Fundamentos de computadores

Tema 1. INTRODUCCIÓN A LOS COMPUTADORES

Objetivos

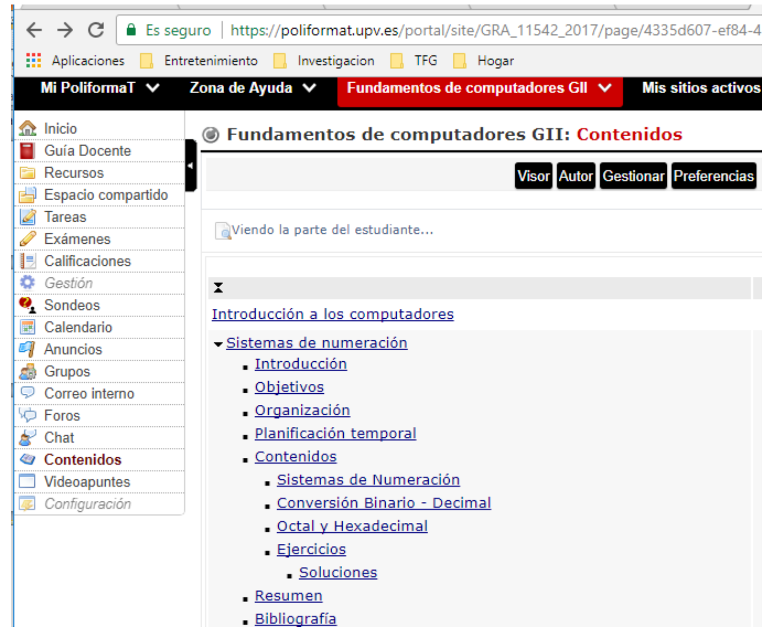
- Conocer los términos básicos de la asignatura
- Ofrecer una perspectiva histórica de los computadores
- Describir las unidades funcionales básicas de un computador
- Introducir los sistemas de representación básicos.

Bibliografía

- Introducción a los Computadores.
  - J. Sahuquillo y otros. Ed. SP-UPV, 1997 (ref. 97.491).
- Fundamentos de los computadores
  - P. de Miguel Miguel Anasagasti, (Ed. Thomson-Paraninfo, 9ª edición)
- Digital design : principles and practices
  - John F. Wakerly (Ed. Upper Saddle River : Pearson Prentice Hall, 2006)

Apartado de Contenidos en Poliformat FCO

- Poliformat, sección “Recursos”
  - Ejercicios sin solución.
  - Ejercicios solucionados.
  - Página web:
    - » *conversión binario – decimal.*
  - Exámenes de años anteriores.



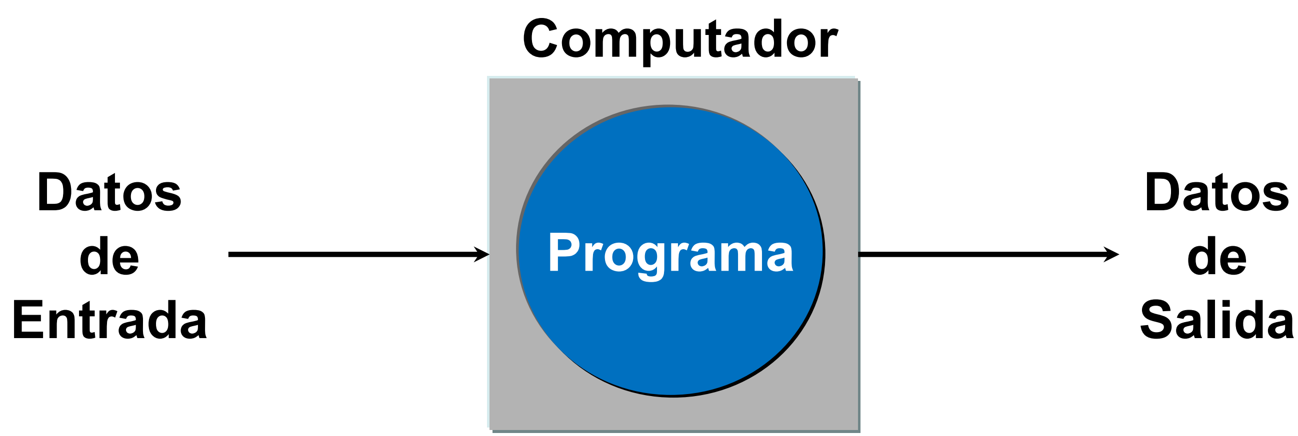
- Poliformat, sección “Lessons”
  - Módulo 2: Sistemas de numeración.
    - » *Incluye teoría y ejercicios*

Índice

- Introducción
- Historia y evolución
- Arquitectura Von Neumann
- Unidades funcionales del computador
- Sistemas de representación básicos

