

Présentiel - 2^{nde}

FICHE 2

Dispositif avec microcontrôleur et capteur Exemple d'un capteur de température

Capacités exigibles (BO 2^{nde}) :

« Mesurer une grandeur physique à l'aide d'un capteur électrique résistif.

Produire et utiliser une courbe d'étalonnage reliant la résistance d'un système avec une grandeur d'intérêt (température, pression, intensité lumineuse, etc.).

Utiliser un dispositif avec microcontrôleur et capteur.»

Objectif :

Utiliser un dispositif avec microcontrôleur et capteur de température (thermistance CTN) pour construire **une alarme à incendie**.

Applications : Détecteur d'échauffement ou de surchauffe (signal d'alerte sonore et/ou lumineux)

Programme de 2^{nde} : Thème « Ondes et signaux » 3. Signaux et capteurs

Compétences travaillées :

- S'approprier
- Analyser/Raisonner
- Réaliser
- Valider
- Communiquer

Prérequis :

- Loi des nœuds. Loi des mailles. Loi d'Ohm.
- Quelques notions de programmation (voir FICHE 1)

Organisation de la ressource :

- Pages 2 : Présentation de la séquence
- Pages 4 à 9 : Activités de l'élève (TP clé en main)
- Pages 11 à 22 : Corrigés et compléments pour le professeur

Mots clefs :

- Capteurs, microcontrôleur, courbe d'étalonnage, programmation, compétences expérimentales et numériques.

Matériel :

- Une thermistance (avec pattes gainées), un Ohmmètre, 2 fils de connexion (+ 2 pinces crocodiles), un thermomètre numérique de référence (précision $\pm 0,5$ °C), un bécher, de l'eau, des glaçons, un ballon, un chauffe ballon, une potence.
- Ordinateur avec le logiciel Arduino et/ou mBlock pour réaliser et téléverser des programmes simples.
- Maquette pédagogique PB200 (distribuée par PIERRON catalogue 2019) ou tout autre dispositif réalisé par l'équipe enseignante.

Présentation de la séquence

Situation déclenchante

Beaucoup de capteurs sont présents dans les objets de la vie quotidienne. Les capteurs de température sont souvent utilisés pour protéger les bâtiments, les machines et les Hommes.



Installer un détecteur de fumée dans les logements est obligatoire depuis le 8 mars 2015.

En complément des détecteurs de fumée, les capteurs de température sont un premier rempart et un excellent moyen pour alerter de manière automatique une surchauffe qui souvent est à l'origine d'un départ de feu.

Problématiques

Un échange entre l'enseignant et la classe peut conduire à dégager différentes problématiques :

Comment fonctionne une alarme à incendie ? Quels types de capteurs électriques sont au service de notre sécurité, de celles des bâtiments, des machines... ?

Comment utiliser un dispositif avec microcontrôleur et capteur de température pour réaliser une alarme à incendie ?

Comment maîtriser le seuil de déclenchement de cette alarme ?

Organisation de la séquence :

- **Partie A** : Le capteur de température (THERMISTANCE CTN) et sa courbe d'étalonnage
- **Partie B** : Le microcontrôleur, au cœur du dispositif de **l'alarme à incendie**
- **Partie C** : Application des lois de la physique
- **Partie D** : Le Convertisseur Analogique-Numérique ou CAN du microcontrôleur
- **Partie E** : Programmation de **l'alarme à incendie**

Intérêts pratiques de cette séquence :

- Cette séquence d'enseignement clé en main permet aux enseignants de se projeter facilement dans les activités élèves concernant les usages pédagogiques des microcontrôleurs.
- Cette séquence alterne activités expérimentales (support de la formation des élèves, cf BO), et apports théoriques (restreints, nécessaires, suffisants) pour comprendre le fonctionnement et l'utilisation des microcontrôleurs.
- Le professeur pourra modifier, ajouter ou supprimer certaines parties de cette séquence en fonction de sa place dans la progression annuelle.
- Cette FICHE 2 a été conçue dans la continuité de la FICHE 1 déjà proposée.

FICHES ELEVE

TP UTILISATION D'UN DISPOSITIF AVEC MICROCONTRÔLEUR ET CAPTEUR - RÉALISATION D'UNE ALARME À INCENDIE -

Objectif : Utiliser un dispositif avec microcontrôleur et capteur de température pour réaliser une alarme à incendie.

Capacités exigibles (2^{nde}) :

Mesurer une grandeur physique à l'aide d'un capteur électrique résistif.

Partie E

Produire et utiliser une courbe d'étalonnage reliant la résistance d'un système avec une grandeur d'intérêt (température, etc.).

Partie A

Utiliser un dispositif avec microcontrôleur et capteur.

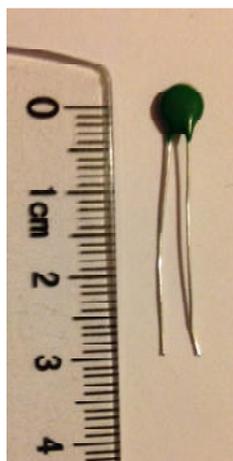
Parties B, C, D

Principe de fonctionnement de l'alarme à incendie :

Si la température du capteur est supérieure à 35°C alors des signaux sonore et lumineux sont émis par le dispositif pour alerter du danger (cette température de déclenchement de l'alarme est modifiable).

Partie A : Le capteur de température (thermistance CTN) et sa courbe d'étalonnage.

APP - REA



Une thermistance est un capteur de température qui se comporte comme une résistance dont la valeur dépend de la température : on l'appelle pour cette raison une **thermistance** (préfixe « **thermi** » pour **thermique** ; suffixe « **istance** » pour **résistance**).

L'objectif de la partie A est de tracer la courbe d'étalonnage de cette thermistance 10K CTN. Pour cela, il faut mesurer avec un Ohmmètre la résistance $R_{th}^{(1)}$ de ce capteur pour différentes températures T du milieu dans lequel il se trouve puis, il faut tracer le graphique $R_{th} = f(T)$ qui montre la dépendance de la résistance du dipôle en fonction de la température.

⁽¹⁾ L'indice $_{th}$ signifie thermistance.

Consigne : Avec le matériel mis à ta disposition, propose et met en place un protocole expérimental afin de construire un tableau de valeurs qui te permettra ensuite de tracer la courbe d'étalonnage de la thermistance.

Matériel :

Un bécher, de l'eau, des glaçons, un thermomètre numérique de référence (précision $\pm 0,5$ °C), un ballon, un chauffe ballon, une potence, un Ohmmètre, la thermistance (avec les pattes gainées), 2 fils de connexion (+ 2 pinces crocodiles) pour l'Ohmmètre.

