



INSTITUTO POLITÉCNICO NACIONAL

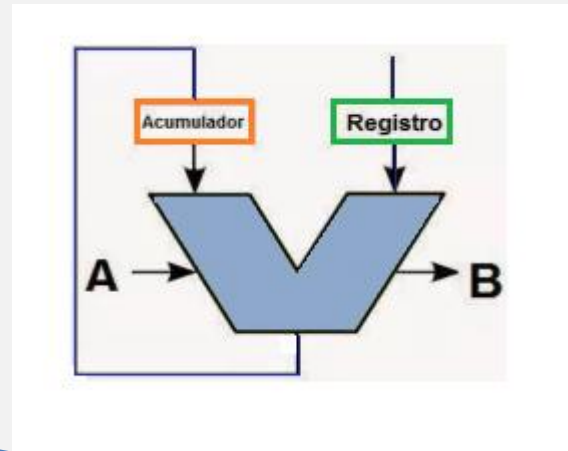
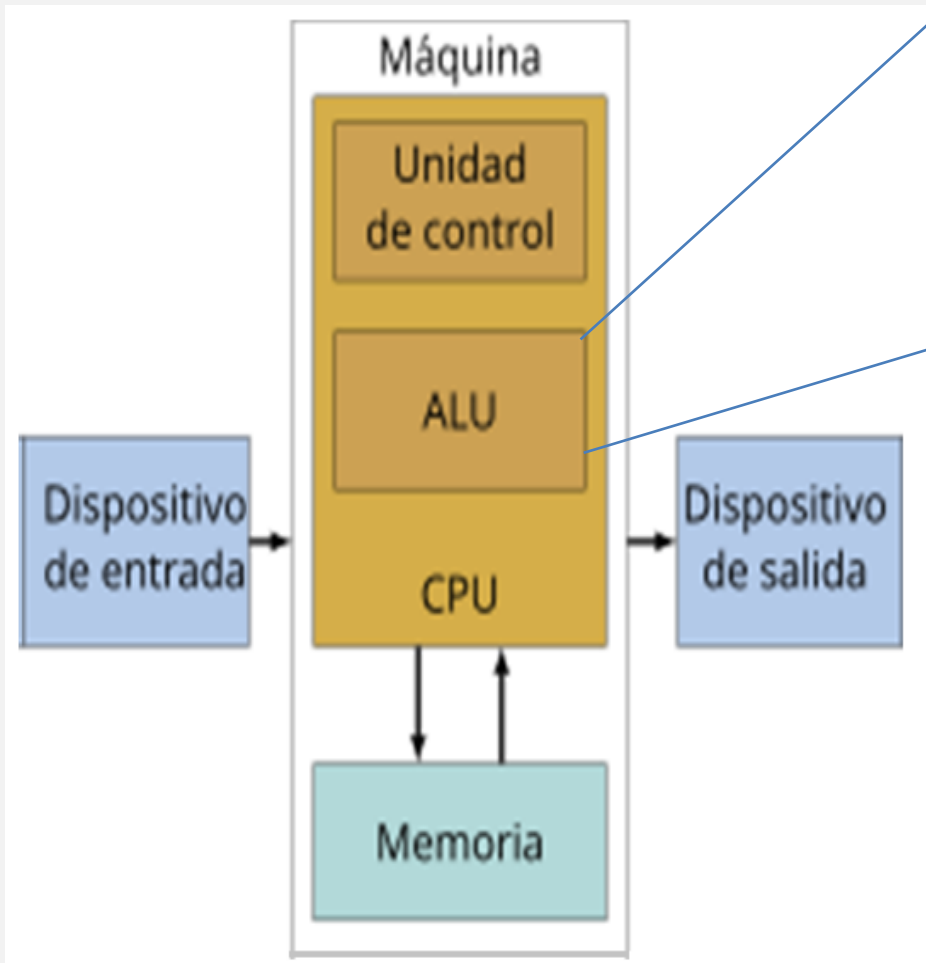
ESCOM

Unidad de Aprendizaje

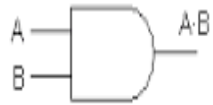
Arquitectura de Computadoras

*Sumador con acarreo en cascada
y sumador con acarreo anticipado*

Marzo 2025



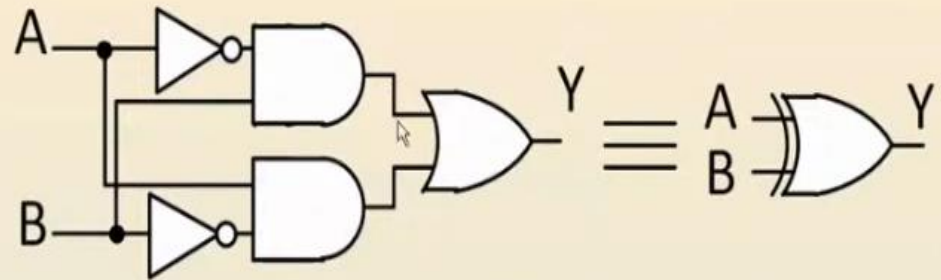
COMPUERTA AND		
A	B	Salida
0	0	0
0	1	0
1	0	0
1	1	1



