Fonaments de computadors

Tema 1. INTRODUCCIÓ ALS COMPUTADORS

Objectius

- Conèixer els termes bàsics de l'assignatura.
- Oferir una perspectiva històrica dels computadors.
- Descriure les unitats funcionals bàsiques d'un computador.
- Introduir els sistemes de representació bàsics.

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)

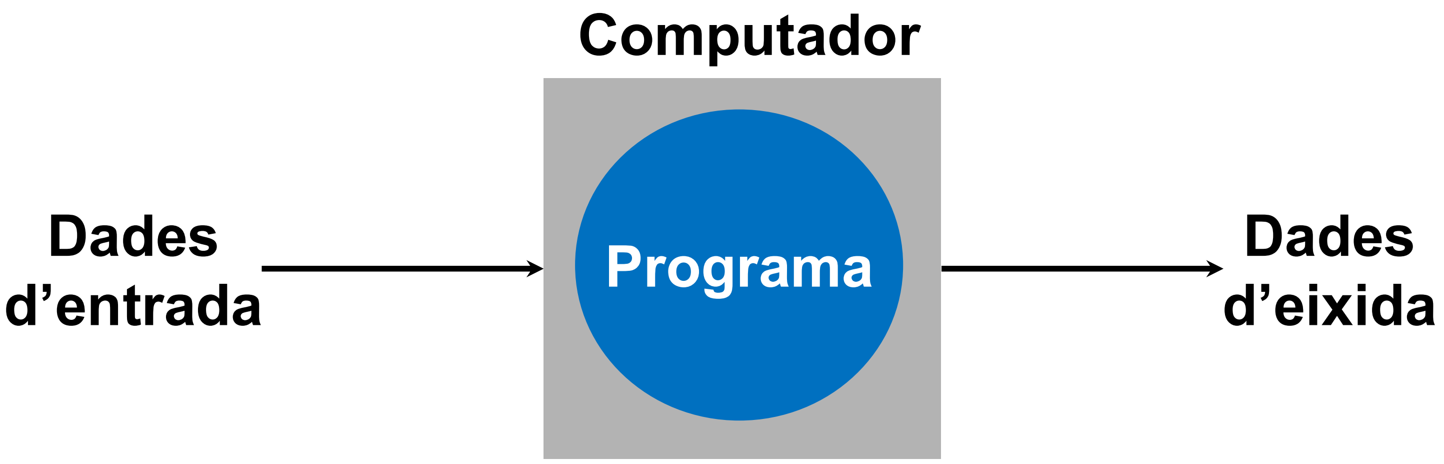
Apartado de Contenidos en Poliformat

Índex

- Introducció
- Història i evolució
- Arquitectura de Von Neumann
- Unitats funcionals del computador
- Sistemes de representació bàsics

Introducció

- Informàtica → INFORmació + autoMÀTICA
- Computador → Màquina de programa emmagatzemat
- Programa → Seqüència d'instruccions que
  s'executa de forma seqüencial

