PRÁCTICA 4

SIMPLIFICACIÓN DE FUNCIONES LÓGICAS MEDIANTE TABLAS DE KARNAUGH

OBJETIVOS:

Una vez finalizada la práctica deberemos ser capaces de:

- Construir tablas de Karnaugh de 3, 4 ó 5 variables.
- Representar una función expresada en forma de suma de productos o producto de sumas en una tabla de Karnaugh.
- Combinar los términos de una tabla para obtener una expresión mínima.
- Convertir una tabla de verdad en una tabla o mapa de Karnaugh para simplificar la correspondiente expresión.
- Utilizar el mapa de Karnaugh para convertir productos de sumas en suma de productos y viceversa.

REFERENCIAS:

- T.L. Floyd, *Fundamentos de Sistemas Digitales*, 9ª Edición, Capítulo 4, "Álgebra de Boole y Simplificación Lógica", secciones 4-8 a 4-11.
- J.Mª Angulo y J. García, *Sistemas Digitales y Tecnología de Computadores*, Capitulo 3 "Álgebra de Boole".
- Transparencias Tema 3 "Lógica Digital". Fundamentos de los Computadores.

ELEMENTOS NECESARIOS:

- Programa de Simulación LogiSim 2.7.1 con:
  - Diferentes tipos de puertas lógicas.
  - Pines de entrada (pin).
  - Herramienta de simplificación de funciones
  - Indicadores (ver).

INTRODUCCIÓN TEÓRICA

Entradas				Salida
A_2	A_1	B_2	B_1	X
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Tabla 1

Los circuitos combinacionales son circuitos lógicos en los que las salidas dependen únicamente de las variables de entrada. En los circuitos combinacionales simples se utilizan las tablas de verdad para describir todos los posibles valores de entrada y las salidas correspondientes; la tabla de verdad describe completamente el comportamiento de un circuito. Los circuitos deben materializarse a partir de la expresión simplificada de la función de salida obtenida a partir de la tabla de verdad.

Una de las técnicas más útiles de simplificación de circuitos lógicos combinacionales de que disponemos fue la desarrollada por M. Karnaugh en 1953. El método consiste en escribir la tabla de verdad dentro de una tabla en la que las celdas (cuadros) adyacentes difieren entre sí solamente en una variable (las celdas adyacentes poseen un borde común, bien sea horizontalmente o verticalmente). Cuando escribimos una tabla de Karnaugh, las variables se escriben siguiendo la secuencia del código Gray, tanto en la parte lateral como en la superior de la tabla. Los valores que adopta la salida aparecen con un ‘0’ ó ‘1’ en la celda correspondiente a su combinación de la tabla de verdad.

Como ejemplo, consideremos el diseño de un comparador de números de 2 bits. Las entradas las denominaremos A_2A_1 y B_2B_1. La salida estará a nivel ALTO cuando A_2A_1 sea mayor o igual a B_2B_1. Empezaremos por construir la tabla de verdad del circuito, que se muestra en la Tabla 6.1. En ella están perfectamente especificadas todas las posibles combinaciones de entrada, así como el valor que debe tomar la salida en cada caso.

A continuación dibujaremos la tabla de Karnaugh, como se muestra en la Figura 1. Los valores correspondientes a la salida los tomamos directamente de la tabla de verdad. La tabla está construida de manera que al agrupar los unos de las celdas adyacentes se obtiene una forma de suma de productos.

El tamaño de los grupos debe ser un entero potencia de 2 (es decir, 1, 2, 4, 8, etc.) y debe contener unos únicamente. Los grupos deben ser tan grandes como sea posible; todos los unos deben estar contenidos en algún grupo y pueden estar incluidos en más de uno si fuera necesario.

Una vez agrupados los unos en la tabla debemos determinar la función de salida. Cada grupo constituye un término producto de la función de salida simplificada. Las adyacencias dentro de cada grupo de celdas formado por más de un ‘1’, nos permitirán eliminar de la expresión de salida las *variables que cambien de valor entre las celdas agrupadas. Un* grupo de dos unos adyacentes tendrá

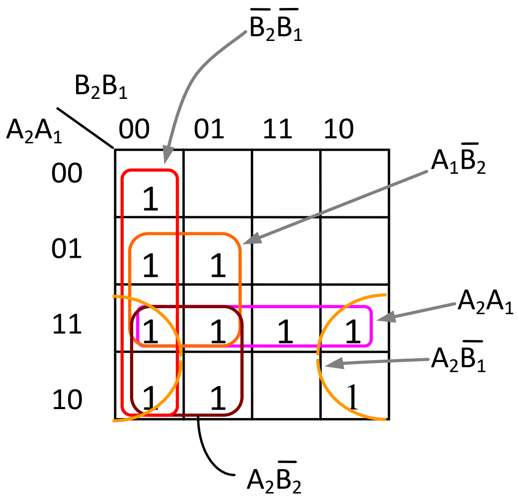

Figura 1

una sola adyacencia y sólo nos permitirá eliminar una variable. En un grupo de 4 unos podremos eliminar dos variables y en un grupo de 8, tres variables.

