Sistemas de numeración

Sitio:	[Agencia de Aprendizaje a lo largo de la Vida]	Imprimido por:	Sebastian Puche
Curso:	Lógica Computacional 1° G	Día:	sábado, 28 de septiembre de 2024, 16:40
Libro:	Sistemas de numeración		

Tabla de contenidos

1. Sistema binario - Introducción

Contar en binario es igual que contar en decimal, si tienen todos pulgares. —Glaser y Way

En la escuela primaria, aprendió el significado de la notación decimal: para interpretar una cadena de dígitos decimales como un número, mentalmente multiplique cada dígito por su valor de posición. Por ejemplo, 5 049 tiene un 5 en el lugar de los millares, un 0 en el lugar de las centenas, un 4 en el lugar de las decenas y un 9 en el lugar de las unidades. Por tanto

5049 = 5.(1000) + 0.(100) + 4.(10) + 9.(1)

Usando la notación exponencial, está ecuación se puede escribir como

5049= 5.103 + 0.102 + 4.101 + 9.100

De manera más general, la notación decimal se basa en el hecho de que cualquier número entero positivo puede ser escrito de manera única como una suma de productos de la forma

d.10n,

donde cada n es un entero no negativo y cada d es uno de los dígitos decimales de 0, 1, 2, 3, 4, 5, 6, 7, 8, o 9. La palabra decimal proviene de la raíz latina deci, que significa “diez”. La notación decimal (o de base 10) expresa un número como una cadena de dígitos en la que cada dígito indica la posición de la potencia de 10 por la que se multiplica. La posición que está más a la derecha es el lugar de las unidades (o el lugar de 100), a la izquierda está el lugar de las decenas (o el lugar 101), a la izquierda está el lugar de las centenas (o el lugar 102) y así sucesivamente, como se muestra a continuación.

Lugar	10^3 miles	10^2 centenas	10^1 decenas	10^0 unidades
Dígito decimal	5	0	4	9

1.1. Algunas consideraciones

Antes de adentrarnos en el tema por completo, les especificamos algunas notaciones que se utilizaron para el desarrollo de los temas.

Donde figura, por ejemplo, 2b10 = significa 2 en base 10 (sistema decimal)

1b2 = signigica 1 en base 2 (sistema binario)

2^4 = significa 2 elevado a la 4 ó 2 potencia de 4

BSS() = binario sin signo (esto significa que de momento, sólo veremos número

NO NEGATIVOS

1.2. Representación binaria de números

No hay nada sagrado acerca del número 10, usamos el 10 como base de nuestro sistema de numeración habitual ya que sucede que tenemos diez dedos. De hecho, cualquier número entero mayor de 1 puede servir como base para un sistema de numeración. En ciencia computacional, la notación de base 2, o la notación binaria es de especial importancia ya que las señales utilizadas en electrónica moderna están siempre en uno de los dos estados.(La raíz latina bi significa “dos”.)

Cualquier número entero se puede representar como una suma única de productos de la forma d.2^n donde cada n es un entero y cada d es uno de los dígitos binarios (o bits) 0 o 1. Por ejemplo,

$$
\begin{aligned}
27 &= 16+8+2+1 \\
&= 1.2^4 + 1.2^3 + 1.2^2 + 1.2^1 + 1.2^0
\end{aligned}
$$

En notación binaria, como en notación decimal, se escriben sólo los dígitos binarios y no las potencias de la base. En notación binaria, entonces

$$1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$

$$27_{10} = 11011_2$$

donde los subíndices indican la base, ya sea 10 o 2, en el que está escrito el número. Los lugares en notación binaria corresponden con las distintas potencias de 2. La posición más a la derecha es el lugar de los unos (o lugar 2^0), a la izquierda está el lugar de los dos (o lugar 2^1), a la izquierda está el lugar de los cuatro (o lugar 2^2) y así sucesivamente, como se muestra a continuación.

