

Simplificación y minimización por adyacencias

Resumen

La simplificación y minimización por adyacencias utiliza los teoremas de complementos y de asociación del álgebra booleana, aunada a la disposición de celdas adyacentes ubicadas por los bits menos significativos. El proceso de combinar celdas adyacentes para obtener una máxima minimización se puede hacer por renglones y filas, gracias a que se pueden encontrar las adyacencias que sólo cambian en los bits menos significativos.

Palabras claves: Simplificación, minimización, bits más significativos y menos significativos, adyacencias.

1. Introducción

El álgebra booleana es la herramienta matemática en la que está basado el diseño lógico y se usa en la descripción y minimización y análisis de las funciones lógicas binarias.

Si tuviéramos una función $f = \bar{a}b + a\bar{b}$ expresada con una combinación de ceros y unos, tendríamos la siguiente tabla de combinaciones lógicas de la función:

a	b	$f = a+b$
0	0	0
0	1	1
1	0	1
1	1	0

En esta tabla sólo se ha escogido los términos de $a+b$ donde aparece un 1.

A cada elemento de f se le conoce como un minitérmino (en f tenemos dos minitérminos separados por el signo +), y a toda la función ($\bar{a}b + a\bar{b}$) se le conoce como la forma canónica expresada en minitérminos. Los elementos a, b se niegan o complementan cuando son cero, por ejemplo en el segundo renglón; a es negado ($\bar{a}$) y b no está negado.

Una técnica para minimizar la cantidad de minitérminos en una función lógica, es por medio del mapa de Karnaugh. Esta técnica es gráfica, y consiste en vaciar la tabla de la forma siguiente: Donde los valores de la función $f = a+b$ son 1, se selecciona la tabla a/b , y en la intersección de los valores a, b se deposita el 1, como a continuación se tiene.

Como $a = 0$ y $b = 1$ en esta posición se escribe un 1, con $a = 1$ y $b = 1$ también en esta posición se escribe un 1. Lo que da como resultado esta tabla.

a/b	0	1
0		
1	1	1

Si se observa en la tabla a/b sólo hay un cambio de un bit de celda a celda. El cambio de un bit de término a término se puede observar mejor en una tabla de cuatro variables.

ab/bc	00	01	11	10
00				
01				
11				
10				

Se tiene que de 00 a 01, de 01 a 11, de 11 a 10 y de 10 a 00 sólo hay un cambio de un bit.

Esto hace posible utilizar propiedad de complemento $\bar{a}+a=1$ (1) y la propiedad asociativa $b\bar{a}+ba=b(\bar{a}+a)=b$ (2).

El sistema gráfico de minimización tiene problemas cuando se trabaja con más de cuatro variables. Porqué es difícil encontrar que de celda a celda sólo haya un cambio de un bit y cuando esto se ha solucionado, en ocasiones la función proporcionada y la función obtenida no son lógicamente equivalentes.

Para poder buscar una función lógicamente equivalente, primero se tiene que buscar los términos adyacentes que solo cambien en un bit.

2. Encontrando términos adyacentes que cambian en un bit

Se sabe que los términos adyacentes que cambian en un bit para cuatro variables son los siguientes:

$$\begin{pmatrix} 00 \\ 01 \\ 11 \\ 10 \end{pmatrix} \quad (3)$$

Pero si se tiene más de cuatro bits, se buscan los demás términos adyacentes de la siguiente forma:

- Se toma como base (3)
- Se procede a tener las combinaciones de la forma siguiente.

$$0 \begin{pmatrix} 00 \\ 01 \\ 11 \\ 10 \end{pmatrix} \quad (4) \quad 1 \begin{pmatrix} 00 \\ 01 \\ 11 \\ 10 \end{pmatrix} \quad (5)$$

- Se rescribe (4) y se le añade (5) comenzando a añadirlo de abajo hacia arriba, dando como resultado:

$$\begin{pmatrix} 000 \\ 001 \\ 011 \\ 010 \\ 110 \\ 111 \\ 101 \\ 100 \end{pmatrix} \quad (6)$$

con esta combinación se tendría una tabla como la siguiente

(7)

abc/def	000	001	011	010	110	111	101	100
000								
001								
011								
010								
110								
111								
101								
100								

que es para 6 variables, pero si se quiere una combinación de cinco variables es suficiente agregar en la primera columna la combinación de cuatro variables, como se observa en la tabla.

abc/de	000	001	011	010	110	111	101	100
00								
01								
11								
10								

Esta tabla da una combinación para cinco variables. Para obtener más combinaciones, ahora se toma como base (6) y se siguen los pasos ii y iii.

