

Organización de computadoras

Clase 1

Universidad Nacional de Quilmes

Lic. Martínez Federico

¿Qué pasó?

¿Qué pasó?

- Binario

¿Qué pasó?

- Binario
- Interpretación

¿Qué pasó?

- Binario
- Interpretación
- Representación

¿Qué pasó?

- Binario
- Interpretación
- Representación
- Aritmética

¿Qué pasó?

- Binario
- Interpretación
- Representación
- Aritmética
- Hexadecimal

¿Qué se viene para hoy?

- Compuertas lógicas:

¿Qué se viene para hoy?

- Compuertas lógicas:
 - ¿Qué?

¿Qué se viene para hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuertas básicas

¿Qué se viene para hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuertas básicas
- Circuitos

¿Qué se viene para hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuertas básicas
- Circuitos
 - Formulas y tablas de verdad

¿Qué se viene para hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuertas básicas
- Circuitos
 - Formulas y tablas de verdad
 - Producto de sumas y suma de productos

¿Qué se viene para hoy?

- Puertas lógicas:
 - ¿Qué?
 - Puertas básicas
- Circuitos
 - Fórmulas y tablas de verdad
 - Producto de sumas y suma de productos
 - Circuitos comunes

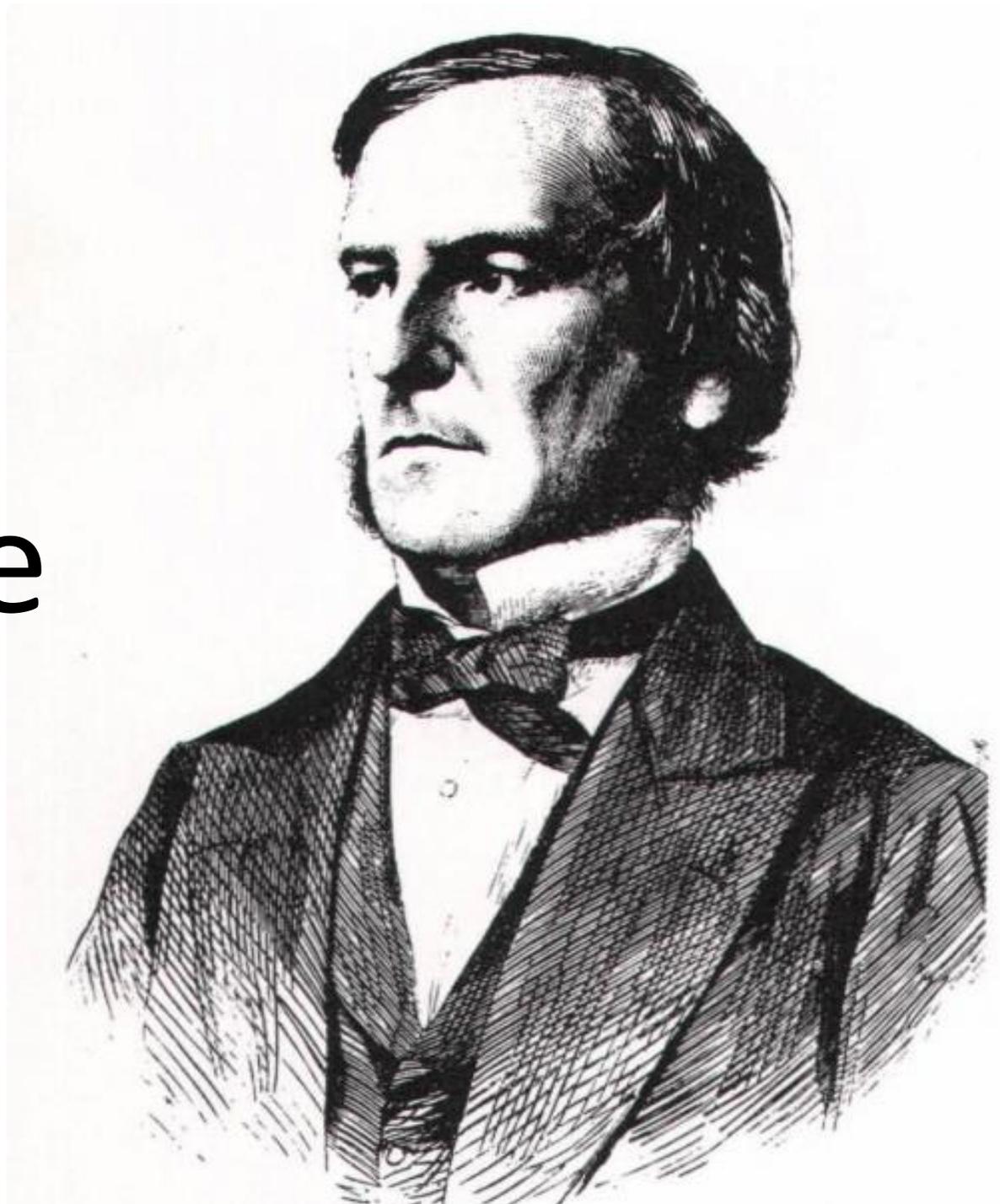
¿Qué se viene para hoy?

- Puertas lógicas:
 - ¿Qué?
 - Puertas básicas
- Circuitos
 - Fórmulas y tablas de verdad
 - Producto de sumas y suma de productos
 - Circuitos comunes
 - Circuitos aritméticos

¿cómo?



