

# ***Introducción al Diseño Lógico (E0301)***

***Ingeniería en Computación***

***Gerardo E. Sager***

***Clase 5 curso 2024***

# Clase 5 Parte I

- Temas a tratar
  - Dualidad
  - Algebra Booleana y el mapa de Karnaugh como herramientas para simplificar y diseñar circuitos lógicos II
  - Operación de circuitos OR-Exclusiva y NOR-exclusiva. Aplicaciones.
  - Circuitos de Habilitación y Deshabilitación.

# Mapas de Karnaugh o K-Maps

- **Dualidad:** Si en una expresión booleana se reemplazan todas las variables por sus expresiones negadas, las operaciones OR por AND y viceversa, la expresión resultante es igual a la expresión original pero negada. A esta propiedad la llamamos DUALIDAD.
  - Basándose en la Dualidad se pueden construir K-Maps, orientados a simplificar las expresiones escribiéndolas como POS. donde las filas y columnas en vez de corresponder al AND de variables, corresponden al OR y además se agrupan los valores '0' en vez de los valores '1'.
  - No vamos a profundizar en este tema, porque vamos a utilizar otra manera de construir los K-Maps ya sea a partir de POS o SOP

# Mapas de Karnaugh o K-Maps

- Cómo llenar un **mapa K** a partir de una expresión Booleana de tipo **SOP**
  - Si la expresión está expresada como SOP puede llenarse el K-MAP, poniendo '1's en los lugares correspondientes a los Productos y '0's en los restantes
- Cómo llenar un **mapa K** a partir de una expresión Booleana de tipo **POS**
  - Si bien se podría aplicar la propiedad de dualidad y construir un K-MAP basado en POS, lo más práctico es negar la expresión propuesta y aplicar deMorgan para convertirla en SOP. Esto describirá la función negada y a partir de allí podremos simplificarla:
  - $F=(A+B)(C+D) \Rightarrow \bar{F} = \overline{(A+B)(C+D)} = \overline{(A+B)} + \overline{(C+D)} = \bar{A} \bar{B} + \bar{C} \bar{D}$
  - A partir de allí podemos agrupar los valores necesarios, obtener la expresión simplificada y luego volverla a negar y aplicar DeMorgan nuevamente.

# Ejemplo:

- Volvemos al ejemplo de la presentación 7, pagina 7. para el caso de POS
- El enunciado pedía diseñar un circuito lógico de tres entradas, A, B, y C. La salida debe ser HIGH cuando la mayoría de las entradas es HIGH.

A	B	C	F	$\bar{F}$	POS
0	0	0	0	1	$\bar{A} \bar{B} \bar{C}$
0	0	1	0	1	$\bar{A} \bar{B} C$
0	1	0	0	1	$\bar{A} B \bar{C}$
0	1	1	1	0	
1	0	0	0	1	$A \bar{B} \bar{C}$
1	0	1	1	0	
1	1	0	1	0	
1	1	1	1	0	

Habíamos obtenido a partir de

$$\bar{F} = \bar{A} \bar{B} \bar{C} + \bar{A} \bar{B} C + \bar{A} B \bar{C} + A \bar{B} \bar{C}$$

$$F = (A+B+C)(A+B+\bar{C})(A+\bar{B}+C)(\bar{A}+B+C)$$

Si aplicamos K-MAP a  $\bar{F}$

	BC	00	01	11	10
A		$\bar{B} \bar{C}$	$\bar{B} C$	$B C$	$B \bar{C}$
0	$\bar{A}$	1	1	0	1
1	A	1	0	0	0

$$\bar{F} = \bar{A} \bar{B} + \bar{B} \bar{C} + \bar{A} \bar{C} = \overline{A+B} + \overline{B+C} + \overline{A+C}$$

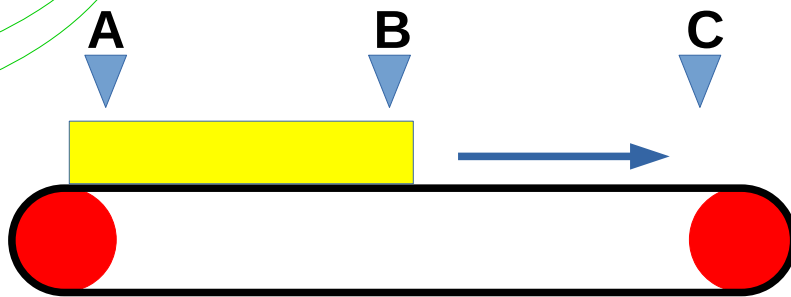
$$\bar{F} = \overline{A+B} + \overline{B+C} + \overline{A+C} = (A+B)(B+C)(A+C)$$

***Se observa que utilizar el K-MAP nos dá una expresión más simple***

# Valores “Don't Care” o “No Importa”

- En algunos casos, cuando escribimos la TdV , hay situaciones en que no nos interesa el valor exacto de la salida cuando se produce cierta combinación de entradas.
- Por ejemplo, si tenemos un circuito de dos entradas, que sabemos que por una restricción externa nunca van a adoptar simultáneamente el valor '1'
- En ese caso, no nos importaría el valor que adopte la salida, y lo marcamos como “don't care” o “No importa” que se marca como X.

# Valores “Don't Care” o “No Importa”



	$\bar{B}\bar{C}$	$B\bar{C}$	$BC$	$\bar{B}C$
$\bar{A}$	0	0	1	0
A	0	1	X	X

$$F = \bar{A}B\bar{C} + \bar{A}BC \quad (X=0; X=0)$$

$$F = AB + BC \quad (X=1; X=0)$$

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

Supongamos un circuito con tres entradas en las cuáles no hay posibilidad de que las tres entradas tengan el mismo valor.

Por ejemplo podría ser una cinta transportadora donde las entradas son sensores que me indican la presencia de un objeto con un '1' y la ausencia con '0',

Si el sistema es como el de la figura, nunca vamos a tener tres sensores activados a la vez. Tampoco se activarán simultáneamente los sensores A y C

Si queremos que la salida valga 1 cuando hay dos sensores activados, escribimos la TdV y el K-Map

Cuando ubicamos la condición X en el K-MAP, depende si le asignamos valor 0 o 1, obtenemos expresiones más simples o más complejas

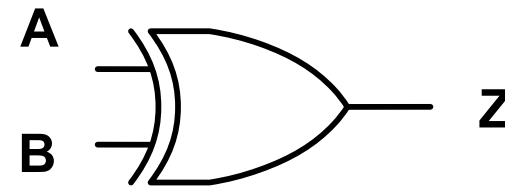
# Circuitos XOR y XNOR

- OR exclusiva (**XOR**)

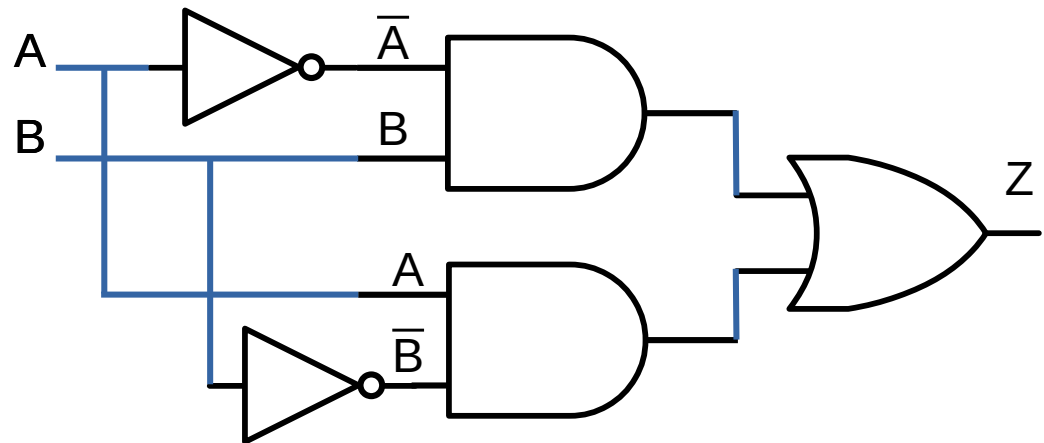
TdV		
A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

K-MAP		
	$\bar{B}$	B
$\bar{A}$	0	1
A	1	0

Símbolo



Implementación



$$A \text{ XOR } B = \bar{A}B + A\bar{B}$$





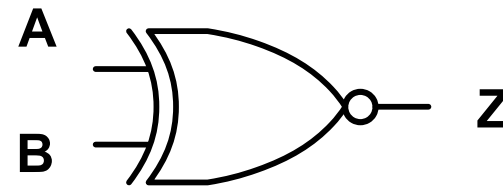
# Circuitos XOR y XNOR

- NOR exclusiva (**XNOR**)