Ejemplo 1. Para una combinación de ocho variables tendríamos:

Paso i. Se toma como base la combinación para seis variables

$$\begin{pmatrix} 000 \\ 001 \\ 011 \\ 010 \\ 110 \\ 111 \\ 101 \\ 100 \end{pmatrix}$$

Paso ii. Se procede a tener las combinaciones de la forma siguiente

$$0 \begin{pmatrix} 000 \\ 001 \\ 011 \\ 010 \\ 110 \\ 111 \\ 101 \\ 100 \end{pmatrix} \quad 1 \begin{pmatrix} 000 \\ 001 \\ 011 \\ 010 \\ 110 \\ 111 \\ 101 \\ 100 \end{pmatrix}$$

Paso iii. Se reescriben y se obtiene

$$\begin{pmatrix} 0000 \\ 0001 \\ 0011 \\ 0010 \\ 0110 \\ 0111 \\ 0101 \\ 0100 \\ 1100 \\ 1101 \\ 1111 \\ 1110 \\ 1010 \\ 1011 \\ 1001 \\ 1000 \end{pmatrix}$$

Con lo que se obtiene la tabla para ocho variables

	0000	0001	0011	0010	0110	0111	0101	0100	1100	1101	1111	1110	1010	1011	1001	1000
0000																
0001																
0011																
0010																
0110																
0111																
0101																
0100																
1100																
1101																
1111																
1110																
1010																
1011																
1001																
1000																

Y para siete variables, en la columna se introduce la combinación para seis variables (6), como a continuación se muestra.

	0000	0001	0011	0010	0110	0111	0101	0100	1100	1101	1111	1110	1010	1011	1001	1000
000																
001																
011																
010																
110																
111																
101																
100																

Y así se puede continuar buscando combinaciones, basándose en las combinaciones anteriores. Esto se puede implementar para su automatización.

3. Algoritmo de minimización.

3.1 Observaciones

- Se divide la tabla en dos partes las cuales son: bits más significativos y menos significativos como se muestra a continuación en una tabla para seis variables.

	BITS MAS SIGNIFICATIVOS							
	000	001	011	010	110	111	101	100
000								
001								
011								
010								
110								
111								
101								
100								
	BITS MENOS SIGNIFICATIVOS							

(8)

- Se le llama tabla de selección, a la tabla donde se vacían los elementos seleccionados de la tabla de combinaciones lógicas f. A continuación se da un ejemplo para seis variables.

(9)

	000	001	011	010	110	111	101	100
000		1	1			1	1	
001	1			1	1			1
011	1			1	1			1
010		1	1			1	1	
110		1	1			1	1	
111	1			1	1			1
101	1			1	1			1
100		1	1			1	1	

iii. Se le llama tabla de minimización, a la que contiene todos los elementos seleccionados en la tabla de combinaciones, a continuación se muestra como ejemplo una tabla para seis variables.

(10)

		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

iv. Las ubicaciones de celda, celda adyacente y celda siguiente a la celda adyacente en la tabla de minimización es de la forma siguiente:

(11)

	Celda	Celda adyacente	Celda siguiente a la adyacente	
	Celda adyacente			
	Celda siguiente a la adyacente			

Mas tarde, en la minimización la celda adyacente se convierte en la celda y la celda siguiente a la adyacente en la celda adyacente etc.

3.2 Consideraciones Previas

- Una vez que se tiene la función lógica, hay que vaciarla en una tabla de selección (9) (combinaciones seleccionadas) para obtener la tabla de minimización (10) , y proceder a la minimización.
- La tabla de minimización (10) está ordenada de tal forma que las celdas adyacentes (verticales) sólo cambia en el bit menos significativo (000011 y 000010) y horizontalmente cambia el ultimo bit de la parte mas significativa (000000 y 001000), en esto nos apoyaremos para eliminar variables utilizando (1) y (2).
- La eliminación de las variables será primero verticalmente y después en forma horizontal, por consideraciones de cómo cambian los bits menos significativos y los bits más significativos.
- Al inicio se seleccionan las celdas de la forma siguiente:

Verticalmente, por parejas adyacentes en las cuales sólo cambia el bit menos significativo.

