**Trabajo Práctico N° 2:**

**Sistema de Numeración en Punto Flotante.**

**Ejercicio 1.**

*Considerando el sistema de Punto Flotante cuya mantisa es fraccionaria, con 6 bits , está expresada en BSS (en el inciso a) o BCS (en el inciso b) y su exponente en BCS con 4 bits, escribir el significado de las siguientes cadenas de bits (mantisa a la izquierda):*

|  |  |  |
| --- | --- | --- |
| **Cadena** | **(a) Mantisa en BSS** | **(b) Mantisa en BCS** |
| 010111 0110 | 010111 \* = ( + + + ) \* = + + 2 + 1= 23 | 0 10111 \* = ( + + + ) \* = + + + 2= 46 |
| 000001 0000 | 0000001 \* = \* = \* 1= | 0 000001 \* = \* = \* 1= |
| 000011 1001 | 000011 \* = ( + ) \* = + = (2 + 1)= | 0 00011 \* = ( + ) \* = + = (2 + 1)= |
| 111111 1111 | 111111 \* = (1 - ) \* = - = (64 + 1)= | 1 11111 \* = (1 - ) \* = - = (32 + 1)= |
| 000000 0000 | 000000 \* = 0 \* = 0 \* 1= 0 | 0 00000 \* = 0 \* = 0 \* 1= 0 |
| 000000 1111 | 000000 \* = 0 \* = 0 | 0 00000 \* = 0 \* = 0 |
| 111111 0000 | 111111 \* = (1 - ) \* = (1 - ) \* 1= | 1 11111 \* = -(1 - ) \* = (1 - ) \* 1= |
| 100000 0000 | 100000 \* = \* = \* 1= | 1 00000 \* = - \* = \* 1= |
| 000001 1111 | 000001 \* = \* = = | 0 00001 \* = \* = = |

**Ejercicio 2.**

*Dado un sistema de Punto Flotante cuya mantisa es fraccionaria , está expresada en BCS con 5 bits y su exponente en BSS con 3 bits, interpretar las siguientes cadenas considerando que la mantisa está sin normalizar, normalizada o normalizada con bit implícito Identificar aquellas cadenas que no pueden ser interpretadas y mencionar por qué.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cadena** | **Sin normalizar** | **Normalizada** | **Normalizada con bit implícito** |
| 01000 111 | 0 1000 \* = \* = = 64 | 0 1000 \* = \* = = 64 | 0 [1]1000 \* = ( + ) \* = + = 64 + 32= 96 |
| 11000 011 | 1 1000 \* = - \* = -= -4 | 1 1000 \* = - \* = -= -4 | 1 [1]1000 \* = -( + ) \* = -( + 2)= -(4 + 2)= -6 |
| 00000 000 | 0 0000 \* = 0 \* = 0 \* 1= 0 | --- | 0 [1]0000 \* = \* = 0,5 \* 1= 0,5 |
| 11111 111 | 1 1111 \* = -(1 - ) \* = -( - )= -(128 - 8)= -120 | 1 1111 \* = -(1 - ) \* = -( - )= -(128 - 8)= -120 | 1 [1]1111 \* = -(1 - ) \* = -( - )= -(128 - 4)= -124 |

**Ejercicio 3.**

*Calcular rango y resolución en extremos inferior negativo, superior negativo, inferior positivo y superior positivo para los siguientes sistemas de representación en punto flotante:*

*Observar que:*

* *En las mantisas BSS, no se puede expresar números negativos, con lo que, aún con exponente negativo, expresaremos un número positivo por un factor de escala menor a 1, pero también positivo. Ejemplo: 2 \* = 0,125.*
* *Las mantisas fraccionarias suponen el punto al principio de la mantisa.*
* *Los exponentes negativos indican factores de escala menores a 1 que mejoran la resolución.*
* *Mantisa normalizada implica que empieza con 1, o sea mantisa mínima 0,1 para la fraccionaria, igual a 0,5 en decimal. Esto hace que no se pueda representar el 0.*
* *Mantisa normalizada con bit implícito, significa agregar un 1 al principio de la misma al interpretarla. Ejemplo: 00000 se interpreta 0,100000 o 0,5 en base 10.*

**(a)** *Mantisa fraccionaria en BSS de 8 bits y exponente en BSS 4 bits.*

Rango= [00000000 \* ; 11111111 \* ]

Rango= [0 \* ; (1 - ) \* ]

Rango= [0 \* 1; ( - )]

Rango= [0; 32640].

Resolución en el extremo inferior= \* = \* 1= = .

Resolución en el extremo superior= \* = = 128.

