

# SOLUCIONES AL FINAL DEL EXAMEN.

## INGENIERÍA TÉCNICA en INFORMÁTICA de SISTEMAS

ASIGNATURA: ESTRUCTURA Y TECNOLOGÍA DE COMPUTADORES I (PLAN NUEVO)

CURSO: 2002/2003 FECHA: 15 de Febrero de 2003 HORA: 9:00 DURACIÓN TOTAL: 2 horas

CÓDIGO CARRERA: CÓDIGO ASIGNATURA: CONVOCATORIA: Febrero – 1ª PP TIPO EXAMEN: E

53

1041

EXAMEN: Reserva

APELLIDOS: .....	NOMBRE: .....	DNI: .....
CENTRO DONDE SE MATRICULÓ: .....	CENTRO DE EXAMEN: .....	
Firma: .....		

!!! Es necesario ENTREGAR ESTA HOJA DE ENUNCIADOS JUNTO CON UNA HOJA DE LECTURA ÓPTICA donde deberá marcar sus respuestas. Ambas hojas deberán estar debidamente cumplimentadas y firmadas !!!

### MATERIAL PERMITIDO DURANTE LA REALIZACIÓN DEL EXAMEN:

- Calculadora no programable
- Ningún otro material: Ni libros, ni ADDENDA, ni fotocopias.

### PRIMERA PARTE: Preguntas tipo TEST de TEORÍA (puntuación máxima: 4 puntos).

Este test es ELIMINATORIO. Mínimo necesario para aprobarlo: 6 aciertos.

La solución a este test se marcará en el espacio RESPUESTAS de la hoja de lectura óptica, cada pregunta en su número respectivo.

Sólo hay una respuesta correcta para cada pregunta.

Cada respuesta correcta vale 0.4 puntos. LAS RESPUESTAS EQUIVOCADAS O EN BLANCO NO PENALIZAN.

#### 1. Señale cual de las siguientes afirmaciones sobre el lenguaje máquina es cierta:

- El lenguaje máquina está ligado al diseño interno del computador, esto es, está relacionado con el diseño a nivel de circuitos electrónicos.
- Los programas escritos en lenguaje máquina sólo son transportables a otro computador siempre y cuando se cumpla la condición de que el procesador sea el mismo.
- Aunque el tamaño del campo del código de operación es fijo, el nº de operandos de una instrucción dada puede ser variable.
- Los modos de direccionamiento que utilizan las instrucciones en código máquina se codifican junto con la operación que realiza la instrucción en el código de operación.

#### 2. La distancia entre la combinación binaria 10011001 y la 10101101 es:

- 8.
- 11101100.
- 3.
- 3.

#### 3. ¿A que denominamos Registro?

- Al lugar donde se almacenan los datos desechables de un sistema informático.
- A un conjunto de biestables que funcionan simultáneamente.
- A un conjunto de datos que hacen funcionar el equipo.
- A la lista de los periféricos disponibles en el computador.

#### 4. Indicar cual de las siguientes letras no designa a alguno de los bits que forman parte del registro de código de condición:

- C
- Y
- Z
- V

#### 5. ¿Qué es un biestable?

- Un elemento de almacenamiento muy grande.
- Un elemento constituido por dos registros.
- Un elemento de software para modificar la información de la memoria.
- Un elemento capaz de almacenar un bit de información.

#### 6. Determinar el nº mínimo de palabras que podría tener una instrucción que especifica 2 operandos en un computador cuya memoria es de 64 kbytes y está organizada en bytes, si el registro de instrucción es de 8 bits y emplea para ambos operandos un direccionamiento directo absoluto:

- 3
- 4
- 5
- 1

#### 7. La representación de un número entero positivo utilizando n cifras:

- Coincide en Módulo y signo y Complemento a 2, pero no en Complemento a 1.
- Coincide en Módulo y signo y en Complemento a 1 pero no en Complemento a 2.
- Coincide en Complemento a 1, Complemento a 2 y en Módulo y signo.
- Coincide en Complemento a 1 y Complemento a 2 pero no en Módulo y signo.

