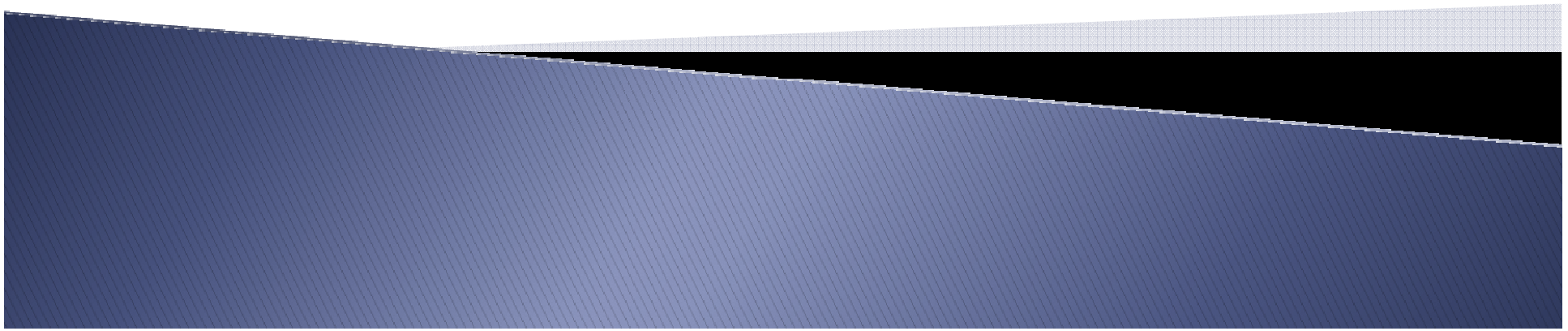
Sumadores

Alfonso González
Cintia González
Eric Alvarado

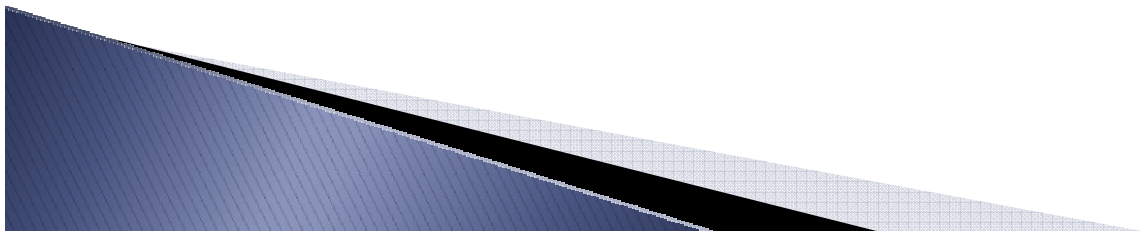


Definición

- En electrónica un sumador es un circuito lógico que calcula la operación suma. En los computadores modernos se encuentra en lo que se denomina Unidad aritmético lógica (ALU). Generalmente realizan las operaciones aritméticas en código binario, decimal: BCD o exceso 3, por regla general los sumadores emplean el sistema binario. En los casos en los que se esté empleando un complemento a dos para representar números negativos el sumador se convertirá en un sumador-restador (Adder-subtracter).

