Unidad 7: Compuertas Lógicas

Profesor: Ing. Israel Chaves Arbaiza

Curso: Electrónica Básica para Ing. Mecánica

UNIVERSIDAD DE COSTA RICA

EIM Escuela de Ingeniería Mecánica

Objetivos

- Entender el uso del álgebra booleana para la toma de decisiones
- Diseñar circuitos digitales con compuertas lógicas
- Implementar el control de un sistema digital

Algebra booleana

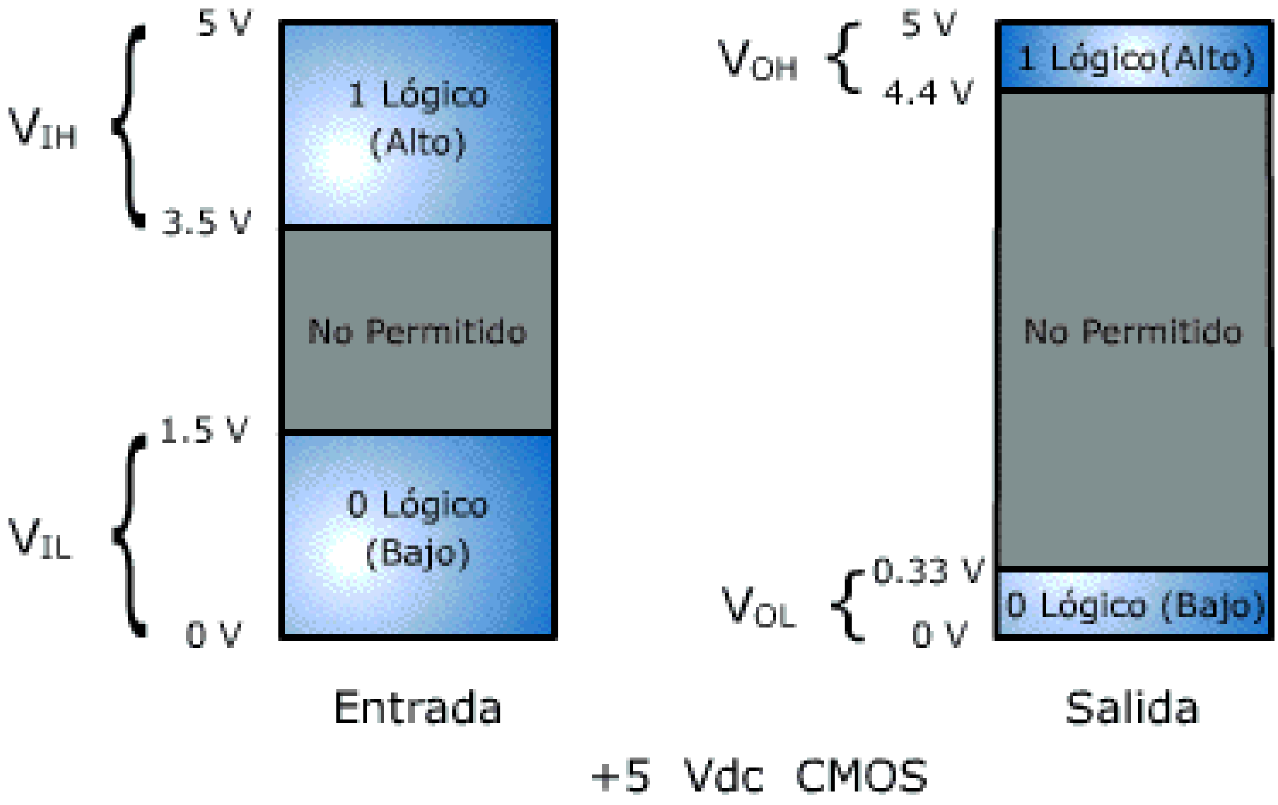
- Busca representar la **toma de decisiones**, en sistemas electrónicos y computacionales
- Desarrollada en 1854 por George Boole, con su libro *"Una investigación sobre las leyes del pensamiento"*
- Los circuitos lógicos utilizan intervalos de voltaje definidos, para representar los estados binarios: **True** o **False**

Algebra booleana

- Los métodos descritos por Boole, se conocen como **lógica booleana**, y el sistema de símbolos y operadores se llama **álgebra booleana**
- El propósito del álgebra booleana, es describir la relación entre la salida del circuito lógico (**la decisión**), y sus entradas (**condiciones**)

Operaciones lógicas

- Las variables booleanas se utilizan para representar el nivel de voltaje presente en un alambre o en las terminales de entrada/salida de un circuito
- Se representan con valores: 0 y 1; que corresponden a valores booleanos, y no a números reales. Esto se conoce como **Nivel lógico**



Operaciones lógicas

- Se utilizan símbolos de letras para representar las llamadas *variables lógicas*
- En el álgebra booleana no hay fracciones, raíces, valores negativos, logaritmos ni imaginarios, **sólo valores de 0 y 1**
- Únicamente hay 3 operaciones básicas: **OR, AND y NOT**, llamdas **Operaciones lógicas**
- Los circuitos digitales, llamados **Compuertas lógicas**, se construyen a partir de diodos, transistores y resistencias.

0 lógico	1 lógico
Falso	Verdadero
Apagado	Encendido
Bajo	Alto
No	Sí
Interruptor abierto	Interruptor cerrado

Tabla de verdad

- Describe la salida (**decisión**) de un circuito lógico, para los diferentes casos de sus señales de entrada (**condiciones**)

A	**B**	**X**
0	0	1
0	1	0
1	0	1
1	1	0

- En el ejemplo, la tabla muestra la salida X, para las diferentes **combinaciones** de las entradas A y B

Tabla de verdad

- Observe que hay 8 combinaciones para un circuito con 3 entradas (A,B,C).
- La cantidad de combinaciones será igual a 2^n con n, la cantidad de entradas