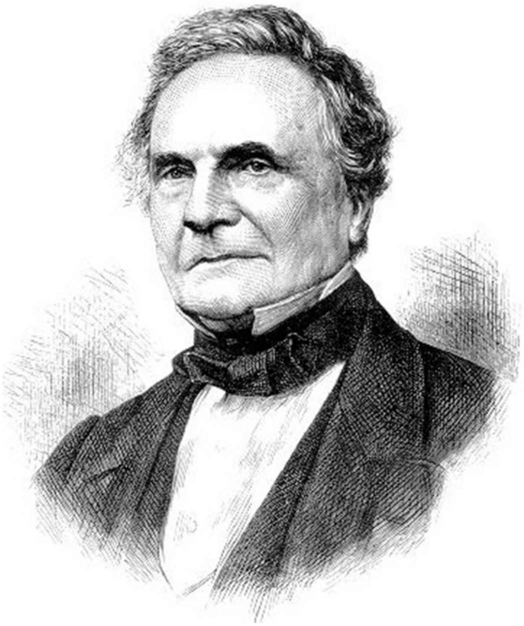


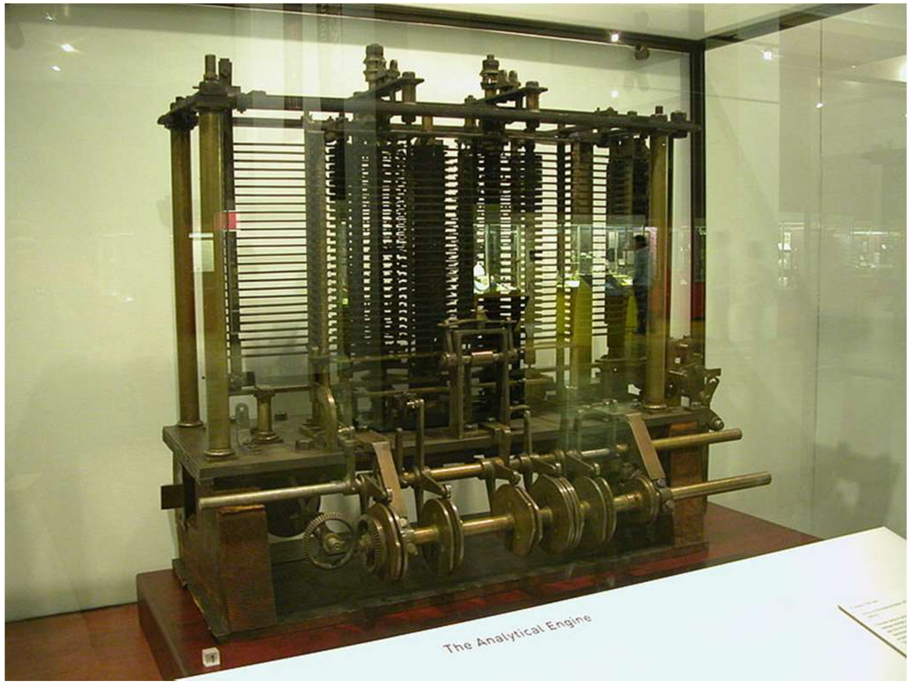
Introducció

- Maquinari → Conjunt d'elements tangibles
  (mecànics o elèctrics)
- Programari → Conjunt d'elements intangibles
  (sistema operatiu, programes)
- Unitat funcional del computador →
  Circuit que realitza una tasca específica
- Bit → Unitat mínima (binària) d'informació
  (0 o 1)
- Byte → Unitat d'informació formada per 8 bits
  (2^8 = 256 combinacions)

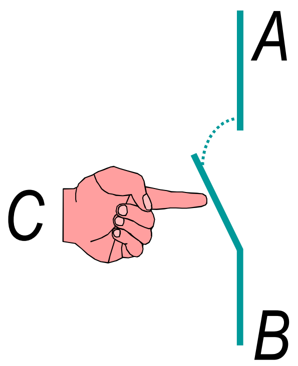
Història i evolució

- El primer dispositiu mecànic considerat un computador va ser dissenyat per Charles Babbage en 1816.
  - Aquesta màquina analítica era un dispositiu mecànic que utilitzava targetes perforades per a la introducció de programes i dades.
  - Mai va ser construïda en la seua totalitat

