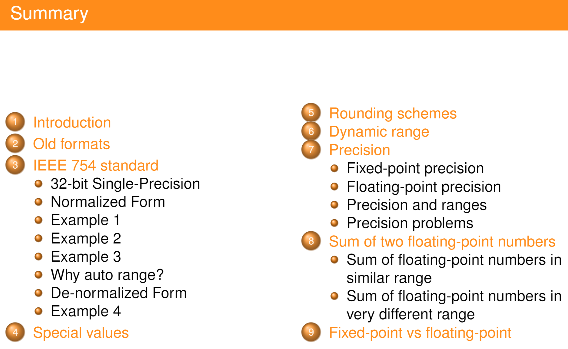
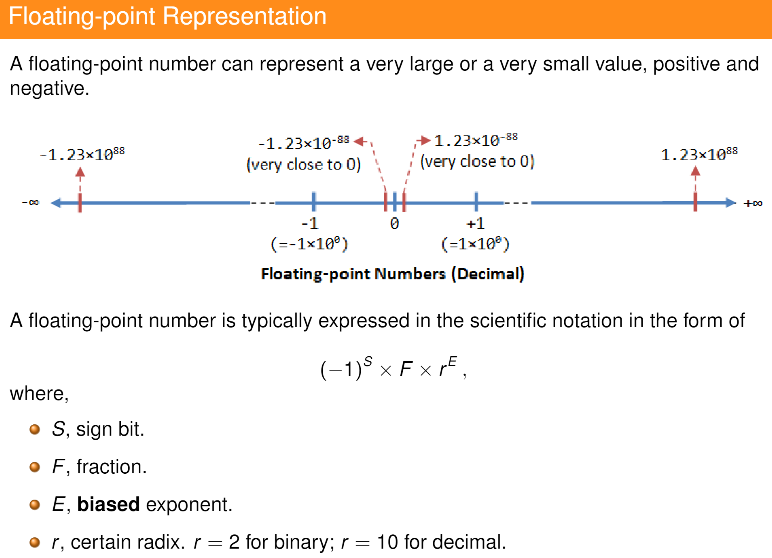
**Representación finita de números reales en formato**

**“*Punto Flotante”***

Vamos a hablar de cómo se representan números reales en una computadora en formato Punto Flotante.





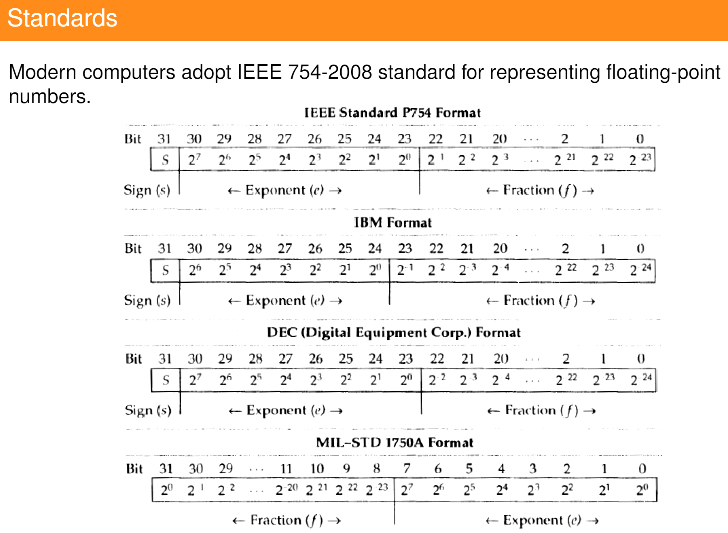
Característica mas importante, es que puede representar una gran cantidad de numeros. Muy grandes y muy pequeños.

Esto se logra representando los números con esta forma.

Signo

Parte Fraccional o Mantisa

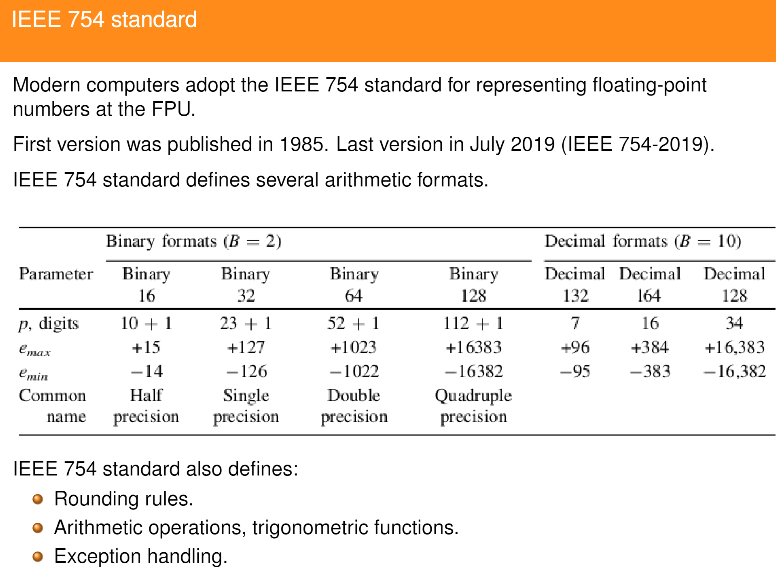
Base y Exponente.



