

Sistemas de Numeración

Electrónica Digital 2021

Docente: Marina Murillo

Inst. de Investigación en Señales, Sistemas e Inteligencia Computacional
Facultad de Ingeniería y Ciencias Hídricas
Universidad Nacional del Litoral

Objetivos

- Representar números en diferentes bases:
 - Decimal
 - Binario
 - Octal
 - Hexadecimal
- Convertir números a diferentes bases
- Representar números con signo

Conceptos Preliminares

- Sistema de numeración → conjunto de símbolos empleados para representar información numérica
- El conjunto de símbolos constituye el alfabeto y depende la base del sistema de numeración
- La base indica la cantidad de símbolos distintos que usa un sistema de numeración

Sistema de numeración decimal

- Sistema numeración posicional → valor dígito depende de su **posición** dentro del número
- 10 símbolos (dígitos) → $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ → base 10
- Posición dígito indica la magnitud de cantidad representada y se asignan **pesos**:
 - N^o enteros: $\dots 10^5 10^4 10^3 10^2 10^1 10^0$
 - N^o fraccionarios: $10^2 10^1 10^0, 10^{-1} 10^{-2} 10^{-3} \dots$
- Valor número decimal → suma de los dígitos luego de haber multiplicado cada dígito por su peso

Polinomio característico

$$N = \sum_{i=-k}^n d_i \cdot 10^i$$

Ejemplos

- $47 = 4 \cdot 10^1 + 7 \cdot 10^0 = 40 + 7$
- $568,23 = 5 \cdot 10^2 + 6 \cdot 10^1 + 8 \cdot 10^0 + 2 \cdot 10^{-1} + 3 \cdot 10^{-2}$
- $67,924 = \dots$
- $157,1 = \dots$

Sistema de numeración binario

- Posicional
- 2 símbolos (bits) $\rightarrow \{0, 1\} \rightarrow$ base 2
- Bit más a la derecha \rightarrow bit menos significativo (LSB)
 - LSB en entero binario \rightarrow peso $2^0 = 1$
- Bit más a la izquierda \rightarrow bit más significativo (MSB) \rightarrow peso depende del tamaño del número binario
 - Estructura de pesos $\rightarrow 2^{n-1} \dots 2^3 2^2 2^1 2^0$
- Números fraccionarios \rightarrow representación binaria
 - Estructura de pesos $\rightarrow 2^{n-1} \dots 2^1 2^0, 2^{-1} 2^{-2} \dots 2^{-n}$
- En general, con n bits se puede contar hasta $2^n - 1$

Contar de 0 hasta 15 en binario

Número decimal	Número binario			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Sistema de numeración hexadecimal

- Posicional
- 16 símbolos $\rightarrow \{0, 1, \dots, 9, A, B, \dots, F\} \rightarrow$ base 16
- Estructura de pesos $\rightarrow 16^{n-1} \dots 16^1 16^0, 16^{-1} \dots 16^{-m}$

Contar de 0 a 15 en hexadecimal

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

Conversión de binario a decimal

El valor decimal de cualquier número binario puede hallarse sumando los pesos de todos los bits con valor 1 y descartando aquellos con valor 0

Ejemplo: convertir el nro. 1101101_2 a decimal

- Primero se determina el peso de cada bit que está en 1 y luego se obtiene la suma de los pesos para obtener el número decimal
- $1101101_2 = 2^6 + 2^5 + 2^3 + 2^2 + 2^0 = 64 + 32 + 8 + 4 + 1 = 109_{10}$

Ejemplo: convertir el nro. $0,1011_2$ a decimal

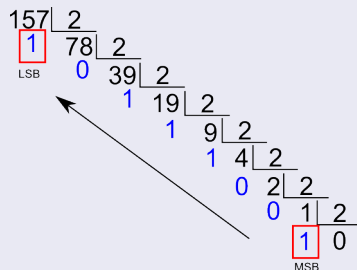
- Primero se determina el peso de cada bit que está en 1 y luego se suman los pesos para obtener la fracción decimal
- $0,1011_2 = 2^{-1} + 2^{-3} + 2^{-4} = 0,5 + 0,125 + 0,0625 = 0,6875_{10}$

Conversión de decimal a binario - Números enteros

Método de la división sucesiva por 2

Se divide la cantidad decimal por 2, anotando los residuos, hasta obtener un cociente **cero**. El último residuo obtenido es el MSB y el primero es el bit menos significativo LSB.

Ejemplo: convertir el número 157_{10} a binario



$$157_{10} = 10011101_2$$

Conversión de decimal a binario - Números fraccionarios

Método de la multiplicación sucesiva por 2

Se multiplica la fracción decimal por 2 y del resultado se extrae su parte entera (MSB). Se multiplica sucesivamente por 2 la parte fraccional restante hasta que la misma sea cero o alcance el número deseado de posiciones decimales. El último residuo o parte entera es el LSB.

