Universidad de Jaén

Compromiso con la sociedad

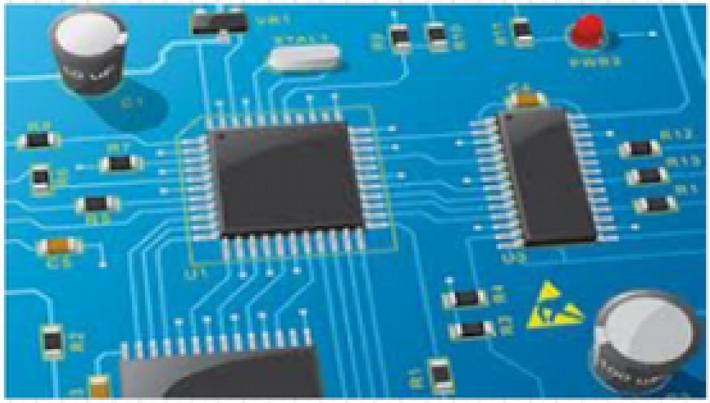
25 AÑOS DE UNIVERSALIDAD 1993-2018

T1. Representación de Información

1.2 Aritmética en Coma Flotante

FUNDAMENTOS DE ARQUITECTURA

DE COMPUTADORES



Departamento de Informática

Contenido del capítulo

- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
  - Suma y Resta
  - Multiplicación y División
- Bibliografía
- Actividades

Contenido del capítulo

- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
  - Suma y Resta
  - Multiplicación y División
- Bibliografía
- Actividades

Aritmética en coma flotante

- La coma flotante no es una representación exacta para reales. Únicamente permite representar números en forma aproximada:
  - Es preciso aplicar cuidadosamente técnicas de redondeo.
  - En los programas no se debe testear igualdad para números en coma flotante.
- La realización de operaciones con datos en coma flotante es compleja: circuitos específicos dedicados a ello.
  - Computadores de potencia baja trabajaban en coma flotante mediante software.
  - 808x: coprocesadores de coma flotante.
  - Actualmente los coprocesadores de coma flotante se integran en el chip del microprocesador.

Aritmética en coma flotante (2)

- **Notación científica normalizada**
  - $0.000000001 = 1.0 \times 10^{-9}$
  - $3155760000 = 3.15576 \times 10^{9}$
- Todo número x se representa en función de Signo, Mantisa (q), y Exponente (e)
  - Ejemplo: $x = \pm q \cdot B^{e}$
  - Ejemplo2: $123409 = 0.123409 \times 10^{6} = +,123409,6$
- En informática se utiliza siempre la base 2:
  - Sólo hay un dígito no nulo a la izquierda de la coma binaria: $1.xxxx \cdot 2^{yyyy}$

Contenido del capítulo

- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
    - Suma y Resta
    - Multiplicación y División
- Bibliografía
- Actividades

Normalización IEEE 754

- Precisión simple (32 bits):

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S	EXPONENTE								MANTISA																						

  - Signo (1 bit):
    - 0 = positivo
    - 1 = negativo

Normalización IEEE 754

- Precisión simple (32 bits):

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>S</td><td colspan="8">EXPONENTE</td><td colspan="23">MANTISA</td></tr>
</table>

- Utilizamos un formato normalizado:
  - $0.232 \cdot 10^3 = 23.2 \cdot 10^1 = \mathbf{2.32 \cdot 10^2}$
  - $01001 = 1.001 \cdot 2^3$
- Mantisa IEEE: 23 bits + 1 bit implícito.
- El **bit implícito** siempre vale 1 y no se almacena:
  - m = 1001.11000 110 · 2^-5 = 1.0011 1001 10 · 2^-2
  - m= 0.0000 0110 1101 · 2^34 = 1.1011 01 · 2^28
- En el primer caso M = 0011 1001 1000 0000 …
- En el segundo caso M = 1011 0100 0000 …

Normalización IEEE 754

- Precisión simple (32 bits):

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S	EXPONENTE								MANTISA																						

- Exponente (8 bits), expresado en binario **<u>sesgado</u>**:
  - $E = S + e = 2^{(|e|-1)}-1 + e$
- Con 8 bits se pueden representar (con signo) números del -127 a 128. El sesgo es por tanto 127
- Ejemplos:
  - Si el exponente es 4, el campo "E" será $4 + 127 = 131$ ($10000011_{]2}$).
  - Si E contiene 01011101 ($93_{]10}$), el exponente real es $93 - 127 = -34$.
- Guardar un exponente sesgado permite comparar valores directamente, para comprobar qué número es mayor.