Lugar	2^4 dieciseises	2^3 ochos	2^2 cuatros	2^1 dos	2^0 unos
Dígito binario	1	1	0	1	1

Al igual que en la notación decimal, se puede agregar o quitar ceros a la izquierda al gusto. Por ejemplo,

$$003_{10} = 3_{10} = 1.\ 2^1 + 1.2^0 = 11_2 = 011_2$$

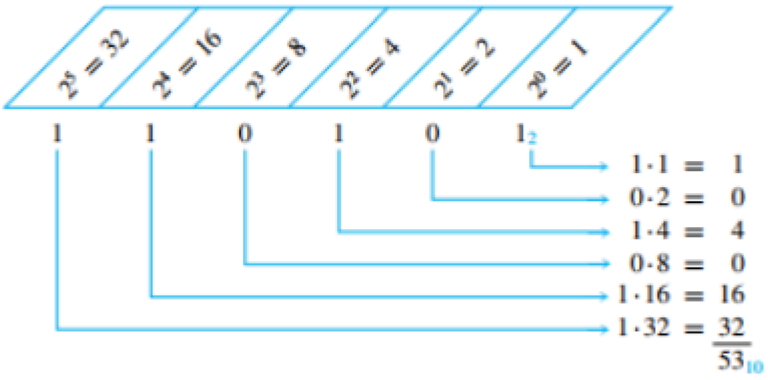
1.3. Conversion de un binario a decimal

Represente 110101_2 en notación decimal.

Solución

$$110101_2 = 1.2^5 + 1.2^4 + 0.2^3 + 1.2^2 + 0.2^1 + 1.2^0$$
$$= 32 + 16 + 4 + 1$$
$$= 53_{10}$$

Por otra parte, se puede utilizar el esquema que se muestra a continuación



1.4. Conversión de un decimal a binario

Este proceso permite construir las cadenas a partir de un valor. Para hacerlo se presenta el mecanismo de las divisiones sucesivas, donde se aplica el siguiente algortimo hasta obtener un cociente igual a 0:

1. Si x > 0 calcular la división entera: x/2,

2. Tomar el resto de la división anterior como un bit (pues es un valor en el conjunto {0,1})

3. Si el cociente es mayor a cero, volver al paso 1 con el cociente como dividendo.

4. Se construye la cadena tomando solo los restos: en el orden que fueron obtenidos se ubican de derecha a izquierda (menos significativo a más significativo).

Suponer por ejemplo que se necesita representar el número 26 en el sistema binario:

1. Se divide el valor 26 por 2 obteniendo resto 0 y cociente 13

2. El resto 0 es el bit menos significativo

3. El nuevo valor de x es 13. Se calcula x=2 obteniendo resto 1 y cociente 6.

4. El resto 1 es el segundo bit de la cadena

5. El nuevo valor de x es 6. Se calcula 6=2 obteniendo resto 0 y cociente 3.

6. El resto 0 es el tercer bit de la cadena

7. El nuevo valor de x es 3. Se calcula 3=2 obteniendo resto 1 y cociente 1.

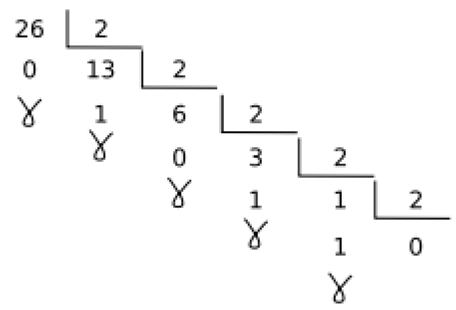
8. El resto 1 es el cuarto bit de la cadena

9. El nuevo valor de x es 1. Se calcula 1=2 obteniendo resto 1 y cociente 0.

10. El resto 1 es el quinto bit de la cadena

11. Se construye la cadena tomando solo los restos, en el orden que fueron obtenidos, de derecha a izquierda: 11010

El proceso anterior se aprecia gráficamente de la siguiente manera:



1.5. Rango del sistema

Hasta este punto se describió el sistema de numeración en términos de sus símbolos, su función de interpretación y su función de representación, pero queda pendiente analizar la capacidad de representación en término del conjunto de números representable o rango . Si bien se dijo que un sistema numérico permite construir un conjunto infinito de cadenas, esto no se cumple en el contexto de un sistema de cómputos pues se tiene una cantidad limitada de bits, por lo que el conjunto de números representables también se ha limitado. Por este motivo se dice que el sistema es un sistema restringido.

cadena	Interpretación	Representación	valor
000	$I(000) = 0$	$R(0) = 000$	0
001	$I(001) = 1 \times 2^0 = 1$	$R(1) = 001$	1
010	$I(010) = 1 \times 2^1 = 2$	$R(2) = 010$	2
011	$I(011) = 1 \times 2^1 + 1{,}2^0 = 3$	$R(3) = 011$	3
100	$I(100) = 1 \times 2^2 = 4$	$R(4) = 100$	4
101	$I(101) = 1 \times 2^2 + 1 \times 2^0 = 5$	$R(5) = 101$	5
110	$I(110) = 1 \times 2^2 + 1 \times 2^1 = 6$	$R(6) = 110$	6
111	$I(111) = 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 7$	$R(7) = 111$	7

Considerar por ejemplo un sistema binario restringido a 3 bits y que sólo contemple los números Naturales, lo llamamos Sin Signo y lo denotamos BSS(3).

Para analizar su rango se debe determinar el valor mínimo y máximo representables. Para el primer caso se interpreta la primer cadena: 000:

$$I(000) = 0.2^2 + 0.2^1 + 0.2^0 = 0$$

Para el segundo caso se interpreta la última cadena: 111

$$I(111) = 1.2^2 + 1.2^1 + 1.2^0 = 7$$

Es decir que el rango de BSS(3) son todos los números naturales comprendidos entre 0 y 7, y se representa de la siguiente manera: [0;7]. El conjunto de valores representables tiene 8 elementos.

Además, con 3 bits se pueden construir 8 cadenas de números representables, es decir, 2^3 = 8.

Generalizando, en un sistema BSS(n) se tiene 2n cadenas y un rango [0; 2n - 1]

Una lista de potencias de 2 es útil para hacer conversiones de binario a decimal y de decimal a binario

Potencias de 2	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Forma decimal	1024	512	256	128	64	32	16	8	4	2	1

1.6. Suma y resta binaria

Los métodos de cálculo de aritmética binaria son análogos a los de aritmética decimal. En aritmética binaria el número 2 (10b2 en notación binaria) desempeña un papel similar al del número 10 en aritmética decimal.

1.7. Suma en notación binaria

Sume 1101b2 y 111b2 usando notación binaria.

Ya que 2b10 = 10b2 y 1b10 = 1b2, la traducción de 1b10 + 1b10 = 2b10 en notación binaria es

$$\begin{array}{r} 1_2 \\ +\ 1_2 \\ \hline 10_2 \end{array}$$

De lo que se deduce que la suma de dos 1 juntos, da como resultado llevar un 1 cuando se usa la notación binaria. Sumar tres 1 juntos, también da como resultado en llevar un 1 ya que 3b10 = 11b2 (“uno uno base dos”)

$$\begin{array}{r} 1_2 \\ +\ 1_2 \\ +\ 1_2 \\ \hline 11_2 \end{array}$$

Así, la suma se puede realizar de la siguiente manera:

$$\begin{array}{cccccc} & & 1 & 1 & 1 & \leftarrow \text{renglón de lo que se lleva} \\ & & 1 & 1 & 0 & 1_2 \\ + & & & 1 & 1 & 1_2 \\ \hline & 1 & 0 & 1 & 0 & 0_2 \end{array}$$

