

# NUMÉRIQUE

LES BASES 2, 10 ET 16

s.bolay, CFPs-EMVs, 2009

## LE DÉCIMAL

\* Soit le chiffre 317

Si l'on souhaite exprimer ce chiffre en terme d'unité, de dizaine, centaine ceci ne demande pas un grand effort, car c'est la base dans laquelle nous sommes habitué à compter.

Exemple:

3 centaines + 1 dizaine + 7 unités

$$3 * 100 + 1 * 10 + 7 * 1$$

$$3 * 10^2 + 1 * 10^1 + 7 * 10^0$$

# LE DÉCIMAL

✿ Soit le chiffre 317

De manière plus systématique, pour exprimer ce chiffre, il faut d'abord déterminer la quantité de "nombre" nécessaire à l'exprimer.

$10^0 = 1$ , permet de compter de 0 à 9

$10^1 = 10$ , permet de compter de 0 à 99

$10^2 = 100$ , permet de compter de 0 à 999

$10^3 = 1000$ , permet de compter de 0 à 9999

...

$10^3$	$10^2$	$10^1$	$10^0$	#
			9	9
		9	9	99
	9	9	9	999
9	9	9	9	9999

# LE DÉCIMAL

✿ Soit le chiffre 317

$10^2 = 100$ , permet de compter de 0 à 999.

sa représentation se retrouve par la division:

$317/100 = 3$  (reste 17)

$17/10=1$  (reste 7)

$7/1=7$  (reste 0)

=> 3 centaines + 1 dizaine + 7 unités

$10^3$	$10^2$	$10^1$	$10^0$	#
			9	9
		9	9	99
	<b>3</b>	<b>1</b>	<b>7</b>	<b>317</b>
9	9	9	9	9999



# LE BINAIRE

✱ Soit le chiffre 317

$2^8 = 256$ , permet de compter de 0 à 511.

sa représentation en binaire s'effectue grace à la division:

$317/256=1$  (reste 61)

$61/128=0$  (reste 61)

$61/64=0$  (reste 61)

$61/32=1$  (reste 29)

$29/16=1$  (reste 13)

$13/8=1$  (reste 5)

$5/4=1$  (reste 1)

$1/2=0$  (reste 1)

$1/1=1$  (reste 0)

=> 100111101

$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	#
								1	1
							1	1	3
						1	1	1	7
					1	1	1	1	15
				1	1	1	1	1	31
			1	1	1	1	1	1	63
		1	1	1	1	1	1	1	127
	1	1	1	1	1	1	1	1	255
1	0	0	1	1	1	1	0	1	317

# LE BINAIRE

✱ Soit le chiffre 317

1 256'taine + ... + 1 unité

sa représentation en décimal s'effectue grace à la multiplication:

$$\begin{aligned}
 & 1 \cdot 2^8 + 0 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\
 &= 1 \cdot 256 + 0 \cdot 128 + 0 \cdot 64 + 1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 \\
 &= 256 + 0 + 0 + 32 + 16 + 8 + 4 + 0 + 1 \\
 &= 317
 \end{aligned}$$

$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	#
								1	1
							1	1	3
						1	1	1	7
					1	1	1	1	15
				1	1	1	1	1	31
			1	1	1	1	1	1	63
		1	1	1	1	1	1	1	127
	1	1	1	1	1	1	1	1	255
1	0	0	1	1	1	1	0	1	317

## CONVERSION DE LA BASE 10 (DÉCIMAL) EN BASE 2 (BINAIRE).

$2^3$	$2^2$	$2^1$	$2^0$	$10^1$	$10^0$	#
0	0	0	0	0	0	1
0	0	0	1	0	1	2
0	0	1	0	0	2	3
0	0	1	1	0	3	4
0	1	0	0	0	4	5
0	1	0	1	0	5	6
0	1	1	0	0	6	7
0	1	1	1	0	7	8
1	0	0	0	0	8	9
1	0	0	1	0	9	10
1	0	1	0	1	0	11
1	0	1	1	1	1	12
1	1	0	0	1	2	13
1	1	0	1	1	3	14
1	1	1	0	1	4	15
1	1	1	1	1	5	16

} 4 bits en binaire ou 2  
digits en décimal permet  
de coder 16 valeurs, soit  
de 0 à 15.