Normalización IEEE 754

- Precisión doble (64 bits):
  - Signo (1 bit).
  - Mantisa (52 bits + 1 bit implícito).
  - Exponente (11 bits); binario sesgado a 1023
- Permite mayor precisión en la representación
- Necesita un mayor tamaño de palabra para el almacenamiento

Casos especiales

$$N = \begin{cases} (-1)^s \times 1.m \times 2^{exp-127} & \text{si } 0 < exp < 255 \\ (-1)^s \times 0.m \times 2^{-126} & \text{si } exp = 0 \text{ (caso desnormalizado)} \\ 0 & \text{si } exp = m = 0 \\ +\infty & \text{si } exp = 255,\ m = 0,\ s = 0 \\ -\infty & \text{si } exp = 255,\ m = 0,\ s = 1 \\ \text{NaN ("\underline{N}ot \underline{a} \underline{N}umber")} & \text{si } exp = 255,\ m \neq 0 \end{cases}$$

- s = signo de la mantisa
- m = módulo de la mantisa almacenada (sin bit implícito)
- exp = exponente almacenado (sin descontar sesgo/exceso)

Ejemplo: Base 10 → IEEE 754

- **$N = –0.75_{10}$**

1. Expresarlo en binario: $–0.75_{10} = –0.11_2$
2. Normalizar: $–0.11_2 = –1.1_2 \times 2^{–1}$
3. Cálculo del exponente sesgado:

   $exp = –1_{10} + 127_{10} = 126_{10} = 01111110_2$
4. Cálculo de mantisa: m = 1.10000000000000000000000 (el bit implícito no se almacena)
5. Bit de signo: s = 1 (nºnegativo)
6. Representación en notación compacta hexadecimal:

   $N = 1011\ 1111\ 0100\ 0000\ 0000\ 0000\ 0000\ 0000_2$ = 0xBF400000

Ejemplo 2: Base 10 → IEEE 754

- **$N = -543.7 \cdot 10^{-17}$**
  - $N = -0.5437 \cdot 10^{-14}$
    - $10^{-14} = 2^x$;
    - $x = -14 \cdot [\log(10)/\log(2)] = -14/0.301029995 = -46.50699333$
  - $N = [-0.5437 \cdot 2^{(-0.50699333)}] \cdot 2^{-46}$
    - $= -0.382594862 \cdot 2^{-46}{}_{10}$
    - $= -0.61F1BCA_H$
  - Normalizando en binario:
    - N = - 0.0110 0001 1111 0001 1011 1100 1010 · 2^-46
    - N = - 1.1000 0111 1100 0110 1111 001 · 2^-48
  - Signo = 1 (negativo)
  - Exponente = S+e = 127 – 48 = 79_{10} = 0x4F = 0100 1111
  - Finalmente, la representación de $-543{,}7 \cdot 10^{-17}$ =
  - 1 | 0100 1111 | 1000 0111 1100 0110 1111 001
  - En formato hexadecimal empaquetado → 0xA7C3E379

Ejemplo: IEEE 754 → Base 10

- **$N = C0A00000_{16} =$**
- **$1100\ 0000\ 1010\ 0000\ 0000\ 0000\ 0000\ 0000_2$**

1. Bit de signo: s = 1 (nºnegativo)
2. Exponente: exp $= 10000001_2 = 129_{10} \rightarrow 129_{10} - 127_{10} = 2_{10}$
3. Mantisa: $1.01000000000000000000000_2 = 2^0 + 2^{-2} = 1.25_{10}$
4. Resultado: $\mathbf{N = -1.25 \times 2^2 = -5_{10}}$

