

CORRECTION ACTIVITE 26 : NUMERISATION D'UN SIGNAL ANALOGIQUE

Compétences

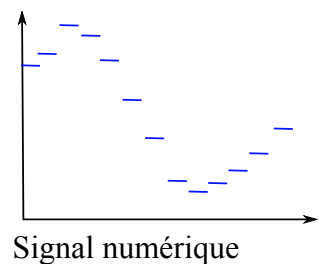
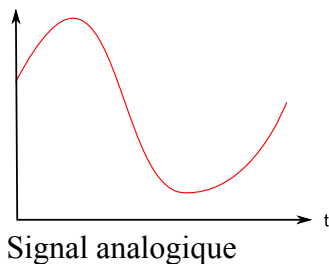
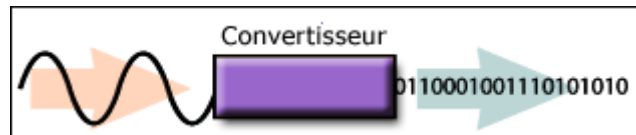
Reconnaître des signaux de nature analogique et des signaux de nature numérique

Mettre en oeuvre un protocole expérimental utilisant un échantillonneur-bloqueur et/ou un convertisseur analogique numérique (CAN) pour étudier l'influence des différents paramètres sur la numérisation d'un signal (d'origine sonore par exemple)

1. Signal analogique, signal numérique :

Un signal analogique est un ensemble **continu** d'informations. Les ordinateurs ne traitant que des données binaires (0 ou 1), pour numériser un signal, il faut **discrétiser** les informations : on parle de **numérisation**. Ces 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** :

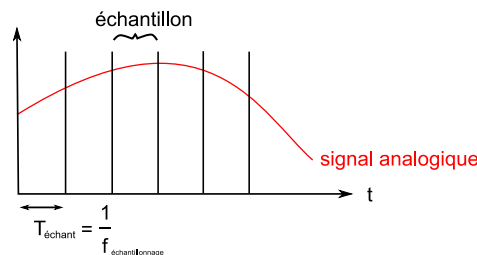


La numérisation est d'autant meilleure que le signal numérique se rapproche du signal analogique initial. Pour cela, plusieurs paramètres ont leur importance.

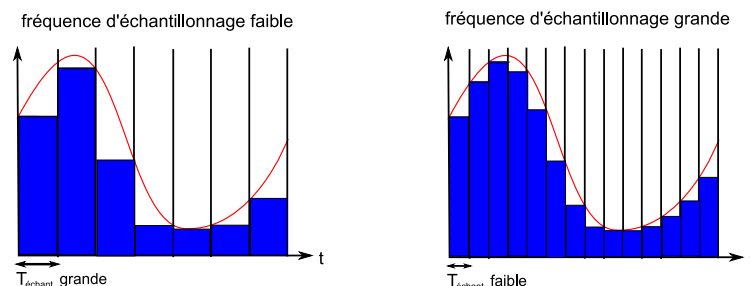
2. La fréquence d'échantillonnage :

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 : $F_e = 1/T_e$



Plus la fréquence d'échantillonnage sera grande, plus le nombre d'échantillons sera grand, plus le signal numérique « collera » au signal analogique et donc meilleure sera la numérisation :



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.

☞ Expliquer pourquoi les sons des CD sont échantillonnés à 44,1 kHz.

Le domaine de fréquences audibles par l'Homme est limité à 20 kHz. Il faut donc, dans un son, conserver les fréquences proches de 20 kHz si l'on veut le numériser correctement. D'où le choix de 44,1 kHz (supérieur au double du 20 kHz).

NB : le 44,1 (et non 44,0) vient d'un choix technologique datant de l'époque du stockage des sons sur la bande magnétique d'un magnéscope (oui j'ai bien écrit « scope » !)

☞ 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 ?

Fe doit être supérieure à $2 \times 3400 \text{ Hz} = 6800 \text{ Hz}$. C'est pourquoi la fréquence échantillonnage de la téléphonie est de 8000 Hz.

