

Bilan de séance



Ce qu'il faut retenir

Numériser consiste à convertir un signal analogique en un signal numérique.

1 – Échantillonnage : On relève à intervalles de temps régulier la valeur du signal

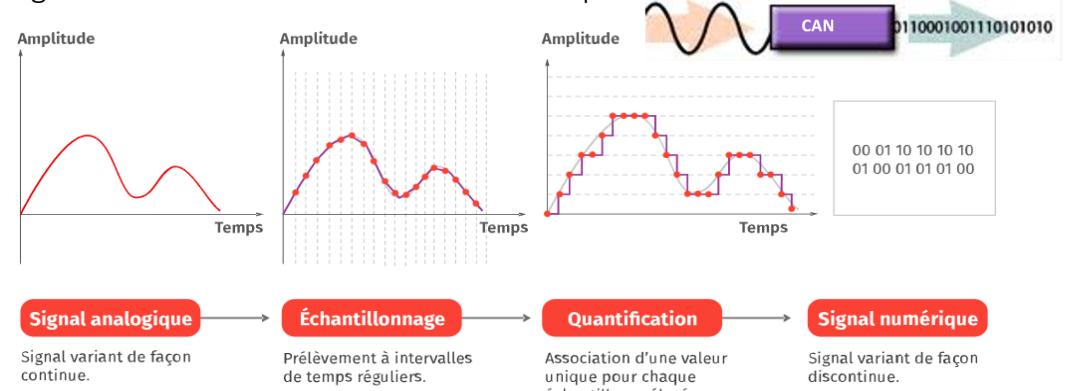
2 – Quantification :
On limite le nombre de valeurs possibles et on affecte aux valeurs relevées lors de l'échantillonnage la valeur la plus proche

3 – Codage en binaire :
On convertit les valeurs en binaire.

Pour aller plus loin :
En base 10 nous avons le chiffre des unités (10^0), le chiffre des dizaines (10^1) le chiffre des centaines (10^2)... La valeur de chaque chiffre va de 0 à 9. En base 2 (binaire) nous avons le chiffre des $2^0=1$, le chiffre des 2^1 , le chiffre des $2^2=4$, le chiffre des $2^3=8$ Un chiffre peut avoir pour valeur 0 ou 1. Pour convertir la première valeur décimale (ici 10) je considère que j'ai 10 billes à ranger dans les cases. Soit je peux remplir une case alors je mets 1 sinon je mets 0. Avec 10 billes je ne peux pas remplir la case qui a une contenance de 128 billes, je mets donc 0. Idem pour les cases de contenance 64, 32,16. Par contre je peux remplir la case de 8 billes. Il me reste 2 billes. Je peux remplir celle de 2 billes.
 $10=1x8+1x2$;
 $10=00001010$

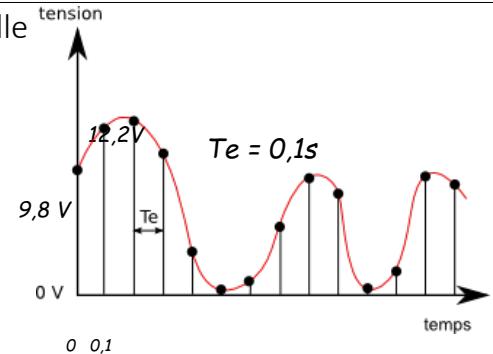
4- Numérisation d'un signal

Un signal analogique doit être converti en numérique pour être traité par le microcontrôleur (interface programmable) : C'est la numérisation du signal. Cette numérisation se fait en 3 étapes :

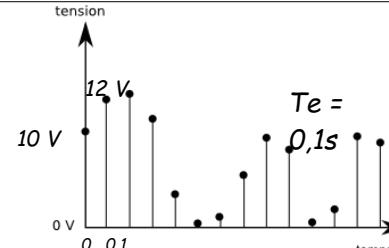


1- Échantillonnage : On relève à intervalle de temps régulier T_e la valeur du signal analogique.

