Faculté Alger 1 Département Mathématiques et Informatique	Spécialité : MI Module : Codage et représentation de l'information	Mme YDROUDJ

Résumé des premiers chapitres

1 Les bases de numérotation:

Système	Base	Symboles	Nombre de symbole
Décimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	10
Binaire	2	0, 1	2
Octal	8	0, 1, 2, 3, 4, 5, 6, 7	8
Hexadécimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	16

2 Conversion entre les systèmes de numérotation :

Base source	Base destination	Procédure	Observations
Décimale	Binaire	Divisions successives sur 2	On s'arrête les divisions quand le quotient de la division= 0
Décimale	Octale	Divisions successives sur 8	
Décimale	Hexadécimale	Divisions successives sur 16	
Décimale → base B : Divisions successives sur la base B			
Binaire	Décimale	Multiplications x2	On s'arrête les divisions quand le quotient de la division= 0
Octale	Décimale	Multiplications x8	
Hexadécimale	Décimale	Multiplications x16	
Base B → décimale : Multiplications successive x B			
Octale	Binaire	1. Passage par le décimal 2. Table de vérité (éclatement de chaque chiffre octal sur 3 bits) 3. Division par chiffre et non pas le nombre complet.	
Hexadécimale	Binaire	Mêmes méthodes que la précédente, seulement ici, on éclate chaque chiffre sur 4 bits.	
Binaire	Octale	1. Table de vérité, on regroupe par 3 bits 2. Passage par le décimal	
Binaire	Hexadécimal	Mêmes méthodes seulement ici, on regroupe par 4 bits.	
Octale	Hexadécimale	1. Passage par le décimal	

		2. Passage par le binaire	
Hexadécimale	Octale	Mêmes méthodes	
Codification et représentation α- Numérique			
Décimale	BCD	1. Table de vérité sur 4 bits 2. Divisions successives par CHIFFRE sur 4 bits	Les chiffres s'arrêtent à 9 dans la table de vérité, on n'a pas (A,B,C..)
BCD	Décimale	1. Regroupement par 4 bits et on utilise la table de vérité	
Dans l'addition BCD, ajouter +6 (0110) sous chaque digit supérieur à 9 (1001) pour corriger le résultat de l'addition.			
Binaire	Gray	- le premier bit binaire est le même que le premier bit Gray, - addition horizontale des bits binaires (deux à deux) pour trouver les bits en Gray équivalents.	
Gray	Binaire	- le premier bit gray est le même que le premier bit binaire, - addition du 2eme bit Gray avec le 1er bit binaire pour trouver le 2eme bit binaire, - 3eme bit Gray avec le 2eme bit binaire pour trouver le 3eme bit binaire, -etc.	

3 Représentation des données :

3.1 Entiers non signés :

- Se sont des nombres positifs, l'intervalle est $[0, 2^n-1]$ tel que n est le nombre de bits.

- Le nombre de valeurs possibles est : 2^n.

- Le nombre minimum requis de bits pour représenter en binaire un nombre N est : $\ln(N) / \ln(2)$.

3.2 **Entiers signés :** 3 méthodes pour les représenter

	SVA : Signe et valeur absolue	C1 : Complément à 1	C2 : Complément à 2	Observations
Intervalle	$[-(2^{n-1}-1), +(2^{n-1}-1)]$	$[-(2^{n-1}-1), +(2^{n-1}-1)]$	$[-(2^{n-1}), +(2^{n-1}-1)]$	n : nombre de bits.
Nombre de valeurs	2^n	2^n	2^n	
Représentation	- premier bit du sine : 0 si positif, 1 sinon	- si c'est positif : reste le même, - si négatif : garder le bit du signe et inverser les autres (0 devient 1 et 1 devient0)	-si c'est positif : reste le même, - si négatif : c'est le C1 +1	
Opérations arithmétiques	Non évidentes	La retenue est additionnée au résultat final	La retenue est négligée	
Décodage (représenter en décimal)	-le bit du signe si 0, devient (+), sinon devient (–). -les autres bits du nombre sont convertis en son équivalent en décimal	- si positif, sa forme en C1=sa forme naturelle -si négatif : remplacer le 1 (bit de signe) par le signe (-) puis inverser le reste des bits	- si positif, sa forme en C1=sa forme naturelle -si négatif : remplacer le 1 (bit de signe) par le signe (-) puis soustraire un 1 (devient en C1) puis inverser les bits. Ou bien : C2[C2(N)]=SVA(N)	

4 Représentation des nombres réels :

4.1 Format virgule fixe :

La position de la virgule est fixe, une partie entière et une partie fractionnaire séparées par une virgule.

4.2 Format virgule flottante :

Normalisé par l'organisme international IEEE :

4.2.1 Norme IEEE 754 :

Le nombre est écrit d'abord sous forme : $\pm 1{,}M \cdot 2^{Er}$

- **Simple précision : (32 bits)**

Signe	Exposant biaisé	Mantisse
1bit	**8 bits**	**23 bits**

- 1 bit de signe de la mantisse
- 8 bits pour l'exposant
- 23 bits pour la mantisse

- **Double précision : (64 bits)**

Signe	Exposant biaisé	Mantisse
1bit	**11 bits**	**52 bits**

- 1 bit de signe de la mantisse
- 11 bits pour l'exposant
- 52 bits pour la mantisse

Processus d'aller d'un nombre décimal vers un nombre sous format IEEE 754 :

1. Convertir le nombre décimal vers un nombre binaire
2. Ecrire le nouveau nombre en binaire sous forme : $\pm$ 1,M . 2^{Er} tel que Er est l'exposant réel
3. Calculer l'exposant biaisé tel que : Eb = Er + 2^{p-1} -1. Convertir le résultat en binaire
4. Composer le nouveau nombre : **bit signe, exposant biaisé, mantisse**

Processus d'aller d'un nombre sous format IEEE 754 vers un nombre décimal :
- Si le nombre est donné en hexadécimal, on le convertit d'abord en binaire,
- Si le nombre est en binaire, on procède comme suit :
1. On peut identifier le bit de signe, le bit du poids le plus fort :
   - si 1 : le nombre est négatif, sinon il est positif,
2. On peut identifier la mantisse,
3. On convertit l'exposant biaisé en décimal,
4. On calcule l'exposant réel : Er = Eb – (2^{p-1} -1),
5. On forme le nombre : : $\pm$ 1,M . 2^{Er} puis on enlève 2^{Er} par le décalage de la virgule par Er positions.

4.3 Addition en virgule flottante :

Elle se fait en trois étapes :

- Dénormaliser les deux nombres afin d'avoir le même exposant (le plus élevé),
- Additionner les mantisses,
- Normaliser le résultat,

4.4 Multiplication en virgule flottante :

Elle se fait en quatre étapes :

- Dénormaliser les deux nombres (exposants naturels),
- Additionner les exposants naturels,
- Multiplier les mantisses,
- Normaliser le résultat,