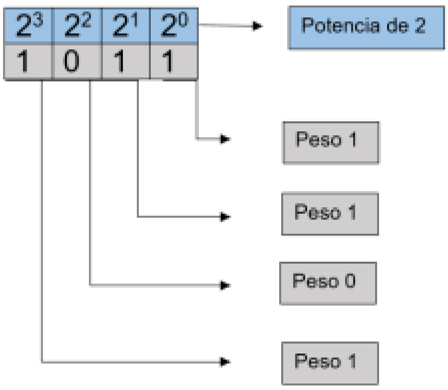
Método de los pesos

Para realizar la conversión de decimal a binario y binario a decimal, podemos aplicar un método abreviado para no tener que realizar tantas cuentas, o al menos cuentas mas simples.

Como sabemos cada dígito de un numero binario, tiene un peso que esta determinado por su posición.



###1011###$_{(2)}$

Esto es una consecuencia del teorema fundamental de la numeración Este teorema establece la forma general de construir números en un sistema de numeración posicional.

- **N** número válido en el sistema de numeración.
- **b** base del sistema de numeración. Número de símbolos permitidos en el sistema.
- $\mathbf{d_i}$ un símbolo cualquiera de los permitidos en el sistema de numeración.
- **n** número de dígitos de la parte entera.
- **,** coma fraccionaria. Símbolo utilizado para separar la parte entera de un número de su parte fraccionaria.
- **k** número de dígitos de la parte decimal.

La fórmula general para construir un número **N**, con un número finito de decimales, en un sistema de numeración posicional de base **b** es la siguiente:

$$N = \sum_{i=-k}^{i=n-1} d_i \, x \, b^i$$

Desarrollado queda

$$N = d_{n-1}xb^{n-1} + d_{n-2}xb^{n-2} + \cdots + d_0xb^0 +, d_{-1}xb^{-1} + \cdots + d_{-k}xb^{-k}$$

Y si lo aplicamos a un numero binario

- **N** número válido en el sistema de numeración.
- **2** base del sistema de numeración. Número de símbolos permitidos en el sistema.
- **d_i** un símbolo cualquiera de los permitidos en el sistema de numeración.
- **n** número de dígitos de la parte entera.
- **,** coma fraccionaria. Símbolo utilizado para separar la parte entera de un número de su parte fraccionaria.
- **k** número de dígitos de la parte decimal.

$$N = d_{n-1}x2^{n-1} + d_{n-2}x2^{n-2} + \cdots + d_0x2^0 +, d_{-1}x2^{-1} + \cdots + d_{-k}x2^{-k}$$

Binario a Decimal

Como vemos, cuando trabajamos en el sistema binario, estamos utilizando todo el tiempo las potencias de 2, entonces podemos elaborar una tabla con estas potencias.

4096	2048	1024	512	256	128	64	32	16	8	4	2	1	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125	0.00390625
2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
12	11	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8

A partir de esta tabla si tenemos un numero en binario

	Binario			
dígitos	1	1	0	1
posición	3	2	1	0

Ubicamos el numero en la tabla de la siguiente manera

1	1	0	1
3	2	1	0

4096	2048	1024	512	256	128	64	32	16	8	4	2	1	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.078125	0.00390625
2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
12	11	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8

Las posiciones del dígito en binario deben coincidir con las posiciones de la tabla. Esto nos indicará que valor representa cada dígito en el ejemplo, tenemos las posiciones 0, 2 y 3 con dígito 1

Si hiciéramos el desarrollo del numero en binario tendríamos

1101 = $1\times2^3+1\times2^2+0x2^1+1\times2^0=8+4+0+1=13_{(10)}$

El dígito 0 no afecta el cálculo, por eso es que si usamos la tabla, directamente sumamos los valores de las posiciones donde el dígito binario es 1. En este caso las posiciones 0,2 y 3.

Mirando directamente en la tabla sacaríamos $8+4+1=13_{(10)}$, como vemos es lo mismo que la forma desarrollada, pero no necesitamos hacer el desarrollo, ya que lo sacamos directo de la tabla

X-VaC9LyjzI **Conversión binario a decimal por el método de los pesos**

Decimal a Binario

Veamos ahora como seria pasar de decimal a binario usando nuestra tabla

Tenemos el numero $275_{(10)}$

Lo que vamos a hacer es buscar en la tabla el numero mas alto menor que 275

4096	2048	1024	512	256	128	64	32	16	8	4	2	1
2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
12	11	10	9	8	7	6	5	4	3	2	1	0

Es 256, esto nos indica que necesariamente ese dígito binario tiene que estar en 1 o "encendido", por lo tanto el numero binario tendría 9 posiciones, ya que la posición 8 ($2^8=256$) debe estar, lo que implica que todas las anteriores estén.

1									
9	8	7	6	5	4	3	2	1	0

Tenemos el dígito que representa 256 en 1 y el resto aun no sabemos. Pero ya hemos representado el valor 256, asi que se lo podemos restar al numero original, o sea **275-256** nos quedaría **19**, es el valor que aun no hemos representado.

Si observamos la tabla, el siguiente dígito tiene un valor 128, el siguiente 64 … todos los que superan 19 no pueden estar porque estaríamos representando un numero mayor, así que nuestro numero en binario quedaría

1	0	0	0	0					
9	8	7	6	5	4	3	2	1	0

El siguiente dígito es 2^4 que según nuestra tabla es **16**, entonces colocamos un 1 en esa posición, y se lo restamos al **19** que queremos representar **19-16**, nos queda **3** por representar que seria **2+1**, o sea las posiciones 0 y 1

1	0	0	0	0	1	0	0	1	1
9	8	7	6	5	4	3	2	1	0

Nos quedaría este numero en binario

El proceso que hicimos, en cuentas es el siguiente.

$$\begin{array}{r} 275 \\ -\ 256 \\ \hline 19 \end{array} \qquad \begin{array}{r} 19 \\ -\ 16 \\ \hline 3 \end{array} \qquad \begin{array}{r} 3 \\ -\ 2 \\ \hline 1 \end{array}$$

R1vVHZaURhQ **Conversión de decimal a binario por el método de los pesos**

— *Martha*

Volver

(273)