Algebra de Boole



Algebra de Boole

- Operaciones sobre binario

Algebra de Boole

- Operaciones sobre binario
- Funciones de la lógica: Or, And, Not

Algebra de Boole

- Operaciones sobre binario
- Funciones de la lógica: Or, And, Not
- Se aplican sobre valores 0 (Falso) y 1 (Verdadero)

Algebra de Boole

- Operaciones sobre binario
- Funciones de la lógica: Or, And, Not
- Se aplican sobre valores 0 (Falso) y 1 (Verdadero)
- Usando las operaciones basicas se puede construir cualquier función binaria

Or

Or

- Un or (o) es verdadero cuando algunas de sus partes es verdadera

Or

- Un or (o) es verdadero cuando algunas de sus partes es verdadera
- O llueve o vamos al parque: Tiene que valer que vayamos al parque o que llueva para que esto sea cierto. Ojo! Si valen los dos también es cierto!

Or

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

And

And

- Un and (y) es verdadero cuando ambas partes son verdaderas

And

- Un and (y) es verdadero cuando ambas partes son verdaderas
- Esta fresco y llueve: Tiene que valer que haga frío y que este lloviendo para que lo que digamos sea cierto

And

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

Not

Not

- Un not (no) es verdadero cuando lo que se niega es falso.

Not

- Un not (no) es verdadero cuando lo que se niega es falso.
- No llueve: Si está lloviendo, lo que decimos es falso

Not

A	$\neg A$
0	1
1	0

Algebra de Boole

- Combinando esos operadores podemos hacer funciones mas complejas
- $F(A,B,C) = A \wedge (B \vee \neg C)$

Algebra de Boole

- Combinando esos operadores podemos hacer funciones mas complejas
- $F(A,B,C) = A \wedge (B \vee \neg C)$
- ¿Cómo sabemos que valores de A, B y C hacen que F valga 1?

Algebra de Boole

- Combinando esos operadores podemos hacer funciones mas complejas
- $F(A,B,C) = A \wedge (B \vee \neg C)$
- ¿Cómo sabemos que valores de A, B y C hacen que F valga 1?

Tabla de verdad

¿Y la compu, amigo?





Compuertas

Compuertas

- Es un dispositivo que implementa una función booleana simple.

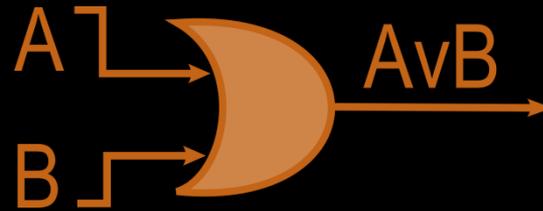
Compuertas

- Es un dispositivo que implementa una función booleana simple.
- Traduce un conjunto de entradas (una o mas) en una salida. La ausencia de electricidad es 0 y su presencia es 1.

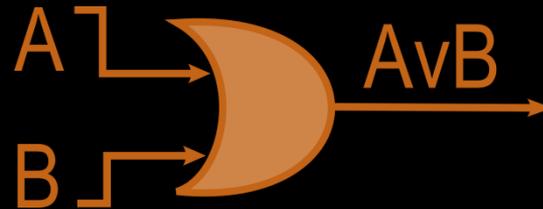
Compuertas

- Es un dispositivo que implementa una función booleana simple.
- Traduce un conjunto de entradas (una o mas) en una salida. La ausencia de electricidad es 0 y su presencia es 1.
- Son la implementación en «Los Fierros» de las operaciones que hacemos con la máquina

