

Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

Este enunciado también corresponde a las siguientes asignaturas:

- 81.518 - Fundamentos de computadores

Ficha técnica del examen

- Comprueba que el código y el nombre de la asignatura corresponden a la asignatura de la que te has matriculado.
 - Tiempo total: **2 horas** Valor de cada pregunta: **P1: 20%; P2: 35%; P3: 35%, i P4: 10%**
 - ¿Puede consultarse algún material durante el examen? **NO** ¿Qué materiales están permitidos?
 - ¿Puede utilizarse calculadora? **NO** ¿De qué tipo? **NINGUNO**
 - Si hay preguntas tipo test, ¿descuentan las respuestas erróneas? **NO** ¿Cuánto?
 - Indicaciones específicas para la realización de este examen: **Razonad todas las respuestas. Las respuestas sin justificar no serán puntuadas.**
-

Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

Enunciados

PROBLEMA 1 [20%]

- a) [10%] Dados los valores $A = 01111110$ y $B = 11101101$, que representan números binarios enteros codificados en signo y magnitud con 8 bits. Realizad la operación $A + B$ en signo y magnitud manteniendo el formato de representación con 8 bits. Si se produce desbordamiento, explicad por qué razón y sino indicad el valor del resultado en base 10.

Dado que en los números están en base 2, podemos comprobar que los dos números son de signo diferente. Por lo tanto, la operación de suma realmente es una operación de resta. Para realizarla, hay que eliminar el bit de signo y restar el número de magnitud más pequeño del otro (el número más grande es el A) y finalmente asignar el signo del número de magnitud más grande al resultado de la operación.

$$\begin{array}{r}
 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad \leftarrow A \\
 1 \quad \text{acarreo} \\
 - 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad \leftarrow B \\
 \hline
 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1
 \end{array}$$

Al resultado se le añade la información del signo del primero operando (número más grande). Por lo tanto, el resultado es $A + B = 00010001$.

No hay desbordamiento puesto que se han sumado números de signo diferente y, por lo tanto, el resultado de la operación es siempre correcto.

El resultado en base 10 es $126 - 109 = 17_{(10)}$.

- b) [10%] Dado el formato de coma flotante siguiente:

S	Exponente		Mantisa	
14	13	9	8	0

Donde:

- El bit de signo, S, vale 0 para cantidades positivas y 1 para negativas.
- El exponente se representa en exceso a 16.
- Hay bit implícito.
- La mantisa está normalizada en la forma $1.X$.

¿A qué número decimal corresponde la secuencia de bits 010100000010101 codificada en este formato de coma flotante?

Para encontrar cuál es el número en decimal que estamos representando en coma flotante, en primer lugar tenemos que ver qué bits corresponden a cada campo (signo, exponente y mantisa) agrupándolos de manera conveniente:

S	Exponente	Mantisa
0	10100	000010101

Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

Signo: El bit de signo es 0, y por lo tanto el número es positivo.

Exponente: $10100_2 = 20_{10}$. Cómo el exponente está representado en exceso a 16, su valor es $20 - 16 = 4_{10}$.

Mantisa: Cómo hay bit implícito, es: 1,000010101.

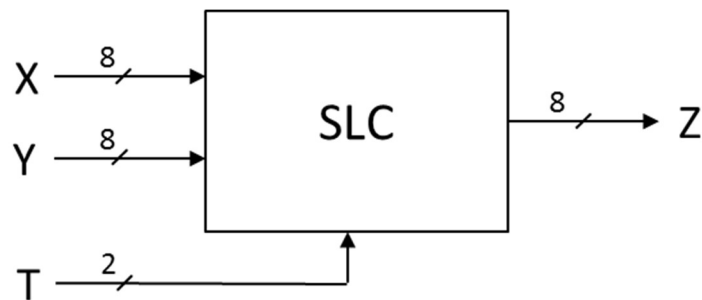
Si aplicamos el TFN, obtendremos el valor:

$$+ 1,000010101 \cdot 2^4 = 10000,10101 = 2^4 + 2^{-1} + 2^{-3} + 2^{-5} = 16,656_{10}.$$

El equivalente decimal es $+16,656_{10}$.

