Les convertisseurs analogiques numériques, permettent d'assurer le traitement des informations sous forme numérique. Un changement de base est souvent nécessaire pour le transport de ces informations (base binaire, hexadécimale). La numération représente ce domaine.

1 La numération

1.1 Principe des systèmes de numération

Quel que soit le système de numération utilisé (binaire, octal, hexadécimal, …) tout nombre X peut se représenter par :

$$X = (a_n, a_{n-1}, a_{n-2}, \dots, a_1, a_0)_B$$

B : représente la base du nombre (2 pour binaire, 16 pour hexadécimale, …)
a : représente le coefficient
n : représente le rang

Ce nombre pourra s'exprimer dans la base décimale de la manière suivante :

$$X = \left(\sum_{i=0}^{n} a_i . \mathrm{B}^{\mathrm{i}} \right)_{10}$$

Ce principe d'écriture des nombres est appelé "numération positionnelle", c'est la position du symbole graphique qui donne sa valeur. Les systèmes de numération ainsi constitués sont dits pondérés.

En base B, avec n rangs, il y a B^n combinaisons. On peut donc compter de 0 à B^n-1.

1.2 Base décimale

Base : 10
Dix symboles : 0, 1, 2, 3, 4, 5, 6, 7, 8 et 9

Exemple : 2022 = $2.10^3 + 0.10^2 + 2.10^1 + 2.10^0$

chiffre, base, rang

1.3 Base binaire

Le système binaire est utilisé depuis le XIXème siècle, et les travaux du mathématicien George Boole.

Les informations stockées ou circulant dans un système numérique sont toujours physiquement binaires (porté par deux niveaux de tension 0 et 5 V par exemple).

Base : 2
Symboles : 0 et 1

Conversion en base 10

$$X = (a_n, a_{n-1}, \dots, a_1, a_0)_2 = \left(a_n . 2^n + a_{n-1} . 2^{n-1} + , \dots, a_1 . 2^1 + a_0\right)_{10}$$

- Un code à n chiffres en base 2 distingue 2^n états ou combinaisons.
- Un état binaire est appelé bit (**bi**nary digi**t**).
- Un bit prend les valeurs 0 ou 1.
- a_n sera nommé comme bit le plus significatif (**MSB** = Most Significant Bit)
- a_0 sera nommé comme bit le moins significatif (**LSB** = Low Significant Bit)
- Les puissances successives de 2 (2^0=1, 2^1=2, 2^2=4, 2^3=8, 2^4=16, 2^5=32, 2^6=64, 2^7=128, 2^8=256, 2^9=512, 2^{10}=1024, 2^{11}=2048, 2^{12}=4096...) sont appelées **poids binaires**.

1.4 Base hexadécimale

Ce système de numération est très utilisé dans les systèmes pour contracter la lecture et l'écriture des données binaires.

Base : 16
Symboles : 0, 1, 2, 3, 4, 5, 6, 7
8, 9, A, B, C, D, E, F

Conversion en base 10

$$X = (a_n, a_{n-1}, \dots, a_1, a_0)_{16}$$
$$= (a_n.16^n + a_{n-1}.16^{n-1} +, \dots, a_1.16^1, +a_0)_{10}$$

Un chiffre hexadécimal (de 0 à F) permet de coder de 0 à 15 (en base 10) avec 4 chiffres binaires (quartet).
Deux chiffres hexadécimaux (de 00 à FF) permettent de coder de 0 à 255 (en base 10) avec 8 chiffres binaires (octet).

Autres notations pour indiquer la nature hexadécimale du nombre :
$(F4)_{16}$ = **$**F4 = $(F4)_H$ = **16#**F4 = **0x**F4

1.5 Changement de base

1.5.1 base B - Décimale

Pour toute base B vers la base 10, on applique la formule polynomiale.

