Université Sultan Moulay Slimane
Ecole Supérieure de Technologie
Département Mécatronique

Electronique Numérique

Chapitre 1 : Systèmes de Numération et Codes

Pr. ARSALANE

Plan du chapitre

I. Système de Numération

II. Codes

III. Quelques définitions

I. Système de Numération

1. **Système Décimal** : C'est le système de base 10 que nous utilisons toujours, il comprend 10 symboles différents : {0, 1, 2, 3, … , 9}

Exemple :

$N = (2356)_{10} = 2.10^3 + 3.10^2 + 5.10^1 + 6.10^0$

$= 2000 + 300 + 50 + 6$

I. Système de Numération

2. **Système Binaire** : Ce système dit de base 2 comprend 2 symboles différents {0 , 1}. Chacun est appelé BIT (Binary digIT)

Exemple :

$N = (10110)_2 = 1.2^4 + 0.2^3 + 1.2^2 + 1.2^1 + 0.2^0$

$= 16 + 4 + 2$

$= (22)_{10}$

I. Système de Numération

3. **Système Hexadécimal** : Ce système de base 16 comprend 16 symboles différents {0 , 1, 2, 3, … , 9, A, B, C, D, E, F}.

Exemple :

$N = (AC53)_{16} = A.16^3 + C.16^2 + 5.16^1 + 3.16^0$

$= 10.16^3 + 12.16^2 + 5.16^1 + 3.16^0$

$= 40960 + 3072 + 80 + 3$

$= (44115)_{10}$

I. Système de Numération

Comptage en Hexadécimal :

Exemple 1 : 38 à 42

38 – 39 – 3A – 3B – 3C – 3D – 3E – 3F – 40 – 41 – 42

Exemple 2 : 6F8 à 700

6F8 – 6F9 – 6FA – 6FB – 6FC – 6FD – 6FE – 6FF – 700

I. Système de Numération

4. **Système Octal** : Ce système dit de base 8 comprend 8 symboles différents {0 , 1, 2, 3, 4, 5, 6, 7}.

Exemple :

$N = (372)_8 = 3.8^2 + 7.8^1 + 2.8^0$

$= 192 + 56 + 2$

$= (250)_{10}$

I. Système de Numération

Comptage en Octal:

Exemple 1 : 65 à 71

65 – 66 – 67 – 70 – 71

Exemple 2 : 275 – 300

275 – 276 – 277 – 300

5. Correspondance entre les bases :

Décimal	Binaire	Hexadécimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

I. Système de Numération

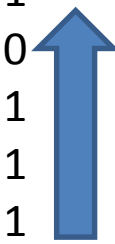
6. Changement de base :

a. Conversion d'un nombre décimal en un nombre d'une autre base :

Méthode : diviser le nombre décimal à convertir par la base B et conserver le reste de la division. Le quotient obtenu est divisé par B et le reste est conservé. Il faut répéter l'opération sur chaque quotient obtenu. Les restent successives sont écrits en commençons par le dernier de la gauche vers la droite pour former l'expression du nombre décimal dans le système de base B.

Cette méthode est dite méthode de la division successive.

3786	2	0
1893	2	1
946	2	0
473	2	1
236	2	0
118	2	0
59	2	1
29	2	1
14	2	0
7	2	1
3	2	1
1		1



Exemple : $(3786)_{10} = (111011001010)_2$

I. Système de Numération

Cas des nombres inférieur à 1 :

Méthode à suivre :

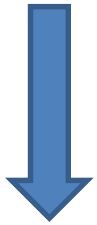
1. multiplier le nombre décimal par 2
2. Garder la partie entière du résultat
3. Soustraire cette partie entière du résultat
4. Recommencer l'opération en 1 tant que le résultat est différent de 0
5. Enfin de travail composer le résultat

Exemple 1 : Convertir en binaire le nombre $(0,7)_{10}$
Exemple 2 : Convertir en décimal le nombre $(0,10110)_2$

I. Système de Numération

Exemple 1 : Convertir en binaire le nombre $(0,7)_{10}$ en binaire :

0,7 x 2 = 1,4 – 1 = 0,4
0,4 x 2 = 0,8 – 0 = 0,8
0,8 x 2 = 1,6 – 1 = 0,6
0,6 x 2 = 1,2 – 1 = 0,2
0,2 x 2 = 0,4 – 0 = 0,4



