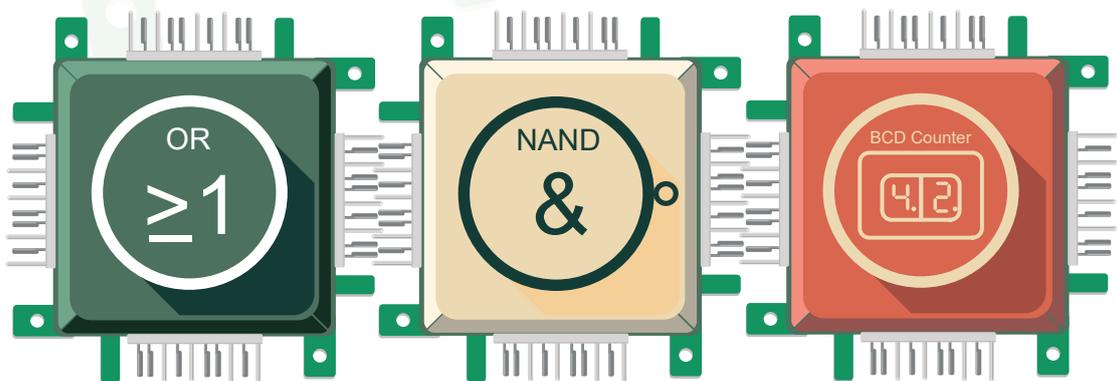


Brick 'R'  
knowledge

# Set lógico

Kit experimental de Brick'R'knowledge  
Experimentierkasten von Brick'R'knowledge



# Aviso legal

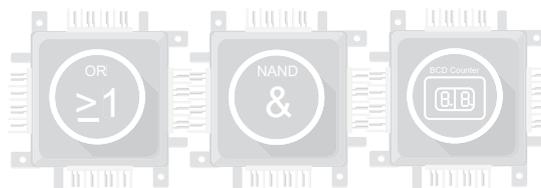
Brick'R'knowledge, manual del set lógico  
Rev. 1.0  
Date: 07.04.2017

ALLNET® y Brick'R'knowledge® son marcas registradas de ALLNET® GmbH Computersysteme.

**ALLNET® GmbH Computersysteme**  
Brick'R'knowledge  
Maistraße 2  
D-82110 Germering

© Copyright 2017 ALLNET GmbH Computersysteme. Todos los derechos reservados.

Toda la información contenida en este manual ha sido recopilada con el mayor cuidado y a nuestro leal saber y entender. Sin embargo, no se pueden descartar completamente los errores. Estamos siempre agradecidos por la notificación de posibles errores. Por favor, envíelos a: [info@brickrknowledge.de](mailto:info@brickrknowledge.de)



DE13101093

# Índice

|  |    |
|--|----|
| 1. Instrucciones de seguridad  | 5  |
| 2. Fundamentos del sistema Brick'R'knowledge                         | 6  |
| 2.1 Brick de tierra  | 6  |
| 2.2 Fuente de alimentación   | 6  |
| 2.3 Conectores   | 7  |
| 3. El set lógico   | 8  |
| 3.1 Bricks del set lógico  | 8  |
| 4. Fundamentos de la tecnología de circuitos digitales               | 16 |
| 4.1 ¿Qué significa exactamente digital?                              | 16 |
| 4.2 Funciones lógicas  | 16 |
| 4.3 Comparación de sistemas numéricos                                | 18 |
| 4.4 Niveles lógicos  | 19 |
| 4.4.1 Rangos de tensión  | 19 |
| 4.4.2 Estados lógicos  | 20 |
| 4.5 Circuitos combinatorios  | 21 |
| 4.6 Circuito secuencial  | 21 |
| 4.7 Fallos de funcionamiento   | 22 |
| 4.8 Dispositivos lógicos programables                                | 23 |
| 4.8.1 Matriz Lógica Programable (PAL) y Matriz Lógica Genérica (GAL) | 23 |
| 4.8.2 Dispositivo Lógico Programable Complejo                        | 23 |
| 4.8.3 Matriz de puertas programables por campo (FPGA)                | 23 |
| 4.8.3.1 Estructura básica de la FPGA                                 | 24 |
| 4.8.3.2 Bloques Lógicos Configurables (CLB)                          | 24 |
| 4.8.3.3 Bloque de Entrada/Salida (IOB)                               | 25 |
| 5. Ejemplos de circuitos   | 26 |
| 5.1 Circuitos básicos  | 26 |
| 5.1.1 Inversor   | 26 |
| 5.1.2 Puerta AND   | 27 |
| 5.1.3 Puerta OR  | 28 |
| 5.1.4 Puerta NAND  | 29 |
| 5.1.5 Puerta NOR   | 30 |
| 5.1.6 Puerta XOR   | 31 |
| 5.1.7 Puerta XNOR  | 32 |
| 5.2 Circuito antirrebotes  | 33 |
| 5.3 Contador digital   | 34 |
| 5.3.1 Semi-sumador de 1-bit  | 34 |
| 5.3.2 Sumador completo de 1-bit                                      | 35 |
| 5.3.3 Sumador completo de 4-bits                                     | 37 |
| 5.4 Flip-flops / Circuito biestable                                  | 38 |
| 5.4.1 Tipos de flip-flop   | 39 |
| 5.4.1.1 Flip-flop sin reloj  | 39 |
| 5.4.1.2 Flip-flop por estado   | 39 |
| 5.4.1.3 Flip-flop disparado por flanco                               | 39 |
| 5.4.1.4 Flip-flop doble disparado por flanco                         | 39 |
| 5.4.2 Flip-flop RS   | 40 |
| 5.4.3 Flip-flop tipo D   | 41 |
| 5.4.4 Flip-flop JK   | 42 |
| 5.5 Registro de desplazamiento                                       | 44 |
| 5.5.1 Registro de desplazamiento de carga serie                      | 44 |

|         |                                      |    |
|---------|--------------------------------------|----|
| 5.6     | Contadores                           | 45 |
| 5.6.1   | Contador asíncrono binario de 4 bits | 46 |
| 5.6.1.1 | Contador asíncrono ascendente        | 47 |
| 5.6.1.2 | Contador asíncrono descendente       | 48 |
| 5.6.2   | Contador BCD asíncrono de 4 bits     | 49 |
| 5.6.3   | Contador síncrono binario de 3 bits  | 51 |
| 5.6.4   | Contador BCD con ejecución y reset   | 53 |
| 5.7     | Divisor de frecuencia                | 55 |
| 6.      | Comunidad brick                      | 56 |
| 7.      | Resumen de los sets de bricks        | 59 |

## Prólogo

El sistema de experimentación Brick'R'knowledge fue presentado por primera vez en la exposición de radio HAM el 28.06.2014 por Rolf-Dieter Klein (Amateurfunkzeichen: DM7RDK).

La característica principal de nuestros sets electrónicos es que los componentes individuales están conectados a través de un sistema de conectores, en el que las piezas a ensamblar son idénticas (hermafrodita). De este modo, incluso los circuitos más complicados se pueden realizar. También es posible el montaje de bloques de construcción individuales en diferentes ángulos!

Hay dos contactos disponibles para el retorno a tierra (0 voltios)!

Esto permite construir circuitos compactos en los que el retorno a tierra garantiza una alimentación de tensión estable para los dispositivos. Otra característica particular es que tales circuitos pueden ser fácilmente explicados y documentados.

Mucha suerte con el set lógico!

*Rolf-Dieter Klein*

## Lista de referencias:

- Compendio electrónico de Patrick Schnabel: <http://www.elektronik-kompodium.de/sites/dig/index.htm>
- Tutor de electrónica por: <http://www.elektroniktutor.de/digital1.html>
- Wikipedia: <https://www.wikipedia.de>

Encontrará conocimientos básicos sobre la tecnología de circuitos digitales y las leyes del álgebra booleana.

## 1. Instrucciones de seguridad

Atención: los Bricks del set de electrónica nunca se deben conectar directamente a la red eléctrica de (115/230V), existe peligro de muerte!

Por favor, utilice solamente la fuente de alimentación que va incluida. El voltaje de nuestra fuente de alimentación es de 9V, el cual no supone ningún riesgo para la salud. Por favor, tenga especial cuidado de que los cables abiertos no entren en contacto o caigan dentro de una regleta de enchufes ya que existe el riesgo de recibir una descarga eléctrica o de electrocutarse. Nunca mire directamente a un diodo de emisor de luz (LED), ya que existe el riesgo de dañar la retina (deslumbrar).

Es importante asegurarse de que la fuente de alimentación (Brick adaptador de Pila) está desconectada después de cada experimento.

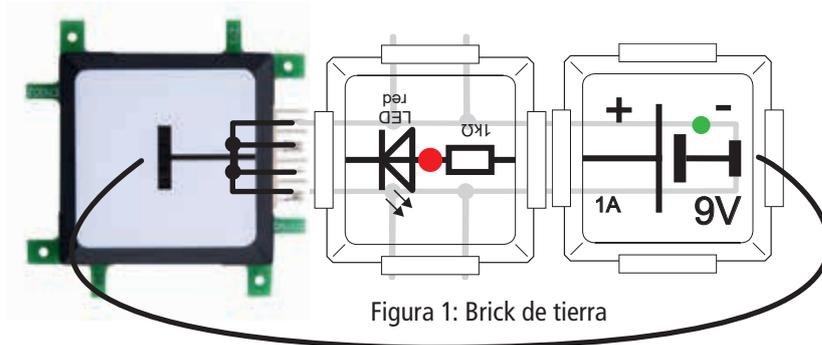
No trague partes del equipo electrónico. En caso de hacerlo, póngase en contacto inmediatamente con un médico!



## 2. Fundamentos del sistema Brick'R'knowledge

### 2.1 Brick de tierra

Uno de los bricks más importantes es el más conocido como brick de tierra. Este tiene un conector de cuatro contactos; los dos contactos centrales están reservados a la transmisión de señal, tal y como lo indica la etiqueta. Los contactos exteriores se utilizan para cerrar el circuito, es decir, para devolver el flujo de corriente a la fuente de tensión. El brick de tierra conecta los dos contactos centrales con los dos exteriores. Se le denomina Brick de tierra porque en la electrónica el término "tierra" no describe el peso de un objeto, sino el potencial de referencia al que se refieren todos los demás potenciales. El brick de tierra establece exactamente esta conexión a 0V en todos los conjuntos de Brick'R'knowledge.



### 2.2 Fuente de alimentación



Figura 2: Fuente de alimentación

El set lógico se alimenta a través de la fuente de alimentación de 9V suministrada. La fuente de alimentación proporciona una corriente de voltaje directa y estabilizada de 9V y una corriente máxima de 1A. En caso de sobrecarga, la fuente de alimentación se desconecta, ya que está protegida contra cortocircuitos.

Las puertas, los flips-flops y los contadores son componentes activos. Para estos bricks usamos la tecnología clásica de CMOS (para más detalles véase el capítulo 4.4) que requieren su propia fuente de alimentación. Téoricamente se pueden conectar a cualquier fuente de alimentación de circuito directo estabilizada en el rango de +5V a +15V. Aún así, recomendamos usar el brick de fuente de alimentación incluido.

Por favor, compruebe siempre el circuito antes de conectar la fuente de alimentación. Después de cada experimento, desconecte la fuente de alimentación de la red eléctrica.

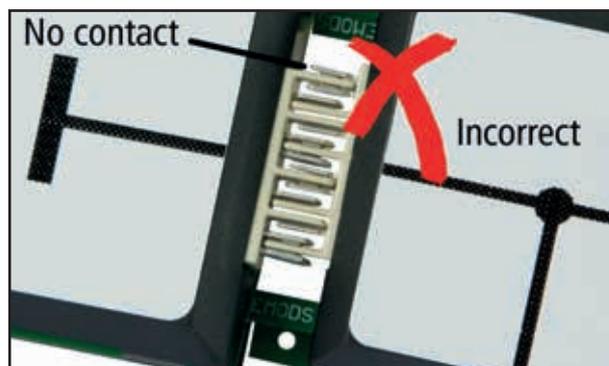
Alternativamente, ofrecemos un brick de batería (ALL-BRICK-0001), usando un bloque de batería de 9V.

## 2.3 Conectores

Al interconectar los Bricks hay que asegurarse de que los contactos se tocan correctamente, ya que de lo contrario podrían producirse interrupciones o incluso cortocircuitos!



Insertado correctamente



Insertado incorrectamente

Figura 3: Conectores

En la imagen de la izquierda tenemos un ejemplo de una conexión realizada correctamente. La conexión consiste en pequeños pines, que se unen mecánicamente y por lo tanto conducen la electricidad.

En la imagen de la derecha tenemos un ejemplo de una conexión incorrecta. Hay espacios entre los contactos que no pueden garantizar el flujo de corriente seguro.

**Prudencia:** Es muy importante controlar el ajuste correcto de los pines, ya que si están muy separados, pueden provocar un cortocircuito. Entonces, el flujo de corriente no se produce a través de nuestros componentes con el efecto deseado, sino que busca el camino más corto de regreso a la fuente de alimentación. Un cortocircuito lleva a un flujo de corriente máximo porque la única resistencia que debe superar la corriente eléctrica es la resistencia interna de la fuente de alimentación, la cual es claramente muy pequeña.

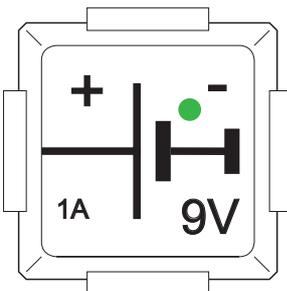
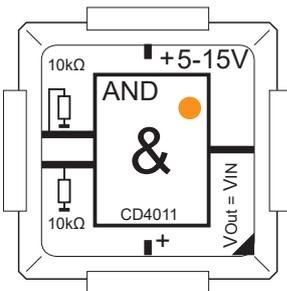
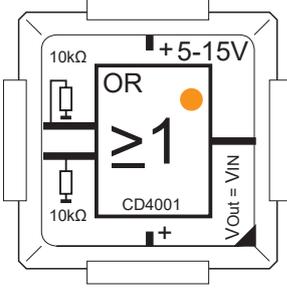
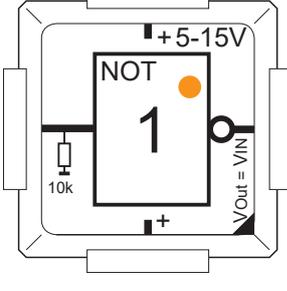


**Importante:** Verificar siempre la posición correcta de los contactos!

### 3. El set lógico

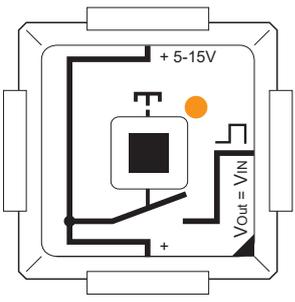
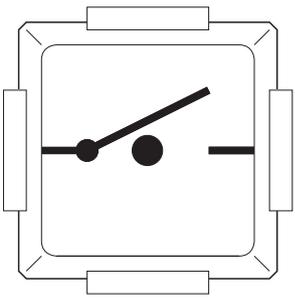
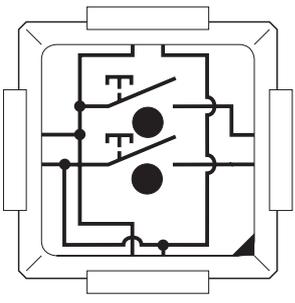
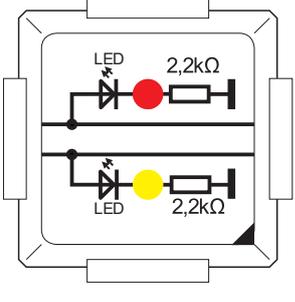
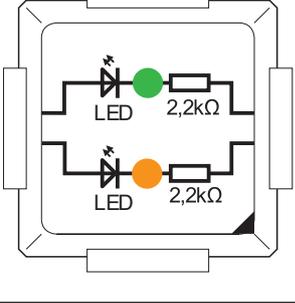
La misión del set lógico de Brick'R'knowledge es enseñar los fundamentos de la tecnología de circuitos digitales de una manera fácil y comprensible. Esto se hace, por un lado, mediante este detallado y didáctico manual y por otro lado experimentando con los bricks. Comenzamos con enlaces lógicos simples desde sumadores semi y completos, hasta circuitos flip-flop y un contador BCD de 4 dígitos. Como reloj utilizamos el conocido temporizador 555.

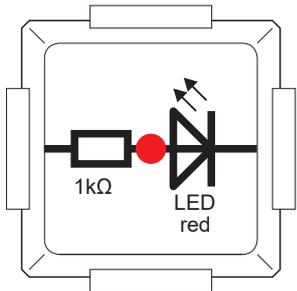
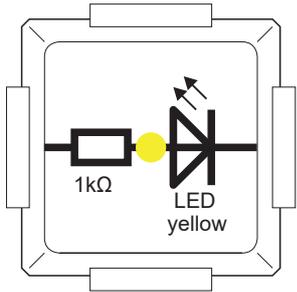
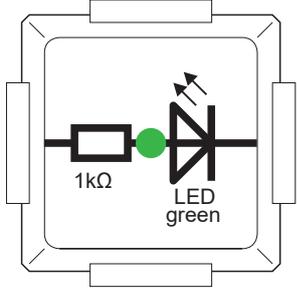
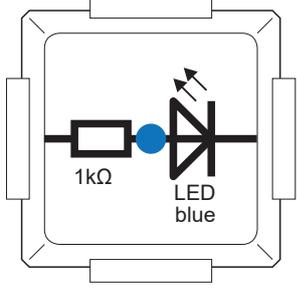
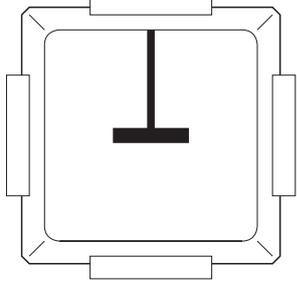
#### 3.1 Bricks del set lógico

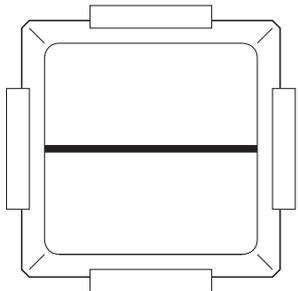
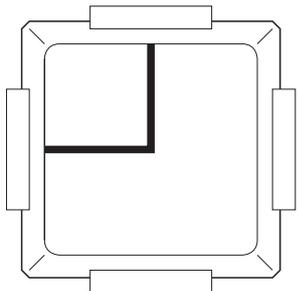
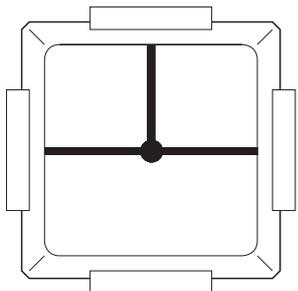
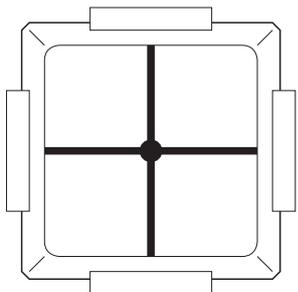
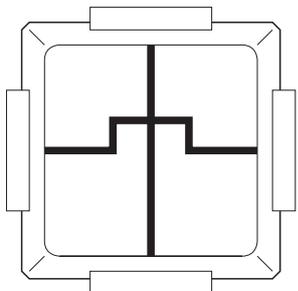
| Ilustración   | Ctd. | Art.Nº / ID del Brick                           | Breve descripción  |
|---|------|---|--|
|    | 1    | Art.Nº: 118627<br>ID del Brick: ALL-BRICK-0221  | <b>Fuente de alimentación de 9V</b><br>El adaptador de alimentación de 9V suministra una corriente máxima de 1A para los bricks. Está estabilizado y protegido contra cortocircuitos. Un LED indica cuando está en funcionamiento. El polo positivo se extrae y el negativo se conecta a tierra.<br>Utilice este brick como el último, después de haber comprobado el circuito de nuevo. |
|   | 2    | Art.Nº: 128276<br>ID del Brick: ALL-BRICK-0437  | <b>Puerta AND</b><br>La puerta realiza una conexión lógica AND con dos entradas, basada en 1/4 CD4011. Cada una de las entradas está conectada a tierra a través de una resistencia desplegable 10kΩ.<br>Tensión de alimentación: +5V...15V.   |
|  | 1    | Art.Nº: 128277<br>ID del Brick: ALL-BRICK-0438  | <b>Puerta OR</b><br>La puerta realiza una conexión lógica OR con dos entradas, basada en 1/4 CD4001. Las entradas se conectan a tierra a través de una resistencia desplegable de 10kΩ cada una.<br>Tensión de alimentación: +5..15V.  |
|  | 1    | Art.Nº: 135015<br>ID del Brick : ALL-BRICK-0634 | <b>Inversor</b><br>El inversor emite la señal de entrada negada en la salida. La entrada se conecta a tierra a través de una resistencia 10kΩ Pulldown.<br>Tensión de alimentación: +5...15V   |

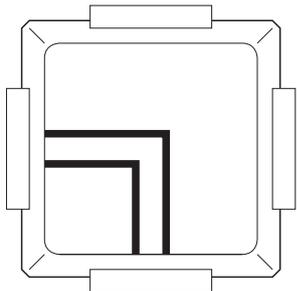
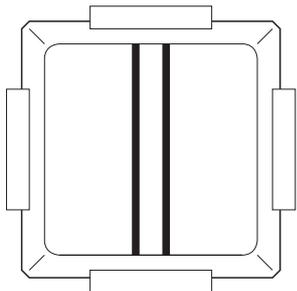
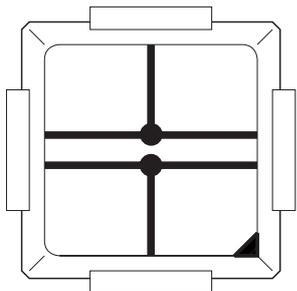
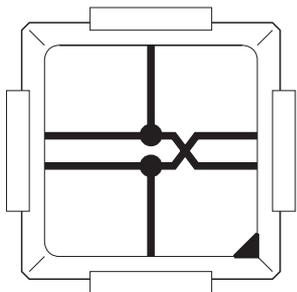
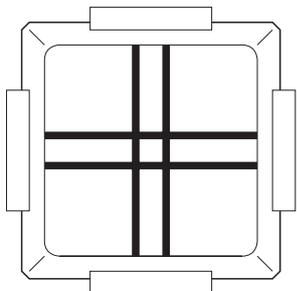
| Ilustración | Ctd. | Art.Nº. / ID del Brick                         | Breve descripción   |
|-------------|------|--|---|
|             | 1    | Art.Nº: 113684<br>ID del Brick: ALL-BRICK-0057 | <b>Puerta NAND</b><br>La puerta realiza una conexión lógica AND con dos entradas y salida negada, basada en 1/4 CD4011. Cada una de las entradas está conectada a tierra a través de una resistencia desplegable de 10 kΩ.<br>Tensión de alimentación: +5..15 V               |
|             | 2    | Art.Nº: 113685<br>ID del Brick: ALL-BRICK-0058 | <b>Puerta NOR</b><br>La puerta realiza una conexión lógica OR con dos entradas y salida negada, basada en 1/4 CD4001. Cada una de las entradas está conectada a tierra a través de una resistencia desplegable 10 kΩ.<br>Tensión de alimentación: +5V..15V                    |
|             | 2    | Art.Nº: 128278<br>ID del Brick: ALL-BRICK-0439 | <b>Puerta XOR</b><br>La puerta realiza una conexión OR exclusiva con dos entradas, basada en 1/4 CD4070. Cada una de las entradas está conectada a tierra a través de una resistencia desplegable 10 kΩ.<br>Tensión de alimentación: +5...15V                                 |
|             | 1    | Art.Nº: 128279<br>ID del Brick: ALL-BRICK-0440 | <b>Puerta XNOR</b><br>La puerta realiza una conexión lógica exclusiva OR con dos entradas y salida negada, basada en 1/4 CD4070. Cada una de las entradas está conectada a tierra a través de una resistencia 10kΩ Pull-down-Widerstand.<br>Tensión de alimentación: +5...15V |
|             | 4    | Art.Nº: 113683<br>ID del Brick: ALL-BRICK-0056 | <b>JK flip-flop</b><br>Flip-flop JK (controlado por un solo flanco) basado en 1/2 CD4027. Las entradas JK y el reloj están conectados a tierra mediante una resistencia de 10kΩ cada uno.   |