Ejemplo: convertir el número $0,875_{10}$ a binario

$$\begin{array}{r}
 0,875 \\
 \times 2 \\
 \hline
 1,75 \\
 \text{MSB}
 \end{array}
 \quad
 \begin{array}{r}
 0,75 \\
 \times 2 \\
 \hline
 1,5 \\
 \text{MSB}
 \end{array}
 \quad
 \begin{array}{r}
 0,5 \\
 \times 2 \\
 \hline
 1,0 \\
 \text{LSB}
 \end{array}$$

$$0,875_{10} = 0,111_2$$

Conversión de hexadecimal a decimal

El valor decimal de cualquier número hexadecimal puede hallarse multiplicando el valor decimal de cada dígito hexadecimal por su peso, y sumando luego estos productos.

Ejemplo: convertir el número $3AB, 1_{16}$ a decimal

$$3AB, 1_{16} = 3 \cdot 16^2 + 10 \cdot 16^1 + 11 \cdot 16^0 + 1 \cdot 16^{-1} = 939,0625_{10}$$

Conversión de decimal a hexadecimal

- Parte entera → método división sucesiva por 16
- Parte fraccionaria → método multiplicación sucesiva por 16

Ejemplo: convertir el número $1243,5_{10}$ a hexadecimal

Parte entera	Parte fraccionaria
$ \begin{array}{r l} 1243 & 16 \\ \hline 11 & 77 \\ 13 & 4 \\ & 0 \end{array} $ <p>LSB ← MSB</p>	$ \begin{array}{r} 0,5 \\ \times 16 \\ \hline 8,0 \end{array} $

Entonces, $1243,5_{10} = 4DB,8_{16}$

Conversión de hexadecimal a binario

Para convertir un número hexadecimal en un número binario se reemplaza **cada símbolo hexadecimal** por el grupo de **cuatro bits** que representan a dicho símbolo.

Ejemplo: convertir el número $4DB,8_{16}$ a binario

- $4_{16} = 0100_2$
- $D_{16} = 1101_2$
- $B_{16} = 1011_2$
- $8_{16} = 1000_2$

Entonces, $4DB,8_{16} = 0100\ 1101\ 1011,1000_2$

Conversión de binario a hexadecimal

- Dividir número binario en grupos de 4 bits → símbolo hexadecimal equivalente.
- Parte entera → agrupar bits desde LSB hacia MSB (completar con ceros a la izquierda del MSB de ser necesario).
- Parte fraccionaria → agrupar desde MSB hacia el LSB (completar con ceros a la derecha del LSB de ser necesario).

Ejemplo: convertir el número $111101010,11_2$ a hexadecimal

$$\underbrace{0001}_1 \underbrace{1110}_E \underbrace{1010}_A, \underbrace{1100}_C_2 = 1EA, C_{16}$$

... en general

Conversión a **base arbitraria**

- Decimal \rightarrow Base **b**
 - Parte entera \rightarrow dividir número decimal por base **b**
 - Parte fraccionaria \rightarrow multiplicar número decimal por base **b**
- Base **b** \rightarrow Decimal
 - Evaluar polinomio característico en base **b**:
$$N_{10} = a_n \cdot \mathbf{b}^n + \dots + a_0 \cdot \mathbf{b}^0 + a_{-1} \cdot \mathbf{b}^{-1} + \dots + a_{-m} \cdot \mathbf{b}^{-m}$$

Tres formas...

- Formato magnitud y signo
- Formato complemento a uno (o a la base reducida)
- Formato complemento a dos (o a la base)

Formato magnitud y signo

Nº bin	Nº dec
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
<hr/>	
1000	-0
1001	-1
1010	-2
1011	-3
1100	-4
1101	-5
1110	-6
1111	-7

- MSB bit signo (BS) (0 → **positivo**, 1 → **negativo**)
- $n - 1$ bits restantes → **magnitud**
- n bits → rango $[-2^{n-1} + 1, 2^{n-1} - 1]$
- 4 bits → $[-2^3 + 1, 2^3 - 1]$
- Doble representación del 0 (+0 y -0)
- Más complejo para operar aritmética binaria

Complemento a Uno

Nº bin	Nº dec
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
<hr/>	
1000	-7
1001	-6
1010	-5
1011	-4
1100	-3
1101	-2
1110	-1
1111	-0

- Se cambian 1s por 0s y 0s por 1s, incluido el BS
- El MSB \rightarrow BS (0 \rightarrow pos, 1 \rightarrow neg)
- n bits \rightarrow núm en $[-2^{n-1} + 1, 2^{n-1} - 1]$
- 4 bits $\rightarrow [-2^3 + 1, 2^3 - 1]$
- Doble representación del 0 (+0 y -0)

Complemento a Dos

Nº bin	Nº dec
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	-8
1001	-7
1010	-6
1011	-5
1100	-4
1101	-3
1110	-2
1111	-1

- Se obtiene adicionando un 1 al complemento a uno ($C2 = C1 + 1$)
- El MSB \rightarrow BS (0 \rightarrow pos, 1 \rightarrow neg)
- n bits \rightarrow núm en $[-2^{n-1}, 2^{n-1} - 1]$
- Con 4 bits $\rightarrow [-2^3, 2^3 - 1]$
- Única representación del 0

Referencias

- Capítulo 2 del libro *Fundamentos de Sistemas Digitales 9na. Ed.*, Thomas Floyd.
- Sección 1.5 y 1.6 (págs 9 - 14) del libro *Diseño Digital 3ra. Ed.*, Morris Mano.