

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

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 2022/23-1


Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

Enunciados

PROBLEMA 1 [20 %]

- a) [2,5 %] Dado el valor decimal $A = -21$, representad A en el formato de signo y magnitud con 9 bits.

Para calcular la representación en formato de signo y magnitud de un número decimal tenemos que aplicar el método de la división entera:

$$\begin{array}{rclcl}
 21 & = & 10 \cdot 2 & + & 1 \\
 10 & = & 5 \cdot 2 & + & 0 \\
 5 & = & 2 \cdot 2 & + & 1 \\
 2 & = & 1 \cdot 2 & + & 0 \\
 1 & = & 0 \cdot 2 & + & 1
 \end{array}$$


Añadimos tres ceros a la izquierda para completar hasta los 8 bits de la magnitud y, finalmente, como el número es negativo añadimos el bit de signo (1) a la izquierda.

$$-21_{(10)} = 100010101_{(SM2)}$$


- b) [2,5 %] Representad el mismo valor A del apartado anterior pero en formato de complemento a 2 con 9 bits.

La codificación de un número negativo X en $Ca2$ es el resultado en binario de la operación $2^n - |X|$ donde $|X|$ es el valor absoluto de X y n el número de bits del formato de representación.

En este caso:

$$2^9 - |-21|_{(10)} = 512_{(10)} - 21_{(10)} = 491_{(10)}$$

Aplicando el método de la división entera:

$$\begin{array}{rclcl}
 491 & = & 245 \cdot 2 & + & 1 \\
 245 & = & 122 \cdot 2 & + & 1 \\
 122 & = & 61 \cdot 2 & + & 0 \\
 61 & = & 30 \cdot 2 & + & 1 \\
 20 & = & 15 \cdot 2 & + & 0 \\
 15 & = & 7 \cdot 2 & + & 1 \\
 7 & = & 3 \cdot 2 & + & 1 \\
 3 & = & 1 \cdot 2 & + & 1 \\
 1 & = & 0 \cdot 2 & + & 1
 \end{array}$$


$$491_{(10)} = 111101011_{(Ca2)}$$

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

- c) **[10 %]** Dados los valores $C = 1011110$ y $D = 0110010$, que representan dos números enteros expresados en signo y magnitud, calculad $C + D$ utilizando el mismo formato. ¿Se produce desbordamiento? Justificad la respuesta.

Para sumar un número negativo y un número positivo representados en signo y magnitud, primero hay que mirar cuál tiene la magnitud mayor, restarle a la magnitud menor y aplicar al resultado el signo del número con la magnitud mayor.

Hacemos la resta de magnitudes:

$$\begin{array}{r}
 C \rightarrow 1|011110 \\
 D \rightarrow 0|110010 \rightarrow \text{mayor magnitud} \\
 \begin{array}{r}
 1\ 1\ 0\ 0\ 1\ 0 \quad \leftarrow D \\
 1\ 1\ 1 \quad \leftarrow \text{acarreo} \\
 -\ 0\ 1\ 1\ 1\ 1\ 0 \quad \leftarrow C \\
 \hline
 0\ 1\ 0\ 1\ 0\ 0
 \end{array}
 \end{array}$$

A continuación, añadimos el signo del número con la magnitud mayor que en este caso es 0.

$$C + D = 0010100_{(SM2)}$$

Nunca se produce desbordamiento en la suma de dos números de distinto signo en formato signo y magnitud.

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

S	Exponente			Mantisa		
14	13	...	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 32.
- Hay bit implícito.
- La mantisa está normalizada en la forma 1,X.

Representad el número $-12,75_{(10)}$ en este formato.