Compuerta OR



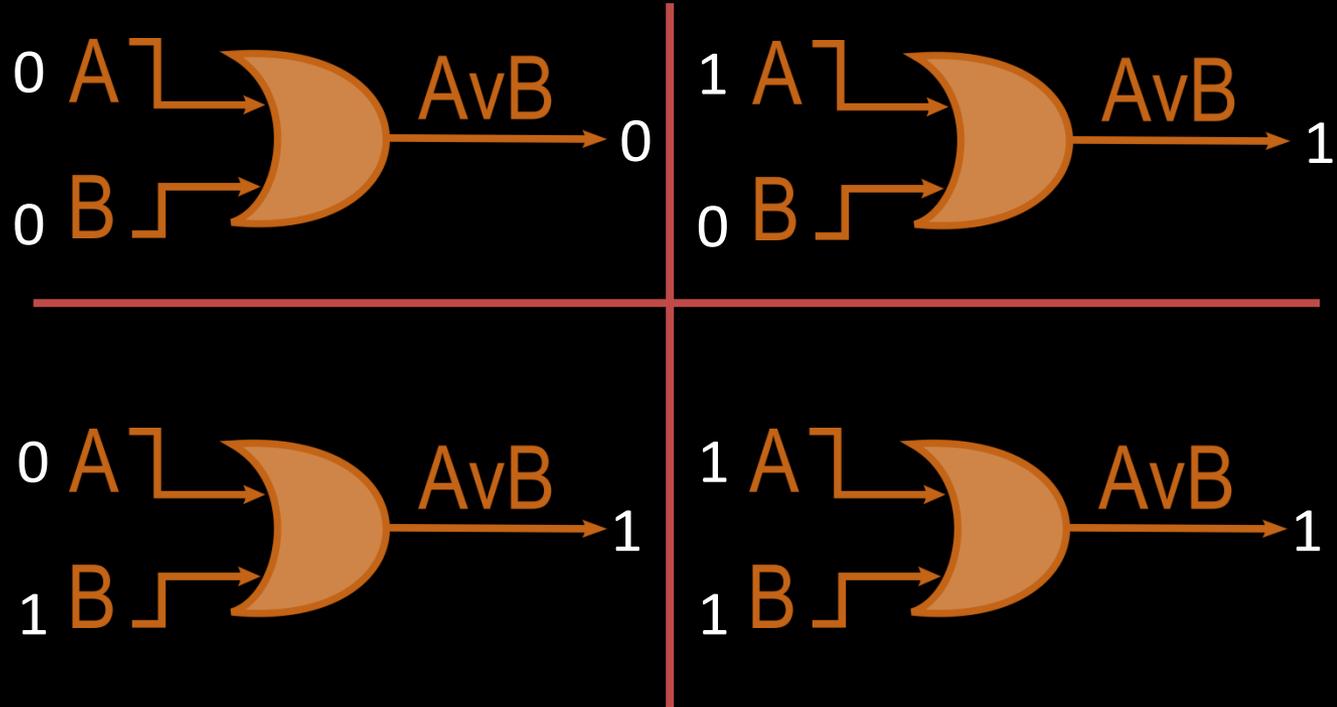
Compuerta OR



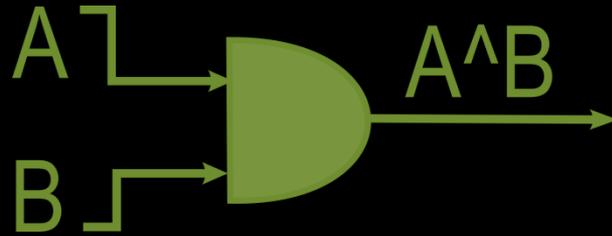
A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

Compuerta OR

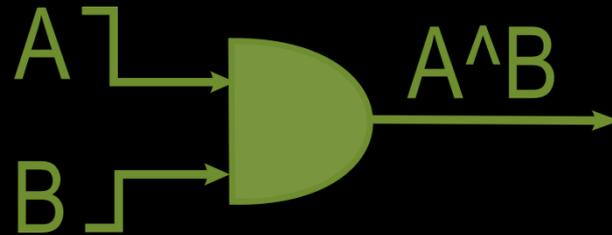
A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1



Compuerta AND

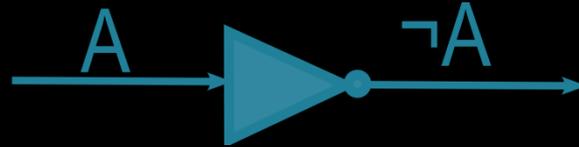


Compuerta AND

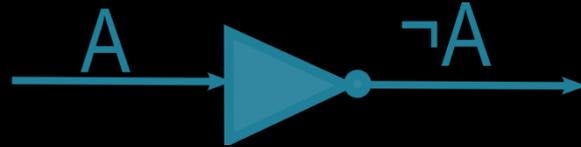


A	B	A^B
0	0	0
0	1	0
1	0	0
1	1	1

Compuerta NOT



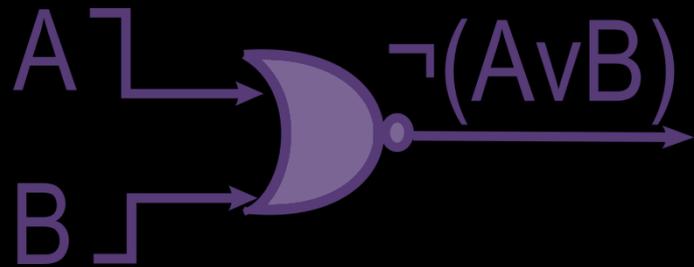
Compuerta NOT



A	$\neg A$
0	1
1	0

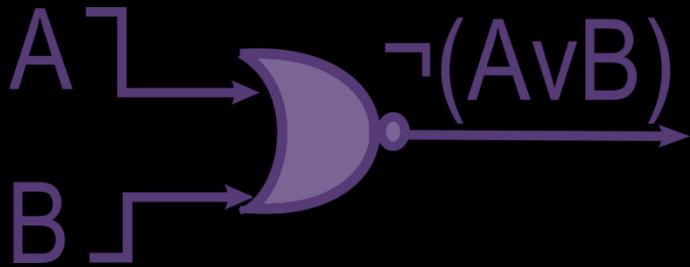
Otras

Otras

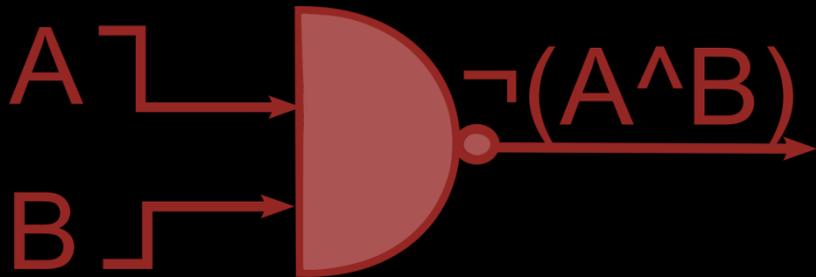


Compuerta NOR

Otras

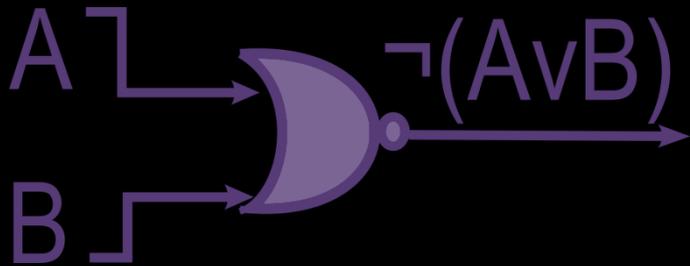


Compuerta NOR

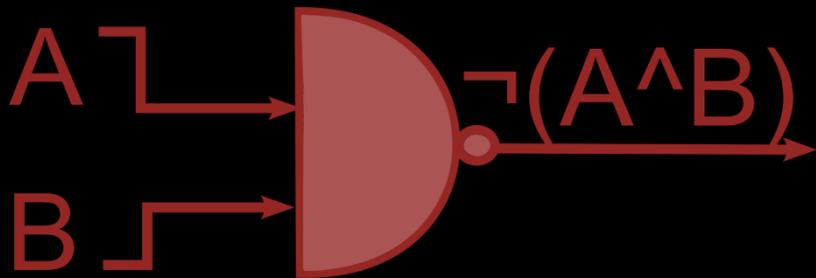


Compuerta NAND

Otras



Compuerta NOR



Compuerta NAND



Compuerta XOR

Circuitos

The image features a complex, glowing blue circuit board pattern on a black background. The circuit lines are bright and have a slight blur, giving a sense of depth and movement. The word "Circuitos" is written in a bold, red, sans-serif font, centered in the upper portion of the image. The overall aesthetic is futuristic and technological.