Contenido del capítulo

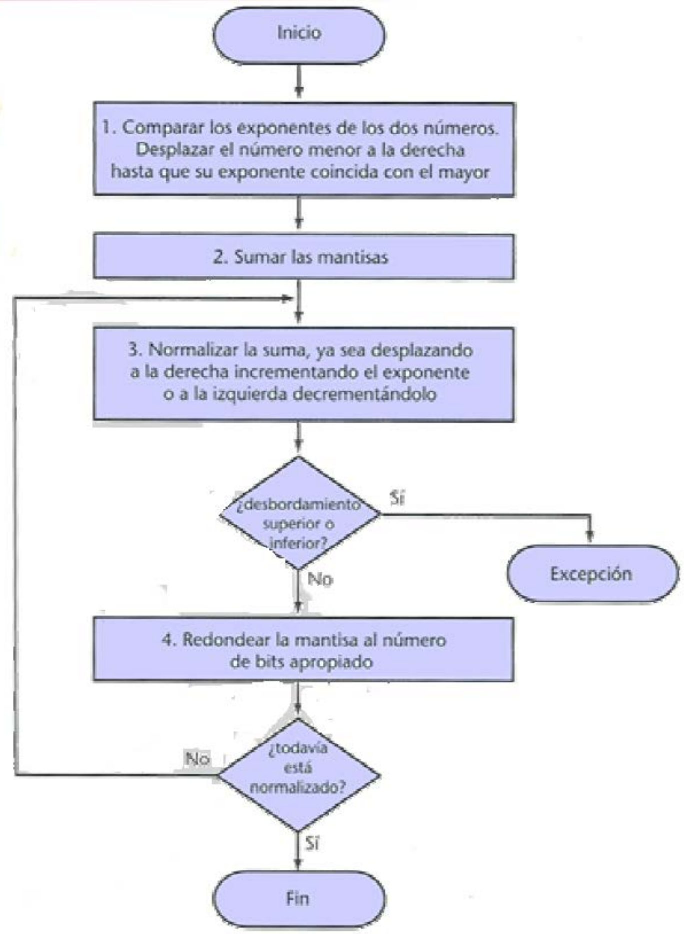
- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
  - Suma y Resta
  - Multiplicación y División
- Bibliografía
- Actividades

Aritmética IEEE 754

- **¡¡Se realizan del mismo modo que para la aritmética decimal!!**
- En caso de suma o resta:
  - Se igualan los exponentes al de mayor grado
  - Se ajusta la mantisa del número menor
  - Se realiza la operación
- En caso de multiplicación o división
  - Se suman o restan los exponentes
  - Se realiza la operación con las mantisas

Suma/Resta en Notación Científica

- $9.999 \times 10^{1} + 1.610 \times 10^{-1}$
  - 3 cifras decimales para la mantisa.
  - 2 para el exponente.
- **Paso 1: Alinear mantisas**
  - $1.610 \times 10^{-1} = 0.01610 \times 10^{1}$ 0.016×10^{1}
- **Paso 2: Sumar mantisas**
  - $9.999 + 0.016 = 10.015$ Suma $= 10.015 \times 10^{1}$
- **Paso 3: Normalizar resultado**
  - $10.015 \times 10^{1} = 1.0015 \times 10^{2}$
- **Paso 4: Redondear resultado**
  - Resultado final $= 1.0015 \times 10^{2} \approx 1.002 \times 10^{2}$



