

SEMINARIO 4 MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

***TECNOLOGÍA Y ORGANIZACIÓN DE
COMPUTADORES***

1º Grado en Ingeniería Informática.

Autor: Pedro Martín Smith

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

- 1. Introducción.
- 2. Conceptos básicos.
- 3. Minimización monofuncional con Mapas de Karnaugh para estructuras AND/OR y NAND/NAND.
- 4. Minimización monofuncional con Mapas de Karnaugh para estructuras OR/AND y NOR/NOR.
- 5. Métodos de minimización.
- 6. Ejercicios.
- 7. APÉNDICE: Minimización Multifuncional.

- BIBLIOGRAFÍA:
Basado en: Hill, F.J., Peterson, G.R., *Teoría de Conmutación y Diseño Lógico*, Limusa Noriega, 1990.

MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

1. Introducción.

En el seminario 2 se trataron: a) el concepto de funciones de conmutación, tanto las completamente especificadas, como las incompletamente especificadas, b) los conceptos de términos producto y términos suma, c) sus representaciones en el mapa de Karnaugh y d) cómo identificarlos en dicho mapa y deducir sus expresiones algebraicas.

En este seminario, trataremos sobre la “minimización” de las funciones de conmutación usando los mapas de Karnaugh. En este contexto, “Minimizar” consistirá en obtener, de entre todas las posibles expresiones de una función de conmutación, aquella que suponga un menor costo en su realización física para estructuras de circuito a dos niveles de puertas lógicas.

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

2. Conceptos básicos.

Un término producto (o cubo) tiene como expresión un solo conjunto de literales operados con el producto lógico (AND). Por ejemplo, en un Algebra de conmutación de 4 variables (A,B,C,D) el cubo $P(5,13)=\{(0101),(1101)\}=(-101)$ tiene la expresión:

$$P(5,13) = B \cdot \bar{C} \cdot D \leftrightarrow -101$$

donde la variable A desaparece. Observe que este cubo toma el valor "1" en 2 casillas adyacentes del mapa de Karnaugh:

	AB	00	01	11	10	
CD	00	0	4	12	8	
	01	1	5	13	9	
	11	3	7	15	11	
	10	2	6	14	10	
			B			

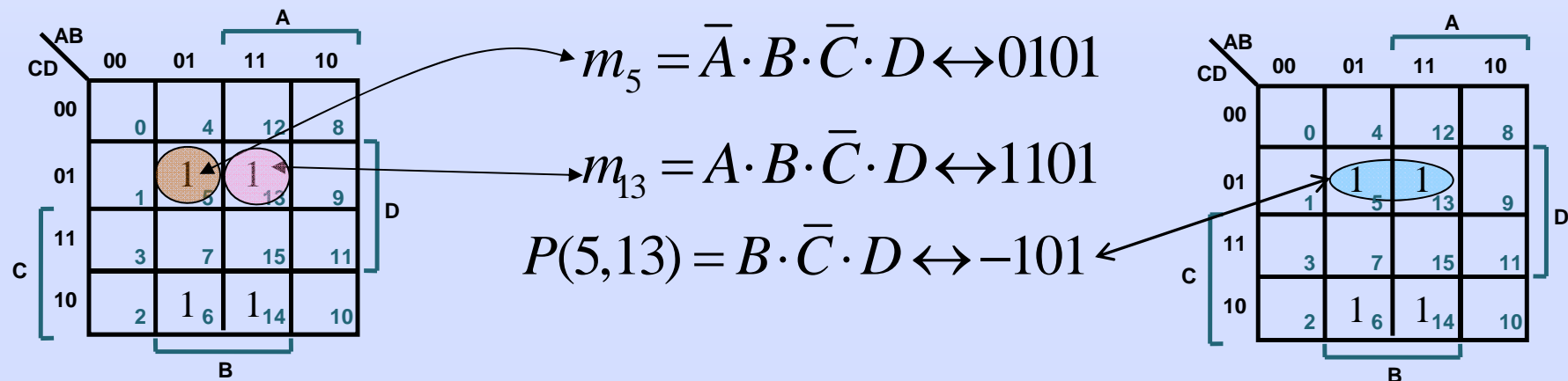
Cubos de orden k .

El número de minterms que forma un cubo siempre es potencia de 2. Diremos que un cubo es de orden k , cuando "desaparecen" k variables, de modo que el cubo toma el valor "1" en 2^k casillas del mapa de Karnaugh

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

2. Conceptos básicos.

Los mapas de Karnaugh se diseñaron para reconocer los cubos "términos producto" de un modo fácil gracias al hecho de que ciertas casillas adyacentes del mapa pueden combinarse. Por ejemplo, la pareja de minterms (cubos de orden 0), m_5 y m_{13} , son adyacentes en el mapa y pueden combinarse para formar el término producto $P(5,13)$.



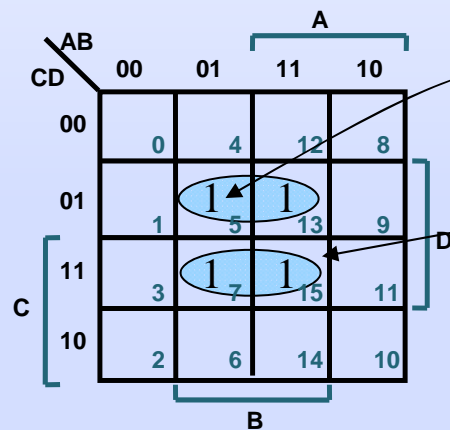
$$P(5,13) = \bar{A} \cdot B \cdot \bar{C} \cdot D + A \cdot B \cdot \bar{C} \cdot D = (\bar{A} + A) \cdot B \cdot \bar{C} \cdot D = 1 \cdot B \cdot \bar{C} \cdot D = B \cdot \bar{C} \cdot D$$

Teorema que justifica matemáticamente esto es: $Alg o = \bar{X} \cdot Alg o + X \cdot Alg o$

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

2. Conceptos básicos.

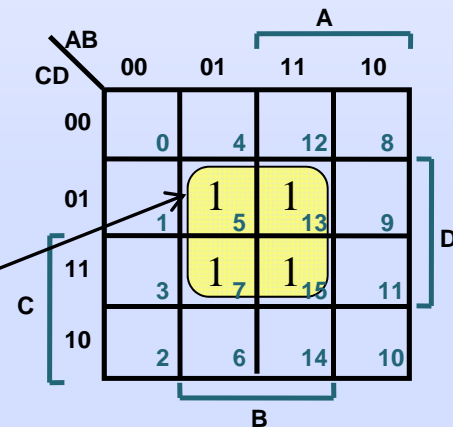
También, dos cubos que sean del mismo orden y adyacentes en el mapa pueden combinarse. Por ejemplo, la combinación de los cubos $P(5,13)$ y $P(7,15)$ dan lugar al cubo $P(5,7,13,15)$



$$P(5,13) = B \cdot \bar{C} \cdot D \leftrightarrow -101$$

$$P(7,15) = B \cdot C \cdot D \leftrightarrow -111$$

$$P(5,7,13,15) = B \cdot D \leftrightarrow -1-1$$



$$P(5,13) + P(7,15) = P(5,7,13,15)$$

$$P(5,7,13,15) = B \cdot \bar{C} \cdot D + B \cdot C \cdot D = (\bar{C} + C) \cdot B \cdot D = 1 \cdot B \cdot D = B \cdot D$$

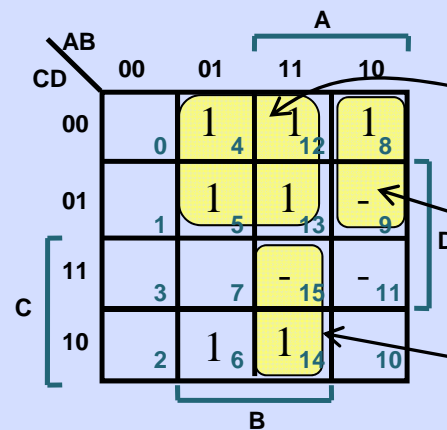
Teorema que justifica matemáticamente esto es: $Alg o = \bar{X} \cdot Alg o + X \cdot Alg o$

MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

2. Conceptos básicos.

Implicante de una función: Dada una función de conmutación f de n variables, se dice que un cubo es implicante de la función, si para aquellos valores de las variables en los que el cubo toma el valor 1, resulta que la función toma el valor **1** o corresponde a una indiferencia de f .

A	B	C	D		f
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	2	0
0	0	1	1	3	0
0	1	0	0	4	1
0	1	0	1	5	1
0	1	1	0	6	1
0	1	1	1	7	0
1	0	0	0	8	1
1	0	0	1	9	-
1	0	1	0	10	0
1	0	1	1	11	-
1	1	0	0	12	1
1	1	0	1	13	1
1	1	1	0	14	1
1	1	1	1	15	-



Ejemplos de
implicantes de f

$P(4,5,12,13)$

$P(8,9)$

$P(14,15)$

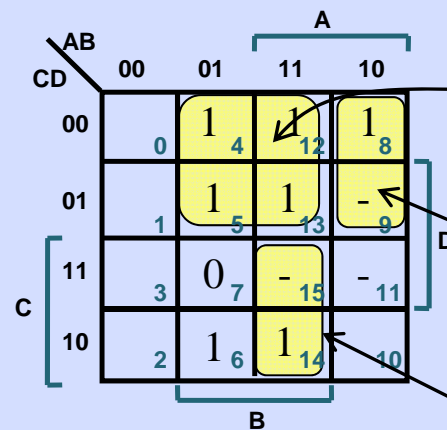
MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

2. Conceptos básicos.

Implicante primo de una función: Dada una función de conmutación f y un implicante de f de orden k , se dice que es **implicante primo de f** , si no existe otro implicante de f de orden p con $p > k$, tal que éste incluya al primero.

Ejemplos de implicantes primos de f y NO primos de f

A	B	C	D		f
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	2	0
0	0	1	1	3	0
0	1	0	0	4	1
0	1	0	1	5	1
0	1	1	0	6	1
0	1	1	1	7	0
1	0	0	0	8	1
1	0	0	1	9	-
1	0	1	0	10	0
1	0	1	1	11	-
1	1	0	0	12	1
1	1	0	1	13	1
1	1	1	0	14	1
1	1	1	1	15	-



P(4,5,12,13) **Sí** es implicante **primo de f**

P(8,9) **No** es implicante **primo de f**

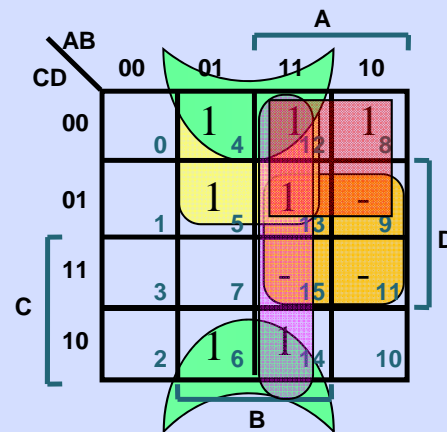
P(14,15) **No** es implicante **primo de f**

MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

2. Conceptos básicos.

Implicante primo esencial de una función: Dada una función de conmutación f y un implicante primo de f : Se dice que este ***implicante primo es esencial de f*** , si dicho implicante contiene, al menos, un minterm de f que no esta contenido en ningún otro implicante primo de f .

A	B	C	D		f
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	2	0
0	0	1	1	3	0
0	1	0	0	4	1
0	1	0	1	5	1
0	1	1	0	6	1
0	1	1	1	7	0
1	0	0	0	8	1
1	0	0	1	9	-
1	0	1	0	10	0
1	0	1	1	11	-
1	1	0	0	12	1
1	1	0	1	13	1
1	1	1	0	14	1
1	1	1	1	15	-



Ejemplo de todos los implicantes primos de f .

- P(4,**5**,12,13) Si esencial
- P(**8**,9,12,13) Si esencial
- P(9,11,13,15) No esencial
- P(12,13,14,15) No esencial
- P(4,**6**,12,14) Si esencial

MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

2. Conceptos básicos.

Cobertura (o Cubrimiento) de una función como suma de productos: Dada una función de conmutación $f(X_n, \dots, X_1, X_0)$ y un conjunto $\mathbf{C} = \{P_1, P_2, \dots, P_m\}$ de implicantes de f : Se dice que \mathbf{C} "**cubre**" a f , si para cualquier minterm de f existe un implicante P_i de \mathbf{C} que lo contiene. Por tanto, dicha función podrá obtenerse como suma lógica de los implicantes P_i del conjunto \mathbf{C} , es decir:

$$f(X_n, \dots, X_1, X_0) = P_1 + P_2 + \dots + P_m$$

Obsérvese que, en general, pueden existir muchos conjuntos \mathbf{C} diferentes y por tanto diversas expresiones alternativas para f que son equivalentes.

Para minimizar f es necesario que este conjunto \mathbf{C} sea de coste mínimo.

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

3. Minimización monofuncional con Mapas K. para estructuras AND/OR y NAND/NAND.

-En la realización de una función f , una posibilidad consiste en “**cubrir**” con “**términos producto p_i** ” todos los “unos” de f , pudiendo incluir o no indiferencias según convenga. Haciendo esto, se consigue una expresión de f como suma lógica de los términos p_i , y por tanto realizable con una estructura de dos niveles de puertas lógicas **AND/OR**.

-Para que la realización sea de menor costo (menor número de puertas lógicas y menor número de entradas por cada puerta lógica):

- a) Los **términos producto p_i** deben ser “**implicantes primos de f** ”.
- b) De todos los implicantes de f , debe **seleccionarse el menor número de ellos**, tal que cubra a la función f .
- c) La solución ha de incluir **todos los implicantes primos esenciales** de la función.

