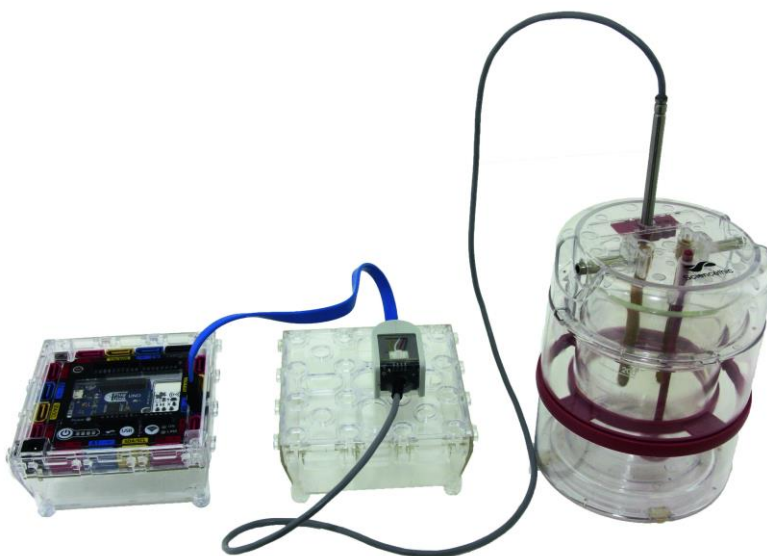


Niveau : seconde	Ondes et signaux partie 3 : Signaux et capteurs
Capteurs électriques.	<i>Utiliser un dispositif avec microcontrôleur et capteur.</i>

Dossier TP Plug'Uino® :

Réalisation d'un thermomètre électronique à l'aide d'une CTN à partir d'un étalonnage simple Programmation en langage Arduino



1. Résumé de l'activité

Cette activité expérimentale a pour but de réaliser la courbe d'étalonnage d'une sonde de mesure de température CTN, et de réaliser ainsi un thermomètre numérique.

2. Thème du programme abordé

Niveau : seconde	Ondes et signaux partie 3 : Signaux et capteurs
Capteurs électriques.	<i>Utiliser un dispositif avec microcontrôleur et capteur.</i>

3. Matériel mis en œuvre

Une interface Plug'Uino[®] Uno Réf. 650 003 ;
 Un capteur de température CTN Plug'Uino[®] Réf. 651 054;
 Un calorimètre Réf. 005 025;
 Un thermomètre numérique Réf. 310 007;
 Un tableur-grapheur.

4. Présentation de l'expérience réalisée

- On verse de l'eau chaude dans un calorimètre. On y plonge la sonde de température CTN Plug'Uino[®], ainsi qu'un thermomètre de référence.
- À l'aide du microcontrôleur, on relève le nombre N fourni par l'entrée analogique sur laquelle est branché le capteur de température CTN, en fonction de la température T . On fait varier la température de l'eau en rajoutant progressivement de la glace pilée et en homogénéisant bien le mélange.
- On entre les valeurs obtenues dans un tableur-grapheur afin de visualiser la courbe d'étalonnage $T = f(N)$
- On réalise alors une courbe de tendance permettant d'obtenir la relation mathématique $T = f(N)$
- On rentre cette fonction dans le microcontrôleur de façon à ce qu'il puisse afficher directement la température mesurée par la sonde CTN.