Se empezó a usar a mediados de los 70s.

Cada fabricante de PC creaba su propio estándar. Lo cual traía inconvenientes.

Ej: Fabricante de compilador tenia que hacerlo compatible para todo tipo de computadora.



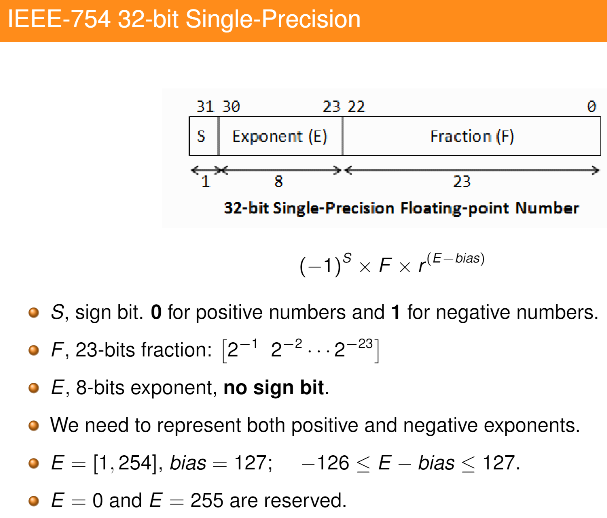
La industria se dio cuenta que había que avanzar hacia un estándar.

El estándar también cubre:

-Esquemas de redondeo

-Operaciones aritméticas y funciones trigonométricas.

-Manejo de excepciones.



Nos enfocamos en el de formato de 32 bits de precisión simple, porque es el más usado en el contexto de sistemas embebidos.

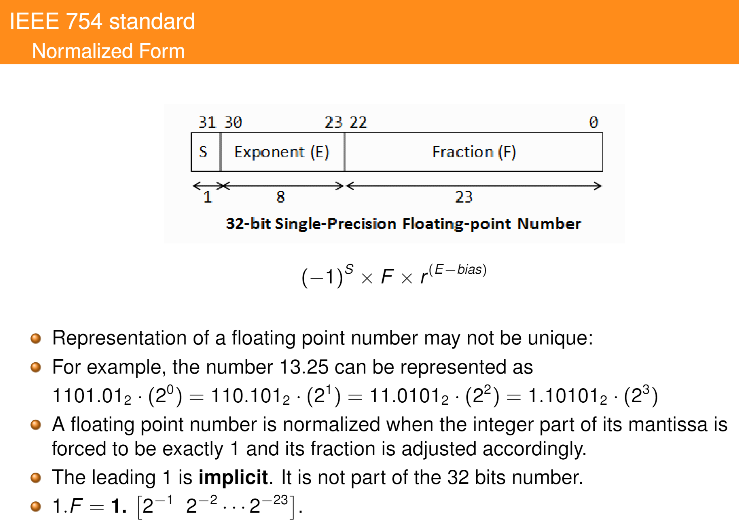
Diagrama de los bits. Palabra de 32 bits.

La mantisa o parte fraccional representa potencias negativas de 2.

E puede valer entre 1 y 254.

El sesgo vale 127. Lo que me permite representar números que tengan un exponente entre -126 y 127.

Esto es un artilugio que se implementa el estándar para evitar ponerle un bit de signo al exponente.

Este formato para representar un numero tiene un inconveniente.

Podemos representar el mismo numero de distintas maneras.

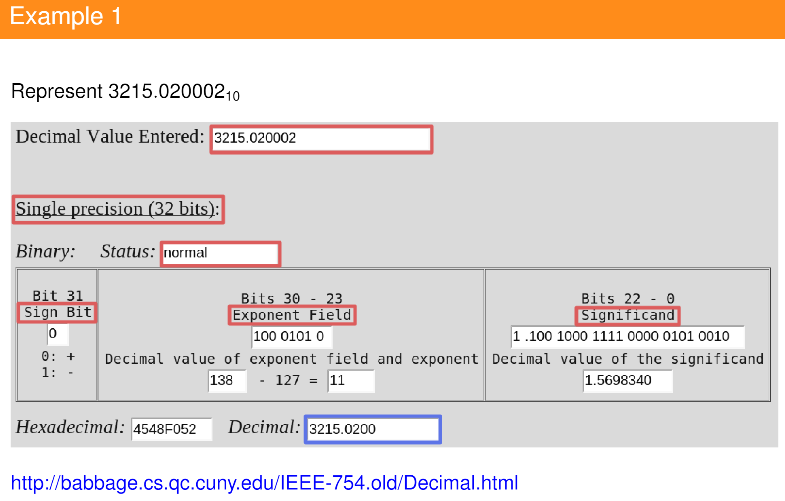
Esto es un problema.

