

Circuito lógico

¿Qué es la Lógica?:

Es una ciencia formal y una rama de la Filosofía que estudia los principios de la demostración e inferencia válida.

La Lógica examina la validez de los argumentos en términos de su estructura, (estructura Lógica), independientemente del contenido específico del discurso y de la lengua utilizada en su expresión y de los estados reales a los que dicho contenido se pueda referir.

Tradicionalmente ha sido considerada como una parte de la Filosofía. Pero en su desarrollo histórico, a partir del final del siglo XIX, y su formalización simbólica ha mostrado su íntima relación con las Matemáticas; de tal forma que algunos la consideran como Lógica Matemática.

En el siglo XX la Lógica ha pasado a ser principalmente la Lógica simbólica. Un cálculo definido por unos símbolos y unas reglas de inferencia. Lo que ha permitido un campo de aplicación fundamental en la actualidad: la Informática.

Lógica Proposicional:

La Lógica proposicional es un sistema formal diseñado para analizar ciertos tipos de argumentos. En Lógica proposicional, las fórmulas representan proposiciones y las conectivas Lógicas son operaciones sobre dichas fórmulas, capaces de formar otras fórmulas de mayor complejidad.

Como otros sistemas lógicos, la Lógica proposicional intenta esclarecer nuestra comprensión de la noción de consecuencia Lógica para el rango de argumentos que analiza.

Tabla de la Verdad:

Es una tabla que despliega el valor de verdad de una proposición compuesta, para cada combinación de valores de verdad que se pueda asignar a sus componentes.

Considérese dos proposiciones A y B . Cada una puede tomar uno de dos valores de verdad: o V (verdadero), o F (falso). Por lo tanto, los valores de verdad de A y de B pueden combinarse de cuatro maneras distintas: o ambas son verdaderas; o A es verdadera y B falsa, o A es falsa y B verdadera, o ambas son falsas. Esto puede expresarse con una tabla simple:

A	B
V	V
V	F
F	V
F	F

Operadores Básicos de la Tabla de la Verdad:

Los operadores fundamentales se definen así:

Negación

La negación es un operador que opera sobre un único valor de verdad, devolviendo el valor contradictorio de la proposición considerada.

A	$\neg A$
V	F
F	V

Conjunción

La conjunción es un operador que opera sobre dos valores de verdad, típicamente los valores de verdad de dos proposiciones, devolviendo el valor de verdad *verdadero* cuando ambas proposiciones son verdaderas, y *falso* en cualquier otro caso.

La tabla de verdad de la conjunción es la siguiente:

A	B	$A \wedge B$
V	V	V
V	F	F
F	V	F
F	F	F

Operadores Básicos de la Tabla de la Verdad:

Disyunción

La disyunción es un operador que opera sobre dos valores de verdad, típicamente los valores de verdad de dos proposiciones, devolviendo el valor de verdad *verdadero* cuando una de las proposiciones es verdadera, o cuando ambas lo son, y *falso* cuando ambas son falsas.

La tabla de verdad de la disyunción es la siguiente:

A	B	$A \vee B$
V	V	V
V	F	V
F	V	V
F	F	F

Implicación o Condicional

El condicional material es un operador que opera sobre dos valores de verdad, típicamente los valores de verdad de dos proposiciones, devolviendo el valor de verdad *falso* sólo cuando la primera proposición es verdadera y la segunda falsa, y *verdadero* en cualquier otro caso.

La tabla de verdad del condicional material es la siguiente:

A	B	$A \rightarrow B$
V	V	V
V	F	F
F	V	V
F	F	V

Operadores Básicos de la Tabla de la Verdad:

Bicondicional

El bicondicional o doble implicación es un operador que funciona sobre dos valores de verdad, típicamente los valores de verdad de dos proposiciones, devolviendo el valor de verdad *verdadero* cuando ambas proposiciones tienen el mismo valor de verdad, y falso cuando sus valores de verdad difieren.

La tabla de verdad del bicondicional es la siguiente:

A	B	$A \leftrightarrow B$
V	V	V
V	F	F
F	V	F
F	F	V

Álgebra de Boole aplicada a la Informática:

Se dice que una variable tiene **valor booleano** cuando, en general, la variable contiene un 0 lógico o un 1 lógico. Esto, en la mayoría de los lenguajes de programación, se traduce en *false* (falso) o *true* (verdadero), respectivamente.

Una variable puede no ser de tipo booleano, y guardar valores que, en principio, no son booleanos; ya que, globalmente, los compiladores trabajan con esos otros valores, numéricos normalmente aunque también algunos permiten cambios desde, incluso, caracteres, finalizando en valor booleano. ...