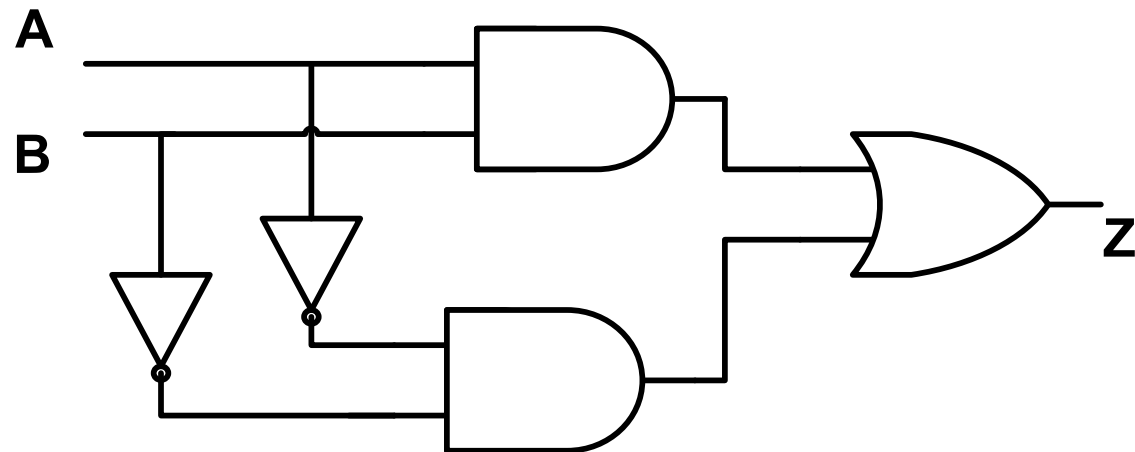
TdV		
A	B	Z
0	0	1
0	1	0
1	0	0
1	1	1

K-MAP		
	$\overline{B}$	B
$\overline{A}$	1	0
A	0	1

Símbolo



Implementación

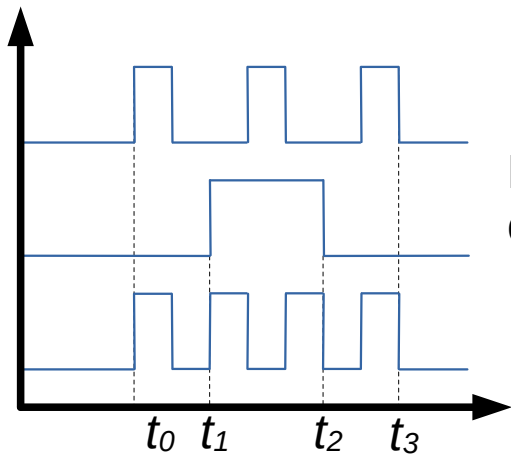


$$A \text{ XNOR } B = \overline{A} \overline{B} + A B$$

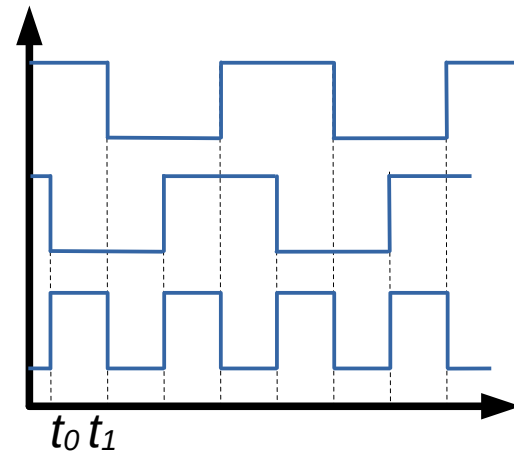


# Circuitos XOR y XNOR

- OR exclusiva (**XOR**) Salida en el tiempo para diferentes entradas

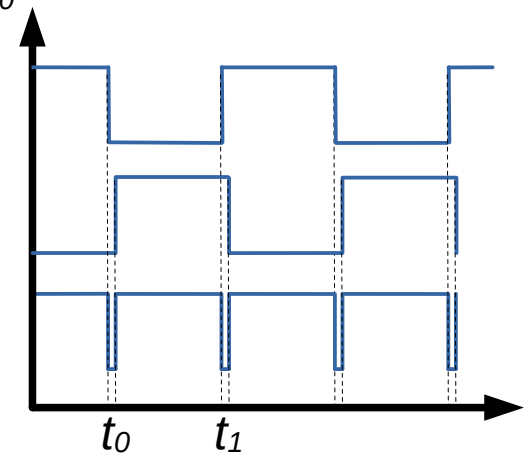
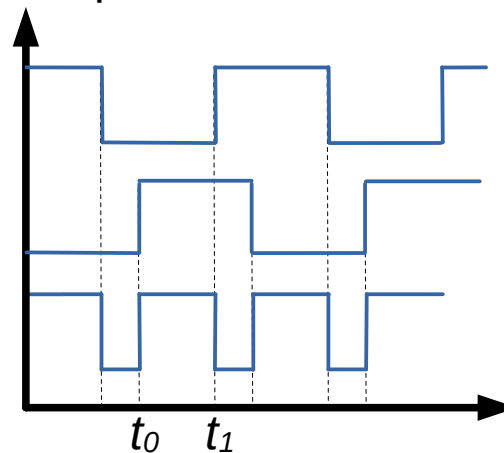
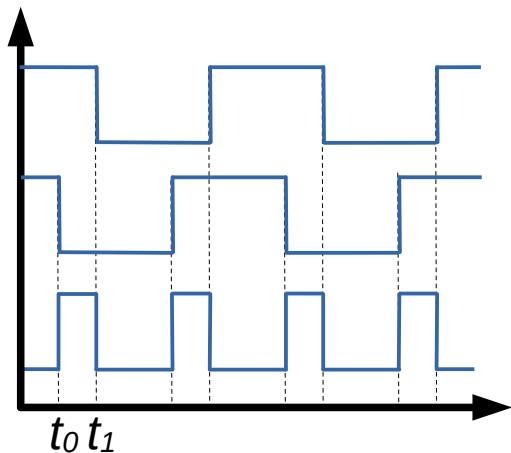


Dos entradas  
cualquiera

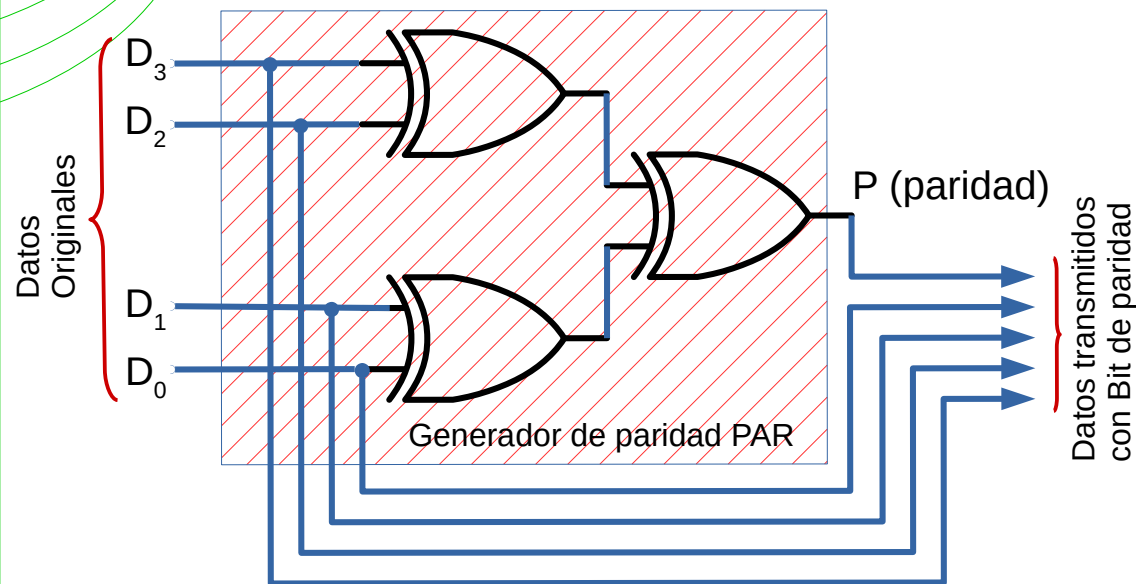


Las entradas  
son dos  
señales de la  
misma  
frecuencia,  
pero defasadas  
90°

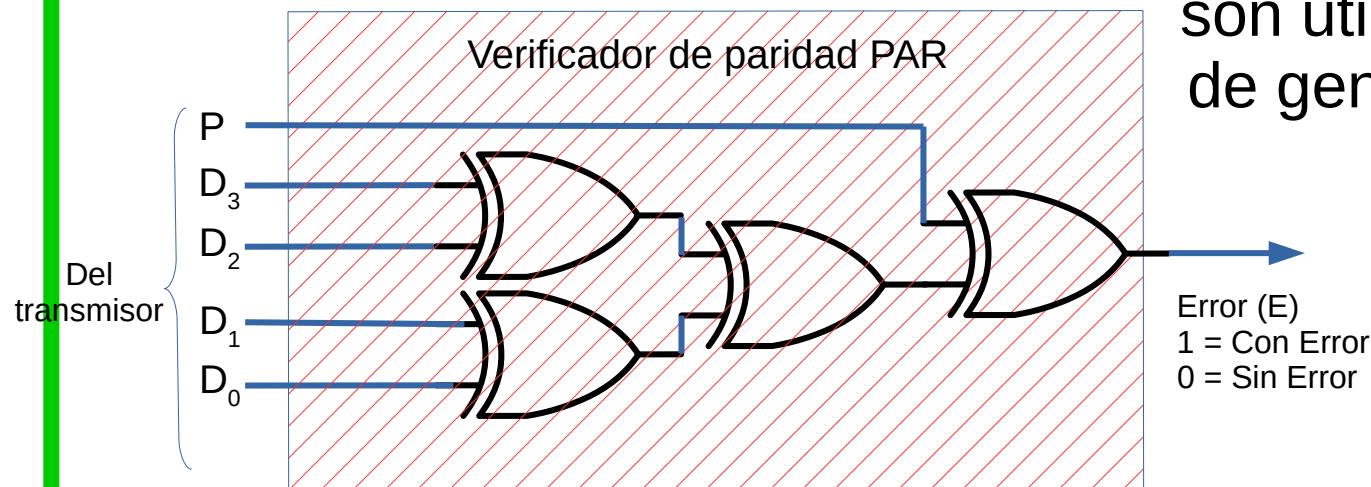
Seguimos con entradas de la misma frecuencia, con distintos defasajes  
Notar como varía el ancho de los pulsos de salida  $\Delta t = t_1 - t_0$



# Generador y Verificador de Paridad

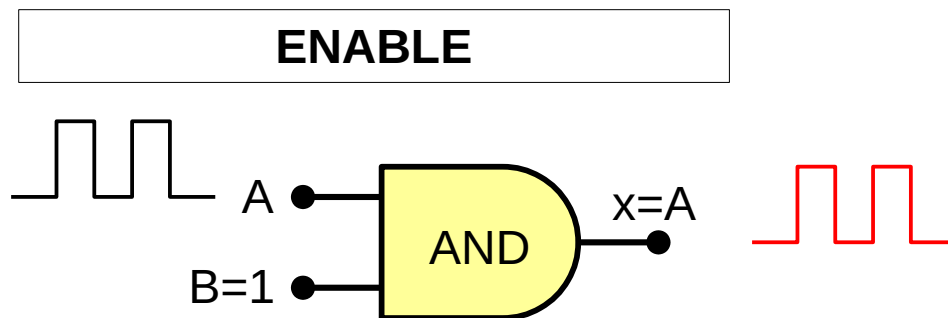


Las compuertas XOR y XNOR son útiles para diseñar circuitos de generación y verificación de paridad

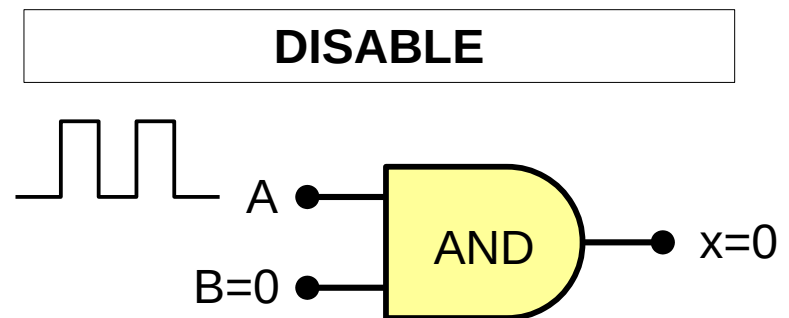


# Circuitos de habilitación y deshabilitación

- En el diseño de circuitos digitales ocurren frecuentemente situaciones donde se requiere circuitos de habilitación/deshabilitación .
  - Un circuito está **habilitado (enabled)** cuando permite el paso de la señal de entrada a la salida.
  - Un circuito está **deshabilitado (disabled)** cuando evita el paso de la señal de entrada a la salida.

