

# Codage informatique

**Achref El Mouelhi**

Docteur de l'université d'Aix-Marseille  
Chercheur en programmation par contrainte (IA)  
Ingénieur en génie logiciel

`elmouelhi.achref@gmail.com`

## Ordinateur

- Unité centrale = { processeur + mémoire + ... }
- Périphériques d'entrée, de sortie ou d'entrée/sortie

## Système informatique : codage binaire

- absence ou présence (d'électricité)
- faux ou vrai
- 0 ou 1

La plus petite unité de mesure (mémoire)

bit : **b**inary **d**igit

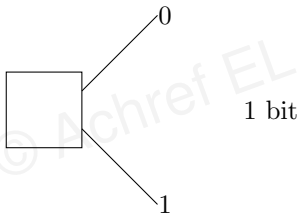


1 bit

# Codage informatique

La plus petite unité de mesure (mémoire)

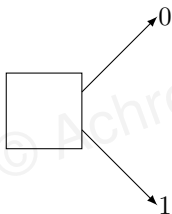
bit : **b**inary **d**igit



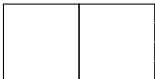
# Codage informatique

La plus petite unité de mesure (mémoire)

bit : **b**inary **d**igit

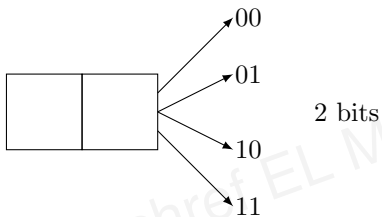


1 bit  $\longrightarrow$   $2^1$  combinaisons

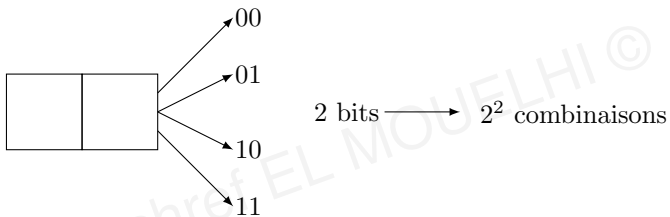


2 bits

© Achref EL MOUELHI ©



# Codage informatique



## Octet (Byte en anglais)

1 octet = 8 bits



1 octet  $\longrightarrow$   $2^8$  combinaisons

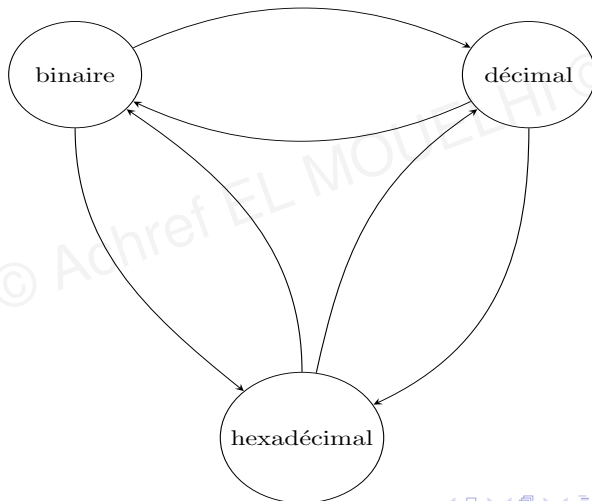
## Autres unités

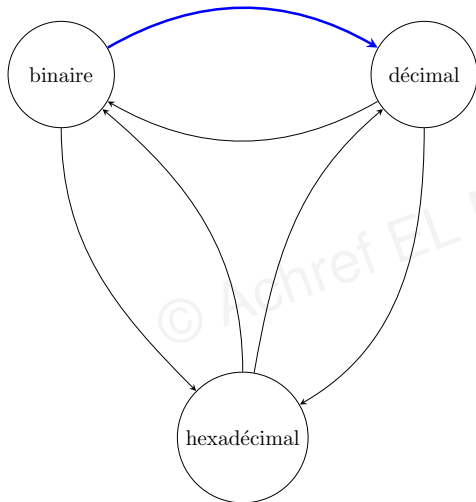
- 8 bits = 1 octet
- 1 kilo octet = 1 024 octets  $\equiv 10^3$  octets
- 1 méga octet  $\equiv 10^6$  octets
- 1 giga octet  $\equiv 10^9$  octets
- 1 téra octet  $\equiv 10^{12}$  octets
- 1 péta octet  $\equiv 10^{15}$  octets
- 1 exa octet  $\equiv 10^{18}$  octets
- 1 zéta octet  $\equiv 10^{21}$  octets
- 1 yotta octet  $\equiv 10^{24}$  octets
- 1 bronto octet  $\equiv 10^{27}$  octets
- 1 geop octet  $\equiv 10^{30}$  octets

## Quelques codages utilisés en informatique

- **binaire** : 0 ou 1
- **octal** : 0, 1, ... 7
- **décimal** : 0, 1, ... 9
- **hexadécimal** : 0, 1, ... 9, A, B, C, D, E, F
- ...