El 0 lógico

El valor booleano de negación suele ser representado como **false**, aunque también permite y equivale al valor natural, entero y decimal (exacto) 0, así como la cadena "false", e incluso la cadena "0".



El 1 lógico

En cambio, el resto de valores apuntan al valor booleano de afirmación, representado normalmente como **true**, ya que, por definición, el valor 1 se tiene cuando no es 0. Cualquier número distinto de cero se comporta como un 1 lógico, y lo mismo sucede con casi cualquier cadena (menos la "false", en caso de ser ésta la correspondiente al 0 lógico).



Isomorfismo:

La aplicación de la Lógica proposicional a los circuitos eléctricos es posible en virtud del isomorfismo existente entre ambos. Llamamos isomorfismo a la relación de igualdad estructural que existe entre dos objetos.

En efecto, el Matemático e Ingeniero norteamericano Claudio Shannon – uno de los diseñadores de las modernas computadoras – descubrió, en 1936, el isomorfismo (igualdad de formas básicas) existentes entre la Lógica de proposiciones y la teoría de los circuitos eléctricos.

Gracias a este descubrimiento se ha desarrollado una teoría sistemática de los circuitos eléctricos y ésta ha hecho posible resolver cualquier problema concerniente a la construcción y funcionamiento de estos circuitos básicos de las computadoras electrónicas.

Para hacer el isomorfismo es necesario considerar sólo 3 funciones Lógicas: la conjunción, la disyunción y la negación. Como a través de esas 3 funciones básicas se puede definir las demás funciones Lógicas, entonces el isomorfismo es total.

Isomorfismo:

La verdad o la falsedad de una proposición puede representarse por "1" y "0". Mientras que el "1" indica presencia, el "0" indica ausencia, de ahí que 1 y 0 se asocian a lo verdadero y lo falso.

PRIMER ISOMORFISMO

Una proposición simple puede ser verdadera o falsa. De igual manera podemos decir que un interruptor puede estar cerrado o abierto.

Ser verdadero es como estar cerrado y ser falso es como estar abierto. Las posibilidades son análogas:

V = "1" = interruptor cerrado → pasa la información

V = "0" = interruptor abierto → no pasa la información

SEGUNDO ISOMORFISMO

Ahora bien, una proposición compuesta puede ser verdadera o falsa. De igual manera, si fluye la información entonces el foquito encenderá, y si no fluye entonces el foquito no encenderá.

El estado de encendido y apagado corresponde a los valores de verdad y falsedad respectivamente. De nuevo, las posibilidades son análogas:

V = "1" = foco prendido → la información está pasando

F = "0" = foco apagado → la información no está pasando

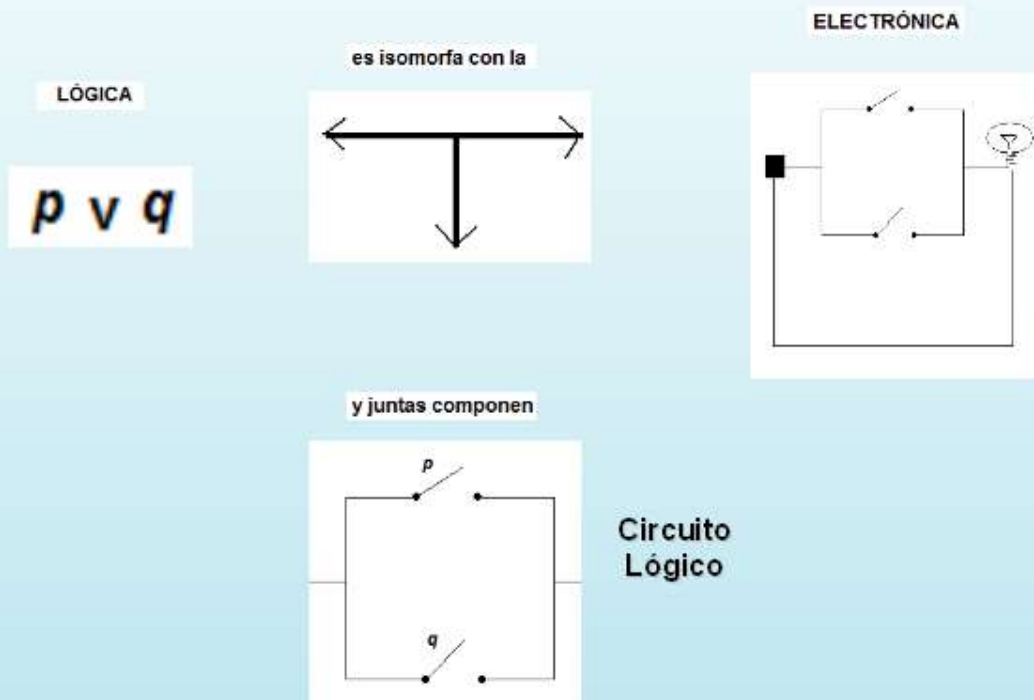
Compuerta Lógica

Una **puerta Lógica**, o **compuerta Lógica**, es un dispositivo electrónico que es la expresión física de un operador booleano en la Lógica de conmutación. Cada puerta Lógica consiste en una red de dispositivos interruptores que cumple las condiciones booleanas para el operador particular. Son esencialmente circuitos de conmutación integrados en un chip.

Claude Elwood Shannon experimentaba con relés o interruptores electromagnéticos para conseguir las condiciones de cada compuerta Lógica, por ejemplo, para la función booleana **Y** (AND) colocaba interruptores en circuito serie, ya que con uno solo de éstos que tuviera la condición «abierto», la salida de la compuerta Y sería = 0, mientras que para la implementación de una compuerta **O** (OR), la conexión de los interruptores tiene una configuración en circuito paralelo. La tecnología microelectrónica actual permite la elevada integración de transistores actuando como conmutadores en redes lógicas dentro de un pequeño circuito integrado. El chip de la CPU es una de las máximas expresiones de este avance tecnológico.

En nanotecnología se está desarrollando el uso de una compuerta Lógica molecular, que haga posible la miniaturización de circuitos.

Circuito Lógico:



Circuito Lógico:

Son estructuras formales (sistemas abstractos) que representan sistemas para la transmisión de información de toda índole (desde la electricidad hasta datos informáticos) simulando el comportamiento real de un circuito eléctrico.

Un Circuito eléctrico es toda de transmisión de impulsos eléctricos.

Los circuitos eléctricos reales tienen los siguientes elementos:

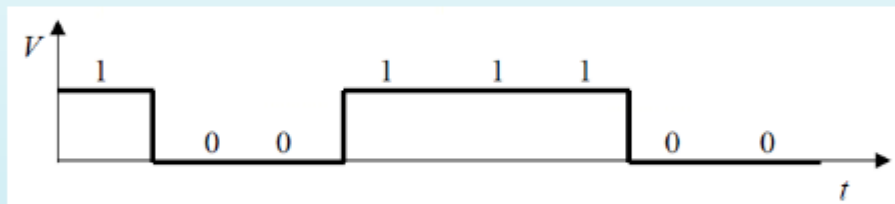
- A. Fuente de energía (batería, pila, tomacorriente)
- B. Cable de transmisión
- C. Interruptores (llamados así porque interrumpen o permiten el flujo de electricidad)
- D. Resistencia o receptor de información (foco, lámpara)

La energía parte del polo negativo de la fuente y se transmite por el cable llega hasta el foco (que se prende) y viaja por el cable hasta llegar al polo positivo de la fuente.

Circuito Lógico:

Un circuito lógico es un dispositivo que tienen una o más entradas y exactamente una salida. En cada instante cada entrada tiene un valor, 0 o 1; estos datos son procesados por el circuito para dar un valor en su salida, 0 o 1.

Los valores 0 y 1 pueden representar ciertas situaciones físicas como, por ejemplo, un voltaje nulo y no nulo en un conductor.



Circuito Lógico:

Aunque los **circuitos electrónicos** podrían parecer muy complejos, en realidad se construyen de un número muy grande de circuitos muy simples.

En un **circuito lógico digital** se transmite información binaria (ceros y unos) entre estos circuitos y se consigue un circuito complejo con la combinación de bloques de circuitos simples.

La información binaria se representa en la forma de: (ver gráficos arriba)

- "0" ó "1",
- "abierto" ó "cerrado" (interruptor),
- "On" y "Off",
- "falso" o "verdadero", etc.

Los **circuitos lógicos** se pueden representar de muchas maneras. En los circuitos representados con gráficos la lámpara puede estar encendida o apagada ("on" o "off"), dependiendo de la posición del interruptor (apagado o encendido), los posibles estados del interruptor o interruptores que afectan un circuito se pueden representar en una **tabla de verdad**.

Circuito Lógico (Compuertas Básicas):

Los circuitos lógicos se construyen a partir de ciertos circuitos elementales denominados compuertas lógicas, entre las cuales diferenciaremos:

- Compuertas lógicas básicas: OR, AND, NOT.
- Compuertas lógicas derivadas: NOR, NAND.

