

REPÚBLICA BOLIVARIANA DE VENEZUELA
MINISTERIO DEL PODER POPULAR PARA LA EDUCACION UNIVERSITARIA
UNIVERSIDAD NACIONAL EXPERIMENTAL MARITIMA DEL CARIBE
CATEDRA: TECNOLOGIA DIGITAL I
INGENIERIA INFORMATICA

COMPUERTAS LOGICAS _CIRCUITOS INTEGRADOS (C. I.)

Profesor:

Pedro Grillet

Alumnos:

Cristhian Franquiz CI: 30225322

Jonathan Alarcon CI: 84423567

Catia la mar, 2025

INTRODUCCION

La **lógica digital** constituye el lenguaje fundamental mediante el cual los circuitos integrados procesan información, permitiendo el funcionamiento de sistemas electrónicos en aplicaciones que van desde dispositivos computacionales hasta sistemas automatizados. Su relevancia en la tecnología moderna es indiscutible, ya que conforma la base de la electrónica digital.

En este informe, exploraremos en detalle el fascinante mundo de los circuitos integrados, abarcando desde sus principios fundamentales hasta sus aplicaciones avanzadas. Inicialmente, definiremos qué son los circuitos integrados y cómo se clasifican, proporcionando una visión clara sobre sus características esenciales. Posteriormente, nos enfocaremos en dos de las familias lógicas más utilizadas en la industria: TTL (Transistor-Transistor Logic) y CMOS (Complementary Metal-Oxide-Semiconductor), analizando sus diferencias, ventajas y aplicaciones.

Para materializar estos conceptos en la práctica, introduciremos el uso del **protoboard**, una herramienta indispensable para la construcción y prueba de prototipos electrónicos, que facilita la experimentación sin necesidad de soldadura. Finalmente, llevaremos a cabo una práctica de laboratorio, donde verificaremos el funcionamiento de las compuertas lógicas básicas y exploraremos la aplicación de los **teoremas de De Morgan**, fundamentales para el análisis y diseño de circuitos digitales.

Este informe no solo proporcionará un marco teórico sólido, sino que también permitirá la interacción con los circuitos digitales en un entorno experimental, facilitando el aprendizaje y aplicación de los principios de la lógica digital.

1.- ¿Que son circuitos integrados (CI)?

Los circuitos integrados, comúnmente llamados chips, son piezas electrónicas esenciales que están presentes en la mayoría de los aparatos electrónicos modernos. Están formados por diversos elementos como transistores, resistencias, diodos y condensadores, todos conectados entre sí dentro de una delgada capa de material semiconductor.

Estos circuitos se clasifican en tres tipos principales según el tipo de señales que manejan. Los circuitos digitales procesan información en formato binario (ceros y unos), siendo fundamentales para equipos como ordenadores y micro controladores. Los circuitos analógicos, por otro lado, trabajan con señales continuas, como las que provienen de sensores de temperatura o sonido. Finalmente, los circuitos mixtos combinan funciones digitales y analógicas dentro de un mismo chip, aprovechando las ventajas de ambos sistemas.

2.- A partir de la siguiente tabla, describa las familias de circuitos digitales i ntegrados (TTL y CMOS)

Familias lógicas	Escalas de integración	Número de compuertas	Función	Características
TTL (Transistor Transistor Logic) SSI (integración a	SSI (integración a pequeña escala) MSI (integración a mediana escala)	Varias compuertas, desde unas pocas hasta decenas de compuertas por chip	Compuertas lógicas básicas (AND, OR, NOT, NAND, NOR, entre otros.), flip-flops, registros, entre otros.	Alta velocidad, Alta fiabilidad, Compatibilidad, Consumo de potencia,
CMOS (Complementary Metal-Oxide Semiconductor)	SSI, MSI, LSI (integración a gran escala), VLSI (integración a muy gran escala)	Varias compuertas, desde unas pocas hasta millones de compuertas por chip	Almacenar la configuración del BIOS/UEFI, Fecha y hora del sistema, Orden de arranque (Boot Order)	Bajo Consumo de Energía, Alta Densidad de Integración, Buena Resistencia al Ruido, Escalabilidad

3.- Que es un protoboard y sus divisiones.

Un **protoboard**, también conocido como **placa de pruebas** o **breadboard** en inglés, es una herramienta utilizada en electrónica para construir circuitos de manera rápida y sin necesidad de soldadura. Su diseño permite conectar componentes eléctricos, como resistencias, transistores, condensadores y microcontroladores, a través de pequeños orificios con contactos metálicos internos.

Estructura y divisiones del protoboard

El protoboard se divide en varias secciones, cada una con una función específica:

1. Filas de alimentación (Power Rails)

- Se encuentran en los extremos superior e inferior de la placa.
- Generalmente están marcadas con líneas rojas y azules para indicar los polos positivo (+) y negativo (-).
- Se utilizan para distribuir la energía a los diferentes componentes del circuito.

