

# Le code binaire



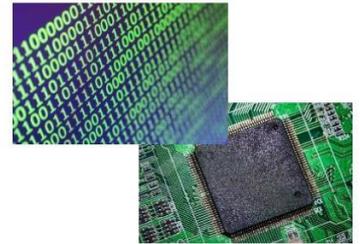
## Rappels

Le **code binaire** est le langage utilisé par le microprocesseur de l'ordinateur pour traiter les informations.

Ce code constitué de 0 et de 1 correspond à des impulsions électriques :

0 | pas de courant électrique ; 1 | courant électrique

Le **BIT** (Binary digiT) est l'unité du codage binaire ayant pour valeur 0 ou 1.



## Le code ASCII

L'**American Standard Code for Information Interchange** (Code américain normalisé pour l'échange d'information), plus connu sous l'acronyme **ASCII**, est une norme informatique de codage de caractères apparue dans les années 60. C'est la norme de codage de caractères la plus influente à ce jour. A l'origine ASCII définit 128 codes à 7 bits allant de 0 à 127.

Le tableau de la page 2 présente le code ASCII | Si je saisis au clavier touche ALT + 65 dans un traitement de texte, le code binaire 1000001 est envoyé au microprocesseur et le caractère « A » s'affiche sur la page.

## Le comptage binaire

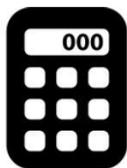
Le système de comptage utilisé en mathématiques est appelé **système décimal**. C'est un système en base 10 car basé sur **10 chiffres** (0 à 9).

Le **système binaire** de comptage est un système en base 2 car basé sur **2 chiffres** (0 et 1). Le comptage binaire a de nombreuses applications informatiques comme par exemple dans le fonctionnement des réseaux informatiques.

Je peux convertir un nombre binaire en nombre décimal et inversement.

Le poids du bit est doublé à chaque colonne de la gauche vers la droite (1,2,4,8,16,32,64).

La valeur du bit 0 ou 1 agit comme un coefficient multiplicateur du poids.



Exemple de comptage sur 7 bits :

Lettre « A » = code binaire 1000001

$$64 \times 1 + 32 \times 0 + 16 \times 0 + 8 \times 0 + 4 \times 0 + 2 \times 0 + 1 \times 1 = 64 + 1 = 65$$

1000001 en binaire correspond à 65 en décimal.

### Comptage binaire sur 7 bits

Comptage décimal

$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
64	32	16	8	4	2	1

0 |  
1 |  
2 |  
.  
.  
65 |  
.  
.  
127 |

0	0	0	0	0	0	0
0	0	0	0	0	0	1
0	0	0	0	0	1	0

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

1	1	1	1	1	1	1
---	---	---	---	---	---	---

## Tableau de la norme ASCII

Decimal	Binary	Char	Decimal	Binary	Char	Decimal	Binary	Char
0	0	[NULL]	48	110000	0	96	1100000	`
1	1	[START OF HEADING]	49	110001	1	97	1100001	a
2	10	[START OF TEXT]	50	110010	2	98	1100010	b
3	11	[END OF TEXT]	51	110011	3	99	1100011	c
4	100	[END OF TRANSMISSION]	52	110100	4	100	1100100	d
5	101	[ENQUIRY]	53	110101	5	101	1100101	e
6	110	[ACKNOWLEDGE]	54	110110	6	102	1100110	f
7	111	[BELL]	55	110111	7	103	1100111	g
8	1000	[BACKSPACE]	56	111000	8	104	1101000	h
9	1001	[HORIZONTAL TAB]	57	111001	9	105	1101001	i
10	1010	[LINE FEED]	58	111010	:	106	1101010	j
11	1011	[VERTICAL TAB]	59	111011	;	107	1101011	k
12	1100	[FORM FEED]	60	111100	<	108	1101100	l
13	1101	[CARRIAGE RETURN]	61	111101	=	109	1101101	m
14	1110	[SHIFT OUT]	62	111110	>	110	1101110	n
15	1111	[SHIFT IN]	63	111111	?	111	1101111	o
16	10000	[DATA LINK ESCAPE]	64	1000000	@	112	1110000	p
17	10001	[DEVICE CONTROL 1]	65	1000001	A	113	1110001	q
18	10010	[DEVICE CONTROL 2]	66	1000010	B	114	1110010	r
19	10011	[DEVICE CONTROL 3]	67	1000011	C	115	1110011	s
20	10100	[DEVICE CONTROL 4]	68	1000100	D	116	1110100	t
21	10101	[NEGATIVE ACKNOWLEDGE]	69	1000101	E	117	1110101	u
22	10110	[SYNCHRONOUS IDLE]	70	1000110	F	118	1110110	v
23	10111	[ENG OF TRANS. BLOCK]	71	1000111	G	119	1110111	w
24	11000	[CANCEL]	72	1001000	H	120	1111000	x
25	11001	[END OF MEDIUM]	73	1001001	I	121	1111001	y
26	11010	[SUBSTITUTE]	74	1001010	J	122	1111010	z
27	11011	[ESCAPE]	75	1001011	K	123	1111011	{
28	11100	[FILE SEPARATOR]	76	1001100	L	124	1111100	
29	11101	[GROUP SEPARATOR]	77	1001101	M	125	1111101	}
30	11110	[RECORD SEPARATOR]	78	1001110	N	126	1111110	~
31	11111	[UNIT SEPARATOR]	79	1001111	O	127	1111111	[DEL]
32	100000	[SPACE]	80	1010000	P			
33	100001	!	81	1010001	Q			
34	100010	"	82	1010010	R			
35	100011	#	83	1010011	S			
36	100100	\$	84	1010100	T			
37	100101	%	85	1010101	U			
38	100110	&	86	1010110	V			
39	100111	'	87	1010111	W			
40	101000	(	88	1011000	X			
41	101001	)	89	1011001	Y			
42	101010	*	90	1011010	Z			
43	101011	+	91	1011011	[			
44	101100	,	92	1011100	\			
45	101101	-	93	1011101	]			
46	101110	.	94	1011110	^			
47	101111	/	95	1011111	_			