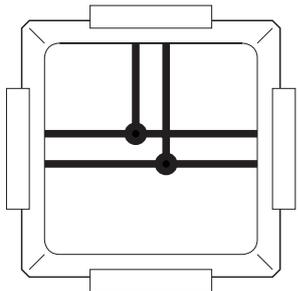
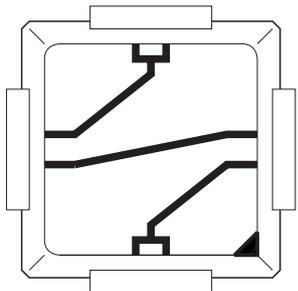
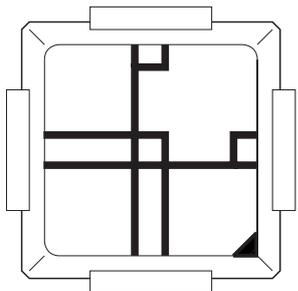
| Ilustración | Ctd. | Art.Nº. / ID del Brick                         | Breve descripción   |
|-------------|------|--|---|
|             | 4    | Art.Nº: 135012<br>ID del Brick: ALL-BRICK-0632 | <p><b>Flip-flop D</b><br/>           Flip-flop D (con control de flancos). La entrada de datos y el reloj están conectados a tierra a través de una resistencia pulldown 10kΩ.<br/>           Tensión de alimentación: +5...15V.</p> <p><b>Advertencia:</b> No se debe conectar ninguna carga adicional, como un LED, a la entrada C del reloj.</p>   |
|             | 1    | Art.Nº: 135014<br>ID del Brick: ALL-BRICK-0633 | <p><b>Flip-flop D con set y reset</b><br/>           D-flipflop (control de flancos) con entrada de set y reset en formato 2x1. Todas las entradas están conectadas a tierra a través de una resistencia pulldown 10kΩ.<br/>           Tensión de alimentación: +5...15V.</p> <p><b>Advertencia:</b> No se debe conectar ninguna carga adicional, como por ejemplo un LED, a la entrada de reloj C!</p>   |
|             | 2    | Art.Nº: 135011<br>ID del Brick: ALL-BRICK-0631 | <p><b>Contador BCD</b><br/>           El contador BCD se basa en el módulo contador síncrono CD4518. Esto facilita la creación BCD de dos dígitos. Las entradas de reloj y de rest están conectadas a tierra a través de una resistencia desplegable de 10 kΩ. Para la conexión en cascada de varios contadores BCD, está disponible la realización de baja actividad (Carry Out) del dígito de las decenas. La entrada del reloj cuenta con un flanco positivo. Tensión de alimentación: +9...15V.</p> |
|             | 1    | Art.Nº: 137826<br>ID del Brick: ALL-BRICK-0643 | <p><b>Reloj</b><br/>           El brick de reloj sirve como fuente de reloj para los bricks lógicos. El circuito se basa en el módulo temporizador universal 555. A través del potenciómetro se puede ajustar una señal de onda cuadrada de aprox. 0,5HZ a 100 Hz. Para generar impulsos individuales de onda cuadrada, el reloj se puede ajustar 1:1 por el brick "boton debounced" (ALL-BRICK-0641). Tensión de alimentación: +5...15V.</p>   |

| Ilustración   | Ctd. | Art.Nº / ID del Brick                          | Breve descripción  |
|---|------|--|--|
|    | 1    | Art.Nº: 137825<br>ID del Brick: ALL-BRICK-0641 | <p><b>Pulsador sin rebotes (Debounced pushbutton)</b></p> <p>Con este interruptor especial puede crear fácilmente un ciclo único sin rebotes para circuitos flip-flop. Alternativamente, también puede utilizar el brick del reloj. Usando el pulsador sin rebotes, hay suficiente tiempo para entender los pasos individuales de un circuito secuencial.<br/>Tensión de alimentación: +5...15V.</p>   |
|    | 3    | Art.Nº: 113644<br>ID del Brick: ALL-BRICK-0017 | <p><b>Botón pulsador</b></p> <p>El botón pulsador es un interruptor electromecánico que permite una conexión conductiva sólo durante la pulsación y mantenimiento del botón. En el momento en que se suelte, la conexión se abrirá de nuevo y el botón volverá a su posición inicial.<br/><b>Advertencia:</b> un pulsador es un componente electromecánico y no proporciona una señal de rebote. Esto puede dar lugar a impulsos falsos cuando se utiliza con circuitos lógicos (véase el capítulo 5.2, página 33)</p> |
|   | 1    | Art.Nº: 137824<br>ID del Brick: ALL-BRICK-0642 | <p><b>Botón pulsador doble</b></p> <p>Este brick contiene dos pulsadores. Esto permite que las entradas de puertas o flip-flops estén convenientemente conectadas. Además, las dos señales están separadas la una de la otra. Conecte la fuente de alimentación al conector superior y/o inferior y ahorre numerosos bricks de línea en muchas situaciones.</p>  |
|  | 1    | Art.Nº: 125693<br>ID del Brick: ALL-BRICK-0410 | <p><b>LED de doble conexión a tierra, rojo/amarillo y señal a través de la conexión</b></p> <p>Este brick contiene dos LED (rojo/amarillo) que están conectados internamente a tierra. Las líneas de señal se conectan por separado. Ambos LED están protegidos contra el flujo excesivo de corriente por una resistencia de serie 2,2 kΩ. Están diseñados para una corriente de 2mA. Ambas resistencias están conectadas a tierra para que el brick pueda conectarse directamente.</p>                                |
|  | 1    | Art.Nº: 125682<br>ID del Brick: ALL-BRICK-0409 | <p><b>LED de doble conexión a tierra, verde/naranja y señal a través de la conexión</b></p> <p>El brick contiene dos LED (verde/naranja). Ambos LED están protegidos contra el flujo excesivo de corriente por una resistencia de serie 2.2 kΩ. Están diseñados para una corriente de 2mA. Las resistencias están conectadas a tierra y se guían por separado. En general, debe asegurarse que el cátodo esté conectado a tierra y el ánodo al potencial positivo.</p>   |

| Ilustración   | Ctd. | Art.Nº ID del Brick                            | Breve descripción   |
|---|------|--|---|
|    | 1    | Art.Nº: 113636<br>ID del Brick: ALL-BRICK-0009 | <b>LED rojo</b><br>LED rojo con una resistencia de serie incluida de 1kΩ. El cátodo (polo negativo) está marcado con una línea en cruz en el símbolo del LED.   |
|    | 1    | Art.Nº: 113638<br>ID del Brick: ALL-BRICK-0011 | <b>LED amarillo</b><br>LED amarillo con una resistencia de serie incluida de 1kΩ. El cátodo (polo negativo) está marcado con una línea en cruz en el símbolo del LED.   |
|   | 1    | Art.Nº: 113639<br>ID del Brick: ALL-BRICK-0012 | <b>LED verde</b><br>LED verde con una resistencia de serie incluida de 1kΩ. El cátodo (polo negativo) está marcado con una línea en cruz en el símbolo del LED.   |
|  | 1    | Art.Nº: 113637<br>ID del Brick: ALL-BRICK-0010 | <b>LED azul</b><br>LED azul con una resistencia de serie incluida de 1kΩ. El cátodo (polo negativo) está marcado con una línea en cruz en el símbolo del LED.   |
|  | 4    | Art.Nº: 113630<br>ID del Brick: ALL-BRICK-0003 | <b>Brick de tierra</b><br>Con el brick de tierra se puede conectar fácilmente la tierra a cualquier punto del circuito. Ponga el brick en el extremo de cada circuito para cerrarlo. El brick de tierra conecta los dos contactos centrales de la conexión con las dos líneas exteriores de tierra. |

| Ilustración   | Ctd. | Art.Nº. / ID del Brick                         | Breve descripción  |
|---|------|--|--|
|    | 10   | Art.Nº: 113631<br>ID del Brick: ALL-BRICK-0004 | <b>Brick de conector recto</b><br>La línea recta conecta dos bricks opuestos.  |
|    | 8    | Art.Nº: 113632<br>ID del Brick: ALL-BRICK-0005 | <b>Brick de esquina</b><br>Une dos partes adyacentes.  |
|   | 8    | Art.Nº: 113633<br>ID del Brick: ALL-BRICK-0006 | <b>Brick cruce en T</b><br>Con este brick puedes conectar los componentes de tu circuito como una "T".   |
|  | 3    | Art.Nº: 113634<br>ID del Brick: ALL-BRICK-0007 | <b>Brick cruzado</b><br>Conecta las cuatro direcciones entre sí. En la esquina del circuito, el brick cruzado se puede usar en línea o como un brick de esquina. |
|  | 7    | Art.Nº: 113635<br>ID del Brick: ALL-BRICK-0008 | <b>Brick de doble conector recto</b><br>Conecta sólo los lados opuestos, de abajo a arriba y de izquierda a derecha.   |

| Ilustración   | Ctd. | Art.Nº. / ID del Brick                         | Breve descripción  |
|---|------|--|--|
|    | 1    | Art.Nº: 113681<br>ID del Brick: ALL-BRICK-0054 | <b>Cable de doble esquina</b><br>Este es un brick de esquina para dos señales separadas.                                       |
|    | 1    | Art.Nº: 113676<br>ID del Brick: ALL-BRICK-0049 | <b>Cable doble recto</b><br>Brick conector recto para dos señales separadas.   |
|   | 2    | Art.Nº: 122443<br>ID del Brick: ALL-BRICK-0381 | <b>Cable cruce en T doble izquierda/derecha</b><br>Se utiliza para separar o conectar señales.                                 |
|  | 1    | Art.Nº: 113675<br>ID del Brick: ALL-BRICK-0048 | <b>Cable doble cruzado</b><br>Con este brick se pueden reenviar las líneas centrales por separado y cruzarlas al mismo tiempo. |
|  | 1    | Art.Nº: 113678<br>ID del Brick: ALL-BRICK-0051 | <b>Cable doble cruzado no conectado</b><br>Este brick permite cruzar señales separadas sin conectarlas.                        |

| Ilustración  | Ctd. | Art.Nº ID del Brick                            | Breve descripción  |
|--|------|--|--|
|   | 4    | Art.Nº: 113679<br>Brick-ID: ALL-BRICK-0052     | <b>Cable doble de cruce en T a la izquierda</b><br>Brick de cruce en T por la izquierda para señales separadas. También ofrecemos un cable doble de cruce en T a la derecha.   |
|   | 4    | Art.Nº: 122442<br>ID del Brick: ALL-BRICK-0380 | <b>Cable doble duplex izquierda/derecha</b><br>Duplex izquierda/derecha para separar señales.  |
|  | 5    | Art.Nº: 113677<br>ID del Brick: ALL-BRICK-0050 | <b>Cable doble especial</b><br>Brick de cable especial para distribuir señales separadas. Este brick se suele utilizar en los circuitos lógicos y en los circuitos con controladores, por ejemplo, el brick Arduino. |

## 4. Fundamentos de la tecnología de circuitos digitales

### 4.1 ¿Qué significa exactamente digital?

El término "digital" viene del latín y significa dedo. Los niños pequeños aprenden a contar y calcular usando los dedos. Sirven como una unidad aritmética elemental. Nos limitamos a la tecnología digital binaria, que sólo conoce dos posibles estados de señal:

- Cero lógico "0" o "L" (inglés low), y
- Lógico Uno "1" o "H" (Inglés high)

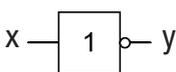
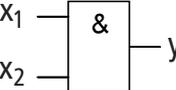
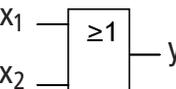
En contraste, el término "analógico" (por ejemplo, cámara analógica frente a cámara digital). La diferencia puede explicarse comparando una pendiente y una escalera. Mientras que una pendiente tiene un cambio continuo de altura, una escalera tiene pequeños escalones discretos. Del mismo modo que el procesamiento digital de la información sólo distingue entre el cero lógico "0" y un "1", la tecnología analógica procesa señales de forma continua. Los principales componentes de los circuitos digitales son puertas lógicas como NOT, AND y OR. Con estas funciones lógicas se pueden construir todas las demás puertas, numeradores, flip-flops, etc. Los circuitos más complejos son circuitos de memoria y procesadores, así como dispositivos lógicos libremente programables.

Si un nivel high de tensión está representado por "1" y un nivel low por "0", hablamos de lógica positiva, si un nivel high de tensión está representado por "0" y un nivel low por "1", hablamos de lógica negativa.

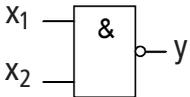
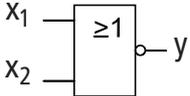
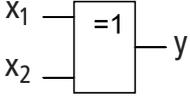
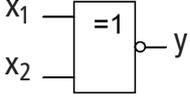
### 4.2 Funciones lógicas

Las funciones lógicas (también llamadas funciones booleanas) se denominan puertas lógicas o simplemente puertas a nivel de hardware. Existen tres funciones lógicas básicas NOT, AND y OR a partir de las cuales se pueden combinar las funciones NAND, NOR, XOR y XNOR. Cada elemento lógico puede describirse mediante una función de conmutación. Las formas comunes de representación son la ecuación booleana y la tabla de verdad. Si un circuito digital contiene sólo elementos lógicos sin realimentación de salida y entrada, se denomina lógica puramente combinatoria o también red de conmutación (véase también el capítulo 4.5 en la página 21). Además de las funciones lógicas, los circuitos digitales también pueden contener elementos de almacenamiento, como flip-flop, que funcionan de forma controlada por reloj o por estado. En cuanto se retroalimenta al menos una salida a una entrada, se denomina circuito secuencial (véase también 4.6 en la página 21).

#### Funciones básicas

| Símbolo de circuito   | Designación      | Ecuación             | Tabla de verdad   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|------------------|----------------------|---|----------------|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | NOT (Negación)   | $y = \bar{x}$        | <table border="1"><thead><tr><th>x</th><th>y</th></tr></thead><tbody><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></tbody></table>   | x              | y              | 0 | 1 | 1 | 0 |   |   |   |   |   |   |   |   |   |
| x   | y                |                      |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1                |                      |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0                |                      |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
|  | AND (Conjunción) | $y = x_1 \wedge x_2$ | <table border="1"><thead><tr><th>x<sub>1</sub></th><th>x<sub>2</sub></th><th>y</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></tbody></table> | x <sub>1</sub> | x <sub>2</sub> | y | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| x <sub>1</sub>  | x <sub>2</sub>   | y                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0                | 0                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1                | 0                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0                | 0                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1                | 1                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
|  | OR (Disyunción)  | $y = x_1 \vee x_2$   | <table border="1"><thead><tr><th>x<sub>1</sub></th><th>x<sub>2</sub></th><th>y</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></tbody></table> | x <sub>1</sub> | x <sub>2</sub> | y | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| x <sub>1</sub>  | x <sub>2</sub>   | y                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0                | 0                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1                | 1                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0                | 1                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1                | 1                    |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |

## Funciones compuestas

| Símbolo de circuito  | Designación            | Ecuación   | Tabla de verdad  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|--|------------------------|--|--|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
|   | NAND (Exclusión)       | $y = \overline{x_1 \wedge x_2}$                                    | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| $x_1$  | $x_2$                  | $y$  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 0                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 1                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 0                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 1                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|   | NOR (Nihilación)       | $y = \overline{x_1 \vee x_2}$                                      | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| $x_1$  | $x_2$                  | $y$  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 0                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 1                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 0                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 1                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|   | XOR (Anticoincidencia) | $y = (\overline{x_1} \wedge x_2) \vee (x_1 \wedge \overline{x_2})$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| $x_1$  | $x_2$                  | $y$  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 0                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 1                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 0                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 1                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|  | XNOR (Equivalencia)    | $y = (x_1 \wedge x_2) \vee (\overline{x_1} \wedge \overline{x_2})$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| $x_1$  | $x_2$                  | $y$  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 0                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0  | 1                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 0                      | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1  | 1                      | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |

La visualización de los símbolos de circuito cumple con IEC 60617-12.

### 4.3 Comparación de sistemas numéricos

| Valor de los dígitos | Número binario     |                    |                    |                    | Número decimal       |                     | Número Hexadecimal* |
|----------------------|--------------------|--------------------|--------------------|--------------------|----------------------|---------------------|---------------------|
|                      | 8(2 <sup>3</sup> ) | 4(2 <sup>2</sup> ) | 2(2 <sup>1</sup> ) | 1(2 <sup>0</sup> ) | 10(10 <sup>1</sup> ) | 1(10 <sup>0</sup> ) | 1(16 <sup>0</sup> ) |
| Cero                 | 0                  | 0                  | 0                  | 0                  | 0                    | 0                   | 0                   |
| Uno                  | 0                  | 0                  | 0                  | 1                  | 0                    | 1                   | 1                   |
| Dos                  | 0                  | 0                  | 1                  | 0                  | 0                    | 2                   | 2                   |
| Tres                 | 0                  | 0                  | 1                  | 1                  | 0                    | 3                   | 3                   |
| Cuatro               | 0                  | 1                  | 0                  | 0                  | 0                    | 4                   | 4                   |
| Cinco                | 0                  | 1                  | 0                  | 1                  | 0                    | 5                   | 5                   |
| Seis                 | 0                  | 1                  | 1                  | 0                  | 0                    | 6                   | 6                   |
| Siete                | 0                  | 1                  | 1                  | 1                  | 0                    | 7                   | 7                   |
| Ocho                 | 1                  | 0                  | 0                  | 0                  | 0                    | 8                   | 8                   |
| Nueve                | 1                  | 0                  | 0                  | 1                  | 0                    | 9                   | 9                   |
| Diez                 | 1                  | 0                  | 1                  | 0                  | 1                    | 0                   | A                   |
| Once                 | 1                  | 0                  | 1                  | 1                  | 1                    | 1                   | B                   |
| Doce                 | 1                  | 1                  | 0                  | 0                  | 1                    | 2                   | C                   |
| Trece                | 1                  | 1                  | 0                  | 1                  | 1                    | 3                   | D                   |
| Catorce              | 1                  | 1                  | 1                  | 0                  | 1                    | 4                   | E                   |
| Quince               | 1                  | 1                  | 1                  | 1                  | 1                    | 5                   | F                   |

El respectivo sistema de numeración se identifica por su base. Esto se hace subscribiendo el conjunto base entre paréntesis, por ejemplo  $1100(2) = 12(10) = C(16)$ .

Además del sistema binario con la base 2, el sistema hexadecimal\* con la base 16 se utiliza a menudo en la tecnología digital. Esto se debe a que los números binarios son relativamente largos y difíciles de entender. Dado que 16 es una potencia de 2, es particularmente fácil convertir números binarios en números hexadecimales. Para ello, se sustituyen cuatro dígitos del número binario por un dígito hexadecimal, lo que aumenta considerablemente la claridad. Los dígitos hexadecimales con el valor 0-9 están separados por los símbolos numéricos 0-9 y los valores 10-15 están representados por las letras mayúsculas A-F. Esto los hace relativamente fáciles de leer. Los dígitos hexadecimales con el valor 0-9 se representan con los símbolos numéricos 0-9. Los valores 10-15 están representados por las letras mayúsculas A-F - por lo que son claramente legibles.

Por ejemplo, es fácil determinar que ADF5 (16) es mayor que ABF1(16), mientras que los números binarios correspondientes  $1010110111111110101(2)$  y  $1010101111110001(2)$  no son tan transparentes.

#### Conversión de sistema binario a sistema decimal

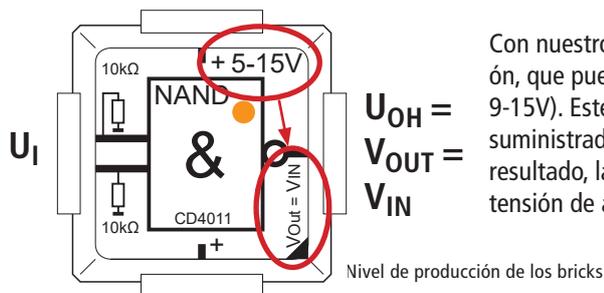
Para convertir un número binario en el correspondiente número decimal, todos los dígitos se multiplican por su valor (potencia correspondiente de dos) y luego se suman.

\*La combinación de palabras latino-griega "hexadecimal" se deriva del griego hexa "seis" y del latín decem "diez".  
Consulte también: <https://de.wikipedia.org/wiki/Hexadezimalsystem>

## 4.4 Nivel lógico

Para poder procesar y visualizar señales digitales, se definen niveles de tensión que corresponden a los estados lógicos. Con respecto a las señales codificadas binarias, dos rangos de voltaje son suficientes para representar la información. Estos se llaman nivel High (o nivel H, high o H) y nivel Low (nivel L, Low o L).

El nivel High, es decir, la tensión más alta, corresponde normalmente a la tensión de funcionamiento (p. ej. 9 voltios para los bricks); el nivel Low, es decir, la tensión más baja, es de 0 V (normalmente llamada a tierra). La transición más o menos pronunciada de nivel Low a nivel High también se llama flanco ascendente o positivo y la transición de nivel High a Low se llama flanco descendente o negativo. El cambio entre los dos niveles debe hacerse con una inclinación mínima de la pendiente, que normalmente se define en la hoja de datos del fabricante.



Con nuestros bricks lógicos, el nivel de salida depende de la tensión de alimentación, que puede variar entre +5V y +15V (excepto el brick contararrestante BCD: 9-15V). Este valor es de 9V cuando se utiliza el adaptador de alimentación suministrado o cuando se alimenta con una batería de bloque de 9V. Como resultado, la salida de nivel High  $U_{OH}$  es aproximadamente el mismo que el de la tensión de alimentación (como es habitual en los componentes CMOS).

### 4.4.1 Rangos de tensión

Los rangos de tensión están definidos por las diferentes "Familias Lógicas". Las familias lógicas son circuitos electrónicos (CI) que tienen idénticos niveles lógicos, comportamientos temporales, propiedades de los controladores, procesos de fabricación, etc.

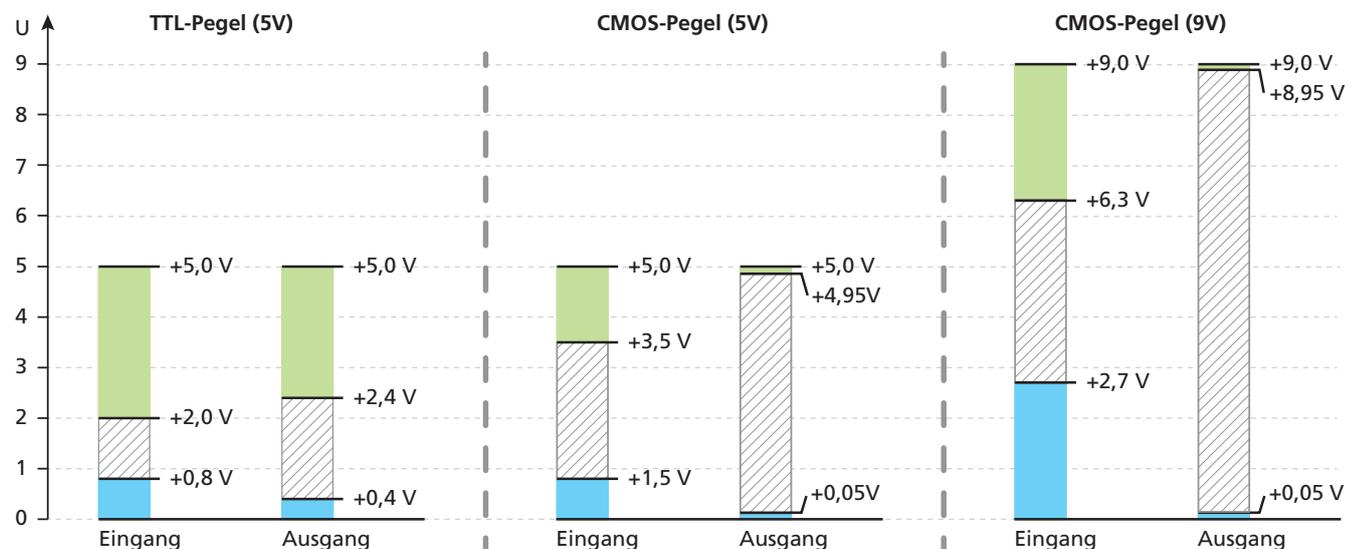


Figura 5: Nivel lógico para tecnología TTL y CMOS

Los dos valores lógicos están representados por rangos de nivel relativamente grandes (azul y verde en el gráfico). Esto tiene sentido para que los circuitos lógicos reales puedan detectar de forma fiable los estados a pesar de las tolerancias. El rango entre los dos rangos de nivel de entrada de una puerta lógica, es decir, entre  $U_{IL}$  y  $U_{IH}$ , no está permitido (rango prohibido), la tensión de la señal no está claramente asignada a un valor lógico (sombreado en el gráfico). Esta es la razón por la que hay un voltaje mínimo de salida  $U_{OH}$  garantizado para un rango de nivel High en la salida y hay un voltaje mínimo de entrada  $U_{IH}$  requerido en la entrada. Si la tensión de salida  $U_{OH}$  es siempre superior que la tensión de entrada  $U_{IH}$ , la diferencia  $U_{OH} - U_{IH}$  se denomina distancia de interferencia estática y garantiza la seguridad de funcionamiento de los circuitos. A un nivel Low hay una tensión máxima de salida  $U_{OL}$ , la tensión máxima de entrada  $U_{IL}$  y la distancia de interferencia estática  $U_{IL} - U_{OL}$ .

## 4.4.2 Estados lógicos

La información de nivel L y H no debe confundirse con los estados lógicos 0 y 1. Los valores L y H indican siempre el nivel de tensión real. Por ejemplo, 0 V (Low) o 5 V (High). Con estos niveles nos referimos a los niveles reales de tensión eléctrica de un circuito. Si se desea describir el funcionamiento lógico de un circuito, se deben asignar las especificaciones de nivel a los estados lógicos. En este contexto, se distingue entre lógica positiva y negativa.

### Lógica positiva

Cuando se utiliza la lógica positiva, la lógica 0 corresponde al nivel Low y la lógica 1 al nivel High.

### Lógica negativa

Cuando se utiliza la lógica negativa, la lógica 0 corresponde al nivel High y el 1 lógico al nivel Low. Las designaciones de las señales de baja actividad suelen estar marcadas con una ralla. Alternativamente, se colocan asteriscos o barras oblicuas delante o detrás. Las anotaciones RESET, \*RESET y /RESET deben indicar que la señal RESET es de baja actividad.

## 4.5 Circuitos combinatorios

El término red de conmutación se refiere a un circuito combinatorio que consiste en puertas básicas simples (por ejemplo, AND, OR, NOT). Una o más variables de salida dependen de una o más variables de entrada en cualquier momento. No hay realimentación, es decir, la salida de una puerta no se retroalimenta a la entrada de la puerta. También se podría decir que un circuito así es un proceso olvidadizo. Los tiempos de funcionamiento de las señales no se tienen en cuenta en el nivel lógico (véase también el capítulo "Fallos" en la página 22).

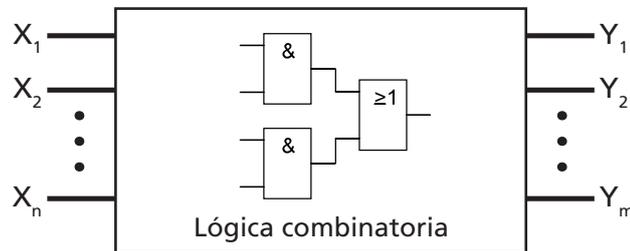


Figura 6: Circuito combinatorio

## 4.6 Circuito secuencial

Para resolver tareas que no sólo dependen de una instantánea, se requieren circuitos con "memoria". Esto significa que se necesita un circuito que enlaza las variables de entrada en un momento determinado con los valores creados antes de ese momento. Por la retroalimentación de las salidas a las entradas, el circuito obtiene un carácter de almacenamiento. Obtenemos los llamados circuitos secuenciales de estado finito.

Este circuito consiste en un bloque de memoria (flip-flop) y un bloque con lógica combinatoria. A partir del estado actual del bloque de memoria y de las variables de entrada  $X$ , la parte del circuito combinatorio forma el estado posterior deseado  $Z_{t+1}$  y las variables de salida  $Y_1$ . Esta parte también se denomina autómata de Mealy. Mientras que las variables de salida  $Y_1$  pueden cambiar dentro de un mismo estado, las salidas  $Y_2$  están sincronizadas y, por lo tanto, libres de posibles impulsos falsos (también llamadas autómatas Moore).

Un circuito secuencial es síncrono cuando las entradas y la realimentación se sincronizan mediante señales de reloj (imagen derecha), en caso contrario son "asíncronas" (imagen izquierda).

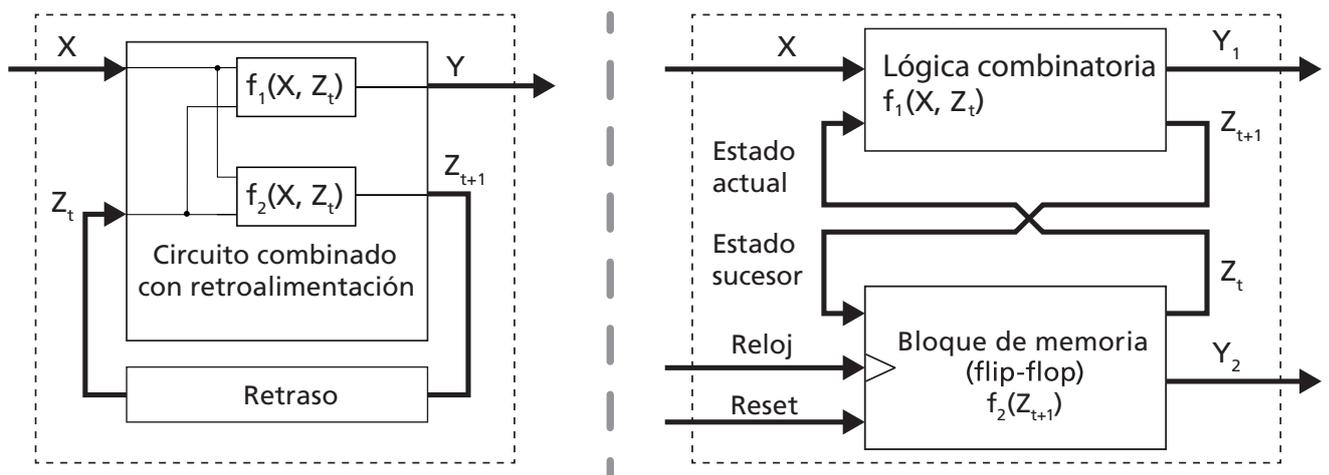


Figura 7: Circuito secuencial asíncrono (izquierda) y circuito secuencial síncrono (derecha)

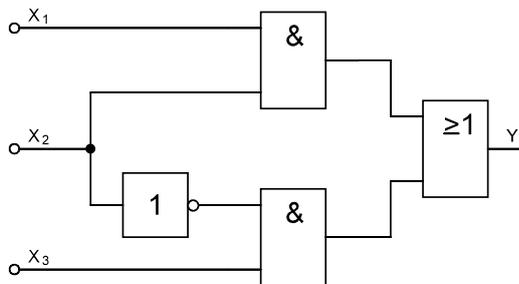
En realidad, cada red de conmutación sólo genera su resultado después de un cierto tiempo, que depende de su estructura. Para vincular los valores de entrada presentes en un momento determinado con los valores correctos almacenados entre ellos, las señales de entrada y las salidas de realimentación se sincronizan con la ayuda de flip-flops y una señal de reloj.

La frecuencia de la señal de reloj debe ser lo suficientemente grande para que todos los circuitos de combinación en el circuito secuencial terminen sus cálculos, es decir, para que transcurran todos los retardos de propagación involucrados.

## 4.7 Fallos de funcionamiento

En el campo de la electrónica, los "glitches" son una afirmación falsa de corta duración en los circuitos lógicos y la falsificación temporal de una función booleana. Esto ocurre porque los tiempos de propagación de la señal de las puertas individuales nunca son completamente idénticos en la realidad. La susceptibilidad a los fallos aumenta con la complejidad, las velocidades de reloj más altas y la miniaturización de los circuitos, pero también puede ocurrir con circuitos muy simples. Representan un problema importante en el desarrollo de circuitos electrónicos modernos y microprocesadores rápidos.