Horizontalmente, por parejas adyacentes en las cuales sólo cambia un bit de la parte más significativa.

Es decir para funciones de cuatro variables horizontal y vertical (00 y 01), (11 y 10).

Para funciones de cinco variables:

Horizontal

(00 y 01), (11 y 10)

Vertical

(000 y 001), (011 y 010), (110 y 111), (101 y 100)

Para funciones de seis variables:

Horizontal y vertical

(000 y 001), (011 y 010), (110 y 111), (101 y 100)

Para funciones de siete variables:

Horizontal

(0000 y 0001), (0011 y 0010), (0110 y 0111), (0101 y 0100), (1100 y 1101), (1111 y 1110)(1010 y 1011), (1001 y 1000).

Vertical

(000 y 001), (011 y 010), (110 y 111), (101 y 100).

Y así sucesivamente para una mayor cantidad de variables

3.3 Algoritmo

I. La tabla esta llena con las combinaciones seleccionadas de la tabla de la función lógica. Tabla de minimización (10).

II. Se selecciona la primera columna, y se localiza la primera celda y su celda adyacente (de arriba hacia abajo).

III. Se comparan las celdas. .

Si cambian en un solo bit como se indica en el inciso d. Se elimina la celda mas baja y de la celda de arriba se elimina el bit cambiante (sólo se señala que ha sido eliminado)

Si no cambia en un bit. Se buscan otras celdas adyacentes (ahora la que era celda adyacente toma el lugar de celda y la que era celda siguiente a la celda adyacente toma el lugar de celda adyacente (11)) y se procesa el punto III hasta terminal la columna.

IV. Al terminar de minimizar todas las columnas en la tabla de minimización, procedemos de la siguiente forma:

V. Se selecciona la primera columna, y se localiza la primera celda y la celda siguiente a la celda adyacente (de arriba hacia abajo).

VI. Se comparan las celdas.

Si cambian en un solo bit como se indica en el inciso d. Se elimina la celda mas baja y de la celda de arriba se elimina el bit cambiante (solo se señala que ha sido eliminado)

Si no cambia en un bit. Se buscan otras celdas (celda y celda siguiente a la adyacente) ahora la que era celda adyacente toma el lugar de celda.

VII. Al terminar de minimizar todas las columnas en la tabla de minimización, procedemos de la siguiente forma:

VIII. Se selecciona la primera fila y se buscan combinaciones adyacentes (de izquierda a derecha).

III. Se comparan las celdas.

Si cambian en un solo bit como se indica en el inciso d. Se elimina la celda que esta a la derecha y de la celda de la izquierda se elimina el bit cambiante (sólo se señala que ha sido eliminado)

Si no cambia en un bit. Se buscan otras celdas adyacentes (ahora la que era celda adyacente toma el lugar de celda y la que era celda si-

guiente a la celda adyacente toma el lugar de celda adyacente (11)) y se procesa el punto III hasta terminar la fila.

IV. Al terminar de minimizar todas las filas en la tabla de minimización, procedemos de la siguiente forma:

V. Se selecciona la primera fila, y se localiza la primera celda y la celda siguiente a la celda adyacente (de izquierda a derecha).

VI. Se comparan las celdas.

Si cambian en un solo bit como se indica en el inciso d. Se elimina la celda de la derecha y de la celda de la izquierda se elimina el bit cambiante (sólo se señala que ha sido eliminado) Si no cambia en un bit. Se buscan otras celdas (celda y celda siguiente a la adyacente) ahora la que era celda adyacente toma el lugar de celda.

VII. Al terminar de minimizar todas las filas en la tabla de minimización, procedemos de la siguiente forma:

VIII. Se selecciona la primera columna, y se selecciona la primera celda,

VIII. Se buscan cambios con las demás celdas en la parte de los bits menos significativos (de arriba a bajo).

Si cambian en un solo bit como se indica en el inciso d. Se elimina la celda más baja y de la celda de arriba se elimina el bit cambiante (sólo se señala que ha sido eliminado). Este es un proceso de una contra todas.

Ahora se toma la siguiente celda y se procesa el punto IX hasta terminar la columna.

