الأقسام التحضيرية للمدارس العليا

المملكة المغربية
ⵜⴰⴳⵍⴷⵉⵜ ⵏ ⵍⵎⵖⵔⵉⴱ

وزارة التربية الوطنية
والتعليم الأولي والرياضة
ⵜⴰⵎⴰⵡⴰⵙⵜ ⵏ ⵓⵙⴳⵎⵉ ⴰⵏⴰⵎⵓⵔ
ⴷ ⵓⵙⵍⵎⴷ ⴰⵎⵣⵡⴰⵔⵓ ⴷ ⵜⵓⵏⵏⵓⵏⵜ

Codage de l'information

Youssef Daghouj

24 Septembre 2025

Introduction

Le mot informatique vient de la contraction de deux mots : information et automatique ; ce qui veut dire le traitement automatique de l'information. Cette information est codée en utilisant les caractéristiques des électrons.

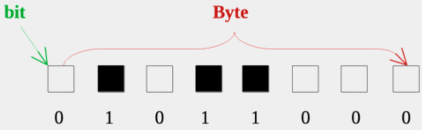
Imaginons 8 petits transistors l'un à côté de l'autre. Chaque transistor peut être, à un moment donné, soit chargé, soit déchargé.

Pour faciliter les choses, on représente le chargé par le caractère 1 et l'autre par le caractère 0.

On appelle l'état du transistor *bit*, alors un bit est égale soit à 0, soit à 1, et

$$8bits = 1Byte = 1Octet$$

.



L'information peut prendre plusieurs formes : Écriture sous forme des caractères et des nombres, son, image, vidéo... En informatique tout est codé sous forme des 0s et des 1s.

Prenons les caractères comme exemple. À chaque caractère est associé un code standard comme *ASCII*.

Caractère	Code binaire
A	1000001
B	1000010
X	1011000
a	1100001
c	1100011
.	101110
,	1010100
+	101011
-	101101
@	1000000
...	...

Table – Extrait de la table *ASCII*

On s'intéresse au codage des nombres, entiers et réels, dans un ordinateur.

Codage des entiers naturels

Théorème

Soit X un entier, il existe une unique suite $(x_n)_{n\in\mathbb{N}}$ d'entiers de $[0, b-1]$ avec $b > 1$, tous nul a partir d'un certain rang, tels que :

$$X = \sum_{n=0}^{+\infty} x_n b^n$$

Définition

On dit que l' écriture $\sum_{n=0}^{+\infty} x_n b^n$ est la représentation en base b de l'entier X.

Si r est le plus grand entier tel que $x_r \neq 0$, on écrira cette représentation sous la forme

$$(x_r x_{r-1} ... x_1 x_0)_b$$

Base décimale ($b = 10$)

Exemple

$$(1991)_{10} = 1 \times 10^0 + 9 \times 10^1 + 9 \times 10^2 + 1 \times 10^3$$

Base binaire ($b = 2$)

Exemple

$$\begin{aligned}(11111000111)_2 &= 1 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + 0 \times 2^3 + 0 \times 2^4 + 0 \times 2^5 \\ &+ 1 \times 2^6 + 1 \times 2^7 + 1 \times 2^8 + 1 \times 2^9 + 1 \times 2^{10} \\ &= (1991)_{10}\end{aligned}$$

Base hexadécimale ($b = 16$)

Exemple

$$(7C7)_{16} = 7 \times 16^0 + 12 \times 16^1 + 7 \times 16^2 = (1991)_{10}$$

Changement de base

Décimale → Binaire

Pour obtenir la représentation d'un décimal en binaire, on effectue des divisions entières successives par 2 jusqu'à ce que le quotient devienne nul.

Le résultat est la concaténation des restes en commançant par le dernier.

Exemple

/ 2

Quotient	Reste
1991	1
995	1
497	1
248	0
124	0
62	0
31	1
15	1
7	1
3	1
1	1
0	

Alors $(1991)_{10} = (11111000111)_2$

Changement de base

Binaire → Décimale

Pour obtenir la valeur décimale d'un binaire on additionne, le produit de la valeur de chaque bit par 2 puissance son indice, en commençant par 0 comme indice du bit de poids faible (premier bit à droite).