### Ejemplo de desarrollo de fallos:

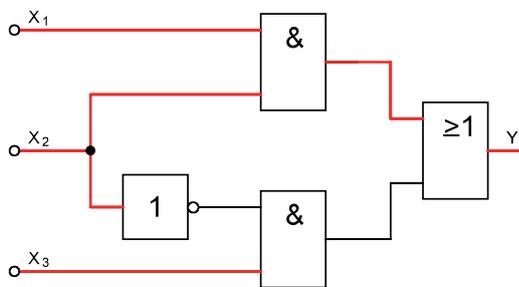


#### El circuito

Hay un circuito con tres entradas:  $x_1$ ,  $x_2$  y  $x_3$ . La salida  $Y$  debe suministrar el valor "1" si se cumple al menos una de las dos condiciones:

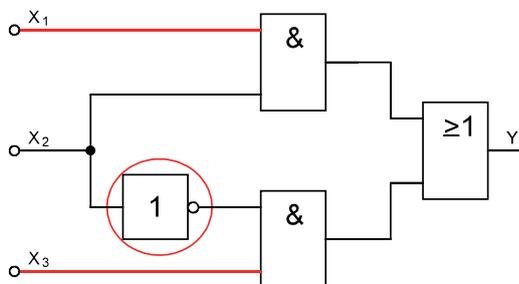
- $x_2$  y  $x_1$  son simultáneamente "1" O
- $x_2$  es igual a "0" y  $x_3$  es igual a "1"

Si no se aplica al menos una de las dos condiciones no se aplica,  $Y$  debe dar el valor "0".



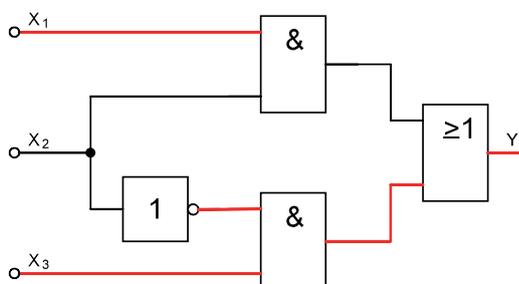
#### Estado 1 - El circuito entrega un 1

Según nuestras especificaciones, se cumple la primera condición, es decir,  $x_2$  y  $x_1$  son "1". Las secciones con la información "1" se muestran en rojo. El inversor convierte el "1" entrante en "0". Por lo tanto, la operación  $Y$  subsiguiente no transmite una señal, es decir, emite un "0". Sin embargo, todo el circuito (operación OR) devuelve un "1", ya que la otra función AND devuelve el "1".



#### Estado 2 - El inversor causa un fallo

En el estado 2,  $x_2 = 0$  y  $x_1 = 1$ . El circuito debe seguir emitiendo un "1". Sin embargo, el inversor necesita cierto tiempo para percibir la conversión de la señal  $x_2$  de "0" a "1". Por poco tiempo, ambos  $x_2 = 0$  y  $\bar{x}_2 = 0$ . Esta circunstancia se procesa como si no se cumpliera ninguna de las condiciones y por lo tanto produce un "0". A esta situación se le llama fallo.



#### Estado 3 - El circuito devuelve el valor correcto

Después de algún tiempo - en el orden de los nanosegundos - el circuito está en estado 3: el inversor ha procesado la nueva información. La salida "1" pasa ahora a la puerta AND, que a su vez también suministra un "1" (de nuevo después de un breve retardo). Ahora la operación OR en  $Y$  emite el valor "1".

Figura 8: Origen de los fallos

En la práctica, también existen diferencias en el tiempo de propagación en puertas del mismo tipo o en líneas de diferentes longitudes. Si desea conocer el valor exacto de la función, debe esperar el tiempo correspondiente hasta que todas las señales se hayan estabilizado. Este hecho limita significativamente la frecuencia de reloj de los procesadores modernos.

El impacto de los fallos en el diseño de los circuitos síncronos se puede prevenir mediante flip-flops D de corriente descendente. La idea es que las salidas de las partes del circuito combinatorio, que consisten en varias puertas lógicas con diferentes tiempos de propagación, sólo deben asumir estados válidos cuando el flanco del reloj toma los valores de salida en los flip-flops D. En el tiempo entre dos flancos de reloj puede ocurrir cualquier número de fallos en la parte combinatoria debido a efectos de tiempo de ejecución, ya que estos estados intermedios no son tomados en cuenta por la corriente descendente del flip-flop D.

El método de proporcionar siempre las salidas de los componentes de circuitos combinatorios con flip-flops D es uno de los principios esenciales para diseños de circuitos digitales estables en FPGAs complejas y libremente programables (véase el capítulo 4.8.3 en la página 23).

## 4.8 Dispositivos lógicos programables

Los bricks lógicos convencionales, no programables, tienen una función fija. Se pueden comprar como circuitos integrados (CI), y también están incorporados en los bricks lógicos. Una de las puertas lógicas más conocidas es la puerta NAND con denominación de tipo 7400 (tecnología TTL) o 4011 (tecnología CMOS). En cambio, los PLD (Programmable Logic Devices), es decir, los dispositivos lógicos programables, sólo obtienen su función después de haber sido fabricados por la programación apropiada del diseñador del circuito (también llamada personalización). Algunos fabricantes de chips proporcionan herramientas de software libre para el diseño, la implementación (lugar y ruta) y la simulación de circuitos.

Aquellos que quieran profundizar en este tema, especialmente en lo que respecta a los dispositivos FPGA, encontrarán un resumen (cada vez más complejo) a continuación:

### 4.8.1 Matriz Lógica Programable (PAL) y Matriz Lógica Genérica (GAL)

Un PAL es un sistema AND programable con un sistema OR fijo. A diferencia de un PAL, un GAL es reescribible. La personalización se realiza con la ayuda de un dispositivo de programación por parte del diseñador de circuitos. El sucesor tecnológico es la CPLD (véase el capítulo siguiente).

### 4.8.2 Dispositivo Lógico Programable Complejo (CPLD)

Un CPLD consiste en bloques que pueden ser interconectados. El elemento principal es la conocida como lógica programable (PLA), que consta de una matriz AND y una matriz OR, siendo ambas matrices programables. También hay bloques de entrada y salida y una realimentación programable. Normalmente se incluye un flip-flop para cada pin de E/S. Esta matriz AND/OR permite cualquier enlace de combinación. Debido a la estructura, los tiempos de producción de la CPLD pueden definirse con precisión, a diferencia de los de la FPGA. Otra diferencia con respecto a la FPGA es la programación permanente por parte del desarrollador de circuitos, de modo que las funciones de la FPGA pueden programarse de forma permanente. No es necesario actualizar la CPLD cada vez que se enciende el dispositivo.

### 4.8.3 Matriz de puertas programables por campo (FPGA)

Al igual que una CPLD, una FPGA consiste en bloques interconectados, pero son más complejos. Las posibilidades de interconexión de estos bloques también se amplían considerablemente en comparación con la CPLD. Mediante la configuración específica de los elementos internos, se pueden realizar muchos circuitos y funciones completamente diferentes en una FPGA. Estos van desde circuitos de baja complejidad, como un decodificador o un simple contador síncrono, hasta circuitos altamente complejos como controladores de memoria o procesamiento digital de señales. Las FPGA modernas a menudo también contienen bloques de función integrados como RAM, PLL o núcleos de CPU enteros.

Las FPGA se utilizan en todos los ámbitos de la tecnología digital, pero sobre todo en aquellos en los que se necesitan FPGA rápidas y fiables. Esto permite cambios o mejoras posteriores en los circuitos implementados sin tener que realizar costosos y lentos cambios de hardware. Esta alta flexibilidad es ideal para prototipos y para la producción rentable de series pequeñas y medianas.

La personalización tiene lugar "en el campo" por el usuario, ya sea leyendo los datos de configuración desde una memoria externa no volátil como una PROM (Memoria de Sólo Lectura Programable) - en este caso la FPGA es el "maestro" - o descargando los datos desde el ordenador a la FPGA (modo esclavo). Después de conectar la tensión de alimentación, la FPGA es inicialmente "estúpida". Sólo después de la fase de configuración de unos pocos milisegundos el aparato está listo para funcionar. Las FPGA pueden ser reconfiguradas en el circuito en cualquier momento sin necesidad de un dispositivo de programación.

### 4.8.3.1 Estructura básica de la FPGA

La estructura básica de la FPGA es una disposición matricial de bloques lógicos configurables (CLB) y de bloques de entrada/salida (IOB). Estos últimos suelen estar conectados a los pines del chip. Consulte la Figura 9 en la página 24.

Entre los bloques (CLB y IOB) hay una red de vías de conexión, también conocida como área de interconexión, que consiste en una jerarquía de "líneas" horizontales y verticales. Las entradas y salidas de los bloques se conectan a estas líneas. Las conexiones remotas se "cablean" a través de puntos de conexión programables en la matriz de interruptores de la red. De esta manera, el enrutamiento de la señal (también llamado enrutamiento) es posible en todo el chip.

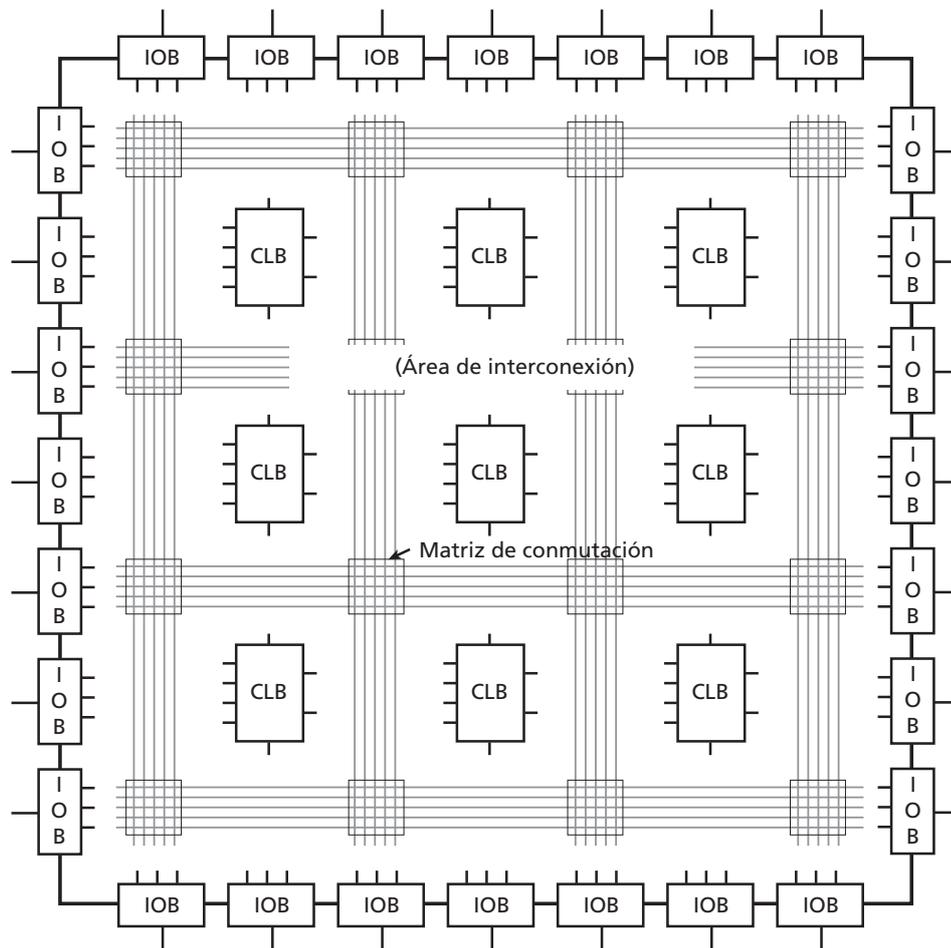


Figura 9: Estructura básica de la FPGA

### 4.8.3.2 Bloques lógicos configurables (CLB)

La estructura básica de una FPGA es una matriz de bloques configurables con una tabla de verdad programable (LUT) y un registro de 1 bit (Flip-flop D). Dependiendo del número de entradas disponibles, el LUT puede implementar cualquier función binaria de n dígitos. Las estructuras LUT con cuatro entradas binarias y más son comunes. Además de los LUT, los multiplexores en los bloques base permiten rutas de señal locales muy rápidas, por ejemplo para integrar o desviar el flip-flop, para la retroalimentación de su salida, para conectar bloques vecinos y similares.

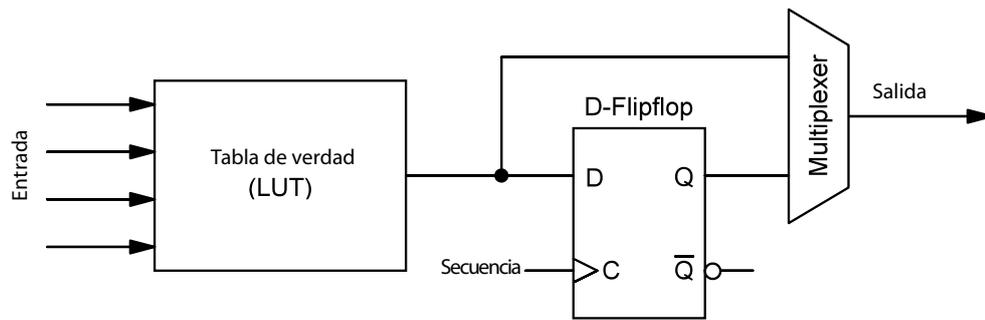


Figura 10: Ejemplo de bloque lógico

#### 4.8.3.3 Bloque de Entrada/Salida (IOB)

Los bloques de Entrada/Salida (IOB) forman la interfaz con el mundo exterior. Se utilizan para conectar los conectores de la FPGA a una matriz de conmutación. Dependiendo de la aplicación, el nivel de voltaje de las entradas y salidas se puede adaptar al estándar de interfaz correspondiente (TTL, LVDS, etc...). Además, se puede definir la corriente de accionamiento de las salidas y la pendiente de las señales de salida, así como se pueden activar los buffers de tristate (pines que se pueden conmutar con alta impedancia) para configurar una arquitectura de bus.

## 5. Ejemplos de circuitos

### 5.1 Circuitos básicos

#### 5.1.1 Inversor

El brick del inversor representa la función NOT y también se denomina negación o inversión. Esto significa que podemos invertir la señal de entrada. Si se aplica un nivel High a la entrada del brick del inversor, se obtiene un nivel Low en la salida y viceversa.

En nuestro circuito de bricks, se conecta un botón a la entrada del brick inversor y se conecta a la tensión de alimentación. Para que el brick del inversor sepa lo que hace cuando el botón está abierto (estado de reposo), tiene una resistencia desplegable integrada, la cual se puede ajustar en cualquier momento. Esto tiene el efecto de que la entrada está inicialmente en un nivel Low. Cualquier brick LED se conecta a la salida para indicar el cambio de estado.

Presionando el botón, el nivel High se aplica a la entrada, la salida pasa a nivel Low y el LED se apaga.

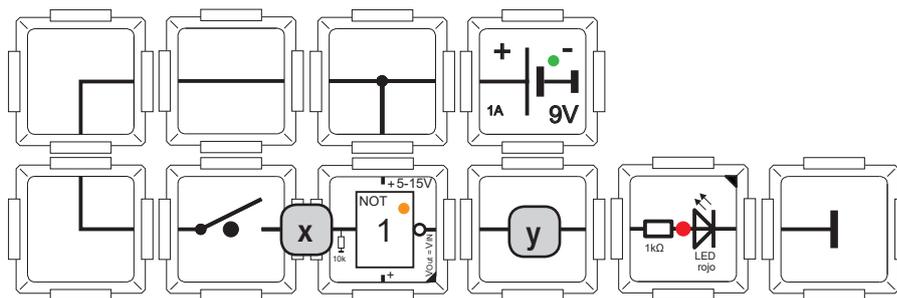


Figura 11: Inversor como brick de circuito

| Símbolo de circuito | Descripción    | Ecuación      | Tabla de verdad  |   |   |   |   |   |   |
|---------------------|----------------|---------------|--|---|---|---|---|---|---|
|                     | NOT (Negación) | $y = \bar{x}$ | <table border="1"> <thead> <tr> <th>x</th> <th>y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table> | x | y | 0 | 1 | 1 | 0 |
| x                   | y              |               |  |   |   |   |   |   |   |
| 0                   | 1              |               |  |   |   |   |   |   |   |
| 1                   | 0              |               |  |   |   |   |   |   |   |

## 5.1.2 Puerta AND

La puerta AND realiza una conexión AND lógica de dos o más entradas. En comparación con la puerta NAND, la salida no se invierte aquí. En nuestro ejemplo, las dos entradas de la puerta AND se conectan a la tensión de alimentación mediante pulsadores. Mientras no se pulse ningún botón, las dos entradas se conectan a tierra (nivel Low) a través de las resistencias desplegables integradas, de forma que hay un nivel definido en las entradas. El LED no se enciende hasta que se pulsamos ambos botones y la salida cambia a nivel High, como se puede ver en la tabla de verdad.

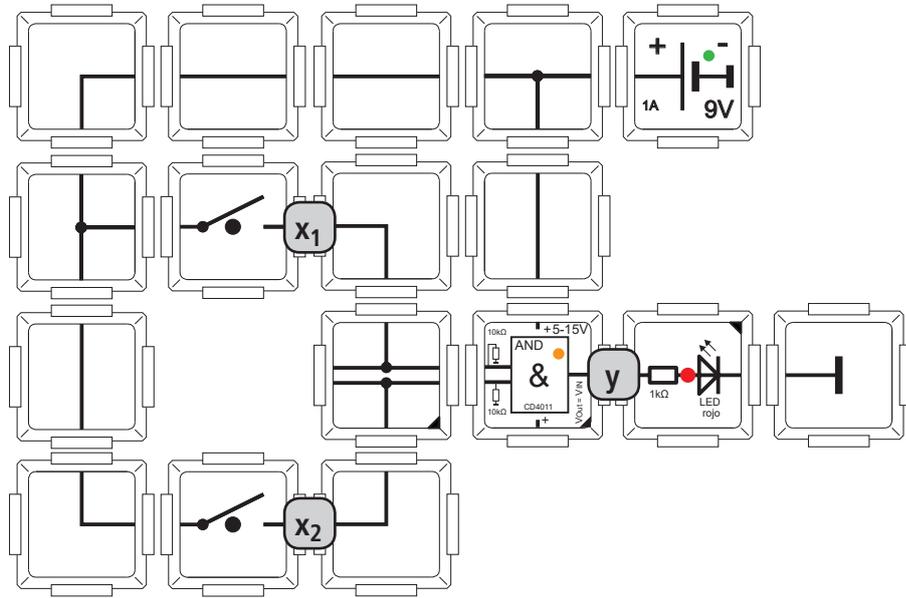


Figura 12: puerta AND como circuito Brick

| Símbolo de circuito | Descripción  | Ecuación             | Tabla de verdad  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|---------------------|--------------|----------------------|--|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
|                     | (Conjunción) | $y = x_1 \wedge x_2$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| $x_1$               | $x_2$        | $y$                  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                   | 0            | 0                    |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                   | 1            | 0                    |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                   | 0            | 0                    |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                   | 1            | 1                    |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |

Un circuito AND también se puede construir simplemente a partir de dos pulsadores que se conectan uno tras otro en el circuito, es decir, en serie. Sólo cuando se pulsamos los dos botones ( $x_1$  y  $x_2$ ), se enciende el LED ( $y$ ). Pruebe la función utilizando la tabla de verdad anterior.

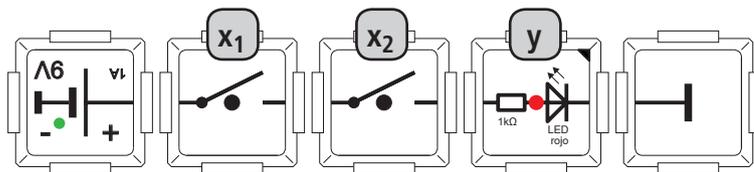


Figura 13: Circuito alternativo AND con pulsador

### 5.1.3 Puerta OR

La puerta OR realiza una conexión lógica OR de dos o más entradas. En comparación con la puerta NOR, la salida no se invierte aquí. En nuestro ejemplo, las dos entradas de la puerta OR están conectadas a la tensión de alimentación mediante pulsadores.

Mientras no se pulse ningún botón, las dos entradas se conectan a tierra (nivel Low) a través de las resistencias desplegadas integradas, de forma que hay un nivel definido en las entradas. El LED no se enciende hasta que se pulsa al menos uno de los dos botones. La salida cambia a nivel High de acuerdo con la tabla de verdad.

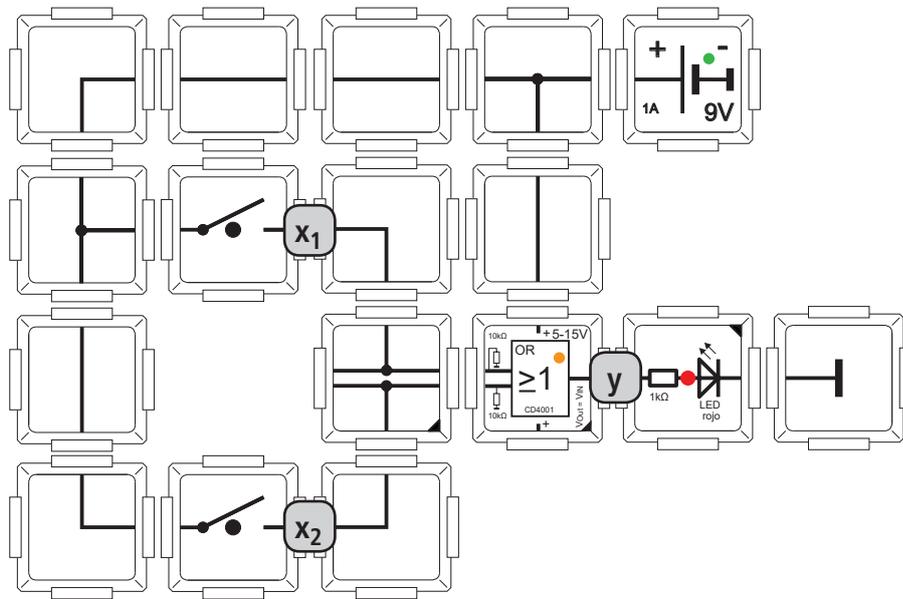


Figure 14: Puerta OR como circuito brick

| Símbolo del circuito | Descripción     | Ecuación           | Tabla de verdad  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|----------------------|-----------------|--------------------|--|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
|                      | OR (Disyunción) | $y = x_1 \vee x_2$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| $x_1$                | $x_2$           | $y$                |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                    | 0               | 0                  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                    | 1               | 1                  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                    | 0               | 1                  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                    | 1               | 1                  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |

Un circuito OR también se puede construir simplemente a partir de dos pulsadores, que se instalan en paralelo en el circuito. En cuanto se pulse al menos uno de los dos botones ( $x_1$  o  $x_2$ ) se enciende el LED ( $y$ ). Pruebe la función utilizando la tabla de verdad anterior.

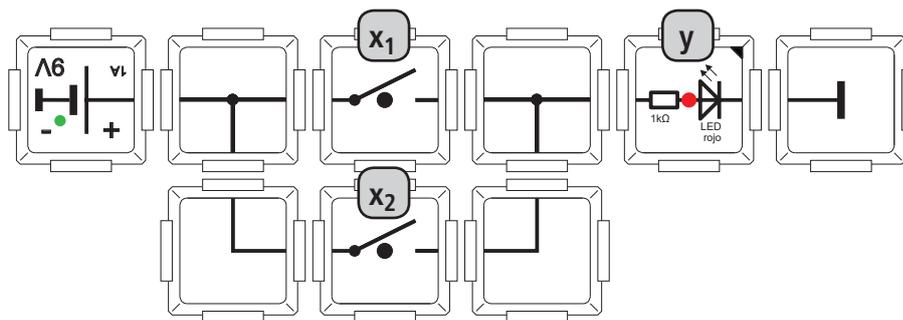


Figura 15: Circuito OR alternativo con pulsador

## 5.1.4 Puerta NAND

La puerta NAND realiza una conexión AND lógica de dos o más entradas con salida invertida. Junto a la puerta NOR, es el tipo de puerta más utilizado. No hay función que no pueda estar compuesta por estos dos tipos de puertas. Esto deja claro por qué estos dos tipos son muy comunes en la práctica.

En nuestro ejemplo, las dos entradas de la puerta NAND se conectan a la tensión de alimentación mediante pulsadores. Mientras no se pulse ningún botón, las dos entradas se conectan a tierra (nivel Low) a través de las resistencias desplegadas integradas. El LED se enciende porque la salida invertida emite un nivel High. Sólo cuando se pulsamos ambos botones la salida pasa a nivel Low (véase también la tabla de verdad) y el LED se apaga.

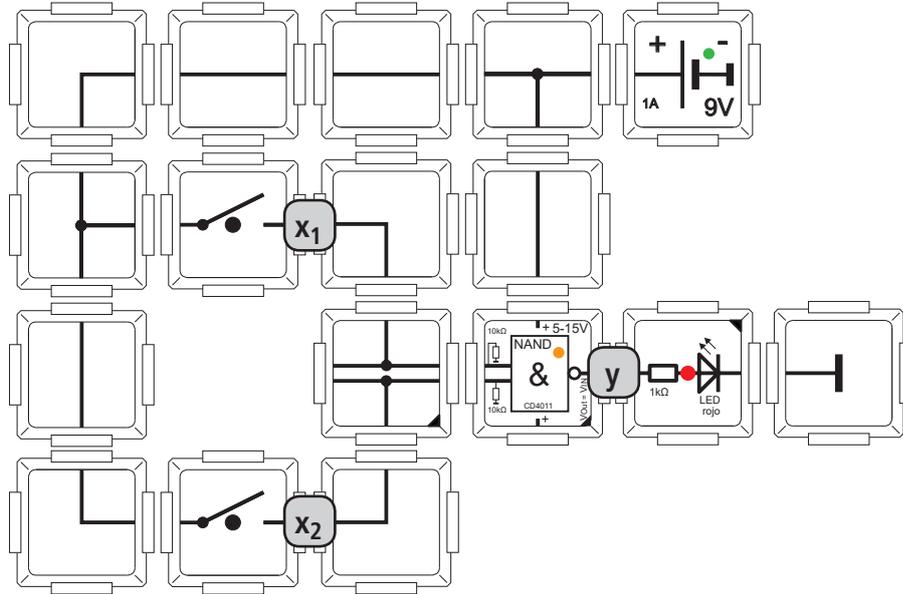
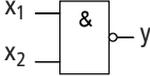


Figura 16: Puerta NAND como circuito brick

| Símbolo del circuito  | Descripción      | Ecuación                        | Tabla de verdad  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|---|------------------|---------------------------------|--|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
|  | NAND (Exclusión) | $y = \overline{x_1 \wedge x_2}$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| $x_1$   | $x_2$            | $y$                             |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0                | 1                               |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1                | 1                               |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0                | 1                               |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1                | 0                               |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |

Alternativamente, también puede hacer que el circuito NAND sea más compacto:

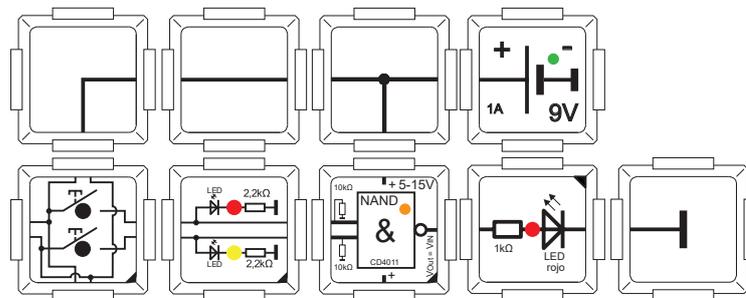


Figura 17: Circuito NAND alternativo

## 5.1.5 Puertas NOR

La puerta NOR realiza una combinación lógica OR de dos o más entradas con salida invertida. Junto con la puerta NAND, es el tipo de puerta más utilizado. No hay función que no pueda estar compuesta de estos dos tipos. Esto deja claro por qué estos dos tipos de puertas son muy comunes en la práctica. En nuestro ejemplo, las dos entradas de la puerta NOR están conectadas a la tensión de alimentación mediante pulsadores. Mientras no se pulse ningún botón, las dos entradas se conectan a tierra (nivel Low) a través de las resistencias desplegadas integradas. El LED se enciende porque la salida invertida emite un nivel High. En cuanto se pulsa al menos una de las dos teclas, la salida se pone a nivel Low (véase también la tabla de verdad), y el LED se apaga.