Introducción

- Informática → INFORmación + autoMÁTICA
- Computador → Máquina de programa almacenado
- Programa → Secuencia de instrucciones que se ejecuta de forma secuencial



Introducción

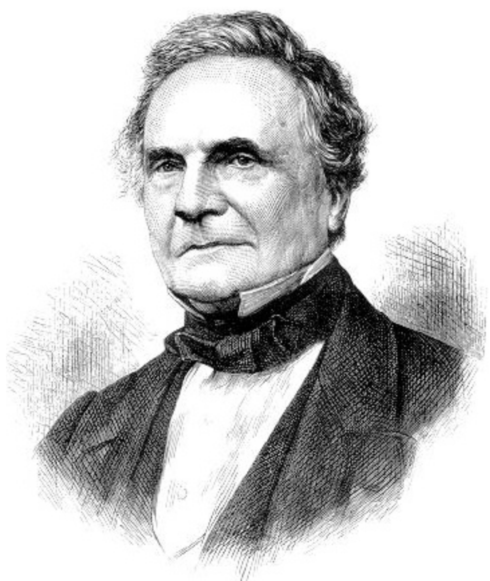
- Hardware → Conjunto de elementos tangibles
  (mecánicos o eléctricos)
- Software → Conjunto de elementos intangibles
  (sistema operativo, programas)
- Unidad Funcional del Computador →
  Circuito que realiza una tarea específica
- Bit → Unidad mínima (binaria) de información
  (0 ó 1)
- Byte → Unidad de información formada por 8 bits
  (2^8 = 256 combinaciones)

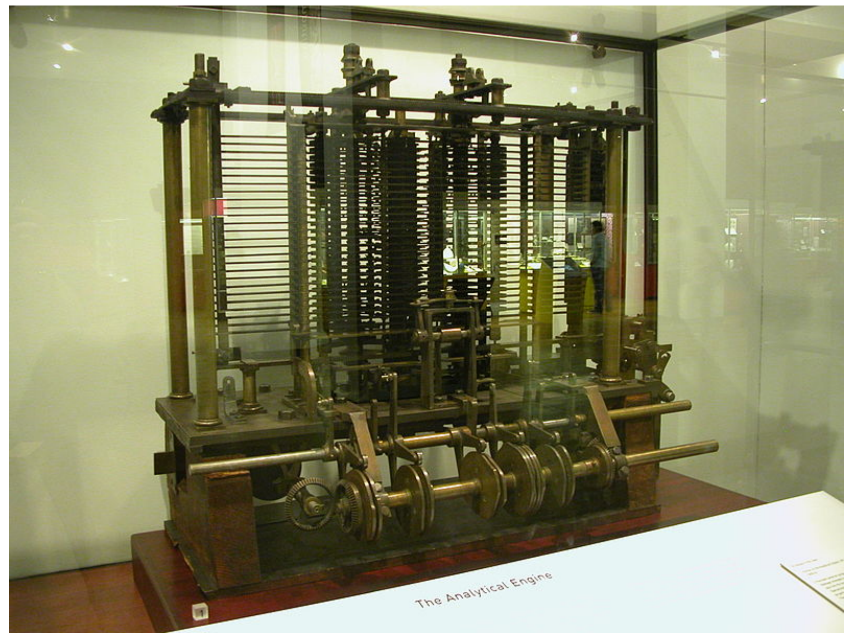
Índice

- Introducción
- Historia y evolución
- Arquitectura Von Neumann
- Unidades funcionales del computador
- Sistemas de representación básicos

Historia y evolución

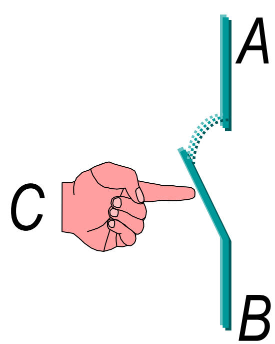
- El primer dispositivo considerado un computador programable fue diseñado por Charles Babbage en 1816.
  - Su máquina analítica era un dispositivo mecánico que usaba tarjetas perforadas para la introducción de programas y datos
  - Nunca se construyó en su totalidad.



