

Numérisation d'une tension

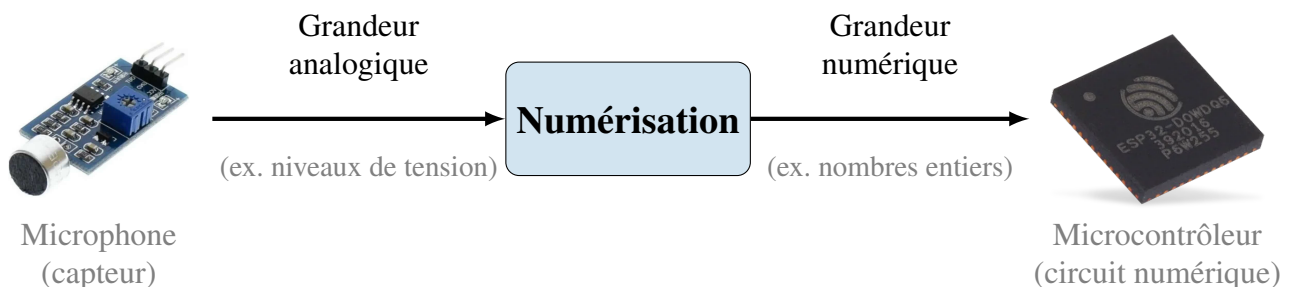
I - Introduction

De nos jours, les **systèmes électroniques** sont essentiellement conçus à base de **circuits numériques** comme des microcontrôleurs (ex. Arduino, ESP32, STM32, ...), des circuits à logiques programmables (FPGA), ...

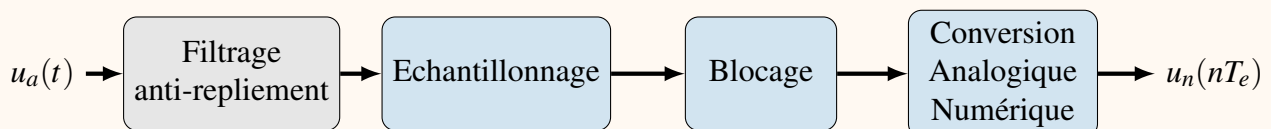


En fonction du type d'application, un système électronique peut avoir besoin d'accéder des **grandeurs physiques** (ex. température, pression, son, ...) par l'intermédiaire d'une **tension analogique** (ex. sortie d'un capteur).

La **numérisation d'une tension analogique** reste donc une opération encore nécessaire pour les systèmes électroniques modernes.



La **chaîne de numérisation** d'un signal analogique est composée de **plusieurs fonctions essentielles** telles que le filtrage anti-repliement, l'**échantillonnage**, le **blocage** et **conversion analogique numérique**.



A la fin de la numérisation, on obtient un **signal numérique** exploitable, par exemple, par un circuit numérique :

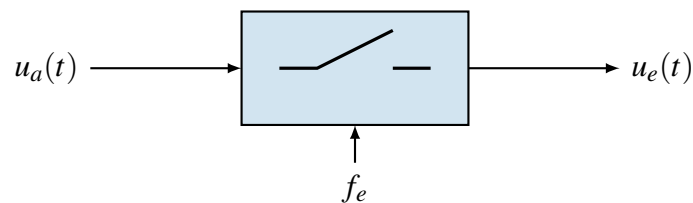
- pour un **traitement en temps réel** (ex. filtrage numérique, ...);
- pour un **stockage en mémoire** (ex. enregistrement musical, oscilloscope, ...)