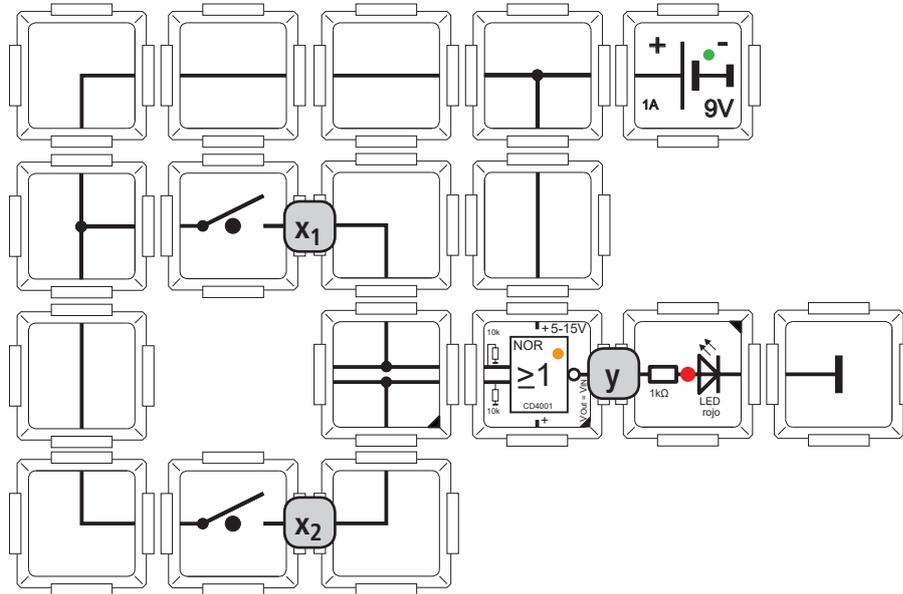


Figura 18: puerta NOR como circuito brick

| Símbolo de circuito | Descripción      | Ecuación                      | Tabla de verdad  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|---------------------|------------------|-------------------------------|--|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
|                     | NOR (Nihilación) | $y = \overline{x_1 \vee x_2}$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| $x_1$               | $x_2$            | $y$                           |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                   | 0                | 1                             |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                   | 1                | 0                             |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                   | 0                | 0                             |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                   | 1                | 0                             |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |

Alternativamente, también puede hacer que el circuito NOR sea más compacto, la funcionalidad es la misma:

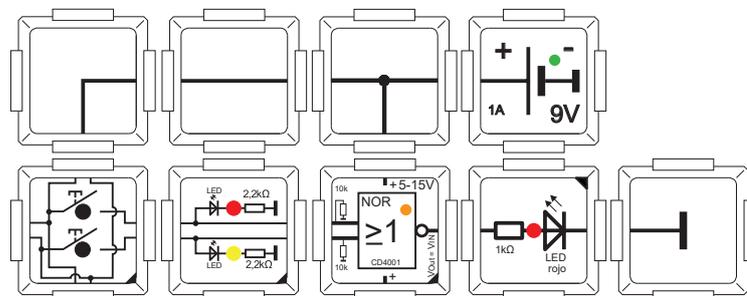


Figura 19: Circuito NOR alternativo

### 5.1.6 Puerta XOR

La puerta XOR se utiliza frecuentemente en el campo electrónico. Realiza una combinación lógica exclusiva OR de dos o más entradas. Un ejemplo sencillo es un interruptor de conmutación común, ya que se utiliza, por ejemplo, para la iluminación de pasillos en el hogar. Cada interruptor se puede utilizar para encender o apagar independientemente del interruptor que se pulse primero. En el lenguaje común, la puerta XOR también puede describirse como "o bien". El enlace XOR también es una parte importante del sumador, que conoceremos mejor en el capítulo 5.3 "Contador digital".

En nuestro ejemplo, las dos entradas de la puerta XOR están conectadas a la tensión de alimentación mediante pulsadores. Mientras no se pulse ningún botón, las dos entradas se conectan a tierra (nivel Low) a través de las resistencias desplegadas integradas. El LED no se enciende porque la salida emite un nivel Low. En cuanto se pulsa "o" uno de los botones "o" el otro, la salida cambia a nivel High y el LED se enciende. Si no se pulsa ningún botón o ambos, la salida vuelve al nivel Low según la tabla de verdad.

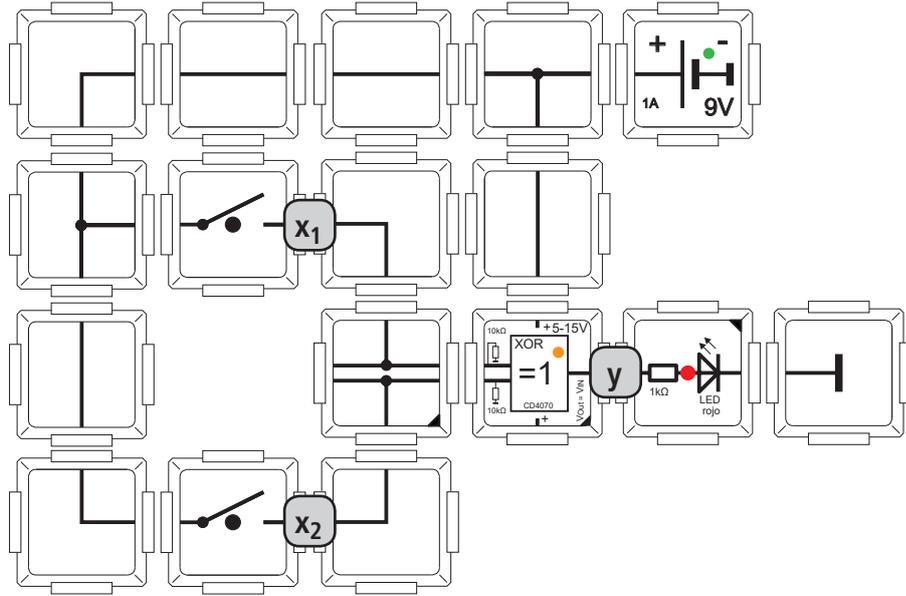


Figura 20: puerta XOR gate como circuito brick

| Símbolo de circuito | Descripción        | Ecuación   | Tabla de verdad   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---------------------|--------------------|--|---|----------------|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|
|                     | XOR (Antivalencia) | $y = (\bar{x}_1 \wedge x_2) \vee (x_1 \wedge \bar{x}_2)$ | <table border="1"> <thead> <tr> <th>x<sub>1</sub></th> <th>x<sub>2</sub></th> <th>y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table> | x <sub>1</sub> | x <sub>2</sub> | y | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| x <sub>1</sub>      | x <sub>2</sub>     | y  |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                   | 0                  | 0  |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                   | 1                  | 1  |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                   | 0                  | 1  |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                   | 1                  | 0  |   |                |                |   |   |   |   |   |   |   |   |   |   |   |   |   |

Alternativamente, también puede hacer que el circuito XOR sea más compacto, la funcionalidad es la misma:

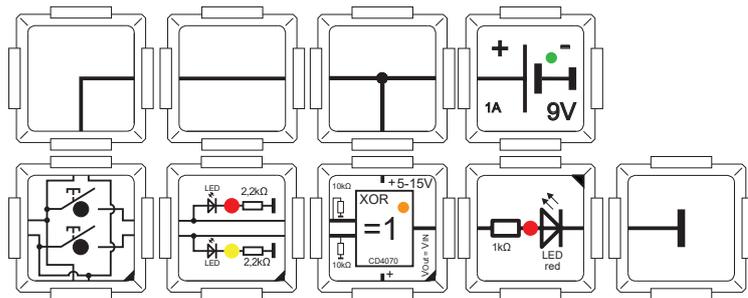


Figura 21: Circuito XOR alternativo

## 5.1.7 Puertas XNOR

La puerta XNOR realiza una conexión lógica exclusiva OR de dos o más entradas. Además de las puertas NAND y NOR, pertenece a las puertas con salida invertida. El enlace OR exclusivo invertido nos indica la igualdad de sus entradas. Esto se puede utilizar, por ejemplo, para comparar dos patrones de bits diferentes. En caso de igualdad del patrón de bits (ambas entradas "0" o ambas entradas "1"), se puede enviar una señal de interrupción al ordenador o simplemente activar una alarma acústica o visual.

En nuestro ejemplo, las dos entradas de la puerta XNOR se conectan a la tensión de alimentación mediante pulsadores. Mientras no se pulse ningún botón, las dos entradas se conectan a tierra (nivel Low) a través de las resistencias desplegadas integradas. En el caso del enlace XNOR, esto significa que la salida tiene un nivel High. Nuestro LED rojo "Alarma" se enciende. Lo mismo se aplica si pulsamos ambos botones simultáneamente. Sin embargo, tan pronto como se pulsa un solo botón, la salida vuelve al nivel Low y el LED se apaga.

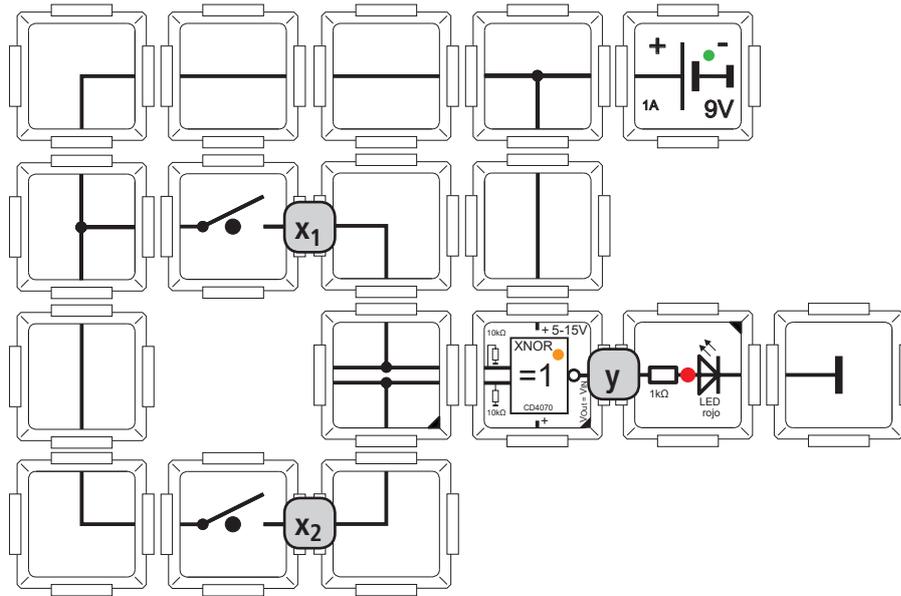


Figura 22: Puerta XNOR como circuito de brick

| Símbolo del circuito | Descripción         | Ecuación   | Tabla de verdad  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
|----------------------|---------------------|--|--|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
|                      | XNOR (Equivalencia) | $y = (x_1 \wedge x_2) \vee (\bar{x}_1 \wedge \bar{x}_2)$ | <table border="1"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>y</math></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> | $x_1$ | $x_2$ | $y$ | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| $x_1$                | $x_2$               | $y$  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                    | 0                   | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                    | 1                   | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                    | 0                   | 0  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                    | 1                   | 1  |  |       |       |     |   |   |   |   |   |   |   |   |   |   |   |   |

Alternativamente también puede construir el circuito XNOR de una manera más compacta, la funcionalidad es la misma:

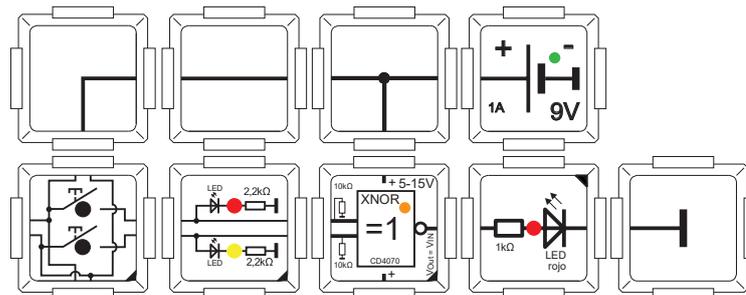


Figura 23: Circuito XNOR alternativo

## 5.2 Circuito antirrebotes

Los pulsadores e interruptores tienen la desventaja de que cuando se accionan, el contacto mecánico (a menudo se utiliza un resorte) causa múltiples cierres y aperturas. Este efecto perturbador también se denomina "rebote" en la tecnología digital.

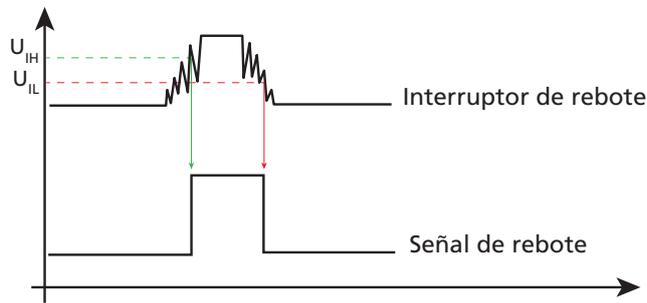


Figura 24: Impulso de interferencia típico por rebote de contacto

El molesto rebote puede ser eliminado con un simple flip-flop RS. Este tipo de flip-flop RS puede construirse a partir de dos puertas NAND o NOR. Utilizamos la variante de puerta NOR, cuyas entradas ya están equipadas con resistencias desplegadas (10 kΩ). Debido a la alta velocidad de conmutación del flip-flop RS, el primer contacto permanece estable en este estado y almacena el valor lógico hasta que se conmuta la otra entrada. Las resistencias desplegadas aseguran que las entradas desconectadas siempre tengan un estado definido. Presione las entradas S(et) y R(eset) para obtener un pulso sin rebote en las salidas Q.

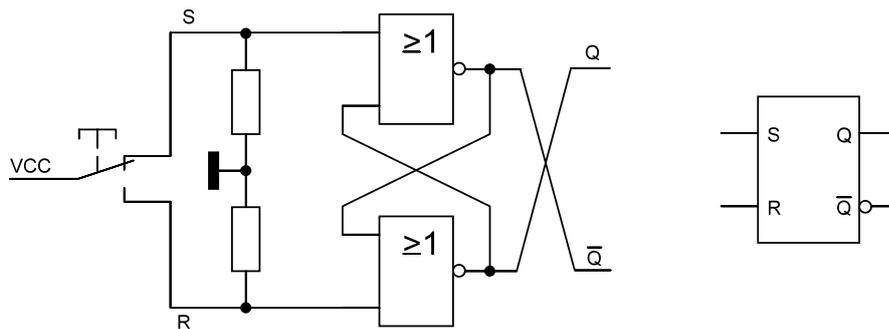


Figura 25: Circuito antirrebote construido con puertas NOR (izquierda) y flip-flop RS (derecha)

Para saber cómo utilizar el Flip-flop D 2x1 para construir un botón debounce, vea el Cap. 5.4.3 en la página 41. Sólo necesita conectar el set y reiniciar la entrada.

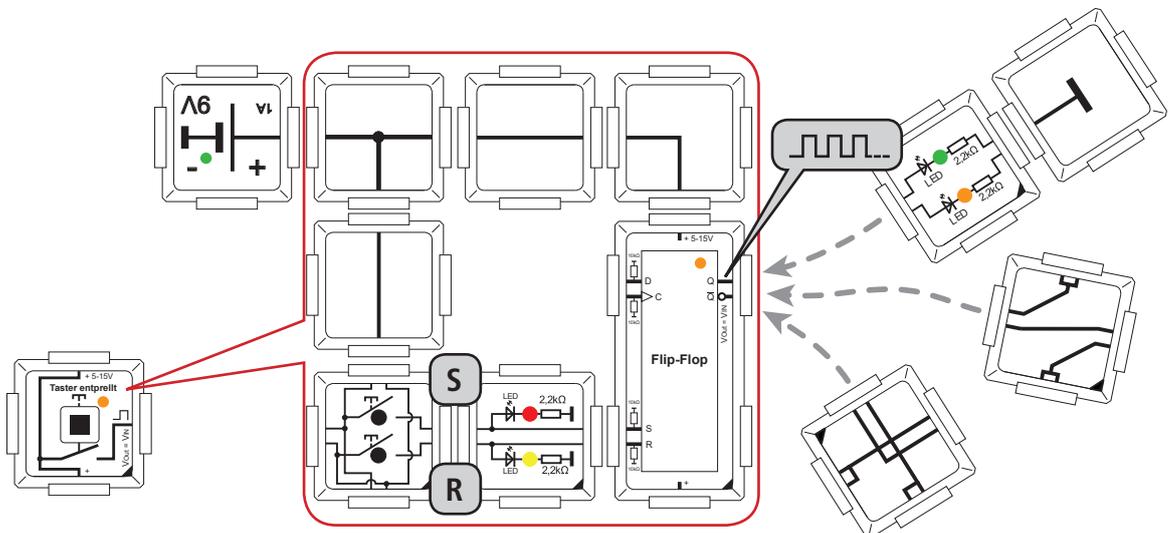


Figura 26: Circuito sin rebotes con brick flip-flop RS (borde rojo) o con pulsador sin rebotes (izquierda)

## 5.3 Contador digital

Los circuitos digitales se pueden ensamblar a partir de las puertas lógicas ya conocidas para realizar operaciones aritméticas en el sistema numérico binario. Este contador digital también se llama ALU (Unidad Aritmético-Lógica). Puede combinar dos valores de entrada mediante operaciones elementales, aritméticas y lógicas. Si la ALU se combina con una unidad de control (seguidor de condición) y un registro de resultados, se obtiene una unidad central de procesamiento, más conocida bajo el término en inglés CPU (Central Processing Unit). El centro de cada CPU es por lo tanto una ALU y su componente elemental es el sumador. En el caso más simple, se pueden añadir dos números binarios de un solo dígito A y B. En el primer paso construimos una semisumador de 1-bit, el cual expandimos en el segundo paso a un sumador completo.

### 5.3.1 Semisumador de 1-bit

El semisumador de 1 bit es el circuito de cálculo más simple y puede sumar dos números binarios de un solo dígito entre sí. La entrada A se suma a la entrada B. La suma S se genera mediante la operación XOR y la transferencia (Ü) mediante la operación AND. En principio, éste es el mismo procedimiento que para la suma escrita en el sistema de números decimales. Puede que recuerde: En cuanto se alcanza la decena, se anota el arrastre correspondiente en el siguiente dígito superior.

Las dos pulsadores corresponden a los sumandos A y B que hay que añadir. Pulsando una de las teclas se obtiene el valor binario "1". El LED rojo de suma en la salida de la puerta XOR se enciende ( $1 + 0 = 1$ ). En cuanto se pulsan ambas teclas, es decir, si calculamos  $1 + 1$ , se obtiene la suma "0" y se enciende el LED verde "1" en el sistema dual.

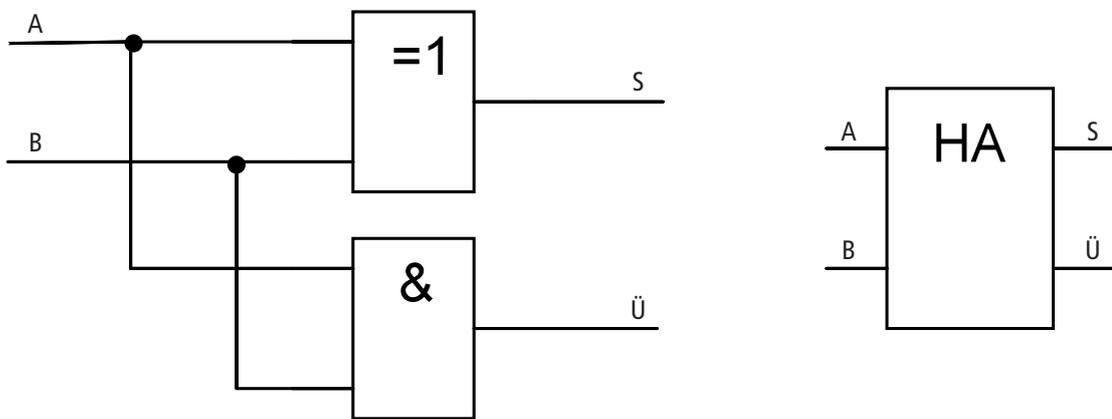


Figura 27: Circuito de un semisumador (izquierda) y su diagrama de bloques (derecha)

Tabla de verdad de un semisumador de 1-bit

| Sumando A | Sumando B | Sum S | Arrastre Ü |
|-----------|-----------|-------|------------|
| 0         | 0         | 0     | 0          |
| 0         | 1         | 1     | 0          |
| 1         | 0         | 1     | 0          |
| 1         | 1         | 0     | 1          |

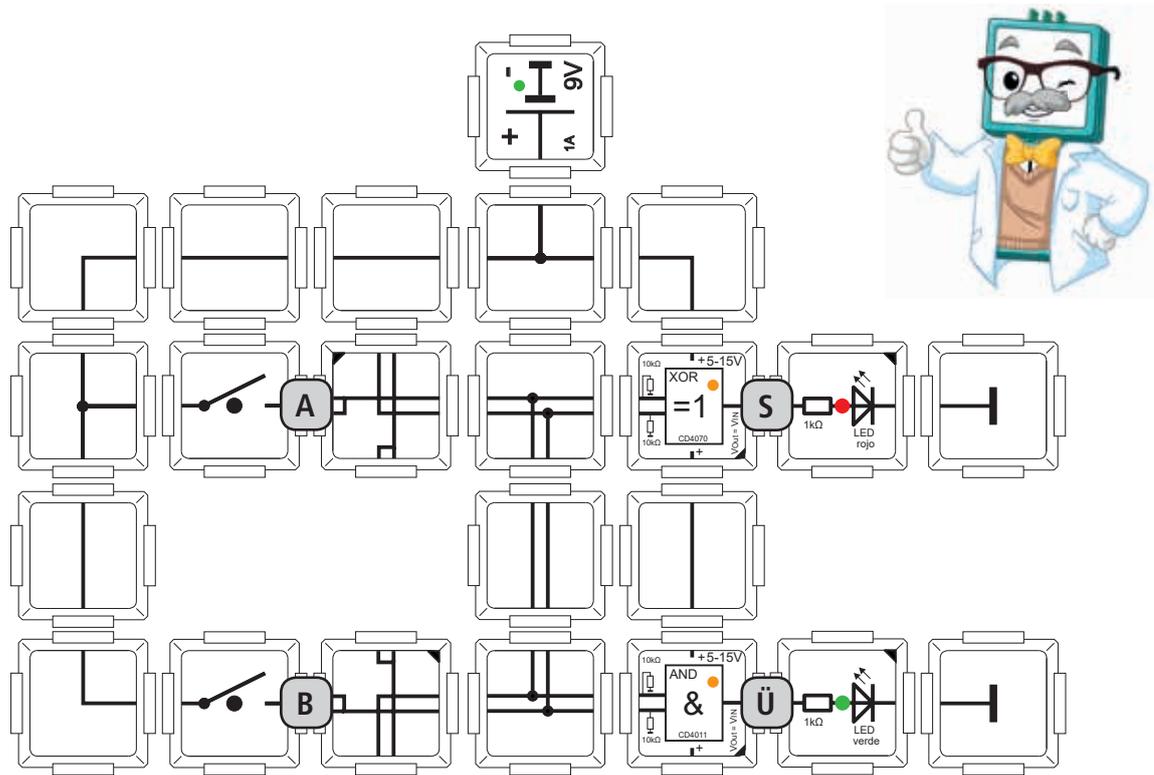


Figura 28: Semisumador de 1-bit como circuito de brick

### 5.3.2 Sumador completo de 1-bit

Si desea añadir números binarios de varios dígitos, el arrastre del dígito inferior en cada caso debe incluirse en la suma  $S$ . Nos referimos a este sumando adicional como acarreo de entrada de  $\ddot{U}_E$ . Combinamos dos semisumadores (HA) con un sumador completo (VA). Esto significa que, además de los sumandos  $A$  y  $B$ , también podemos utilizar el acarreo de entrada  $\ddot{U}_E$ . El resultado se visualiza como la suma  $S$  (LED rojo) y como el acarreo de salida  $\ddot{U}_A$  (LED verde).

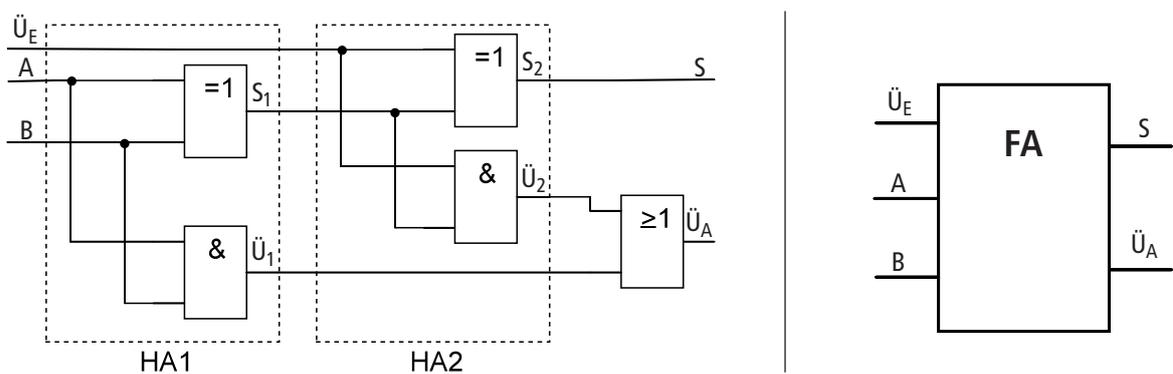


Figura 29: Circuito de un sumador completo (izquierda) y diagrama de bloque

#### Tabla de verdad para sumador completo de 1-bit

| Acarreo de entrada $\ddot{U}_E$ | Sumando A | Sumando B | Sum S | Acarreo de salida $\ddot{U}_A$ |
|---------------------------------|-----------|-----------|-------|--------------------------------|
| 0                               | 0         | 0         | 0     | 0                              |
| 0                               | 1         | 0         | 1     | 0                              |
| 0                               | 0         | 1         | 1     | 0                              |
| 0                               | 1         | 1         | 0     | 1                              |
| 1                               | 0         | 0         | 1     | 0                              |
| 1                               | 1         | 0         | 0     | 1                              |
| 1                               | 0         | 1         | 0     | 1                              |

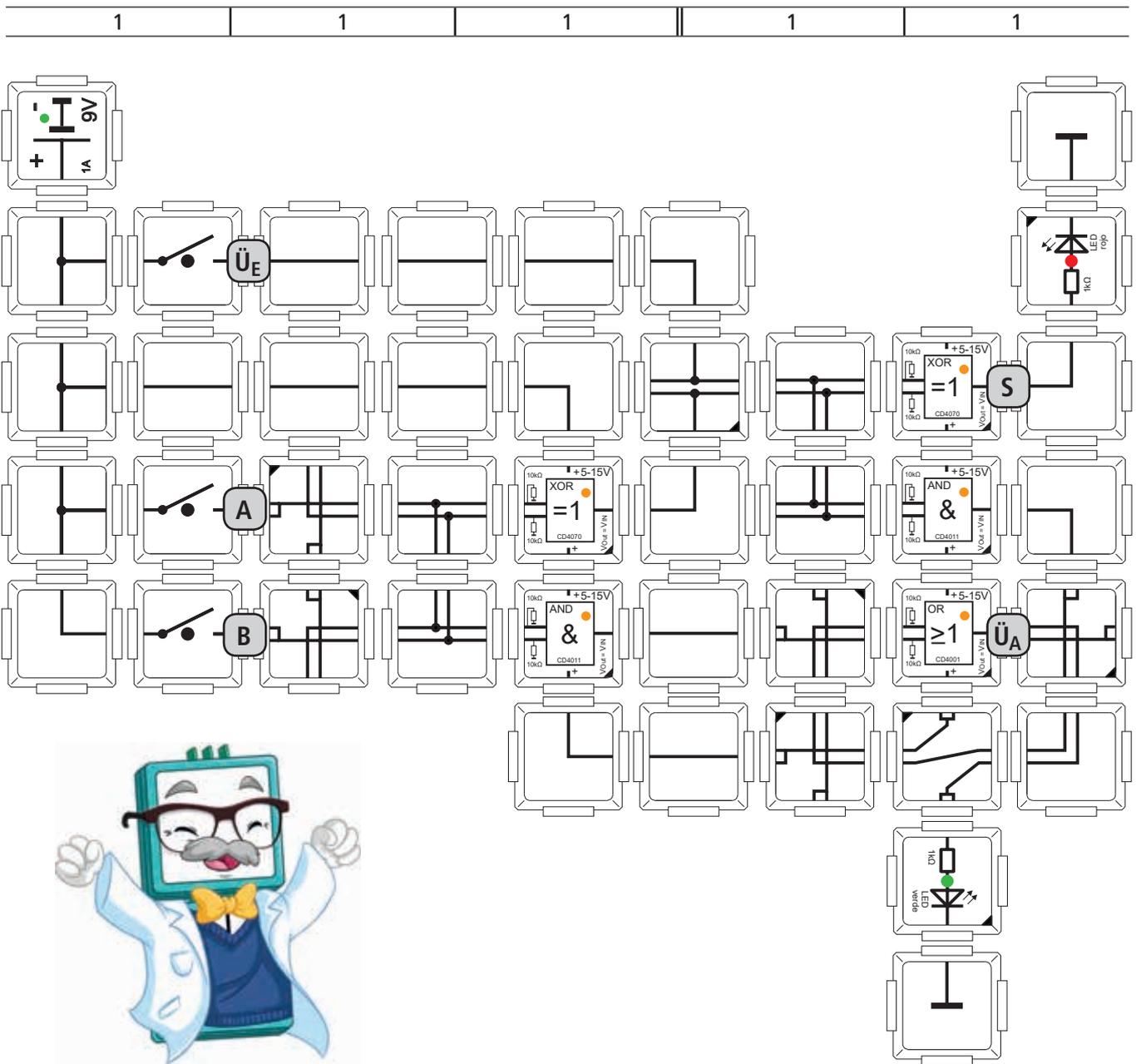


Figura 30: sumador completo de 1-bit como brick de circuito

### 5.3.3 Sumador completo de 4-bits

El siguiente diagrama de bloques muestra un sumador completo de 4 bits (VA0 a VA3) que consiste en cuatro sumadores completos de 1 bit. Esto permite agregar dos números binarios de cuatro dígitos. La entrada  $\ddot{U}_{E0}$  se puede conectar a tierra, ya que no se debe tener en cuenta ningún arrastre para el dígito inferior. Las transferencias de VA0 a VA3 se transmiten conectando las salidas de acarreo  $\ddot{U}_{An}$  con las entradas de acarreo de la correspondiente suma superior  $\ddot{U}_{En+1}$ . Aquí ya no se utiliza  $\ddot{U}_{A3}$ .

