Álgebra de Boole

Sistemas de Procesamiento de Datos

Prof. Lic Verónica Lourdes Tomich
Prof. TUP Rodrigo Soto
Prof. TUSI Leonardo Chiessa
Prof. Lic Eduardo Monaco
PDI Guillermo Gimenez

Álgebra de Boole

La circuitería digital en computadoras y otros sistemas digitales, se diseña y se analiza con el uso de una disciplina matemática denominada..

Álgebra de Boole

Álgebra de Boole

Resulta ser una herramienta útil en dos áreas:

- Análisis: es una forma concisa de describir el funcionamiento de los circuitos digitales.
- Diseño: dada una función deseada, se puede aplicar el álgebra para simplificar dicha función.

Álgebra de Boole

- Componentes de un álgebra:
  - Variables
  - Operaciones

Álgebra de Boole

- En este caso dichos componentes son lógicos.
  - Variables: Pueden tomar valores 0 o 1.
  - Operaciones: Op. Log. básicas
    - AND
    - OR
    - NOT

Álgebra de Boole

Representación de las operaciones básicas.

- A AND B = A . B

La operación and es verdadera (1) si y sólo si los dos valores son verdaderos.

Álgebra de Boole

A	B	**A AND B**
0	0	0
0	1	0
1	0	0
1	1	1

Álgebra de Boole

- A OR B = A+ B

Es verdadero si al menos 1 de los dos

operandos es verdadero.

- NOT A = $\bar{A}$ o ~A

Álgebra de Boole

A	B	**A OR B**
0	0	0
0	1	1
1	0	1
1	1	1

Álgebra de Boole

- NOT A = $\bar{A}$ o ~A

Si A es verdadero entonces $\bar{A}$ va a ser falso. Pero

si A es falso entonces $\bar{A}$ va a ser verdadero.

10

Álgebra de Boole

A	$\bar{A}$
0	1
0	1
1	0
1	0

Álgebra de Boole

- Ejemplo:

$$D = A + (\sim B + C)$$

D es igual a 1 si A es 1 o si B = 0 o si C = 1.

En otro caso D es igual a 0.

Álgebra de Boole

- Resolución del Ejemplo: $D = A + (\sim B + C)$
- $D = 1 \Rightarrow A = 1$, $B = 0$ y $C = 1$
- $D = 1 + (\sim 0 + 1) \quad \Rightarrow$
- $D = 1 + (1 + 1)$
- $D = 1$

Álgebra de Boole

- Resolución del Ejemplo: D = A + (~B + C)
- D = 1 => A= 1 , B=1 y C = 0
- D = 1 + (~1+0) =>
- D= 1+(0+0)
- D= 1

14

Álgebra de Boole

- Resolución del Ejemplo: $D = A + (\sim B + C)$
- $D = 1 \Rightarrow A = 0$, $B = 1$ y $C = 0$
- $D = 0 + (\sim 1 + 0) \Rightarrow$
- $D = 0 + (0 + 0)$
- $D = 0$

Álgebra de Boole

<u>Aclaraciones:</u>

En ausencias de paréntesis la operación AND es preferente a la operación OR. Si no hay ambigüedad la operación AND se presenta como una concatenación.

$$A + B.C = A + (B.C) = A + BC$$

Álgebra de Boole

La Tabla define las operaciones básicas en una forma conocida como TABLA de VERDAD.

- Se enumeran otro tres operadores
  - ➔ XOR
  - ➔ NAND
  - ➔ NOR

Álgebra de Boole

Los dos postulados agregados anteriormente denotan la manera en las que se interpretan las expresiones.

- ➔ A NAND B = NOT(A AND B) = NOT A OR NOT B
- ➔ A NOR B = NOT(A OR B) = NOT A AND NOT B

Las dos expresiones anteriores se denomina *Teorema De Morgan*

18

Álgebra de Boole

A NAND B = NOT(A AND B) = NOT A OR NOT B

A	B	A AND B	A OR B	**A NAND B**	**~(A AND B)**	~A	~B	**~A OR ~B**
0	0	0	0	1	1	1	1	1
0	1	0	1	1	1	1	0	1
1	0	0	1	1	1	0	1	1
1	1	1	1	0	0	0	0	0

