

Solución de Jose Antonio Vaque Urbaneja al examen de ETC I sistemas Febrero 2003

Gracias a TKA por escribir las preguntas. Gracias a Siro Pérez por las respuestas

Las respuestas mias las pongo en fondo VERDE.

Las respuestas en que Siro Perez no coincide las pongo en fondo verde la mia, y en fondo amarillo la suya.

Añado en texto rojo comentario y la página del MORA en que se encuentra la solución (no tengo el CERRADA y no puedo comparar)

1º - Acerca del M68000 NO ES CIERTO que:

- a) Incluye 8 registros de datos D0...D7
- b) El registro de código de condición coincide con los 5 bits menos significativos del registro de estado
- c) El registro D7 también se llama SP**
- d) Los registros de direcciones son de 32 bits

Pag. 434: D7 se llama PPS (puntero de pila de usuario) o USP (user stack pointer),

2º - La pseudoinstrucción que se identifica con EQU, DEF o un signo igual (según el ensamblador) sirve para:

- a) Manipular el contador de dirección de ensamblado
- b) Reserva de espacio en memoria
- c) Definir constantes
- d) Definir símbolos**

Pág. 480: “Se utiliza para definir un símbolo que se va a utilizar ...”

3º - Según la estructura del computador digital estudiado en la asignatura, señale cuantos bloques o unidades comprende la UCP:

- a) 2**
- b) 3**
- c) 4
- d) No es un número definido

Página 234: “La UCP se puede considerar constituida a su vez por dos unidades funcionales; la unidad de control y la unidad aritmética-lógica”.

La unidad de memoria y la unidad de E/S son externas a la CPU

Puede existir cierta confusión si miras el esquema del ordenador de la página 256, ya que aparece una tercera unidad que denomina Unidad de Interfase BUS, que parece estar aparte, pero realmente está compuesto de registros que pertenecen a la A/L y señales que pertenecen a la de control, estando repartida entre las dos unidades.

4° - Señala que afirmación sobre direccionamiento directo absoluto es FALSA:

- a) La instrucción contiene la dirección exacta en la que está el objeto.
- b) El objeto está en una pos de la memoria principal
- c) El tamaño del operando direccionado queda limitado por el nº de bits del campo de dirección de la instrucción
- d) El rango de posiciones direccionables queda limitado por el tamaño del campo del operando

Pág. 369. No aclara más que la respuesta a) como cierta, por lo que evidentemente la b) también lo es. Para descartar recorro a la lógica:

Me dicen que en el libro de CERRADA pone: "La instrucción contiene la dirección de memoria exacta, sin compactar, en que se encuentra el objeto. Por tanto, el objeto está en una posición de la memoria principal. El rango de posiciones de memoria direccionables queda limitado por el tamaño del campo del operando."

5° - Cuál de los siguientes elementos de un computador digital interpreta y secuencia las instrucciones:

- a) Unidad de E/S
- b) Unidad de control
- c) UAL
- d) Unidad de memoria

Pag. 234 y 249, aunque me parece que en el CERRADA está escrito literalmente así, en el MORA no lo dice con esas mismas palabras.

6° - En el IEEE 754 se cumple que:

- a) La mantisa está en complemento a 2
- b) El exponente está en exceso a $2^{(n-1)}$
- c) La coma está a la izquierda del bit implícito
- d) Utiliza un formato de precisión ampliada, valiendo siempre 1 el bit implícito

Pag. 75, aunque me parece que en el CERRADA está escrito literalmente así, en el MORA no lo dice con esas mismas palabras. De todas formas por eliminación:

- a) Falso sin dudarlo
- b) 2^{n-1} para $n=8$ bits dará 128, lo cierto es que es $(2^{n-1} - 1)$, que para $n=8$ bits dará $128 - 1 = 127$
- c) Está a la derecha del bit implícito, que es un uno
- d) Es la definición exacta de la mantisa en el libro de Cerrada, añadiendo que el cero tiene una representación especial, por lo que es la cierta

7° - En un sistema de numeración la base es:

- a) El conjunto de cantidades representables
- b) El conjunto de reglas utilizadas para la representación
- c) El nº de símbolos usados en la representación
- d) El nº de cifras empleadas en la representación.

