

**SOLUCIONES COMENTADAS AL EXAMEN DE  
ESTRUCTURAS DE LOS COMPUTADORES.  
SEPTIEMBRE DE 1.999**

(I.T.INFORMATICA DE SISTEMAS – I.T. INFORMATICA DE GESTION)

1º) Se tiene un computador conectado a una red telefónica por la que llegan números en coma flotante relativos a las temperaturas en una cámara frigorífica de carnes. Los datos viene protegidos mediante código Hamming. El formato de representación es en complemento a 2 tanto para el exponente como para la mantisa que además emplea la técnica del bit implícito.

A nuestro terminal ha llegado el dato siguiente:

0	1	1	1	0	0	1	1	0	1	0	1	0	1	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Se pide:

- a) Sabiendo que tanto la mantisa como el exponente se expresan en complemento a 2; que el ancho del exponente es de un bit más que los bits de paridad necesarios para proteger el dato; que el resto de los bits del dato pertenecen a la mantisa y que ésta emplea la técnica del bit implícito se pide calcular el rango de numeración para la representación anterior.

En el caso de protección de la información mediante el código de Hamming, el número de bits de paridad necesarios debe satisfacer la inecuación:  $2^p \geq n + p + 1$

En nuestro caso tenemos que  $n + p + 1 = 18 \Rightarrow n + p = 17$ . Como se debe satisfacer que  $2^p \geq 17 \Rightarrow p = 5$  ( $2^5 = 32$ ). Por lo tanto el ancho del exponente son 6 bits (uno más que los necesarios para proteger el dato mediante el código Hamming)

En nuestro caso, tenemos 18 bits. Si le quitamos los 5 necesarios para el bit de paridad, nos quedan 13 bits; de los cuales 6 son para el exponente con lo que nuestra mantisa será de 7 bits. Por lo tanto nuestro dato se representa con una mantisa de 7 bits (empleando la técnica del bit implícito) y en complemento a 2 y con un exponente de 6 bits representado en complemento a 2.

Como nos dicen que la mantisa emplea la técnica del bit implícito deberemos contar ese bit para el rango de la mantisa.

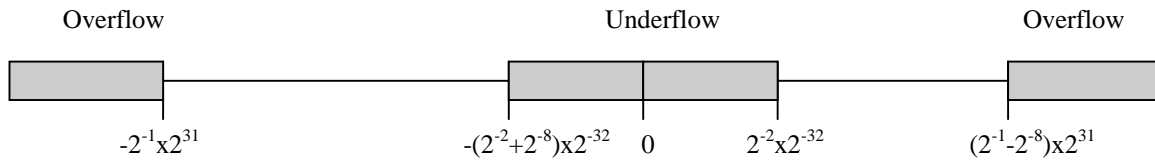
Rango Mantis:

Positivos:  $[2^{-2}, 2^{-1}-2^{-8}]$  y para los negativos  $[-(2^{-2}+2^{-8}), -2^{-1})$

Rango del Exponente:

El rango para el exponente será de:  $[-2^{n-1}, 2^{n-1}-1]$  con lo que como el ancho es de 6 bits tendremos:  $[-2^5, 2^5-1] = [-32, 31]$

Con lo que la recta real queda de la forma:



- b) Verificar si el número llega correctamente y si fuese necesario realizar la corrección correspondiente.

Si se supone que recibimos el siguiente número en coma flotante, protegido mediante código Hamming, ver si el número llega correctamente o no.

**0111001101010110**

Nos indican que el número siguiente está codificado según el código de Hamming y se nos pide que comprobemos si el número ha sido recibido de forma correcta.

En el diagrama inferior se han resaltado en **negrita** los bits de paridad de Hamming. Que ocupan las posiciones correspondientes a las potencias de dos: 1, 2, 4, 8, 16 ... A partir de las cuales se descomponen los demás números tal y como sigue:

1 = paridad	10 = 8 + 2
2 = paridad	11 = 8 + 2 + 1
3 = 2 + 1	12 = 8 + 4
4 = paridad	13 = 8 + 4 + 1
5 = 4 + 1	14 = 8 + 4 + 2
6 = 4 + 2	15 = 8 + 4 + 2 + 1
7 = 4 + 2 + 1	16 = paridad
8 = paridad	17 = 16 + 1
9 = 8 + 1	18 = 16 + 2

**0** **1** 1 1 0 0 1 1 0 1 0 1 0 1 0 **1** 1 0  
 $b_1$   $b_2$   $b_3$   $b_4$   $b_5$   $b_6$   $b_7$   $b_8$   $b_9$   $b_{10}$   $b_{11}$   $b_{12}$   $b_{13}$   $b_{14}$   $b_{15}$   $b_{16}$   $b_{17}$   $b_{18}$

El  $b_1$  protegerá a todos aquellos valores en cuya descomposición aparezca un 1. El  $b_2$  protegerá todos aquellos valores en los que se encuentra un 2 en su descomposición y así sucesivamente.

