

Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

Este enunciado corresponde también 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 matriculada.
- Tiempo total: **2 horas** Valor de cada pregunta: **P1: 20%, P2: 35%, P3: 35% y P4: 10%.**
- En el caso de que los estudiantes no puedan consultar algún material durante el examen, ¿cuáles son?:
- Se puede utilizar calculadora? **NO** De que tipo? **NINGUNO**
- En el caso de que haya preguntas tipo test: ¿descuentan las respuestas erróneas? **NO**
¿Cuánto?
- Indicaciones específicas para la realización de este examen
 - **Razonad las respuestas en cada ejercicio.**
 - **Las respuestas sin justificar no recibirán puntuación.**

Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

Enunciados

PROBLEMA 1 [20%]

- a) [10%] Dados los valores $A = 10000010$ y $B = 00010011$, 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. ¿Se produce desbordamiento?

Dado que en los números con signo y magnitud el primer dígito codifica el signo, 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 B) y finalmente asignar el signo del número de magnitud más grande al resultado de la resta.

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

← acarreo

Al resultado se le añade la información del signo del primer 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.

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

S	Exponente		Mantisa		
13	12	8	7	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 01010000001000 codificado 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	00001000

Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

- Signo: El bit de signo es 0, y por lo tanto el número es positivo.
- Exponente: $10100_2 = 20_{10}$. Como que el exponente está representado en exceso a 16, su valor es $20 - 16 = 4_{10}$.
- Mantisa: Como hay bit implícito, es: 1,00001000.

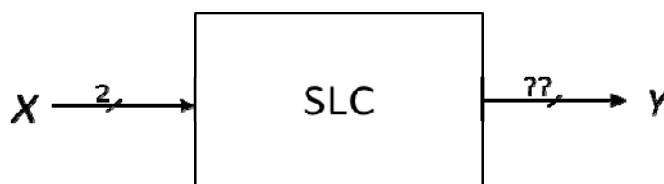
Si aplicamos el TFN, obtendremos el valor:

$$+ 1,0000100 \cdot 2^4 = 10000,100 = 2^4 + 2^{-1} = 16,5_{10}.$$

El equivalente decimal es $+16,5_{10}$.

PROBLEMA 2 [35%]

a) [25%] Un sistema combinacional SLC responde al siguiente esquema:



donde X e Y son números enteros codificados en complemento a 2. La entrada X tiene 2 bits. La salida Y calcula la siguiente función matemática:

$$Y = f(X) = X^2 - 2 \cdot X + 1$$

Se pide que respondáis los siguientes apartados:

- i. [5%] ¿Cuántos bits debe tener como mínimo la salida Y para que no se produzca nunca desbordamiento?

Como $X \in [-2, 1] \rightarrow X^2 \in [0, 4]$ y $-2 \cdot X + 1 \in [-1, 5]$. Así $X^2 - 2 \cdot X + 1 \in [0, 9]$, puesto que el valor máximo de la expresión lo tenemos por $X = -1$ (9) y el mínimo por $X = 1$ (0). El +9 se codifica como 01001 en Ca2, lo cual implica que **Y tiene que tener como mínimo 5 bits para que no se produzca nunca desbordamiento.**

- ii. [10%] Implementad la salida Y con una memoria ROM y las puertas lógicas que consideréis necesarias. Indicad en hexadecimal el contenido de todas las posiciones de la memoria ROM.

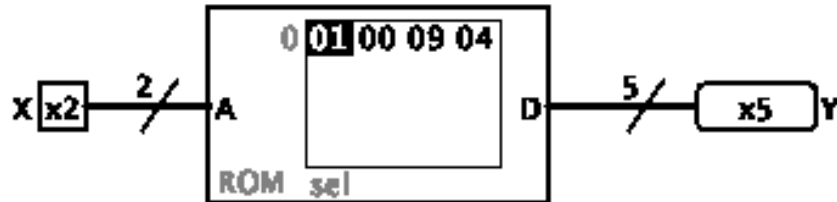
Calculamos el valor de la función para cada uno de los posibles valores de Y , así:

X	Y
00	00001
01	00000
10	01001
11	00100

En consecuencia, si usamos los dos bits de X para direccionar una memoria ROM de 4 celdas de 5 bits, podemos implementar la salida Y .

Examen 2019/20-2

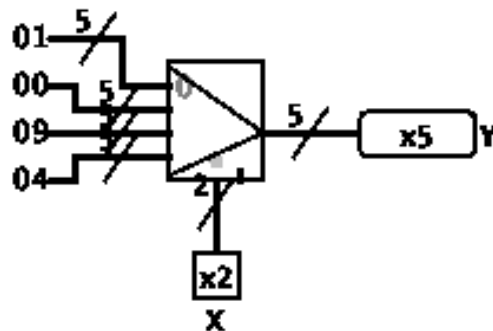
Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00



- iii. **[10%]** Implementad la salida Y usando un multiplexor y las puertas lógicas que consideréis necesarias.

Usamos los valores calculados para la función Y en el apartado anterior y los ponemos como entradas de un multiplexor que tiene como señales de control los bits de la variable X .