2. Zona de conexiones (Terminal Strips)

- Es la parte central del protoboard, donde se insertan la mayoría de los componentes.
- Cada columna de orificios está conectada internamente de forma horizontal en grupos de cinco.
- Se usa para conectar resistencias, transistores y otros elementos del circuito.

3. Canal central o separación (Ditch)

- Divide la zona de conexiones en dos partes.
- Se usa para colocar circuitos integrados (ICs), ya que sus pines necesitan estar separados y no conectarse directamente.

4.- Que son LEDS, funcionamiento, montaje práctico- protoboard.

¿Qué son los LEDS?

Los **LEDs** (*Light Emitting Diodes*) son dispositivos electrónicos que emiten luz cuando una corriente eléctrica pasa a través de ellos. Son ampliamente utilizados en iluminación, indicadores y pantallas debido a su eficiencia energética y durabilidad.

Funcionamiento de un LED

Un LED es un diodo semiconductor que permite el paso de corriente en un solo sentido. Cuando la corriente fluye correctamente a través de él, los electrones se

recombinan dentro del material semiconductor, liberando energía en forma de luz. Los LED tienen **dos terminales**:

- **Ánodo (+)** → Conectado al positivo de la fuente.
- **Cátodo (-)** → Conectado al negativo de la fuente.

Si se conectan de manera incorrecta (polarización inversa), el LED no emitirá luz.

Montaje práctico en un protoboard

Para conectar un LED en un **protoboard**, sigue estos pasos:

Material necesario:

- Protoboard
- LED
- Resistencia de **220Ω** o **470Ω** (para limitar la corriente y evitar dañar el LED)
- Cables de conexión
- Fuente de alimentación (puede ser una batería de 9V o un Arduino)

Pasos de conexión en el protoboard:

1. **Coloca el LED** en la zona de conexiones del protoboard. Recuerda que la pata larga es el **ánodo (+)** y la corta es el **cátodo (-)**.
2. **Añade la resistencia** en serie con el ánodo del LED. Esto evitará que una corriente excesiva dañe el componente.
3. **Conecta la fuente de alimentación:**
 - La terminal positiva (+) de la fuente va a la resistencia.
 - La terminal negativa (-) de la fuente va al cátodo del LED.
4. **Enciende la fuente de alimentación** y observa cómo el LED emite luz.

PRACTICA N.1

1.- Ud. debe comprobar el funcionamiento de las compuertas AND, OR, NAND, NOR, NOT, X-OR, X-NOR.(Circuitos integrados: AND-7408, OR-7432, NOT-7404, NAND-7400, NOR-7402 y XOR-7486)

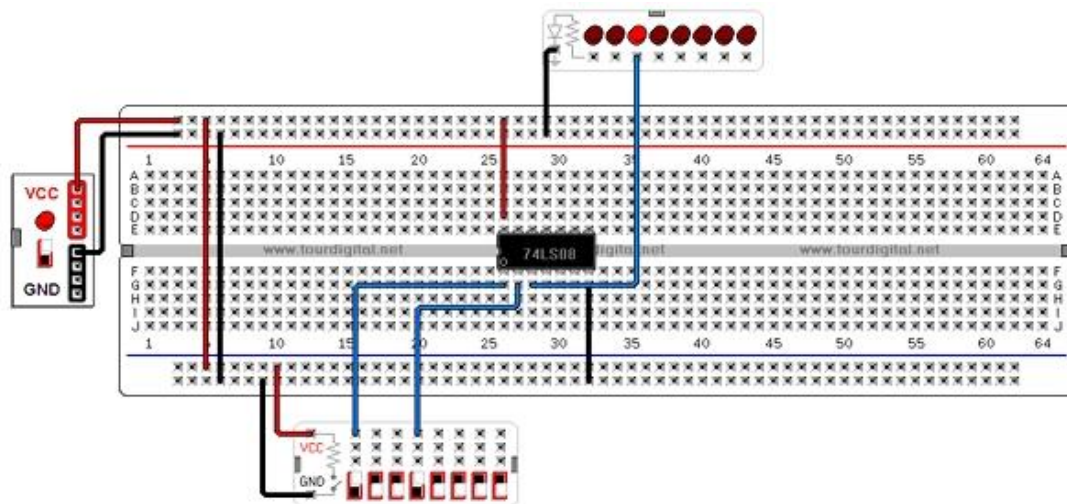
Montaje Práctico de la compuerta AND-7408.

- 1.- Conectar la tierra GND a la patilla 7.
- 2.- Conectar la alimentación +5 Vcc a la patilla 14 del 7408. (AND)
- 3.- Conectar el interruptor de entrada I0 a la patilla 1. Según la compuerta seleccionada.
- 4.- Conectar el interruptor de entrada I1 a la patilla 2. Según la compuerta seleccionada.
- 5.- Conectar la patilla 3 de salida del 7408 al LED correspondiente.

