

ETC-Tema-4.pdf



TurbApuntPatata



Estructura de Computadores



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingeniería Informática
Universidad Politécnica de Valencia

BBVA

1/6
Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

Ábrete la Cuenta Online de BBVA y llévate 1 año de **Wuolah PRO**

cómo??



Ventajas Cuenta Online de BBVA

0€
Sin comisión de administración o mantenimiento de cuenta.
(0 % TIN 0 % TAE)

0€
Sin comisión por emisión y mantenimiento de Tarjeta
Aqua débito.

0
Sin necesidad de domiciliar nómina o recibos.

Las ventajas de **WUOLAH PRO**

✖ Di adiós a la publi en los apuntes y en la web

✖ Descarga carpetas completas de un tirón

✖ Acumula tickets para los sorteos

cómo???

**1/6**

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

ESTRUCTURA DE COMPUTADORES

Tema 4

ventajas**PRO**

Di adiós a la publi en los apuntes y en la web



Acumula tickets para los sorteos



Descarga carpetas completas

estudia sin publi
WUOLAH PRO



TEMA 4

Aritmética en coma flotante

Representación de números

La representación del computador es limitada y no siempre es exacta: Con 32 bits se pueden obtener 2^{32} palabras diferentes. Por tanto, como máximo se pueden representar 2^{32} valores del conjunto R (Los reales)

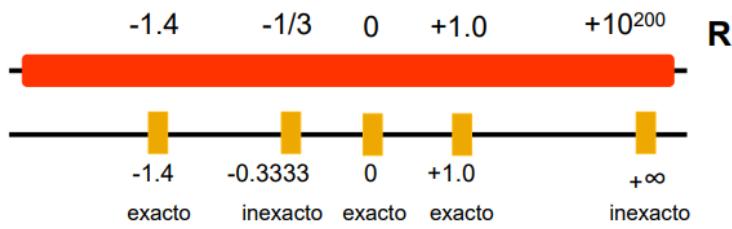
Representación exacta: Hay **números reales** que tienen **representación exacta**, *con 32 bits se puede expresar su valor*, mientras que **otros que no**, *como los números con parte decimal periódica*

Codificación: Se Aplica un **formato arbitrario**, como el **IEEE 754**, estructurado en tres *campos de bits para el signo, el exponente y la parte significativa (mantisa)*

Representación: El formato impone más **restricciones a la representación**: *habrá algunas palabras de bits con una significación matemática especial, como el valor infinito o el cero*

En el computador interesa aumentar La cantidad de **números representados** (densidad) y El **rango** de la **representación**

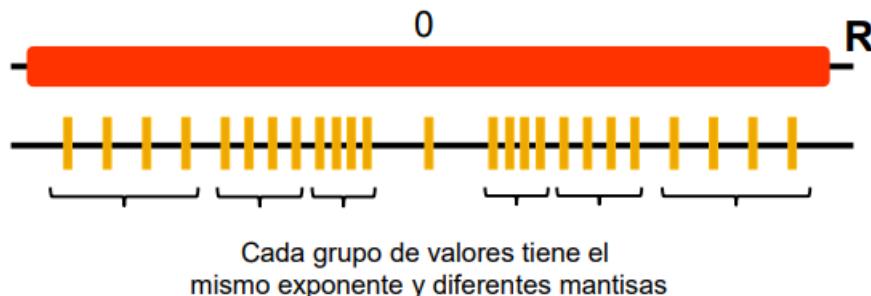
Estos dos aspectos dependen de los campos de la parte significativa (mantisa) y del exponente del formato



Patrón de la representación de los números reales

No hay valores representados muy **cerca del cero**, y Para un mismo valor de exponente los **números representados están separados por la misma distancia**

Cuanto más grande es el exponente **más distancia hay entre dos números representados consecutivos** (*la densidad de representación disminuye*) y se puede representar menos números de ese rango/tamaño





1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

Ábrete la Cuenta Online de BBVA y llévate 1 año de Wuolah PRO



cómo??