**(b)** *Mantisa fraccionaria normalizada en BSS de 15 bits y exponente en CA1 10 bits.*

Rango= [100000000000000 \* ; 111111111111111 \* ]

Rango= [0,5 \* ; (1 - ) \* ]

Rango= [0,5 \* ; ( - )].

Resolución en el extremo inferior= \* = .

Resolución en el extremo superior= \* = .

**(c)** *Mantisa fraccionaria normalizada con bit implícito en BCS de 15 bits y exponente en Exceso 5 bits.*

Rango negativo= [1 [1]11111111111111 \* ; 1 [1]00000000000000 \* ]

Rango negativo= [-(1 - ) \* ; - \* ]

Rango negativo= [-( - 1); -].

Rango positivo= [0 [1]00000000000000 \* ; 0 [1]11111111111111 \* ]

Rango positivo= [ \* ; (1 - ) \* ]

Rango positivo= [; ( - 1)].

Resolución en el extremo superior negativo / extremo inferior positivo= \* = .

Resolución en el extremo inferior negativo / extremo superior positivo= \* = 1.

**(d)** *Mantisa fraccionaria normalizada con bit implícito en BCS de N bits y exponente en CA2 de M bits.*

Rango negativo= [-(1 - ) \* ; - \* ].

Rango positivo= [ \* ; (1 - ) \* ].

Resolución en el extremo superior negativo / extremo inferior positivo= \* .

Resolución en el extremo inferior negativo / extremo superior positivo= \* .

**Ejercicio 4.**

*Dado un sistema de Punto Flotante cuya mantisa es fraccionaria , está expresada en BCS con 10 bits y su exponente en CA2 con 5 bits, obtener la representación de los siguientes números, considerando que la mantisa está sin normalizar, normalizada o normalizada con bit implícito.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cadena** | **Sin normalizar** | **Normalizada** | **Normalizada con bit implícito** |
| 0 | 0 000000000 \* | --- | --- |
| 1 | 0 100000000 \* =  0 010000000 \* =  0 001000000 \* … | 0 100000000 \* | 0 [1]000000000 \* |
| 9 | 0 100100000 \* =  0 010010000 \* =  0 001001000 \* … | 0 100100000 \* | 0 [1]001000000 \* |
| -5,0625 | 1 101000100 \* =  1 010100010 \* =  1 001010001 \* … | 1 101000100 \* | 1 [1]010001000 \* |
| 34000,5 | --- | --- | --- |
| 0,015625 | 0 000001000 \* =  0 000010000 \* =  0 000100000 \* … | 0 100000000 \* | 0 [1]000000000 \* |
| Número máximo | 0 111111111 \* = (1 - ) \* = - = 32704 | 0 111111111 \* = (1 - ) \* = - = 32704 | 0 [1]111111111 \* = (1 - ) \* = - = 32736 |
| Número mínimo | 1 111111111 \* = -(1 - ) \* = -( - )= -32704 | 1 111111111 \* = -(1 - ) \* = -( - )= -32704 | 1 [1]111111111 \* = -(1 - ) \* = -( - )= -32736 |

**Ejercicio 5.**

*Decir cómo influyen las siguientes variantes en el rango y resolución:*

**(a)** *Mantisa con signo y sin signo.*

Mantisa con signo (supongo mantisa entera y exponente en BCS):

Rango= [-( - 1) \* ; ( - 1) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

Mantisa sin signo (supongo mantisa entera y exponente en BCS):

Rango= [0; ( - 1) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

**(b)** *Exponente con signo y sin signo.*

Exponente con signo ( supongo mantisa entera y en BCS):

Rango= [-( - 1) \* ; ( - 1) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

Exponente sin signo (supongo mantisa entera y en BCS):

Rango= [-( - 1) \* ; ( - 1) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

**(c)** *Tamaño de mantisa.*

Supongo mantisa entera en BCS y exponente en BCS:

Rango= [-( - 1) \* ; ( - 1) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

**(d)** *Tamaño de exponente.*

Supongo mantisa entera en BCS y exponente en BCS:

Rango= [-( - 1) \* ; ( - 1) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

**(e)** *Mantisa fraccionaria, fraccionaria normalizada y fraccionaria normalizada con bit implícito.*

Mantisa fraccionaria (supongo mantisa y exponente en BCS):

Rango= [-(1 - ) \* ; (1 - ) \* ].

Resolución en el extremo inferior= \* .

Resolución en el extremo superior= \* .

Mantisa fraccionaria normalizada (supongo mantisa y exponente en BCS):

Rango negativo= [-(1 - ) \* ; - \* ].