Para cada compuerta, construya la tabla de la Verdad correspondiente y registre los datos obtenidos. ("0" Apagado= 0 volts, "1" Encendido= 5 volts)

Compuerta lógica AND (7408): Compuerta lógica básica. La salida es 1 (verdadera) solo si todas las entradas son 1 (verdaderas).

Prueba en simulador:



```

AND: Bloc de notas
Archivo Edición Formato Ver Ayuda
Version 0.9.7
5
4 10 175 77 128 0 1
5 100 400 206 81 2 2
0 110 110 877 276 8 3
3 498 20 203 58 180 8
1 270 410 206 81 189 9
0
12
255 5 71 189 71 189 71 189 71 124 163 124 163 124 0 1 0 3
0 5 71 249 71 249 94 249 94 137 163 137 163 137 0 1 0 3
0 5 202 137 202 137 202 137 202 371 202 371 202 371 0 3 0 3
255 5 189 124 189 124 189 124 189 358 189 358 189 358 0 3 0 3
255 5 285 421 285 421 254 421 254 421 254 358 254 358 0 9 0 3
0 5 285 479 285 479 241 479 241 479 241 371 241 371 0 9 0 3
0 5 540 280 540 280 540 280 540 371 540 371 540 371 0 3 0 3
255 5 462 215 462 215 462 215 462 124 462 124 462 124 0 3 0 3
65793 5 509 57 509 57 509 57 501 57 501 137 501 137 0 8 0 3
16744448 5 462 280 462 280 326 280 326 280 326 421 326 421 0 3 0 9
16744448 5 475 280 475 280 475 301 383 301 383 421 383 421 0 3 0 9
16744448 5 488 280 488 280 584 280 584 280 584 64 584 64 0 3 0 8
1
7008 1 14 456 228 5 0 8649028 0 3

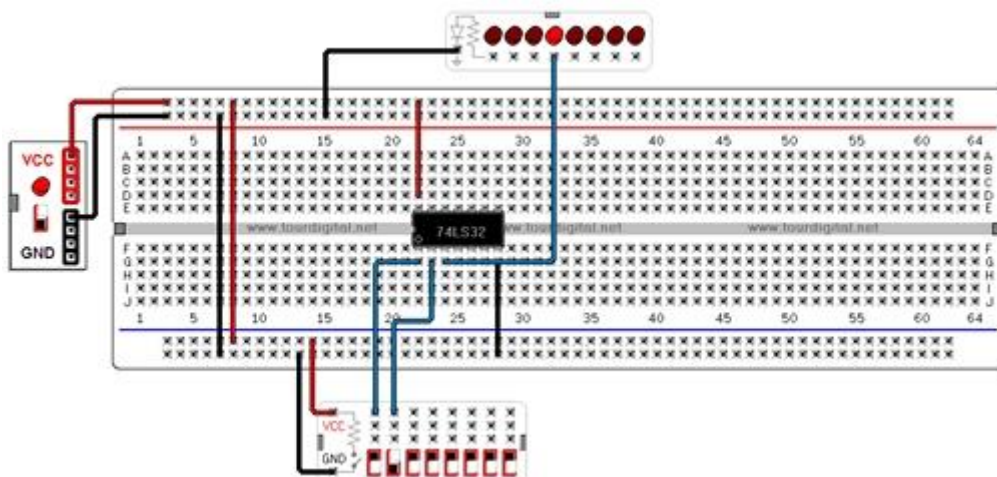
```

Tabla de la verdad AND:

Entrada A	Entrada B	Salida (Q)
0	0	0
0	1	0
1	0	0
1	1	1

Compuerta lógica OR (7432): Compuerta lógica básica. La salida es 1 (verdadera) si al menos una de las entradas es 1 (verdadera).

