

Analogique vs Numérique

Distinguons deux mode de représentation des grandeurs : l'analogique et le numérique.

Le codage des informations et leur transport sous forme numérique est une technique assez récente. Jusque dans les années 70, avant les calculatrices, les CD ou la microinformatique, les informations étaient surtout codées et transportées sous forme analogique.

Les systèmes analogiques utilisent des grandeurs continues, comme une tension électrique, pour représenter les valeurs à traiter. On parle d'analogique car il y a « analogie » entre la donnée et sa représentation.

- Les systèmes analogiques paraissent maintenant plus rudimentaires et manquent de précision (approximation, tolérance, etc.) dès que le support qui sert à son écriture ou à sa transmission est de qualité insuffisante.
- A chaque manipulation des signaux analogique les erreurs d'imprécision viennent dégrader l'information (exemples : photocopies, enregistrements sur K7, transmissions radio etc.)

Les systèmes numériques ne manipulent que des nombres. C'est à la fois plus simple et plus compliqué !

- Plus simple parce que :
 - la mémorisation de données numériques est plus facile.
 - elles peuvent être recopiées avec exactitude.
L'information chiffrée reste intacte.
 - on dispose de nombreuses méthodes arithmétiques et logiques pour traiter ces codes : les vérifier, les crypter et les compresser.
- Plus compliqué, parce que :
 - pour avoir ces signaux sous forme digitale il faut d'abord les numériser
 - et à l'inverse il faut souvent les reconvertir en analogique quand la reproduction sous cette forme est nécessaire.

Exemple : le son.

Le son est un phénomène de vibration de l'air qui nous parvient à l'oreille par des variations de pression sur nos tympanes. Ces variations sont continues et donc de nature analogique. Graphiquement on pourrait représenter le son comme une courbe qui oscille autour d'une valeur moyenne en variant en amplitude et en fréquence.

Ces fluctuations peuvent être enregistrées de manière analogique. C'est ce que l'on faisait en retranscrivant ces oscillations dans les sillons des disques vinyles

A l'heure du son numérique ; les signaux analogiques captés par le micro sont numérisés (digitalisés) en échantillonnant à intervalles réguliers les mesures de ces fluctuations.

La numérisation, comme toutes retranscriptions d'ailleurs, n'est jamais réalisée avec une précision absolue. La figure 1 illustre l'échantillonnage d'un son avec des valeurs pouvant aller de -7 à +7 ce qui pourrait être réalisé avec un code de 4 bits. On voit sur ce graphique que les valeurs des échantillons sont très approximatives et la restitution de la courbe à partir des échantillons risque d'être bien moins nuancée que la courbe initiale.

