

A

Instrucciones generales para la realización de este examen
La respuesta debe escribirse en el hueco existente a continuación de cada pregunta **con letra clara**.

Cada respuesta correcta suma un punto Cada respuesta incorrecta, ilegible o vacía no suma ni resta. El total de puntos se dividirá entre el total de preguntas y se multiplicará por 10 para obtener la nota del examen.

— El valor 41D90000h representa un nº real en formato IEEE-754, ¿a qué número decimal corresponde?

27,125

En la CPU elemental se han mapeado los siguientes dispositivos:

- Dispositivo RAM: en el rango C000-C7FF.
- Dispositivo ROM: en el rango 0800 – 0BFF.
- Dispositivo PROM: en el rango F110 – F117.

— ¿Cuál es la organización MxN del dispositivo PROM?

8x16

— El dispositivo de memoria ROM está formado por 32 chips organizados en 16 bancos, ¿Cuál es la organización MxN de cada chip?

64x8

— Se desea mapear un nuevo dispositivo de organización 256x16 a partir del dispositivo de memoria ROM. Dibuja su circuito de activación etiquetando correctamente las líneas.

~a15 · ~a14 · ~a13 · ~a12 · a11 · a10 · ~a9 · ~a8

— Se desea añadir al juego de instrucciones de la CPU teórica una nueva instrucción que permita decrementar en una unidad el contenido de un dato en memoria, DEC [Ri]. Completar el cuadro siguiente con las señales necesarias para que se ejecute, en el menor número de ciclos de reloj posible, la instrucción DEC [R0]

Pa so	Señales de Control
4	RO-IB, IB-MAR, LEER
5	(ciclo de espera)
6	MDR-IB, TMPE-SET, ADD, ALU-TMPS, ALU-SR
7	TMPS-IB, IB-MDR, ESCRIBIR
8	FIN

— Se sabe que el código que asigna ISO-Latin1 al carácter ‘a’ es 61h, al carácter ‘p’ el 112 (decimal) y al carácter ‘á’ el E1h. Codifica la cadena “papá” en UTF-8. Responder en hexadecimal.

706170C3A1h

— Determina cuál o cuáles de las siguientes afirmaciones son CIERTAS. En caso de que ninguna sea cierta, contesta “ninguna”.

- A) Cuando la CPU elemental recibe una interrupción, tarda como mínimo tres pasos de ejecución en atenderla.
- B) Siempre que el interfaz de un periférico activa la señal INT, después la CPU activa la señal INTA.
- C) Durante la ejecución de una instrucción IRET el registro R7 se incrementa en 2.
- D) Durante el proceso de atención a una interrupción, la CPU recibe un número de vector de interrupción a través del registro MDR

C, D

— En una ALU de 4 bits se suman el mayor positivo codificable en signo-magnitud y el mayor positivo codificable en exceso a Z central. Interpretar el resultado de la suma en binario natural e indicarlo en decimal. Indicar también el valor de los bits de carry y overflow.

Suma: 6 C: 1 O: 0



Determinar cuáles de las siguientes afirmaciones son CIERTAS. Responder TODAS o NINGUNA si se considera que es la respuesta adecuada.

- A) La instrucción JMP Inm8 modifica los flags del registro de estado.
- B) La instrucción que más rápido se ejecuta en la CPU elemental tarda 3 ciclos de reloj.
- C) La señal de control SUB se activa en la instrucción NEG.
- D) Una palabra de control contiene tantos bits como instrucciones distintas sea capaz de ejecutar la CPU.

C

La CPU elemental está ejecutando el siguiente fragmento de código:

```
Leer: MOV R1, [R0]
      ADD R3, R1, R1
      INC R0
      DEC R5
      BRNZ Leer
      MOVL R2, 3
```

En un instante dado, la CPU se halla ejecutando una de las instrucciones anteriores, y en ese instante los valores de sus registros son:

R0	7030h	R4	6576h
R1	0034h	R5	0100h
R2	0000h	R6	1FFAh
R3	0000h	R7	7422h
PC	7141h	IR	4324h

- Si en el instante aludido el valor de registro TMPS es 0068h ¿qué ciclo de ejecución acaba de finalizar?

El ciclo 5

Se ha desarrollado el siguiente programa para la CPU teórica. Este programa utiliza tres periféricos un teclado y dos pantallas (. El código fuente está compuesto de las siguientes partes: un **programa principal**, una **rutina de interrupción** correspondiente al periférico teclado y un **procedimiento llamado borra_pantalla**

El **programa principal** lleva a cabo las siguientes acciones:

1. Instala la rutina de interrupción en memoria.
2. Comienza escribiendo el carácter indicado en la variable “caracter” en la pantalla_0 o pantalla_1 dependiendo del valor de la variable “pantalla”; comenzando en la posición indicada por la variable “pos0” si corresponde escribir en la pantalla_0 o “pos1” si corresponde escribir en la pantalla_1.

Por lo tanto “pos0” y “pos1” reflejan la última posición de la pantalla donde se ha escrito el carácter en cada pantalla respectivamente.

La **rutina de interrupción** de teclado se genera cada vez que se pulsa una tecla. Si la tecla pulsada es una letra MINUSCULA se cambia el valor de la variable “caracter” con el ascii correspondiente a dicha letra en MAYUSCULA. Si la tecla pulsada es el espacio se cambia de pantalla_0 a pantalla_1 ó de pantalla_1 a pantalla_0.