1.8. Resta en notación binaria

Reste 10112 de 110002 usando notación binaria

En la resta decimal el hecho de que 10b10 - 1b10 = 9b10 se usa para prestar a través de varias columnas. Por ejemplo, considere lo siguiente:

$$
\begin{array}{rcccl}
 & & 9 & 9 & \\
 & & \cancel{1} & 1 & \leftarrow \text{renglón de préstamos} \\
 & \cancel{1} & \cancel{0} & \cancel{0} & 0_{10} \\
- & & & 5 & 8_{10} \\
\hline
 & & 9 & 4 & 2_{10}
\end{array}
$$

En la resta binaria, también puede ser necesario pedir prestado a través de más de una columna. Pero cuando usted pide prestado un 1b2 de 10b2, lo que queda es 1b2.

$$
\begin{array}{rr}
 & 10_2 \\
- & 1_2 \\
\hline
 & 1_2
\end{array}
$$

Así, la resta se puede realizar de la siguiente manera:

$$
\begin{array}{rccccl}
 & & & 0 & 1 & 1 \\
 & & & \cancel{1} & \cancel{1} & 1 \quad \leftarrow \text{renglón de préstamos} \\
 & 1 & \cancel{1} & \cancel{0} & \cancel{0} & 0_2 \\
- & & 1 & 0 & 1 & 1_2 \\
\hline
 & & 1 & 1 & 0 & 1_2
\end{array}
$$

2. Sistema hexadecimal

Ahora debería ser obvio que los números escritos en notación binaria ocupan mucho más espacio que los números escritos en notación decimal. Sin embargo, muchos aspectos del funcionamiento de la computadora pueden ser mejor analizados usando números binarios. La notación hexadecimal es mucho más compacta que la notación decimal y es mucho más fácil para convertir de ida y vuelta entre la notación hexadecimal y binaria que entre la notación binaria y la decimal. La palabra hexadecimal proviene del griego hex- que significa "seis" y la raíz latina deci-, que significa "diez". Por tanto hexadecimal se refiere a "dieciséis" y la notación hexadecimal también se llama notación de base 16. La notación hexadecimal se basa en el hecho de que cualquier número entero se puede expresar de manera única como una suma de números de la forma

d.16^n

donde cada n es un entero no negativo y cada d es uno de los números enteros de 0 a 15. Con el fin de evitar ambigüedad, cada dígito hexadecimal se debe representar por un solo símbolo. Los enteros del 10 al 15 están representados por los símbolos A, B, C, D, E y F.

En la siguiente tabla, se muestran los dieciséis dígitos hexadecimales, junto con sus equivalentes decimales y, para futura referencia, sus 4 bits equivalentes binarios.

Decimal	Hexadecimal	4-bit binario equivalente
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Este sistema de numeración también contiene un conjunto de funcionalidades para poder operar con él y de la misma manera que en el sistema binario podremos:

1. Interpretar

2. Representar

3. Calcular su rango

4. Agrupación de bits

2.1. Convirtiendo de notación hexadecimal a decimal

Traspolando el mismo razonamiento que hicimos para el sistema binario, en este caso asignamos una potencia de 16 a cada posición comenzando de derecha a izquierda (desde el 0) y ubicamos nuestra cadena respetando las posiciones.

Consideremos los siguientes ejemplos:

Queremos interpretar la cadena 128, entonces:

$$1 \,.\, 16^2 + 2 \,.\, 16^1 + 8 \,.\, 16^0 = 296$$

Ahora bien, qué pasa si la cadena que necesitamos interpretar contiene una letra: 2A, entonces, siguiendo la misma lógica quedaría:

$$2 \,.\, 16^1 + A \,.\, 16^0 = 42$$

Teniendo en cuenta que el valor de A es 10 como figura en la tabla del capítulo anterior

Como último ejemplo, considerar la cadena A3F

$$I(A3F) = 10 \,.\, 16^2 + 3 \,.\, 16^1 + 15 \times 16^0$$

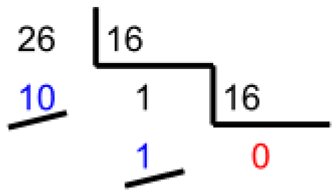
2.2. De hexadecimal a binario

Siguiendo la lógica del sistema binario, para representar valores mediante cadenas se deben realizar sucesivas divisiones por la base, que en este caso es 16, hasta obtener un cociente igual a 0 tomando cada resto como bits de la cadena.

