

Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

Este enunciado también corresponde a las siguientes asignaturas:

- 81.518 - Fundamentos de computadores

Ficha técnica del examen

- No es necesario que escribas tu nombre. Una vez resuelta la prueba final, solo se aceptan documentos en formato .doc, .docx (Word) y .pdf.
 - 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%; P4:10%**
 - ¿Puede consultarse algún material durante el examen? **NO** ¿Qué materiales están permitidos? **Ninguno.**
 - ¿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 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

Enunciados

PROBLEMA 1 [20%]

- a) **[10%]** Dados los números $A = 101100110$ y $B = 101110011$, que representan números enteros binarios codificados en complemento a 2 y 9 bits, realizad la operación $A - B$. Explicad si se produce o no desbordamiento y por qué razón. Indicad cuál es el rango del resultado de esta operación expresándolo en decimal.

Para restar dos números en formato de Ca2 convertimos la operación $A - B$ en $A + (-B)$, cambiando el signo del sustraendo.

Para cambiar de signo un número representado en Ca2 complementamos todos sus bits y sumamos 1 al resultado. Por lo tanto, para cambiar el signo del número B:

$$\begin{array}{r}
 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1 \quad \leftarrow B \\
 \downarrow \\
 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0 \quad \leftarrow B' \\
 + \quad \quad \quad 1 \\
 \hline
 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1 \quad \leftarrow -B
 \end{array}$$

A continuación hacemos la operación de suma de los dos números:

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

No hay desbordamiento ya que es una suma de dos operandos de signo contrario. El resultado final es $111110011_{(Ca2)}$.

El rango de representación de un número en complemento a 2 es $[-(2^{n-1}), 2^{n-1}-1]$. El rango de representación del resultado de esta operación es el mismo puesto que podemos obtener estos mismos valores extremos con múltiples combinaciones de los operandos de entrada. Por lo tanto, en este caso que tenemos números de 9 bits ($n=9$) el rango es $[-(2^8), 2^8-1] = [-256, 255]$.

- b) **[5%]** Dado el número $C = 100010011$ que representa un número natural expresado en binario natural con 9 bits, indicad su valor en base 10.

Para obtener el valor decimal de un número natural binario aplicamos el TFN:

$$1 \cdot 2^8 + 0 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 256 + 16 + 2 + 1 = 275_{(10)}$$

- c) **[5%]** Dado el formato de coma flotante siguiente:

S	Exponente			Mantisa		
12	11		8	7		0

Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

Donde:

- El bit de signo (posición 12), S, vale 0 para cantidades positivas y 1 para negativas.
- El exponente se representa en exceso a 8, con 4 bits de la posición 11 a la 8.
- La mantisa está normalizada en la forma 1,X. Se representa con bit implícito y con 8 bits de la posición 7 a la 0.

¿A qué número decimal corresponde la secuencia de bits 0111101100001 codificado en este formato de coma flotante?

Analizando el número según el formato tenemos:

- Signo: positivo, $S=0$
- Exponente: 1111_2 en exceso a 8, $15-8 = 7_{(10)}$.
- Mantisa: $1,01100001_2$.

Si juntamos la mantisa y el exponente, obtenemos el número:

$$1,01100001_2 \cdot 2^7 = 10110000,1_2$$

Y si aplicamos el TFN:

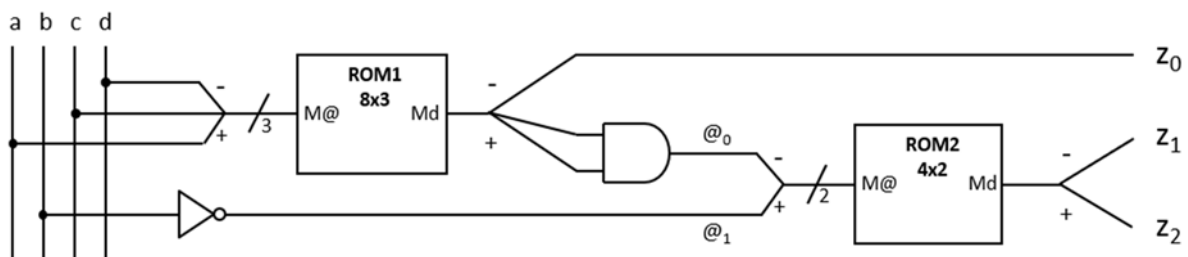
Parte entera: $1 \cdot 2^7 + 1 \cdot 2^5 + 1 \cdot 2^4 = 128 + 32 + 16 = 176$

Parte decimal: $1 \cdot 2^{-1} = 0,5$

Por lo tanto, en decimal el número que nos han dado es **+176,5₍₁₀₎**.