Los bits de paridad (resaltados en negrita en la parte superior) protegen a los bits siguientes:

$b_1 = \mathbf{b_3} \oplus b_5 \oplus \mathbf{b_7} \oplus b_9 \oplus b_{11} \oplus b_{13} \oplus b_{15} \oplus \mathbf{b_{17}} = 0$  *incorrecto ya que hay tres bits a uno*

$b_2 = \mathbf{b_3} \oplus b_6 \oplus \mathbf{b_7} \oplus \mathbf{b_{10}} \oplus b_{11} \oplus \mathbf{b_{14}} \oplus b_{15} \oplus b_{18} = 0$  como  $b_2$  tiene 1 es *incorrecto*.

$b_4 = b_5 \oplus b_6 \oplus \mathbf{b_7} \oplus \mathbf{b_{12}} \oplus b_{13} \oplus \mathbf{b_{14}} \oplus b_{15} = 1$  *correcto*.

$b_8 = b_9 \oplus \mathbf{b_{10}} \oplus b_{11} \oplus \mathbf{b_{12}} \oplus b_{13} \oplus \mathbf{b_{14}} \oplus b_{15} = 1$  *correcto*.

$b_{16} = \mathbf{b_{17}} \oplus b_{18} = 1$  *correcto*.

Con lo que el número no es correcto. Sumando los pesos de los bits de paridad erróneos nos indica que  $1 + 2 = 3$  luego es el bit  $b_3$  el erróneo debe tener un 0, por lo que el número que nos llega será el 000101 0101010

c) Calcular el valor del número recibido.

El exponente viene expresado en complemento a 2 y es el 000101. Es un número positivo y equivale al 5

La mantisa es 0101010 a la que le falta el bit implícito con lo que debemos calcular el valor de la mantisa 10101010 que es negativa y vale  $-(2^{-2}+2^{-4}+(2^{-5}-2^{-7}))$ . El número valdrá por lo tanto  $-(2^{-2}+2^{-4}+(2^{-5}-2^{-7})) \times 2^5 = -(8 + 2 + 0,5 + 0,25) = -10,75$

2º) Se pide realizar un **procedimiento** en ensamblador del 8086 que pase un número de 16 bits representado en complemento a dos a su correspondiente valor representado en exceso  $2^{n-1}$ , siendo  $n = 16$ . El dato numérico en complemento a dos se encuentra en la variable de tipo palabra NumCa2 y se debe dejar el resultado en la variable de tipo palabra NumExceso. Se valorará especialmente el empleo de máscaras para su resolución.

Para pasar de complemento a 2 a exceso  $2^{n-1}$  solamente se tiene que invertir el bit más significativo que nos indica el signo. Por lo tanto nuestro procedimiento será:

Conversion PROC FAR

PUSH AX	; Guardamos AX en la pila
MOV AX, NumCa2	; Obtenemos el número en complemento a 2
XOR AX, 8000h	; Empleamos la función XOR que con el
	; 8000h solamente invertirá el bit más significativo.
MOV NumExc2n, AX	; Guardamos el resultado
POP AX	; Recuperamos el valor de AX
RET	; Salimos del procedimiento

Conversion ENDP

3º) Se quiere diseñar una memoria para un circuito electrónico que se adaptará a una tarjeta de felicitación de cumpleaños de forma que al abrirla suene la melodía del cumpleaños feliz. El archivo de sonido es de tipo WAV y ocupa 384 K. y el programa que reproducirá la melodía ocupará, como máximo, 128 K.

Además se tienen que tener en cuenta los siguientes puntos:

- El cliente puede pedir que el fichero de sonido sea en formato MP3 192 K. con lo que el programa ocupará un máximo de 320 K. Dado que no existirán diferentes tarjetas de felicitación se deberá cambiar el contenido de la memoria.
- Se podrá cambiar las baterías sin que se pierda la información de las memorias.

