75.562 · Fundamentos de Computadores · 2023-24

PEC2 - Segunda prueba de evaluación continua

Apellidos: *Gámez García*
Nombre: *Elías*

Formato y fecha de entrega

- Para dudas y aclaraciones sobre el enunciado debéis dirigiros al consultor responsable de vuestra aula.
- Hay que entregar la solución en un fichero PDF utilizando una de las plantillas entregadas conjuntamente con este enunciado.
- Se debe entregar a través de la aplicación de **Entrega y registro de EC** del apartado Evaluación de vuestra aula.
- La fecha límite de entrega es el **8 de noviembre** (a las 24 horas).
- **Razonad la respuesta en todos los ejercicios. Las respuestas sin justificación no recibirán puntuación.**

Respuestas

Ejercicio 1:

a) Expresad la función f como suma de mintérminos:

Los minterminos de una función son todas las combinaciones para las que la función toma el valor de 1, por lo que observamos la tabla y anotamos aquellos mintérminos:

M	a	b	c	d	F
0	1	0	0	0	0
1	1	0	0	1	0
2	1	0	1	0	0
3	1	0	1	1	0
4	1	1	0	0	0
5	1	1	0	1	0
6	1	1	1	0	1
7	1	1	1	1	1
8	0	0	0	0	0
9	0	0	0	1	0
10	0	0	1	0	0
11	0	0	1	1	0
12	0	1	0	0	0
13	0	1	0	1	0
14	0	1	1	0	1
15	0	1	1	1	1

Los mintérminos son, por lo tanto:

$$\boldsymbol{M_6, M_7, M_{14}, M_{15}}$$

Finalmente, escribimos la función como suma de los mintérminos, negando aquellas variables que están negadas, y manteniendo aquellas que son verdaderas:

$$f = \overline{a}bc\overline{d} + \overline{a}bcd + abc\overline{d} + abcd$$

b) Minimizad la expresión algebraica obtenida en el apartado anterior usando las leyes del álgebra de Boole que consideréis adecuadas, indicando en cada paso la que habéis utilizado.

1. Aplicamos la propiedad distributiva: $AB + AC = A(B + C)$

$$\overline{A}BC(\overline{D} + D) + ABC\overline{D} + ABCD$$

2. Aplicamos la ley del complemento: $A + \overline{A} = 1$

$$\overline{A}BC1 + ABC\overline{D} + ABCD$$

3. Aplicamos la ley distributiva:

$$BC(A\overline{D} + \overline{A}) + ABCD$$

4. Aplicamos la ley de absorción: $AB + \overline{A} = B + \overline{A}$

$$BC(\overline{D} + \overline{A}) + ABCD$$

5. Distribuimos el primer elemento en:

$$BC\overline{D} + BC\overline{A} + ABCD$$

6. Aplicamos la ley distributiva:

$$BC\overline{D} + BC\overline{A} + ABCD$$

7. Aplicamos la ley de absorción:

$$BC\overline{D} + BC\overline{A} + ABCD$$

8. Reordenamos:

$$BC\overline{D} + BC\overline{A} + ABCD$$

9. Aplicamos la ley distributiva:

$$BC(\overline{A} + A) + BC\overline{D}$$

10. Aplicamos la ley del complemento:

$$BC1 + BC\overline{D}$$

11. Aquí aplicamos la ley de la identidad: $A1 = A$

$$BC + BC\overline{D}$$

12. Y finalmente aplicamos la ley de absorción:

$$BC$$

Ejercicio 2:

a) Sintetizad e implementad con puertas lógicas de manera mínima a dos niveles la función f mediante el método de Karnaugh.