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

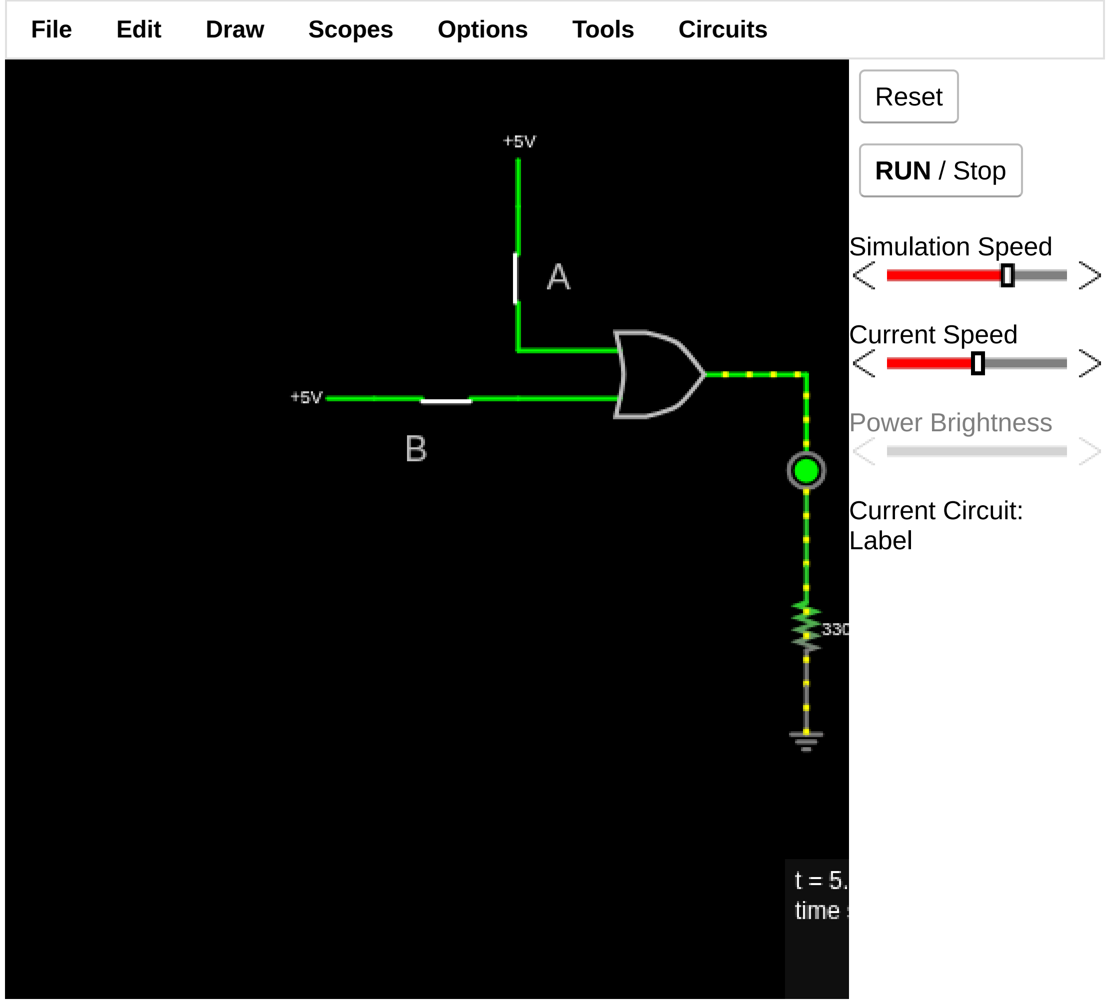
Operación OR

- La operación OR (o), representa una situacion donde es suficiente que se cumpla una u otra condición (pueden ser más de 2 condiciones), para activar una salida.
- El horno de la cocina es un buen ejemplo, la luz en el horno, se enciende si el interruptor está encendido **O** si la puerta está abierta.

Operación OR

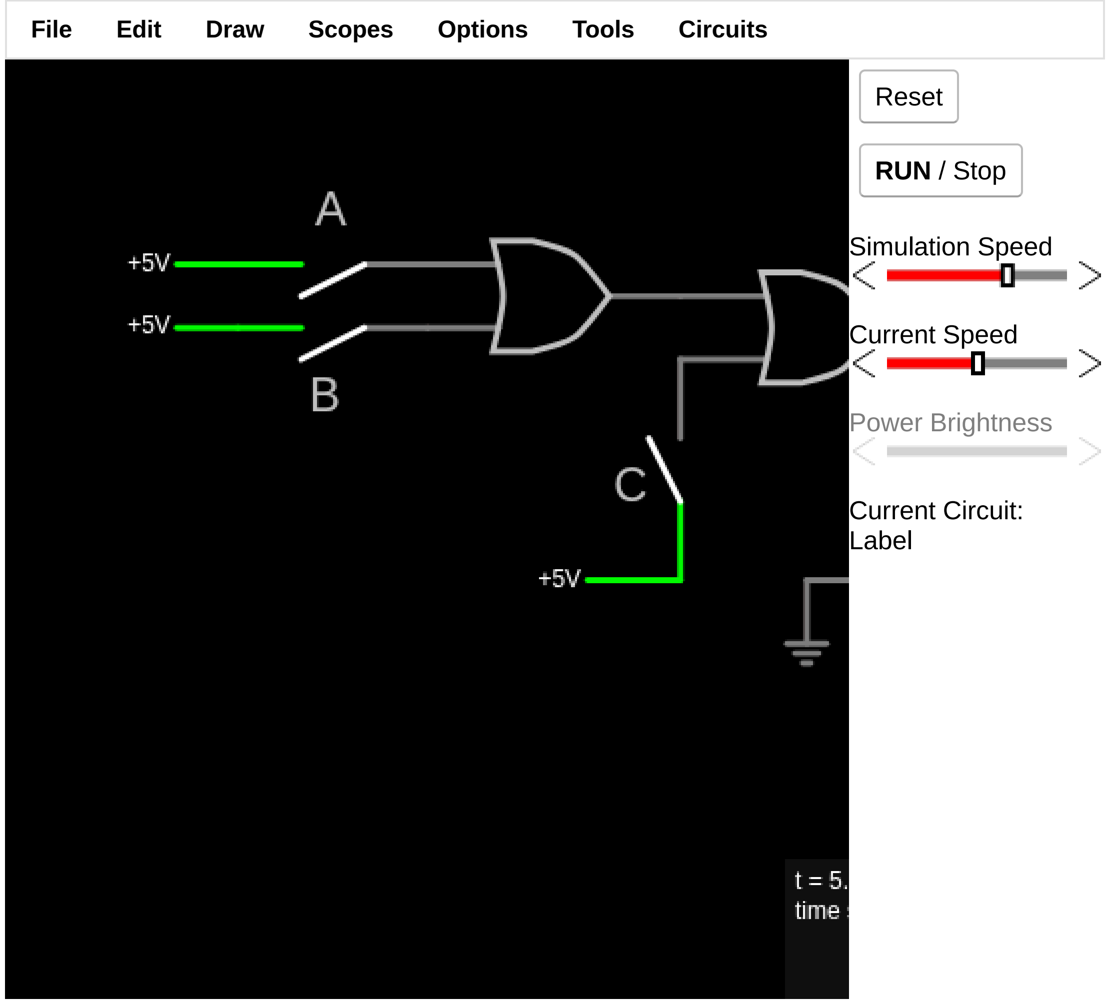
- A y B, representan el interruptor y la puerta, X, la salida.
- La operación OR se ejecuta como una **suma** de los valores lógicos
- **No es una suma ordinaria**