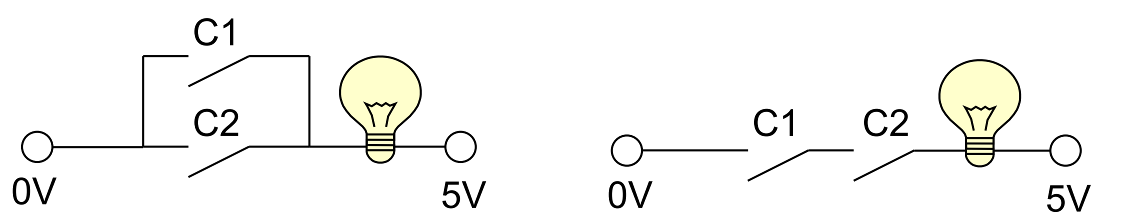


Historia y evolución

- La historia del computador moderno durante el siglo XX gira alrededor de la introducción y posterior evolución del interruptor electrónico (*electronic switch)*
  - Es un dispositivo que controla el paso de una corriente eléctrica en función de una señal eléctrica externa
  - Permite la implementación de operaciones lógicas sencillas que se combinan para construir un computador

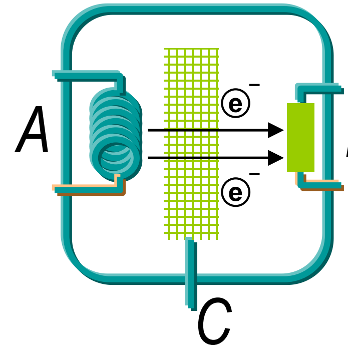


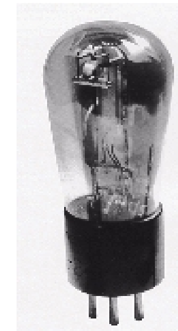
- Ejemplo: Bajo que condiciones se encenderán las bombillas?

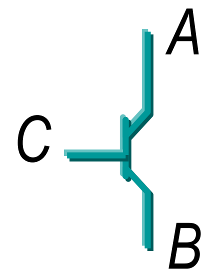


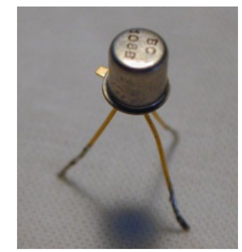
Historia y evolución

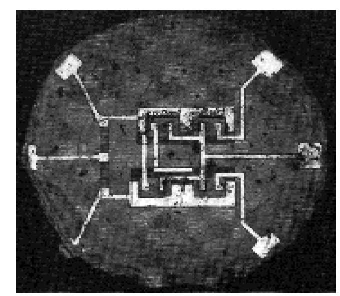
- Generaciones
  - Primera generación (1940-1956)
    - Válvulas de vacío
    - Alto consumo y disipación de calor
    - Baja fiabilidad
  - Segunda generación (1956-1963)
    - Transistor
    - Grandes mejoras en consumo, disipación y fiabilidad
    - Reduce costes e inicia el camino de la miniaturización
  - Tercera generación (1964-1971)
    - Circuitos integrados (chips) con múltiples transistores
    - Minicomputadores
  - Cuarta generación (1971-presente)
    - Microprocesador
    - Alta escala de integración
    - Computador personal



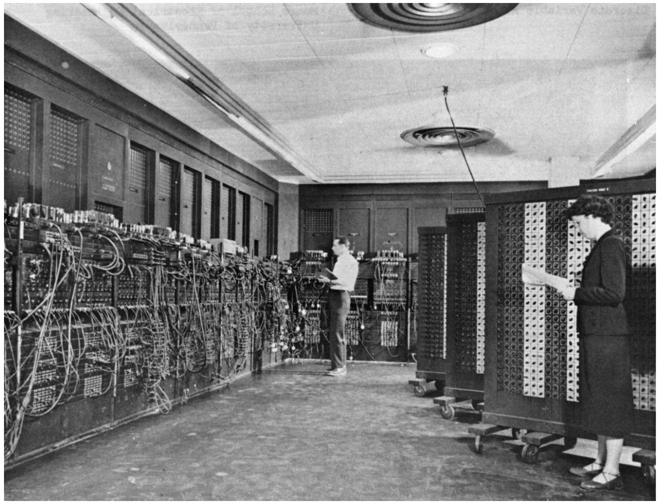






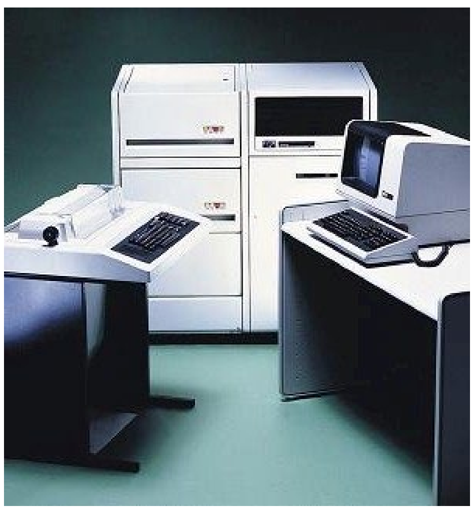


Historia y evolución



ENIAC
1ª gen.

IBM 608
2ª gen.



PDP-11
3ª gen.

Apple II
4ª gen.