Entonces lo que se establece, es que la única representación valida es la que empieza con “1.” … algo .. el resto de la parte fraccional.

Este “1.” Se toma implícito en el campo F de la mantisa. No es parte de los 32 bits.

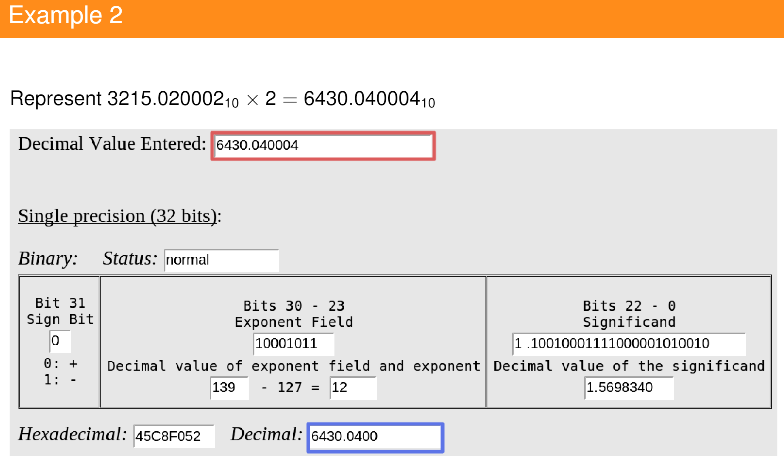
Con F se representa el número “1. + las potencias negativas de 2”. Dese la -1 a la -23.

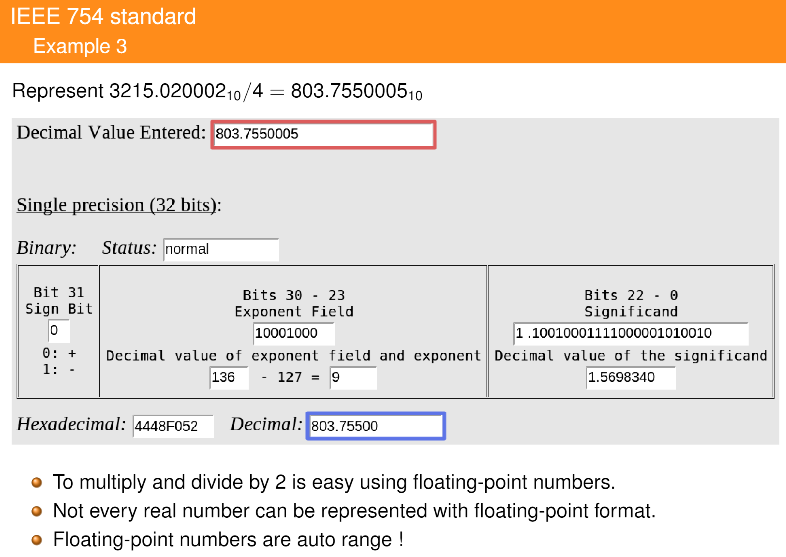
Esta manera de representar el numero en punto flotante la IEEE la llama Forma NORMALIZADA.



Si comparamos el numero original con el que puedo representar realmente con este formato punto flotante, se observa que se perdieron algunos decimales.

1.5698340 x 2^11





Se observa que lo único que varia es el exponente.

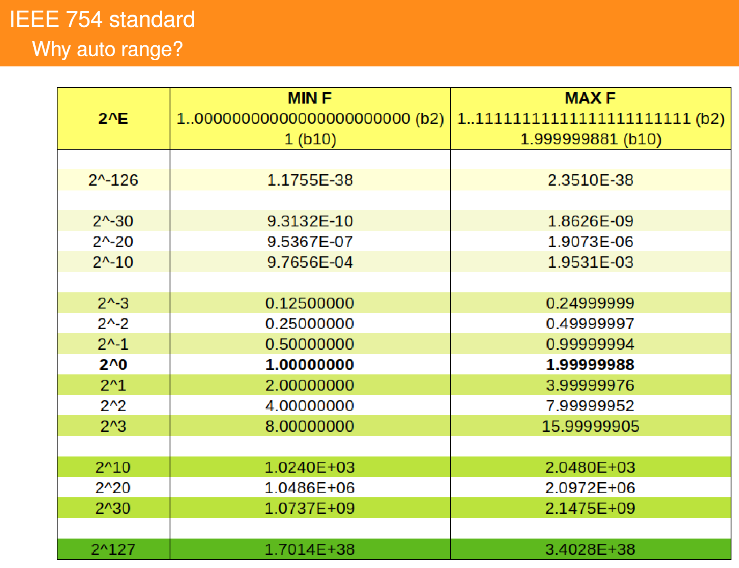
Y que la representación no es exactamente la misma.