Para el resto de teclas no se realiza ninguna función.



```
ORIGEN 2600h
INICIO main
.PILA ¿?
.DATOS
  caracter VALOR 'A'
  pantalla VALOR 0
  pos0 VALOR 0F900h
  pos1 VALOR 0F980h

.CODIGO

PROCEDIMIENTO rut_teclado
  PUSH R0
  PUSH R1
  PUSH R2
  PUSH R3
  PUSH R4

  ; R0 = Registro de datos del teclado
  MOVH R0, 0FAh
  MOVL R0, 0h
  MOV R0, [R0]
  MOVH R0, 0

  ; R1 = Ascii tecla espacio
  ---- HUECO 1 ----
  BRNZ no_espacio
  MOVH R2, BYTEALTO DIRECCION pantalla
  MOVL R2, BYTEBAJO DIRECCION pantalla
  MOV R3, [R2]
  ; R4 = Mascara
  MOVH R4, 0h
  MOVL R4, 1h
  ---- HUECO 2 ----
  MOV [R2], R3
  JMP final
no_espacio:
  MOVH R1, 0
  MOVL R1, 'a'
  COMP R0, R1
  ---- HUECO 3 ----
  MOVL R1, 'z'
  COMP R0, R1
  BRNC compruebaZF
  JMP es_letra
compruebaZF:
  BRNZ final
es_letra:
  ; R4 = Mascara
  ---- HUECO 4 ----
  MOVH R2, BYTEALTO DIRECCION caracter
```

<pre> MOVL R2, BYTEBAJO DIRECCION caracter MOV [R2], R0 final: POP R4 POP R3 POP R2 ---- HUECO 5 ---- FINP main: ; Instalar la rutina de interrupcion MOVH R0, 0 MOVL R0, 50 ---- HUECO 6 ---- MOVH R0, BYTEALTO DIRECCION pantalla MOVL R0, BYTEBAJO DIRECCION pantalla bucle: CLI ; R1 = Auxiliar MOV R1, [R0] ; R2 = Direccion de memoria de la ; variable pos0 o pos1 MOVH R2, BYTEALTO DIRECCION pos0 MOVL R2, BYTEBAJO DIRECCION pos0 ADD R2, R2, R1 ; R3 = Direccion de la pantalla ; en la que se ha escrito MOV R3, [R2] ; Borrar caracter anterior MOVH R4, 7h MOVL R4, ' ' MOV [R3], R4 INC R3 ; Comprobar final Pantalla 0 MOVH R1, 0F9h MOVL R1, 78h COMP R1, R3 BRNZ comprobar_Pantalla_1 MOVH R3, 0F9h MOVL R3, 0h comprobar_Pantalla_1: ; Comprobar final Pantalla 0 MOVH R1, 0F9h MOVL R1, 0F8h </pre>	<pre> COMP R1, R3 BRNZ escribe MOVH R3, 0F9h MOVL R3, 80h escribe: MOVH R5, BYTEALTO DIRECCION caracter MOVL R5, BYTEBAJO DIRECCION caracter MOV R4, [R5] MOVH R4, 7h ---- HUECO 7 ---- STI JMP bucle FIN </pre>	<div> <div> – ¿Qué instrucción o instrucciones falta/n en el --- Hueco 4 --? </div> <div> <div>MOVH R4, 0FFh</div> <div>MOVL R4, 0DFh</div> <div>AND R0, R0, R4</div> </div> </div>
	<div> <div> – ¿Qué instrucción o instrucciones falta/n en el --- Hueco 1 --? </div> <div> <div>MOVH R1, 0</div> <div>MOVL R1, ‘ ‘</div> <div>COMP R0, R1</div> </div> </div>	<div> <div> – ¿Qué instrucción o instrucciones falta/n en el --- Hueco 5 --? </div> <div> <div>POP R1</div> <div>POP R0</div> <div>IRET</div> </div> </div>
	<div> <div> – ¿Qué instrucción o instrucciones falta/n en el --- Hueco 2 --? </div> <div> <div>XOR R3, R3, R4</div> </div> </div>	<div> <div> – Qué instrucción o instrucciones falta/n en el --- Hueco 6 --? </div> <div> <div>MOVH R1, BYTEALTO DIRECCION rut_teclado</div> <div>MOVL R1, BYTEBAJO DIRECCION rut_teclado</div> <div>MOV [R0], R1</div> </div> </div>
	<div> <div> – ¿Qué instrucción o instrucciones falta/n en el --- Hueco 3 --? </div> <div> <div>BRC final</div> </div> </div>	<div> <div> – ¿Cuál es el tamaño óptimo de la pila? Contestar en decimal </div> <div> <div>7</div> </div> </div>



- ¿Cuál es la dirección de memoria que le corresponde a la esquina inferior derecha de la pantalla_1 utilizada en el programa?
Contestar en **hexadecimal**

F9F7h

- ¿Cuál es el número de vector de interrupción correspondiente al teclado? Contestar en **hexadecimal**

32h

- ¿Qué instrucción o instrucciones falta/n en el
--- **Hueco 7** --?

MOV [R3], R4

MOV [R2], R3

Sin signo	Relación	Con signo
ZF = 1	=	ZF = 1
ZF = 0	≠	ZF = 0
CF=0 AND ZF=0	>	SF=OF AND ZF=0
CF = 0	≥	SF=OF
CF = 1	<	SF ≠ OF
CF=1 OR ZF =1	≤	SF≠OF OR ZF=1