

Funciones lógicas: tabla de verdad

Apellidos, nombre	Martí Campoy, Antonio (amarti@disca.upv.es)
Departamento	Departament d'Informàtica de Sistemes i Computadors (DISCA)
Centro	Escola Tècnica Superior d'Enginyeria Informàtica (ETSINF)

1 Resumen de las ideas clave

En este artículo trabajarás con funciones lógicas, tablas de verdad, circuitos combinacionales y otros términos relacionados. Todo ello es necesario para diseñar y construir con éxito un circuito digital. El único conocimiento previo que necesitas es saber representar números en el sistema binario.

2 Introducción

El primer paso en el diseño y construcción de un circuito digital es determinar con exactitud qué es lo que queremos que ese circuito haga. Obvio, ¿verdad?

El problema es que el lenguaje humano no está libre de ambigüedades y una frase puede admitir más de una interpretación. Por ello existen diferentes herramientas que nos permitirán especificar, sin ninguna ambigüedad, que es lo que el circuito debe hacer.

Se utiliza el término función lógica para referirse al comportamiento de un circuito digital combinacional, y existen varias formas de representar, de expresar, una función lógica. En este artículo veremos una de estas formas, la tabla de verdad.

Antes de empezar quiero darte una definición, bueno, en realidad serán dos.

Decimos que un circuito es **combinacional** cuando su salida en un instante dado depende únicamente del valor de las entradas en ese instante. Por ejemplo, un circuito que realiza la suma de dos números es combinacional, porque la salida (el resultado) depende sólo de los operandos que hay en cada momento, y no de los operandos ni resultados anteriores.

Por otro lado, decimos que un circuito es **secuencial** si su salida en un instante dado depende tanto de las entradas actuales como de las anteriores. Por ejemplo, el monedero de una máquina de café, ya que cuando metes monedas lo que aparece en la pantalla no es la última moneda, sino la suma de las monedas que se han introducido anteriormente.

Ahora que ya sabes que hay dos tipos de circuitos digitales, ten presente que todo lo que vamos a trabajar en este artículo se refiere sólo a circuitos combinacionales.

3 Objetivos

Una vez hayas terminado de leer este documento, repetir los ejemplos y resolver los ejercicios propuestos, serás capaz de:

- Representar una función lógica mediante su tabla de verdad.
- Representar una función lógica con entradas indiferentes mediante su tabla de verdad.
- Representar gráficamente la interfaz de un circuito digital.
- Todo lo anterior trabajando con sistemas combinacionales.

4 Función lógica

Esta es la definición de función lógica:

“Representación formal del comportamiento de un circuito digital.”

Hay dos cosas muy importantes en esta definición. La primera es que una función lógica es cualquier forma de **decir** (expresar, representar) lo que queremos que haga un circuito. Puede ser un dibujo, un diagrama, una tabla, una lista... La segunda cosa es que tiene que ser **formal**. Es decir, tiene que haber unas reglas que nos digan cómo hemos de representar el comportamiento del circuito.

Además de la definición de función lógica, hay dos definiciones más que necesitas conocer:

- **Aridad:** Es el número de variables lógicas de entrada de la función lógica, o del circuito, que es lo mismo.
- **Valoración:** es una las posibles combinaciones de valores de las entradas. Por tanto, una función lógica tendrá 2^{aridad} valoraciones.

Veamos un ejemplo. “Queremos hacer un circuito con tres entradas llamadas A, B y C, y una salida llamada F, y que la salida se active si el número de entradas a 1 es impar”.

Vamos por partes. ¿Sabrías decirme cuál es la aridad de esta función? Correcto, es 3, ya que hemos dicho que tenemos tres entradas. ¿Y cuántas valoraciones tendrá esta función? Pues 2^3 valoraciones, que son 8.

Y por último, ¿la descripción que te he dado te parece formal y sin ambigüedades? Yo diría que no, ya que ¿cuál debe ser la salida si no hay ninguna entrada a 1?

No importa si decidimos que cero entradas a 1 es par o impar, lo importante es que la definición que te he dado no especifica completamente el comportamiento del circuito. En los siguientes apartados verás una herramienta para representar funciones lógicas sin equívocos ni ambigüedades.

4.1 Tabla de verdad

La tabla de verdad es la representación de una función lógica en forma de tabla. Hay unas reglas sencillas, pero estrictas, para construir esta tabla.