II - Échantillonnage d'un signal analogique

1) Principe de l'échantillonnage

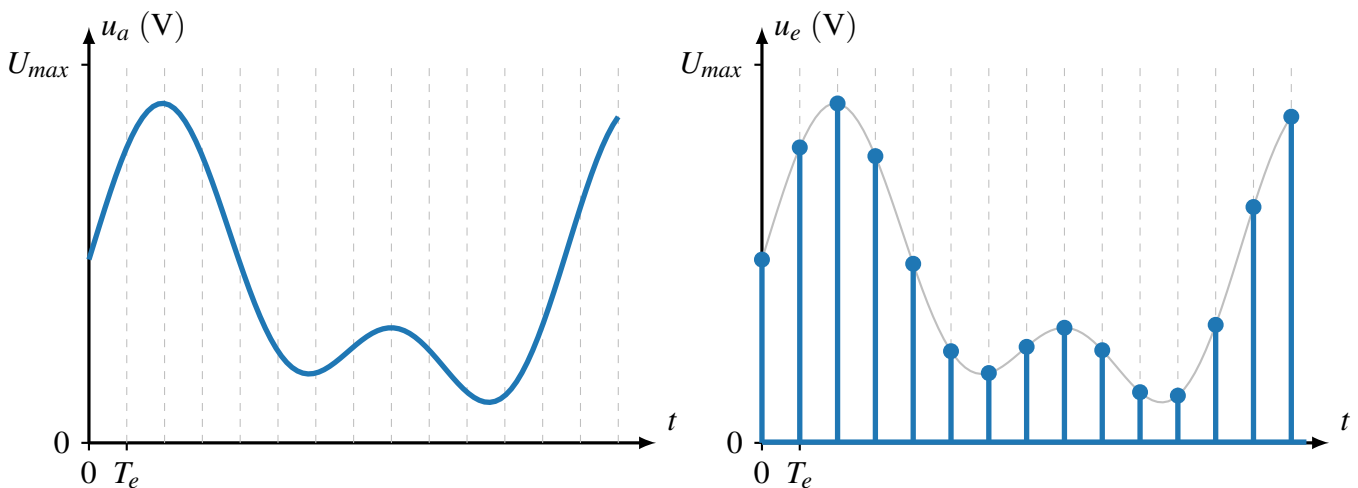
L'échantillonnage est une opération que consiste à **prélever des valeurs** (échantillons) d'un signal analogique à une **fréquence d'échantillonnage** :

$$f_e = \frac{1}{T_e} \quad (\text{Hz})$$

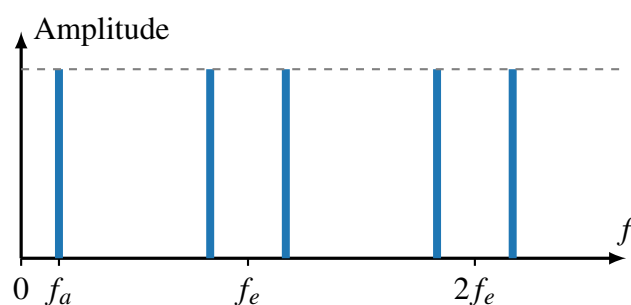
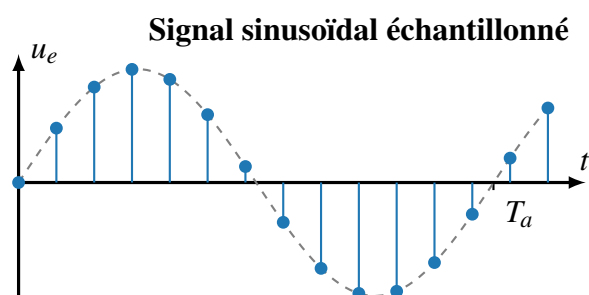
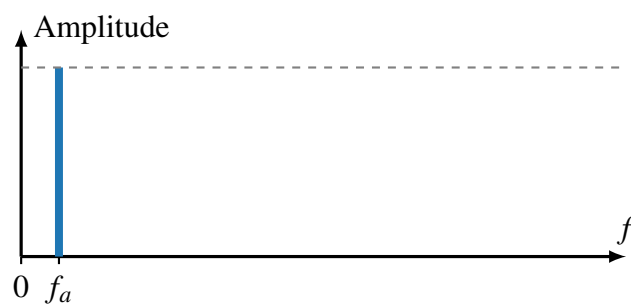
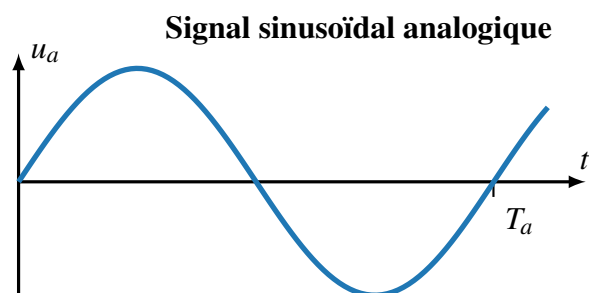