2.2. 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

Le son en 8 kHz est moins riche et surtout possède moins d'aigu. Ceci est en accord avec ce qui a été dit en cours : baisser la fréquence d'échantillonnage élimine les informations sur les hautes fréquences.

- Ré-échantillonner le son « **8.wav** » en son 48 kHz. L'enregistrer en « **8vers48.wav** ». L'écouter. Le son est-il meilleur maintenant ?

Le son en 8 kHz ré-échantillonné en 44 kHz reste le même. Augmenter le nombre d'échantillons ne rajoute pas les informations manquantes entre un 8 kHz et un 48 kHz. Il est donc inutile de réaliser ce ré-échantillonnage. (on a juste un son « mauvais » qui nécessite beaucoup d'octets pour être décrit : absurde !)

☞ Exercice :

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

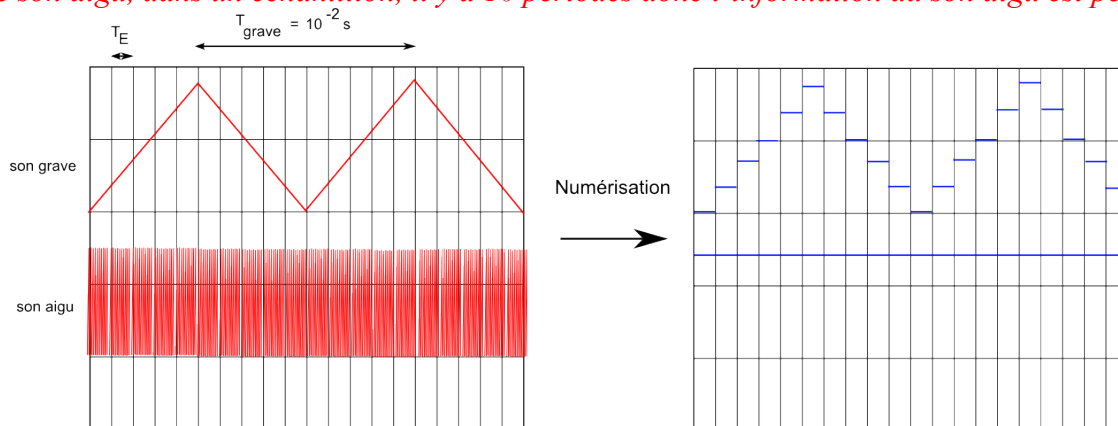
1. Calculer les périodes de ces deux sons.
2. Si la fréquence d'échantillonnage choisie pour numériser ces sons est de 1 kHz, calculer la durée des échantillons.
3. Conclure : Si l'on réduit la fréquence d'échantillonnage, quel type de son est alors mal numérisé ?

1. $T_{\text{aigu}} = 10^{-4} \text{ s}$ et $T_{\text{grave}} = 10^{-2} \text{ s}$.

2. Si $F_E = 1 \text{ kHz}$ alors $T_E = 10^{-3} \text{ s}$

3. Pour le son grave, chaque période est décomposée en 10 échantillons

Pour le son aigu, dans un échantillon, il y a 10 périodes donc l'information du son aigu est perdue.



2.3. Conclusion :

☞ Deux idées à retenir sur le choix de la fréquence d'échantillonnage :

- Il faut choisir F_E de manière à ce que $F_E > F_{\text{maxi du signal analogique}}$ (en fait $F_E > 2F_{\text{maxi du signal analogique}}$)
- Une fréquence F_E trop faible enlève l'information portant sur les hautes fréquences du signal analogique.

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**.

Le nombre de valeurs dont on dispose pour définir l'amplitude s'appelle la quantification. Elle s'exprime en « bit ».

