

## Preguntas Más Frecuentes Tema 4

### Contenido

- P.4.1: ¿Como se resuelve el *ejercicio 2.3 (pag. 400)* de la Preparación de la Evaluación de este tema? ... 2
- P.4.2: ¿Cómo se obtienen las expresiones de  $S_i$ ,  $C_i$  y  $D_i$  del *ejercicio E.7.4 (pag.213)*? ..... 6
- P.4.3: P4.3: ¿Encuentro serias dificultades a la hora de comprender el *apartado 7.3: "Transistores de puerta flotante (FAMOS) y Mecanismos de Borrado"*. Se utilizan conceptos que desconozco. ¿Es necesario el estudio y comprensión de este apartado para continuar? ..... 6
- P.4.4: No entiendo la respuesta dada al *ejercicio E.7.5*. ¿Podrían explicarla de forma más detallada?..... 6
- P.4.5: ¿Podrían explicar de dónde salen las expresiones de A, B, C y D del *apartado C) del problema E.7.7 de la pag 224* del libro de problemas? ..... 7

**P.4.1:** ¿Como se resuelve el ejercicio 2.3 (pag. 400) de la Preparación de la Evaluación de este tema?.

**R.4.1:** Los componentes con los que se realiza el diseño condiciona el tipo de diseño a realizar. En este caso queremos diseñar con PLDs, por lo tanto, si tenemos en cuenta el tipo de circuito que vamos a usar, nos damos cuenta que debemos conseguir representar las expresiones del circuito a implementar en función de los términos mínimos de las variables de entrada y que, en este caso, no tiene sentido su minimización, sino todo lo contrario. Es decir, para diseñar con PLDs lo que tenemos que hacer es expandir las funciones en vez de minimizarlas. Si se realiza el diseño a partir de la tabla de verdad las expresiones obtenidas se usan directamente, sin minimizar.

El *ejercicio de autoevaluación 2.3* consiste en la síntesis de un sumador de acarreo adelantado para palabras de 4 bits y usando una PROM. Dado que el diseño para palabras de 4 bits es muy laborioso y que no se pierde el valor pedagógico perseguido con su resolución, lo vamos a resolver para palabras de 2 bits.

Los pasos a seguir para realizar el diseño son:

1. En este caso concreto podemos partir de las expresiones 5.9 y 5.10 de la pag.277 del texto que son las expresiones recursivas que tenemos que ir desarrollando para los distintos bits de las palabras binarias.

$$P_i = A_i \oplus B_i, \quad G_i = A_i B_i$$

$$S_i = P_i \oplus C_i, \quad C_{i+1} = G_i + P_i C_i$$

2. Particularizamos estas expresiones para los dos bits de las palabras que queremos sumar y que son los datos del problema junto con el acarreo de entrada,  $C_0$ . Es decir, debemos poner las expresiones en función de los bits de las palabras  $A(A_1 A_0)$  y  $B(B_1 B_0)$  y del acarreo de entrada,  $C_0$ , siendo  $A_1$  y  $B_1$  los bits más significativos. Así,

$$P_0 = A_0 \oplus B_0, \quad G_0 = A_0 B_0 \qquad P_1 = A_1 \oplus B_1, \quad G_1 = A_1 B_1$$

$$S_0 = P_0 \oplus C_0, \quad C_1 = G_0 + P_0 C_0 \quad S_1 = P_1 \oplus C_1, \quad C_2 = G_1 + P_1 C_1$$

3. Sustituimos los valores de las  $P_i$  y  $G_i$  por sus expresiones en función de los bits de las palabras:

$$S_0 = A_0 \oplus B_0 \oplus C_0, \quad C_1 = A_0 B_0 + (A_0 \oplus B_0)C_0$$

$$S_1 = A_1 \oplus B_1 \oplus C_1 = A_1 \oplus B_1 \oplus [A_0 B_0 + (A_0 \oplus B_0)C_0],$$

