



# INFORMATIQUE DU BÂTIMENT

## Mathématiques 1er semestre

### Systemes de numération





# Table des matières

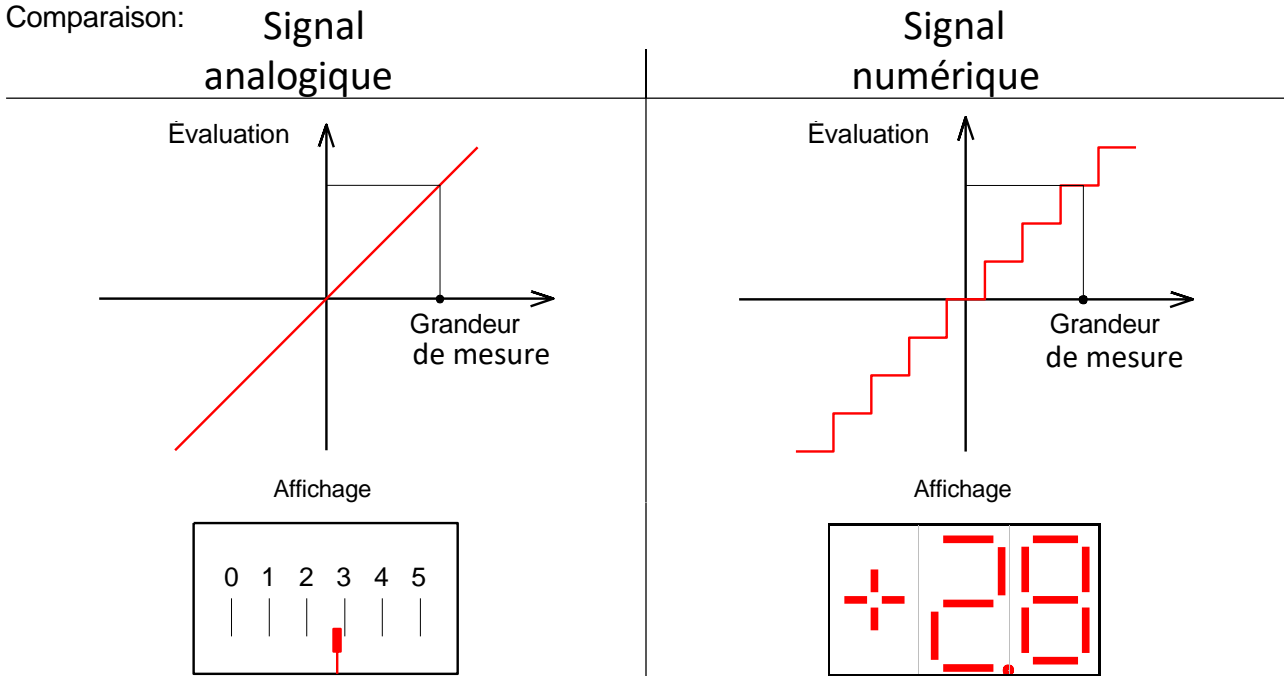
<b>1</b>	<b><i>Introduction</i></b> .....	<b>2</b>
1.1	Signaux analogiques et numériques .....	2
1.2	Signaux binaires .....	2
<b>2</b>	<b><i>Systèmes de numération</i></b> .....	<b>3</b>
2.1	Système décimal .....	3
2.2	Système binaire .....	3
2.3	Système octal et hexadécimal .....	4
2.4	Comparaison des systèmes de numération.....	5
<b>3</b>	<b><i>Conversion de nombres</i></b> .....	<b>6</b>
3.1	Décimal $\leftarrow \rightarrow$ Binaire .....	6
3.2	Décimal $\leftarrow \rightarrow$ HEX .....	7
<b>4</b>	<b><i>Calculer avec des nombres binaires</i></b> .....	<b>8</b>
4.1	Représentation de nombres négatifs.....	8
4.2	Addition et soustraction .....	9
4.3	Soustraction par addition du complément à deux.....	10
4.4	Multiplication et division .....	11
<b>5</b>	<b><i>Exercices divers</i></b> .....	<b>12</b>

# 1 Introduction

## 1.1 Signaux analogiques et numériques

Un **signal** est une grandeur variable dans le temps (p. ex. une tension électrique) qui porte une **information** qui lui est propre.

