

Aritmetica del Computador

Hermes Pantoja Carhuavilca

Facultad de Ingeniería Mecánica
Universidad Nacional de Ingeniería

Métodos Numéricos



Contenido

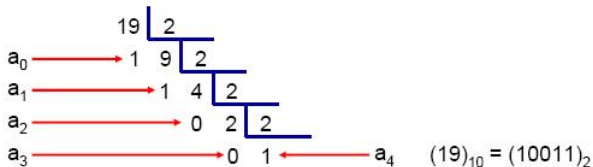
- 1 Conversión entre bases
- 2 Estándar IEEE-754



Conversión entre bases

Convertir de base 10 a cualquier base (s)

Ejemplo: convertir el número $(19)_{10}$ a binario.



Ejemplo: convertir el número $(127)_{10}$ a hexadecimal.



Conversión entre bases

Comandos de MatLab

```
>> dec2bin(19)
```

```
ans =
```

```
10011
```

```
>> dec2hex(127)
```

```
ans =
```

```
7F
```



Conversión entre bases

Otros Comandos de MatLab

- **dec2base** `dec2base(63,8)` retorna 77
- **bin2dec** `bin2dec('010111')` retorna 23
- **hex2dec** `hex2dec('3ff')` retorna 1023
- **base2dec** `base2dec('31',5)` retorna 16



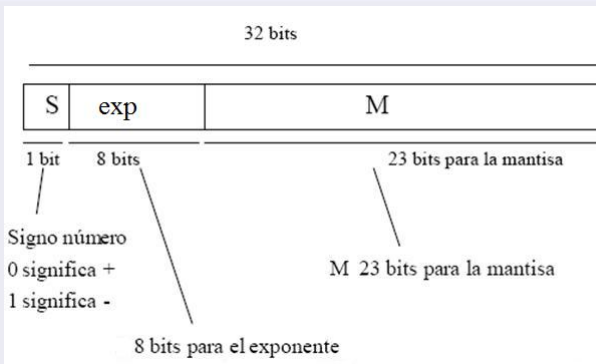
Ejercicios

- Convertir 1234 a base 5
- Convertir 101001011_2 a base 8
- Convertir 101001110011_2 a hexadecimal.
- Convertir 3271245_8 en decimal
- Convertir 314 en octal.
- Convertir $1F6_{16}$ a base 2.



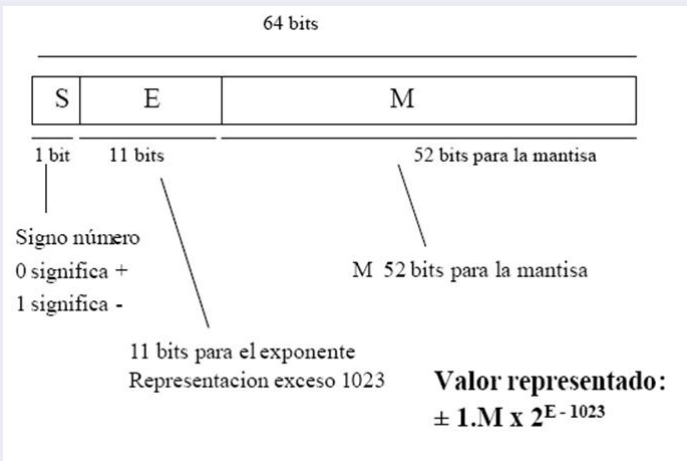
Estándar IEEE-754

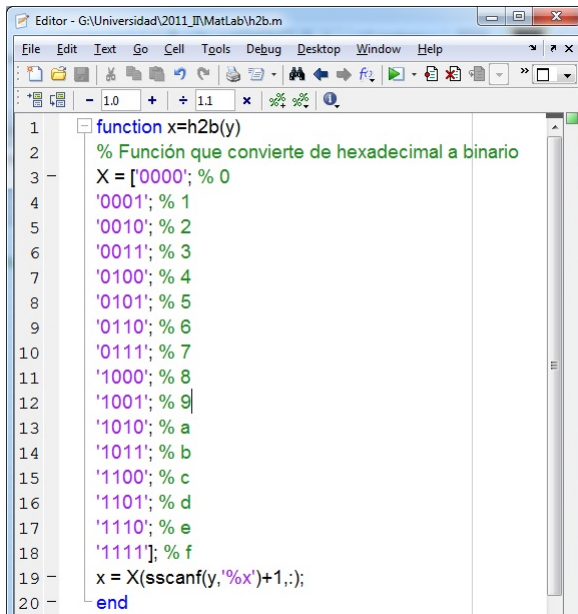
Precisión Simple: 32 bits



Estándar IEEE-754

Precisión Doble: 64 bits

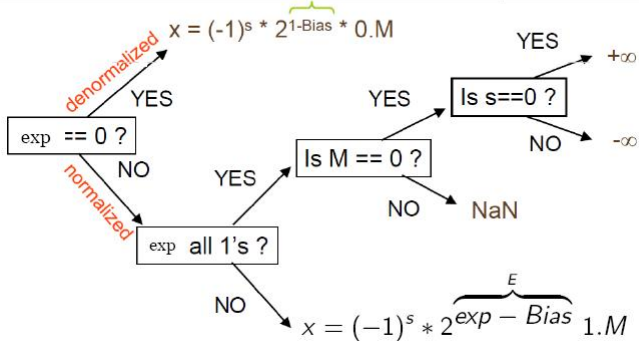
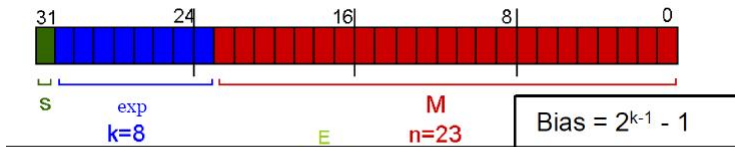




