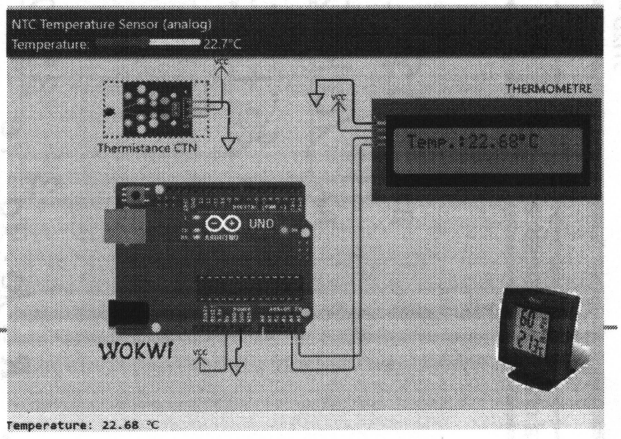
Lycée Polyvalent Pierre Emile Martin	**Traitement algorithmique**	académie d'Orléans-Tours
Microcontrôleur	**Thermomètre**	

CORRECTION

Document réponse du TP **Date** _____

Nom : _____ **Classe** : _____

Prénom : _____ **Groupe** : _____



Mots-clés : capteur analogique, conditionneur de capteur, *configuration des E/S, variables booléennes, entières et réelles, affectation, structures alternatives, programmation en langage C, tests dans la console.*

ACTIVITÉS

1. Chaîne de mesure
   a. Linéariser la réponse d'un capteur
   b. Modéliser une chaîne de mesure
2. Logiciel (Software)
   a. Utiliser des variables de type booléen, entier et réel.
   b. Coder un algorithme dans un langage spécifique.

PRÉREQUIS

Cours d'algorithmique et chaîne de mesure.

MATÉRIELS ET LOGICIELS

Micro-ordinateur, navigateur, simulateur en ligne WOKWI, Arduino Uno, module KY-013, afficheur graphique.

CONDITION DE RÉALISATION

1 élève par poste informatique

PAGE DE PRÉSENTATION DU TP

https://bit.ly/3vkIAGS

RESSOURCES

Wiki « Microcontrôleur – Entrées, sorties (GPIO) » https://bit.ly/3E2zQ97
Conditionnement des capteurs https://bit.ly/3tA3X6y

