

# SOLUCIONES AL FINAL DEL EXAMEN.

## INGENIERÍA TÉCNICA en INFORMÁTICA de SISTEMAS

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

CURSO: 2002/2003 FECHA: 8 de Septiembre de 2003 HORA: 9:00 DURACIÓN TOTAL: 2 horas

CÓDIGO CARRERA: CÓDIGO ASIGNATURA: CONVOCATORIA: Septiembre – 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. Para construir un código de Hamming válido para ser utilizado con datos de 14 bits es preciso añadir:**

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

**2. Indique cual es la misión del biestable N del registro de estado:**

- Se pone a 1 si el resultado del último valor calculado fue cero.
- Se pone a 1 si el resultado del último valor calculado fue negativo.
- Se pone a 1 si el resultado del último valor calculado fue positivo
- Se pone a 1 si el resultado del último valor calculado fue cero o negativo.

**3. En el estándar IEEE 754 es CIERTO que:**

- La mantisa se representa en el sistema de complemento a dos.
- El exponente se representa en exceso a  $2^{n-1}$ .
- La coma está a la izquierda del bit implícito.
- Utiliza el formato de precisión ampliada, valiendo siempre 1 el bit implícito.

**4. Acerca del M68000 NO ES CIERTO que:**

- Incluye 8 registros de datos que se denominan D0, D1, D2, D3, D4, D5, D6 y D7.
- El registro de códigos de condición coincide con los 5 bits menos significativos del registro de estado.
- El registro D7 recibe también el nombre de SP.
- Los registros de direcciones tienen un tamaño de 32 bits.

**5.Cuál de los siguientes elementos principales de un computador digital es el encargado de interpretar y secuenciar las instrucciones:**

- La unidad de entrada-salida
- La unidad de control
- La unidad aritmético lógica
- La unidad de memoria

**6. La distancia entre la combinación binaria 111010 y la 110101 es:**

- 4.
- 4.
- 5.
- 5.

**7. ¿A qué 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.

**8. ¿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.

**9. 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 de memoria:**

- 3
- 4
- 5
- 1

**10. 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

**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  $f = m_1 + m_4 + m_6 + m_7$**

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

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

**12. Obtener la representación binaria del número decimal  $1,4848 \cdot 10^4$  en formato normalizado IEEE 754 para coma flotante de 32 bits:**

- a) \$68640000
- b) \$06E80000
- c) \$46680000
- d) \$046E0000

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

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

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

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

**14. Indicar el contenido del registro D0 después de ejecutarse el siguiente fragmento de código del M68000:**

- a) \$0000542
- b) \$00051040
- c) \$000F0443
- d) \$0000B410

```

COM   ORG       2500
      EQU      $F5F
      MOVE.L   #$00F04S1,D0
      ADD.L    NUL,D0
      AND.W    #COM,D0
NUL   DC.L     $42
      END
    
```