PROBLEMA 2 [35%]

a) [10%] Dado el circuito lógico combinacional siguiente:



Donde el contenido de la memoria ROM1, especificado en hexadecimal, es:

M@	[Md]
0	6
1	5
2	2
3	7
4	4
5	3
6	6
7	1

Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

Y el contenido de la memoria ROM2, también en hexadecimal, es:

M@	[Md]
0	3
1	2
2	1
3	0

Completad la tabla de verdad que especifica las salidas z_0 , z_1 , z_2 en función de las entradas a , b , c y d . Calculad previamente los valores intermedios indicados en el circuito y añadidos a la tabla de verdad siguiente:

a	b	c	d	@ ₁	@ ₀	z ₂	z ₁	z ₀
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					

Para evaluar la función $@_0$ debemos tener en cuenta que es una operación AND de las dos salidas más significativas de la ROM1. Además, la salida z_0 es la salida menos significativa de la ROM1. Así que considerando el contenido de la ROM y sus entradas, podemos expresar la tabla del contenido de la ROM en función de las conexiones de este circuito, es decir:

M@ →	a c d	[Md] →	Md ₂ Md ₁ Md ₀	@ ₀ = Md ₂ · Md ₁	z ₀
0	0 0 0	6	1 1 0	1	0
1	0 0 1	5	1 0 1	0	1
2	0 1 0	2	0 1 0	0	0
3	0 1 1	7	1 1 1	1	1
4	1 0 0	4	1 0 0	0	0
5	1 0 1	3	0 1 1	0	1
6	1 1 0	6	1 1 0	1	0
7	1 1 1	1	0 0 1	0	1

Ahora podemos completar la tabla de verdad del enunciado teniendo en cuenta la tabla anterior. Finalmente, también podemos evaluar la columna $@_1$ considerando que $@_1 = b'$.

Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

a	b	c	d	@ ₁	@ ₀	Z ₂	Z ₁	Z ₀
0	0	0	0	1	1			0
0	0	0	1	1	0			1
0	0	1	0	1	0			0
0	0	1	1	1	1			1
0	1	0	0	0	1			0
0	1	0	1	0	0			1
0	1	1	0	0	0			0
0	1	1	1	0	1			1
1	0	0	0	1	0			0
1	0	0	1	1	0			1
1	0	1	0	1	1			0
1	0	1	1	1	0			1
1	1	0	0	0	0			0
1	1	0	1	0	0			1
1	1	1	0	0	1			0
1	1	1	1	0	0			1

Finalmente, las salidas z_2 y z_1 dependen del contenido de la ROM2 y de las entradas conectadas a esta ROM. Por eso, procedemos de la misma forma que en el caso de la ROM1, representando su contenido considerando entradas y salidas conectadas en el circuito:

M@ →	@ ₁ @ ₀	[Md] →	Z ₂ Z ₁
0	0 0	3	1 1
1	0 1	2	1 0
2	1 0	1	0 1
3	1 1	0	0 0

Ahora ya podemos completar las dos columnas de las funciones z_2 y z_1 :

a	b	c	d	@ ₁	@ ₀	Z ₂	Z ₁	Z ₀
0	0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	1	1
0	0	1	0	1	0	0	1	0
0	0	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0
0	1	0	1	0	0	1	1	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	1	1	0	1
1	0	0	0	1	0	0	1	0
1	0	0	1	1	0	0	1	1
1	0	1	0	1	1	0	0	0
1	0	1	1	1	0	0	1	1
1	1	0	0	0	0	1	1	0
1	1	0	1	0	0	1	1	1
1	1	1	0	0	1	1	0	0
1	1	1	1	0	0	1	1	1

Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

b) [10%] Dada la tabla de verdad siguiente:

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

Sintetizad de manera mínima a dos niveles la función f mediante el método de Karnaugh. No hace falta que implementéis el circuito.

