Codage de l'information numérique

Le système décimal plus proche de nos habitudes (l'homme possède 10 doigts!...), utilise dix symboles (0...9), mais n'est pas directement utilisable en technologie électrique, puisqu'il faudrait par exemple dix niveaux de tension pour coder directement ces dix symboles.
D'autres systèmes de numération et codage de l'information sont utilisés dans les techniques numériques.
L'information binaire élémentaire, nommée bit (binary digit), ou eb (élément binaire) correspond à deux états du signal électrique codés 0 et 1.
C'est la juxtaposition de ces éléments binaires selon différentes conventions et codes, sous forme de nombres ou mots, qui permet d'exprimer des messages numérisés qui seront traités, analysés, transmis pour finalement fournir un résultat utile (affichage de données, numérisation du son, de l'image, transmission codée, calculs...).

I. CODES PONDERES

I.1 Construction

Dans cette famille usuelle de codes, c'est le rang ou la position de chaque chiffre composant un nombre, qui lui donne un poids, une valeur selon la base utilisée.
Le chiffre de droite est de poids le plus faible ***(LSB, less signifiant bit en binaire)***, le chiffre de gauche est celui de poids fort ***(MSB, most signifiant bit en binaire)***.

Exemple en base 10 **459** $\mathbf{459 = 4.10^2 + 5.10^1 + 9.10^0}$

centaines dizaines unités <==== Poids attribués à la position

La forme générale d'un nombre de n chiffres en base b est alors la suivante:

$(\mathbf{a_{n-1} \ldots a_1 .a_0})_b$ composé de n chiffres du domaine [0,b-1]

Les symboles utilisés sont :

En décimal	(base 10)	0 1 2 3 4 5 6 7 8 9
En binaire	(base 2)	0 1
En hexadécimal	**(base 16)**	**0 1 2 3 4 5 6 7 8 9 A B C D E F**

Les 16 symboles différents de l'hexadécimal offrent 2^4 possibilités soit l'équivalent de 4 bits.

Le code hexadécimal permet donc de simplifier l'expression de nombres binaires par regroupement de blocs de 4 bits.
Il permet un compromis entre le code binaire des machines et une base de numération pratique à utiliser pour les ingénieurs.
L'hexadécimal correspond rend les conversions très simples et fournissant une écriture plus compacte.
L'hexadécimal a été utilisé la première fois en 1956 par les ingénieurs de l'ordinateur Bendix G-15.
Par exemple, l'adresse $FF01_H$ correspond en binaire à 1111 1111 0000 0001.

- La valeur décimale (base 10), d'un nombre de n chiffres significatifs dans une base b quelconque est fournie par l'expression:

$$N_{10} = \sum_{i=0}^{i=n-1} a_i.b^i$$

avec a_i valeur décimale de chaque chiffre et i le rang de ce chiffre à partir de la droite.

La <u>valeur décimale maximale</u> que l'on peut obtenir pour un nombre de n rangs, est $\mathbf{b^n - 1}$.

Exemple en base 2 : $(1\ 0\ 0\ 1\ 0\ 1\ 0\ 1)_2$ est un mot binaire de 8 bits ou octet

Son poids décimal est : $(1.\ 2^7) + (1.2^4) + (\ 1.2^2) + (1.\ 2^0) = 149$

Le poids maxi obtenu sur un octet est donné par le code $(1\ 1\ 1\ 1\ 1\ 1\ 1\ 1)_2$
Il lui correspond en décimal la valeur 2^8-1= 255

Exemple en base 16 ou code HEXA : $(4\ A\ 2\ 0)_{16}$ est un mot hexa de 16 bits
Son poids décimal est : $(4.\ 16^3) + (10.16^2) + (2.\ 16^1) = (4.\ 4096) + (10.\ 256) + (2.\ 16) = 18976_{10}$

I.2 Changements de base – Conversion décimal vers binaire

On divise successivement par 2 le nombre décimal. Les restes correspondent aux bits consécutifs dans l'ordre inverse. C'est terminé au premier dividende nul (que l'on de compte pas).

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

44 2
0 22 2
0 11 2 44_{10} => 101100_2
1 5 2
1 2 2
0 **1**
<---------------------------- Sens de lecture des restes

II. CODES NON PONDERES

II.1 Code BCD (Binaire Codé Décimal) ou DCB

Ce codage permet une lecture décimale directe à partir d'un nombre binaire, il est surtout utilisé pour l'affichage.
Le principe est de coder chaque chiffre décimal, de 0 à 9, par quatre bits, les dernières combinaisons de 1010 à 1111 n'étant pas utilisées.
Exemple : 587_{10} => 0101 1000 0111_{BCD}

Remarque : dans le cas d'un affichage sept segments, il faut prévoir un décodage BCD/ 7 segments.

III.2 Code Gray ou binaire réfléchi

Ce code supprime les aléas découlant du changement simultané de plusieurs variables pour deux combinaisons adjacentes du code binaire pur. On l'emploi par exemple dans les codeurs de position absolue.