Conclusiones:

-Tanto en P Flotante como fijo es fácil multiplicar y dividir por 2.

-No se pueden representar todos los números reales en formato punto flotante.

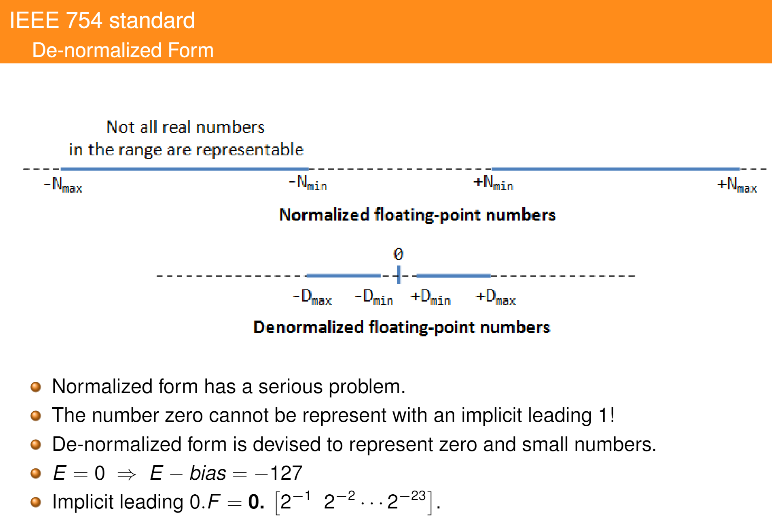
-Son AUTO RANGO!

¿Por qué se dicen que son auto rango?

Lo que pasa es que cunado variamos el valor del exponente, lo que estamos haciendo es movernos por diferentes tramos de la recta de los números reales.

Vemos que, tomando un valor fijo de exponente, tenemos un valor mínimo y máximo de F, estos 3 valores van a definir un rango en el cual estoy operando mientras no modifique el valor del exponente.

Tiene que quedar claro que a medida que pasamos de exponentes muy negativos, a exponentes muy positivos, lo que estamos haciendo es tomar la recta de los reales por pedazos. Y, además, también, estos tramos van creciendo en tamaño.

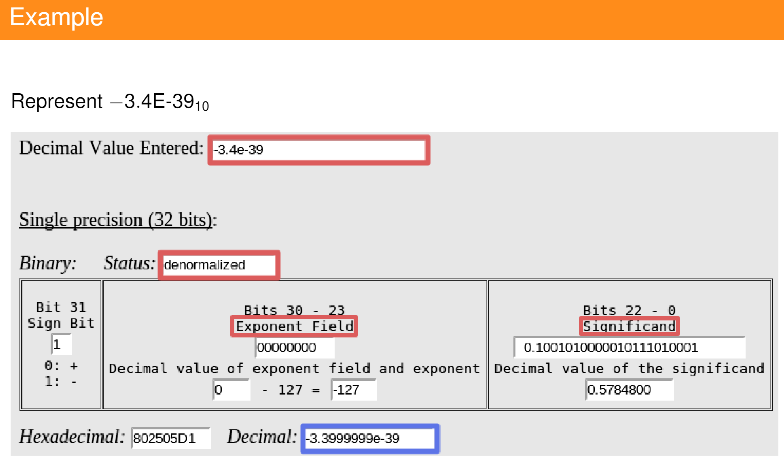


Habíamos dicho que el formato tiene un 1 implícito. (Forma normalizada)

Esta representación tiene un problema, con este formato, no se puede representar el cero.

Por lo que, para solucionar este inconveniente, se considera, que cuando el E=0, en lugar de un 1, el numero tiene un 0 implícito.

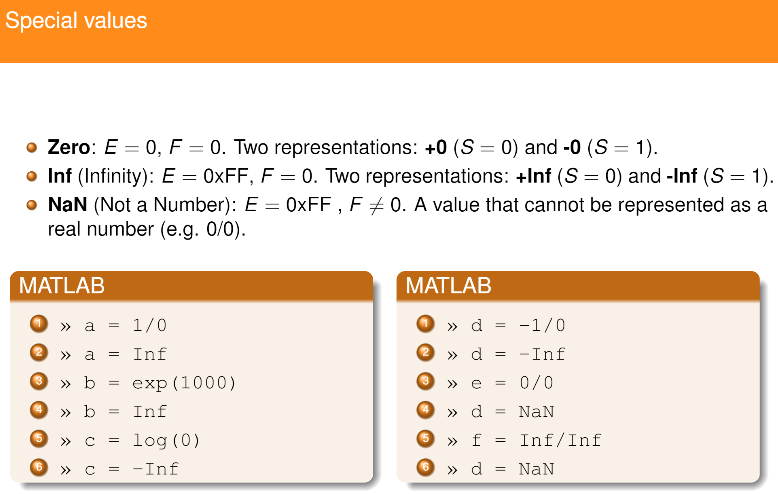
Este formato, se lo conoce como forma DESNORMALIZADA.