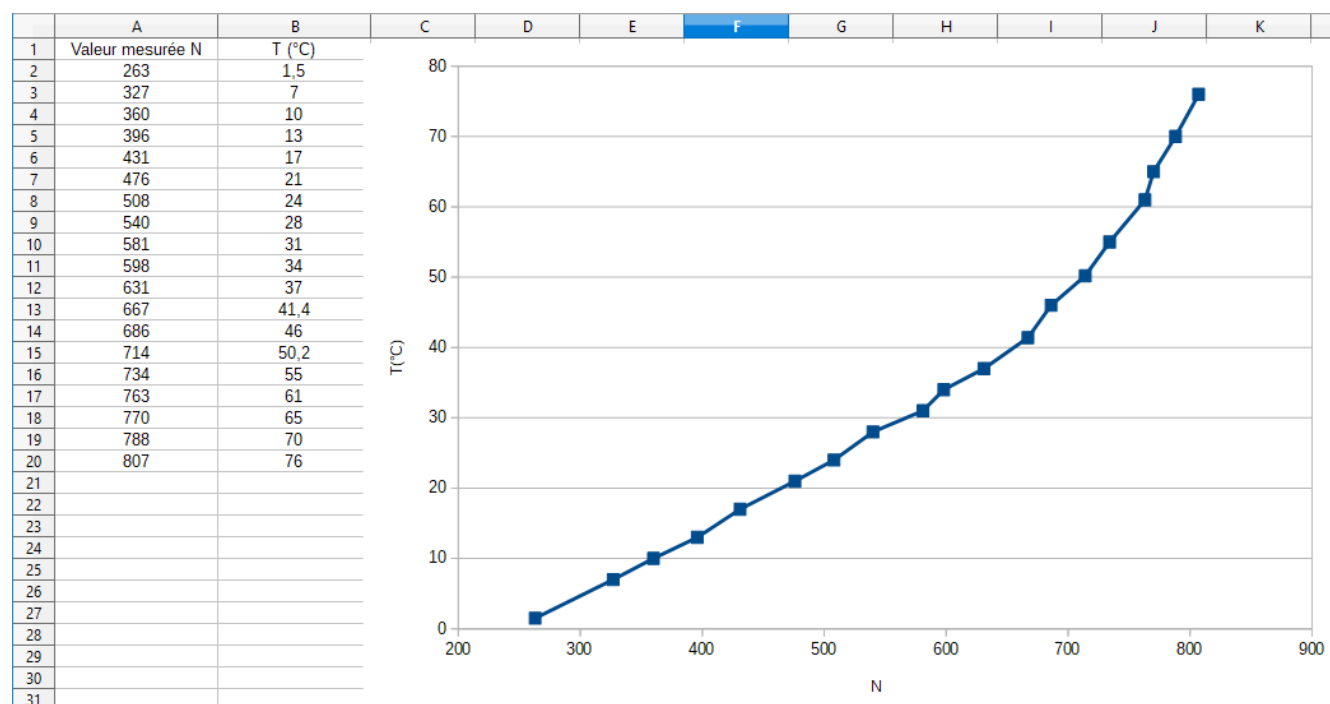
5. Programme Arduino

Le programme Arduino fonctionnel, permettant de relever le nombre mesuré par le microcontrôleur et délivré par la sonde de température CTN est le suivant :