Rango positivo= [ \* ; (1 - ) \* ].

Resolución en el extremo superior negativo / extremo inferior positivo= \* .

Resolución en el extremo inferior negativo / extremo superior positivo= \* .

Mantisa fraccionaria normalizada con bit implícito (supongo mantisa y exponente en BCS):

Rango negativo= [-(1 - ) \* ; - \* ].

Rango positivo= [ \* ; (1 - ) \* ].

Resolución en el extremo superior negativo / extremo inferior positivo= \* .

Resolución en el extremo inferior negativo / extremo superior positivo= \* .

**Ejercicio 6.**

*Efectuar las siguientes sumas para un sistema de punto flotante con mantisa en BSS de 8 bits y exponente en BCS de 8 bits. Observar que los factores de escala deben ser los mismos, sino se sumarían dos mantisas con pesos distintos (recordar que se puede correr los unos y sumar o restar este corrimiento al exponente para obtener una cadena equivalente).*

**(a)** *00001111 00000011 +**00001000 00000010.*

Opción 1:

00001111 00000011 +

00001000 00000010 =

00001111 00000011 +

00000100 00000011 =

00010011 00000011= 19 \* = 19 \* 8= 152.

Opción 2:

00001111 00000011 +

00001000 00000010 =

00011110 00000010 +

00001000 00000010 =

00100110 00000010= 38 \* = 38 \* 4= 152.

**(b)** *01111111 00000000 + 11111100 10000001.*

Opción 1:

01111111 00000000 +

11111100 10000001 =

01111111 00000000 +

01111110 00000000 =

11111101 00000000= 253 \* = 253 \* 1= 253.

Opción 2:

01111111 00000000 +

11111100 10000001 =

11111110 10000001 +

11111100 10000001 =

[1]11111010 10000001 =

11111101 00000000= 253 \* = 253 \* 1= 253.

**(c)** *00000001 00000111 + 00011100 00000000.*

Opción 1:

00000001 00000111 +

00011100 00000000 =

10000000 00000000 +

00011100 00000000 =

10011100 00000000= 156 \* = 156 \* 1= 156.

Opción 2:

00000001 00000111 +

00011100 00000000 =

00100000 00000010 +

00000111 00000010 =

00100111 00000010= 39 \* = 39 \* 4= 156.

**Ejercicio 7.**

*Suponiendo que los números que no son representables se aproximan al más próximo, obtener las representaciones o aproximaciones de los números 8.625, 0.4 y 2.5 en los sistemas.*

|  |  |  |
| --- | --- | --- |
| **Número** | **(a) Mantisa fraccionaria normalizada de 5 bits BSS y exponente de 4 bits CA2** | **(b) Mantisa fraccionaria normalizada de 10 bits BCS y exponente de 3 bits CA2** |
| 8,625 | 10001 \* = 8,5 | 0 111111111 \* = 7,984375 |
| 0,4 | 11010 \* = 0,40625 | 0 110011010 \* = 0,400390625 |
| 2,5 | 10100 \* = 2,5 | 0 101000000 \* = 2,5 |

**Ejercicio 8.**

*Se define Error Absoluto y Error Relativo de un número x en un sistema de la siguiente forma: EA (x)= y ER (x)= , donde es el número representable del sistema más próximo a x. Calcular los errores absolutos y relativos para los casos del ejercicio anterior.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Número** | **(a)** | | **(b)** | |
| **EA** | **ER** | **EA** | **ER** |
| 8,625 | = 0,125 | = 0,0145 | = 0,640625 | = 0,0743 |
| 0,4 | = 0,00625 | = 0,015625 | = 0,000390625 | = 0,0009765625 |
| 2,5 | = 0 | = 0 | = 0 | = 0 |

**Ejercicio 9.**

*Considerando que, en los procesos de truncamiento o redondeo, la elección se basa en la representación más cercana, estimar el Error Absoluto Máximo cometido en las representaciones del ejercicio 7. Recordar que la distancia entre 2 representaciones sucesivas se conoce como resolución (R), por lo que EAmáx .*

|  |  |  |
| --- | --- | --- |
| **Número** | **EAmáx en (a)** | **EAmáx en (b)** |
| 8,625 | \* \* = = 0,25 | --- |
| 0,4 | \* \* = = 0,0078125 | \* \* = = 0,00048828125 |
| 2,5 | \* \* = = 0,0625 | \* \* = = 0,00390625 |

**Ejercicio 10.**

*Tomar un sistema de punto flotante cualquiera y dibujar la forma del gráfico de cada tipo de error en función del número que se quiere representar.*

Supongo mantisa fraccionaria en BSS de 3 bits y exponente en BCS de 3 bits.