A	B	X = A + B
0	0	0
0	1	1
1	0	1
1	1	1



Operación OR: 3 entradas

A	B	C	X = A + B + C
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1



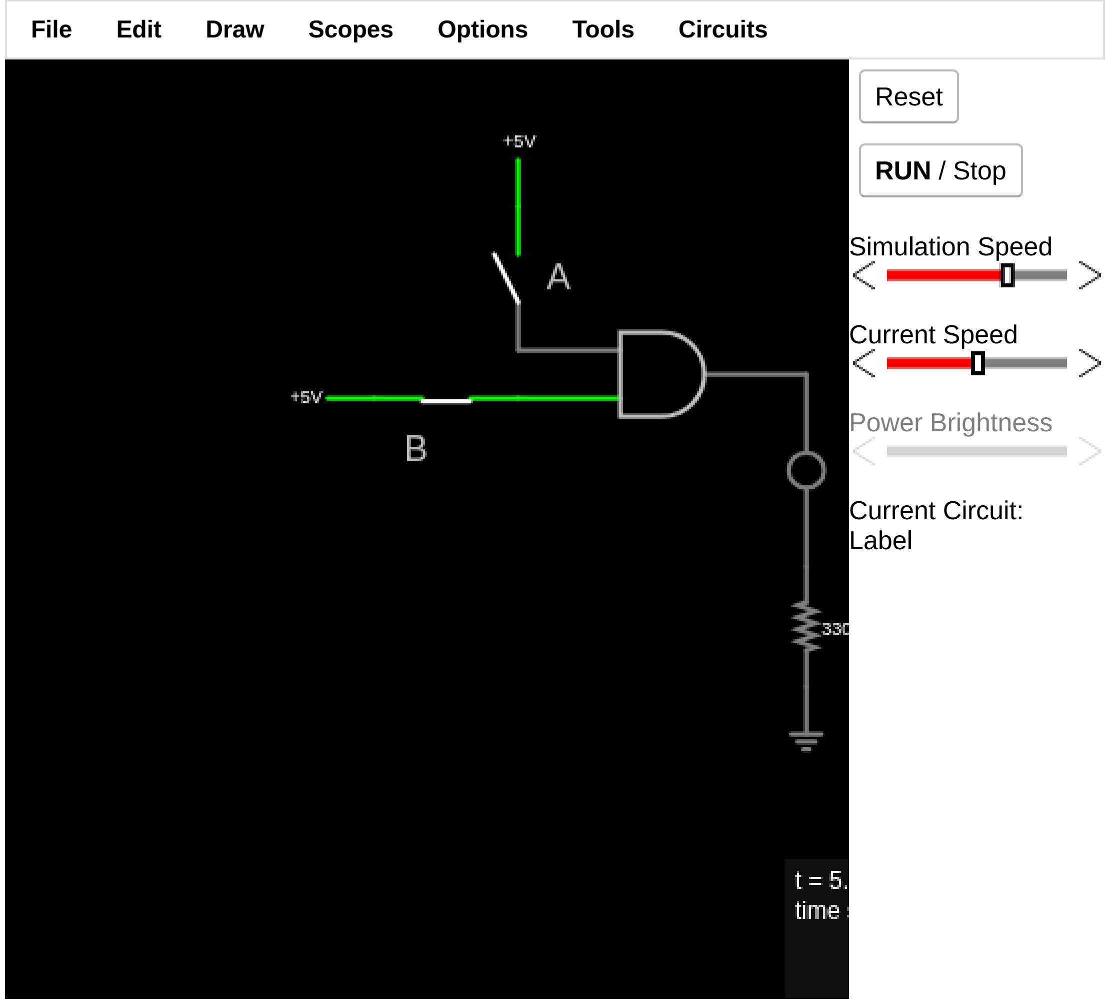
Operación AND

- La operación AND (y), representa una situacion donde se deben cumplir (valor lógico 1) todas las condiciones, para activar una salida.
- Una secadora de ropa, seca ropa (calienta y gira) **sólo si** el temporizador (A) está por encima de cero **Y** la puerta (B) está cerrada;

Operación AND

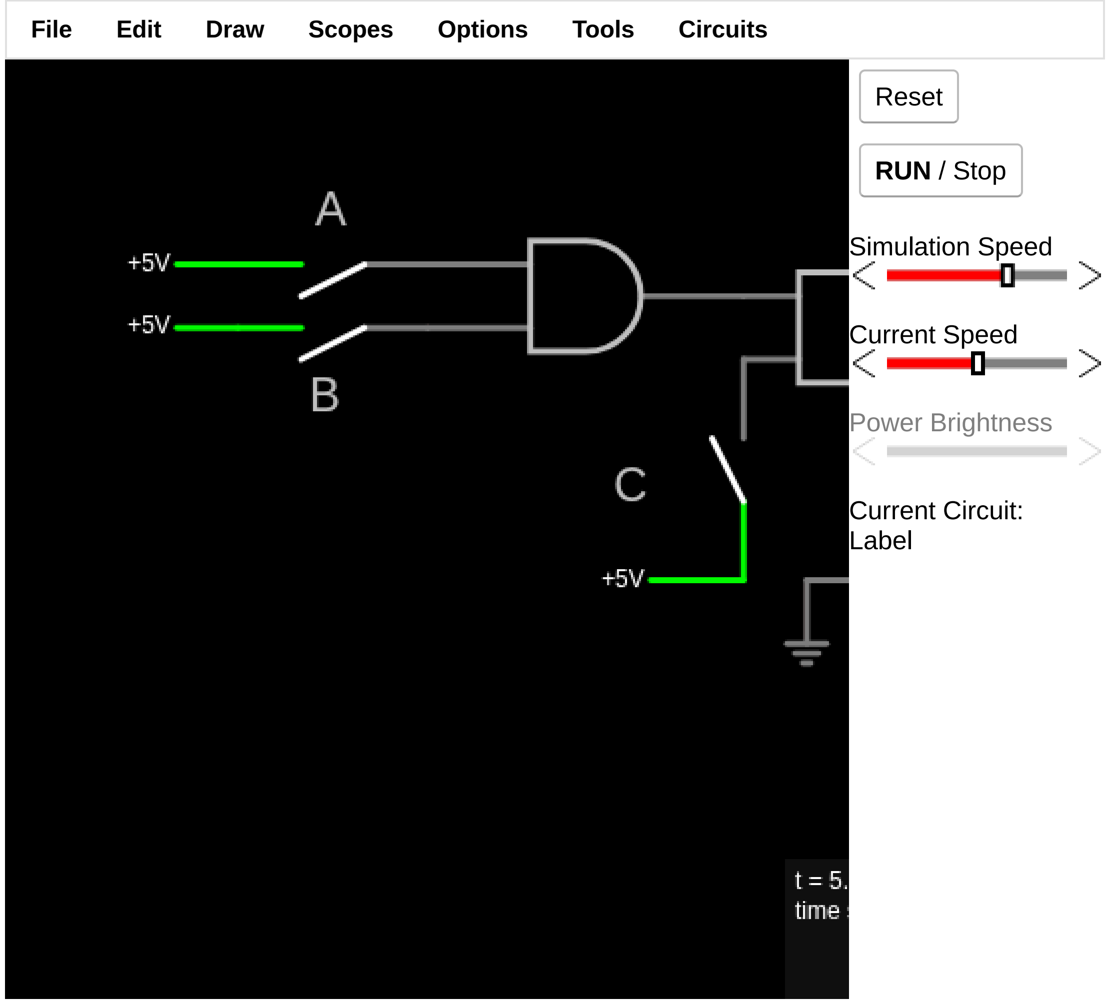
- Si la X representa la decisión (salida) de secar la ropa
- Las letras A y B, representan el temporizador y la puerta, respectivamente.
- La operación AND se ejecuta como una **multiplicación** de los valores lógicos