Tipos de sumadores

- Half-Adder
- Full-Adder
- Método Ripple
- Carry-Look-Ahead



Semisumador

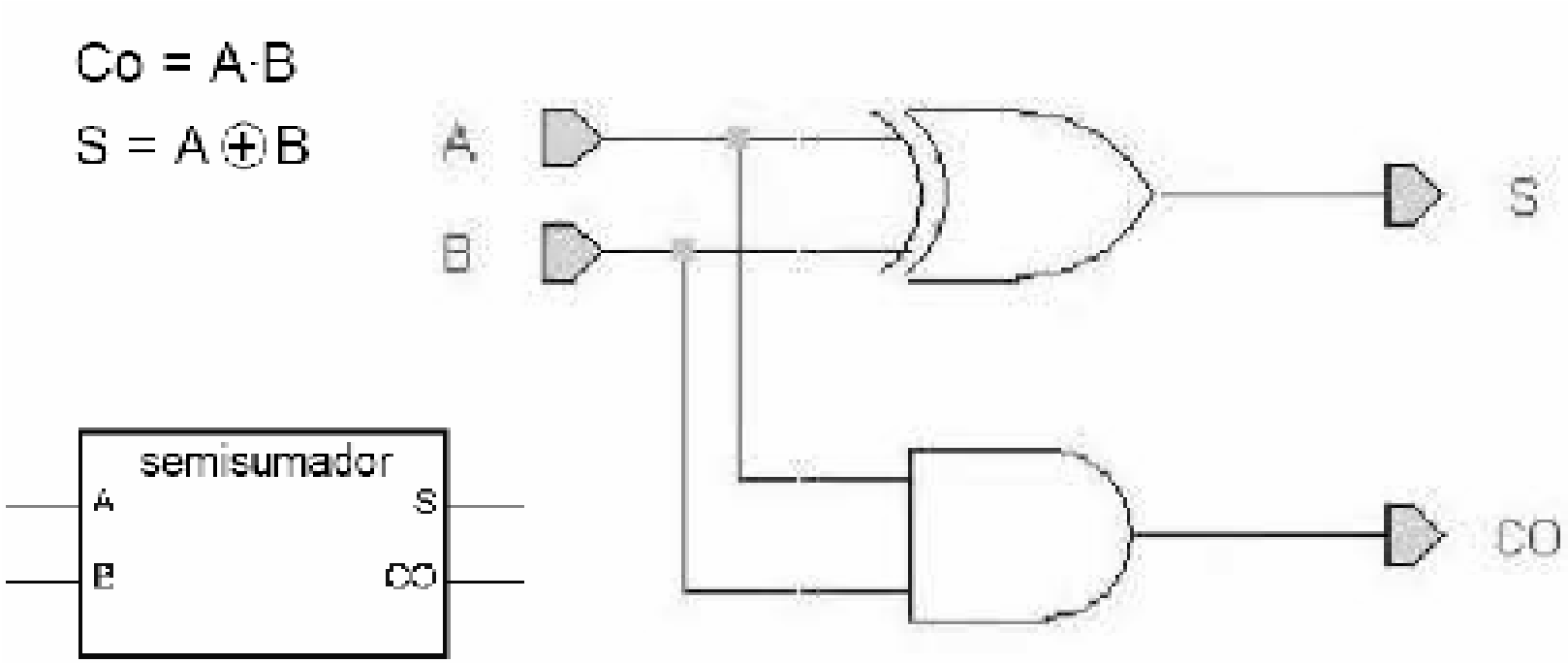
- Se denomina semisumador aun circuito que admite 2 bits como entrada y genera como salida:
  - Un bit que representa la suma de los 2 bits de entrada
  - Otro bit que representa el acarreo generado por la suma

La Tabla de verdad de este circuito puede deducirse a partir de las reglas de la suma binaria

A	B	Co	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

A,B → sumandos
Co → acarreo de salida
S → suma

- A partir de esta tabla de verdad se puede observar que la suma puede implementarse con una operación XOR y el acarreo de salida con una operación AND



Full-Adder

- La principal diferencia entre el sumador completo y el semisumador es que este admite un valor que represente un acarreo de entrada.

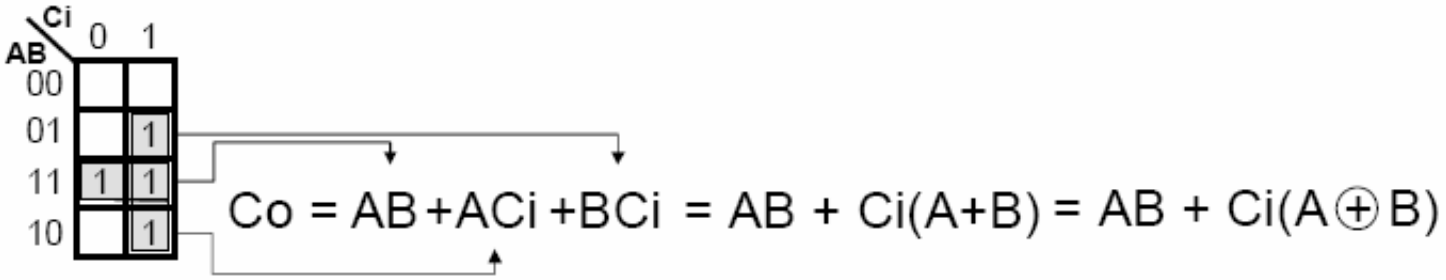
Ci	A	B	Co	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

