

Numérisation de l'information

Cours p 522-526

Exercices corrigés: 9* 10* 13* 16* 22* 21 29p 529-536 Exercices : 12 - 15 - 18 - 24- 26-30 p 529-536

1- Quelle est la différence entre un signal analogique et un signal numérique ?

1.1- Signal analogique et signal numérique

Un signal est la représentation physique d'une information qui est transportée d'une source à un destinataire. Ce signal peut être capté et converti en un signal électrique. Il en existe deux types :

- Si le signal électrique varie de façon **continue** au cours du temps, il est **analogique**.

Ex : Un microphone transforme un son en signal électrique analogique.

- Si le signal électrique varie de façon **discrète** (ou discontinue), **par paliers**, il est **numérique**.

Ex : Les ordinateurs ne traitent que des signaux numériques.

Livre doc 2 p. 522 Hachette

1.2- Codage binaire d'un signal numérique

a- Bits et octet

Dans un système numérique, chaque circuit électronique qui le compose peut produire un signal numérique avec deux niveaux de tension électrique :

- une tension basse, **codée 0**,
- une tension haute, **codée 1**.

On parle de codage binaire.

Le fichier numérique associé au signal, est donc la succession de « 0 » ou « 1 », appelés bits.

Un bit est la plus petite unité d'information numérique. Il ne peut prendre que deux valeurs (0 ou 1).

Avec un bit, on ne peut donc représenter que deux états différents, 0 ou 1. Avec deux bits on peut représenter quatre états différents (2×2) (01 – 10 - 11 - 00)

On peut étendre cette progression et énoncer une loi fondamentale : **n bits $\Rightarrow 2^n$ états différents.**

En informatique, l'unité d'information composée de 8 bits constitue l'octet.

1 octet = 8 bits soit $2^8 = 256$ états différents.

Remarque : La manipulation d'un octet lors d'une phase de traitement ou de stockage informatique est l'équivalent d'un seul caractère, tel qu'une lettre, un chiffre ou un signe de ponctuation.

b- Représentation d'un nombre en binaire

Base 10	Base 2	Décomposition	Base 10	Base 2	Décomposition
0	0	0	5	101	$1 \times 2^2 + 0 \times 2^1 + 1$
1	1	1	6	110	$1 \times 2^2 + 1 \times 2^1 + 0$
2	10	$1 \times 2^1 + 0$	7	111	$1 \times 2^2 + 1 \times 2^1 + 1$
3	11	$1 \times 2^1 + 1$	8	1000	$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0$
4	100	$1 \times 2^2 + 0 \times 2^1 + 0$	9	1001	$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1$

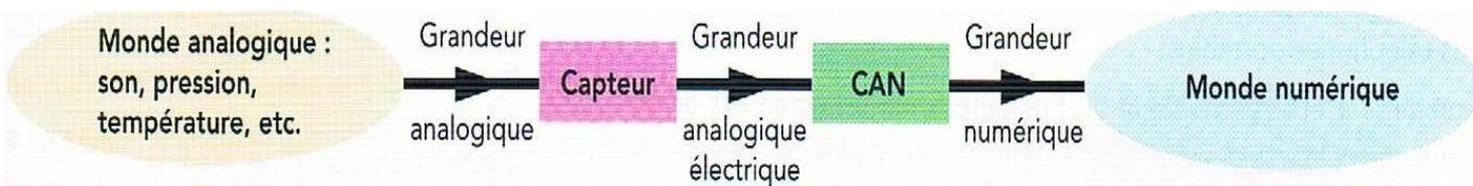
Dans ces conditions, il y a 256 représentations possibles, du nombre le plus petit : 0 0 0 0 0 0 0 au nombre le plus grand possible avec huit bits : 1 1 1 1 1 1 1 1 qui est, en décimal, 255.

2- Quelles sont les principales étapes de la numérisation d'un signal analogique ?

La transformation d'un signal analogique en signal numérique est appelée numérisation.