COMPUERTA OR		
A	B	Salida
0	0	0
0	1	1
1	0	1
1	1	1



Compuerta Lógica XOR (OR exclusiva)



A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

$$Y = \bar{A}B + A\bar{B}$$

$$Y = A \oplus B$$

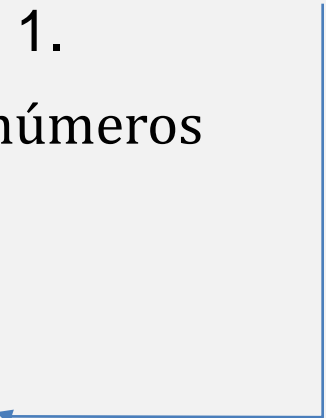
Conceptos

Sumador: Un circuito que realiza la suma de por lo menos dos números binarios.

Acarreo: El bit que se genera cuándo la suma de dos bits (junto con un acarreo anterior, si existe) supera el valor de 1.

Existen cuatro posibles combinaciones en la suma de números binarios:

A	B	Resultado
0	0	0
0	1	1
1	0	1
1	1	1 0

$$\begin{array}{r} 1 \ 1 \\ + \ 1 \ 1 \\ \hline 0 \ 1 \\ \hline 1 \ 0 \ 0 \end{array}$$


Regla	2^n	2^3	2^2	2^1	2^0
Valor		8	4	2	1

Número binario 1 0 = 2 número decimal

Circuito Sumador (sumador medio-Half Adder - HA)

A	0	0	1	1
	+	+	+	+
B	0	1	0	1
	<hr/>	<hr/>	<hr/>	<hr/>
	0	1	1	1 0

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

C_{out}

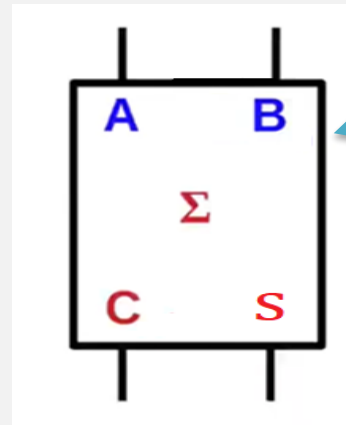
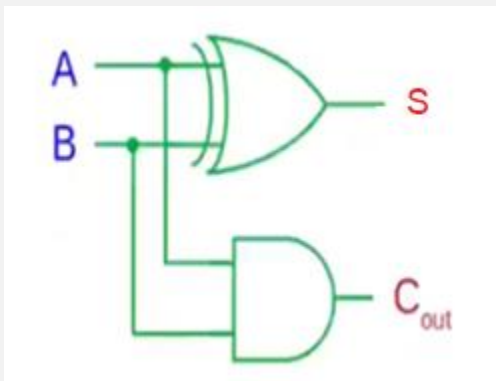
La función lógica de C es:

$$C = AB$$

La función lógica de S es:

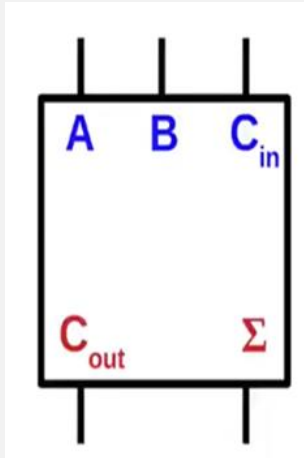
$$S = \bar{A}B + A\bar{B} = A \oplus B$$

Suma números de 1 bit de entrada, en cada operación genera una *suma* y un *acarreo* (No tiene entrada de acarreo)



Circuito sumador de dos números de 1 bit

Forma estándar del Sumador Completo



A	B	C _{in}	C _{out}	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Cout

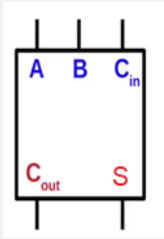
B \ A C _{in}	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$C_{out} = AC_{in} + AB + BC_{in}$$