A,B → sumandos
Ci → acarreo de entrada
Co → acarreo de salida
S → suma

- Dado que podemos expresar la suma de dos bits con la operación XOR, podemos expresar la suma de dos bits y un acarreo de la siguiente forma:

$$S = A \oplus B \oplus Ci$$

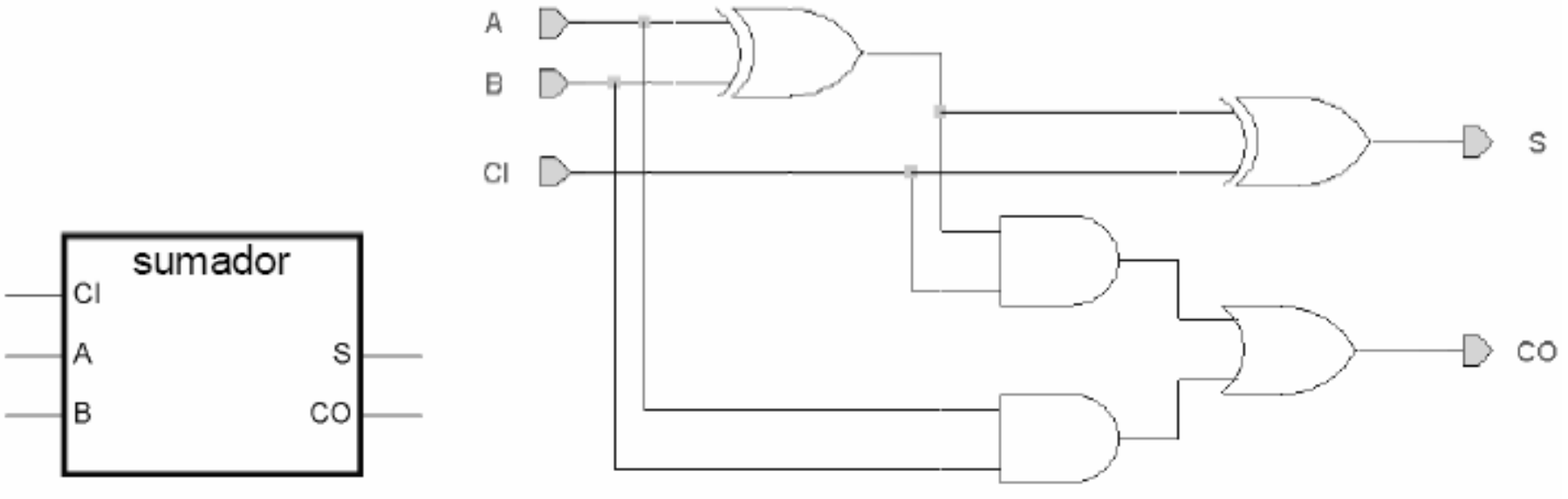
- El acarreo de salida será uno en dos circunstancias:
  - Cuando las 2 entradas A y B sean 1
  - Cuando la suma de las 2 entradas sea 1 y el acarreo de entrada tambien sea 1.



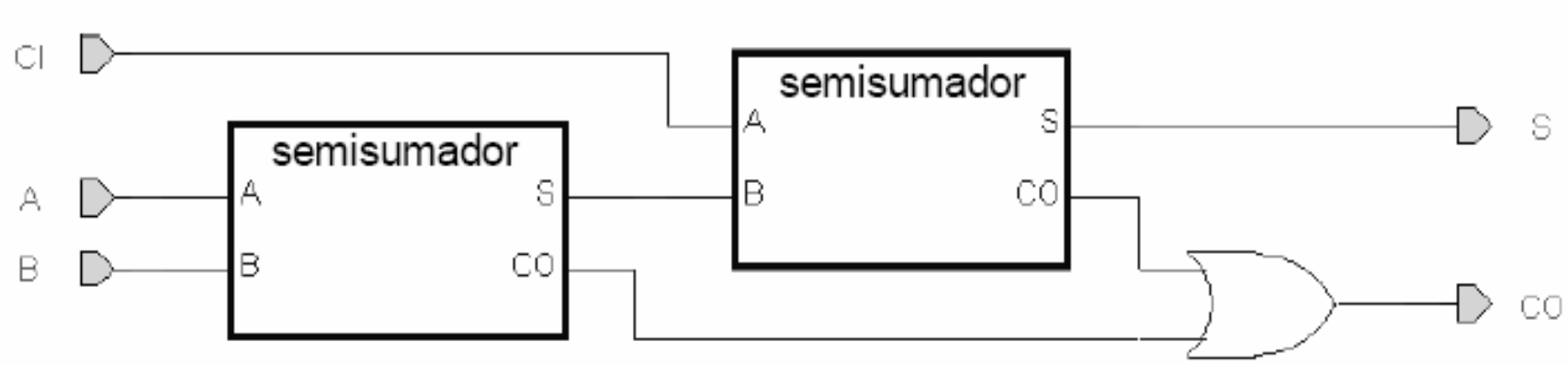
- Se puede implementar el circuito sumador full-adder usando dos puertas XOR, dos puertas AND y una puerta OR.

$$S = A \oplus B \oplus Ci$$

$$Co = AB + Ci(A \oplus B)$$



- Es posible implementar el circuito sumador full-adder utilizando dos semisumadores
  - El primer semisumador suma los 2 bits.
  - El segundo suma el resultado con el acarreo de entrada.
  - Habrá acarreo de salida si cualquiera de los dos semisumadores genera un acarreo.



