MISE EN ŒUVRE

→ **TRAITER : ARDUINO UNO (EDI Arduino)**

→ **AQUERIR : Potentiomètre**

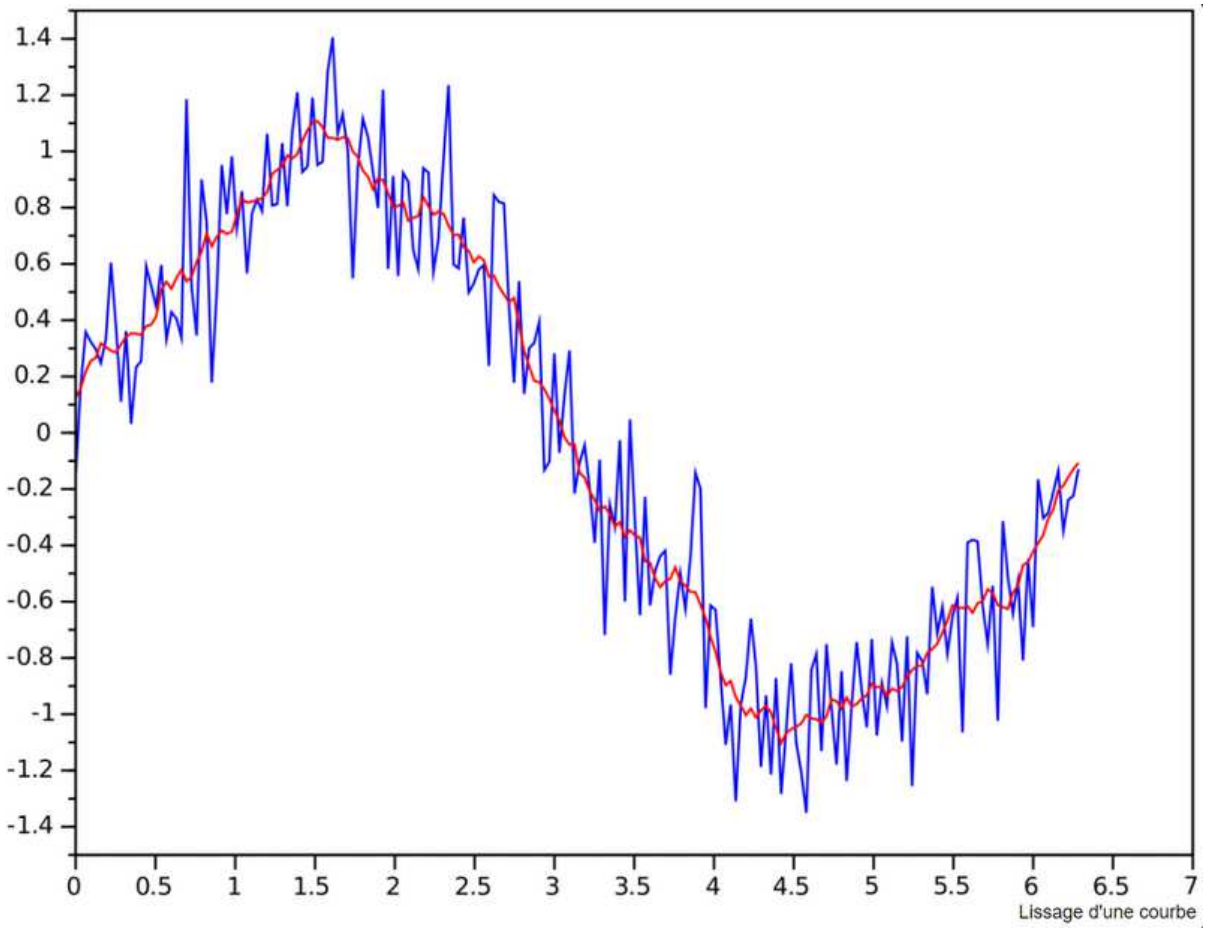
→ **COMMUNIQUER : Moniteur série**

1 – Mise en situation

Les courbes obtenues après importation des valeurs brutes issues d'un capteur peuvent être parfois difficiles à exploiter en raison des **bruits de mesures**.