B \ A C _{in}	00	01	11	10
0	0	1	0	1
1	1	0	1	0

$$S = A\bar{B}\bar{C}_{in} + \bar{A}\bar{B}C_{in} + ABC_{in} + \bar{A}B\bar{C}_{in}$$

Forma estándar del sumador completo

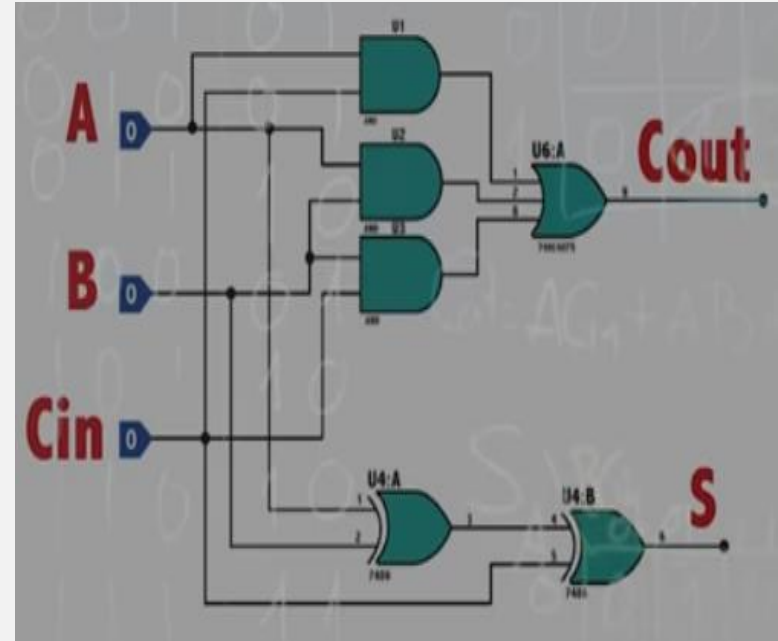


Un sumador completo tiene tres entradas:

- **A** (bit de entrada 1)
- **B** (bit de entrada 2)
- **C_{in}** (acarreo de la posición anterior)

Genera dos salidas:

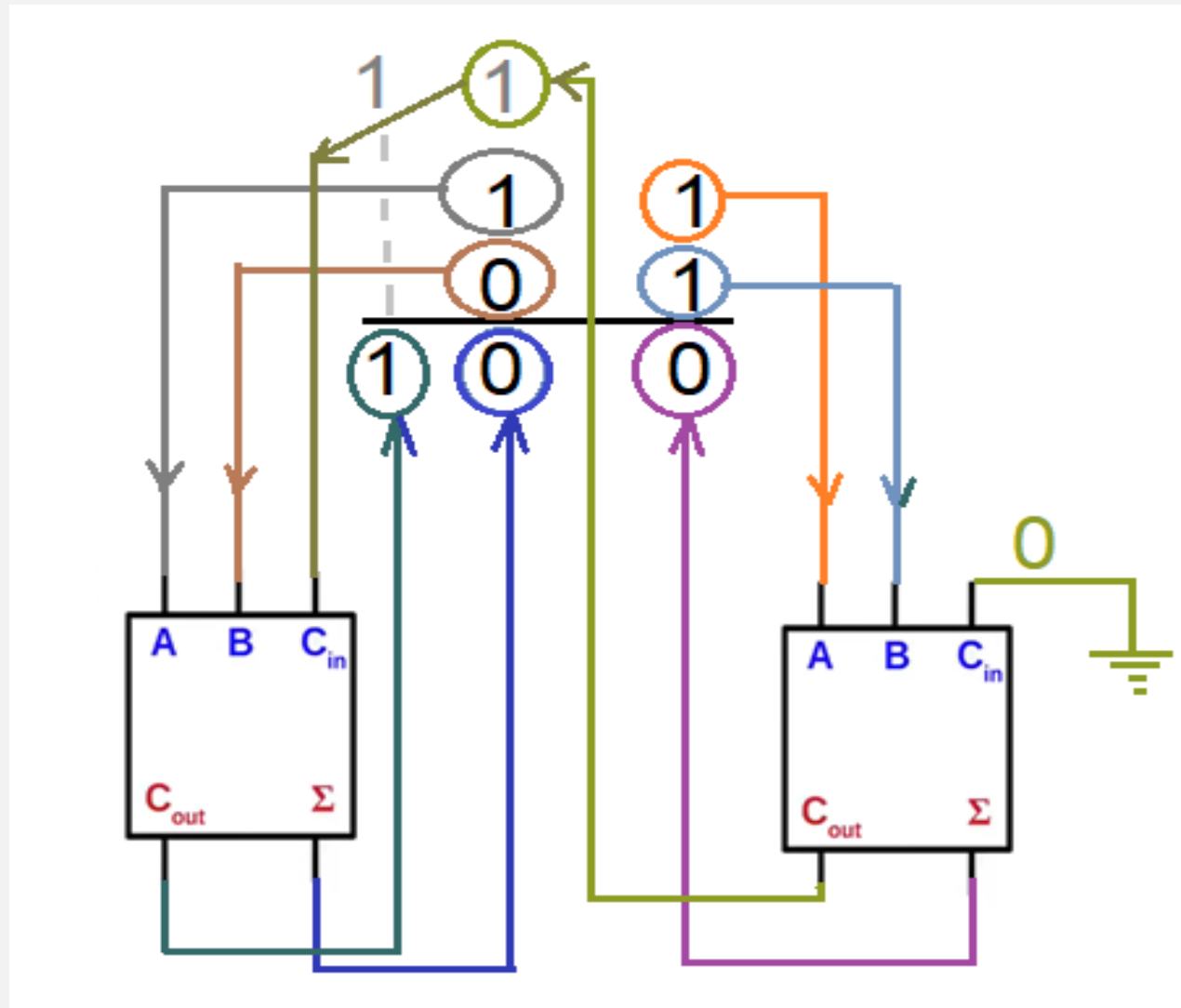
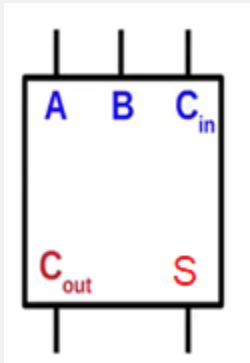
- **S** (suma)
- **C_{out}** (acarreo de salida)



