

# Chapitre 3 Echantillonnage des signaux

## 3.1 Introduction

Les signaux primaires porteurs d'information sont pratiquement toujours de type analogique (temps et amplitude continus). Un ordinateur, ou tout autre système électronique numérique est un dispositif qui traite des données (nombres). Si on veut traiter un signal par voie numérique, il faut le représenter par une suite de valeurs numériques. Donc une opération préliminaire de conversion analogique numérique est nécessaire. La conversion analogique numérique est la succession de trois effets sur le signal analogique de départ :

- ✓ *l'échantillonnage* pour rendre le signal discret,
- ✓ *-la quantification* pour associer à chaque échantillon une valeur,
- ✓ *le codage* pour associer un code à chaque valeur.

## 3.2 Echantillonnage

### 3.2.1 Définition

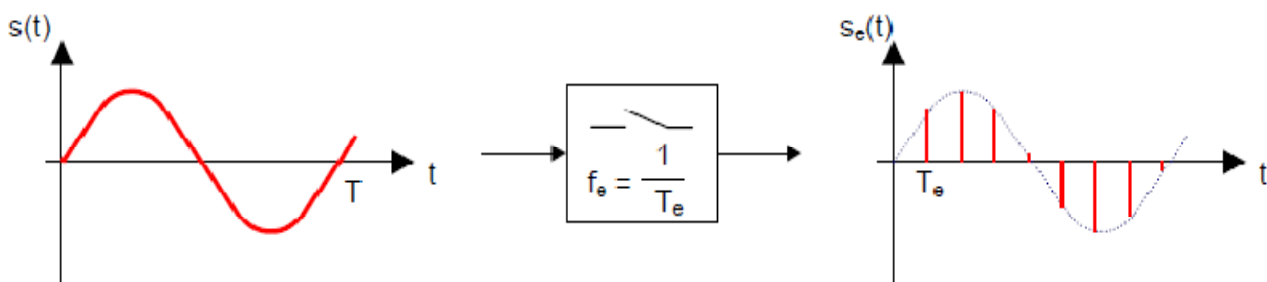
L'échantillonnage consiste à prélever à des instants précis, le plus souvent équidistants, les valeurs instantanées d'un signal. Le signal analogique  $s(t)$ , continu dans le temps, est alors représenté par un ensemble de valeurs discrètes :

$$s_e(t) = s(n.T_e)$$

Avec  $n$  : entier

$T_e$  : période d'échantillonnage.

Cette opération est réalisée par un échantillonneur souvent symbolisé par un interrupteur.



### 3.2.2 Echantillonnage idéal

L'échantillonnage idéal est modélisé par la multiplication du signal continu  $s(t)$  et d'un peigne de Dirac de période  $T_e$ .

$$s_e(t) = s(t) \cdot \delta_{T_e}(t) = s(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT_e) = s(nT_e) \sum_{n=-\infty}^{+\infty} \delta(t - nT_e)$$

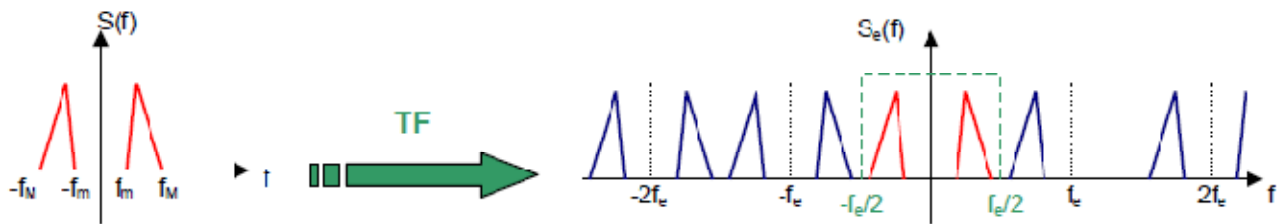
Le spectre du signal échantillonné est donc le suivant :

$$S_e(f) = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} S(f) * \delta(f - nf_e)$$

$$S_e(f) = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} S(f - nf_e)$$

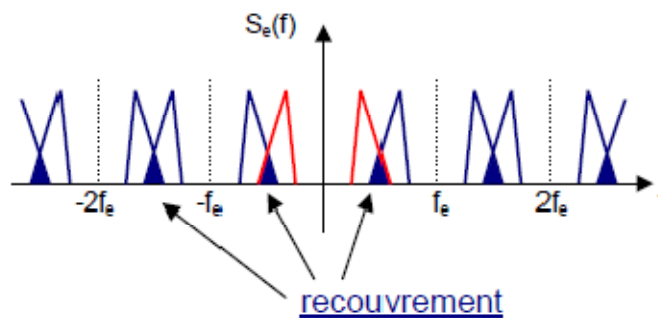
On obtient donc un spectre infini qui provient de la périodisation du spectre du signal d'origine autour des multiples de la fréquence d'échantillonnage.

