1ère NSI Codage des informations : Représentation des entiers relatifs

1. Entiers signés

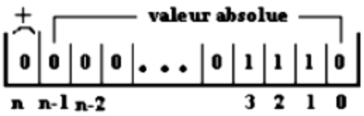
Comme son nom l'indique, un **nombre signé** est un nombre pourvu d'un signe **positif** ou **négatif**.

Pour représenter les entiers relatifs en notation binaire, on doit étendre la **représentation des positifs aux nombres négatifs**.

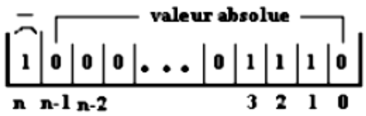
1ère solution avec les premières machines :

Une solution est de réserver le bit de **poids fort** (c'est le bit le plus à gauche) pour le signe de l'entier à représenter (**1 pour –** et **0 pour +**) et d'utiliser **les autres pour représenter sa valeur absolue**

Exemple du codage en binaire signé des nombres +**14** et -**14** :



+14 est représenté par 0000...01110



-14 est représenté par 1000...01110

Un exemple :

On représente l'entier 5 sur 8 bits par 00000101,
−5 serait donc représenté par 10000101

1) En utilisant la méthode décrite ci-dessus, représentez −15 (représentation sur 8 bit)

Ce codage n'est pas **utilisé en pratique**.

En effet :

- le **traitement spécifique du signe coûte cher** en circuits électroniques et en **temps de calcul**.
- 0 est représenté par deux **nombres distincts**.
  Un zéro positif (00000000) et un zéro négatif (10000000) !
- l'addition **ne fonctionne plus** !

C'est une version améliorée qui est utilisée dans la plupart des calculateurs : elle se nomme le complément à deux.

2. Complément à deux

C'est pour remédier à ces problèmes que l'on utilise la notation en **complément à deux**.

Avant de représenter un entier relatif, il est nécessaire de **définir le nombre de bits** qui seront **utilisés** pour cette représentation (souvent 4, 8, 16, 32 ou 64 bits)

Les nombres positifs sont **représentés comme attendu**, en revanche **les nombres négatifs sont obtenus de la manière suivante** :

- On **représente** le nombre sur le **nombre de bits de codage** demandés ;
- On inverse les bits de l'écriture binaire de sa valeur absolue (opération binaire NON), les bits à 1 passent à 0 et vice versa (c’est la phase dite de *complément à un*);
- On ajoute 1 au résultat (les dépassements sont ignorés).

Exemples :

- Déterminons la représentation de -12 sur 8 bits
    - Commençons par représenter 12 sur 8 bits (sachant que pour représenter 12 en binaire seuls 4 bits sont nécessaire, les 4 bits les plus à gauche seront à 0) : 00001100
    - Inversons tous les bits : 11110011
    - Ajoutons 1 au nombre obtenu à l'étape précédente : les retenues sont notées en rouge
    - La représentation de −12 sur 8 bits est donc : 11110100

$$\begin{array}{r} \scriptstyle 11 \\ 11110011 \\ +\ 00000001 \\ \hline 11110100 \end{array}$$

Comment peut-on être sûr que 11110100 est bien la représentation de −12 ?

Nous pouvons affirmer sans trop de risque de nous tromper que
12 + (-12) = 0.

Vérifions que cela est vrai pour notre représentation sur 8 bits.

$$\begin{array}{r} \scriptstyle 11111 \\ 00001100 \\ +\ 11110100 \\ \hline 00000000 \end{array}$$

Dans l'opération ci-contre, nous avons un 1 pour le 9e bit, mais comme notre représentation se limite à 8 bits, il nous reste bien 00000000.

- Pour coder (−3) :

sur 4 bits :
On prend le nombre positif 3 : 0011
On inverse les bits : 1100
On ajoute 1 : 1100 + 0001 = 1101
Donc −3 = 1101

sur 8 bits :
On prend le nombre positif 3 : 0000 0011
On inverse les bits : 1111 1100
On ajoute 1 : 1111 1100 + 0000 0001 = 1111 1101
Donc −3 = 1111 1101

L’espace **codable** est compris entre $\mathbf{-2^{n-1}}$ **et** $\mathbf{2^{n-1} - 1}$ où n représente le nombre de bits.

2) Comment est alors représenté l'entier négatif -1 ?

3) Quels entiers relatifs peut-on représenter avec des mots de 8 bits ? Combien sont-ils ?

4) En utilisant le complément à 2, représentez -15 (représentation sur 8 bits)

5) Représentez sur 8 bits l'entier 4 puis représentez, toujours sur 8 bits, l'entier –5.
   Additionnez ces 2 nombres (en utilisant les représentations binaires bien évidemment).
   Vérifiez que vous obtenez bien -1.

6) Quel est le plus petit entier négatif que l'on peut représenter sur 8 bits ?

7) Quelles sont les bornes inférieure et supérieure d'un entier relatif codé sur 16 bits ?