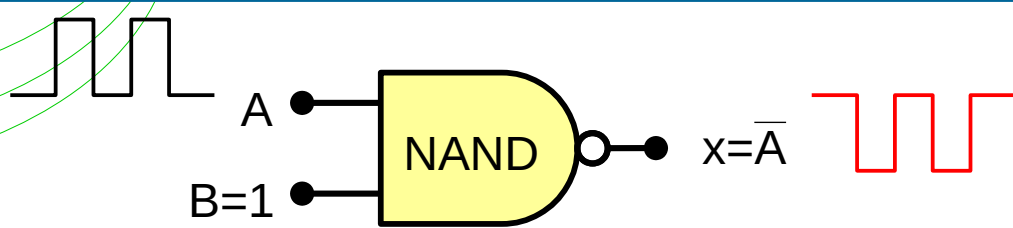


B=1, HABILITA que pase la señal A

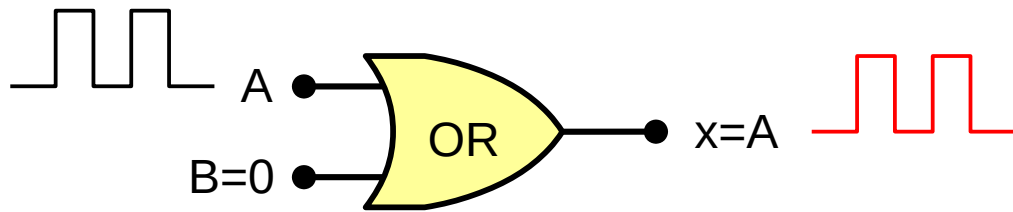


B=0, DESHABILITA el paso de la señal A

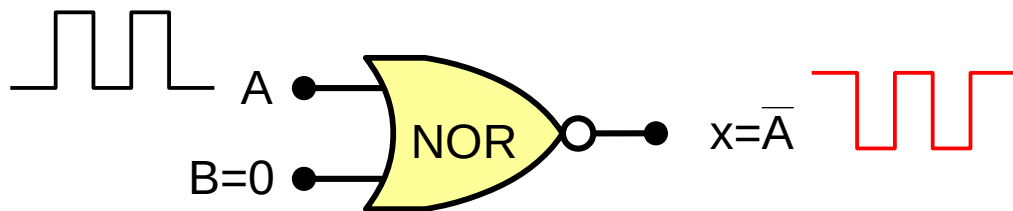
# Circuitos de habilitación y deshabilitación



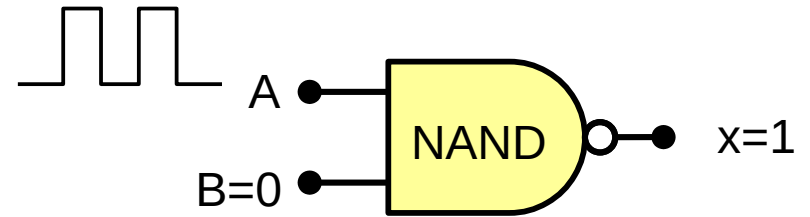
B=1, HABILITA que pase la señal A INVERTIDA a la salida



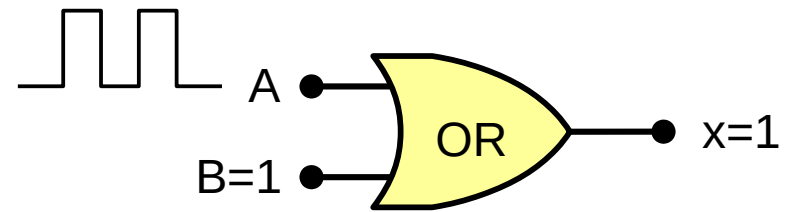
B=0, HABILITA que pase la señal A a la salida



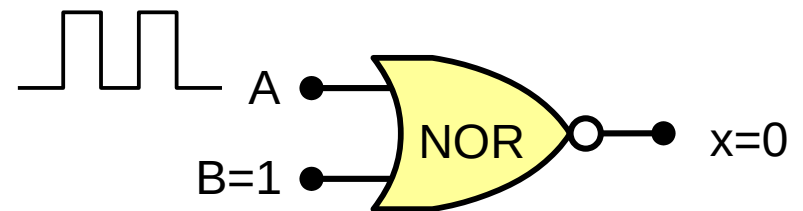
B=0, HABILITA que pase la señal A INVERTIDA a la salida



B=0, DESHABILITA que pase la señal A a la salida

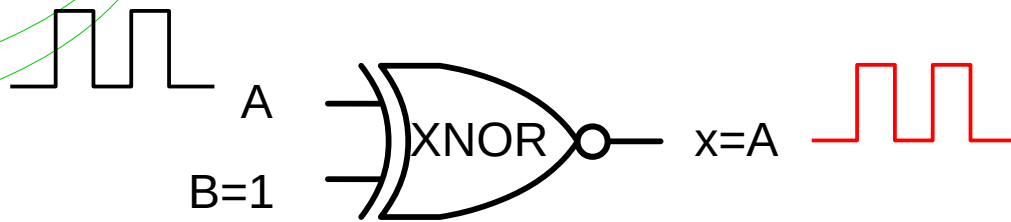


B=1, DESHABILITA que pase la señal A a la salida

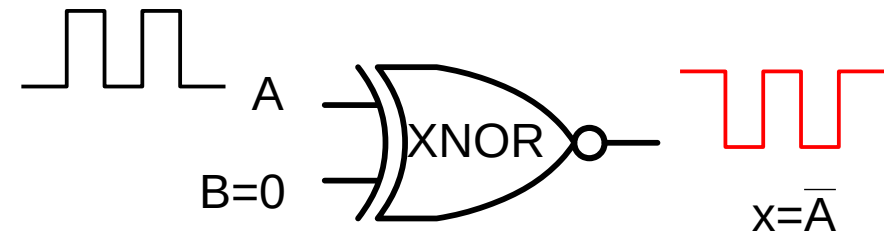


B=1, DESHABILITA que pase la señal A a la salida

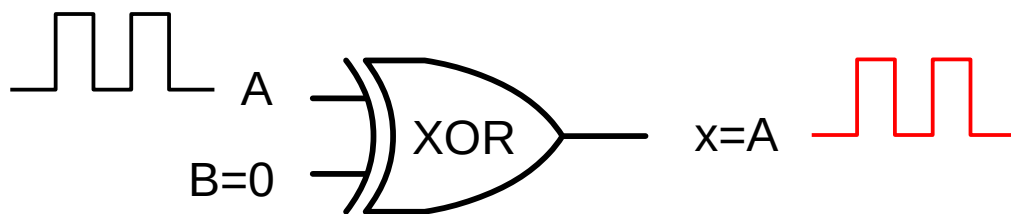
# Circuitos inversores con XOR y XNOR



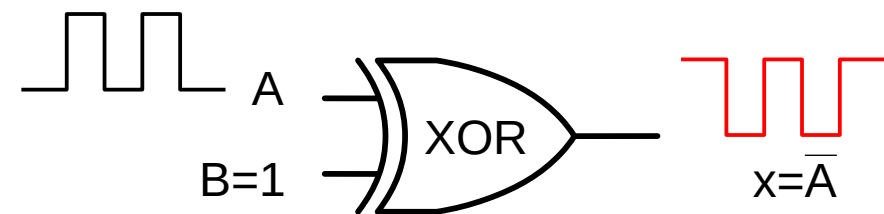
B=1, HABILITA que pase la señal A a la salida



B=0, HABILITA que pase la señal A INVERTIDA a la salida



B=0, HABILITA que pase la señal A a la salida



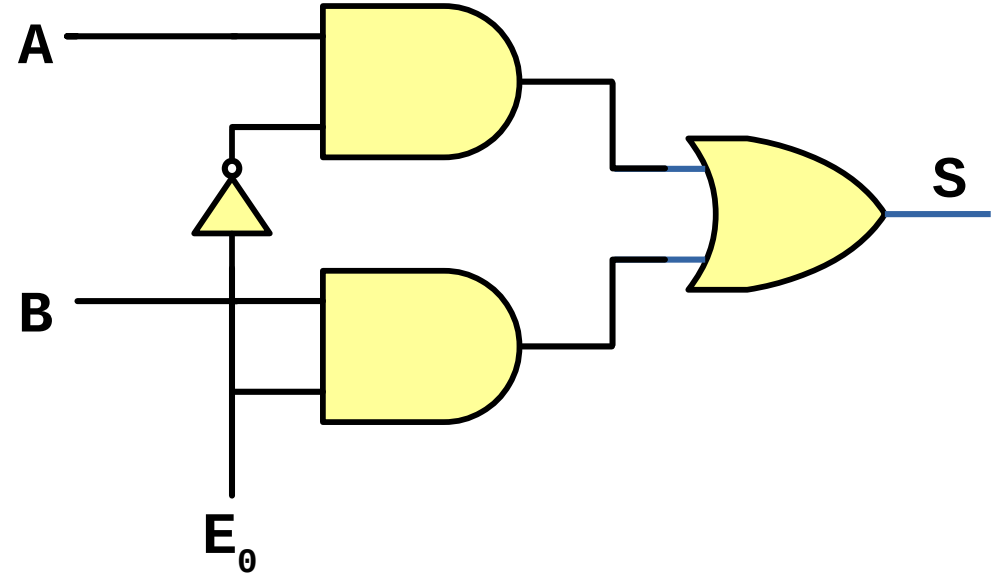
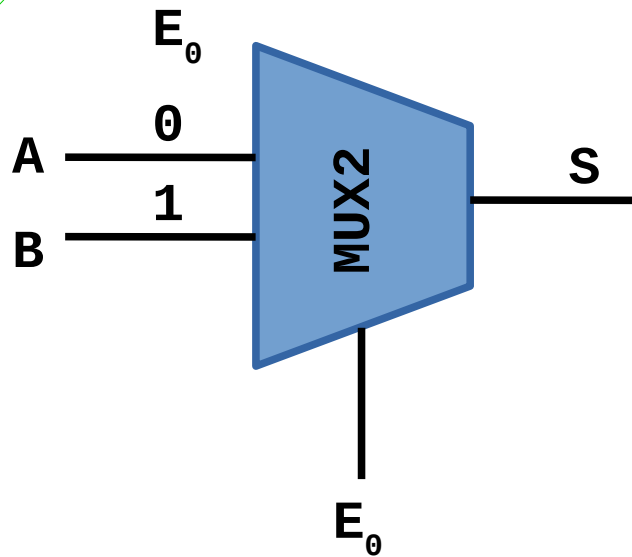
B=1, HABILITA que pase la señal A INVERTIDA a la salida