A	B	X = A · B
0	0	0
0	1	0
1	0	0
1	1	1



Operación AND: 3 entradas

A	B	C	X = A · B · C
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



Operación NOT

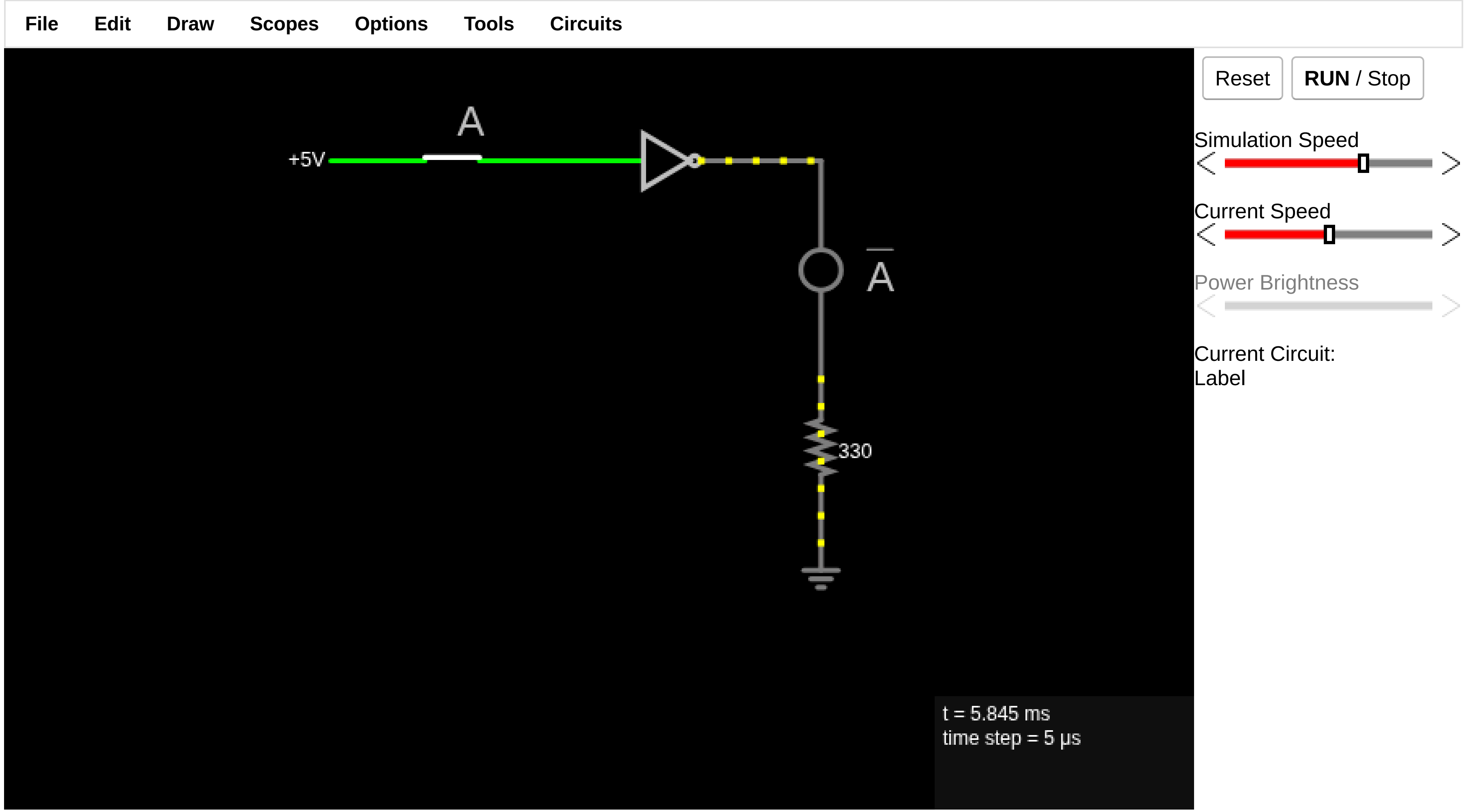
- La operación NOT (negación), representa una situacion donde la salida se convierte en el valor inverso (contrario) de la entrada
- Por ejemplo, si la variable A tiene un valor lógico de 1, la salida X sería un cero

$$X = \bar{A}$$

- La barra representa la operación de negación, también se puede representar con A'

