1NSI

A - REPRÉSENTATION DES DONNÉES -TYPES ET VALEURS DE BASE

R

Arithmétique binaire

I- L'ADDITION BINAIRE

Les *additions binaires* fondamentales entre bits sont récapitulées dans le tableau ci-contre. **L'addition binaire** fait apparaître la notion de **retenue** :

Addition binaire						
				Retenue		
0	+	0	=		0	
0	+	1	=		1	
1	+	0	=		1	
1	+	1	=	1	0	

Calculer les additions binaires suivantes.

$$1 + 1 + 1 + 1 = \dots$$

$$1 + 1 + 1 + 1 + 1 = \dots$$

Effectuer les additions binaires des nombres décimaux suivants.

II- RAPPELS D'ALGÈBRE

$$a + b = (+a) + (+b)$$

$$a - b = (+a) + (-b)$$

$$-a - b = (-a) + (-b)$$

$$-a + b = (-a) + (+b)$$

En généralisant, on peut dire que **l'addition** ou la **soustraction** de nombres se résument à une seule et même opération :

L'addition de nombres signés.

III- ÉCRITURE D'UN NOMBRE BINAIRE SIGNÉ

Les symboles du signe positif (+) et du signe négatif (-) ne pouvant être utilisés directement en logique booléenne, il existe une convention d'écriture permettant de tenir compte du signe d'un nombre binaire signé.

Cette convention se résume à deux critères :

- 1. Le **bit de poids fort** du format est réservé pour exprimer le signe :
 - a) Si le bit de poids fort vaut 0, le nombre binaire est positif,
 - b) Si le bit de poids fort vaut 1, le nombre binaire est négatif.
- 2. Le **format** d'un nombre binaire signé doit donc être spécifié (4 bits, 8 bits...).

III.A- Nombre binaire signé positif

Sachant que le **bit** de **poids fort** est égal à **zéro**, la **valeur décimale** du nombre est déduite directement des bits restants.

Exemple

Avec 4 bits, 0.101_2 correspond au décimal $+5_{10}$:

Binaire ·	b3	b2	b1	b0
binaire	0	1	0	1
Décimal	+		5	

Avec 8 bits, 0.000.0101₂ correspond aussi au décimal +5₁₀:

Binaire	b7	b6	b5	b4	b3	b2	b1	ь0
	0	0	0	0	0	1	0	1
Décimal	+				5			

• Transcoder les décimaux suivants en binaires signés sur 8 bits.

+17 =	=	
+56 =	=	
+92 =	=	
+110	=	
+127	=	
+128	=	

III.B- Nombre binaire signé négatif

Sachant que le **bit** de **poids fort** est égal à **un**, la valeur décimale (absolue) est déduite du **complément à deux** du nombre binaire négatif lui-même.

La méthode de calcul de la valeur binaire absolue d'un nombre binaire signé négatif est la suivante :

- Complémenter bit par bit le nombre binaire signé négatif :
 - Complémenter à 1
- Rajouter 1 au résultat de la complémentation précédente :
 - Complémenter à 2.

Cette méthode utilise donc les notions de **complémentation** et d'**addition** binaire.

Démonstration

```
Soit X un binaire codé sur n bits et X' son complémentaire alors: X + X' = 2^n - 1 avec 2^n \equiv 0 donc X + X' = -1 et -X = X' + 1 • X': complément (à 1) • X' + 1: complément à 2 En généralisant: X - Y = X + (Y' + 1)
```

Exemple: Soit le nombre binaire signé 1.101 :

Il est négatif car son bit de poids fort est égal à 1.

Son complément à 1 est : C1(1.101) = 0.010.

Son complément à 2 est : $C2(1.101) = C1(1.101) + 1 = 0.010 + 1 = 0.011 = +3_{10}$.

Conclusion

• $1.101 = -3_{10}$

Exemple: Soit le nombre binaire signé 1.000.0101 :

Il est négatif car son bit de poids fort est égal à 1.

Son « complément à 1 » est : C1(1.000.0101) = 0.111.1010.

Son « complément à 2 » est : C2(1.000.0101) = C1(1.000.0101) + 1 = 0.111.1010 + 1 =**0**.111.1011 = +123₁₀.

Conclusion

• $\mathbf{1}.000.0101 = \mathbf{-}123_{10}$

Exemple: Soit le nombre décimal signé -7:

Il est négatif. Son équivalent binaire est C2(+7)

Sur 4 bits par exemple : $+7_{10} = 0.111_2$

Son complément à 1 est : $C1(+7) = 1.000_2$.