Prueba en simulador:



```

OR: Bloc de notes
Archivo Edición Formato Ver Ayuda
Version 0.9.7
5
4 148 302 77 128 0 1
5 80 400 206 81 2 2
0 250 250 877 276 8 3
1 451 557 206 81 140 4
3 577 175 203 58 150 5
0
12
255 5 209 316 210 316 210 263 303 263 303 264 303 264 0 1 0 3
0 5 209 376 209 376 233 376 233 277 303 277 303 277 0 1 0 3
255 5 368 264 368 264 368 264 368 498 368 498 368 498 0 3 0 3
0 5 355 277 355 277 355 277 355 511 355 511 355 511 0 3 0 3
255 5 446 498 446 498 446 568 446 568 466 568 466 568 0 3 0 4
0 5 433 511 433 511 433 626 433 626 466 626 466 626 0 3 0 4
0 5 628 420 628 420 628 420 628 511 628 511 628 511 0 3 0 3
255 5 550 355 550 355 550 264 550 264 550 264 0 3 0 3
0 5 588 212 588 212 588 213 459 213 459 277 459 277 0 5 0 3
12615680 5 550 420 550 420 508 420 508 568 507 568 507 568 0 3 0 4
12615680 5 563 420 563 420 563 478 526 478 526 568 526 568 0 3 0 4
12615680 5 576 420 576 420 683 420 683 219 683 219 0 3 0 5
1
7032 1 14 544 368 5 0 8649028 0 3

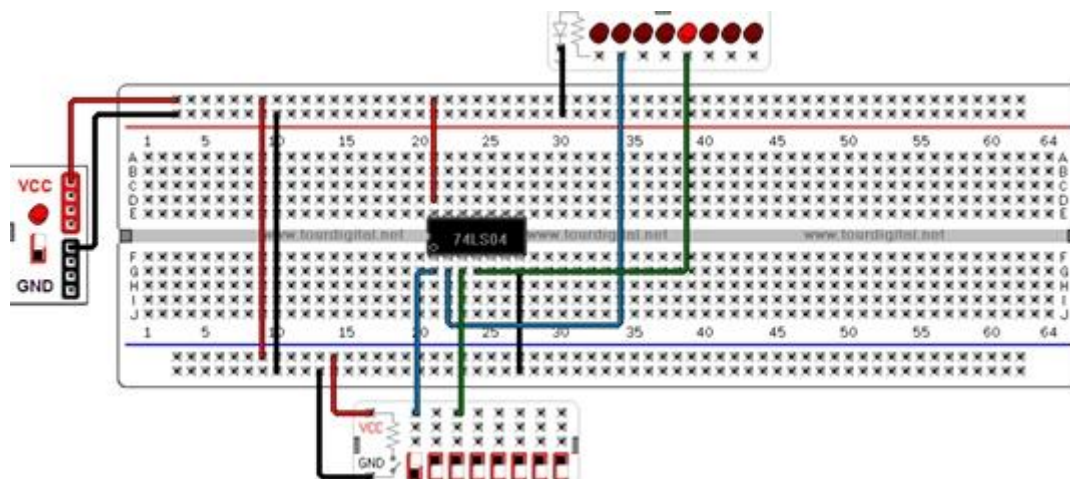
```

Tabla de la verdad OR:

Entrada A	Entrada B	Salida (Q)
0	0	0
0	1	1
1	0	1
1	1	1

Compuerta lógica NOT (7404): Compuerta lógica básica. Invierte la salida. Si la entrada es 1(verdadera) la salida es 0 (falsa) y viceversa.

Prueba en simulador:




```

NOT: Bloc de notas
Archivo Edición Formato Ver Ayuda
Version 0.9.7
5
4 96 274 77 128 0 1
5 80 400 206 81 2 2
0 200 200 877 276 8 3
3 591 130 203 58 140 4
1 414 488 206 81 149 5
0
13
255 5 157 288 157 288 157 288 157 214 253 214 253 214 0 1 0 3
255 5 331 214 331 214 331 214 331 448 331 448 331 448 0 3 0 3
0 5 157 348 157 348 179 348 179 227 253 227 253 227 0 1 0 3
0 5 344 227 344 227 344 227 344 461 344 461 344 461 0 3 0 3
0 5 383 461 383 461 383 557 383 557 429 557 429 557 0 3 0 5
0 5 604 227 604 227 604 167 604 167 602 167 602 167 0 3 0 4
255 5 487 305 487 305 487 305 487 214 487 214 487 214 0 3 0 3
65793 5 565 370 565 370 565 370 565 461 565 461 565 461 0 3 0 3
12615680 5 487 370 487 370 472 370 472 499 470 499 470 499 0 3 0 5
12615680 5 500 370 500 370 500 419 657 419 657 174 657 174 0 3 0 4
32768 5 513 370 513 370 512 370 512 499 508 499 508 499 0 3 0 5
32768 5 526 370 526 370 717 370 717 370 717 174 717 174 0 3 0 4
255 5 396 448 396 448 396 499 396 499 429 499 429 499 0 3 0 5
1
7004 1 14 481 318 5 0 8649028 0 3

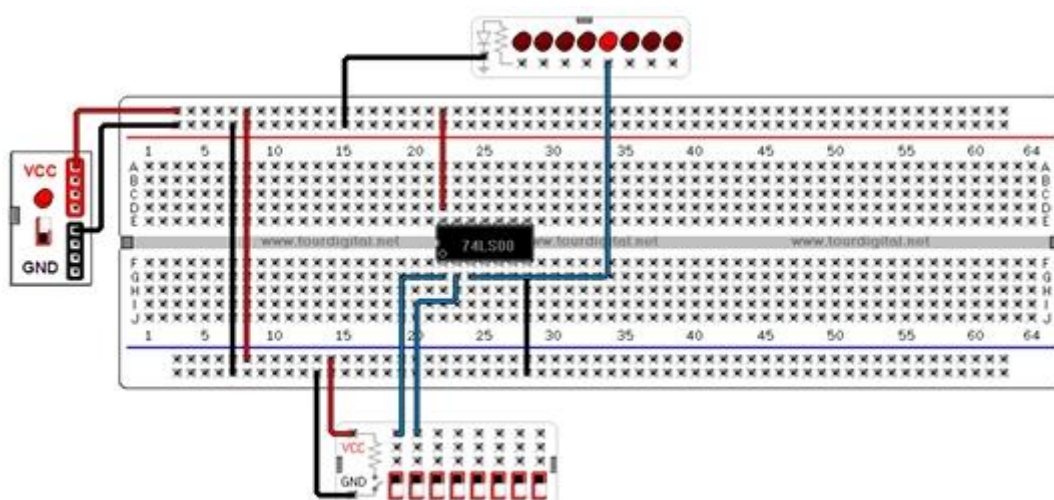
```