Comparaison:



Les signaux analogiques peuvent prendre **n'importe quelle** valeur dans une certaine plage.

Les signaux numériques peuvent uniquement varier de manière discrète, autrement dit, **quantifiée**.

## 1.2 Signaux binaires

Un signal binaire est un signal numérique qui n'est reproduit que sur 2 valeurs (niveau logique).

- .....
- .....

Un interrupteur par exemple est un élément binaire. Dans le cas le plus simple, il ne peut adopter que 2 états. Nous pouvons par exemple procéder à l'attribution suivante:

Interrupteur ouvert →

.....

Interrupteur fermé →

.....

Les signaux binaires présentent entre autres les avantages suivants:

- .....
- .....

## 2 Systèmes de numération

### 2.1 Système décimal:

Nous calculons généralement dans le *système décimal* ou *système de base 10*. La valeur de chaque chiffre dépend de sa position. On parle alors d'un **système de position**.

Exemple:

*MSD*<sup>1</sup>

*LSD*

$2573_{10}$	2	5	7	3
-------------	---	---	---	---

← suite de chiffres

Base 10

← valeur

$2573_{10} =$

-----  
-----

### 2.2 Système binaire:

Le *système binaire* est un système de numération à 2 valeurs. Il ne comprend que les chiffres 0 et 1.

Exemple:

*MSB*<sup>2</sup>

*LSB*

$1101_2$	1	1	0	1
----------	---	---	---	---

← suite de chiffres

Base 2

← valeur

$1101_2 =$

-----  
-----

- Les *systèmes de numération* construits selon le système de position possèdent une réserve de chiffres correspondant à la valeur de la base (p.ex. système binaire = 2 chiffres).
- La valeur du chiffre le plus élevé est inférieure de 1 à la base.
- La *valeur* du voisin de gauche d'un chiffre est plus grande du facteur de la base, celle du voisin de droite est plus petite du même facteur.

<sup>1</sup> *MSD, LSD* engl. most significant, least significant digit (chiffre le plus ou le moins significatif)

<sup>2</sup> *MSB, LSB* engl. most significant, least significant bit (bit le plus ou le moins significatif)



### 2.3 Système octal et hexadécimal

Le système *binnaire* est idéal pour un ordinateur mais pour nous, plutôt inconfortable. Pour simplifier l'utilisation, nous regroupons donc dans le *système octal*, 3 chiffres binaires pour former un **chiffre octal** ou dans le *système hexadécimal*, nous regroupons 4 chiffres binaires pour former un **chiffre HEX**.

	Base	Réserve de chiffres
Système octal		
Système HEX <sup>3</sup>		

**Exemple:** donné: chiffre binaire = 1010101001<sub>2</sub>  
recherché: a) chiffre octal  
b) chiffre HEX

a) binaire → octal

001      010      101      0012      Groupes de 3

     Pondération

     Chiffre octal

b) binaire → HEX

0010      1010      1001<sub>2</sub>      Groupes de 4

     Pondération

     Chiffre HEX

**Essai:**

-----  
-----

<sup>3</sup> les chiffres HEX sont souvent désignés par un H placé à la suite, p. ex: 0FH



## 2.4 Comparaison des systèmes de numération

**Exercice:** Complétez le tableau suivant!

Décimal	Binaire	Octal	HEX
0	00000	00	00
1	00001	01	01
2	00010	02	02
3	00011	03	03
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			
28			
29			
30			
31			