Ejemplo: Se necesita representar el número 26 en hexadecimal:

1. Se divide el valor 26 por 16 hasta encontrar un cociente 0

2. Se construye la cadena tomando solo los restos, empezando por el último



Uno de los restos es 10, entonces debemos traducirlo a la letra correspondiente aplicando la tabla de interpretación de hexadecimal. El valor 10 es equivalente a la letra A, quedando entonces 1A. Esto quiere decir que el valor 26 en decimal se corresponde con la cadena 1A en hexadecimal.

2.3. Rango del sistema

De la misma manera que en el sistema binario debemos calcular el mínimo número representable interpretando la cadena más chica y la más grande. Siendo el rango todos los números comprendidos entre ambos. Supongamos el sistema hexadecimal de 2 dígitos:

El mínimo valor representable es el resultado de interpretar la cadena 00, es decir:

$$0 \times 16^1 + 0 \times 16^0 = 0$$

El máximo valor representable es el resultado de interpretar la cadena FF

$$15 \times 16^1 + 15 \times 16^0 = 255$$

(aplicando la tabla de interpretación de hexadecimal)

Por lo tanto el rango de este sistema es: [0; 255]

2.4. Convirtiendo de binario a hexadecimal

Agrupación de bits

Este es un método establece una relación directa entre cadenas del sistema binario y cadenas del sistema hexadecimal, que nos permite convertir de manera directa cadenas en binario a cadenas en hexadecimal, sin pasar por la interpretación de la cadena original. Para esto, la cadena binaria se segmenta formando cuartetos de bits comenzando por el bit menos significativo (b0). Supongamos por ejemplo la cadena 1001011010100101, que al ser segmentada se obtiene:

1001 0110 1010 0101

Dado que cada cuarteto es alguna de las combinaciones de 4 bits del sistema BSS(4) y por lo tanto el rango que cubren es

[0;15]

Considerando que dichos valores del rango se pueden representar por un solo caracter hexadecimal, entonces se aplica la siguiente tabla para convertir, uno a uno, los cuartetos de la cadena.

Símbolo Cadena $BSS(4)$	0	1	2	3	4	5	6	7
	0000	0001	0010	0011	0100	0101	0110	0111
	8	9	A	B	C	D	E	F
	1000	1001	1010	1011	1100	1101	1110	1111

En el ejemplo mencionado:

1001 0110 1010 0101

9 6 A 5

Por lo tanto, las cadenas 96A5 y 1001 0110 1010 0101 representan el mismo valor. Notar que no hizo falta obtener ese valor, dado que no se aplicó el proceso de interpretación.

2.5. Ejercicio de volcado de memoria

La unidad más pequeña de memoria direccionable en la mayoría de las computadoras es un byte, u ocho bits. En algunas operaciones de depuración de un volcado es de contenido de la memoria, es decir, se muestra o se imprime en orden el contenido de cada posición de memoria. Para ahorrar espacio y hacer la salida más fácil a ojo, se les dan las versiones hexadecimales del contenido de la memoria, en lugar de las versiones binarias. Supongamos, por ejemplo, que un segmento del volcado de memoria se parece a

A3 BB 59 2E

¿Cuál es el contenido real de las cuatro posiciones de memoria?

2.6. Solución a ejercicio de volcado de memoria

$$
\begin{aligned}
A3_{16} &= 10100011_2 \\
BB_{16} &= 10111011_2 \\
59_{16} &= 01011001_2 \\
2E_{16} &= 00101110_2
\end{aligned}
$$