- $u_a(t)$ est le **signal analogique** en fonction du temps.
- $u_e(t)$ est le **signal échantillonné** en fonction du temps.

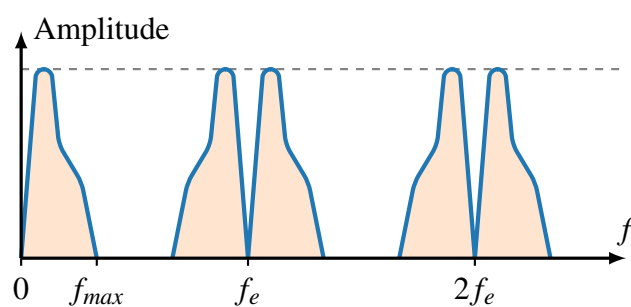
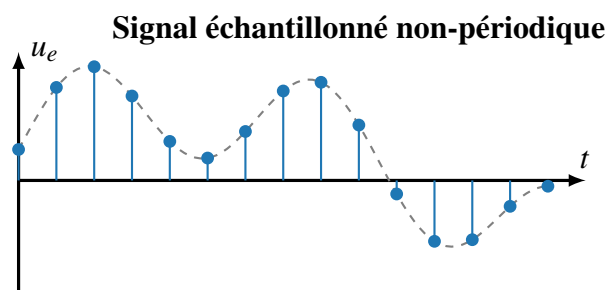
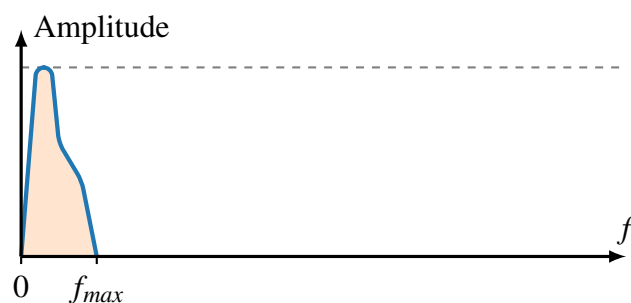
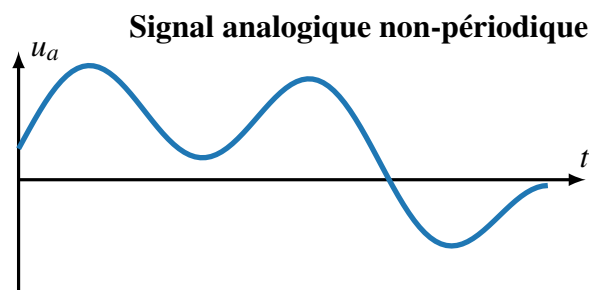
Exemple :



2) Spectres d'amplitude



Le spectre d'un signal échantillonné se répète !



III - Condition de Shannon

Pour que l'échantillonnage d'un signal analogique se fasse correctement, la condition de Shannon impose :