La conversion d'un nombre d'une base à une autre est possible





Exemple : 11010

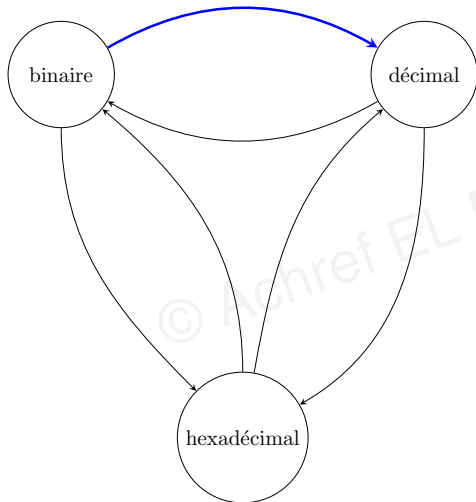
$(11010)_2$

$$= 1 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0$$

$$= 2^4 + 2^3 + 2^1$$

$$= 16 + 8 + 2$$

$$= (26)_{10}$$



Exemple : 11010

$(11010)_2$

$$= 1 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0$$

$$= 2^4 + 2^3 + 2^1$$

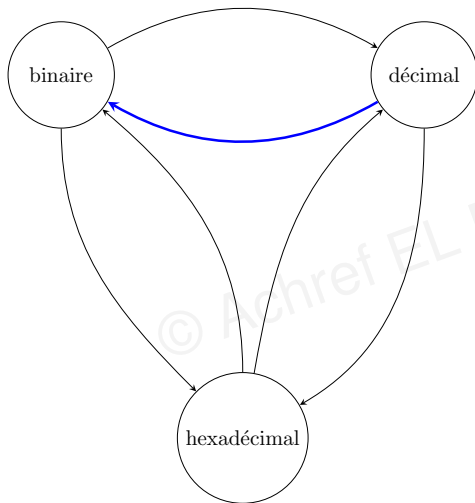
$$= 16 + 8 + 2$$

$$= (26)_{10}$$

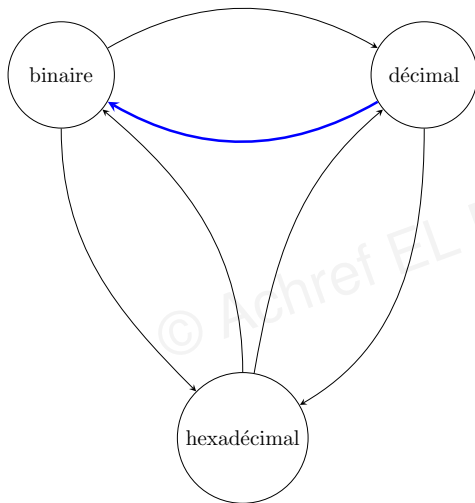
Exercice : montrez que

•  $(100010010)_2 = (274)_{10}$

•  $(10111010)_2 = (186)_{10}$

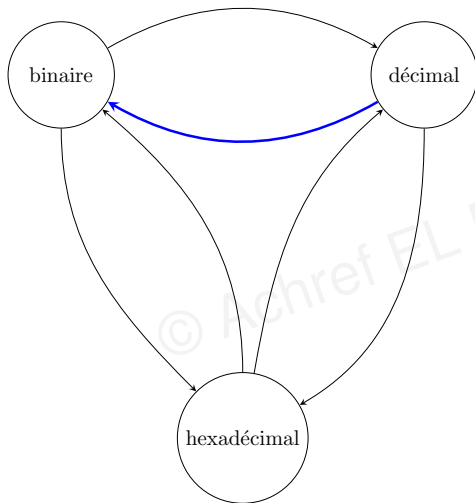


$$\begin{array}{r|l} 80 & 2 \\ \hline \end{array}$$

