

# Numérisation d'un signal analogique

## 1. Signal analogique, signal numérique :

Un signal **analogique** est un ensemble **continu** d'informations. (exple : son, capteur de température...)

Un signal **numérique** est un ensemble **discret** (c'est-à-dire discontinu) d'informations.

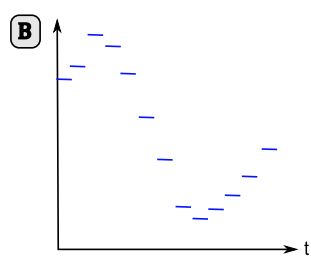
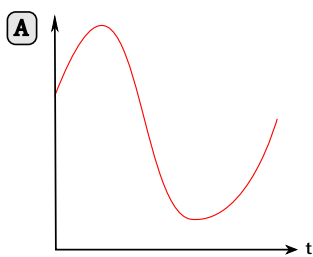
Pour transformer un signal analogique en signal numérique, il faut **discrétiser** les informations : on parle de **numérisation**.

Les ordinateurs ne traitant que des données binaires (0 ou 1), les informations sont ensuite traduites en binaire, c'est-à-dire en ensemble de 0 ou de 1.

La numérisation est faite par un **convertisseur analogique-numérique** (en abrégé : **CAN**):



Classer les signaux décrits ou représentés ci-dessous en « analogique » ou « numériques »



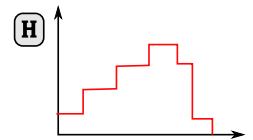
**C** Evolution de la température au cours d'une journée

**D** Affichage toutes les heures des températures

**E** Film diffusé en streaming sur internet

**F** Son émis par un haut-parleur relié à un lecteur CD (via un amplificateur)

**G** Film enregistré par un magnéscope sur cassette VHS



**La numérisation est d'autant meilleure que le signal numérique se rapproche du signal analogique initial.**

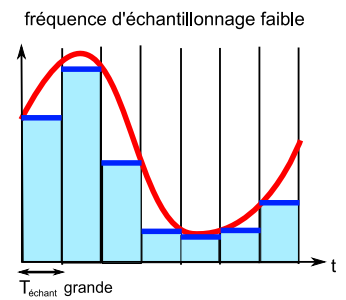
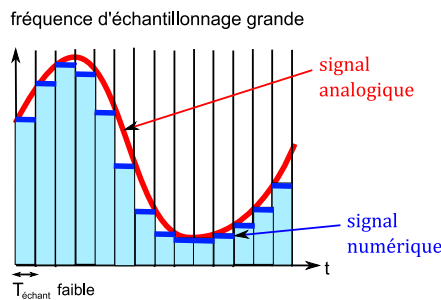
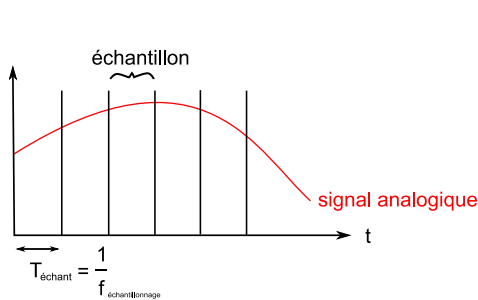
Pour cela, plusieurs paramètres ont leur importance, en particulier la fréquence d'échantillonnage et la quantification.

## 2. La fréquence d'échantillonnage :

### 2.1. Généralités :

Pour numériser un signal, il faut le découper en **échantillons** (« samples » en anglais) de durée égale  $T_e$ .

La **fréquence d'échantillonnage** correspond au nombre d'échantillons par seconde et s'exprime :  $F_e =$



Compléter en barrant un des deux adjectifs entre les crochets :

Plus la fréquence d'échantillonnage sera grande, plus la période d'échantillonnage sera [grande / petite], plus le nombre d'échantillons sera [grand / petit], plus le signal numérique sera [proche/éloigné] du signal analogique et donc [meilleure/moins bonne] sera la numérisation :

### 2.2. Approche expérimentale:

• Régler le GBF de la manière suivante : Le GBF délivre un signal électrique analogique (signal continu au sens mathématique du terme)

- Signal sinusoïdal
- Fréquence 500 Hz
- Amplitude mesurée au voltmètre (sur AC donc valeur efficace mesurée) : 2 V.

- Nous allons réaliser une acquisition avec Latispro : régler les paramètres de manière à réaliser une acquisition de durée totale 10 ms

☞ *1<sup>er</sup> cas : faible fréquence d'échantillonnage* : régler le nombre de points de manière à ce que la fréquence d'échantillonnage soit  $F_E = 1$  kHz. Observer et conclure.

☞ *2<sup>nd</sup>e cas : grande fréquence d'échantillonnage* : régler le nombre de points de manière à ce que la fréquence d'échantillonnage soit de 20 kHz. Observer et conclure quant au choix de la fréquence d'échantillonnage.