2.1- Principe

A l'entrée du **convertisseur analogique numérique(CAN)** est appliquée une tension électrique, grandeur analogique représentative du phénomène physique étudié à l'aide d'un capteur approprié ; à sa sortie, est généré un nombre binaire, codé sur un certain nombre de bits



2.2- Etapes de la numérisation

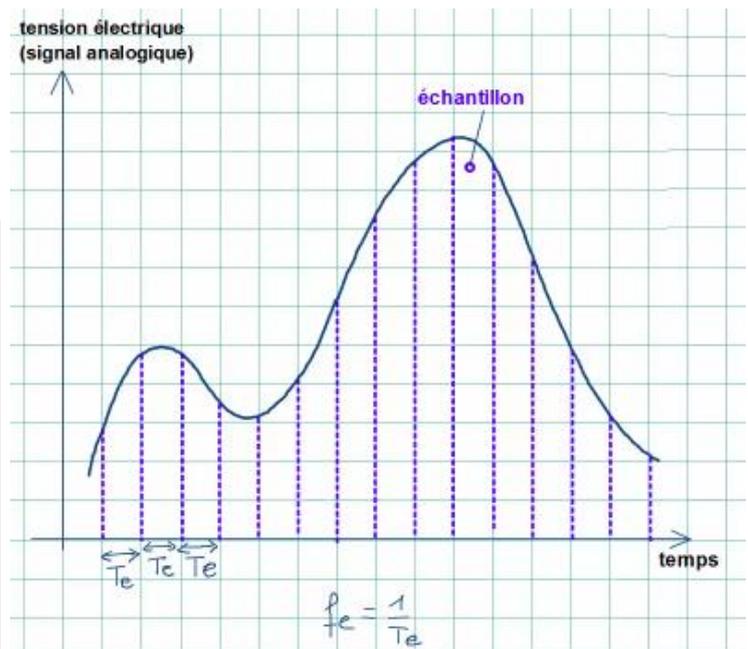
Toute numérisation comprend trois étapes : **l'échantillonnage, la quantification et le codage.**

a- L'échantillonnage :

Le CAN prélève, à intervalles de temps régulier les valeurs de tension du signal analogique.

L'intervalle de temps entre deux prélèvements est appelé **période d'échantillonnage et notée T_e** . Elle se mesure en secondes (s). Elle dépend du CAN utilisé.

La **fréquence d'échantillonnage f_e** , correspond ainsi au nombre de points retenus par seconde sur le signal analogique. f_e en Hz, définie par **$f_e = 1 / T_e$** .



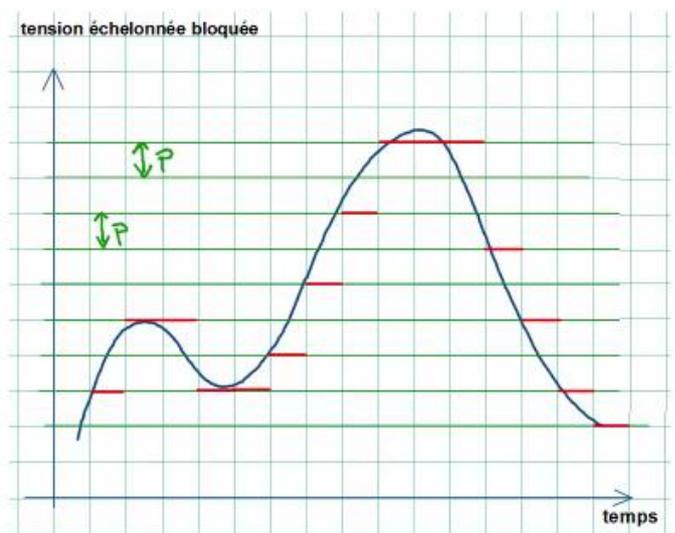
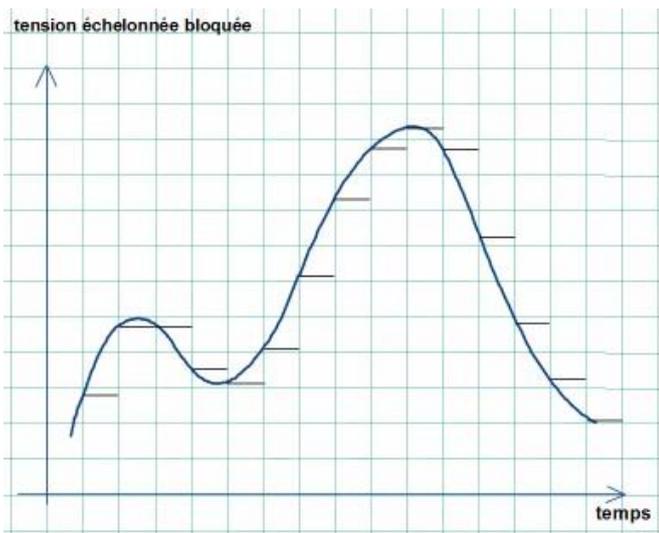
La fréquence d'échantillonnage doit être suffisamment grande pour pouvoir reconstituer convenablement les variations du signal analogique ($f_e \geq 2f$)
($\Leftrightarrow T_e$ doit être suffisamment petite devant la période du signal analogique à échantillonner.)

