

Organización y Arquitectura de Computadoras 2019-2

Práctica 3: Circuitos Combinacionales

Sandra del Mar Soto Corderi
Edgar Quiroz Castañeda

3 de marzo del 2019

1 Ejercicios

1. Desarrolla un circuito que simule el comportamiento de la implicación lógica. Sólo puedes hacer uso de fuentes de alimentación power y ground, transistores tipo PNP y NPN y pines de entrada y salida.
Como es un operador binario de valores lógicos, entonces las entradas son dos variables lógicas $\in \{1, 0\}^2$ y la salida es otro valor lógico $\in \{1, 0\}$.
Luego, la implicación se define como sigue

Table 1: Tabla de verdad de la implicación

p	q	$p \implies q$
0	0	1
0	1	1
1	0	0
1	1	1

Y el mapa de Karnaugh correspondiente es

		p	
		0	1
q	0	1	0
	1	1	1

Por lo que la regla de correspondencia es

$$(p \implies q) = p' + q$$

2. Sean $x, y \in \{0, 1, 2, 3\}$. Desarrolla un comparador electrónico de 2 bits, las salidas del comparador deben ser
 - $x < y$
 - $x = y$
 - $x > y$

Podemos considerar los números del 0 al 3 en su representación binaria. Como sólo son dos dígitos, entonces cada número puede ser representado con dos variables lógicas, lo que da un total de 4 variables lógicas, una para cada bit

de cada número.

Y a su vez, hay tres salidas, una por cada relación que hay que modelar.

Table 2: Tabla de verdad de las relaciones anteriores

a	b	c	d	$ab < cd$	$ab = cd$	$ab > cd$
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

Los mapas de Karnaugh correspondientes son

- Igualdad

		ab			
		00	01	11	10
cd	00	1	0	0	0
	01	0	1	0	0
	11	0	0	1	0
	10	0	0	0	1

Por lo que la regla de correspondencia es

$$(ab = cd) = a'b'c'd' + a'bc'd + abcd + ab'cd'$$

- Menor que

		<i>ab</i>			
		00	01	11	10
<i>cd</i>	00	0	1	1	1
	01	0	0	1	1
	11	0	0	0	0
	10	0	0	1	0

Por lo que la regla de correspondencia es

$$(ab < cd) = ac' + bc'd' + abd'$$

- Mayor que

		<i>ab</i>			
		00	01	11	10
<i>cd</i>	00	0	0	0	0
	01	1	0	0	0
	11	1	1	0	1
	10	1	1	0	0

Por lo que la regla de correspondencia es

$$(ab < cd) = a'c + b'cd + a'b'd$$

2 Preguntas

1. ¿Cuál es el procedimiento a seguir para desarrollar un circuito que recuelva un problema que involucre lógica combinacional?
 - (a) Identificar las variables lógicas de entrada y las salidas lógicas.
 - (b) Construir la tabla de verdad del circuito.
 - (c) Minimizar la expresión que representa la tabla de verdad. Esto se puede hacer usando manipulación algebraica, mapas de Karnaugh, el algoritmo de Quine-McCluskey o el método de Expreso.
 - (d) Diseñar o especificar el circuito correspondiente a la expresión minimizada.

2. Si una función de conmutación se evalúa a más ceros que unos, ¿es conveniente usar mintérminos o maxtérminos? ¿Y en caso contrario?

Si se toman mintérminos, entonces la cantidad de términos en la expresión es la misma que la cantidad de casos donde la función es 1. Si se usan maxtérminos, entonces esta cantidad es la misma que la cantidad de casos donde la función es 0.

Por lo que si hay más unos, la expresión quedará más corta si se usan maxtérminos, y si hay más ceros, quedará más corta usando mintérminos.

Aunque si se minimiza la expresión, se debería obtener algo más o menos del mismo tamaño sin importar si se inició con mintérminos o maxtérminos.

3. En base al trabajo realizado, ¿cuáles son los inconvenientes de desarrollo de circuitos de forma manual?
El proceso de representación por expresiones y el proceso de minimización es tedioso, y la complejidad de los patrones y longitud de las expresiones aumenta exponencialmente con el número de variables. Entonces, hay un punto en el que no es viable hacerlo a mano.