

Semana #2 – Sistema Binario, Octal y Hexadecimal / Aritmética en binario / Códigos de detección de errores

SISTEMA BINARIO

El sistema binario, en matemáticas e informática, es un sistema de numeración en el que los números se representan utilizando solamente las cifras cero y uno (0 y 1). Los ordenadores trabajan internamente con dos niveles de voltaje, por lo que su sistema de numeración natural es el sistema binario (encendido 1, apagado 0) y cada dígito es llamado bit.

El sistema binario se lee de la siguiente manera:

1	1	1	1	1	1	1	1	1	1
512	256	128	64	32	16	8	4	2	1

Así pues, por ejemplo, los siguientes valores binarios, serían sus respectivos decimales:

$$100101 = 37$$

$$0111 = 7$$

$$10000 = 16$$

$$1100000 = 96$$

Ejercicio

Convierta en binario los siguientes valores decimales:

128	64	32	16	8	4	2	1

$$10 = \underline{\hspace{2cm}}$$

$$78 = \underline{\hspace{2cm}}$$

$$32 = \underline{\hspace{2cm}}$$

$$25 = \underline{\hspace{2cm}}$$

$$108 = \underline{\hspace{2cm}}$$

$$44 = \underline{\hspace{2cm}}$$

$$56 = \underline{\hspace{2cm}}$$

$$80 = \underline{\hspace{2cm}}$$

$$14 = \underline{\hspace{2cm}}$$

SISTEMA HEXADECIMAL

El sistema hexadecimal, a veces abreviado como hex, es el sistema de numeración posicional de base

16 —empleando por tanto 16 símbolos—. Su uso actual está muy vinculado a la informática y ciencias de la computación, pues los computadores suelen utilizar el byte u octeto como unidad básica de memoria; y, debido a que un byte representa 28 valores posibles, y esto puede representarse como:

$2^8 = 2^4 \cdot 2^4 = 16 \cdot 16 = 1 \cdot 16^2 + 0 \cdot 16^1 + 0 \cdot 16^0$, que, según el teorema general de la numeración posicional, equivale al número en base 16 10016, dos dígitos hexadecimales corresponden exactamente —permiten representar la misma línea de enteros— a un byte.

En principio dado que el sistema usual de numeración es de base decimal y, por ello, sólo se dispone de diez dígitos, se adoptó la convención de usar las seis primeras letras del alfabeto latino para suplir los dígitos que nos faltan. El conjunto de símbolos sería, por tanto, el siguiente:

$$S = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$$

Se debe notar que A = 10, B = 11, C = 12, D = 13, E = 14 y F = 15. En ocasiones se emplean letras minúsculas en lugar de mayúsculas.

EJERCICIO

Convierta los siguientes valores hexadecimales a decimales y luego a binarios:

Valor hexadecimal	Valor decimal	Valor binario
2F8		
DE1		
BC		
A0		
AAB		
FD2		

SISTEMA OCTAL

El sistema numérico en base 8 se llama octal y utiliza los dígitos 0 a 7.

Por ejemplo, el número binario para 74 es 1001010, lo agruparíamos como 1 001 010. De modo que el número decimal 74 en octal es 112.

En informática, a veces se utiliza la numeración octal en vez de la hexadecimal. Tiene la ventaja de que no requiere utilizar otros símbolos diferentes de los dígitos. Sin embargo, para trabajar con bytes o conjuntos de ellos, asumiendo que un byte es una palabra de 8 bits, suele ser más cómodo el sistema hexadecimal, por cuanto todo byte así definido es completamente representable por dos dígitos hexadecimales.

Es posible que la numeración octal se usara en el pasado en lugar de la numeración decimal, por ejemplo, para contar los espacios interdigitales o los dedos distintos de los pulgares. Esto explicaría por qué en latín nueve (novem) se parece tanto a nuevo (novus). Podría tener el significado de número nuevo.

Tabla de conversión entre decimal, binario, hexadecimal y octal

Deci	Binario	Hexadec	Oct
0	00000	0	0
1	00001	1	1
2	00010	2	2
3	00011	3	3
4	00100	4	4
5	00101	5	5
6	00110	6	6
7	00111	7	7
8	01000	8	10
9	01001	9	11
10	01010	A	12
11	01011	B	13
12	01100	C	14
13	01101	D	15
14	01110	E	16
15	01111	F	17
16	10000	10	20
17	10001	11	21
18	10010	12	22
....
30	11110	1E	36
31	11111	1F	37
32	100000	20	40

SUMA DE NÚMEROS BINARIOS

Las posibles combinaciones al sumar dos bits son:

$$\begin{array}{l} 0 + 0 = 0 \\ 0 + 1 = 1 \\ 1 + 0 = 1 \\ 1 + 1 = 10 \end{array}$$

Ejemplo:

$$\begin{array}{r} 100110101 \\ + 11010101 \\ \hline 1000001010 \end{array}$$

Se puede convertir la operación binaria a una operación decimal resolver la decimal y del resultado de la operación decimal se convierte a un resultado (número) binario.

Operamos como en el sistema decimal: comenzamos a sumar desde la derecha, en nuestro ejemplo, 1 +

1 = 10, entonces escribimos 0 en la fila del resultado y llevamos 1 (este "1" se llama acarreo o arrastre).

A continuación, se suma el acarreo a la siguiente columna: 1 + 0 + 0 = 1, y seguimos hasta terminar todas las columnas (exactamente como en decimal).

RESTA DE NÚMEROS BINARIOS

El algoritmo de la resta en binario es el mismo que en el sistema decimal. Pero conviene repasar la operación de restar en decimal para comprender la operación binaria, que es más sencilla. Los términos que intervienen en la resta se llaman minuendo, sustraendo y diferencia.

Las restas básicas 0 - 0, 1 - 0 y 1 - 1 son evidentes:

$$\begin{array}{l} 0 - 0 = 0 \\ 1 - 0 = 1 \\ 1 - 1 = 0 \end{array}$$

0 - 1 = equivale a 10 - 1 = 1. El dígito 1, se toma *prestado* de la posición siguiente.

La resta 0 - 1 se resuelve, igual que en el sistema decimal, tomando una unidad prestada de la posición siguiente: 10 - 1 = 1 y me llevo 1, lo que equivale a decir en decimal, 2 - 1 = 1.

Esa unidad prestada debe devolverse, sumándola, a la posición siguiente. Ejemplos:

$$\begin{array}{r} 10001 \\ -01010 \\ \hline 00111 \end{array}$$

$$\begin{array}{r} 11011001 \\ -10101011 \\ \hline 00101110 \end{array}$$