Exemple en binaire pur, passage de 01 à 10. (fig 1) :

Les deux variables ne peuvent pas physiquement changer simultanément d'état, il existe une transition soit par l'état 00, soit par l'état 11, codes incorrects qui peuvent être pris en compte à tort par une électronique rapide.
Le code est lu par un système optique selon les conventions suivantes:

- Blanc ➢ 0 logique
- Noir ➢ 1 logique

fig. 1

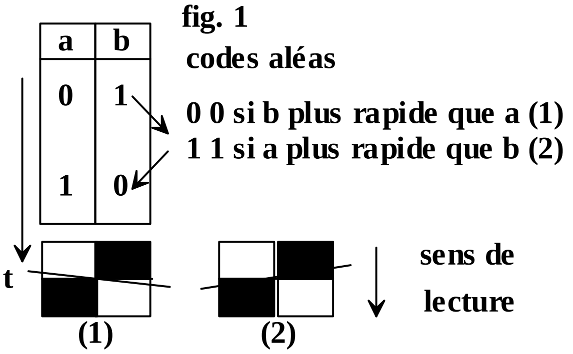
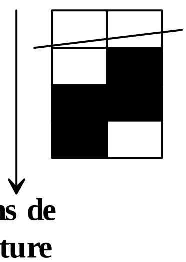


fig.2



Le code Gray ou binaire réfléchi ne doit donc pas entraîner le changement de plus d'une variable entre deux combinaisons successives, il est construit par symétrie d'où le nom de réfléchi (voir exemple plus loin).

III. LES DESIGNATIONS DU STOCKAGE DE L'INFORMATION

III.1 OCTET ou BYTE ; MOT SIMPLE ou MOT DOUBLE

L'**octet** (en anglais *byte* ou *B* avec une majuscule dans les notations) est une unité d'information composée de 8 bits. Il permet par exemple de stocker un caractère, tel qu'une lettre ou un chiffre.

Ce regroupement de nombres par série de 8 permet une lisibilité plus grande, au même titre que l'on apprécie, en base décimale, de regrouper les nombres par trois pour pouvoir distinguer les milliers. Le nombre « 1 256 245 » est par exemple plus lisible que « 1256245 ».
Un **mot (en anglais *word*),** désigne une unité d'information composée de 16 bits.

Un mot double (en anglais *double word*, où *dword*), désigne une unité d'information de 32 bits de longueur.

Pour un octet, le plus petit nombre est 0 (représenté par huit zéros 00000000), et le plus grand est 255 (représenté par huit chiffres « un » 11111111), ce qui représente 256 possibilités de valeurs différentes.

2^7 =128	2^6 =64	2^5 =32	2^4 =16	2^3 =8	2^2 =4	2^1 =2	2^0 =1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1

III.2 KiloOCTET, MégaOCTET

Longtemps l'informatique s'est singularisée par l'utilisation de différentes valeurs pour les unités du système international. Ainsi beaucoup d'informaticiens ont appris que 1 kilooctet valait 1024 octets. Or, depuis décembre 1998, l'organisme international *IEC* a statué sur la question (http://physics.nist.gov/cuu/Units/binary.html).

Voici donc les unités standardisées :

- Un kilooctet (ko ou kB) = 1000 octets
- Un Mégaoctet (Mo ou MB) = 1000 ko = 1 000 000 octets
- Un Gigaoctet (Go ou GB) = 1000 Mo = 1 000 000 000 octets
- Un Téraoctet (To) = 1000 Go = 1 000 000 000 000 octets

Attention ! De nombreux logiciels (parfois même certains systèmes d'exploitation) utilisent toujours la notation antérieure à 1998 pour laquelle :

- Un kilooctet (ko) = 2^{10} octets = 1024 octets
- Un Mégaoctet (Mo) = 2^{20} octets = 1024 ko = 1 048 576 octets
- Un Gigaoctet (Go) = 2^{30} octets = 1024 Mo = 1 073 741 824 octets
- Un Téraoctet (To) = 2^{40} octets = 1024 Go = 1 099 511 627 776 octets

Montrer que Windows 32 bits ne peut adresser que 4Go de mémoire vive, et qu'il faut Windows 64 bits pour aller au delà...

Exercices :

1) Etablir le tableau de correspondance de 0 à 20, entre les différents codes, selon la présentation suivante:

Décimal	Binaire naturel	Hexadécimal	BCD	Binaire réfléchi ou code Gray
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

2) Donner l'expression développée et calculer la valeur en base 10 des nombres suivants:

$10111_2 =$

$C3A_{16} =$

3) Coder 429_{10} en BCD

Transcoder $1000\ 0101_{BCD}$ en décimal.

Coder 67_{10} en binaire

Coder 63_{10} en hexadécimal.

4) Un codeur relatif génère 12 impulsions par mm, donner le code hexa obtenu après un déplacement de 18mm.

5) Déterminer le nombre de pistes minimales nécessaire à un codeur rotatif absolu pour coder une position angulaire avec une précision de 45' (45 secondes) d'arc. Il y a 3600' dans un ° d'angle...