$$C_2 = A_1 B_1 + (A_1 \oplus B_1)[A_0 B_0 + (A_0 \oplus B_0)C_0]$$

4. Operando resulta:

$$\begin{aligned}
 S_0 &= (\bar{A}_0 B_0 + A_0 \bar{B}_0) \oplus C_0 = \overline{(\bar{A}_0 B_0 + A_0 \bar{B}_0)} C_0 + (\bar{A}_0 B_0 + A_0 \bar{B}_0) \bar{C}_0 \\
 &= (\bar{A}_0 \bar{B}_0 + A_0 B_0) C_0 + (\bar{A}_0 B_0 + A_0 \bar{B}_0) \bar{C}_0 = \bar{A}_0 \bar{B}_0 C_0 + A_0 B_0 C_0 + \bar{A}_0 B_0 \bar{C}_0 + A_0 \bar{B}_0 \bar{C}_0 \\
 C_1 &= A_0 B_0 + (A_0 \oplus B_0) C_0 = A_0 B_0 + (\bar{A}_0 B_0 + A_0 \bar{B}_0) C_0 = A_0 B_0 + \bar{A}_0 B_0 C_0 + A_0 \bar{B}_0 C_0
 \end{aligned}$$

Análogamente,

$$S_1 = (\bar{A}_1 B_1 + A_1 \bar{B}_1) \oplus C_1 = \bar{A}_1 \bar{B}_1 C_1 + A_1 B_1 C_1 + \bar{A}_1 B_1 \bar{C}_1 + A_1 \bar{B}_1 \bar{C}_1$$

pero,  $C_1 = A_0 B_0 + \bar{A}_0 B_0 C_0 + A_0 \bar{B}_0 C_0 = A_0 B_0 + B_0 C_0 + A_0 C_0$

y  $\bar{C}_1 = \overline{A_0 B_0 + \bar{A}_0 B_0 C_0 + A_0 \bar{B}_0 C_0} = \bar{A}_0 \bar{B}_0 + \bar{B}_0 \bar{C}_0 + \bar{A}_0 \bar{C}_0$

por tanto, operando obtenemos:

$$\begin{aligned}
 S_1 &= (\bar{A}_1 \bar{B}_1 + A_1 B_1) C_1 + (\bar{A}_1 B_1 + A_1 \bar{B}_1) \bar{C}_1 = \\
 &= (\bar{A}_1 \bar{B}_1 + A_1 B_1) (A_0 B_0 + B_0 C_0 + A_0 C_0) + (\bar{A}_1 B_1 + A_1 \bar{B}_1) (\bar{A}_0 \bar{B}_0 + \bar{B}_0 \bar{C}_0 + \bar{A}_0 \bar{C}_0) = \\
 &= \bar{A}_1 \bar{B}_1 A_0 B_0 + A_1 B_1 A_0 B_0 + \bar{A}_1 \bar{B}_1 B_0 C_0 + A_1 B_1 B_0 C_0 + \bar{A}_1 \bar{B}_1 A_0 C_0 + A_1 B_1 A_0 C_0 + \\
 &\quad + \bar{A}_1 B_1 \bar{A}_0 \bar{B}_0 + A_1 \bar{B}_1 \bar{A}_0 \bar{B}_0 + \bar{A}_1 B_1 \bar{B}_0 \bar{C}_0 + A_1 \bar{B}_1 \bar{B}_0 \bar{C}_0 + \bar{A}_1 B_1 \bar{A}_0 \bar{C}_0 + A_1 \bar{B}_1 \bar{A}_0 \bar{C}_0
 \end{aligned}$$

De igual forma calcularíamos  $C_2$

5. Para implementar estas expresiones mediante una PROM tenemos que ponerlas en función de los términos mínimos de las señales de entrada o datos del problema,  $A_1$ ,  $B_1$ ,  $A_0$ ,  $B_0$  y  $C_0$ .

