

La numération Les chiffres et les lettres

Lors d'une transmission, les systèmes ne reconnaissent que des suites de bits à 1 et à 0.

Nous savons que pour des raisons spécifiques, il a été choisi d'associer les bits par blocs de 8 que nous appelons octets. Ces blocs représentent sans difficulté des nombres. Mais nous humains, nous utilisons essentiellement des lettres pour nous exprimer. Comment une machine peut elle se représenter les lettres et les mots ?

Une lettre est un caractère spécifique. Il existe différents codages normalisés des caractères. Le plus connu et le plus utilisé est le code ASCII : *American Standard Code for Information Interchange* créé dans les années 60 au USA.

Le code ASCII permet la reconnaissance de 128 caractères par un code allant de 0 à 127 en décimal (ou 7F en hexadécimal).

La table en annexe donne une définition de ces codes.

Les 32 premiers codes sont appelés codes de contrôle. Il ne génèrent pas de caractères visibles, mais des actions d'affichage

A partir de 32 , on trouve les chiffres (caractère affichable de '0' à '9')

A partir de 65 , les lettres majuscules ('A' à 'Z')

A partir de 97 , les lettres minuscules ('a' à 'z')

Enfin les espaces intermédiaires sont complétés par les caractères de ponctuation.

Cette table permet la rédaction de mots et de textes par une succession de caractères, donc une succession d'octets. Ce même texte peut être stocké sans problème dans une mémoire ou un disque à raison d'un caractère par octet.

Nous noterons toutefois, qu'un système différencie le nombre 136, du mot '136' qui est une suite caractères.

Le nombre est codé sur 1 seul octet : valeur $(88)_{16}$

Le mot est codé sur 3 octets : $(31)_{16}$; $(33)_{16}$; $(36)_{16}$

Si l'on utilise l'octet comme élément de dimensionnement, on s'aperçoit que dans le cas présent le premier bit est toujours à 0. Il a, en effet, été convenu que ce code est défini sur 7 bits. La norme affirme également qu'il est possible d'utiliser une version étendue du code ASCII ou les code de 128 à 255 (80 à FF en hexadécimal) sont à définir par l'utilisateur. Ce système permet de gérer les caractères spécifique à certaine langue. Par exemple en français l'existence des accents sur les voyelles qui n'existent pas en anglais.

Certaines extensions sont normalisées :

- Code ASCII étendu ANSI : pour nos accents ou

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	□	□	,	f	//	...	†	‡	^	‰	Š	<	œ	□	□	□
9	□	\	/	ˆ	˜	•	–	—	˝	‰	š	>	œ	□	□	ÿ
A		ı	ı	£	¤	¥	¦	§	¨	©	ª	«	¬	–	®	—
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ø	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

- Code ASCII étendu OEM : pour les caractères semi graphiques

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	ç	ü	é	â	ä	à	&	ç	ê	ë	è	ï	î	ì	ñ	ß
9	é	æ	ff	ô	ö	ò	û	ù	ÿ	ö	ü	ç	£	¥	℞	f
A	á	í	ó	ú	ñ	Ñ	©	©	¿	¡	¬	½	¾	ı	«	»
B	▩	▩	▩		†	‡		¶	¶			¶	¶	¶	¶	¶
C	⌞	⌞	⌞	†	–	†	†		⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞
D	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞
E	α	β	Γ	Π	Σ	σ	μ	τ	θ	θ	Ω	δ	ω	ω	€	π
F	≡	±	≥	≤	†	J	÷	∞	o	.	.	√	”	z		

Annexe : Table ASCII standard:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SoH	SoT	ETx	EoT	Enq	Ack	Bell	BS	Tab	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	Syn	ETB	Can	EoM	Sub	Esc	FS	GS	RS	US
2	Space	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Del