**Con una compuerta XOR o XNOR puedo hacer un INVERSOR CONFIGURABLE**

# Parte II

- Temas a tratar
  - Multiplexores
  - Uso de Multiplexores para implementar funciones lógicas.
  - Componentes pasivos: Llaves Normalmente Abiertas (NA) y Normalmente Cerradas (NC).
  - Componentes Activos: Diodos – LED – Transistores Bipolares – MOSFET – CMOS.
  - Uso como llaves de MOSFETs PMOS y NMOS
  - INVERSOR, AND y OR implementados como circuitos con llaves.
  - INVERSOR, NAND y NOR implementados como circuitos CMOS.
  - Efectos producidos por la implementación real. Tiempo de propagación, y Resistencia de salida.
  - Pautas para representar esquemas de circuitos correctamente

# Multiplexor de dos entradas (MUX2)

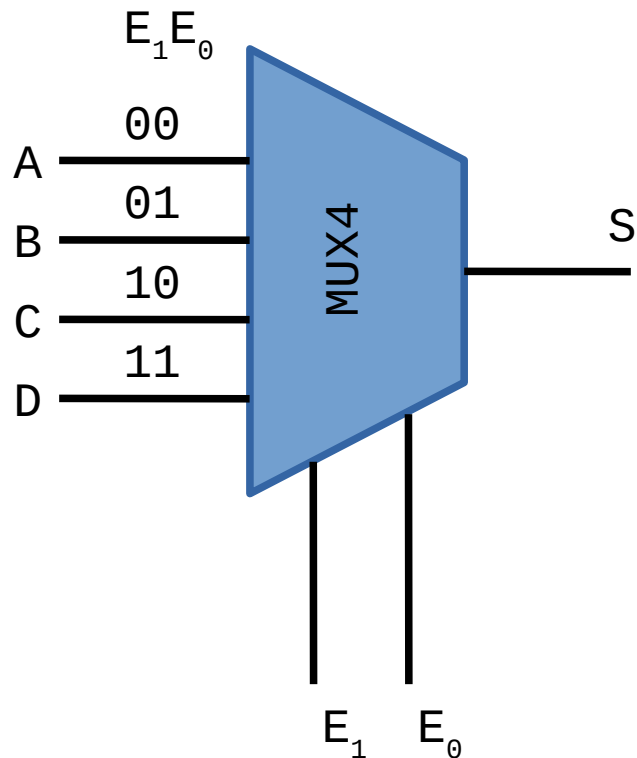


E <sub>0</sub>	S
0	A
1	B

$$S = \overline{E_0} A + E_0 B$$



# Multiplexor de cuatro entradas (MUX4)

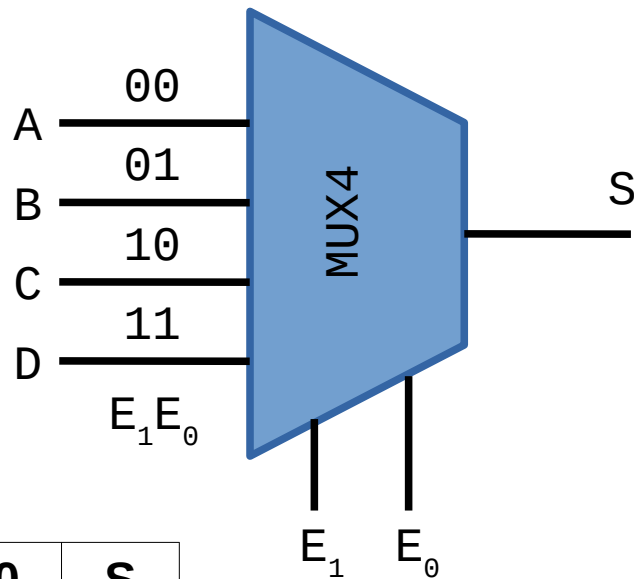


$$S = \bar{E}_1 \bar{E}_0 A + \bar{E}_1 E_0 B + E_1 \bar{E}_0 C + E_1 E_0 D$$

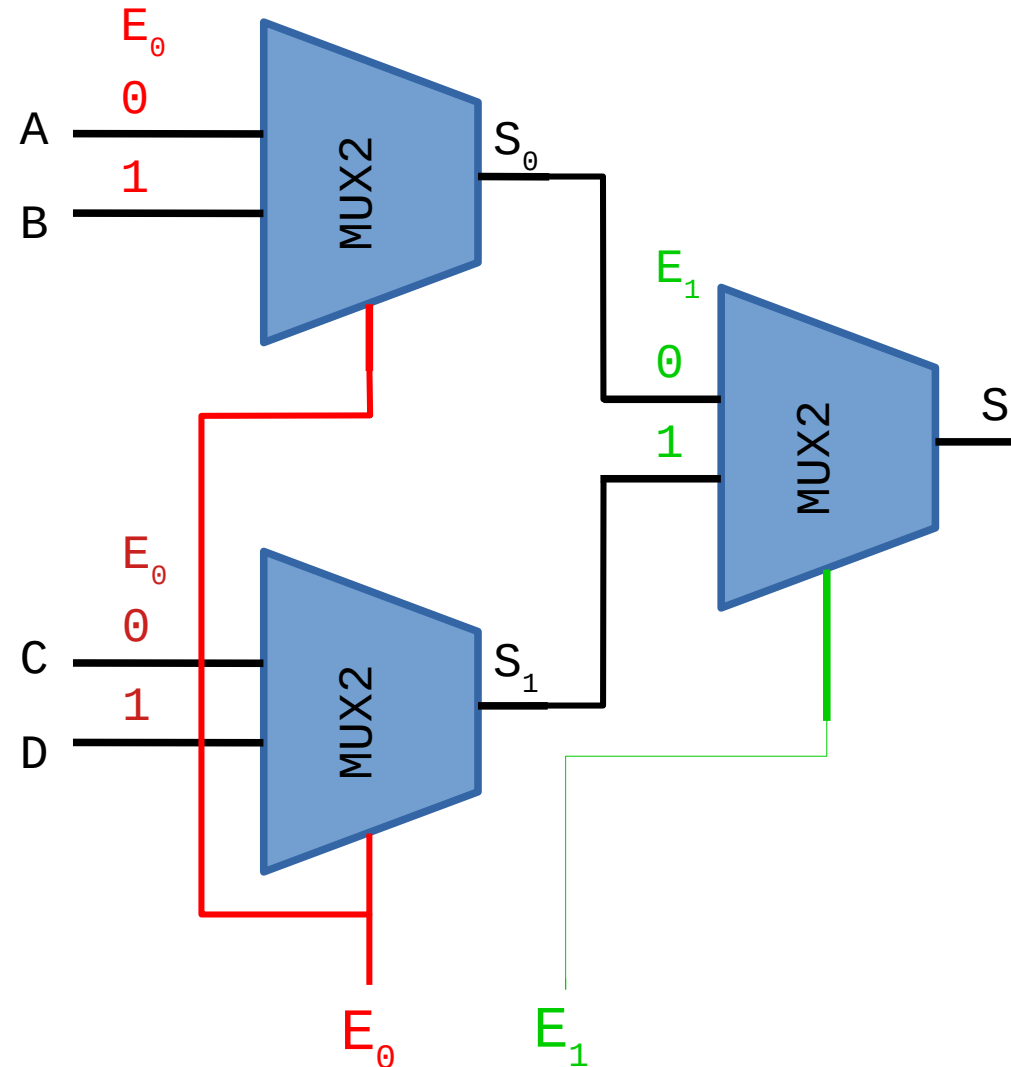
E1	E0	S
0	0	A
0	1	B
1	0	C
1	1	D

# Implementar un MUX4 a partir de MUX2

$$S = \bar{E}_1 \bar{E}_0 A + \bar{E}_1 E_0 B + E_1 \bar{E}_0 C + E_1 E_0 D$$

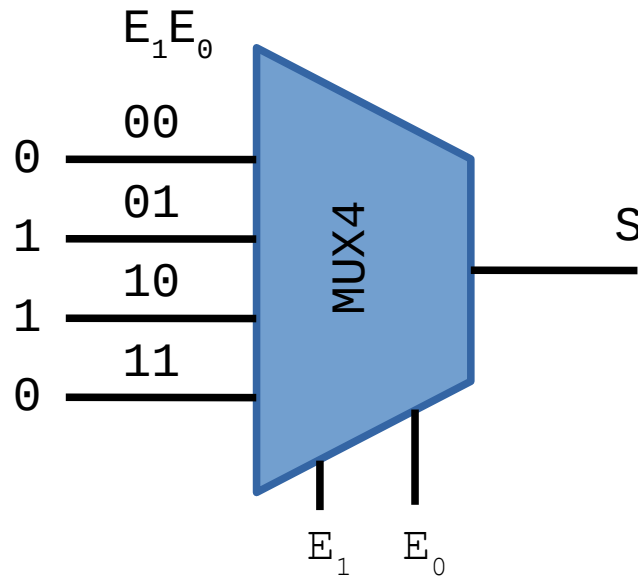


E1	E0	S
0	0	A
0	1	B
1	0	C
1	1	D



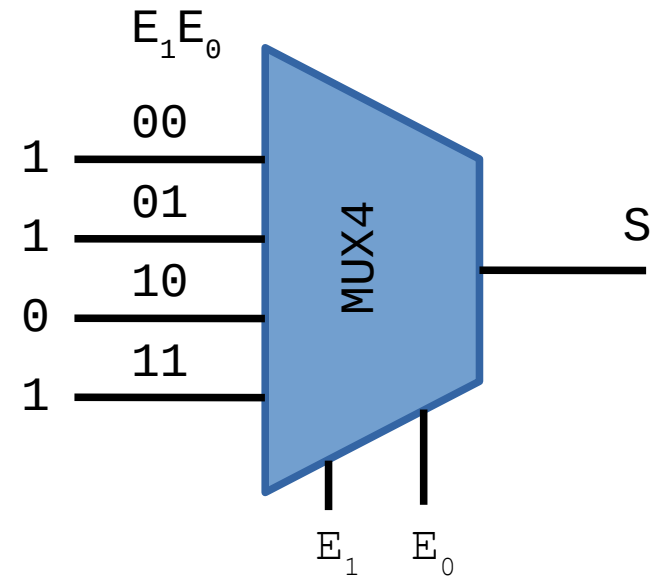
# Uso de MUX4 para implementar funciones lógicas de dos entradas