Las ventajas de Wuolah PRO



Di adiós a la publi en
los apuntes y en la web



Descarga carpetas
completas de un tirón



Acumula tickets
para los sorteos

Ventajas Cuenta Online de BBVA

0€

Sin comisión de administración o
mantenimiento de cuenta.
(0 % TIN 0 % TAE)

0€

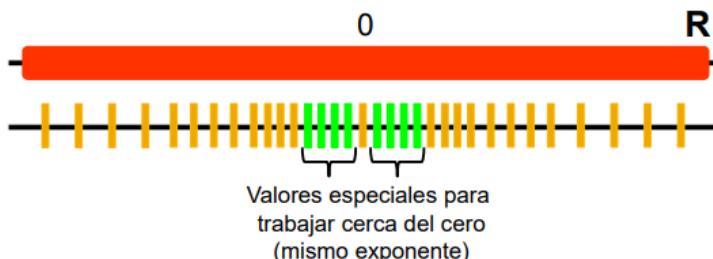
Sin comisión por emisión y
mantenimiento de Tarjeta
Aqua débito.

0

Sin necesidad de domiciliar
nómina o recibos.

Los valores cercanos al cero

El formato IEEE 754 **reserva un subconjunto de palabras** de bits para representar **números reales cerca del cero** y que se interpretan de forma diferente del resto de valores (*valores desnormalizados*). No todas las unidades de coma flotante soportan este subconjunto



La norma IEEE 754

Alcance de la norma

La norma IEEE 754, y su ampliación de punto fijo IEEE 854, especifican:

Codificación: Cómo representar los números en **diversos formatos** (*precisiones simple, doble y extendida, SP DP EP*) y el tratamiento de casos particulares:

NaN (Not a Number), **$\pm\infty$** (infinity), **0** (zero)

Modos de funcionamiento: Ej: el método de redondeo aplicable durante los cálculos

Conjunto de operaciones: Operaciones que se puedan **implementar en el hardware** o en forma de **bibliotecas**

Excepciones: El soporte que ha de dar el **sistema de excepciones** de los procesadores (para que se puedan diseñar buenas bibliotecas de cálculo numérico)

Representación

Símbolos

- **S** es el **signo**
 - **M** la **magnitud de la mantisa**
 - **E** el **exponente**
 - **Simple precisión (SP)**
 - **Doble precisión (DP)**
- | | | |
|---|---|----|
| 1 | 8 | 23 |
| S | E | M |
- $$(-1)^S \cdot 1.M \cdot 2^{E-127}$$
-
- | | | |
|---|----|----|
| 1 | 11 | 52 |
| S | E | M |
- $$(-1)^S \cdot 1.M \cdot 2^{E-1023}$$

Valores desnormalizados (SP y DP)

S	000..00	M ≠ 0	$(-1)^S \cdot 0.M \cdot 2^{-126}$
S	000..00	M ≠ 0	$(-1)^S \cdot 0.M \cdot 2^{-1022}$

± 0 **S** **000..00** **000..00**

$\pm\infty$ **S** **111...11** **000..00**

Nan **x** **111...11** **M ≠ 0**

Valores especiales (SP y DP)

0: Da igual el signo (hay 2 0s), todo a 0s, incluso la mantisa

$\pm\infty$: Todo a 1, la mantisa a 0s, y el signo te dirá si + o -

NaN: Todo a 1, la mantisa diferente de 0, el signo te da igual

Los valores especiales

Son manipulados por las operaciones **junto con los reales corrientes**

Cero e infinito: se entienden como límites matemáticos, por eso

- $+\infty + +\infty = +\infty - \infty + -\infty = -\infty \dots$
- $+\infty \cdot \text{positivo} = +\infty + \infty \cdot \text{negativo} = -\infty \dots$
- $\text{positivo} / +0 = +\infty \text{ positivo} / -0 = -\infty \dots$
- **Atención a las comparaciones:** $+0$ y -0 **son iguales**

Not a Number (NaN):

Propagación: cualquier operación donde un **operando es NaN** dará como **resultado NaN**

Generación: NaN es el resultado de $(+\infty) + (-\infty)$, $\pm 0 \cdot \pm \infty$, $\pm 0 / \pm 0$, $\pm \infty / \pm \infty \dots$

Comparación: Una comparación ($=, <, \dots$) entre **NaN** y **otro número** resulta siempre **FALSA**

La norma y los lenguajes de programación: Los **valores especiales** permiten **tratar los incidentes del cálculo**, *El desbordamiento aritmético produce un resultado representable*