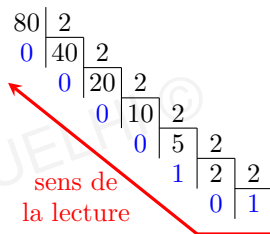
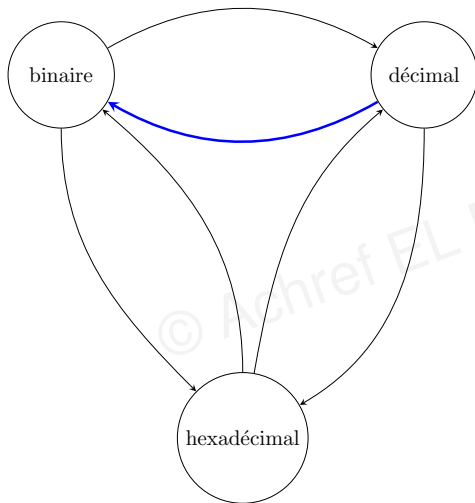


$$\begin{array}{r|l} 80 & 2 \\ \hline 0 & 40 \end{array}$$

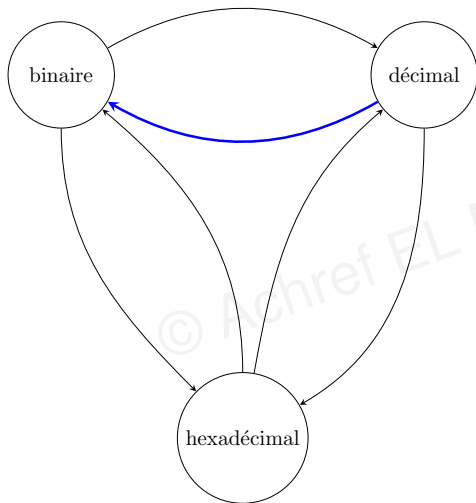
# Codage informatique



$$\begin{array}{r|l} 80 & 2 \\ \hline 0 & 40 \quad 2 \\ & 0 \quad 20 \quad 2 \\ & & 0 \quad 10 \quad 2 \\ & & & 0 \quad 5 \quad 2 \\ & & & & 1 \quad 2 \quad 2 \\ & & & & & 0 \quad 1 \end{array}$$



# Codage informatique



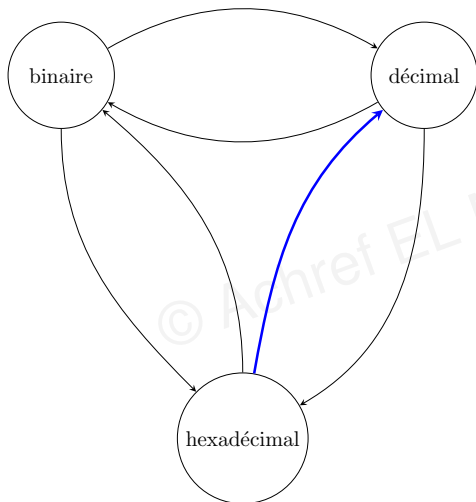
$$\begin{array}{r|l} 80 & 2 \\ \hline 0 & 40 \quad 2 \\ & 0 \quad 20 \quad 2 \\ & & 0 \quad 10 \quad 2 \\ & & & 0 \quad 5 \quad 2 \\ & & & & 1 \quad 2 \quad 2 \\ & & & & & 0 \quad 1 \end{array}$$