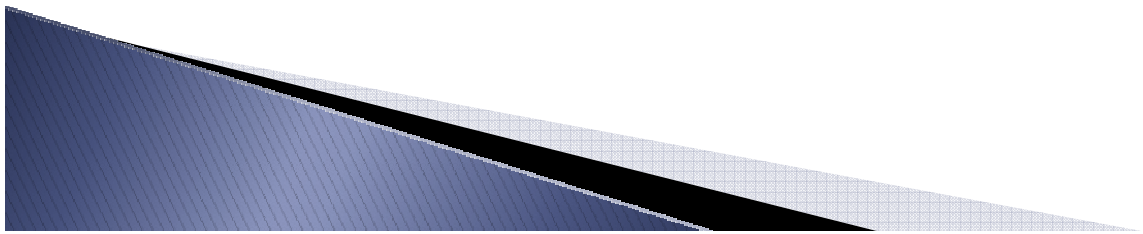
Vhdl full-adder

```
Library ieee;
Entity Sumador is
Port ( a, b, co: in bit ;s , cout : out bit);
End sumador;
Architecture caso1 of sumador is
Begin
  Process(a,b,cin)
    Begin
     If( a='1' and b='1' or
         a='1' and co='1' or
         b='1' and co='1')
      Then  cout <='1' ;
      Else   cout  <='0';
     End if;
     If (  a='0' and b='0' and c='1'  or
           a='0' and b='1' and c='0' or
           a='1' and b='0' and c='0' or
           a='1' and b='1' and c='1')
      Then s<='1';
      Else  s<='0';
    End if;
  End process;
End caso1;
```