Appréciation

_____

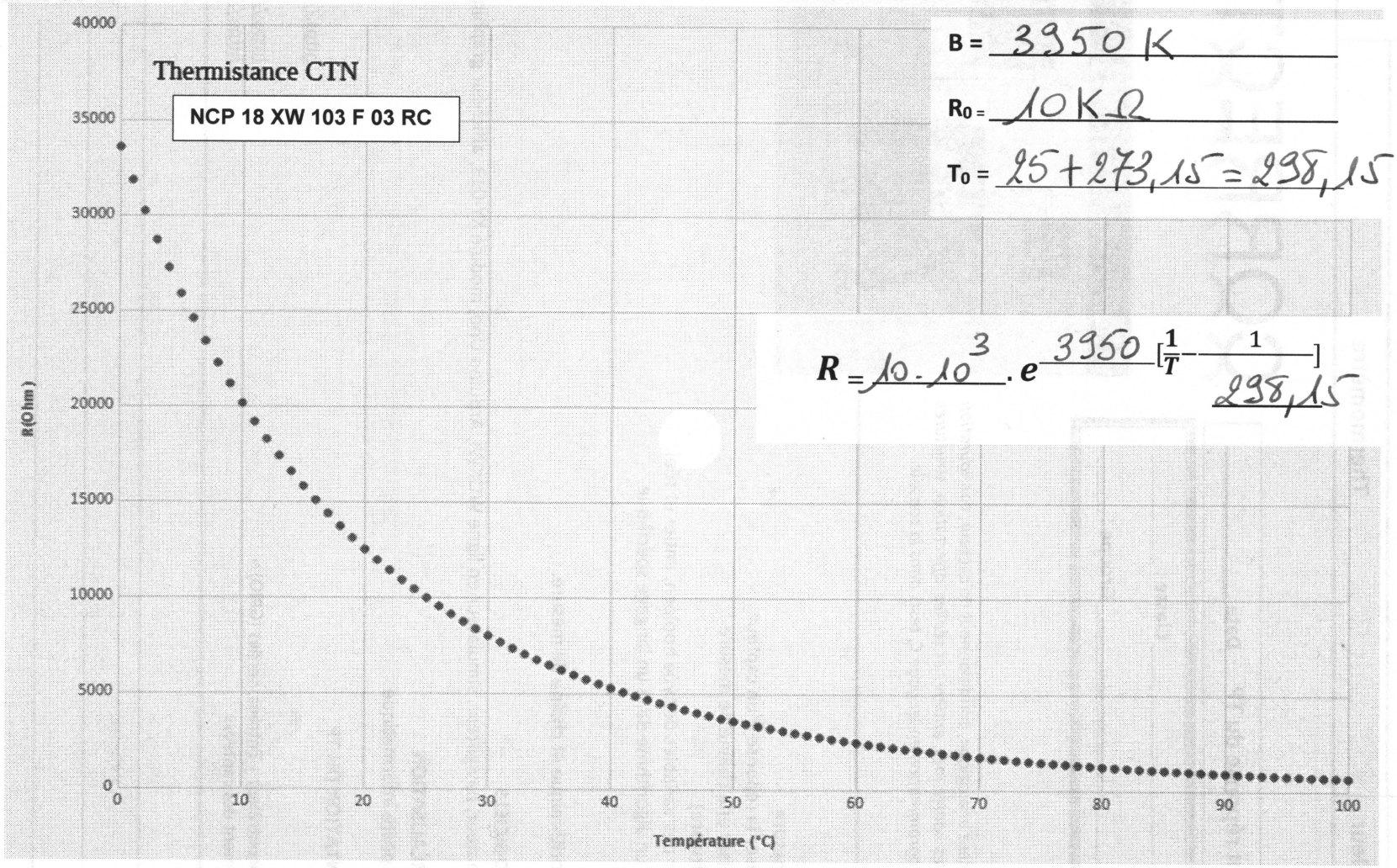
_____

_____

Sujet

DR1 – La thermistance CTN dans la chaîne de mesure

Q1.



B = 3950 K

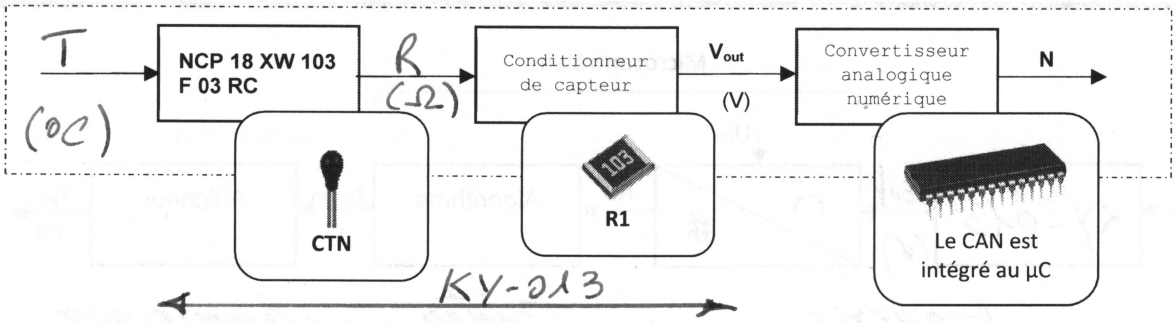
R_0 = 10 KΩ

T_0 = 25 + 273,15 = 298,15 K

$$R = 10 \cdot 10^3 \cdot e^{3950\left[\frac{1}{T} - \frac{1}{298,15}\right]}$$

Q2.