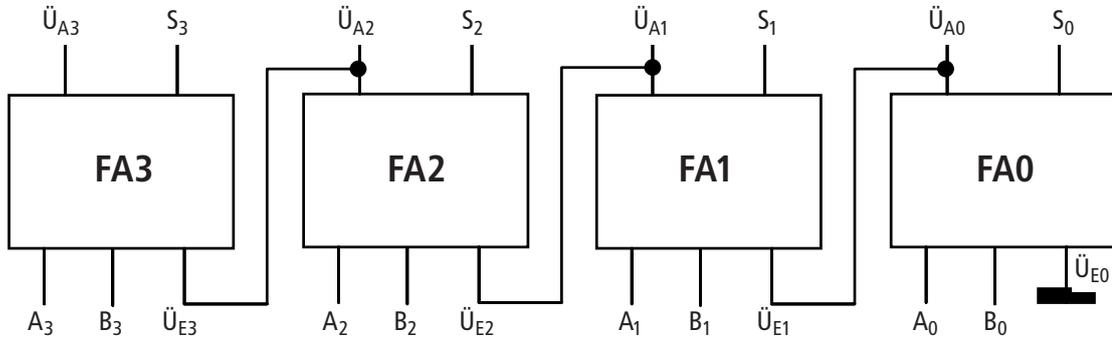


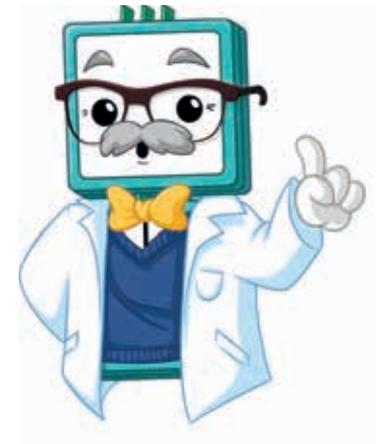
Figura 31: Diagrama de bloque sumador completo de 4-bits

**Por cierto:** En la tecnología digital es común empezar los índices de las líneas de bus o unidades lógicas con "0". Hay que tener en cuenta también que la valencia de los dígitos se muestra como la potencia de dos en el sistema binario:  $2^n$  - empezando por  $n = 0$ . Por ejemplo:

|           |           |           |           |
|-----------|-----------|-----------|-----------|
| $2^3 = 8$ | $2^2 = 4$ | $2^1 = 2$ | $2^0 = 1$ |
|-----------|-----------|-----------|-----------|



Debido a la complejidad del circuito, no es posible construir el sumador completo de 4 bits con los bricks del conjunto lógico.



## 5.4 Flip-flops

Suponga que está esperando un paquete pero no está en casa para recibirlo. Es bueno que haya un buzón para que el remitente o cartero pueda dejarlo para usted. De esta manera el mensaje no se perderá y usted puede recogerlo del buzón y leerlo tan pronto como tenga tiempo. El buzón de correo también podría describirse como una especie de búfer. El indicador rojo indica si la información está lista o no. Por lo tanto, puede señalar dos estados. Así es como se puede imaginar el llamado flip-flop, como el elemento de memoria digital más pequeño (1 bit).



El término "flip-flop" indica en la tecnología digital un circuito electrónico que puede tener dos estados eléctricos estables (0 y 1). Cambiando las señales de entrada, el flip-flop puede cambiar su estado de uno a otro.

Por lo tanto tenemos la posibilidad de almacenar una información digital (1 bit), siempre y cuando haya una tensión de alimentación. El flip-flop es el centro de los circuitos secuenciales y se utiliza en varios tipos. En los microprocesadores, varios flip-flops se conectan en paralelo. El ancho de un registro nos muestra cuántos flip-flop están conectados en paralelo en el registro. Las unidades comunes son: 8 bits = 1 byte; 16 bits = 1 palabra; 32 bits = 1 palabra larga.

A menudo, el flip-flop RS también se compara con un balancín mecánico:

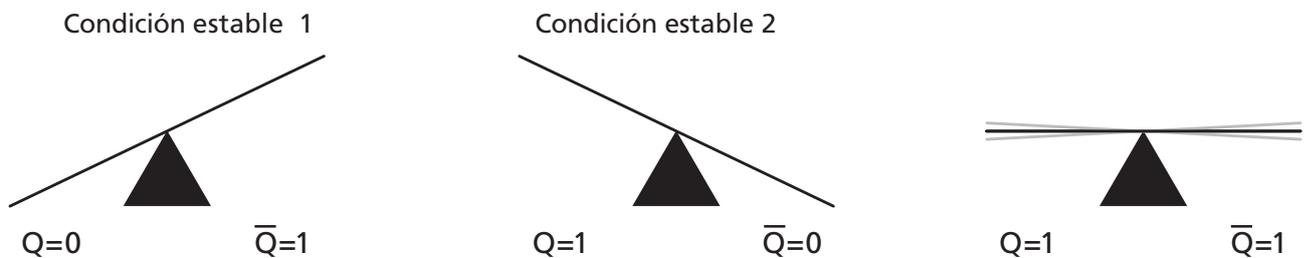


Figura 32: Flip-flop comparado con un balancín

Un balancín tiene dos estados estables: o bien su extremo izquierdo o bien su extremo derecho toca el suelo. Con el esfuerzo físico apropiado, es posible cambiar su estado. En un flip-flop esto ocurre si se activan las entradas,  $R = 0, S = 1$  o  $R = 1, S = 1$ . Si no se hace ningún esfuerzo físico para cambiar el estado del balancín, éste se queda en el último estado: el estado se almacena. Si el balancín está en posición horizontal, tiene un estado metaestable ( $R = 1$  y  $S = 1$ ): en la práctica, el balancín nunca puede descansar en esta posición. En algún momento cambiará a un lado o al otro. Incluso se puede saber en qué estado cambia porque nunca se conocen todas las interferencias exactas. Un flip-flop cronometrado es equivalente a un balancín, en el que la fuerza de control sólo se aplica durante un período determinado por una señal de reloj externa.

## 5.4.1 Tipos de Flip-flop

Los diferentes tipos de flip-flop difieren en el tipo y número de entradas y su comportamiento. Tienen entradas con diferentes efectos y sólo cambian su estado bajo ciertas condiciones definidas. Un flip-flop simple tiene dos entradas y dos salidas. Los flip-flops dependientes del reloj tienen una adicional entrada de reloj correspondiente.

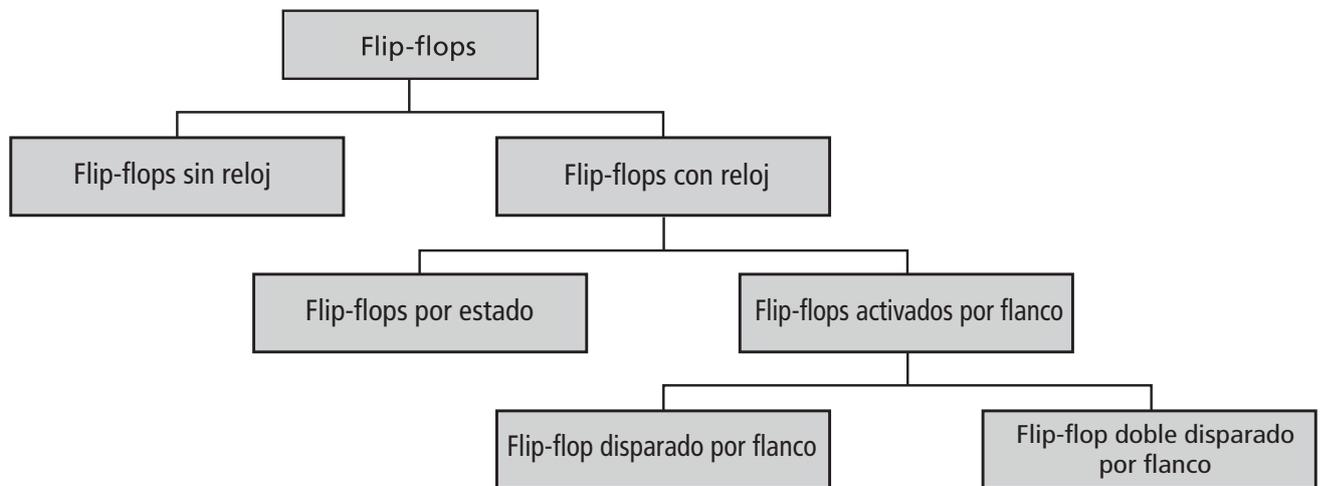
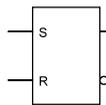


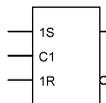
Fig. 33: Tipos de Flip-flops

### 5.4.1.1 Flip-flops sin reloj



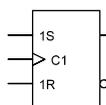
El flip-flop RS sin entrada de reloj es completamente independiente del reloj. Sus entradas de ajuste y reset, S y R, pueden activarse en cualquier momento. Véase también el capítulo 5.4.2 en la página 40.

### 5.4.1.2 Flip-flops por estado



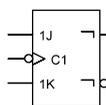
Las entradas de ajuste y reset, S y R, de un flip-flop RS controlado por estado sólo son efectivas mientras exista un nivel de señal (en este caso High) en la entrada C del reloj. Este tipo de flip-flop puede derivarse del flip-flop RS cambiando una puerta AND antes de cada entrada de un flip-flop RS que consiste en puertas NOR y combinando una entrada, cada una como entrada de reloj.

### 5.4.1.3 Flip-flop disparado por flanco



Con flip-flop controlado por un solo flanco, la entrada de ajuste y restablecimiento (1S y 1R) sólo es efectiva cuando se cambia el flanco de la entrada de reloj C1. Se reduce la susceptibilidad al fracaso. El control del flanco del reloj se indica mediante el triángulo en el símbolo del circuito. Véase también el capítulo 5.4.3 en la página 41.

### 5.4.1.4 Flip-flop doble disparado por flanco



El disparador de dos flancos recoge el estado de entrada durante el primer flanco del reloj (aquí flanco positivo) y lo emite con el siguiente flanco del reloj (negativo). La susceptibilidad a los fallos se minimiza. El control del flanco del reloj se indica mediante el triángulo en el símbolo del circuito. Véase también el capítulo 5.4.4 en la página 42.

Los flip-flops de doble flanco controlados, cuyos estados de entrada aparecen retardados en la salida, también se denominan flip-flops maestro-esclavo. Esto se puede ver en el símbolo de conmutación en el ángulo recto de la salida. Un conocido referente de este tipo es el llamado flip-flop maestro-esclavo JK.

En los siguientes capítulos conocerás los flip-flops que se incluyen en el set lógico de Brick'R'knowledge.

## 5.4.2 Flip-flop RS

El flip-flop RS no controlado por reloj también puede constar de dos puertas NOR. Este circuito básico también se llama flip-flop NOR. En principio, el circuito también es posible con dos flip-flops NAND, pero esto requiere entradas negadas.

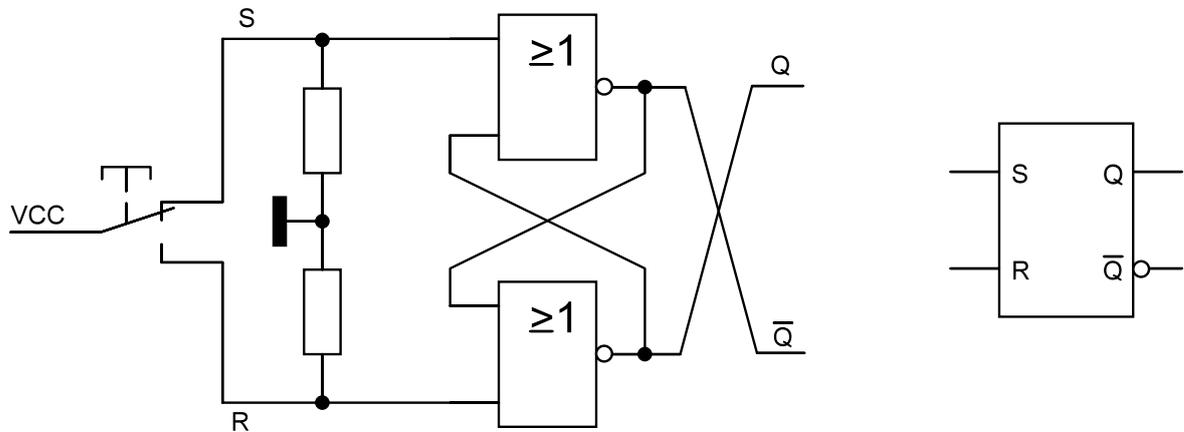


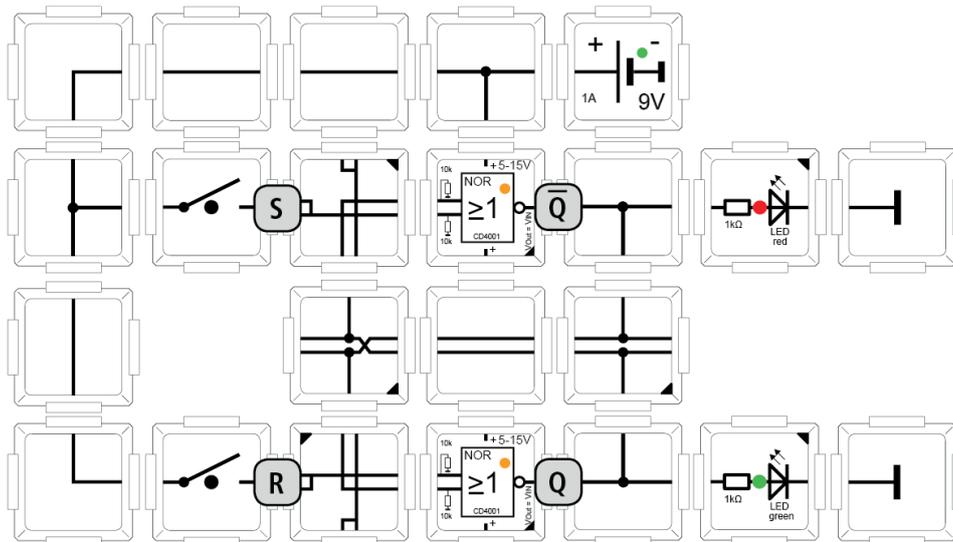
Figura 34: Flip-flop construido con dos puertas NOR (izquierda) y el símbolo de circuito del flip-flop RS (derecha)

Las entradas de ajuste y reset también se denominan R y S, como se muestra en el símbolo de circuito flip-flop RS. Las dos salidas Q y  $\bar{Q}$  están negadas entre sí.

### Tabla de verdad de flip-flop RS no controlado por reloj

| Entrada set S | Entrada reset R | Salida Q | Salida $\bar{Q}$ | Descripción      |
|---------------|-----------------|----------|------------------|------------------|
| 1             | 0               | 1        | 0                | Set (SET)        |
| 0             | 0               | Q        | $\bar{Q}$        | Sin cambios      |
| 0             | 1               | 0        | 1                | Reset (RESET)    |
| 1             | 1               | 0        | 0                | Estado prohibido |

Tenga en cuenta que en el siguiente circuito de brick la salida Q está por debajo y la salida  $\bar{Q}$  por encima!



- Set: Pulsar la tecla de la entrada S para ajustar la salida Q a "1".
- Sin cambios: Si no se pulsa ninguna tecla, las salidas permanecen inalteradas.
- Reset: Pulsar la tecla de la entrada R para ajustar la salida Q a "0".
- Estado prohibido

### 5.4.3 Flip-flop D

El Flip-flop D consiste en un Flip-flop RS, donde la entrada de reset se conecta a la entrada de ajuste a través de un inversor. Esto evita que se produzca el estado indefinido.

El flip-flop D está disponible como flip-flop de activación por estado o por flanco, como en este Logic Set. Si un flip-flop D tiene entradas de ajuste y restablecimiento de LED también se puede ajustar o resetear de forma asíncrona (véase también el capítulo 5.2 en la página 33).

El flip-flop D es el elemento de memoria básico de lectura-escritura. En la práctica, muchos flip-flop D suelen estar conectadas en paralelo y sincronizadas mediante un reloj común. Esta disposición de flip-flops 4, 8, 16 o 32 D se denomina registro. Una aplicación práctica es, por ejemplo, un registro de estado que debe leerse para recibir información sobre un dispositivo o un registro de comandos para transmitir comandos a un dispositivo. El puerto D es la entrada de datos. En el caso de un flip-flop D controlado por un solo flanco, el valor en la entrada de datos ("0" o "1") se toma con el flanco positivo en la entrada de reloj C.

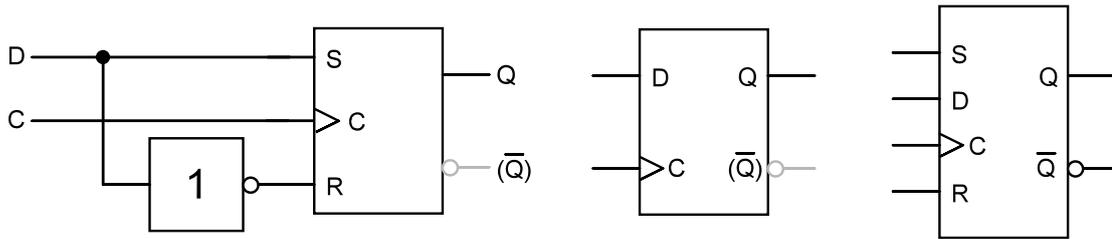


Figura 36: Flip-flop D con control de flancos Flip-flop RS(izquierda), símbolo de circuito de un simple Flip-flop D (centro) y con entradas RS (derecha)

Tenga en cuenta que el brick estándar flip-flop D que se incluye en el Logic Set no contiene una salida  $\bar{Q}$ .

**Tabla de verdad para Flip-flop D controlado por flanco positivo (opc. con entradas de set y reset)**

| Entrada de datos D | Entrada del reloj C | Salida Q | (Salida $\bar{Q}$ ) | Discripción |
|--------------------|---------------------|----------|---------------------|-------------|
| 1                  | 0 → 1               | 1        | (0)                 | Set         |
| 0                  | 0 → 1               | 0        | (1)                 | Reset       |
| 1                  | 1 → 0               | Q        | ( $\bar{Q}$ )       | Sin cambios |
| 0                  | 1 → 0               | Q        | ( $\bar{Q}$ )       | Sin cambios |

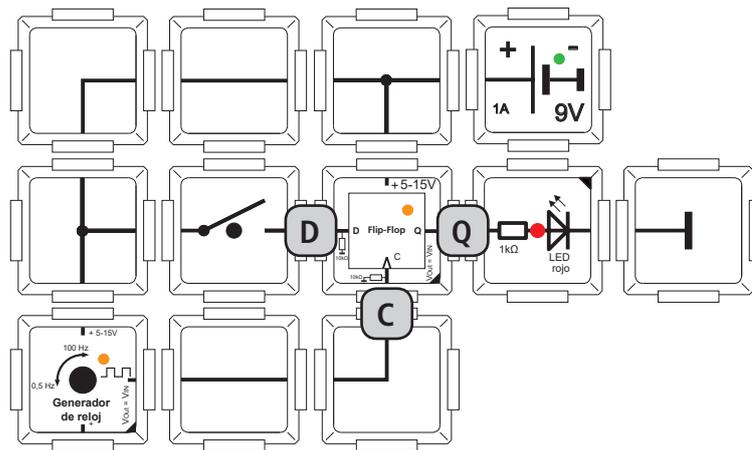


Figure 37: Circuito con Flip-flop D estándar

Flip-flop D con set asíncrono y entrada de reset

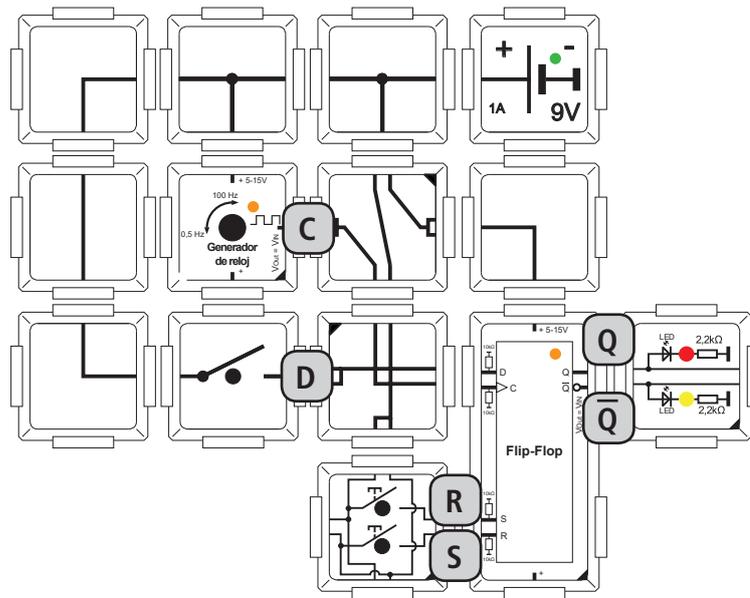


Figura 38: Circuito de brick con Flip-flop y entrada asíncrona de set y reset (izquierda), símbolo de flip-flop D.

#### 5.4.4 Flip-flop JK

El flip-flop JK está disponible como flip-flop controlado por reloj y por estado. El set lógico de Brick'R'knowledge contiene cuatro flip-flops JK controlados por un solo flanco. Cuando se aplica un flanco de reloj positivo, el estado de salida cambia si "1" está presente en las dos entradas J y K. A este comportamiento se le denomina conmutación o "Toggling". En un flip-flop en T también hay dos entradas J y K conectadas que están disponibles para alternar.

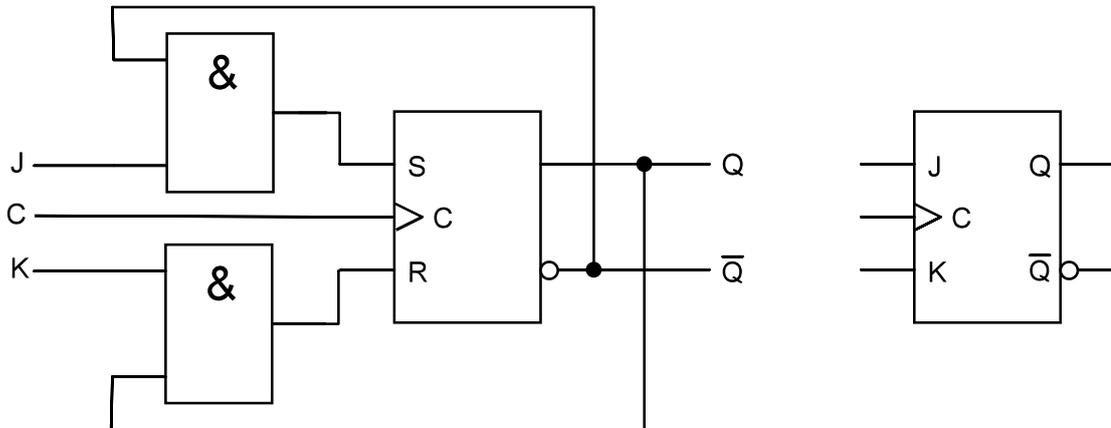


Figura 39: Estructura de un flip-flop JK (izquierda) y símbolo de circuito para un flip-flop JK disparado por un flanco (derecha)

Se debe aplicar una señal de onda cuadrada a la entrada de reloj C (del inglés "Clock") del flip-flop JK. El flip-flop JK que utilizamos siempre reacciona a un flanco de reloj positivo, es decir, la transición en C de "0" a "1". Las dos entradas J y K son entradas de control. Las salidas Q y  $\bar{Q}$  se controlan en función de las entradas J y K.