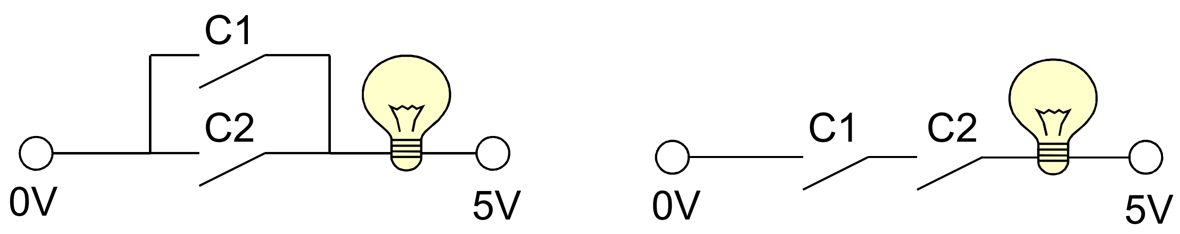




- La història del computador modern durant el segle XX gira al voltant de la introducció i posterior evolució de l'interruptor electrònic (*electronic switch*).
  - És un dispositiu que controla el pas d'un corrent elèctric en funció d'un senyal eléctric extern.
  - Permet la implementació d'operacions lògiques senzilles que es combinen per a construir un computador.

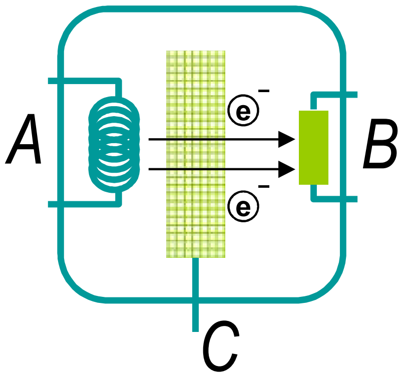


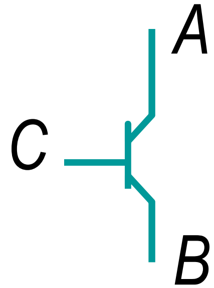
- Exemple: Amb quines condicions s'encendran les peretes?

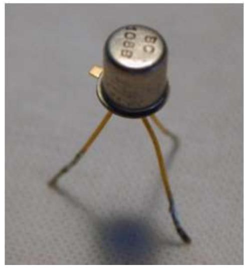


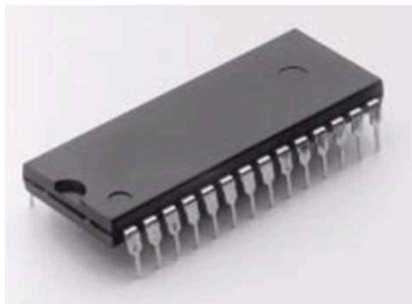
Història i evolució

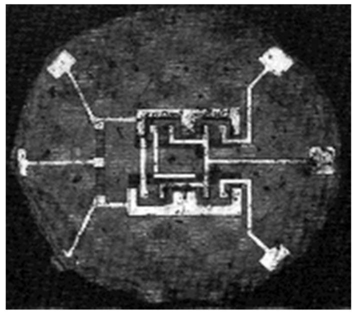
- Generacions
  - Primera generació (1940-1956)
    - Vàlvules de buit
    - Alt consum i dissipació de calor
    - Baixa fiabilitat
  - Segona generació (1956-1963)
    - Transistor
    - Grans millores en consum, dissipació i fiabilitat
    - Redueix costos i inicia el camí de la miniaturització
  - Tercera generació (1964-1971)
    - Circuits integrats (xips) amb múltiples transistors
    - Minicomputadors
  - Quarta generació (1971-present)
    - Microprocessador
    - Alta escala d'integració
    - Computador personal