X. Al terminar de minimizar todas las columnas en la tabla de minimización, procedemos de la siguiente forma:

XI. Se selecciona la primera fila, se selecciona la primera celda

XII. Se buscan cambios con las demás celdas en la parte de los bits mas significativos (de izquierda a derecha).

Si cambian en un solo bit como se indica en el inciso d. Se elimina la celda que está a la derecha y de la celda de la izquierda se elimina el bit cambiante (sólo se señala que ha sido eliminado). Este es un proceso de una contra todas.

Ahora se toma la siguiente celda y se procesa el punto XII hasta terminar la fila.

XIII. Al terminar de minimizar todas las filas en la tabla de minimización, procedemos de la siguiente forma:

XIV. Si no se eliminó ninguna variable horizontalmente (bits más significativos) se ha terminado el proceso, de lo contrario se comienza nuevamente desde el punto VIII.

4. Implementación (Sumador completo de tres bits)

Se tiene la siguiente tabla, donde se realizaron las combinaciones y se obtuvieron las distintas adiciones de los bits $abc + def$. Donde abc y def son sumandos y no una función lógica.

Nota. Los números dos que aparecen en las tablas de minimización corresponden a las variables eliminadas.

a	b	c	d	e	f	S0	S1	S2	carry
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	1	0	0
0	0	0	0	1	1	1	1	0	0
0	0	0	1	0	0	0	0	1	0
0	0	0	1	0	1	1	0	1	0
0	0	0	1	1	0	0	1	1	0
0	0	0	1	1	1	1	1	1	0
0	0	1	0	0	0	1	0	0	0
0	0	1	0	0	1	0	1	0	0
0	0	1	0	1	0	1	1	0	0
0	0	1	0	1	1	0	0	1	0
0	0	1	1	0	0	1	0	1	0
0	0	1	1	0	1	0	1	1	0
0	0	1	1	1	0	1	1	1	0
0	0	1	1	1	1	0	0	0	1
0	1	0	0	0	0	0	1	0	0
0	1	0	0	0	1	1	1	0	0
0	1	0	0	1	0	0	0	1	0
0	1	0	0	1	1	1	0	1	0
0	1	0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1	1	0
0	1	0	1	1	0	0	0	0	1
0	1	0	1	1	1	1	0	0	1
0	1	1	0	0	0	1	1	0	0
0	1	1	0	0	1	0	0	1	0
0	1	1	0	1	0	0	1	1	0
0	1	1	0	1	1	1	0	0	1
0	1	1	1	0	0	0	0	0	1
0	1	1	1	0	1	1	0	0	1
0	1	1	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1	1	0
1	0	0	0	0	0	0	0	1	0
1	0	0	0	0	1	1	0	1	0
1	0	0	0	1	0	0	1	1	0
1	0	0	0	1	1	1	1	1	0
1	0	0	1	0	0	0	0	0	1
1	0	0	1	0	1	1	0	0	1
1	0	0	1	1	0	0	1	0	1
1	0	0	1	1	1	1	1	0	1
1	0	1	0	0	0	1	0	1	0
1	0	1	0	0	1	0	1	1	0
1	0	1	0	1	0	0	0	0	1
1	0	1	0	1	1	1	0	0	1
1	0	1	1	0	0	0	1	1	0
1	0	1	1	0	1	1	1	1	0
1	0	1	1	1	0	0	0	0	1
1	0	1	1	1	1	1	1	0	1
1	1	0	0	0	0	0	1	1	0
1	1	0	0	0	1	1	1	1	0
1	1	0	0	1	0	0	0	0	1
1	1	0	0	1	1	1	0	0	1
1	1	0	1	0	0	1	0	0	1
1	1	0	1	0	1	0	1	0	1
1	1	0	1	1	0	0	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	1	0	0	0	1	1	1	0
1	1	1	0	0	1	0	0	0	1
1	1	1	0	1	0	1	0	0	1
1	1	1	0	1	1	0	1	0	1
1	1	1	1	0	0	0	1	1	0
1	1	1	1	0	1	1	0	0	1
1	1	1	1	1	0	0	0	0	1
1	1	1	1	1	1	1	1	1	0