Álgebra de Boole

A NOR B = NOT(A OR B) = NOT A AND NOT B

A	B	Ā	A NOR B	A OR B	**~(A OR B)**	~A	~B	**~A AND ~B**
0	0	1	1	0	1	1	1	1
0	1	1	0	1	0	1	0	0
1	0	0	0	1	0	0	1	0
1	1	0	0	1	0	0	0	0

Álgebra de Boole

De esta manera tenemos la tabla de verdad completa

21

Álgebra de Boole

A	B	Ā	A AND B	A OR B	A XOR B	A NAND B	A NOR B
0	0	1	0	0	0	1	1
0	1	1	0	1	1	1	0
1	0	0	0	1	1	1	0
1	1	0	1	1	0	0	0

Álgebra de Boole

Ahora analicemos los postulados:

23

Álgebra de Boole

Postulados básicos		
A.B=B.A	A+B=B+A	Ley Conmutativa

Álgebra de Boole

Postulados básicos		
A.(B+C)= (A.B)+(A.C)	A+(B.C)= (A+B).(A+C)	Ley Distributiva

Álgebra de Boole

Postulados básicos		
1.A=A	0+A=A	Elemento Neutro

Álgebra de Boole

Postulados básicos		
$A.\bar{A}=0$	$A+\bar{A}=1$	Elemento complemento

Álgebra de Boole

Ahora analicemos otras identidades:

28

Álgebra de Boole

Otras identidades		
0.A=0	1+A=1	Ley Nula

Álgebra de Boole

Otras identidades		
A.A=A	A+A=A	Idempotencia

Álgebra de Boole

Otras identidades		
A.(B.C)=(A.B).C	A+(B+C)=(A+B)+C	Ley Asociativa

Álgebra de Boole

Otras identidades		
~(A.B)=~A+~B	~(A+B)=~A.~B	Teorema De Morgan

Álgebra de Boole

Puertas: Bloque fundamental de construcción de todos los circuitos lógicos digitales.

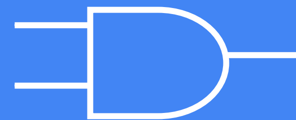
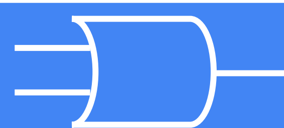
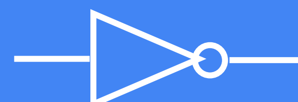
- Las funciones lógicas se implementan interconectando puertas.

33

Álgebra de Boole

- Cada puerta se define de tres formas:
  - Símbolo grafico
  - Notación algebraica
  - Tabla de verdad

34

Nombre	Símbolo	Función
AND		F=A.B
OR		F=A+B
XOR		$F=A \oplus B$
NOT		$F=\bar{A}$
NAND		F=~(AB)
NOR		F=~(A+B)

Álgebra de Boole

- Por lo tanto para poder llevar todo esto a la práctica vamos a tener qué tener muy presente las siguientes tres diapositivas:
- Postulados Básicos
- Otras Identidades
- Simbología de funciones básicas

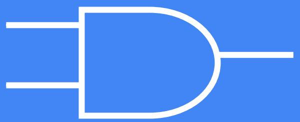
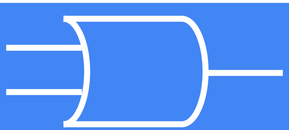
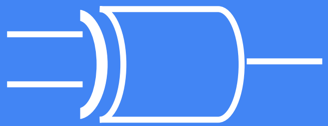
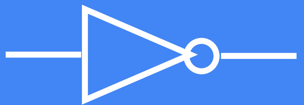
36

Álgebra de Boole

Postulados Básicos		
A.B=B.A	A+B=B+A	Ley Conmutativa
A.(B+C)= (A.B)+(A.C)	A+(B.C)= (A+B).(A+C)	Ley Distributiva
1.A=A	0+A=A	Elemento Neutro
$A.\bar{A}=0$	$A+\bar{A}=1$	Elemento Complemento
A.(A +B)=A	A+A.B=A	Ley de Absorción

Álgebra de Boole

Otras identidades		
0.A=0	1+A=1	Elemento Neutro
A.A=A	A+A=A	Idempotencia
A.(B.C)=(A.B).C	A+(B+C)=(A+B)+C	Ley Asociativa
~(A.B)=~A+~B	~(A+B)=~A.~B	Teorema De Morgan

Nombre	Símbolo	Función
AND		F=A.B
OR		F=A+B
XOR		$F=A \oplus B$
NOT		$F=\bar{A}$
NAND		F=~(AB)
NOR		F=~(A+B)

Expresiones Canónicas

- Existen dos formas básicas de expresión canónica que pueden ser interpretadas en dos niveles de compuertas:
  - Suma de productos o expansión de miniterminos.
  - Productos de sumas o expansión de maxiterminos.