Exemple

$$
\begin{aligned}
(11111000111)_2 &= 1 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + 0 \times 2^3 + 0 \times 2^4 + 0 \times 2^5 \\
&+ 1 \times 2^6 + 1 \times 2^7 + 1 \times 2^8 + 1 \times 2^9 + 1 \times 2^{10} \\
&= (1991)_{10}
\end{aligned}
$$

L'addition binaire

Méthode

L'addition en binaire s'effectue de droite à gauche entre les bits du même rang des deux nombres comme suit :

- $0 + 0 = 0$
- $1 + 0 = 0 + 1 = 1$
- $1 + 1 = 0$ et 1 comme retenue

Exemple

$$
\begin{array}{cccccccccc}
 & & & {\scriptstyle 1} & {\scriptstyle 1} & & & & & \\
 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 74 \\
+ & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 153 \\
\hline
 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 225
\end{array}
$$

Débordement (Overflow)

Définition

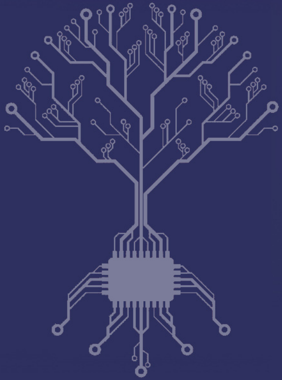
C' est un bit supplémentaire qui indique un dépassement de capacité de calcul. Le débordement intervient lorsque le résultat de l' opération n' est pas représentable dans le système utilisé, c' est à dire lorsque le nombre des bits significatifs du résultat est supérieur au nombre des bits utilisé.

Exemple

Sur 8 bits, le nombre des entiers possibles à coder est $2^8 = 256$, soit comme intervalle $[0, 255]$.

$$
\begin{array}{cccccccccccc}
 & {\scriptstyle 1} & & & {\scriptstyle 1} & {\scriptstyle 1} & & & & & & \\
 & & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & & 202 \\
+ & & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & & 153 \\
\hline
 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & & 355>255
\end{array}
$$

Codage des entiers relatifs



Bit de signe

Cette méthode consiste à réserver le bit du poids fort (le premier à gauche) au signe du nombre : 1 si il est négatif et 0 si non. Sur les autres bits on code la valeur absolue du nombre.

Exemples

Sur un octet :

$$\overbrace{0}^{+}\ \overbrace{1000110}^{70}$$

$$\overbrace{1}^{-}\ \overbrace{1000110}^{70}$$

Dans le cas d'un octet (8 bits) l'intervalle des entiers possibles à coder est $[-127, 127] = [-(2^{8-1} - 1), 2^{8-1} - 1]$

D'une manière générale sur x bits l'intervalle est $[-(2^{x-1} - 1), 2^{x-1} - 1]$

Complément à 2

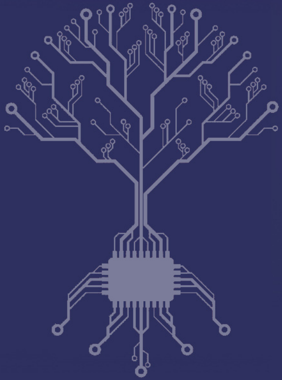
La différence ici est qu'un entier négatif est codé en ajoutant la valeur 1 au complément à 1 (l'inverse de chaque bit) du cas positif.

Exemple

Codons en complément à 2 et sur 8 bits le nombre $(-70)_{10}$:
On a $(+70)_{10} = (01000110)_{C2}$ dont le complément à 1 est 10111001, en l'ajoutant 1 on aura $(10111010)_{C2} = (-70)_{10}$

L'intervalle des entiers possibles à coder sur x bits est $[-2^{x-1}, 2^{x-1} - 1]$

