Álgebra de Boole

Apunte N° 3

George Boole y Claude Shannon

La finalidad de la Electrónica Digital es procesar la información. Para ello utiliza las operaciones definidas por George Boole en su «investigación sobre las leyes del pensamiento», publicada en 1854. En una época de triunfo de las matemáticas en la tarea de «modelizar» el mundo físico, George Boole dio también forma matemática a la combinación de proposiciones; Boole introdujo, a la vez, un lenguaje formal (la lógica proposicional) y una estructura matemática (el álgebra de Boole) capaz de representar y de validar tal lenguaje.

Casi un siglo después, en 1938, al estudiar los complejos circuitos de relés que se utilizaban en la comunicación telefónica, Claude E. Shannon demostró que las operaciones booleanas son aptas para describir los circuitos con conmutadores y, también, para expresar cálculos en el sistema de numeración de base 2. Shannon estableció la posibilidad de utilizar la misma estructura matemática (el álgebra de Boole) como soporte de un sistema de numeración y cálculo (el sistema binario) y proporcionó una forma de «construir» las operaciones del álgebra booleana mediante la conexión de dispositivos físicos (los conmutadores).

Boole y Shannon fijaron los «cimientos» conceptuales para el procesamiento digital de la información. Gracias a ellos disponemos de un lenguaje formalizado capaz de expresar la combinación de proposiciones, de un sistema de numeración capaz de soportar cálculos aritméticos y de una forma de «materializar» (es decir, de construir máquinas capaces de ejecutar) tanto el lenguaje como el sistema de numeración.

La base matemática que soporta todo esto corresponde a la estructura de álgebra de Boole de dos elementos (el 0 y el 1): álgebra booleana binaria. Las «máquinas digitales», aunque solamente «saben» trabajar con el 0 y el 1 (una lógica dual muy limitada), son capaces de manejar, a más alto nivel (por programación), la lógica difusa, las redes neuronales, la inferencia matemática, la inteligencia artificial,...

Estructura del Álgebra de Boole

El álgebra de Boole es una estructura matemática definida sobre un conjunto de elementos {a, b, c,...} por tres operaciones con las propiedades siguientes:

Operación	Representación	Postulados básicos
Suma - OR	$f = a + b$ Propiedad Asociativa: $a + b + c = (a + b) + c$ Propiedad Conmutativa: $a + b = b + a$	$0 + 0 = 0$ $a + 0 = a$ $0 + 1 = 1$ $a + 1 = 1$ $1 + 1 = 1$ $a + a = a$ $a + \overline{a} = 1$
Multiplicación - AND	$f = a \cdot b$ Propiedad Asociativa: $a \cdot b \cdot c = (a \cdot b) \cdot c$ Propiedad Conmutativa: $a \cdot b = b \cdot a$	$0 \cdot 0 = 0$ $a \cdot 0 = 0$ $0 \cdot 1 = 0$ $a \cdot 1 = a$ $1 \cdot 1 = 1$ $a \cdot a = a$ $a \cdot \overline{a} = 0$
Distributiva AND - OR	$a \cdot (b + c) = a \cdot b + a \cdot c$	$a + (b \cdot c) = (a + b) \cdot (a + c)$
Complemento - NOT	$f = \overline{a}$	$\overline{0} = 1$ $\overline{1} = 0$ $\overline{\overline{a}} = a$

Estructura del Álgebra de Boole

A partir de los axiomas que definen el álgebra de Boole pueden deducirse directamente, los siguientes teoremas operativos:

- **Idempotencia:** $a + a = a a \cdot a = a$
- **Absorción:** $a + a \cdot b = a$ $\qquad a + \overline{a} \cdot b = a + b$

  $a \cdot (a + b) = a$ $\qquad a \cdot (\overline{a} + b) = a \cdot b$
- **Adyacencia:** $a \cdot b + a \cdot \overline{b} = a$ $\qquad (a + b) \cdot (a + \overline{b}) = a$
- **Leyes de De Morgan:** $\overline{a \cdot b} = \overline{a} + \overline{b}$ $\qquad \overline{a + b} = \overline{a} \cdot \overline{b}$
- **Leyes de Consenso:** $a \cdot b + a \cdot c + b \cdot c = a \cdot b + a \cdot c$

  $(a + b) \cdot (a + c) \cdot (b + c) = (a + b) \cdot (a + c)$

Las leyes de De Morgan deben su nombre a su creador, Augustus De Morgan (1806-1871), matemático de origen inglés nacido en la India que fue el primer presidente de la Sociedad de Matemáticas de Londres.