Compuerta And: La operación *And* requiere que todas las señales sean simultáneamente verdaderas para que la salida sea verdadera. Así, el circuito de la figura necesita que ambos interruptores estén cerrados para que la luz encienda.

En una compuerta AND con entradas A y B, la salida Y resulta: $Y = A \cdot B$

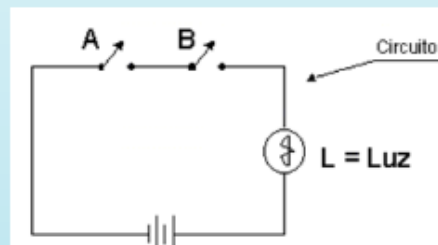


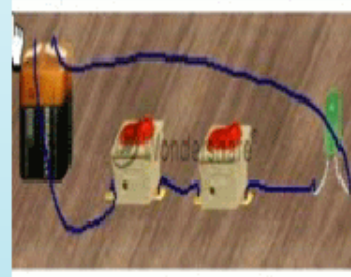
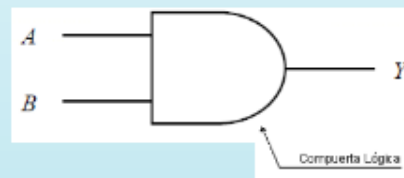
Tabla de Verdad

A	B	L
0	0	0
0	1	0
1	0	0
1	1	1

Circuito Lógico (Compuertas Básicas):

Compuerta And: Los estados posibles del circuito se pueden modelar en la Tabla de Verdad que tiene asociada. Sabemos que los interruptores sólo pueden tener dos estados, abiertos o cerrados, si el interruptor abierto se representa mediante el cero (0 o falso) y el cerrado mediante el valor uno (1 o verdadero) entonces en la tabla de verdad asociada se puede ver la situación que se describía en el párrafo anterior, cuando se decía que la luz sólo prende cuando ambos interruptores están cerrados, es decir, si $A = 1$ y $B = 1$ entonces $L = 1$.

Para efectos de este trabajo, la operación And la representaremos como la función $\text{And}(A, B)$, donde A y B serían los parámetros de entrada (los mismos valores de A y B en el circuito) y $L = \text{And}(A, B)$, correspondería a la forma de asignación de valor a L. En este caso el parámetro de salida es la misma función And.



Circuito Lógico (Compuertas Básicas):

Compuerta Or. La operación Or tiene similares características a la operación And, con la diferencia que basta que una señal sea verdadera para que la señal resultante sea verdadera. En la figura se puede ver tal situación.

Note que en el circuito los interruptores están en paralelo, por lo cual basta que uno de ellos esté cerrado para que el circuito se cierre y encienda la luz.

En una compuerta OR con entradas A y B, la salida Y resulta: $Y = A + B$

La operación Or también tiene una representación funcional como Or (A, B) donde A y B serían los parámetros de entrada (los mismos valores de A y B en el circuito) y $L = \text{Or}(A, B)$, correspondería a la forma de asignación de valor a L. En este caso, el parámetro de salida es la misma función Or.

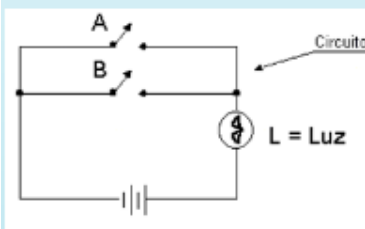
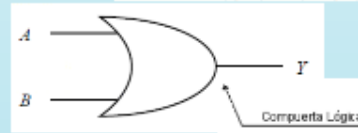


Tabla de Verdad

A	B	L
0	0	0
0	1	1
1	0	1
1	1	1



Circuito Lógico (Compuertas Básicas):

Compuerta Not: La última de las tres operaciones fundamentales, la cual también se conoce como negación, complemento o inversión, es mucho más simple que las anteriores. En la figura se puede observar el circuito, que en este caso tiene la particularidad de que al estar el interruptor abierto la luz enciende, cuando él está en posición de cerrado la luz permanecería apagada.