sens de la lecture

$$(80)_{10} = (1010000)_2$$

Exercice : trouvez le codage binaire de

- $(50)_{10}$
- $(257)_{10}$



Exemple : 11010

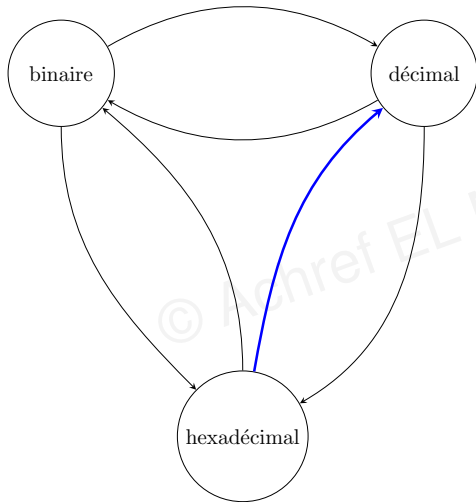
$(AFE2)_{16}$

$$= A * 16^3 + F * 16^2 + E * 16^1 + 2 * 16^0$$

$$= 10 * 16^3 + 15 * 16^2 + 14 * 16^1 + 2 * 16^0$$

$$= 40\,960 + 3\,840 + 224 + 2$$

$$= (45\,026)_{10}$$



Exemple : 11010

$(AFE2)_{16}$

$$= A * 16^3 + F * 16^2 + E * 16^1 + 2 * 16^0$$

$$= 10 * 16^3 + 15 * 16^2 + 14 * 16^1 + 2 * 16^0$$

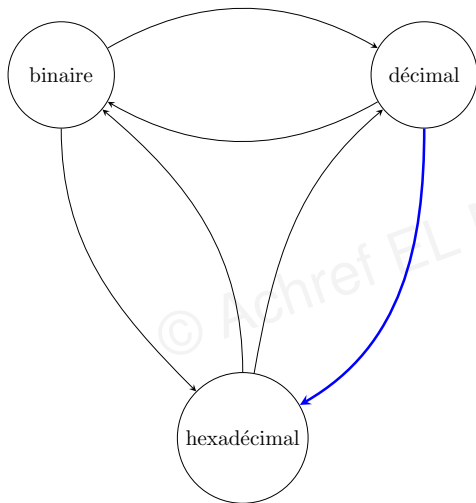
$$= 40\,960 + 3\,840 + 224 + 2$$

$$= (45\,026)_{10}$$

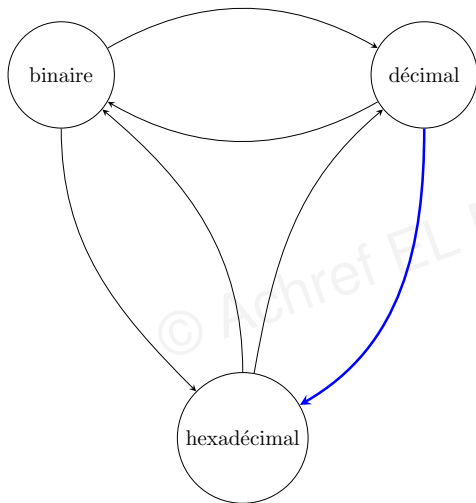