Codage des réels



Théorème

Soit $X \in \mathbb{R}_+$. Il existe $n \in \mathbb{Z}$ et des entiers $x_i \in [0, b-1]$ avec $b > 1$, pour $i \in]-\infty, n]$ tels que $x_n \neq 0$ et

$$X = \sum_{i=-\infty}^{n} x_i b^i$$

Définition

Soit $X = \sum_{i=-\infty}^{n} x_i b^i$ un développement en base b de X, on note :

$$X = (x_n x_{n-1} ... x_0, x_{-1} x_{-2} ...)_b$$

De plus, s' il existe $m < 0$ tel que pour tout $r < m, x_r = 0$, on utilisera une représentation bornée, en omettant l' ultime série de 0 :

$$X = (x_n x_{n-1} .... x_0, x_{-1} x_{-2} ... x_m)_b$$

Virgule fixe

Cette méthode est utilisée par les premières machines et possède une partie entière et une partie décimale séparées par une virgule. La position de la virgule est fixe d' où le nom.
Soit $X \in \mathbb{R}_+$, la représentation en binaire de la partie entière de X est déjà vue ; il nous reste à représenter sa partie décimale $(0, x_{-1}x_{-2}....)_2$:

Méthode

Cette représentation est la concaténation des parties entières obtenues en multipliant successivement la partie décimale par 2 jusqu'à ce que cette dernière soit nulle ou le nombre de bits est atteint.

Exemple

$(0,375)_{10} = (?)_2$

- $0,375 \times 2 = 0,75$
- $0,75 \times 2 = 1,5$
- $0,5 \times 2 = 1,0$

D'où $(0,375)_{10} = (0,011)_2$ et $(70,375)_{10} = (1000110,011)_2$

Méthode réciproque

Inversement, la forme décimale de la partie fractionnaire est obtenue en additionnant le produit de la valeur de chaque bit par 2 puissance son indice.

Exemple

$$(0,10101)_2 = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 0 \times 2^{-4} + 1 \times 2^{-5} = (0,65625)_{10}$$

Notation scientifique

Définition

Soit $b \in \mathbb{N}$, $b \geq 2$. Soit $x \in \mathbb{R}$. Il existe un unique entier $e \in \mathbb{Z}$ et un unique réel $m \in [1, b[$ tels que

$$x = m.b^e$$

Exemples en base 10

- $503 = 5,03 \times 10^2$.
- $0,0016 = 1,6 \times 10^{-3}$.

Virgule flottante - Norme IEEE 754

Définition

C'est la norme utilisée actuellement sur les machines sous la forme scientifique binaire :

$$(-1)^s 1, m.2^e$$

Cela revient à coder le signe (s), la mantisse (m) et l'exposant (e).

Simple précision (sur 32 bits)

- 1 bit de signe (0 pour + et 1 pour -);
- 8 bits pour l'exposant : Biaisé pour être positif (Ajout de $2^{8-1} - 1 = 127$) et placé avant la mantisse pour simplifier la comparaison ;
- 23 bits pour la mantisse : Virgule placée après le bit à 1 de poids fort.

Double précision (sur 64 bits)

- 1 bit de signe ;
- 11 bits pour l'exposant : Biaisé (Ajout de $2^{11-1} - 1$);
- 52 bits pour la mantisse.

Un réel sous la forme de la norme IEEE 754

Exemple

$(35,5)_{10} = (?)_{IEEE754_{32bits}}$

- Nombre positif alors $s = 0$
- $(35,5)_{10} = (100011,1)_2$ (Virgule fixe)
- $(35,5)_{10} = (1,000111.2^5)_2$ (Virgule flottante)
  - $1,m = 1,000111$ (On complète 23 bits par des 0s à droite)
  - $e = 5 + 127 = 132 = (10000100)_2$

Finalement,

$$(35,5)_{10} = (\underbrace{\overbrace{\ 0\ }^{signe}\ \underbrace{10000100}_{exposant}\overbrace{00011100000000000000000}^{mantisse}}_{32bits})_{IEEE754}$$

De la norme IEEE 754 à la valeur réelle

Exemple

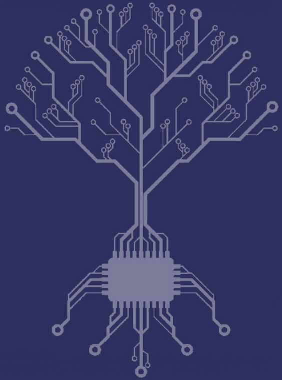
$$(11000000111100000000000000000000)_{IEEE754} = (?)_{10}$$

$$\overbrace{1}^{\textit{nombre négatif}} \qquad \underbrace{10000001}_{=129,\ \textit{alors } e=129-127=2} \qquad \overbrace{11100000000000000000000}^{m=111}$$

Finalement,

$$(-1)^s 1,m.2^e = (-1)^1 1,111.2^2 = (-7,5)_{10}$$

Système informatique



Définition

Un système informatique est un ensemble d'éléments en interaction entre eux afin de traiter automatiquement l'information.

