

## Operación de complementar a 2

Hemos visto en la teoría que la operación de complementar un binario a 2, que da como resultado el opuesto, se logra invirtiendo todos los bits y sumando 1. Además, *el complemento del complemento* da el número original, es decir, haciendo dos veces la misma operación volvemos al número de partida.

Pero, ¿por qué funciona este método? ¿Por qué estos dos pasos de invertir y sumar 1 dan el opuesto aritmético?

## Dificultad en la resta

Olvidémonos por un momento de los binarios y volvamos a los números decimales, “los de siempre”. De las operaciones que sabemos hacer de la escuela primaria, seguramente la suma nos ha resultado siempre la operación más fácil. Posiblemente la siguiente sea la resta, salvo en ciertos casos especiales.

Esos casos especiales aparecen, seguramente, cuando hay que “pedir prestado”. Por ejemplo,  $356 - 189$ . Aquí hay una dificultad extra, porque 9 es mayor que 6 y 8 es mayor que 5. *Hay dígitos en el sustraendo (189) que son mayores que los correspondientes del minuendo (356).*

Cuando se presenta este caso hay que recordar en qué posiciones “pedimos prestado” y en qué posiciones se disminuyen en 1 los dígitos del minuendo. Debido a estas complicaciones, estos casos son algo más difíciles para el cálculo mental.

## Método de complemento a la base

Un método alternativo de restar, llamado “de complemento a la base”, evita la complicación de los casos especiales de la resta, y funciona como sigue.

Llamemos A y B a los números que queremos restar. Es decir, queremos resolver  $x = A - B$ . Si A y B tienen n cifras, se construye un número N, con un 1 y n ceros. Por ejemplo, en el caso anterior donde los números tienen tres cifras, construimos  $N = 1000$ . Además, llamemos M a  $N - 1$ . Por ejemplo, para A y B de tres cifras,  $M = 999$ .

Ahora reescribimos la ecuación  $x = A - B$  usando algunos sencillos trucos aritméticos:

- $x = A - B$
- $x = A - B + 1000 - 1000$  // Sumamos y restamos N, la ecuación no cambia
- $x = A - B + (999 + 1) - 1000$  // Escribimos N como  $M + 1$ , nada cambia
- $x = A + (999 - B) + 1 - 1000$  // Reordenamos los términos, nada cambia

Si resolvemos la ecuación de esta manera, el resultado será el mismo que antes, sólo que ahora las operaciones son diferentes. Hay más operaciones que antes; **pero ahora son más fáciles**, porque en la resta  $999 - B$  no puede haber ningún caso de “pedir prestado”. Siempre los dígitos del minuendo serán mayores que los del sustraendo. Hemos convertido una resta “difícil” en dos sumas y dos restas fáciles.

## Ejemplo resuelto

Resolvamos el ejemplo de antes,  $x = 356 - 189$ . Calculamos  $999 - 189$  que es 810. Sumamos  $356 + 810 = 1166$ . Ahora sumamos 1: 1167. Y finalmente restamos 1000, que no es otra cosa que eliminar el 1 más significativo: 167. Efectivamente,  $356 - 189 = 167$ .

Si los números tuvieran más o menos de tres cifras, en el primer paso solamente restamos y sumamos la unidad seguida de tantos ceros como haga falta (100, 10000, 100000. . .).

## Aplicación a los binarios

Esto que ocurre en el mundo de los decimales, que nos es familiar, en realidad se cumple en cualquier base. En otras bases, al construir el número  $N$ , siempre tomamos la unidad seguida de ceros. Pero cuando separemos  $N$  en  $M + 1$ , el número  $M$ , en lugar de ser una sucesión de “nueves”, será **una sucesión de los dígitos más altos disponibles en el sistema de esa base**. En base 2, se tratará de una sucesión de “unos”.

Supongamos siempre que  $A$  y  $B$  tienen tres cifras, pero ahora en base 2:

- $x = A - B$
- $x = A - B + 1000 - 1000$
- $x = A - B + (111 + 1) - 1000$
- $x = A + (111 - B) + 1 - 1000$

Ejemplo en base 2: resolvamos  $4 - 3$  en binario. ¿Cuánto debería dar?. Éste es un caso de resta “con dificultad” porque las dos últimas cifras del sustraendo son mayores que las del minuendo.

- $x = 100 - 011$
- $x = 100 - 011 + 1000 - 1000$
- $x = 100 - 011 + (111 + 1) - 1000$
- $x = 100 + [(111 - 011) + 1] - 1000$

Ahora observemos que las operaciones que aparecen entre corchetes en la última ecuación corresponden efectivamente a **invertir los bits del sustraendo y luego**

**sumar 1**, tal como dice nuestra “receta” para calcular el complemento a 2. Cada vez que restemos un dígito binario de 111, si lo que restamos es 1, quedará 0, y si restamos un 0, quedará un 1. Esto es equivalente a **invertir los bits** del numeral que restamos.

Hagamos las operaciones completas:

$$\blacksquare x = 100 + (111 - 011) + 1 - 1000 = 100 + 100 + 1 - 1000 = 1000 + 1 - 1000 = 1001 - 1000 = 1$$

## Obtener el opuesto

Este procedimiento de “restar sumando”  $A - B$ , cuando queremos obtener el opuesto de un número (que, como sabemos, es su complemento a 2), se generaliza si decimos **sea A igual a 0**.

En efecto, si hacemos A igual a 0, con lo cual desaparece de todas las ecuaciones presentadas anteriormente, obtenemos el procedimiento para complementar a 2 un número B, y, así, **obtener el opuesto**. La operación de complementar a 2 el número B no es más que resolver  $x = 0 - B$ , donde B está expresado en el sistema binario, por el método de complemento a la base.