La représentation des entiers naturels en base binaire constitue une des premières « briques » de l'informatique : elle permet à l'ordinateur de compter et de faire des opérations arithmétiques élémentaires telles l'addition et la multiplication.

Partie I : Combien de bits ?

Les nombres entiers naturels sont codés classiquement en binaire avec des mots de 8 bits, 16 bits, 32 bits ou 64 bits (1, 2, 4 ou 8 octets), suivant la machine sur laquelle on travaille et le type de calcul. La question se pose maintenant de savoir sur combien de bits il est préférable de coder un nombre.

I.1. En utilisant 8 bits, quelle est la valeur du plus grand entier naturel que l'on puisse coder ? Mêmes questions avec 16 bits, 32 bits et 64 bits.

I.2. En vous aidant de la question précédente, proposer une méthode permettant de déterminer si un entier *N* peut être codé sur 8, 16, 32 ou 64 bits.

I.3. Appliquer la méthode précédente pour déterminer si les entiers suivants peuvent être codés sur 8, 16, 32 ou 64 bits.

a. $7{,}9\times10^5$ **b.** $1{,}2\times10^2$ **c.** $4{,}0\times10^9$ **d.** $1{,}9\times10^{19}$

Intéressons-nous maintenant au nombre de bits nécessaires pour afficher le résultat d'une multiplication.

Pour commencer, regardons le cas des nombres en base décimale. L'étude du nombre de chiffre nécessaire à l'écriture du résultat d'une multiplication nécessite d'utiliser la représentation du nombre en notation scientifique.

Rappel : Notation scientifique d'un nombre N en base 10

- *On trouve l'encadrement du nombre pour déterminer l'exposant :*

  Si $10^X \leq N \leq 10^{X+1}$*, l'exposant est X*
- *Le nombre multiplicateur m, appelé mantisse, est tel que* $m\times10^X = N$*. Ainsi,* $m = \frac{N}{10^X}$

On a alors $N = m\times10^X$

I.4. **<u>Sans procéder à la multiplication</u>**, déterminer le nombre de chiffres nécessaires pour écrire le résultat des opérations suivantes.

a. 5×16 **b.** 500×35

La même méthode peut être appliquée aux entiers représentés en base binaire, en représentant les nombre en notation scientifique binaire.

Pour écrire le nombre décimal N en écriture scientifique binaire :

- On trouve l'encadrement du nombre pour déterminer l'exposant :

  Si $2^X \leq N < 2^{X+1}$, l'exposant est X
- Le nombre multiplicateur m, appelé mantisse, est tel que $m\times2^X = N$. Ainsi, $m = \frac{N}{2^X}$

On a alors $N = m\times2^X$

Exemple : 12345 en écriture scientifique binaire

$2^{13} \leq 12345 \leq 2^{14}$*, l'exposant sera 13.*

La mantisse est, $m = \frac{12345}{2^{13}} = 1{,}506958008$*.*

Donc $12345 = 1{,}506958008\times2^{13}$

I.5. Déterminer le nombre de bits nécessaires pour écrire le résultat des multiplications suivantes en <u>binaire</u>.

a. 5×16 **b.** 500×35

Partie II : Opérations binaires

Faire des opérations sur des nombres en représentation binaire suit la même logique que lorsque l'on travaille en représentation décimale. Les règles sont les mêmes, mais encore faut-il les connaitre.

A. Addition

II.1. Commençons par étudier l'addition en base 10. Faire l'addition décimale ci-dessous, en indiquant chaque étape (par quoi commencez-vous, que faite vous ensuite, etc…) ainsi que les retenues.

$$\begin{array}{r} 12\,671 \\ \underline{+\ \ 2\,454} \end{array}$$

Etapes :

II.2. En appliquant la même logique, quel est, en binaire, le résultat des additions suivantes.

a. $\begin{array}{r} \overline{1}^2 \\ \underline{+\ \overline{1}^2} \end{array}$ **b.** $\begin{array}{r} \overline{10}^2 \\ \underline{+\ \ \overline{1}^2} \end{array}$ **c.** $\begin{array}{r} \overline{10}^2 \\ \underline{+\ \overline{11}^2} \end{array}$

II.3. Poser les additions suivantes en binaire et les résoudre. Vérifier ensuite le résultat de chaque addition en convertissant les nombres et le résultat en base 10.

a. $\overline{10001}^2 + \overline{11100}^2$ **b.** $\overline{11}^2 + \overline{101}^2$ **c.** $\overline{11110}^2 + \overline{10110}^2$

d. $\overline{1100}^2 + \overline{1001}^2$ **e.** $\overline{1111}^2 + \overline{10000011}^2$ **f.** $\overline{101}^2 + \overline{10001}^2 + \overline{110}^2$

B. Soustraction

De manière similaire, la soustraction en binaire suis les mêmes règles que la soustraction en décimal, comme le montre les exemples suivants :

(On ne considère pour l'instant que les soustractions où le premier nombre est supérieur au second.)