Ejemplo, Tiene que ser un numero muy pequeño, ya que en formato desnormalizado se representan los números alrededor del cero.

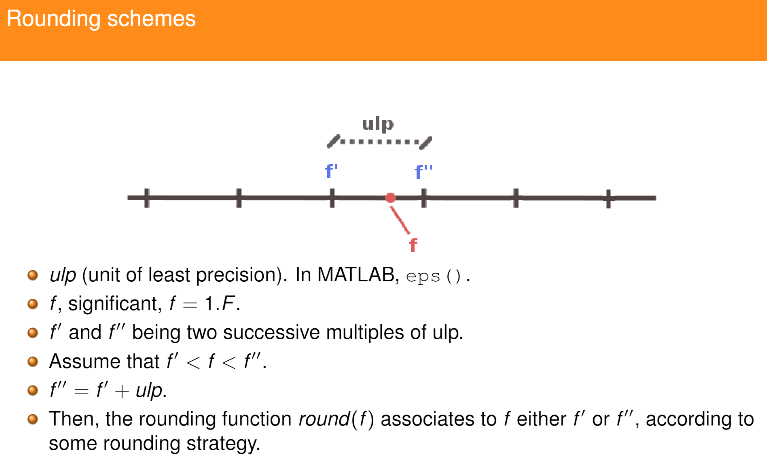
Se observa que el numero que se puede representar no es igual al original, pero está bastante cerca.

Valores Especiales.

La norma de la IEEE contempla 3 valores especiales: 0, infinito y NaN.

NaN: estar reservado para operaciones que no tienen un valor cuando uno lo implementa en una computadora. Ej: 0/0.

Estas operaciones nos dan el mismo valor en una computadora, que si lo resolviéramos con papel y lápiz.

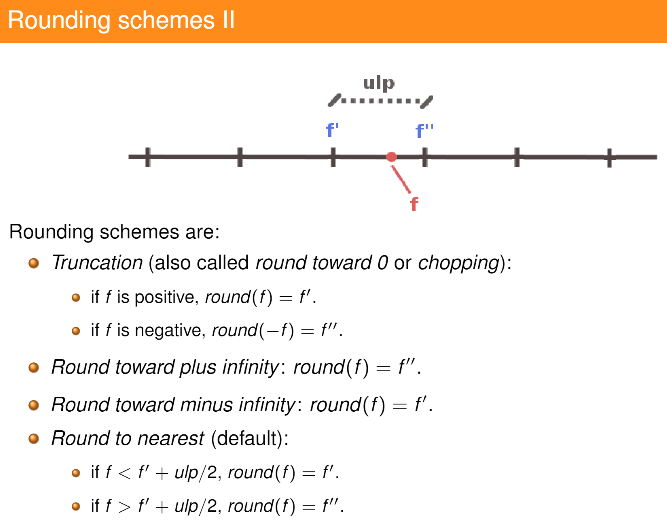
Esquemas de Redondeo:

Como es de esperar el estándar punto flotante también cuenta con esquemas de redondeo, como todo tipo de variable numérica que se representa en una computadora.

En la grafica aparecen variables que vamos a utilizar para definir los esquemas de redondeo.

ULP: lo que llamamos precisión en punto fijo.

El estándar de la IEEE soporta 4 esquemas de redondeo.



Redondeo hacia:

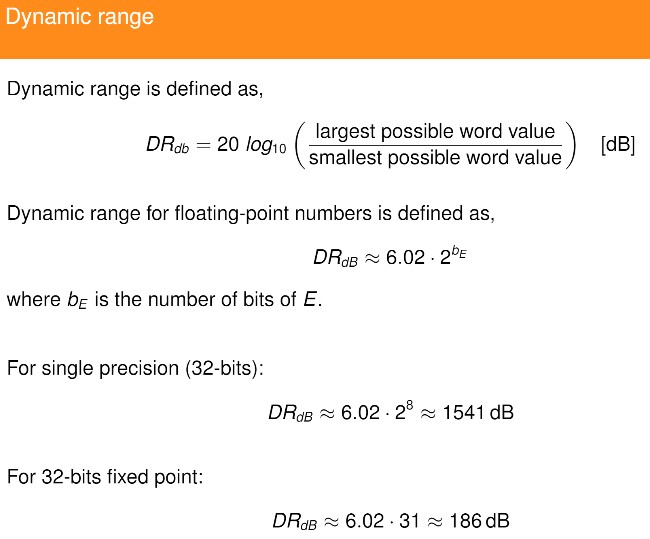
- cero.

