Apuntes de Lógica Digital

Daniel Araya Román

2023-05-08

1. Sistemas Binarios

1.1 Sistemas Digitales

En los sistemas digitales electrónicos actuales, las señales emplean sólo dos valores discretos $\rightarrow$ *binarios*. Un dígito binario, llamado **bit**, que puede tomar los valores 0 y 1. Un sistema digital es una interconexión de módulos digitales. Para entender como funciona cada módulo digital, se necesiatan conocimientos básicos de circuitos digitales y de su función lógica.

Un lenguaje importante para el diseño digital es el **(HDL, Hardware Description Language)**. Sirve para simular sistemas digitales y verificar su funcionamiento antes de crearlos en hardware.

Sistema digitales

$\downarrow$

Manejan cantidades discretas de información

$\downarrow$

Representadas en forma binario

1.2 Números Binarios

El número decimal 7392, contiene potencias de 10 que están implícitas en la posición de los coeficientes, e.g:

$$7 \times 10^3 + 3 \times 10^2 + 9 \times 10^1 + 2 \times 10^0$$

Un número con punto decimal se representa con una serie de coeficientes, así:

$$a_5a_4a_3a_2a_1a_0 \cdot a_{-1}a_{-2}a_{-3}$$

los coeficientes a_j son cualesquiera de los 10 dígitos (0...9); el valor del subíndice j indica la posición, y la potencia de 10 que se deberá multplicar ese coeficiente. De modo que:

$$10^5a_5 + 10^4a_4 + 10^3a_3 + 10^2a_2 + 10^1a_1 + 10^0a_0 + 10^-1a_{-1} + 10^-2a_{-2} + 10^-3a_{-3}$$

El sistema binario es distinto al decimal, sus coeficientes solo pueden tener 2 valores, 0 o 1. Cada coeficiente a_j se multiplica por 2^j. 11010.11 es 26.75 en decimal, porque:

$$1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 26.75$$

En general, un número expresado en un sistema de base **r** consiste en coeficientes que se multiplican por potencias de **r**:

$$a_n \cdot r^n + a_{n-1} \cdot r^{n-1} + ... + a_2 \cdot r^2 + a_1 \cdot r + a_0 + a_{-1} \cdot r^{-1} + a_{-2} \cdot r^{-2} + ... + a_{-m} \cdot r^{-m}$$

1.4 Números Octales y Hexadecimales

Las conversiones entre binario, octal y hexadecimal son importantes en las computadoras digitales. Puesto que $2^3 = 8$ y $2^4 = 16$, cada dígito octal corresponde a **tres** dígitos binarios, y cada dígito hexadecimal corresponde a **cuatro** dígitos binarios.

Binario → octal: agrupando los dígitos binarios de 3 en 3, de derecha a izquierda, y reemplazando cada grupo por su equivalente octal.

$$(10\,110\,001\,101\,011 \cdot 111\,100\,000\,110)_2 = (26153.7406)_8$$

Binario → hexadecimal: agrupando los dígitos binarios de 4 en 4, de derecha a izquierda, y reemplazando cada grupo por su equivalente hexadecimal.

$$(10\,1100\,0110\,1011 \cdot 1111\,0010)_2 = (2C6B.F2)_{16}$$

Cuando se habla de binario es más deseable expresarlo en términos de números octales o hexadecimales, porque son más compactos y fáciles de leer. Así $(111\,111\,111\,111)_2$ este número en binario de 12 bits, se puede escribir como $(7777)_8$ en octal o $(FFF)_{16}$ en hexadecimal.

Decimal (base 10)	Binary (base 2)	Octal (base 8)	Hexadecimal (base 16)
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Table 1: Numbers in different bases