- De la realización **AND/OR** de f , se puede deducir la realización **NAND/NAND**, complementando 2 veces la expresión **AND/OR** y aplicando las leyes de **DEMORGAN**.

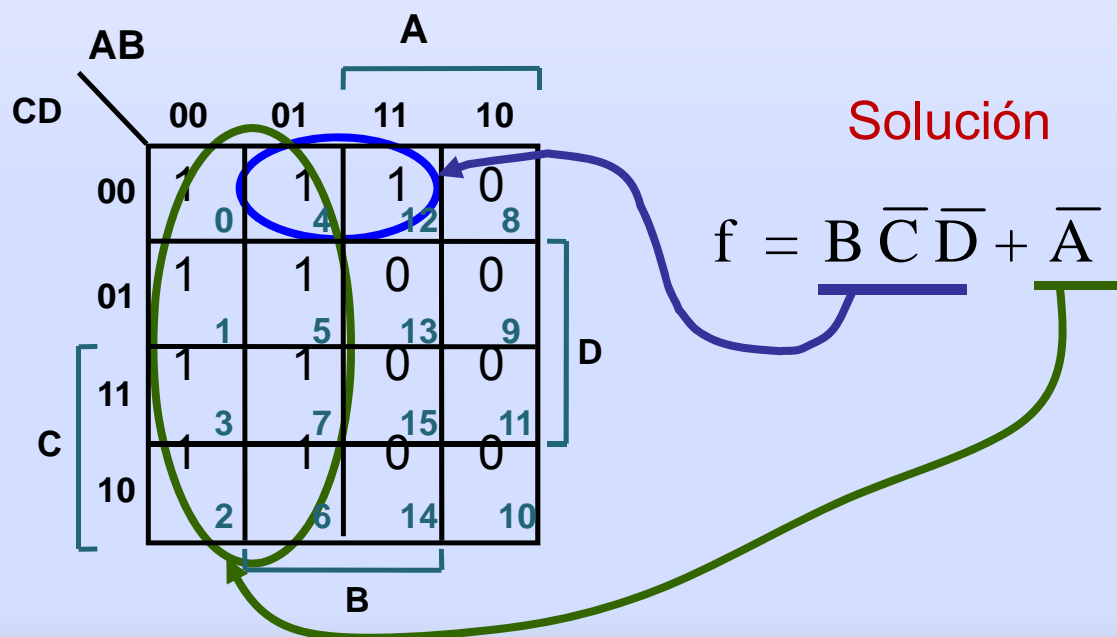
Ejemplo 1 sencillo de minimización y realización de una función con estructura AND/OR y su conversión a NAND/NAND

1) Minimizar con estructura AND/OR la función:

$$f(A,B,C,D) = \sum m_i(0,1,2,3,4,5,6,7,12)$$

A B C D		f
0 0 0 0	0	1
0 0 0 1	1	1
0 0 1 0	2	1
0 0 1 1	3	1
0 1 0 0	4	1
0 1 0 1	5	1
0 1 1 0	6	1
0 1 1 1	7	1
1 0 0 0	8	0
1 0 0 1	9	0
1 0 1 0	10	0
1 0 1 1	11	0
1 1 0 0	12	1
1 1 0 1	13	0
1 1 1 0	14	0
1 1 1 1	15	0

Tabla de Verdad



2) Realizar la función f con estructura de puertas lógicas AND/OR y convertirla a la estructura NAND/NAND

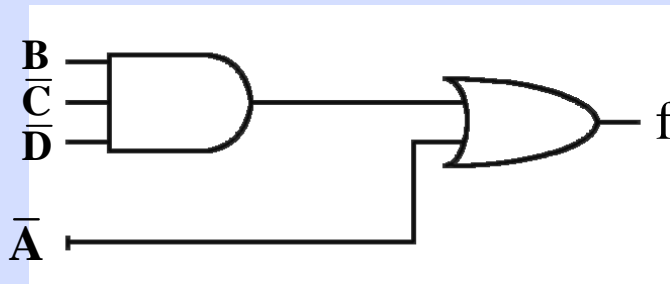
$$f(A,B,C,D) = \sum m_i(0,1,2,3,4,5,6,7,12)$$

Una vez se tiene la realización **AND/OR** de f , su conversión a **NAND/NAND** se hace complementando 2 veces la expresión AND/OR de f y aplicando las leyes de *DEMORGAN*.

Solución

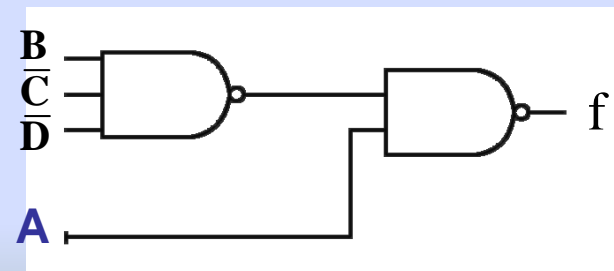
AND/OR

$$f = B \bar{C} \bar{D} + \bar{A}$$



NAND/NAND

$$\begin{aligned} f &= \overline{\overline{B \bar{C} \bar{D} + \bar{A}}} = \\ &= \overline{\overline{B \bar{C} \bar{D}} \cdot \overline{\bar{A}}} = \overline{\overline{B \bar{C} \bar{D}} \cdot A} \end{aligned}$$



MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

4. Minimización monofuncional con Mapas K. para estructuras OR/AND y NOR/NOR.

En la realización de una función f , con estructura **OR/AND** lo que se hace es "**cubrir todos los ceros de f** ", en vez de los unos, con **términos suma S_i** , pudiendo incluir o no indiferencias de la función según convenga.

-Una cobertura de la función consiste ahora en obtener un conjunto S de términos suma $S = \{S_1, S_2, \dots, S_m\}$, tal que la función pueda expresarse como producto de dichos términos suma:

$$f(X_n, \dots, X_1, X_0) = S_1 \cdot S_2 \cdot \dots \cdot S_m$$

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

El aspecto en el mapa de Karnaugh de los términos suma son análogos a los términos producto, de hecho un término suma se puede conseguir como el complemento de un término producto.

Ejemplo:

En un Álgebra de conmutación de 4 variables (A,B,C,D), obtener el término suma $S_i = (\bar{A} + B + C)$ como complemento de un término producto P_i .

Solución: $S_i = (\bar{A} + B + C) = \overline{\overline{\bar{A} + B + C}} = \overline{\overline{\bar{A}} \cdot \overline{B} \cdot \overline{C}} = \overline{A \cdot \bar{B} \cdot \bar{C}} = \bar{P}_i$

		A			
AB		00	01	11	10
CD	00	1 ₀	1 ₄	1 ₁₂	0 ₈
	01	1 ₁	1 ₅	1 ₁₃	0 ₉
	11	1 ₃	1 ₇	1 ₁₅	1 ₁₁
	10	1 ₂	1 ₆	1 ₁₄	1 ₁₀
C		D			
		B			

$S_i = \bar{P}_i$

		A			
AB		00	01	11	10
CD	00	0 ₀	0 ₄	0 ₁₂	1 ₈
	01	0 ₁	0 ₅	0 ₁₃	1 ₉
	11	0 ₃	0 ₇	0 ₁₅	0 ₁₁
	10	0 ₂	0 ₆	0 ₁₄	0 ₁₀
C		D			
		B			

P_i

MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

4. Minimización monofuncional con Mapas K. para estructuras OR/AND y NOR/NOR.

-Un **método rápido** para conseguir una realización **OR/AND** de una función, sin introducir nuevas definiciones consiste en lo siguiente:

- 1) Especificar la función **f** y **el complemento de f** en una tabla de verdad y representar el **complemento de f** en un mapa de Karnaugh.
- 2) Minimizar el **complemento de f** como si se tratase de una minimización con estructura **AND/OR**.
- 3) Sobre la expresión mínima **AND/OR** del complemento de **f** volver a complementar para rescatar la función original **f** y aplicar las leyes de DEMORGAN, de modo que se exprese como **producto de sumas**, con lo que se obtiene la realización **OR/AND**.

-De la realización **OR/AND** de **f** , se puede deducir la realización **NOR/NOR**, complementando 2 veces la expresión **AND/OR** y aplicando las leyes de DEMORGAN.

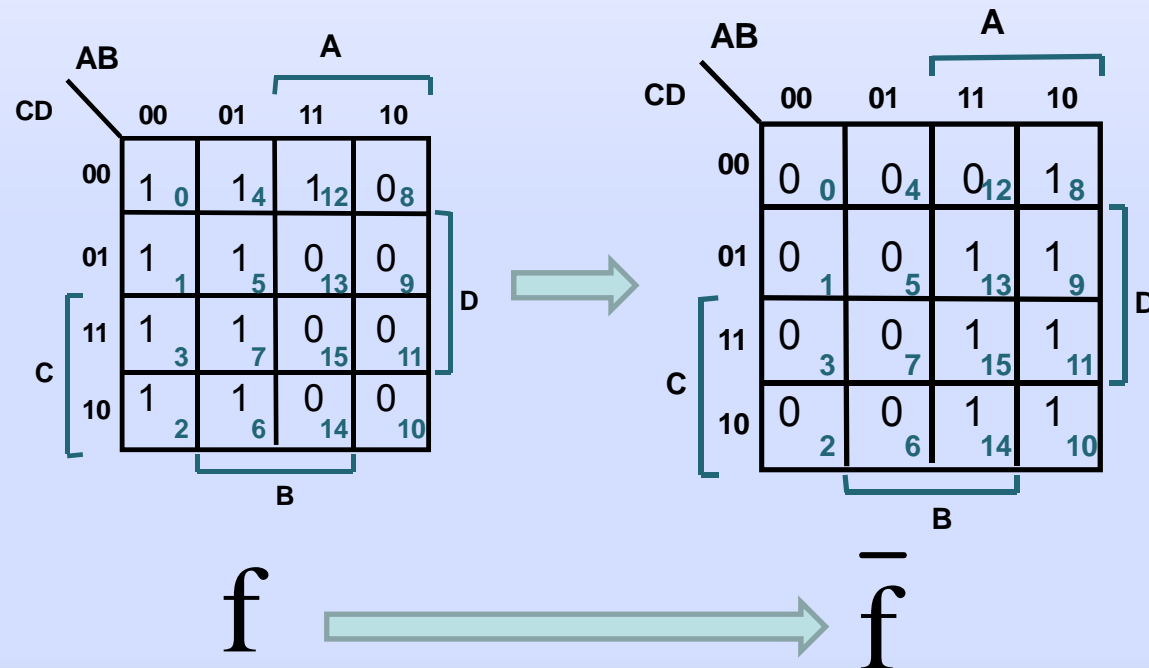
Ejemplo 1 sencillo de minimización y realización de una función con estructura **OR/AND** y su conversión a **NOR/NOR**

1) Especificar la función f y el complemento de f en una tabla de verdad y representar el complemento de f en un mapa de Karnaugh.

A B C D		f	\overline{f}
0 0 0 0	0	1	0
0 0 0 1	1	1	0
0 0 1 0	2	1	0
0 0 1 1	3	1	0
0 1 0 0	4	1	0
0 1 0 1	5	1	0
0 1 1 0	6	1	0
0 1 1 1	7	1	0
1 0 0 0	8	0	1
1 0 0 1	9	0	1
1 0 1 0	10	0	1
1 0 1 1	11	0	1
1 1 0 0	12	1	0
1 1 0 1	13	0	1
1 1 1 0	14	0	1
1 1 1 1	15	0	1

Tabla de Verdad

$$f(A,B,C,D) = \sum m_i(0,1,2,3,4,5,6,7,12)$$



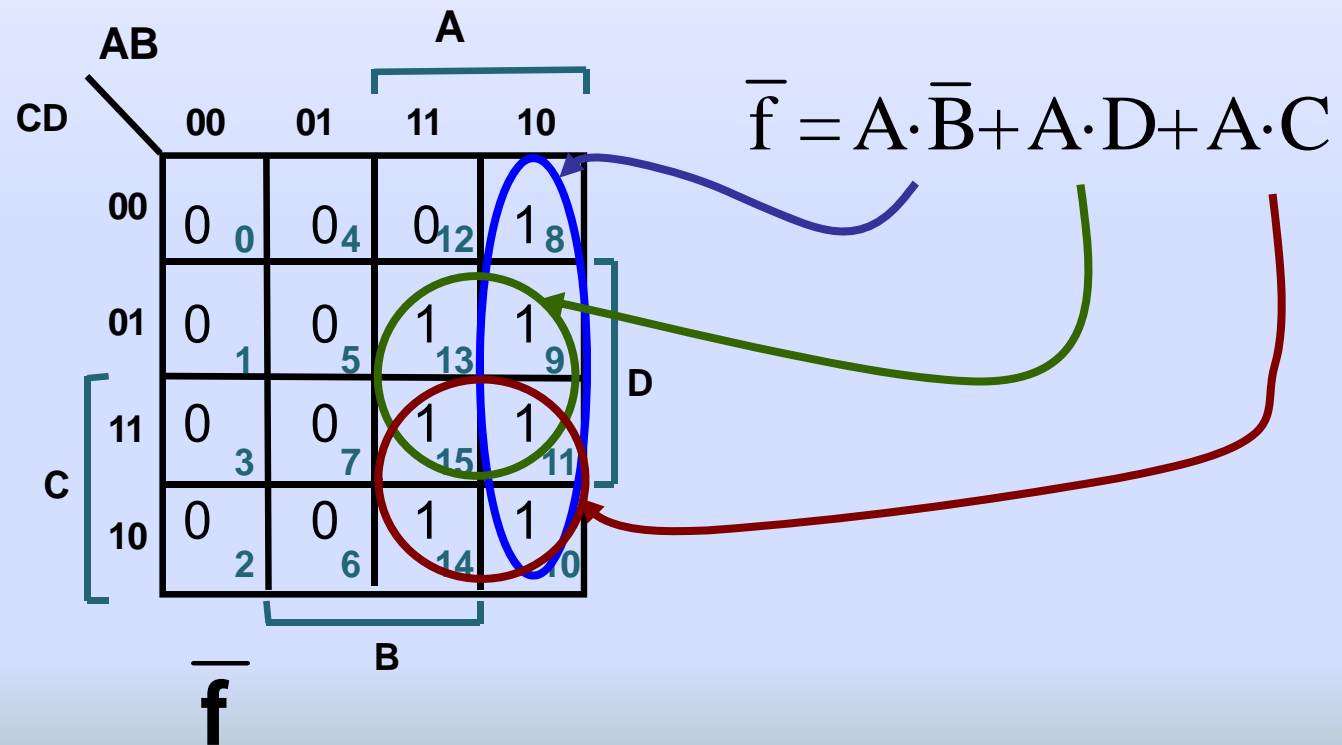
Ejemplo 1 sencillo de minimización y realización de una función con estructura **OR/AND** y su conversión a **NOR/NOR**

