



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

**TECNOLÓGICO NACIONAL DE MÉXICO**  
**INSTITUTO TECNOLÓGICO DE TLAXIACO**

---

**INVESTIGACIÓN DE INTERPRETACIÓN DE 0 Y 1 A NIVEL DE  
HARDWARE**

---

**CARRERA:**

INGENIERÍA EN SISTEMAS COMPUTACIONALES

**ASIGNATURA:**

ARQUITECTURA DE COMPUTADORAS

**SEMESTRE:** 5BS

**ALUMNO:**

CRUZ CRUZ DIEGO

**DOCENTE:**

OSORIO SALINAS EDWARD

Tlaxiaco Oax. 14 de octubre de 2024

***“Educación, ciencia y tecnología, progreso día con día” ®***



## INTRODUCCIÓN

Es común escuchar que las computadoras solo entienden ceros y unos. Lo vemos en gráficos, películas, e incluso en la publicidad. Esos famosos ceros y unos, que parecen ser el lenguaje secreto que "solo los hackers pueden entender", realmente son la base de cómo funcionan todos los sistemas digitales. Pero, ¿es esto una verdad absoluta o simplemente una creencia popular? ¿Cómo es posible que las computadoras, capaces de reproducir videos, mostrar imágenes, ejecutar inteligencia artificial y gestionar complejas?

Pues sí, es verdad. Además de todas estas funciones avanzadas, las computadoras solo trabajan con dos estados: encendido y apagado, que se representan como 1 y 0 en el sistema binario. Este sistema es la clave que permite a las máquinas modernas procesar todo tipo de información. En este artículo, vamos a descubrir cómo y por qué el sistema binario es fundamental para la informática, así como la fascinación.

## ¿Cómo funciona el sistema binario?

Para entender el sistema binario, primero pensemos en cómo funciona el sistema decimal, que usamos a diario. En decimal, tenemos 10 dígitos: del 0 al 9. Cuando llegamos al 9, pasamos a números de dos dígitos, comenzando con 10, 11, 12, y así sucesivamente. Luego, repetimos el patrón con 20, 30, y cuando agotamos los números de dos dígitos, pasamos a los de tres, como 100.

En binario, solo tenemos dos dígitos: 0 y 1. El proceso es similar. Primero contamos 0 y 1, pero al no haber más dígitos, el siguiente número es 10 (que equivale a 2 en decimal). Luego viene el 11 (que es 3 en decimal). Para representar el 4, necesitamos tres dígitos: 100. Seguimos con 101 (5 en decimal), 110 (6), 111 (7), y para el 8, usamos cuatro dígitos: 1000.



Diagram illustrating the sequence of binary numbers:

1 dígito:	0, 1	← binario
	0, 1	← decimal
2 dígitos:	10, 11	
	2, 3	
3 dígitos:	100, 101, 110, 111	
	4, 5, 6, 7	
4 dígitos:	1000	
	8	

Si te fijas, aumentamos un dígito solo en las potencias de 2 (2, 4, 8, 16, 32, 64, 128, 256, 512, 1024)

Una manera simple de convertir un número decimal a binario es realizar divisiones sucesivas por 2, guardando el resto de cada división. Veamos cómo convertir el número 11 a binario:



Diagram illustrating the conversion of the decimal number 11 to binary using successive division by 2:

	Cociente	Resto
11/2	5	1
5/2	2	1
2/2	1	0
11 (decimal) = 1011 (binario)		

El sistema binario es fundamental en la informática, y su base se encuentra en el trabajo de George Boole, quien en 1854 desarrolló el álgebra booleana, que incluye tres operaciones lógicas: AND, OR y NOT. Estas operaciones solo producen dos resultados: verdadero (1) y falso (0), y se utilizan para tomar decisiones lógicas.

Claude Shannon, en 1937, combinó el álgebra booleana con circuitos eléctricos en su trabajo sobre el análisis simbólico de circuitos, demostrando que se podían realizar operaciones matemáticas utilizando interruptores que representaban 0s y 1s. Esto dio origen a las puertas lógicas, esenciales para el diseño de computadoras.