PROBLEMA 2 [35%]

a) [25 %] Un sistema combinacional SLC tiene las siguientes entradas y salidas:



donde $Z = X + Y$ teniendo en cuenta que los números están representados en el formato que especifica T , según indica la siguiente tabla:

t_1	t_0	Representación X, Y y Z
0	0	Binario natural
0	1	Ca2
1	0	Signo y Magnitud
1	1	Combinación no usada

Implementad el circuito SLC usando los bloques y puertas que consideréis necesarios (excepto memoria ROM). El circuito NO tiene que comprobar ninguna situación de desbordamiento. Sólo tiene que hacer la operación de suma aritmética.

Para hacer la suma en binario natural y Ca2 sin tener en cuenta si se produce desbordamiento sólo hace falta un sumador (marcado en verde a la figura).

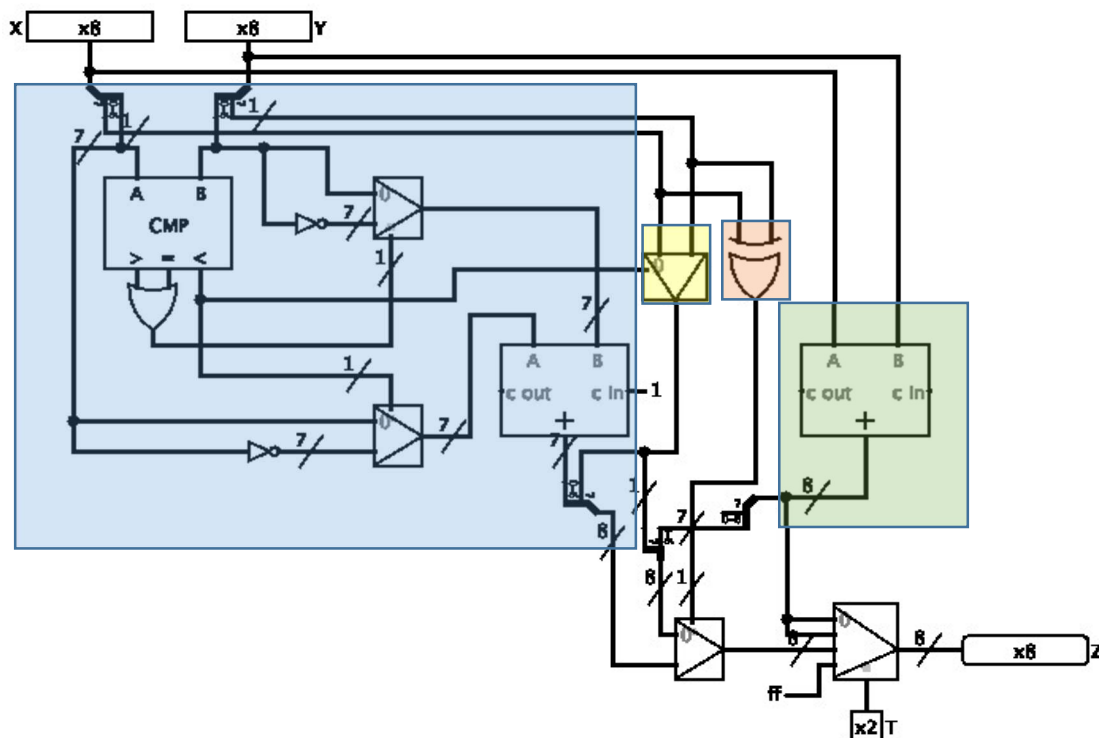
Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

Para hacer la suma en Signo y Magnitud tenemos que hacer:

- Comprobar si los operandos tienen el mismo signo o no (puerta xor marcada en naranja)
- Si los signos son iguales (xor con valor 0) las magnitudes se suman, para eso aprovechamos los 7 bits menos significativos del sumador marcado en verde (el que hemos usado para los otros dos casos).
- Si los signos son diferentes (xor con valor 1) se resta la magnitud menor de la magnitud más grande. Para eso usamos el comparador, los multiplexores y el sumador del bloque marcado en azul. Con el comparador determinamos cuál es la magnitud menor y determinamos cuál de las dos magnitudes tenemos que negar para hacer el resto.
- Finalmente, hay que determinar el signo del resultado, para lo cual volvemos a usar la salida del comparador como selector de un multiplexor (marca en amarillo) que tiene como entradas los bits más significativos de X y Y.