$$C_{out} = AC_{in} + AB + BC_{in}$$

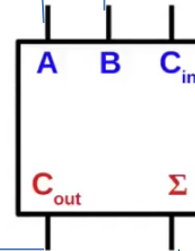
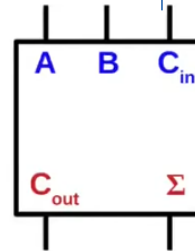
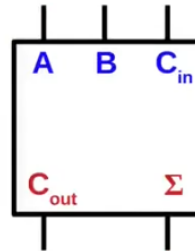
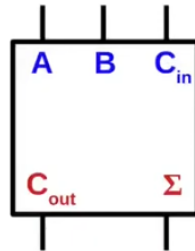
$$S = A\bar{B}\bar{C}_{in} + \bar{A}\bar{B}C_{in} + ABC_{in} + \bar{A}B\bar{C}_{in}$$

Sumador de 2 números de 2 bits cada uno



Sumador completo de 2 números de 4 bit cada uno

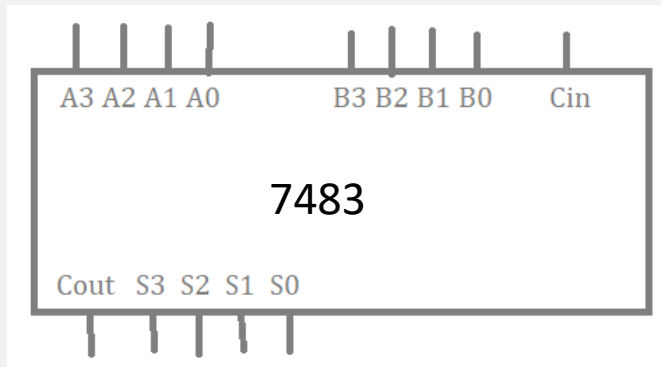
$$\begin{array}{r} 0101 \\ + 1111 \\ \hline \end{array}$$