Suma/Resta en Coma Flotante

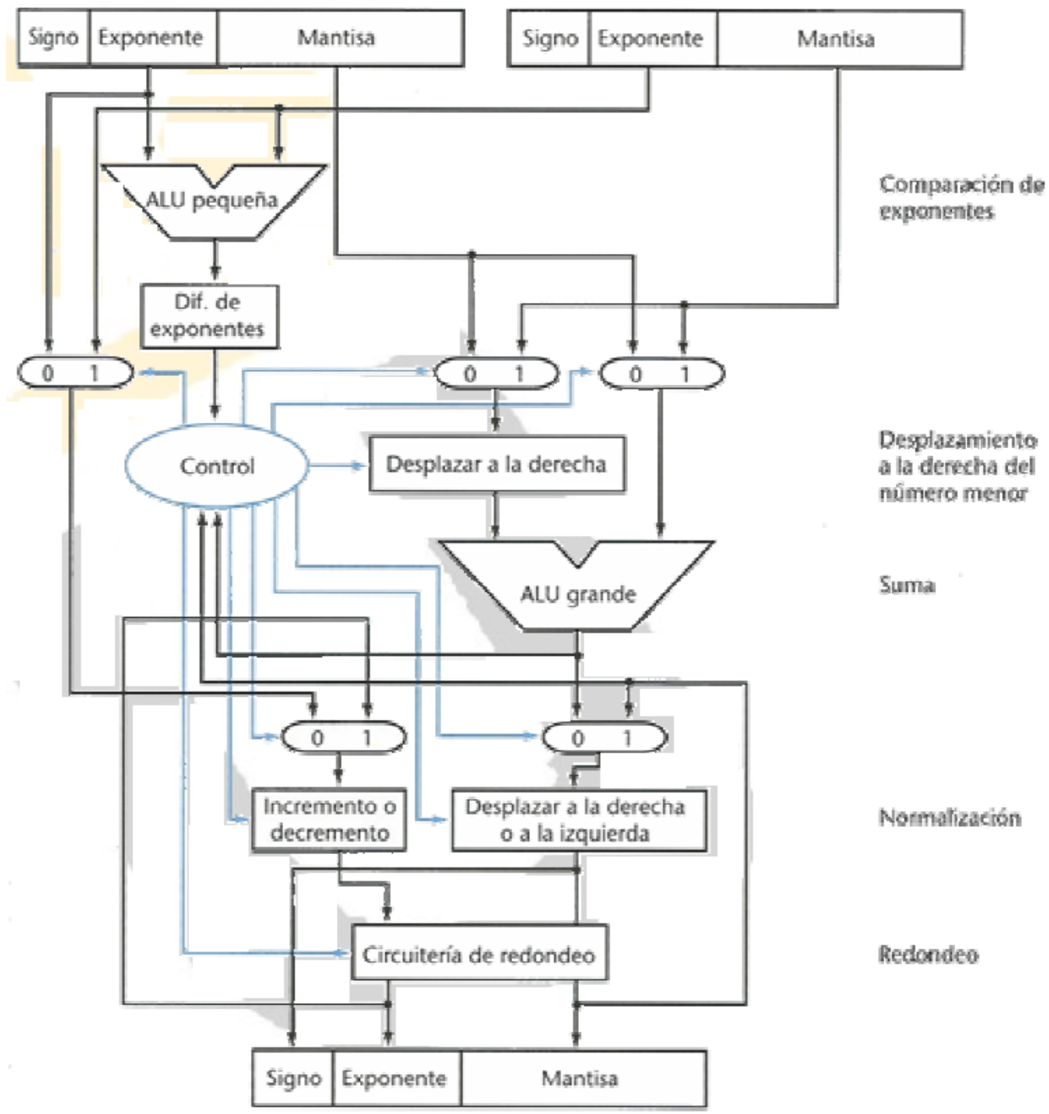
- Recordemos el caso decimal:
  - N = 3,5 · 10^3 + (-2,8 · 10^2)
  - N = 3,5 · 10^3 - 0.28 · 10^3 = 3,22 · 10^3
- El caso binario es **EXACTAMENTE** igual (ej. distinto al anterior):
  - 1 | 1010 0111 | 1000 000 + 0 | 1010 0011 | 1111 000
  - Igualo exponentes a "1010 0111" .
  - Ajusto mantisas: N2 debe desplazar su mantisa 4 posiciones a la derecha (1010 0111 – 1010 0011 = $4_{]10}$):
    - 1.1111 000 → 0.00011 111
    - Puedo hacer la cuenta en decimal: 167 – 163 = 4
  - …

Suma/Resta en Coma Flotante (3)

- …
- El signo será el del número mayor = 1 (negativo)
- Resto las mantisas (signo contrario) [OJO al **BIT OCULTO**!!!]:
- 1.1000 000 - 0.00011 111 = *[Comp2 – N2]* = 1.1000 000 + 1.11100 001 = 11.0110 001;
  - Puesto que es mayor que el número original hay que despreciar el bit más significativo → 1.0110 001
- Normalizo si es necesario para dejar el bit oculto.
  - En este caso ya se encuentra normalizado.
- El número final será:
  - N = 1 | 1010 0111 | 0110 001

Construcción de una unidad aritmético lógica (coma flot.)

- Diagrama de bloques de una ALU dedicada a la suma en coma flotante.

