Informatique-Electronique - AGIL Elec B

Electronique numérique : Représentation des nombres et systèmes numériques

Renato Martins, ICB UMR CNRS
Université Bourgogne Europe, 2025

Remerciements

- Les slides du cours sont basés pour la plupart sur le support gentillement mis à disposition par **Amira Bousselmi** et par de nombreuses autres personnes.

- Je n’ai pas crédité ces personnes dans la plupart des slides (ce qui n'est pas bien...mes excuses.)

Table des Matières

Chapitre 1 :
Numérisation et arithmétique binaire

Question : Quelles quantités représentent ces nombres/textes ?

10

77

FF

FFD700

C1. Systèmes logiques et numérisation

Question : Quelles quantités représentent ces nombres/textes ?

10

77

FF

FFD700

Question en retour : ils sont représentés dans quelle(s) base(s) ?

C1. Systèmes logiques et numérisation

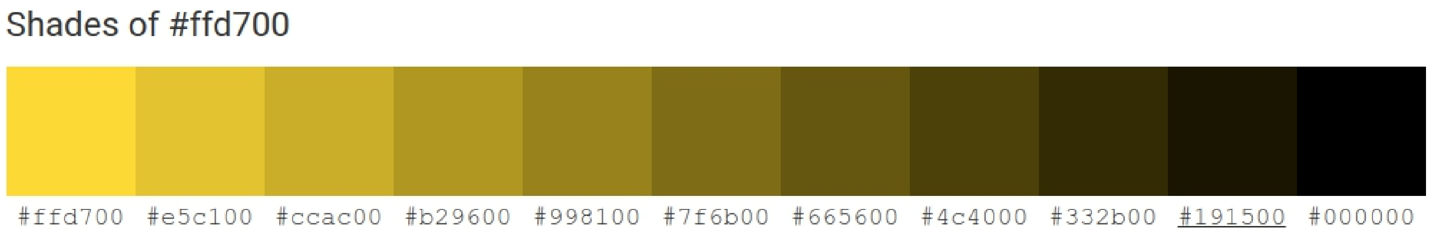
Question : Quelles quantités représentent ces nombres/textes ?

10

77

FF

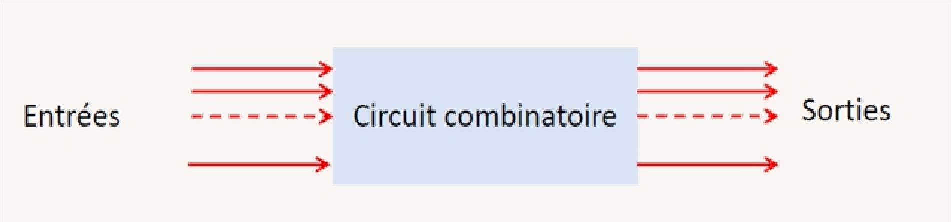
FFD700



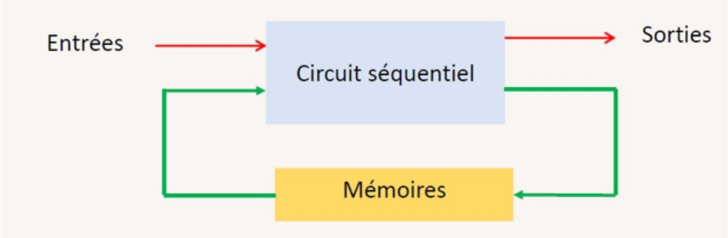
Question en retour : ils sont représentés dans quelle(s) base(s) ?

C1. Systèmes logiques et numérisation

1. Familles logiques



- **1. Logique combinatoire** :
  - La fonction de sortie dépend uniquement des variables d'entrée.
  - Un circuit combinatoire est défini par une ou plusieurs fonctions logiques.
- **2. Logique séquentielle :**
  - Les fonctions de sortie dependent non seulement de l’état des variables d’entrée mais également de l’état de certaines variables de sortie.