Tabla de la verdad NOT:

Entrada (A)	Salida (Q)
0	1
1	0

Compuerta Lógica NAND (7400): Se puede considerar como la negación de una compuerta AND. Si todas las entradas son 1(verdaderas), la salida es 0 (falso). En cualquier otro caso, la salida es 1(verdadero).

Prueba en simulador:



```

NAND: Bloc de notas
Archivo Edición Formato Ver Ayuda
Version 0.9.7
5
4 148 302 77 128 0 1
5 80 400 206 81 2 2
0 250 250 877 276 8 3
1 451 557 206 81 140 4
3 577 175 203 58 150 5
0
12
255 5 209 316 210 316 210 263 303 263 303 264 303 264 0 1 0 3
0 5 209 376 209 376 233 376 233 277 303 277 303 277 0 1 0 3
255 5 368 264 368 264 368 264 368 498 368 498 368 498 0 3 0 3
0 5 355 277 355 277 355 277 355 511 355 511 355 511 0 3 0 3
255 5 446 498 446 498 446 568 446 568 466 568 466 568 0 3 0 4
0 5 433 511 433 511 433 626 433 626 466 626 466 626 0 3 0 4
0 5 628 420 628 420 628 420 628 511 628 511 628 511 0 3 0 3
255 5 550 355 550 355 550 355 550 264 550 264 550 264 0 3 0 3
0 5 588 212 588 212 588 213 459 213 459 277 459 277 0 5 0 3
12615680 5 550 420 550 420 512 420 512 568 507 568 507 568 0 3 0 4
12615680 5 563 420 563 420 562 420 562 443 526 443 526 568 0 3 0 4
12615680 5 576 420 576 420 703 420 703 219 703 219 703 0 3 0 5
1
7000 1 14 544 368 5 0 8649028 0 3

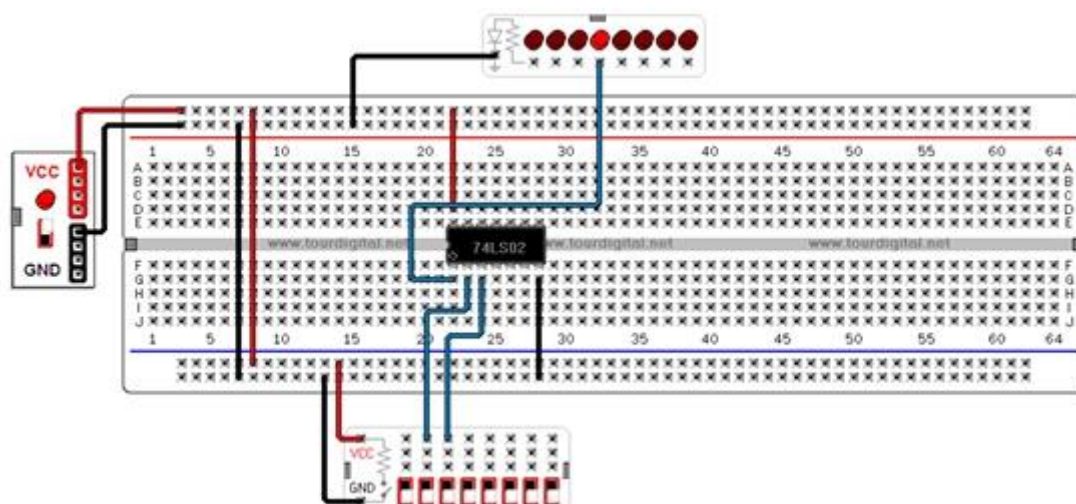
```

Tabla de la verdad NAND:

Entrada A	Entrada B	Salida (Q)
0	0	1
0	1	1
1	0	1
1	1	0

Compuerta Lógica NOR (7402): Negación de OR. La salida es 1(verdadera) si las entradas son 0(falsas).