40

Expresiones Canónicas

- Permiten asociar a una función una expresión algebraica única.

- La tabla de verdad también es una representación única para una función booleana.

Suma de productos

Los términos son productos o miniterminos.

- Formado por los productos AND que para las diferentes combinaciones de entradas producen salida verdadera.
- En cada producto cada variable aparece una vez, esta puede estar invertida.

42

Suma de productos

- También conocida como expansión de miniterminos.

F = 001 011 101 110 111

$$F = A'B'C + A'BC + AB'C + ABC' + ABC$$

A	B	C	F	F′
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

$$F' = A'B'C' + A'BC' + AB'C'$$

Suma de productos

- Primero tomamos los F qué tienen como resultado el 1

A	B	C	F	F′
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

001 => A´B´C

011 => A´B C

101 => A B´C

110 => A B C´

111 => ABC

Suma de productos

Los términos de miniterminos los tendremos

que sumar

45

Suma de productos

F=	001	011	101	110	111

46

Suma de productos

F=	001	011	101	110	111
F=	A´.B´.C +	A´.B.C +	A.B´.C +	A.B.C´ +	ABC

Suma de productos

F=	001	011	101	110	111
F=	A´.B´.C +	A´.B.C +	A.B´.C +	A.B.C´ +	ABC
F(001)=	1.1.1 +	1.0.1 +	0.1.1+	0.0.0+	001

48

Suma de productos

F=	001	011	101	110	111
F=	A´.B´.C +	A´.B.C +	A.B´.C +	A.B.C´ +	ABC
F(001)=	1.1.1 +	1.0.1 +	0.1.1+	0.0.0+	0.0.1
F(001)=	1 +	0 +	0 +	0 +	0
F(011)=	1.0.1+	1.1.1+	0.0.1+	0.1.0+	011

Entonces sabemos qué si el primer término nos da 1 todo lo qué luego sumamos no importa porque su resultado es 1.

Forma Canónica de Miniterminos buscamos los F=1

F en forma canónica:

$$
\begin{aligned}
F(A, B, C) &= \Sigma m(1,3,5,6,7) \\
&= m1 + m3 + m5 + m6 + m7 \\
&= A'B'C + A'BC + AB'C + ABC' + ABC
\end{aligned}
$$

Forma Canónica a Forma Mínima

RECORDEMOS:

Terminos = ABC

Variables = Cada letra

Objetivo:

Reducir la función aplicando las leyes anteriormente nombradas

Forma Canónica a Forma Mínima

Reducir la función aplicando las leyes anteriormente:

forma canónica $\neq$ forma minima

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC' \\
&= C + ABC' \\
&= ABC' + C \\
&= AB + C
\end{aligned}
$$

Forma Canónica a Forma Mínima

Veamos el Paso a Paso

53

Forma Canónica a Forma Mínima

Buscar qué tienen en común los términos:

$$F(A, B, C) = A'B'C + A'BC + AB'C + ABC + ABC'$$

54

Forma Canónica a Forma Mínima

Los primeros cuatro términos tienen C en común.

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC'
\end{aligned}
$$

Forma Canónica a Forma Mínima

Ahora tratamos de reducirlo más..

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC'
\end{aligned}
$$

Forma Canónica a Forma Mínima

Seguimos simplificando, aplico ley distributiva.