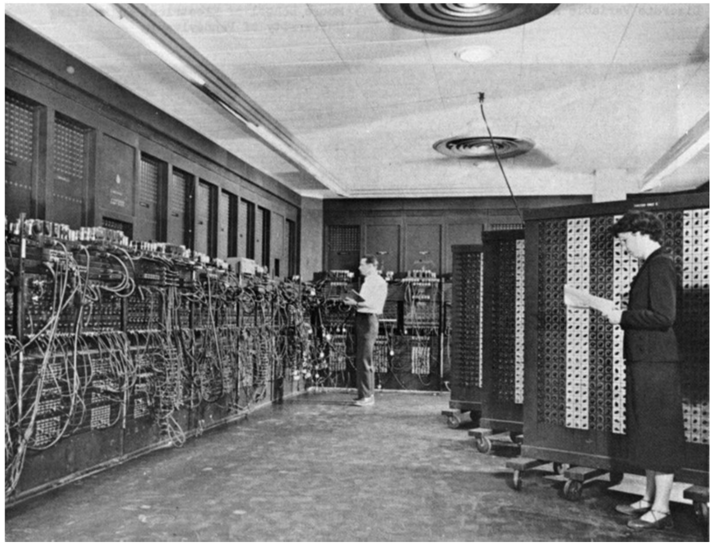






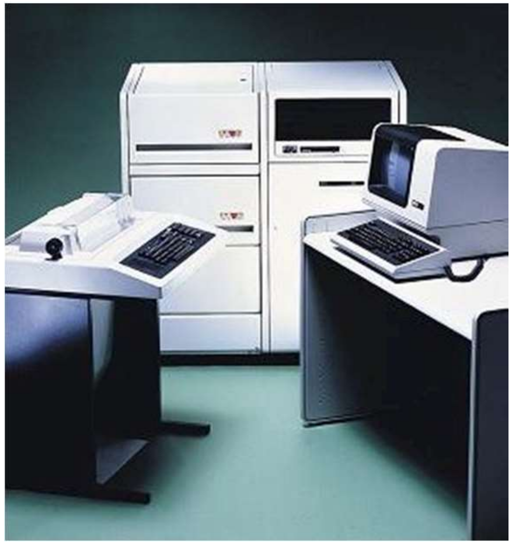


Història i evolució



ENIAC
1ª gen.

IBM 608
2ª gen.



PDP-11
3ª gen.

Apple II
4ª gen.

Història i evolució

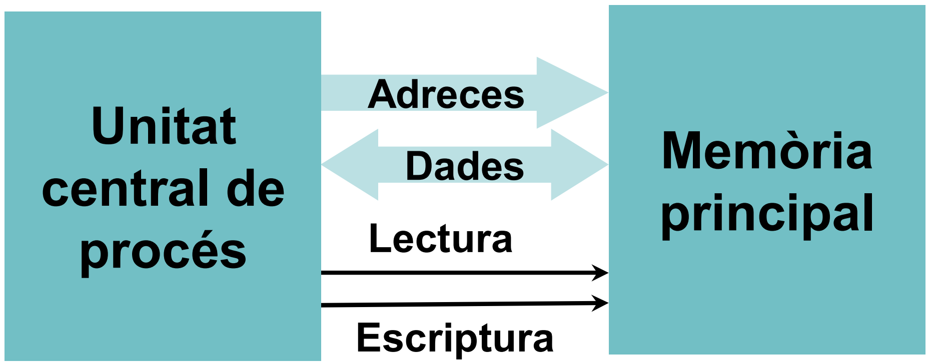
- Cinquena generació (present i futur)
  - Noves tecnologies (òptica, quàntica, etc.)
  - Processadors multinucli
  - Grans sistemes multicomputadors, exascale
  - Processament distribuït i paral·lel, computació en núvol i *grid*
  - Computació i comunicacions ubiqües (Internet, dispositius mòbils, xarxes socials, etc.)
  - Aplicacions de la intel·ligència artificial (xarxes neuronals, sistemes experts, sistemes de reconeixement de veu, robòtica, etc.)

Arquitectura de Von Neumann

- És la base de la immensa majoria de computadors actuals.
  - La memòria principal emmagatzema instruccions i dades.
  - La unitat central de procés executa instruccions.
  - L'execució d'una instrucció pot tindre com a conseqüència la lectura i/o escriptura en memòria principal o l'accés al sistema d'entrada/eixida.