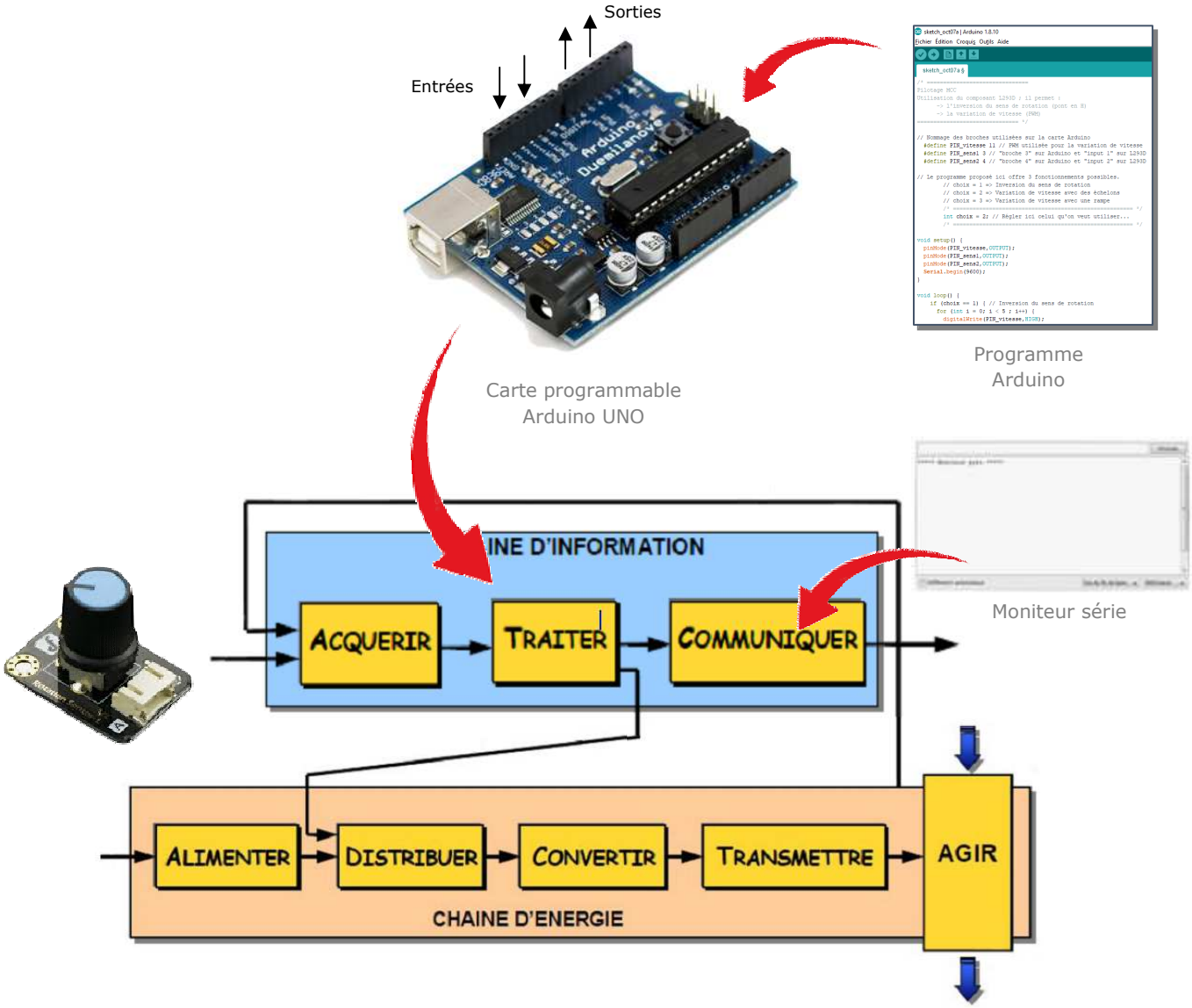
Il est donc intéressant de filtrer ces défauts dont la fréquence des variations est élevée (comparée aux variations du signal que l'on souhaite observer) pour ne conserver que l'évolution de la grandeur qui nous intéresse. Comme on peut le voir sur le graphe ci-après, **on supprime ainsi les fluctuations transitoires** de façon à souligner les tendances à plus long terme.

On parle de **l'opération de filtrage** qui est une étape indispensable du processus de conversion d'un signal analogique en signal numérique.