C1. Systèmes logiques et numérisation

2. Numérisation de l’information

- Un système de numération permet de coder une information (texte, image, nombre, parole ...) en lui attribuant un symbole ou combinaison de symboles compréhensible par le processeur.
- Le système conventionnel de comptage en base 10 est incompatible avec la machine, d'où la nécessité d'introduire d'autres systèmes de numération. Les systèmes de numération binaire et Hexadécimal sont les plus utilisés dans le domaine de l'électronique et de l'informatique.

- Il existe plusieurs systèmes de numérotation, appelés «**base**» :

  - Base Décimale, (10) : Les chiffres décimales sont : (0,1,2,3,4,5,6,7,8,9)
  - Base Binaire, (2) : Les chiffres binaires sont : (0,1) appelés bits ou digits
  - Base Octale, (8) : Les chiffres octales sont : (0,1,2,3,4,5,6,7)
  - Base Hexadécimale, (16) : Les chiffres hexadécimales sont : (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

C1. Systèmes logiques et numérisation

- En général, dans le système de numérotation de base « B » le nombre $(N)_B$ s'écrit :

$$(N)_B = (a_{n-1} \dots a_1\, a_0, a_{-1}\, a_{-2} \dots)_B = (a_{n-1}.B^{n-1} + a_{n-2}.B^{n-2} + \dots\dots\dots + a_1.B^1 + a_0.B^0 + a_{-1}.B^{-1} + a_{-2}.B^{-2} + \dots)$$

- **Exemple :**
  - En numérotation décimale, le nombre $(78{,}13)_{10}$ s'écrit :

$$(78{,}13)_{10} = 7.10^1 + 8.10^0 + 1.10^{-1} + 3.10^{-2}$$

  - En numérotation binaire, le nombre $(1011{,}101)_2$ s'écrit :

$$(1011{,}101)_2 = 1.2^3 + 0.2^2 + 1.2^1 + 1.2^0 + 1.2^{-1} + 0.2^{-2} + 1.2^{-3}$$

3. Conversion d'un nombre d'une base à une autre

Il s'agit du processus de conversion d'un nombre écrit dans une base b1 à une autre base b2.

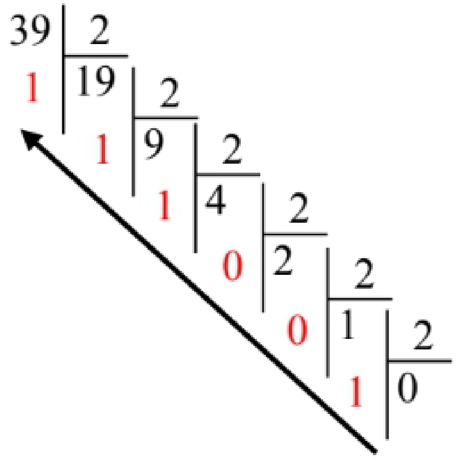
1- Le codage

La conversion d'un nombre décimal en une autre base non décimale est appelée codage

1.1 Codage en un nombre binaire

Le principe consiste à faire des divisions successives du nombre sur 2 et prendre le reste de la division dans l'ordre inverse.

- Exemple 1: $(39)_{10} = (?)_2$



- Après la division, le nombre recherché est ?

1.2 Codage en un nombre octal

On utilise la méthode de divisions successives par 8 jusqu'à un quotient égale à 0. Les restes successifs pris de bas en haut forment le nombre codé en octal.

- Exemple 1: $(423)_{10} = (?)_8$

1.3 Codage en un nombre hexadécimal

On utilise la méthode de divisions successives par 16 jusqu'à un quotient égale à 0. Les restes successifs pris de bas en haut forment le nombre code en hexadécimal.