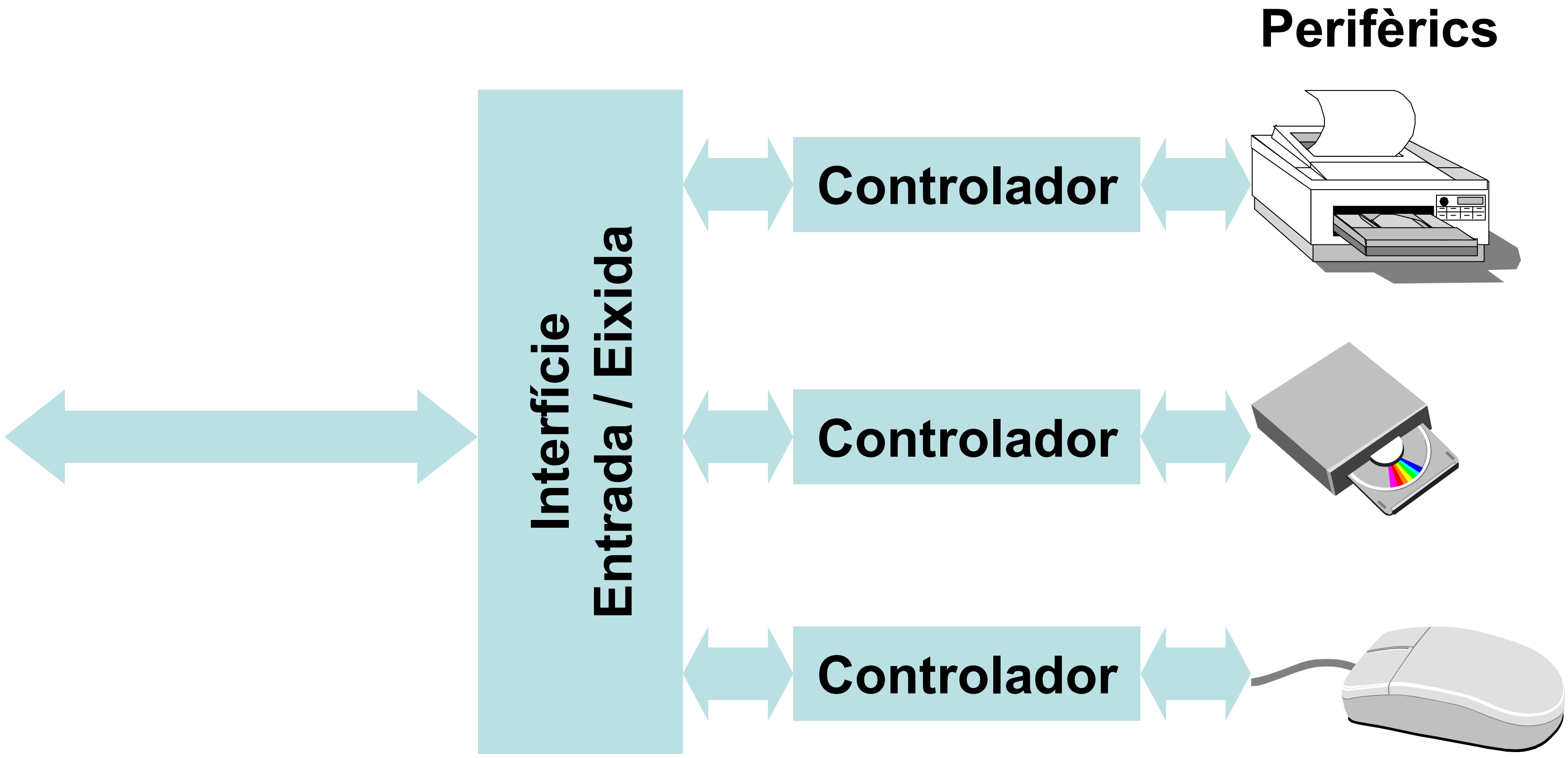
Unitats funcionals del computador

- Unitat central de procés (UCP o CPU)
  - És el component que interpreta les instruccions i processa les dades contingudes en els programes.
- Memòria principal
  - Dispositiu d'emmagatzemament (permet lectura i escriptura)
  - En general, el processador accedeix a la memòria principal com si aquesta fóra un vector indexat per *adreces.*



Unitats funcionals del computador

- Sistema d’entrada/eixida
  - Permet la comunicació de la UCP amb l'exterior.