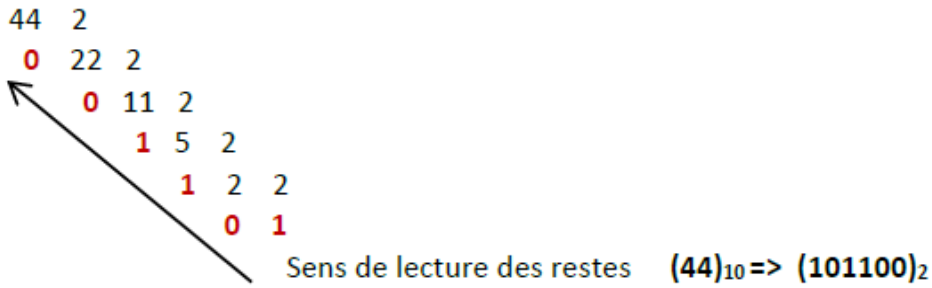
$$X = \sum_{i=0}^{n} a_i.B_i = (a_n.10^n + a_{n-1}.10^{n-1} +, \dots, a_1.10^1 + a_0)_{10}$$

1.5.2 Décimal – Binaire

On divise successivement par 2 le nombre décimal. Les restes correspondent aux bits consécutifs à lire en sens inverse d'apparition dans la division.

La division se termine au premier dividende nul.

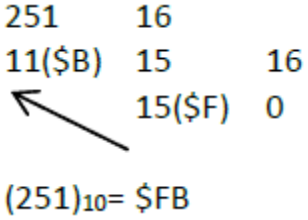
Exemple de transcodage décimal, binaire: $(44)_{10}$ => base 2



1.5.3 Décimal – Hexadécimal

Idem précédemment : **on divise par 16 et on lit les restes dans l'ordre inverse d'apparition**.

Ex : $(251)_{10}$



$(251)_{10}$= $FB

Cette stratégie peut servir à réaliser plus rapidement la conversion décimal / binaire d'un grand nombre.

1.5.4 Binaire – Hexadécimal

On regroupe les bits 4 par 4 (en quartets).

Chaque quartet peut alors être converti directement en hexadécimal (max. : $1111_2 = F_{16}$).

Ex : $1111\ 1011_2 = FB_{16}$

1.5.5 Hexadécimal – Binaire

Chaque symbole hexadécimal (chiffre ou lettre) est converti dans le quartet binaire correspondant.

Ex : $F0A8_{16}$ = (1111 0000 1010 1000)$_2$ = (1111000010101000)$_2$

2 Codage dans les systèmes numériques

2.1 Codes pondérés

2.1.1 Code binaire

Chaque bit est représenté par son rang, bit de poids fort à gauche, faible à droite

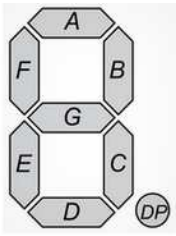
Base binaire			Base décimale
Rang n : **bit de poids fort (MSB)**	Rang 1	Rang 0 : **bit de poids faible (LSB)**	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

2.1.2 Code Décimal Codé Binaire (DCB)

Le codage BCD consiste à :

- conserver le codage binaire sur 4 bits pour tous les chiffres allant de 0 à 9,
- ne pas utiliser les codes sur 4 bits allant de 10 à 15 (respectivement 1010_2 à 1111_2).

Ce codage permet une lecture décimale directe à partir d'un nombre binaire, il est surtout utilisé pour l'affichage (afficheur 7 segments notamment).



Exemple : $582_{10} \equiv (0101\ 1000\ 0010)_{DCB}$

Remarque :

- Le code DCB (ou BCD) est un code pondéré à condition de propager la retenue en base 10 entre deux quartets.
- Dans le cas d'un affichage sept segments, pour lequel il est très utilisé, il faut prévoir un décodage DCB / 7 segments.