Schéma de l'expérience :

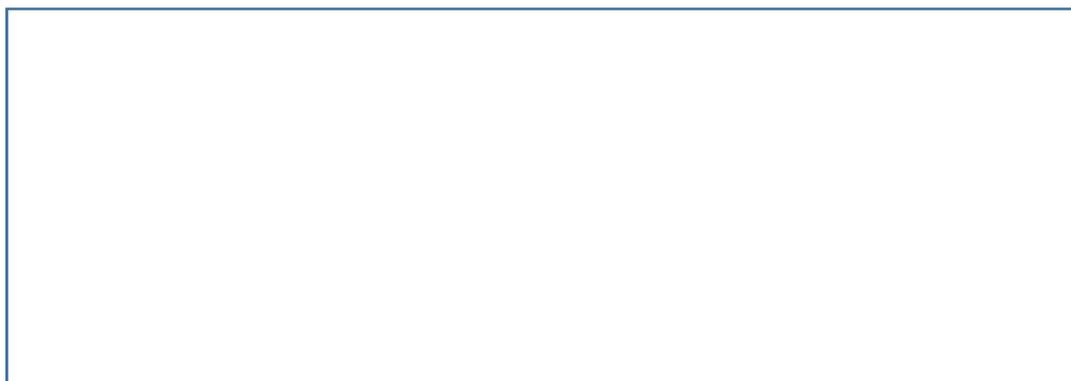


Tableau des mesures :

température T (en °C)	0										100
résistance R_{th} (en $k\Omega$)											

Courbe d'étalonnage de la thermistance :

Sur une feuille de papier millimétré, trace la courbe d'étalonnage $R_{th} = f(T)$.

Ou bien avec le langage de programmation PYTHON, trace la courbe d'étalonnage $R_{th} = f(T)$

Collez votre courbe d'étalonnage ici

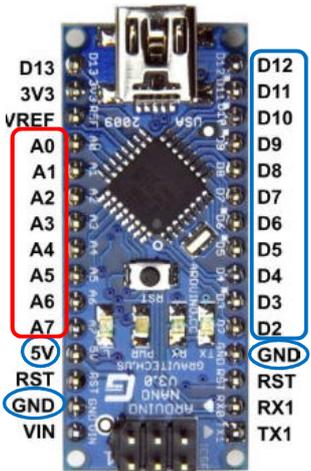
Collez votre programme PYTHON ici

Partie B : Le microcontrôleur, au cœur du dispositif de l'alarme à incendie

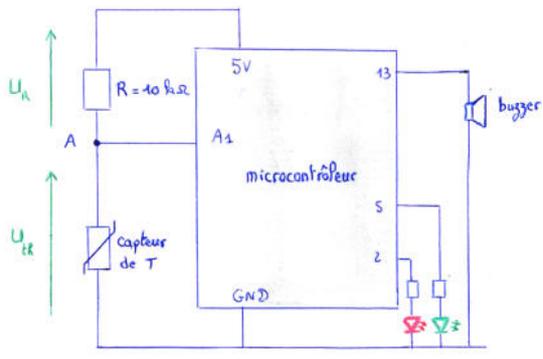
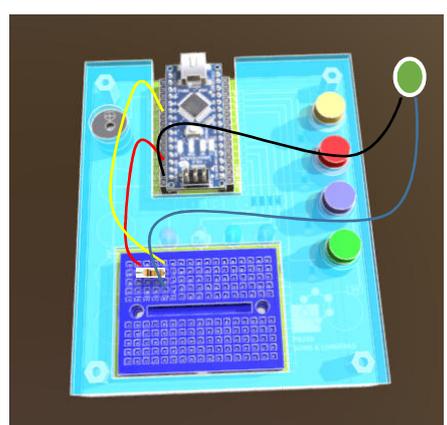
APP - REA

« Les capteurs écoutent le monde physique.
 Les actionneurs agissent dans le monde physique.
 Les microcontrôleurs écoutent les capteurs et parlent aux actionneurs. »

Source : le livre de projets Arduino

<p>Dans ce TP n°2, vous allez utiliser d'autres types de broches, il s'agit des entrées analogiques, il y en a 8, numérotées de A0 jusqu'à A7. C'est sur ces broches que l'on branche les capteurs.</p>	 <p>Microcontrôleur Arduino™ type Nano</p>	<p>Dans le TP n°1, vous avez repéré :</p> <ul style="list-style-type: none"> - les broches 5V et GND pour alimenter vos circuits avec du +5 V et une masse (0V), - ainsi que les broches numériques D2 à D13 qui envoient, selon le programme téléversé, un signal HAUT (+5V) ou BAS (0V) à des actionneurs (DEL ou buzzer) reliés à ces broches.
--	---	--

Il ne faut pas brancher un capteur directement entre la broche 5V et la broche GND (0V) ! En effet, selon la valeur de la grandeur physique qu'il mesure, le capteur peut avoir une résistance très faible. L'intensité du courant électrique qui le traverse peut dans ce cas être très élevée, et vous risquez de griller le microcontrôleur. Ceci explique que dans le dispositif d'alarme à incendie, on ajoute une résistance ($R=10\text{ k}\Omega$) en série avec le capteur selon le schéma suivant :

<p>Schéma électrique de l'alarme à incendie</p> 	<p>Circuit avec la maquette PB200</p>  <p>Capteur de T</p>
---	--

Si vous utilisez la maquette pédagogique PB200 (distribuée par PIERRON), les deux DEL et le buzzer sont déjà câblés, il ne reste donc plus qu'à brancher la thermistance et la résistance $R=10\text{ k}\Omega$ en utilisant la plaque multitrouss bleue. **N'OUBLIEZ PAS** le fil de connexion qui relie le nœud A à la broche A1.

APPELEZ LE PROFESSEUR POUR VERIFIER VOTRE CIRCUIT

Schéma électrique de l'alarme à incendie	Schéma électrique équivalent, seule la maille contenant le capteur est représentée
	<div style="border: 1px solid red; padding: 5px; width: fit-content; margin: 10px auto;"> $U_{th} = U_G \times \frac{R_{th}}{R + R_{th}}$ </div>

Remarque très importante : Le courant électrique ne circule pas dans la branche A-A1. L'intensité du courant électrique qui circule dans la résistance $R=10\text{ k}\Omega$ est donc la même que celle qui circule dans la thermistance.

- 1) Lorsque le capteur est à la température ambiante, mesurez les tensions U_R , U_{th} et U_G et vérifiez la loi des mailles.

- 2) À l'aide de la loi d'Ohm et de la loi d'unicité du courant électrique dans un circuit en série, établir la relation entre la tension U_{th} aux bornes de la thermistance, la tension U_R et les deux valeurs des résistances R_{th} et R .

- 3) À l'aide de la loi des mailles, en déduire la relation encadrée en rouge sous le schéma électrique.

APPELEZ LE PROFESSEUR POUR VERIFIER VOTRE TRAVAIL

Partie D : Le convertisseur Analogique Numérique ou CAN
cas où $U_G = 4,64 \text{ V}$
ANA

Avec la courbe d'étalonnage (partie A), nous avons vu que si nous connaissons R_{th} alors, nous pouvons en déduire la température T .

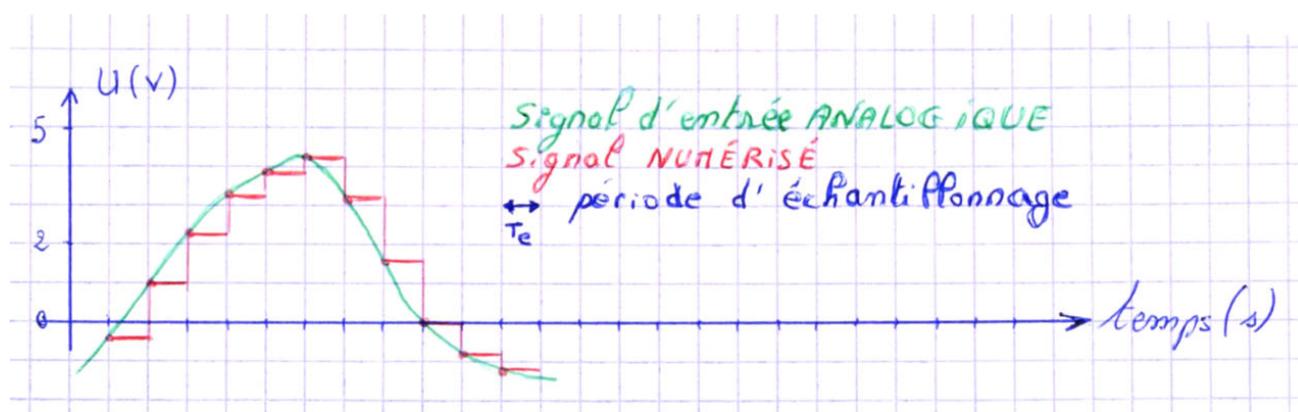
Les lois de la physique (partie C) nous donnent la relation entre R_{th} et U_{th} .

