

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

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%**
 - ¿Se puede consultar material durante la prueba? **No** ¿Qué materiales están permitidos?
Ninguno
 - ¿Puede utilizarse calculadora? **No** ¿De qué tipo? **Ninguno**
 - Si hay preguntas tipo test, ¿descuentan las respuestas erróneas? ¿Cuánto?
 - Indicaciones específicas para la realización de este examen: **Razonad todas las respuestas. Las respuestas sin justificar no serán puntuadas.**
-

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

Enunciados

PROBLEMA 1 [20 %]

- a) **[5 %]** Dados los números $A = 011001$ y $B = 101000$, que representan números binarios enteros codificados en signo y magnitud con 6 bits, calcula $A - B$ usando el mismo formato. Explica si se produce o no desbordamiento y por qué razón.

La operación de restar A y B en signo y magnitud de 6 bits implica el cambio de signo de B y la suma de las magnitudes.

Para aplicar el cambio de signo al valor de B : $-B = 001000_{(SM2)}$

Para realizar la operación pedida al ejercicio realizamos la operación de suma de las magnitudes $A_{(SM2)}$ y $-B_{(SM2)}$

$$\begin{array}{rcccccc}
 & 1 & & & & & \leftarrow \text{acarreo} \\
 & 1 & 1 & 0 & 0 & 1 & \leftarrow \text{magnitud } A_{(SM2)} \\
 + & 0 & 1 & 0 & 0 & 0 & \leftarrow \text{magnitud } B_{(SM2)} \\
 \hline
 1 & 0 & 0 & 0 & 0 & 1 & \leftarrow \text{resultado}
 \end{array}$$

El resultado tiene acarreo y, por tanto, el resultado no se puede expresar con 6 bits con signo y magnitud.

- b) **[5 %]** Dado el número $C = -45$ que representa un número entero, indica su valor en complemento a 2 con 7 bits.

La codificación del número C a un valor en complemento a 2 con 7 bits es el resultado en binario de la operación $2^n - |C|$, donde $|C|$ es el valor absoluto de C . Como que $n = 7$ aplicaremos la operación de conversión en binario de $2^7 - 45_{(10)}$

Realizamos la conversión de los dos números decimales a números base 2 mediante el método de la división entera:

$$\begin{array}{rclcl} 2^7 & = & 2^6 \cdot 2 & + & 0 \\ 2^6 & = & 2^5 \cdot 2 & + & 0 \\ 2^5 & = & 2^4 \cdot 2 & + & 0 \\ 2^4 & = & 2^3 \cdot 2 & + & 0 \\ 2^3 & = & 2^2 \cdot 2 & + & 0 \\ 2^2 & = & 2^1 \cdot 2 & + & 0 \\ 2^1 & = & 2^0 \cdot 2 & + & 0 \\ 2^0 & = & 0 \cdot 2 & + & 1 \end{array}$$

Entonces $2^7 = 10000000_{(2)}$

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

De la misma forma lo haremos con $|-45_{(10)}| = 45_{(10)}$

$$\begin{array}{rcl}
 45 & = & 22 \cdot 2 + 1 \\
 22 & = & 11 \cdot 2 + 0 \\
 11 & = & 5 \cdot 2 + 1 \\
 5 & = & 2 \cdot 2 + 1 \\
 2 & = & 1 \cdot 2 + 0 \\
 1 & = & 0 \cdot 2 + 1
 \end{array}$$

Entonces $|-45_{(10)}| = 101101_{(2)}$

Para hacer la conversión a Complemento a 2 debemos hacer la operación binaria siguiente:

$$\begin{array}{r}
 2^7 - |-45_{(10)}| = 10000000_{(2)} - 101101_{(2)} \\
 \begin{array}{r}
 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \leftarrow 2^7 \\
 1\ 1\ 1\ 1\ 1\ 1\ 1 \leftarrow \text{acarreo} \\
 - \quad 1\ 0\ 1\ 1\ 0\ 1 \leftarrow 45_{(10)} \\
 \hline
 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1 \leftarrow \text{resultado}
 \end{array}
 \end{array}$$

Así pues, la codificación en Ca2 del valor $-45_{(10)}$ es **$1010011_{(Ca2)}$**

También podemos conseguir la codificación en Ca2 de una magnitud negativa mediante un cambio de signo a la codificación de la magnitud positiva $45_{(10)}$.

c) **[10 %]** Dado el formato de coma flotante siguiente:

S	Exponente				Mantisa			
13	12			8	7			0

Donde:

- El bit de signo (posición 13), S, vale 0 para cantidades positivas y 1 para negativas.
- El exponente se representa en exceso a 16, con 5 bits, desde la posición 12 a la 8.
- La mantisa está normalizada en la forma $1,X$. Se representa con bit implícito y con 8 bits, desde la posición 7 a la 0.

