Carry Lookahead Adder

Carry Lookahead Addition (CLA) is a fast binary addition technique that reduces propagation delay by computing carry bits in parallel rather than sequentially. Unlike a ripple carry adder, which propagates carry bit by bit, a CLA adder generates carry signals using the concepts of **generate (G) and propagate (P) functions**, defined as

$$G_i = A_i B_i, \quad P_i = A_i \oplus B_i$$

The carry output at each stage is determined using precomputed expressions such as

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

enabling faster addition. This parallel carry computation significantly improves speed, making CLA adders crucial in high-speed arithmetic circuits like ALUs and DSP processors.

Example: 8-bit Binary Addition Using Carry Lookahead Method

$$A = 11011011_2 = 219_{10}$$
$$B = 10101101_2 = 173_{10}$$

Step 1: Compute Generate (G) and Propagate (P) Bits

For each bit-pair (A_i, B_i), the Generate and Propagate signals are defined as:

$$G_i = A_i \cdot B_i \quad \text{(Carry Generate)}$$
$$P_i = A_i \oplus B_i \quad \text{(Carry Propagate)}$$

Bit Position	A_i	B_i	$G_i = A_i \cdot B_i$	$P_i = A_i \oplus B_i$
0	1	1	1	0
1	1	0	0	1
2	0	1	0	1
3	1	1	1	0
4	1	0	0	1
5	0	1	0	1
6	1	0	0	1
7	1	1	1	0

Table 1: Generate and Propagate Bits

Step 2: Compute Carry Bits

Using Carry Lookahead logic:

$$
\begin{aligned}
&C_0 = 0 \quad \text{(Initial Carry)} \\
&C_1 = G_0 + (P_0 \cdot C_0) = 1 + (0 \cdot 0) = 1 \\
&C_2 = G_1 + (P_1 \cdot C_1) = 0 + (1 \cdot 1) = 1 \\
&C_3 = G_2 + (P_2 \cdot C_2) = 0 + (1 \cdot 1) = 1 \\
&C_4 = G_3 + (P_3 \cdot C_3) = 1 + (0 \cdot 1) = 1 \\
&C_5 = G_4 + (P_4 \cdot C_4) = 0 + (1 \cdot 1) = 1 \\
&C_6 = G_5 + (P_5 \cdot C_5) = 0 + (1 \cdot 1) = 1 \\
&C_7 = G_6 + (P_6 \cdot C_6) = 0 + (1 \cdot 1) = 1 \\
&C_8 = G_7 + (P_7 \cdot C_7) = 1 + (0 \cdot 1) = 1
\end{aligned}
$$

Step 3: Compute Sum Bits

The sum bits are computed as:

$$S_i = P_i \oplus C_i$$

Bit Position	P_i	C_i	$S_i = P_i \oplus C_i$
0	0	0	0
1	1	1	0
2	1	1	0
3	0	1	1
4	1	1	0
5	1	1	0
6	1	1	0
7	0	1	1
C_{cout}		C_8	1

Table 2: Final Sum Computation

Final Result

$$
\begin{aligned}
&\text{Final Sum} = \mathbf{110001000_2} = \mathbf{392_{10}} \\
&\text{Carry Out} = \mathbf{C_8 = 1}
\end{aligned}
$$

Thus, the final result of $219 + 173$ in binary is:

$$\mathbf{11011011_2 + 10101101_2 = 110001000_2}$$