Según el formato dado, analizamos primero el número separando la parte entera y la parte fraccionaria: Pasamos la parte entera a binario, aplicando el método de la división entera:


$$\begin{array}{rclcl}
 12 & = & 6 \cdot 2 & + & 0 \\
 6 & = & 3 \cdot 2 & + & 0 \\
 3 & = & 1 \cdot 2 & + & 1 \\
 1 & = & 0 \cdot 2 & + & 1
 \end{array}
 \quad \uparrow$$

$$12_{(10)} = 1100_{(2)}$$

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

Para la parte fraccionaria aplicamos el método correspondiente:

$$\begin{array}{rclcl} 0,75 & \cdot 2 = & 0,5 & + & 1 \\ 0,5 & \cdot 2 = & 0,0 & + & 1 \end{array}$$


$$0,75(10 = 0,11(2$$

Juntamos la parte entera y fraccionaria: $1100,11(2$

Para normalizar la mantisa hay que mover la coma 3 posiciones hacia la izquierda:

$$1100,11(2 = 1,10011(2 \cdot 2^3$$

Identificamos cada campo:

- Signo: Negativo, $S=1$.
- Exponente: 3. Hay que representarlo en exceso a 32. Por lo tanto, tenemos que sumarle el exceso, $32 + 3 = 35$, que en base 2 es $100011(2$.
- Mantisa: $1,10011(2$. Como tenemos que representar el número con bit implícito eliminamos el 1 de la parte entera. Así pues, la mantisa será 10011 . Como la mantisa tiene disponibles 8 posiciones, añadimos tres ceros por la derecha hasta completar el formato.

El número en el formato solicitado es:

1	100011	10011000
---	--------	----------

PROBLEMA 2 [35 %]

a) [20 %] Dada la tabla de verdad siguiente:

a	b	c	d	f	g
0	0	0	0	X	101
0	0	0	1	1	011
0	0	1	0	1	100
0	0	1	1	X	011
0	1	0	0	1	101
0	1	0	1	X	111
0	1	1	0	X	110
0	1	1	1	0	000
1	0	0	0	0	100
1	0	0	1	1	010
1	0	1	0	X	101
1	0	1	1	0	110
1	1	0	0	X	011
1	1	0	1	0	101
1	1	1	0	1	111
1	1	1	1	X	001

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

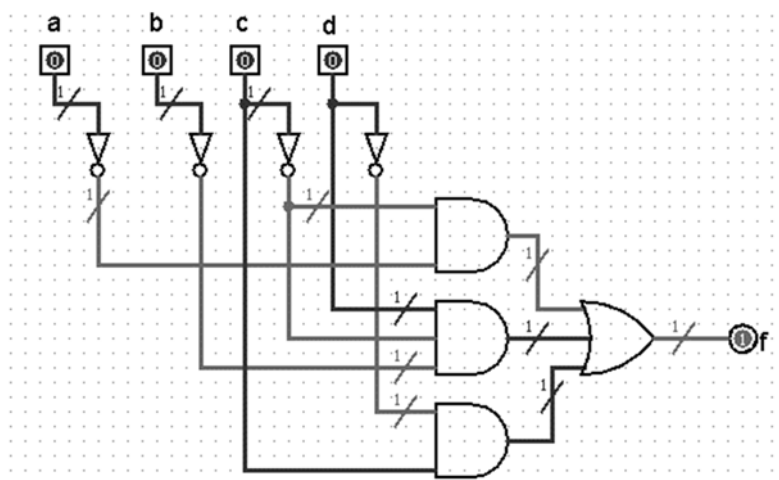
- i. **[10 %]** Sintetizad la función f de manera mínima a dos niveles mediante el método de Karnaugh y dibujad el circuito resultante.

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

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

Y obtenemos esta expresión mínima y el correspondiente circuito:

$$f = c \cdot d' + b'c'd + a' \cdot c$$

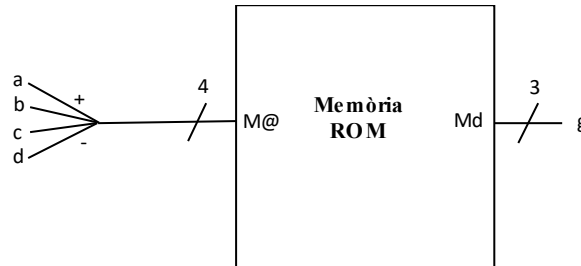


- ii. **[10 %]** Implementad la función g usando una memoria ROM y las puertas lógicas que consideréis necesarias. Indicad el contenido de la ROM y la dimensión de todos los buses del circuito.