## CONVERSION DE LA BASE 10 (DÉCIMAL) EN BASE 16 (HEXADÉCIMAL).

$16^1$	$16^0$	$10^1$	$10^0$	#
0	0	0	0	1
0	1	0	1	2
0	2	0	2	3
0	3	0	3	4
0	4	0	4	5
0	5	0	5	6
0	6	0	6	7
0	7	0	7	8
0	8	0	8	9
0	9	0	9	10
0	A	1	0	11
0	B	1	1	12
0	C	1	2	13
0	D	1	3	14
0	E	1	4	15
0	F	1	5	16

} 1 chiffre en hexadécimal  
ou 2 digit en décimal  
permet de coder 16  
valeurs, soit de 0 à 15.

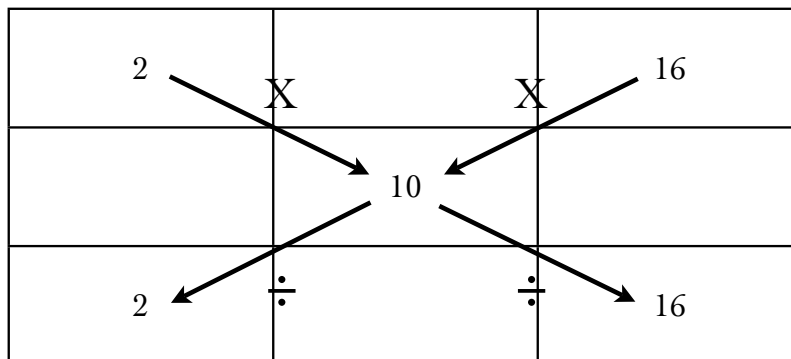
## CONVERSION DE LA BASE 16 (HEXADÉCIMAL) EN BASE 2 (BINAIRE).

$2^3$	$2^2$	$2^1$	$2^0$	$16^1$	$16^0$	#
0	0	0	0	0	0	1
0	0	0	1	0	1	2
0	0	1	0	0	2	3
0	0	1	1	0	3	4
0	1	0	0	0	4	5
0	1	0	1	0	5	6
0	1	1	0	0	6	7
0	1	1	1	0	7	8
1	0	0	0	0	8	9
1	0	0	1	0	9	10
1	0	1	0	0	A	11
1	0	1	1	0	B	12
1	1	0	0	0	C	13
1	1	0	1	0	D	14
1	1	1	0	0	E	15
1	1	1	1	0	F	16

}

4 bits en binaire ou 1 chiffre en hexadécimal permet de coder 16 valeurs, soit de 0 à 15.

## LA BASE 10 COMME PIVOT

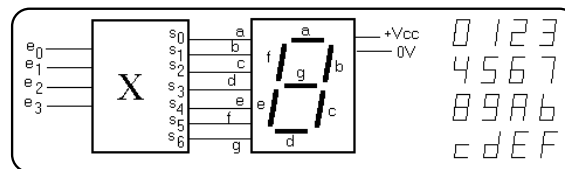


## LE DÉCIMAL CODÉ EN BINAIRE (DCB OU BCD EN ANGLAIS)

Si vous achetez un voltmètre numérique, la valeur mesurée est transmise à l'afficheur en numérique. Mais elle est auparavant transformée en décimal, chaque chiffre décimal est transmis à un afficheur en binaire naturel (sur 4 bits). C'est le BCD : la juxtaposition des valeurs binaires (sur quatre bits) des chiffres décimaux. Donc  $583_{10}$  se notera  $0101\ 1000\ 0011_{bcd}$ . Cette codification pose deux problèmes principaux :

- un certain nombre de combinaisons ne sont pas utilisées (celles qui correspondent à A à F en hexa). Sur 8 bits on représente les nombres de 0 à 99 au lieu de 0 à 255 en binaire. Sur 16 bits, on se limite à 9999 au lieu de 65535...
- les calculs sont compliqués : rien que pour faire un programme d'incrémement (ajouter 1), il faut ajouter 1 aux 4 bits de droite, si on obtient 1010 ( $10_{10}$ ), on les remplace par 0000 et on ajoute 1 aux quatre bits suivants (dizaines), sans oublier de vérifier si l'on ne passe pas à la centaine suivante...

Dès qu'il y a des calculs à effectuer, les systèmes numériques traduisent les nombres BCD en binaire dès leur acquisition, les résultats seront transformés en BCD au moment de leur sortie. On peut remarquer que pour transformer un nombre binaire en décimal (bcd), l'ordinateur est obligé de faire des divisions successives par 1010 (10 en binaire)



## EXEMPLES D'APPLICATION DU NUMÉRIQUE:

- ✿ Ordinateurs
- ✿ Téléphonie mobile
- ✿ Automates programables
- ✿ Les cartes bancaires/téléphoniques