Donc nombre $(0,7)_{10} = (0,10110)_2$

I. Système de Numération

Exemple 2 : Convertir en décimal le nombre $(0,10110)_2$

$(0,10110)_2 = 1.2^{-1} + 0.2^{-2} + 1.\ 2^{-3} + 1.\ 2^{-4} + 0.\ 2^{-5}$

$= (0,6875)_{10}$

I. Système de Numération

b. Conversion d'un nombre Hexadécimal en binaire

Méthode : chaque symbole écrit dans un nombre Hexadécimal est remplacé par son équivalent écrit en binaire

Exemple :

$N = (ECA)_{16} = (1110\ 1100\ 1010)_2$

I. Système de Numération

c. Conversion d'un nombre binaire en Hexadécimal

Méthode : c'est l'inverse de la précédente, il faut donc regrouper les '1' et les '0' du nombre par quatre en commençant par la droite, puis chaque groupe est remplacé par le symbole Hexadécimal correspondant.

Exemple :

$$N = (1100001101111)_2 = (186F)_{16}$$

I. Système de Numération

d. Conversion d'un nombre octal en binaire

Méthode : chaque symbole du nombre écrit dans le système octal est remplacé par son équivalent écrit dans le système binaire.

Exemple :

$N = (472)_8 = (100\ 111\ 010)_2$

I. Système de Numération

e. Conversion d'un nombre binaire en octal

Méthode : c'est l'inverse de la précédente, il faut donc regrouper les '1' et les '0' par trois en commençant par la droite, puis chaque groupe est remplacé par le symbole Octal correspondant.

Exemple :

$$N = (\overbrace{110}\overbrace{011}\overbrace{111})_2 = (637)_8$$

II. Les Codes

- Un code constitue une correspondance entre les symboles et des objets à désigner.

- Les codes étudiés précédemment sont pondérés : Dans une base de travail donnée, la valeur d'un rang donné est un multiple de la base de celle du rang inférieur.

- D'autre codes ne sont pas pondérés, c'est-à-dire que la position d'écriture ne correspond pas à un poids. Il ne permettent pas d'effectuer les opérations arithmétiques,

II. Les Codes

1. Codes pondérés :

a. Code naturel :

Le code Binaire naturel et ces dérivés (Octal et Hexadécimal) répondent aux règles classiques de l'arithmétique des nombres positives.

b. Code décimal codé binaire (BCD : Binary Coded Decimal)

Dans ce codage, chaque digit décimal est écrit en binaire. Cette représentation est commode pour traiter les nombres dans le mode de représentation le plus adapté à l'opérateur humain lors de l'affichage.

Exemple : $(7239)_{10} = (0111\ 0010\ 0011\ 1001)_{BCD}$

II. Les Codes

c. Le binaire signé:

Dans ce type de représentation, sur un format de 8 bits, il reste 7 bits significatives, le 8ème bit indique le signe du nombre codé,

Exemple : bit de signe (0 : positive, 1 : négative)

+ 52 : 00110100

– 52 : 10110100

- On utilise le bit de signe pour indiquer si le nombre binaire est positif ou négatif.
- Dans le cas des nombres positifs, les bits restant représente toujours la grandeur exacte du nombre binaire.
- Dans le cas des nombres négatifs, il y a trois façons de représenter la norme d'un nombre :
  - La notation en grandeur exact (GE)
  - La notation en complément à 1 (ou Complément Restreint : CR)
  - La notation en complément à 2 (ou Complément Vrai : CV)

II. Les Codes

La notation en grandeur exact (GE) :

- Le nombre de l'exemple du slide précédent, contient 1 bit de signe et 7 bits indiquant la norme ou la grandeur. On dit qu'on écrit les nombres binaires signés selon la notation en grandeur exact.

- Le système de notation exact est très facile à comprendre mais n'est pas aussi utile pour écrire des nombres signés que les deux autres systèmes (CR et CV).

II. Les Codes

La notation en complément à 1 (CR) :

La notation en complément à 1 d'un nombre binaire quelconque s'obtient en changeant chaque '0' du nombre par '1' et chaque '1' par un '0'. Autrement dit, on complémente chacun des bits du nombre.