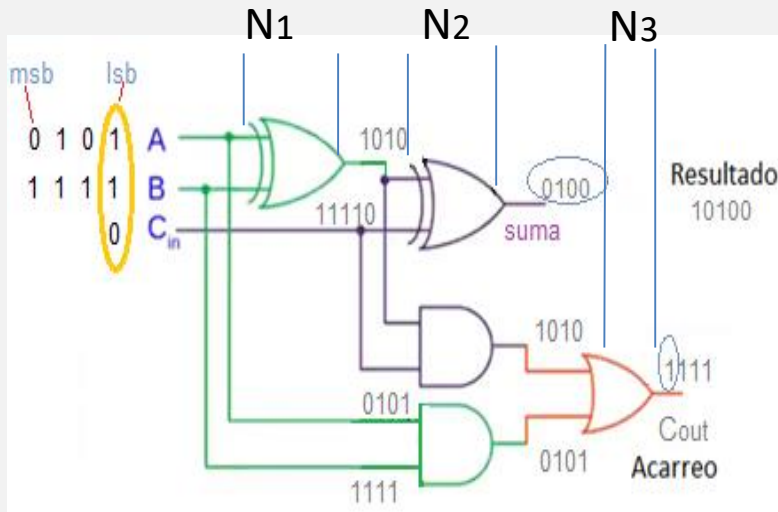
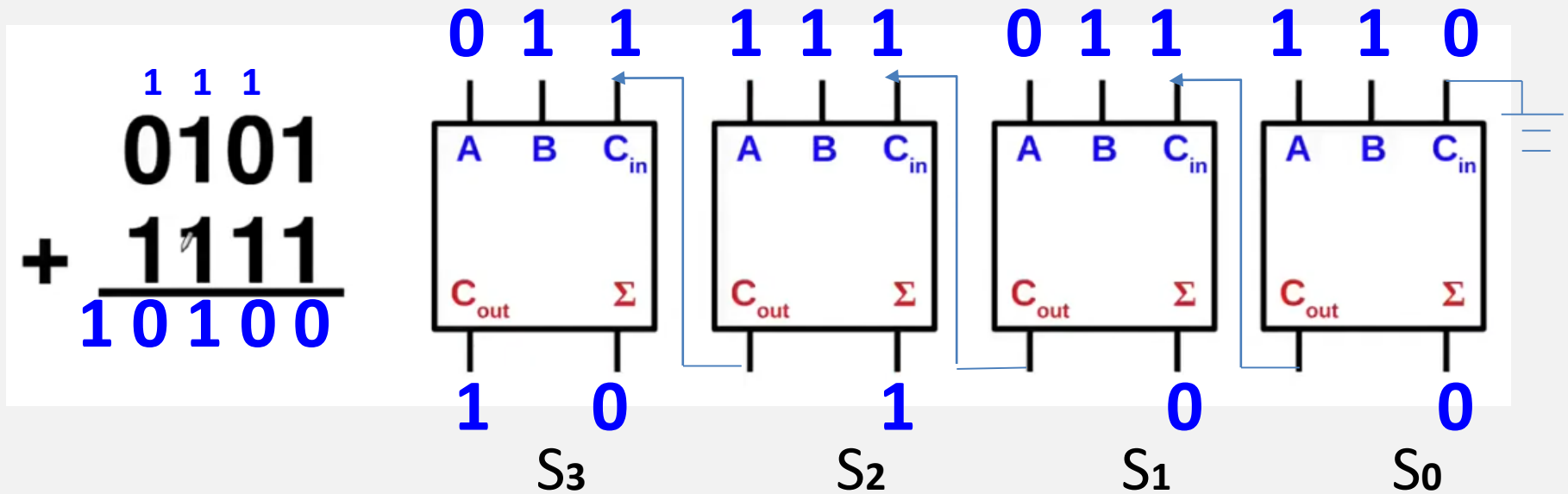
$A = A_3 A_2 A_1 A_0$
 8 4 2 1