## OR EXCLUSIVA



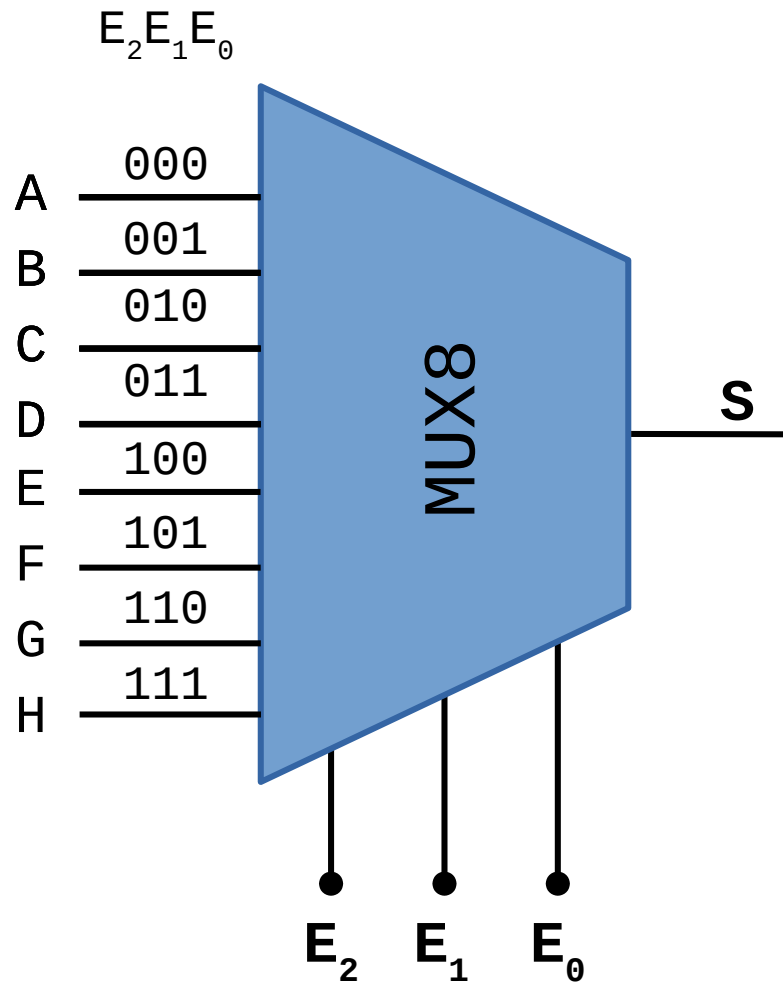
$E_1$	$E_0$	$S$
0	0	0
0	1	1
1	0	1
1	1	0

## FUNCIÓN ARBITRARIA



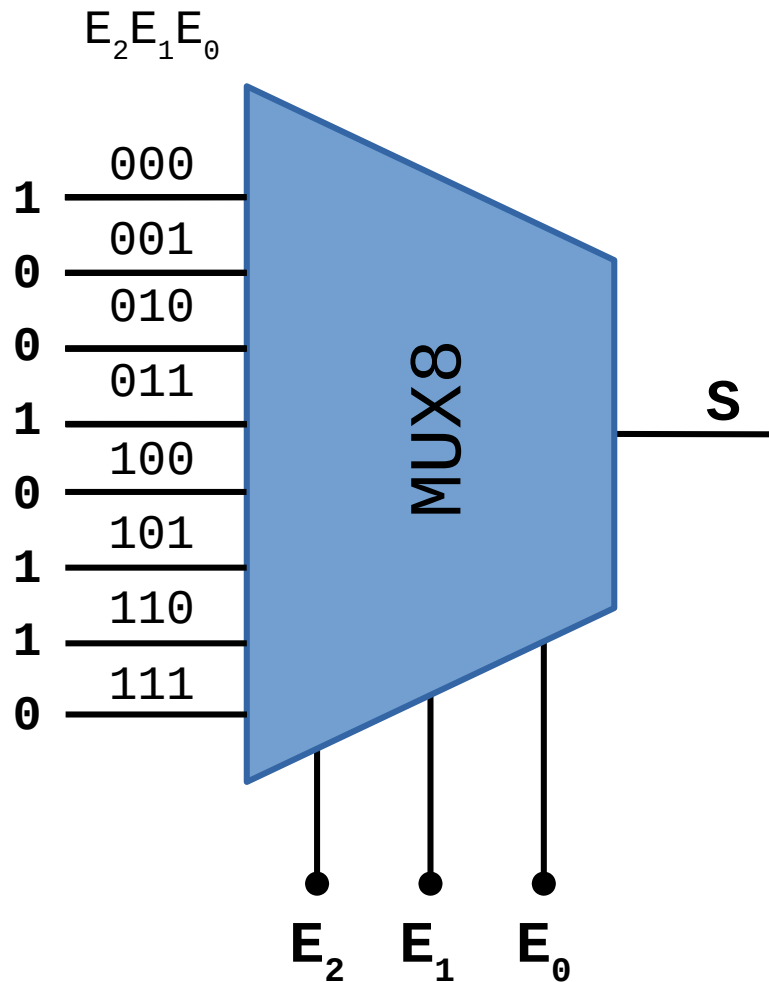
$E_1$	$E_0$	$S$
0	0	1
0	1	1
1	0	0
1	1	1

# Multiplexor de ocho entradas (MUX8)



$E_2$	$E_1$	$E_0$	S
0	0	0	A
0	0	1	B
0	1	0	C
0	1	1	D
1	0	0	E
1	0	1	F
1	1	0	G
1	1	1	H

# Uso de MUX8 para implementar funciones lógicas de tres entradas



$E_2$	$E_1$	$E_0$	$S$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

# Componentes

## Llaves / switches / Interruptores

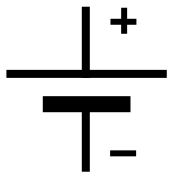


Normalmente Abierto (NA)

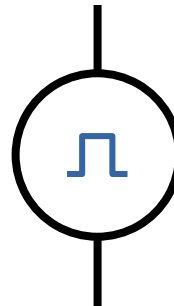


Normalmente Cerrado (NC)

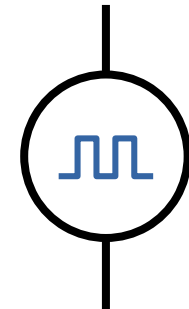
## Fuentes



Batería  
Fuente de  
Tensión Continua  
VCC



Fuente Digital  
Fuente de Pulsos

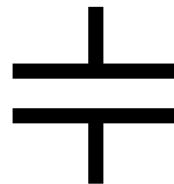


Reloj  
Clock  
Ck

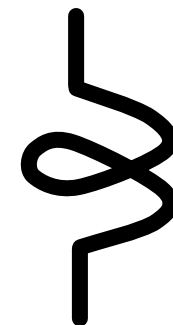
## Componentes Pasivos



Resistor (R)



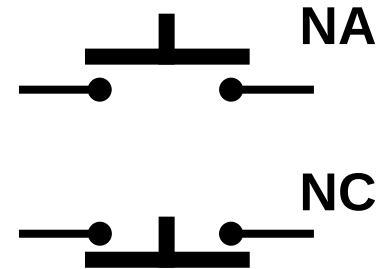
Capacitor (C)



Inductor (L)

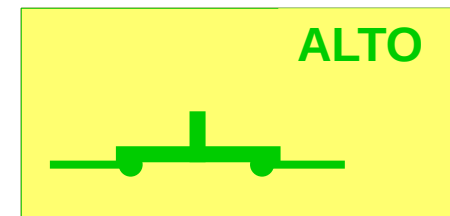
# Llaves y Variables Lógicas

- Las llaves o interruptores pueden ser de dos tipos:
  - Normalmente Abiertas (**NA**): Cuando son accionadas, conectan el circuito, por ejemplo pulsador de timbre.
  - Normalmente Cerradas (**NC**): Cuando son accionadas, desconectan el circuito.



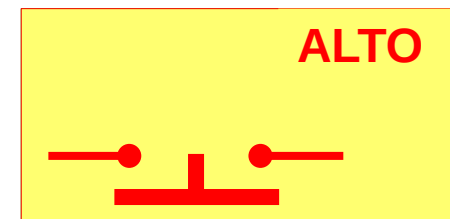
- Se puede interpretar que una llave **NA** puede accionarse por una variable lógica

- Cuando está en ALTO, el circuito está establecido.
- Cuando está en BAJO, el circuito está interrumpido.

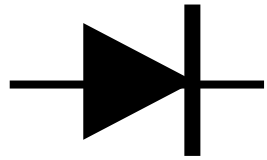


- Se puede interpretar que una llave **NC** puede accionarse por una variable lógica

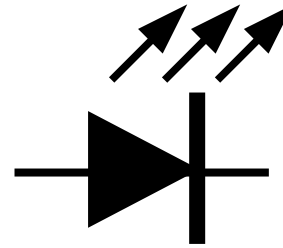
- Cuando está en BAJO, el circuito está establecido.
- Cuando está en ALTO, el circuito se interrumpe.



# Componentes Activos

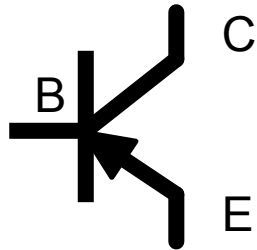


DIODO

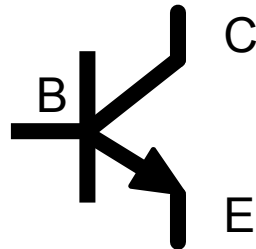


DIODO EMISOR DE LUZ (LED)

## TRANSISTORES BIPOLARES

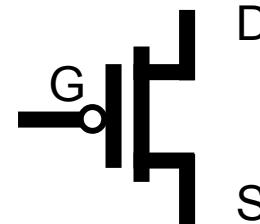


Transistor PNP

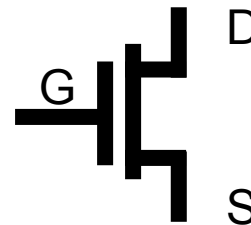


Transistor NPN

## TRANSISTORES DE EFECTO DE CAMPO METAL-ÓXIDO SEMICONDUCTOR (MOSFET)



Transistor P-MOS

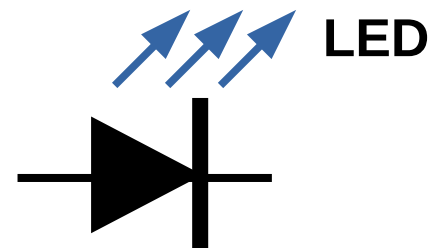
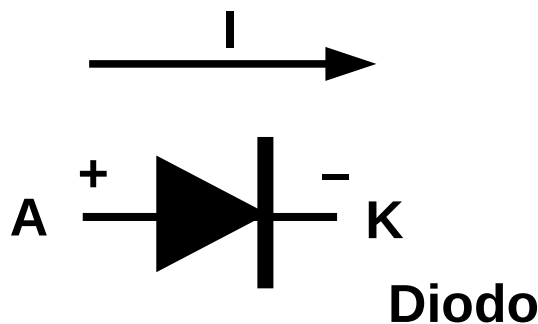


Transistor N-MOS



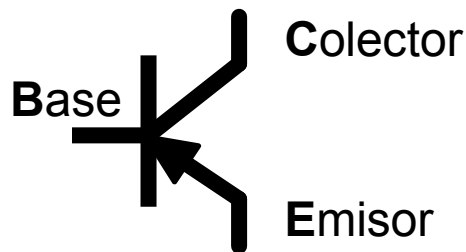
# Componentes Activos: Diodo

- Los terminales del Diodo reciben nombre: Anodo (A) y cátodo (K)
- El diodo permite que la corriente circule en un solo sentido, de ánodo a cátodo por dentro del dispositivo.
- Para que circule la corriente el Anodo debe ser positivo con respecto al cátodo.
- La corriente que circula por el diodo crece exponencialmente con la tensión aplicada, así que en muchos casos es necesario poner un resistor en serie para limitar la corriente y que el dispositivo no se dañe.
- El LED es un tipo especial de diodo que emite luz cuando circula corriente a través de él.
- Actualmente existen LEDs que emiten luz de distintos colores, desde el infrarrojo hasta el Ultravioleta cercano.