Exemples en base 10

Soustraction simple

$$\begin{array}{r} 7\,5\,4 \\ \underline{-\ \ \ 2\,3} \\ 7\,3\,1 \end{array}$$

On soustrait symbole par symbole :

Unité : 4 - 3 = 1
Dizaine : 5 − 2 = 3
Centaine : 7 − 0 = 7

Soustraction avec retenue

$$\begin{array}{r} {\scriptstyle -1}\ {\scriptstyle +10}\ \ \\ 2\,3\,8 \\ \underline{-\ \ \ 7\,2} \\ 1\,6\,6 \end{array}$$

Lorsque le premier symbole est inférieur au second, on ajoute 10 au premier symbole, on fait la soustraction pour avoir le symbole résultat et on retient -1 au symbole suivant :

Unité : 8 − 2 = 6
Dizaine : 10 + 3 − 7 = 6
Centaine : − 1 +2 − 0 = 1

Exemples en base 2

Soustraction simple

$$\begin{array}{r} \overline{1\,1\,0\,1}^2 \\ \underline{-\ \ \ \overline{1\,0\,0}^2} \\ \overline{1\,0\,0\,1}^2 \end{array}$$

On soustrait symbole par symbole :

Bit #1 : 1 − 0 = 1
Bit #2 : 0 − 0 = 0
Bit #3 : 1 − 1 = 0
Bit #4 : 1 − 0 = 1

Soustraction avec retenue

$$\begin{array}{r} {\scriptstyle -1}\ {\scriptstyle +2}\ \ \\ \overline{1\,1\,0\,1}^2 \\ \underline{-\ \overline{1\,0\,1\,0}^2} \\ \overline{1\,1}^2 \end{array}$$

Lorsque le premier symbole est inférieur au second, on ajoute 2 ($\overline{10}^2$) au premier symbole, on fait la soustraction pour avoir le symbole résultat et on retient -1 au symbole suivant :

Bit #1 : 1 − 0 = 1
Bit #2 : 2 + 0 − 1 = 1
Bit #3 : −1 + 1 − 0 = 0
Bit #4 : 1 − 1 = 0

II.4. Poser les soustractions suivantes en binaire et les résoudre. Vérifier ensuite le résultat de chaque soustraction en convertissant les nombres et le résultat en base 10.

a. $\overline{10111}^2 - \overline{110}^2$ **b.** $\overline{101}^2 - \overline{11}^2$ **c.** $\overline{1001110}^2 - \overline{10110}^2$

C. Multiplication

Ici, par chance, il n'y a que des multiplications par 0 ou par 1, facile ! Mais il faut faire attention aux possibles décalages lorsqu'on passe d'un bit à l'autre. Conseil : utiliser la méthode « multiplier puis additionner » !

Exemple 1 :

$$
\begin{array}{rl}
\overline{11}^{2} & \\
\times \quad \overline{10}^{2} & \\
\hline
\overline{0}^{2} & \text{On multiplie par bit \#1} \\
+ \quad \overline{11\mathbf{0}}^{2} & \text{On décale d'un bit avec un 0 puis on multiplie par bit \#2} \\
\hline
\overline{110}^{2} & \text{On additionne}
\end{array}
$$

Exemple 2 :

$$
\begin{array}{rl}
\overline{1101}^{2} & \\
\times \quad \overline{101}^{2} & \\
\hline
\overline{1101}^{2} & \text{Multiplication par bit \#1} \\
+ \quad \overline{0\mathbf{0}}^{2} & \text{Décalage de 1 bit et multiplication par bit \#2} \\
+ \quad \overline{1101\mathbf{00}}^{2} & \text{Décalage de 2 bits et multiplication par bit \#3} \\
\hline
\overline{1000001}^{2} & \text{On additionne}
\end{array}
$$

II.5. Poser les multiplications suivantes en <u>binaire</u> et les résoudre. Vérifier ensuite le résultat de chaque multiplication en convertissant les nombres et le résultat en base 10.

a. $\overline{101}^{2} \times \overline{11}^{2}$ **b.** $\overline{10111}^{2} \times \overline{110}^{2}$ **c.** $\overline{10010}^{2} \times \overline{1110}^{2}$

II.6. Considérant un nombre binaire de la forme $N = \overline{n_p n_{p-1} \ldots n_2 n_1 n_0}^{2}$, quelle serait alors la représentation binaire de $2 \times N$? Et de $4 \times N$?

Pour aller un peu plus loin…

<u>Opérations hexadécimale</u>

Poser les opérations suivantes en hexadécimal et les résoudre.

a. $\overline{8}^{16} + \overline{9}^{16}$ **b.** $\overline{A}^{16} + \overline{2A}^{16}$ **c.** $\overline{375}^{16} + \overline{269B}^{16}$

d. $\overline{A2}^{16} - \overline{9}^{16}$ **e.** $\overline{269B}^{16} - \overline{3D5}^{16}$

Ce qu'il faut retenir

❖ **La notation scientifique en binaire**

Pour écrire le nombre décimal N en écriture scientifique binaire :

- On trouve l'encadrement du nombre pour déterminer l'exposant :
  Si $2^X \leq N \leq 2^{X+1}$, l'exposant est X
- Le nombre multiplicateur m, appelé mantisse, est tel que $m \times 2^X = N$. Ainsi, $m = \frac{N}{2^X}$

On a alors $N = m \times 2^X$

Savoir faire

❖ **Evaluer le nombre de bits nécessaires pour écrire un entier naturel**

❖ **Evaluer le nombre de bits nécessaires pour écrire le résultat de l'addition ou de la multiplication d'entiers naturels**

❖ **Addition, soustraction et multiplication binaire**