Historia y evolución

- Generaciones
  - Quinta generación (1981-1991)
    - El gobierno japonés lanza el programa “quinta generación” junto a 6 empresas privadas, con objetivo de desarrollar un computador con “inteligencia humana”.
      - Respuesta a lenguaje natural
      - Capacidad de aprendizaje y organización autónoma.
      - Lenguaje máquina basado en programación lógica (tipo PROLOG)
    - Resultado: fracaso.
      - Se concluye que hace falta mejoras tecnológicas por descubrir para obtener unas buenas prestaciones en sistemas de este tipo.

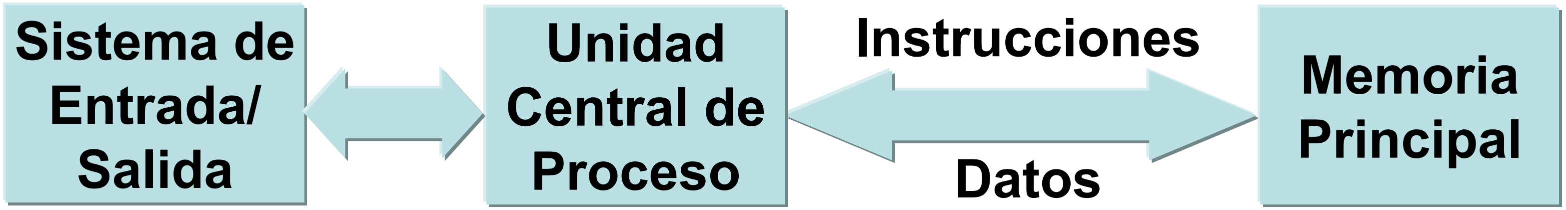
Historia y evolución

- Actualidad: en la bibliografía se dejan de clasificar los computadores por generaciones.
- La tecnología va avanzando siguiendo las siguientes líneas:
  - Computación cuántica, basada en las propiedades físicas de los átomos.
  - Procesadores multinúcleo
  - Grandes sistemas multicomputadores, exascale
  - Procesamiento distribuído y paralelo, computación en nube y grid
  - Computación y comunicaciones ubícuas (Internet, dispositivos móviles, redes sociales, telemedicina, etc.)
  - Aplicaciones de la inteligencia artificial (redes neuronales, sistemas expertos, sistemas de reconocimiento de voz, robótica, etc.)

Índice

- Introducción
- Historia y evolución
- Arquitectura Von Neumann
- Unidades funcionales del computador
- Sistemas de representación básicos

Arquitectura Von Neumann



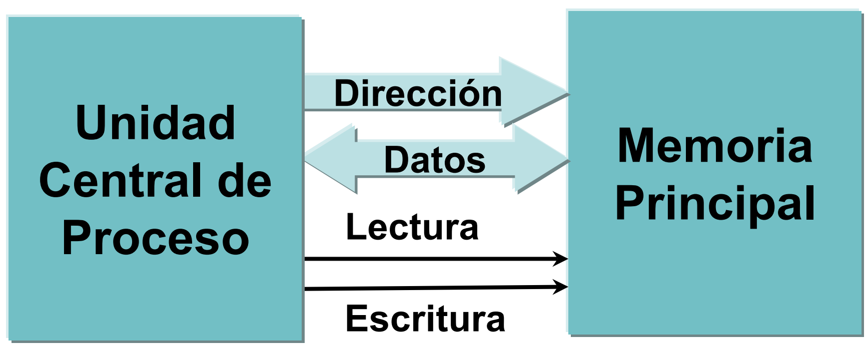
- Es la base de la inmensa mayoría de computadores actuales
  - La memoria principal almacena instrucciones y datos
  - La unidad central de proceso ejecuta instrucciones
  - La ejecución de una instrucción puede tener como consecuencia la lectura y/o escritura en memoria principal o el acceso al sistema de entrada/salida

Índice

- Introducción
- Historia y evolución
- Arquitectura Von Neumann
- Unidades funcionales del computador
- Sistemas de representación básicos