El redondeo

Esta es una Situación frecuente, **una operación genera una mantisa M de longitud más larga** (p bits) **que la prevista** en el formato (m bits). *Pues los m primeros bits de la mantisa M se llaman retenidos*

Possibilidades:

- **M es representable** de forma exacta en el formato: *los $p-m$ bits no retenidos son 0 y se pueden eliminar*: $010000 \rightarrow 0100$
- **M se encuentra entre dos valores representables** $M-$ y $M+$ ($M- < M < M+$) **y hay que redondear**: *escoger uno de ellos como representación inexacta de M*

Modos de redondeo: La norma admite cuatro modos de redondeo

- Hacia $+\infty$
- Hacia 0
- Hacia $-\infty$
- Escoger el **más próximo de los dos** (este es el modo por omisión)

El redondeo hacia el más próximo (sesgado al par)

La variante por omisión es “*tie to even*”: en caso de que **M equidiste de $M-$ y $M+$** hay que **escoger la mantisa representable par** (o sea, la que acabe en 0)

M	se elige	M resultante
010000	(exacta)	0100
010001	M_{-} (más próxima)	0100
010010	M_{-} (par)	0100
010011	M_{+} (más próxima)	0101
010100	(exacta)	0101
010101	M_{-} (más próxima)	0101
010110	M_{+} (par)	0110
010111	M_{+} (más próxima)	0110
011000	(exacta)	0110

cómo???



1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

ventajas

PRO



Di adiós a la publicidad en los apuntes y en la web



Acumula tickets para los sorteos



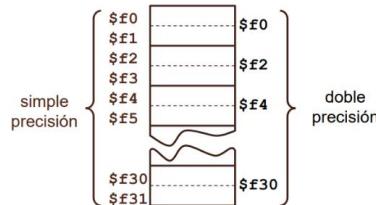
Descarga carpetas completas

estudia sin publicidad
WUOLAH PRO

La coma flotante en el MIPS

Hay 32 registros de 32 bits para tipo **float** los cuales se suelen usar por pares (ya que un número en coma flotante de doble precisión ocupa 64 bits).

De estos tipos **doble** se puede hacer hasta 16 registros al hacer las parejas ($32/2 = 16$)



Intercambio con la memoria y los registros de enteros

Los registros en coma flotante utilizan **instrucciones diferentes** a las de los registros normales, hacen falta instrucciones específicas para la conversión de un tipo a otro (*casting*)

Operación	Instrucción	
lectura $$ft \leftarrow Mem[X+\$rs]$	lwc1 \$ft,X(\$rs)	fs i ft: registros de coma flotante
escritura $Mem[X+\$rs] \leftarrow \ft	swc1 \$ft,X(\$rs)	
transferencia $\$fs \leftarrow \rt	mtc1 \$rt,\$fs	rs i rt: son registros de enteros
transferencia $\$rt \leftarrow \fs	mfc1 \$rt,\$fs	

Lwc1 \$ft (En coma flotante), x(\$rs): *guarda en \$ft el valor en coma flotante de la direc. de rs+x*

Swc1 x(\$rs), \$ft (En coma flotante): *guarda en la direc. Rs+x el valor en coma flotante de \$ft*

Mtc1 y **Mfc1** pasan número de coma flotante a enteros normales y viceversa

Funcionamiento: Carga en memoria un número en coma flotante. La mitad del número en \$f0 y la otra mitad en \$f1. Ya que el número en coma flotante ocupa 64 bits y los registros son de 32, mitad y mitad DOBLE PRECISIÓN

El banco de registros

Hay **32 registros de 32 bits**, \$f0, \$f1, ..., \$f31 para tipo float

Convenio de uso

Nombre del registro	Utilización
\$f0	Retorno de función (parte real)
\$f2	Retorno de función (parte imaginaria)
\$f4, \$f6, \$f8, \$f10	Registros temporales
\$f12, \$f14	Paso de parámetros a funciones
\$f16, \$f18	Registros temporales
\$f20, \$f22, \$f24, \$f26, \$f28, \$f30	Registros a preservar entre llamadas

