

PEC2 - Segunda prueba de evaluación continua

Presentación

Después de conocer los sistemas de representación de la información, esta PEC se focaliza en los circuitos lógicos combinacionales. Las funciones lógicas nos permiten describir la funcionalidad de un circuito y mediante las puertas lógicas y los bloques combinacionales podemos implementarlas en un circuito. Todo esto, no se podría hacer sin saber mecanismos de minimización, como mapas de Karnaugh, que nos permiten reducir las dimensiones del circuito que se tiene que implementar.

Competencias

- Conocer y saber aplicar el álgebra de Boole para la manipulación de funciones lógicas.
- Tener nociones tecnológicas de los circuitos digitales y entender la relación entre los circuitos digitales y las funciones lógicas.
- Conocer y saber utilizar las puertas lógicas y los módulos combinacionales en el diseño de circuitos lógicos.

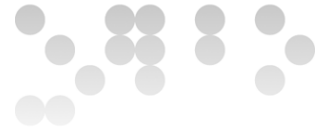
Objetivos

- Saber aplicar las operaciones básicas y los axiomas del álgebra de Boole.
- Saber representar las funciones lógicas mediante tablas de verdad.
- Saber representar las funciones lógicas mediante expresiones algebraicas.
- Saber analizar un circuito combinacional.
- Saber realizar un cronograma a partir de un circuito digital combinacional.
- Saber sintetizar una función a dos niveles.
- Saber diseñar un circuito combinacional sencillo a partir de los bloques combinacionales de los materiales.

Recursos

Los recursos que se recomienda utilizar para esta PEC son los siguientes:

- **Básicos:** El módulo 3 de los materiales. En cada pregunta, se indica qué sección de los materiales se puede encontrar la información para resolverlos. Notad que también existe en los materiales una cantidad muy extensa de ejercicios para ver cómo se pueden resolver
- **PEC anteriores:** En el aula de CANVAS, en recursos adicionales podéis encontrar PECs de otros semestres resueltas.
- **Complementarios:** KeMap, VerilCIRC, VerilCHART y el Wiki de la asignatura.



Criterios de valoración

- Razonad la respuesta en todos los ejercicios. Las respuestas sin justificación no recibirán puntuación.
- Los ejercicios realizados con IA generativa no recibirán puntuación.
- Los ejercicios que se detecten como plagio mediante la herramienta de plagio de la universidad no recibirán puntuación.
-
- La valoración está indicada en cada uno de los subapartados

Uso de herramientas de IA

En esta actividad no está permitido el uso de herramientas de inteligencia artificial.

En el plan docente y en el [web sobre integridad académica y plagio](#) de la UOC encontraréis información sobre qué se considera conducta irregular en la evaluación y las consecuencias que puede tener.

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 de la Actividad** correspondiente del apartado **Contenidos** de vuestra aula.
- La fecha límite de entrega es el **9 de abril** (a las 24 horas).

Enunciado de la PEC

Ejercicio 1 [25 %]

Dada la siguiente expresión lógica de la función $f(a,b,c,d)$:

$$f(a, b, c, d) = (\overline{a + b + c}) + (a \cdot b \cdot c) + \bar{c} + (a \oplus b) \cdot d + \bar{d}$$

- [10 %] Simplificad la función f usando las propiedades y las leyes del álgebra de Boole. (Secciones 1.2. Álgebra de Boole y 1.3.1. Expresiones algebraicas)
- [5%] Obtened la tabla de verdad de la función. (Sección 1.3.2. Tablas de verdad)

[illegible]

-
- The diagram shows five digital signals (a, b, c, d, f) over time, with vertical dashed lines marking time points t_1 through t_5 .
- Signal a:** Low until t_3 , then high.
 - Signal b:** Low until t_1 , high until t_2 , low until t_4 , and high until t_5 .
 - Signal c:** High until t_1 , low until t_2 , high until t_3 , low until t_4 , high until t_5 , and low.
 - Signal d:** High until t_1 , low until t_3 , high until t_4 , low until t_5 , and high.
 - Signal f:** High until t_1 , low until t_2 , high until t_3 , low until t_4 , and high until t_5 .

la tabla de verdad siguiente:

Dada la tabla de verdad siguiente:



a	b	c	d	f	g
0	0	0	0	X	1
0	0	0	1	0	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	X	0
0	1	0	1	0	0
0	1	1	0	1	1
0	1	1	1	0	0
1	0	0	0	0	1
1	0	0	1	X	0
1	0	1	0	0	1
1	0	1	1	1	1
1	1	0	0	0	1
1	1	0	1	X	0
1	1	1	0	1	1
1	1	1	1	1	0

- a) [7,5%] Utilizando el método de Karnaugh, indicad las dos funciones mínimas posibles de la función f . ¿Cuál de las dos expresiones mínimas utiliza una cantidad menor de puertas lógicas? Implementad esta última opción con puertas lógicas en dos niveles. (Sección 2.3.2. Síntesis mínima a dos niveles. Método de Karnaugh)
- b) [5%] Hemos preguntado a una herramienta de IA generativa la minimización mediante el método de Karnaugh de la función g , y nos ha dado el siguiente resultado (Sección 2.3.2. Síntesis mínima a dos niveles. Método de Karnaugh):

Paso 1: Organización en un mapa de Karnaugh 4x4

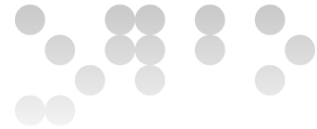
ab\cd	00	01	11	10
00	1	0	1	0
01	0	0	0	1
11	1	0	1	1
10	1	0	1	1