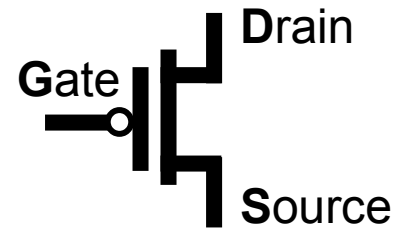
# Componentes Activos: BJT y MOSFET

- Tanto los Transistores de Juntura Bipolares de Juntura (BJT) como los MOSFET, pueden utilizarse como llaves.
- Cuando la tecnología de fabricación utiliza ambos tipos de MOSFET (PMOS y NMOS) en el mismo circuito, se la denomina tecnología CMOS (Complementary MOS)
- Tienen un elemento de control (Base EN BJT o Gate en CMOS), que permite accionarla con un nivel lógico ALTO o BAJO, según como estén conectadas.



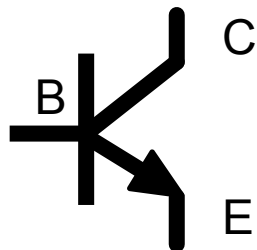
Transistor  
(BJT) **PNP**

Se acciona cuando la Base está en un nivel más BAJO que el Emisor



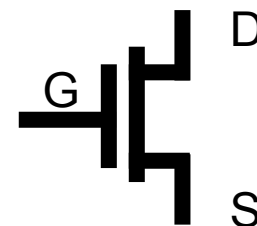
Transistor  
**P-MOS**

Se acciona cuando Gate está en un nivel más BAJO que Source



Transistor  
(BJT) **NPN**

Se acciona cuando la Base está en un nivel más ALTO que el Emisor

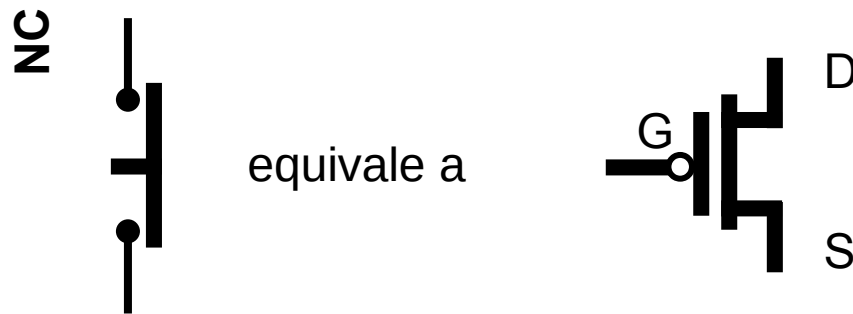


Transistor  
**N-MOS**

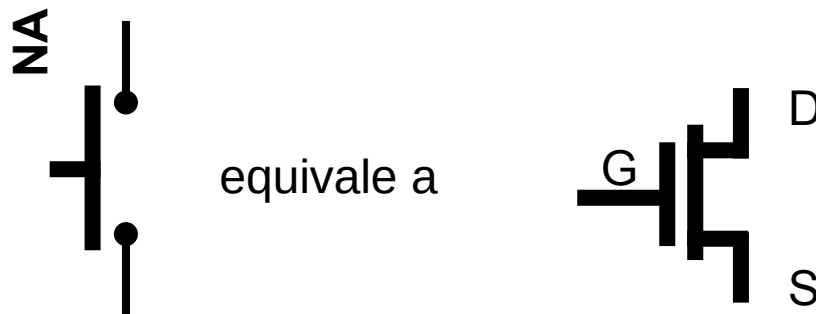
Se acciona cuando Gate está en un nivel más ALTO que Source

# Equivalencia Llave -> MOSFET

- En Sistemas Digitales los BJT y MOSFET se usan como Llaves o interruptores.
- La implementación de circuitos digitales con MOSFET tiene ventajas tecnológicas respecto a la implementación con transistores bipolares y es la que más se usa actualmente.



Cuando se aplica un nivel BAJO entre G y S, un PMOS permite la circulación de corriente entre D y S



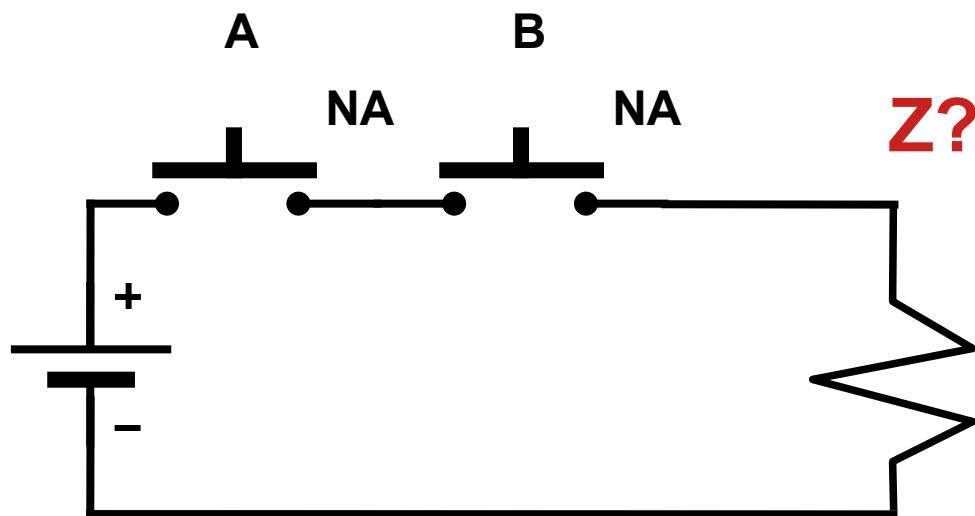
Cuando se aplica un nivel ALTO entre G y S, un NMOS permite la circulación de corriente entre D y S

# Equivalente de Compuerta AND

Si queremos que el punto **Z** tenga un nivel ALTO, necesitamos cerrar el circuito accionando las llaves **A** y **B** a la vez

Como vimos antes, en una llave NA, aplicar un valor alto a la entrada de control (pulsador) cierra el circuito

Podemos asimilar **A** a un nivel ALTO y  $\bar{A}$  a un nivel BAJO.



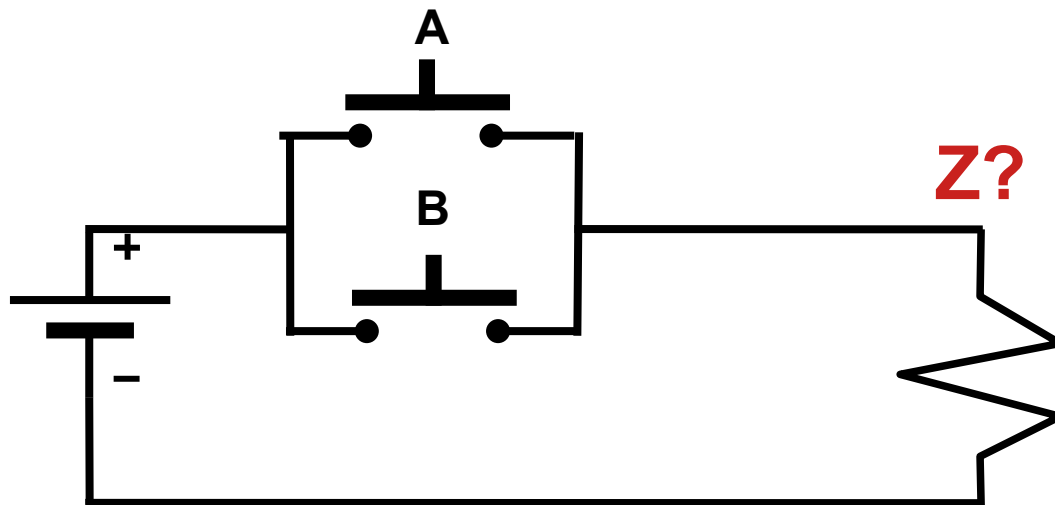
A	B	Z
BAJO (abierto)	BAJO (abierto)	BAJO
BAJO (abierto)	ALTO (cerrado)	BAJO
ALTO (cerrado)	BAJO (abierto)	BAJO
ALTO (cerrado)	ALTO (cerrado)	ALTO

# Equivalente de Compuerta OR

Si queremos que el punto **Z** tenga un nivel ALTO, necesitamos cerrar el circuito accionando cualquiera de las llaves **A**, **B** o ambas a la vez

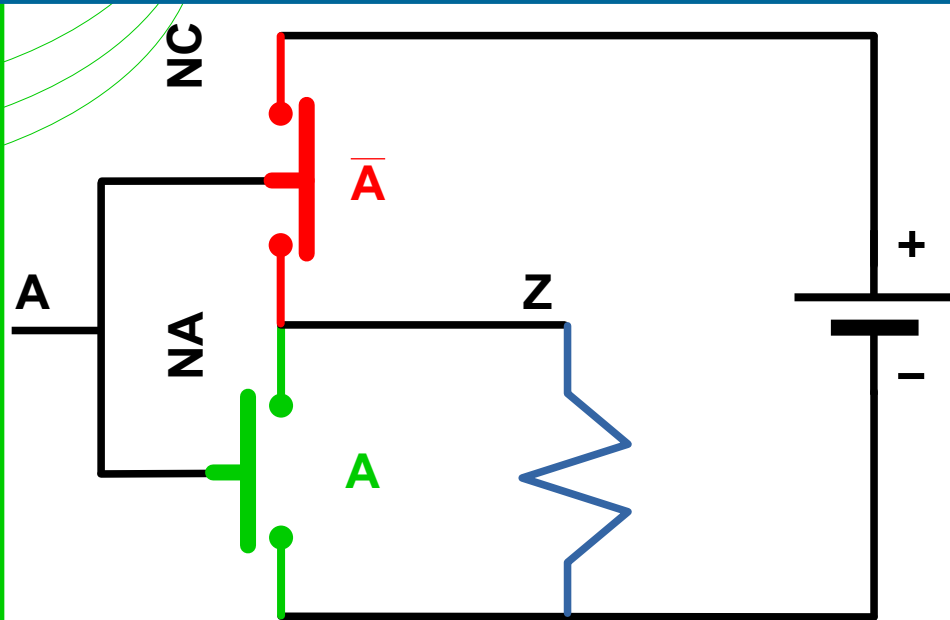
Como vimos antes, en una llave **NA**, aplicar un valor alto a la entrada de control (pulsador) cierra el circuito

Podemos asimilar **A** a un nivel ALTO y  $\bar{A}$  a un nivel BAJO.