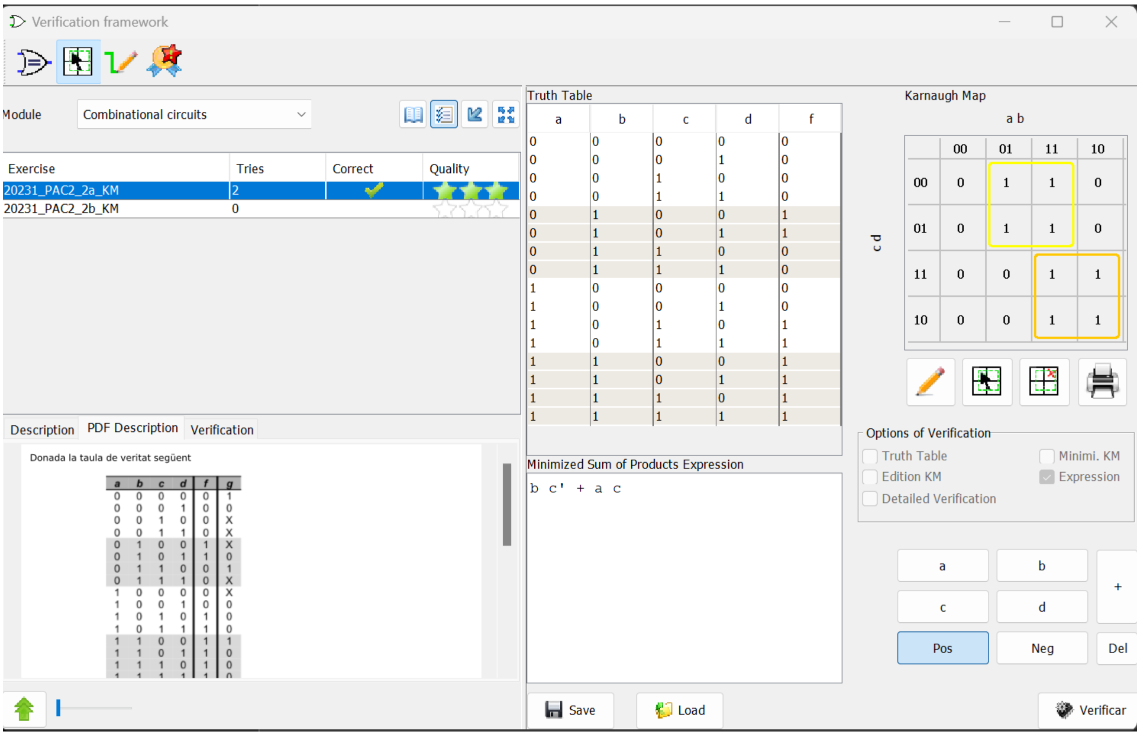
Primero de todo, analizamos la tabla de verdad, y situamos en el mapa de Karnaugh aquellas combinaciones de las variables ab y cd que den como resultado 1 en la función f. Aquellas que den como resultado 0, se indican como tal, y realizamos las agrupaciones más grandes que podamos, teniendo en cuenta que solamente podemos hacer agrupaciones en 2^N, siendo N entero:

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

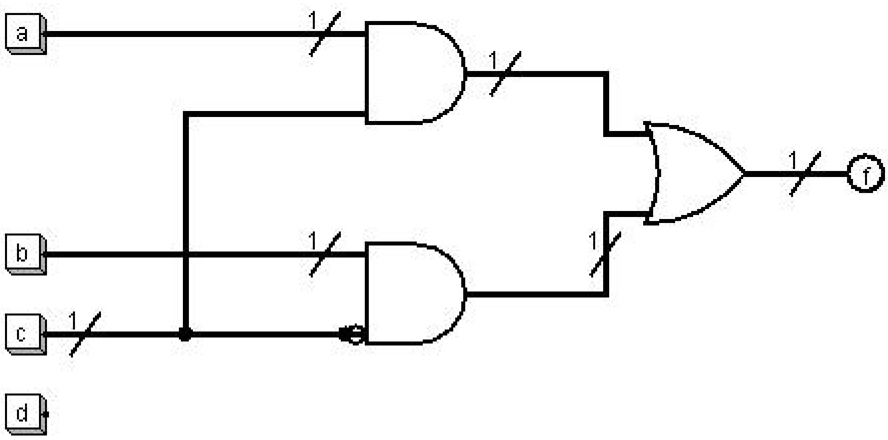
Vemos como podemos realizar dos agrupaciones de 4 términos, por lo que, a continuación, nos fijamos si las variables cambian o no en los grupos que hemos marcado. Por tanto, si una variable cambia de 0 a 1, o de 1 a 0, no la escribiremos en el resultado. En cambio, si la variable no cambia de valor, si la escribiremos. Si estas variables resultan en un 0, se negarán, y si se tratan de 1, no la negaremos.