Si observamos estas expresiones vemos que  $S_0$  y  $C_1$  son funciones sólo de  $A_0$ ,  $B_0$  y  $C_0$ , y que las expresiones de  $S_1$  y  $C_2$  son funciones de  $A_0$ ,  $B_0$ ,  $C_0$ ,  $A_1$ , y  $B_1$ , pero los productos que forman parte de ellas no son los términos mínimos de las 5 variables de entrada. Por tanto, el siguiente paso es expandirlas, o sea, expresarlas en función de los términos mínimos de las 5 variables de entrada. La forma de conseguir esto es ver la/s variable/s que falta/n en cada uno de los sumandos de cada una de las expresiones y multiplicarlos por la suma de esta/s variable/s y su/s complementada/s. Así, como en la expresión de  $S_0$  faltan las variables  $A_1$  y  $B_1$  en todos los sumandos, deberemos multiplicarla por  $(A_1 + \bar{A}_1)(B_1 + \bar{B}_1)$ . En realidad, lo que estamos haciendo es dejarla como está, puesto que la estamos multiplicando por la unidad, pero la estamos cambiando de representación. Justo, estamos haciendo lo contrario de lo que hacemos al minimizar. Así,

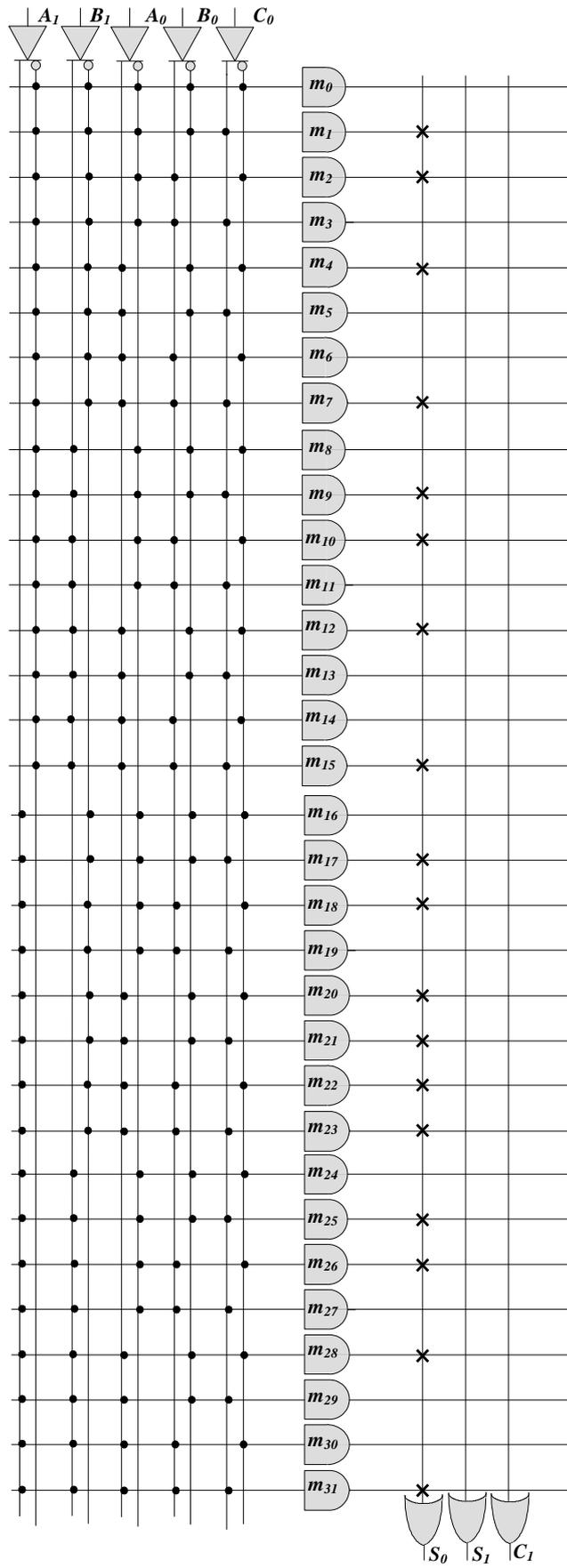
$$\begin{aligned}
 S_0 &= \bar{A}_0 \bar{B}_0 C_0 + A_0 B_0 C_0 + \bar{A}_0 B_0 \bar{C}_0 + A_0 \bar{B}_0 \bar{C}_0 = \\
 &= (\bar{A}_0 \bar{B}_0 C_0 + A_0 B_0 C_0 + \bar{A}_0 B_0 \bar{C}_0 + A_0 \bar{B}_0 \bar{C}_0) (A_1 + \bar{A}_1) (B_1 + \bar{B}_1)
 \end{aligned}$$