### 3 Conversion de nombres

#### 3.1 Décimal $\leftarrow \rightarrow$ Binaire

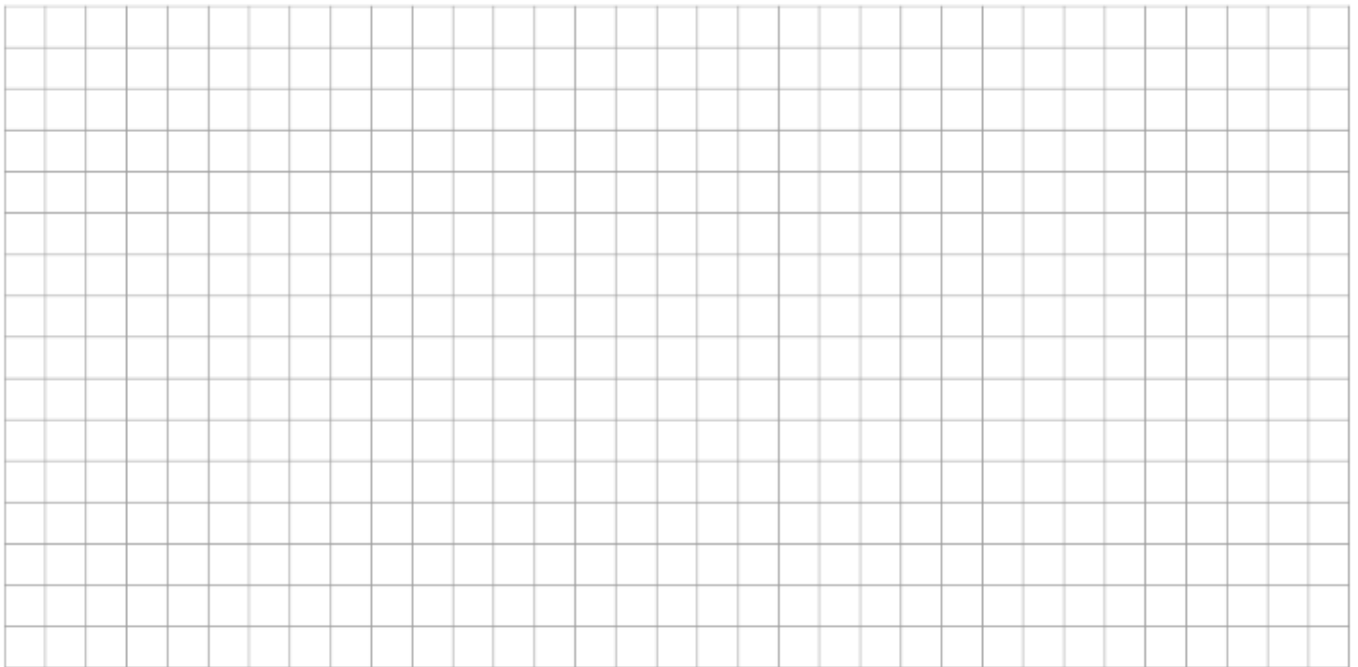
Les nombres décimaux sont transformés en nombres binaires par une division continue par la base 2.

Exemple:    donné:  $91_{10}$   
              rech.:  $x_2$

91	:	2 =		reste	
			.....	reste	.....
			.....	reste	.....
			.....	reste	.....
			.....	reste	.....
			.....	reste	.....
			.....	reste	.....
			.....	reste	.....
			.....	reste	.....

$x_2 =$  .....

Exercice:    Représenter le nombre  $200_{10}$  en nombre binaire!



La conversion **binaire**  $\rightarrow$  **décimal** s'obtient en additionnant les puissances de 2.

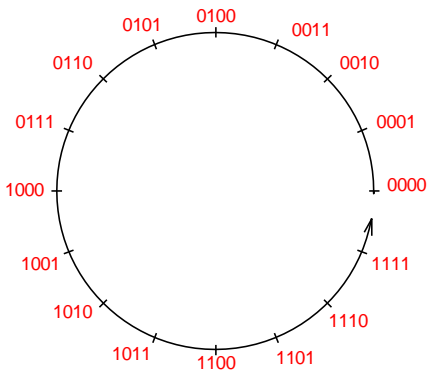
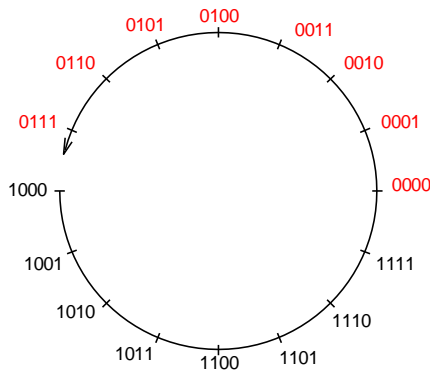
Exemple:  $10110110_2$  .....