Exercice : trouvez le codage décimal de

•  $(EF3)_{16}$

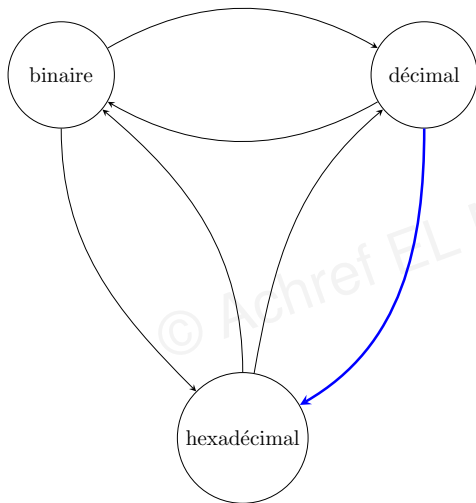
•  $(A2B9)_{16}$



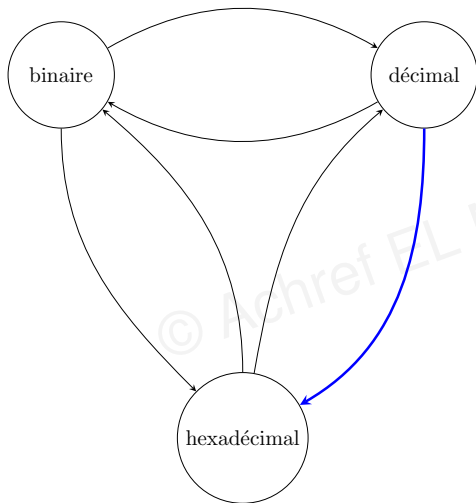
250 | 16



$$\begin{array}{r|l} 250 & 16 \\ \hline 10 & 15 \end{array}$$

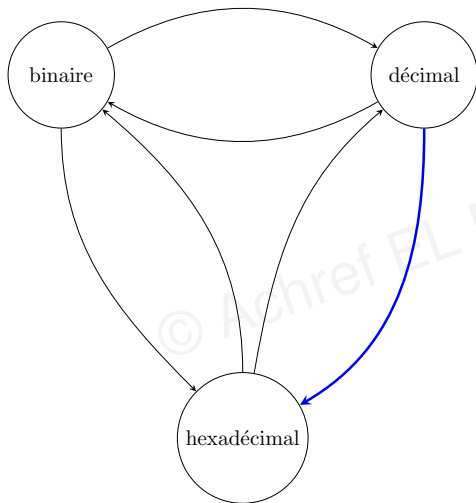


$$\begin{array}{r|l} 250 & 16 \\ \hline A & 15 \end{array}$$



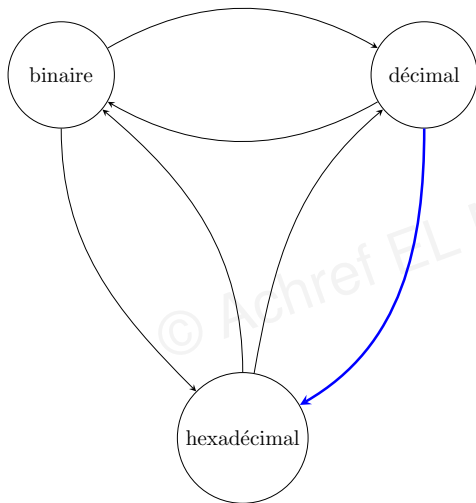
$$\begin{array}{r|l} 250 & 16 \\ \hline A & F \end{array}$$

# Codage informatique



250 | 16  
A | F

sens de  
la lecture



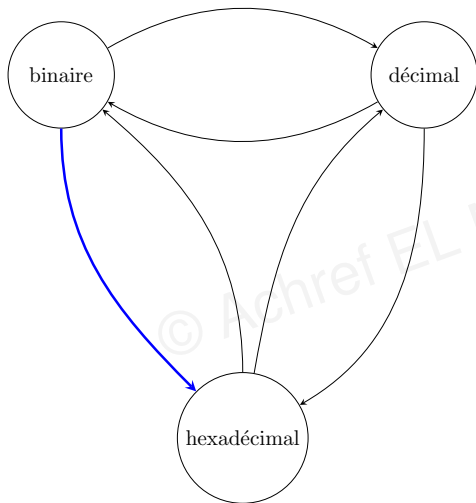
$$\begin{array}{r|l} 250 & 16 \\ \hline A & F \end{array}$$