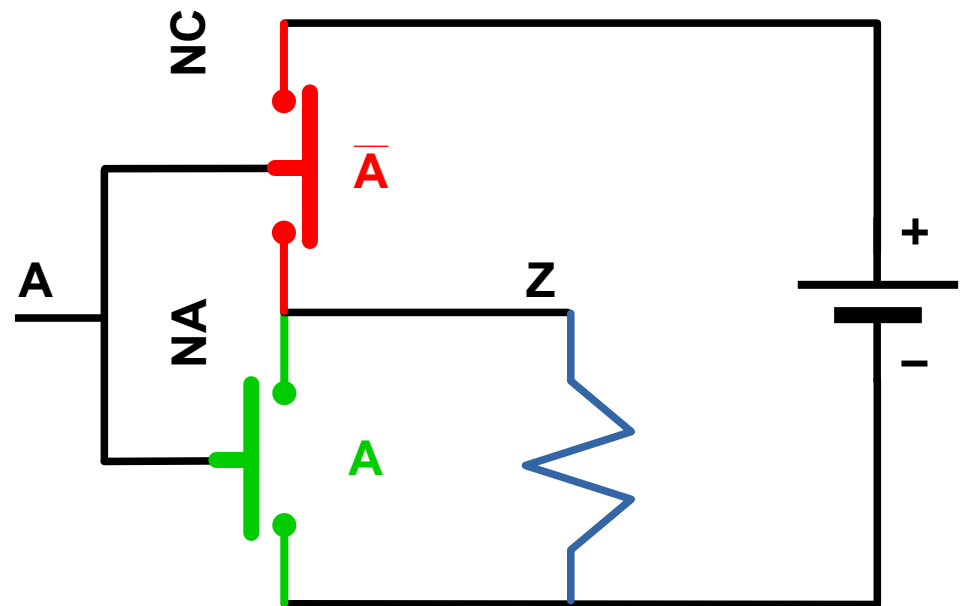
A	B	Z
BAJO (abierto)	BAJO (abierto)	BAJO
BAJO (abierto)	ALTO (cerrado)	ALTO
ALTO (cerrado)	BAJO (abierto)	ALTO
ALTO (cerrado)	ALTO (cerrado)	ALTO

# Equivalente de Inversor

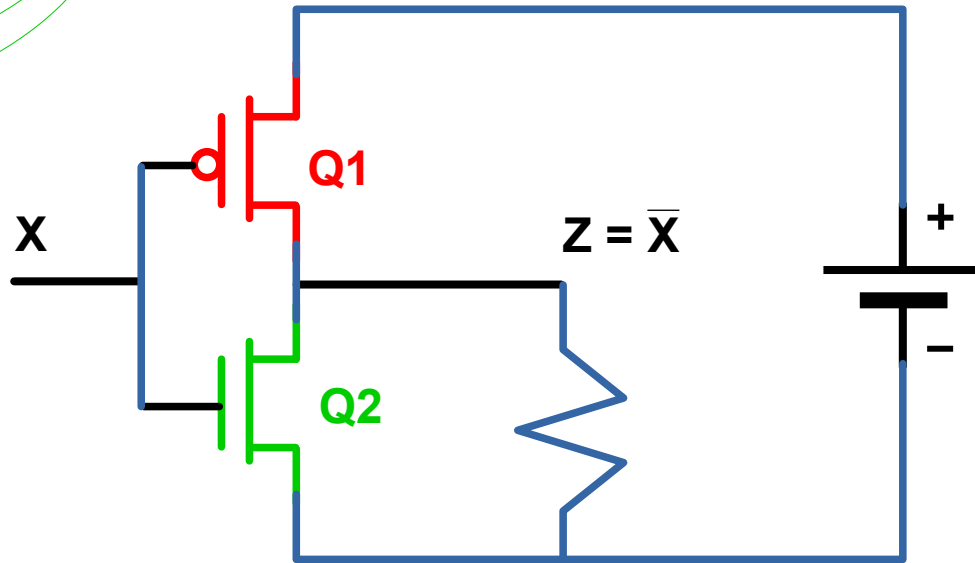


Si aplicamos un nivel ALTO en A, se accionará la llave  $\overline{A}$ , Mientras que la llave  $A$  permanecerá abierta. Si seguimos el circuito, vemos que Z tendrá un nivel BAJO

Si aplicamos un nivel BAJO en A, se accionará la llave  $A$ , Mientras que la llave  $\overline{A}$  permanecerá abierta. Si seguimos el circuito, vemos que Z tendrá un nivel ALTO



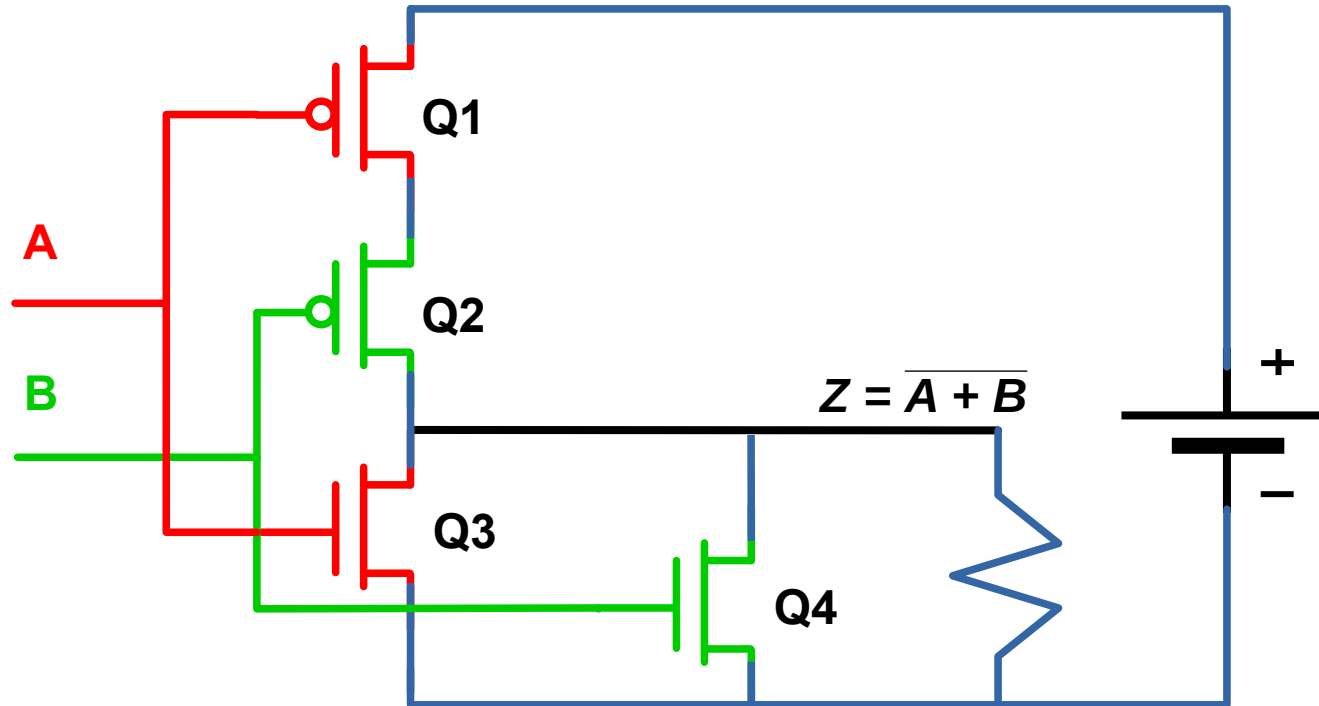
# Implementación de inversor con CMOS



X	Q1	Q2	Z
L	ON	OFF	H
H	OFF	ON	L

- Q1 y Q2 Son MOSFET de tipo P y de tipo N respectivamente
- Actúan como las llaves que vimos antes