Les grandeurs physiques R_{th} et U_{th} sont donc des images de la température. Si nous pouvons mesurer l'une de ces deux grandeurs, alors nous pourrions en déduire la valeur de la température.

Mesurer U_{th} avec un voltmètre est très simple. Néanmoins, le microcontrôleur est capable de mesurer seul et très rapidement cette tension électrique U_{th} entre sa masse et ses entrées analogiques A0 à A7.

Le microcontrôleur mesure U_{th} à intervalle de temps régulier, bloque cette donnée, l'exploite éventuellement, puis refait une mesure, la bloque à nouveau, l'exploite éventuellement etc. à une fréquence très rapide appelée fréquence d'échantillonnage : en physique, cette fonction s'appelle un échantillonneur bloqueur.

La tension analogique U_{th} (**signal continu**) et ainsi remplacée par une tension numérisée (**signal discontinu**).



Une autre caractéristique importante d'un microcontrôleur est le nombre de bits utilisé pour coder les valeurs des tensions mesurées. Le microcontrôleur (ATmega328P utilisé ici) code l'amplitude de la tension en « mots » de **10 bits**.

Un « **bit** » (de l'anglais *binary digit*) est un chiffre binaire (**0** ou **1**).

Avec un mot de **2 bits**, on peut écrire : 00, 01, 10 et 11 soit 4 valeurs. ($4=2^2$)

Avec un mot de **3 bits**, on peut écrire : 000, 001, 010, 011, 100, 101, 110, 111 soit 8 valeurs. ($8=2^3$) etc.

Avec un mot de **10 bits**, on peut écrire $2^{10} = 1024$ valeurs.

EXPÉRIENCE : Alimente ton microcontrôleur et mesure avec un voltmètre la tension électrique entre sa borne notée « 5V » et sa masse notée « GND ». Note la valeur mesurée $U_G = U_{max} = \dots \text{ V}$ c'est la tension maximale du microcontrôleur.

Avec un microcontrôleur Arduino™ de type Nano,

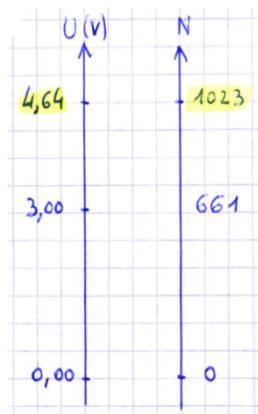
La tension minimale $U_{min} = 0,00 \text{ V}$ sur une entrée analogique est codée par le nombre **N=0**

La tension maximale $U_{max} = 4,64 \text{ V}$ sur une entrée analogique est codée par le nombre **N=1023**

Il y a une **relation de proportionnalité** entre la tension analogique U et la valeur numérique N mesurée par le microcontrôleur au niveau des entrées analogiques.

Par exemples :

- Si la tension mesurée au niveau de l'entrée analogique A1 vaut $U = 3,00 \text{ V}$ alors, le nombre correspondant à cette tension est $N = 1023 * 3,00 / 4,64 = 661$.
- Si la valeur numérique lue par le microcontrôleur au niveau d'une entrée analogique est $N = 224$, alors la valeur de la tension électrique correspondante est $U = 4,64 * 224 / 1023 = 1,02 \text{ V}$.



Partie E : Programmation de l'alarme à incendie

ANA - REA - VAL

- D'après la courbe d'étalonnage (partie A), quelle est la valeur de la résistance R_{th} lorsque la température est égale à 35 °C ?
- D'après les résultats des lois de la physique (partie C), en déduire la valeur de la tension U_{th} aux bornes de la thermistance lorsque la température est égale à 35 °C ?
- D'après la partie C, en déduire la valeur numérique N mesurée par le microcontrôleur lorsque la température est égale à 35°C ?

Voici un exemple de programme (réalisé avec le logiciel mBlock Version3) à téléverser dans le dispositif pour faire fonctionner l'alarme.

D'après les calculs précédents, par quel nombre N faut-il remplacer le nombre 500 dans ce programme pour que l'alarme se déclenche à partir de 35 °C ?

**ANALYSE DU PROGRAMME :**

D'après le programme proposé ci-dessus et le schéma du dispositif d'alarme page 6, décrivez en quelques lignes ce qu'il va se passer lorsque vous téléverserez ce programme dans le dispositif :

ÉCRIVEZ AVEC MBLOCK LE PROGRAMME CI-DESSUS (CORRIGÉ AVEC LA BONNE VALEUR DE N) ET TÉLÉVERSEZ-LE DANS LE DISPOSITIF.

VÉRIFIEZ LE BON FONCTIONNEMENT DE L'ALARME AVEC DE L'EAU CHAUDE OU UN SÈCHE-CHEVEUX PAR EXEMPLE.

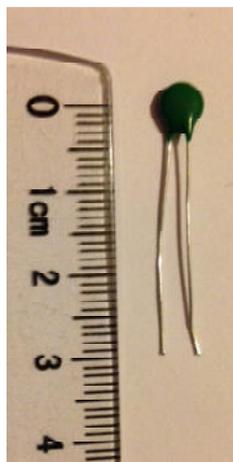
FICHES PROF

Corrections

Corrigé PARTIE A

Partie A : Le capteur de température (thermistance CTN) et sa courbe d'étalonnage.

APP - REA



Une thermistance est un capteur de température qui se comporte comme une résistance dont la valeur dépend de la température : on l'appelle pour cette raison une **thermistance** (préfixe « **thermi** » pour **thermique** ; suffixe « **istance** » pour **résistance**).

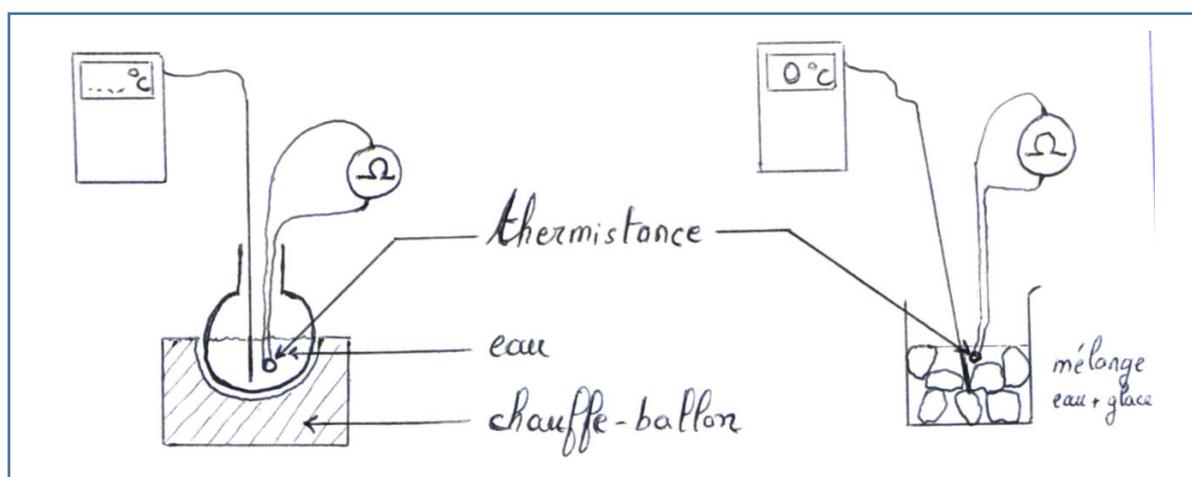
L'objectif de la partie A est de tracer la courbe d'étalonnage de cette thermistance 10K CTN. Pour cela, il faut mesurer avec un Ohmmètre la résistance $R_{th}^{(1)}$ de ce capteur pour différentes températures T du milieu dans lequel il se trouve puis, il faut tracer le graphique $R_{th} = f(T)$ qui montre la dépendance de la résistance du dipôle en fonction de la température.

