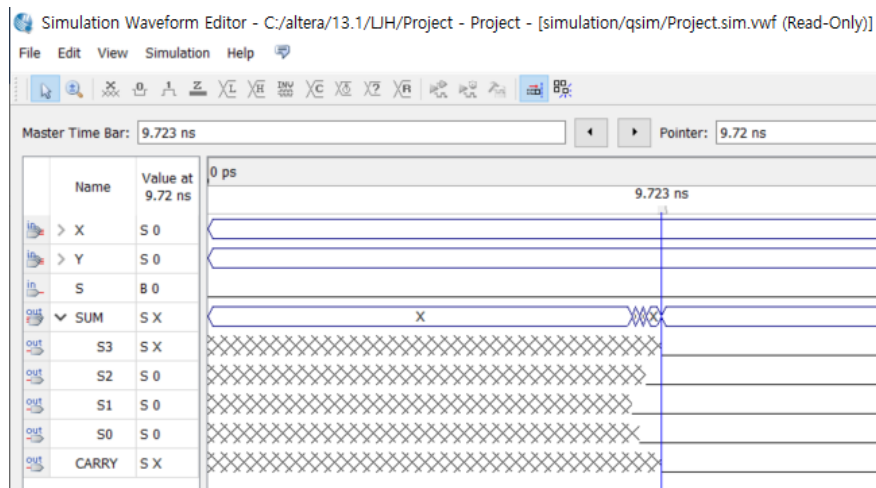
<Functional Simulation - Adder>



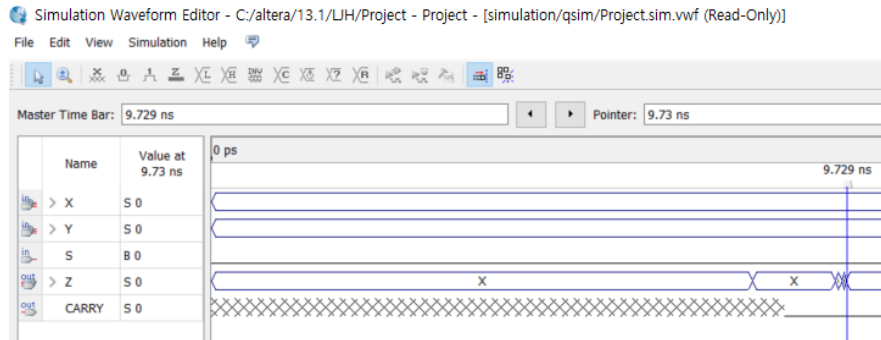
<Functional Simulation - Subtractor>

## 문제 4. Function Simulation / Timing Simulation

<Explain & Compare the Propagation Delay with RCA (Timing Simulation)>



<CLA 4bit Parallel Adder/Subtractor> (9.723ns)



<RCA 4bit Parallel Adder/Subtractor> (9.729ns)

RCA Delay (ns)	CLA Delay (ns)	Diff [RCA – CLA] (ns)
9.723	9.729	0.006

CLA 4 비트 가감산기 회로에서는 9.723ns RCA 4 비트 가감산기 회로에서는 9.729ns 로 지연 시간이 측정되어서, CLA 회로가 6ps 빨랐으나 가시적인(ns 수준의) 차이가 나진 않았다.

$$\begin{aligned}
 C_1 &= G_0 + P_0 C_0 \\
 C_2 &= G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0) = G_1 + P_1 G_0 + P_1 P_0 C_0 \\
 C_3 &= G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0) = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0 \\
 C_4 &= G_3 + P_3 C_3 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0) = G_3 + P_3 G_2 + P_3 P_2 G_1 + \\
 &\quad + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0 \\
 S_i &= X_i \oplus Y_i \oplus C_i = P_i \oplus C_i
 \end{aligned}$$

<4 비트 가감산기에서 사용되는 C1~C4(Carry, 올림), S(Sum, 합)>

하지만 최하위비트에서 발생된 올림이 최상위비트까지 전파된 후에야 합이 계산되는 RCA 회로와 다르게 각 항들의 연산이 두 단계 게이트, C<sub>i</sub>의 계산이 n 개 입력 AND 게이트 한번과 n 개 입력 OR 게이트 한번, 총 두 단계의 게이트로 일시에 끝나버리는 CLA 회로의 특성을 생각하면 8 비트, 16 비트, 32 비트 등 가감산기를 확장하여 n 비트 가감산기를 RCA 와 CLA 로 고려하였을 때 RCA 의 경우 n 비트 RCA 가감산기는 (4 비트 가감산기의 게이트 단계)\*(배수)에 근사하는 지연을 가지는 반면, 모든 캐리 연산이 병렬적으로 수행되는 CLA 가감산기는 배수가 큰 가산기를 만들수록 RCA 가감산기보다 훨씬 빠르게 작동할 것임을 예상할 수 있다.

## 느낀 점과 내가 아직 모르는 것들

계산 소자를 설계할 때 어떤 방식으로 최적화가 이루어지는 지 알게되어 좋았습니다. 부가적으로 회로를 설계하면서 N input AND/OR gate의 지연 시간은

입력 개수와 어떤 상관 관계를 가지고 있는지 궁금증이 생겼습니다. (어떤  $n$  에 대해 가령  $\text{Delay} = an^2 + bn + c$  같은 지연 시간 식이 존재 하는지, 존재한다면 어떤 절차에 의해 해당 지연식이 유도되는지 등)