Exemple : CR(101101) = 010010

Pour écrire des nombres négatifs en complément à 1, on attribue au bit de signe la valeur '1' et on transforme la grandeur exact binaire en sa notation en complément à '1'.

Exemple :	– 57	10111001	GE
	– 57	11000110	CR

Remarque : le bit de signe n'est pas complémenté, il reste à 1 afin d'indiquer un nombre négatif.

II. Les Codes

La notation en complément à 2 (CV) :

La notation en complément à deux d'un nombre binaire quelconque s'obtient simplement en prenant la complément à 1 du même nombre et en additionnant 1 au bit de poids le plus faible (LSB).

– 57 10111001

MSB : le bit de poids le plus fort (Most Significant Bit)

LSB : le bit de poids le plus faible (Least Significant Bit)

CV(10111001) 11000110

\+ 1

= 11000111

II. Les Codes

Résumé:

	Notation GE	Notation CR	Notation CV
+ 57	00111001	00111001	00111001
– 57	10111001	11000110	11000111

- Dans les modes d'écriture GE, CR et CV, les nombres positifs sont toujours écrits suivant la notation en grandeur exact, et le bit de signe est toujours '0'. Ce qui distingue ces trois modes sont les nombres négatifs.
- La conversion d'un nombre écrit en CR ou CV en sa valeur binaire exact ne pose pas vraiment de problème :
  - Pour passer du CR à la valeur binaire exact, il suffit de complémenter chaque bit à nouveau. CR [CR(x)] = x.
  - De même, pour passer du CV à la valeur binaire exact, il suffit de complémenter chaque bit et ajouter '1' au LSB. CV [CV(x)] = x.

II. Les Codes

Dynamique sur N bits :

- Mode non-signé :
  - Valeur minimale en Décimal : 0
  - Valeur maximale en Décimal : $2^N - 1$
- Mode signé :
  - Valeur minimale en Décimal : -2^{N-1}
  - Valeur maximale en Décimal : $+2^{N-1} - 1$

II. Les Codes

Exemple : Dynamique sur 8 bits :

- Mode non-signés :
  - Valeur minimale en Décimal : 0
  - Valeur maximale en Décimal : 255
- Mode signés :
  - Valeur minimale en Décimal : -128
  - Valeur maximale en Décimal : $+127$

II. Les Codes

Codage en virgule flottante :

- Représentation : exposant et mantisse

$$x = \text{signe}.mantisse.2^{(exposant - décalage)}$$

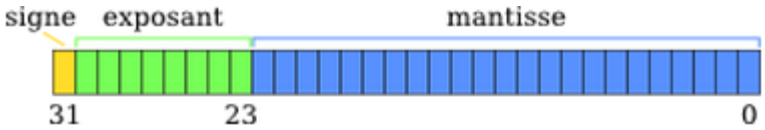
- Norme IEEE 754 :

	Encodage	Signe	Exposant	Mantisse
Simple précision	32 bits	1bit	8 bits	23 bits
Double précision	64 bits	1 bit	11 bits	52 bits

II. Les Codes

- **Codage en virgule flottante : 32 bits**
  - 1 bit de signe,
  - 8 bits d'exposant (-126 à 127)
  - 23 bits de mantisse, avec bit 1 implicite

$$x = (-1)^S . 2^{(e-127)} * 1, m$$



Exemple :

- **0,15625 => 1,25x2^{-3} => e=127-3=124 et m=0,25 => 0 01111100 01000000000000000000000**
- **-5.75 => -1,4375x2^{2} => e=127+2=129 et m=0,4375 => 1 10000001 01110000000000000000000**

II. Les Codes

1. Codes non-pondérés :

a. Code cyclique (code binaire réfléchi ou Code de Gray) :

Dans ce code, un seul bit change d'état entre deux valeurs adjacentes, il est employé dès que l'on doit représenter une évolution réel des variables où une seule change à chaque instant (exemple dans la table de Karnaugh).

C'est le système de codage qui, contrairement au code binaire pure, est arrangé de manière à nous faire changer l'état d'une seule variable à la fois.