⁽¹⁾ L'indice th signifie thermistance.

Consigne : Avec le matériel mis à ta disposition, propose et met en place un protocole expérimental afin de construire un tableau de valeurs qui te permettra ensuite de tracer la courbe d'étalonnage de la thermistance.

Matériel :

Un bécher, de l'eau, des glaçons, un thermomètre numérique de référence (précision $\pm 0,5$ °C), un ballon, un chauffe ballon, une potence, un Ohmmètre, la thermistance (avec les pattes gainées), 2 fils de connexion (+ 2 pinces crocodiles) pour l'Ohmmètre.

Schéma de l'expérience :**Tableau des mesures :**

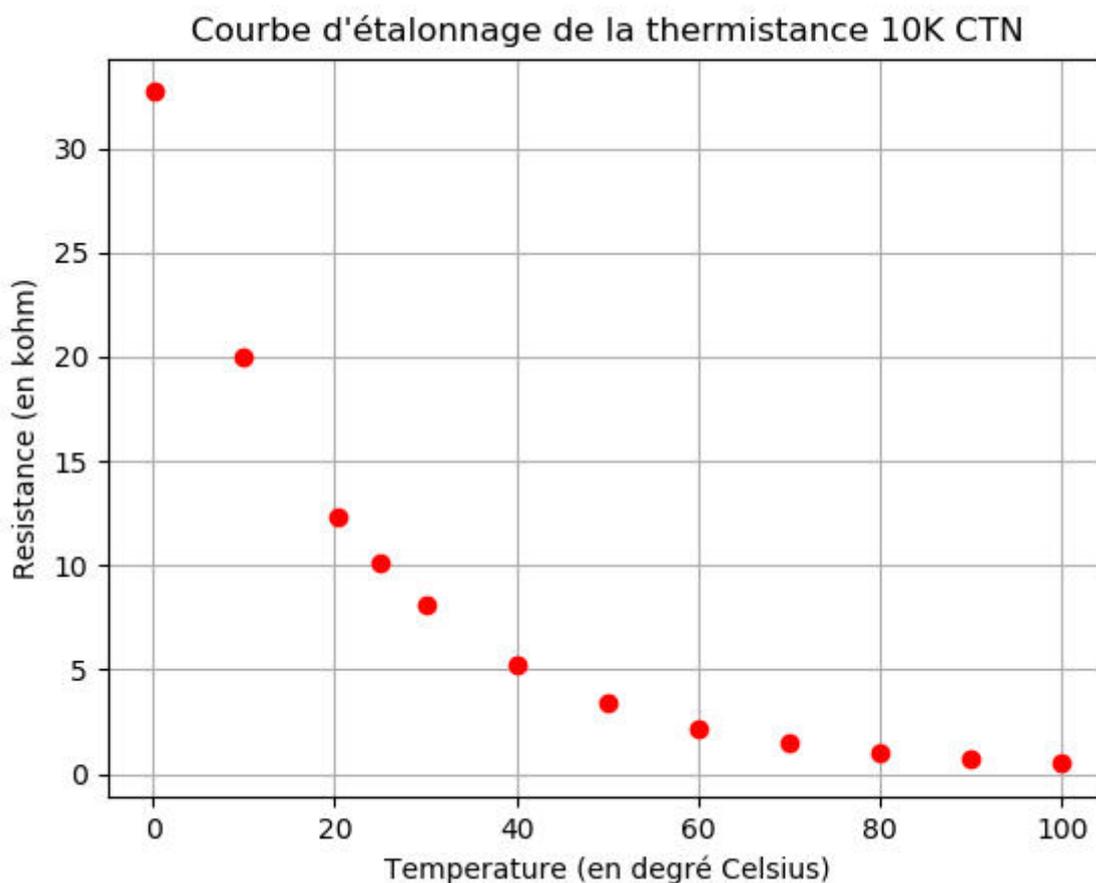
température T (en °C)	8,6	10	20	30	40	50	60	70	80	90	100
résistance R_{th} (en k Ω)	22.0	20.0	13.2	7.9	5.0	3.3	2.2	1.5	1.1	0.8	0.5

Courbe d'étalonnage de la thermistance :

Sur une feuille de papier millimétré, trace la courbe d'étalonnage $R_{th} = f(T)$.

Ou bien avec le langage de programmation PYTHON, trace la courbe d'étalonnage $R_{th} = f(T)$

Corrigé PARTIE A (suite)

Programme PYTHON pour tracer la courbe d'étalonnage :

```

PyScripter - C:\Users\pbuffer\Desktop\FICHE 2 2nde Temperature\ARDUINO_PYTHON_Fiche_2_Programmes\
Fichier  Edition  Rechercher  Affichage  Projet  Exécuter  Outils  Aide
Nouveau Fichier...
"""
Formation des enseignants lycée 2018-2019
courbe d'étalonnage Thermistance 10K CTN
programme réalisé par patrice.bufferet@ac-amiens.fr
"""

import matplotlib.pyplot as plt
import numpy as np

Tmes=np.array([0.1,10.0,20.4,25.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0])
Rmes=np.array([32.7,20.0,12.3,10.1,8.1,5.2,3.4,2.2,1.5,1.0,0.7,0.5])

plt.plot(Tmes,Rmes,"ro")
plt.title("Courbe d'étalonnage de la thermistance 10K CTN")

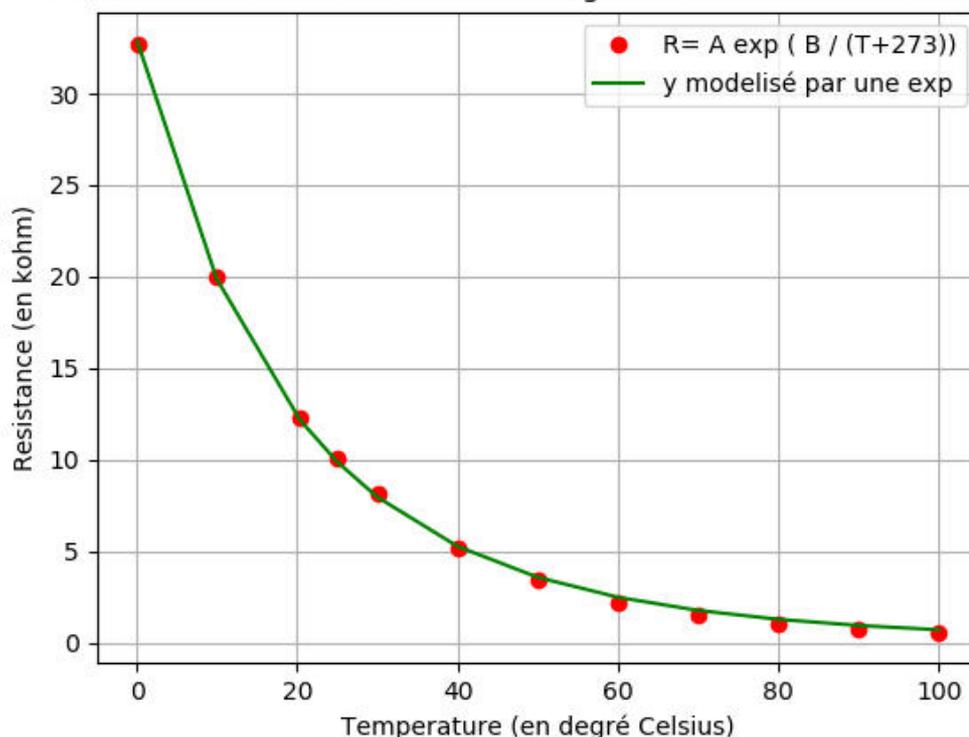
plt.grid()
plt.ylabel("Resistance (en kohm)")
plt.xlabel("Temperature (en degré Celsius)")
plt.show()