The image shows a screenshot of a MATLAB editor window titled "Editor - G:\Universidad\2011_II\MatLab\h2b.m". The window contains a function definition for converting a hexadecimal string to a binary string. The code is as follows:

```
1 function x=h2b(y)
2 % Función que convierte de hexadecimal a binario
3 X = ['0000'; % 0
4 '0001'; % 1
5 '0010'; % 2
6 '0011'; % 3
7 '0100'; % 4
8 '0101'; % 5
9 '0110'; % 6
10 '0111'; % 7
11 '1000'; % 8
12 '1001'; % 9
13 '1010'; % a
14 '1011'; % b
15 '1100'; % c
16 '1101'; % d
17 '1110'; % e
18 '1111']; % f
19 x = X(sscanf(y, '%x')+1,:);
20 end
```





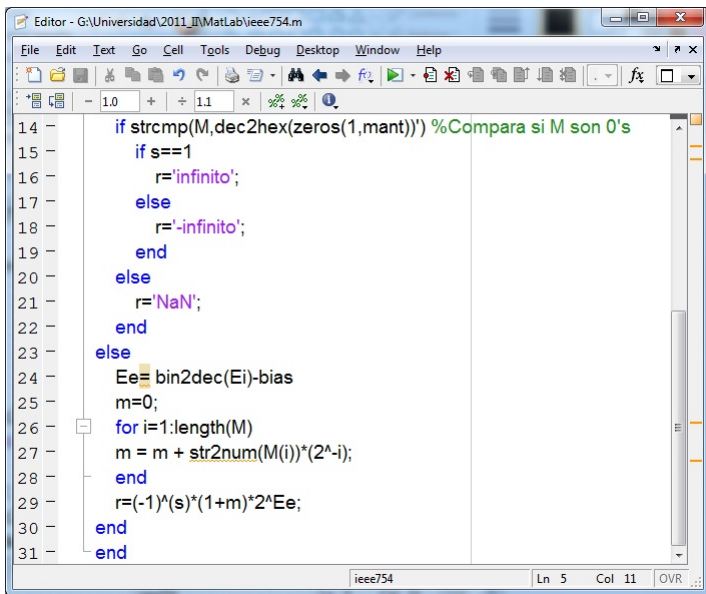
```

1 function r=ieee754(x,k,mant)
2     Ei = x(2:k+1)
3     M = x(k+2:mant+k+1)
4     if (x(1)=='0')
5         s = 1;
6     else
7         s = -1;
8     end
9     if strcmp(Ei,dec2hex(zeros(1,k)))
10        r='Número No Normalizado';
11    else
12        bias = 2^(k-1)-1;
13        if strcmp(Ei,dec2hex(ones(1,k))) %Compara si Ei son 1's
14            if strcmp(M,dec2hex(zeros(1,mant))) %Compara si M son 0's

```

ieee754 Ln 5 Col 11 OVR





The image shows a screenshot of a MATLAB editor window titled "Editor - G:\Universidad\2011_I\MatLab\ieee754.m". The window contains the following MATLAB code:

```
14 - if strcmp(M,dec2hex(zeros(1,mant))) %Compara si M son 0's
15 -     if s==1
16 -         r='infinito';
17 -     else
18 -         r='-infinito';
19 -     end
20 - else
21 -     r='NaN';
22 - end
23 - else
24 -     Ee= bin2dec(Ei)-bias
25 -     m=0;
26 -     for i=1:length(M)
27 -         m = m + str2num(M(i))*(2^(-i));
28 -     end
29 -     r=(-1)^(s)*(1+m)*2^Ee;
30 - end
31 - end
```

The status bar at the bottom of the editor shows "ieee754", "Ln 5", "Col 11", and "OVR".



Ejercicio

En el formato de doble precisión, se pide:

- 1 El epsilon de la maquina. Comparar con el eps de matlab.
- 2 Mayor valor positivo normalizado. Comparar con el realmax de matlab.
- 3 Menor valor positivo normalizado. Comparar con el realmin de matlab.
- 4 el valor de:

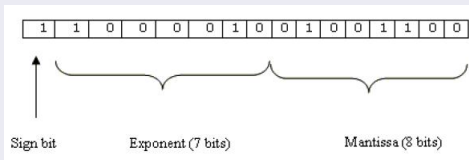
010000000110

0001000001111001010110000001000001100010010011011101



Ejercicio:

Vamos a considerar un hipotético computador que en números de punto flotantes están representados en una palabra de 16-bit. Un ejemplo se muestra en la Figura 1:



Muestre la representación en punto flotante y los bits del:

- 1 El número eps (epsilon de la maquina)
- 2 Mayor valor positivo normalizado
- 3 Menor valor positivo normalizado
- 4 El número 1 y -10.375
- 5 El infinito y NaN