Paso 2: Agrupaciones y obtención de la función mínima

- **Grupo 1:** $g = a$ (seleccionando la columna $a=1$ donde $g=1$)
- **Grupo 2:** $g = c \cdot d$ (para agrupar las celdas donde solo depende de c y d)

Resultado final: Función mínima simplificada:

$$g(a, b, c, d) = a + (c \cdot d)$$



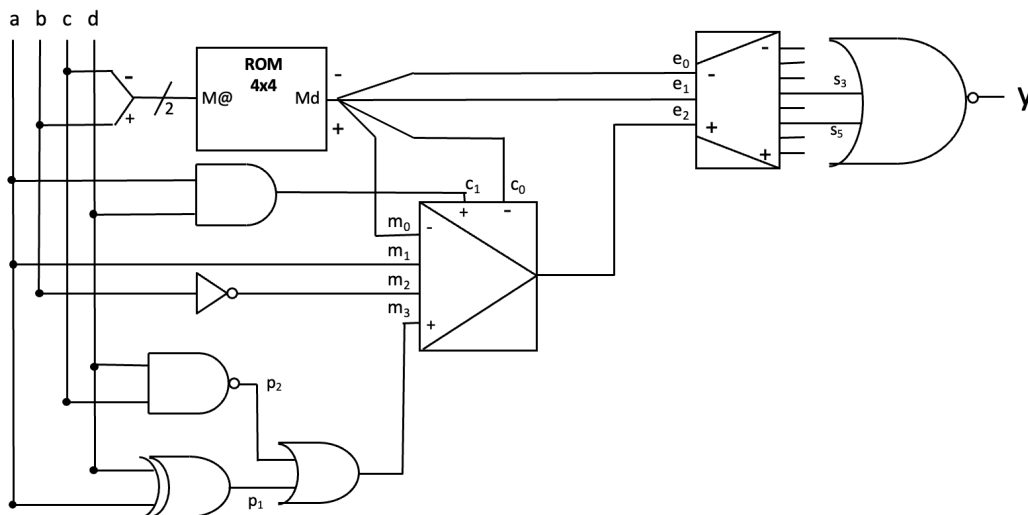
Razonad la corrección de la solución y, en caso de ser incorrecta, indicad en cada paso los errores detectados, pero no hace falta obtener la solución correcta.

- c) [7,5%] Sintetizad la función g con un multiplexor de 3 variables de control y el menor número de puertas lógicas. (Sección 3.1. Multiplexor. Multiplexor de buses. Demultiplexor)

NOTA: Tenéis disponible el ejercicio en VerilCIRC y en KeMAP para verificarlo. Os recomendamos hacer pruebas con KeMAP con otros ejercicios antes de hacer el de la PEC.

Ejercicio 3 [25 %]

Dado el circuito lógico combinacional siguiente:



Donde el contenido de la memoria ROM es el siguiente:

M@	Md
0	0000
1	0111
2	1100
3	1001

Completad la tabla de verdad que especifica la salida y en función de las entradas a , b , c y d . Calculad previamente los valores intermedios (p_2 , p_1 , m_3 , m_2 , m_1 , m_0 , c_1 , c_0 , e_2 , e_1 , e_0 , s_5 , s_3) indicados en el circuito y añadidos en la tabla de verdad siguiente (Sección 3. Bloques combinacionales):

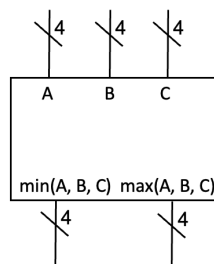


a	b	c	d	p ₂	p ₁	m ₃	m ₂	m ₁	m ₀	c ₁	c ₀	e ₂	e ₁	e ₀	s ₅	s ₃	y
0	0	0	0														
0	0	0	1														
0	0	1	0														
0	0	1	1														
0	1	0	0														
0	1	0	1														
0	1	1	0														
0	1	1	1														
1	0	0	0														
1	0	0	1														
1	0	1	0														
1	0	1	1														
1	1	0	0														
1	1	0	1														
1	1	1	0														
1	1	1	1														

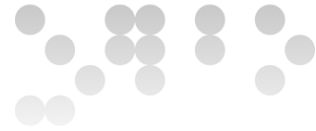
NOTA: Tenéis disponible el ejercicio en VerilChart para verificarlo.

Ejercicio 4 [30 %]

Se desea diseñar un circuito que, dados 3 números de 4 bits en complemento a 2, muestre el menor de los tres valores en la salida $\min(A,B,C)$ y el mayor de los tres valores en la salida $\max(A,B,C)$ siguiendo la siguiente estructura (Secció 3. Bloques combinacionales):



- [20%] Diseñad un circuito que, dados dos números A y B de 4 bits en complemento a 2, obtenga el menor y el mayor de los dos operandos a partir de bloques combinacionales (excepto ROMs) y las puertas que consideréis necesarias. Llamad a las salidas del circuito como $\min2$ para $\min(A,B)$ y $\max2$ para $\max(A,B)$.
- [10%] Utilizando el circuito del apartado a), bloques combinacionales (excepto ROMs) y las puertas que consideréis necesarias, diseñad el circuito $\min(A,B,C)$ y $\max(A,B,C)$. Llamad a las salidas del circuito como $\min3$ y $\max3$ respectivamente.



NOTA: Tenéis disponible el ejercicio en VerilCIRC. No hay límite en el número de intentos que uséis para comprobar vuestra solución.

NOTA2: En el caso de diseñar el circuito en VerilCIRC, la herramienta no admite utilizar bloques diseñados por el usuario. Por lo tanto, en caso de reutilizar un circuito, como por ejemplo el circuito *del apartado a)*, tendréis que copiar el circuito tantas veces como necesitéis utilizarlo.