El circuito resultante es el siguiente:



- b) **[10%]** Minimizad la siguiente función mediante Karnaugh y expresad algebraicamente la función resultante:

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

Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

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

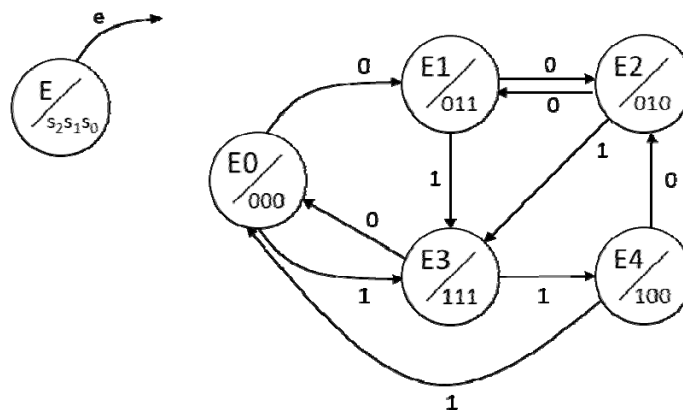
ab \ cd	00	01	11	10
00	0	x	1	x
01	x	1	x	x
11	1	0	0	x
10	0	0	0	0

Y obtenemos esta expresión mínima:

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

PROBLEMA 3 [35%]

a) [15%] Dado el grafo de estados siguiente:



Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

- i) **[5%]** ¿Cuántos bits de entrada tiene el circuito que implementa este grafo? ¿Cuántos bits de salida? ¿Cuál será el número mínimo de biestables para implementarlo? Si lo implementamos usando una memoria ROM, ¿cuál será el tamaño (bus de direcciones, ancho de las palabras) de esta memoria?

El circuito tiene un bit de entrada, denominado e . Los bits de salida son tres, denominados s_2 , s_1 , s_0 . Como el circuito tiene 5 estados necesitamos 3 bits para representarlos y, por lo tanto, necesitamos 3 biestables para poder almacenar estos 3 bits.

La memoria ROM que puede implementar este circuito tendrá 4 bits de direccionamiento, 3 bits para el estado más 1 bit para la entrada. El ancho de las palabras de la ROM deberá ser de 6 bits, 3 bits para guardar la salida y 3 bits para codificar el estado futuro.

- ii) **[10%]** Asumiendo que la codificación de cada estado corresponde al número de estado, tal y como es habitual en este tipo de ejercicios, indicad en la tabla adjunta el contenido en binario de la memoria ROM para las direcciones de memoria especificadas:

Dirección	Contenido
0x05	
0x06	
0x07	
0x08	
0x09	
0x0A	

El direccionamiento de la memoria ROM viene definido por las señales q_2 , q_1 , q_0 y e especificadas en orden de mayor a menor peso. Los tres primeros bits representan el estado y el bit de menos peso es la entrada.

Cada posición de la memoria almacena los bits q_2+ , q_1+ , q_0+ , s_2 , s_1 y s_0 , especificados en orden de mayor a menor peso. Los tres primeros bits especifican el estado futuro y los otros tres son las salidas.

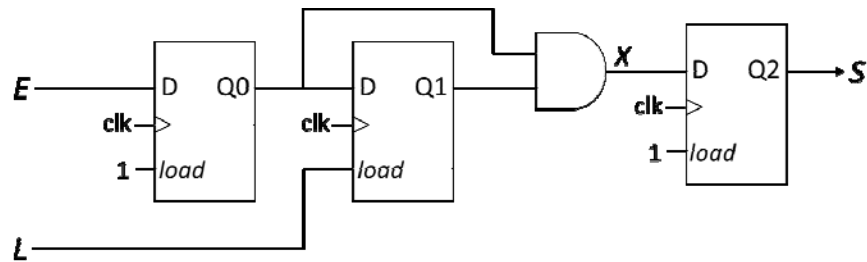
El contenido de las direcciones 0x05 hasta 0x0A de la memoria ROM será:

Dirección	Contenido
0x05	011 010
0x06	000 111
0x07	100 111
0x08	010 100
0x09	000 100
0x0A	xxx xxx