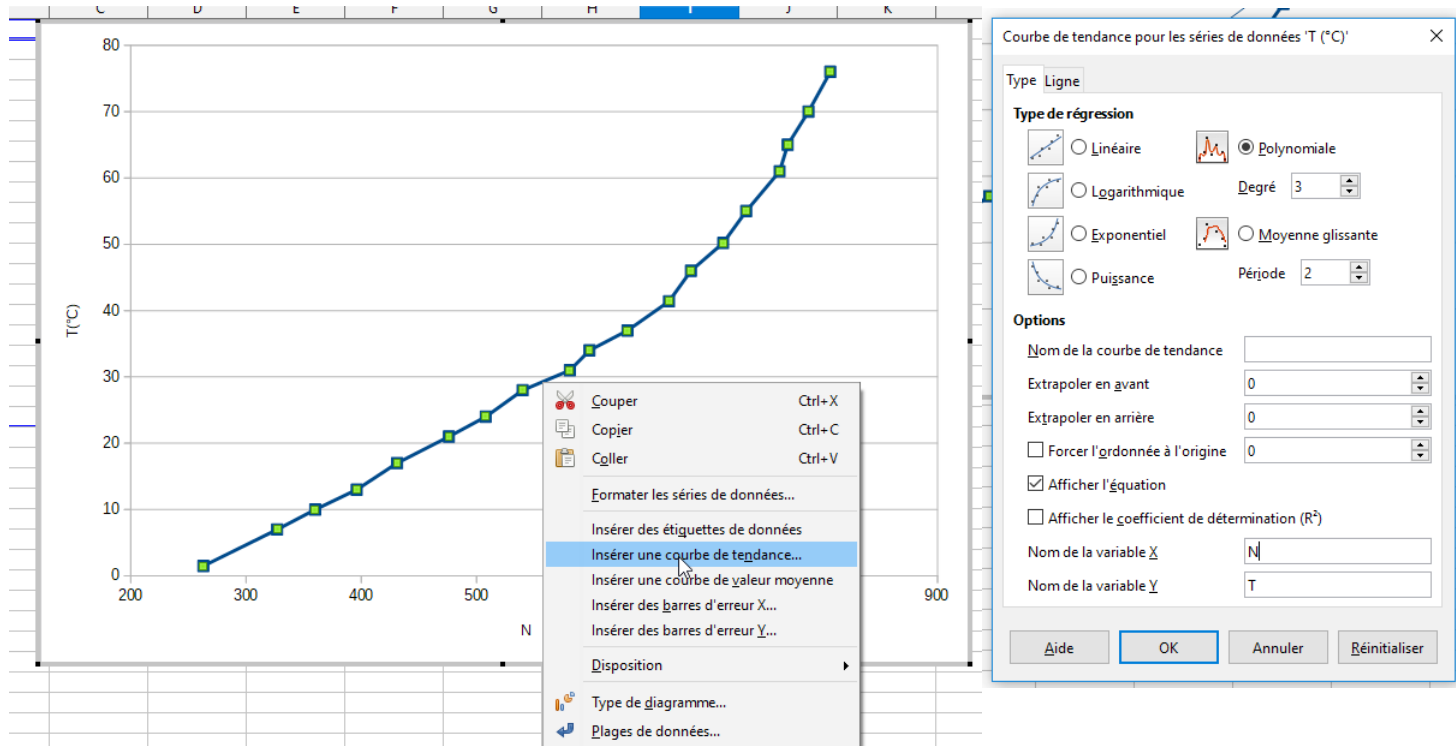
```
ScEthic_Etalonnage_Simple_CTN $
1 //
2 //   Sciencéthic
3 //
4 //   Etalonnage simple d'une sonde CTN pour microcontrôleur
5 //
6
7 float N;                // Définit un nombre flottant N
8
9 #define BrocheConnexion A0 // Définit la broche sur laquelle sera connectée la sonde de température CTN
10                             // Ici, elle sera reliée à l'entrée A0
11
12 void setup() {           // Initialisation du programme
13   Serial.begin(9600);     // Initialise la vitesse de communication entre la carte Plug'Uino et le PC
14                           // à 9600 bps
15 }
16
17 void loop() {            // Boucle principale
18   N = analogRead(BrocheConnexion); // Met dans N la mesure faite sur l'entrée BrocheConnexion
19   Serial.println(N);      // Affiche le nombre N
20   delay(1000);            // Attend une seconde avant de boucler
21 }
```