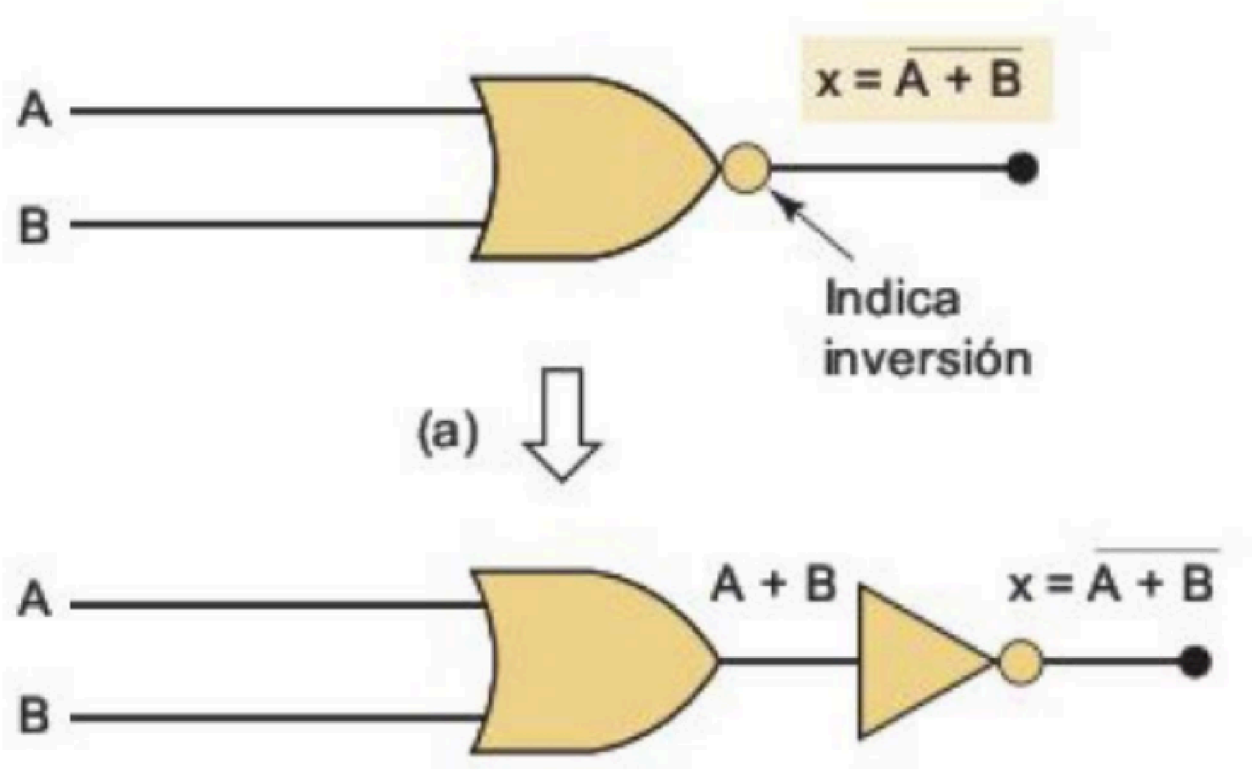
Operación NOT

A	$X = \bar{A}$
0	1
1	0



Operaciones derivadas: NOR

- La **compuerta NOR**, opera como una compuerta OR, seguida de un inversor (NOT)
- Se representa con el mismo símbolo que una OR, pero seguido de un pequeño círculo que representa la negación



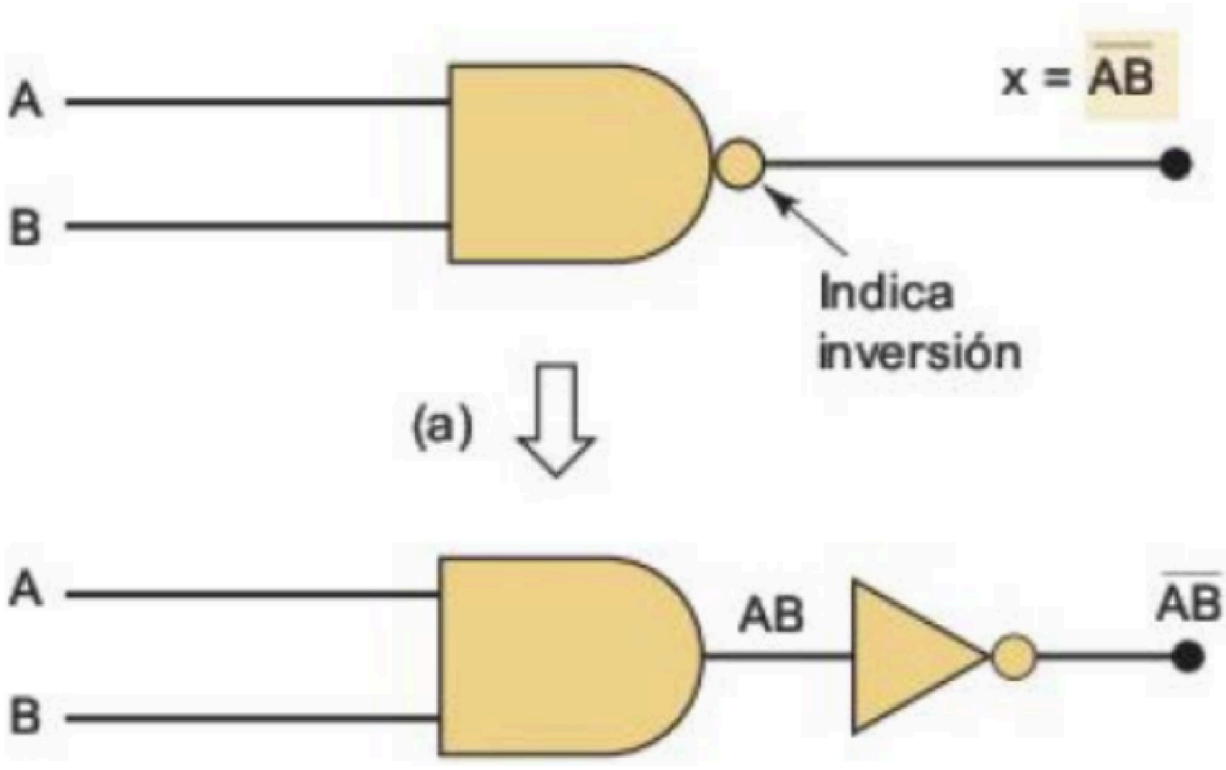
Operaciones derivadas: NOR

- Su tabla de verdad, muestra que la salida es el inverso exacto de la compuerta OR, para todas las posibles combinaciones de las condiciones

		OR	NOR
A	B	A + B	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Operaciones derivadas: NAND

- La **compuerta NAND**, opera como una compuerta AND, seguida de un inversor (NOT)
- Se representa con el mismo símbolo que una AND, pero seguido de un pequeño círculo que representa la negación



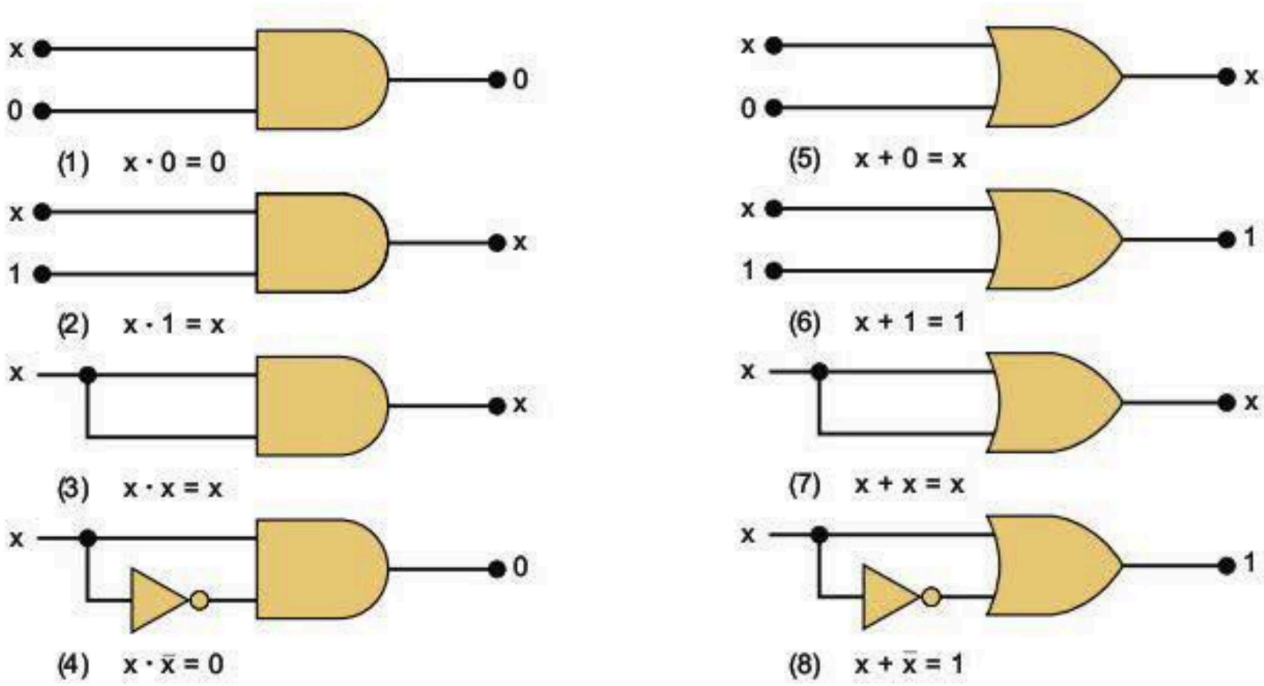
Operaciones derivadas: NAND

- Su tabla de verdad, muestra que la salida es el inverso exacto de la compuerta AND, para todas las posibles combinaciones de las condiciones