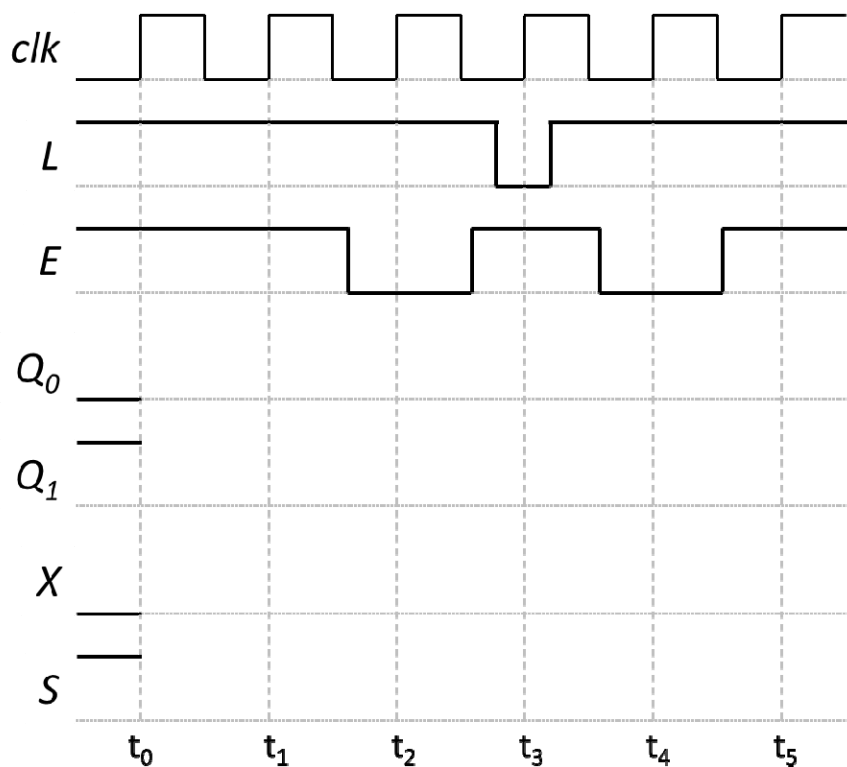
Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

b) **[20%]** Dado el circuito secuencial siguiente:



Completad el cronograma siguiente, incluyendo el valor de la salida de la puerta AND (señal X en el circuito):



Empezamos analizando la señal Q_0 . El biestable asociado se carga a cada ciclo de reloj con la entrada E .

El biestable asociado a la señal Q_1 se carga con la señal Q_0 si la señal de carga $L=1$. Los instantes en que el biestable se carga son: t_0 , t_1 , t_2 , t_4 y t_5 . Como ya hemos calculado la señal Q_0 para todo el cronograma, ahora podemos completar la señal Q_1 para todo el cronograma también.

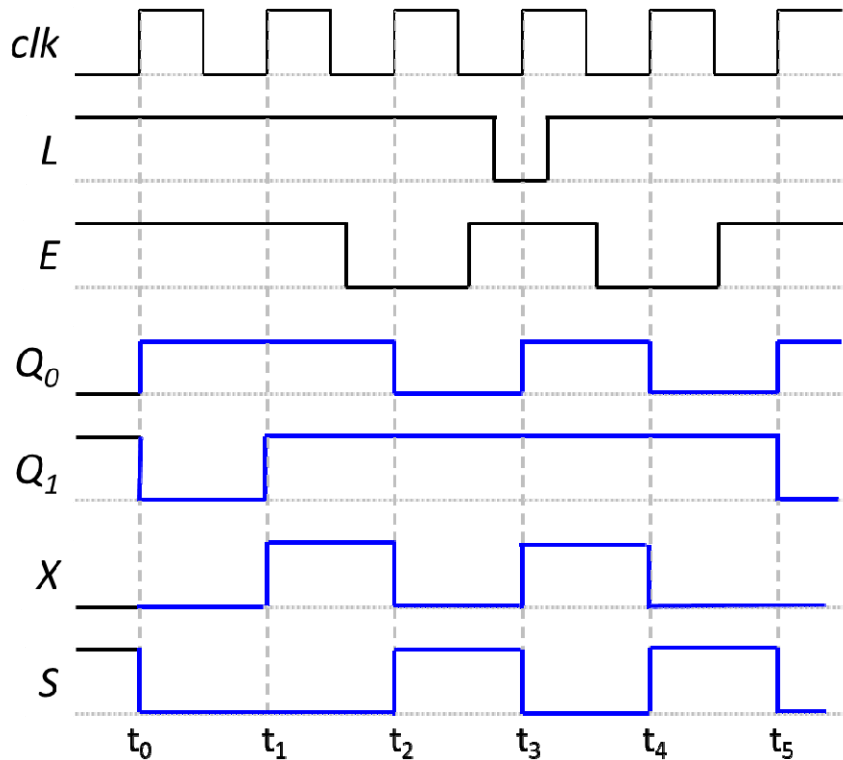
Para establecer los valores de Q_2 necesitamos calcular antes la señal X . Esta señal X es la salida de una puerta AND entre los bits Q_1 y Q_0 . Podemos, pues, completar el valor de la señal X para todo el cronograma.

Examen 2019/20-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	10/06/2020	09:00

Por último, pasamos a razonar el valor de Q_2 . El biestable asociado se carga siempre con el valor de la señal X . Por lo tanto, la salida S , que se corresponde con el valor de Q_2 , pasará a ser la señal X desplazada un ciclo de reloj.

El cronograma completo es el siguiente.



PROBLEMA 4 [10%]

- a) [5%] ¿Cuál es la función de los buses en un computador?

Conectar dos o más componentes entre ellos.

- b) [5%] ¿En qué consiste el cálculo de la instrucción siguiente?

En determinar la dirección de memoria de la instrucción siguiente.