Tabla de verdad para flip-flops JK disparados por el flanco positivo

| Entrada J | Entrada K | Entrada de reloj C | Salida Q  | Salida $\bar{Q}$ | Descripción            |
|-----------|-----------|--------------------|-----------|------------------|------------------------|
| 0         | 0         | 0 → 1              | Q         | $\bar{Q}$        | Sin cambios            |
| 1         | 0         | 0 → 1              | 1         | 0                | Set                    |
| 0         | 1         | 0 → 1              | 0         | 1                | Reset                  |
| 1         | 1         | 0 → 1              | $\bar{Q}$ | Q                | Salidas de conmutación |

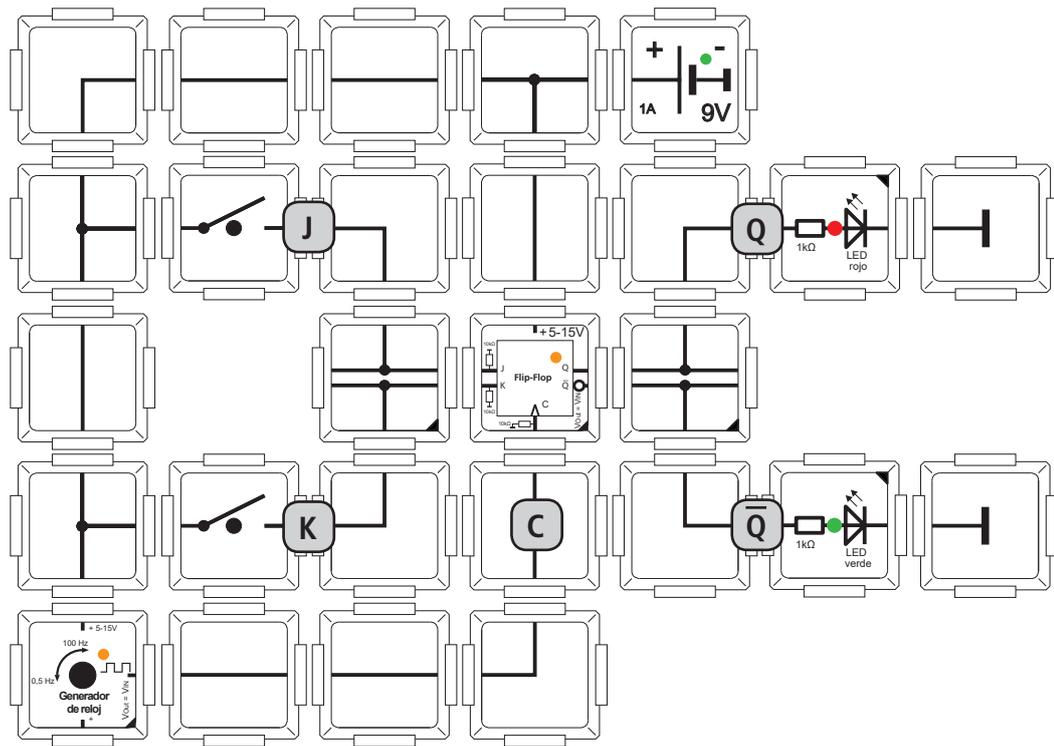


Figura 40: JK flip-flop como circuito de brick

- Sin cambio: Si no se pulsa ninguna tecla, las salidas Q y  $\bar{Q}$  permanecen inalteradas.
- Set: Pulsar la tecla de la entrada J para ajustar la salida Q a "1" y  $\bar{Q}$  a "0".
- Reset: Pulse el botón de la entrada K (no pulse el botón J) para ajustar la salida Q a "0" y  $\bar{Q}$  a "1".
- Conmutación de salidas: Si se pulsamos ambos pulsadores, las salidas se conmutan. Esto significa para la salida Q: un "1" en el flanco del reloj se convierte en un "0" y un "0" en el flanco del reloj se convierte en un "1". La salida  $\bar{Q}$  es equivalente invertida.

### Circuito flip-flop JK alternativo

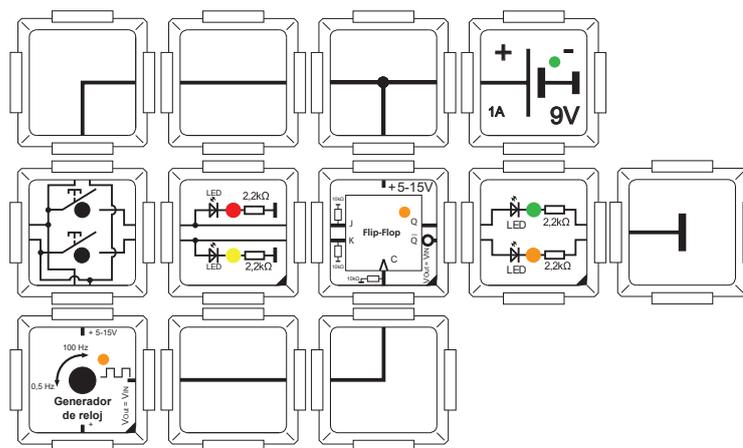


Figura 41: circuito flip-flop JK alternativo

## 5.5 Registro de desplazamiento

Los registros de cambio consisten en flip-flops conectados en serie, que son sincronizados. Además de la suma (véase el capítulo 5.3 en la página 34), el cambio de bits es una de las operaciones elementales de unidad aritmética. Por ejemplo, si el número binario importado se desplaza un dígito a la derecha, el resultado decimal corresponde a una división de 2. Para la multiplicación, el número binario correspondiente debe moverse una posición a la izquierda. Dependiendo de la aplicación, los registros de desplazamiento tienen diferentes anchos de bits, se mueven a la derecha o a la izquierda y pueden cargarse en serie o en paralelo. Los registros de desplazamiento son adecuados para muchas tareas como circuitos integrados (ICs), que ya contienen todas las funcionalidades estándar.

Las aplicaciones posibles para los registros de cambio son:

- Operaciones aritméticas (multiplicación, división)
- Buffer circular
- Convertidores serie-paralelo y paralelo-serie
- Retraso de señal

### 5.5.1 Registro de desplazamiento de carga serie

Si se carga un registro de desplazamiento con un "1" en la entrada D, este valor se desplaza un dígito (flip-flop) con cada reloj.

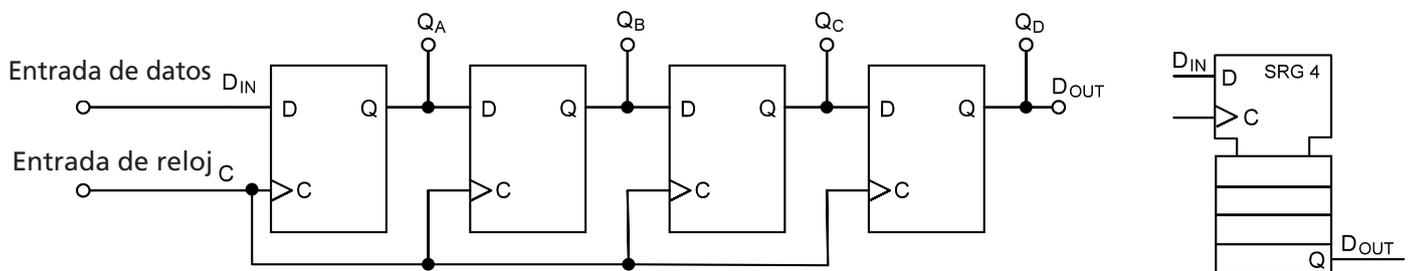


Figura 42: Registro de desplazamiento de 4 bits construido con un flip-flop D con entrada y salida serial (izquierda), símbolo de circuito (derecha)

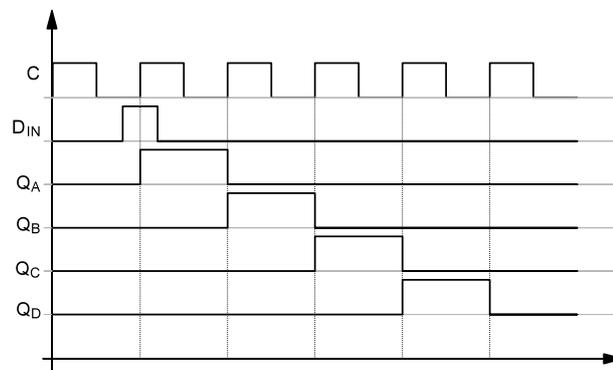


Figura 43: Diagrama de temporización de registro de desplazamiento

En el siguiente circuito de brick, el registro de desplazamiento consta de cuatro flip-flops D. Mientras se pulsa el botón, hay un nivel High en la entrada de datos del primer flip-flop. Con el siguiente flanco positivo del reloj C se acepta "1" y se emite a  $Q_A$ . Al mismo tiempo, el nivel de  $Q_A$  se transmite a la entrada de datos del segundo flip-flop. El principio continúa hasta que después de cuatro ciclos de reloj la palabra de datos de 4 bits de ancho está en el registro de desplazamiento ( $Q_A$  a  $Q_D$ ). En nuestro ejemplo,  $Q_D$  es también la salida de datos serie  $D_{OUT}$ . Alternativamente, esto podría conectarse a  $D_{IN}$  para construir un registro de anillo en el que los bits se desplazan en un círculo.

El reloj se genera con el brick del reloj. El conocido módulo temporizador tipo 555 se utiliza debajo de la campana. La frecuencia de reloj se puede ajustar entre 0,5 Hz y 100 Hz (véase también 5.2.2 en la página 33). El botón de desestimación no es recomendable para este ejercicio!

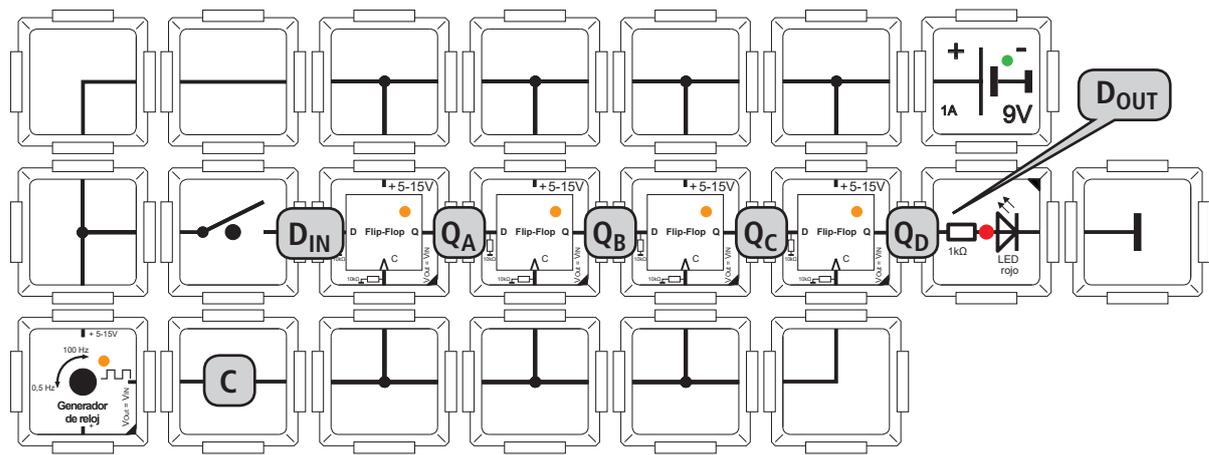


Figura 44: Registro de desplazamiento como circuito de bricks

## 5.6 Contador

Los circuitos Flip-flop son ideales como contadores digitales. Los contadores binarios se clasifican según su control y dirección de conteo. Existen dos tipos básicos de contadores, asíncronos y síncronos, dependiendo del tipo de conteo. La señal del reloj debe estar siempre libre de rebotes para evitar errores de conteo causados por rebotes en los contactos (véase también el capítulo 5.2 en la página 33).

En los contadores síncronos, todos los flip-flops de memoria son pulsados por el mismo pulso de conteo. No hay adición de los tiempos de ejecución de la señal, como es el caso de los contadores asíncronos causados por la transferencia del reloj de nivel a nivel. En comparación con los contadores asíncronos, el método síncrono permite frecuencias de conteo máximas considerablemente más altas. Mientras que el esfuerzo del circuito es mucho mayor con los contadores síncronos.

El reloj contador sólo controla el primer flip-flop. Todos los siguientes son controlados por los niveles de salida de su predecesor. Sin embargo, como con todos los procesos asíncronos, esto puede llevar a circuitos confusos con problemas de tiempo de ejecución. Sin embargo, el esfuerzo de la puerta es menor. Donde el esfuerzo del circuito es mucho mayor con contadores síncronos. A los contadores asíncronos también se les denomina "contador de ondulaciones" porque el reloj de control entre los flip-flops, recuerda a una onda.

Tanto los contadores síncronos como los asíncronos están disponibles como contadores ascendentes o descendentes. Se pueden utilizar circuitos adicionales para conmutar la dirección de conteo y limitar el rango de conteo. Si es necesario, el valor del contador puede ser emitido en diferentes códigos (por ejemplo, binario, BCD o contador Gris). Véase también la tabla de la página 18.

El componente básico universal de un contador binario es un flip-flop JK conectado como un flip-flop de palanca. Cada flip-flop corresponde a un dígito binario y almacena un bit. Con una combinación adecuada de los flip-flops (ver capítulos siguientes) puedes crear contadores binarios y de varios dígitos.

La capacidad K de un contador binario depende del número n de memorias individuales (flip-flops). La siguiente fórmula se aplica a un contador binario:

$$K = 2^n - 1$$

Los valores para n = 2 a 10 se muestran en la siguiente tabla

### Dependencia de la profundidad de conteo y de la capacidad de conteo

| Número de flip-flops n / (contando la profundidad en bits) | Capacidad de conteo (K) |
|--|-------------------------|
| 2  | 3                       |
| 3  | 7                       |
| 4  | 15                      |
| 5  | 31                      |
| 6  | 63                      |
| 7  | 127                     |
| 8  | 255                     |
| 9  | 511                     |
| 10   | 1023                    |

En la práctica, se utilizan normalmente contadores de 4 u 8 bits de ancho. Si es necesario, por ejemplo, se pueden enlazar dos contadores de 8 bits para formar un contador de 16 bits, por ejemplo. En el diseño de circuitos modernos, los contadores se integran como bloques lógicos en dispositivos programables como FPGA (Field-Programmable Gate Arrays). Véase también el capítulo 4.8 en la página 23.



### 5.6.1 Contador asíncrono binario de 4 bits

Los contadores que funcionan asíncronamente no tienen un pulso de reloj común. Las flip-flop se conectan en serie, es decir, la salida de la primera flip-flop se conecta a la entrada del reloj del segundo flip-flop, y así sucesivamente. Debido a los tiempos de propagación de la señal, cada flip-flop recibe el flanco del reloj a diferentes horas en la entrada del reloj. Los tiempos de propagación de la señal de las distintas etapas de la memoria se suman. Esto tiene un efecto particular en contadores binarios con gran profundidad de conteo y altas frecuencias de reloj. Para un conteo correcto, el tiempo de ciclo debe ser siempre mayor que el tiempo total de funcionamiento de la señal a través del contador completo.

Si se conoce el tiempo de conmutación más corto  $t$  (en segundos) de una memoria para  $n$  flip-flops idénticos, se calcula la frecuencia máxima (frecuencia de corte  $f_g$  en hercios [Hz] ) del reloj de conteo:

$$f_g = \frac{1}{(n + 1) \times t}$$

Con flip-flops en tecnología CMOS o TTL, el tiempo de propagación de la señal tarda unos pocos nanosegundos. Cuanto mayor sea la profundidad de contaje (por ejemplo, 16 bits), más tiempo tardará el pulso en propagarse desde el primer hasta el último Flip-Flop. Este tiempo de funcionamiento del impulso de contaje puede provocar fallos y, por lo tanto, errores durante el conteo. Cuanto mayor sea la frecuencia de conteo, antes se producirán los problemas. Mientras el conteo sólo esté en el rango de hasta 100 Hz - como con nuestro brick de reloj - un contador asíncrono de 4 bits no es ningún problema.

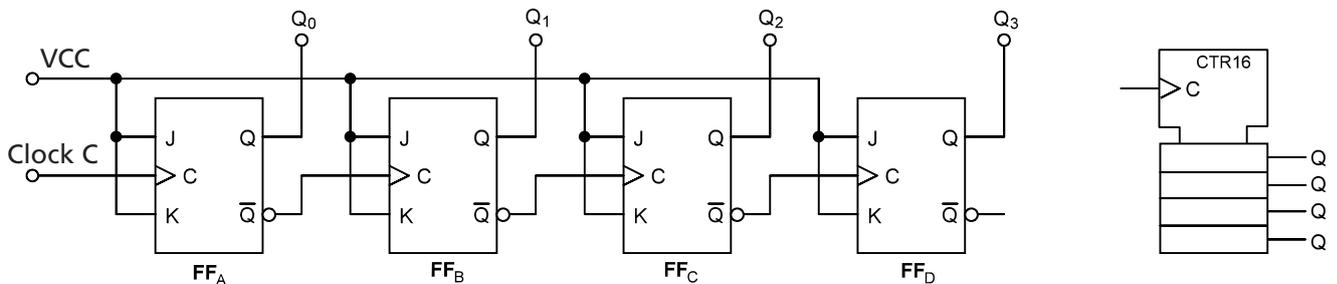


Figura 45: Contador binario asíncrono de 4 bits (izquierda), símbolo de circuito (derecha)

### 5.6.1.1 Contador asíncrono ascendente

El siguiente circuito de bricks representa un contador ascendente asíncrono. El proceso de conteo se puede observar fácilmente utilizando los LED naranjas en los bricks flip-flop, que están conectados internamente a la salida Q (véase también el diagrama de temporización Fig. 47). Las etapas de flip-flop se conectan en serie conectando la salida  $\bar{Q}$  de la primera etapa a la entrada de reloj C de la segunda etapa, y así sucesivamente. Como utilizamos flip-flops JK controlados de una sola arista, la transición del estado de ajuste al de reposición provoca un flanco positivo en la salida  $\bar{Q}$ . Dado que tanto la entrada J como la entrada K están conectadas a nivel High, esto provoca la conmutación de las siguientes flip-flop.

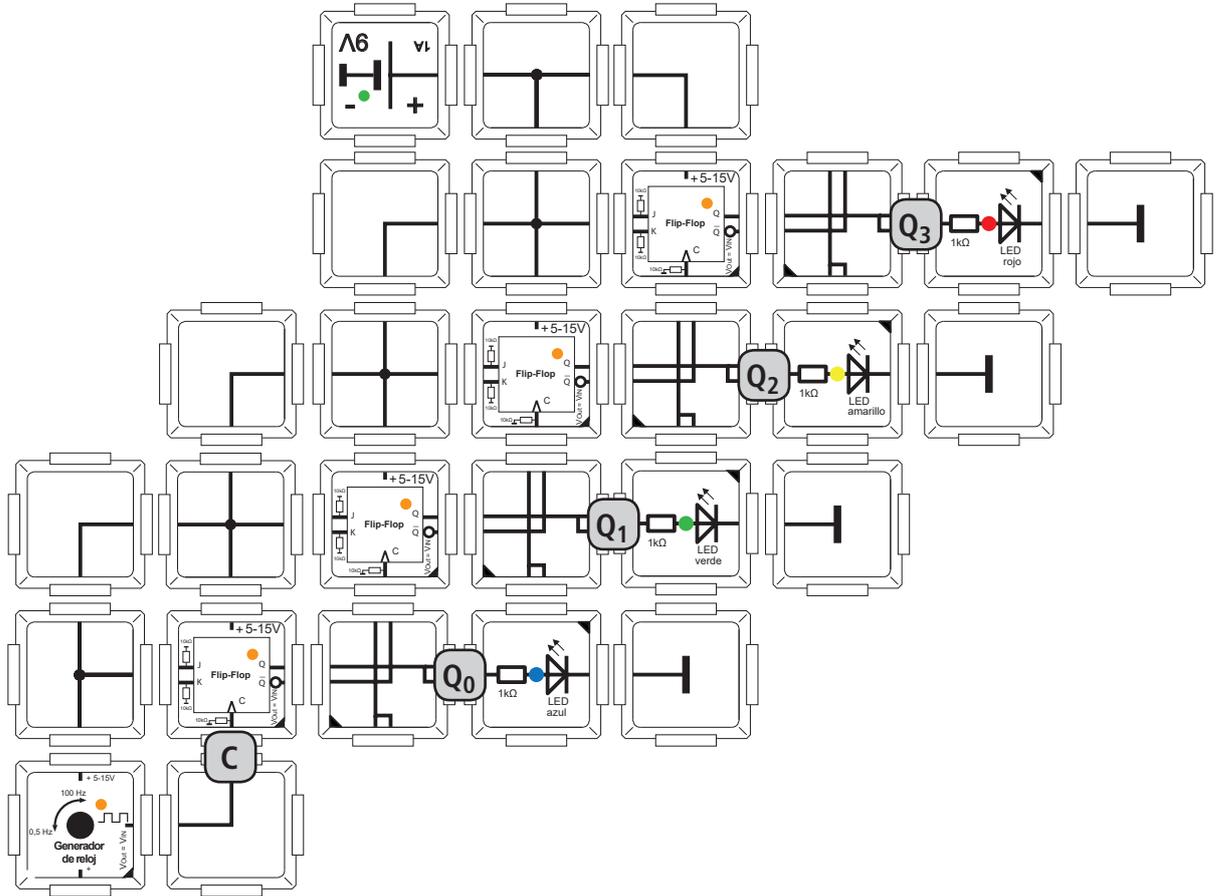


Figura 46: Contador asíncrono binario de 4 bits ascendente construido con flip flops como circuitos de bricks

El diagrama de tiempo que se muestra a continuación muestra un contador binario de subida de 4 bits que cuenta de 0 a 15 y luego comienza de nuevo desde 0. Por favor, tenga en cuenta que necesita una señal de rebote en la entrada de reloj C. Alternativamente, puede utilizar el botón pulsador de rebotes para entender mejor los pasos de conteo.

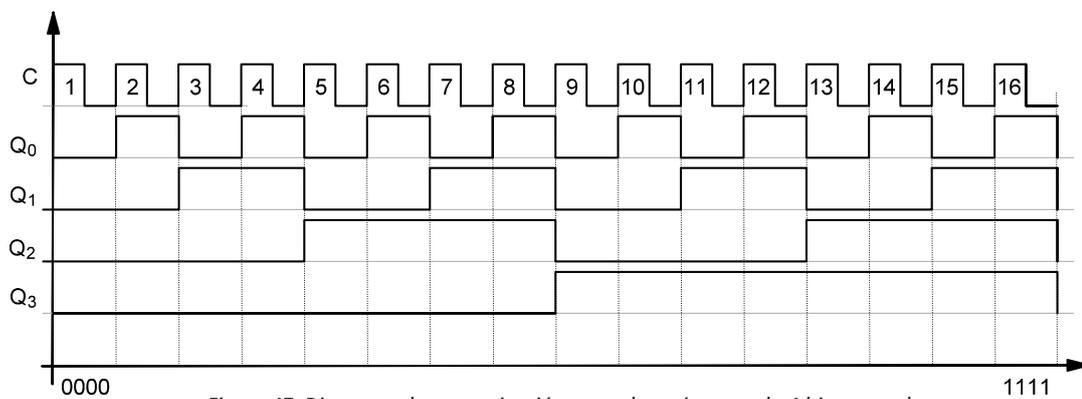


Figure 47: Diagrama de temporización contador asíncrono de 4 bits ascendente.

### 5.6.1.2 Contador asíncrono descendente

El siguiente circuito de bricks muestra un contador asíncrono descendente. El proceso de conteo puede leerse en el LED naranja incorporado en el flip-flop, que está conectado a la salida Q (véase también el diagrama de temporización Fig. 49). Las etapas de flip-flop se conectan en serie conectando la salida Q<sub>0</sub> de la primera etapa a la entrada de reloj C de la segunda etapa, y así sucesivamente. Como utilizamos flip-flops JK controlados por un solo flanco, la transición del estado de reset al estado de set provoca un flanco positivo en la salida Q. Dado que tanto la entrada J como la entrada K están conectadas a un nivel High, esto hace que el siguiente flip-flop se conmute.

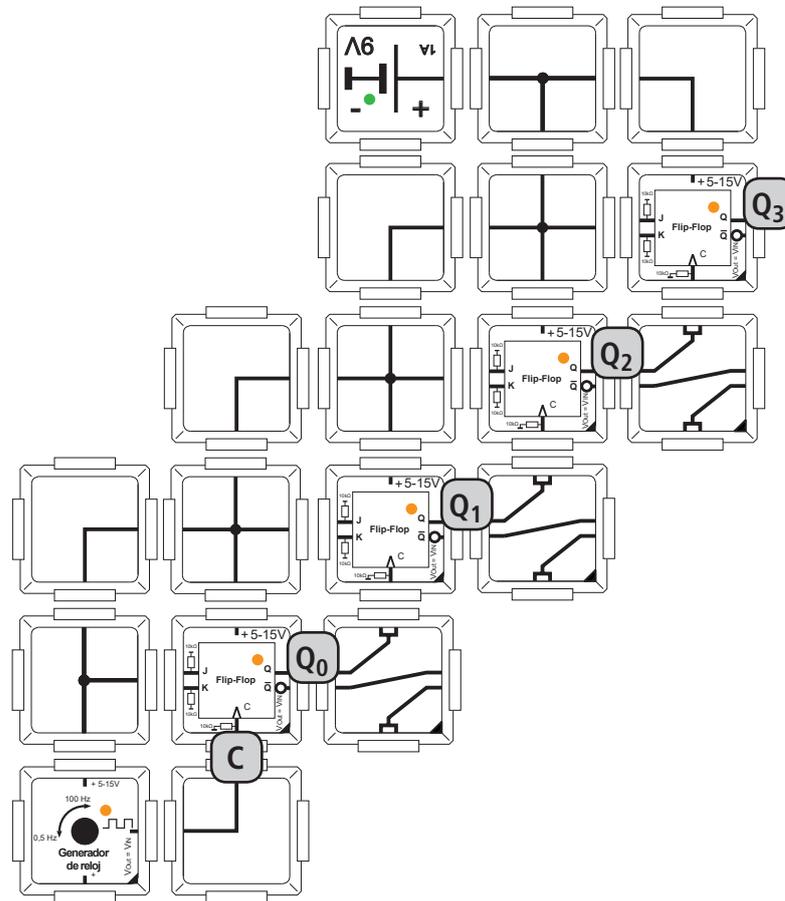


Figura 48: Contador asíncrono binario de 4 bits descendente construido con flip-flops JK como un circuito brick.

El diagrama de temporización que se muestra a continuación muestra un contador binario de 4 bits que cuenta de 15 a 0 y luego comienza de nuevo a partir de 15. Tenga en cuenta que la entrada de reloj C debe estar cableada sin rebote. Usando el brick de reloj, esto siempre está garantizado. Alternativamente, puede utilizar el botón pulsador de rebotes para entender mejor los pasos de conteo.

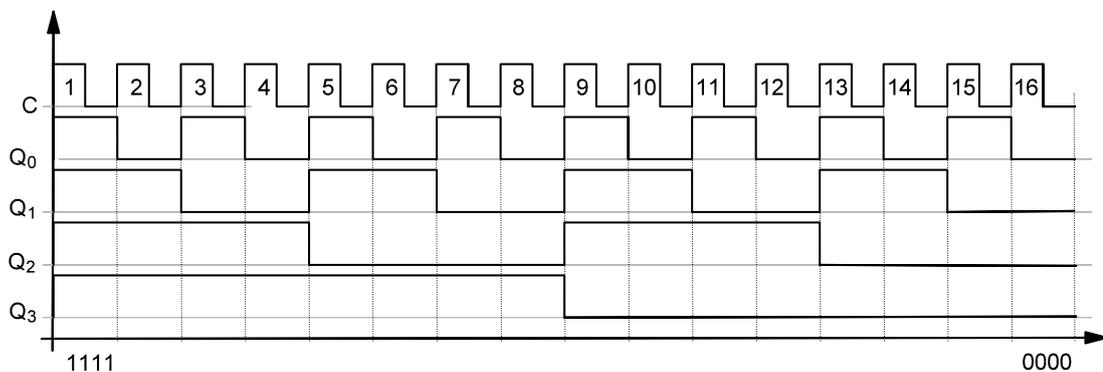
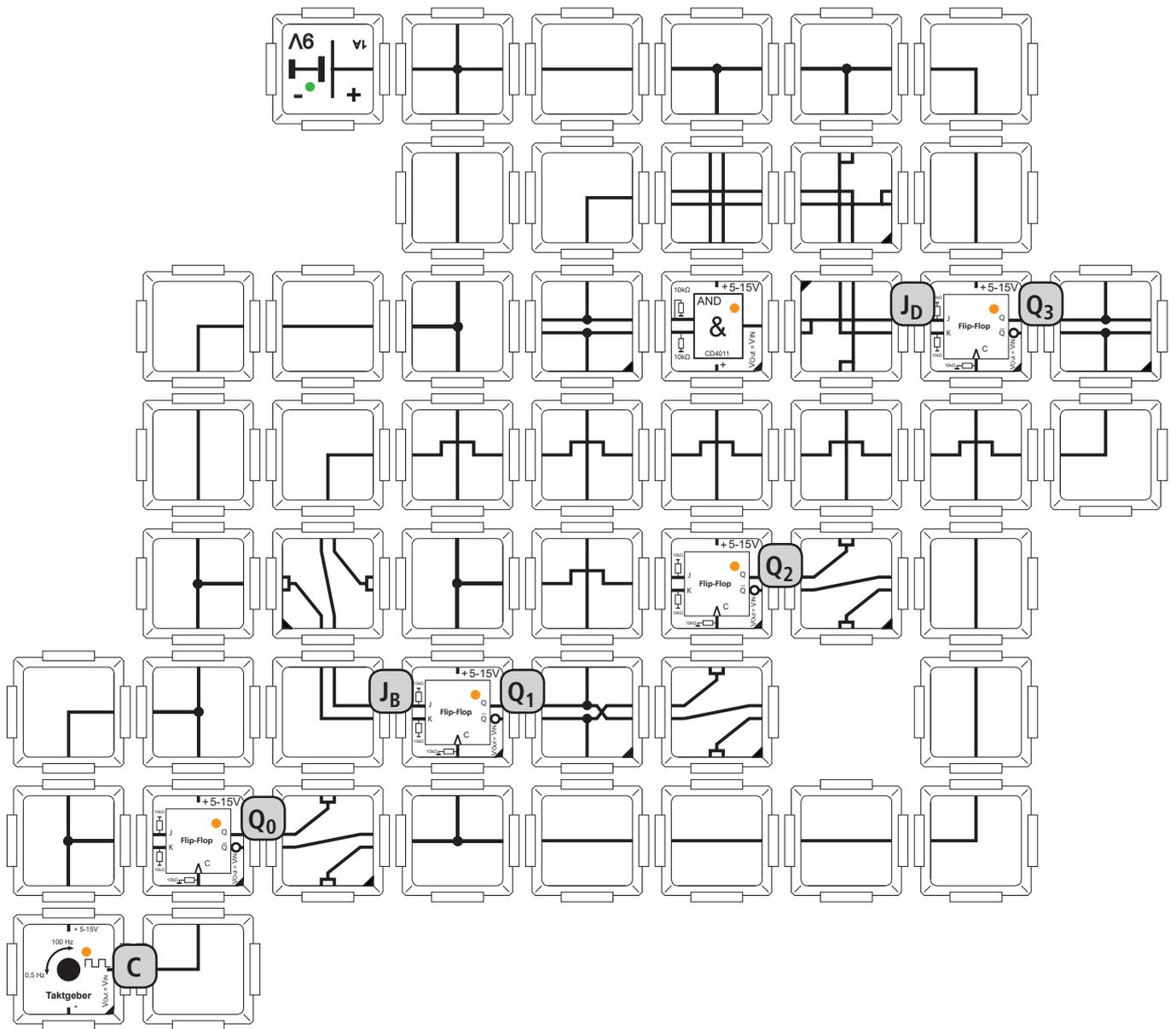


Figura 49: Diagrama de temporización contador asíncrono de 4 bits descendente.