**Ejercicio 11.**

*Detallar las características del estándar IEEE 754 para simple precisión y doble precisión.*

|  |  |  |
| --- | --- | --- |
| **Característica** | **Simple precisión** | **Doble precisión** |
| Bits en signo | 1 | 1 |
| Bits en exponente | 8 | 11 |
| Bits en fracción | 23 | 52 |
| Total de bits | 32 | 64 |
| Exponente en exceso | 127 | 1023 |
| Rango de exponente | [-126; 127] | [-1022; 1023] |
| Rango de números | [ ] | [ ] |

**Ejercicio 12.**

*¿Qué valores están representados por las siguientes cadenas si responden al estándar IEEE 754?*

**(a)**

0 11000100 [1]00000000000000000000000= \*

0 11000100 [1]00000000000000000000000= \* 1

0 11000100 [1]00000000000000000000000= .

**(b)**

1 11111110 [1]10100000000000000000000= - ( + + )

1 11111110 [1]10100000000000000000000= - (1 + 0,5 + 0,125)

1 11111110 [1]10100000000000000000000= -1,625 \* .

**(c)**

0 00000000[1] 00000000000000000000001= \*

0 00000000[1] 00000000000000000000001= .

**(d)**

0 00000000[1] 10011000000000000000000= ( + + )

0 00000000[1] 10011000000000000000000= + + .

**(e)**

1 00000000 00000000000000000000000= 0.

**(f)**

0 11111111 00000000000000000000000= +.

**(g)**

0 11111111 00000100000000000000000= NaN.

**(h)**

0 01100010100 [1]0000000000000000000000000000000000000000000000000000=

\* = \* 1= .

**(i)**

0 11010101110 [1]1010000000000000100000000000000000000000000000000000=

( + + + )= + + + .

**(j)**

0 00000000000[1] 0101000000000000000000000000000000000000000000000000=

( + )= + .

**(k)**

1 11111111111 1111100000000000000000000000000000000000000000000000= NaN.

**Ejercicio 13.**

*Hallar la representación en simple precisión del estándar IEEE 754 de los siguientes números: 1, 13, 257, -40000, 0.0625.*

|  |  |
| --- | --- |
| **Número** | **Simple precisión del estándar IEEE 754** |
| 1 | 0 01111111 [1]00000000000000000000000 |
| 13 | 0 10000010 [1]10100000000000000000000 |
| 257 | 0 10000111 [1]00000001000000000000000 |
| -40000 | 1 10001110 [1]00111000100000000000000 |
| 0,0625 | 0 01111011 [1]00000000000000000000000 |

**Ejercicio 14.**

*Calcular rango y resolución en extremos inferior negativo y superior positivo para los sistemas de simple precisión y doble precisión del estándar IEEE 754. ¿Cuál es el menor número positivo distinto de “0” que se puede representar?*

|  |  |  |
| --- | --- | --- |
| **Característica** | **Simple precisión** | **Doble precisión** |
| Rango negativo | [-(2 - ) \* ; - \* ] | [-(2 - ) \* ; - \* ] |
| Rango positivo | [ \* ; (2 - ) \* ] | [ \* ; (2 - ) \* ] |
| Resolución extremo inferior negativo | \* | \* |
| Resolución extremo superior positivo | \* | \* |
| Menor número positivo distinto de 0 | \* | \* |

**Ejercicio 15.**

*Efectuar las siguientes sumas (las cadenas son representaciones en el estándar IEEE 754):*

**(a)** *00001111 010000000000000000000000 + 00010000 010000000000000000000000.*

00001111 [1],010000000000000000000000 +

00010000 [1],010000000000000000000000 =

\* [1],010000000000000000000000 +

\* [1],010000000000000000000000 =

\* [1],010000000000000000000000 +

\* [1] 0 ,10000000000000000000000 =

\* [1]1,11000000000000000000000 =

\* [1],111000000000000000000000= 1,875 \* .

**(b)** *11111111 101010101010101010101010 + 11111100 100000011111000001101010.*

11111111 [1]101010101010101010101010 +

11111100 [1]100000011111000001101010 =

NaN +

\* [1]100000011111000001101010 =

NaN.

**Ejercicio 16.**

*En el estándar IEEE 754, ¿para qué sirve, cuando el exponente es 0 y la mantisa no es nula, que la mantisa no esté normalizada?*

Cuando el exponente es 0 y la mantisa no es nula, que la mantisa no esté normalizada (desnormalizar) sirve para representar números por debajo de y para garantizar la menor brecha entre el menor número normalizado y el mayor número desnormalizado.