

|                                       |                        |                           |
|---------------------------------------|------------------------|---------------------------|
| CURSO DE ARQUITECTURA DE COMPUTADORAS | Daniel García Martínez | FECHA ENTREGA<br>10/03/17 |
| DATAPATH O RUTA DE DATOS              | TAREA NÚMERO 10        | GRUPO 3CM3                |

## Normalización de un número binario con Radix

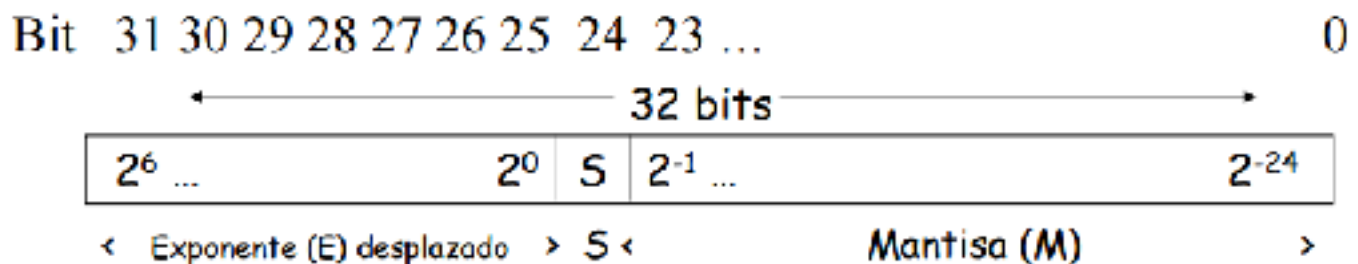
AUTOR: Daniel García Martínez  
INSTITUTO POLITÉCNICO NACIONAL  
ESCOM

### Cómo funcionan los números de punto flotante

La idea es descomponer el número en dos partes:

- Una ***mantisa*** (también llamada coeficiente o significando) que contiene los dígitos del número. Mantisas negativas representan números negativos.
- Un ***exponente*** que indica dónde se coloca el punto decimal (o binario) en relación al inicio de la mantisa. Exponentes negativos representan números menores que uno.

En el formato estándar IEE para un número binario de simple precisión, el bit de signo (S) es el que se encuentra en el bit 24, el exponente (E) incluye los 7 bits de más a la izquierda y la mantisa (M) incluye los restantes 24 bits.



Este formato cumple todos los requisitos:

- Puede representar números de órdenes de magnitud enormemente dispares (limitado por la longitud del exponente).
- Proporciona la misma precisión relativa para todos los órdenes (limitado por la longitud de la mantisa).
- Permite cálculos entre magnitudes: multiplicar un número muy grande y uno muy pequeño conserva la precisión de ambos en el resultado.

Los números de coma flotante decimales normalmente se expresan en notación científica con un punto explícito siempre entre el primer y el segundo dígitos. El exponente o bien se escribe explícitamente incluyendo la base, o se usa una e para separarlo de la mantisa.

| Mantisa | Exponente | Notación científica | Valor en punto fijo |
|---------|-----------|---------------------|---------------------|
| 1.5     | 4         | $1.5 \cdot 10^4$    | 15000               |
| -2.001  | 2         | $-2.001 \cdot 10^2$ | -200.1              |
| 5       | -3        | $5 \cdot 10^{-3}$   | 0.005               |
| 6.667   | -11       | 6.667e-11           | 0.0000000000667     |

## El estándar

Casi todo el hardware y lenguajes de programación utilizan números de punto flotante en los mismos formatos binarios, que están definidos en el estándar IEEE 754.

Presenta dos precisiones:

- Precisión Sencilla o Simple Precisión (32 bits, es decir dos palabras de 16 bits)
- **Doble Precisión (64 bits, es decir cuatro palabras de 16 bits).**

| Formato            | Bits totales | Bits significativos | Bits del exponente | Número más pequeño     | Número más grande    |
|--------------------|--------------|---------------------|--------------------|------------------------|----------------------|
| Precisión sencilla | 32           | 23 + 1 signo        | 8                  | $-1.2 \cdot 10^{-38}$  | $3.4 \cdot 10^{38}$  |
| Precisión doble    | 64           | 52 + 1 signo        | 11                 | $-5.0 \cdot 10^{-324}$ | $1.8 \cdot 10^{308}$ |

Aparición frecuente de la representación interna en hexadecimal. La base usada en el estándar IEEE es 16.

Hay algunas peculiaridades:

- La secuencia de bits es primero el bit del signo, seguido del exponente y finalmente los bits significativos.
- El exponente no tiene signo; en su lugar se le resta un desplazamiento (127 para sencilla y 1023 para doble precisión). Esto, junto con la secuencia de bits, permite que los números de punto flotante se puedan comparar y ordenar correctamente incluso cuando se interpretan como enteros.
- Se asume que el bit más significativo de la mantisa es 1 y se omite, excepto para casos especiales.
- Hay valores diferentes para cero positivo y cero negativo. Estos difieren en el bit del signo, mientras que todos los demás son 0. Deben ser considerados iguales aunque sus secuencias de bits sean diferentes.
- Hay valores especiales no numéricos (NaN, «not a number» en inglés) en los que el exponente es todo unos y la mantisa no es todo ceros. Estos valores representan el resultado de algunas operaciones indefinidas (como multiplicar 0 por infinito, operaciones que involucren NaN, o casos específicos). Incluso valores NaN con idéntica secuencia de bits no deben ser considerados iguales.