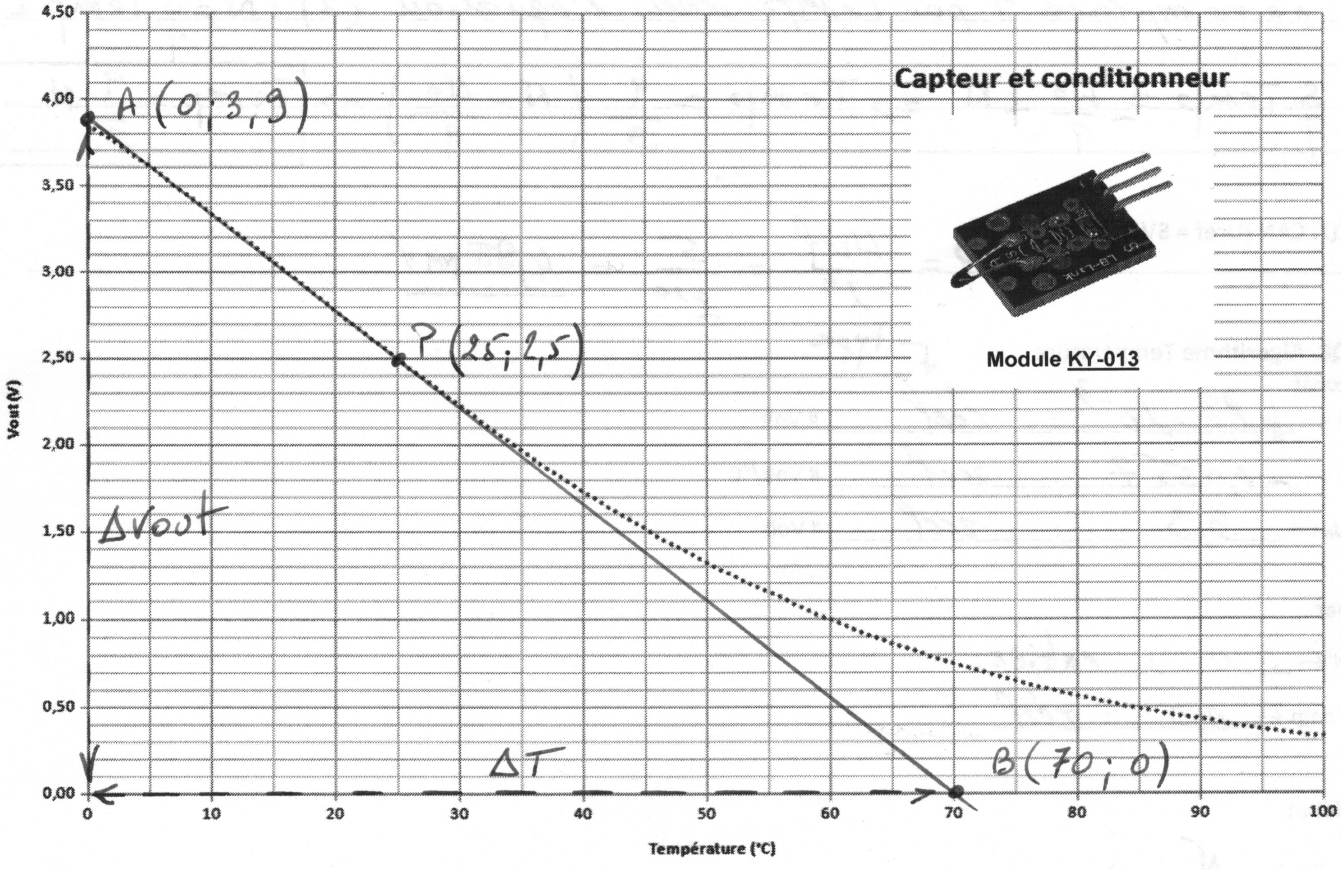
a)



b) La thermistance CTN est un capteur passif car elle ne fonctionne pas en générateur mais en récepteur.

DR2 – Linéarisation du modèle "CTN intégré à un conditionneur"

Q3.



Équation de la tangente au point P (25 ; 2,5)

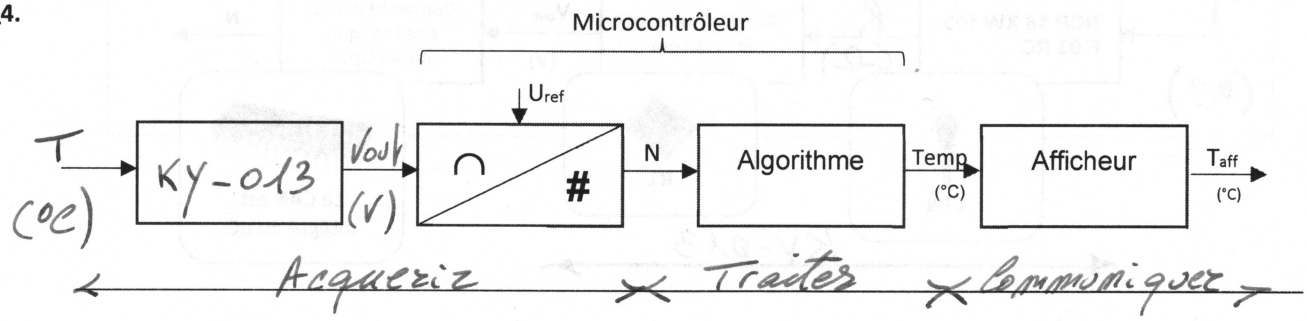
a) $S = \frac{\Delta V_{out}}{\Delta T} = \frac{0 - 3,9}{70 - 0} = -0,0557\ V/°C$ $\qquad V_0 = 3,9\ V$

$$\boxed{V_{out\,(V)} = -0,0557 \cdot T_{(°C)} + 3,9}$$

b) Le cahier des charges est respecté car $T \in [0, 40°C]$ ±1°C

DR3 – SOLUTION 1 – Modèle linéarisé de la chaîne de mesure

Q4.



Q5.