sens de la lecture

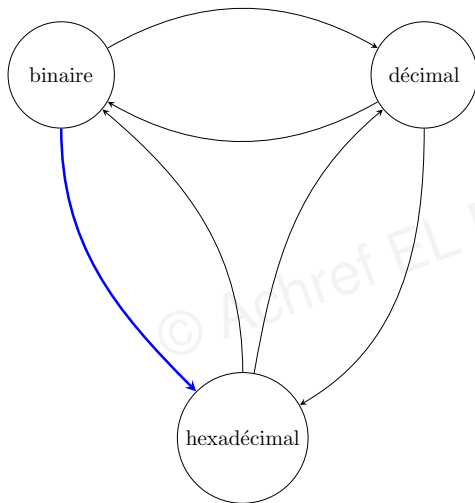
$$(250)_{10} = (FA)_{16}$$

Exercice : trouvez le codage hexadécimal de

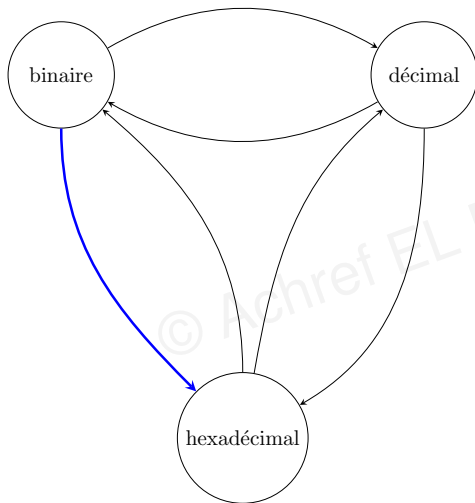
- $(1615)_{10}$
- $(2495)_{10}$



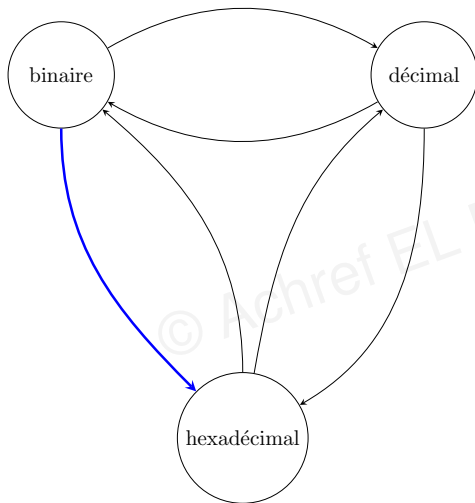
1 1 0 1 0 0 1 0



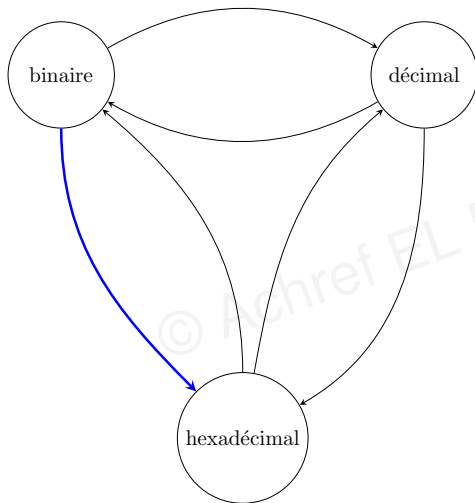
1 1 0 1, 0 0 1 0



1 1 0 1 0 0 1 0  
D 2



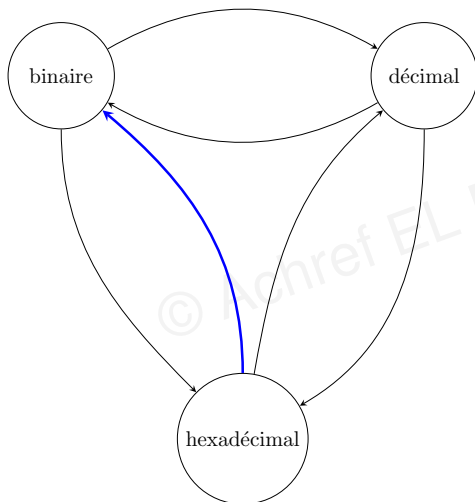
$$\begin{array}{c} \underline{1\ 1\ 0\ 1} \ \underline{0\ 0\ 1\ 0} \\ \text{D} \quad \quad \text{2} \\ (11010010)_2 = (D2)_{16} \end{array}$$



$$\begin{array}{ccc} \underline{1101} & \underline{0010} & \\ D & 2 & \\ (11010010)_2 & = & (D2)_{16} \end{array}$$

Exercice : trouvez le codage hexadécimal de

- $(1001011)_2$
- $(1001001000)_2$



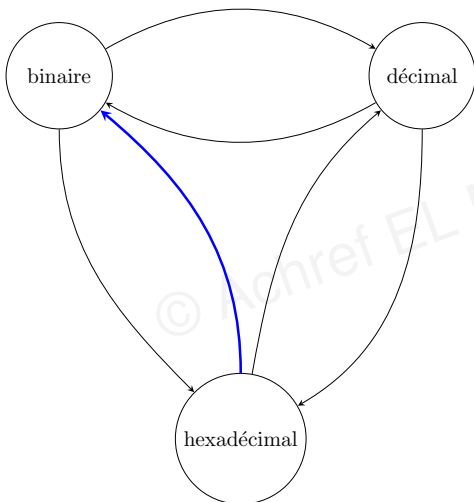
## Règle

Remplacer chaque caractère hexadécimal par son codage binaire sur 4 bits

$(F8D)_{16}$

$= 1111 + 1000 + 1101$  (+  $\Rightarrow$  concaténation)

$= (111110001101)_2$



## Règle

Remplacer chaque caractère hexadécimal par son codage binaire sur 4 bits

$(F8D)_{16}$

$= 1111 + 1000 + 1101$  (+  $\Rightarrow$  concaténation)

$= (111110001101)_2$

Exercice : trouvez le codage binaire de

•  $(23A)_{16}$

•  $(C17)_{16}$