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

Conectamos a la entrada de direcciones de la memoria ROM las variables a , b , c , d con el mismo orden de peso que en la tabla de verdad. La memoria ROM tiene una entrada de direcciones de 4 bits i un total de 16 palabras de 3 bits:



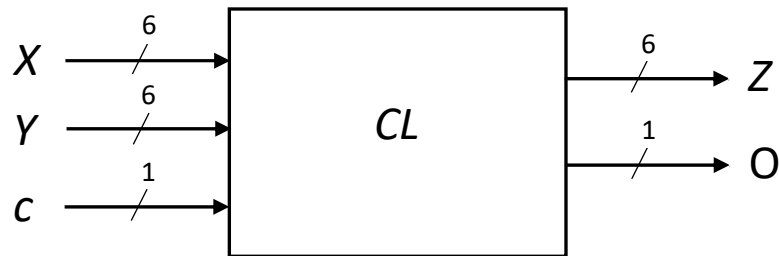
Donde el contenido de la memoria ROM es:

$M@$	$[Md]$
0	101
1	011
2	100
3	011
4	101
5	111
6	110
7	000
8	100
9	010
10	101
11	110
12	011
13	101
14	111
15	001

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

b) [15 %] Diseñad un circuito lógico combinacional, denominado CL, con la estructura siguiente:



El circuito tiene dos entradas X e Y que se corresponden con dos números enteros de 6 bits, codificados **en complemento a 2**, y de una entrada de control c , de un bit. La salida Z , también **en complemento a 2** de 6 bits, se comporta de la manera siguiente:

- Cuando $c = 0$ entonces $Z = X + Y$
- Cuando $c = 1$ entonces $Z = X - Y$

La salida O de un bit, valdrá 1 si se produce desbordamiento al hacer la operación y 0 si no.

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

Se debe realizar una operación de suma o de resta en función de la señal de entrada c . Por lo tanto, en ambos casos se utiliza un bloque sumador donde uno de los operandos siempre es el número X y la otra entrada depende del valor de c .

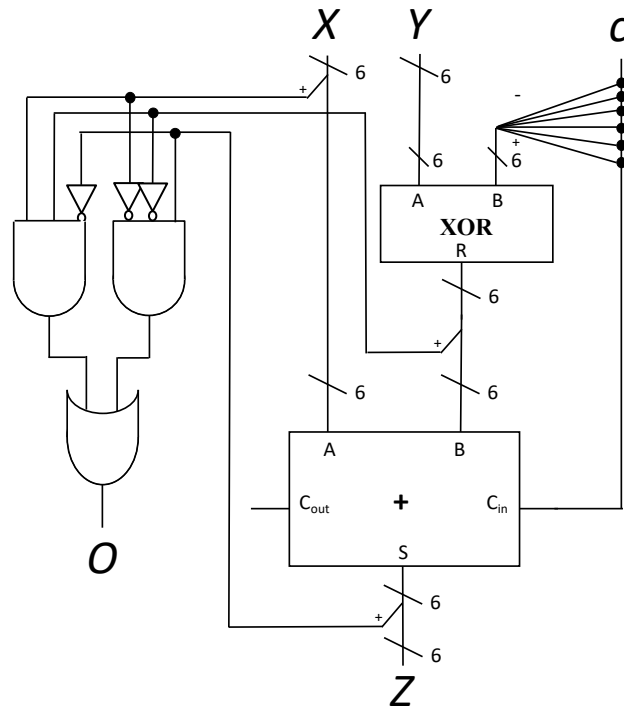
- Si $c = 0$, la otra entrada del bloque sumador debe ser Y .
- Si $c = 1$, hay que hacer la operación de cambio de signo en $Ca2$, que consiste al complementar los bits del número Y y sumar 1. El primer paso se consigue haciendo la operación XOR bit a bit entre el correspondiente bit del número Y y la entrada c . Sumar 1 se consigue haciendo que c sea la entrada C_{in} del bloque sumador. Hay que tener en cuenta que esta implementación no afecta cuando $c = 0$, ya que la operación XOR cuando $c = 0$ hace que la entrada del sumador sea el mismo número Y , y la entrada $C_{in} = 0$.