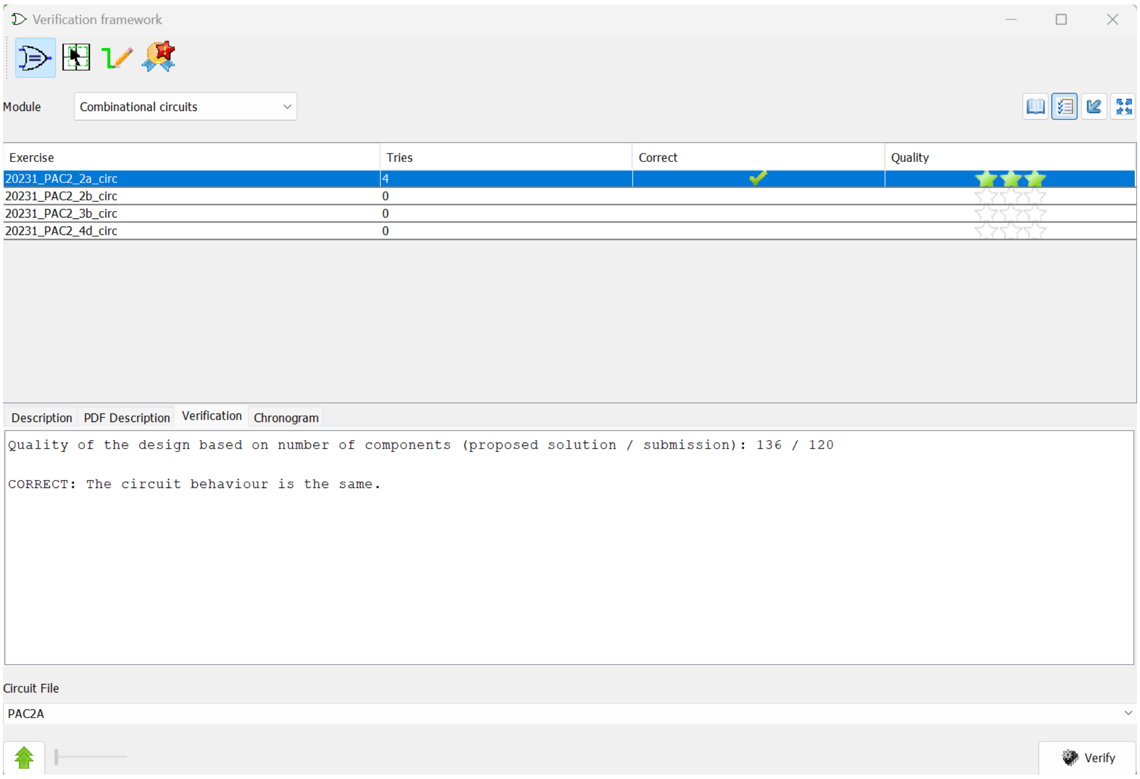
La expresión, por tanto, queda de la siguiente forma:

$$f = b\overline{c} + ac$$



Que resulta en el siguiente circuito:





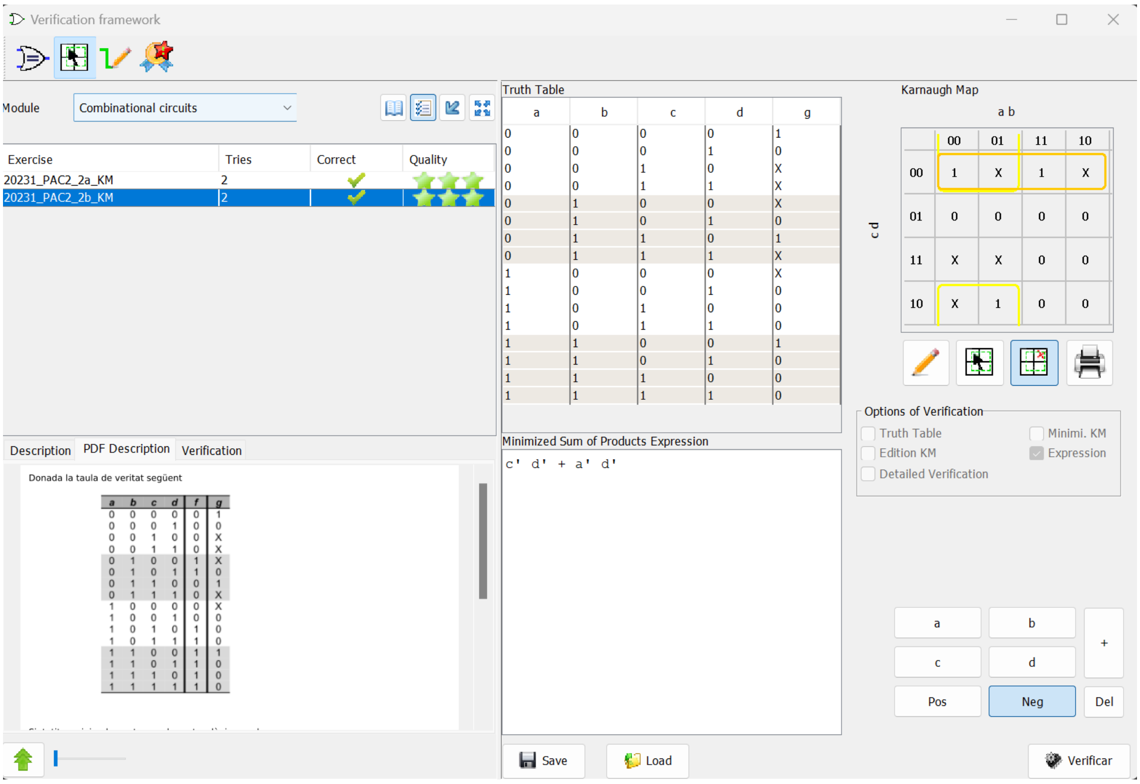
b) Sintetizad e implementad con puertas lógicas de manera mínima a dos niveles la función g mediante el método de Karnaugh.