- Se suelen utilizar los números pares \$f0, \$f2, ..., \$f30

Emparejables para formar 16 registros de 64 bits para tipo double

- Si \$f0 “**contiene**” un **double**: \$f0 tiene la **parte baja** y \$f1 la **parte alta** (\$f1 | \$f0)

Conversión de formatos

Los registros de CF pueden contener:

símbolo	tipo
S	números en coma flotante en SP
D	(por parejas) números en coma flotante en DP
W	números enteros de 32 bits

La instrucción **cvt._.fd,fs** hace las conversiones posibles entre los tres tipos. Donde están las barras bajas le pones los tipos:

Ejemplo **cvt.d.w \$f4,\$f7**. Convierte el **entero** de **\$f7** a una **doble precisión** en **\$f4|/\$f5**

En combinación con las instrucciones de transferencia con el banco de registros de enteros, se puede hacer aritmética con variables de tipos diversos

Instrucciones aritméticas lógicas básicas

Hay **dos versiones** de cada operación: **S** (simple precisión) y **D** (doble precisión)

Aritméticas

operación	instrucción
suma	add._ fd,fs,ft
resta	sub._ fd,fs,ft
multiplicación	mul._ fd,fs,ft
división	div._ fd,fs,ft
comparación	c.cond._ fs,ft
copia	mov._ fd,fs
cambio de signo	neg._ fd,fs
valor absoluto	abs._ fd,fs

Ejemplo

- add.s \$f0,\$f1,\$f2 //suma de floats
- add.d \$f2,\$f4,\$f6 //suma de doubles

Comparación

Las instrucciones de comparación escriben un **bit implícito FPc** que codifica **cierto=1** y **falso=0**

Este **bit** se encuentra en un **registro de control del coprocesador** y puede ser consultado por las **instrucciones de salto**

fd>fs	fd=fs	fd<fs
gt	eq	lt
le	neq	ge

Para cada tipo de datos, hay un conjunto de comparaciones codificables.

Las **más importantes**:
c._.s fd,fs o **c._.d fd,fs**

Control de flujo

- **bc1t eti** si (FPc == 1) bifurcar a eti
- **bc1f eti** si (FPc == 0) bifurcar a eti

```
; mirar si $f0>$f2
c.gt.s $f0,$f2
; saltar si afirmativo
bc1t eti
```

```
; mirar si $f0<=$f2
c.le.s $f0,$f2
; saltar si negativo
bc1f eti
```

- Combinadas con las instrucciones de comparación, permiten bifurcar con condiciones aritméticas complejas.
- Cada **condición** permite **dos implementaciones** según se quiera **comparar** que el bit **FPc** esté a **0** o a **1**

```
Float AreaCirculo(int r)
PI: float 3.14159
Area:    mtc1 $a0,$f0
          cvt.s.w $f0,$f0
          mult.s $f0,$f0,$f0
          la $t0,PI
          lwc1 $f2,0($t0)
          mult.s $f12,$f0,$f2
          jr $ra
```

Operadores de Coma flotante

Los operadores de coma flotante toman como entrada uno o dos operandos en un formato de CF. El resultado dado será codificado en CF según la normal (exceptuando los operandos de comparación)

Cambio de signo NEG.S NEG.D

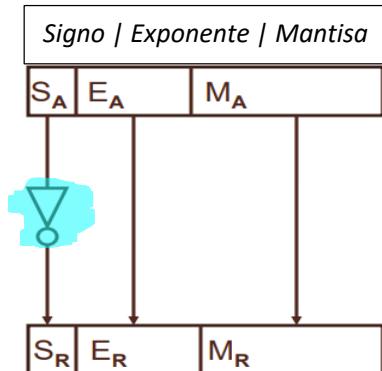
Tiene **Dos versiones**, de simple precisión **NEG.S** y de doble precisión **NEG.D**

Simple

- Entrada:** SA (1 bit), EA (8 bits) y MA (23 bits)
- Salida:** SR (1 bit), ER (8 bits) y MR (23 bits)

Doble

- Entrada:** SA (1 bit), EA (11 bits) y MA (52 bits)
- Salida:** SR (1 bit), ER (11 bits) y MR (52 bits)