```

Complément PARTIE A

pour le PROFESSEUR uniquement (niveau Terminale)

Modélisation de la courbe d'étalonnage de le thermistance 10K CTN



modélisation de la courbe d'étalonnage par une équation du type $R = Ae^{B/(T+273)}$

Programme PYTHON pour tracer la courbe d'étalonnage et sa modélisation :

```

PyScripter - C:\Users\pbuffer\Desktop\FICHE 2 2nde Temperature\ARDUINO_PYTHON_Fiche_2_Programmes\T_LC
Fichier Edition Rechercher Affichage Projet Exécuter Outils Aide
Nouveau Fichier...
"""
Formation des enseignants lycée 2018-2019
courbe d'étalonnage Thermistance 10K CTN
programme réalisé par patrice.bufferet@ac-amiens.fr
Remerciements à jeremy.laforet@utc.fr pour ses conseils
"""

import matplotlib.pyplot as plt
import numpy as np
from scipy.optimize import curve_fit

Tmes=np.array([0.1,10.0,20.4,25.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0])
Rmes=np.array([32.7,20.0,12.3,10.1,8.1,5.2,3.4,2.2,1.5,1.0,0.7,0.5])

def mdl(temp,a,b):
    return a*np.exp(b/(temp+273))

popt, pcov = curve_fit(mdl,Tmes,Rmes)

plt.plot(Tmes,Rmes,"ro",label="R= A exp ( B / (T+273))")
plt.plot(Tmes,mdl(Tmes,"popt"),"g-",label="y modelisé par une exp")
plt.title("Modélisation de la courbe d'étalonnage de le thermistance 10K CTN")

plt.legend()
plt.grid()
plt.ylabel("Resistance (en kohm)")
plt.xlabel("Temperature (en degré Celsius)")
plt.show()

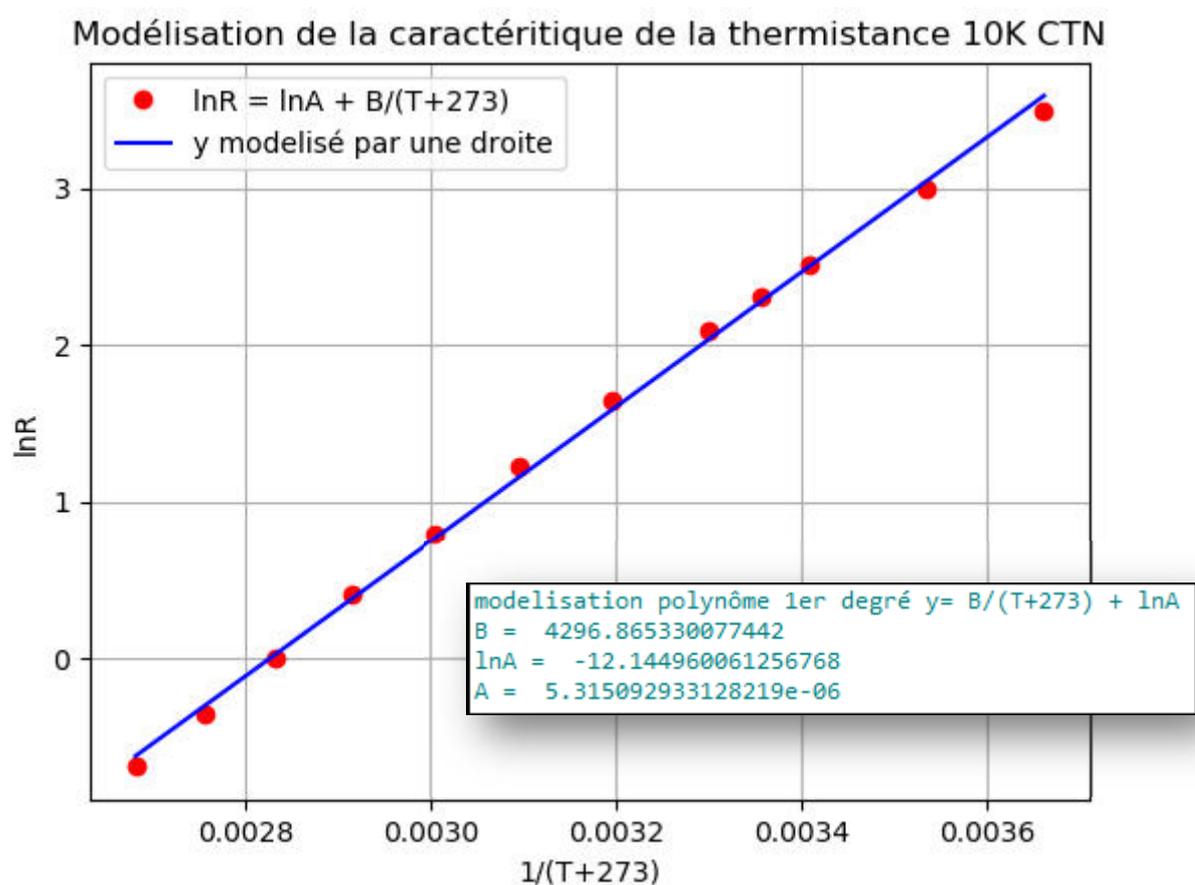
```

Complément PARTIE A

pour le PROFESSEUR uniquement (niveau Terminale)

La modélisation de la courbe d'étalonnage par une exponentielle décroissante ne peut se traiter du point de vue mathématique qu'à partir de la classe de Terminale.

Néanmoins, ce complément d'information est intéressant à ce niveau de l'activité.