Estructura del Álgebra de Boole

Demostración:

- **Absorción:** $a + \overline{a} \cdot b = a + b$

$a + \overline{a} \cdot b = (a + a \cdot b) + \overline{a} \cdot b$ por Absorción $a = a + a \cdot b$

$(a \cdot a + a \cdot b) + \overline{a} \cdot b$ por postulado $a = a \cdot a$

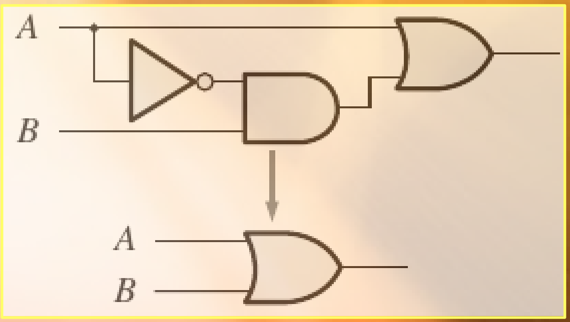
$a \cdot a + a \cdot b + a \cdot \overline{a} + \overline{a} \cdot b$ agregando $a \cdot \overline{a} = 0$

$(a + \overline{a}) \cdot (a + b)$ por factorización

$1 \cdot (a + b)$ por postulado $a + a = 1$

$a + b$ por postulado $1 \cdot a = a$

a	b	$\overline{a} \cdot b$	C0 + C2	a + b
0	0	0	0	0
0	1	1	1	1
1	0	0	1	1
1	1	0	1	1



Función Lógica

Se denomina función lógica a toda expresión algebraica formada por variables binarias que se relacionan mediante las operaciones básicas del álgebra de Boole. $f(a,b,c) = \overline{a} \cdot \overline{b} \cdot (a + b) + c + a \cdot c$

Simplificar una función lógica implica utilizar los postulados y axiomas de Boole, junto a las Leyes de De Morgan para llegar a una representación reducida de la función sin alterar esta, la función y su simplificada deben tener la misma tabla de verdad.

Ej: $f = \overline{a + b} \cdot (a + b)$

Aplico 1Er Ley de De Morgan: $\overline{a + b} = \overline{a} \cdot \overline{b}$

$f = \overline{a} \cdot \overline{b} \cdot (a + b)$

Aplico propiedad distributiva

$f = \overline{a} \cdot \overline{b} \cdot a + \overline{a} \cdot \overline{b} \cdot b$

Aplico postulado de Boole $a \cdot \overline{a} = 0$

$f = \overline{b} \cdot 0 + \overline{a} \cdot 0$

Aplico postulado de Boole $a \cdot 0 = 0$

$f = 0$

a	b	$\overline{a + b}$	(a + b)	C2·C3
0	0	1	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	1	0

Simplificación

Ej: $f = \overline{(a \cdot b + c) \cdot (a + b \cdot c)}$

Aplico 2da Ley de De Morgan: $\overline{a \cdot b} = \overline{a} + \overline{b}$

$f = \overline{(a \cdot b + c)} + \overline{(a + b \cdot c)}$

Aplico 1Er Ley de De Morgan: $\overline{a + b} = \overline{a} \cdot \overline{b}$

$f = \overline{(a \cdot b)} \cdot \overline{c} + \overline{a} \cdot \overline{(b \cdot c)}$

Aplico 2da Ley de De Morgan: $\overline{a \cdot b} = \overline{a} + \overline{b}$

$f = (\overline{a} + \overline{b}) \cdot \overline{c} + \overline{a} \cdot (\overline{b} + \overline{c})$

Aplico distributiva

$f = \overline{a} \cdot \overline{c} + \overline{b} \cdot \overline{c} + \overline{a} \cdot \overline{b} + \overline{a} \cdot \overline{c}$

Aplico postulado a + a = a

$f = \overline{a} \cdot \overline{c} + \overline{b} \cdot \overline{c} + \overline{a} \cdot \overline{b}$