2) Minimizar el complemento de f como si se tratase de una minimización con estructura AND/OR

A B C D		f	\overline{f}
0 0 0 0	0	1	0
0 0 0 1	1	1	0
0 0 1 0	2	1	0
0 0 1 1	3	1	0
0 1 0 0	4	1	0
0 1 0 1	5	1	0
0 1 1 0	6	1	0
0 1 1 1	7	1	0
1 0 0 0	8	0	1
1 0 0 1	9	0	1
1 0 1 0	10	0	1
1 0 1 1	11	0	1
1 1 0 0	12	1	0
1 1 0 1	13	0	1
1 1 1 0	14	0	1
1 1 1 1	15	0	1

Tabla de Verdad

$$f(A,B,C,D) = \sum m_i(0,1,2,3,4,5,6,7,12)$$



Ejemplo 1 sencillo de minimización y realización de una función con estructura **OR/AND** y su conversión a **NOR/NOR**

3) Sobre la expresión mínima AND/OR del complemento de f volver a complementar para rescatar la función original f , aplicando las leyes de DEMORGAN

A B C D		f	\overline{f}
0 0 0 0	0	1	0
0 0 0 1	1	1	0
0 0 1 0	2	1	0
0 0 1 1	3	1	0
0 1 0 0	4	1	0
0 1 0 1	5	1	0
0 1 1 0	6	1	0
0 1 1 1	7	1	0
1 0 0 0	8	0	1
1 0 0 1	9	0	1
1 0 1 0	10	0	1
1 0 1 1	11	0	1
1 1 0 0	12	1	0
1 1 0 1	13	0	1
1 1 1 0	14	0	1
1 1 1 1	15	0	1

\overline{f}
↓
 f

$$\overline{f} = A \cdot \overline{B} + A \cdot D + A \cdot C$$



$$f = \overline{\overline{f}} = \overline{A \cdot \overline{B} + A \cdot D + A \cdot C} = \overline{(A \cdot \overline{B}) \cdot (A \cdot D) \cdot (A \cdot C)}$$

$$f = \overline{(A \cdot \overline{B}) \cdot (A \cdot D) \cdot (A \cdot C)} = (\overline{A} + B) \cdot (\overline{A} + \overline{D}) \cdot (\overline{A} + \overline{C})$$

4) Realizar la función f con estructura de puertas lógicas OR/AND y convertirla a la estructura NOR/NOR

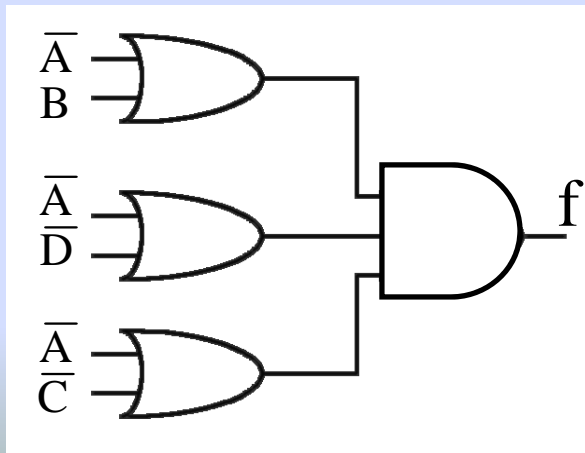
$$f(A,B,C,D) = \sum m_i(0,1,2,3,4,5,6,7,12)$$

Una vez se tiene la realización **OR/AND** de f , su conversión a **NOR/NOR** se hace complementando 2 veces la expresión OR/AND de f y aplicando las leyes de *DEMORGAN*.

Solución

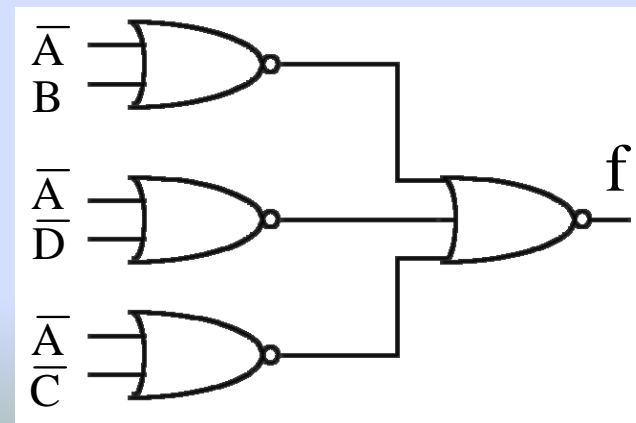
OR/AND

$$f = (\bar{A} + B)(\bar{A} + \bar{D})(\bar{A} + \bar{C})$$



NOR/NOR

$$\begin{aligned} f &= \overline{\overline{(\bar{A} + B)(\bar{A} + \bar{D})(\bar{A} + \bar{C})}} \\ &= \overline{\overline{(\bar{A} + B)} + \overline{\overline{(\bar{A} + \bar{D})}} + \overline{\overline{(\bar{A} + \bar{C})}}} \end{aligned}$$



MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

5. Métodos de minimización.

Introducción.

- Por razones pedagógicas, en este apartado trataremos 2 métodos, el primero, que denominaremos "**Método intuitivo**" servirá para la mayoría de los casos sencillos que se dan en la asignatura TOC. En este método se utilizan los mapas de Karnaugh, nuestra habilidad para reconocer los implicants primos esenciales, y aquellos opcionales para el cubrimiento de la función de conmutación a realizar.
- El segundo método, que denominamos "**Método riguroso**", se basa en el método algorítmico de "Quine-McCluskey". La diferencia con dicho método es que utilizaremos los mapas de Karnaugh para identificar visualmente los implicants primos de la función, en vez de la habitual construcción algorítmica de implicants primos partiendo de los minterms de la función, implicants de orden 1, orden 2,...,orden n.

MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

5. Métodos de minimización.

5.1 Método intuitivo con mapas de Karnaugh.

Para casos sencillos, este método obtendrá buenos resultados de minimización. Ilustramos los pasos a seguir con un ejemplo.

5.2 Método riguroso.

Para casos más complejos, este método es más seguro, dado que se basa en el algoritmo determinista de “**Quine-McCluskey**”

Existen numerosos programas de optimización (no deterministas) para tratar con la complejidad del problema de minimización funcional (NP completo), cuando el un número de variables de la función aumenta.

5.1 Método intuitivo. Ejemplo 2 de minimización.

Paso 1) Especificación de la función.

Ejemplo : Obtener una realización mínima AND/OR de la función :

$$F(X_3X_2X_1X_0) = \sum m_i(0,1,2,7,8,9,10,12,13,14,15)$$

$X_3 X_2 X_1 X_0$		f
0 0 0 0	0	1
0 0 0 1	1	1
0 0 1 0	2	1
0 0 1 1	3	0
0 1 0 0	4	0
0 1 0 1	5	0
0 1 1 0	6	0
0 1 1 1	7	1
1 0 0 0	8	1
1 0 0 1	9	1
1 0 1 0	10	1
1 0 1 1	11	0
1 1 0 0	12	1
1 1 0 1	13	1
1 1 1 0	14	1
1 1 1 1	15	1

Paso 1) Se especifica la función mediante una “Tabla de Verdad”.

En la tabla es conveniente introducir una columna con los equivalentes en decimal de las combinaciones de los valores binarios que toman las entradas. Esto facilita la identificación de los valores de f en las celdas del mapa de Karnaugh.

5.1 Método intuitivo. Ejemplo 2 de minimización.

Paso 2) Mapa de Karnaugh de la función

Ejemplo : Obtener una realización mínima AND/OR de la función :

$$f(X_3 X_2 X_1 X_0) = \sum m_i(0,1,2,7,8,9,10,12,13,14,15)$$

$X_3 X_2 X_1 X_0$		f
0 0 0 0	0	1
0 0 0 1	1	1
0 0 1 0	2	1
0 0 1 1	3	0
0 1 0 0	4	0
0 1 0 1	5	0
0 1 1 0	6	0
0 1 1 1	7	1
1 0 0 0	8	1
1 0 0 1	9	1
1 0 1 0	10	1
1 0 1 1	11	0
1 1 0 0	12	1
1 1 0 1	13	1
1 1 1 0	14	1
1 1 1 1	15	1

Tabla de Verdad

		X_3			
		$X_3 X_2$			
		00	01	11	10
$X_1 X_0$	00	1 ₀	4	1 ₁₂	1 ₈
	01	1 ₁	5	1 ₁₃	1 ₉
	11	3	1 ₇	1 ₁₅	1 ₁₁
	10	1 ₂	6	1 ₁₄	1 ₁₀
		X_2			

Mapa de Karnaugh

5.1 Método intuitivo. Ejemplo 2 de minimización.

Paso 3) Identificar los implicantes primos esenciales de la función.

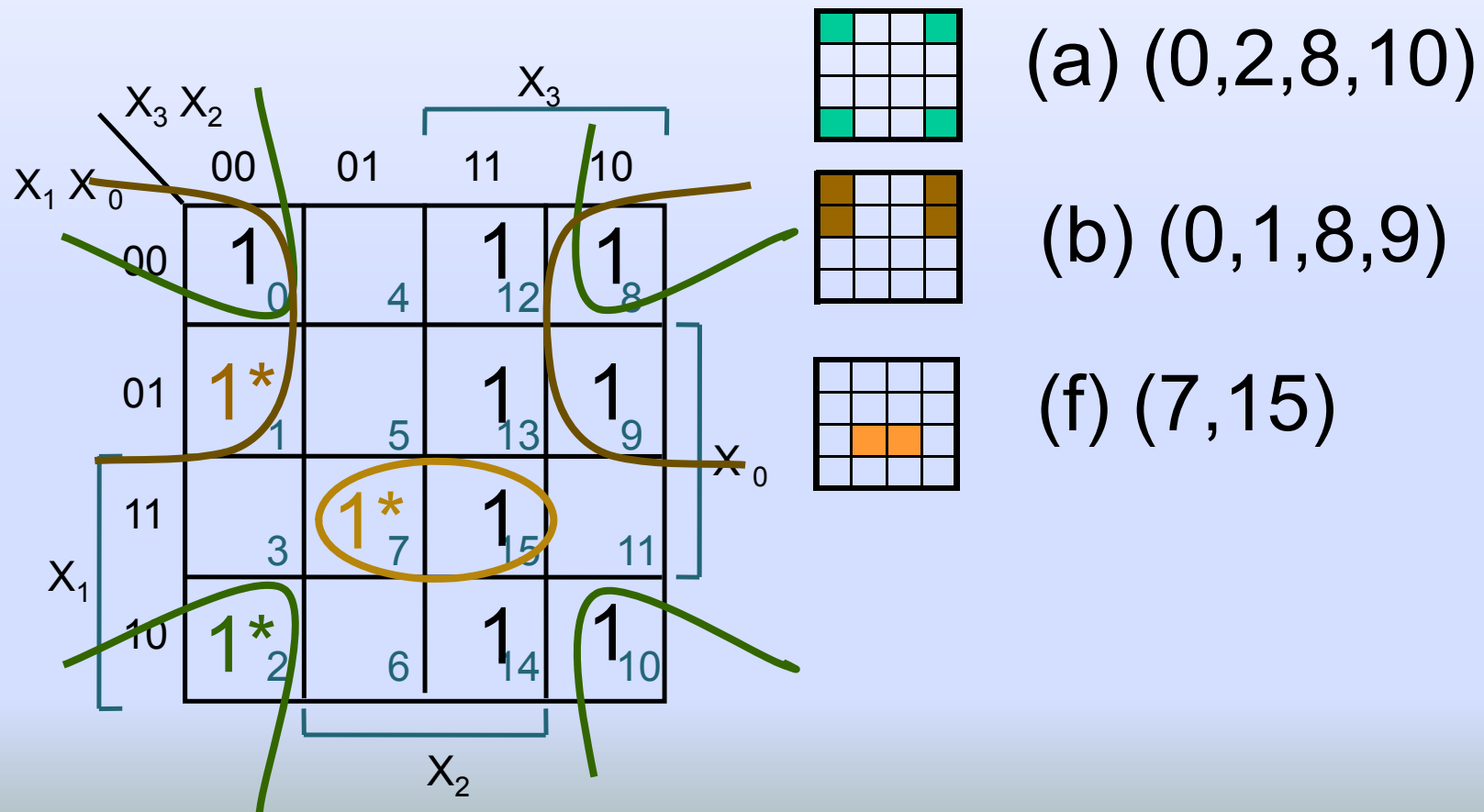
$X_3 X_2$		00	01	11	10		
$X_1 X_0$	00	1 ₀	4	1 ₁₂	1 ₈		
	01	1* ₁	5	1 ₁₃	1 ₉		
	11	3	1* ₇	1 ₁₅	11		
	10	1* ₂	6	1 ₁₄	1 ₁₀		
				X_2			

Mentalmente, nos fijamos en celdas del mapa con “1”_s de f tal que: Para cada una de estas celdas:

- Imaginamos los implicantes primos de f que cubren a la celda considerada.
- Descubrimos una de estas celdas donde sólo un único implicante primo de f cubre a esa celda. Entonces, este implicante primo es **ESENCIAL** y hay que seleccionarlo.
- Continuar los pasos anteriores hasta descubrir **todos los** implicantes primos **esenciales de f .**

5.1 Método intuitivo. Ejemplo 2 de minimización.

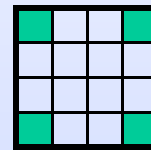
Paso 3a) Comprobamos que hemos identificado correctamente los implicantes primos esenciales de f .