Linéarisation de la modélisation de la courbe d'étalonnage (page précédente)

par une équation du type $\ln R = \ln A + \frac{B}{T+273}$

Complément PARTIE A

pour le PROFESSEUR uniquement (niveau Terminale)

Programme PYTHON pour tracer la courbe de modélisation linéarisée :

```

PyScripter - C:\Users\pbuffer\Desktop\FICHE 2 2nde Temperature\ARDUINO_PYTHON_Fiche_2_Programmes\T_LC
Fichier  Edition  Rechercher  Affichage  Projet  Exécuter  Outils  Aide
Nouveau Fichier...
"""
Programme réalisé par patrice.bufferet@ac-amiens.fr le 21/02/2019
revu et amélioré par jeremy.laforet@utc.fr le 25/02/2019
Modélisation d'une exponentielle décroissante du type  $R = A \exp ( B / (T+273))$ 
en passant par  $\ln R = \ln A + B/(T+273)$ 
La console Python donne après modélisation  A = 6,0.10^(-6)  et B = 4264
"""
import matplotlib.pyplot as plt
import numpy as np
from scipy.optimize import curve_fit

Tmes=np.array([0.1,10.0,20.4,25.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0])
Rmes=np.array([32.7,20.0,12.3,10.1,8.1,5.2,3.4,2.2,1.5,1.0,0.7,0.5])
T=1/(Tmes+273)
R=np.log(Rmes)
x=T
y=R

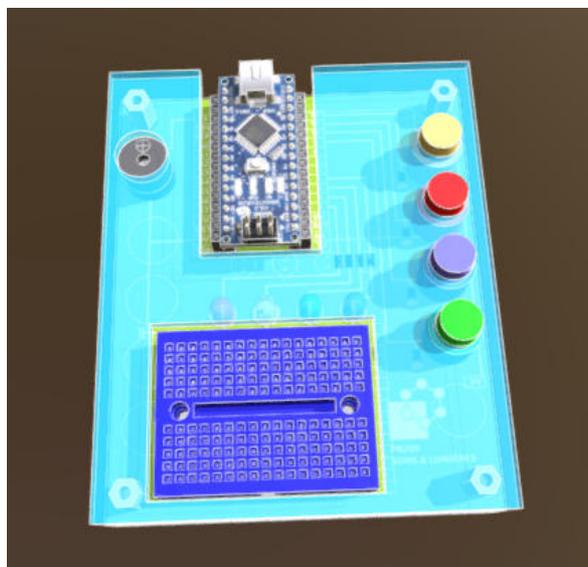
modelisation=np.polyfit(x,y,1)
print("modélisation polynôme 1 degré y=bx+c", modelisation)
print("b = ",modelisation[0])
print("c = ",modelisation[1])
print("modélisation polynôme 1er degré y= B/(T+273) + lnA", modelisation)
print("B = ",modelisation[0])
print("lnA = ",modelisation[1])
print("A = ",np.exp(modelisation[1]))
ymod=modelisation[0]*x+modelisation[1]

plt.plot(x,y,"ro",label="lnR = lnA + B/(T+273)")
plt.plot(x,ymod,"b-",label="y modélisé par une droite")
plt.title("Modélisation de la caractéristique de la thermistance 10K CTN")
plt.legend()
plt.grid()
plt.xlabel("1/(T+273)")
plt.ylabel("lnR")
plt.show()

```

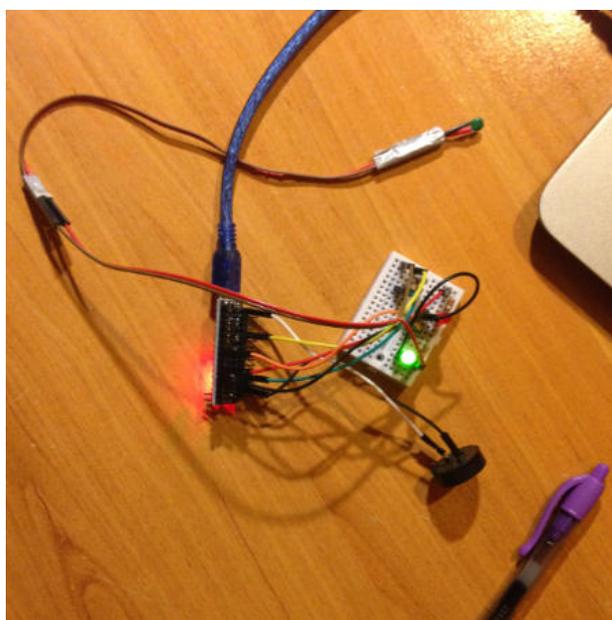
Corrigé PARTIE B

Prendre en photo la maquette PB200 (Pierron) cablée

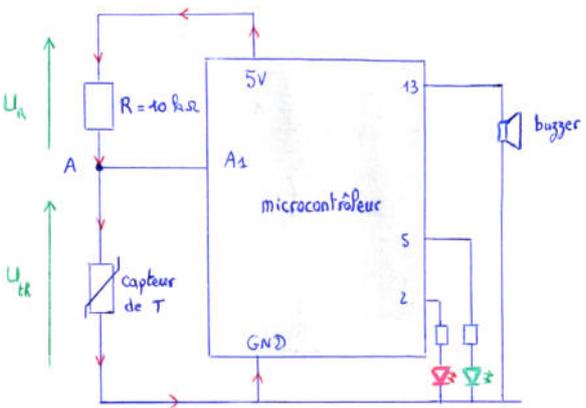
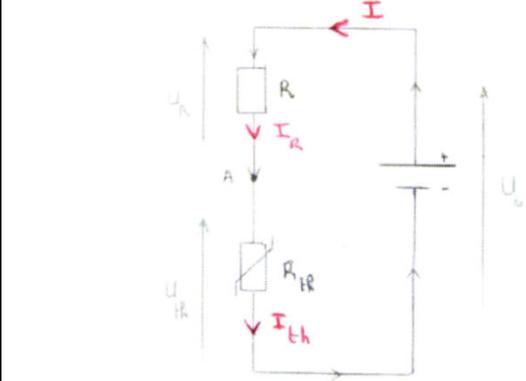


ET / OU

Un autre dispositif équivalent préparé par le professeur



Corrigé PARTIE C

<p style="text-align: center;">Schéma électrique de l'alarme à incendie</p> 	<p style="text-align: center;">Schéma électrique équivalent, seule la maille contenant le capteur est représentée</p>  <div style="border: 1px solid red; padding: 5px; width: fit-content; margin: 10px auto;"> $U_{th} = U_G \times \frac{R_{th}}{R + R_{th}}$ </div>
---	--

Remarque très importante : Le courant électrique ne circule pas dans la branche A-A7. L'intensité du courant électrique qui circule dans la résistance $R=10\text{ k}\Omega$ est donc la même que celle qui circule dans la thermistance.

- 1) Lorsque le capteur est à la température ambiante, mesurez les tensions U_R , U_{th} et U_G et vérifiez la loi des mailles.

Avec un voltmètre, les mesures directes sur le circuit donnent $U_G = 4,64\text{ V}$
 $U_R = 2,11\text{ V}$ $U_{th} = 2,53\text{ V}$. On vérifie la loi des mailles $U_G = U_R + U_{th}$ (1)

- 2) À l'aide de la loi d'Ohm et de la loi d'unicité du courant électrique dans un circuit en série, établir la relation entre la tension U_{th} aux bornes de la thermistance, la tension U_R et les deux valeurs des résistances R_{th} et R .

Loi d'Ohm $U_R = R I_R$
 Loi d'ohm $U_{th} = R_{th} I_{th}$
 Loi d'unicité $I = I_R = I_{th}$ } \Rightarrow $U_{th} = U_R \frac{R_{th}}{R}$ (2)

- 3) À l'aide de la loi des mailles, en déduire la relation écrite en rouge sous le schéma électrique.