Unitats funcionals del computador

- Perifèrics
  - D'entrada: ratolí, teclat, llapis òptic, pantalla tàctil…
  - D'eixida: pantalla, altaveu, impressora…
  - D'emmagatzemament: disc dur, DVD, memòria flaix…
  - De comunicació: mòdem, xarxa sense cable, Ethernet …
- UCP vs perifèrics
  - Diferents tecnologies
  - Diferents velocitats de transferència d'informació
  - Diversitat de modes d'operació (ex: R,W,RW) i funcionament
  - Diferents formats de representació de dades
- Interfície o controlador
  - Dispositiu maquinari/programari que permet la comunicació entre la UCP i el perifèric
  - Soluciona les diferències entre la UCP i el perifèric

Sistemes de representació bàsics FCO

- Sistema de numeració
  - Conjunt de signes, regles i convencions que permeten expressar quantitats verbalment i gràficament.
  - Exemple. Decimal, binari
- Base d'un sistema de numeració
  - Nombre de símbols distints que s'empren. Cada un d'aquests símbols es denomina dígit.
  - Exemple. Decimal (10 signes), binari (2 signes)
- Sistema de numeració posicional
  - Un nombre ve definit per una cadena de dígits, on cada un està afectat per un factor d'escala.
  - Aquell en què l'ordre dels símbols és important.
    - En decimal, $32 \neq 23$

Sistemes de representació bàsics

- En el sistema binari,
  - Base = 2, Dígits = 0 i 1 (anomenats bits)
  - Una quantitat N es representa per mitjà d'una seqüència de bits
    - Exemple. N = **1** 0 1 **1**

MSB (Most Significant Bit) LSB (Least Significant Bit)

- Per a calcular la quantitat representada, es calcula el <u>polinomi de potències de la base</u>
  - Exemple. $N = 1011_2 = 1x2^3 + 0x2^2 + 1x2^1 + 1x2^0 = 8 + 0 + 2 + 1 = 11_{10}$
  - Exemple. $R = 10{,}11_2 = 1x2^1 + 1x2^{-1} + 1x2^{-2} = 2 + 0{,}5 + 0{,}25 = 2{,}75_{10}$

- El polinomi de potències de la base es pot utilitzar per a obtindre l'equivalència decimal de qualsevol quantitat representada en qualsevol base (no sols binari).

Sistemes de representació bàsics

- Algunes quantitats comunes

P.P.B.	Binari	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.	Binari	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

- Canvi de base (decimal a binari)
  - Mètode de les divisions successives
    - Aplicable a nombres sense part fraccionària.
    - Consisteix a dividir la quantitat entre la nova base (b=2). Mentre el quocient siga major o igual que la nova base, dividim de nou (aquesta vegada, només el quocient).
    - Una vegada fetes totes les divisions, la seqüència de dígits és la concatenació de l'últim quocient i els residus de les divisions anteriors, començant per l'última.
  - Exemple: passem el nombre 348_{10} a binari.

    348÷2=174÷2=87÷2=43÷2=21÷2=10÷2=5÷2=2÷2= **1** (MSB)

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

    Solució: $348_{10} = 101011100_2$
  - Aquest mètode també és útil per a passar de decimal a qualsevol base (no sols binari).