Notad que existe otra solución donde las operaciones de suma $X + Y$ y resta $X - Y$ se realizan por separado con dos sumadores y podemos obtener el valor de la salida Z con un multiplexor controlado por c .

Para determinar cuando hay desbordamiento hay que fijarse en el bit más significativo (bit de signo) de los operandos (números en las entradas del bloque sumador) y del resultado (Z). El desbordamiento solo se puede producir cuando los operandos son del mismo signo y el resultado de la suma tiene signo opuesto. Por lo tanto, podemos expresar O con esta función lógica $x_5 \cdot r_5 \cdot \bar{z}_5 + \bar{x}_5 \cdot \bar{r}_5 \cdot z_5$, donde $R5$ es el bit más significativo de la salida del bloque XOR.

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

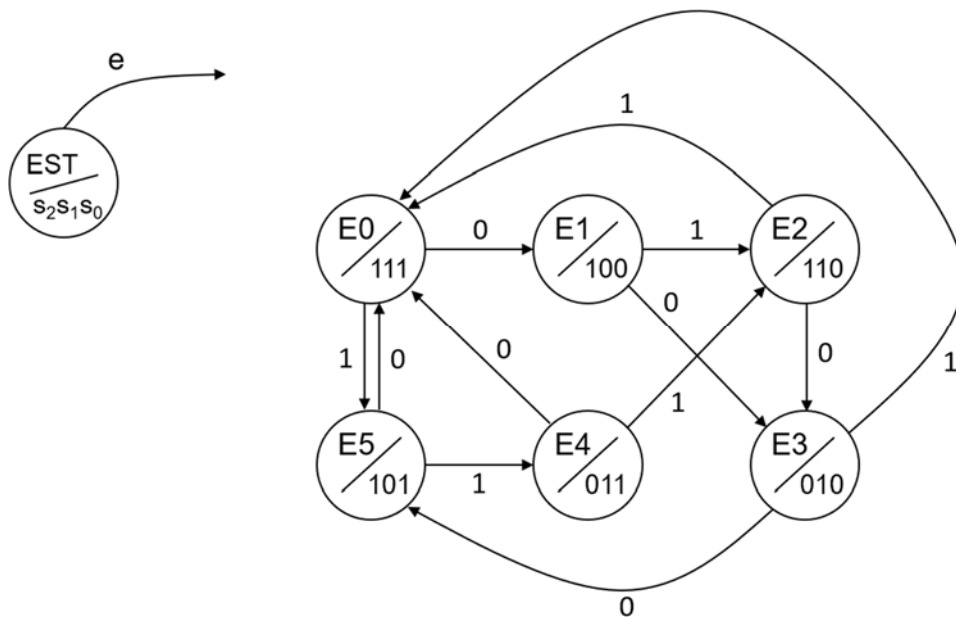


Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

PROBLEMA 3 [35 %]

a) [17,5 %] Dado el grafo de estados siguiente:



Se pide:

- i. [7,5 %] ¿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, ¿cuántos bits de direcciones y cuántos bits de datos necesitará esta memoria?

El número de bits de entrada lo vemos por ejemplo en la leyenda: es **1 bit** (señal *e*).

Igualmente vemos que el circuito tiene **3 bits de salida** (*s*₂, *s*₁ y *s*₀).

El grafo tiene 6 estados, y por tanto harán falta 3 bits para codificarlos (ya que 2 bits sólo pueden tomar 4 combinaciones de valores). Así, el circuito deberá tener al menos **3 biestables**, uno para cada bit de codificación de los estados. Podemos denominar *q*₂, *q*₁ y *q*₀ a sus señales de salida.

Al implementar un circuito secuencial con una memoria ROM, la ROM debe tener una palabra por cada estado y cada posible combinación de variables de entrada. Es decir, una palabra por cada combinación de las señales [*q*₂ *q*₁ *q*₀ *e*]. Por lo tanto, estos son las señales que llegarán a su entrada de direcciones, que será de **4 bits** (y la ROM tendrá $2^4 = 16$ palabras).