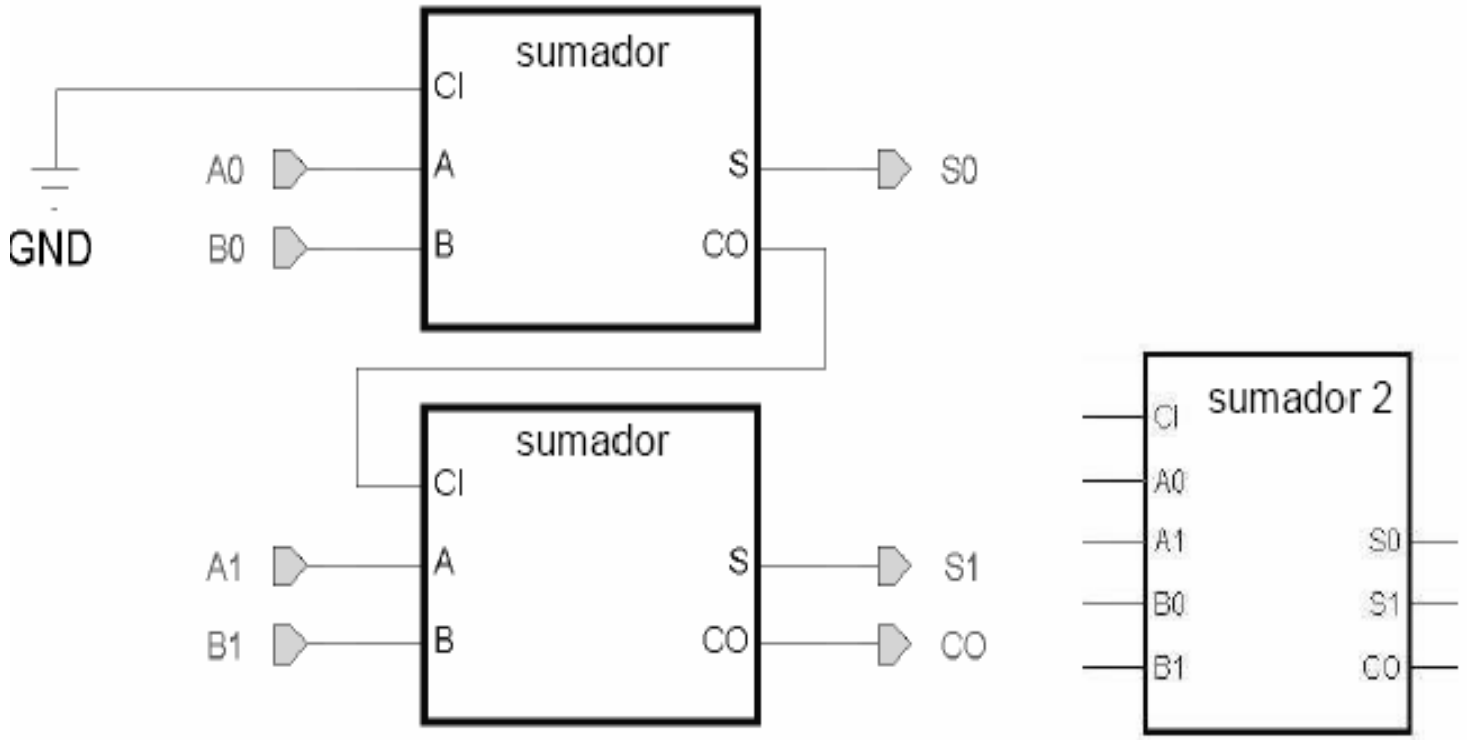
Método Ripple

- Un circuito sumador completo permite sumar dos números de un bit con un acarreo de entrada y generar un acarreo de salida.

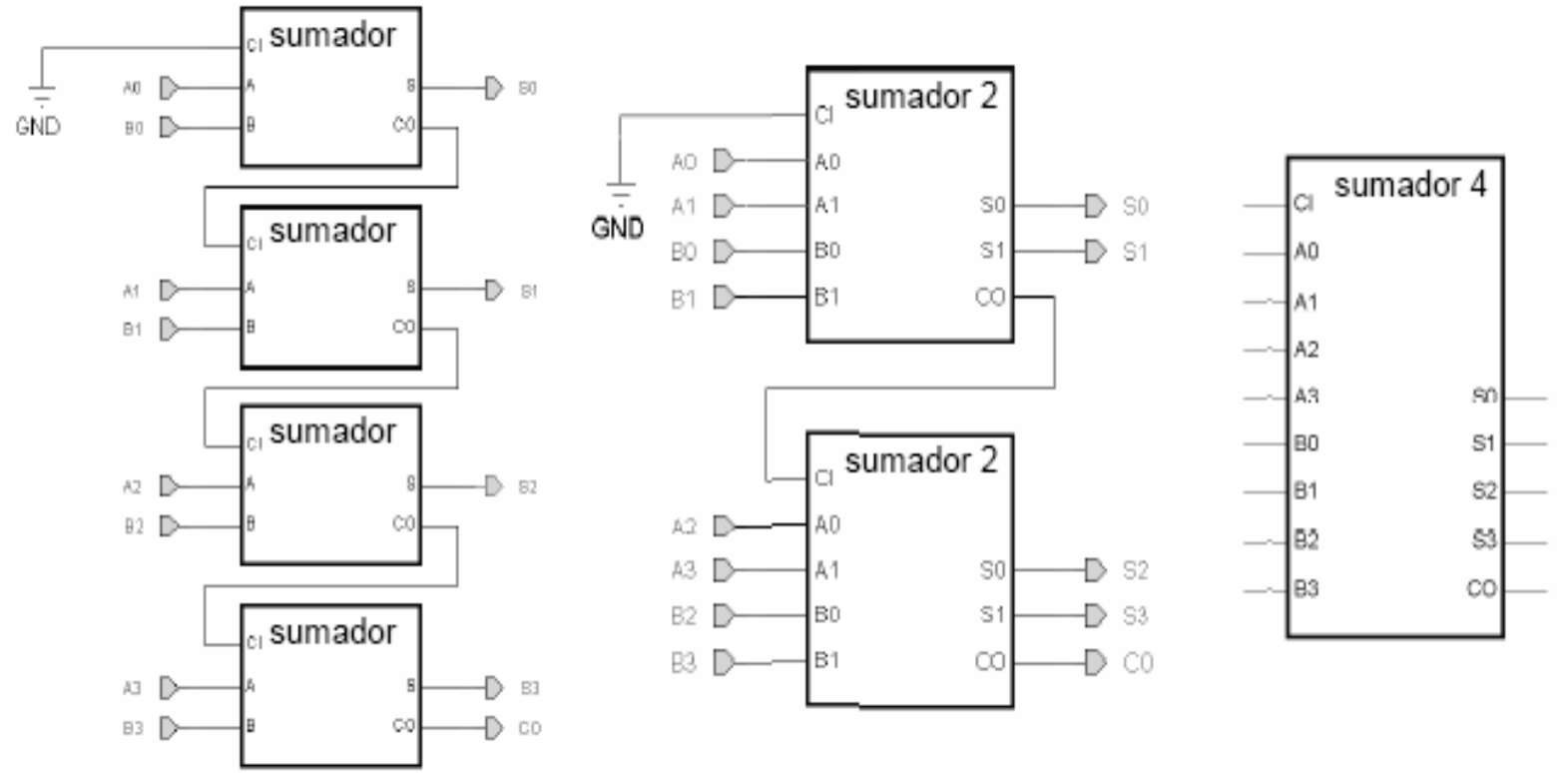
- Como regla general, un sumador binario de cualquier numero de bits puede realizarse conectando en cascada varios sumadores completos de un bit



