Fundamentos de Arquitectura de Computadores

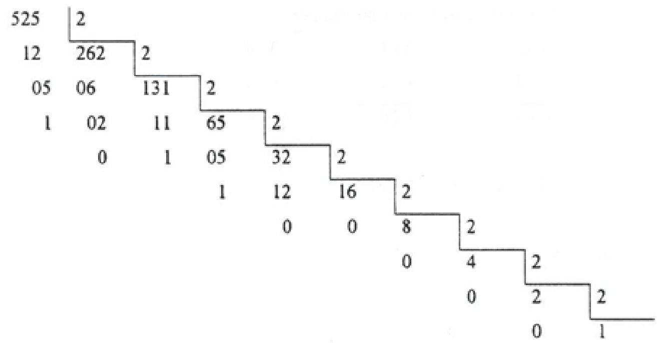
<u>Relación de Problemas Temas 1</u>

3.1. Transformar a binario natural los siguientes números:

(a) decimales:	**525;**	**0.17;**	**24.32**
(b) octales:	**372;**	**0.0375;**	**47.05407**
(c) hexadecimales:	**37F;**	**0.0AC54;**	**43AC.F32**

(a) $525)_{10}$.

Dividiendo sucesivamente por 2:



luego

$525)_{10} = 1000001101)_2$

(b) $0.17)_{10}$.

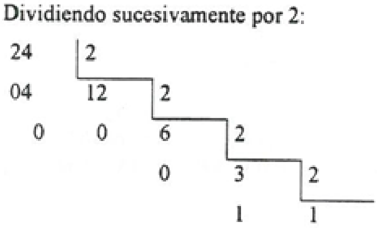
Multiplicando sucesivamente por 2:

0.17	0.34	0.68	0.36
x 2	x 2	x 2	x 2
0.34	0.68	1.36	0.72

$0.17)_{10} = 0.0010...)_2$.

(c) $24.32)_{10}$.

Dividiendo sucesivamente por 2:



Multiplicando sucesivamente por 2:

0.32	0.64	0.28	0.56	0.12
x 2	x 2	x 2	x 2	x 2
0.64	1.28	0.56	1.12	0.24

$24.32)_{10} = 24)_{10} + 0.32)_{10} = 11000.01010...)_2$.

(d) 372)8 = 001 111 010)2

(e) 0.0375)8 = 0.000 011 111 101)2

(f) 47.05407)8 = 100 111.000 101 100 000 111)2

(g) 0x37F = 0011 0111 1111)2

(h) 0x0.0AC54 = 0.0000 1010 1100 0101 0100)2

(i) 0x43AC.F32 = 0100 0011 1010 1100.1111 0011 0010)2

2. Indique la representación de los siguientes números, razonando su respuesta:

a) -16 en complemento a dos con 5 bits

El rango de representación es [-16, 15], por tanto el número es representable. -16 en complemento a dos con 5 bits es 10000

b) 4 en complemento a uno con 5 bits

4 en binario puro con 5 bits es 00100. Como el número es positivo, ya se encuentra representado también en complemento a uno.

c) 14 en signo magnitud con 5 bits.

01110, El primer dígito indica el signo, en este caso positivo

d) -14 en complemento a dos con 5 bits

El rango de representación es [-16, 15], por tanto el número es representable. 14 en binario puro con 5 bits es 01110. Se complementa y se suma uno: 10001 + 1 = 10010

3. Represente en el estándar de coma flotante IEEE 754 de 32 bits el siguiente número: -32.5

-32,5(10 = -100000,1(2 = -100000,1 x 20 = 1,000001 x 25
Signo = 1, por ser negativo
Exponente = 5 + 127 = 132 = 10000100
Mantisa = 00000100000…. 00000
El número es 11000010000000100000000….00 = 0xC2020000

3. Obtener la representación interna en simple precisión según las especificaciones IEEE 754 del número N = 37,485 · 10^{-17} Redondear y dar el resultado en hexadecimal empaquetado.

N = 37,485 · 10^{-17} = 0,37485 · 10^{-15}

$10^{-15} = 2^{x}$ → x = - 15 / log_10 2 = - 49,82892…

N = (0,37485 · $2^{-0.82892...}$)·2^{-49} = 0.2110217 · 2^{-49}

N_16 = 0.3605862 · 2^{-49} = 0.0011 0110 0000 0101 1000 0110 0010… · 2^{-49}

N_2 = 1.1011 0000 0010 1100 0011 000|1 0… · 2^{-52} (se trunca)

E = 127 – 52 = 75$)_{10}$ = 4B$)_{16}$ = 0100 1011

S = 0 (número positivo).