Unidades funcionales del computador FCO

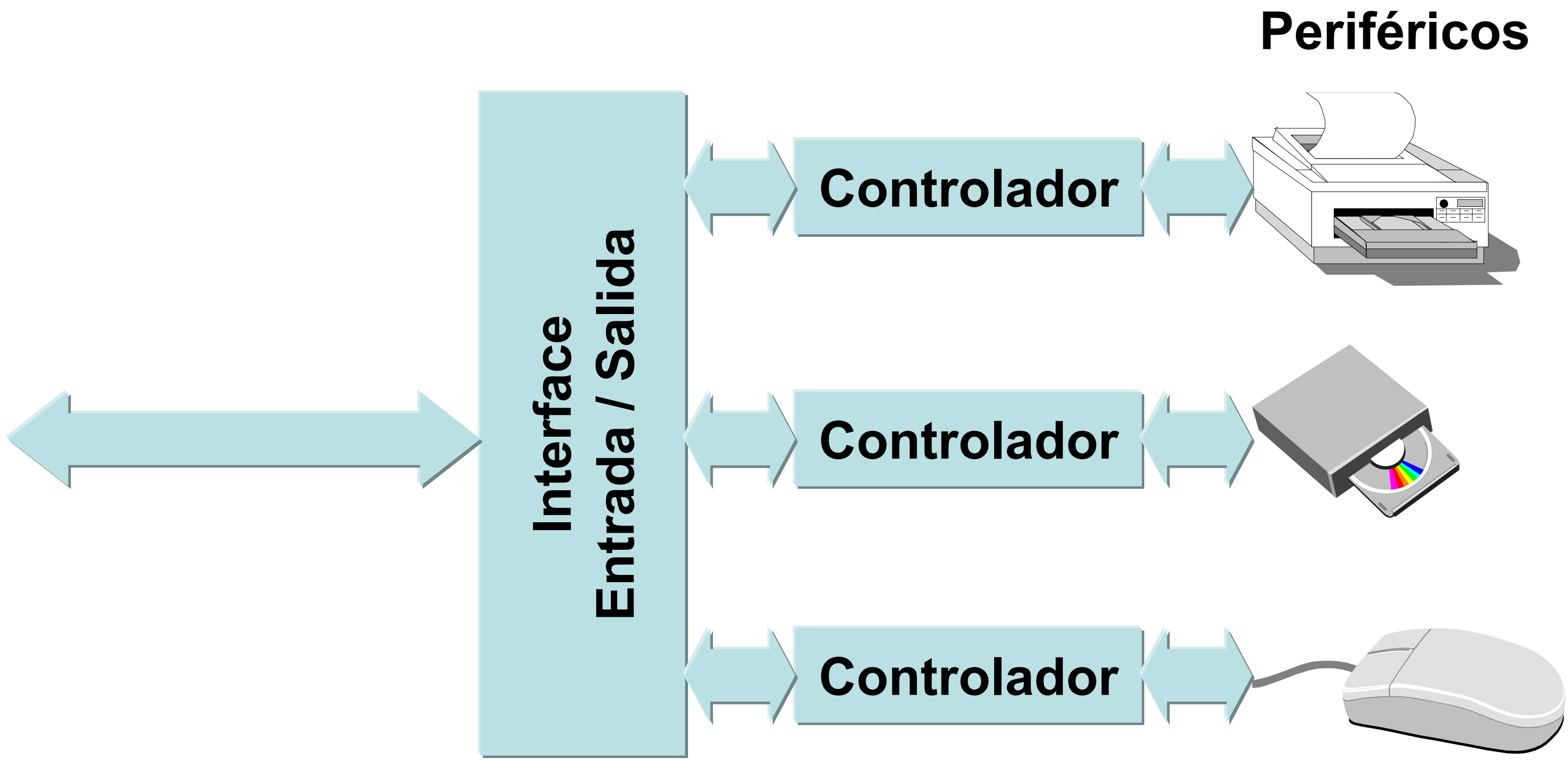
- Unidad Central de Proceso (UCP o CPU)
  - Es el componente que interpreta las instrucciones y procesa los datos contenidos en los programas
- Memoria Principal
  - Dispositivo de almacenamiento (permite lectura y escritura)
  - En general, el procesador accede a la memoria principal como si esta fuera un vector indexado por *direcciones*



DISCA

Unidades funcionales del computador

- Sistema de Entrada/Salida
  - Permite la comunicación de la UCP con el exterior



Unidades funcionales del computador FCO

- Periféricos
  - De entrada: Ratón, teclado, lápiz, pantalla táctil …
  - De salida: Pantalla, altavoz, impresora …
  - De almacenamiento: Disco duro, DVD, memoria flash …
  - De comunicación: Modem, red wireless, ethernet …
- UCP vs periféricos
  - Diferentes tecnologías
  - Diferentes tasas de transferencia de información
  - Diversidad de modos de operación (ej: R,W,RW) y funcionamiento
  - Diferentes formatos de representación de datos
- Interfaz o controlador
  - Dispositivo hardware/software que permite la comunicación entre la UCP y el periférico
  - Soluciona las diferencias entre la UCP y el periférico

Índice

- Introducción
- Historia y evolución
- Arquitectura Von Neumann
- Unidades funcionales del computador
- Sistemas de representación básicos

Sistemas de representación básicos

- Sistema de numeración
  - Conjunto de signos, reglas y convenciones que permiten expresar cantidades verbal y gráficamente
  - Ejemplo. Decimal, binario
- Base de un sistema de numeración
  - Número de símbolos distintos que se emplean. Cada uno de estos símbolos se denomina dígito
  - Ejemplo. Decimal (10 signos), binario (2 signos)
- Sistema de numeración posicional
  - Un número viene definido por una cadena de dígitos, donde cada uno de ellos está afectado por un factor de escala.
  - Aquél en el que el orden de los símbolos es importante
    - En decimal, $32 \neq 23$

