Leçon 2 - OPÉRATIONS ARITHMÉTIQUES DANS LE SYSTÈME BINAIRE

Avec les connaissances que nous venons d'acquérir, nous sommes en mesure maintenant d'écrire la suite naturelle des nombres binaires.

Base dix	Base deux	
00	0000	
01	0001	
02	0010	
03	0011	
04	0100	
05	0101	
06	0110	
07	0111	
08	1000	
09	1001	
10	1010	
11	1011	etc

Nous ne perdons pas de vue que le but de notre travail est de faire réaliser ces opérations par des machines. Aussi nous devrons nous imposer les deux contraintes suivantes :

1 - Les machines travaillent sur des nombres qui ont toujours la même longueur (ou même format). Ainsi si nous devons fournir à une machine 8 bits le nombre 11 1011, nous devrons en réalité lui entrer : 0011 1011.

2 - Les machines ne travaillent que sur deux nombres à la fois, donc si nous voulons faire A + B + C = S, la machine fera A + B = S_1 puis S_1 + C = S ou (A + B) + C = S

Additions en base deux

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0 et report 1

Exemple :

reports	(1)	0 1 1 1 0 0 0 0
		1 0 0 1 1 1 0 0
	+	1 0 1 1 1 0 1 0
carry (9ème bit)	(1)	0 1 0 1 0 1 1 0

Nous pouvons constater dans l'exemple ci-dessus que le résultat de notre addition "déborde" du format.

On dit qu'il y a OVERFLOW, dépassement de la capacité qui, ici, se traduit par la génération d'un report ou CARRY.

Soustractions en base deux

0 - 0 = 0
1 - 0 = 1
0 - 1 = 1 retenue 1
1 - 1 = 0

Exemple :
Faisons A - B = D
avec : A = 1001 0110 et B = 0110 1100

	A	1 0 0 1 0 1 1 0
	- B	0 1 1 0 1 1 0 0
retenues	fi	1 1 0 1 0 0 0 0
	D	0 0 1 0 1 0 1 0

On remarquera que, ce que nous avions communément l'habitude d'appeler "retenue", se nomme maintenant :

- pour l'addition report (CARRY)
- pour la soustraction retenue (BORROW)

Complément à 2 et complément à 1 d'un nombre binaire

Lorsqu'on fait l'opération suivante :

	1 0 0 0 0 0 0 0 0
-	0 1 1 0 1 1 0 0
	0 1 0 0 1 0 1 0 0

on dit que 1001 0100 est le **complément à 2** de 0110 1100

De même, lorsqu'on fait l'opération suivante :

	1 1 1 1 1 1 1 1
-	0 1 1 0 1 1 0 0
	1 0 0 1 0 0 1 1

1001 0011 est le **complément à 1** de 0110 1100
Il est à noter que pour faire le complément à 1 d'un nombre binaire il suffit de changer les 1 en 0 et les 0 en 1

Si	0	1	1	0	1	1	0	0	est le nombre binaire N
	↓	↓	↓	↓	↓	↓	↓	↓	
	1	0	0	1	0	0	1	1	est le complément à 1 de N

Puisque 1 0000 0000 = 1111 1111 + 1,
le complément à 2 de N étant 1 0000 0000 - N
et le complément à 1 de N étant 1111 1111 - N

Le complément à 2 d'un nombre est donc égal au complément à 1 +1. On en déduit la méthode simple énoncée ci-dessous:

> *Le complément à deux d'un nombre binaire peut s'obtenir en inversant les bits (complément à 1) puis en y ajoutant 1.*

On remarquera que le complément à deux, du complément à deux, est le nombre lui même

Si		0 1 1 0 1 1 0 0	est le nombre binaire N
		↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	
		1 0 0 1 0 0 1 1	est le complément à 1 de N
	+	0 0 0 0 0 0 0 1	
		1 0 0 1 0 1 0 0	est le complément à 2 de N
		↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	
		0 1 1 0 1 0 1 1	nouveau complément à 1
	+	0 0 0 0 0 0 0 1	
		0 1 1 0 1 1 0 0	nouveau complément à 2 qui est égal à N

Soustraction de nombres binaires par la méthode du complément à 2

Reprenons la soustraction déjà effectuée précédemment:

A - B = D

1001 0110 - 0110 1100 = 0010 1010

Si nous faisons ceci:

1001 0110 + (1 0000 0000 - 0110 1100) - 1 0000 0000= 0010 1010

Le même résultat est obtenu

L'opération précédente montre que pour soustraire un nombre binaire à un nombre de 8 bits il suffit de lui ajouter le complément à deux du nombre à soustraire et de soustraire 1 au 9ème bit. Mais comme nous l'avons vu précédemment les machines travaillent sur un format constant ainsi une machine 8 bits devra effectuer des opérations spécifiques lorsqu'elle voudra s'intéresser au 9ème bit donc si l'on se dispense de ces opérations la soustraction finale (- 1 0000 0000) devient inutile.
On peut également utiliser cette technique en passant par le complément à 1

1001 0110 + (1111 1111 - 0110 1100) - 1 0000 0000+ 1 = 0010 1010

Dans ce dernier cas la soustraction A - B par la méthode du complément à deux se ramène à ceci:

Au nombre A on ajoute B inversé, on ajoute 1 et on néglige le 9ème bit

La soustraction par la méthode du complément à 2 est simple en toutes occasions. Elle permet également, en électronique, de réaliser une soustraction avec le même circuit que l'addition. Il suffit pour cela de fournir au circuit additionneur non pas le nombre à soustraire, mais ce nombre passé par des inverseurs et d'ajouter 1 (ce qui peut se faire sur le circuit lui même). Nous étudierons ce circuit ultérieurement.

Il est à noter que lorsque le résultat d'une soustraction est négatif, le nombre résultant est le complément à 2 de la valeur positive. Exemple si nous faisons 150 – 151= -1:

1001 0110 - 1001 0111 = 1111 1111

ce qui est bien le complément à 2 de +1 (0000 0001)

Multiplications en base deux

A	*	B	=	P
0	*	0	=	0
0	*	1	=	0
1	*	0	=	0
1	*	1	=	1

L'opération s'effectue comme en base dix

Exemple:

```
          1  1  0  1
       *  1  0  1  1
          __________
          1  1  0  1
       1  1  0  1  .
 1  1  0  1  .  .  .
 ___________________
1  0  0  0  1  1  1  1
```

Il est à noter que la multiplication de deux nombres de 4 bits donne un résultat sur 8 bits

Cas particulier: la multiplication d'un nombre par une puissance de deux
Dans le **système décimal**, multiplier par 10^3 qui s'écrit 1000 revient à ajouter 3 "0" sur la droite du nombre, de même en **binaire** multiplier par 2^3 qui s'écrit 1000 en binaire revient à ajouter 3 "0" sur la droite. Multiplier un nombre par une puissance n de 2 revient à décaler le nombre de n rang vers la gauche

Divisions en base deux

La division s'effectuera de la même manière qu'en base dix

Exemple:

```
   1  1  1  0  0  1  1  1  |  1  0  1
-  1  0  1                 |___________
   ------------            |
   0  1  0  0  0           |  1  0  1  1  1  0
-        1  0  1           |
   ---------------------   |
   0  0  0  1  1  1        |
-           1  0  1        |
   ------------------------|
   0  0  0  0  1  0  1     |
-              1  0  1     |
   ------------------------|
   0  0  0  0  0  0  0  1  |
```

d'où

111100111 : 101 = 101110 reste 1

Cas particulier:

La division d'un nombre exprimé en binaire naturel, par une puissance n de deux, consiste à supprimer les n bits de poids le plus faible (sur la droite de ce nombre). On sera amené à introduire un nombre égal de "0" à gauche afin que le nombre conserve le même format

Exercices

Opérations arithmétiques en binaire

a/ Effectuer les additions suivantes (*):

1100 + 0011=
1111 + 0101=
10101010 + 00110011=
11001101 + 11100011=

b/ Effectuer les soustractions suivantes(*):

1111 - 0101=
1100 - 0011=
10101010 - 00110011=
11001101 - 01100011=

c/ Multiplier les nombres suivants par 2, 6, 8

00001100, 00010101, 10101000

c/ Diviser les nombres suivants par 2,4, 8

11000000, 01010000, 11001100