Son complément à 2 est : $C2(+7) = C1(+7) + 1 = 1.000 + 1 = 1.001_2$.

Conclusion

$$-7_{10} = 1.001_2$$

• Calculez les compléments à 2 des décimaux signés sur 8 bits.

```
-17 =
-127 =
-128 =
```

IV- SOUSTRACTION DE NOMBRES BINAIRES

Nous avons vu également que l'opération de soustraction se résume à une opération d'addition de nombres signés. Donc pour soustraire un nombre, il suffit :

- de coder le nombre positif à soustraire en un nombre négatif par la méthode du complément à 2,
- puis de l'additionner.
 - Si le résultat est positif, le travail est terminé.
 - Si le résultat est négatif, il faut le complémenter à 2 pour connaître sa valeur absolue.

Exemple

Comment effectuer la soustraction suivante : 5 - 3 ; les nombres binaires étant exprimés dans un format de 4 bits.

$$\begin{aligned} &5_{10} = 0.101_2 \\ &3_{10} = 0.011_2 \\ &C1(3_{10}) = 1.100_2 \\ &C2(3_{10}) = C1(3_{10}) + 1 = 1.100 + 1 = 1.101_2 = -3_{10} \\ &5_{10} + (-3_{10}) = 0.101_2 + 1.101_2 = 1.0.010_2 = \textbf{0}.010_2 \text{ car format de 4 bits.} \end{aligned}$$

Conclusion

$$\mathbf{0}.010_2 = +2_{10}$$

Exemple

Comment effectuer la soustraction suivante : 3 - 5 ; les nombres binaires étant exprimés dans un format de 4 bits.

$$\begin{split} &3_{10}=0.011_2\\ &5_{10}=0.101_2\\ &C1(5_{10})=1.010_2\\ &C2(5_{10})=C1(5_{10})+1=1.010+1=1.011_2=-5_{10}\\ &3_{10}+(-5_{10})=0.011_2+1.011_2=\textbf{1}.110_2. \ C'est\ un\ nombre\ négatif\ car\ le\ bit\ de \end{split}$$

poids fort est égal à 1.

$$C1(1.110_2) = 0.001_2$$

 $C2(0.001_2) = 0.010_2 = 2_{10}$

Conclusion

$$\mathbf{1}.110_2 = -2_{10}$$

• **Effectuez** les soustractions binaires sur 5 bits des décimaux suivants.

V- NOTION D'INDICATEURS D'ÉTATS

Les **indicateurs d'états** ou **bits d'états** ou **drapeaux** (flag en anglais) sont les bits spéciaux d'un **registre d'état**.

Ce registre spécial est réservé pour prévenir le CPU effectuant les opérations logiques et arithmétiques qu'un **résultat singulier** est survenu.

Nous verrons principalement les 3 bits d'état suivants :

- Le bit de **signe** (Negative sign bit), ou bit N,
- Le bit de retenue (Carry), ou bit C,
- Le bit de **débordement** (o**V**erflow), ou bit **V**.

Ces bits sont traités par des instructions spéciales du CPU.

V.A- Le bit de signe N

Il indique si le résultat d'une opération est positif (N=0) ou négatif (N=1).

Exemple

$$0 011 > 0 => N=0$$

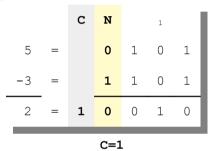
 $1 011 < 0 => N=1$

V.B- Le bit de retenue C

Il indique si le résultat d'une opération nécessite une retenue (C=1) ou ne nécessite pas de retenue (C=0).

• Cela se produit généralement quand il apparaît un bit supplémentaire lors d'une addition par exemple.

Exemple (sur 4 bits)

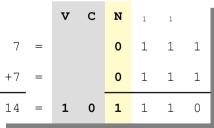


V.C- Le bit de débordement V

Un **débordement** se produit quand **le résultat** n'est pas du bon signe (**V=1**).

• Cela se produit généralement quand on additionne ou on soustrait de grands nombres positifs ou négatifs.

Exemple (sur 4 bits)



Le CPU fait une addition de 2 nombres positifs mais un changement de signe est apparu (N=1)

V=1

• **Effectuez** les opérations binaires, sur 7 bits + 1 bit de signe, suivantes en précisant la valeur des bits d'états N, C et V.

	N	С	v
127 - 1 = 126			
127 + 1 = 128			
19 + 24 = 43			
25 + 4 = 29			
-64 - 65 = -129			
1 - 2 = -1			