Así, ya tenemos los bloques que calculan $X + Y$ en las tres representaciones y usamos la señal T para seleccionar cuál será el resultado que producirá el circuito.



Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

b) **[10 %]** Minimiza la siguiente función por Karnaugh y haz la síntesis del circuito a dos niveles:

a	b	c	d	h
0	0	0	0	1
0	0	0	1	0
0	0	1	0	X
0	0	1	1	1
0	1	0	0	X
0	1	0	1	0
0	1	1	0	X
0	1	1	1	1
1	0	0	0	X
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

El mapa de Karnaugh de la función h es el siguiente:

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

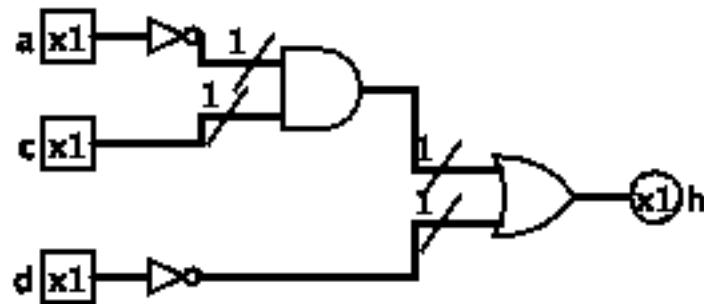
Y obtenemos la expresión mínima siguiente:

$$h = a' \cdot c + d'$$

y el circuito de dos niveles que la sintetiza:

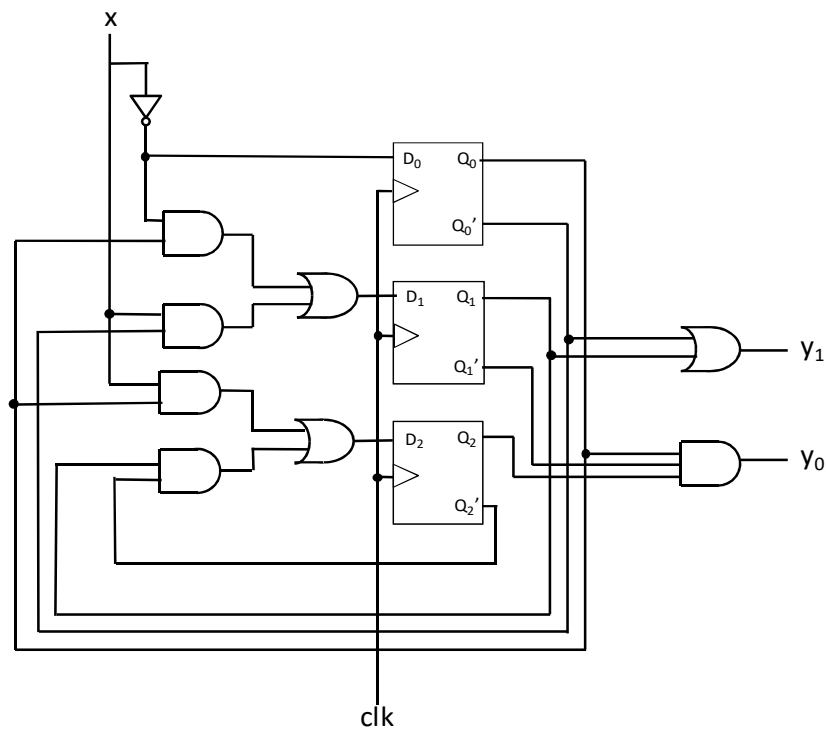
Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30



PROBLEMA 3 [35%]

a) [20%] Dado el circuito siguiente:



Se pide analizar el circuito y construir la tabla de transiciones y la tabla de salidas correspondientes. Al escribir la tabla de transiciones, poned en primer lugar las variables que codifican el estado y a continuación las variables de entrada.

Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

El circuito tiene una señal de entrada, x , y dos señales de salida, y_1 e y_0 , y está formado por puertas lógicas y biestables. Tres biestables forman la parte central del circuito, de forma que se pueden guardar $2^3 = 8$ estados diferentes.

Los nuevos estados se generan a partir del propio estado actual (los tres biestables) y de la señal de entrada x , aplicando lógica combinacional (puertas lógicas). En concreto:

- $d_2 = x \cdot q_0 + q_1 \cdot q_2'$
- $d_1 = x' \cdot q_0 + x \cdot q_0'$
- $d_0 = x'$

Y las salidas se generan a partir del estado actual aplicando lógica combinacional. En concreto:

- $y_1 = q_0' + q_1$
- $y_0 = q_0 \cdot q_1' \cdot q_2$

Así pues, la tabla de transiciones es la siguiente:

q_2	q_1	q_0	x	d_2	d_1	d_0
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	1
0	0	1	1	1	0	0
0	1	0	0	1	0	1
0	1	0	1	1	1	0
0	1	1	0	1	1	1
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	1	0
1	0	1	0	0	1	1
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	1	0
1	1	1	0	0	1	1
1	1	1	1	1	0	0

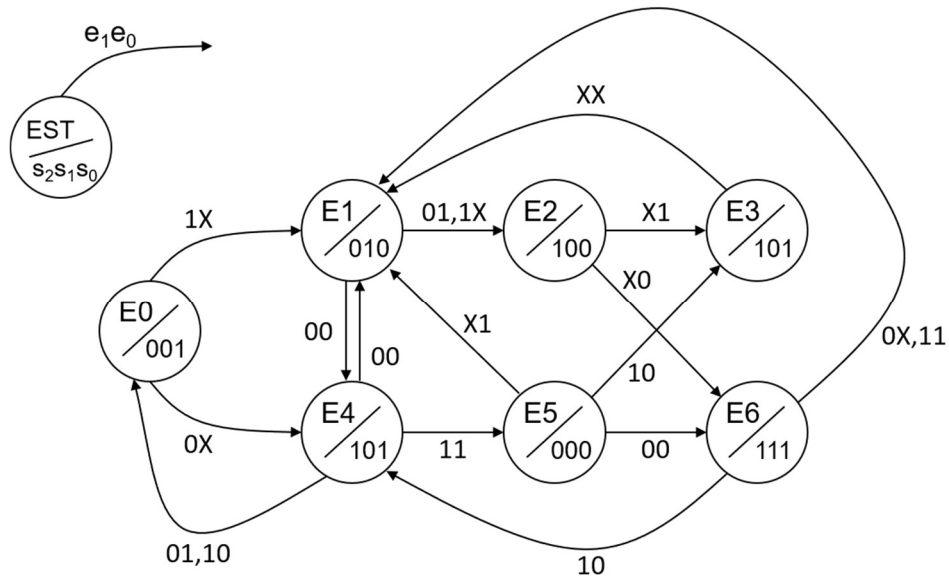
Y la tabla de salidas es la siguiente:

q_2	q_1	q_0	y_1	y_0
0	0	0	1	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	0
1	0	1	0	1
1	1	0	1	0
1	1	1	1	0