#### 8. ¿Qué significado tienen las siglas BCD?

- Es un código según el cual cada cifra decimal se representa por su valor en el sistema de numeración binario.
- Es un código detector de errores
- Es un código según el cual cada cifra binaria se representa por su valor en el sistema de numeración decimal.
- No significa nada en concreto.

#### 9. Cual de los siguientes no es un procedimiento básico para provocar una operación de entrada salida:

- E/S controlada por programa
- E/S controlada por interrupción
- E/S por acceso directo a memoria
- E/S por controlador de interfase

#### 10. Para construir un código de Hamming válido para ser utilizado con datos de 11 bits es preciso añadir:

- 5 bits de paridad.
- 4 bits de paridad.
- 3 bits de paridad.
- 2 bits de paridad.

**ESTE EXAMEN CONSTA DE DOS PARTES DE TIPO TEST: TEORÍA Y EJERCICIOS PRÁCTICOS  
 EL TEST ELIMINATORIO DE TEORÍA FIGURA EN EL REVERSO DE ESTA HOJA**

**SEGUNDA PARTE: Preguntas tipo TEST de EJERCICIOS PRÁCTICOS (puntuación máxima: 6 puntos).**

Este test se corregirá sólo si se ha superado el test de TEORÍA.

La solución a este test se marcará en el espacio RESPUESTAS de la hoja de lectura óptica, cada pregunta en su número respectivo.

Sólo hay una respuesta correcta para cada pregunta.

Cada respuesta correcta vale 1 punto. LAS RESPUESTAS EQUIVOCADAS O EN BLANCO NO PENALIZAN.

**11. Sea la función lógica de tres variables  $f = m_3 + m_5 + m_7$**

**La segunda forma canónica de esta función es:**

- a)  $M_1 \cdot M_3 \cdot M_5 \cdot M_6 \cdot M_7$
- b)  $M_3 \cdot M_5 \cdot M_7$
- c)  $M_0 \cdot M_1 \cdot M_2 \cdot M_4 \cdot M_6$
- d)  $M_0 \cdot M_2 \cdot M_4$

**12. Obtener la representación binaria del número decimal  $7,424 \cdot 10^3$  en formato normalizado IEEE 754 para coma flotante de 32 bits:**

- a) \$686E0000
- b) \$86E80000
- c) \$45E80000
- d) \$456E0000

**13. Simplificar la siguiente expresión utilizando los teoremas del álgebra de Boole:**

$$\overline{(B + C + D) \cdot (B + C + \overline{D}) \cdot (A \cdot \overline{B} + C \cdot \overline{D})}$$

(las negaciones en las respuestas se representan mediante “ ’ ”)

- a)  $D' + (A' + B)C$
- b)  $C' + (A' + B)D$
- c)  $D' + (A + B')C$
- d) Ninguna de las anteriores

**14. Obtener el equivalente decimal del número \$49FC0000 suponiendo que se utiliza el formato normalizado IEEE 754 para coma flotante de 32 bits:**

- a)  $1,015808 \cdot 10^6$
- b)  $-1,015808 \cdot 10^5$
- c)  $2,064384 \cdot 10^6$
- d)  $4,063232 \cdot 10^5$

**15. Después de ejecutarse el siguiente segmento de código del M68000 cuál es el contenido correcto de las siguientes posiciones de memoria:**

- a) (\$6006) = \$AB
- b) (\$6006) = \$FD
- c) (\$6007) = \$EF
- d) (\$6007) = \$CD

	<i>ORG</i>	<i>\$6000</i>
<i>DAT</i>	<i>DC.W</i>	<i>\$1234,\$ABCD</i>
	<i>DC.W</i>	<i>\$5678,\$90EF</i>
<i>COD</i>	<i>CLR.L</i>	<i>D1</i>
	<i>MOVEA.L</i>	<i>#DAT,A2</i>
	<i>MOVE.L</i>	<i>(A2)+,D0</i>
	<i>MOVE.W</i>	<i>(A2)+,D1</i>
	<i>EOR.W</i>	<i>D0,D1</i>
	<i>MOVE.W</i>	<i>D0,(A2)+</i>
	<i>MOVE.L</i>	<i>D1,-(A2)</i>