Prueba en simulador:



```

*XOR: Bloc de notas
Archivo Edición Formato Ver Ayuda
Version 0.9.7
5
4 148 302 77 128 0 1
5 80 400 206 81 2 2
0 250 250 877 276 8 3
1 451 557 206 81 140 4
3 577 175 203 58 150 5
0
12
255 5 209 316 210 316 210 263 303 263 303 264 303 264 0 1 0 3
0 5 209 376 209 376 233 376 233 277 303 277 303 277 0 1 0 3
255 5 368 264 368 264 368 264 368 498 368 498 368 498 0 3 0 3
0 5 355 277 355 277 355 277 355 511 355 511 355 511 0 3 0 3
255 5 446 498 446 498 446 568 446 568 466 568 466 568 0 3 0 4
0 5 433 511 433 511 433 626 433 626 466 626 466 626 0 3 0 4
0 5 628 420 628 420 628 420 628 511 628 511 628 511 0 3 0 3
255 5 550 355 550 355 550 355 550 264 550 264 550 264 0 3 0 3
0 5 588 212 588 212 588 213 459 213 459 277 459 277 0 5 0 3
12615680 5 550 420 550 420 509 420 509 568 507 568 507 568 0 3 0 4
12615680 5 563 420 563 420 563 441 526 441 526 568 526 568 0 3 0 4
12615680 5 576 420 576 420 703 420 703 219 703 219 0 3 0 5
1
7086 1 14 544 368 5 0 8649028 0 3

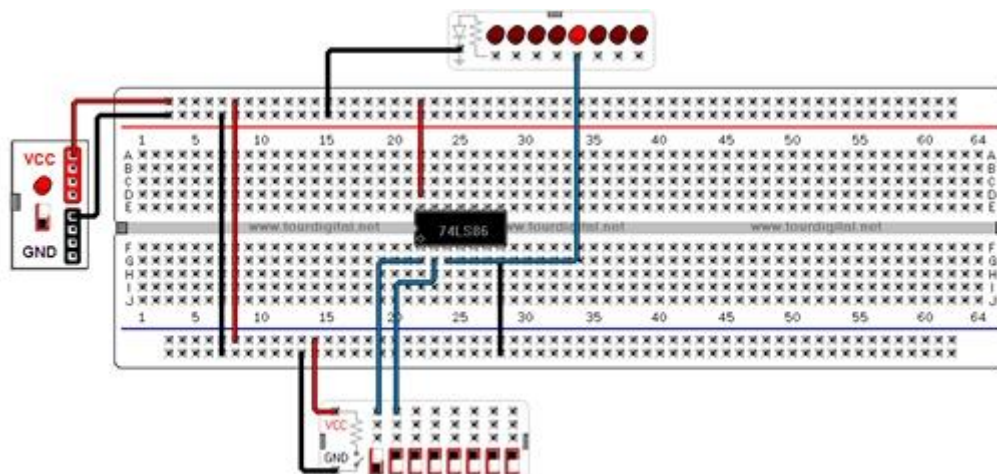
```

Tabla de la verdad NOR:

Entrada A	Entrada B	Salida (Q)
0	0	1
0	1	0
1	0	0
1	1	0

Compuerta Lógica XOR (7486): Abreviatura de "exclusive OR" (O Exclusivo). La salida será 1(verdadera) solo si las entradas son diferentes.

Prueba en simulador:



```

XOR: Bloc de notas
Archivo Edición Formato Ver Ayuda
Version 0.9.7
5
4 148 302 77 128 0 1
5 80 400 206 81 2 2
0 250 250 877 276 8 3
1 451 557 206 81 140 4
3 577 175 203 58 150 5
0
12
255 5 209 316 210 316 210 263 303 263 303 264 303 264 0 1 0 3
0 5 209 376 209 376 233 376 233 277 303 277 303 277 0 1 0 3
255 5 368 264 368 264 368 264 368 498 368 498 368 498 0 3 0 3
0 5 355 277 355 277 355 277 355 511 355 511 355 511 0 3 0 3
255 5 446 498 446 498 446 568 446 568 466 568 466 568 0 3 0 4
0 5 433 511 433 511 433 626 433 626 466 626 466 626 0 3 0 4
0 5 628 420 628 420 628 420 628 511 628 511 628 511 0 3 0 3
255 5 550 355 550 355 550 355 550 264 550 264 550 264 0 3 0 3
0 5 588 212 588 212 588 213 459 213 459 277 459 277 0 5 0 3
12615680 5 550 420 550 420 509 420 509 568 507 568 507 568 0 3 0 4
12615680 5 563 420 563 420 563 441 526 441 526 568 526 568 0 3 0 4
12615680 5 576 420 576 420 703 420 703 420 703 219 703 219 0 3 0 5
1
7086 1 14 544 368 5 0 8649028 0 3

```

Tabla de la Verdad XOR:

Entrada A	Entrada B	Salida (Q)
0	0	0
0	1	1
1	0	1
1	1	0

Funcionamiento y Ejemplos de Cada Compuerta Lógica

1. Compuerta AND (Y)

- Función: Solo produce una salida alta (1) si todas sus entradas son altas (1). Es como decir "y" en lenguaje natural.
- Símbolo: \wedge
- Ejemplo: Un circuito de alarma que se activa solo si se detecta movimiento (entrada A) y se abre una puerta (entrada B).