$B = B_3 B_2 B_1 B_0$
 8 4 2 1

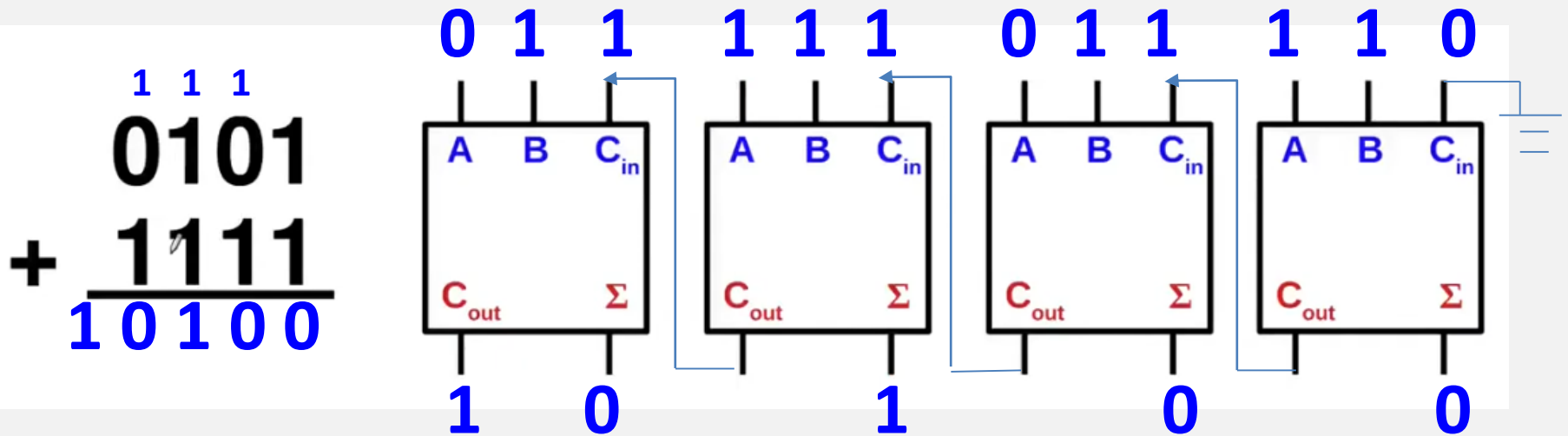
S_0
 S_1
 S_2
 S_3
 S_4



Sumador completo de 2 números de 4 bits cada uno

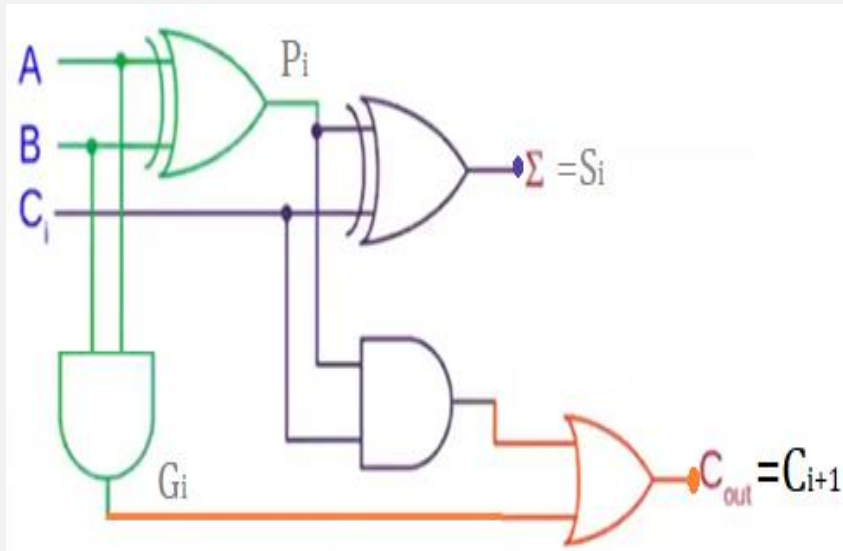


- Cada acarreo recorre 3 niveles de compuertas.
- La atenuación y el retardo son crecientes conforme se incrementa el número de sumadores (tarda en ejecutar las operaciones).



- Suma de 2 números binarios que pueden generar **acarreo** (carry) en cada columna, el resultado de esa columna afecta la siguiente.
- El acarreo debe ser propagado a través de todos los bits, esto puede ser lento para operaciones con números grandes.

sumador de acarreo anticipado (carry look-ahead adder)



$$P_i = A_i \oplus B_i$$

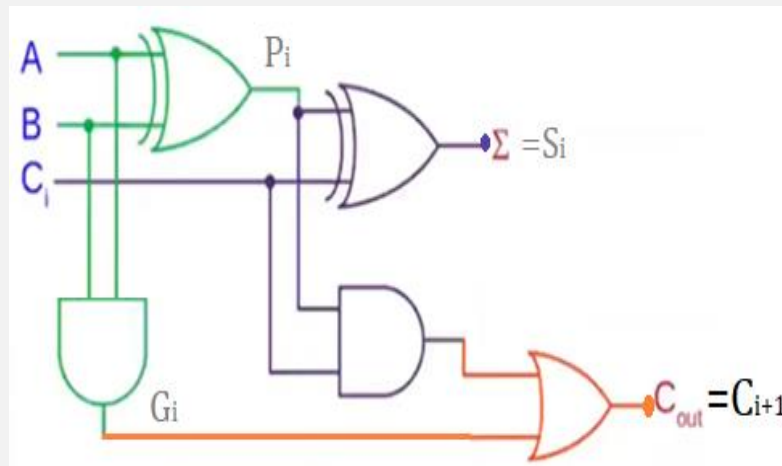
$$G_i = A_i \cdot B_i$$

$$S_i = P_i \oplus C_i$$

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

El *sumador de acarreo anticipado* es un tipo de circuito sumador que mejora la velocidad de las operaciones aritméticas en comparación con un **sumador en serie** o **sumador de acarreo**. Tiene la capacidad de *calcular* o *anticipar* los valores de acarreo sin esperar a que se propaguen de un bit a otro.