**16. Señalar cuál es el contenido del registro D2, siendo inicialmente D0: \$0000 000F, D1: \$0000 0010 y D2: \$1357 AF86 después de ejecutar las siguientes instrucciones:**

- a) \$0000 F0F0
- b) \$0000 F0D0
- c) \$0005 F0F0
- d) \$0005 F0D0

<i>ROR.W</i>	<i>D0, D2</i>
<i>MULU</i>	<i>D1, D2</i>
<i>ANDI.B</i>	<i>#\$F0, D2</i>

# SOLUCIONES NO OFICIALES

Por Jose Antonio Vaqué

1. **A** (página 346)
2. **C** (página 86). Distancia es el número total de bits que hay que cambiar para igualar los códigos.
3. **B** (página 240) aunque no está así en el libro.
4. **B** (página 435)
5. **D** (página 244)
6. **C**. El direccionamiento directo requiere tener la dirección del elemento en la propia instrucción, por lo que al se la memoria de 16 bits, necesitamos 2 octetos para cada dirección, mas el octeto para el código de instrucción, total 5 octetos.
7. **C** (a partir de la página 55)
8. **A** (página 88)
9. **D** (página 270)
10. **B**. (página 106). Si consideramos la ecuación  $2^k \geq n + k + 1$ , con  $n = 11$  probamos  $k$  a partir del 1, resultando los valores ( $k=1$ )  $2 \geq 13$  [NO], ( $k=2$ )  $4 \geq 14$  [NO], ( $k=3$ )  $8 \geq 12$  [NO], ( $k=4$ )  $16 \geq 16$  [SI]
11. **A** (página 127). Si montamos la tabla es rápido, sencillo y seguro, tachamos los que nos dan, y los otros son el resultado:

m0	M7
m1	M6
m2	M5
<del>m3</del>	<del>M4</del>
m4	M3
<del>m5</del>	<del>M2</del>
m6	M1
<del>m7</del>	<del>M0</del>

Resultando M1.M3.M5.M6.M7

12. **C**. Si pasamos a binario  $7.424 \times 10^3 = 111010000000 =$  normalizando  $= 1.1101 \times 2^{12}$   
Montamos el numero a partir de **s=0**, **e=12+127=139=10001011**, **m=(1.)1101**:  
**01000101.11101000.00000000.00000000 = 45.E8.00.00**
13. **B**. Aplicamos las reglas, pero para no liarnos, creamos unas variables auxiliares:  
 $x1 = B+C+D \rightarrow x1' = B'C'D'$   
 $x2 = B+C+D' \rightarrow x1' = B'C'D$   
 $y1 = A.B' \rightarrow y1' = A'+B$   
 $y2 = C.D' \rightarrow y2' = C'+D$   
 $x3 = y1+y2 \rightarrow x3' = y1'y2' = (A'+B)(C'+D) = A'C'+A'D+BC'+BD$   
 $(x1.x2.x3)' = x1' + x2' + x3' = B'C'D' + B'C'D + A'C'+A'D+BC'+BD$   
 $\rightarrow$  Por absorción de estos términos  $= B'C' + A'C'+A'D+BC'+BD$   
 $\rightarrow$  Por absorción de estos términos  $= C' + A'C'+A'D+BD$   
 $\rightarrow$  Por absorción de estos términos  $= C' + A'D+BD$   
 $\rightarrow$  Sacamos factor común  $= C' + (A'+B).D$
14. **C**. Si pasamos a binario  $49FC0000 = 01001001.11111000.00000000.00000000$   
Montamos el numero a partir de **s=0**, **e=10010011-127=147-127=20**, **m=(1.) 1111=1.96875**  
 $n=2^{20} \times 1.96875 = 2064387 = 2.064384 \times 10^6$