Circuitos

- Traducen un conjunto de entradas en un conjunto de salidas.

Circuitos

- Traducen un conjunto de entradas en un conjunto de salidas.
- Una o mas funciones booleanas.

Circuitos

- Traducen un conjunto de entradas en un conjunto de salidas.
- Una o mas funciones booleanas.
- Se obtienen combinando compuertas.

Circuitos

- Se pueden construir a partir de una fórmula booleana o a partir de una tabla de verdad
- Ejemplo: Construir un circuito que compute cada una de las siguientes funciones:
 - $B \wedge (C \vee A)$
 - $(A \wedge B) \vee (\neg A \wedge C)$

Ejercicio

- Realizar un circuito de 3 entradas que compute la función mayoría, es decir, si dos o mas entradas valen 1 debe obtenerse un 1, y un 0 si no.

Circuitos

- ¿Cómo pasar de la tabla al circuito?

Circuitos

- Suma de productos:
 - Una formula tiene la forma de suma de productos si tiene la siguiente pinta:
 - $A_1 \vee A_2 \vee A_3 \vee \dots \vee A_n$, donde cada A_i usa solo and y not
 - Ej: $(A \wedge \neg B) \vee (\neg A \wedge C) \vee (B \wedge C)$

Circuitos

- Producto de sumas:
 - Una formula tiene la forma de producto de sumas si tiene la siguiente pinta:
 - $A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_n$, donde cada A_i usa solo or y not
 - Ej: $(A \vee \neg B) \wedge (\neg A \vee C) \wedge (B \vee C)$

Circuitos

- ¿Cómo pasar de la tabla al circuito?
 1. Armamos la tabla
 2. Si hay menos filas con resultado 1:
 1. Escribimos un producto por cada una de estas filas
 2. Las sumamos
 3. Armamos el circuito a partir de la fórmula
 3. Si hay menos filas con resultado 0:
 1. Escribimos una suma por cada una de estas filas
 2. Hacemos el producto entre ellas
 3. Armamos el circuito a partir de la fórmula

Ejemplo

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Ejemplo

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Ejercicio

- Realizar un circuito de 3 entradas que compute la función mayoría, es decir, si dos o mas entradas valen 1 debe obtenerse un 1, y un 0 si no.