Aplico factorización

$f = \overline{a} \cdot (\overline{c} + \overline{b}) + \overline{b} \cdot \overline{c}$

a	b	c	(a·b+c)	(a+b·c)	$\overline{C3 \cdot C4} = C6 + C7$	$\overline{a} \cdot (\overline{c} + \overline{b})$	$\overline{b} \cdot \overline{c}$
0	0	0	0	0	1	1	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	0
0	1	1	1	1	0	0	0
1	0	0	0	1	1	0	1
1	0	1	1	1	0	0	0
1	1	0	1	1	0	0	0
1	1	1	1	1	0	0	0

Simplificación

Ej: $f = [\, a\overline{b}\,(c + bd) + \overline{a}\,\overline{b}\,]\, c$

Aplico distributiva

$f = [\, a\overline{b}c + a\overline{b}bd + \overline{a}\,\overline{b}\,]\, c$

Aplico postulado $b\overline{b} = 0$

$f = [\, a\overline{b}c + a\cdot 0\cdot d + \overline{a}\,\overline{b}\,]\, c$

Aplico postulado $a\cdot 0\cdot d = 0$ y se elimina

$f = [\, a\overline{b}c + \overline{a}\,\overline{b}\,]\, c$

Aplico distributiva

$f = a\overline{b}cc + \overline{a}\,\overline{b}c$

Aplico postulado $cc = c$

$f = a\overline{b}c + \overline{a}\,\overline{b}c$

Aplico factorización

$f = \overline{b}c\,(a + \overline{a})$

Aplico postulado $a + \overline{a} = 1$

$f = \overline{b}c\cdot 1$

Aplico postulado $a \cdot 1 = a$

$f = \overline{b}c$

Simplificación

Existe una relación directa entre la complejidad de la red de puertas que constituyen un circuito lógico determinado y la complejidad de su expresión booleana.

Por ello, el objetivo de la simplificación de un circuito lógico consiste en minimizar su expresión para conseguir su implementación utilizando un número mínimo de puertas lógicas conectadas adecuadamente.

Para valorar las prestaciones de un diseño digital se tienen en consideración principalmente dos factores: la velocidad de respuesta y el coste. La velocidad de respuesta de un sistema digital disminuye con el retardo que sufre la señal al propagarse por los niveles o número de puertas que compone el camino más largo entre las entradas y las salidas del sistema, buscando aquel diseño que tenga el menor número de niveles de puertas.

La reducción de su coste se consigue utilizando un número mínimo de puertas, lo que determina una disminución de interconexiones, diseños de circuitos impresos más simples, mayor facilidad de mantenimiento, etc. En la valoración del coste no se incluye los posibles complementos de las variables de entrada que se supone que van a estar disponibles. Sin embargo, sí hay que tener en cuenta, principalmente con componentes discretos, las conexiones de las entradas y las conexiones necesarias entre las puertas del primer nivel y las puertas de salida o de segundo nivel.

El criterio para comparar el coste de dos circuitos digitales es: dado dos circuitos digitales se considera de menor coste aquel que tenga menos puertas, y a igual número de puertas el que necesite menos conexiones.

Formas canónicas de una función

Entre las múltiples expresiones algebraicas con las que se puede representar una función lógica, destacan dos tipos según la expresión esté formada por:

- sumas de productos

$$f_3 = f_3(c, b, a) = b\,a + c\,b\,\overline{a} + \overline{b}\,a + a$$

- productos de sumas

$$f_4 = f_4(c, b, a) = (b + a)(c + b + \overline{a})(\overline{b} + a)$$

Se define como término canónico de una función lógica a todo producto o suma en el que aparecen todas las variables en su forma directa a o complementada $\overline{a}$. Por ejemplo, en una función de tres variables, son términos canónicos, entre otros: $c\,\overline{b}\,a$ y $c + b + \overline{a}$.

A los términos producto se les llama productos canónicos o minitérminos. Esta denominación se debe al hecho de que, este término, toma el valor 1 para una sola combinación de las variables de entrada, de ahí el prefijo mini.

A los términos suma se les llama sumas canónicas o maxitérminos, denominándose así por el hecho de que, este término, toma el valor 1 tantas veces como lo hagan los sumandos que lo forman, de ahí el prefijo maxi.

Una función formada, exclusivamente, por términos de sumas canónicas o bien de productos canónicos recibe el nombre de función canónica. Si esta función tiene n variables, cada uno de sus productos o sumas canónicas tendrá n variables. Como cada variable se puede representar en su forma directa o complementada, el número de productos canónicos posibles será 2^n , al igual que el de sumas canónicas.

Formas canónicas de una función

Cada minitérmino se obtiene multiplicando las n variables en su forma directa si toman el valor 1 y complementada si tienen el valor 0.