Pag. 30 a 33, aunque me parece que en el CERRADA está escrito literalmente así, en el MORA no lo dice con esas mismas palabras.

8° - Se envía una secuencia de 14 bits de datos con Hamming. Hay que añadir:

a) 5 bits de paridad

b) 4

c) 3

d) 2

Pág 106, $2^k > n + k + 1$ donde n es el nro de bits a transmitir y k es el número de bits a añadir a los n que se transmiten. Podemos calcular:

Con 4, saldría $2^4 > 14 + 4 + 1 \rightarrow 16 > 19$ que no es cierto

Con 5, saldría $2^5 > 14 + 5 + 1 \rightarrow 32 > 20$ que si es cierto

Por la cuenta de la vieja, que es más rápido y efectivo:

Con 1 cubrimos 1 \rightarrow No llega

Con 2 cubrimos 2 \rightarrow No llega

Con 3 cubrimos 4 \rightarrow No llega

Con 4 cubrimos 11 \rightarrow No llega

Con 5 cubrimos 26 \rightarrow Como necesitamos 14, tomamos este que el siguiente que se pasa

Con 6 cubrimos 57 ...

9° - Cuantas casillas adyacentes tiene una casilla de un mapa de Karnaugh de 4 variables:

a) Todos 4

b) Todos 15

c) Las del centro 4, las de las esquinas 2 y el resto del borde 3.

d) Según la función que sea.

Pág. 172: "En los mapas de Karnaugh de n variables, cada cuadro es adyacente a n cuadros"

10° - Indique la misión del bit N en el registro de estado:

a) Se pone a 1 si el resultado del último cálculo fue 0

b) Se pone a 1 si el resultado del último cálculo fue negativo

c) Se pone a 1 si el resultado del último cálculo fue positivo

d) Se pone a 1 si el resultado del último cálculo fue negativo ó 0

Pág, 382, "Igual al bit más significativo (signo) del resultado" Recuerdo que positivos comienzan por cero, negativos por uno.

LOS PROBLEMAS

11° - Simplificar $[(A+C+D) \cdot (B+C+D') \cdot (A \cdot B' + C + D')]'$

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

Operamos directamente, ya que es muy sencillo, ni Karnaugh ni tablas de verdad:

$$[(A+C+D) \cdot (B+C+D') \cdot (A \cdot B' + C + D')] \rightarrow (\text{lo contrario de un producto suma de contrarios}) =$$

$$(A+C+D)' + (B+C+D')' + (A \cdot B' + C + D') \rightarrow \text{Lo divido en tres, para que lo veais mejor:}$$

$$(A+C+D)' \rightarrow (\text{lo contrario de una suma es el producto de los contrarios}) = A' \cdot C' \cdot D'$$

$$(B+C+D')' \rightarrow (\text{lo contrario de una suma es el producto de los contrarios}) = B' \cdot C' \cdot D$$

$$(A \cdot B' + C + D') \rightarrow (\text{igual}) = (A \cdot B')' \cdot C' \cdot D \rightarrow (\text{Desarrollo } (A \cdot B')' = A' + B) = (A' + B) \cdot C' \cdot D \rightarrow$$

$$(\text{multiplico por la propiedad distributiva}) = A' \cdot C' \cdot D + B \cdot C' \cdot D$$

Ahora los sumo:

$$A' \cdot C' \cdot D' + B' \cdot C' \cdot D + A' \cdot C' \cdot D + B \cdot C' \cdot D \rightarrow (\text{Los reordeno para seguirlos mejor}) =$$

$$A' \cdot C' \cdot D' + A' \cdot C' \cdot D + B' \cdot C' \cdot D + B \cdot C' \cdot D \rightarrow (\text{saco } A' \cdot C' \text{ factor común de los dos primeros, y } C' \cdot D' \text{ de los dos últimos}) =$$

$$A' \cdot C' \cdot (D' + D) + C' \cdot D (B' + B) \rightarrow (\text{Como } D' + D = 1, \text{ y } X \cdot 1 = X, \text{ elimino lo que está en los paréntesis}) =$$

$$A' \cdot C' + C' \cdot D \rightarrow (\text{Ahora saco } C' \text{ factor común, y ya está}) = C' \cdot (A' + D)$$