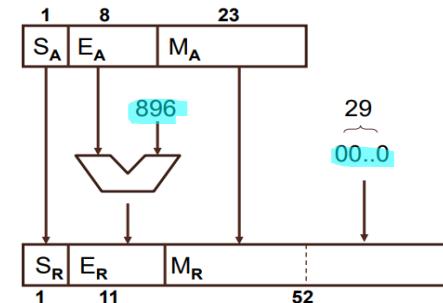
Conversión de simple a doble precisión cvt.d.s

Entrada: SA (1 bit), EA (8 bits) y MA (23 bits) **Salida:** SR (1 bit), ER (11 bits) y MR (52 bits)

- Signo:** *no cambia* → $SR = SA$
- Exponente:** *hay que cambiar de exceso 127 a exceso 1023* → $ER = EA + 896$
- Mantisa:** *hay que añadir 52-23=29 ceros a la derecha* → $MR = MA || 00 \dots 0$

Valores especiales

E _A	S _R	E _R	M _R
zero y subnormal: 00000000_2	S _A	00000000000_2	$M_A 00 \dots 0$
$\pm\infty$ y NaN: 11111111_2	S _A	11111111111_2	$M_A 00 \dots 0$
valores corrientes: (otros valores)	S _A	$E_A + 896$	$M_A 00 \dots 0$

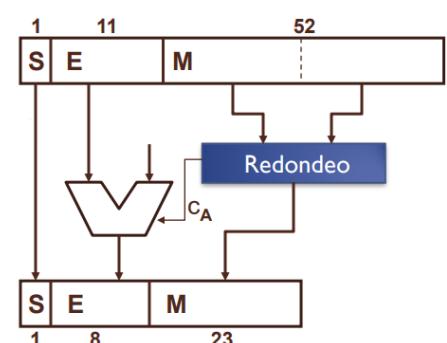


Conversión de doble a simple precisión cvt.s.d

Entrada: SA (1 bit), EA (11 bits) y MA (52 bits) **Salida:** SR (1 bit), ER (8 bits) y MR (23 bits)

- Signo:** *no cambia* → $SR = SA$
- Exponente:** *hay que cambiar de 11 bits en exceso 1023 a 8 bits en exceso 127 por el cambio de tamaño puede desbordar*
- Mantisa:** *hay que eliminar 29 bits por la derecha y redondear*

Hay un circuito para redondear al más próximo, pero no me cabe :p



Conversión de entero a doble precisión cvt.d.w

- Si es **positivo**, un entero W de 32 bits se puede escribir como $+0.Wx2^{32}$.
 - Si es **negativo**, W se reescribe como $-0.(-W)x2^{32}$.
 - La **mantisa W comienza por** una serie de **Z ceros** ($0 \leq Z \leq 32$).
 - Habrá que **normalizar la mantisa desplazándola Z+1 posiciones hacia la izquierda** (eliminando el bit entero) y **restar Z+1 al exponente**

Entrada: **w** (32 bits)

Salida: **SR** (1 bit), **ER** (11 bits) y **MR** (52 bits)

- **Signo:** $SR = Signo(W)$
 - **Exponente:** $ER = 1023 + 32 - Z - 1$
 - **Mantisa:** $MR = |W| << Z + 1$ (*desplazamiento*)
 - Si $W < 0$, hay que hacer $W = -W$ (*cambio de signo*)

Esquema

La normalización ha de contar el número Z de bits a cero por la izquierda (hasta el primer 1), Ha de desplazar la mantisa Z posiciones hacia la izquierda

- Hay que **representar el exponente**: $31 - Z$ y Sumando el exceso, tenemos $E = 1023 + 31 - Z$
 - Hay que **completar la mantisa** con ceros

La normalización

El Circuito de normalización está compuesto por un **codificador prioritario** y un **barrer shifter**

- **Codificador:** Codifica la entrada de menor índice con un 1, calcula Z
 - **Barrel shifter:** Desplaza mantisa hacia izquierda y elimina los ceros sobrantes
 - Por último, se descarta el bit implícito