Si operamos, la expresamos en función de los términos mínimos de forma simplificada, o sea en función de  $m_i$ , siendo  $i$  el equivalente en decimal del término mínimo (que puede tomar  $2^5-1$  valores, desde 0 hasta 31) y tomamos para el orden de los bits de las entradas el mismo que después consideraremos en las entradas de la PROM,  $A_1 B_1 A_0 B_0 C_0$ , siendo  $A_1$  el bit más significativo (MSB) y  $C_0$  el menos significativo (LSB) obtenemos:

$$S_0 = m_1 + m_2 + m_4 + m_7 + m_9 + m_{10} + m_{12} + m_{15} + m_{17} + m_{18} + m_{20} + m_{23} + m_{25} + m_{26} + m_{28} + m_{31}$$

De igual forma operaríamos con las expresiones  $S_1$  y  $C_2$ .

Una vez obtenidas las 3 expresiones,  $S_0$ ,  $S_1$  y  $C_2$  las implementamos en la PROM. Así el circuito resultante para  $S_0$  es:



**P.4.2: ¿Cómo se obtienen las expresiones de  $S_i$ ,  $C_i$  y  $D_i$  del ejercicio E.7.4 (pag.213)?**

**R.4.2:** Las expresiones  $S_i$  y  $C_i$  son las de la salida de un sumador completo expresada en función de los términos mínimos. Es decir, son las expresiones de  $S_i$  y  $C_{i+1}$  de las ecuaciones 5.7 y 5.8. de la pag.272 del texto base, sólo que al acarreo de entrada lo considera como  $C_{i-1}$  y el de salida como  $C_i$ , pero sólo es un cambio de nomenclatura, lo importante es que el de salida es uno más que el de entrada.:

En el texto base las expresiones son:  $S_i = C_i \oplus A_i \oplus B_i$  y  $C_{i+1} = A_i B_i + C_i(A_i \oplus B_i)$

y en el problema hemos considerado:  $S_i = C_{i-1} \oplus A_i \oplus B_i$  y  $C_i = A_i B_i + C_{i-1}(A_i \oplus B_i)$

Si desarrollamos la expresión de  $S_i$  obtenemos:

$$\begin{aligned} S_i &= C_{i-1} \oplus A_i \oplus B_i = \overline{C_{i-1}}(A_i \oplus B_i) + C_{i-1} \overline{A_i \oplus B_i} = \overline{C_{i-1}}(\overline{A_i B_i} + A_i \overline{B_i}) + C_{i-1}(A_i B_i + \overline{A_i} \overline{B_i}) = \\ &= \overline{C_{i-1}} \overline{A_i} B_i + \overline{C_{i-1}} A_i \overline{B_i} + C_{i-1} A_i B_i + C_{i-1} \overline{A_i} \overline{B_i} \end{aligned}$$

De igual forma operamos en  $C_i$ .

La expresión de  $D_i$  procede del hecho de querer hacer un sumador/restador y complementar  $B$  mediante el uso de la función XOR, como se explica en el problema E.5.3.



**P.4.3: P4.3: ¿Encuentro serias dificultades a la hora de comprender el apartado 7.3: "Transistores de puerta flotante (FAMOS) y Mecanismos de Borrado". Se utilizan conceptos que desconozco. ¿Es necesario el estudio y comprensión de este apartado para continuar?**

**R.4.3:** Este apartado es a título informativo y por completitud del texto. No es necesario que lo estudiéis y lo comprendáis, pero sí que debéis saber que existen diferentes tipos y que su borrado es diferente. No es objeto de examen y no es necesario para entender el resto del tema.