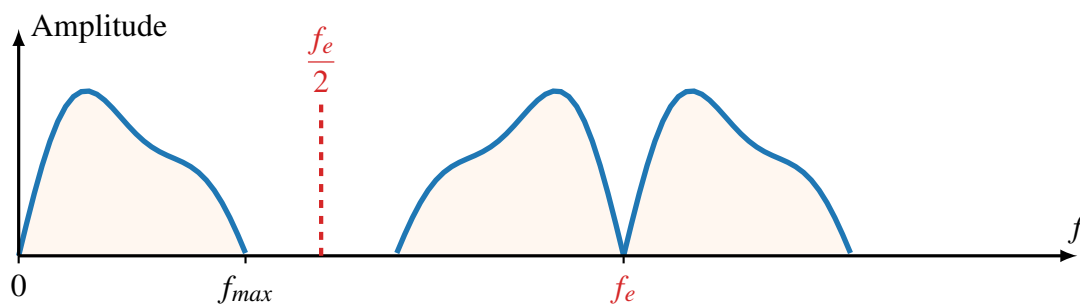
$$f_e > 2f_{max}$$

ou

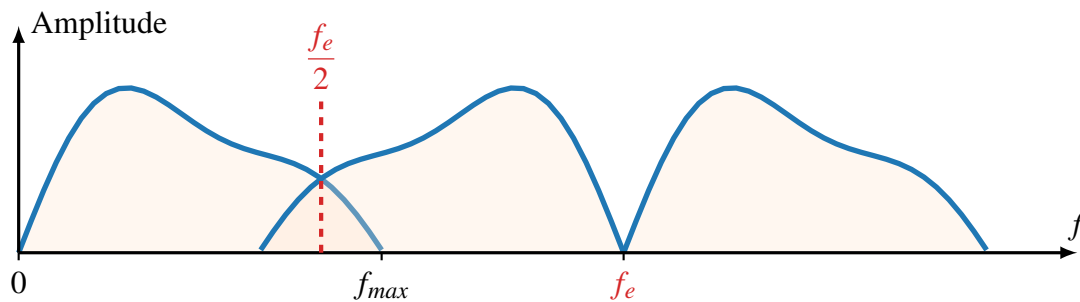
$$f_{max} < \frac{f_e}{2}$$

- f_e est la fréquence d'échantillonnage.
- f_{max} est la fréquence maximale du signal analogique.

Exemple : Signal analogique respectant la condition de Shannon.

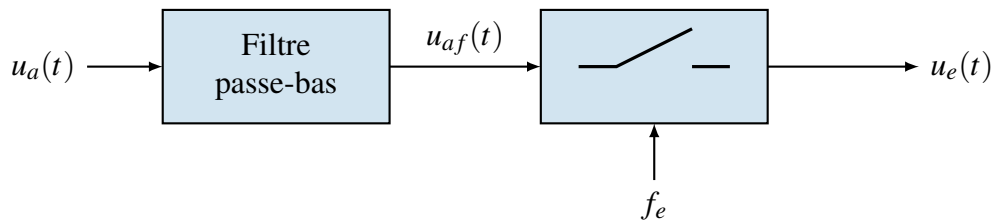


Exemple : Signal analogique ne respectant pas la condition de Shannon.



Le dernier exemple met en évidence un problème appelé **repliement du spectre** !

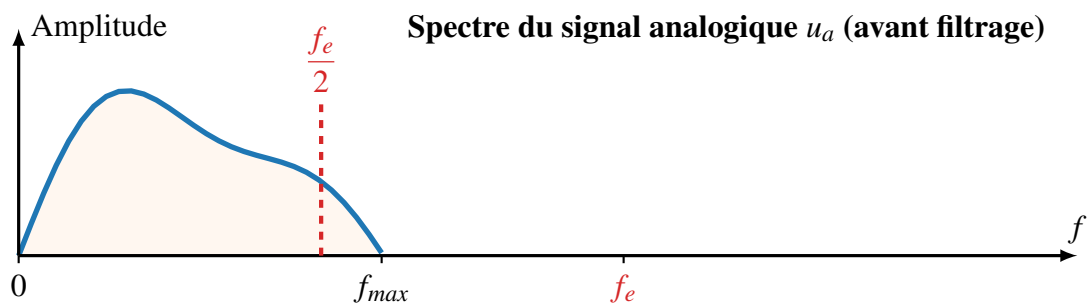
IV - Filtre anti-repliement (anti-aliasing)