15. B. Solo hay que seguir el programa, que es laborioso pero nada difícil:

Paso	OPERACIÓN	D0	D1	A2	Comentarios
1	DAT DC.W \$1234,\$ABCD DC.W \$5678,\$9EF				Guardamos en la memoria
2	COD CLR.L D1		00.00.00.00		Limpiamos D1
3	MOVEA.L #DAT,A2		00.00.00.00	6000	Ponemos la dirección en A2
4	MOVE.L (A2)+,D0	12.34.AB.CD	00.00.00.00	6004	Cargamos D0, y movemos luego A3 (L=4 bytes)
5	MOVE.W (A2)+,D1	12.34.AB.CD	00.00.56.78	6006	Cargamos D1 (solo W), y movemos A2 (W=2 bytes)
6	EOR.W D0,D1	12.34.AB.CD	00.00FD.B5	6006	Or exclusivo entre los bytes bajos de D0 y D1, en D1.
7	MOVE.W D0,(A2)+	12.34.AB.CD	00.00FD.B5	6008	Pasamos a memoria los bytes bajos de D0, moviendo luego A2.
8	MOVE.L D1,-(A2)	12.34.AB.CD	00.00FD.B5	6004	Movemos A2 (L = 4 bytes), y luego copiamos a memoria todo D1

POSICION	6000	6001	6002	6003	6004	6005	6006	6007
Tras el paso 1	12	34	AB	CD	56	78	90	EF
Tras el paso 7	12	34	AB	CD	56	78	AB	CD
Tras el paso 8	12	34	AB	CD	00	00	FD	B5

16. C. Trazamos el programa

OPERACIÓN	D2	Comentarios
	13.57.AF.86	Valor de D2 al iniciarse el programa
ROR.W D0,D2	13.57.5F.0D	Rotación de la parte W de D2 15 veces, no afectando a la parte alta
MULU D1,D2	00.05.F0.D0	Multiplica sin signo D1 por D2, OPERANDO SIEMPRE EN W
ANDI.B #\$F0,D2	00.05.F0.D0	AND Inmediato, que afecta a la parte B, pero cuando hacemos (X AND 1) siempre es X, y (X AND 0) siempre es cero, no hace falta ni plantearlo en binario al ser $(X_1 X_2 X_3 X_4 X_5 X_6 X_7 X_8 \text{ AND } 11110000) = X_1 X_2 X_3 X_4 0000$

Para que veáis la rotación, aunque si nos fijamos, rotar 15 veces solo es cambiar el primer bit por el último (pongo de colores el bit que se va moviendo cada vez):

Valor inicial Hexa	A				F				8				6			
Inicialmente	1	0	1	0	1	1	1	1	1	0	0	0	0	1	1	0
Rotación 01	0	1	0	1	0	1	1	1	1	1	0	0	0	0	1	1
Rotación 02	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0	1
Rotación 03	1	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0
Rotación 04	0	1	1	0	1	0	1	0	1	1	1	1	1	0	0	0
Rotación 05	0	0	1	1	0	1	0	1	0	1	1	1	1	1	0	0
Rotación 06	0	0	0	1	1	0	1	0	1	0	1	1	1	1	1	0
Rotación 07	0	0	0	0	1	1	0	1	0	1	0	1	1	1	1	1
Rotación 08	1	0	0	0	0	1	1	0	1	0	1	0	1	1	1	1
Rotación 09	1	1	0	0	0	0	1	1	0	1	0	1	0	1	1	1
Rotación 0A	1	1	1	0	0	0	0	1	1	0	1	0	1	0	1	1
Rotación 0B	1	1	1	1	0	0	0	0	1	1	0	1	0	1	0	1
Rotación 0C	1	1	1	1	1	0	0	0	0	1	1	0	1	0	1	0
Rotación 0D	0	1	1	1	1	1	0	0	0	0	1	1	0	1	0	1
Rotación 0E	1	0	1	1	1	1	0	0	0	0	0	1	1	0	1	0
Rotación 0F	0	1	0	1	1	1	1	1	0	0	0	0	1	1	0	1
Valor final Hexa	5				F				0				D			