

Opérations Arithmétiques

1 Addition en Binaire

1.1 Principe

L'addition de deux nombres binaires est réalisée de la même façon que l'addition décimale.

L'addition est l'opération qui consiste à effectuer :

- Dans un premier temps, la somme **S_i** de deux digits de même rang tels que **A_i** et **B_i** par exemple,
- Puis dans un second temps, une deuxième somme entre les digits de rang supérieur **A_{i+1}** et **B_{i+1}** et la valeur de la retenue issue de l'addition précédente **R_i**.

L'addition de deux nombres binaires se résume par la table suivante :

0	+	0	=	0	
0	+	1	=	1	
1	+	0	=	1	
1	+	1	=	10	1+1=2 donc 10 en binaire (je pose 0 et je retiens 1)

Si l'on désire additionner des nombres binaires comportant plusieurs bits ($a_n, \dots, a_2, a_1, a_0$) + ($b_n, \dots, b_2, b_1, b_0$), on doit ajouter les unités (rang 0), puis les bits de rang 1, puis 2, etc...

Exemple : soit à additionner **101 + 111**, on pose l'opération :

$$\begin{array}{r}
 \\
 \\
 \\
 \\
 \hline
 1 \ 1 \ 0 \ 1 \ \text{Retenues} \\
 1 \ 0 \ 1 \\
 1 \ 1 \ 1 \\
 \hline
 1 \ 1 \ 0 \ 0
 \end{array}$$

Il faut tenir compte, dès le deuxième rang que l'on peut avoir une retenue ; donc il faut ajouter au résultat précédent la retenue de l'addition de rang inférieur. Pour l'addition de deux nombres binaires, on peut donc considérer qu'il faut ajouter à chaque rang, 3 bits :

$$S_n = (a_n + b_n) + R_{n-1}$$

1.2 Table de vérité

c (R _{n-1})	b	a	S _n	R _n
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

La retenue est égale à 1 dès que le résultat de la somme aval atteint la valeur 2 (soit 10 en binaire) ou 3 (soit 11 en binaire).

1.3 Décomposition de la procédure

Soit à effectuer l'addition de deux nombres binaires A et B tels que :

A= 11001011 (203 en décimal)
 B= 10011110 (158 en décimal)

	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
R	1			1	1	1	1		
A	1	1	0	0	1	0	1	1	
+									
B	1	0	0	1	1	1	1	0	
S	1	0	1	1	0	1	0	0	1

- Au premier rang (2^0), il n'y a pas de retenue et le total A_0+B_0 est bien égal à 1,
- Au rang suivant (2^1), il n'y a pas de retenue aval et le total A_1+B_1 est égal à 2. on pose donc 0 et on retient 1 puisque 2 s'écrit 10 en binaire.
- Au rang suivant (2^2), on additionne la retenue aval soit 1 avec A_2 et B_2 se qui donne $1+0+1=10$. On pose 0 et on retient 1.
- Au rang suivant (2^3), on additionne la retenue aval soit 1 avec A_3 et B_3 se qui donne $1+1+1=3$ ou 11 en binaire. On pose 1 et on retient 1
-

Le résultat définitif est donc **101101001** soit **361** en décimal qui correspond bien à **203 + 158**.

2 Addition en hexadécimal

Les règles sont les mêmes que celles de l'addition en décimal ou en binaire à la seule différence que la retenue est égale à 1 dès que la somme de deux digits atteint 16.

Exemple soit à additionner les deux nombres :

A = CB soit $203_{(10)}$
 B = 9E soit $158_{(10)}$

	16^2	16^1	16^0
R	1	1	
A		C	B
+			
B		9	E
S	1	6	9

$B + E = 11 + 14 = 25$ en décimal soit 19 en hexadécimal. On pose 9 et on retient 1
 $1 + C + 9 = 1 + 12 + 9 = 22$ en décimal soit 16 en hexadécimal. On pose 6 et on retient 1

Le résultat définitif est donc **169₍₁₆₎** soit **361** en décimal qui correspond bien à **203 + 158**.