Exemples

Ordinateur, téléphone, guichet bancaire, drone, ...

Chaque système informatique est composé de deux parties :

- Partie matérielle (*Hardware*)
- Partie logicielle (*Software*)

Partie matérielle

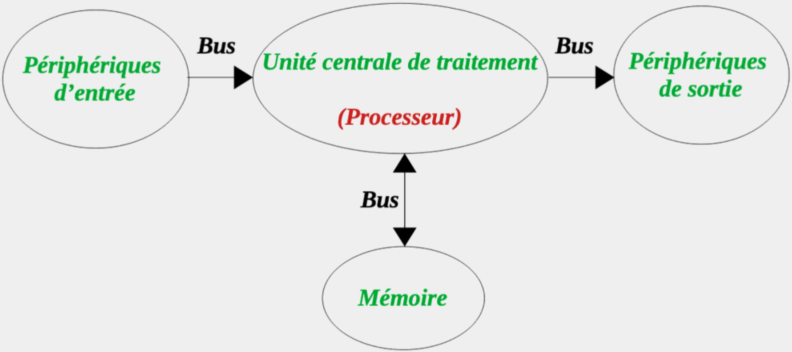


Figure – Structure générale d'un ordinateur

Processeur

Constitue le cerveau de l'ordinateur, il est chargé du contrôle, du calcul et de l'exécution des instructions. Sa vitesse est mesurée en *Hertz*(*Hz*) telle que :

$$1Hz = 1 \textit{ opération par seconde}$$

$$1KHz = 10^3 Hz$$

$$1MHz = 10^6 Hz$$

$$1GHz = 10^9 Hz$$

Mémoire

Sert à stocker les données sous forme binaire. Sa capacité est mesurée généralement en $Byte(B)$ telle que :

$$1KB = 2^{10}Bytes = 1024Bytes$$

$$1MB = 2^{10}KB = 2^{20}Bytes$$

$$1GB = 2^{10}MB = 2^{30}Bytes$$

$$1TB = 2^{10}GB = 2^{40}Bytes$$

Types de mémoire

- *RAM*(*Random Access Memory*)
  Mémoire électronique très rapide qui contient provisoirement les données de toute chose en cours d' exécution, mais elle se vide complétement après la coupure du courant électrique, alors on dit qu' elle est volatile.
- *ROM*(*Read Only Memory*)
  Mémoire non volatile qui contient un logiciel nécessaire pour le démarrage de l'ordinateur.
- *HDD*(*Hard Disk Drive*) et *SSD*(*Solid State Drive*)
  Deux mémoires non volatiles de grande capacité. La première est plus rapide car elle stocke les données d'une manière électronique, tandis que la deuxième les stocke d'une manière magnétique.

Périphériques

Les périphériques d' entrée permettent de captuer et d'entrer l'information à l'ordinateur, comme le clavier, la souris, ... Contrairement aux périphériques de sortie qui permettent de restituer l'information pour qu'elle soit compréhensible par l'être humain, comme l'écran, les écouteurs, ...

Carte mère

Sur cette carte tous les composants de l'ordinateur sont connectés pour l'échange de l'information à travers le *bus*.

Partie logicielle

Définition

Un logiciel est un ensemble de programmes, et un programme est une suite d'instructions exécutées dans un ordre pour réaliser une tâche.

On trouve deux types de logiciels :

- *Système d'exploitation*
  Un grand logiciel obligatoirement installé et en interaction direct avec le matériel afin de l'exploiter.

Exemples

- Pour les ordinateurs : *Unix*(*MacOS*, *Ubuntu*, ...), *Windows*, ...
- Pour les téléphones : *iOS*, *Android*, ...

- *Applications*
  Des logiciels installés selon le besoin de l'utilisateur.

Exemples

Navigateur web, *mail*, *lecteur audio/vidéo*, ...