Se utiliza el *método de anticipación* que consiste en calcular de *forma paralela* si se va a generar un acarreo en cada columna.



Generación de acarreo (G): Si hay un acarreo en una posición, este se generará independientemente de si hay un acarreo entrante.

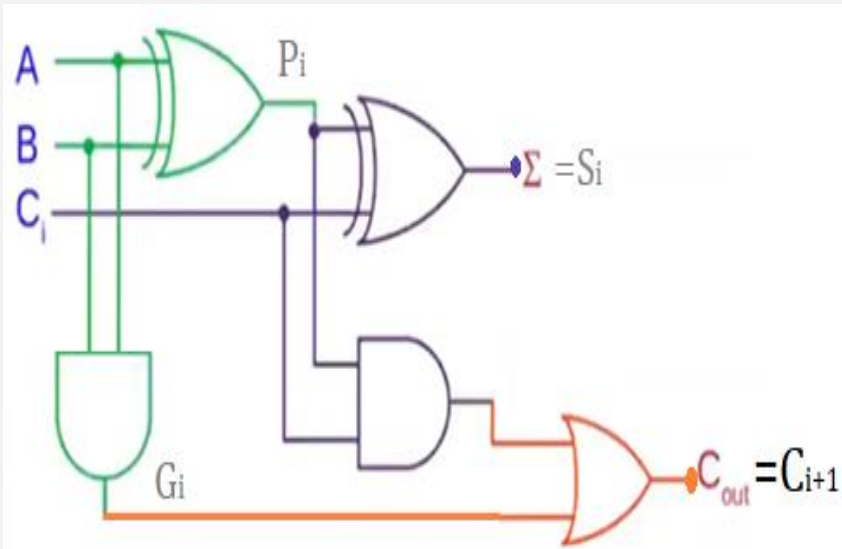
$$G_i = A_i \cdot B_i$$

donde A_i y B_i son los bits de los números que se están sumando en la posición i .

Propagación de acarreo (P): Un acarreo se propagará a la siguiente posición si el bit correspondiente de la suma es 1.

$$P_i = A_i \oplus B_i$$

El sumador de **acarreo anticipado** puede calcular los acarreos para todos los bits de manera paralela, lo que resulta en un tiempo de suma mucho más rápido.



$$P_i = A_i \oplus B_i$$

$$G_i = A_i \cdot B_i$$

$$S_i = P_i \oplus C_i$$

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

$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 G_0 + P_0 P_1 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 G_1 + P_1 P_2 G_0 + P_0 P_1 P_2 C_0$$

$$C_4 = G_3 + G_2 P_3 + G_1 P_2 P_3 + G_0 P_1 P_2 P_3 + C_0 P_0 P_1 P_2 P_3$$

$$\vdots = \vdots$$

$$C_{n+1} = G_n + \sum_{k=1}^n \left[G_{k-1} \prod_{i=k}^n P_i \right] + C_0 \prod_{i=0}^n P_i$$

En una suma binaria, cada bit sumado puede generar un acarreo que se transfiere al siguiente bit más significativo.

En un sumador convencional, este acarreo debe propagarse de un bit al siguiente, lo que puede hacer que la operación sea lenta.

El sumador de acarreo anticipado mejora esto al calcular el acarreo en todas las posiciones de bits *simultáneamente*, en lugar de esperar a que se propague de uno en uno. Esto se logra utilizando puertas lógicas que predicen los valores de acarreo basándose en las entradas antes de que la suma se complete.