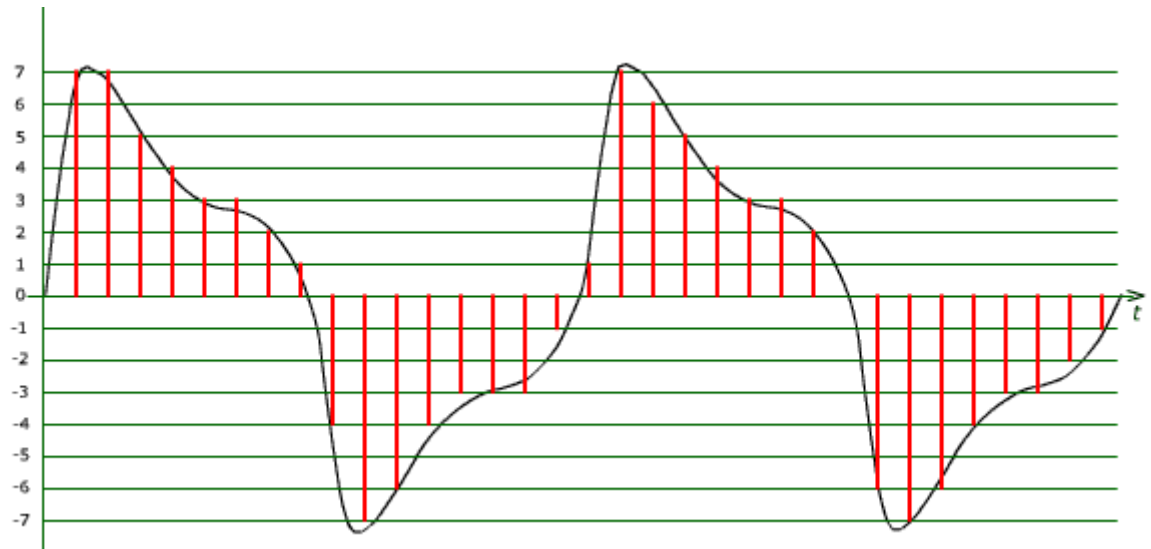


Figure 1 Echantillonnage avec échantillons codés sur 4 bits

Pour améliorer la qualité de cette numérisation il est possible d'agir sur deux facteurs :

- Le taux d'échantillonnage
- La précision des mesures des échantillons

Le taux ou fréquence d'échantillonnage est fonction de la qualité attendue. L'oreille humaine est capable dans les meilleures conditions d'entendre les fréquences allant de 20 Hz à 20.000 Hz. On comprendra que pour avoir un signal de haute fidélité HiFi on devra avoir une fréquence d'échantillonnage qui soit d'au moins égale au double de la fréquence à reproduire la plus haute. Dans ce cas $2 \times 20\text{kHz} = 40 \text{ kHz}$. Ainsi les CD sont enregistrés avec une fréquence de 44,1 kHz.

La numérisation représentée par la figure 2 est meilleure que la précédente car les échantillons y sont deux fois plus nombreux.

Les valeurs échantillonnées sont de plus codées de manière beaucoup plus fine puisqu'elles sont reprises avec des valeurs discrètes qui vont de -15 à +15 au lieu de -7 à +7. Il faudrait dans ce cas coder ces nombres sur 5 bits.

Dans le cas des CD audio, les échantillons sont codés sur 16 bits et peuvent donc alors prendre $2^{16} = 65.536$ valeurs distinctes.

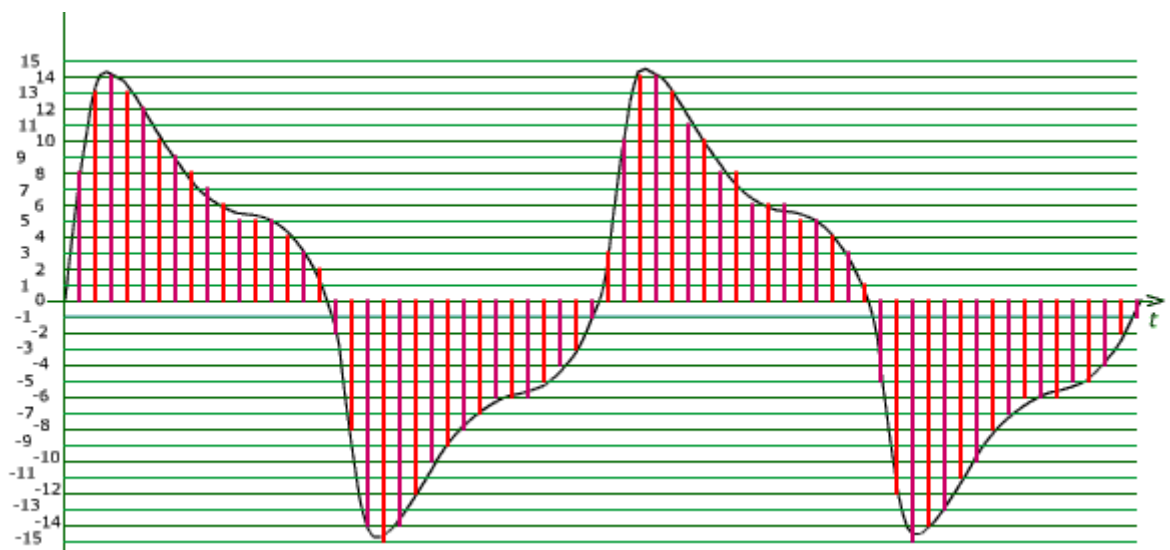


Figure 2 Echantillonnage à fréquence double avec échantillons codés sur 5 bits