

Arquitecturas Computacionales

Miguel Solis

Clase 02

Facultad de Ingeniería / Escuela de Informática
Universidad Andrés Bello, Viña del Mar.

- Número binario:
 - base 2
 - dígitos 0 o 1
 - $110 = 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2$

- Número hexadecimal:
 - base 16
 - dígitos de 0 a 9 \rightarrow A a F
 - $AB = 10 \cdot 16^1 + 11 \cdot 16^0$

Convierta los siguientes números binarios en hexadecimales:

- 10110
- 10010101
- 100100001001

Suma binaria

- $0 + 0 = 0$
- $0 + 1 = 1$
- $1 + 0 = 1$
- $1 + 1 = 0$ (con acarreo)

- $001 + 011$
- $101 + 011$
- $010 + 011$

- $1 - 1 = 0$
- $0 - 0 = 0$
- $1 - 0 = 1$
- $0 - 1 = 1$ (descontando 1 en el acarreo)

- 011 - 001
- 101 - 011
- 010 - 011

Se pueden representar de dos formas:

- en signo y magnitud
- en complemento

Representación en signo y magnitud

- El número está formado por dos partes: magnitud y signo (que puede ser +: 0, o -: 1)
- El signo se representa mediante un único bit adicional, con la posición del MSB
- +123: 01111011
- -123: 11111011

Al sumar con representación en signo y magnitud, se debe comparar tanto los signos como las magnitudes de los operandos.

- Si el signo de ambos números es el mismo: se suman las magnitudes y el signo es el mismo
- Si los signos son diferentes, hay que comparar las magnitudes:
 - Si son iguales, el resultado es 0 (dah)
 - Si son distintas, restamos a la magnitud mayor la más pequeña, con el mismo signo del número de magnitud mayor

Considere la representación de negativos en signo y magnitud:

- $0110 + 1011$
- $1011 + 1110$
- $0100 + 1111$

Representación en complemento 2

- Se usa para sumar y restar de forma más sencilla sin usar comparadores.
- Si N' es el complemento de N , $-N$ viene dado por $N' + 1$
- +7: 0111
- -7: 1001

Considere la representación de negativos en complemento 2:

- $0110 + 1011$
- $1011 + 1110$
- $0100 + 1111$

- Representación en punto fijo: cantidad de bits pre-definida para parte entera y decimal
- Representación en punto flotante: se normaliza el número y se usa notación científica

Considere un número real x :

$$x = q \cdot 2^m,$$

- q : mantisa
- m : exponente

Un número en punto flotante con 32 bits acomoda sus bits según:

- signo del número real x : 1 bit
- signo del exponente m : 1 bit
- magnitud del exponente: 7 bits
- magnitud de la mantisa: 23 bits

Considere $x = 26,32$:

- fijo : 0 00000000011010 . 0101000111101011
- flotante: 0 1000101 10100101000111101011100