Circuitos útiles

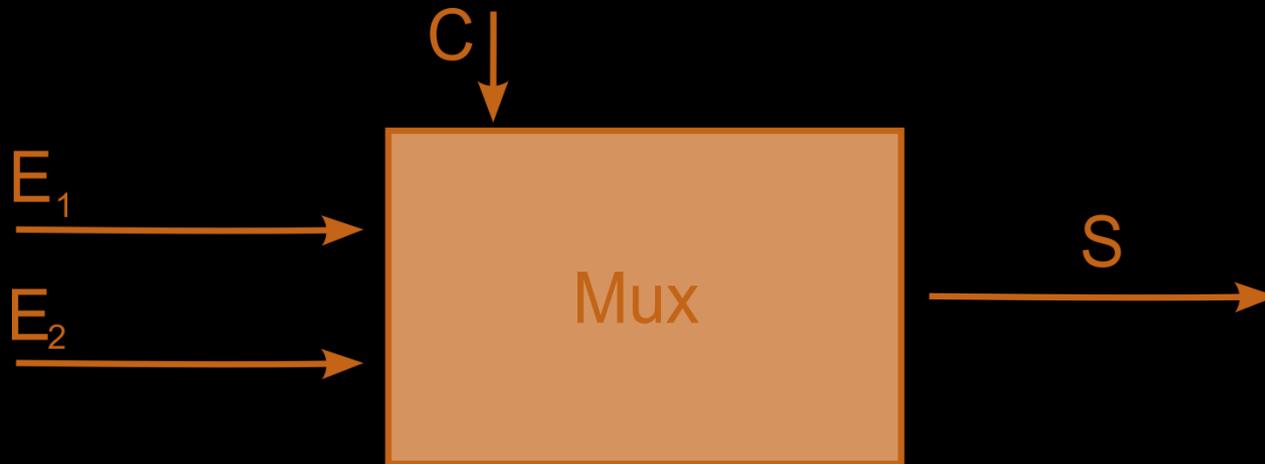
Multiplexor simple

Multiplexor simple

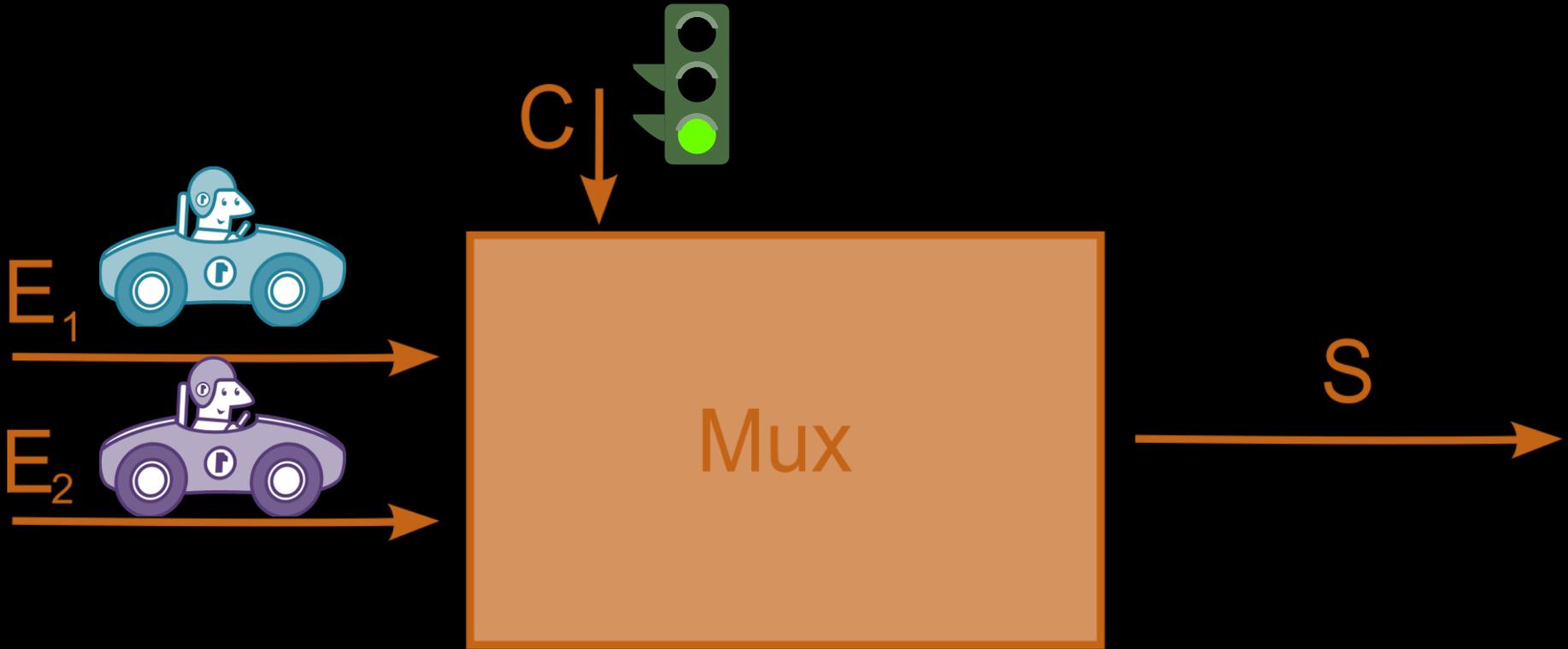
- 2 entradas
- 1 salida
- Una línea de control que elige cuál de las entradas se proyecta a la salida.

