

### 3. Sustractores: Conversión de sumadores en sustractores.

La resta entre dos números no es mas que una suma con alguno de los operandos negativo. Veamos cómo se obtiene un número negativo en binario.

#### 3.1. Números con signo

Un número binario con signo queda determinado por su magnitud y signo. Este último es el bit más a la izquierda en un número binario y si su valor es 0 indica que el número es positivo y 1 que es negativo.

Existen tres formatos binarios para representar números enteros con signo:

- Signo-magnitud:

Unicamante hay que añadir el bit de signo a la magnitud binaria deseada y asignar al bit de signo un 1 o un 0 si queremos que sea negativo o positivo respectivamente.

$$+5 = 0101 \quad -5 = 1101$$

- Complemento a 1:

En este formato, los números positivos se expresan igual que en el formato signo-magnitud. Sin embargo, los negativos se obtienen realizando el complemento a uno del número positivo (incluido el signo).

$$+3 = 0011 \quad -3 = 1100$$

El número positivo equivalente a un número negativo se obtiene realizando el complemento a 1 del número negativo.

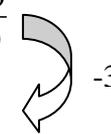
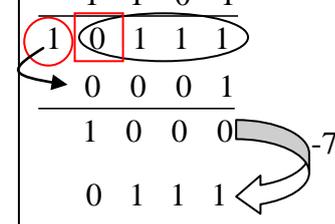
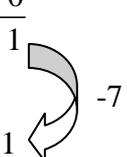
- Complemento a 2:

En este formato, los números positivos se expresan igual que en el formato signo-magnitud. Sin embargo, los negativos se obtienen realizando el complemento a dos del número positivo (incluido el signo). El complemento a dos se obtiene realizando el complemento a 1 y sumando uno a dicho número complementado a 1.

$$+5 = 0101 \quad -5 = (-5)_{c1} + 1 = 1011$$

El número positivo equivalente a un número negativo se obtiene realizando el complemento a 2 del número negativo.

En la Tabla 26 se muestran tres operaciones utilizando las tres representaciones. Se observa que el resultado de algunas operaciones es incorrecto, es decir, no todas las representaciones son adecuadas para relizar operaciones matemáticas.

	$(+5) + (+2) = (+7)$	$(-5) + (+2) = (-3)$	$(-5) + (-2) = (-7)$
M-S	$\begin{array}{r} 0\ 1\ 0\ 1 \\ 0\ 0\ 1\ 0 \\ \hline 0\ 1\ 1\ 1 \end{array}$	$\begin{array}{r} 1\ 1\ 0\ 1 \\ 0\ 0\ 1\ 0 \\ \hline 1\ 1\ 1\ 1 \end{array}$ ERROR	$\begin{array}{r} 1\ 1\ 0\ 1 \\ 1\ 0\ 1\ 0 \\ \hline 1\ 0\ 1\ 1 \end{array}$ ERROR
C1	$\begin{array}{r} 0\ 1\ 0\ 1 \\ 0\ 0\ 1\ 0 \\ \hline 0\ 1\ 1\ 1 \end{array}$	$\begin{array}{r} 1 \\ 1\ 0\ 1\ 0 \\ 0\ 0\ 1\ 0 \\ \hline 1\ 1\ 0\ 0 \end{array}$ 	$\begin{array}{r} 1\ 0\ 1\ 0 \\ 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 1\ 1 \\ 1\ 0\ 0\ 0 \\ \hline 0\ 0\ 0\ 1 \\ 1\ 0\ 0\ 0 \\ \hline 0\ 1\ 1\ 1 \end{array}$ 
C2	$\begin{array}{r} 0\ 1\ 0\ 1 \\ 0\ 0\ 1\ 0 \\ \hline 0\ 1\ 1\ 1 \end{array}$	$\begin{array}{r} 1 \\ 1\ 0\ 1\ 1 \\ 0\ 0\ 1\ 0 \\ \hline 1\ 1\ 0\ 1 \end{array}$ 	$\begin{array}{r} 1\ 1 \\ 1\ 0\ 1\ 1 \\ 1\ 1\ 1\ 0 \\ \hline 1\ 1\ 0\ 0\ 1 \\ 0\ 0\ 1\ 1\ 1 \end{array}$ 

**Tabla 26**

A la hora de trabajar con números con signo es muy importante definir el registro con el que se va a trabajar, es decir, el número de bits y por lo tanto los números que se pueden representar con ese número de bits. Si se realizan operaciones con un registro de  $n$  bits, tras una operación, los bits que superen esta cifra deberán ser ignorados (overflow o desbordamiento). Por ello, es importante tener presente en todo momento, a partir de un determinado registro, los números que pueden representarse. Esta información se obtiene a partir de la siguiente expresión:

$$\text{Números que pueden representarse con } n \text{ bits } [-2^{n-1}, 2^{n-1}-1]$$

Veamos un ejemplo. Supongamos que se debe realizar la siguiente operación utilizando la representación en C2.  $(-7) + (-2) = (-9)$

Con un registro de 4 bits será suficiente para representar los operandos (-7 y -2) con signo. Sin embargo, al realizar la operación obtenemos el siguiente resultado:

$$\begin{array}{r} 1\ 0\ 0\ 1 \\ 1\ 1\ 1\ 0 \\ \hline 1\ 0\ 1\ 1\ 1 \end{array}$$

Al ser el registro es de 4 bits, en el resultado también debemos examinar 4 bits. El resultado nos indica que es un número positivo, lo cual es erróneo. Esto es consecuencia del Overflow, ya que, el resultado al ser 9, no está dentro del margen  $-2^{n-1}$  y  $2^{n-1}-1$ .

Este es un hecho muy importante a la hora de diseñar sistemas que operen con números con signo.