Sistemes de representació bàsics

- Canvi de base (decimal a binari)
  - Mètode de les multiplicacions successives
    - Aplicable a nombres que només tenen part fraccionària.
    - Consisteix a multiplicar el nombre per la nova base (b=2). La part entera resultant (0 o 1) serà un dels dígits de la seqüència.
    - Apliquem de nou la multiplicació a la part fraccionària restant.
  - Exemple: convertim $0{,}375_{10}$ 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)
```

Solució: $0{,}375_{10} = 0{,}011_2$

  - És possible que una quantitat que es representa amb un nombre finit de dígits en decimal requerisca infinits dígits en binari (exemple: 0,9).
  - Aquest mètode també és útil per a passar de decimal a qualsevol base (no sols binari).

Sistemes de representació bàsics FCO

- Conversió d'un nombre R = e,f a una base b
  - Convertim la part entera (e), amb la qual cosa obtindrem una seqüència de dígits de la base b, $a_n a_{n-1} \ldots a_1 a_0$
  - Convertim la part fraccionària (f), amb la qual cosa obtindrem una altra seqüència de dígits de la base b, $a_{-1} a_{-2} \ldots a_{-p}$
  - Reunim els dígits que s'han obtingut per separat, mantenint la posició de la coma entre els dígits de e i els de f
  - R en base b s'escriu $a_n a_{n-1} \ldots a_1 a_0 \, , \, a_{-1} a_{-2} \ldots a_{-p}$
- Exemple: convertim $10{,}375_{10}$ a binari
  - $10_{10} = 1010_2$ i $0{,}375_{10} = 0{,}011_2 \rightarrow 10{,}375_{10} = 1010{,}011_2$
  - Podem verificar el resultat només calculant el valor decimal de la seqüència binària obtinguda:
    $1010{,}011_2 = 2^3 + 2^1 + 2^{-2} + 2^{-3} = 8 + 2 + 0{,}25 + 0{,}125 = 10{,}375_{10}$

Sistemes de representació bàsics

- A més del sistema binari, s'utilitzen també:
  - Octal (base 8 = 2^3)
    - Cada dígit octal representa un grup exactament de 3 bits.
    - Dígits octals: 0, 1, 2, 3, 4, 5, 6, 7
- Hexadecimal (base 16 = 2^4)
  - Cada dígit hexadecimal representa un grup exactament de 4 bits.
  - Dígits: 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})
- I l'ús d'aquest s'ha estés per
  - La facilitat de conversió a / des de binari, i