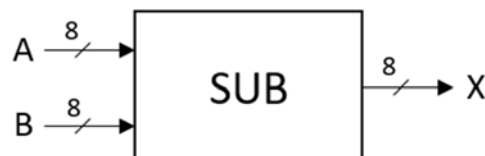
El mapa de Karnaugh de la función f es el siguiente:

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

Y obtenemos esta expresión mínima:

$$f = a'b'c'd' + bc'd + ad + a'bcd' + ab'c$$

c) [15%] Se quiere diseñar un circuito combinacional denominado *SUB* con la estructura siguiente:



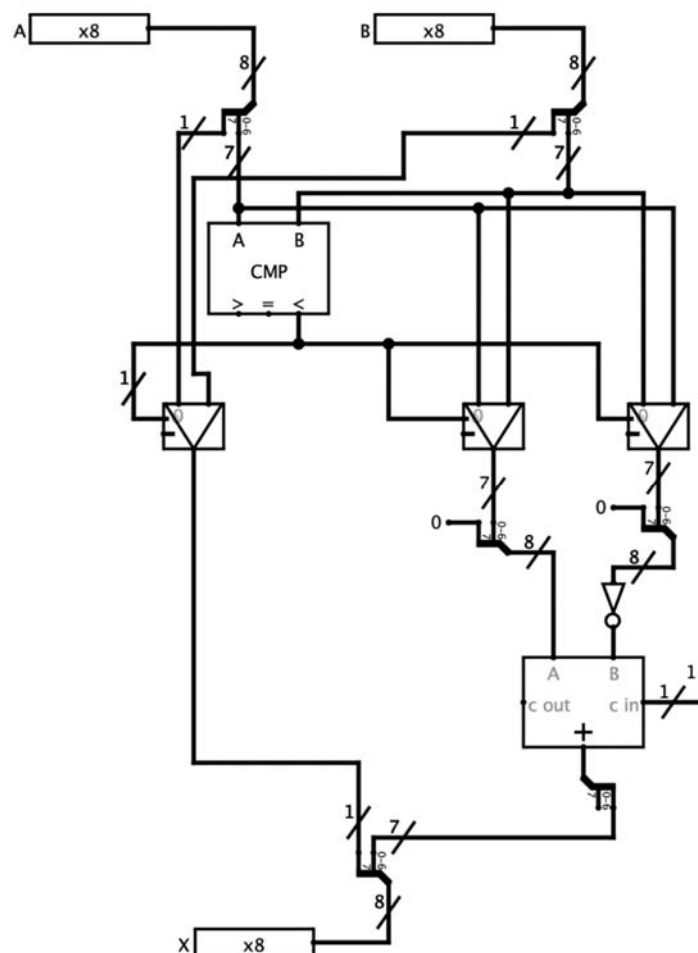
Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

Tanto las entradas A y B como la salida X representan números enteros de 8 bits codificados en SM2. Además, podéis asumir que los números A y B tendrán siempre un valor positivo. La salida X tiene que valer el resultado de restar A menos B , es decir, $X = A - B$.

Se pide que diseñéis el circuito *SUB* usando bloques y puertas combinacionales, y especificando claramente la dimensión de todos los buses utilizados.

La resta de dos números en SM2 positivos se lleva a cabo restando el valor de la magnitud mayor menos el valor de la magnitud menor. El resultado tendrá el mismo signo que la entrada con mayor magnitud. Por eso, el circuito tiene que separar las magnitudes de los dos operandos, y en función del resultado de la comparación de estos dos, hacer $|A| - |B|$ si $|A| > |B|$, o $|B| - |A|$ si $|A| < |B|$. Así que el circuito conectará la salida de un comparador a dos MUXs que seleccionarán B como primer operando y A como segundo si $|A| < |B|$, y al revés en el caso contrario. Estas dos salidas de los MUXs se tienen que restar. Para restar dos números, utilizamos un circuito sumador transformando el segundo operando a $Ca2$. Por eso, tenemos que añadir un bit 0 a los dos operandos (para transformarlos a $Ca2$) y cambiar de signo del segundo operando (complementando el valor de sus bits y sumando el valor 1). El resultado siempre será positivo ya que hemos hecho la magnitud mayor menos la menor, así que podemos descartar el bit más significativo del resultado. Finalmente, hará falta un tercer MUX para seleccionar el valor del signo de la entrada A o de la entrada B en función del resultado de la comparación. El resultado final se obtiene juntando el signo seleccionado con el resultado de la resta de magnitudes.



Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

El circuito se puede describir con las siguientes expresiones algebraicas que usamos para obtener los valores de la tabla:

- $d_1 = xy + q_1$
- $d_0 = q_0'x' + q_0y'$

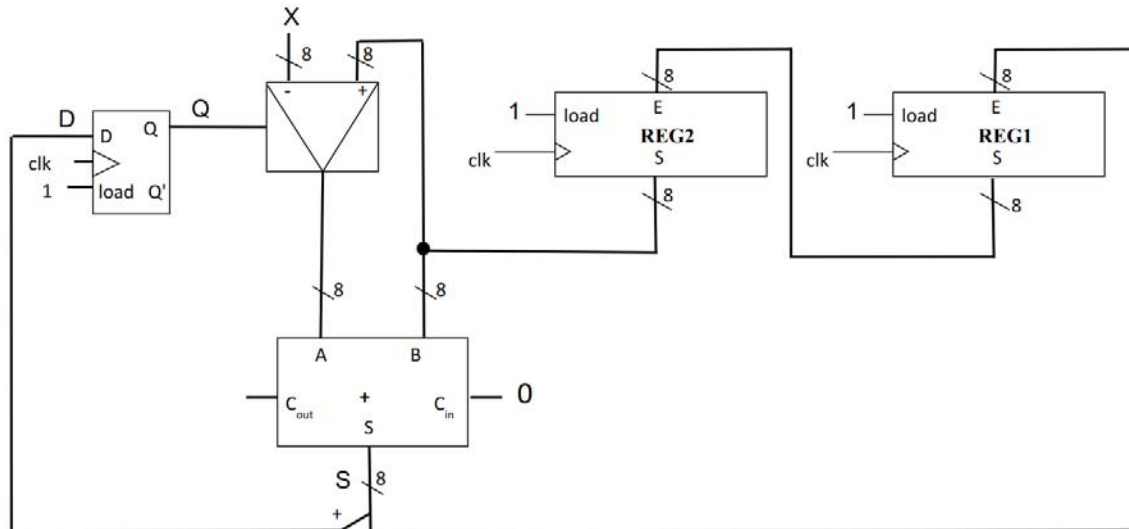
Por lo tanto, podemos rellenar los valores de d_1 , d_0 , z_1 , z_0 a partir de q_1 , q_0 , x e y , teniendo en cuenta que q_1 , q_0 determinan la dirección de la memoria ROM.

q_1	q_0	x	y	d_1	d_0	z_1	z_0
0	0	0	0	0	1	1	1
0	0	0	1	0	1	1	1
0	0	1	0	0	0	1	1
0	0	1	1	1	0	1	1
0	1	0	0	0	1	1	0
0	1	0	1	0	0	1	0
0	1	1	0	0	1	1	0
0	1	1	1	1	0	1	0
1	0	0	0	1	1	0	1
1	0	0	1	1	1	0	1
1	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1
1	1	0	0	1	1	0	0
1	1	0	1	1	0	0	0
1	1	1	0	1	1	0	0
1	1	1	1	1	0	0	0

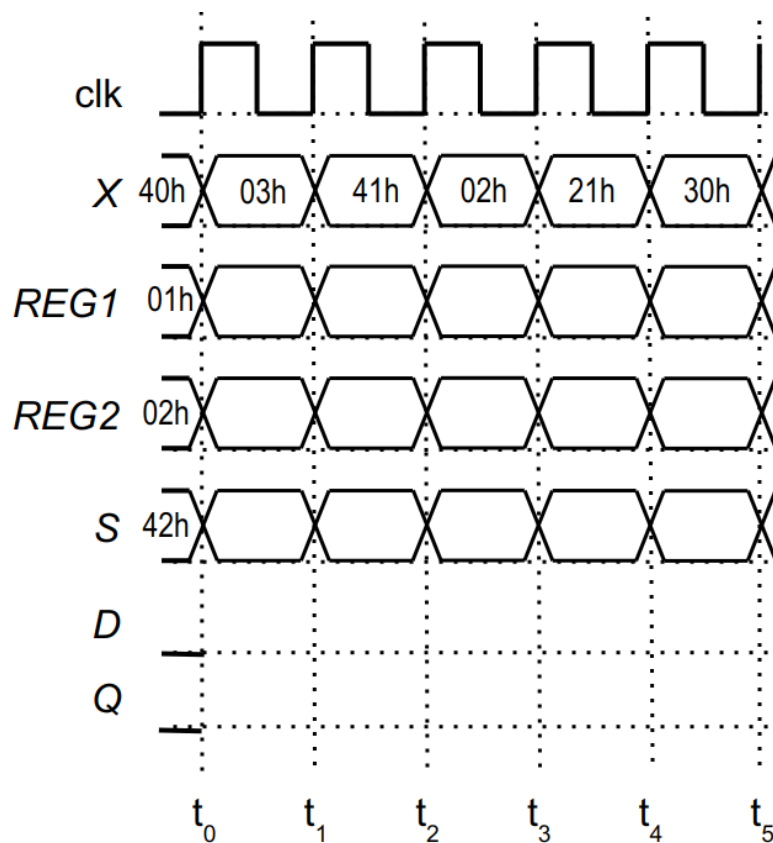
Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

b) [17.5%] Dado el circuito siguiente:



Completad el cronograma siguiente. No hace falta que justificuéis la respuesta.