12° - Obtener $1'4848 \cdot 10^4$ en IEEE754 estándar de 32 bits

a) \$68640000

b) \$46680000

[las respuestas c y d no las tengo]

$$1.4848 \times 10^4_d = 14848_d = 11101000000000_b = 1.1101000000000_b \times 2^{13}_d$$

$$127_d + 13_d = 140_d = 10001100_b$$

S	Exponente								Mantisa																											
0	1	0	0	0	1	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4				6				6				8				0				0				0				0							

13° - Hamming, 6 bits de datos: secuencia 0010011101

- a) Error en bit 4
- b) Error en bit 6
- c) Error en bit 7
- d) No hay error**

Montamos la tabla (que es un tostón, pero más sencillo de operar), teniendo en cuenta que el bit menos significativo lo sabemos por que si lo pones al contrario, da error en el bit 11, y solo son 10.

Solo debemos reflejar en la parte derecha de la tabla los valores para los que se recibe un 1, calcular su paridad por columnas, y completar para paridad par:

Valor	Bit	Recibido				
0001	P0	0				
0010	P1	0				
0011	D0	1	0	0	1	1
0100	P2	0				
0101	D1	0				
0110	D2	1	0	1	1	0
0111	D3	1	0	1	1	1
1000	P3	1	1	0	0	0
1001	D4	0				
1010	D5	1	1	0	1	0
Paridad de la columna			Par	Par	Par	Par
Bit a añadir para paridad par			0	0	0	0

14° - La primera forma canónica de una función de 3 variables es $m1+m4+m6+m7$. La segunda forma canónica es:

- a) $M0 \cdot M2 \cdot M3 \cdot M5$
- b) $M2 \cdot M3 \cdot M5$
- c) $M2 \cdot M4 \cdot M5 \cdot M7$**
- d) $M0 \cdot M5 \cdot M7$

Vamos por la cuenta de la vieja, que es más rápido y no te equivocas, pones los minterms en una columna, en otra los Maxterms en orden inverso, tachas los minterms que tienes, y lo que queda son los que te piden::

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

Quedan M2, M4, M5 y M7. Sale en aproximadamente 5 segundos, y no te equivocas, con las operaciones si te puedes equivocar.

15° - Como afecta a D4 instrucción ADDI.W #\$1011,D4, suponiendo que antes de ejecutarla D4=\$45C8F9AF:

- a)\$ 45C909C0
- b)\$ 45C909B0
- c)\$ 45C809C0**
- d)\$ 45C809B0

Montamos otra tablita de las que tanto me gustan:

Valor del registro D4	45	C8	F9	AF
Valor a SUMAR (W)	--	--	10	11
SUMA	45	C8	09	C0

\$F9 + \$10 = \$109, pero al ser W, se pierde el acarreo.

16° - Dado el siguiente código en ensamblador:

```

COM      ORG      2500
         EQU      $F5F
         MOVE.L   #$000f0481,D0
         ADD.L    NUL,D0
         AND.W    #COM,D0
NUL      DC.L     $42
         END
    
```

¿Cual sería el valor final de D0?

- a) \$000F0443**
- b)\$ 00000542
- c)\$ 00051040
- d)\$ 0000B410

Paso a paso:

ORG 2500

El programa comenzará a ensamblarse en esta dirección

COM EQU \$F5F

Definimos un símbolo

MOVE.L #\$000f0481,D0

Carga este valor en D0, usando los 4 bytes del mismo

ADD.L NUL,D0

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 2 líneas después, es un problema para los ensambladores de una fase, pero no para la ejecución).

Valor del registro D0	00	0F	04	81
Valor a SUMAR (L)	00	00	00	42
SUMA que pasa a D0	00	0F	04	C3

AND.W #COM,D0

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.

Valor del registro D0	00	0F	04	C3
Valor a operar (W)	--	--	0F	5F
AND que pasa a D0	00	0F	04	43

NUL DC.L \$42

Guarda un lugar en la memoria para almacenar un LONG (4 bytes), inicializando el valor con \$00.00.00.42

END

Sacabao.

Espero que hayáis tendido suerte y aprobemos todos, que este examen no ha sido tan difícil como esperábamos (yo por lo menos). Un saludo y a seguir con el segundo cuatrimestre.

Jose Antonio Vaqué
javaque@ya.com