Asimismo, aplicando el principio de dualidad, cada maxitérmino se obtiene sumando las n variables en su forma directa si toman el valor 0 y complementada si tienen el valor 1.

Los miniterminos se representan por m_i y los maxiterminos por M_i , siendo el subíndice i igual al valor decimal del número binario que corresponde al término canónico.

Dec.	abc	Miniterminos		Maxiterminos	
0	000	$\bar{a}\,\bar{b}\,\bar{c}$	m_0	$a+b+c$	M_7
1	001	$\bar{a}\,\bar{b}\,c$	m_1	$a+b+\bar{c}$	M_6
2	010	$\bar{a}\,b\,\bar{c}$	m_2	$a+\bar{b}+c$	M_5
3	011	$\bar{a}\,b\,c$	m_3	$a+\bar{b}+\bar{c}$	M_4
4	100	$a\,\bar{b}\,\bar{c}$	m_4	$\bar{a}+b+c$	M_3
5	101	$a\,\bar{b}\,c$	m_5	$\bar{a}+b+\bar{c}$	M_2
6	110	$a\,b\,\bar{c}$	m_6	$\bar{a}+\bar{b}+c$	M_1
7	111	$a\,b\,c$	m_7	$\bar{a}+\bar{b}+\bar{c}$	M_0

Conversión entre expresiones canónicas de una función

- Para obtener una expresión canónica en suma de productos (miniterminos) se usarán las combinaciones de variables binarias en las que la función vale uno.
- En el caso de querer obtener una expresión canónica en producto de sumas (maxiterminos) se usarán las combinaciones de variables binarias en las que la función vale cero.
- Para n variables, utilizando las leyes de De Morgan, se deduce la relación existente entre el complemento de un minitermino y su maxitermino equivalente:

$$m_i = M_{2^n-1-i} \qquad \text{ej: } \overline{m_3} = M_{2^3-1-3} = M_4 \qquad \sim(\overline{a}\; b\; c) = a+\overline{b}+\overline{c}$$

o viceversa, la relación existente entre el complemento de un maxitermino y su minitermino equivalente

$$M_i = m_{2^n-1-i} \qquad \text{ej: } \overline{M_5} = m_{2^3-1-5} = m_2 \qquad \sim(a+\overline{b}+c) = \overline{a}\; b\; \overline{c}$$

Conversión entre expresiones canónicas de una función

Las expresiones normalizadas son aquellas en las que no todos sus términos son canónicos y están únicamente formadas por suma de productos (Forma Normal Disyuntiva) o por producto de sumas (Forma Normal Conjuntiva).

Para convertir una expresión normalizada a canónica:

- En el caso de suma de productos, se multiplica cada término producto no canónico por la variable que falta más ella misma negada.
- En el caso de producto de sumas, se suma en cada factor no canónico la variable que falta por ella misma negada.

$$f_3\,(c,b,a) = c\,b + c\,\overline{b}\,a + \overline{b} = c\,b\,(a + \overline{a}) + c\,\overline{b}\,a + \overline{b}\,(c + \overline{c})(a + \overline{a}) =$$

$$= cba + cb\overline{a} + c\overline{b}a + c\,\overline{b}\,a + c\,\overline{b}\,\overline{a} + \overline{c}\,\overline{b}\,a + \overline{c}\,\overline{b}\,\overline{a} =$$

$$= cba + cb\overline{a} + c\,\overline{b}\,a + c\,\overline{b}\,\overline{a} + \overline{c}\,\overline{b}\,a + \overline{c}\,\overline{b}\,\overline{a}$$

$$= \sum_m (0, 1, 4, 5, 6, 7)\ \text{FND}$$

$$= \prod_M (2, 3)\ \text{FNC}$$

Bibliografía y Licencia

- Acha, Santiago, Castro, Manuel, Rioseras, Miguel, "Electrónica Digital Introducción a la Lógica Digital" 2da Ed. (Ra-Ma 2010)
- Floyd, Thomas, "Fundamentos de sistemas digitales" 9na Ed. (Pearson 2006)
- Gonzalez Gomez, Juan,"Circuitos y Sistemas Digitales" (Madrid 2002)

- Autor:

Matías E. García

Prof. & Tec. en Informática Aplicada
www.profmatiasgarcia.com.ar
info@profmatiasgarcia.com.ar

creative commons