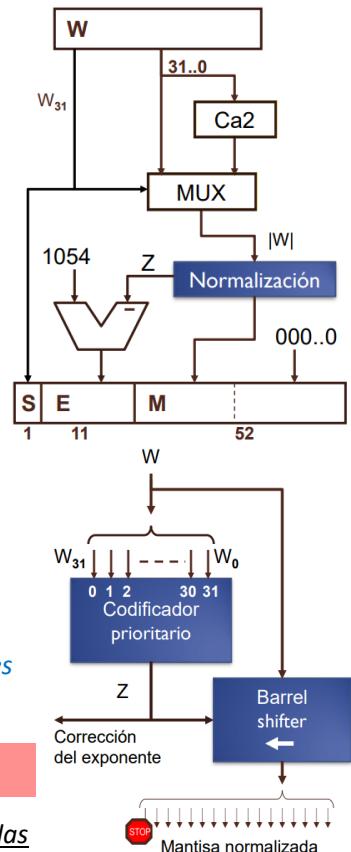
La multiplicación *mul.s* / *mul.d*

Tiene **Dos versiones**, una de **simple** precisión y otra de **doble** precisión y **2 entradas**

Entradas: **SA** (1 b), **EA** (8/11 bs) y **MA** (23/52 bs) || **SB** (1 b), **EB** (8/11 bs) y **MB** (23/52 b)

Salida: SR (1 bit), ER (8/11 bits) v MR (23/52 bits)

- **Signo:** $SR = SA \text{ xor } SB$
 - **Exponente:** Hay que **sumar** y **compensar** el **exceso**
 - Simple precisión $ER = EA + EB - 127$
 - Doble precisión $ER = EA + EB - 1023$
 - **Mantisa:** Multiplicar $1.MA \times 1.MB$ (*considerando el bit implícito de delante "1"*)
 - El multiplicador depende de la precisión: 24×24 bits o 53×53 bits
 - Habrá que renormalizar (quitando el bit implícito) y redondear la mantisa resultante dejándola en $23/52$ bits



cómo???



1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

BBVA está adherido al Fondo de Garantía de Depósitos de Entidades de Crédito de España. La cantidad máxima garantizada es de 100.000 euros por la totalidad de los depósitos constituidos en BBVA por persona.

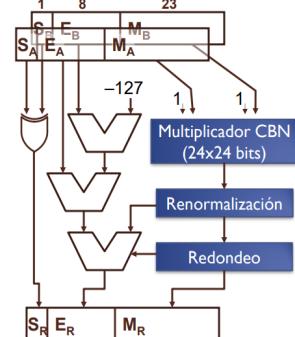
Muls

Dos sumadores para **sumar los exponentes** representados en exceso 127

Multiplicador para las mantisas, Opera con el bit implícito añadido: **24x24 bits**.

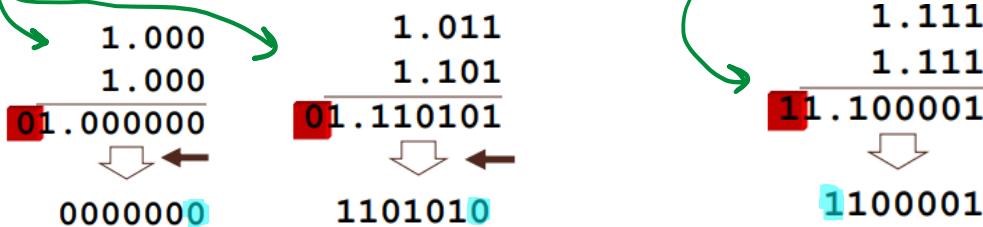
Habrá que **renormalizar el resultado** y **quitar el bit implícito** Tal vez habrá que **decrementar el exponente**

El **resultado ocupará 47 bits** y habrá que redondearlo a 23: **Tal vez habrá que incrementar el exponente**

Renormalización

Renormalización después de un producto

1. Si la **mantisa** del producto **comienza por 0**: *Desplazar a la izquierda una posición*
2. Si la mantisa del producto **comienza por 1**: *Incrementar el exponente en 1*
3. **En cualquier caso**: *Eliminar el bit entero implícito*



ventajas

PRO



Di adiós a la publi en los apuntes y en la web



Acumula tickets para los sorteos



Descarga carpetas completas

estudia sin publi

WUOLAH PRO

Seguidme en wuolah “TurboApuntesPat” hay apuntes de todas las asignaturas, no te arrepentirás juju. Sígueme en twitter @MaikPatataOtaku ;) Y en Instagram también juju @potatproductions :p