**Exemple :**



**Remarques :**

- On voit sur le spectre du signal échantillonné qu'il est possible de restituer le signal original par un simple filtrage passe-bas.
- Si  $f_M$ , la fréquence maximale du spectre du signal à échantillonner, est supérieure à  $f_e/2$ , la restitution du signal original sera impossible car il va apparaître un **recouvrement spectrale** lors de l'échantillonnage. On dit qu'on est en **sous-échantillonnage**.



Le théorème de **Shannon** montre que la reconstitution correcte d'un signal nécessite que la fréquence d'échantillonnage  $f_e$  soit au moins deux fois plus grande que la plus grande des fréquences  $f_M$  du spectre du signal :

$$f_e \geq 2 \cdot f_M$$

- Lorsqu'il y a recouvrement spectral, nous avons vu qu'il était impossible de reconstruire correctement le signal. Pourtant dans la plupart des situations, le spectre du signal à échantillonner s'étale sur tout le domaine des fréquences (tout en diminuant du côté des hautes fréquences), mais il

n'existe pas une fréquence  $f_{max}$  au-delà de laquelle l'énergie est nulle. Il y a donc un problème pour choisir la fréquence d'échantillonnage. On se fixe donc en pratique une  $f_{max}$  à partir de laquelle on estime la représentation de notre signal satisfaisante pour les applications que l'on veut en faire. Puis on effectue un filtrage passe-bas (à  $f_{max}$ ) avant l'échantillonnage afin de remédier aux repliements de spectre. On appelle ce filtre un filtre **anti-repliement**.

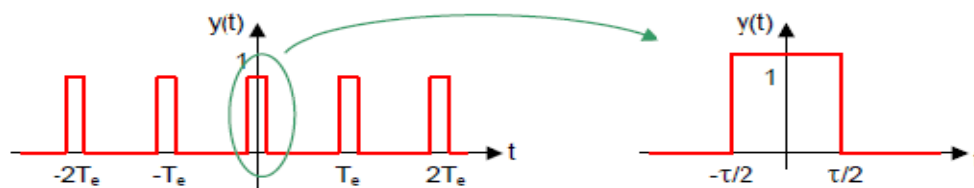
**Exemple :**

C'est par exemple le cas de la parole. Le spectre des sons audibles s'étend jusqu'à environ 20kHz. Dans le cas des CD audio, le signal est échantillonné à 44.1 kHz alors que dans le cas du téléphone numérique le signal est échantillonné à 8 kHz seulement. En effet, en téléphonie, on estime que le message est compréhensible pourvu que les composantes basses fréquences soient transmises correctement alors que l'on veut conserver tous les harmoniques pour avoir un son de qualité en audio. On limite ainsi le spectre à 22.05 kHz pour un CD audio et à 4 kHz pour la téléphonie (3.4kHz en pratique).

- Si  $f_e > 2f_M$ , il y a **sur-échantillonnage**. Alors les motifs successifs obtenus par périodisation du spectre sont disjoints et éloignés l'un de l'autre. Le filtrage passe-bas pour la récupération du signal est facilité. Plus on prendra d'échantillons par période, plus le signal sera facile à reconstruire.

### 3.2.3 Echantillonnage réel

En pratique, l'échantillonnage s'effectue en commandant un interrupteur par un train d'impulsions étroites. Il est donc impossible d'obtenir des échantillons de durée quasiment nulle. La modélisation de l'échantillonnage par un peigne de Dirac est donc erronée. En fait, chaque impulsion va avoir une durée très courte  $\tau$ . L'échantillonnage peut donc être modélisé par la multiplication du signal par une suite de fonction rectangle (ou porte) de largeur  $\tau$ .