		AND	NAND
A	B	AB	$\overline{AB}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Teoremas booleanos

- También llamados *reglas booleanas*, simplifican expresiones y los circuitos lógicos
- En cada teorema, **x** puede ser 0 o 1

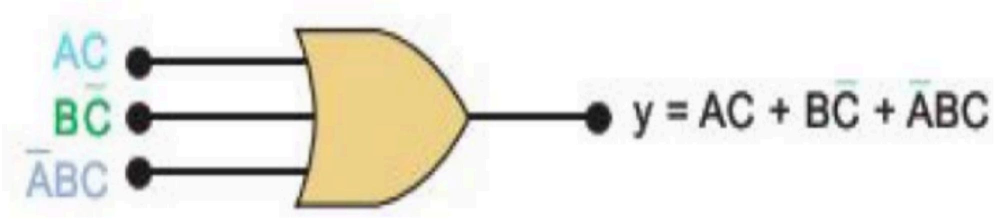


Implementación de circuitos lógicos a partir de expresiones

- Cuando la operación de un circuito, se define mediante una expresión booleana, se puede dibujar el diagrama del circuito lógico directamente a partir de la expresión
- El principio básico es dibujar el circuito de **derecha a izquierda**, es decir, comenzando por la decisión, hacia las condiciones
- Cada operación de **multiplicación**, se dibuja como una **compuerta AND**, y cada **suma** se dibuja como una **operación OR**

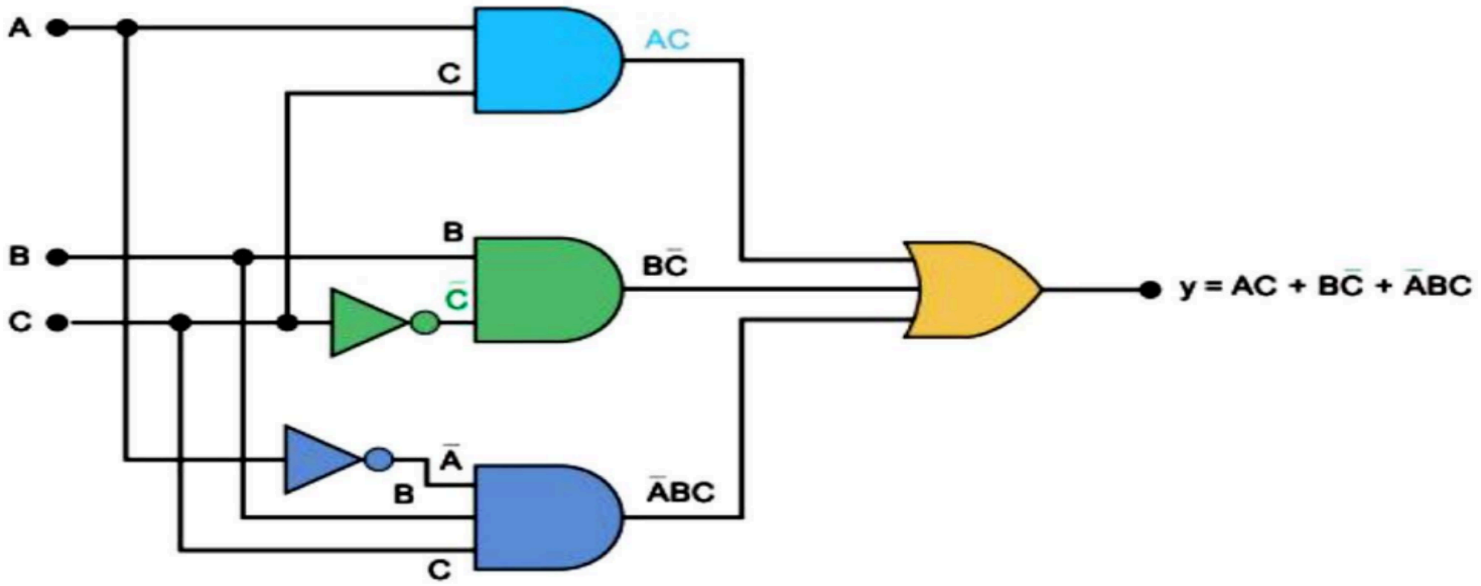
Implementación de circuitos lógicos a partir de expresiones

- Ejemplo: $y = AC + B\bar{C} + \bar{A}BC$
- Esta expresión se conforma de una suma de 3 términos (AC,$B\bar{C}$,$\bar{A}BC$) que a su vez son multiplicaciones de 2 o más condiciones
- Así, dibujando de derecha a izquierda, se inicia con una compuerta OR (suma)



Implementación de circuitos lógicos a partir de expresiones

- Expresión: $y = AC + B\bar{C} + \bar{A}BC$
- De esta manera, se obtiene:



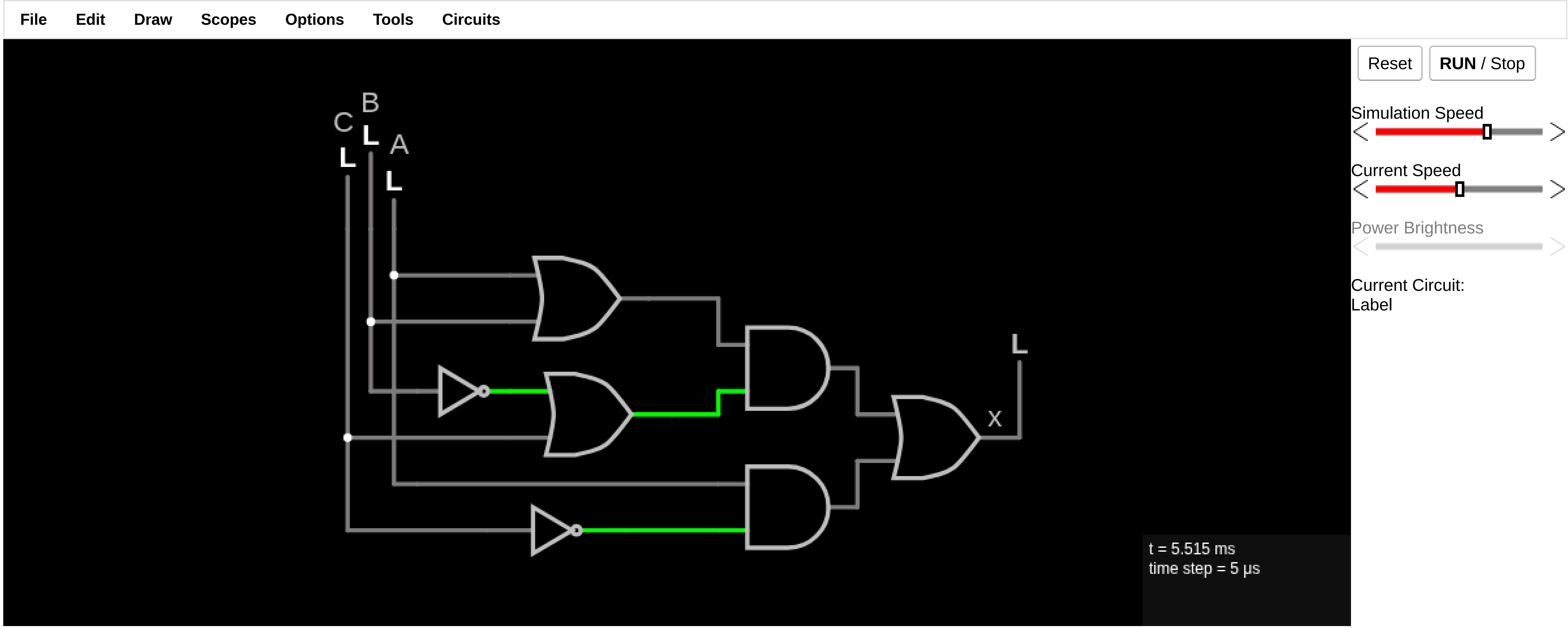
Implementación de circuitos lógicos a partir de expresiones

- Dibuje el circuito de la expresión:

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

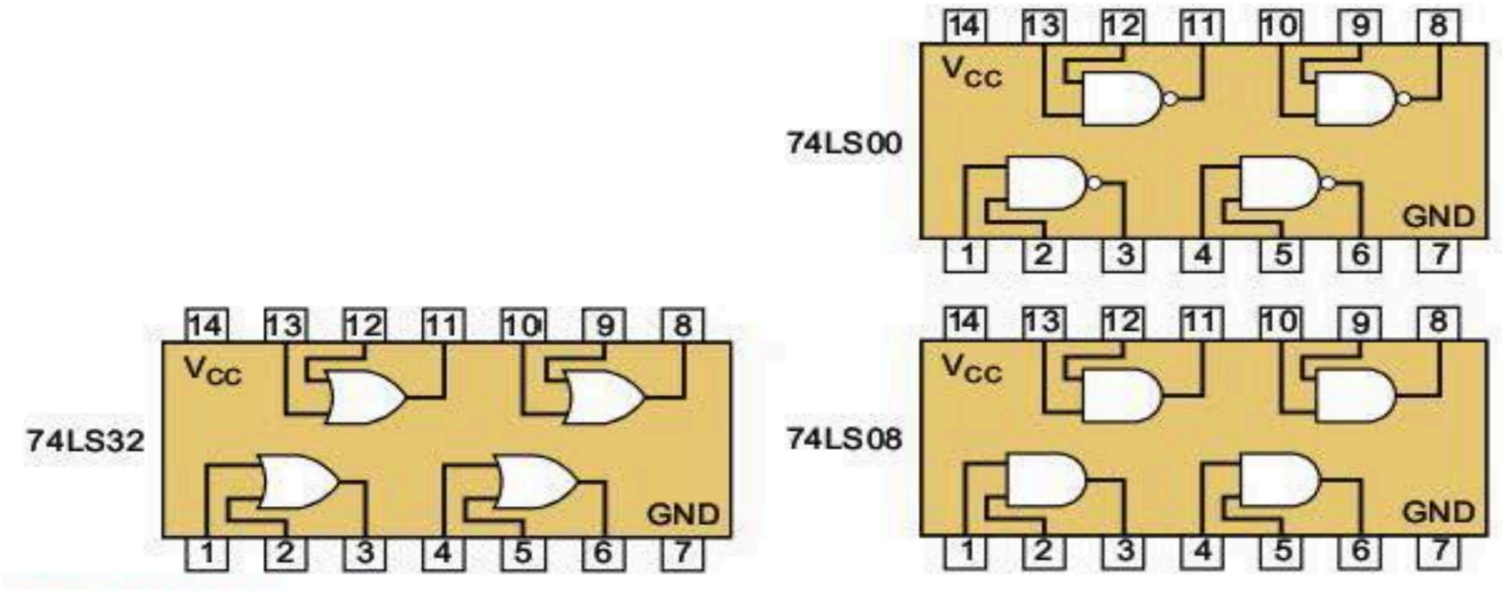
Implementación de circuitos lógicos a partir de expresiones

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



Circuitos integrados

- Los circuitos integrados, se presentan típicamente en chips de 14 patillas, conteniendo cuatro compuertas lógicas de 2 condiciones internamente
- Pero también se pueden presentar chips con compuertas de 3 o hasta 4 condiciones, y una salida