- La tabla tiene tantas columnas como entradas y salidas tenga función lógica.
- La tabla tiene tantas filas como valoraciones tenga la función lógica. Recuerda, el número de valoraciones es dos elevado a la aridad (número de entradas).
- A cada valoración se le asigna una fila.
- Las valoraciones se ordenan siguiendo la numeración binaria, empezando por la que representa al cero.
- Las entradas se ponen a la izquierda, y las salidas a la derecha.
- La tabla de verdad de una función es única. Es decir, para una función lógica sólo existe una tabla de verdad.

Aquí tienes un ejemplo. Queremos hacer un circuito que encienda la luz interior de un coche si alguna de las puertas, o las dos puertas del coche están abiertas. Este circuito tiene dos entradas llamadas I (de izquierda) y D (de derecha) que valen 1 cuando la puerta correspondiente está abierta y 0 en caso contrario. También tiene una entrada llamada M (de manual) que si vale 0, la luz se encenderá en función de las puertas. Por el contrario, si la entrada M vale 1 se encenderá la luz independientemente de si las puertas están abiertas o cerradas. Por último el circuito tiene una salida L (de luz) que valdrá 1 si hay que encender la luz y 0 si hay que dejar la luz apagada.

Para empezar a construir la tabla nos hacemos la pregunta siguiente: ¿cuántas columnas y filas tendrá la tabla? La respuesta es fácil: tres entradas más una salida igual a cuatro columnas, y cómo la aridad es 3, el número de valoraciones es $2^3 = 8$ por lo que tendremos 8 filas. ¡Perfecto! Ya podemos dibujar la tabla (Tabla 1).

Tabla 1. Estructura de una tabla de verdad para una función de aridad 3 y una salida.

Ahora ponemos las entradas a la izquierda, y las salidas, en este caso la salida, a la derecha. **Importante.** ¿Qué orden utilizamos para poner las entradas? El orden es fundamental, ya que le da sentido a las valoraciones. No es lo mismo 001_2 que 100_2 por lo que hay que decidir si vamos a escribir M, I y D o D, I y M. Cualquier combinación es válida, y puedes elegir la que más te guste o te resulte más cómoda. Pero una vez has elegido una ordenación, no puedes cambiarla. En este documento yo te sugiero utilizar M, I y D. De esta forma la tabla quedaría como ves en la Tabla 2:

Tabla 2. Tabla de verdad con las entradas M, I, D y la salida L.

M	I	D	L

El siguiente paso es escribir las valoraciones, en binario, empezando desde 0, considerando que la variable de la derecha, es decir la entrada D, es la de menor peso. De este modo, la tabla quedaría cómo ves a continuación en la Tabla 3.

Tabla 3. Tabla de verdad con las valoraciones.

M	I	D	L
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Por último queda asignar el valor de la salida para cada valoración. En ese caso el enunciado es claro y el funcionamiento de este circuito es bien conocido, por lo que no es difícil completar la tabla como puedes ver en la Tabla 4.

Tabla 4. Tabla de verdad para la función lógica L.

M	I	D	L
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

4.1.1 Ejercicio

Vamos a retomar el ejercicio del detector de paridad, pero usando cuatro entradas. La descripción, en lenguaje natural sería: "Queremos construir un circuito que dadas cuatro entradas llamadas D, C, B y A nos diga si el número de entradas que tienen valor 1 es **impar**, poniendo a 1 una salida llamada I." Como hemos hablado antes, nos surge la duda de si ninguna entrada a 1 es par o impar. En este caso, decidimos que si ninguna entrada está activa hay un número par¹ de entradas activas.

Por favor, coge papel y lápiz (o el equivalente digital) e intenta construir la tabla de verdad de la función lógica I. Te sugiero que uses el orden D, C, B, A para las entradas. Tienes la solución en la sección 5.

4.2 Entradas indiferentes

Te he insistido mucho en que en una función lógica es necesario que el valor de la salida esté perfectamente definido para cada una de las valoraciones.

Sin embargo, podemos encontrar situaciones reales en la que una determinada combinación de entradas es imposible. O simplemente, nos da igual cuál será el valor de la salida. En estos casos decimos que tenemos entradas indiferentes, y lo indicaremos en la tabla de verdad poniendo X en la salida. Ni 0 ni 1, ponemos una X.

¹ La detección de paridad se utiliza para detectar errores en la comunicación de datos digitales, y es habitual considerar que el valor cero tiene un número par de unos.