- + infinito

- - infinito

- al mas cercano.

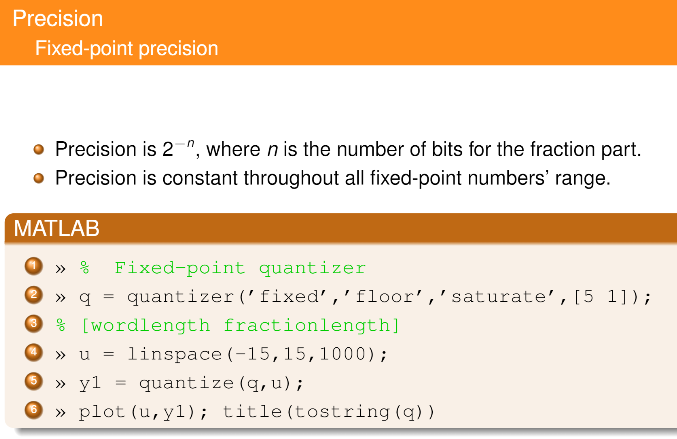
Método utilizado por defecto por la FPU cuando arranca una PC.

Rango Dinámico:

Es un número que nos indica la cantidad de números que podemos representar en una computadora con un formato determinado.

VENTAJA: Con la misma cantidad de bits, en P. Flotante podemos representar muchos más números respecto a P. Fijo.



Precisión en punto Flotante:

-Definición

-Expresión matemática

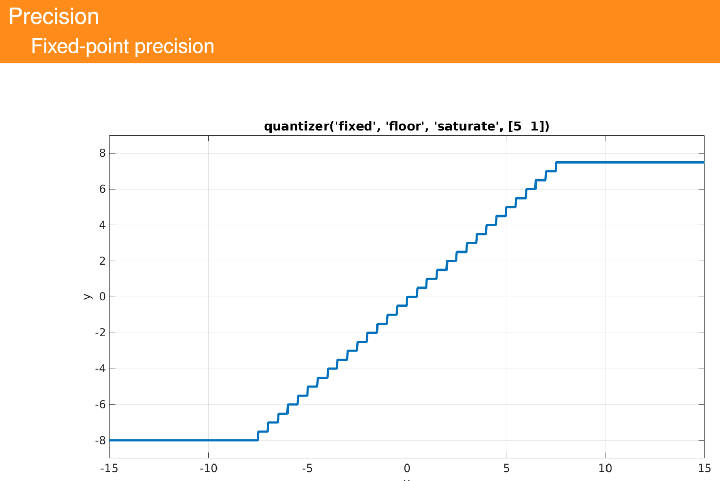
-Característica

Vemos 1ro P.Fijo para poder comparar.

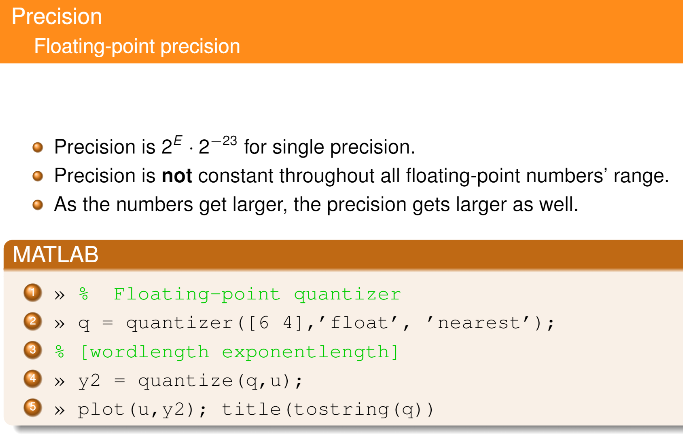
Precisión es la distancia que hay entre 2 posibles números dentro de un sistema de numeración.

Constante mientras no se modifique “n” en P. Fijo

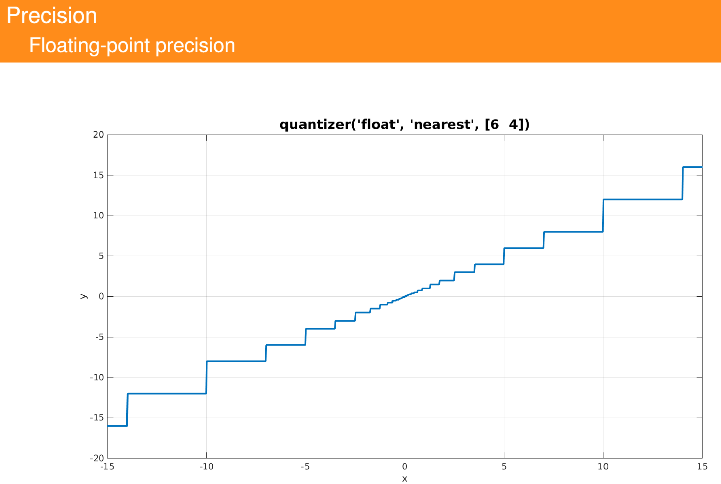
Quantizer: Es una función que recibe un numero P. Flotante de 64 bits. Y no devuelve un numero con otro tipo de precisión.