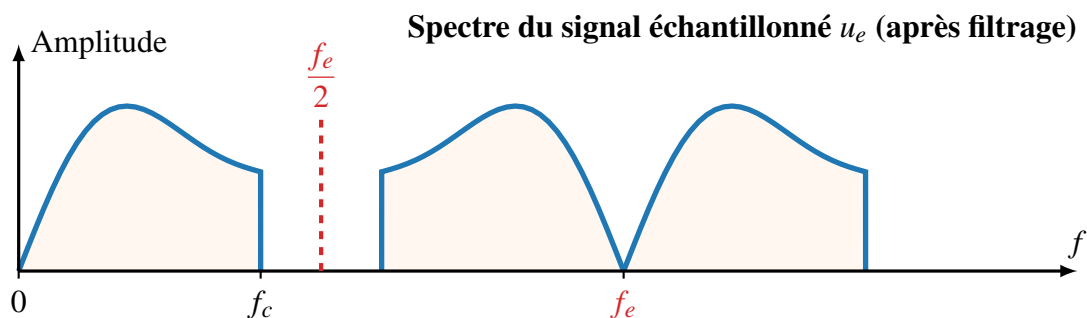
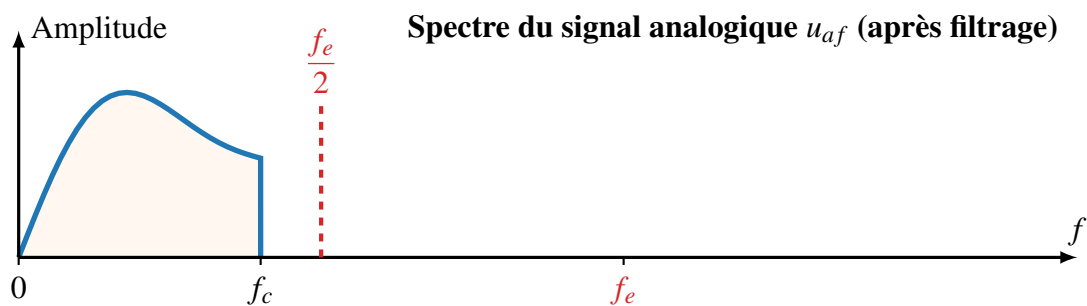
Pour éviter le phénomène de repliement de spectre, il faut limiter la fréquence maximale du signal analogique avec un filtre passe-bas tel que :

$$f_c < \frac{f_e}{2}$$

Exemple : Filtrage anti-repliement d'un signal ne respectant la condition de Shannon.



On applique un filtre passe bas (idéal) de fréquence de coupure $f_c < \frac{f_e}{2}$.



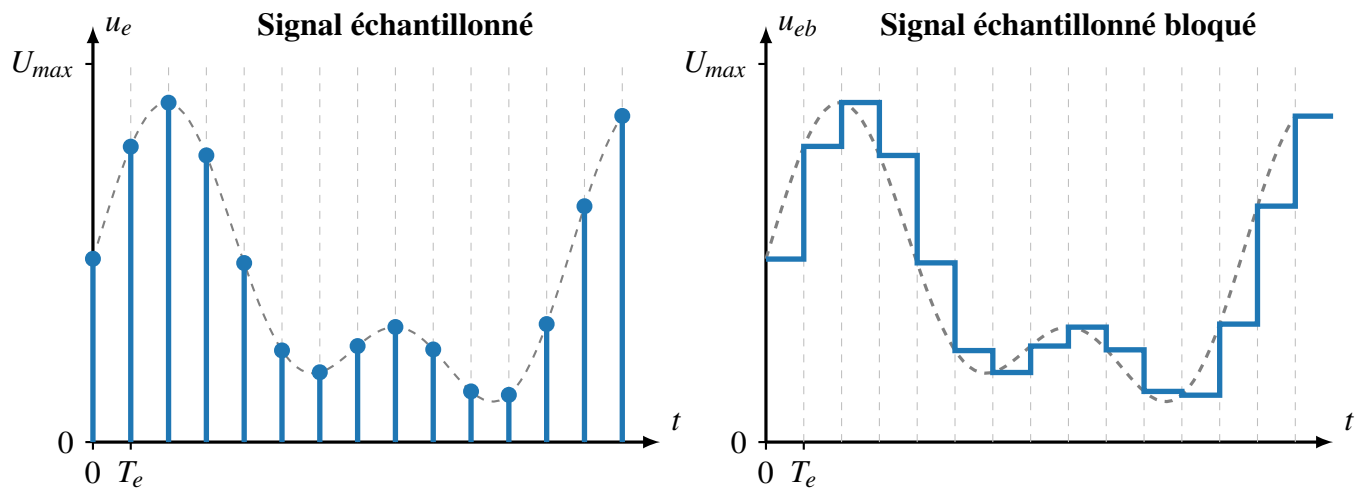
Avec le filtrage passe-bas, il n'y a plus de repliement de spectre après échantillonnage !