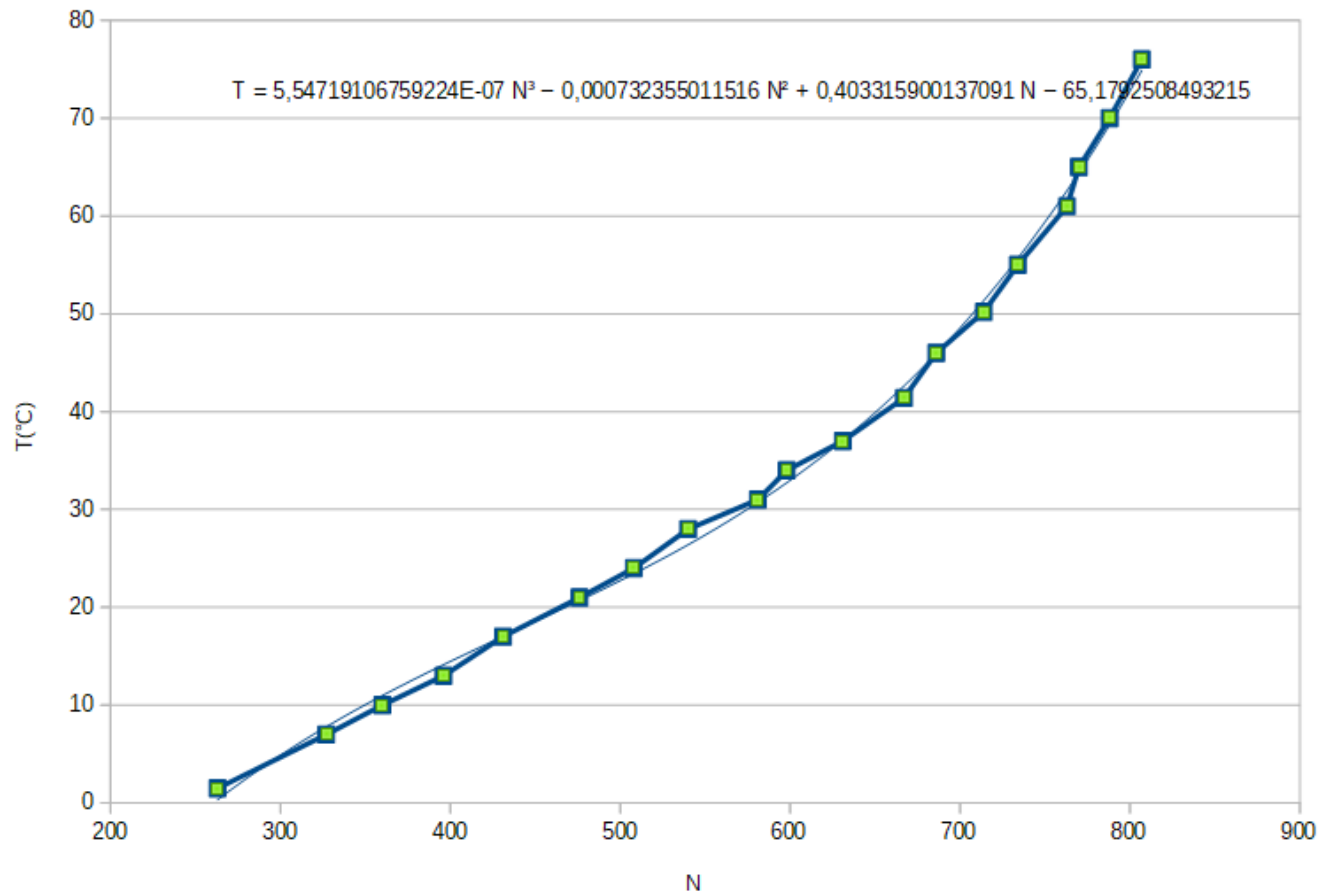
6. Résultats obtenus

La courbe d'étalonnage obtenue est la suivante :



On sélectionne une courbe de tendance polynomiale de degré 3, et on demande d'afficher l'équation comme ci-contre.





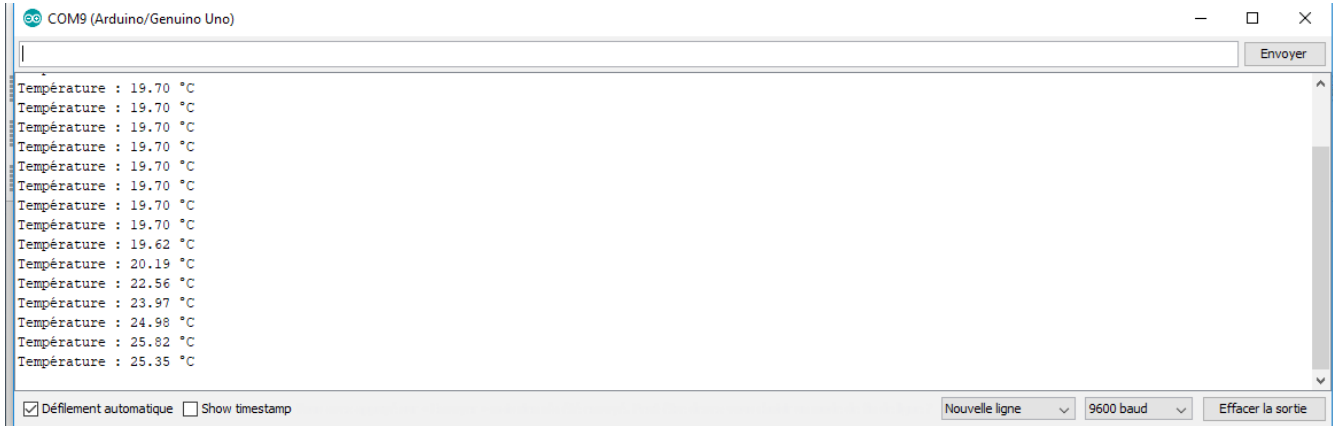
La courbe de tendance ainsi que son équation sont alors affichées :