***Théorème de Shannon***

Pour numériser convenablement un signal, il faut que la fréquence d'échantillonnage soit au moins deux fois supérieure à la fréquence du signal à numériser.

1. En se remémorant le domaine de fréquence audible par l'Homme, expliquer pourquoi les sons des CD audio sont échantillonnés à 44,1 kHz.
2. La voix humaine est comprise dans une bande de fréquence comprise entre 100 et 3400 Hz. Quelle fréquence d'échantillonnage doit-on choisir pour la téléphonie ?

***2.3. Influence de la fréquence d'échantillonnage sur les hautes fréquences du signal analogique :***

- A l'aide du logiciel Audacity (voir notice ci-jointe)
  - enregistrer un son à l'aide du logiciel et d'un micro en 44kHz et 16 bits. L'enregistrer dans votre dossier personnel sous le nom : « **44.wav** »
  - Ré-échantillonner le son à l'aide du logiciel en 8kHz. L'enregistrer sous le nom : « **8.wav** »
  - Ecouter ces deux sons en passant par le poste de travail. Conclure
  - Ré-échantillonner le son « **8.wav** » en son 48 kHz. L'enregistrer en « **8vers48.wav** ». L'écouter. Le son est-il meilleur maintenant ?

☞ *Exemple d'application :*

Un son aigu a une fréquence de 10 kHz. Un son grave a une fréquence de 100 Hz

1. Calculer les périodes de ces deux sons.
2. Si la fréquence d'échantillonnage choisie pour numériser ces sons est de 1 kHz, calculer la durée des échantillons.
3. Conclure : Si l'on réduit la fréquence d'échantillonnage, quel type de son est alors mal numérisé ?

***2.4. Conclusion :*** Deux idées à retenir sur le choix de la fréquence d'échantillonnage :

- Il faut choisir  $F_E$  de manière .....
- Une fréquence  $F_E$  trop faible enlève l'information portant sur .....

Ordres de grandeurs :

Type de support de sons	$F_E$ choisie
CD audio	44,1 kHz
DVD	48 kHz
Téléphonie	8 kHz
Radio numérique	22,5 kHz

**3. La quantification :**

***3.1. Présentation de la quantification***

Lors de la numérisation, il faut également discrétiser les **valeurs de l'amplitude du signal**. La quantification consiste, pour chaque échantillon, à lui associer **une** valeur d'amplitude. Cette valeur de l'amplitude s'exprime en « bit » et l'action de transformer la valeur numérique de l'amplitude en valeur binaire s'appelle le **codage**.

**Qu'est-ce qu'un bit ?**

Un « bit » (de l'anglais *binary digit*) est un chiffre binaire (0 ou 1)

Avec 2 bits, on peut écrire : 00, 01, 10 et 11 soit 4 valeurs. ( $4 = 2^2$ )

Avec 3 bits, on peut écrire : 000, 001, 010, 011, 100, 101, 110, 111 soit 8 valeurs ( $8 = 2^3$ )

Avec 4 bits, on peut écrire  $2^4 = \dots$  valeurs

Avec n bits, on peut écrire ..... valeurs

**Conversion d'un nombre binaire en nombre décimal** : un exemple vaut mieux qu'un long discours :  
Que vaut l'octet (ensemble de 8 bits) 10110010 en décimal ?

	$2^7$ = 128	$2^6$ = 64	$2^5$ = 32	$2^4$ = 16	$2^3$ = 8	$2^2$ = 4	$2^1$ = 2	$2^0$ = 1
Octet =	1	0	1	1	0	0	1	0
somme de:	1 x 128	0 x 64	1 x 32	1 x 16	0 x 8	0 x 4	1 x 2	0 x 1

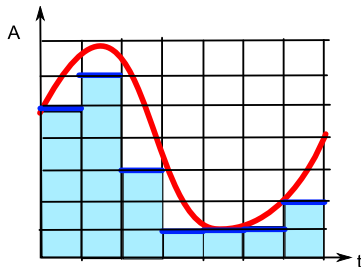
Ici 10110010 =  $1 \times 128 + 0 \times 64 + 1 \times 32 + 1 \times 16 + 0 \times 8 + 0 \times 4 + 1 \times 2 + 0 \times 1 = 178$

Exemples d'écriture binaire :

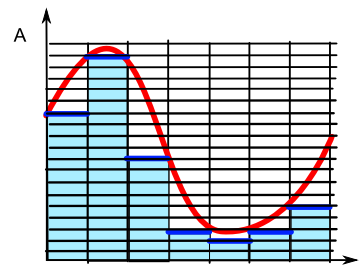
1. Ecrire la valeur décimale de l'octet 01001101.
2. Écrire, en binaire, la valeur maximale que peut prendre une grandeur codée sur 4 bits puis calculer sa valeur en décimal.
3. Ecrire l'octet correspondant au nombre 15. Même question pour le 16. Peut-on « coder » 16 sur 4 bits ?