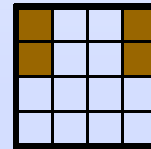
5.1 Método intuitivo. Ejemplo 2 de minimización.

Paso 3b) Por claridad, ponemos “indiferencias” en las casillas con unos que han sido cubiertos por los implicantes primos ESENCIALES.

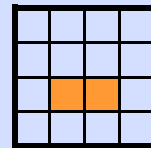
		X_3			
		$X_3 X_2$			
		00	01	11	10
X_1	00	- 0	4	1 12	- 8
	01	- 1	5	1 13	- 9
	11	3	- 7	- 15	11
	10	- 2	6	1 14	- 10



(a) (0,2,8,10)



(b) (0,1,8,9)



(f) (7,15)

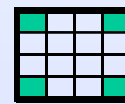
Vemos que todavía hay “unos” sin cubrir.

Tenemos que elegir el menor número de Implicantes primos que los cubra.

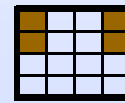
5.1 Método intuitivo. Ejemplo 2 de minimización.

Paso 4) Seleccionar ahora el resto de implicantes primos de F para "cubrir" totalmente a la función.

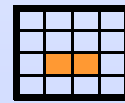
		X ₃			
		X ₂		X ₃	
		00	01	11	10
X ₁	X ₀	00	01	11	10
	00	- 0	4	12	- 8
	01	1	5	13	- 9
	11	3	- 7	- 15	11
X ₁	10	- 2	6	14	- 10



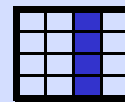
(a) (0,2,8,10)



(b) (0,1,8,9)



(f) (7,15)



(d) (12,13,14,15)

$$F = (a) + (b) + (f) + (d)$$

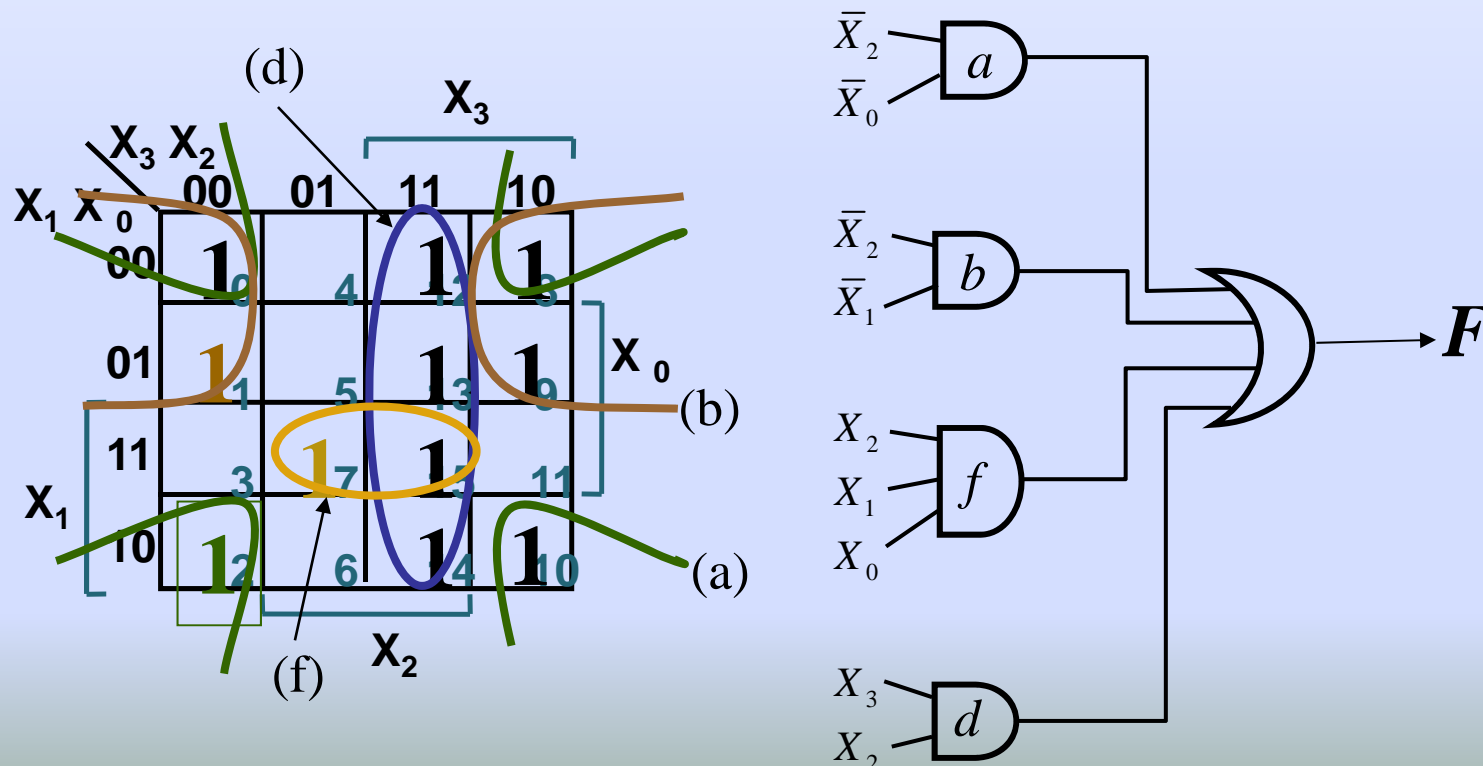
EN ESTE CASO ¡YA! TENEMOS LA SOLUCIÓN

¡No existe una solución que cubra todos los unos con un número inferior de implicantes primos!

5.1 Método intuitivo. Ejemplo 2 de minimización.

Paso 5) Obtener la expresión algebraica de los implicants primos seleccionados en la cobertura de F y realizar el esquemático del circuito con puertas lógicas

$$F = a + b + f + d = \bar{X}_2\bar{X}_0 + \bar{X}_2\bar{X}_1 + X_2X_1X_0 + X_3X_2$$



MINIMIZACIÓN DE FUNCIONES DE COMPUTACIÓN.

5.2 Método riguroso.

Pasos a seguir en la minimización monofuncional.

- 1) Especificación de la función
- 2) Mapa de Karnaugh de la función.
- 3) Identificación de TODOS los implicants primos.
- 4) Construir la Tabla de implicants primos.
- 5) Selección de implicants primos esenciales.
- 6) Si es necesario, construir la tabla de implicants primos reducida y aplicar la regla de dominancia de filas para obtener los implicants primos esenciales secundarios.
- 7) Obtener un número mínimo de implicants primos opcionales para una total cobertura de la función.
- 8) Obtener las expresiones de los implicants primos seleccionados, expresar F algebraicamente con dichos implicants y realizar el esquema del circuito con puertas lógicas.

5.2 Método Riguroso. Ejemplo 2 de minimización.

Pasos 1 y 2) Especificar Tabla de verdad y Mapa de Karnaugh de la función.

Ejemplo: Obtener una realización mínima AND/OR de la función :

$$F(X_3 X_2 X_1 X_0) = \sum m_i(0,1,2,7,8,9,10,12,13,14,15)$$

Mapa de Karnaugh

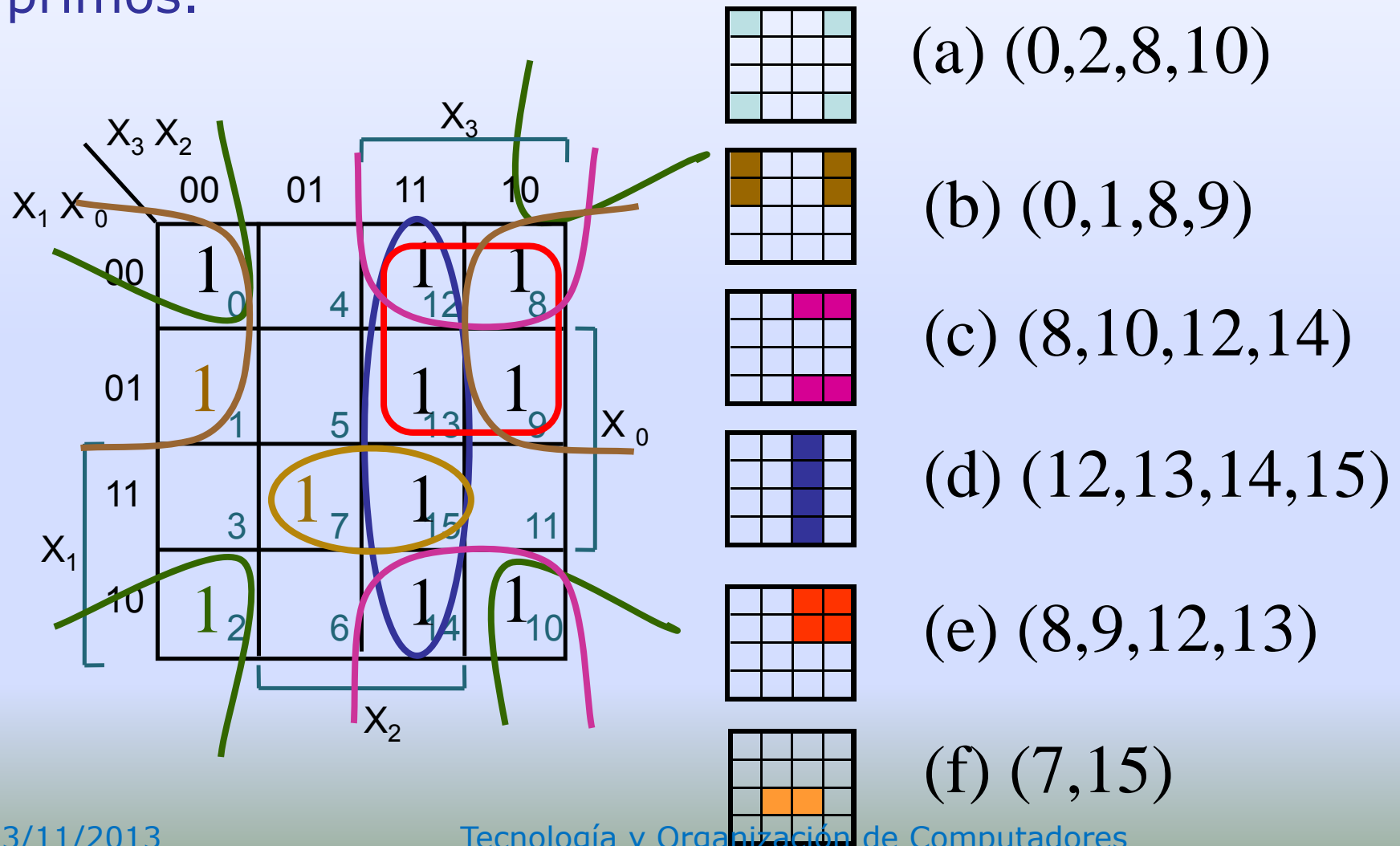
Tabla de Verdad

$X_3 X_2 X_1 X_0$		f
0 0 0 0	0	1
0 0 0 1	1	1
0 0 1 0	2	1
0 0 1 1	3	0
0 1 0 0	4	0
0 1 0 1	5	0
0 1 1 0	6	0
0 1 1 1	7	1
1 0 0 0	8	1
1 0 0 1	9	1
1 0 1 0	10	1
1 0 1 1	11	0
1 1 0 0	12	1
1 1 0 1	13	1
1 1 1 0	14	1
1 1 1 1	15	1

$X_3 X_2$		X_3			
		00	01	11	10
$X_1 X_0$	00	1 ₀	4	1 ₁₂	1 ₈
	01	1 ₁	5	1 ₁₃	1 ₉
	11	3	1 ₇	1 ₁₅	11
	10	1 ₂	6	1 ₁₄	1 ₁₀

5.2 Método Riguroso. Ejemplo 2 de minimización.

Paso 3) Identificación de **TODOS** los implicantes primos.



5.2 Método Riguroso. Ejemplo 2 de minimización.

Pasos 4 y 5) Tabla de implicants primos y selección de implicants primos esenciales.