Remarque : Il est primordial de bien choisir la fréquence d'échantillonnage pour ne pas trop perdre d'informations. (Shannon : fréquence échantillonnage > 2 fois la fréquence_{max} signal analogique)



2 – Quantification : On limite le nombre de valeurs possibles sur l'axe des ordonnées (en fonction du nombre de bits) et on attribue la valeur la plus proche à chaque valeur relevée lors de l'échantillonnage.



3 – Codage : On convertit en binaire (0 et 1) chaque valeur décimale relevée aux intervalles de temps réguliers T_e

temps	Valeur décimale	Valeur binaire							
		2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0 s	10	0	0	0	0	1	0	1	0
0,1 s	12	0	0	0	0	1	1	0	0

Remarque : J'ai codé le signal ci-dessus sur 8 bits ce qui nous donne 255 valeurs possibles sur l'axe des ordonnées. Plus le nombre de bits utilisés pour le codage et plus la fréquence d'échantillonnage sont importants plus le signal numérique de sortie sera fidèle au signal analogique d'entrée.



Exercices

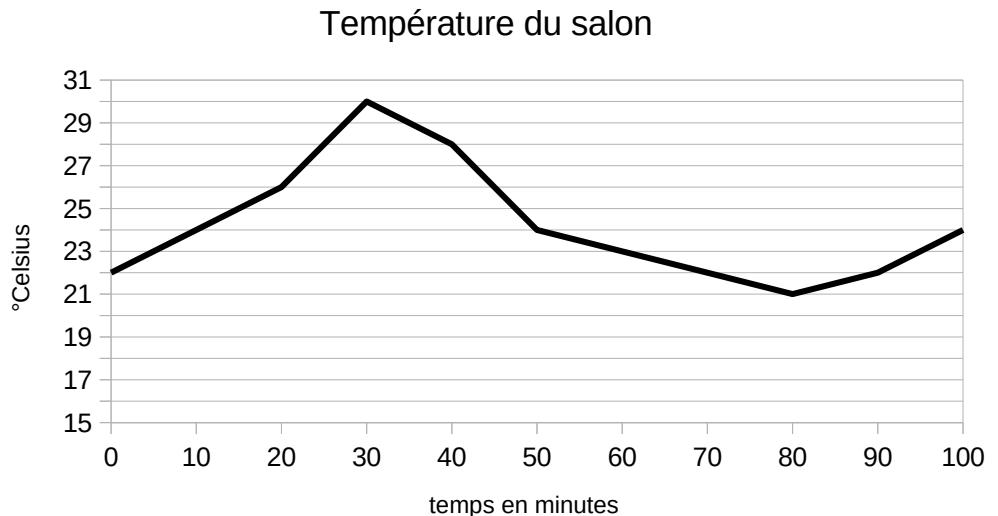
Un capteur de température mesure l'information ci-contre :

1- De quel type est cette information ?

.....

2- Quel type de signal permet de la transporter

.....



3- Afin de réduire les pertes d'information nous décidons de la numériser avec une fréquence d'échantillonnage de 10 minutes.

3.1- Reformuler avec vos propres mots l'expression « fréquence d'échantillonnage de 10 minutes »

.....

.....

3.2- Représenter sur le diagramme ci-dessus le signal échantillonné et la quantification. (voir page précédente)

3.3- Quelle est la valeur de la température à $t=10$ minutes ?

3.4- Quelle est la valeur de la température à $t=20$ minutes ?

3.5- Convertir en binaire sur un octet (1 pour chaque valeur) les valeurs décimales de ces deux températures.

temps	Température en valeur décimale	Valeur binaire							
		2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
		128	64	32	16	8	4	2	1
10 minutes									
20 minutes									

3.6- Représenter sur l'axe ci-dessous le signal numérique permettant de transporter l'information de température à $t=10$ minutes et $t=20$ minutes.