b- La quantification et le codage

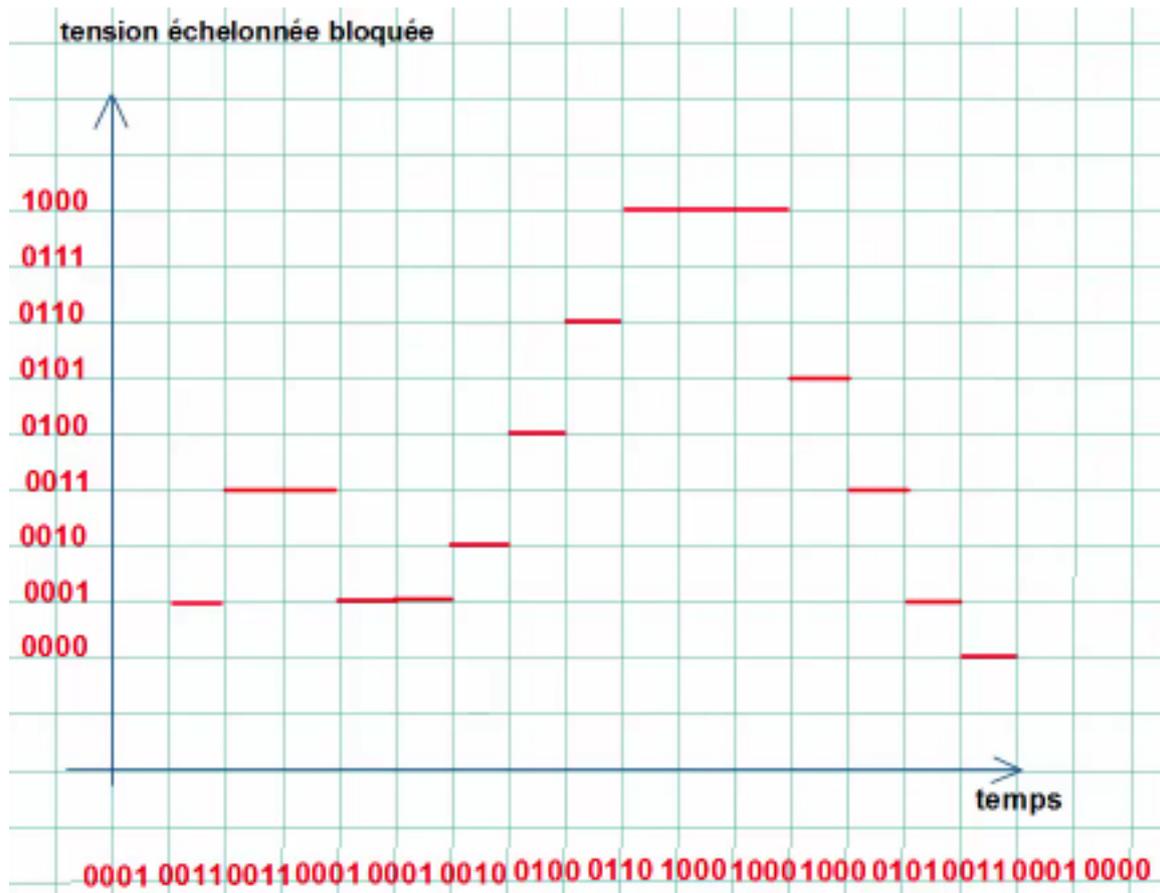
Le CAN possède une gamme de valeurs comprises entre S_{\min} et S_{\max} entre lesquelles les valeurs du signal à numériser doivent situer.

Les valeurs des tensions sont « bloquées » pendant la durée d'échantillonnage T_e . Le CAN crée des niveaux de quantifications afin de coder les valeurs du signal

Chaque valeur de tension échelonnée va prendre une valeur binaire, dont le niveau est le plus proche possible des niveaux de quantifications.



**Le nombre de niveaux dépend du nombre de bits du convertisseurs = 2^N
niveaux de quantification (N bits)**



La numérisation permet d'obtenir une suite de nombres binaires qui correspondra, au plus près, à l'évolution du signal analogique. On obtient donc un signal numérique, à valeurs discrètes.