Sumador con Acarreo Anticipado

$$C_1 = G_0 + P_0 C_0$$

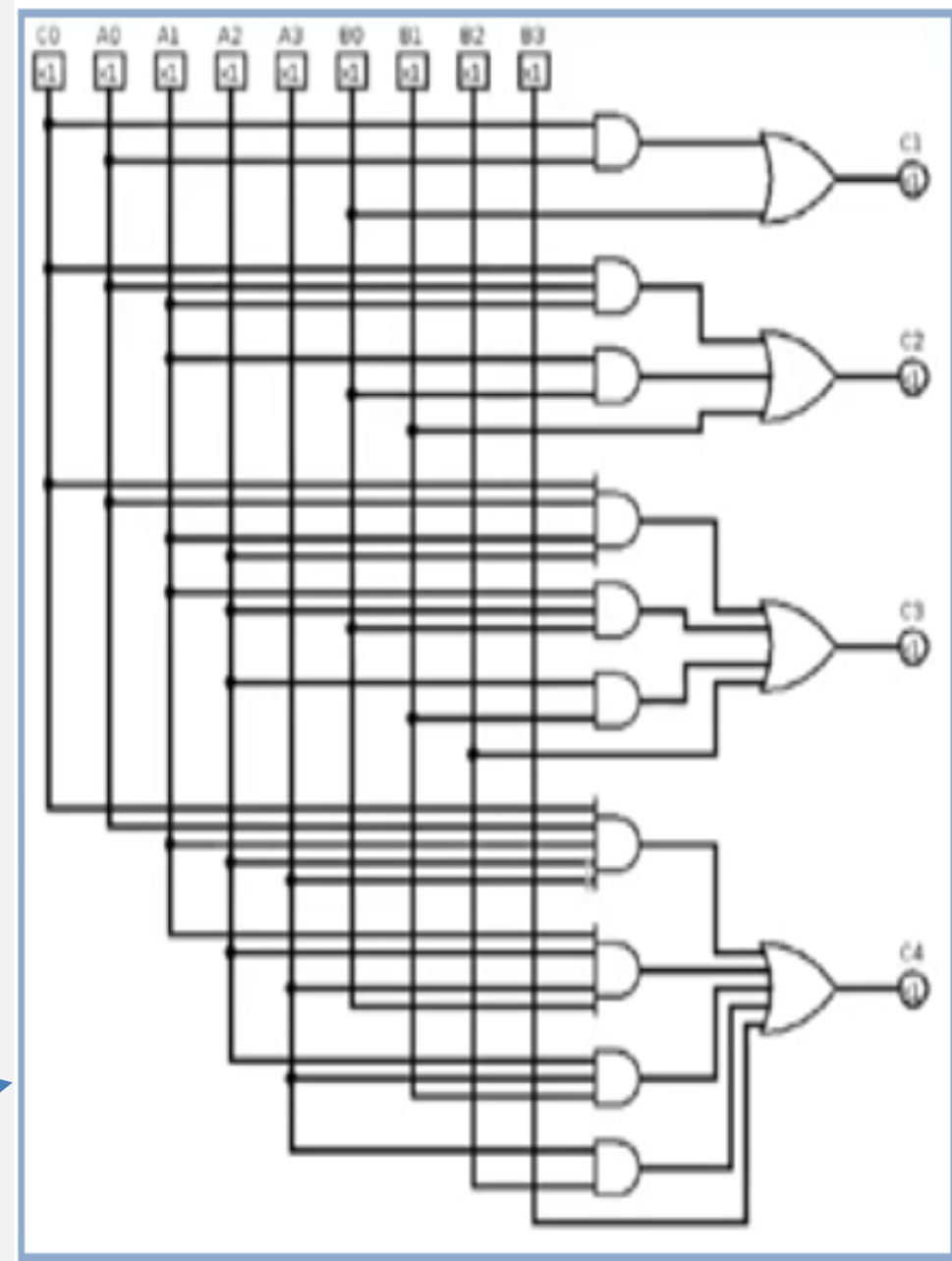
$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 G_0 + P_0 P_1 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 G_1 + P_1 P_2 G_0 + P_0 P_1 P_2 C_0$$

$$C_4 = G_3 + G_2 P_3 + G_1 P_2 P_3 + G_0 P_1 P_2 P_3 + C_0 P_0 P_1 P_2 P_3$$

$$\vdots = \vdots$$

$$C_{n+1} = G_n + \sum_{k=1}^n \left[G_{k-1} \prod_{i=k}^n P_i \right] + C_0 \prod_{i=0}^n P_i$$



(Carry lookahead adder-CLA)

Ventajas

Velocidad: El tiempo de suma no depende del número de bits de los operandos, debido a que todos los acarreos se calculan de manera paralela. Esto lo hace mucho más rápido que los sumadores en serie (requieren que cada acarreo se propague secuencialmente).

Eficiencia en circuitos grandes: Para procesadores o sistemas que realizan operaciones de suma en paralelo (procesamiento de señales o cómputo científico), un sumador de acarreo anticipado mejorará el rendimiento.

Desventajas

Complejidad: Son más complejos en términos de diseño y consumo de área en un circuito (requieren más puertas lógicas como AND, OR, XOR para calcular los acarreos de manera anticipada).

Consumo de recursos: Son rápidos pero consumen más energía y espacio en comparación con otros tipos de sumadores más simples como el sumador en serie.

Característica	Sumador en cascada (RCA)	Sumador en paralelo (LCA)
Velocidad	Lenta (propagación del acarreo)	Rápida (cálculo anticipado)
Complejidad	Baja (simple)	Alta (requiere más puertas lógicas)
Consumo de hardware	Menor	Mayor
Uso recomendado	Circuitos simples y de bajo costo	Aplicaciones de alta velocidad