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC'
\end{aligned}
$$

Forma Canónica a Forma Mínima

Analicemos los términos con: A’, B’ B y A

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC'
\end{aligned}
$$

Aplico Elemento Complemento: (A’ + A) = 1 y (B’ + B) = 1

Forma Canónica a Forma Mínima

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC' \\
&= C + ABC'
\end{aligned}
$$

Aplico ley distributiva NUEVAMENTE

Forma Canónica a Forma Mínima

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC' \\
&= C + ABC'
\end{aligned}
$$

Aplicó la distributiva (C+C') . (A.B+C) y luego simplifico

Forma Canónica a Forma Mínima

forma canónica $\neq$ forma minima

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC' \\
&= C + ABC' \\
&= ABC' + C \\
&= AB + C
\end{aligned}
$$

61

A	B	C	minterms	
0	0	0	$A'B'C'$	m0
0	0	1	$A'B'C$	m1
0	1	0	$A'BC'$	m2
0	1	1	$A'BC$	m3
1	0	0	$AB'C'$	m4
1	0	1	$AB'C$	m5
1	1	0	ABC'	m6
1	1	1	ABC	m7

forma corta de escribir minterms

(ejemplo de 3 terminos o $2^3 = 8$ minterms)

F en forma canónica:

$$
\begin{aligned}
F(A, B, C) &= \Sigma m(1,3,5,6,7) \\
&= m1 + m3 + m5 + m6 + m7 \\
&= A'B'C + A'BC + AB'C + ABC' + ABC
\end{aligned}
$$

forma canónica $\neq$ forma minima

$$
\begin{aligned}
F(A, B, C) &= A'B'C + A'BC + AB'C + ABC + ABC' \\
&= (A'B' + A'B + AB' + AB)C + ABC' \\
&= ((A' + A)(B' + B))C + ABC' \\
&= C + ABC' \\
&= ABC' + C \\
&= AB + C
\end{aligned}
$$

Producto de sumas

- También conocida como expansión de maxiterminos.

$$F = \quad 000 \qquad 010 \qquad 100$$

$$F = (A + B + C)\,(A + B' + C)\,(A' + B + C)$$

A	B	C	F	F′
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Producto de sumas

- También conocida como expansión de maxiterminos. Tomamos como seria con el F´

F = 000 010 100

$F = (A + B + C)\,(A + B' + C)\,(A' + B + C)$

A	B	C	F	F′
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

$F' = (A + B + C')\,(A + B' + C')\,(A' + B + C')\,(A' + B' + C)\,(A' + B' + C')$

Producto de sumas

- Los términos son sumas o maxitérminos.
  - Formado por las suma OR que para las diferentes combinaciones de entradas producen salida falsa.
  - En cada producto cada variable aparece una vez, esta puede estar invertida.

65

A	B	C	maxterms	
0	0	0	$A+B+C$	M0
0	0	1	$A+B+C'$	M1
0	1	0	$A+B'+C$	M2
0	1	1	$A+B'+C'$	M3
1	0	0	$A'+B+C$	M4
1	0	1	$A'+B+C'$	M5
1	1	0	$A'+B'+C$	M6
1	1	1	$A'+B'+C'$	M7

F en forma canónica:

$$\begin{aligned} F(A, B, C) &= \Pi M(0,2,4) \\ &= M0 \bullet M2 \bullet M4 \\ &= (A + B + C)\,(A + B' + C)\,(A' + B + C) \end{aligned}$$

forma canónica $\neq$ forma minima

$$\begin{aligned} F(A, B, C) &= (A + B + C)\,(A + B' + C)\,(A' + B + C) \\ &= (A + B + C)\,(A + B' + C) \\ &\quad (A + B + C)\,(A' + B + C) \\ &= (A + C)\,(B + C) \end{aligned}$$

Maxi

forma corta de escribir [redacted]terminos

Maxi

(ejemplo de 3 términos o $2^3 = 8$ [redacted]terminos)

Ejemplo 1

- Se desea construir un circuito que contiene tres entradas A, B, C, que representan los bits de un número binario entero no negativo, y la salida vale 1, si el número que representa en decimal, es una potencia exacta de 2.

67

Ejemplo 1

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Son potencias de dos: 1, 2 y 4.

Utilizamos Miniterminos:

F= A´.B´.C + A´.B.C´+ A.B´.C´

Ejemplo 2

- Se desea diseñar un circuito que permita determinar cuando un semáforo se encuentra dañado.

A= VERDE B = AMARILLO C = ROJO

1= PRENDIDO 0= APAGADO

Ejemplo 2

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

A= VERDE B = AMARILLO C = ROJO

1= PRENDIDO 0= APAGADO

Usamos Maxiterminos:

F= (A+B+C).(A´+B+C´).(A´+B´+C´)