Sistemas de representación básicos FCO

- En el sistema binario,
  - Base = 2, Dígitos = 0 y 1 (denominados bits)
  - Una cantidad N se representa mediante una secuencia de bits
    - Ejemplo. N = **1** 0 1 **1**$_2$

MSB LSB

(Most Significant Bit) (Least Significant Bit)

- En el sistema decimal,
  - Base = 10, Dígitos = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 Ejem: 34_{10}
- En el sistema hexadecimal,
  - Base = 16, Dígitos = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F Ejem: $4A_{16}$
- En el sistema octal,
  - Base = 8, Dígitos = 0, 1, 2, 3, 4, 5, 6, 7 Ejem: 450_8

Sistemas de representación básicos FCO

sistema binario $\rightarrow$ sistema decimal

- Para calcular la cantidad representada en decimal, se desarrolla el polinomio de potencias de la base
  - Ejemplo. $N = 1011_2 = 1x2^3 + 0x2^2 + 1x2^1 + 1x2^0 = 8 + 0 + 2 + 1 = 11_{10}$
  - Ejemplo. $R = 10{,}11_2 = 1x2^1 + 1x2^{-1} + 1x2^{-2} = 2 + 0{,}5 + 0{,}25 = 2{,}75_{10}$

sistema hexadecimal $\rightarrow$ sistema decimal

- El desarrollo de potencias de la base se puede utilizar para obtener la equivalencia decimal de cualquier cantidad representada en cualquier base (no sólo binario)
  - $N = 4A_{16} = 4x16^1 + Ax16^0 = 4x16^1 + 10x16^0 = 74_{10}$

Sistemas de representación básicos

- Algunas cantidades comunes

P.P.B.	Binario	Decimal
2^{-4}	0,0001	0,0625
2^{-3}	0,001	0,125
2^{-2}	0,01	0,25
2^{-1}	0,1	0,5
2^{0}	1	1
2^{1}	10	2
2^{2}	100	4
2^{3}	1000	8
2^{4}	10000	16
2^{5}	100000	32
2^{6}	1000000	64
2^{7}	10000000	128
2^{8}	100000000	256
2^{9}	1000000000	512
2^{10}	10000000000	1024
2^{11}	100000000000	2048

P.P.B.	Binario	Decimal
	0	0
2^{0}	1	1
2^{1}	10	2
$2^{1}+2^{0}$	11	3
2^{2}	100	4
$2^{2}+2^{0}$	101	5
$2^{2}+2^{1}$	110	6
$2^{2}+2^{1}+2^{0}$	111	7
2^{3}	1000	8
$2^{3}+2^{0}$	1001	9
$2^{3}+2^{1}$	1010	10
$2^{3}+2^{1}+2^{0}$	1011	11
$2^{3}+2^{2}$	1100	12
$2^{3}+2^{2}+2^{0}$	1101	13
$2^{3}+2^{2}+2^{1}$	1110	14
$2^{3}+2^{2}+2^{1}+2^{0}$	1111	15

Sistemas de representación básicos

sistema decimal → sistema binario

- Método de las divisiones sucesivas
  - Aplicable a números sin parte fraccionaria
  - Consiste en dividir la cantidad entre la nueva base (b=2). Mientras el cociente sea mayor o igual que la nueva base, dividir de nuevo (esta vez, sólo el cociente).
  - Una vez realizadas todas las divisiones, la secuencia de dígitos es la concatenación del último cociente y los restos de las divisiones anteriores, empezando por la última.
- Ejemplo: Pasar el número 348_{10} a binario

  $348 \div 2 = 174 \div 2 = 87 \div 2 = 43 \div 2 = 21 \div 2 = 10 \div 2 = 5 \div 2 = 2 \div 2 = \mathbf{1}$ (MSB)

  (LSB) **0** ← **0** ← **1** ← **1** ← **1** ← **0** ← **1** ← **0** ↵

  Solución: $348_{10} = 101011100_2$
- Este método también es útil para pasar de decimal a cualquier base (no sólo binario), se sustituye el 2 por la base.

Sistemas de representación básicos FCO

sistema decimal $\rightarrow$ sistema binario

- Método de las divisiones sucesivas
- Este método también es útil para pasar de decimal a cualquier base (no sólo binario), se sustituye el 2 por la base.

Ejemplo: sistema decimal $\rightarrow$ sistema octal

Ahora se divide por la base, 8.