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 $2^4 = 16$ valeurs

Avec n bits, on peut écrire 2^n 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 = 1x128 + 0x64 + 1x32 + 1x16 + 0x8 + 0x4 + 1x2 + 0x1 = 178

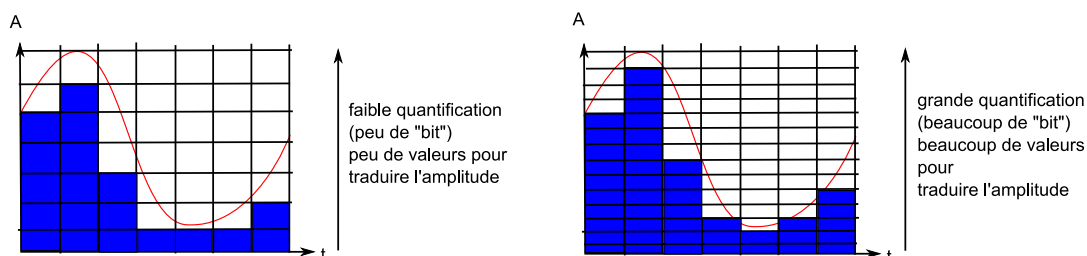
Exemples de quantifications :

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 ?

On dispose de $2^{16} = 65536$ valeurs pour traduire l'amplitude du signal dans chaque échantillon.

Même question avec une quantification de 8 bit (soit une séquence binaire de 8 zéros ou un).

On dispose de $2^8 = 256$ valeurs seulement pour traduire l'amplitude du signal dans chaque échantillon.



Plus la quantification est grande, plus l'amplitude du signal numérique sera proche de celle du signal analogique.

Ordres de grandeurs :

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

☞ Exercice :

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

1. En 24 bits : $2^{24} = 16\,777\,216$ possibilités et en 4 bits : $2^4 = 16$ possibilités
2. Avec 16 possibilités, on ne pourra pas distinguer deux sons d'intensité sonore très proche.

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

- ☞ 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**
Que remarque-t-on lorsque l'on réduit la quantification ?

En réduisant la quantification de 16 bits à 8 bits, on perd de l'information sur l'amplitude et l'ordinateur ce qui a pour effet de créer du bruit (un souffle).

4. Choix des critères de numérisation :

En résumé, plus la fréquence d'échantillonnage et la quantification sont grandes, meilleure sera la numérisation.

☞ *Alors pourquoi se restreindre au niveau de ces valeurs ?*

La limite vient du nombre d'octets qui vont être nécessaires pour numériser le signal car ce nombre sera écrit sur un support de stockage (disque dur, clé USB, DVD...). La capacité de stockage de ces supports n'est pas illimitée.

De plus, il faut penser qu'il faut du temps pour écrire toutes ces données sur un support (durée qui dépend de beaucoup de paramètres : type de support, version du port USB etc....) Les informaticiens parlent de « flux » ou « débit binaire » (en ko/s ou Mo/s). Cette vitesse d'écriture ne peut pas être infinie !

Piste de réflexion et information :

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 :

- son d'un CD audio (44,1 kHz et 16 bits, stéréo):
 $N = 44\,100 \times (16/8) \times 60 \times 2 = 10\,584\,000$ octets
On divise par 1024 : $N = 10\,335$ ko
On divise par 1024 : $N = 10,9$ Mo
- son d'un film sur DVD(48 kHz et 24 bits, stéréo):
 $N = 48\,000 \times (24/8) \times 60 \times 2 = 17\,280\,000$ octets = 16,5 Mo

✂ 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.

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

1. $N = 16\,000 \times (8/8) \times 3 \times 60 \text{ (3 minutes)} \times 1 \text{ (mono)} = 2,88 \cdot 10^6 \text{ o} = 2,8 \text{ Mo}$

2. $N = 48\,000 \times (24/8) \times 3 \times 60 \times 2 \text{ (stéréo)} = 51,8 \cdot 10^6 \text{ o} = 49,4 \text{ Mo}$

3. F_E faible : des aigus semblent absents de la chanson

Quantification faible : on entend beaucoup de bruit et peu la distinction son intense/peu intense.

4. *NON, les fréquences aigues absentes ne peuvent pas être « inventées » et le bruit reste présent. Il est juste codé sur plus de bits.*

5. *Elle ne peut pas !*