2. Compuerta OR (O)

- Función: Produce una salida alta (1) si al menos una de sus entradas es alta (1). Es como decir "o" en lenguaje natural.
- Símbolo: \vee
- Ejemplo: Un circuito que enciende una luz si se presiona un botón (entrada A) o se detecta movimiento (entrada B).

3. Compuerta NOT (NO)

- Función: Invierte la entrada. Si la entrada es 0, la salida es 1, y viceversa.
- Símbolo: \neg

- Ejemplo: Un interruptor que enciende una luz cuando está apagado y la apaga cuando está encendido.

4. Compuerta NAND (NO Y)

- Función: Es la negación de una compuerta AND. Produce una salida baja (0) solo si todas las entradas son altas (1).
- Símbolo: \uparrow
- Ejemplo: Un circuito que impide el paso si ambas puertas de seguridad están abiertas.

5. Compuerta NOR (NO O)

- Función: Es la negación de una compuerta OR. Produce una salida baja (0) si al menos una de las entradas es alta (1).
- Símbolo: \downarrow
- Ejemplo: Un circuito que desactiva una alarma solo si todas las ventanas están cerradas y todas las puertas están aseguradas.

6. Compuerta XOR (O Exclusivo)

- Función: Produce una salida alta (1) solo si las entradas son diferentes.
- Símbolo: \oplus
- Ejemplo: Un circuito que enciende una luz cuando solo se presiona uno de dos botones.

2.- Comprobar los Teoremas de Morgan, usando las compuertas correspondientes en el simulador.

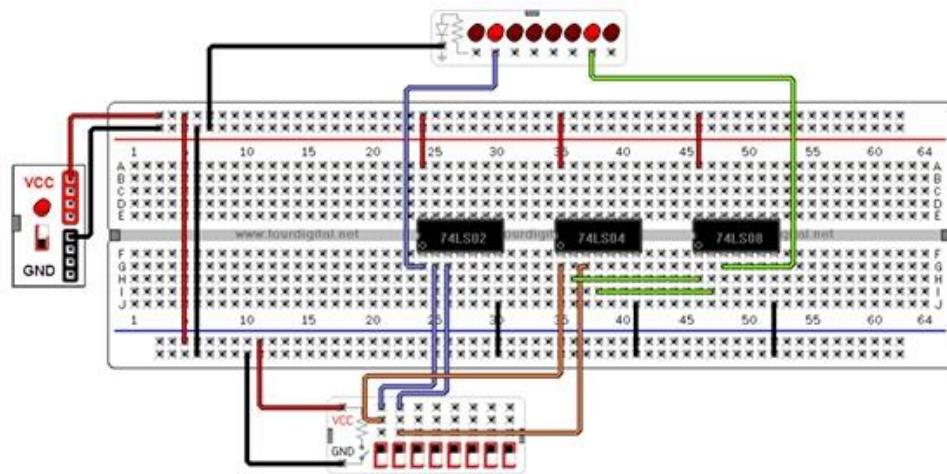
El teorema de Morgan es una herramienta fundamental en lógica matemática y álgebra de Boole. Sirve para simplificar expresiones lógicas complejas, especialmente aquellas que involucran negaciones, conjunciones (AND) y disyunciones (OR).

Primera Ley de Morgan:

La primera ley de De Morgan establece que la negación de una conjunción (AND) es igual a la disyunción (OR) de las negaciones.

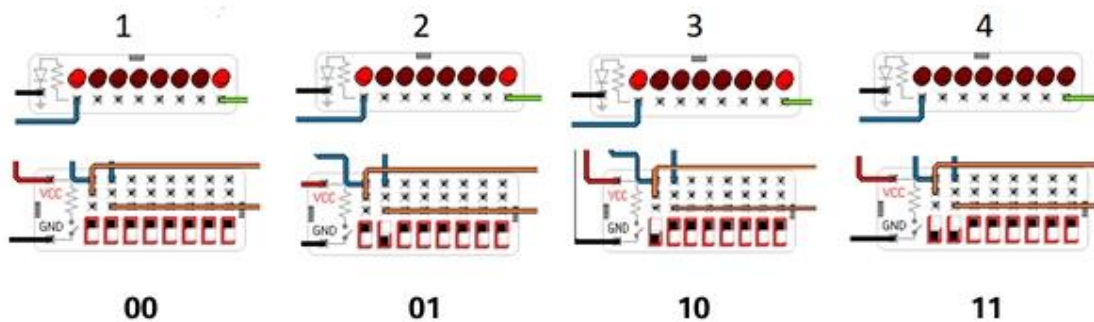
$$\overline{AB} = \overline{A} + \overline{B}$$

Planteamiento en simulador:



Para comprobar este teorema, ambos Leds donde se hicieron las conexiones deben estar en el mismo estado, si uno prende el otro también y si uno se apaga el otro también se apaga.