- El primero de los acarreos de entrada debe estar siempre a cero ya que representa el acarreo inicial en los bits menos significativos.



- El principal problema de esta conexión en serie de sumadores es que el retardo del circuito depende de la propagación del acarreo a lo largo de todo el sumador.



Carry Look Ahead

- Este sumador, llamado también sumador paralelo con acarreo anticipado, realiza la suma aumentando la velocidad de proceso sobre la conexión en serie. Lo logra mediante la generación de todos los bits de acarreo en el mismo proceso de calculo de las sumas parciales.

- Para poder anticipar el valor del acarreo hay que dividir la función que lo expresa en otras dos funciones

$$C_{i+1} = \underbrace{AB}_{G_i} + C_i\underbrace{(A \oplus B)}_{P_i}$$

P_i → propagación de acarreo

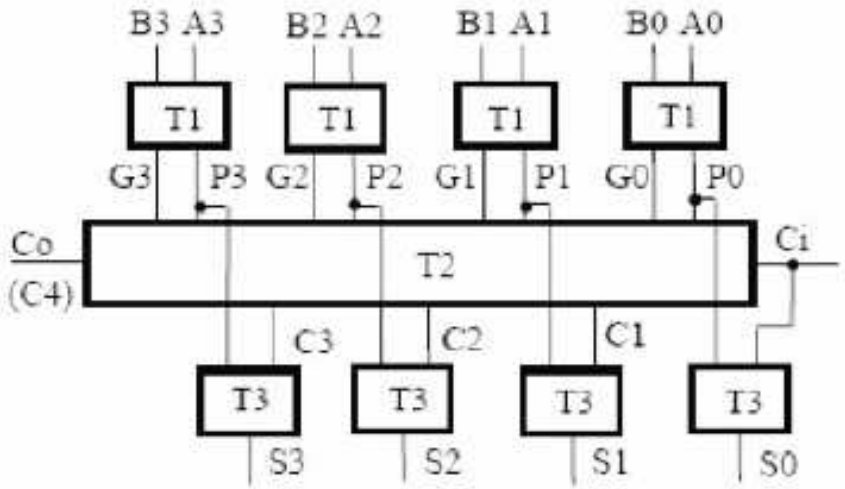
G_i → generación de acarreo

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

- Si suponemos un sumador de 4 bits podemos calcular cada uno de los acarreos intermedios

$$C1 = G0 + P0Ci$$
$$C2 = G1 + P1\,C1 = G1 + P1\,(G0 + P0\,Ci) = G1 + P1G2 + P1P0Ci$$
$$C3 = G2 + P2\,C2 = G2 + P2G1 + P2P1G0 + P2P1P0Ci$$
$$Co = C4 = G3 + P3\,C3 = G3 + P3G2 + P3P2G1 + P3P2P1G0 + P4P3P2P1Ci$$

$$Sj = Ai \oplus Bj \oplus Ci = Pj \oplus Ci$$



Circuito comercial 74283

54LS283/DM54LS283/DM74LS283
4-Bit Binary Adders with Fast Carry

General Description

These full adders perform the addition of two 4-bit binary numbers. The sum (Σ) outputs are provided for each bit and the resultant carry (C_4) is obtained from the fourth bit. These adders feature full internal look ahead across all four bits. This provides the system designer with partial look-ahead performance at the economy and reduced package count of a ripple-carry implementation.