Examen 2022/23-2

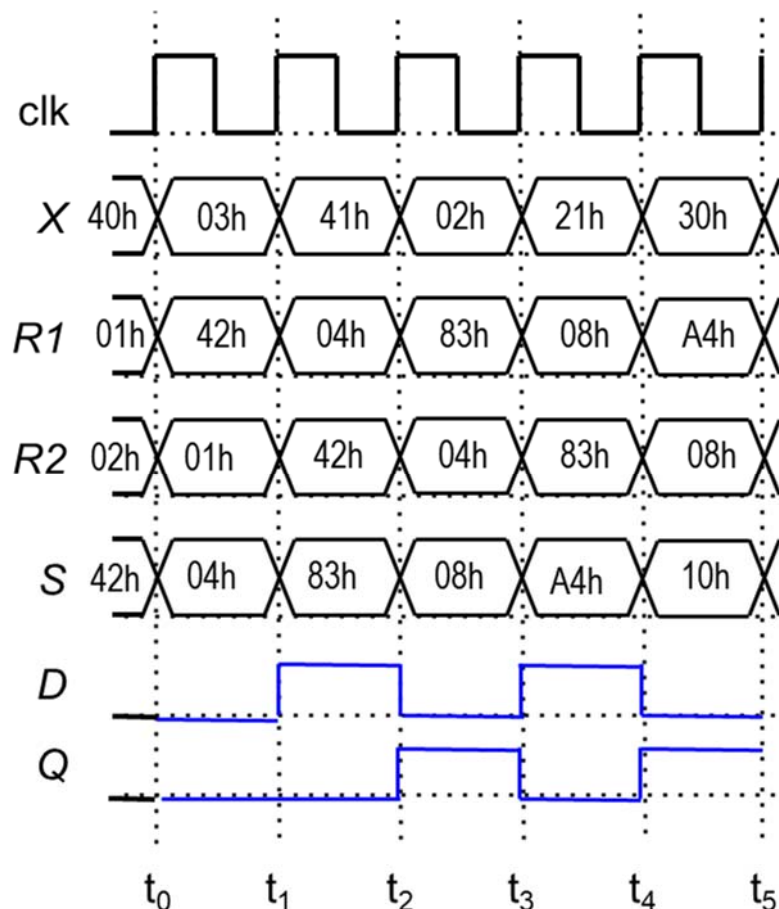
Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

Para rellenar el cronograma miramos en qué momentos se cargan los registros y con qué valores. En cuanto a los momentos, vemos que se cargan en cada flanco ascendente de reloj, ya que en la entrada *load* de los dos registros hay conectado un 1. El mismo razonamiento se aplica al biestable *D*.

En cuanto a los valores que se cargan, vemos que en el registro *REG1* se guarda el valor del resultado de la operación de suma. A su vez, en el registro *REG2* se guarda el valor de *REG1*. Por tanto, podremos escribir sus nuevos valores cuando conozcamos el resultado de la suma.

Para determinar el valor a la salida del sumador debemos considerar el multiplexor y el biestable *D*, del cual depende el valor de la entrada de selección del multiplexor. Observamos que el valor de entrada *D* del biestable es el bit más significativo (bit de más peso) de la suma. A cada flanco activo de la señal de reloj, el biestable propaga el valor de la entrada *D* a la salida *Q*. Si *Q* vale 0, se propaga a la entrada *A* del sumador el valor de *X*, en caso contrario se propaga el valor de *REG2*. A la entrada *B* del sumador siempre se tiene el valor de *REG2*. Por tanto, si la entrada de selección del multiplexor es 0, se realiza la operación $X + \text{REG2}$, en cambio, si es 1 realiza la operación $\text{REG2} + \text{REG2}$.

A partir de este razonamiento rellenamos el cronograma, que queda tal y como se muestra a continuación



Examen 2022/23-2

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	8/7/2023	19:30

PROBLEMA 4 [10%]

- a) **[5%]** ¿Para qué se usan los módulos de acceso directo a memoria (DMA)?
[Para liberar la CPU de las transferencias de información entre periféricos y memoria principal.](#)
- b) **[5%]** ¿Qué tipo de procesador permitiría ejecutar varios programas simultáneamente?
[Uno con varias unidades de procesamiento o cores.](#)