## Question

Pourquoi un octet = 8 bits et pas 10, 100 ou 1 000 ?

© Achref EL MOU

## Question

Pourquoi un octet = 8 bits et pas 10, 100 ou 1 000 ?

## Réponse

L'octet est le plus petit nombre de bits pouvant contenir un seul caractère (son code **ASCII**).

## Code ASCII : American Standard Code for Information Interchange

- Standardisé dans les années 60
- Première version permettant de coder les caractères sur 7 bits (soit 128 caractères possibles)
  - 0 à 31 : pour les caractères de contrôle (retour à la ligne, tabulation, escape...)
  - 48 à 57 : pour les caractères numériques
  - 65 à 90 : pour les lettres en majuscule
  - 97 à 122 : pour les lettres en minuscule
- Deuxième version permettant de coder les caractères sur 8 bits (soit 256 caractères possibles) pour ajouter les lettres accentuées

Pour plus de détails, voir

[https://fr.wikibooks.org/wiki/Les\\_ASCII\\_de\\_0\\_%C3%A0\\_127/La\\_table\\_ASCII](https://fr.wikibooks.org/wiki/Les_ASCII_de_0_%C3%A0_127/La_table_ASCII)

## Autres codages

- **EBCDIC** (Extended Binary-Coded Decimal Interchange Code) : proposé par **IBM** et permettant de coder des caractères sur 8 bits.
- **Unicode** : permettant le codage de tous les alphabets (arabe, hébreu...) sur 16 bits ([Documentation officielle](#)).