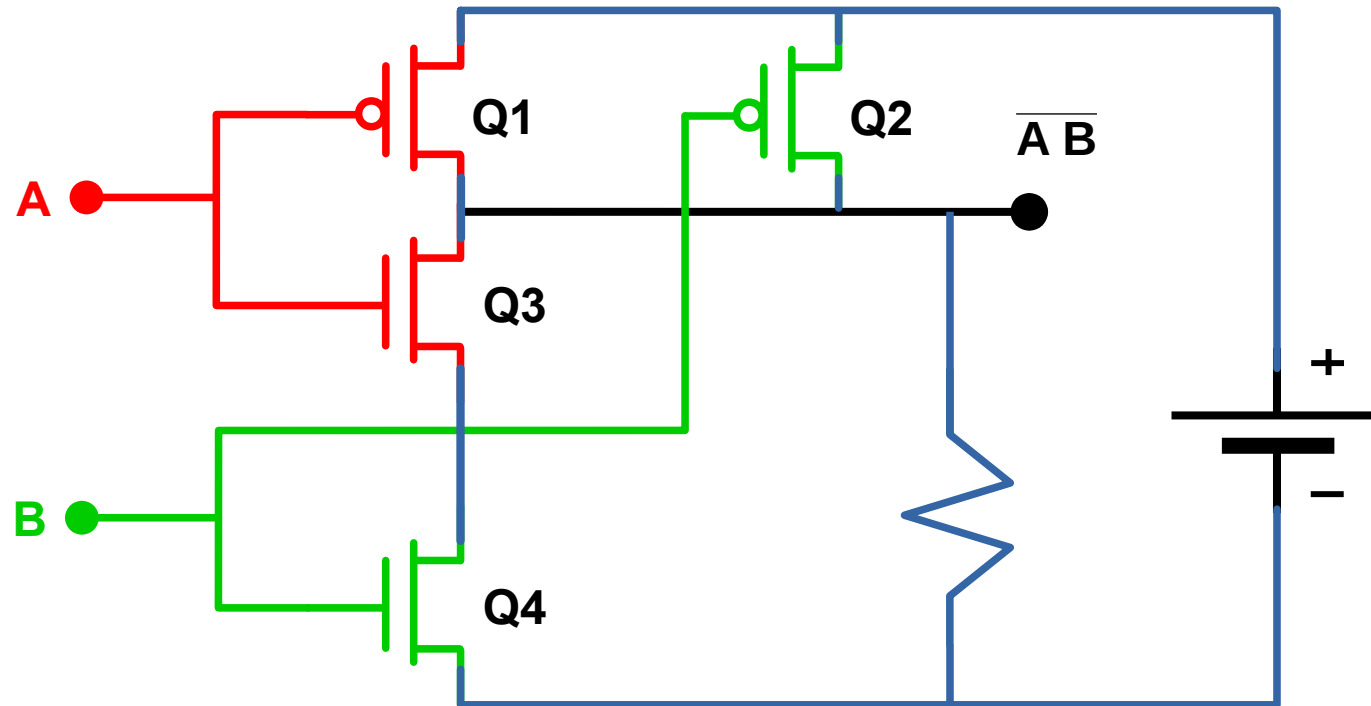
# Compuerta NOR



A	B	Q1	Q2	Q3	Q4	Z
L	L	ON	ON	OFF	OFF	H
L	H	ON	OFF	OFF	ON	L
H	L	OFF	ON	ON	OFF	L
H	H	OFF	OFF	ON	ON	L



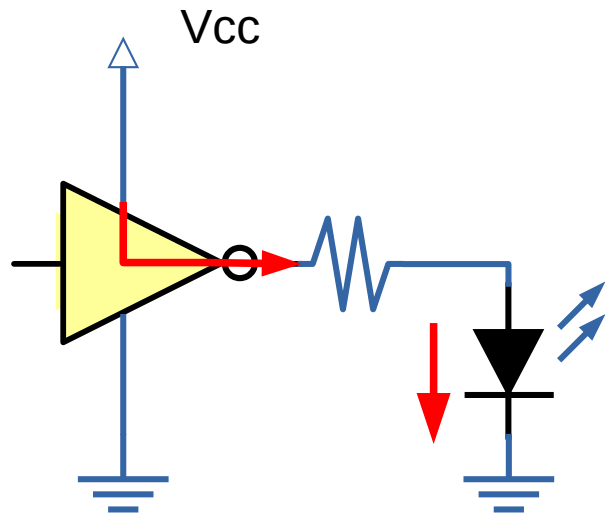
# Compuerta NAND



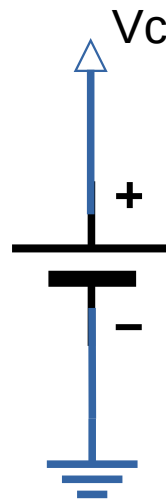
A	B	Q1	Q2	Q3	Q4	Z
L	L	ON	ON	OFF	OFF	H
L	H	ON	OFF	OFF	ON	H
H	L	OFF	ON	ON	OFF	H
H	H	OFF	OFF	ON	ON	L

# Conectar un LED

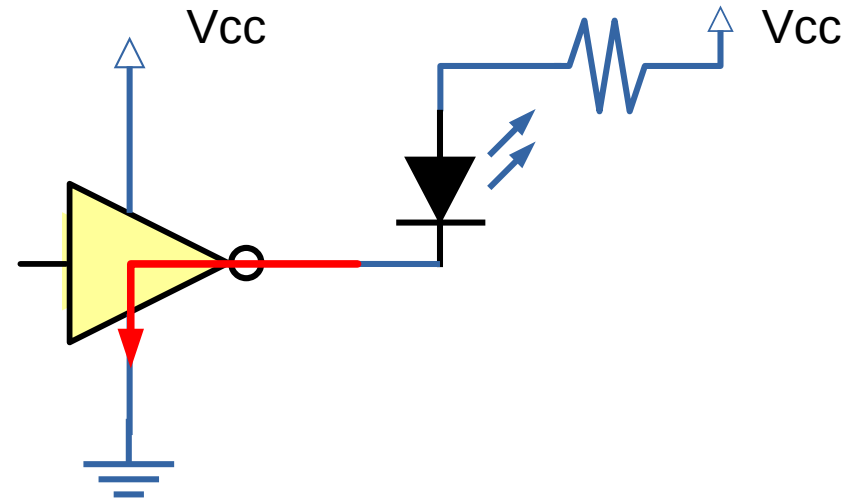
Enciende cuando la salida se pone en ALTO



Fuente de alimentación



Enciende cuando la salida se pone en BAJO



En rojo se marca el camino interno que sigue la corriente que enciende el LED cuando el circuito está activo

Note que en el primer caso, la corriente SALE de la Compuerta y en el segundo caso ENTRA a la compuerta

En ambos casos se marcaron las entradas de alimentación ( $V_{CC}$  y Tierra).

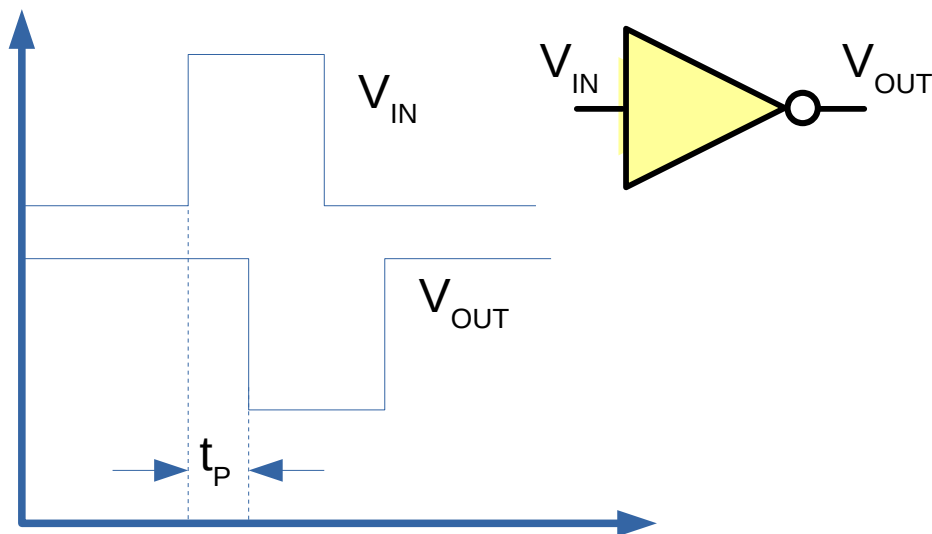
Esto es una convención para no trazar las líneas que unen la batería con los distintos elementos del circuito.

# Compuertas Reales

## TIEMPO DE PROPAGACIÓN

Desde el instante en que cambia el valor de la entrada de un circuito lógico hasta que la salida adopta el valor correspondiente y se mantiene estable, pasa un tiempo que llamamos tiempo de propagación.

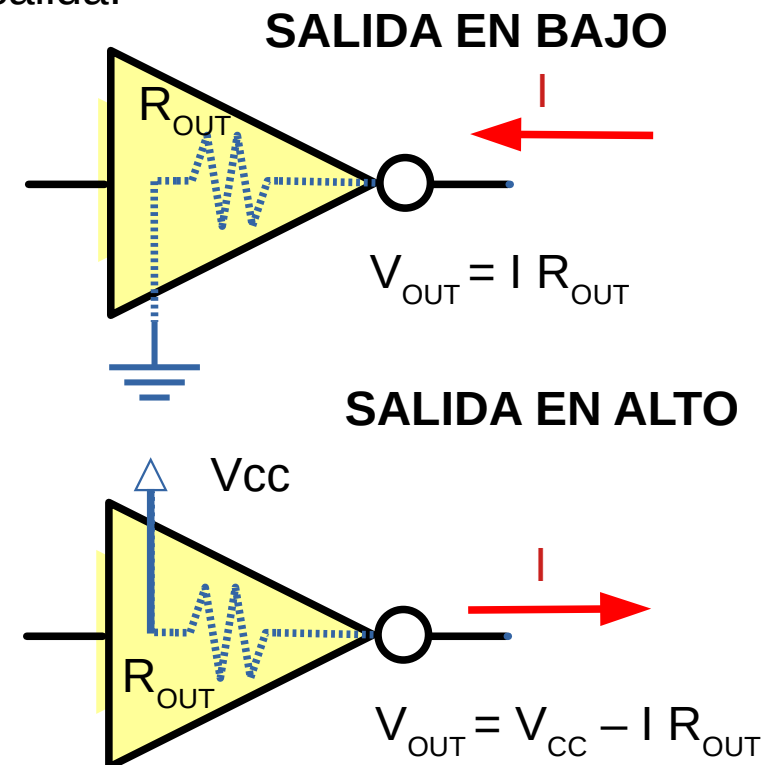
En la figura un ejemplo para un inversor.



## RESISTENCIA DE SALIDA

Los componentes no son ideales y por eso tienen un valor de resistencia en el camino de salida.

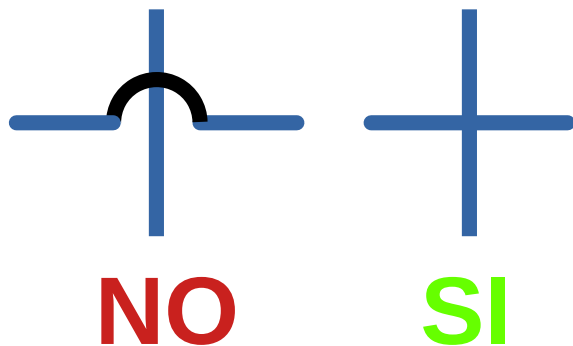
Esto afecta el valor de tensión que se obtiene cuando aumenta la corriente de salida.



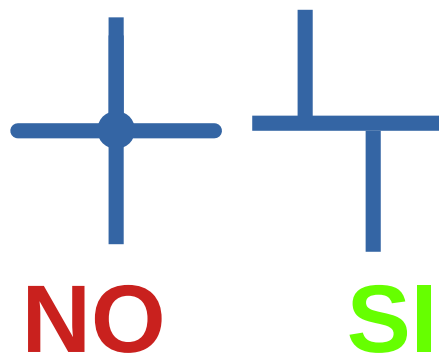
# Dibujar circuitos

- Hay muchas maneras de representar un circuito, la más habitual es realizar un “Diagrama Esquemático”
- En los diagramas suele darse la necesidad de unir líneas que conectan dispositivos, pero también a veces es necesario efectuar cruces de líneas que no están conectadas entre sí
- La convención que utilizamos es:
  - Dos líneas que se cruzan no están conectadas.
  - Las conexiones se indican haciendo terminar una línea sobre otra formando una “T”
  - Si es necesario cambiar la dirección de una línea, hacerlo siguiendo ángulos rectos, o en el peor de los casos a  $30^\circ$ ,  $45^\circ$  o  $60^\circ$

## CRUCES



## UNIONES



## CAMBIOS DE DIRECCIÓN