Pruebas:



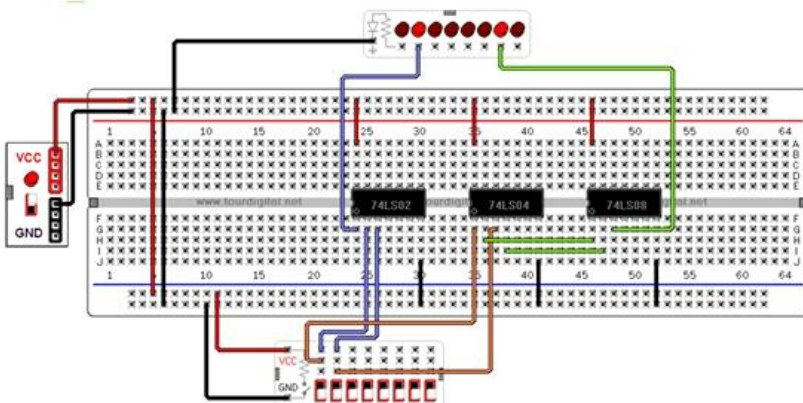
Resultado:

Este teorema es correcto pues las salidas fueron iguales en su representación AND ($X'Y'$) y en su representación OR con sus complementos negados individualmente ($X' + Y'$).

Segunda Ley de Morgan: La segunda ley de De Morgan es el dual de la primera. Establece que la negación de una disyunción (OR) es igual a la conjunción (AND) de las negaciones.

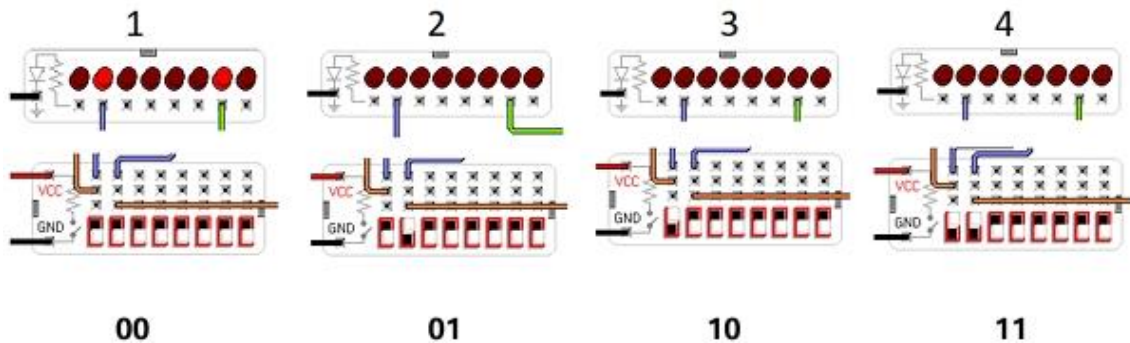
$$\overline{X + Y} = \bar{X} \bar{Y}$$

Planteamiento en simulador:



Para comprobar este teorema, ambos Leds donde se hicieron las conexiones deben estar en el mismo estado, si uno prende el otro también y si uno se apaga el otro también se apaga.

Pruebas:



Resultado:

Este teorema es correcto pues las salidas fueron iguales en su representación. Negar el resultado de una operación OR es equivalente a negar cada una de las entradas y luego realizar una operación AND.

POST-LABORATORIO

En este trabajo exploramos de manera integral el fascinante mundo de los circuitos integrados, abarcando desde su definición y clasificación hasta su aplicación práctica en la construcción de circuitos digitales. A lo largo del desarrollo teórico y experimental del tema, comprendimos que los circuitos integrados constituyen la base fundamental de la electrónica, desempeñando un papel clave en la evolución de la tecnología. Su versatilidad y capacidad de integración han permitido la creación de una vasta gama de dispositivos utilizados en la industria y otras áreas del conocimiento.

El entrenamiento práctico realizado fue esencial para consolidar los conocimientos adquiridos, ya que permitió verificar de forma empírica el funcionamiento de las puertas lógicas básicas y la validez de los teoremas de De Morgan, herramientas fundamentales en el análisis y diseño de sistemas digitales.

En definitiva, los circuitos integrados no solo representan componentes esenciales en cualquier sistema electrónico, sino que también son protagonistas de la gran revolución tecnológica que vivimos en la actualidad. Comprender su funcionamiento, utilidad y aplicación representa un paso crucial para todos aquellos que buscan adentrarse en el apasionante mundo de la electrónica, un campo en constante crecimiento que continúa transformando la manera en que interactuamos con el entorno y la tecnología.