0	1	1	0	0	1	0	0	1	0
0	1	1	0	1	0	1	0	1	0
0	1	1	0	1	1	0	1	1	0
0	1	1	1	0	0	1	1	1	0
0	1	1	1	0	1	0	0	0	1
0	1	1	1	1	0	1	0	0	1
0	1	1	1	1	1	0	1	0	1
1	0	0	0	0	0	0	0	1	0
1	0	0	0	0	1	1	0	1	0
1	0	0	0	1	0	0	1	1	0
1	0	0	0	1	1	1	1	1	0
1	0	0	1	0	0	0	0	0	1
1	0	0	1	0	1	1	0	0	1
1	0	0	1	1	0	0	1	0	1
1	0	0	1	1	1	1	1	0	1
1	0	1	0	0	0	1	0	1	0
1	0	1	0	0	1	0	1	1	0
1	0	1	0	1	0	1	1	1	0
1	0	1	0	1	1	0	0	0	1
1	0	1	1	0	0	1	0	0	1
1	0	1	1	0	1	0	1	0	1
1	0	1	1	1	0	0	0	0	1
1	0	1	1	1	1	1	0	0	1
1	1	0	0	0	0	0	1	1	0
1	1	0	0	0	1	1	1	1	0
1	1	0	0	1	0	0	0	0	1
1	1	0	0	1	1	1	0	0	1
1	1	0	1	0	0	1	1	0	1
1	1	0	1	0	1	0	0	0	1
1	1	0	1	1	0	1	1	0	1
1	1	0	1	1	1	0	0	1	1
1	1	1	0	0	0	1	1	1	0
1	1	1	0	0	1	0	0	0	1
1	1	1	0	1	0	1	0	0	1
1	1	1	0	1	1	0	1	0	1
1	1	1	1	0	0	0	1	1	0
1	1	1	1	0	1	1	0	0	1
1	1	1	1	1	0	0	0	0	1
1	1	1	1	1	1	1	1	1	0

TABLA 1. SUMADOR COMPLETO DE TRES BITS.

Se vacía la Tabla 1 de acuerdo a sus entradas ($abcdef$) y salidas ($carry, s2, s1, s0$) en la tabla de selección, como se muestra a continuación.

CARRY

	000	001	011	010	110	111	101	100
000								
001						1		
011					1	1	1	
010					1	1		
110			1	1	1	1	1	1
111		1	1	1	1	1	1	1
101			1		1	1	1	1
100					1	1	1	1

TABLA DE TÉRMINOS SELECCIONADOS CARRY								
						111001		
					110011	111011	101011	
					110010	111010		
			011110	010110	110110	111110	101110	100110
		001111	011111	010111	110111	111111	101111	100111
			011101		110101	111101	101101	100101
					110100	111100	101100	100100

SALIDA S2

	000	001	011	010	110	111	101	100
000					1	1	1	1
001			1		1		1	1
011		1	1	1				1
010			1	1			1	1
110	1	1			1	1		
111	1				1	1	1	
101	1	1		1		1		
100	1	1	1	1				

TÉRMINOS SELECCIONADOS S2								
					110000	111000	101000	100000
			011001		110001		101001	100001
		001011	011011	010011				100011
			011010	010010			101010	100010
	000110	001110			110110	111110		
	000111				110111	111111	101111	
	000101	001101		010101		111101		
	000100	001100	011100	010100				

SALIDA S1

	000	001	011	010	110	111	101	100
000			1	1	1	1		
001		1		1	1		1	
011	1		1			1		1
010	1	1					1	1
110	1	1					1	1
111	1		1			1		1
101		1		1	1		1	
100				1	1	1		

TABLA DE TÉRMINOS SELECCIONADOS S1								
			011000	010000	110000	111000		
		001001		010001	110001		101001	
	000011		011011			111011		100011
	000010	001010					101010	100010
	000110	001110					101110	100110
	000111		011111			111111		100111
		001101		010101	110101		101101	
				010100	110100	111100		

SALIDA S0

	000	001	011	010	110	111	101	100
000		1	1			1	1	
001	1			1	1			1
011	1			1	1			1
010		1	1			1	1	
110		1	1			1	1	
111	1			1	1			1
101	1			1	1			1
100		1	1			1	1	

TÉRMINOS SELECCIONADOS S0								
		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

Se obtienen las tablas de minimización a partir de las tablas de selección.

Ahora se procede a minimizar la salida carry.