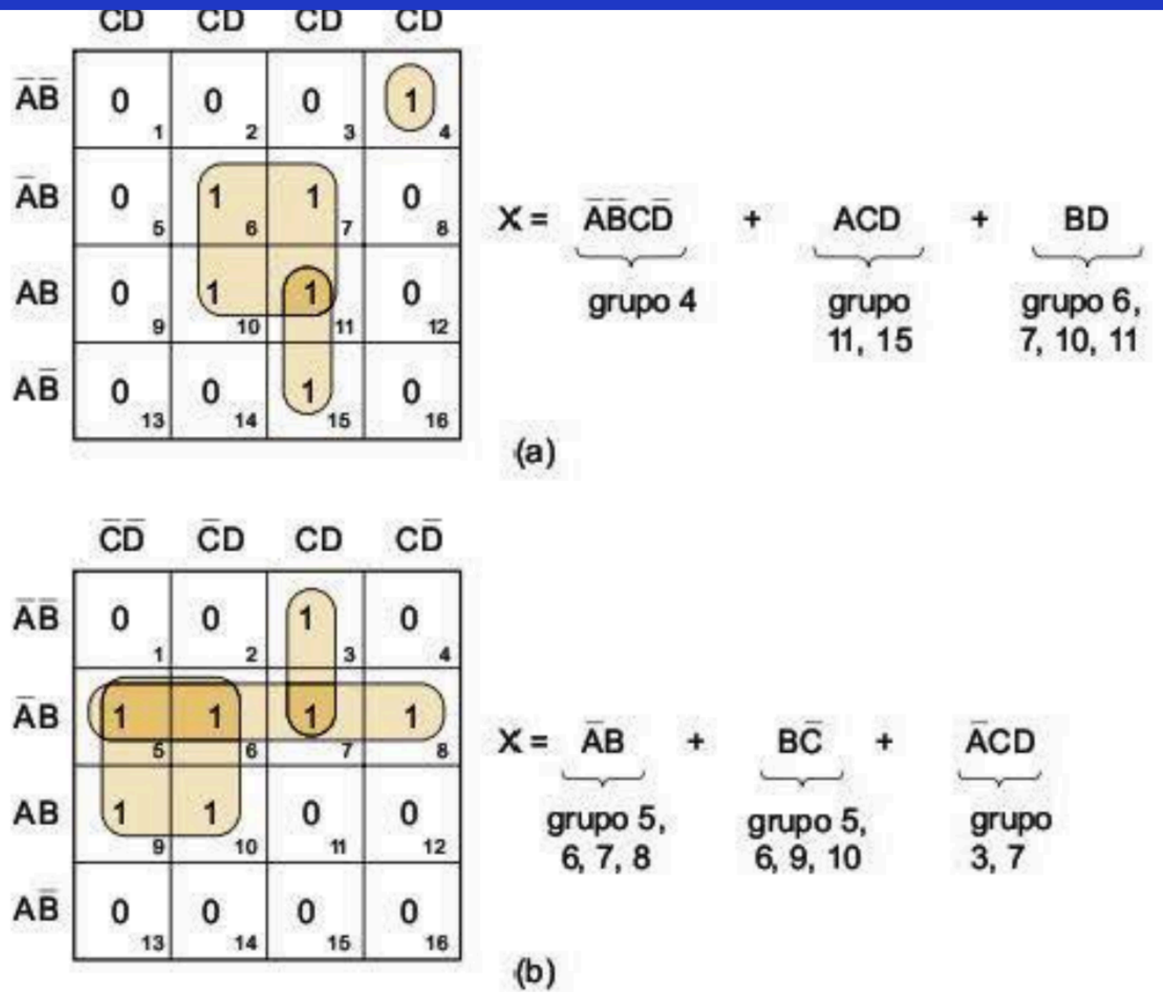
Mapas de Karnaugh

- Los mapas K, representan la relación entre las condiciones y la salida lógica deseada
- El mapa K, permite **agrupar los 1's** para obtener una **expresión booleana, a partir de la tabla de verdad**
- Los grupos en el mapa deben ser de 1, 2, 4, 8 o 16 (potencias de 2)

A	B	C	D	X	
0	0	0	0	0	
0	0	0	1	1	$\rightarrow \bar{A}\bar{B}\bar{C}D$
0	0	1	0	0	
0	0	1	1	0	
0	1	0	0	0	
0	1	0	1	1	$\rightarrow \bar{A}B\bar{C}D$
0	1	1	0	0	
0	1	1	1	0	
1	0	0	0	0	
1	0	0	1	0	
1	0	1	0	0	
1	0	1	1	0	
1	1	0	0	0	
1	1	0	1	1	$\rightarrow AB\bar{C}D$
1	1	1	0	0	
1	1	1	1	1	$\rightarrow ABCD$

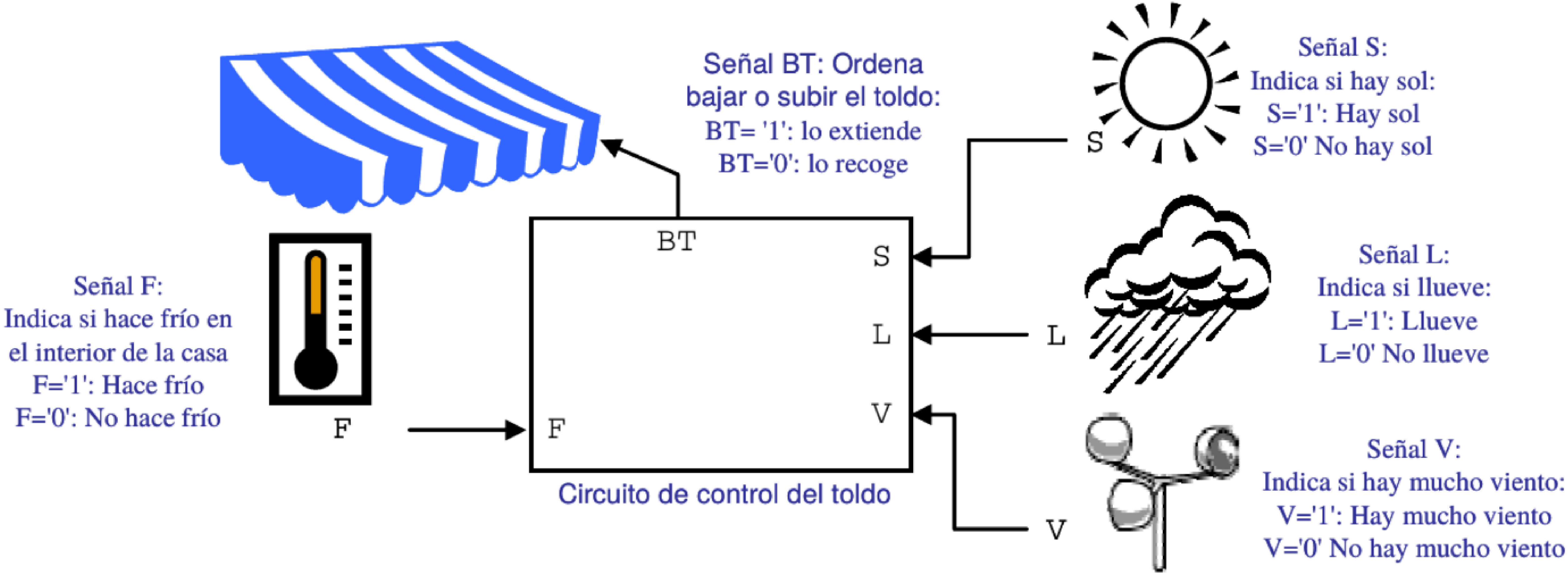
$$\left\{ \begin{aligned} X = {} & \bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D \\ & + AB\bar{C}D + ABCD \end{aligned} \right\}$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	1	0	0
$\bar{A}B$	0	1	0	0
AB	0	1	1	0
$A\bar{B}$	0	0	0	0



Solucionador de Karnaugh online

Ejemplo de sistema de control



Ejemplo de sistema de control

Utilizando las variables del esquema, el circuito que acciona el toldo, debe funcionar según las siguientes características:

- Independientemente del resto de señales de entrada, siempre que llueva se debe de extender el toldo para evitar que se moje la terraza. No se considerará posible que simultáneamente llueva y haga sol.
- Si hace viento se debe extender el toldo para evitar que el viento moleste. Sin embargo, hay una excepción: aún cuando haya viento, si el día está soleado y hace frío en la casa, se recogerá el toldo para que el sol caliente la casa.
- Por último, si no hace viento ni llueve, sólo se bajará el toldo en los días de sol y cuando haga calor en el interior, para evitar que se caliente mucho la casa.

Obtenga la tabla de verdad, la función binaria, y el correspondiente
circuito lógico del sistema