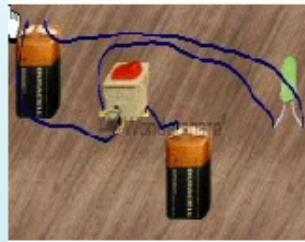
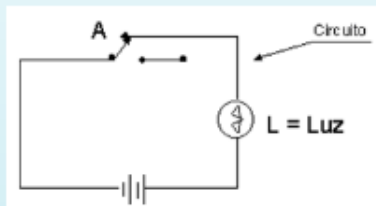
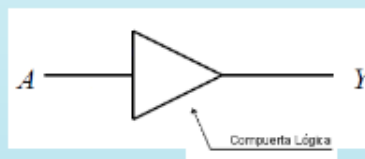


Tabla de Verdad

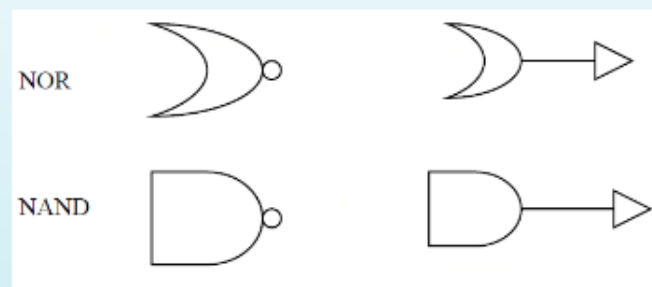
A	L
0	1
1	0

La notación funcional para esta operación será $\text{Not}(A)$, donde A corresponde a la señal de entrada y $\text{Not}(A)$ corresponde al valor complementario de A . **En una compuerta NOT con entrada A , la salida Y resulta: $Y = \bar{A}$**



Circuito Lógico (Compuertas):

Compuertas NOR y NAND: Las compuertas NOR y NAND no son básicas. Una *compuerta NOR equivale a una compuerta OR seguida de una compuerta NOT*. Una *compuerta NAND equivale a una compuerta AND seguida de una compuerta NOT*.

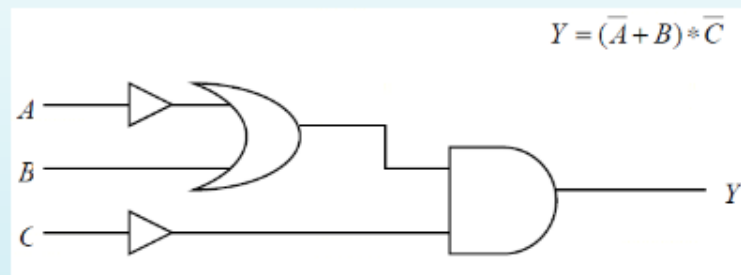


Por lo tanto, cuando las entradas son A y B , las salidas de estas compuertas resultan:

- NOR: $Y = A + B$
- NAND: $Y = A \cdot B$

Circuito Lógico:

Los circuitos lógicos se forman combinando compuertas lógicas. La salida de un circuito lógico se obtiene combinando las tablas correspondientes a sus compuertas componentes. Por ejemplo:



Es fácil notar que las tablas correspondientes a las compuertas OR, AND y NOT son respectivamente idénticas a las tablas de verdad de la disyunción, la conjunción y la negación en la lógica de enunciados, donde sólo se ha cambiado V y F por 0 y 1. Por lo tanto, los circuitos lógicos, de los cuales tales compuertas son elementos, forman un *álgebra de Boole* al igual que los enunciados de la *lógica de enunciados*.

Circuito Lógico:

Adoptaremos, entonces, aquí las mismas convenciones adoptadas en el caso del álgebra de Boole.

- Omitimos el símbolo *, usándose en su lugar la yuxtaposición de variables.
- Establecemos que + es más fuerte que * y * es más fuerte que .

Puesto que tanto el álgebra de Boole es la estructura algebraica tanto de los circuitos como de la lógica de enunciados, la salida de un circuito lógico también puede expresarse en el lenguaje de la lógica de enunciados. Por ejemplo, la salida del circuito anterior resulta:

$$(\bar{A} + B) * \bar{C} \quad (\neg p \vee q) \wedge \neg r$$

Diseño Circuito Lógico:

A continuación se realizará el análisis respectivo de la construcción de un circuito lógico en físico el cual servirá de ejemplo para la aplicación de las herramientas previamente descritas en este trabajo.

Planteamiento: Nuestro objetivo es realizar un circuito lógico en físico el cual demostrará la aplicación de varias reglas de la lógica a través de compuertas lógicas y además proporcionará de una meta al usuario, al cual se le pedirá que demuestre mediante la aplicación de las reglas cual camino se debe seguir para realizar una Tautología, ésta solo podrá ser demostrada por un único camino posible.

Análisis: para llevar a cabo nuestro diseño se utilizarán varios tipos de herramientas las cuales serán:

- Diagrama de flujo
- Algebra de Boole
- Lógica Proposicional
- Tabla de la verdad
- Circuito Eléctrico (en serie y en paralelo)

Diseño Circuito Lógico:

Lo primero que se hizo fue comprobar mediante una serie de pasos ordenados y lógicos, las diferentes combinaciones de las reglas de la disyunción conjunción y negación enfocándonos solo en el uso de 2 variables y los diferentes resultados posibles.