Se sabe que el bus de datos es de 16 bits y el de direcciones de 20 bits y que se cuenta con los siguientes módulos de memoria disponibles para realizar el circuito:

Memoria ROM	Memoria EEPROM	Memoria RAM
128 K x 8	128 K x 8	128 K x 8
256 K x 8	256 K x 8	256 K x 8
512 K x 8	512 K x 8	512 K x 16

- a) Razonar brevemente el tipo de módulos de memoria que se deberán emplear para realizar el circuito con las especificaciones requeridas.

Dado que el contenido de la memoria se podrá cambiar a petición del cliente, el tipo de memorias a utilizar será el EEPROM.

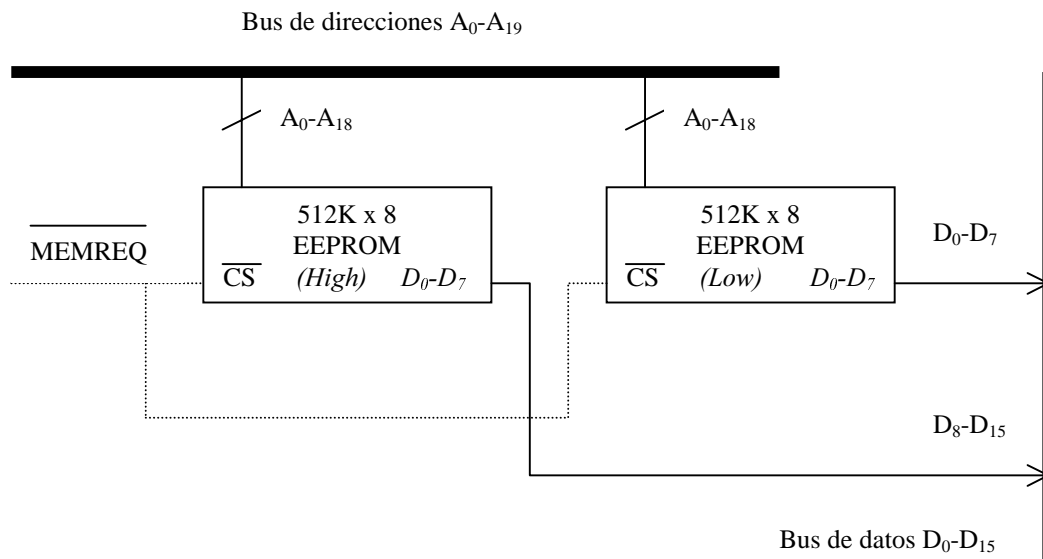
- b) Calcular el menor número de módulos necesario para realizar las especificaciones anteriores.

En los dos casos, las especificaciones de memoria, nos dan una necesidad de 512K. Como el bus de direcciones es de 20 bits que permite direccionar un máximo de 1MB. podremos direccionar los 512K que nos piden (solamente 19 bits  $A_0-A_{18}$ )

El menor número de módulos de memoria, dado que tenemos que escoger la opción de módulos EEPROM se da con dos módulos de 512K x 8 (tipo EEPROM)

- c) Realizar el mapa de memoria con una capacidad máxima para almacenar el fichero de sonido (WAV o MP3) y el programa correspondiente.

El mapa de memoria es el que se requiere para poder direccionar una pastilla de 512Kx8 de EEPROM. Es decir, tendré una fila formada por dos pastillas de memoria que se direccionarán a la vez, una contendrá los 8 bits superiores y otra los 8 bits inferiores de cada dirección.



**LOS ALUMNOS QUE DESEEN REVISIÓN DEL EXAMEN DEBERÁN ENTREGAR EN LA SECRETARÍA DEL DEPARTAMENTO DE AUTOMÁTICA UN FOLIO INDICANDO SUS DATOS PERSONALES Y EL PROBLEMA O PROBLEMAS QUE DESEAN REVISAR JUSTIFICADANDO EL MOTIVO DE SU DESACUERDO CON RESPECTO A LAS SOLUCIONES PUBLICADAS. NO SE ATENDERÁ NINGUNA REVISIÓN QUE NO SIGA DICHO CRITERIO.**

**LA FECHA LÍMITE PARA LA SOLICITUD DE REVISIÓN SERÁ EL MIÉRCOLES 15 DE SEPTIEMBRE DE 1999 A LAS 12:00.**

**LA REVISIÓN SERÁ EL JUEVES 16 DE 11:00-12:00 EN EL LABORATORIO L4 DEL EDIFICIO ESTE.**