Vamos a ver un ejemplo muy sencillo. Vamos a construir la tabla de verdad de un circuito que nos diga si el nivel de un depósito de agua está por debajo de un nivel mínimo, por encima de un nivel máximo, o entre el mínimo y el máximo. Para ello el depósito tiene dos sensores de agua, uno a pocos centímetros de la base y otro a pocos centímetros del borde superior. Cuando el sensor detecta agua, tenemos un 1, y en caso contrario tenemos un cero. Las señales se llaman NI para el nivel inferior y NS para el nivel superior. El circuito tiene 3 salidas, B (bajo), M (medio) y S (Superior) que nos dirán con un 1 si el nivel del agua está por debajo del mínimo (salida B), entre el mínimo y el máximo (salida M) y por encima del nivel superior (salida S). En la Figura 1 puedes ver el depósito con los sensores y el símbolo lógico o interfaz² del circuito que queremos construir.

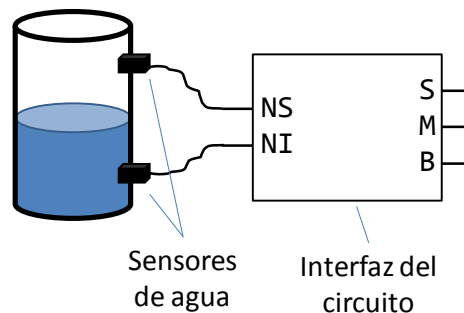


Figura 1. Circuito para la detección del nivel de agua.

Aprovecho para decirte que en este circuito tenemos tres salidas, por lo que podemos decir que tenemos tres funciones lógicas, S, M y B, aunque las tres funciones pueden representarse en una sola tabla de verdad.

La tabla de verdad de este circuito tendrá 5 columnas (2 entradas más 3 salidas), y 4 filas (2 entradas, aridad = 2, $2^2 = 4$). En la Tabla 5 puedes ver la tabla con las cuatro valoraciones.

Tabla 5. Estructura y valoraciones para la tabla de verdad de B, M y S.

NS	NI	B	M	S
0	0			
0	1			
1	0			
1	1			

Vamos a completar las salidas por filas. Así, para la primera fila o valoración, donde el agua está por debajo del nivel mínimo (no está activo ninguno de los sensores), activaremos la salida B, pero no la M ni la S. Lo puedes ver en la Tabla 6.

Tabla 6. Tabla de verdad para la valoración 0.

NS	NI	B	M	S
0	0	1	0	0
0	1			
1	0			
1	1			

² El símbolo lógico o interfaz de un circuito es una representación gráfica donde aparecen las entradas y salidas, **siempre** en el interior del símbolo.

Para la segunda fila o valoración el agua está por encima del nivel mínimo pero no llega al nivel máximo, por lo que activaremos la salida M pero no las salidas B ni S, tal como puedes ver en la Tabla 7.

Tabla 7. Tabla de verdad para las valoraciones 0 y 1.

NS	NI	B	M	S
0	0	1	0	0
0	1	0	1	0
1	0			
1	1			

Vamos ahora a por la tercera fila, es decir la valoración NS = 1 y NI = 0. En este caso, los sensores nos dicen que hay agua a la altura del sensor superior, pero no hay agua en el sensor inferior. En condiciones normales, esto es imposible, ¿no? Puede que se haya estropeado algún sensor, que el depósito se haya dado la vuelta, pero en condiciones normales esto no es posible.

Cuando una valoración, una combinación de las entradas es imposible, es decir, no puede suceder en condiciones normales, se llama entrada indiferente. Y el valor de las salidas para esta entrada se indica con una X, como puedes ver en la Tabla 8.

Tabla 8. Tabla de verdad para las valoraciones 0, 1 y 2 con entradas indiferentes.

NS	NI	B	M	S
0	0	1	0	0
0	1	0	1	0
1	0	X	X	X
1	1			

Un par de reflexiones sobre las entradas indiferentes. Fíjate que es la entrada la que es imposible, por lo que a todas las salidas se les asigna la X. También se asignan X cuando en los requerimientos del diseño nos dicen que para una determinada valoración no importa el valor de las salidas. Es decir, imposible o no importa, se llama entrada indiferente y las salidas toman valor X para esta valoración.

Por otro lado, podrías pensar que si la entrada es imposible o no importa el valor que tome, ¿qué más da que pongamos ceros o unos en las salidas? Efectivamente tu razonamiento es correcto, PERO, (siempre hay un pero) cuando aprendas a simplificar funciones verás que las X nos resultan muy útiles para obtener circuitos más sencillos y baratos. Así que si hay una entrada imposible o que no importa su salida, tú pon X en todas las salidas y al final ahorrarás recursos.