Procedemos a dibujar el mapa de Karnaugh para la función g, y teniendo en cuenta que los valores indeterminados de X los contamos como 1 para optimizar la función en el caso de valer 1:

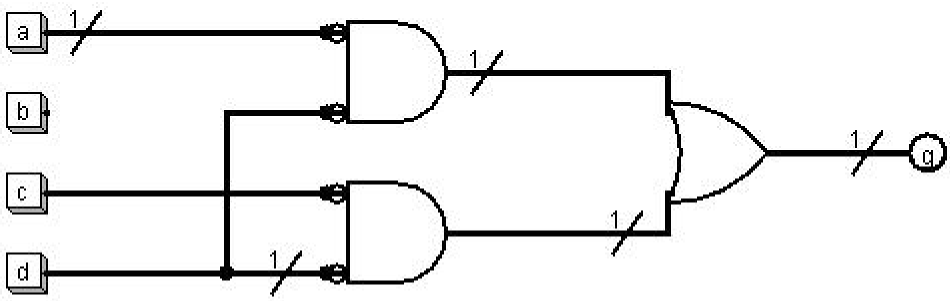
cd \ ab	00	01	11	10
00	1	X	1	X
01	0	0	0	0
11	X	X	0	0
10	X	1	0	0

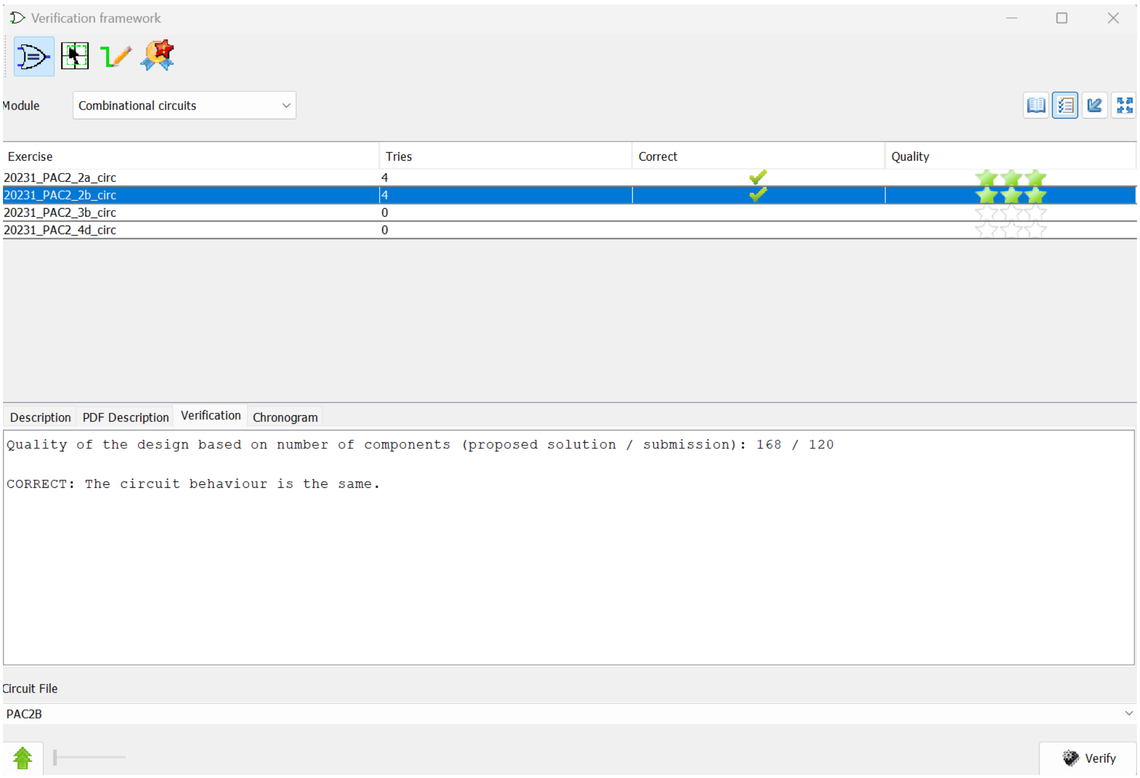
Por lo que la función resultante queda en:

$$f = \overline{cd} + \overline{ad}$$



Que resulta en el siguiente circuito:





c) Dado el siguiente cronograma, en el cual a, b y c son las señales de entrada de una ROM (en este orden) y o 1 y o 0 son sus salidas, obtened la tabla de verdad de la ROM expresando los valores en decimal.

Observando el cronograma, nos resulta en la siguiente tabla de verdad:

A	**B**	**C**	**O1**	**O0**
0	0	0	1	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	1	0
1	1	1	0	1

Ejercicio 3:

a) Rellenad la tabla de verdad de la función X(a, b, c, d), incluyendo los valores intermedios (e 0, e 1, e 2, e 3, s 1, s 0 y s) especificados en la tabla:

Ejercicio 4:

a) ¿Cuál es el valor máximo que puede tener X? ¿Cuál es el valor máximo que puede tener el producto 4·X? Indicadlos los dos en decimal. ¿Cuál es el valor máximo que puede tener la salida Y? ¿Cuál será, pues, la dimensión en número de bits de la salida Y?

X ha de ser mayor que 4, ya que si X es menor que 4, el producto de 4X – 16 sería negativo; por lo que rompería con la condición de que Y ha de ser un número natural (igual o mayor que 0).

Por tanto, puedes representar un total de 2^4 = 16 combinaciones diferentes, que van desde 0000 (0 en decimal) hasta 1111 (15 en decimal), que hace que X tenga un valor máximo de 15.

La salida Y, entonces, puede tener un valor máximo de: 4*15 – 16 = 44, que corresponde a una dimensión de 6 bits (un máximo de 64, que corresponde con 2^6).

b) ¿Qué rango de valores de la entrada X activarán la salida OVF?

La salida OVF se activará cuando X sea menor que 4. Cualquier valor de entrada X menor que 4 hará que la salida OVF sea igual a 1, indicando que el resultado Y no es representable como un número natural, y en ese caso, la salida Y será 0.

c) Si se quisiera implementar con una memoria ROM, indicad las 10 primeras posiciones. No hay que hacer el circuito, solo indicar las 10 primeras posiciones de la ROM.

Se indican los casos suponiendo que se desea implementar una ROM en la que la salida OVF = 0.

Dirección 0 (X = 4): Y = 4 * 4 - 16 = 0
Dirección 1 (X = 5): Y = 4 * 5 - 16 = 4
Dirección 2 (X = 6): Y = 4 * 6 - 16 = 8
Dirección 3 (X = 7): Y = 4 * 7 - 16 = 12
Dirección 4 (X = 8): Y = 4 * 8 - 16 = 16
Dirección 5 (X = 9): Y = 4 * 9 - 16 = 20
Dirección 6 (X = 10): Y = 4 * 10 - 16 = 24
Dirección 7 (X = 11): Y = 4 * 11 - 16 = 28
Dirección 8 (X = 12): Y = 4 * 12 - 16 = 32
Dirección 9 (X = 13): Y = 4 * 13 - 16 = 36

A continuación, los 10 primeros casos, independientemente de la salida OVF:

Dirección 0 (X = 0): Y = 4 * 0 - 16 = -16 (0)
Dirección 1 (X = 1): Y = 4 * 1 - 16 = -12 (0)
Dirección 2 (X = 2): Y = 4 * 2 - 16 = -8 (0)
Dirección 3 (X = 3): Y = 4 * 3 - 16 = -4 (0)
Dirección 4 (X = 4): Y = 4 * 4 - 16 = 0
Dirección 5 (X = 5): Y = 4 * 5 - 16 = 4
Dirección 6 (X = 6): Y = 4 * 6 - 16 = 8
Dirección 7 (X = 7): Y = 4 * 7 - 16 = 12
Dirección 8 (X = 8): Y = 4 * 8 - 16 = 16
Dirección 9 (X = 9): Y = 4 * 9 - 16 = 20