¿A qué número decimal corresponde la secuencia de bits 11010011010111 codificado en este formato de coma flotante?

La identificación del formato de la numeración nos indica que el primer bit de la cadena de bits indica el signo. En este caso, el primer bit es un 1 y por tanto, el número es **negativo**.

Respecto la mantisa, vemos que ocupa las 8 posiciones menos significativas de la cadena de bits. Por lo tanto se corresponde con la cadena: 11010111. Ahora bien, el formato indica que hay bit implícito y está normalizada de la forma $1,X$. Esto implica que solo muestra la parte variable de las mantisas normalizadas y se asume la parte fija como conocida y definida en el formato de representación. En este ejercicio el número representado por la mantisa es **$1,11010111_{(2)}$** .

Respecto al exponente, el formato nos indica que está representado por 5 bits y además se nos indica que se ha codificado con exceso a 16. Esto implica que el valor del exponente se conseguirá a partir de restar el número codificado con exceso y este exceso. En este ejercicio el exponente es $10100_{(2)} = 1 \cdot 2^4 +$

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

$0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 20_{(10)}$ sabiendo que hay una codificación con exceso a 16, el exponente será igual a $20_{(10)} - 16 = 4_{(10)}$.

Ahora se puede unir el signo, el exponente y la mantisa para obtener el número representado

$$- 1,11010111_2 \cdot 2^4$$

Haremos un cambio de base para obtener el valor decimal

$$\begin{aligned}
 - 1,11010111_2 \cdot 2^4 &= - (1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 1 \cdot 2^{-4} + 0 \cdot 2^{-5} + 1 \cdot 2^{-6} + 1 \cdot 2^{-7} + 1 \cdot 2^{-8}) \cdot 2^4 \\
 &= - 29,4375_{(10)}
 \end{aligned}$$

PROBLEMA 2 [35%]

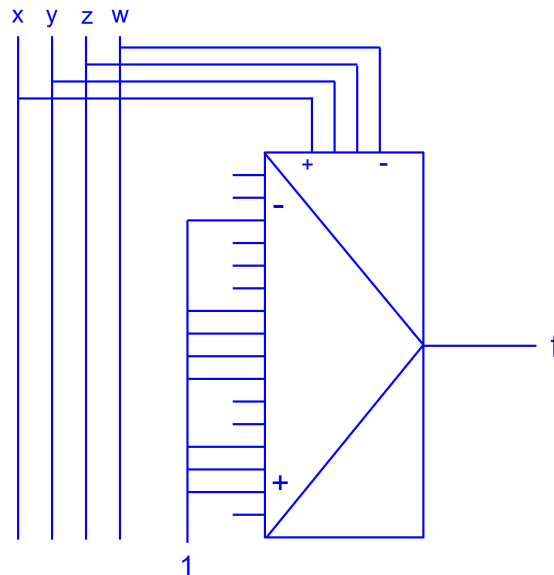
- a) [5%] Implementad la función f utilizando un multiplexor de 4 entradas de control. Podéis utilizar las puertas y bloques que consideréis necesarios.

x	y	z	w	f
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Una función lógica se puede implementar con un multiplexor conectando directamente las variables $xyzw$ en las entradas de control del multiplexor, y conectando el valor 1 en las entradas del multiplexor correspondientes a las posiciones para las cuales la función f vale 1 (en el resto de entradas debe haber un 0). El circuito resultante se muestra a continuación:

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30



- b) **[15%]** Escribid la expresión algebraica mínima a dos niveles de la función g , obteniéndola mediante el método de Karnaugh. No hace falta de diseñéis el circuito mínimo a dos niveles.

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

El mapa de Karnaugh para la función g es el siguiente:

Examen 2023/24-1

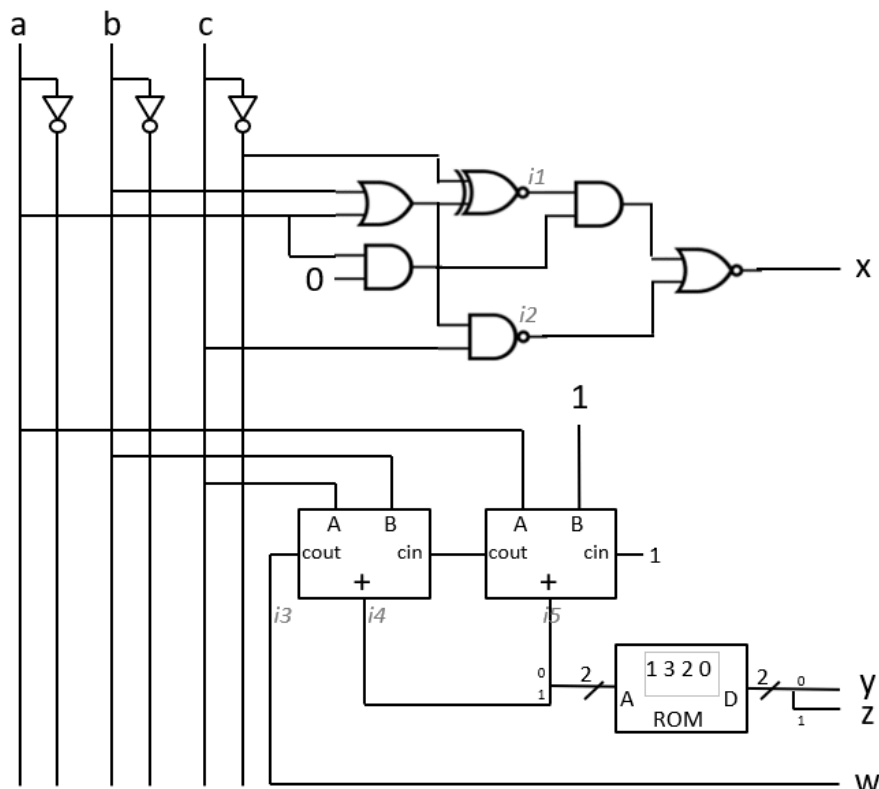
Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

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

Y obtenemos esta expresión mínima:

$$g = b'c'd' + bc'd + a'c$$

c) [15%] Dado el circuito lógico combinacional siguiente:



Se pide que rellenéis la tabla de verdad siguiente, que especifica la salida x , y , z , w en función de las entradas a , b , c . Hay que calcular previamente los valores intermedios indicados en la tabla.

Nota: No hace falta que expliquéis textualmente como obtenéis el valor de cada señal.

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

<i>a</i>	<i>b</i>	<i>c</i>	<i>i</i> ₁	<i>i</i> ₂	<i>i</i> ₃	<i>i</i> ₄	<i>i</i> ₅	<i>x</i>	<i>y</i>	<i>z</i>	<i>w</i>
0	0	0									
0	0	1									
0	1	0									
0	1	1									
1	0	0									
1	0	1									
1	1	0									
1	1	1									

En primer lugar, escribiremos la expresión algebraica correspondiente a cada punto intermedio y la simplificaremos aplicando las leyes de De Morgan siempre que sea posible.

- *i*₁: (*c*' XNOR *a*+*b*). Cuando *a*+*b*=0 valdrá *c* y cuando *a*+*b*=1 valdrá *c*'.
- *i*₂: (*a*+*b* NAND *c*). Cuando *a*+*b*=0 valdrá 1 y cuando *a*+*b*=1 valdrá *c*'.
- *x*: (0 NOR *i*₂). Cuando *i*₂=1 valdrá 0 y cuando *i*₂=0 valdrá 1.
- *i*₅: *a* dado que la entrada *B* del sumador siempre vale 1 y la entrada *cin*=1.
- *i*₄: Dado que *cin*=1 siempre, valdrá el bit menos significativo de la suma aritmética (*b*+*c*+1).
- *i*₃: Es la salida de acarreo de la operación anterior.
- (*z*,*y*) es la salida de la ROM la cual contiene los valores 1,3,2,0 en las direcciones 0,1,2,3. La señal que llega a la entrada de direcciones de la ROM es (*i*₄, *i*₅).
- *w*: *i*₃

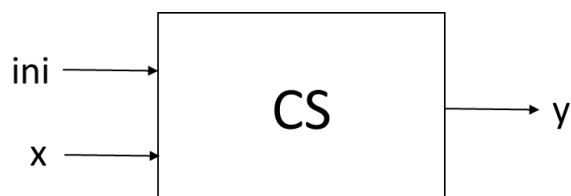
<i>a</i>	<i>b</i>	<i>c</i>	<i>i</i> ₁	<i>i</i> ₂	<i>i</i> ₃	<i>i</i> ₄	<i>i</i> ₅	<i>x</i>	<i>y</i>	<i>z</i>	<i>w</i>
0	0	0	0	1	0	1	0	0	0	1	0
0	0	1	1	1	1	0	0	0	1	0	1
0	1	0	1	1	1	0	0	0	1	0	1
0	1	1	0	0	1	1	0	1	0	1	1
1	0	0	1	1	0	1	1	0	0	0	0
1	0	1	0	0	1	0	1	1	1	1	1
1	1	0	1	1	1	0	1	0	1	1	1
1	1	1	0	0	1	1	1	1	0	0	1

PROBLEMA 3 [35 %]

- a) [15 %] Considerad un circuito secuencial que tiene una señal de entrada de 1 bit (*x*), aparte de la señal de inicialización asíncrona, y una señal de salida (*y*) de 1 bit también.