Remarque : la numérisation est meilleur pour une période d'échantillonnage T_e et un pas p faibles.

2.3- Caractéristiques d'un CAN.

Le convertisseur est caractérisé par son pas p , en volt,

C'est la plus petite variation du signal que le CAN peut détecter, soit l'écart entre deux niveaux de quantification :

$$p = \frac{\text{plage de mesure}}{2^N}$$

Plage de mesure de la tension analogique = gamme de valeur

3- Quel est le principe du codage d'une image numérique ?

3.1- La pixellisation

Un appareil photographique numérique, ou un scanner, assure la numérisation d'une image. Pour cela, l'appareil découpe l'image en un quadrillage. Chaque case de ce quadrillage est appelée un **pixel** (px).

La pixellisation est donc le fait de convertir une image en pixels.

Chaque pixel est composé de trois sous-pixels colorés en rouge, vert et bleu = RVB

3.2- Principe de codage d'une image en RVB

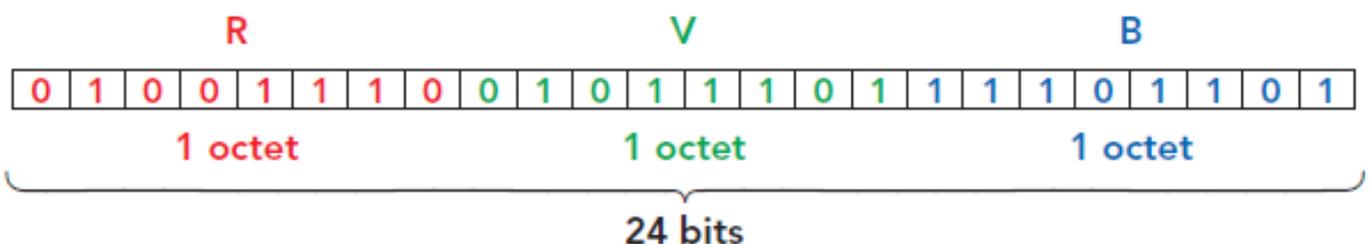
L'intensité lumineuse, grandeur analogique, est convertie par la cellule de l'appareil en un signal analogique sous forme de tension électrique. Ce signal est échantillonné, quantifié et codé sous forme d'un tableau de nombres binaires. C'est le **codage RVB**.

codage RVB 24 bits

Un pixel est constitué de 3 sous pixels chacun:

- associés à une couleur Rouge- Vert -Bleu
- codé par 8 bits = $8 \times 3 = 24$ bits pour un pixel d'où le terme

1 ss pixel = 8 bits = 1 octet => 1 pixel 3 x1 = 3 octets



Un octet (8 bits) peut prendre $2^8 = 256$ valeurs différentes, soit 256 nuances pour chaque sous-pixel.

L'intensité de chaque couleur est donc codée par un nombre entier compris entre 0 et 255.

Le codage d'une image numérique est représenté par un tableau de nombres. Chaque pixel de l'image est représenté par une case de trois nombres indiquant l'intensité de chaque couleur fondamentale.

255	255	255
0	0	0
0	0	0
255	255	255
0	0	0
0	0	0
255	255	255
0	0	0
0	0	0

Codage d'un carré rouge de 3x3 pixels.

Livre p.524 et doc 10 p. 525

Le codage en niveaux de gris est obtenu en indiquant une intensité égale pour chaque couleur fondamentale.

(même valeur pour chaque sous pixel)

Ex :	40	200
	40 gris foncé	200 gris clair
	40	200

Livre doc 11 p.525

3.3- Définition et taille d'une image numérique.

La **définition** d'une image correspond au **nombre de pixels qui la constitue.**

La **taille** d'une image, exprimée en octet, est la place occupée par le codage de tous les pixels qui la constituent.

Taille d'une image = nombre d'octet par pixel x définition (en octets)

Remarque : en réalité, la taille du fichier est légèrement supérieure car quelques octets supplémentaires sont utilisés pour coder les caractéristiques (nombre de lignes, de colonnes, nom du fichier, etc...)

Remarque 2 : en informatique, on utilise des préfixes multipliant par une puissance de 2.

Nom	kibiocet	mébiocet	gébiocet	tébiocet
Symbole	Kio	Mio	Gio	Tio
Valeur (en octet)	2^{10}	2^{20}	2^{30}	2^{40}