V - Blocage d'un signal échantillonné

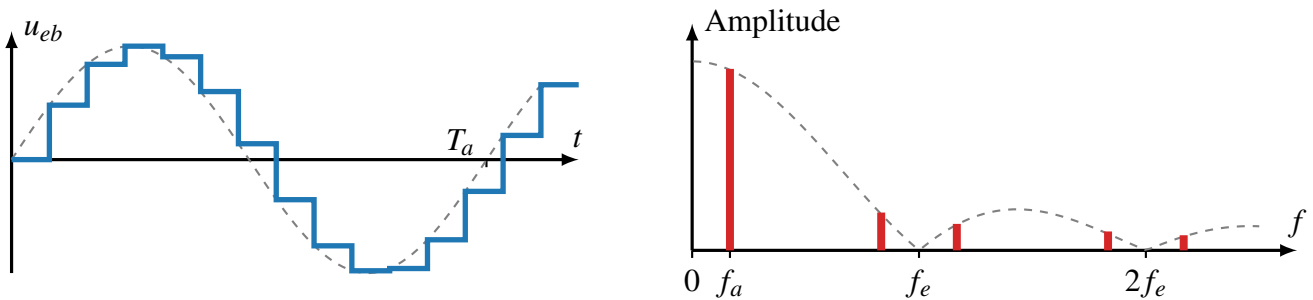
1) Principe du blocage

Pour laisser du temps à la conversion analogique-numérique dans la phase suivante, le signal échantillonné doit être **bloqué pendant la durée de la période d'échantillonnage T_e** .

Exemple :



2) Spectre d'amplitude

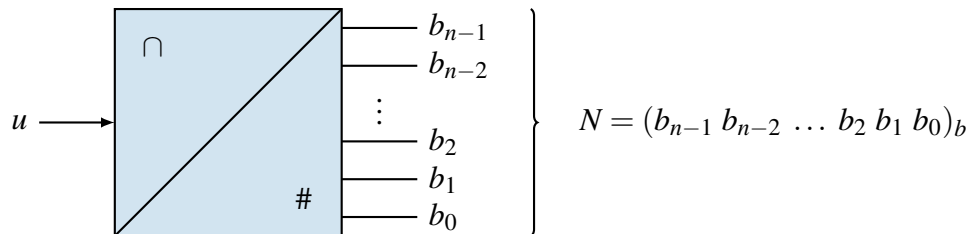


L'opération de blocage provoque une atténuation du spectre d'amplitude en $\left| \frac{\sin x}{x} \right|$

VI - Conversion analogique-numérique

1) Principe

Un CAN (Convertisseur Analogique Numérique) réalise la conversion d'une tension analogique en un nombre binaire N .



2) Résolution

La résolution d'un CAN est le **nombre de bits** utilisés pour la conversion.



$n = 10$ bits pour le CAN d'un Arduino Uno R3 (entrées analogiques).

3) Nombre maximal

Le nombre maximal obtenu en sortie du CAN est lié à la résolution n par la relation :

$$N_{max} = 2^n - 1$$



$N_{max} = 2^{10} - 1 = 1023$ pour le CAN d'un Arduino Uno R3 (entrées analogiques).

4) Valeur pleine échelle

La valeur pleine échelle U_{PE} est la **tension maximale** acceptable en entrée du CAN.