Ejemplo: Pasar el número 348_{10} a octal

DISCA

sistema decimal → sistema binario

- Método de las multiplicaciones sucesivas
  - Aplicable a números que sólo tengan parte fraccionaria
  - Consiste en multiplicar el número por la nueva base (b=2). La parte entera resultante (0 ó 1) será uno de los dígitos de la secuencia
  - Aplicar de nuevo la multiplicación a la parte fraccionaria restante
- Ejemplo: convertir $0{,}375_{10}$ a base 2

```
0,375 x 2 = 0,750 → 0 (MSB)
0,750 x 2 = 1,50  → 1
0,50  x 2 = 1     → 1 (LSB)
```

Solución: $0{,}375_{10} = 0{,}011_2$

- Es posible que una cantidad que se representa con un número finito de dígitos en decimal requiera infinitos dígitos en binario (ejemplo: 0,9)
- Este método también es útil para pasar de decimal a cualquier base (no sólo binario), se sustituye el 2, por la base.

Sistemas de representación básicos

RESUMEN: sistema decimal $\rightarrow$ sistema en base b

- Conversión de un número $R = e,f_{10}$ a una base b
  - Convertir la parte entera (e), con lo que obtendremos una secuencia de dígitos de la base b, $a_n a_{n-1} \ldots a_1 a_0$
  - Convertir la parte fraccionaria (f), con lo que obtendremos otra secuencia de dígitos de la base b, $a_{-1} a_{-2} \ldots a_{-p}$
  - Reunir los dígitos que se han obtenido por separado, manteniendo la posición de la coma entre los dígitos de e y los de f
  - R en base b se escribe $a_n a_{n-1} \ldots a_1 a_0 \, , \, a_{-1} a_{-2} \ldots a_{-p}$
- Ejemplo: Convertir $10{,}375_{10}$ a binario
  - $10_{10} = 1010_2$ y $0{,}375_{10} = 0{,}011_2 \rightarrow 10{,}375_{10} = 1010{,}011_2$
  - Podemos verificar el resultado sin más que calcular el valor decimal de la secuencia binaria obtenida:
    $1010{,}011_2 = 2^3 + 2^1 + 2^{-2} + 2^{-3} = 8 + 2 + 0{,}25 + 0{,}125 = 10{,}375_{10}$

Sistemas de representación básicos

- En informática además del sistema binario, se utilizan también:
  - Octal (base 8 = 2^3)
    - Cada dígito octal representa un grupo de exactamente 3 bits
    - Dígitos octales: 0, 1, 2, 3, 4, 5, 6, 7
  - Hexadecimal (base 16 = 2^4)
    - Cada dígito hexadecimal representa un grupo de exactamente 4 bits
    - Dígitos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A (=10_{10}), B (=11_{10}), C (=12_{10}), D(=13_{10}), E (=14_{10}), F(=15_{10})
- Y su uso está extendido por
  - La facilidad de conversión a / desde binario, y
  - Porque permiten representar largas secuencias de bits con pocos dígitos (más fáciles de manejar que las secuencias de bits)

Sistemas de representación básicos

- Cambio de bases binaria, octal, hexadecimal
  - Dado que las bases octal y hexadecimal son potencias de 2 (la base binaria), se puede demostrar que
    - En octal (base 2^3) un dígito representa a un grupo de 3 bits
    - En hexadecimal (base 2^4) un dígito representa a un grupo de 4 bits
    - En ambos casos, el cambio de una representación a otra se realiza utilizando una tabla, agrupando los bits en bloques de 3 ó 4

Octal	Binario
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Hexadecimal	Binario
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111

Hex.	Binario
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

Sistemas de representación básicos FCO

sistema binario → octal o sistema binario → hexadecimal

- Cuando el grupo de 3/4 bits no está completo, se rellena con ceros
  - Ceros a la izquierda si los bits son de la parte entera
  - Ceros a la derecha si los bits son de la parte fraccionaria
- Un grupo de bits nunca puede incluir la coma
  - No se pueden mezclar bits de la parte entera y de la fraccionaria en el mismo grupo
  - Comenzar las agrupaciones alrededor de la coma

Relleno

$111000011011{,}10000001_2 = 111\ 000\ 011\ 011\ ,\ 100\ 000\ 01\underbrace{0}_{\text{Relleno}}{}_2 = 7033{,}402_8$

$111000011011{,}10000001_2 = 1110\ 0001\ 1011\ ,\ 1000\ 0001_2 = E1B{,}81_{16}$

sistema octal → hexadecimal o sistema hexadecimal → octal

– Pasando a binario estaríamos en el caso anterior:

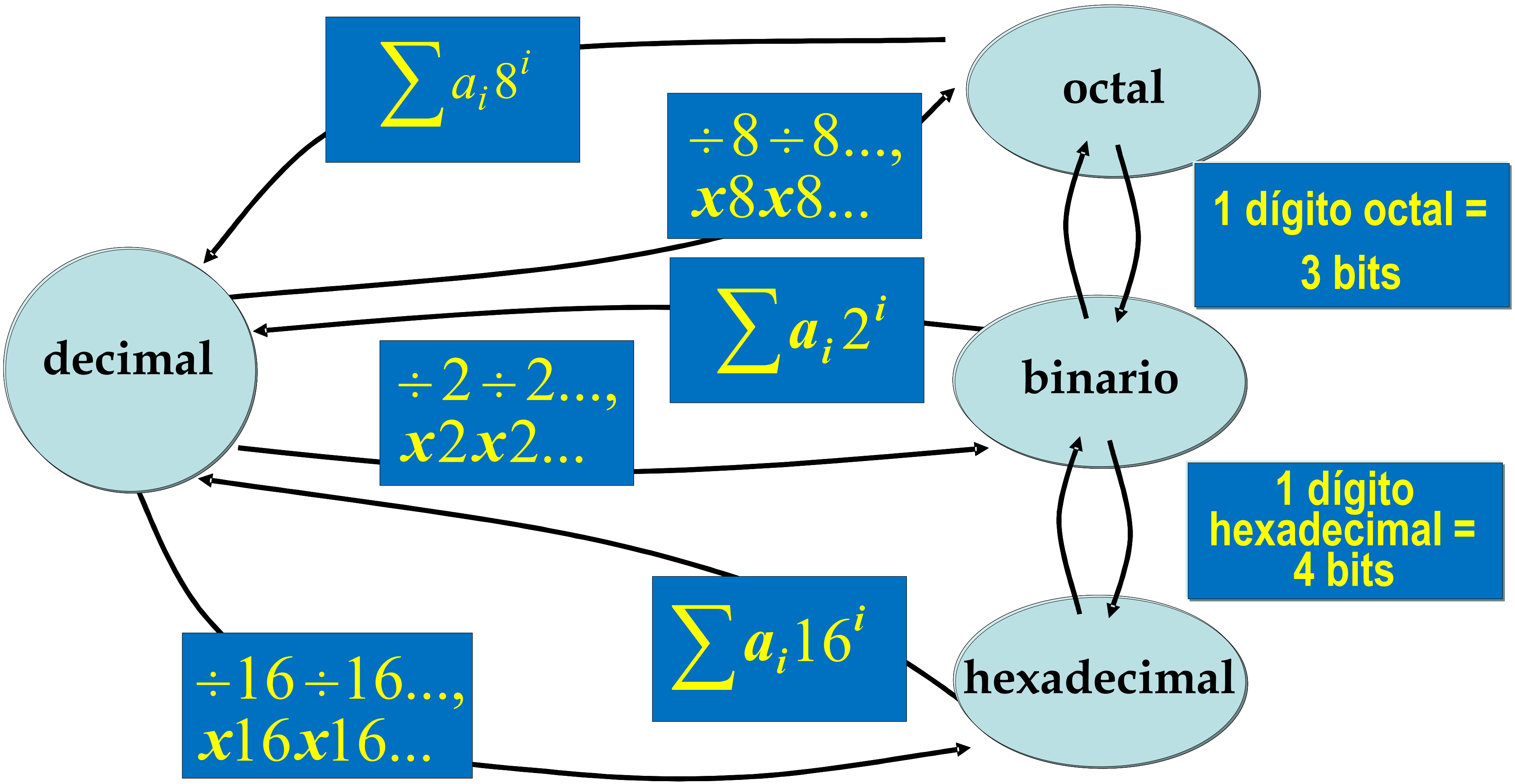
Relleno

$7033{,}402_8 = 111000011011{,}10000001_2$

$= 111\ 000\ 011\ 011,\ 100\ 000\ 010_2$

$= 1110\ 0001\ 1011,\ 1000\ 0001_2 = E1B{,}81_{16}$

Sistemas de representación básicos



Sistemas de representación básicos FCO

- Código BCD (Binary Coded Decimal)
  - Método sencillo de codificación de cantidades utilizando dígitos binarios
  - Se utilizan cuatro bits (denominados D, C, B y A), para codificar un dígito decimal
  - Cada dígito decimal se codifica por separado, mediante una tabla

Dígito decimal	Dígito BCD			
	D	***C***	***B***	***A***
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

- Ejemplo. Codificar 348_{10} en BCD
  $3_{10} = 0011_{BCD}$, $4_{10} = 0100_{BCD}$, $8_{10} = 1000_{BCD}$
  $348_{10} = 001101001000_{BCD}$

- Ejemplo. ¿Qué cantidad es 00101001_{BCD}?
  $0010_{BCD} = 2_{10}$, $1001_{BCD} = 9_{10}$
  $00101001_{BCD} = 29_{10}$