Vamos a terminar la tabla de verdad de las funciones B, M y S. Nos queda la última valoración, con los dos sensores activados, lo que indica que el nivel de agua está a la altura del sensor superior o por encima, por lo que debemos activar la salida S cómo puedes ver en la Tabla 9.

Tabla 9. Tabla de verdad de las funciones B, M y S.

NS	NI	B	M	S
0	0	1	0	0
0	1	0	1	0
1	0	X	X	X
1	1	0	0	1

Y ahora ya tienes la tabla de verdad del circuito, y por tanto tienes, de forma precisa e inequívoca, su funcionamiento.

4.2.1 Ejercicio con entradas indiferentes

En este ejercicio nos piden que diseñemos un circuito que nos diga si una caja está correctamente posicionada. Para detectar la posición de la caja tenemos 5 sensores, cuatro de ellos formando un cuadrado en el suelo, y el quinto en el centro del cuadrado. Cuando la caja se sitúa sobre uno de los sensores (en parte o completamente) este nos lo indica con un 1 en la señal correspondiente: N, E, S, O y C (Norte, Este, Sur, Oeste y Centro). Una caja estará bien posicionada si se activa el sensor central y ninguno de los cuatro sensores perimetrales, en cuyo caso se activará la salida P. Hay una segunda salida llamada H que se activa si hay al menos un sensor activado. La Figura 2 muestra la posición de los sensores y la interfaz del circuito.

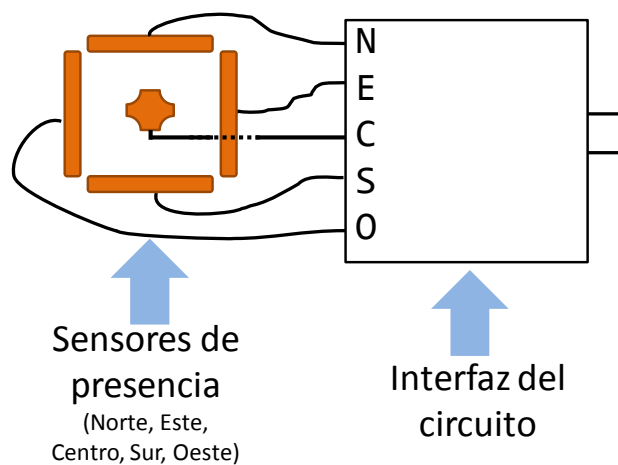


Figura 2. Sensores e interfaz para detectar la posición de la caja.

El cuadrado interior que dibujan los sensores mide 75 cm de lado, mientras que todas las cajas que se utilizan son cubos y su arista mide 50 cm, y se ubicarán de una en una. Esta información es muy importante, ya que te permitirá identificar casos imposibles, es decir, entradas indiferentes. El primer caso imposible es evidente, y es que una caja active los cinco sensores a la vez. Pero, ¿hay más casos imposibles? ¿Puede una caja activar a la vez N y E? ¿Y activar a la vez N y S? ¿Si? ¿No? ¿Seguro? Tendrás que calcular la longitud de la diagonal de la cara de la caja para responder a las preguntas anteriores³. En la Figura 3 tienes algunos ejemplos de cajas en diferentes posiciones y orientaciones que te pueden ayudar a identificar los casos que has de comprobar, pero has de hacer los cálculos para estar seguro.

Por favor, coge papel y lápiz (o el equivalente digital) e intenta construir la tabla de verdad de la función lógica P. Tienes la solución en la siguiente sección.

³ La longitud de la diagonal de un cuadrado de lado L se calcula cómo $\sqrt{2} * L$ o usando el teorema de Pitágoras para calcular la hipotenusa del triángulo rectángulo que forma medio cuadrado.

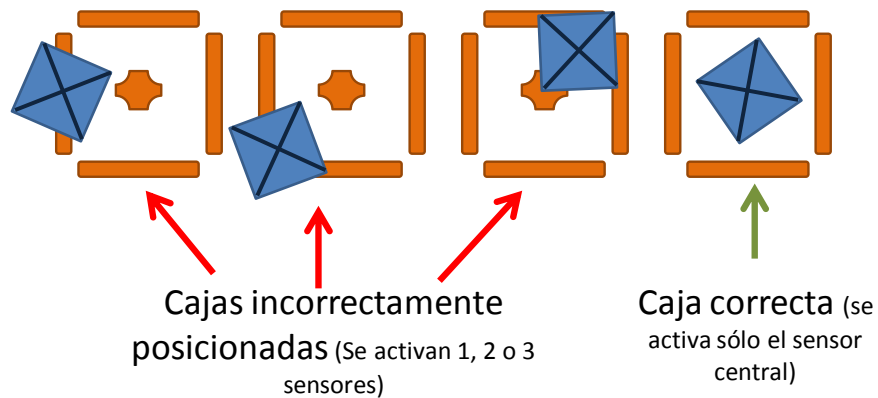


Figura 3. Ejemplos de posicionamiento de una caja.