Il suffit alors de modifier le programme Arduino pour y rentrer la modélisation obtenue (ici en ligne 20) :

ScEthic_ThermometreNumerique_Simple_CTN \$

```
1 //
2 //   Sciencéthic
3 //
4 //   Réalisation d'un thermomètre numérique à l'aide d'une CTN
5 //
6
7 float N;           // Définit un nombre flottant N
8 float T;           // Définit un nombre flottant T
9
10 #define BrocheConnexion A0 // Définit la broche sur laquelle sera connectée la sonde de température CTN
11                             // Ici, elle sera reliée à l'entrée A0
12
13 void setup() {       // Initialisation du programme
14   Serial.begin(9600); // Initialise la vitesse de communication entre la carte Plug'Uino et le PC
15                       // à 9600 bps
16 }
17
18 void loop() {        // Boucle principale
19   N = analogRead(BrocheConnexion); // Met dans N la mesure faite sur l'entrée BrocheConnexion
20   T = 5.54719106789224E-7*N*N*N - 0.000732355011516*N*N + 0.403315900137091*N - 65.1792508493215;
21   // La ligne 20 calcule la valeur de la température qui correspond à N
22   Serial.print("Température : "); // Affiche "Température :"
23   Serial.print(T);                // Affiche la valeur de la température calculée
24   Serial.println(" °C");          // Affiche l'unité et passe à la ligne
25   delay(1000);                    // Attend une seconde avant de boucler
26 }
```

Les mesures sont alors affichées ainsi :



The screenshot shows the Arduino IDE serial monitor window for a COM9 (Arduino/Genuino Uno) connection. The window displays a list of temperature readings in degrees Celsius. The readings are: 19.70, 19.70, 19.70, 19.70, 19.70, 19.70, 19.70, 19.70, 19.70, 19.62, 20.19, 22.56, 23.97, 24.98, 25.82, and 25.35. The window also includes a search bar, an 'Envoyer' button, and checkboxes for 'Défilement automatique' (checked) and 'Show timestamp' (unchecked). The baud rate is set to 9600 and the line type is 'Nouvelle ligne'.

```
COM9 (Arduino/Genuino Uno)
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.70 °C
Température : 19.62 °C
Température : 20.19 °C
Température : 22.56 °C
Température : 23.97 °C
Température : 24.98 °C
Température : 25.82 °C
Température : 25.35 °C
```

7. Exploitation

7.1. Programmation

En fonction des capacités des différents élèves on peut leur demander diverses choses :

- à un niveau **d'initiation**, on peut enlever certains commentaires du programme et demander aux élèves à quoi correspond cette ligne ;
- à un niveau **avancé** on peut par exemple enlever la ligne 20 et demander aux élèves de rentrer l'équation obtenue lors de la phase d'étalonnage ;
- à un niveau **expert** on peut demander aux élèves de programmer complètement le microcontrôleur.

7.2. Résultats des mesures

On peut discuter avec les élèves de la mesure de température affichée en fonction de celle mesurée par un thermomètre : l'écart est-il grand ? Varie-t-il en fonction de la plage de température sur laquelle on travaille ? Les mesures réalisées à partir du microcontrôleur sont-elles compatibles avec les grandeurs mesurées au thermomètre ?