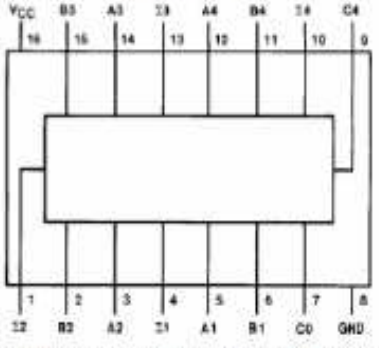
The adder logic, including the carry, is implemented in its true form meaning that the end-around carry can be accomplished without the need for logic or level inversion.

Features

- Full-carry look-ahead across the four bits
- Systems achieve partial look-ahead performance with the economy of ripple carry
- Typical add times
  - Two 8-bit words 25 ns
  - Two 16-bit words 45 ns
- Typical power dissipation per 4-bit adder 95 mW
- Alternate Military/Aerospace device (54LS283) is available. Contact a National Semiconductor Sales Office/Distributor for specifications.

Connection Diagram

Dual-In-Line Package



Order Number 54LS283DMQB, 54LS283FMQB, 54LS283LMQB, DM54LS283J, DM54LS283W, DM74LS283M or DM74LS283N
See NS Package Number E20A, J16A, M16A, N16E or W16A

Recommended Operating Conditions

Symbol	Parameter	DM54LS283 Min	DM54LS283 Nom	DM54LS283 Max	DM74LS283 Min	DM74LS283 Nom	DM74LS283 Max	Units
V_{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V_{IH}	High Level Input Voltage	2			2			V
V_{IL}	Low Level Input Voltage			0.7			0.8	V
I_{OH}	High Level Output Current			−0.4			−0.4	mA
I_{OL}	Low Level Output Current			4			8	mA
T_A	Free Air Operating Temperature	−55		125	0		70	°C

Electrical Characteristics (over recommended operating free air temperature range (unless otherwise noted))

Symbol	Parameter	Conditions		Min	Typ (Note 1)	Max	Units
V_I	Input Clamp Voltage	V_{CC} = Min, I_I = −18 mA				−1.5	V
V_{OH}	High Level Output Voltage	V_{CC} = Min, I_{OH} = Max, V_{IL} = Max, V_{IH} = Min	DM54	2.5	3.4		V
			DM74	2.7	3.4		
V_{OL}	Low Level Output Voltage	V_{CC} = Min, I_{OL} = Max, V_{IL} = Max, V_{IH} = Min	DM54		0.25	0.4	V
			DM74		0.35	0.5	
		I_{OL} = 4 mA, V_{CC} = Min	DM74		0.25	0.4	
I_I	Input Current @ Max Input Voltage	V_{CC} = Max, V_I = 7V	A, B			0.2	mA
			C0			0.1	
I_{IH}	High Level Input Current	V_{CC} = Max, V_I = 2.7V	A, B			40	μA
			C0			20	
I_{IL}	Low Level Input Current	V_{CC} = Max, V_I = 0.4V	A, B			−0.8	mA
			C0			−0.4	
I_{OS}	Short Circuit Output Current	V_{CC} = Max (Note 2)	DM54	−20		−100	mA
			DM74	−20		−100	
I_{CCH}	Supply Current	V_{CC} = Max (Note 3)			19	34	mA
I_{CCL}	Supply Current	V_{CC} = Max (Note 4)			22	39	mA

Switching Characteristics at V_{CC} = 5V and T_A = 25°C (See Section 1 for Test Waveforms and Output Load)