3 Soustraction en Binaire

3.1 Etude comparative

DECIMAL	BINAIRE
$\begin{array}{r} 15 \\ - 96 \\ \hline 09 \end{array}$ <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 10px;"> <div style="text-align: center;"> 19 → 19 9₁ → 9+1 → -10 </div> <div style="text-align: center;"> 09 </div> </div> <p style="margin-top: 10px;">195 - 96 = 99</p>	$\begin{array}{r} 11000011 \\ - 11000000 \\ \hline 1100011 \end{array}$ <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 10px;"> <div style="text-align: center;"> 11 → 11 1₁ → 1+1 → -10 </div> <div style="text-align: center;"> 01 </div> </div> <p style="margin-top: 10px;">11000011 - 11000000 = 1100011</p>

A + (-B) =

$$\begin{array}{r}
 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ A \\
 +\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ -B \\
 \hline
 1\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ -47
 \end{array}$$

Signe (-) ———

Le résultat est bien celui attendu (29 – 76 = - 47). En effet $11010001_{(2)} = -47_{(10)}$

Rappel : pour obtenir la valeur absolue du nombre binaire négatif il faut le complémenter à 2.

$$\begin{array}{r}
 \text{Résultat (négatif)}\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 1 \\
 \text{Complément à 1}\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0 \\
 +\ 1 \\
 \hline
 \text{Complément à 2}\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1
 \end{array}$$

00101111 = 47 donc 11010001 représente bien -47

4 Soustraction dans les autres bases

La méthode du complément à deux en binaire peut être extrapolée pour toutes les autres bases.

En binaire il faut faire le complément à 2
 Le complément à 2 = (complément à 1) + 1

En décimal il faut faire le complément à 10
 Le complément à 10 = (complément à 9) + 1

En Hexadécimal il faut faire le complément à 16
 Le complément à 16 = (complément à 15) + 1

4.1 Soustraction en base 10

Exemple : 76 – 54

Complément à 10 de 54

$$\begin{array}{r}
 \ 5\ 4 \\
 \text{Complément à 9}\ 4\ 5 \\
 +\ 1 \\
 \hline
 \text{Complément à 10}\ 4\ 6
 \end{array}$$

76 – 54 = 22 peut s’obtenir en faisant 76 + 46 (retenue négligée)

$$\begin{array}{r}
 \ 7\ 6 \\
 +\ 4\ 6 \\
 \hline
 \boxed{1}\ 2\ 2
 \end{array}$$

4.2 Soustraction en Hexadécimal

Exemple : F7 – 2E

Complément à 16 de 2E

$$\begin{array}{r}
 \phantom{\text{Complément à 15}} \phantom{\text{Complément à 16}} \\
 \text{Complément à 15} \phantom{\text{Complément à 16}} \\
 \phantom{\text{Complément à 15}} +1 \phantom{\text{Complément à 16}} \\
 \hline
 \text{Complément à 16} \\
 \phantom{\text{Complément à 16}}
 \end{array}$$

F7 – 2E = C9 peut s'obtenir en faisant F7 + D2 (retenue négligée)

$$\begin{array}{r}
 \\
 \\
 \\
 \hline
 \\

 \end{array}$$

1	Addition en Binaire.....	1
1.1	Principe.....	1
1.2	Table de vérité.....	1
1.3	Décomposition de la procédure.....	1
2	Addition en hexadécimal	2
3	Soustraction en Binaire	2
3.1	Etude comparative.....	2
3.2	Soustraction dans le cas des calculateurs	4
3.2.1	Premier cas : $ A > B $	4
3.2.2	Deuxième cas : $ A < \text{ou} = B $	4
4	Soustraction dans les autres bases.....	5
4.1	Soustraction en base 10.....	5
4.2	Soustraction en Hexadécimal.....	6