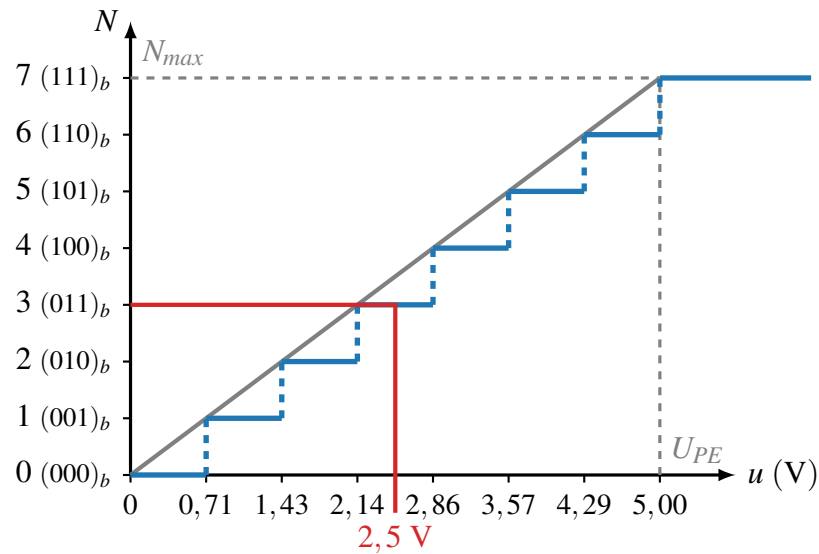


$U_{PE} = 5$ V (tension d'alimentation par défaut) pour le CAN d'un Arduino Uno R3 (entrées analogiques).

5) Caractéristique de transfert

La conversion analogique-numérique impose une **quantification** de la tension analogique sur 2^n niveaux de tension de 0 V à une tension maximale U_{PE} (valeur pleine échelle). La caractéristique de transfert est en **marche d'escalier** (non-linéaire).

Exemple : $n = 3$ bits et $U_{PE} = 5$ V



$N = 3$ est le même nombre obtenu pour toutes les valeurs de tension entre 2,14 V et 2,86 V (non comprise) !

6) Quantum

Le quantum (ou résolution analogique) est le **pas de quantification** tel que :

$$q = \frac{U_{PE}}{2^n - 1} \quad (\text{V})$$

ou

$$q = \frac{U_{PE}}{N_{max}} \quad (\text{V})$$



$q = \frac{5}{1023} \approx 4,89$ mV pour le CAN d'un Arduino Uno R3 (entrées analogiques).

7) Temps de conversion

Pour convertir correctement un signal échantillonné avec une période T_e en un signal numérique, le **temps de conversion** T_{conv} du CAN doit respecter la condition :

$$T_{conv} < T_e$$



$T_{conv} \approx 100$ μ s en moyenne pour le CAN d'un Arduino Uno R3 (entrées analogiques).

VII - Profondeur de mémoire

Certains systèmes électroniques ont besoin d'enregistrer les échantillons de la tension numérisée dans une mémoire.

La profondeur de mémoire est le **nombre d'échantillons** (ou nombre de points) que peut stocker la mémoire du système électronique lors d'une acquisition.

La durée d'acquisition d'une tension est liée à la profondeur de mémoire et à la fréquence d'échantillonnage par la relation :

$$\Delta t = PM \times T_e \quad \Rightarrow \quad \Delta t = \frac{PM}{f_e}$$

Une bonne profondeur de mémoire permet :

- d'acquérir des signaux de **longue durée** ;
- d'observer des **détails** dans un signal (ex. Zoom).



Les oscilloscopes numériques modernes disposent de grande profondeur de mémoire (jusqu'à des centaines de Mpts) pour acquérir les signaux et permettre des analyses profondes (zoom).



Mais attention, une trop grande profondeur peut ralentir les post-traitements (ex. FFT pour un oscilloscope).

Il faut donc un bon compromis entre profondeur de mémoire et fréquence d'échantillonnage. La plupart des oscilloscopes le gère automatiquement.