N = 0 | 0100 1011 | 1011 0000 0010 1100 0011 000 → 0010 0101 1101 1000 0001 0110 0001 1000 → 25D81618

3.18 Obtener la representación interna del número 3754.8976 x 10^8 en notación IEEE 754.

Haciendo la transformación $10^8 = 2^y$ tenemos que:

$$y = 26.57542476$$

De donde N = 5595.233444 x 2^{26}; Pasando a binario N:

$$N = 1010111011011.001110111100 \times 2^{26}$$

Normalizando N, nos queda:

$$N = 1.0101110110110011101111 00 \times 2^{38}$$

Calculamos el exponente a almacenar:

$$e = E + S = 38 + 127 = 165 \quad (S = 2^{ne-1})$$

$$e = 10100101)_2$$

Por tanto la representación interna sería:

0 10100101 01011101101100111011110

O bien, en hexadecimal, N → 52AED9DE

*** 7.11 (13/2/95) El contenido en hexadecimal de dos datos, en precisión sencilla, representado en notación IEEE 754, es:**

X = DEB0 0000   Y = 5DE0 0000

a) **Reproducir las operaciones que efectuaría el computador (en binario o hexadecimal) para obtener X+Y**
b) **Reproducir las operaciones que efectuaría el computador (en binario o hexadecimal) para obtener X/Y**

X= DEB0 0000 ⇒ 1 | 101 1110 1 | 011 0000 0000 ... ⇒ s_X =1; e_X=1011 1101; M_X=1.0110 0000
Y= 5DE0 0000 ⇒ 0 | 101 1101 1 | 110 0000 0000 ... ⇒ s_Y =0; e_Y=1011 1011; M_Y=1.1100 0000

a) R=X+Y
e_X-e_Y= 0000 0010 = 2$)_{10}$ ⇒ M'$_X$=M_X ; M_Y= 0.0111·2^2 ⇒ M'$_Y$=0.0111; e'$_R$=e_X=1011 1101;
Como X<0 e Y>0, hay que restar las mantisas:

$$\begin{array}{rl} M'_X & \rightarrow 1.0110 \\ -\ M'_Y & \rightarrow 0.0111 \\ \hline M'_R & \rightarrow 0.1111 \end{array} = 1.111 \cdot 2^{-1} \Rightarrow M_R = 1.1110 \text{, y hay que corregir el exponente.}$$

$e_r = e'_r - 1 = 1011\ 1101 - 1 = 1011\ 1100$

Luego:

	signo	*exponente*	*mantisa*	
R=X+Y →	1	1011 1100	1110 0000 0000 0000 0000 000	→ DE70 0000

b) R=X/Y

$e'_R = e_X - e_Y + S = 1011\ 1101 + 0111\ 1111 - 1011\ 1011 = 1000\ 0001;$

$$\frac{M_X}{M_Y} = \frac{1.011}{1.110} = \frac{1011}{1110}$$

```
                 |1110
 1011.0          0'1100100100....100
  111 0
 010000
 - 1110
   00010000
    -  1110
      00010000
```

Luego $M'_R = 0.1100\ 100\ 100\ 100 \ldots = 1.100\ 100\ 100\ 100 \ldots \cdot 2^{-1}$

$\Rightarrow$ $M_R = 1.100\ 100\ 100 \ldots$, y hay que corregir el exponente: $e_R = e'_R - 1 = 1000\ 0001 - 1$ =

= 1000 0000

como $s_X = 1$; $s_Y = 0$ $\Rightarrow s_R = 1$; con lo que:

	signo	*Exponente*	*mantisa*	
R=X/Y →	1	1000 0000	1001 0010 0100 1001 0010 010	→ C049 2492

El contenido en hexadecimal de dos datos en coma flotante con representación IEEE 754 simple precisión es:

A = 40D6 5F23 y B = B207 DD23

¿Cómo calcularía internamente y qué resultado obtendría un computador al efectuar la operación A + B?

A = 40D6 5F23 → 0 100 0000 1 101 0110 1001 1111 0010 0011
B = B207 DD23 → 1 011 0010 000 0111 1101 1101 0010 0011

Sa = 0;	Ea = 1000 0001;	Ma = 1,101 0110 1001 1111 0010 0011
Sb = 1;	Eb = 0110 0100;	Mb = 1,000 0111 1101 1101 0010 0011

Ea > Eb; Ea – Eb = 0x81 – 0x64 = 129 – 100 = 29)10 ➔ B << A ➔ A + B = A → 40D6 5F23