Implicants
primos
esenciales

	0	1	2	7	8	9	10	12	13	14	15
a*	x		x*		x		x				
b*	x	x*			x	x					
c					x		x	x		x	
d								x	x	x	x
e					x	x		x	x		
f*				x*							x
	x	x	x	x	x	x	x				x

5.2 Método Riguroso. Ejemplo 2 de minimización.

	12	13	14
C _{**}	x - - - - x -		
d	x	x	x
e	x - - x - - - -		
	d	d	d

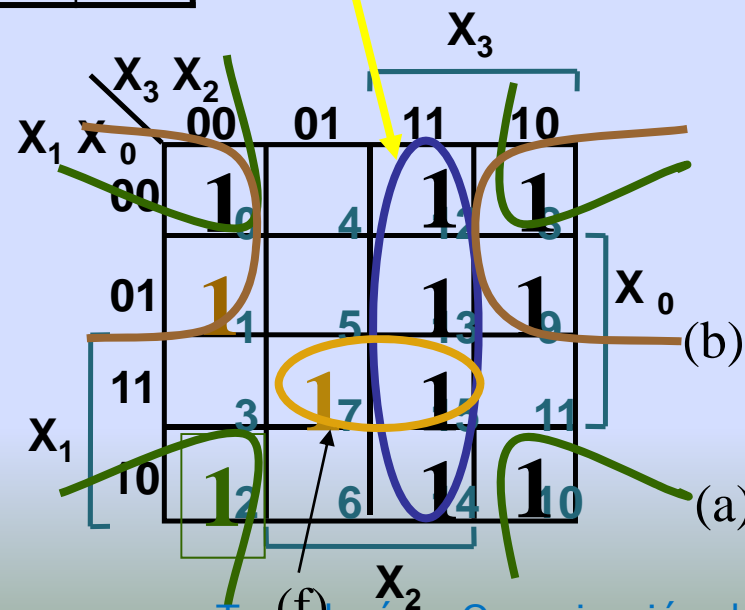
Paso 6) Tabla de implicants primos reducida, aplicar regla de dominancia y seleccionar los implicants primos esenciales secundarios.

(d)

d domina a c y e. Por tanto se eliminan.

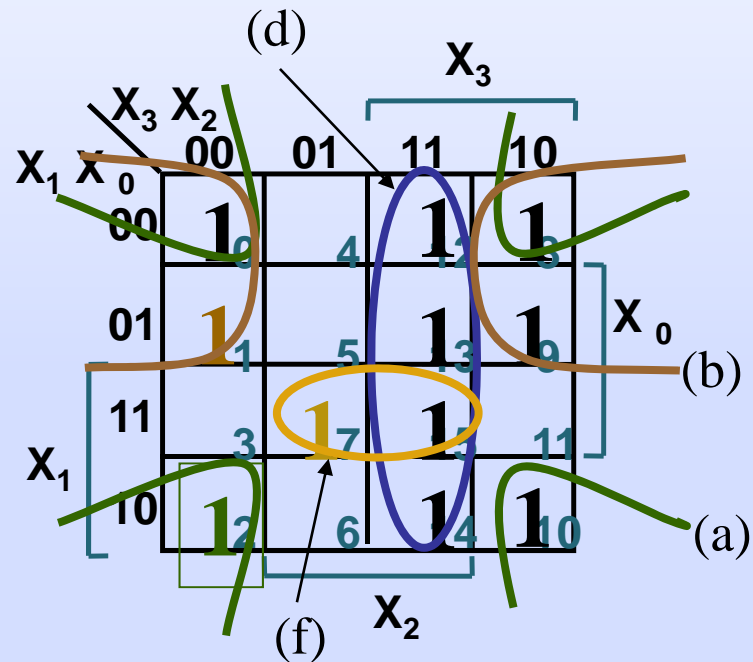
Después de este paso, d se comporta como implicants primo esencial y se denomina

esencial secundario



5.2 Método Riguroso. Ejemplo 2 de minimización.

Paso 7) Obtener un número mínimo de implicants primos opcionales para una total cobertura de la función.



En este ejemplo, hemos obtenido los implicants primos esenciales de la función: (a), (b) y (f). Respecto a los implicants primos opcionales, ha resultado que el impicante (d), en realidad es un impicante primo esencial secundario, tras aplicar la “regla de dominancia”

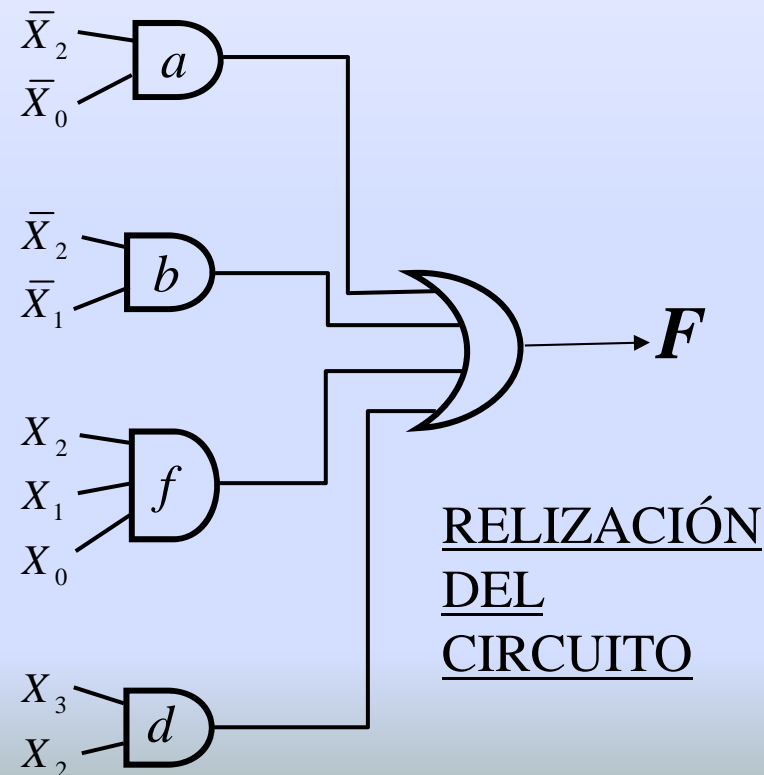
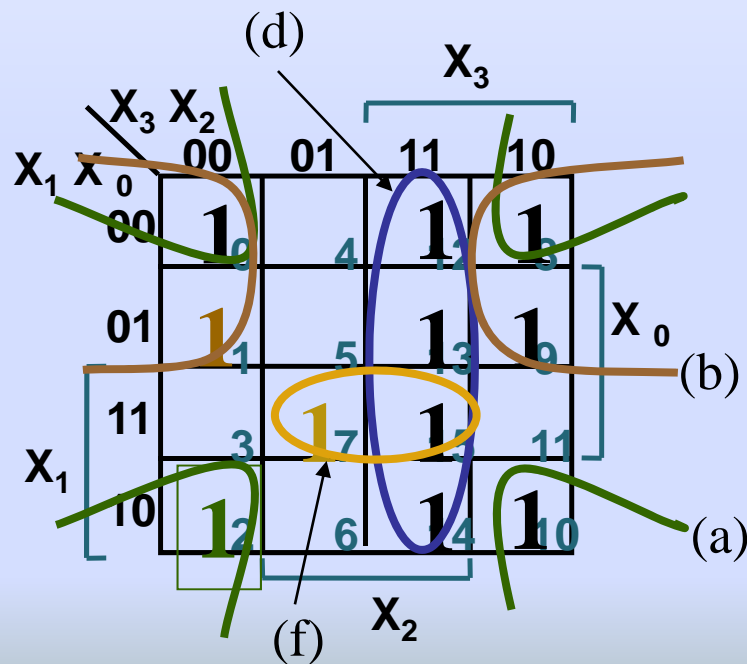
Observamos, que el conjunto de implicants primos { (a), (b), (f) y (d) } forma una cobertura mínima de F, no siendo necesario seleccionar más implicants .

$$F = a + b + f + d$$

5.2 Método Riguroso. Ejemplo 2 de minimización.

Paso 8) Obtener las expresiones de los implicantes primos seleccionados, expresar F algebraicamente con dichos implicantes y realizar el esquema del circuito con puertas lógicas.

$$F = a + b + f + d = \bar{X}_2\bar{X}_0 + \bar{X}_2\bar{X}_1 + X_2X_1X_0 + X_3X_2$$



6. Ejercicios.

Minimizar las funciones y obtener el esquema de circuito para las estructuras: AND/OR; NAND/NAND; OR/AND y NOR/NOR de las siguientes funciones:

a) $f(A,B,C) = \sum m(0,1,2,4,5) + d(3,7)$

b) $f(X,Y,Z) = \prod M(1,2,4,) + d(7)$

c) $f(X,Y,Z,V) = \sum m(0,2,3,4,6,7,8,12,14,15) + d(10,11,13)$

d) $f(A,B,C,D) = \prod M(2,3,4,5,10,11) + d(0,1)$

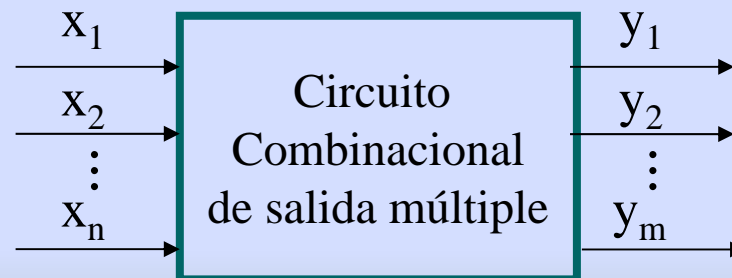
7. APÉNDICE

Minimización Multifuncional.

Minimización Multifuncional.

En general los circuitos combinacionales suelen consistir en un conjunto de entradas y un conjunto de salidas. Esto significa que han de realizar un conjunto de funciones de conmutación que dependen del mismo conjunto de entradas.

Si el conjunto de funciones se minimiza por separado (una a una) el resultado que se obtiene no asegura un coste mínimo global. En este Apéndice se explican las bases de los métodos multifuncional, utilizando los conocimientos adquiridos en los apartados anteriores de este seminario.



$$y_i = f_i(x_1, x_2, \dots, x_n)$$

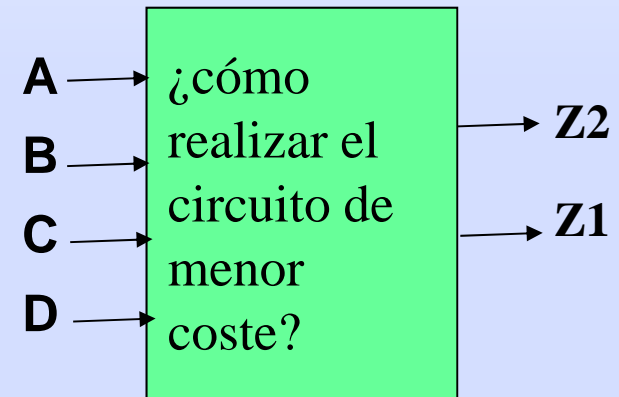
Minimización Multifuncional.

Un ejemplo.

Realizar un circuito que tiene dos salidas Z2 y Z1, cuyo comportamiento debe ser el indicado por la tabla siguiente.

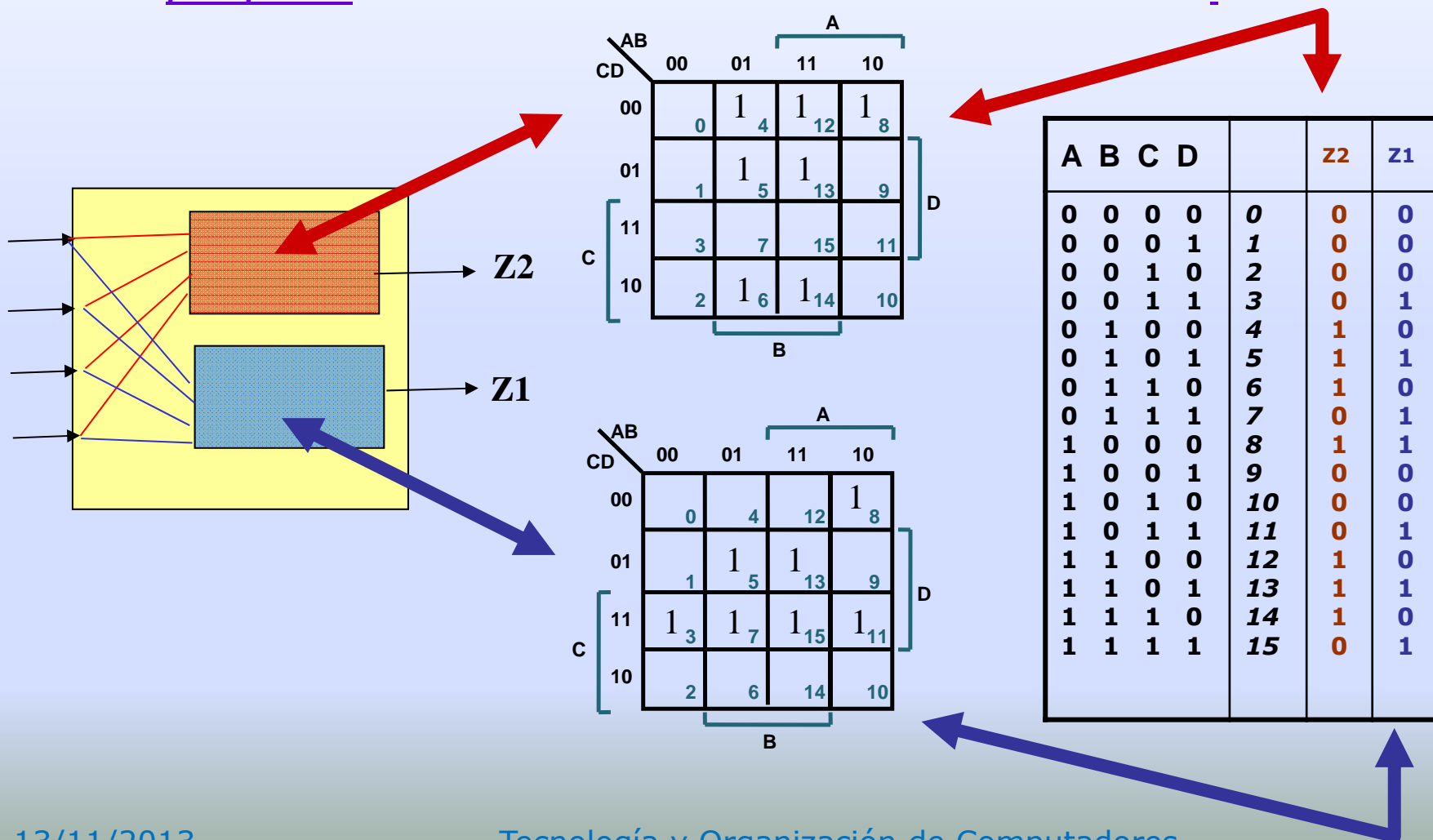
Los circuitos de salida múltiple son circuitos que realizan un conjunto de funciones que dependen del mismo conjunto de variables de entrada.