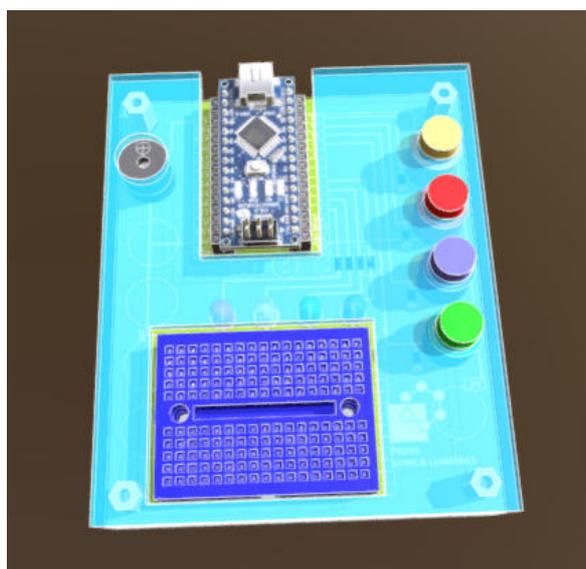
(1) } \Rightarrow $U_{th} = U_G \frac{R_{th}}{R + R_{th}}$ (3)
 (2)

APPELEZ LE PROFESSEUR POUR VERIFIER VOTRE TRAVAIL

Corrigé PARTIE D

EXPÉRIENCE : Alimente ton microcontrôleur et mesure avec un voltmètre la tension électrique entre sa borne notée « 5V » et sa masse notée « GND ». Note la valeur mesurée $U_G = U_{\max} = \dots\dots$ V c'est la tension maximale du microcontrôleur.

Prendre en photo la maquette PB200 (Pierron) cablée avec le voltmètre



ET / OU tout autre dispositif équivalent préparé par le professeur

Avec un microcontrôleur Arduino™ de type Nano,

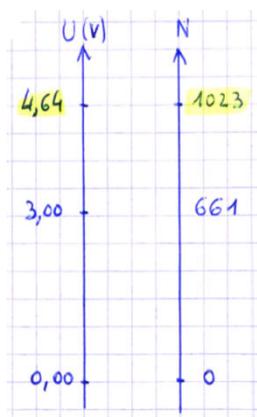
La tension minimale $U_{\min} = 0,00$ V sur une entrée analogique est codée par le nombre **N=0**

La tension maximale $U_{\max} = 4, \dots$ V sur une entrée analogique est codée par le nombre **N=1023**

Il y a une **relation de proportionnalité** entre la tension analogique U et la valeur numérique N mesurée par le microcontrôleur au niveau des entrées analogiques.

Par exemples :

- Si la tension mesurée au niveau de l'entrée analogique A1 vaut $U = 3,00$ V alors, le nombre correspondant à cette tension est $N = 1023 * 3,00 / 4, \dots = \dots$
- Si la valeur numérique lue par le microcontrôleur au niveau d'une entrée analogique est $N = 224$, alors la valeur de la tension électrique correspondante est $U = 4, \dots * 224 / 1023 = \dots$ V.



Corrigé PARTIE E

- D'après la courbe d'étalonnage (partie A), quelle est la valeur de la résistance R_{th} lorsque la température est égale à 35 °C ?

D'après la courbe d'étalonnage $R_{th} = 6 \text{ k}\Omega$

- D'après les résultats des lois de la physique (partie C), en déduire la valeur de la tension U_{th} aux bornes de la thermistance lorsque la température est égale à 35 °C ?

D'après (3) page 7 $U_{th} = 4,64 \times \frac{6}{10+6}$ $U_{th} = 1,74 \text{ V}$

- D'après la partie C, en déduire la valeur numérique N mesurée par le microcontrôleur lorsque la température est égale à 35°C ?

D'après page 8 $N = 1023 \times \frac{1,74}{4,64}$ $N = 384$

Voici un exemple de programme (réalisé avec le logiciel mBlock Version3) à téléverser dans le dispositif pour faire fonctionner l'alarme.

D'après les calculs précédents, par quel nombre N faut-il remplacer le nombre 500 dans ce programme pour que l'alarme se déclenche à partir de 35 °C ?

384

ANALYSE DU PROGRAMME :

D'après le programme proposé ci-dessus et le schéma du dispositif d'alarme page 6, décrivez en quelques lignes ce qu'il va se passer lorsque vous téléverserez ce programme dans le dispositif :

Si la température est supérieure à 35 °C ($N < 384$) alors

- . la DEL verte s'éteint.
- . la DEL rouge clignotte et un signal sonore retentit.

sinon

- . la DEL verte est allumée.
- . la DEL rouge est éteinte.

ÉCRIVEZ AVEC MBLOCK LE PROGRAMME CI-DESSUS (CORRIGÉ AVEC LA BONNE VALEUR DE N) ET TÉLÉVERSEZ-LE DANS LE DISPOSITIF.

VÉRIFIEZ LE BON FONCTIONNEMENT DE L'ALARME AVEC DE L'EAU CHAUDE OU UN SÈCHE-CHEVEUX PAR EXEMPLE.

Corrigé PARTIE E (complément)

Programme Arduino de l'alarme à incendie
pour ceux qui préfèrent programmer en langage Arduino plutôt qu'avec mBlock

Alarme à incendie en langage Arduino

```

capteur_N_tensionU_et_alarme §
// Alarme à incendie qui se déclenche
// lorsque T>20°C environ soit N=500

void setup() {
  pinMode(A1, INPUT);
  pinMode(5, OUTPUT);
  pinMode(2, OUTPUT);
  pinMode(13, OUTPUT);
}

void loop() {
  if(analogRead(A1) < (500)){
    digitalWrite(5, LOW);
    digitalWrite(2, HIGH);
    tone(13, 650, 500);
    delay(500);
    digitalWrite(2, LOW);
    tone(13, 750, 500);
    delay(500);
  }
  else{
    digitalWrite(5, HIGH);
    digitalWrite(2, LOW);
  }
}

```

Alarme à incendie en langage mBlock

```

Arduino - générer le code
répéter indéfiniment
si la valeur sur la broche Analogique 1 < 500 alors
  mettre l'état logique de la broche 5 à bas
  mettre l'état logique de la broche 2 à haut
  jouer un son sur la broche 13 : note 650 fréquence un demi
  mettre l'état logique de la broche 2 à bas
  jouer un son sur la broche 13 : note 750 fréquence un demi
sinon
  mettre l'état logique de la broche 5 à haut
  mettre l'état logique de la broche 2 à bas

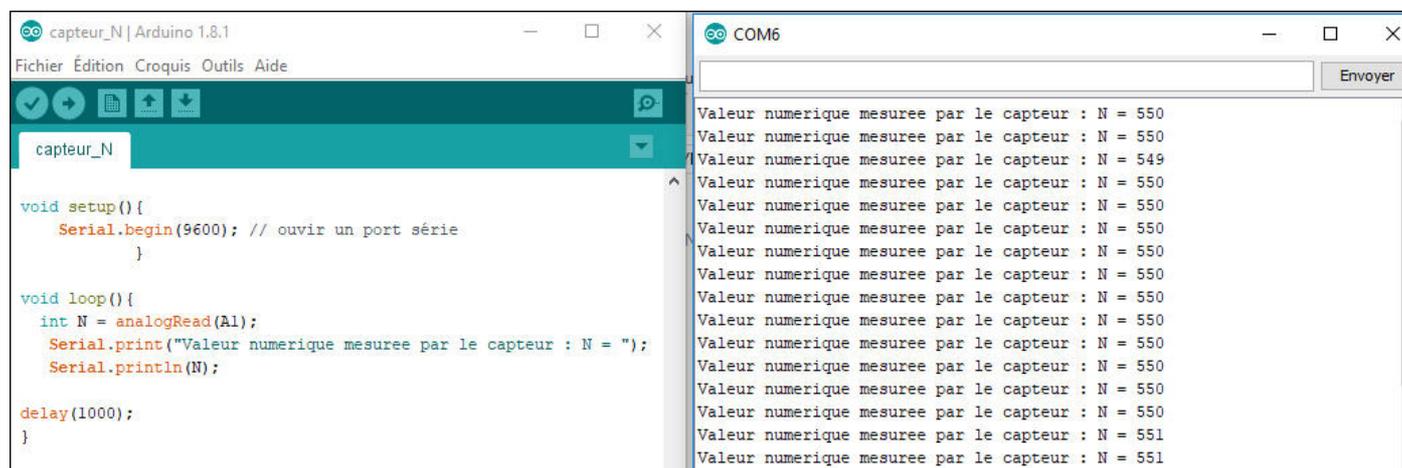
```