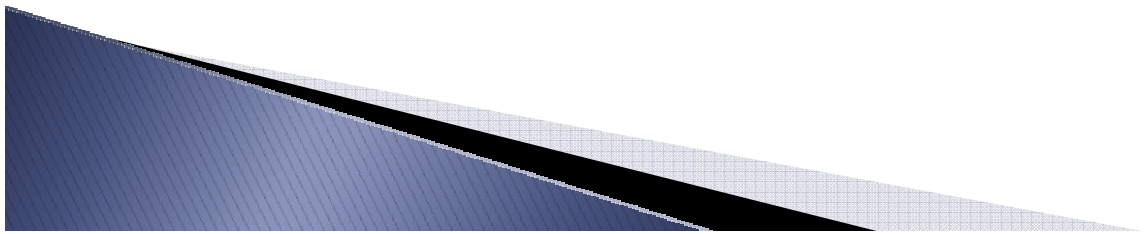
Symbol	Parameter	From (Input) To (Output)	R_L = 2 kΩ				Units
			C_L = 15 pF		C_L = 50 pF		
			Min	Max	Min	Max	
t_{PLH}	Propagation Delay Time Low to High Level Output	C0 to Σ1, Σ2		24		28	ns
t_{PHL}	Propagation Delay Time High to Low Level Output	C0 to Σ1, Σ2		24		30	ns
t_{PLH}	Propagation Delay Time Low to High Level Output	C0 to Σ3		24		28	ns
t_{PHL}	Propagation Delay Time High to Low Level Output	C0 to Σ3		24		30	ns
t_{PLH}	Propagation Delay Time Low to High Level Output	C0 to Σ4		24		28	ns
t_{PHL}	Propagation Delay Time High to Low Level Output	C0 to Σ4		24		30	ns
t_{PLH}	Propagation Delay Time Low to High Level Output	A_i or B_i to Σ_i		24		28	ns
t_{PHL}	Propagation Delay Time High to Low Level Output	A_i or B_i to Σ_i		24		30	ns
t_{PLH}	Propagation Delay Time Low to High Level Output	C0 to C4		17		24	ns
t_{PHL}	Propagation Delay Time High to Low Level Output	C0 to C4		17		25	ns
t_{PLH}	Propagation Delay Time Low to High Level Output	A_i or B_i to C4		17		24	ns
t_{PHL}	Propagation Delay Time High to Low Level Output	A_i or B_i to C4		17		25	ns

Function Table

Input				Outputs					
				When C0 = L			When C0 = H		
				When C2 = L			When C2 = H		
A1 / A3	B1 / B3	A2 / A4	B2 / B4	Σ1 / Σ3	Σ2 / Σ4	C2 / C4	Σ1 / Σ3	Σ2 / Σ4	C2 / C4
L	L	L	L	L	L	L	H	L	L
H	L	L	L	H	L	L	L	H	L
L	H	L	L	H	L	L	L	H	L
H	H	L	L	L	H	L	H	H	L
L	L	H	L	L	H	L	H	H	L
H	L	H	L	H	H	L	L	L	H
L	H	H	L	H	H	L	L	L	H
H	H	H	L	L	L	H	H	L	H
L	L	L	H	L	H	L	H	H	L
H	L	L	H	H	H	L	L	L	H
L	H	L	H	H	H	L	L	L	H
H	H	L	H	L	L	H	H	L	H
L	L	H	H	L	L	H	H	L	H
H	L	H	H	H	L	H	L	H	H
L	H	H	H	H	L	H	L	H	H
H	H	H	H	L	H	H	H	H	H

Problema

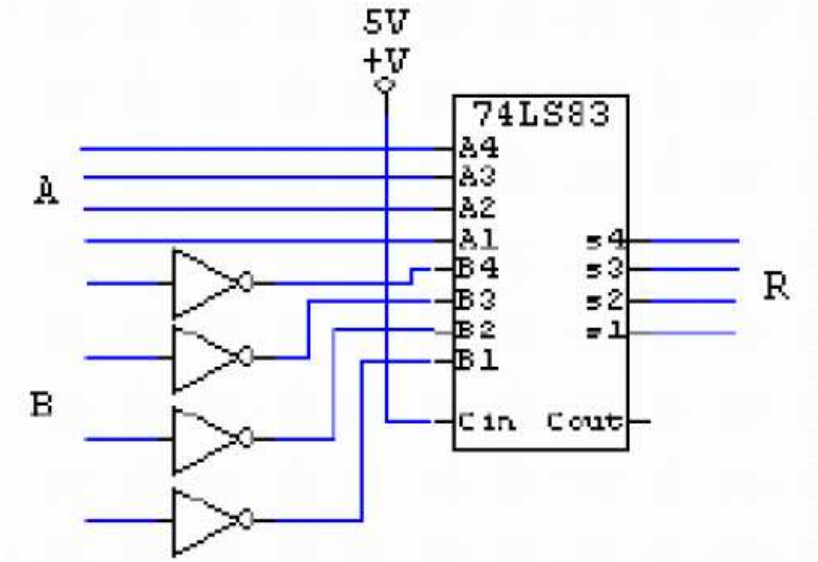
- A base de sumadores desarrollar otras aplicaciones como por ejemplo un restador



Este dispositivo puede verse como

A – B = A + (–B) , para la conversión del

operando B se emplea la codificación en complemento a dos.



Problema

- Diseñar un circuito que realice la operación aritmética:

  $O = 5X + 2Y + Z$ para operandos X (x1x0), Y (y1y0) y Z (z1z0) de dos bits, utilizando el menor número posible de semisumadores de dos bits de operandos de entradas A (a1a0) y B (b1b0).