II. Les Codes

1. Codes non-pondérés :

a. Code cyclique (code binaire réfléchi ou Code de Gray) :

Principe:

1. On choisit un code de départ : zéro est codé 0 et un est codé 1

2. A chaque fois qu'on a besoin d'un bit supplémentaire, on symétrise la liste des codes déjà obtenus (comme une réflexion dans un miroir)

3. On rajoute un 0 puis un 1 au début (à gauche) de chacun des codes. On a ainsi doublé le nombre de codes formés.

II. Les Codes

1. Codes non-pondérés :

a. Code cyclique (code binaire réfléchi ou Code de Gray) :

Exemple:

0	0			0	.0	0	00	0	.00	0	000
1	1			1	.1	1	01	1	.01	1	001
		miroir→						2	.11	2	011
				2	.1	2	11	3	.10	3	010
				3	.0	3	10	---	---		
								4	.10	4	110
								5	.11	5	111
								6	.01	6	101
								7	.00	7	100

Décimal	Code Binaire pure	Code de Gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

II. Les Codes

b. Code redondants : détecteur et correcteur d'erreurs

Ces codes sont utilisés pour Controller les transmissions, ils permettent de détecter une erreur de bit lors d'une transmission et parfois même une correction de l'erreur.

- Code de parité paire : Nombre des '1' est paire
- Code de parité impaire : Nombre des '1' est impaire

Exemple :

10111011 : parité pair

00111011 : parité impair

II. Les Codes

Description du code

• Lors de la transmission de caractères de texte, si on utilise le code ASCII, chaque caractère occupe 8 bits. Par exemple, le mot « HELLO » est représenté par 10010000 10001011 10011001 10011001 10011111.

• Chaque caractère est codé sur 7 bits plus 1 bit de parité (bit de contrôle) en général placé avant les bits d'information

• Le bit de parité est calculé de telle sorte que le nombre total de 1 soit toujours pair (par exemple).

• Mot d'information : 100 1101 Mot de code : 0 100 1101

• Mot d'information : 110 0111 Mot de code : 1 110 0111 Paramètres du code

• Longueur des mots d'information : m = 7

• Longueur des messages émis : n = 8

II. Les Codes

c. Le code ASCII (American Standard Code for Information Interchange)

- Dans les premières années des télégraphes, il fut définit des codes internationaux pour communiquer des messages écrits. Le code ASCII est maintenant généralisé pour tout les traitements informatiques.
- Il permet de coder tout caractère alphanumérique en binaire.
- Par exemple, le code ASCII est utilisé pour coder les touches du clavier d'un micro-ordinateur.
- L'ensemble des codes ASCII correspond à chaque caractère alphanumérique est contenu dans le tableau suivant :

II. Les Codes

c. Le code ASCII (American Standard Code for Information Interchange)

Le code ASCII a été mis au point pour la langue anglaise, il ne contient donc pas de caractères accentués, ni de caractères spécifiques à une langue. Pour coder ce type de caractère il faut recourir à un autre code.

Le code ASCII a donc été étendu à 8 bits (un octet) pour pouvoir coder plus de caractères (on parle d'ailleurs de code ASCII étendu...).

Ce code attribue les valeurs 0 à 255 (donc codées sur 8 bits, soit 1 octet) aux lettres majuscules et minuscules, aux chiffres, aux marques de ponctuation et aux autres symboles (caractères accentués).

II. Les Codes

F f	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	**BS**	HT	**LF**	VT	NP	**CR**	SO	SI
1.	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	**ESC**	FS	GS	RS	US
2.		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3.	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4.	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5.	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6.	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7.	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	Del

II. Les Codes

En langage de programmation C, nous utilisons les types de variables suivant :

Nom du type	Signification	Codage
char	Entier signé	8 bits
unsigned int	Entier non signé	32 bits
int	Entier signé	32 bits
float	Réel signé	32 bits • 24 bits de mantisse • 8 bits d'exposant
double	Réel signé	64 bits • 53 bits de mantisse • 11 bits d'exposant

III. Quelques définitions

- BIT (Binary digIT) : le bit est une unité élémentaire d'information ne pouvant prendre de deux valeurs distinctes : '0' ou '1'.
- Mot binaire : en informatique, l'unité de traitement de l'information est le mot binaire.
    - Un ensemble de 4 bits (mot de 4 bits) est appelé Quartet
    - Un ensemble de 8 bits (mots de 8 bits) est appelé Octet.
- Octet : 10111001

  MSB LSB
- Kilooctet : 1Ko = 1024 Octets
- Mégaoctet : 1Mo = 1024 Ko
- Gigaoctet : 1Go = 1024 Mo