Luego de comprobado las combinaciones posibles se decide que camino seguir, para poder dar diferentes resultados controlados y solo un resultado que dé tautología.

Se utilizó la tabla de la verdad en cada combinación y sus resultados los cuales fueron expresados en 0 y 1 se expresarán físicamente en el circuito mediante bombillos led de la siguiente manera:

Led Verde = 1
Led Rojo = 0

Tablas de la verdad Circuito Lógico:

A continuación se presentan las tablas de la verdad realizadas para nuestras deducciones:

P	Q	Conjunción (P ^ Q)			Negación (~ P) (~ Q)			Disyunción (P ó ~ Q)			Conjunción (~ P ^ ~ Q)		
1	1	1	1	0	1	0	0	1	0	0	1	0	
1	0	1	0	0	0	1	1	0	0	1	0	0	
0	1	0	0	1	1	0	0	1	1	0	0	1	
0	0	0	0	0	1	0	1	0	1	1	0	0	

Disyunción

(P ^ Q) ó (P ó Q)	(P ó Q)	(P ó Q)	(P ó Q)
1	1	1	1
1	0	0	1
0	0	1	1
0	0	0	0

Negación

(~ (P ^ Q))	(~ (P ^ Q))	(~ (P ^ Q))	(~ (P ^ Q))
0	1	1	1
1	1	0	0
1	0	0	1
1	0	0	0

Bicondicional

(P ^ Q) ó (P ó Q) <-> (~ P ó ~ Q)	(P ó Q) <-> (~ P ó ~ Q)	(P ó Q) <-> (~ P ó ~ Q)	(P ó Q) <-> (~ P ó ~ Q)
1	1	1	1
1	0	0	1
0	0	1	1
0	0	0	0

Condicional

((P ^ Q) ó (P ó Q) -> (~ P ó ~ Q))	((P ^ Q) ó (P ó Q) -> (~ P ó ~ Q))	((P ^ Q) ó (P ó Q) -> (~ P ó ~ Q))	((P ^ Q) ó (P ó Q) -> (~ P ó ~ Q))
1	1	1	1
1	0	0	1
0	0	1	1
0	0	0	0

Disyunción

(~ (P ^ Q) ^ (~ P ^ ~ Q))	(~ (P ^ Q) ^ (~ P ^ ~ Q))	(~ (P ^ Q) ^ (~ P ^ ~ Q))	(~ (P ^ Q) ^ (~ P ^ ~ Q))
0	1	1	0
1	1	0	0
1	0	0	1
1	0	0	0

Bicondicional

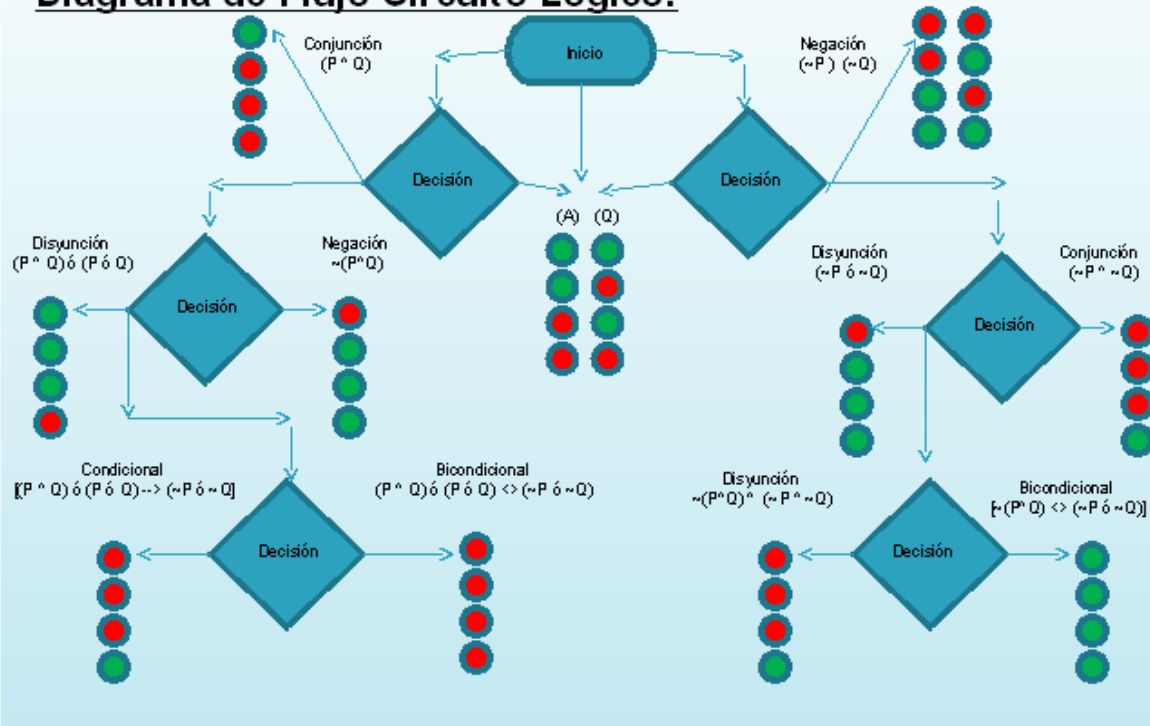
(~ (P ^ Q) <-> (~ P ó ~ Q))	(~ (P ^ Q) <-> (~ P ó ~ Q))	(~ (P ^ Q) <-> (~ P ó ~ Q))	(~ (P ^ Q) <-> (~ P ó ~ Q))
0	1	1	1
1	1	0	1
1	0	0	1
1	0	0	0