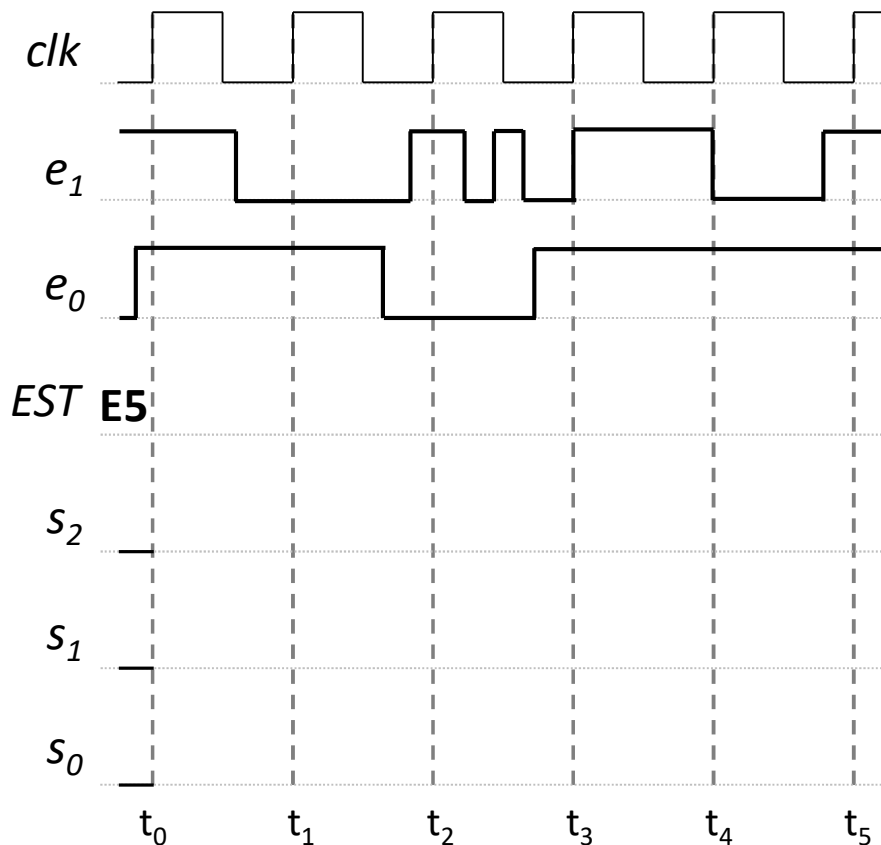
b) **[15%]** Dado el grafo de estados siguiente, que implementa un circuito lógico secuencial:

Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30

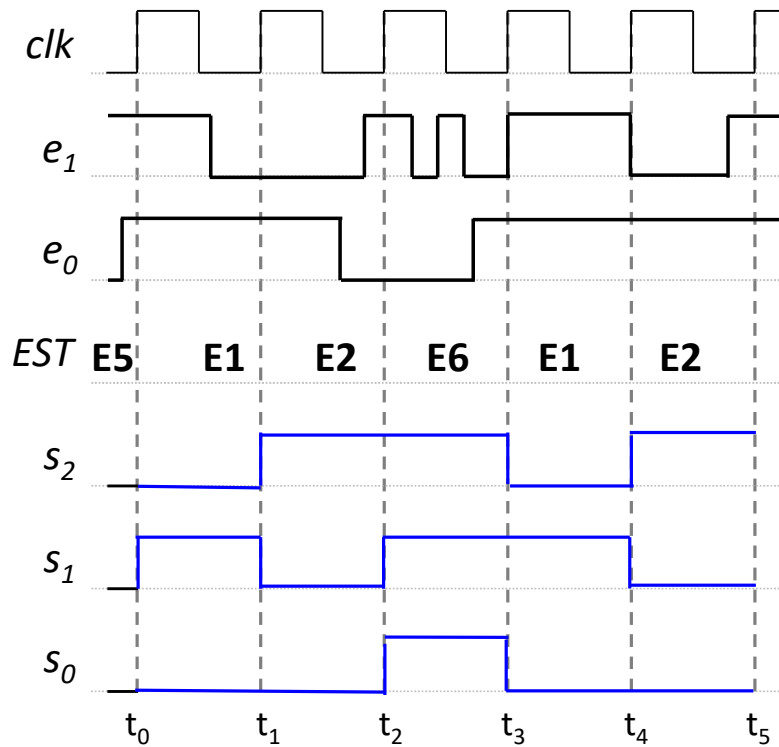


Completad el cronograma siguiente:



Examen 2020/21-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/1/2021	18:30



PROBLEMA 4 [10%]

a) [5%] ¿Qué es la arquitectura de Harvard?

Una manera de construir máquinas que tienen una memoria para las instrucciones y otra para los datos.

b) [5%] ¿Para qué se usa el PC (*program counter*)?

Para guardar la posición de memoria de la instrucción a ejecutar.