



Representación de Números Binarios en la Computadora

Las computadoras utilizan representación binaria para los números y se le conoce como tamaño de registro. Actualmente las PCs cuentan con una capacidad de registro de 64 bits, esto significa que cada registro cuenta con un total de 64 posiciones, celdas o casillas para representar números binarios.

					...									
63	62	61	60	59	...	4	3	2	1	0				

Los sub índices que se han colocado debajo son la potencia de dos de cada celda. Por tanto, el número que queramos representar debe ser, primero, convertido a binario, y después, escribirlo en el registro justificado a la derecha. Ejemplo:

- Representar: $(10)_{10} = (1010)_2$, con un total de 10 casillas.

0	0	0	0	0	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---

Notar que el rango de representación de números enteros, utilizando N casillas es $[0, 2^N - 1]$, y la cantidad total de números que se pueden representar es 2^N . Por ejemplo, con tres casillas podemos representar del 0 al 7, un total de 8.

0	=	0	0	0
1	=	0	0	1
2	=	0	1	0
3	=	0	1	1
4	=	1	0	0
5	=	1	0	1
6	=	1	1	0
7	=	1	1	1

¿Cómo representamos un número negativo?

Reservando una de las casillas para contener el signo: la de más a la izquierda. Y como solo podemos almacenar 0 o 1, la convención es:

- 0 representa un número positivo.
- 1 representa un número negativo.

Ahora bien, utilizando una casilla para el signo se reduce la capacidad de representar en un módulo, pues se pierde el dígito con la mayor potencia, pero se gana la capacidad de representar números negativos.

Notar que el rango de representación de números enteros, utilizando N casillas es $[-2^{N-1}, 2^{N-1}-1]$, y la cantidad total de números que se pueden representar es 2^N . Por ejemplo, con tres casillas podemos representar del -4 al 3, un total de 8.

3	=	0	1	1
2	=	0	1	0
1	=	0	0	1
0	=	0	0	0
-1	=	1	0	1
-2	=	1	1	0
-3	=	1	1	1
-4	=	1	0	0

OJO con el -4: Esta combinación de bits corresponde al cero negativo, que en realidad no tendría sentido. Pero la convención tomada en todos los sistemas digitales actuales es que representará al número negativo más pequeño del rango de representación. De esta manera ninguna combinación de bits se desperdicia y ninguna queda sin sentido. Más adelante, cuando veamos las operaciones de suma dentro de la ALU, veremos que sí toma sentido.

Formatos de representación de Números Binarios en la computadora

Magnitud verdadera:

Consiste en la representación de números binarios tal cual es. Es decir, se realiza la conversión por el método conocido, se escribe en las casillas, justificado a la derecha y se coloca el bit de signo en la posición más a la izquierda. Luego las casillas intermedias se rellenan con cero. Esta es la representación que más se parece a la de los binarios naturales vista en las clases anteriores. Ejemplo:

- Representar en MV: $(10)_{10} = (1010)_2$, con un total de 10 casillas.

0	0	0	0	0	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---

- Representar en MV: $(-10)_{10} = (-1010)_2$, con un total de 10 casillas.

1	0	0	0	0	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---

Complemento a Uno:

Consiste en obtener primero la representación en magnitud verdadera; luego se reemplazan los unos por ceros y los ceros por unos, excepto el signo. En otras palabras, se niegan todos los bits de la magnitud. Ejemplo:

- Representar en C1: $(10)_{10} = (1010)_2$, con un total de 10 casillas.

0	1	1	1	1	1	1	1	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---

- Representar en C1: $(-10)_{10} = (-1010)_2$, con un total de 10 casillas.

1	1	1	1	1	1	1	1	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---

Complemento a Dos:

Consiste en obtener primero la representación en magnitud verdadera; luego se obtiene la representación en complemento a uno y finalmente, se le suma uno. Ejemplo:

- Representar en C2: $(10)_{10} = (1010)_2$, con un total de 10 casillas.

0	1	1	1	1	1	1	1	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---

- Representar en C2: $(-10)_{10} = (-1010)_2$, con un total de 10 casillas.

1	1	1	1	1	1	1	1	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---

Convenio de Complemento a dos:

Para la representación de cantidades numéricas en las computadoras digitales actuales existe un convenio, el cual se conoce como “convenio de complemento a dos”. Este convenio nos dice que, dentro de las computadoras:

- a) Las cantidades positivas se representan en magnitud verdadera.
- b) Las cantidades negativas se representan en complemento a dos.

Para todo ejercicio de conversión a cualquiera de estas tres representaciones nos deberán indicar con cuantos bits se está trabajando.

Conversión de regreso a base diez:

Para convertir a base diez una cantidad representada en cualquiera de estos formatos, se procede así:

- a) Si está en magnitud verdadera: simplemente se realiza la suma de las potencias de dos de las posiciones que tienen uno, excepto el signo. Luego, el signo se coloca dependiendo si es 0 u 1 en esa posición.
- b) Si está en complemento a uno: se niegan los bits de magnitud para regresar a magnitud verdadera y luego se procede como indica el literal a.
- c) Si está en complemento a dos: se niegan los bits, se le suma 1 y luego se procede como en el literal a).