Cada palabra tiene que indicar el estado futuro (*q*₂⁺ *q*₁⁺ *q*₀⁺) y el valor de las 3 señales de salida en el estado actual. Por lo tanto, cada palabra tendrá $3 + 3 =$ **6 bits**.

Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

- ii. **[10 %]** Escribid la tabla de salidas y la tabla de transiciones del sistema representado por el grafo, codificando los estados según su índice asociado. Al escribir la tabla, poned en primer lugar las variables que codifican el estado y a continuación las variables de entrada.

Para obtener la tabla de transiciones observamos en el grafo a qué estado se llega desde cada estado con cada valor posible de la variable de entrada. Como los estados E6 y E7 no existen, las filas correspondientes de la tabla son combinaciones *don't care*.

q ₂	q ₁	q ₀	e	q ₁ ⁺	q ₁ ⁺	q ₀ ⁺
0	0	0	0	0	0	1
0	0	0	1	1	0	1
0	0	1	0	0	1	1
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	0	1	0	0	0
0	1	1	0	1	0	1
0	1	1	1	0	0	0
1	0	0	0	0	0	0
1	0	0	1	0	1	0
1	0	1	0	0	0	0
1	0	1	1	1	0	0
1	1	0	0	x	x	x
1	1	0	1	x	x	x
1	1	1	0	x	x	x
1	1	1	1	x	x	x

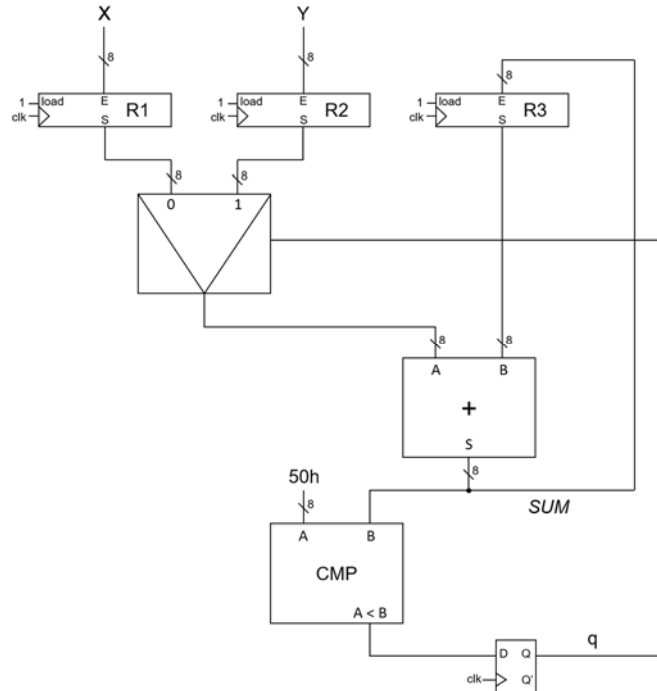
La tabla de salidas expresa cuánto valen [s₂ s₁ s₀] en cada estado. Al igual que en la tabla anterior, las filas correspondientes a los estados E6 y E7 son combinaciones *don't care*.

q ₂	q ₁	q ₀	s ₂	s ₁	s ₀
0	0	0	1	1	1
0	0	1	1	0	0
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	1	1
1	0	1	1	0	1
1	1	0	x	x	x
1	1	1	x	x	x