L'expression du signal d'échantillonnage devient donc :

$$y(t) = \sum_{n=-\infty}^{+\infty} \text{rect}_{\tau}(t - nT_e) = \text{rect}_{\tau}(t) * \sum_{n=-\infty}^{+\infty} \delta(t - nT_e)$$

Et par conséquent, sa transformée de Fourier est égale à :

$$Y(f) = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} \tau \cdot \text{sinc}(\pi n f_e \tau) \delta(f - n f_e)$$

Comme l'expression du signal échantillonné est :

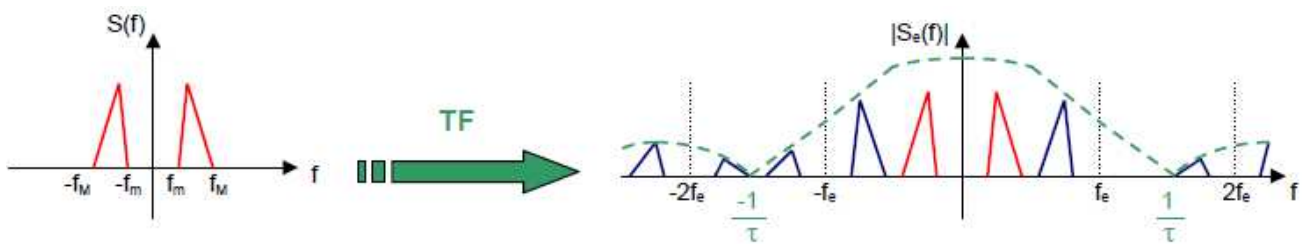
$$s_e(t) = s(t) \cdot y(t)$$

Sa transformée de Fourier devient :

$$S_e(f) = S(f) * Y(f) = S(f) * \frac{\tau}{T_e} \sum_{n=-\infty}^{+\infty} \text{sinc}(\pi n f_e \tau) \delta(f - n f_e)$$

$$S_e(f) = \frac{\tau}{T_e} \sum_{n=-\infty}^{+\infty} \text{sinc}(\pi n f_e \tau) S(f - n f_e)$$

On retrouve la même allure de spectre modulé en amplitude par une fonction en sinus cardinal.



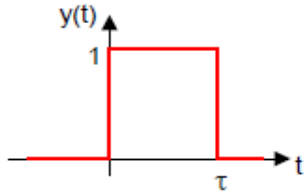
### Remarques :

- Pour se rapprocher d'un échantillonnage idéal et qu'ainsi le signal soit facilement reconstituable, il faut que  $\tau$  soit le plus petit possible.
- Dans le cas où  $\tau$  est du même ordre de grandeur que  $T_e$ , il faudra que  $f_e \gg 2f_m$ .

### 3.2.4 Echantillonnage-blocage

En pratique, on n'échantillonne pas un signal pour le reconstruire juste après. L'échantillonnage est utilisé pour prélever le signal à des instants multiples de  $T_e$  et ensuite convertir les échantillons sous forme d'un code binaire (8, 12, 16 bits, ...). Cette conversion est effectuée par l'intermédiaire d'un convertisseur analogique-numérique (CAN). Cette conversion n'est pas instantanée. Si le signal à convertir varie trop rapidement, il est nécessaire de procéder au blocage du signal pour avoir une conversion sans erreur. On utilise donc un **échantillonneur-bloqueur** qui mémorise la tension à convertir et la maintient constante pendant toute la durée de conversion.

L'effet de blocage peut être modélisé par une fonction porte décalée de  $\tau/2$  :



$$y(t) = \sum_{n=-\infty}^{+\infty} \text{rect}_{\tau} \left( t - \frac{\tau}{2} - nT_e \right) = \text{rect}_{\tau} \left( t - \frac{\tau}{2} \right) * \sum_{n=-\infty}^{+\infty} \delta(t - nT_e)$$

$$y(t) = \sum_{n=-\infty}^{+\infty} \text{rect}_{\tau} \left( t - \frac{\tau}{2} - nT_e \right) = \text{rect}_{\tau} \left( t - \frac{\tau}{2} \right) * \sum_{n=-\infty}^{+\infty} \delta(t - nT_e)$$

L'échantillonnage-blocage consiste donc à la multiplication du signal par  $y(t)$ . La transformée de Fourier du signal échantillonné est donc :

$$S_e(f) = \frac{\tau}{T_e} \sum_{n=-\infty}^{+\infty} \text{sinc}(\pi n f_e \tau) S(f - n f_e) e^{-j\pi n f_e \tau}$$

### Remarques :

Le spectre est identique au précédent. Le terme en  $e^{-j\pi n f_e \tau}$  traduit un déphasage entre le signal initial et le signal échantillonné. En principe, on maintient la valeur de l'échantillon sur toute la période d'échantillonnage donc  $\tau = T_e$ . Ainsi, pour  $f = f_e$ , on a un déphasage de  $-\pi$ .