Tabla de resultados al eliminar el menos significativo vertical

						111001		
					110012	111012	101011	
			011112	010112	110112	111112	101112	100112
		0011111						
			011101		110102	111102	101102	100102

Tabla de resultados al analizar al que está enseguida del término adyacente vertical

						111001		
					110012	111012	101011	
			011112	010112	110122	111122	101122	100122
		0011111						
			011101					

Tabla de resultados de adyacencias horizontal

						111001		
					112012		101011	
			012112		112122		102122	
		0011111						
			011101					

Tabla de resultados al término del análisis horizontal del término siguiente al adyacente

						111001		
					112012		101011	
			012112		122122			
		0011111						
			011101					

El resultado final para la salida del carry es

$$\text{Carry} = 0011111 + 012112 + 011101 + 112012 + 122122 + 111001 + 101011$$

Si se sustituyen los valores por variables y recordando que los 2 son variables canceladas entonces la función minimizada queda:

$$\text{Carry} = \bar{a}\bar{b}cdef + \bar{a}bde + \bar{a}bcd\bar{e}f + ab\bar{d}e + ad + abc\bar{d}\bar{e}f + \bar{a}bcd\bar{e}f$$

Se presenta la secuencia de la minimización de la salida S2

TÉRMINOS SELECCIONADOS								
					110000	111000	101000	100000
			011001		110001		101001	100001
		001011	011011	010011				100011
			011010	010010			101010	100010
	000110	001110			110110	111110		
	000111				110111	111111	101111	
	000101	001101		010101		111101		
	000100	001100	011100	010100				

Tabla de resultados al eliminar el menos significativo vertical

						110002		101002
								100002
			011001					
		001011	011012	010012				100012
							101010	
	000112	001110			110112	111112		
							101111	
	000102	001102		010102		111101		

Tabla de resultados al analizar al que está enseguida del término adyacente vertical.

						110002		101002
								100022
			011001					
		001011	011012	010012				
							101010	
	000122	001110			110112	111112		
							101111	
		001102		010102		111101		

Tabla de resultados de adyacencias horizontal

					110002		101002	100022
				011001				
			001011	012012				
							101010	
	000122	001110			112112			
							101111	
		001102		010102		111101		

El resultado final para la salida de S2 es

$$S2 = 000122 + 001011 + 001110 + 001102 + 011001 + 012012 + 010102 + 110002 + 112112 + 111101 + 101002 + 101010 + 101111 + 100022$$

Si se sustituyen los valores por variables y recordando que los 2 son variables canceladas entonces la función minimizada queda:

$$S2 = \bar{a}\bar{b}\bar{c}d + \bar{a}\bar{b}c\bar{d}ef + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}de + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f} + \bar{a}\bar{b}cde\bar{f}$$

Se presenta la secuencia de la minimización de la salida S1

TABLA DE TÉRMINOS SELECCIONADOS								
			011000	010000	110000	111000		
		001001		010001	110001		101001	
	000011		011011			111011		100011
	000010	001010					101010	100010
	000110	001110					101110	100110
	000111		011111			111111		100111
		001101		010101	110101		101101	
				010100	110100	111100		

Tabla de resultados al eliminar el menos significativo vertical

			011000	010002	110002	111000		
		001001					101001	
	000012		011011			111011		100012
		001010					101010	
	000112	001110					101110	100112
			011111			111111		
		001101		010102	110102		101101	
						111100		

Tabla de resultados al analizar al que está enseguida del término adyacente vertical

			011000	010002	110002	111000		
		001001					101001	
	000012		011011			111011		100012
		001010					101010	
	000112	001110					101110	100112
			011111			111111		
		001101		010102	110102		101101	
						111100		
						111100		

TABLA DE RESULTADOS DE ADYACENCIAS HORIZONTAL								
			011000	010002	110002	111000		
		001001					101001	
	000012		011011			111011		100012
		001010					101010	
	000112	001110					101110	100112
			011111			111111		
		001101		010102	110102		101101	
						111100		

Tabla de resultados al término del análisis horizontal del término siguiente al adyacente

			011000	010002	110002	111000		
		001001					101001	
	000012		011011			111011		100012
		001010					101010	
	000112	001110					101110	100112
			011111			111111		
		001101		010102	110102		101101	
						111100		

Primera interacción de uno contra todos vertical y horizontal

Tabla de resultados de uno contra todos vertical

			011000	010202	110202	111200		
		001201					101201	
	000212		011211			111211		100212
		001210					101210	