Complément n°1

MONITEUR SÉRIE

Objectif : Afficher les valeurs de N et de U_{th} sur l'écran de l'ordinateur

Les valeurs de U_{th} et N mesurées ou calculées dans cette séquence peuvent s'afficher sur l'écran de l'ordinateur. Pour cela il suffit d'écrire le petit programme suivant :



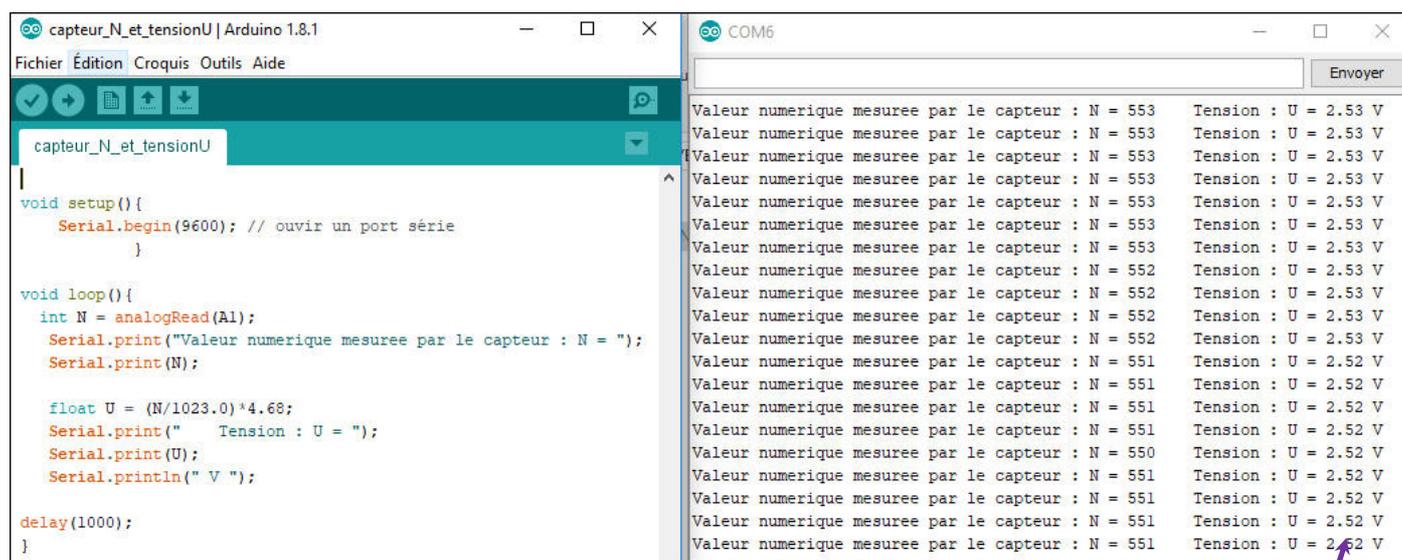
```

void setup() {
  Serial.begin(9600); // ouvrir un port série
}

void loop() {
  int N = analogRead(A1);
  Serial.print("Valeur numerique mesuree par le capteur : N = ");
  Serial.println(N);

  delay(1000);
}

```



```

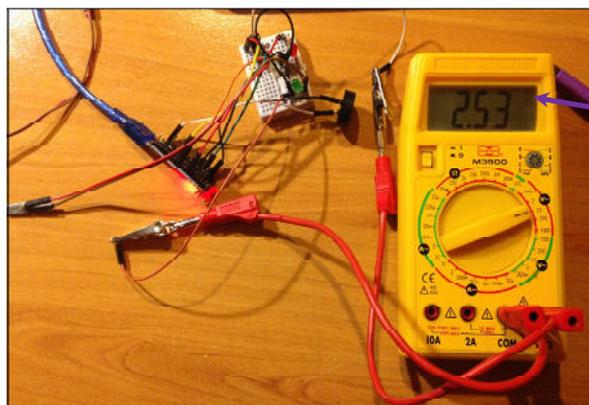
void setup() {
  Serial.begin(9600); // ouvrir un port série
}

void loop() {
  int N = analogRead(A1);
  Serial.print("Valeur numerique mesuree par le capteur : N = ");
  Serial.println(N);

  float U = (N/1023.0)*4.68;
  Serial.print("    Tension : U = ");
  Serial.print(U);
  Serial.println(" V ");

  delay(1000);
}

```



Complément n°2

AFFICHEUR LCD

Objectif : Afficher les valeurs de N et de U_{th} sur un afficheur LCD

Les valeurs de U_{th} et N mesurées ou calculées dans cette séquence peuvent s’afficher sur un afficheur LCD. Pour cela il suffit d’écrire le petit programme suivant :

```

capteur_N_ecranLCD $

// Formation Compétences numériques - Réforme Lycée 2019 -
// Programme et maquette réalisés par patrice.buffet@ac-amiens.fr
// Maquette PB200 :
// Afficher la valeur de N d'un capteur sur un écran LCD ANNEXE.

#include <LiquidCrystal.h>
LiquidCrystal EcranLCD(12, 11, 5, 4, 3, 2);

void setup() {
  EcranLCD.begin(16, 2);
}

void loop() {
  int N = analogRead(A1);

  EcranLCD.clear();
  EcranLCD.setCursor(0, 0);
  EcranLCD.print(" nombre N = ");
  EcranLCD.print(N);
  delay(1000);
}

```



```

capteur_N_et_tensionU_ecranLCD $

// Formation Compétences numériques - Réforme Lycée 2019 -
// Programme et maquette réalisés par patrice.buffet@ac-amiens.fr
// Maquette PB200 :
// Afficher la valeur de N et Uth d'un capteur sur un écran LCD ANNEXE.

#include <LiquidCrystal.h>
LiquidCrystal EcranLCD(12, 11, 5, 4, 3, 2);

void setup() {
  EcranLCD.begin(16, 2);
}

void loop() {
  int N = analogRead(A1);
  float U = N*4.68/1023;

  EcranLCD.clear();
  EcranLCD.setCursor(0, 0);
  EcranLCD.print(" nombre N = ");
  EcranLCD.print(N);
  EcranLCD.setCursor(0, 1);
  EcranLCD.print("tension U=");
  EcranLCD.print(U);
  EcranLCD.print("V");
  delay(1000);
}

```