A	B	C	D		Z2	Z1
0	0	0	0	0	0	0
0	0	0	1	1	0	0
0	0	1	0	2	0	0
0	0	1	1	3	0	1
0	1	0	0	4	1	0
0	1	0	1	5	1	1
0	1	1	0	6	1	0
0	1	1	1	7	0	1
1	0	0	0	8	1	1
1	0	0	1	9	0	0
1	0	1	0	10	0	0
1	0	1	1	11	0	1
1	1	0	0	12	1	0
1	1	0	1	13	1	1
1	1	1	0	14	1	0
1	1	1	1	15	0	1

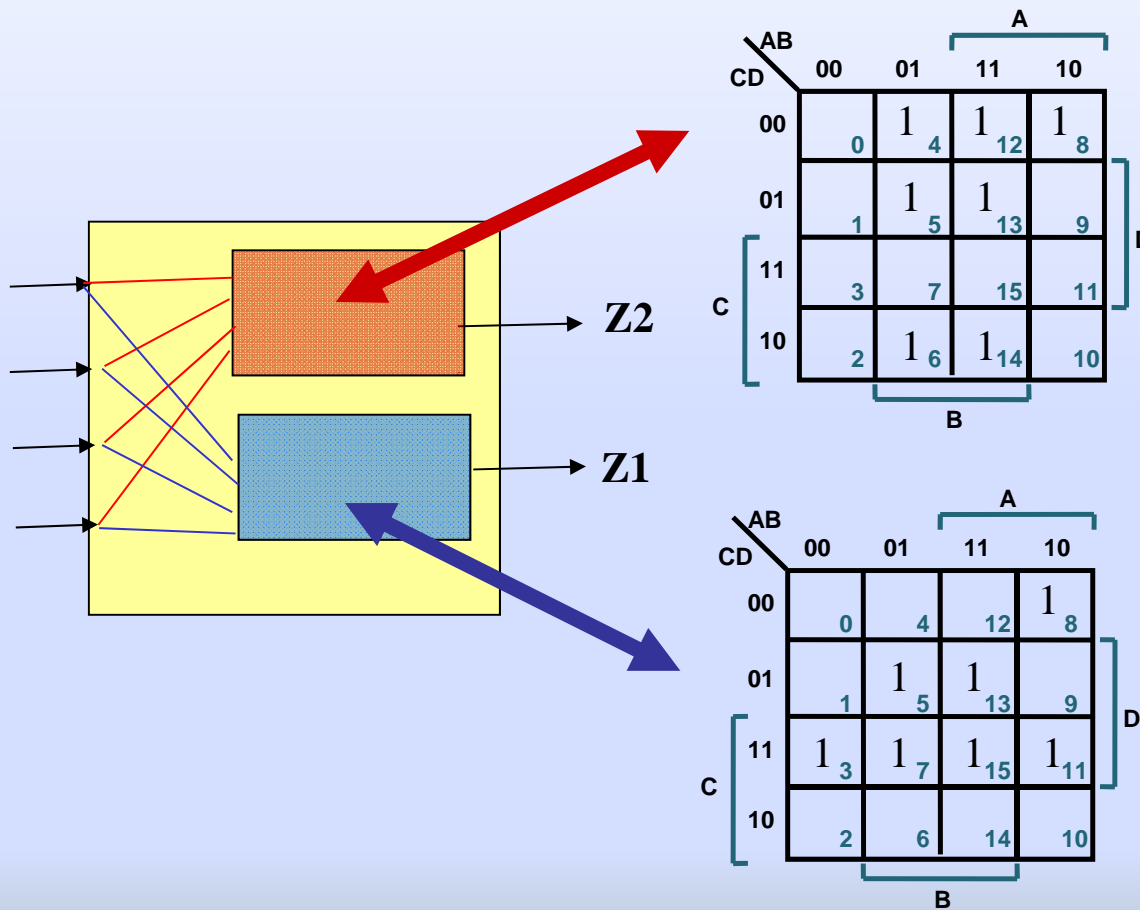


Minimización Multifuncional.

Una alternativa para realizar el circuito de salida múltiple es minimizar y realizar cada función de salida por separado. Pero ESTO NO CONDUCE A UN CIRCUITO de MÍNIMO COSTE.



Minimización Multifuncional.



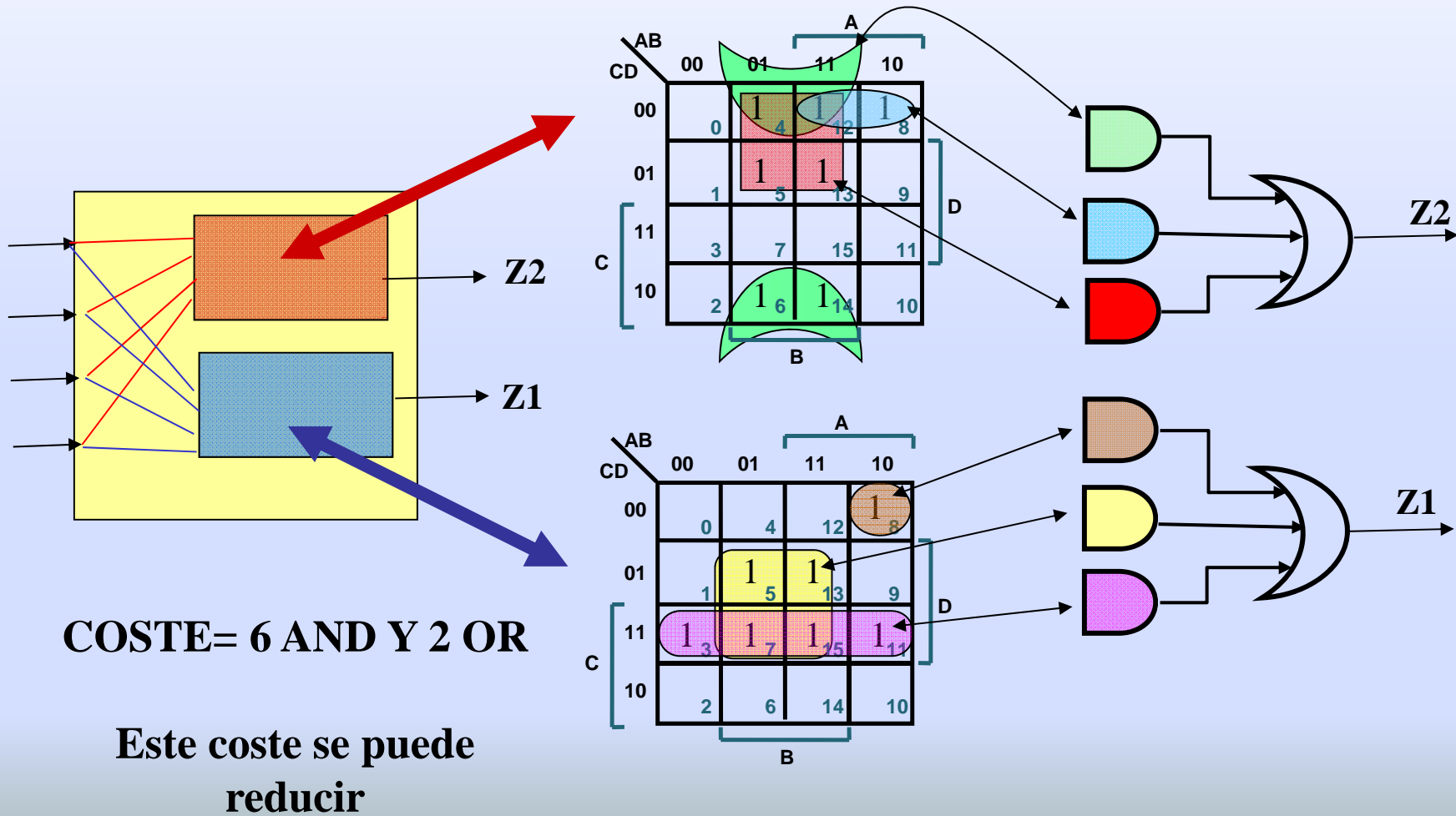
Obtener el coste del circuito según la minimización monofuncional (por separado) de cada salida

¿ Es esta solución de coste mínimo?

NO

¿Se podría encontrar una solución de menor coste?

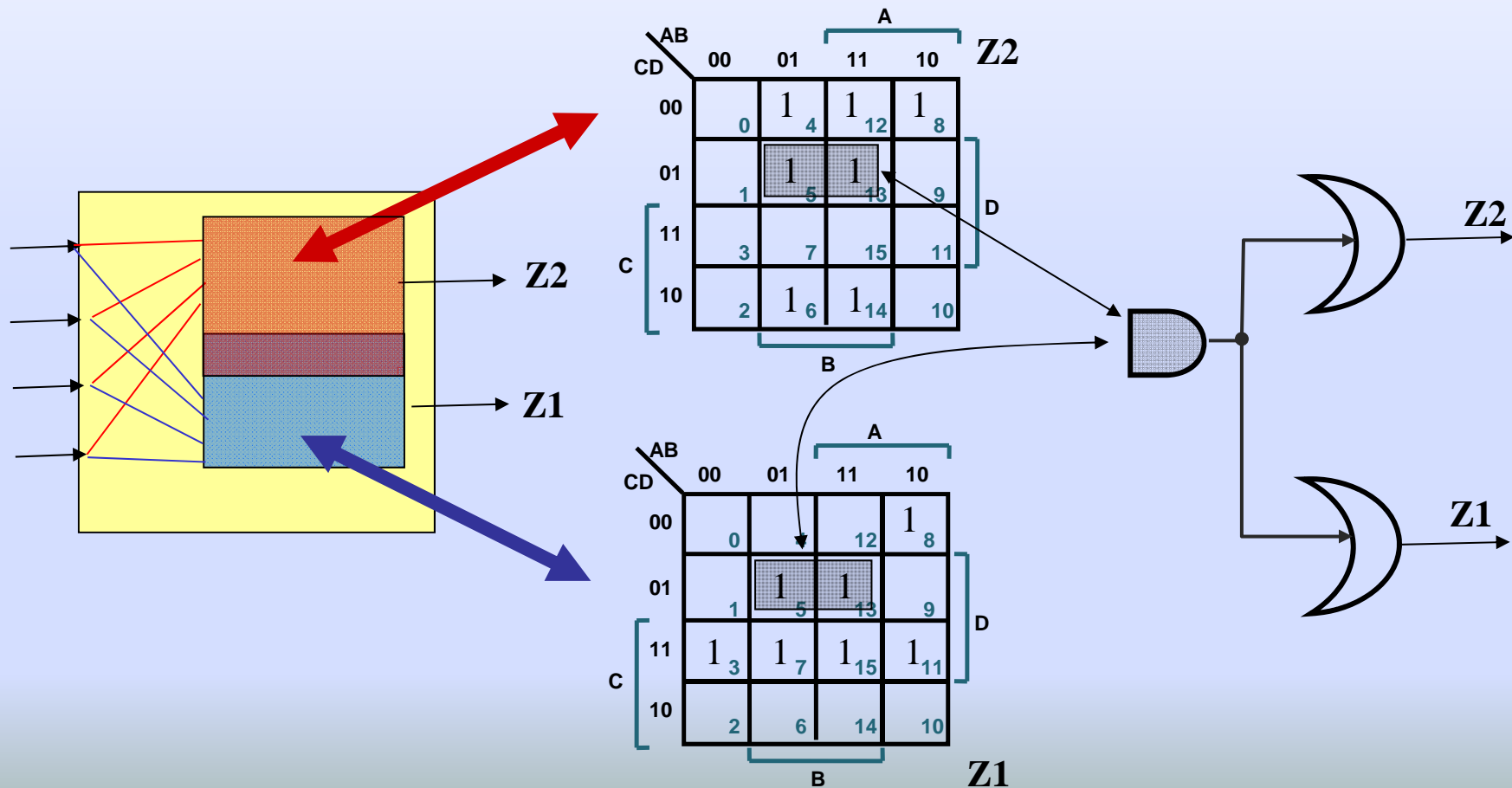
Minimización Multifuncional.