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30



El circuito debe calcular el opuesto del número entero representado en complemento a 2 que se va introduciendo por la entrada, teniendo en cuenta que el bit de menor peso de este número entero es el primero después de la inicialización y los bits con pesos crecientes van llegando a cada ciclo de reloj. La salida del estado inicial debe ser 0.

Para calcular el opuesto se debe aplicar la operación de cambio de signo en complemento a 2 consistente en dejar todos los bits de menor peso iguales hasta llegar al primer 1. A partir de este bit la salida se debe complementar. La conversión de un nuevo número se realiza reiniciando el circuito mediante la señal asíncrona *ini*.

Ejemplo de funcionamiento, en que en la entrada tenemos el número 10101100 y la salida debe ser 01010100:

Entrada x	0	0	1	1	0	1	0	1	
Salida y	0	0	0	1	0	1	0	1	0

Dibujad el grafo de estados siguiendo el modelo de Moore que realice esta especificación. Hay que especificar textualmente cuál es el significado de cada uno de los estados que forman este grafo.

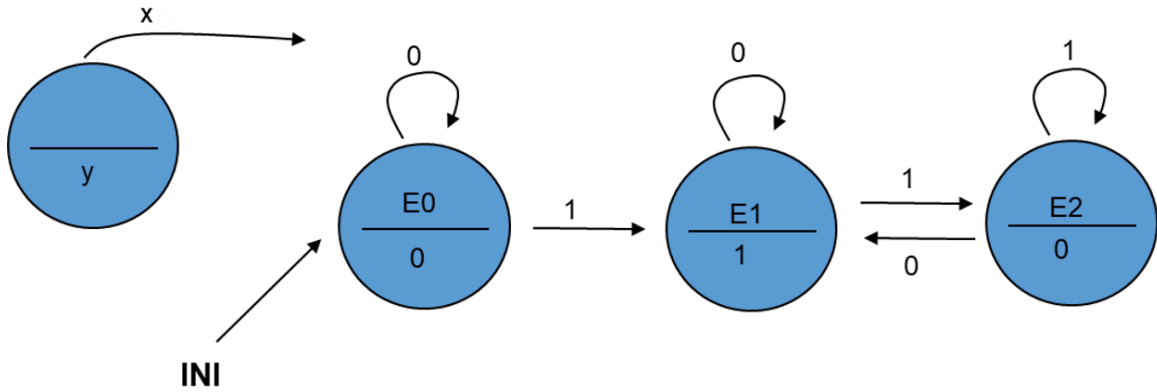
Para conseguir el funcionamiento deseado, el circuito debe tener los siguientes estados:

Estado	Descripción	Salida
<i>E0</i>	Estado inicial, la salida es 0 a la espera de recibir el primer valor. Se mantendrá en este estado si la entrada se mantiene a 0. La salida es 0.	0
<i>E1</i>	La primera vez que entre un 1 entrará en este estado. A partir de entonces, se mantiene o entra en este estado cada vez que entra un 0. La salida es 1.	1
<i>E2</i>	Una vez ha pasado al estado <i>E1</i> , entra en el estado <i>E2</i> cada vez que entra un 1. La salida es 0.	0

El grafo representa el comportamiento deseado:

Examen 2023/24-1

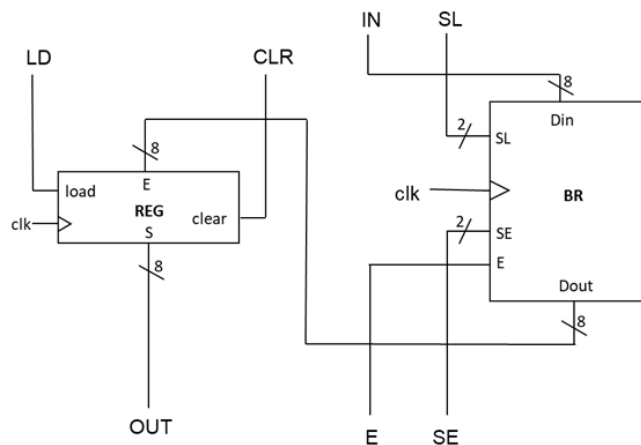
Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30