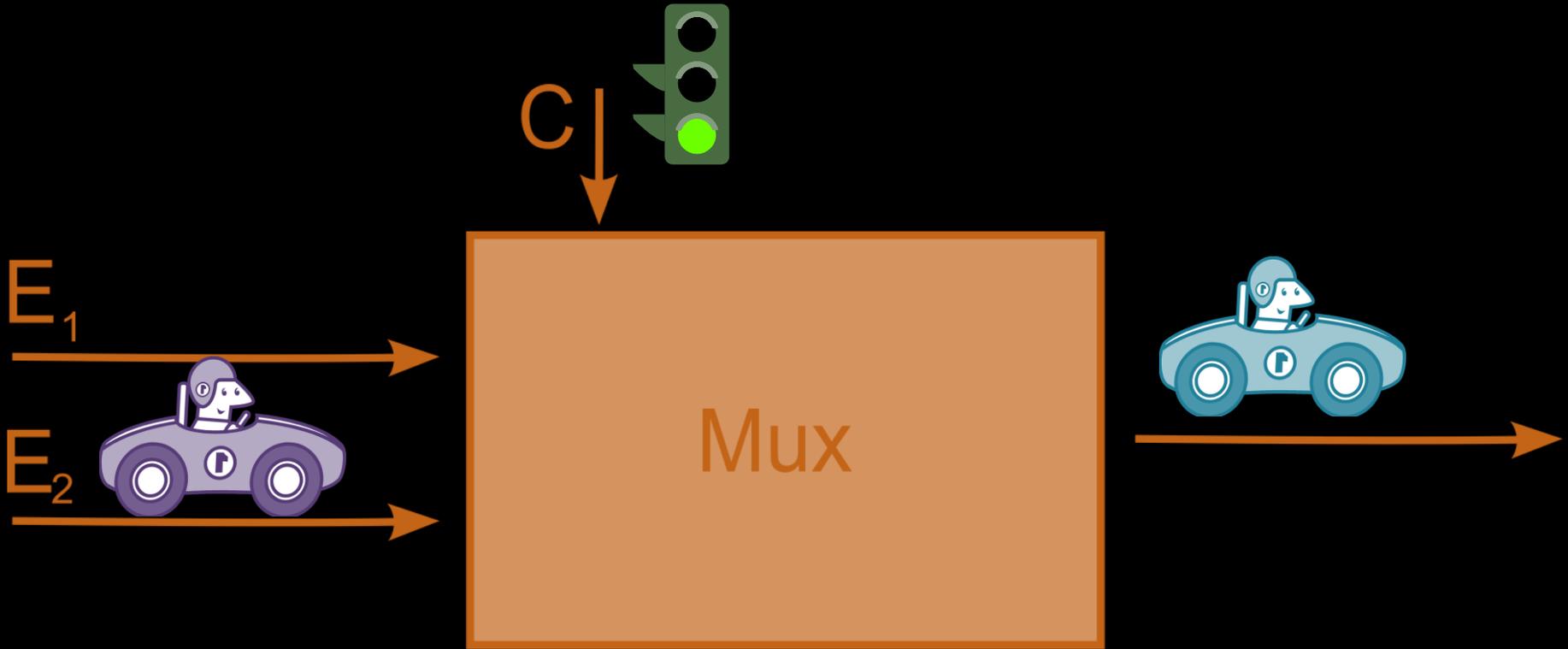
Multiplexor simple

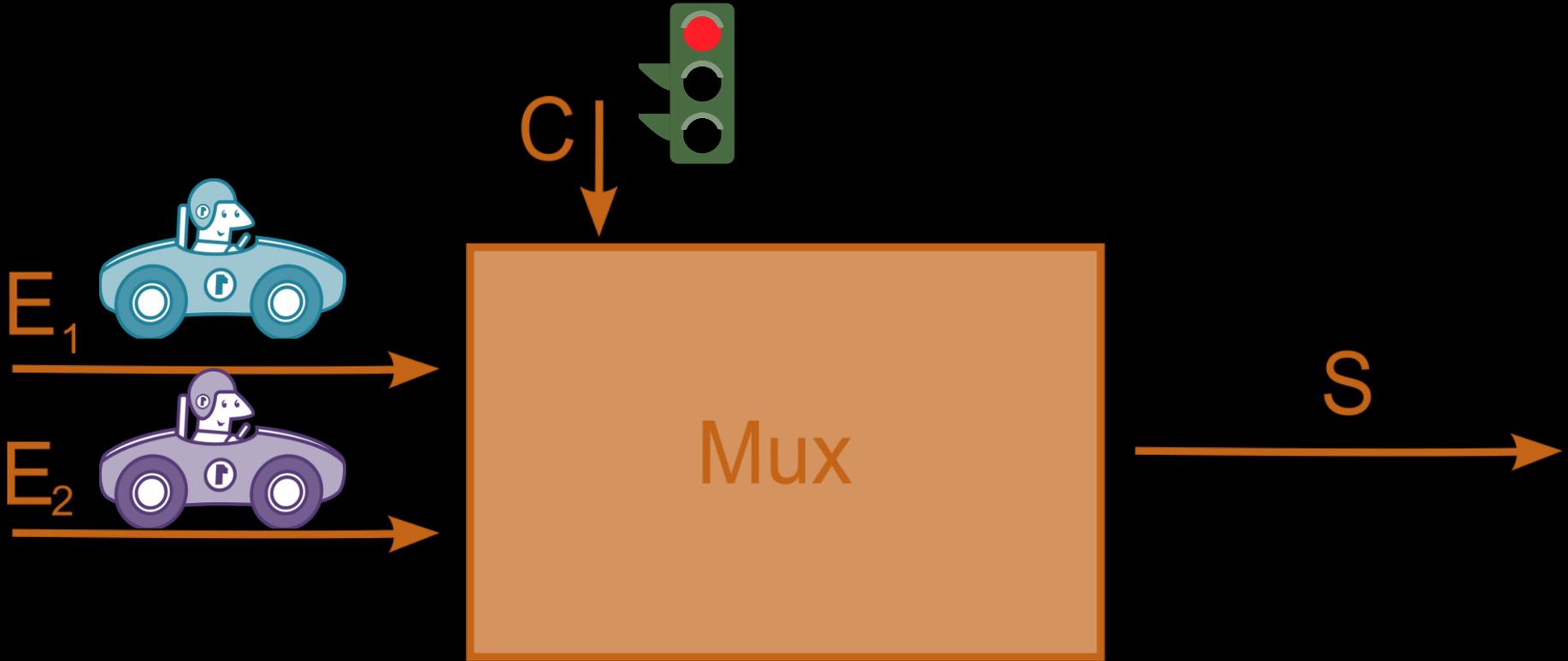
- 2 entradas
- 1 salida
- Una línea de control que elige cuál de las entradas se proyecta a la salida.