2.2 Code non pondéré : code Gray ou binaire réfléchi

	Gray			
	d	c	b	a
0	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	0	1	0
4	0	1	1	0
5	0	1	1	1
6	0	1	0	1
7	0	1	0	0
8	1	1	0	0
9	1	1	0	1
10	1	1	1	1
11	1	1	1	0
12	1	0	1	0
13	1	0	1	1
14	1	0	0	1
15	1	0	0	0

Ce code supprime les aléas découlant du changement simultané de plusieurs variables (a, b, c ou d du tableau ci-dessous) pour deux combinaisons adjacentes du code binaire pur.

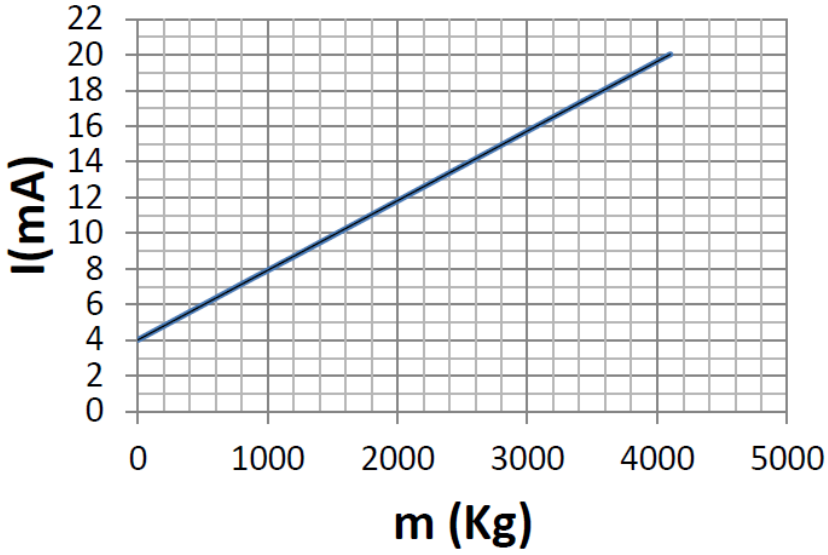
Il est construit par symétrie (voir les traits horizontaux du tableau ci-dessous) d'où le nom de réfléchi.

Remarque :

- C'est un code CYCLIQUE : pour passer de la position 15 à la position 0, il n'y a qu'un seul changement d'état.
- Le code étant non pondéré (il ne permet pas de faire du calcul), on n'affecte pas de poids aux colonnes mais un rang ou un symbole (d, c, b, a).
- Code utilisé notamment dans les codeurs de positions absolus.

Exercice : Capteur analogique et numérique

Un capteur-transmetteur linéaire délivre un signal 4 / 20 mA pour une masse m variant de 0 à 4095 Kg.



I- La caractéristique du capteur-transmetteur est donnée ci-contre.

1) Donner la valeur de la sensibilité du capteur-transmetteur. Préciser l'unité.

2) Donner l'équation de la caractéristique du capteur-transmetteur I = f(m).
3) Quelle est la valeur de l'intensité I transmise si la masse mesurée est de m= 1 600 Kg ?

II- Le signal fourni par le capteur-transmetteur est ensuite converti en binaire à l'aide d'un convertisseur analogique-numérique 16 bits. (4mA correspond à la valeur 0 et 20mA correspond à la valeur Nmax)

4) Donner la valeur Nmax en décimal puis en hexadécimal sur 16 bits.
5) Donner la valeur du quantum (en mA) du convertisseur.
6) Donner la valeur de la résolution (en g) du capteur-transmetteur.
7) Donner la valeur N en décimal, binaire puis hexadécimal correspondant à la masse m=1600 Kg ?

III- Le signal numérique codé sur 16 bits est constituée de deux octets au format NRZ.
Niveau haut = 1, niveau bas = 0, LSB en premier.

B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 B11 B12 B13 B14 B15

Lors d'une mesure, le capteur fournit le signal ci-contre.

8) Donner la valeur en décimal du nombre correspond au signal capteur.
9) En déduire la valeur de la masse mesurée.
10) Quelle est alors la valeur de l'intensité I fournit par le capteur analogique ?