## 4 Calculer avec des nombres binaires

### 4.1 Représentation des nombres négatifs

Plage de nombres	Nombres positifs	Nombres positifs et négatifs
Arithmétique 4 bits		
Arithmétique 8 bits	<div style="border: 1px solid black; height: 100px; width: 100%;"></div>	<div style="border: 1px solid black; height: 100px; width: 100%;"></div>

L'ordinateur gère les nombres négatifs de manière binaire en complément à deux.

**Exemple:** à quoi correspond le nombre  $-2_{10}$  eines d'un ordinateur 4 bits en complément à 2?

Z	
$\bar{Z}$	
$-Z$	

chiffre positif

complément à 1

complément à 2

En général:

Le complément à 2 s'obtient en ajoutant 1 au complément à 1.



## 4.2 Addition et soustraction

Les nombres binaires s'additionnent ou se soustraient de la même manière que les nombres décimaux.

Règles de calcul

Opération	résultat	report
$0 + 0$		
$0 + 1$		
$1 + 0$		
$1 + 1$		
$1 + 1 + 1$		

Opération	résultat	report
$0 - 0$		
$0 - 1$		
$1 - 0$		
$1 - 1$		
$0 - 1 - 1$		
$1 - 1 - 1$		

Exemples d'addition et de soustraction directe

1.	<i>décimal</i>	<i>binair</i>
	11	1011
report (carry):	<u>+ 19</u>	<u>+ 10011</u>
résultat:	<u><u>          </u></u>	<u><u>          </u></u>
2.	<i>décimal</i>	<i>binair</i>
	30	11110
report (borrow):	<u>- 11</u>	<u>- 1011</u>
résultat:	<u><u>          </u></u>	<u><u>          </u></u>
3.	<i>décimal</i>	<i>binair</i>
	7	111
report (borrow):	<u>- 8</u>	<u>- 1000</u>
résultat:	<u><u>          </u></u>	<u><u>          </u></u>

### 4.3 Soustraction par addition du complément à 2

Les ordinateurs effectuent généralement une soustraction sous la forme d'une addition avec un nombre négatif:

$$Z1 - Z2 = Z1 + (-Z2)$$



chiffre négatif,  
représentation en  
complément à 2.

**Exemple:** Un ordinateur 8 bits doit effectuer la soustraction 10111 – 110.

Procédure:

1. Complément à la largeur de bit de l'ordinateur, ici 8 chiffres
2. Formation du complément à 2 du soustracteur
3. Exécution de l'addition

1. complément pour atteindre 8 positions

10111 → 00010111  
110 → 00000110

2. Formation du complément à 2

00000110 (chiffre)  
11111001 (compl. à 1)  
11111010 (compl. à 2 = compl. à 1+1)

3. soustraction / addition

*direct*

*indirect*

00010111  
- 00000110

00010111  
+ 11111010

résultat:

**Exercice:** effectuer le calcul 110 – 10111 à l'aide du complément à 2!



## 4.4 Multiplication et division

Règles de calcul

*Multiplication*

0	•	0	=	0
0	•	1	=	0
1	•	0	=	0
1	•	1	=	1

*Division*

0	:	1	=	0
1	:	1	=	1

### Exemples:

1.

*Décimal*

$$\underline{20} \cdot 11$$

*Binaire*

$$10100 : 1011$$

2.

*Décimal*

$$33 : 11 =$$

*Binaire*

$$100001 : 1011 =$$

### Fractions binaires

Des chiffres après la virgule peuvent apparaître après la division.

Exemple:  $0,111_2 =$  .....

## 5 Exercices divers

	donné:		recherché:	
	nombre	système	système	nombre
1.	145'999	déc	hd	
2.	4832	déc	hd	
3.	2623	déc	hd	
4.	9CA	hd	déc	
5.	123F7	hd	déc	
6.	ACDC	hd	déc	
7.	1346	déc	2er	
8.	1537	déc	2er	
9.	63	déc	bin	
10.	1023	déc	dual	
11.	1010101	2	10	
12.	111000111	2	10	
13.	100100100	2	10	
14.	1111011101101	2	10	
15.	1000	déc	oct	
16.	1985	déc	oct	
17.	1234	oct	déc	
18.	5670	oct	déc	
19.	BBC	hd	déc	
20.	BCD	hd	déc	