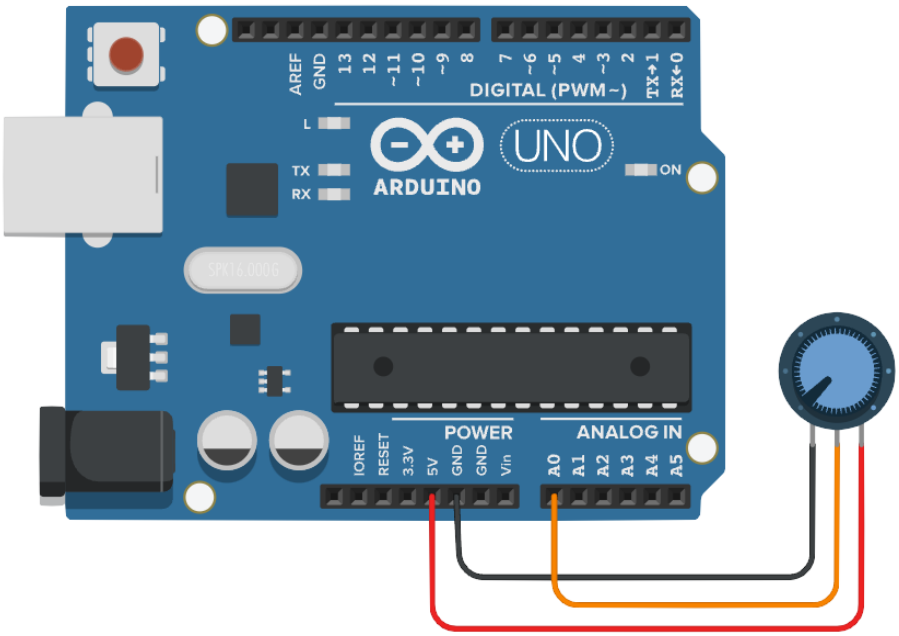


En **bleu** : le signal non traité

En **rouge** : le signal traité à l'aide d'une moyenne mobile



2 – Plan de câblage / Montage



3 – Programme

Ce programme affiche dans le moniteur série les valeurs calculées de la moyenne mobile.

➘ Bibliothèques requises : RunningAverage.h

➘ Source : https://github.com/RobTillaart/Arduino/tree/master/libraries/RunningAverage

```
// ==================================
// Lissage : moyenne glissante
// ==================================
// source : https://github.com/RobTillaart/Arduino/tree/master/libraries/RunningAverage
/*
Ce programme ne fonctionne pas dans le simulateur TinkerCad.
c'est normal, ceci est du au fait que le bibliothèque "RunningAverage.h"
n'existe pas dedans.
*/

// Appel des bibliothèques
    #include "RunningAverage.h"

// Paramètres de la moyenne mobile (à régler comme on veut)
    int inputPin = A0;    // Broche d'un signal analogique ou mettre un nom de variable...
    int taille = 10;      // nombre de valeurs à prendre pour calculer la moyenne mobile
    int pause = 100;       // valeur en ms du delay() à la fin du calcul de la moyenne mobile


// Création d'un objet "Moyenne glissante"
    RunningAverage MoyGliss(taille);  // objet "Moyenne Glissante" avec nombre d'échantillons pour le calcul

// Paramètres nécessaire au fonctionnement
    int MoGl_cpt = 0; // Compteur d'échantillons pris
    float moyenneG = 0; // Moyenne glissante calculée sur les échantillons

void setup(void) {
    Serial.begin(9600);
    MoyGliss.clear(); // Initialisation de l'objet "Moyenne Glissante"
}

void loop(void) {
    int echantillon = analogRead(inputPin); // Lecture du capteur
    MoyGliss.addValue(echantillon); // Ajout de l'échantillon à l'objet "Moyenne Glissante"
    moyenneG = MoyGliss.getAverage(); // Récupération de la moyenne glissante

    // Affichage des résultats
        Serial.print(echantillon);
        Serial.print(" - ");
        Serial.println(moyenneG);

    MoGl_cpt++; // Incrémentation du compteur
    if (MoGl_cpt == 300) { // Effacement de l'historique pour éviter les débordements
        MoGl_cpt = 0;
        MoyGliss.clear();
    }
    delay(pause);
}
```