Contador BCD asíncrono de 4 bits construido con flip-flops como circuito de bricks

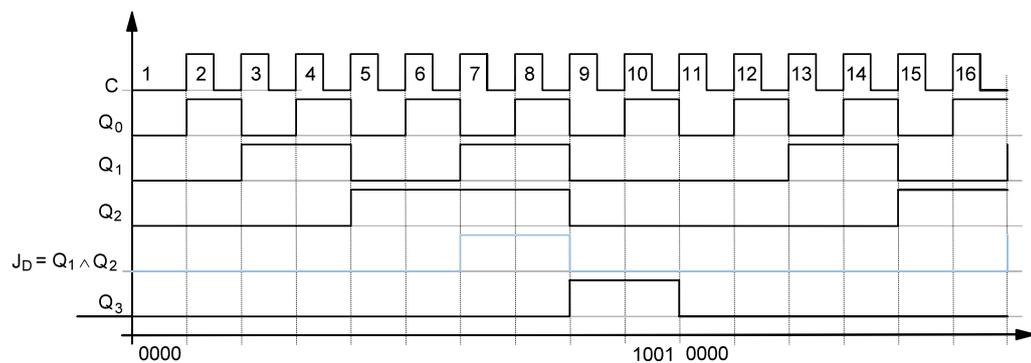


Figura 52: Diagrama de temporización de un contador BCD asíncrono de 4 bits



Tenga en cuenta que no debe conectarse ninguna carga adicional, por ejemplo un LED, a la entrada C del reloj de los flip-flop JK.

### 5.6.3 Contador síncrono binario de 3 bits

Los contadores síncronos tienen una estructura más clara y son fácilmente ampliables. Su circuito combinatorio es más complejo, pero no hay problemas de tiempo de ejecución.

Cuando el pulso cambia, los niveles de entrada de cada paso de memoria deciden si el flip-flop se restablece o descansa en su estado anterior. Puesto que toda la información en sus entradas tiene que ser ajustada al principio del pulso del reloj, los contadores síncronos no pueden ser contruidos con flip-flops T. Los flip-flop RS también son adecuados de forma limitada porque el esfuerzo del circuito es demasiado grande debido al circuito adicional.

El flip-flop más adecuado es, por lo tanto, el flip-flop JK-Master-Slave, cuyas entradas J y K pueden controlarse a través de puertas adicionales. En nuestro ejemplo usamos un flip-flop JK controlado por un solo flanco. Las entradas J y K del primer flip-flop tienen un nivel High que equivale a la operación de conmutación. La salida controla las entradas J y K conectadas de la etapa siguiente. Las entradas J y K de todos los flip-flop siguientes están conectadas con la puerta AND que interpreta los niveles de salida del paso previo de memoria.

El siguiente ejemplo para un contador síncronico, tiene sólo 3 bits de ancho. De esta manera es posible contar hasta 7.

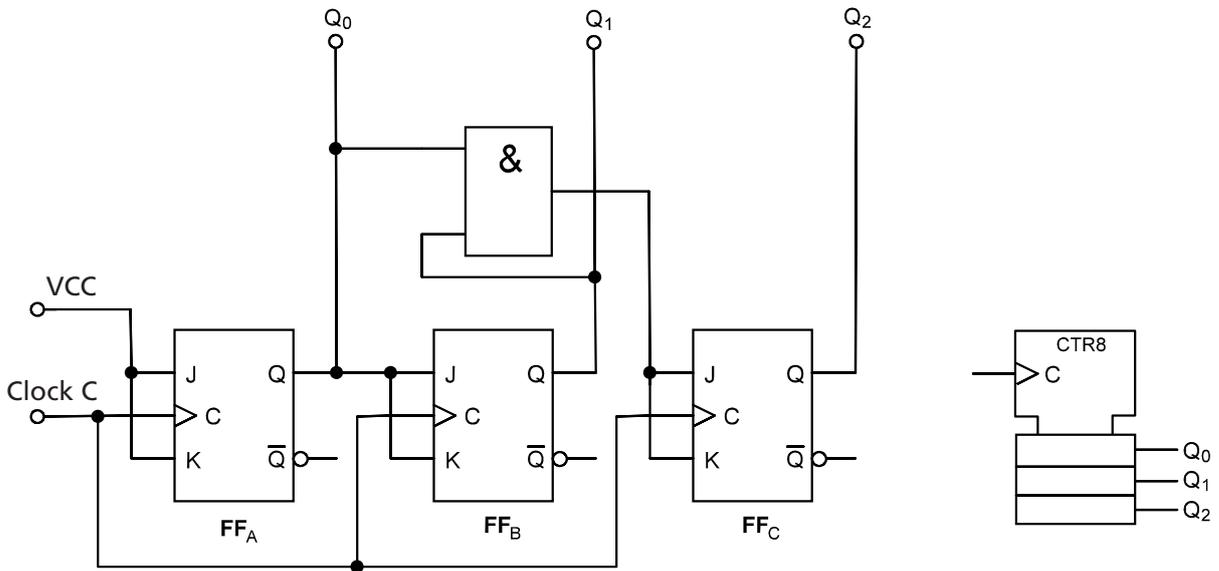


Figura 53: Contador síncrono binario de 3 bits ascendente (izquierda), símbolo de circuito (derecha)

Tabla de estados para un contador síncrono ascendente de 3 bits

| Reloj | Q <sub>2</sub> | Q <sub>1</sub> | Q <sub>0</sub> | Ecuación         |
|-------|----------------|----------------|----------------|------------------|
| 1     | 0              | 0              | 0              |                  |
| 2     | 0              | 0              | 1              |                  |
| 3     | 0              | 1              | 0              |                  |
| 4     | 0              | 1              | 1              | $Q_0 \wedge Q_1$ |
| 5     | 1              | 0              | 0              |                  |
| 6     | 1              | 0              | 1              |                  |
| 7     | 1              | 1              | 0              |                  |
| 8     | 1              | 1              | 1              |                  |

El contador tiene tres flip-flops de memoria y puede contar binarios desde 000 hasta 111. Con ayuda de la tabla de estados de las salidas Q<sub>0</sub> a Q<sub>2</sub>, se puede explicar el circuito de la puerta. El Flip-flop A cambia en cada pulso, lo que resulta en una reducción a la mitad de la frecuencia del reloj. El Flip-flop B sólo se ajusta de nuevo si la salida Q<sub>0</sub> está a nivel lógico High, de lo contrario se guarda el estado anterior. El Flip-flop C debe ajustarse en cuanto Q<sub>0</sub> y Q<sub>1</sub> tengan un nivel High al mismo tiempo.

Los nuevos niveles de salida aparecen después del flanco ascendente del reloj. Todas las señales de salida se ajustan sincrónicamente, es decir, al mismo tiempo. En comparación con los sumadores asíncronos, los retardos de señal de los pasos de memoria individuales no se acumulan.



**Tenga en cuenta que no debe conectarse ninguna carga adicional, por ejemplo, un LED a la entrada C del reloj de los flip-flops JK.**

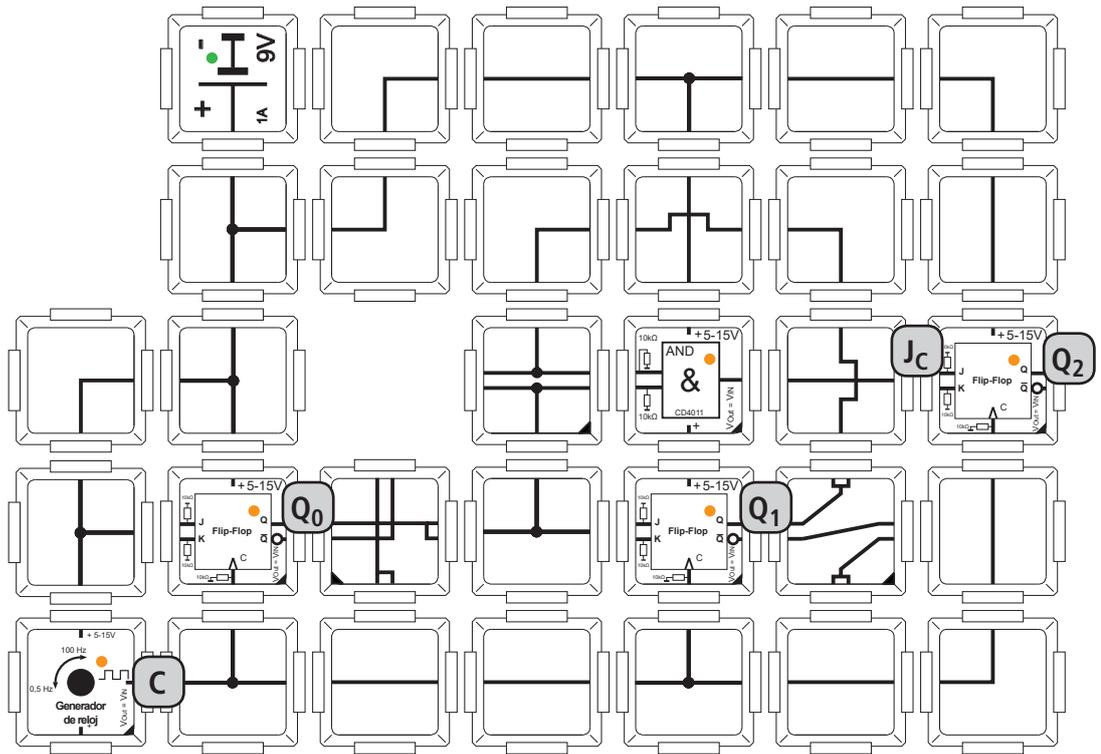


Figura 54: Contador sincronico binario de 3 bits con flip flop JK como circuito de brick

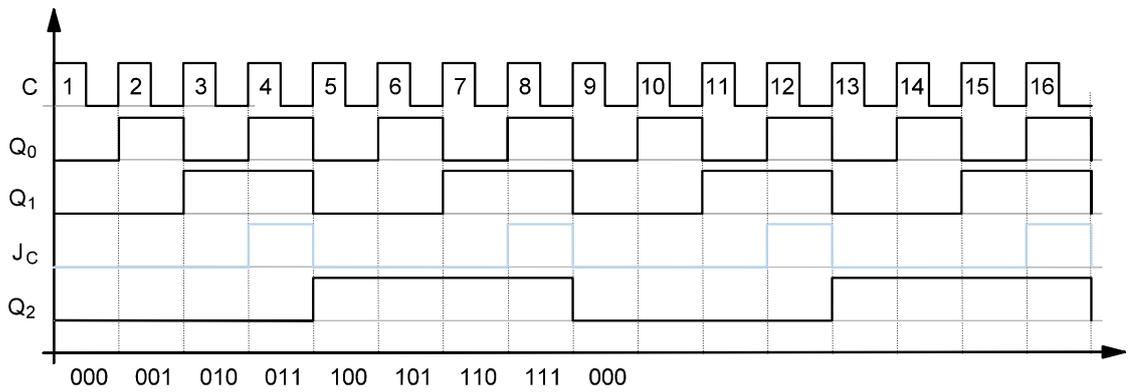


Figura 55: Diagrama de temporización de un contador síncrono binario ascendente de 3 bit

**Alternativa: Contador descendente**

Añadiendo LED en las salidas  $\bar{Q}$ , se puede transformar el contador en un contador descendente.

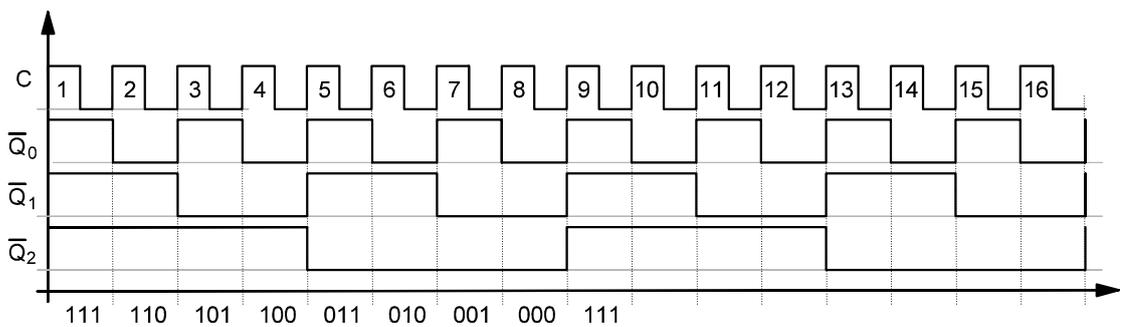


Figura 56: Diagrama de temporización de un contador síncrono binario descendente de 3 bits

### 5.6.4 Contador BCD con ejecución y reset

En este ejemplo usamos por primera vez el contador BCD de Brick'R'knowledge. Internamente están contruidos con dos contadores BCD síncronos que se conectan asincrónicamente mediante un bit de transmisión. Al conectarlos en serie, puede contar hasta números de varios dígitos. La decodificación de la pantalla de BCD a 7 segmentos ya está integrada en el contador BCD.

La siguiente figura muestra los cables de un contador de BCD con dos posibilidades de pulsación diferentes. En el lado izquierdo se puede ver la versión con un botón de supresión de rebotes y en el lado derecho la versión con el reloj que puede implementar frecuencias de hasta 100 Hz.

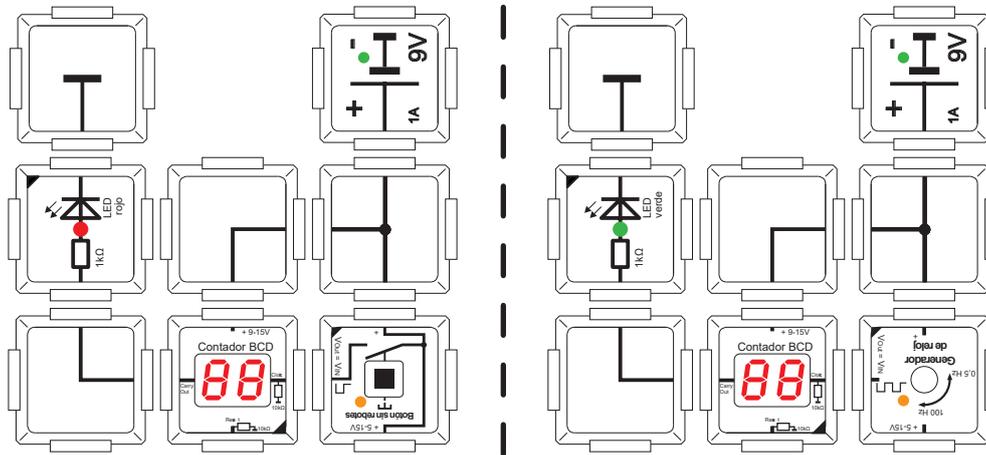


Figura 57: Contador BCD con pulsador de rebotes (izquierda) y reloj (derecha)

En nuestro ejemplo construiremos un contador BCD de 4 dígitos conectando en cascada dos de los bricks de contador BCD. La señal de Reloj y las señales de transmisión (Carry 1 y Carry 2) son visualizadas por los LED. Un reset asíncrono ("high active") es posible a través de un pulsador. El diagrama de bloques de nuestro circuito es el siguiente.

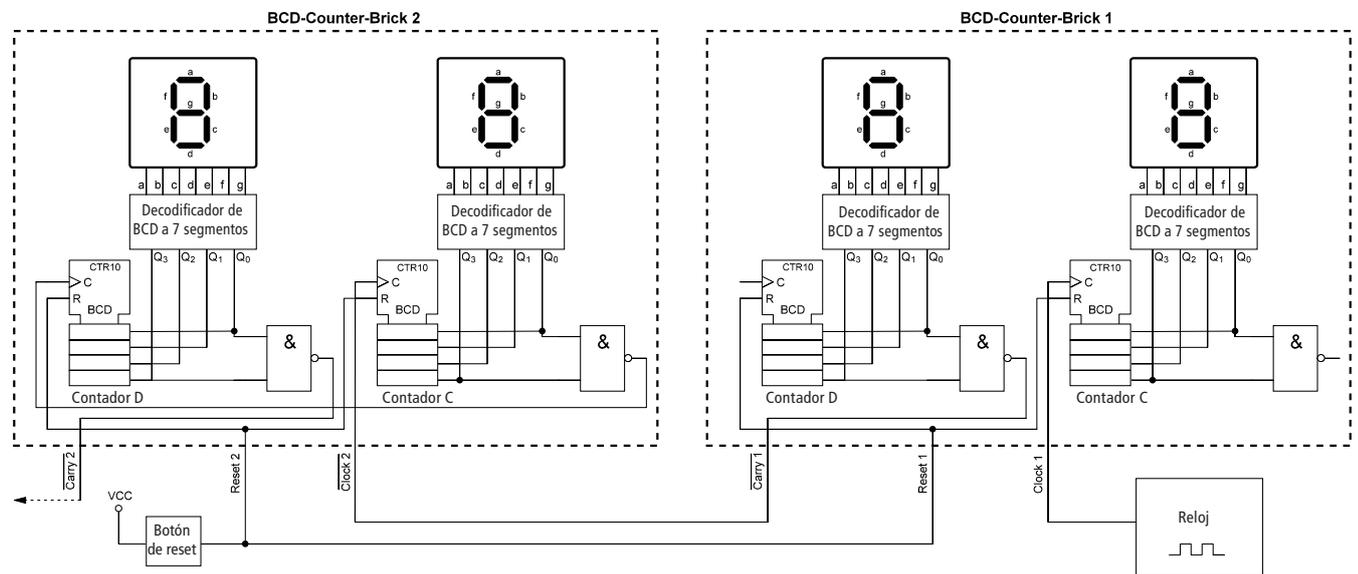


Figura 58: Diagrama de bloques de un contador BCD de 4 dígitos

En cuanto los bricks del contador BCD alcanzan el número máximo de 99 se activan los "carry out". Dado que esta señal es "low active", debemos escribir correctamente "Carry Out" para mostrar la negación. Para poder distinguir entre las dos salidas "carry out" del circuito de arriba, escribimos Carry1 y Carry 2.

La señal Carry Out es en el caso de nuestro contador BCD de 10 pulsos de duración. Esto significa que la realización se anunciará ya en el número 90. Puede ver este aviso por la salida porque cambia a "Low" y vuelve a "High" (es decir, no activo) cuando el número cambia de 99 a 00.

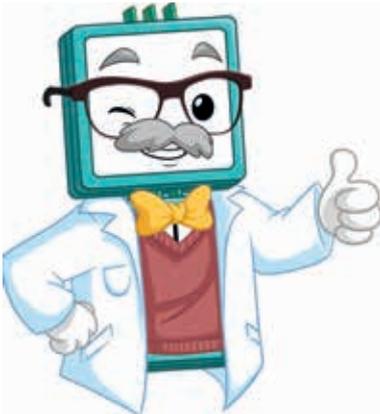
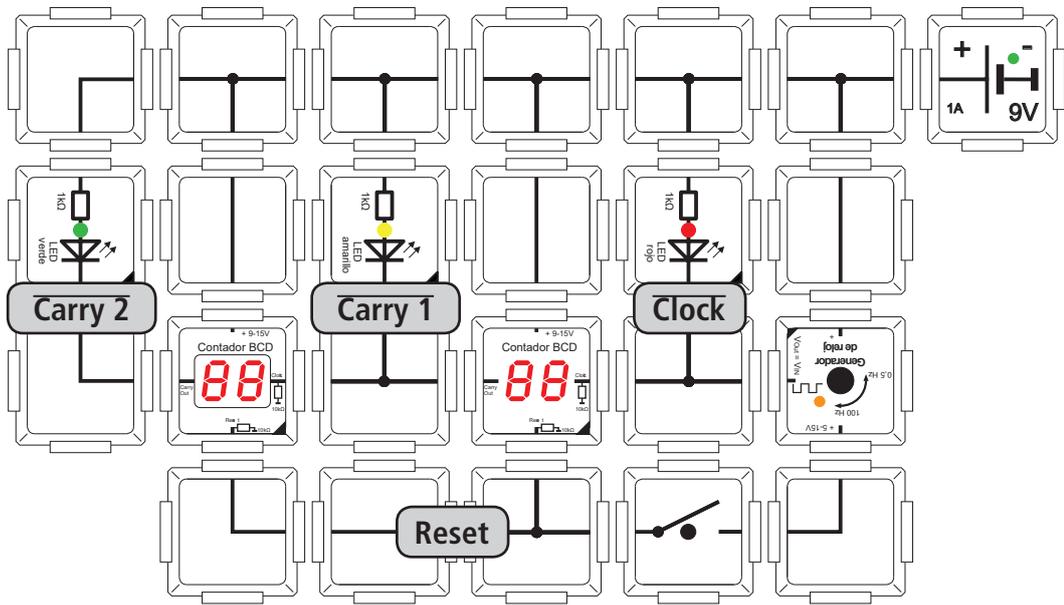


Figura 59: Contador BCD de 4 dígitos con ejecución de salidas y entradas de reset

## 5.7 Divisor de frecuencia

Los divisores de frecuencia son circuitos que dividen las frecuencias de señales a una cierta ratio de división. La mayoría de divisores de frecuencias tienen una ratio de división integral fija. Aparte de esto, puedes encontrar divisores de frecuencia ajustables que tienen una entrada adicional que establece la ratio divisoria.

En principio, cada contador también es un divisor de frecuencia. Circuitos de ejemplo:

- Contador asincronico binario: vea más en el capítulo 5.6.1 en la página 46
- Contador sincronico binario: vea más en el capítulo 5.6.3 en la página 51

Un solo flip-flop alcanza un divisor de frecuencia a un ratio de 2 : 1 (salida  $\bar{Q}_0$ ). Con dos flip-flops puedes construir un divisor de frecuencia a una ratio de 4: 1 (salida  $\bar{Q}_1$ ), etc...

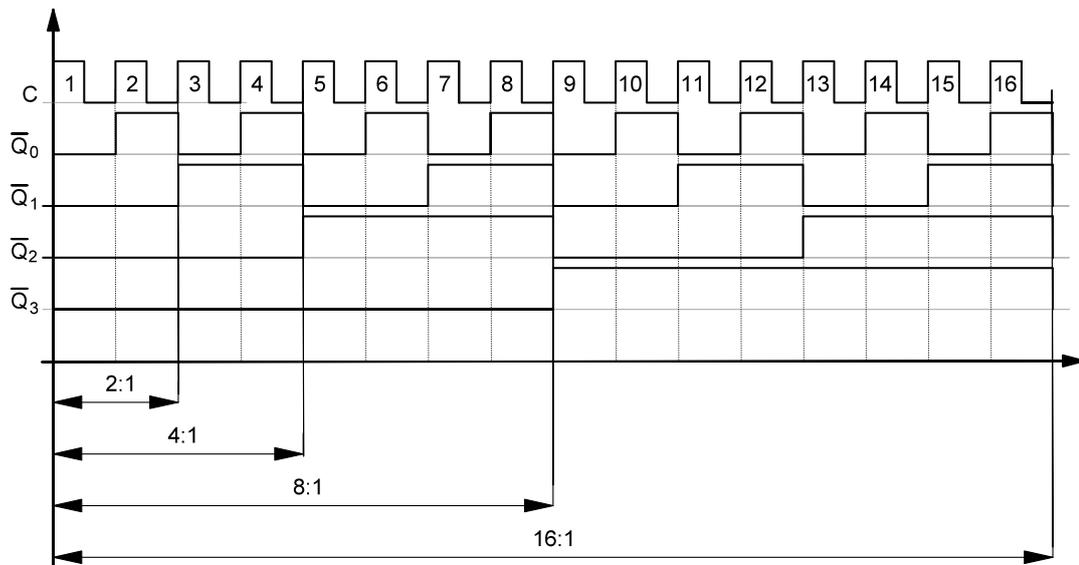


Figura 60: Diagrama de temporización de un frecuencímetro (véase también el contador binario de 4 bits del capítulo 5.6.1 en la página 46)

A un pulso de reloj de 100 Hz (Máxima frecuencia del brick de reloj) recibes las siguientes frecuencias en las salidas  $\bar{Q}$ :

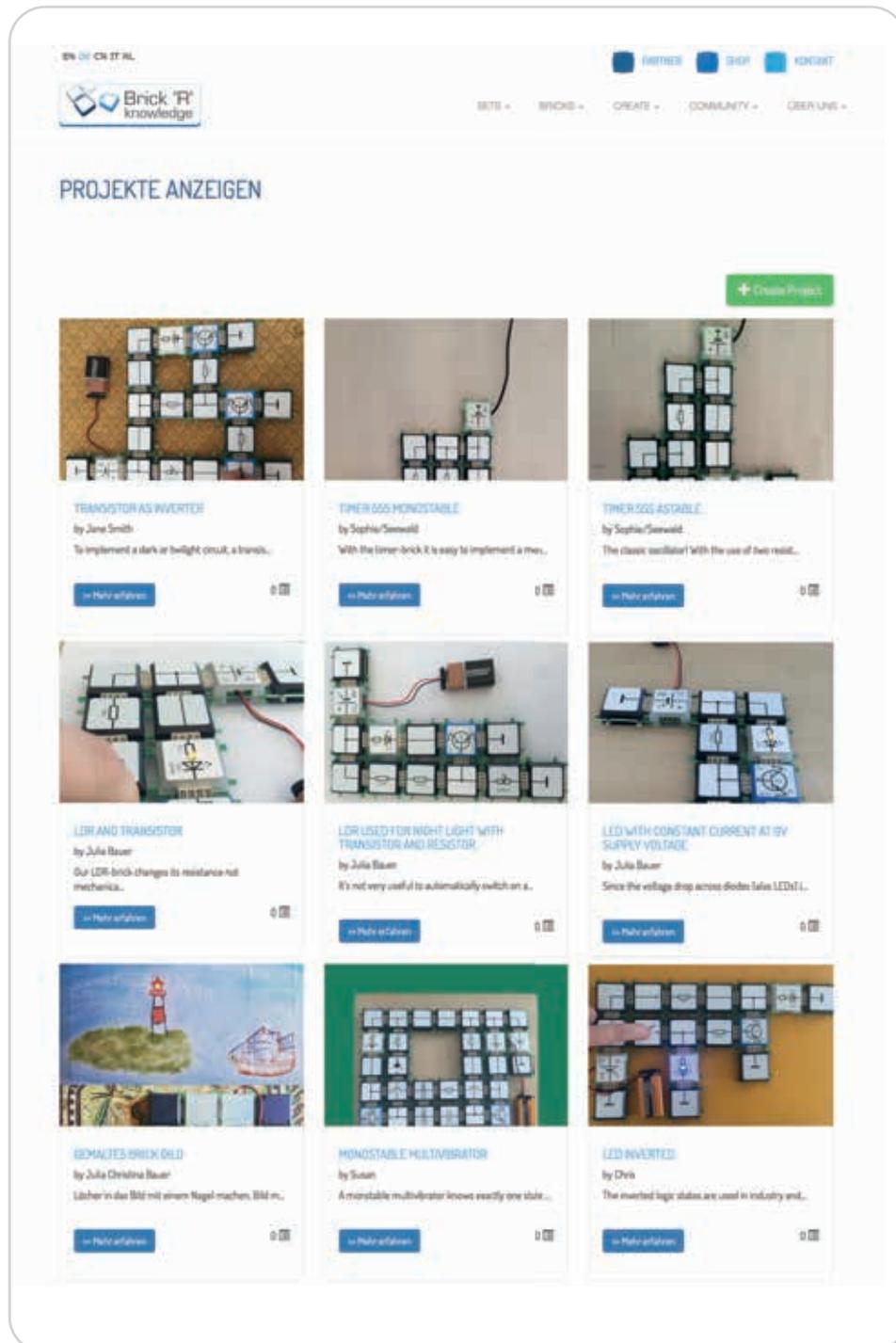
| Ratio      | 2:1                | 4:1                | 8:1                | 16:1               |
|------------|--------------------|--------------------|--------------------|--------------------|
| Salida     | Salida $\bar{Q}_0$ | Salida $\bar{Q}_1$ | Salida $\bar{Q}_2$ | Salida $\bar{Q}_3$ |
| Frecuencia | 50 Hz              | 25 Hz              | 12,5 Hz            | 6,25 Hz            |

## 6. Comunidad brick

El universo del brick se está expandiendo: encuentra más ideas, experimentos y bricks en ferías, en nuestra página web, en YouTube o en redes sociales. Impulsa tu creatividad!