5 Soluciones a los ejercicios

Aquí tienes las tablas de verdad de los ejercicios propuestos en los puntos 4.1.1 y 4.2.1. Es importante que intentes completar las tablas sin mirar la solución. Es más productivo que lo intentes y te equivoques a que las hagas bien a la primera ayudándote de la solución.

Para el detector de paridad impar habíamos decidido o asumido que ninguna entrada activa se considera un número par, por lo que la primera valoración (0 0 0 0) tiene salida 0. El resto de valoraciones no tienen mayor complicación, como puedes ver en la Tabla 10.

Tabla 10. Tabla de verdad de un detector de paridad Impar.

D	C	B	A	I
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Por cierto, igual te has dado cuenta de que en esta tabla de verdad no importa en qué orden ponemos las entradas, la tabla es siempre la misma. ¡Cuidado! Esto no pasa casi nunca, así que presta mucha atención al orden de las entradas por que es **fundamental**.

En el caso del circuito que detecta que la caja está correctamente posicionada, los casos imposibles corresponden con la activación simultánea de N y S o de E y O, ya que la arista de la caja mide 50cm y la diagonal no llega a los 71cm ($\sqrt{2} * 50$), por lo que no es posible colocar una caja de forma que active dos sensores opuestos que se encuentran a 75cm de distancia.

Por cierto, fíjate en la salida H, la que nos dice si hay al menos un sensor activado. Por ejemplo, en la valoración 00101 hay al menos un sensor activado y por tanto podríamos pensar que hay que activar la salida H. Pero no, ponemos una X ya que es una entrada imposible, y por tanto indiferente, y para algo que no es posible las salidas, todas, son X. "Si no puede pasar, ¿qué más da lo que haga el circuito?"

La solución a este ejercicio la tienes en la Tabla 11.

Tabla 11. Tabla de verdad del circuito o función lógica P que detecta la posición de una caja.

C	N	E	S	O	P	H
0	0	0	0	0	0	0
0	0	0	0	1	0	1
0	0	0	1	0	0	1
0	0	0	1	1	0	1
0	0	1	0	0	0	1
0	0	1	0	1	X	X
0	0	1	1	0	0	1
0	0	1	1	1	X	X
0	1	0	0	0	0	1
0	1	0	0	1	0	1
0	1	0	1	0	X	X
0	1	0	1	1	X	X
0	1	1	0	0	0	1
0	1	1	0	1	X	X
0	1	1	1	0	X	X
0	1	1	1	1	X	X
1	0	0	0	0	1	1
1	0	0	0	1	0	1
1	0	0	1	0	0	1
1	0	0	1	1	0	1
1	0	1	0	0	0	1
1	0	1	0	1	X	X
1	0	1	1	0	0	1
1	0	1	1	1	X	X
1	1	0	0	0	0	1
1	1	0	0	1	0	1
1	1	0	1	0	X	X
1	1	0	1	1	X	X
1	1	1	0	0	0	1
1	1	1	0	1	X	X
1	1	1	1	0	X	X
1	1	1	1	1	X	X

6 Conclusiones

En este objeto de aprendizaje has trabajado con varios conceptos:

- Circuito combinacional.
- Aridad.
- Valoración.
- Función lógica.
- Tabla de Verdad.
- Entradas indiferentes.
- Interfaz del circuito.

Si eres capaz de definir todos estos conceptos también serás capaz de construir la tabla de verdad de una función lógica, es decir, de describir formalmente y sin ambigüedades el comportamiento de un circuito combinacional, que es el objetivo que queríamos alcanzar con este objeto de aprendizaje.

Este es sólo el primer paso para llegar a diseñar un circuito digital, pero es un paso muy importante, ¡puedes estar satisfecho de dominarlo!

7 Bibliografía

7.1 Libros:

John F. Wakerly "Digital design : principles and practices", Upper Saddle River : Pearson Prentice Hall. 2006

Antonio Lloris Ruiz; Alberto Prieto Espinosa; Luis Parrilla Roure "Sistemas digitales", Aravaca, Madrid : McGraw-Hill/Interamericana de España. 2003