Examen 2023/24-1

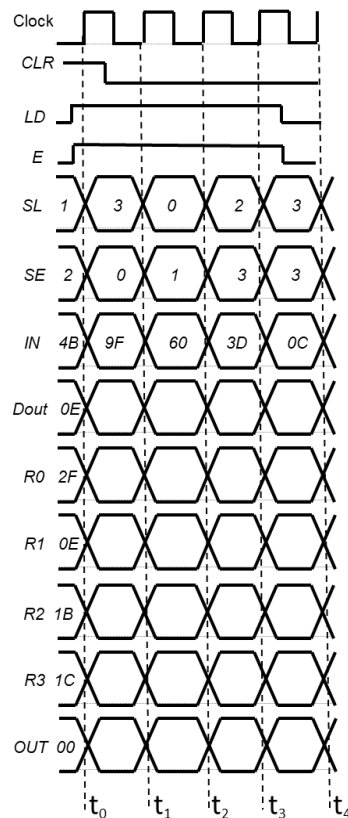
Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

a) [20 %] Considerad el siguiente circuito secuencial.



Se pide que completéis el siguiente cronograma a partir del circuito dado. Tened en cuenta que los 4 registros de BR se denominan R0, R1, R2 y R3, respectivamente.

Indicad textualmente el razonamiento hecho para todos los valores completados del cronograma entre t_0 y t_1 .



Examen 2023/24-1

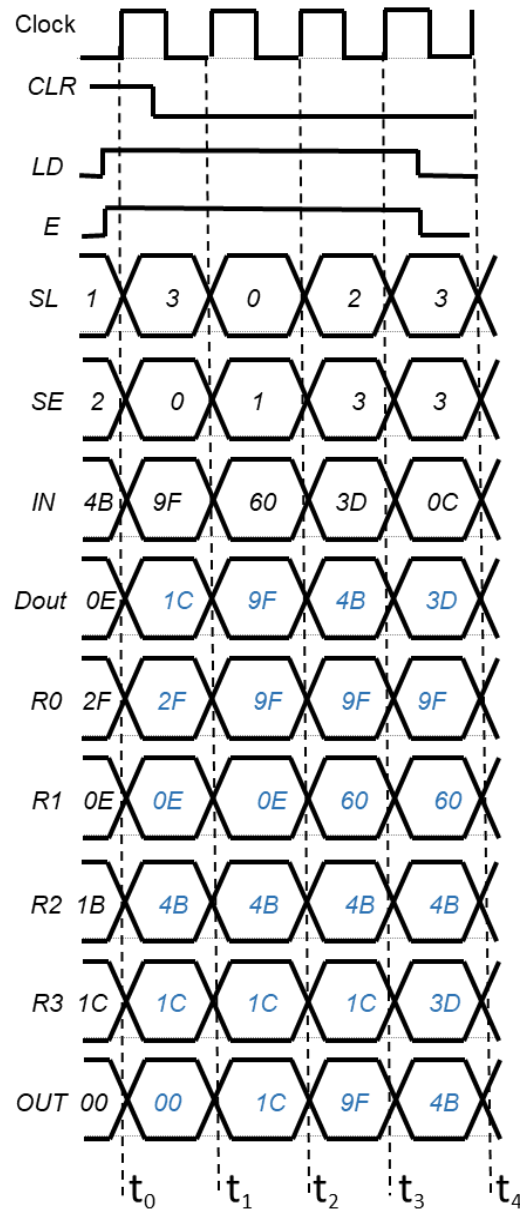
Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

Para rellenar el cronograma tenemos que mirar en qué momentos se carga el registro y el banco de registros y con qué valores. En relación con los momentos, vemos que el registro se carga en cada flanco ascendente de reloj siempre que *LD* sea 1 y *CLR* sea 0. Si *LD* es 0 la salida del registro se mantendrá en el valor que tenga cargado y si *CLR* es 0 la salida del registro será 0x00 de forma asíncrona.

En cuanto al banco de registros BR, observamos que tenemos por un lado la señal *E* que indicará si tenemos habilitada la escritura, y *SE* que nos indicará el registro en el cual se escribe la entrada *IN*. Por otro lado, tenemos *SL* (entrada de selección de registros) que nos indica el valor de qué registro se observará en todo momento en la salida del banco de registros, la cual está conectada a la entrada del registro.

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30



PROBLEMA 4 [10%]

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

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

b) [5%] ¿Qué es una ALU?

Examen 2023/24-1

Asignatura	Código	Fecha	Hora inicio
Fundamentos de computadores	75.562	24/1/2024	12:30

[Un recurso de cálculo programable.](#)