$N = V_{out}/q$
$V_{out} = S.T + U_0$ } $N = \frac{ST + U_0}{q}$ => $\boxed{N = \frac{S}{q}T + \frac{U_0}{q}}$ (1)

Q6. On souhaite Temp = T

on remplace T par Temp dans l'equation (1) : $N = \frac{S}{q}Temp + \frac{U_0}{q}$

$\frac{S}{q}Temp = -\frac{U_0}{q} + N$ => $Temp = \frac{q}{S}\left(N - \frac{U_0}{q}\right)$ => $\boxed{Temp = \frac{q}{S}N - \frac{U_0}{q}}$

Q7. CAN (Uref = 5V, n = 10)

$q = \frac{U_{ref}}{2^n} = \frac{5}{2^{10}} = \underline{4{,}88\ mV}$

Q8. Algorithme Température

const

type

q ← $4{,}88 \cdot 10^{-3}$: réel # Volt

S ← $-0{,}057$: réel # Volt/°C

U_0 ← 3,9 : réel # Volt

var

N ← 0 : entier

Temp ← 0,0 : réel

début

lire (N)

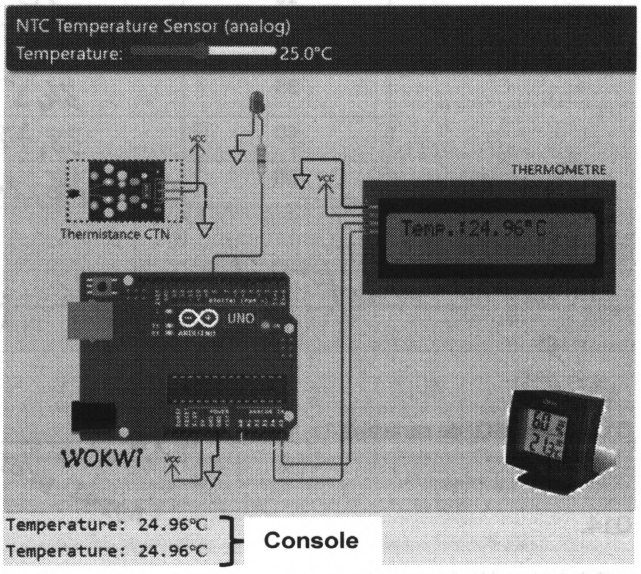
Temp ← $\frac{q}{S}N - \frac{U_0}{S}$ # Application numérique

écrire (Temp) # Temp = − 0,0876 . N + 70

fin

DR4 – PROGRAMMATION

Q9, Q10, Q11




```
// Bibliothèque
#include <LiquidCrystal_I2C.h>

// Paramètres de l'objet lcd
#define I2C_ADDR    0x27
#define LCD_COLUMNS 16
#define LCD_LINES   2

// CTN
const float BETA = 3950;
const float S = - 0.0557 ;
const float U0 = 3.9 ;

// CAN
const float q = 5/1024 ;

// Création de l'objet lcd
LiquidCrystal_I2C lcd(I2C_ADDR, LCD_COLUMNS, LCD_LINES);

void setup() {
  // Initialisation de la console
  Serial.begin(9600);

  // Initialisation du LCD
  lcd.init(); // A placer en premier
  // Commande du rétroéclairage
  lcd.backlight();
  lcd.print("Thermometre");
  delay(3000);
}

// Programme principal
void loop() {
  // Lire N
  int N = analogRead(A0) ;

// Traiter
  // Modèle linéarisé
  float Temp = (q*N - U0)/S ;

  // Écrire dans la console
Serial.print("Température") ;
Serial.print(Temp) ;
Serial.println("°C") ;

  // Une mesure par seconde
delay(1000);
}
```

Appel prof

Validation du code dans Wokwi Oui ☐ Partiel ☐ Non ☐

Q12. Simulation 1 sous Wokwi (Modèle linéarisé de la chaîne de mesure)

Température (°C)	Température affichée (°C)	Écart = Température – Température affichée (°C)
0	0,85	-0,85
10	10	0
25	25	0
35	34,5	0,5
40	38,78	1,22
50	46,10	3,9

Cahier des charges respecté ?

Non, l'écart est > 1°C pour T = 40°C

Q13. Code ASCII du caractère ° : df16

Appel prof

Validation du code dans Wokwi **Oui** ☐ **Partiel** ☐ **Non** ☐

Q14.

```
void setup() {
  Serial.begin(9600);
 // Configuration de la broche LED
  pinMode(LED, OUTPUT);
// ---------------------------------------------------
// Alarme
  if ((Temp < 10) || (Temp > 40)) {
    digitalWrite(LED, HIGH);
  }
  else{
    digitalWrite(LED, LOW);
  }
```

Appel prof

Validation du code dans Wokwi **Oui** ☐ **Partiel** ☐ **Non** ☐

Q15. Simulation 2 sous Wokwi (Modèle non linéarisé de la chaîne de mesure)

a)

Température (°C)	Température affichée (°C)	Écart = Température – Température affichée (°C)
0	-0,05	0,05
10	10	0
25	24,96	0,04
35	35,02	0,02
40	40,05	0,05
50	50,02	0,02

b) L'écart de la simulation 2 est plus faible que celui de la simulation 1 pour T ∈ [0, 50]