**3.2. Exemples de quantifications**

1. Avec une quantification de 16 bit (soit une séquence binaire de 16 zéros ou un), de combien de valeurs dispose-t-on pour traduire l'amplitude du signal dans chaque échantillon ?
2. Même question avec une quantification de 8 bit (soit une séquence binaire de 8 zéros ou un).



faible quantification (peu de "bit") peu de valeurs pour traduire l'amplitude



grande quantification (beaucoup de "bit") beaucoup de valeurs pour traduire l'amplitude

3. Compléter en barrant un des deux adjectifs entre les crochets :

Lors de la quantification, plus le codage s'effectue avec un nombre important de bits, plus l'amplitude du signal numérique sera [*proche/éloignée*] de celle du signal analogique et donc [*meilleure/moins bonne*] sera la numérisation.

Ordres de grandeurs :

Type de support de sons	Quantification choisie
CD audio	16 bits
DVD	24 bits
Téléphonie	8 bits
Radio numérique	8 bits

**4. Exercice :**

- a. Calculer le nombre le « paliers » dont on dispose pour décrire l'amplitude en 24 bits. Idem en 4 bits.
- b. Lequel permettra de bien distinguer un son intense d'un son moins intense ?

### 3.3. Influence de la quantification sur la qualité d'un son :

Les fichiers se trouvent dans le dossier **Commun >> Physique\_Chimie >> TS >> Sons**

- Ouvrir Audacity et le fichier : **piano\_44kHz\_16bits.wav**  
Modifier la quantification du fichier audio en 8 bits. L'enregistrer en **piano\_44kHz\_8bits.wav**  
Fermer Audacity.
- A partir du poste de travail, ouvrir et écouter le fichier audio suivant : **piano\_44kHz\_16bits.wav**  
Ouvrir et écouter maintenant le fichier : **piano\_44kHz\_8bits.wav**  
Quelle grandeur, liée à la numérisation, ces deux fichiers ont-ils en commun.  
Que remarque-t-on lorsque l'on réduit la quantification ?

## 4. Choix des critères de numérisation :

Regarder l'animation : **Resume\_numerisation.swf** (issue de <http://xpose.avenir.asso.fr/viewxpose.php?site=8&subpage=/general/echantillonnage.html>)

En résumé, L'échantillonnage consiste à prélever périodiquement des échantillons d'un signal analogique. La quantification consiste à affecter une valeur numérique à chaque échantillon prélevé. Plus la fréquence d'échantillonnage et la quantification sont grandes, meilleure sera la numérisation.

🔗 Alors pourquoi se restreindre au niveau de ces valeurs ?

### Pistes de réflexion et informations :

Le nombre N d'octets (ensemble de 8 bits) nécessaires pour « décrire » numériquement une minute de son est :

$$N = F \times (Q/8) \times 60 \times n$$

avec F fréquence échantillonnage en Hz

Q : quantification en bits

n : nombre de voies (si le son est stéréo, n = 2 ; en mono : n = 1)

N s'exprime en octet

### Exemples :

1. Calculer la taille occupée, en octets puis Mo, d'une minute du son d'un CD audio (44,1 kHz et 16 bits, stéréo):
2. Même question pour le son d'un film encodé au format « ac3 » sur un DVD (48 kHz et 24 bits, stéréo):
3. Un réseau informatique domestique de mauvaise qualité possède un débit binaire (nombre d'octets pouvant circuler sur le réseau par seconde) de 230 ko/s. Le son du CD pourra-t-il être transmis sur ce réseau ? Et celui du DVD ?

### 🔗 Exercice bilan :

Une personne mal attentionnée télécharge sur un forum une chanson de 3 minutes au format mp3.

La chanson a été numérisée par un pirate à 16 kHz et 8 bits mono.

La personne, voulant une qualité « DVD » pour la chanson, modifie le fichier et le transforme en 48 kHz et 24 bits stéréo.

- a. Calculer le poids en octet de la chanson avant transformation.
- b. Même question après transformation.
- c. Décrire la sensation auditive que l'on éprouve en écoutant le fichier téléchargé avant transformation.
- d. La qualité de la chanson a-t-elle été améliorée par la transformation ?
- e. Comment la personne peut-elle améliorer la qualité du fichier téléchargé ?

### Remarque (intéressante mais) hors programme :

Traditionnellement, en informatique, 1 ko n'est pas 1000 octets mais 1024 octets (en fait  $2^{10}$ ). De même, 1 Mo n'est pas  $10^3$  ko mais 1024 ko !! Nous n'avons pas à nous en soucier ici et nous conserverons en tête que 1 kilo =  $10^3$  et 1 méga =  $10^6$ .