Definimos un numero en P. Fijo, con redondeo a menos infinito y que aplique aritmética de saturación, Q3.1

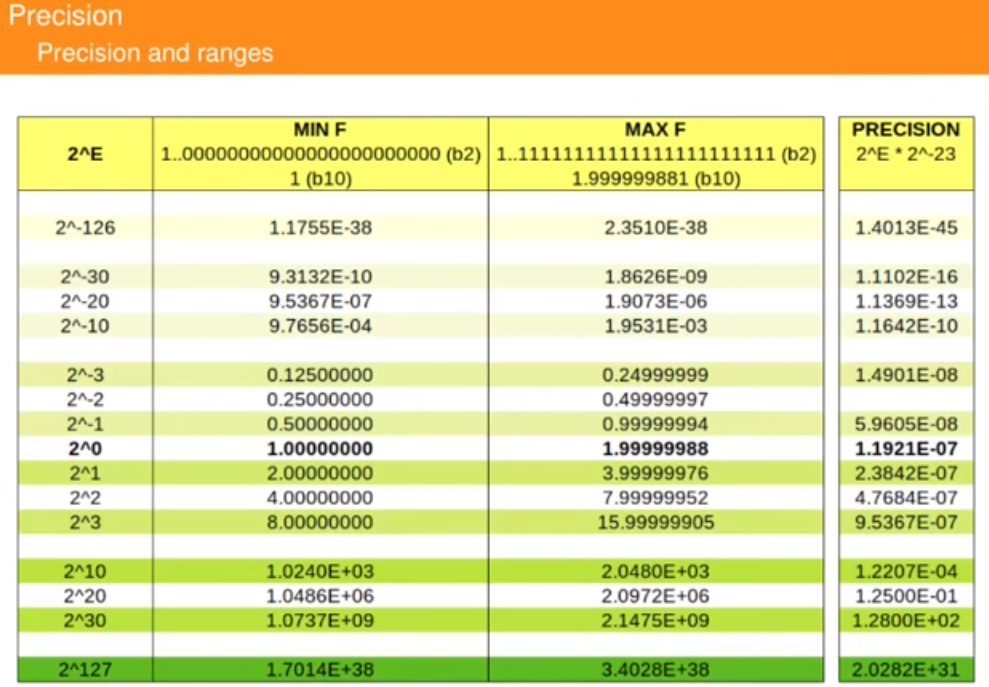
Vector u, con 1000 elementos cuyo valor va entre -15 y 15

En P. Flotante se define la precisión como 2^E x 2^-(Cantidad de bits de F)

La precisión no es constante

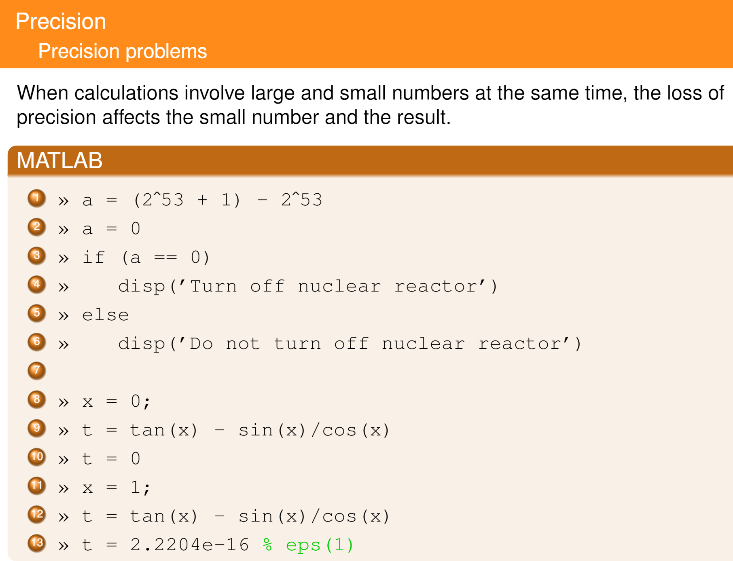


Cerca del cero el valor de la precisión es muy pequeño y aumenta a medida que nos alejamos.



Observamos que la precisión está en función del valor del exponente, para cada rango de números p=ctte.

Pero varia junto con el exponente.

Que la precisión no sea constante y tenga una variación tan impórtate va a tener impacto cuando operemos aritméticamente con números en P. Flotante.

Matlab trabaja con float de 64 Bits

Ejemplos.

Hay que tener cuidado cunado operamos en punto flotante con cantidades muy diferente en valor.