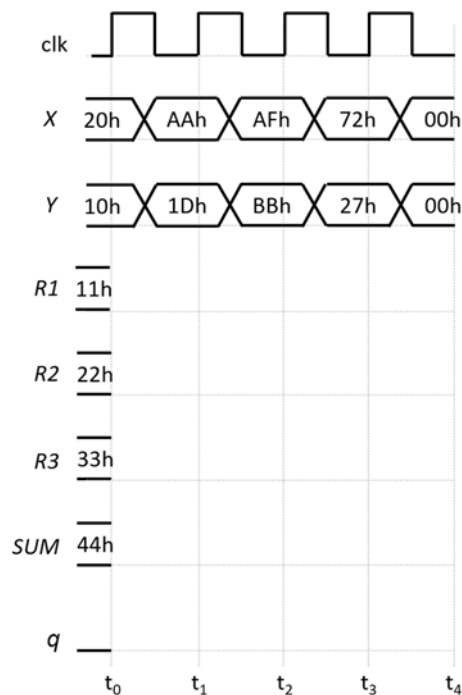
Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

b) [17,5 %] Dado el circuito siguiente:



Completad el cronograma siguiente expresando el valor de los registros *R1*, *R2* y *R3* en hexadecimal, e incluyendo la señal intermedia *SUM*, también en hexadecimal. No hace falta que justifiéis el resultado.



Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

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 a cada flanco de reloj, ya que en la entrada load de los tres registros está conectado un 1. El biestable también se cargará en cada flanco, porque no se ha dibujado su entrada load, lo que indica que siempre le llega 1.

En cuanto a los valores que se cargan, vemos que a la entrada de R1 y de R2 llegan siempre las señales X e Y respectivamente. Por lo tanto, podemos escribir ya de entrada todos sus valores en el cronograma (a cada flanco se escribe el valor de X o de Y en el instante anterior al flanco).

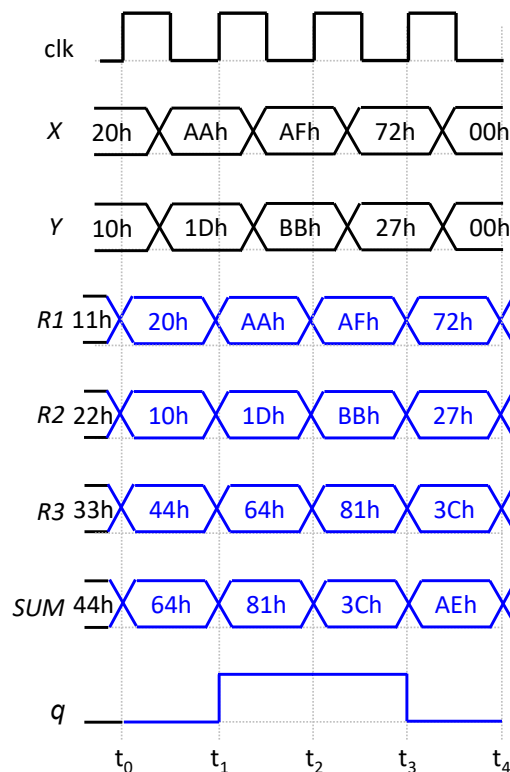
En R3 se carga la señal *SUM*, que es el resultado de la suma de R3 más R1 o R2, en función de cuánto valga *q*: R1 si $q=0$, R2 si $q=1$.

Finalmente vemos que, en cada flanco, si *SUM* es mayor que 50h se cargará un 1 en el biestable (y por tanto *q* pasará a valer 1), de lo contrario se cargará un 0.

Antes del instante t_0 $SUM=44h$, que es menor que 50h. Por lo tanto, en este instante en R3 se cargará 44h y en el biestable un 0; es decir, *q* seguirá valiendo 0, como mínimo hasta t_1 . Por eso a partir de t_0 SUM valdrá $R3+R1 = 44h+20h = 64h$.

Como este valor es mayor que 50h, en el instante t_1 se cargará un 1 en el biestable, y por tanto *q* tomará el valor 1. En R3 se escribirá 64h. Por lo tanto *SUM* pasará a valer $64h+1Dh = 81h$. Puesto que también es mayor que 50h, en t_2 se escribirá un 1 en el biestable (*q* se mantendrá a 1). Y en R3 se cargará el valor de *SUM* antes de t_2 , es decir 81h.

Razonando de forma análoga rellenamos todo el cronograma, que queda tal como se muestra a continuación.



Examen 2022/23-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	28/1/2023	09:30

PROBLEMA 4 [10%]

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

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

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

Conectar dos o más componentes entre ellos.