

Descripción de la Base 2 y su Importancia, procesamiento de datos en computadoras

El sistema binario utiliza solo dos símbolos: el 0 y el 1, las computadoras interpretan estos símbolos como estados eléctricos: apagado y encendido.[1]

Claude Shannon propuso usar interruptores para representar operaciones lógicas, esta propuesta permitió aplicar principios de álgebra booleana a circuitos eléctricos.[1]

Los procesadores modernos emplean el sistema binario para almacenar y procesar datos. Esta codificación facilita la construcción de sistemas digitales eficientes y precisos, incluyendo el manejo de carry y borrow[2]

Suma Binaria

La suma binaria se realiza de derecha a izquierda como en el sistema decimal, el sistema combina bits y aplica reglas simples para generar resultados.[3]

Cuando se suman $1 + 1$, el sistema produce un 0 y genera un *carry* hacia la izquierda. Este acarreo se añade a la siguiente columna en la operación. [3]

La presencia del *carry* asegura que la suma binaria funcione correctamente entre columnas cada acarreo se propaga según sea necesario en el cálculo. [3]

EJEMPLO:

$$\begin{array}{r} 10011000 \\ + 00010101 \\ \hline 10101101 \end{array}$$

acarreo \longrightarrow

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

Resta Binaria

La resta binaria también se realiza de derecha a izquierda, usando reglas básicas. El sistema reconoce cuándo se requiere un *borrow* para continuar la operación. [3]

Cuando se intenta restar 1 de 0, el sistema pide prestado un 1 de la posición siguiente, esto convierte el 0 en 10 (binario) y permite completar la resta. [3]

El *borrow* garantiza que las restas se realicen correctamente en operaciones más largas. El sistema propaga el préstamo si la posición siguiente también es 0. [3]

EJEMPLO:

$$\begin{array}{r} 10001 \\ -01010 \\ \hline 00111 \end{array} \quad \begin{array}{r} 11011001 \\ -10101011 \\ \hline 00101110 \end{array}$$

Multiplicación Binaria

La multiplicación binaria solo tiene cuatro combinaciones posibles entre ceros y unos, estas reglas hacen que el proceso sea rápido y predecible. [4]

El sistema trata al 1 como el elemento neutro de la multiplicación, cuando se multiplica 1×1 , el resultado siempre es 1. [4]

La multiplicación completa requiere desplazar resultados parciales y sumarlos estos pasos se parecen al proceso manual de multiplicación decimal. [4]

EJEMPLO:

$$\begin{array}{r} 10100 \\ \times 1101 \\ \hline 10100 \\ 00000 \\ 10100 \\ 10100 \\ \hline 100000100 \end{array}$$

División Binaria

La división binaria funciona como la decimal, pero con números binarios, el sistema usa restas binarias para obtener el cociente y el residuo. [4]


El divisor se compara con partes del dividendo, y se resta si es menor o igual coloca un 1 en el cociente cuando realiza una resta exitosa. [4]

Si no se puede restar, el sistema coloca un 0 en el cociente y baja el siguiente bit, el proceso se repite hasta terminar con todos los bits del dividendo. [4]

EJEMPLO:

$$\begin{array}{r} 11011101 \mid 101 \\ \underline{-101} \\ 111 \\ \underline{-101} \\ 0101 \\ \underline{-101} \\ 00001 \end{array}$$

residuo



- [1] F. Yazdanpanah, C. Alvarez-Martinez, D. Jimenez-Gonzalez, and Y. Etsion, "Hybrid Dataflow/von-Neumann Architectures," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 6, pp. 1489–1509, Jun. 2014, doi: 10.1109/TPDS.2013.125.
- [2] E. Gregorio, Q. Gutiérrez, V. Manuel, H. Suárez, and A. Morales González, "UN MUNDO DE UNOS Y CEROS. NUEVOS RECURSOS DIDÁCTICOS DEL SISTEMA DE NUMERACIÓN BINARIO EN EDUCACIÓN PRIMARIA."
- [3] Tema, "Estructura de la información," 2017. Accessed: Jun. 07, 2025. [Online]. Available: <https://exa.unne.edu.ar/ingenieria/computacion/Tema3.pdf>
- [4] M. De Guzmán, "Aventuras matemáticas," *Madrid*, 1995.