## 3.3 Quantification

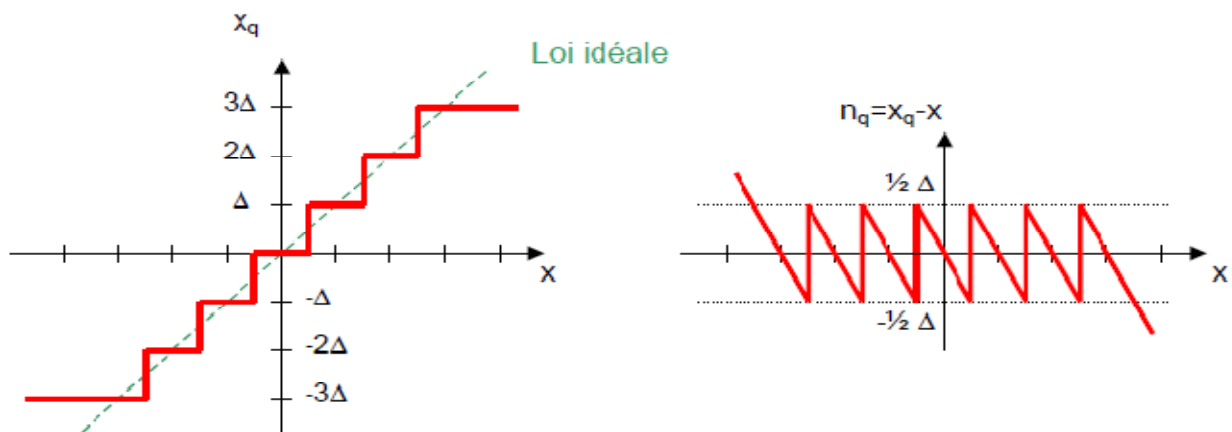
### 3.3.1 Définition

La quantification consiste à associer à une valeur réelle  $x$  quelconque, une autre valeur  $x_q$  appartenant à un ensemble fini de valeurs et ce suivant une certaine loi : arrondi supérieur, arrondi le plus proche, etc...

L'écart entre chaque valeur  $x_q$  est appelé ***pas de quantification***. Le fait d'arrondir la valeur de départ entraîne forcément une ***erreur de quantification*** que l'on appelle le ***bruit de quantification***.

### 3.3.2 Quantification uniforme

La loi de quantification uniforme utilise un pas de quantification ( $\Delta$ ) constant entre chaque valeur  $x_q$ .



Le bruit de quantification  $n_q$  est dans ce cas un signal aléatoire. Ces caractéristiques sont donc définies par ses propriétés statistiques. On peut alors démontrer que la puissance du bruit de quantification (si sa densité de probabilité est uniforme) est égale à :

$$P_{n_q} = \frac{\Delta^2}{12}$$

Le rapport signal sur bruit dû à la quantification est donc égale à :

$$\left(\frac{S}{N}\right)_{dB} = 10 \cdot \log \frac{P_s}{P_{n_q}}$$

La puissance du signal à quantifier est égale à sa valeur efficace au carré (*voir remarque*) :

$$\left(\frac{S}{N}\right)_{dB} = 10 \cdot \log \left[ 12 \left( \frac{V_{seff}}{\Delta} \right)^2 \right]$$

Si l'on décompose la plage de variation  $V_{PE}$  du signal à quantifier en  $2^n$  intervalles de largeur  $\Delta$  (avec  $n$  le nombre de bits utilisés pour coder le signal quantifié).

$$\text{Alors : } V_{PE} = 2^n \times \Delta \text{ et } \Delta = \frac{V_{PE}}{2^n}$$

$$\text{Ainsi : } \left(\frac{S}{N}\right)_{dB} = 10 \cdot \log(12) + 20 \cdot \log \left( 2^n \cdot \frac{V_{seff}}{\Delta} \right) = 10 \cdot \log(12) + 20 \cdot \log 2^n + 20 \cdot \log \left( \frac{V_{seff}}{V_{PE}} \right)$$

$$\left(\frac{S}{N}\right)_{dB} \approx 6.02 n + 10.8 + 20 \cdot \log \left( \frac{V_{seff}}{V_{PE}} \right)$$

Ainsi, dans le cas d'un convertisseur analogique-numérique, chaque fois que l'on rajoutera un bit dans le résultat de conversion, on améliorera le rapport signal sur bruit dû à la quantification d'environ 6dB.

### **Remarque :**

En traitement du signal, on considère la puissance d'un signal aux bornes d'une résistance de  $1\Omega$ . La puissance est donc égale au carré de la valeur efficace.

### **Exemple :**

Si l'on veut numériser une sinusoïde et que l'on fixe  $V_{PE} = 2 \cdot V_{max}$

Dans ce cas,

## **3.4 Codage**

Le codage consiste à associer à un ensemble de valeurs discrètes un code composé d'éléments binaires. Les codes les plus connus : code binaire naturel, code binaire décalé, code complément à 2, code DCB, code Gray.