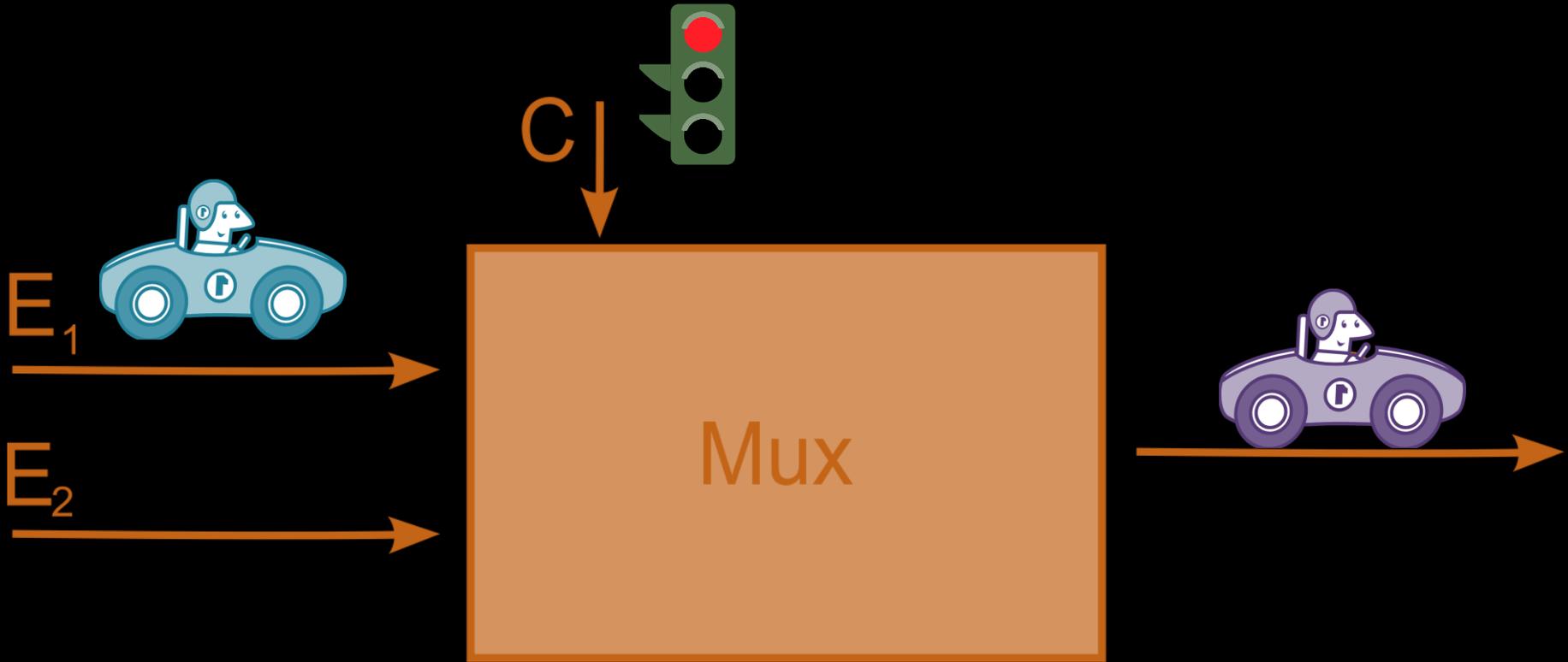












Multiplexor simple

- Tabla abreviada:

C	S
0	E1
1	E2

Multiplexor Simple

- Tabla completa:

C	E1	E2	S
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Multiplexor Simple

- Tabla completa:

C	E1	E2	S
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Multiplexor de 4 entradas

- 4 entradas
- 1 salida
- 2 líneas de control



Multiplexor de 4 entradas

- Tabla abreviada:

C1	C2	S
0	0	E1
0	1	E2
1	0	E3
1	1	E4

Multiplexor de 4 entradas

- Tabla complete y circuito

Multiplexor de 4 entradas

- Tabla complete y circuito

!!!TAREA!!!

Decodificador

- 2 entradas
- 4 salidas

Decodificador

- 2 entradas
- 4 salidas



Decodificador

- 2 entradas
- 4 salidas



Decodificador

- 2 entradas
- 4 salidas



Decodificador

- 2 entradas
- 4 salidas



00 -> 0

Decodificador

- 2 entradas
- 4 salidas



00 -> 0

Decodificador

- 2 entradas
- 4 salidas



00 -> 0 01 -> 1

Decodificador

- 2 entradas
- 4 salidas



00 -> 0 01 -> 1

Decodificador

- 2 entradas
- 4 salidas



00 -> 0 01 -> 1 10 -> 2

Decodificador

- Tabla:

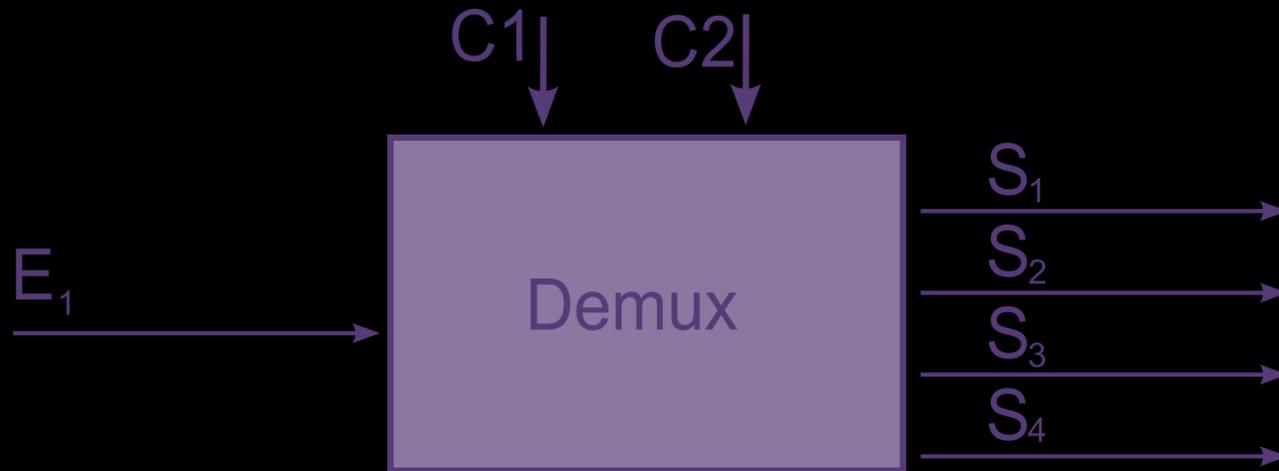
E1	E2	S1	S2	S3	S4
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Demultiplexor

- 1 Entrada
- 2 Entradas de control
- 4 salidas

Demultiplexor

- 1 Entrada
- 2 Entradas de control
- 4 salidas



Demultiplexor

- Tabla:

E	C1	C2	S1	S2	S3	S4
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

Circuitos aritméticos

- Implementan funciones aritméticas, como la suma

Half adder

- Suma dos bits
- 2 Entradas:
 - Los bits a sumar
- 2 Salidas:
 - La suma
 - El carry

Half adder

- Tabla:

X1	X2	S	C
0	0		
0	1		
1	0		
1	1		

Half adder

- Tabla:

X1	X2	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Full adder

- Suma dos bits
- 3 entradas
 - Los dos bits a sumar
 - El carry “anterior”
- 2 Salidas:
 - La suma
 - El carry de salida

Full adder

- Tabla:

X1	X2	Ci	S	Co
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

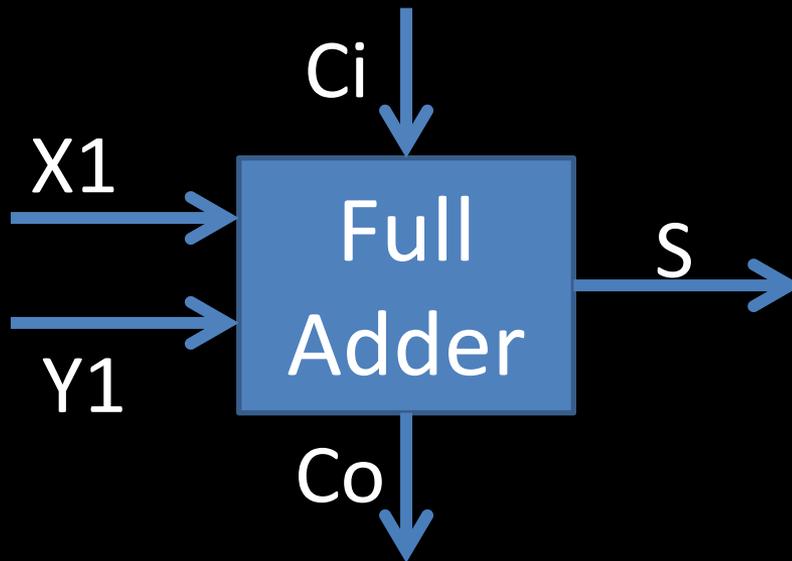
Full adder

- Tabla:

X1	X2	Ci	S	Co
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

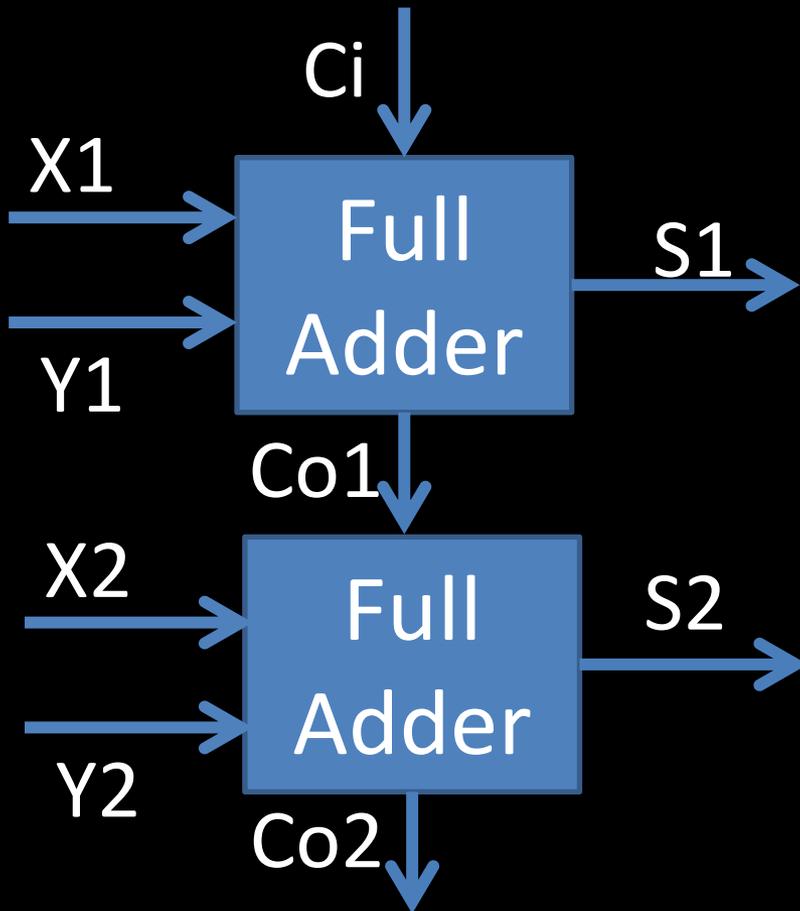
Sumar varios bits

- Una vez que ya tenemos armado el full adder de un bit, ¿Cómo puedo sumar varios bits?



Sumar varios bits

- Una vez que ya tenemos armado el full adder de un bit, ¿Cómo puedo sumar varios bits?



Restador

- Resta dos bits
- 2 Entradas:
 - Los bits a restar
- 2 Salidas:
 - La resta
 - El borrow (Le pedí uno al compañero)

Restador

- Tabla:

X1	X2	R	B
0	0		
0	1		
1	0		
1	1		

Restador

- Tabla:

X1	X2	R	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

¿Qué pasó hoy?

- Compuertas lógicas:

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuerta OR

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuerta OR
 - Compuerta AND

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuerta OR
 - Compuerta AND
 - Compuerta NOT

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuerta OR
 - Compuerta AND
 - Compuerta NOT
 - Otras compuertas

¿Qué pasó hoy?

- Puertas lógicas:
 - ¿Qué?
 - Puerta OR
 - Puerta AND
 - Puerta NOT
 - Otras puertas
- Circuitos

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuerta OR
 - Compuerta AND
 - Compuerta NOT
 - Otras compuertas
- Circuitos
 - Formulas y tablas de verdad

¿Qué pasó hoy?

- Puertas lógicas:
 - ¿Qué?
 - Puerta OR
 - Puerta AND
 - Puerta NOT
 - Otras puertas
- Circuitos
 - Fórmulas y tablas de verdad
 - Producto de sumas y suma de productos

¿Qué pasó hoy?

- Puertas lógicas:
 - ¿Qué?
 - Puerta OR
 - Puerta AND
 - Puerta NOT
 - Otras puertas
- Circuitos
 - Fórmulas y tablas de verdad
 - Producto de sumas y suma de productos
 - Circuitos comunes

¿Qué pasó hoy?

- Compuertas lógicas:
 - ¿Qué?
 - Compuerta OR
 - Compuerta AND
 - Compuerta NOT
 - Otras compuertas
- Circuitos
 - Formulas y tablas de verdad
 - Producto de sumas y suma de productos
 - Circuitos comunes
 - Circuitos aritméticos

Bibliografía

- Organización y Arquitectura de computadoras, Stallings, Apéndice A: Lógica digital (Notar que el libro muestra mas circuitos que los vistos en clase y llega a un nivel de detalle mayor)





Thank You