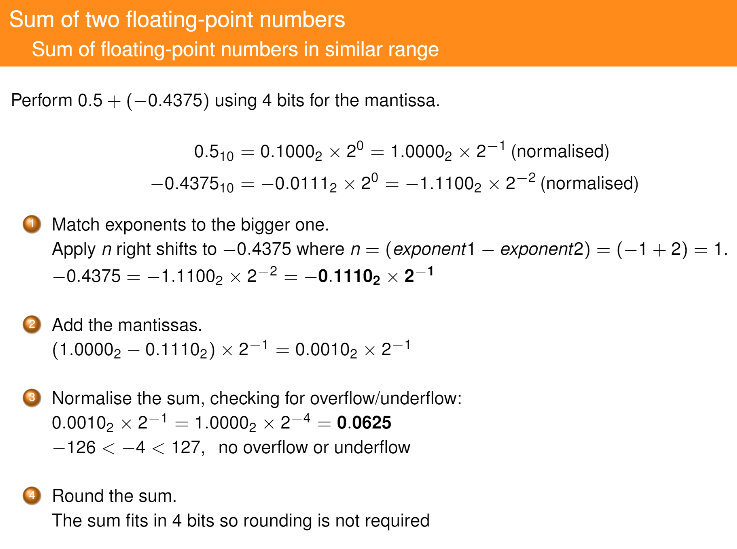
También cuando se usa en estructuras de bifurcación.

2do ejemplo. Con funciones trigonométricas.

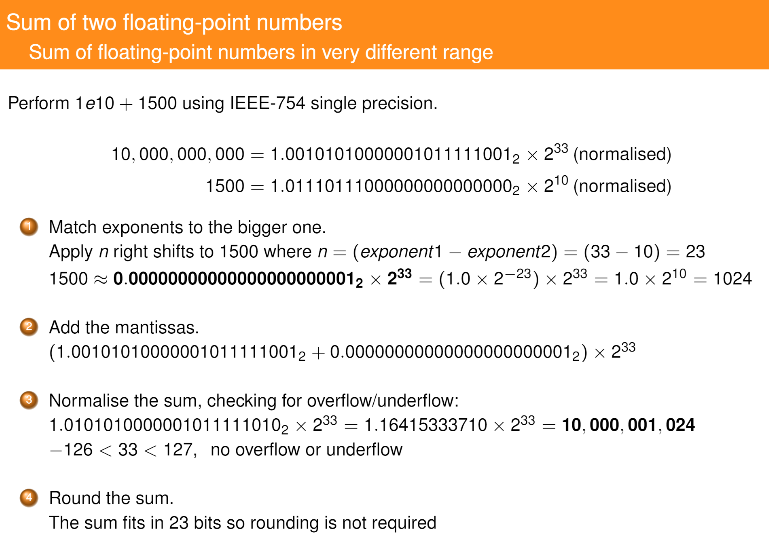
Debería dar 0 en ambos casos.

Hay que tener cuidado.

Suma en P. Flotante

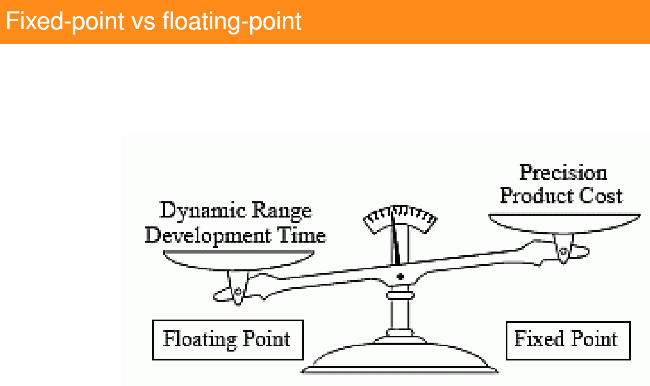
Vamos a ver la suma de 2 números y que consecuencias tiene que la precisión no sea la misma para números con diferentes exponentes.

1ro igualo los exponente al mayor de ambos



Lo que se hace al aumentar el exponente, es forzar a representar el numero 1500 en el rango de números, definido por 2^33. En este caso se comete un error, ya que pasamos de tener el numero 1500 al 1024, que es lo que se puede representar en ese rango.

Problemas asociados a operar con exponente diferentes.



Cuando se trabaja en DPS siempre surge la pregunta si se debe trabajar en Punto Fijo o Flotante.

Cada sistema tiene sus ventajas y desventajas.

Punto flotante tiene un gran rango dinámico y tal vez el tiempo de desarrollo es menor, ya que el programador no tiene que verificar si se produjo un Overflow o Underflow, ya que difícilmente suceda.

Punto fijo: La precisión es constante en todo el rango de números a representar y generalmente los microprocesadores que solo soportan P. Fijo suelen ser mas baratos.

Cortex M3 sin FPU $2 dólares … M4 con FPU - $4 dolares … El doble