- Exemple 1: $(423)_{10} = (1A7)_{16}$

2 - Le décodage

Le décodage est la conversion d'un nombre quelconque écrit en base m (octal, binaire ou hexadécimal) en base décimal. La forme générale est :

$$(y)_{10} = \sum_{n=0}^{i-1} X_n . m^n \qquad avec \begin{cases} m : la\ base \\ i : nombre\ de\ symbole \\ n : le\ rang \end{cases}$$

2 - Le décodage

Le décodage est la conversion d'un nombre quelconque écrit en base **m** (octal, binaire ou hexadécimal) en base décimal. La forme générale est :

$$(y)_{10} = \sum_{n=0}^{i-1} X_n . m^n \qquad avec \begin{cases} m : la\ base \\ i : nombre\ de\ symbole \\ n : le\ rang \end{cases}$$

- Exemples

$(100111)_2 = 1*2^5+0*2^4+0*2^3+1*2^2+1*2^1+1*2^0=(39)_{10}$

$(647)_8$ =?

$(1A7)_{16}$ =?

Rappel : *Tout nombre numérique peut s'écrire sous la forme suivante :*

$$N_{(B)} = C_n \cdot B^n + C_{n-1} \cdot B^{n-1} + \ldots\ldots + C_1 \cdot B^1 + C_0 \cdot B^0$$

- B est la base du système.
- C_n est le coefficient compris entre 0 et B-1.
- B_n est le poids du coefficient C_n.

Exercice 1 :

Convertir les nombres suivants dans les différentes bases mentionnées :

(1001001101)2 = (...)16 = (...)10
(E85)16 = (...)2 = (...)10
(279)10 = (...)2 = (...)16
(E85)16 = (...)2 = (...)10
(117)10 = (...)2 = (...)16

- **Exercice :** Quelle est la valeur maximale en decimal représentée avec 8 bits (en base binaire) ?

3 - Le transcodage

C’est la conversion d'une base A à une base B (A et B ne sont pas des bases décimales). L’idée est de convertir le nombre de base A à la base 10 puis coder le résultat à la base B.

Conversion Octal ➔ Binaire

En octal, chaque symbole de la base s'écrit sur 3 bits en binaire. L’idée est de remplacer chaque symbole dans la base octale par sa valeur en binaire.

Exemples :

$(34)_8 = (011100)_2$

Conversion Binaire ➔ Octal

L’idée est de faire des regroupements de **3** bits puis remplacer chaque regroupement par la valeur octale correspondante. Le regroupement se fait de droite à gauche.

Exemples :

$(11001010010110)_2 = (31226)_8$

- **Conversion Hexadécimal ➔ Binaire**

Chaque symbole de base hexadécimal s'écrit sur 4 bits

Hexadécimal ➔ $(1\ A\ F\ 3\ )_{16}$

Binaire ➔ $(0001\ 1010\ 1111\ 0011)_2$

- **Conversion Binaire ➔ Hexadécimal**

Binaire ➔ $(1101\ 0000\ 1100)_2$

Hexadécimal ➔ $(D0C)_{16}$

Exercice 1 : Convertir en binaire, octal et en hexadécimal, les nombres décimaux suivants :

a) 22_{10}

b) 63_{10}

c) 242_{10}

d) 3_{10}

Exercice 2 : Convertir les nombres suivants :

a) $(11011)_2 = (.....)_{10} = (.....)_8$

b) $(111101{,}101001)_2 = (.....)_{10}$

c) $(DA37)_{16} = (.....)_{10}$

C1. Systèmes logiques et numérisation

Exercice 1 : Convertir en binaire, octal et en hexadécimal, les nombres décimaux suivants :

a)	22_{10} = 10110	26	16
b)	63_{10} = 111 111	77	3F
c)	242_{10} = 11110010	362	F2
d)	3_{10} = 11	03	03

4 - Conversion des parties fractionnaires :

- **Multiplications successives par B.**
  - **Exemples**:

0,8125	
*2	
1,6250	MSB
*2	
1,2500	
*2	
0,5000	
*2	
1,0000	LSB

(0,8125)10 = (0,1101)2

0,8125	
*8	
6,5000	MSB
*8	
4,0000	LSB

(0,8125)10 = (0,64)8

Exercice 2 : Convertir en binaire les nombres décimaux suivants :

a) $(1{,}25)_{10}$

b) $(255{,}255)_{10}$

c) $(0{,}001)_{10}$