En paralelo, Alan Turing presentó un modelo teórico de computadora en 1936, basado en un sistema binario. Aunque las primeras computadoras, como el Harvard Mark I y Mark II, eran decimales, la computación binaria se demostró más eficiente para circuitos electrónicos.

La primera computadora electrónica y binaria, Colossus, se construyó en 1943 para descifrar códigos nazis. Sin embargo, el ENIAC, lanzado en 1946, fue reconocido como la primera computadora moderna, aunque era decimal. Posteriormente, John Von Neumann desarrolló el EDVAC, la primera computadora binaria y de programa almacenado, que sentó las bases para las computadoras modernas.

Desde entonces, todas las computadoras han utilizado el sistema binario debido a su eficiencia en circuitos electrónicos, evolucionando de tubos de vacío a transistores y chips.

## **Del binario al ensamblador y lenguajes de programación**

El sistema binario permite a las computadoras realizar cálculos a gran velocidad, utilizando solo dos valores (1 y 0). Sin embargo, programar directamente en binario era un proceso laborioso y propenso a errores, donde los programadores usaban

tarjetas perforadas para escribir su código. Si ocurriera un error, debían revisar cada tarjeta manualmente.

Para simplificar esta tarea, se desarrolló el lenguaje ensamblador, que utiliza abreviaturas para instrucciones directas al procesador. El ensamblador es un programa que traduce estas instrucciones a binario, ya que las computadoras solo entienden ceros y unos.

A pesar de ser una mejora, el lenguaje ensamblador seguía siendo complicado y lento. Esto llevó a la creación de lenguajes de programación de alto nivel, que son más accesibles para los humanos. Sin embargo, estos lenguajes también necesitan ser convertidos a binario mediante un compilador. El primer compilador, creado por Grace Hopper en 1952 para el lenguaje Flow-Matic, fue un hito en la programación. Posteriormente, el lenguaje Fortran, lanzado en 1957, se convirtió en el primer lenguaje de alto nivel ampliamente utilizado, seguido por COBOL en 1959.

Aunque en la década de 1960 se empezó a adoptar el código binario por su eficiencia, el poder de cómputo actual permite que se podría usar el sistema decimal. Sin embargo, cambiar toda la infraestructura existente sería impracticable. Por lo tanto, el sistema binario sigue siendo la norma en la computación.

En contraste, la computación cuántica utiliza qubits, que son más poderosos que los bits binarios, aunque este tema es diferente. Los bits, abreviatura de "Binary Digit", son la unidad básica de información, y un byte (8 bits) puede representar 256 valores diferentes. Las unidades de almacenamiento se definen de la siguiente manera:

1 kilobyte (KB) = 1024 bytes

1 megabyte (MB) = 1024 KB

1 gigabyte (GB) = 1024 MB

1 Terabyte (TB) = 1024 GB

1 petabyte (PB) = 1024 TB

1 Exabyte (EB) = 1024 PB

En resumen, toda la información en computadoras, desde imágenes hasta código, se traduce a ceros y unos, lo que refleja la importancia del sistema binario en la informática.

## **CONCLUSIÓN.**

La historia de la computación demuestra que, aunque el sistema binario pueda parecer sencillo, su eficacia es fundamental para el funcionamiento de las computadoras modernas. Desde el arduo proceso de programar en binario hasta el desarrollo del lenguaje ensamblador y los lenguajes de alto nivel, cada avance ha simplificado y optimizado la programación. La creación de compiladores facilitó aún más esta evolución, permitiendo a los programadores centrarse en soluciones más complejas sin lidiar con la tediosa traducción a código binario. Aunque el sistema binario sigue siendo la norma, la computación cuántica abre nuevas posibilidades al utilizar qubits. A lo largo de esta historia, queda claro que todos los aspectos de la informática, desde el almacenamiento de datos hasta el procesamiento de información, dependiente de la representación binaria, consolidando su importancia en la era digital.

## **Referencias.**

<https://ed.team/blog/por-que-las-computadoras-solo-entienden-0-y-1-codigo-binario>