Diseño Circuito Lógico:

Se utilizó en nuestro circuito conexiones eléctricas en serie y en paralelo para poder lograr que con el paso o bloqueo del flujo de la corriente eléctrica, se pueda simular la toma de decisiones para la aplicación de las leyes.

Luego se realizó un diagrama de flujo para saber de que manera se ordenaría el circuito con la aplicación de las reglas ya expresadas de manera tal, que pueda cumplir con nuestros objetivos.

Diagrama de Flujo Circuito Lógico:



Diseño Circuito Lógico:

Para lograr que el circuito solo le permitiera al usuario una sola vía posible para la tautología, se realizaron las conexiones de manera tal que solo el uso de las reglas lógicas y ordenadas correctas diera la mencionada ley.

El uso de cualquier otra vía simplemente no daría tautología, pero se previó en el circuito un sistema de comprobación aleatoria para ocultar el camino más obvio a la respuesta.

Luego de conectado el circuito se comprobó mediante el uso de una tabla de la verdad, las diferentes posibilidades del usuario para lograr la tautología y evitar errores de conexión. Esta Comprobación de posibilidades arrojó que solo había una manera posible de lograr tautología.

Maqueta del Circuito Lógico:

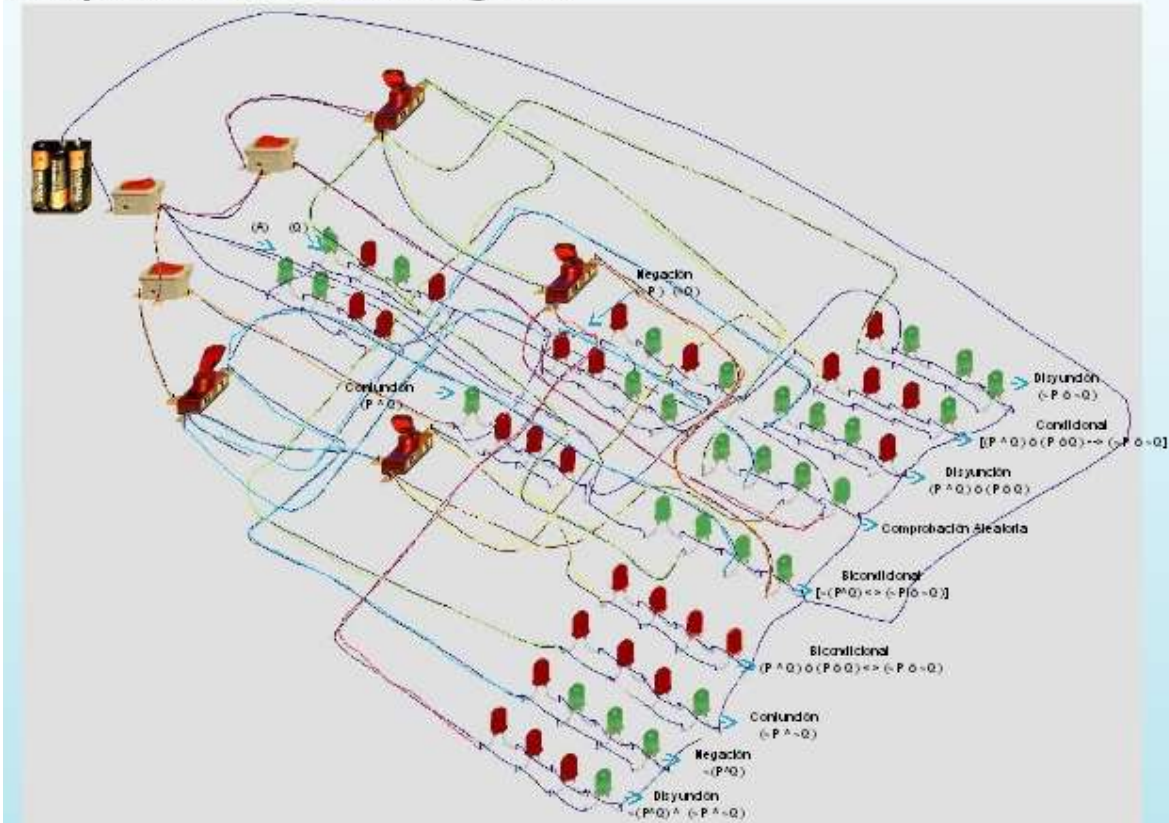
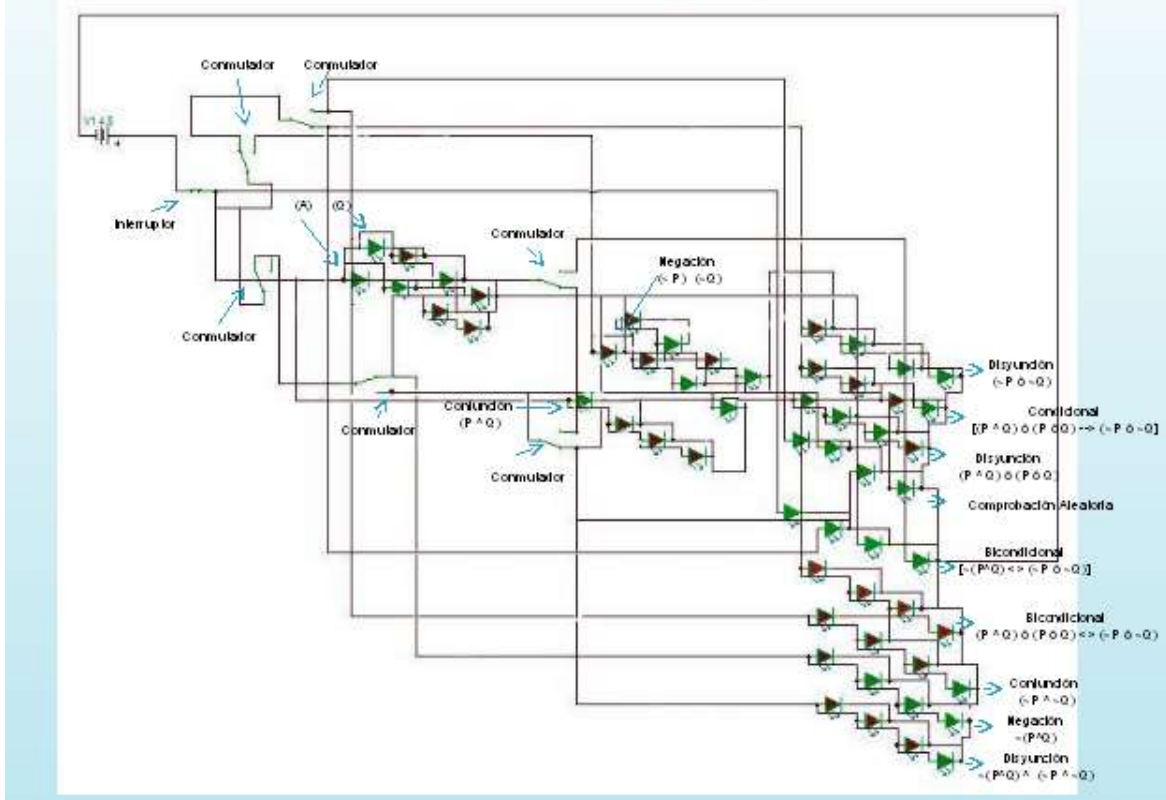


Diagrama Eléctrico del Circuito Lógico:



Bibliografía:

Internet

<https://www.monografias.com/trabajos104/circuito-logico/circuito-logico>

http://www.unicrom.com/Tut_circuitoslogicos_.asp

http://html.rincondelvago.com/circuitos-logicos_1.html

<http://es.wikipedia.org/wiki/Lógica>

<http://www.econ.uba.ar/www/departamentos/humanidades/plan97/logica/Legris/apuntes/AP-Circuitos.pdf> Universidad de Buenos Aires

<http://www.slideshare.net/rafael.mora/circuitos-logicos-presentation>