**P.4.4: No entiendo la respuesta dada al ejercicio E.7.5. ¿Podrían explicarla de forma más detallada?**

**R.4.4:** En este ejercicio tenemos que sintetizar una mini-ALU para que opere sobre 2 bits y que ejecute, bajo las señales de control  $S_1$ ,  $S_0$  y  $M$ , las operaciones lógicas y aritméticas que se describen en la tabla de la figura 7.5.1.

Empecemos por sintetizar la primera función que, como podemos observar en la tabla, es la operación  $F=AB$ . Según la tabla, esta función la ejecuta la ALU siempre que  $S_1=0$ ,  $S_0=0$  y  $M=1$ . Al ser una operación lógica la realiza bit a bit, por tanto, las expresiones de los dos bits de salida de la ALU deberán ser función de la función lógica a realizar ( $AB$ ), del término mínimo que la controla ( $S_1=0$ ,  $S_0=0$ ) y de la variable que define que la operación a realizar es una función lógica ( $M$ ). Es decir:

$F(S_1 S_0 M) = F(0 0 1) =$  (Función lógica a realizar) (Término mínimo de control) (Variable que define que la operación a realizar es una función lógica)  $= (A B) (S_1=0, S_0=0) (M=1)$ .  
Así, para el bit  $F_0$  resulta:

$$F_0(S_1 S_0 M) = F_0(0 0 1) = (A_0 B_0) (S_1=0, S_0=0) (M=1)$$

y para el  $F_1$ :  $F_1(S_1 S_0 M) = F_1(0 0 1) = (A_1 B_1) (S_1=0, S_0=0) (M=1)$

Por tanto,  $F_0(001) = A_0 B_0 \bar{S}_1 \bar{S}_0 M$  (línea +7, pag.216)

y  $F_1(001) = A_1 B_1 \bar{S}_1 \bar{S}_0 M$  (línea -4, pag.216)

Si pasamos a la siguiente función tenemos:

Ahora, la ALU debe realizar la operación  $F=A+B$  siempre que  $S_1=0, S_0=1$  y  $M=1$ . Por tanto, de nuevo:

$F(S_1 S_0 M) = F(0 1 1) =$  (Función lógica a realizar) (Término mínimo de control) (Variable que define que la operación a realizar es una función lógica)  $= (A+B) (S_1=0, S_0=1) (M=1)$

Es decir:  $F_0(S_1 S_0 M) = F_0(0 1 1) = (A_0 + B_0) (S_1=0, S_0=1) (M=1)$

y  $F_1(S_1 S_0 M) = F_1(0 1 1) = (A_1 + B_1) (S_1=0, S_0=1) (M=1)$

Por tanto,  $F_0 = (A_0 + B_0) \bar{S}_1 S_0 M$  (línea -10, pag.216)

$F_1 = (A_1 + B_1) \bar{S}_1 S_0 M$  (línea -3, pag.216)

De igual forma se obtienen el resto de las expresiones.



**P.4.5:** ¿Podrían explicar de dónde salen las expresiones de A, B, C y D del apartado C) del problema E.7.7 de la pag 224 del libro de problemas?.

**R.4.5:** Estas expresiones proceden del apartado D del problema 6.4, pero al transcribirlas se han cometido errores.

Como podemos comprobar las expresiones correctas son:

$$A = d_8 + d_9$$

$$B = d_4 + d_5 + d_6 + d_7$$

$$C = d_2 + d_3 + d_6 + d_7$$

$$D = d_1 + d_3 + d_5 + d_7 + d_9$$

Sin embargo, la figura 7.7.3 de la página 226 es correcta. Obsérvese que responde a estas ecuaciones.

Otra forma de obtener estas expresiones es a partir de las del apartado B del mismo problema 7.7 y que figuran en la parte superior de la misma pag. 224. Obsérvese que A aparece sin negar en las expresiones de  $d_8$  y  $d_9$ . Por tanto, podemos poner que  $A=d_8+d_9$ .

Análogamente, B aparece sin negar en las expresiones de  $d_4, d_5, d_6$  y  $d_7$ . Por tanto, podemos poner que  $B=d_4+d_5+d_6+d_7$ . Lo mismo para el resto de las expresiones.