Tabla de resultados de uno contra todos horizontalmente

			011000	210202		111200		
		201201						
	200212		211211					
		201210						

Segunda interacción de uno contra todos vertical y horizontal

Tabla de resultados de uno contra todos vertical

			011000	210202		111200		
		201221						
	200212		211211					

Tabla de resultados de uno contra todos horizontalmente

			011000	210202		111200		
		201221						
	200212		211211					

El resultado final para la salida de S2 es

$$S1 = 200212 + 201221 + 011000 + 211211 + 210202 + 111200$$

Si se sustituyen los valores por variables y recordando que los 2 son variables canceladas entonces la función minimizada queda:

$$S1 = \bar{b}\bar{c}e + \bar{b}cf + \bar{a}bcd\bar{e}\bar{f} + bcef + b\bar{c}\bar{e} + ab\bar{c}\bar{e}\bar{f}$$

Se presenta la secuencia de la minimización de la salida S0

TÉRMINOS SELECCIONADOS								
		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

Tabla de resultados al eliminar el menos significativo vertical

		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

Tabla de resultados al analizar al que está enseguida del término adyacente vertical

		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

Tabla de resultados de adyacencias horizontal

		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

Tabla de resultados del análisis horizontal del término siguiente adyacente

		001000	011000			111000	101000	
	000001			010001	110001			100001
	000011			010011	110011			100011
		001010	011010			111010	101010	
		001110	011110			111110	101110	
	000111			010111	110111			100111
	000101			010101	110101			100101
		001100	011100			111100	101100	

Primera interacción de uno contra todos vertical y horizontal

Tabla de resultados de uno contra todos vertical

		001020	011020			111020	101020	
	000021			010021	110021			100021
		001120	011120			111120	101120	
	000121			010121	110121			100121
		001100	011100					

Tabla de resultados de uno contra todos horizontal

		021020				121020	101020	
	020021				120021			
		021120				121120		
	020121			210121				100121
		021100						

Segunda interacción de uno contra todos vertical y horizontal

Tabla de resultados de uno contra todos vertical

		021220				121220	101020	
	020221				120021			

				210121				100121
		021100						

Tabla de resultados de uno contra todos horizontal

		221220						101020
	020221				120021			
				210121				100121
		021100						

Tercera interacción de uno contra todos vertical y horizontal

Tabla de resultados de uno contra todos vertical

		021220				121220	101020	
	020221				120021			
				210121				100121
		021100						

Tabla de resultados de uno contra todos horizontal

		221220						101020
	020221				120021			
				210121				100121
		021100						

El resultado final para la salida de S2 es

$$S_0 = 020221 + 221220 + 021100 + 210121 + 120021 + 101020 + 100121$$

Si se sustituyen los valores por variables y recordando que los 2 son variables canceladas entonces la función minimizada queda:

$$S0 = \bar{a}cf + c\bar{f} + \bar{a}cd\bar{e}\bar{f} + b\bar{c}df + a\bar{c}\bar{d}f + a\bar{b}c\bar{d}\bar{f} + a\bar{b}\bar{c}df$$

Nota: Casi todas las salidas se pueden minimizar más si se utiliza el *or* exclusivo.

5. Conclusiones

La simplificación y minimización en la electrónica digital es fundamental, ya que se abaratan los costos de implementación y el diseño suele ser menos tedioso, por esto se busca una mejor técnica para la minimización de funciones lógicas.

La técnica que se presenta es sencilla de aplicar y proporciona una buena simplificación, además no tiene pasos o procedimientos ocultos, la forma de cancelación es bastante conocida.

Por esto resulta una forma práctica de aplicar la simplificación y minimización por adyacencias. 

6. Bibliografía

M. MORRIS MANO

1982 Logica Digital y Diseño de Computadoras, 36-115 p.

VICTOR P. NELSON, H. TROY NAGLE, BILL D. CARROLL, J. DAVID IRWIN

1996 Análisis y Diseño de Circuitos Lógicos Digitales, 174-235 p.

ROBERT L. MORRIS Y JOHN R. MILLAR

1982 Diseño con Circuitos Integrados TTL, 133-169 p.

M.C. Vicente Sosa Ortiz

Universidad Tecnológica de Puebla