### Más proyectos

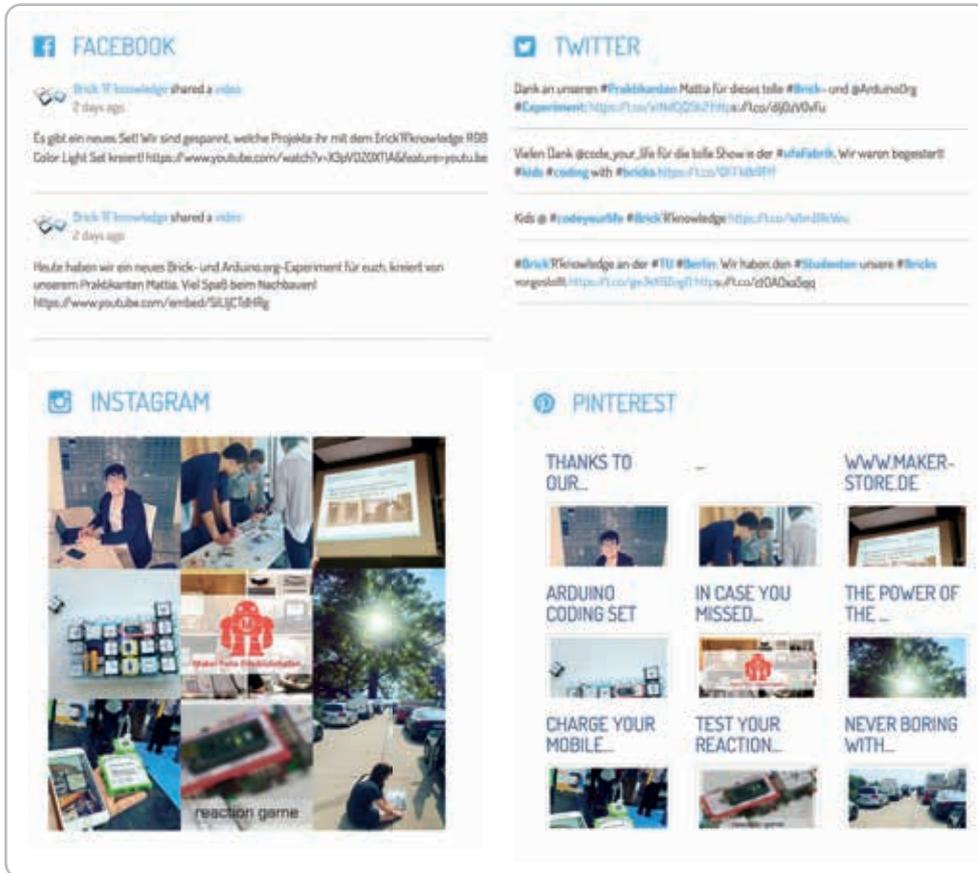
Haciendo clic en "Crear" puedes probar experimentos de otros usuarios o mostrar tus propios circuitos.





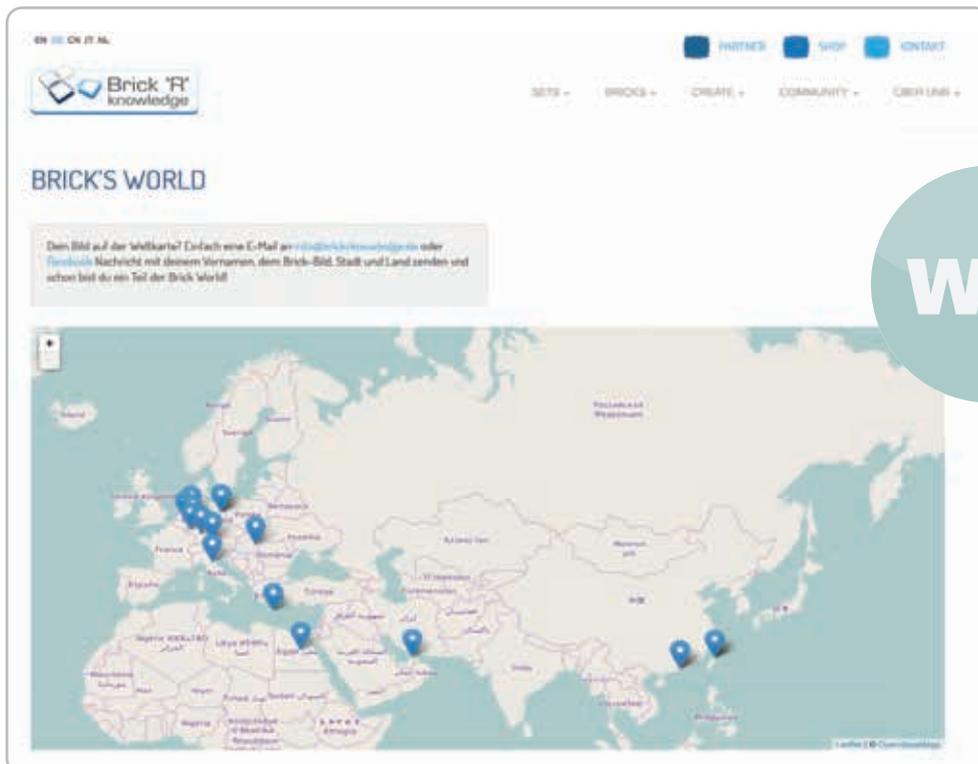
## Redes sociales

Haciendo clic en "Comunidad" puedes encontrar todas nuestras redes sociales. Mantente al día!

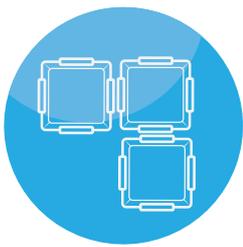


## Por todo el mundo

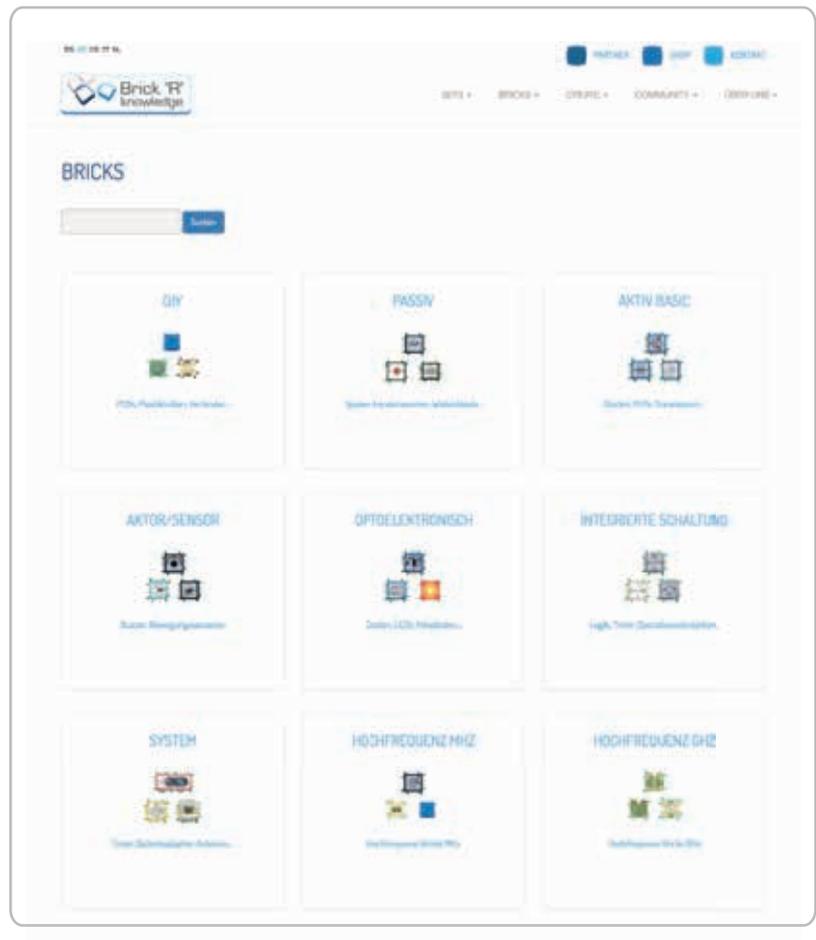
En "Comunidad" también puedes saber dónde están nuestros bricks. ¿Tienes una foto de bricks en tu ciudad? Envíanosla y pronto la encontrarás en nuestra página web!



## Más bricks!



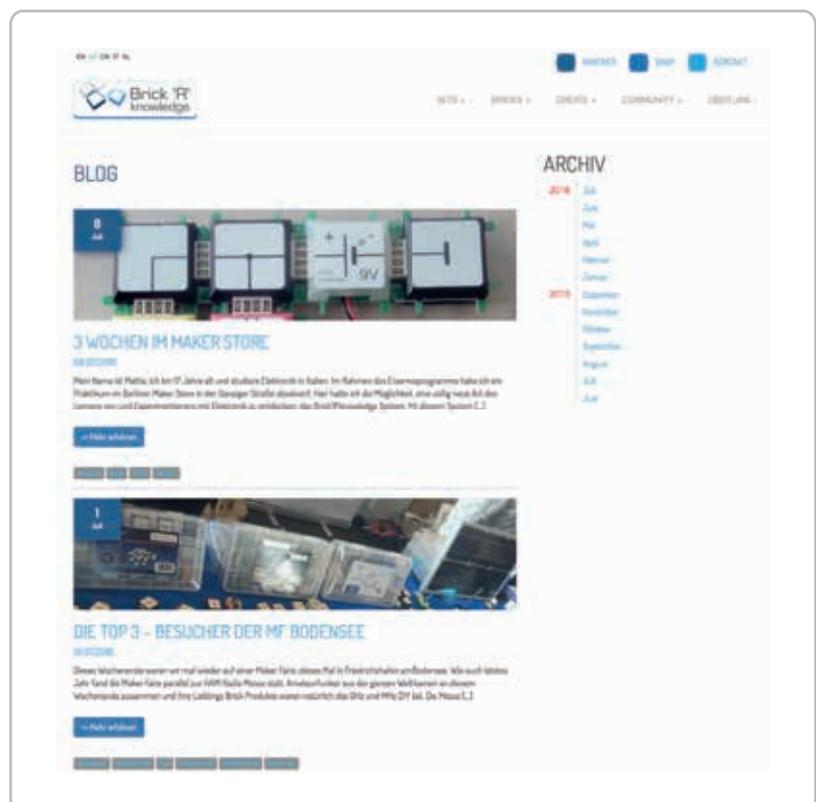
Haciendo clic en "Bricks" puedes encontrar todos los bricks disponibles con información e ideas para experimentos.



## Blog



Cada semana subimos un nuevo post en el blog. Puedes leer sobre nuestras experiencias en ferias, nuevos circuitos, divertidas historias e información sobre el mundo de la electrónica.



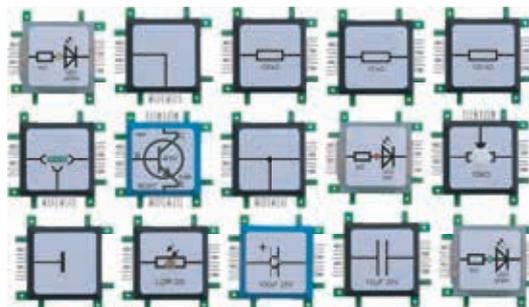
## 7. Resumen de los sets de bricks



### Set básico

ALL-BRICK-0374

El set básico contiene 19 bricks seleccionados para ofrecer un fácil y rápido comienzo en el mundo de Brick'R'Knowledge, así como la posibilidad de crear numerosos circuitos. El set básico es perfecto para que los niños adquieran sus primeras experiencias con experimentos electrónicos y técnicos.

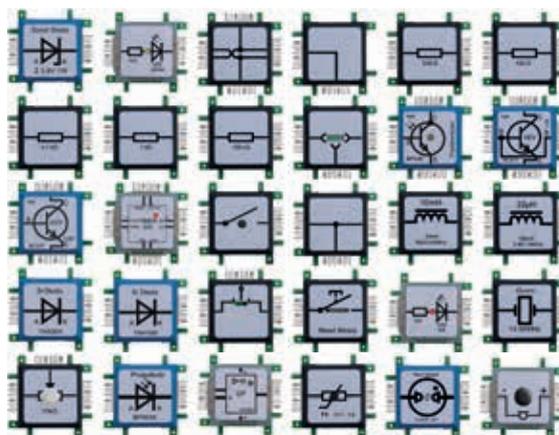


### Set avanzado

ALL-BRICK-0223

Nuestro set avanzado contiene 111 componentes que te permiten desarrollar soluciones más complejas y complicadas. Gracias al sistema educativo, el conocimiento puede ser recopilado, de manera que no sólo vosotros sino que también nuestra próxima generación pueda beneficiarse de ello. Puedes construir circuitos individuales conectando bricks diferentes juntos. Simple a la vez que complejo, se puede experimentar con temas electrónicos y tecnológicos de una manera totalmente nueva. A través del factor open-source, puedes crear tus propios bricks y desarrollar tus propias soluciones.

El Brick'R'Knowledge no se trata solamente sobre electrónica básica, también se pueden realizar experimentos RF, lo que lo convierte en un sistema único a nivel mundial.

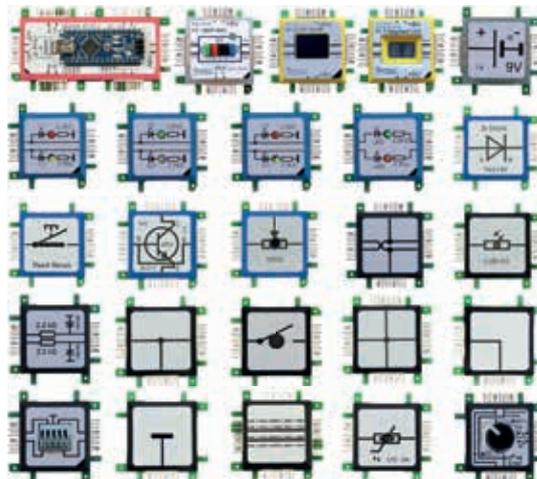




## Set de codificación de Arduino

ALL-BRICK-0414

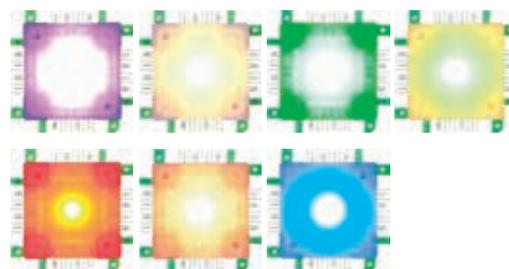
Entra en contacto con la electrónica digital y empieza a entender la programación con Arduino® Nano, que se incluye en el kit. Es nuestro primer set con componentes digitales, como pantallas de 7 segmentos, display OLED, convertidor A/D o bricks I2C, complementario a todos los bricks analógicos. Para empezar con el popular microcontrolador, te ayudamos con varios ejemplos de programación.



## Set de luces de 7 colores

ALL-BRICK-0398

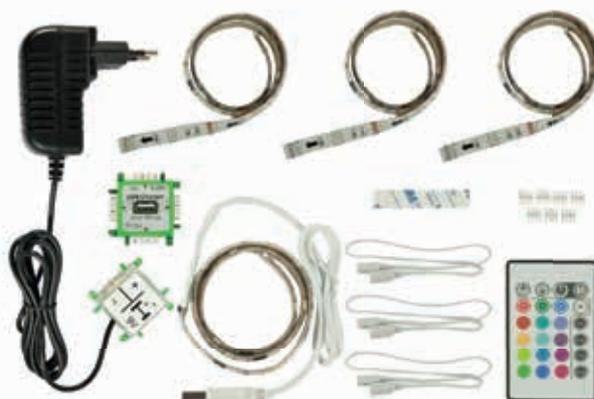
El set de luces de 7 colores contiene 28 bricks LED en 7 colores diferentes para crear impresionantes efectos de luz en una arquitectura horizontal y vertical. Los LED rojos, amarillos, azules, naranja, violetas, verdes y blancos cálidos de 1 vatio son perfectos para la iluminación individual o como solución a la iluminación móvil.

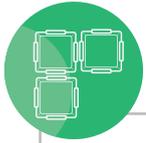


## Set RGB de luces de colores

ALL-BRICK-0619

Crea tu show de luces! El set RGB de luces de colores viene con cuatro tiras LED flexibles que contienen un total de 36 LED, los cuales se pueden controlar con el mando a distancia por infrarrojos incluido. Puedes pegar, cortar y conectar las tiras del LED como quieras. El mando a distancia por infrarrojos tiene 16 teclas de colores diferentes y 4 programas de luz.

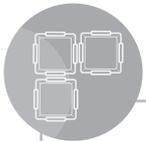
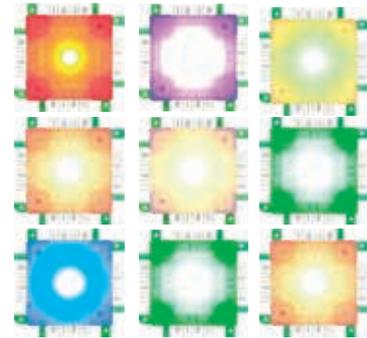




## Set de LED programable

ALL-BRICK-0483

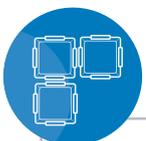
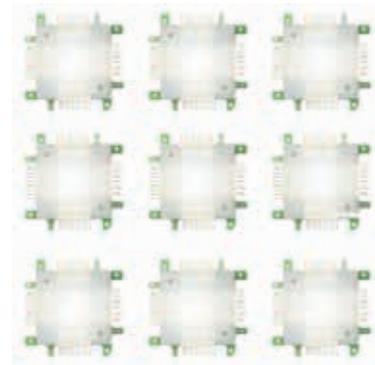
El kit contiene 49 bricks LED RGB programables y controlables, cada uno con dos o tres conectores y un brick de conjunción para la gestión de Arduino y la fuente de alimentación. Además, el conjunto de LED programables de Brick' R' knowledge incluye un brick adaptador de Arduino y un Arduino Nano. Con este set se pueden realizar animaciones de LED coloridas y otras ideas individuales. Y lo mejor de todo: con la realización de diferentes proyectos se puede aprender fácilmente la programación de microcontroladores.



## Set del LED de alta potencia

ALL-BRICK-0399

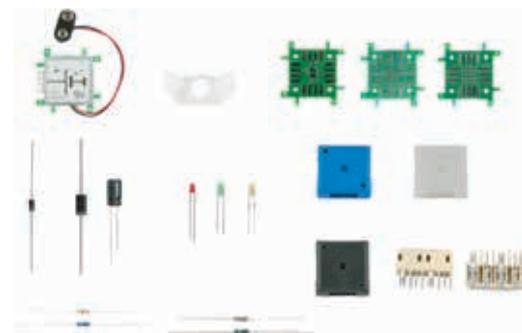
Potenciados por 1 vatio, cada uno de los bricks LED de alta potencia incluidos en el kit irradian toda la zona circundante en blanco brillante. Construye soluciones individuales en cada arquitectura imaginable e inventa luces de noche Brick, lámparas de mesa Brick o cualquier otro iluminante creativo. La fuente de alimentación con 12V 8A soporta la intensa luminosidad para ofrecer una atmósfera elegante y acogedora. El set de alta potencia permite tratar con un diseño de luz moderno y al mismo tiempo aprender sobre electrónica.

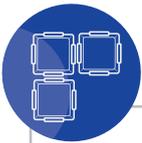


## Set DIY

ALL-BRICK-0397

El set de bricolaje va un paso más allá. Los componentes incluidos ofrecen una visión mucho más detallada de la arquitectura del brick y permiten incluso la producción de bricks individuales. El kit de bricolaje ofrece una enorme flexibilidad para la generación del fabricante o para las personas que crean bricks individuales.

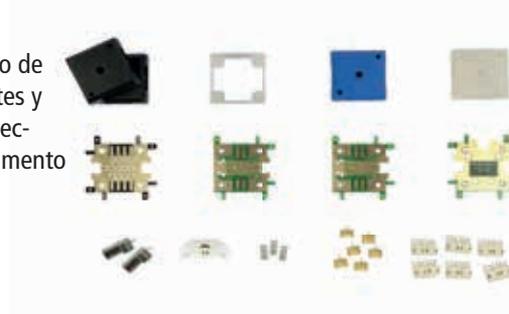




## Set DIY de MHz

ALL-BRICK-0457

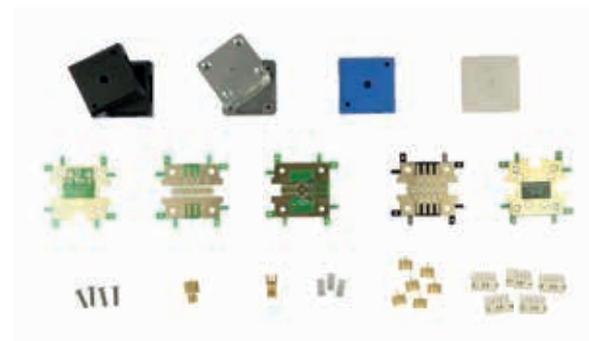
Se pueden crear proyectos individuales desafiantes dentro del rango de frecuencias de MHz con el sistema DIY de MHz. Tres rejillas diferentes y paneles de experimentación, tomas BNC, enchufes P-SMP y los conectores necesarios hacen que el kit sea perfecto para cualquier experimento de alta frecuencia. El set contiene conectores hermafroditas y una plantilla de soldadura para que los conectores SMD desarrollen sus propios bricks u otros componentes para el sistema Brick.



## Set DIY de GHz

ALL-BRICK-0458

Realice experimentos avanzados y complicados en el rango de alta frecuencia hasta frecuencias de GHz. El kit ofrece cuatro placas de circuito impreso diferentes, P-SMP, tomas SMA, conectores P-SMP y conectores hermafroditas específicos del brick. El conjunto DIY de GHz es perfecto para operadores de radio HAM y aficionados a la medición.



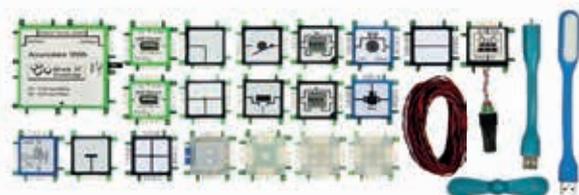
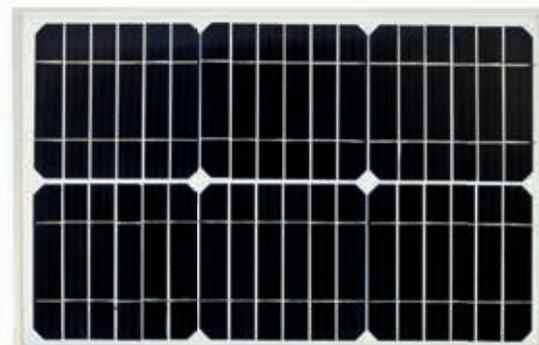
## Set solar

ALL-BRICK-0484

El set solar garantiza la diversión de experimentar en familia y acerca a los niños a las energías renovables de una manera lúdica.

- ¿Cómo funciona una célula solar?
- ¿Cómo almacena la energía una batería?
- ¿Cómo construir una luz nocturna con detector de movimiento?

El set solar proporciona respuestas a estas y otras preguntas. Con este set os convertís en miembros oficiales de la generación Maker!



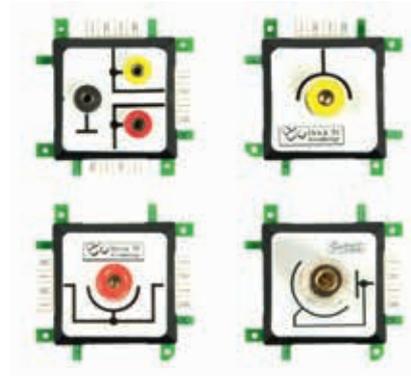


## Set de medición 1

contiene 4 Bricks

ALL-BRICK-0637

El set de medición 1 permite medir la tensión, la corriente y otras magnitudes con instrumentos de medición estándar. El set contiene adaptadores de medición (3x2mm), con abrazadera de cable adicional y adaptadores de medición (4mm) con un punto final amarillo.



## Set de medición 2

contiene 6 Bricks

ALL-BRICK-0638

El set de medición 2 permite medir la tensión, la corriente y otras magnitudes con instrumentos de medición estándar. Contiene adaptadores de medición (4mm) con extremo cerrado GND, adaptadores de medición (4mm) rojo en línea y adaptadores de medición (4mm) con extremo abierto GND negro.



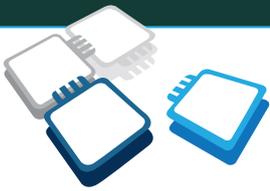
## Set Internet de las Cosas

contiene 17 Bricks

ALL-BRICK-0646

Con el Internet de las Cosas programado, es posible configurar los bricks a través de internet. Con el brick IdC aprenderá a construir su primera página web y a controlar los pines E/S con su smartphone. El set también contiene un sensor de temperatura y humedad, cuyos valores se pueden visualizar en una pantalla: El primer paso hacia la domótica!

También se pueden introducir datos, como la cotización del dólar desde internet y visualizarlo. Para controlar el display de 7 segmentos, se utiliza el conocido bus I<sup>2</sup>C, que pronto conocerá.



# Brick 'R' knowledge



ALLNET® GmbH Computersysteme  
Maistrasse 2  
D-82110 Germering  
[www.brickrknowledge.com](http://www.brickrknowledge.com)

Telefon: +49 (0)89 894 222  
Fax: +49 (0)89 894 222 33  
[info@brickrknowledge.com](mailto:info@brickrknowledge.com)



**Maker Store Berlin**  
Prenzlauer Allee 173  
D-10409 Berlin  
[www.maker-store.de](http://www.maker-store.de)

Telefon: +49 (0)30 473 756 80  
[service@allknow.de](mailto:service@allknow.de)