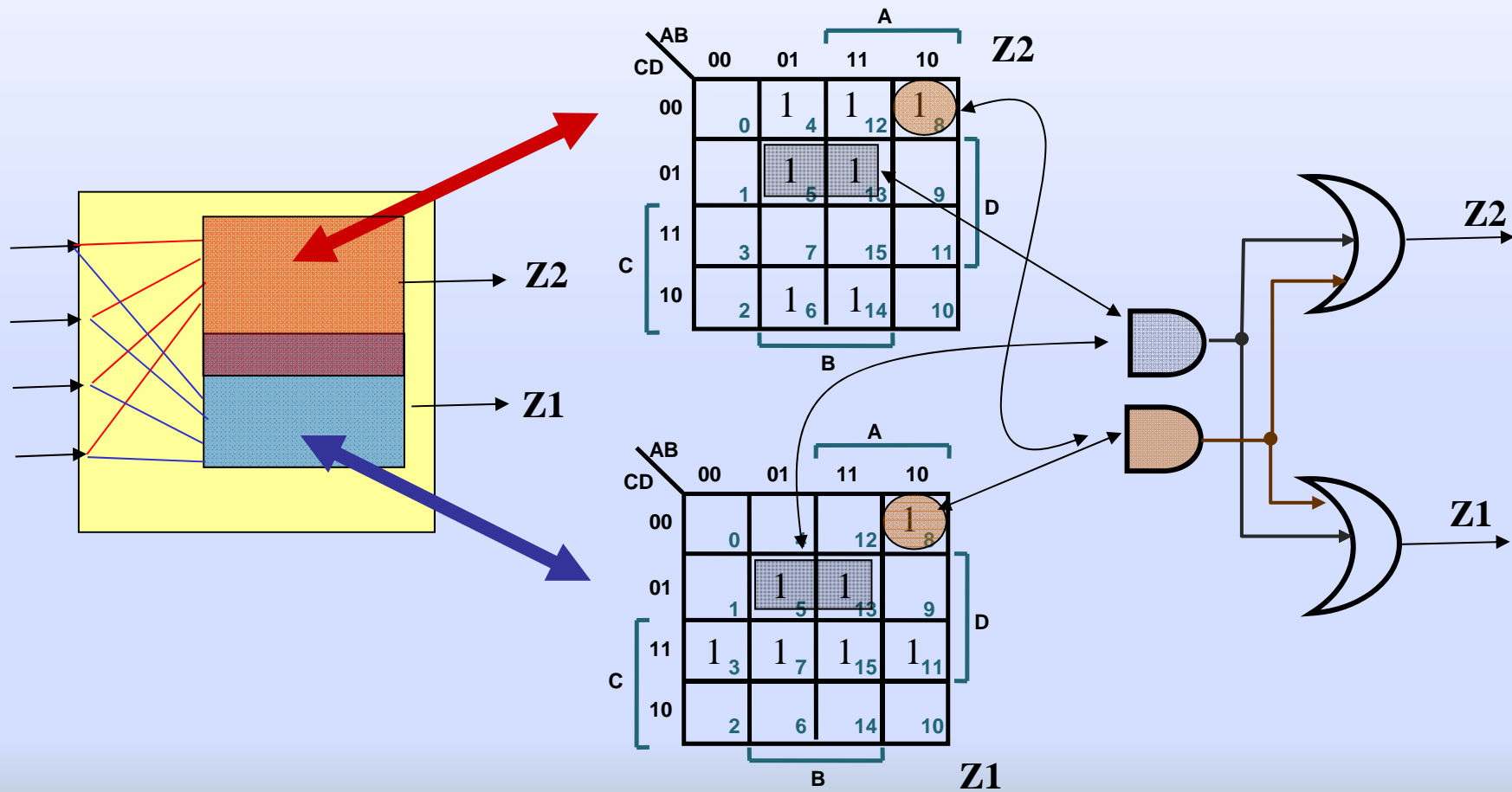
Minimización Multifuncional.

Existe una alternativa MÁS ADECUADA CON UN COSTE DEL CIRCUITO MENOR.

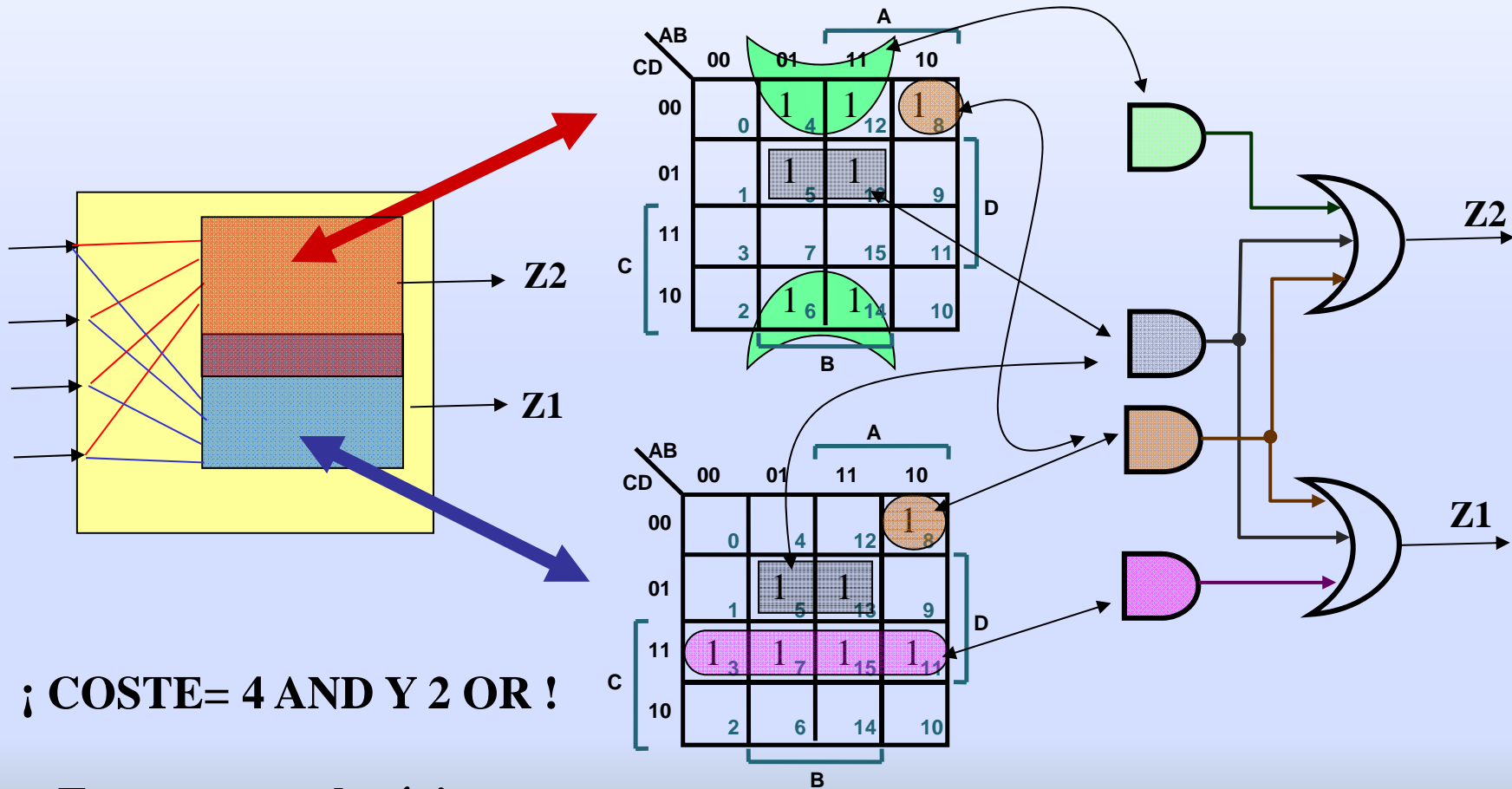
MINIMIZACIÓN MULTIFUNCIONAL.



Minimización Multifuncional.



Minimización Multifuncional.



¡ COSTE= 4 AND Y 2 OR !

Este coste es el mínimo.

Minimización Multifuncional.

Puntos a tener en cuenta en la minimización Multifuncional.

1. En el paso de "IDENTIFICACIÓN DE TODOS LOS IMPLICANTES (IMPLICADOS) PRIMOS" se tomarán también los implicantes (implicados) primos correspondientes a los mapas "producto entre funciones".

Ejemplo: Si el sistema consta de las funciones F_a , F_β y F_γ , entonces hay que realizar los mapas "producto" $F_a F_\beta F_\gamma$, $F_a F_\beta$, $F_a F_\gamma$, $F_\beta F_\gamma$, además de los mapas F_a , F_β y F_γ . De todos estos mapas se identifican los implicantes (implicados) primos sin repetirlos, comenzando por los mapas producto de mayor número de funciones.

2. En la TABLA DE IMPLICANTES (IMPLICADOS) PRIMOS:

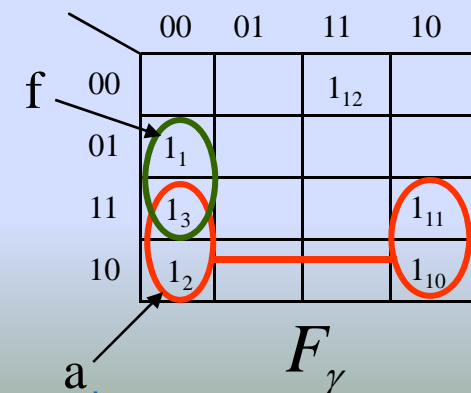
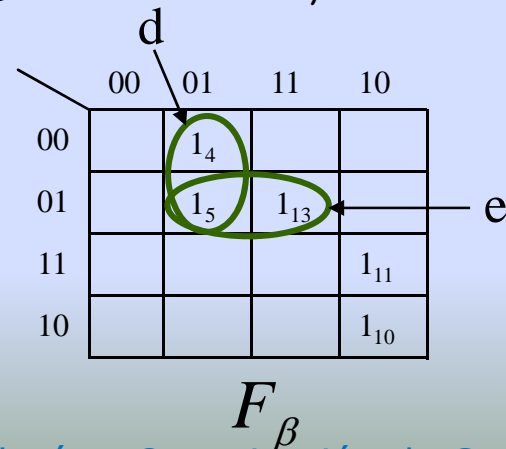
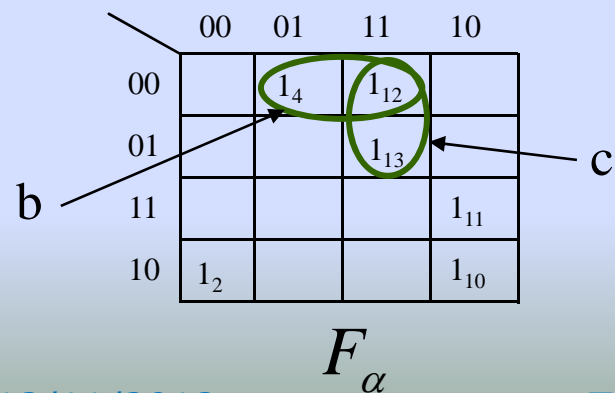
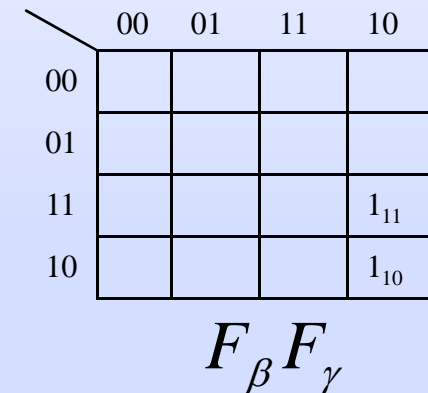
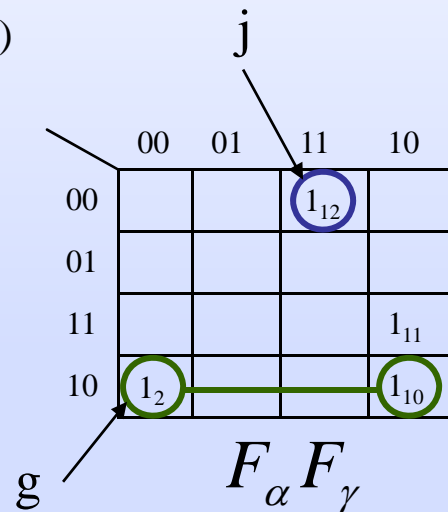
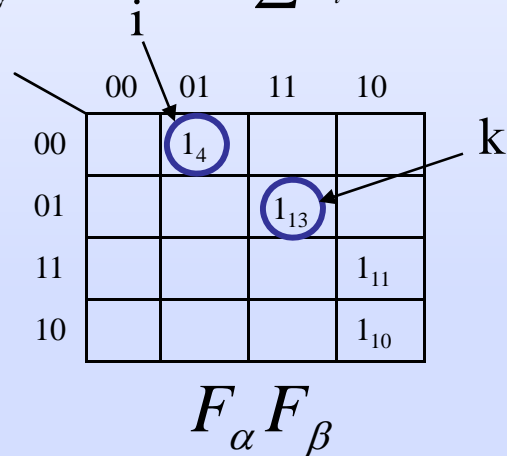
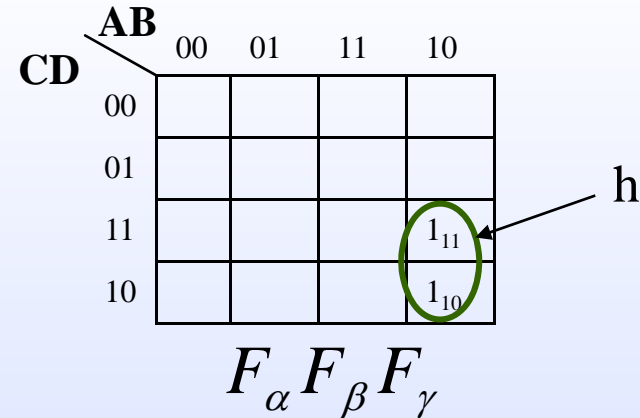
- a) Junto a cada implicante (implicado), se anota las funciones a las que pertenece. Por ejemplo, si es del mapa producto $F_a F_\beta$ se anotará $a\beta$.
- b) Se reservan columnas independientes para cada función, especificando en la primera fila los minterminos (maxterminos) a cubrir .
- c) Se marcan los minterminos (maxterminos) de cada implicante (implicado), únicamente en las columnas que correspondan a funciones a las que pertenece dicho implicante (implicado) (según punto (a)).

Minimización Multifuncional.

$$F_{\alpha}(A, B, C, D) = \sum m_i(2, 4, 10, 11, 12, 13)$$

$$F_{\beta}(A, B, C, D) = \sum m_i(4, 5, 10, 11, 13)$$

$$F_{\gamma}(A, B, C, D) = \sum m_i(1, 2, 3, 10, 11, 12)$$



Minimización Multifuncional.