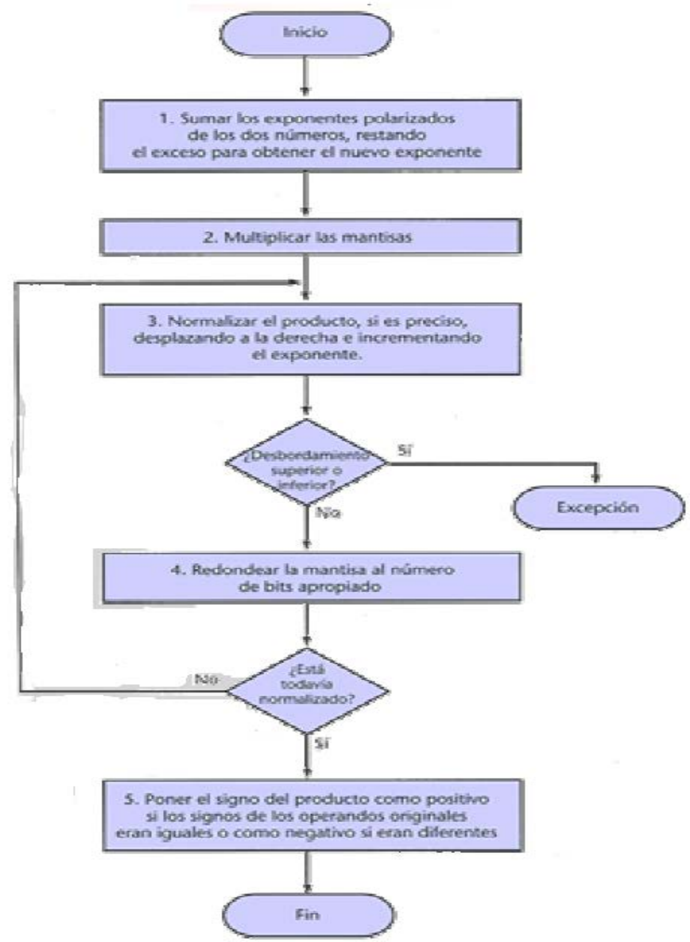


Contenido del capítulo

- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
  - Suma y Resta
  - Multiplicación y División
- Bibliografía
- Actividades

Multiplicación en Notación científica

- $1.110 \times 10^{10} \times 9.200 \times 10^{-5} = ?$
  - 3 cifras decimales para la mantisa.
  - 2 para el exponente.
- Paso 1: Sumar exponentes
  - Exponente $= 10 + (-5) = 5$
- Paso 2: Multiplicar mantisas
  - $1.110 \times 9.200 = 10.212$
- Paso 3: Normalizar resultado
  - $10.212 \times 10^5 = 1.0212 \times 10^6$
- Paso 4: Redondear resultado
  - $1.0212 \times 10^6 \approx 1.021 \times 10^6$
- Paso 5: Colocar signo
  - Resultado final $= +1.021 \times 10^6$



Multiplicación en coma flotante

- La multiplicación es más sencilla porque no es necesario que los exponentes sean iguales
  1. Las mantisas se multiplican como enteros en coma fija
  2. Los exponentes se suman
  3. Si cualquiera de los operandos es cero, el resultado también
- Existe la posibilidad de que sea necesario normalizar el resultado de la multiplicación
- También es posible que se produzca un desbordamiento a cero o a infinito del exponente:
  - Hay que controlar que el resultado sea representable

División en coma flotante

- La división consiste en dividir las mantisas y restar al exponente del dividendo el exponente del divisor
  1. Si el dividendo es cero el resultado es cero
  2. Si el divisor es cero se considera desbordamiento
  3. Si el dividendo y el divisor son cero, el resultado se identifica como un número desconocido
- Existe la posibilidad de que sea necesario normalizar el resultado de la división
- También es posible que se produzca un desbordamiento a cero o a infinito del exponente:
  - Hay que controlar que el resultado sea representable

Contenido del capítulo

- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
  - Suma y Resta
  - Multiplicación y División
- Bibliografía
- Actividades

Bibliografía

- Patterson y Hennessy: Estructura y Diseño de Computadores: Capítulo 3.
- Prieto, Lloris, Torres: Introducción a la Informática: Capítulo 3.
- Murdocca y Heuring: Principios de Arquitectura de Computadoras: Capítulos 2 y 3.

Contenido del capítulo

- Introducción
- Representación Coma Flotante Simple Precisión (IEEE-754)
- Aritmética IEEE-754
  - Suma y Resta
  - Multiplicación y División
- Bibliografía
- Actividades

Actividades

- Representar 7,5 y 1,5 usando el formato IEEE-754
- Indique el valor binario en IEEE-754 y el valor decimal del siguiente número hexadecimal representado en IEEE-754 de 32 bits: 0x3FE00000
- Indique, razonando brevemente su respuesta:
  - La representación en el estándar de coma flotante IEEE 754 de 32 bits del número decimal -6.625
  - El valor decimal del número hexadecimal 0x40A00000 que representa un número en coma flotante según IEEE 754 (precisión simple)
  - El valor decimal del número hexadecimal 0x00700000 que representa un número en coma flotante según IEEE 754 (precisión simple)

Actividades (2)

- Indique, razonando brevemente su respuesta:
  - La representación en el estándar de coma flotante IEEE 754 de 32 bits del número decimal -12.75
  - El valor decimal del número hexadecimal 0xC0A00000 que representa un número en coma flotante según IEEE 754 (precisión simple)
  - El valor decimal del número hexadecimal 0x00600000 que representa un número en coma flotante según IEEE 754 (precisión simple)
- ¿Cuál es el rango de números decimales (mayor y menor, negativo y positivo) que se pueden representar en coma flotante IEEE-754