  - Perquè permeten representar llargues seqüències de bits amb pocs dígits (més fàcils de manejar que les seqüències de bits).

Sistemes de representació bàsics

- Canvi de bases binària, octal, hexadecimal
  - Atès que les bases octal i hexadecimal són potències de 2 (la base binària), es pot demostrar que
    - En octal (base 2^3) un dígit representa un grup de 3 bits.
    - En hexadecimal (base 2^4) un dígit representa un grup de 4 bits.
    - En els dos casos, el canvi d'una representació a una altra es realitza utilitzant una taula, agrupant els bits en blocs de 3 o 4.

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

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

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

Sistemes de representació bàsics

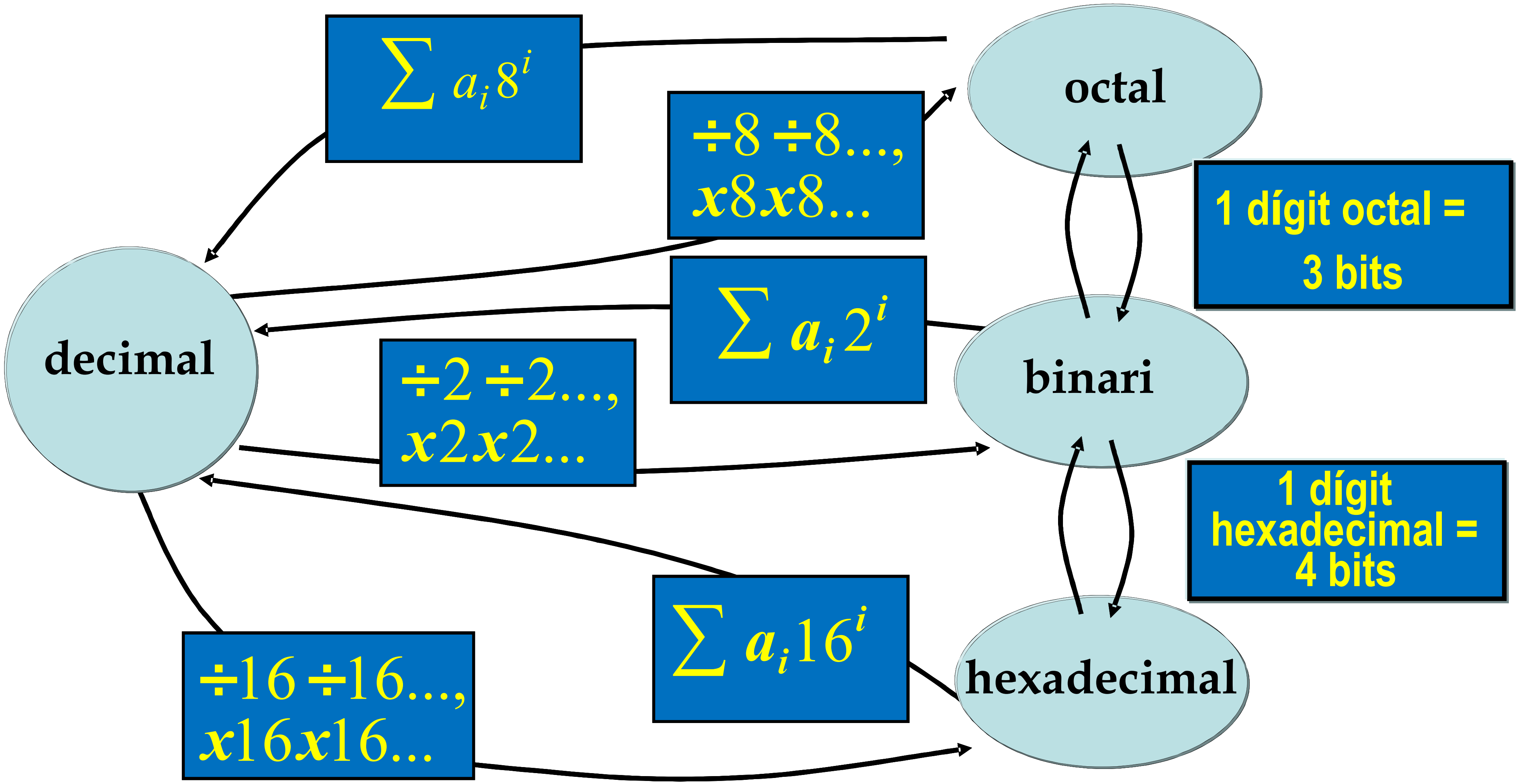
- Canvi a/de binari des d'octal i hexadecimal
  - Quan el grup de 3/4 bits no està complet, s'ompli amb zeros.
    - Zeros a l'esquerra si els bits són de la part entera.
    - Zeros a la dreta si els bits són de la part fraccionària.
  - Un grup de bits mai pot incloure la coma.
    - No es poden barrejar bits de la part entera i de la fraccionària en el mateix grup.
    - Cal començar les agrupacions al voltant de la coma.

Omplit amb zeros

$111000011011{,}10000001_2 = 111\ 000\ 011\ 011\ ,\ 100\ 000\ 010_2 = 7033{,}402_8$

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

Sistemes de representació bàsics



Sistemes de representació bàsics

- Codi BCD (Binary Coded Decimal)
  - Mètode senzill de codificació de quantitats utilitzant dígits binaris.
  - S'utilitzen quatre bits (denominats D, C, B i A), per a codificar un dígit decimal.
  - Cada dígit decimal es codifica per separat, per mitjà d'una taula.

Dígit decimal	Dígit 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

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

- Exemple. Quina quantitat és 00101001_{BCD}?
  $0010_{BCD} = 2_{10}$, $1001_{BCD} = 9_{10}$
  $00101001_{BCD} = 29_{10}$

Fonaments de computadors

Tema 1. INTRODUCCIÓ ALS COMPUTADORS