TODOS los Implicantes Primos	F_α						F_β					F_γ					
	2	4	10	11	12	13	4	5	10	11	13	1	2	3	10	11	12
γ a													X	X	X	X	
α b		X			X												
α c					X	X											
β d							X	X									
β e								X			X						
* γ f												X		X			
* $\alpha\gamma$ g	X		X										X		X		
* $\alpha\beta\gamma$ h			X	X					X	X					X	X	
$\alpha\beta$ i		X					\bar{X}										
* $\alpha\gamma$ j					X												X
$\alpha\beta$ k						X					X						
	X		X	X	X				X	X		X	X	X	X	X	X
Se eligen para el sistema los I.P. Esenciales = [f, g, h, j]																	

Minimización Multifuncional.

Implicantes Primos	F_α		F_β		
	4	13	4	5	13
γ a					
α b	X				
$\rightarrow \alpha$ c		X			
β d			X	X	
$\rightarrow \beta$ e				X	X
$\rightarrow \alpha\beta$ i	X		X		
$\alpha\beta$ k		X			X
	i	c	i	e	e

Por inspección, o utilizando el método de Petrick, una posible solución para cubrir el Sistema es añadir al conjunto de implicantes seleccionados los siguientes:

$c \rightarrow (12,13)$ de F_α
 $e \rightarrow (5, 13)$ de F_β
 $i \rightarrow (4)$ de F_α y F_β

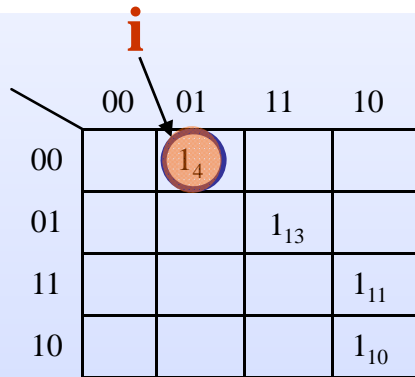
Conjunto de I.P. Seleccionados para el Sistema = $\{ [f, g, h, j] , [c, e, i] \}$

Esenciales

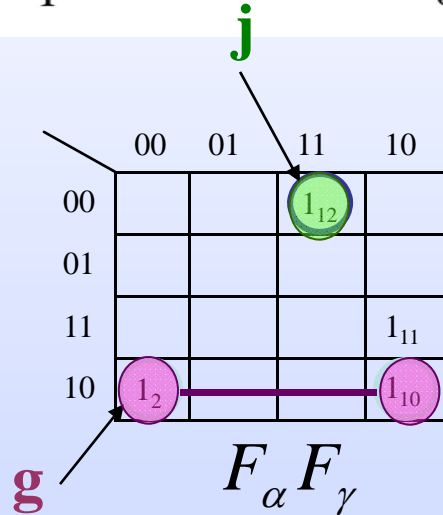
Opcionales

Minimización Multifuncional.

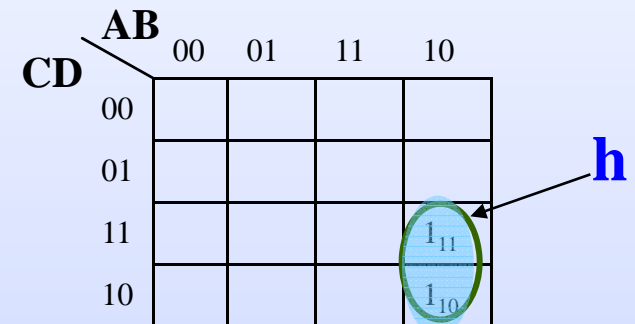
Conjunto de I.P. Seleccionados para el Sistema = $\{ \underline{[f, g, h, j]}, \underline{[c, e, i]} \}$



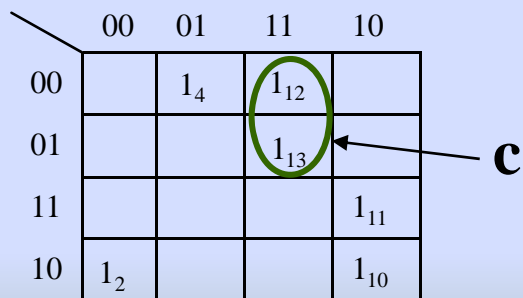
$F_\alpha F_\beta$



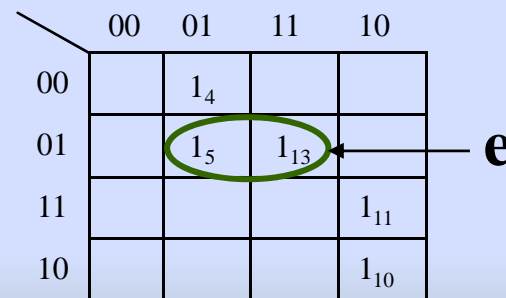
$F_\alpha F_\gamma$



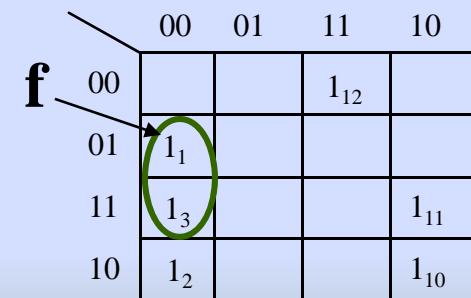
$F_\alpha F_\beta F_\gamma$



F_α



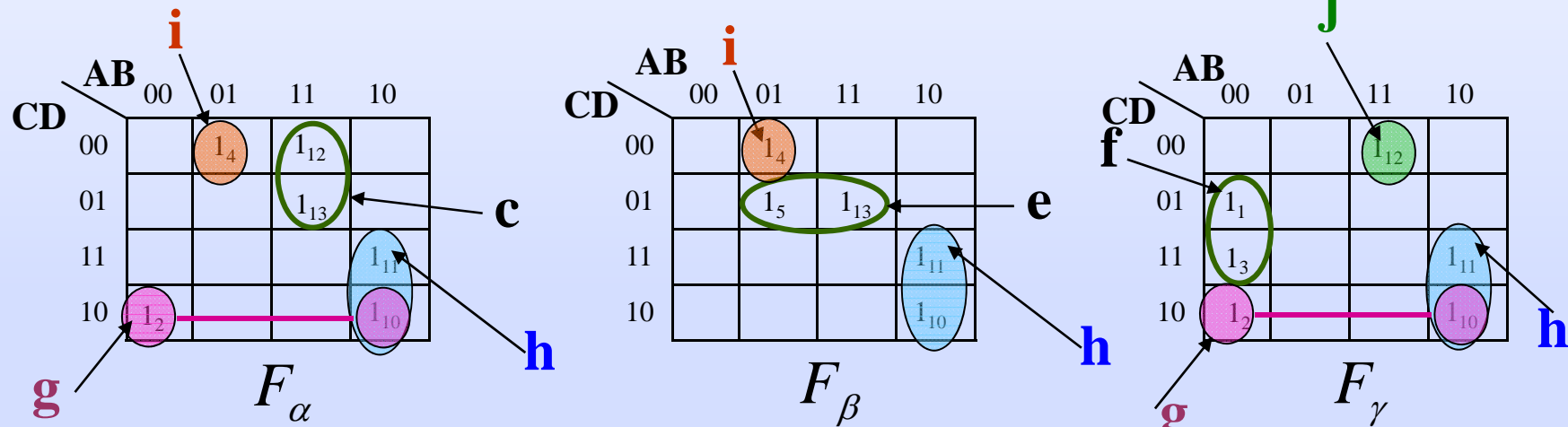
F_β



F_γ

Minimización Multifuncional.

OBTENCIÓN DE LAS FUNCIONES A PARTIR DEL CONJUNTO DE IMPLICANTES PRIMOS SELECCIONADOS PARA EL SISTEMA.



Por inspección vemos que una solución adecuada al sistema es:

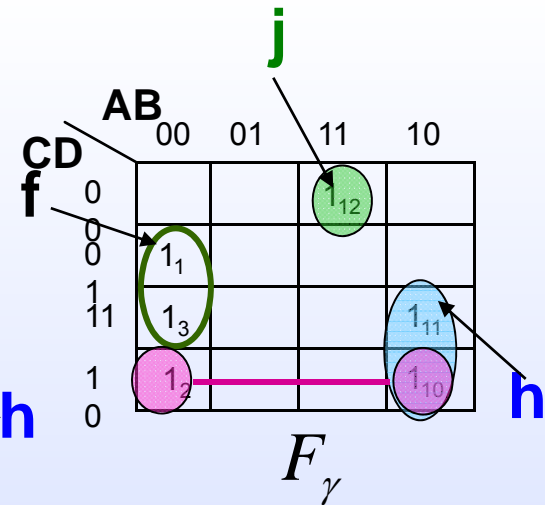
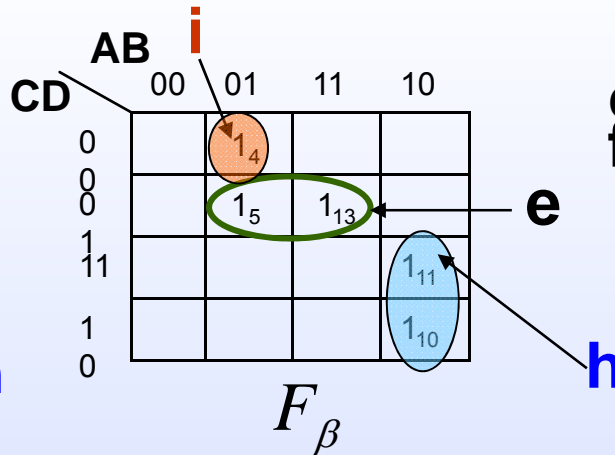
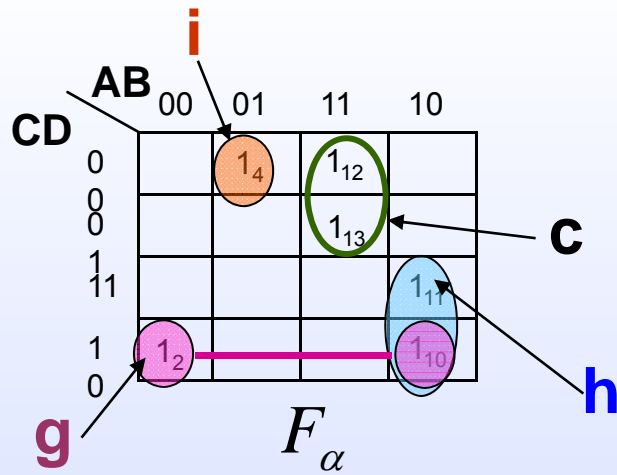
$$F_{\alpha} = c + g + h + i$$

$$F_{\beta} = e + h + i$$

$$F_{\gamma} = f + g + h + j$$

Ejercicio :

Dibuje el circuito que realiza a las funciones F_{α} , F_{β} y F_{γ}



$$g = \bar{B} \cdot C \cdot \bar{D}$$

$$c = A \cdot B \cdot \bar{C}$$

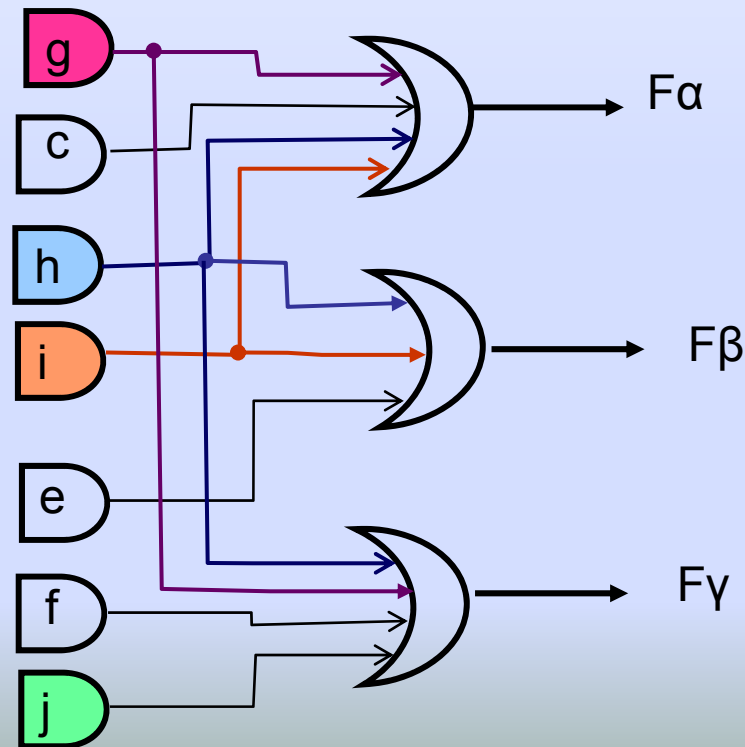
$$h = A \cdot \bar{B} \cdot C$$

$$i = \bar{A} \cdot B \cdot \bar{C} \cdot \bar{D}$$

$$e = B \cdot \bar{C} \cdot D$$

$$f = \bar{A} \cdot \bar{B} \cdot D$$

$$j = A \cdot B \cdot \bar{C} \cdot \bar{D}$$



$$F_{\alpha} = c + g + h + i$$

$$F_{\beta} = e + h + i$$

$$F_{\gamma} = f + g + h + j$$

Solución al ejercicio anterior

SEMINARIO 4 MINIMIZACIÓN DE FUNCIONES DE COMUTACIÓN.

***TECNOLOGÍA Y ORGANIZACIÓN DE
COMPUTADORES***

1º Grado en Ingeniería Informática.