**15. 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)  $4,063232 \cdot 10^5$

**16. 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

```

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

# SOLUCIONES NO OFICIALES

Por Jose Antonio Vaqué

1. **A.** (página 106). Si consideramos la ecuación  $2^k \geq n + k + 1$ , con  $n = 11$  probamos  $k$  a partir del 1, resultando los valores  $(k=1) 2 \geq 16$  [NO],  $(k=2) 4 \geq 17$  [NO],  $(k=3) 8 \geq 18$  [NO],  $(k=4) 16 \geq 19$  [NO],  $(k=5) 32 \geq 17$  [SI]
2. **B**
3. **D** (página 75).
4. **C**
5. **C** (página 234)
6. **A** (página 86). Distancia es el número total de bits que hay que cambiar para igualar los códigos
7. **B.**
8. **D**
9. **C.** El direccionamiento directo requiere tener la dirección del elemento en la propia instrucción, por lo que al ser la memoria de 64Kb necesitamos 16 bits, necesitamos 2 octetos para cada dirección, mas el octeto para el código de instrucción, total 5 octetos.
10. **D** (página 270)
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            |
| <del>m1</del> | <del>M6</del> |
| m2            | M5            |
| m3            | M4            |
| <del>m4</del> | <del>M3</del> |
| m5            | M2            |
| <del>m6</del> | <del>M1</del> |
| <del>m7</del> | <del>M0</del> |

Resultando M2.M4.M5.M7

12. **C.** Si pasamos a binario  $1.4848 \times 10^4 = 11101000000000 =$  normalizando  $= 1.1101 \times 2^{13}$   
Montamos el numero a partir de **s=0, e=13+127=140= 10001100, m=(1.)1101:**

**01000110.01101000.00000000.00000000 = 46.68.00.00**

13. **B.** Aplicamos las reglas, pero para no liarnos, creamos unas variables auxiliares:

$$x1 = A+C+D \rightarrow x1' = A'C'D'$$

$$x2 = B+C+D' \rightarrow x1' = B'C'D$$

$$y1 = A.B' \rightarrow y1' = A'+B$$

$$x3 = y1+C+D' \rightarrow x3' = y1'C'D = (A'+B)C'D = A'C'D + BC'D$$

$$(x1.x2.x3)' = x1' + x2' + x3' = A'C'D' + B'C'D + A'C'D + BC'D$$

$$\rightarrow \text{Por absorción de estos términos} = A'C' + B'C'D + BC'D$$

$$\rightarrow \text{Por absorción de estos términos} = A'C' + C'D$$

$$\rightarrow \text{Sacamos factor común} = C' (A'+D)$$

14. C. Trazamos el programa:

| LÍNEA | OPERACIÓN            | D0          | Comentarios   |
|-------|----------------------|-------------|---|
| 1     | ORG 2500             |             | El programa comenzará a ensamblarse en esta dirección   |
| 2     | COM EQU \$F5F        |             | Definimos un símbolo para el valor \$F5F  |
| 3     | MOVE.L #\$00F0481,D0 | 00.0F.04.81 | Carga este valor en D0, usando los 4 bytes del mismo  |
| 4     | ADD.L NUL,D0         | 00.0F.04.C3 | Le suma a D0 lo que esté en la posición de memoria marcada como NUL, y el resultado lo guarda en D0, utilizando 4 bytes para D0 y para el Operando (NUL se define en la línea 6, esto será un problema para los ensambladores de una sola fase, pero no para la ejecución):<br>Valor del registro D0 00 0F 04 81<br>Valor a SUMAR (L) 00 00 00 42<br>SUMA que pasa a D0 00 0F 04 C3 |
| 5     | AND.W #COM,D0        | 00.0F.04.43 | Hacemos un AND entre D0 y el valor que hemos marcado como COM, guardando el resultado en D0, solo afecta a los dos bytes de menor peso de D0:<br>Valor del registro D0 00 0F 04 C3<br>Valor a operar (W) --- --- 0F 5F<br>AND que pasa a D0 00 0F 04 43   |
| 6     | NUL DC.L \$42        | 00.0F.04.43 | Guarda un lugar en la memoria para almacenar un LONG (4 bytes), inicializando el valor con \$00.00.00.42  |
| 7     | END                  | 00.0F.04.43 | Se acaba el programa  |

15. C. Si pasamos a binario 49FC0000 = **01001001.11111000.0000000.00000000**  
 Montamos el numero a partir de **s=0, e=10010011-127=147-127=20, m=(1.) 1111=1.96875**

$$n=2^{20} \times 1.96875 = 2064387 = 2.064384 \times 10^6$$

16. 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<br>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 2 | 12   | 34   | AB   | CD   | 56   | 78   | 90   | EF   |
| Tras el paso 3 | 12   | 34   | AB   | CD   | 56   | 78   | 90   | EF   |
| Tras el paso 4 | 12   | 34   | AB   | CD   | 56   | 78   | 90   | EF   |
| Tras el paso 5 | 12   | 34   | AB   | CD   | 56   | 78   | 90   | EF   |
| Tras el paso 6 | 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   |