En la Figura 1 podemos ver las agrupaciones para el comparador de 2 bit. Puesto que en este caso todos los grupos son de 4 unos, cada término producto tendrá dos variables (se han eliminado dos de cada uno de ellos). La expresión resultante es la suma de todos los términos producto. El circuito que la implementa se muestra en la Figura 2.

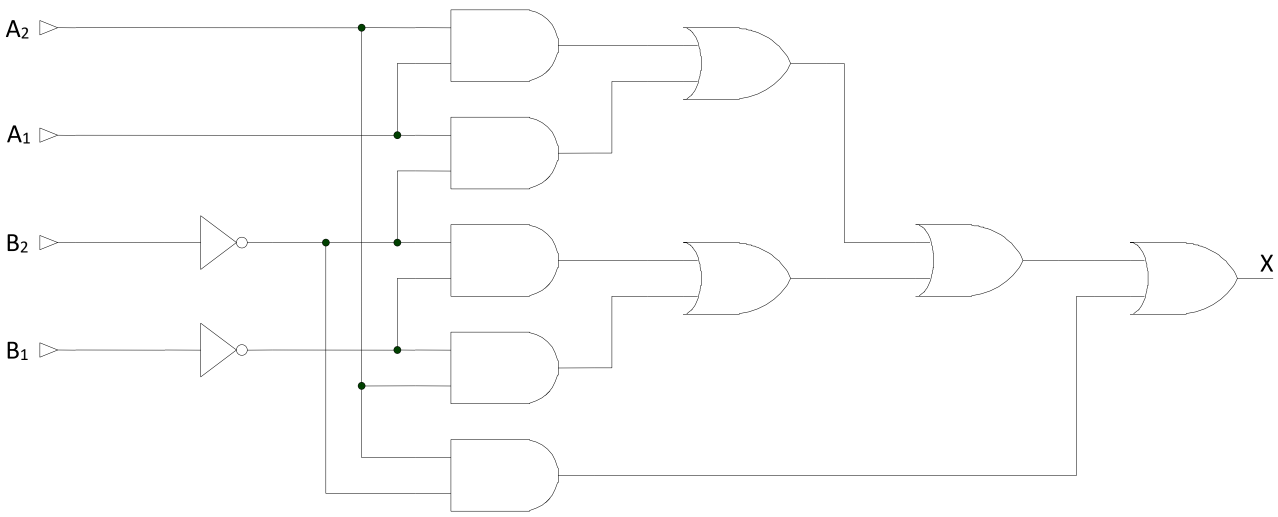

Figura 2

REALIZACIÓN PRÁCTICA

1. Se dispone de la función:

$$g = \sum_{4}(1,2,4,5,7,8,10,14) + \sum_{0}(0,11)$$

a) Escribe su tabla de verdad.

b) Obtén su equivalente en forma de producto de sumas.

c) Simplifica la tabla de verdad e indica la expresión obtenida.

d) Haciendo uso de LogiSim, implementa la función simplificada con el menor número de puertas básicas posible.

e) Comprueba que la tabla de verdad del circuito implementado coincide con la original.

2. Se desea diseñar un circuito lógico para generar una salida a nivel alto si y sólo si la entrada, representada por un número binario de cuatro bits, es múltiplo de dos o tres.

   a) Escribe su tabla de verdad

   b) Construye la tabla de Karnaugh asociada y agrupa los unos de la forma indicada en la introducción teórica. Obtén la expresión algebraica simplificada de la salida en forma de suma de productos.

   c) Implementa en LogiSim dicha expresión utilizando las puertas lógicas necesarias.

   d) Vuelve a simplificar la tabla de Karnaugh anterior de forma que la expresión quede ahora en forma de producto de sumas. Impleméntala y comprueba que su tabla de verdad coincide con la original.

3. Simplifica mediante tablas de Karnaugh la función:

$$f = \sum_{5}(0,3,7,10,11,12,13,14,26,27)$$

   a) Impleméntala haciendo uso del menor número de puertas posible.
   b) Comprueba que la salida del circuito es correcta para cada una de las combinaciones y obtén su tabla de verdad.