

Projet de Surveillance de Température Automobile avec Interface Qt - Utilisation d'Arduino et de capteurs de température :



préparé par
ONS MEJRI
NAIMA MANSOUR
KHOULOUD OTHMANI
AYA ZIADI
Abdslem Ardhaoi

- Mesurer la température d'un moteur (simulé ou réel) avec Arduino .



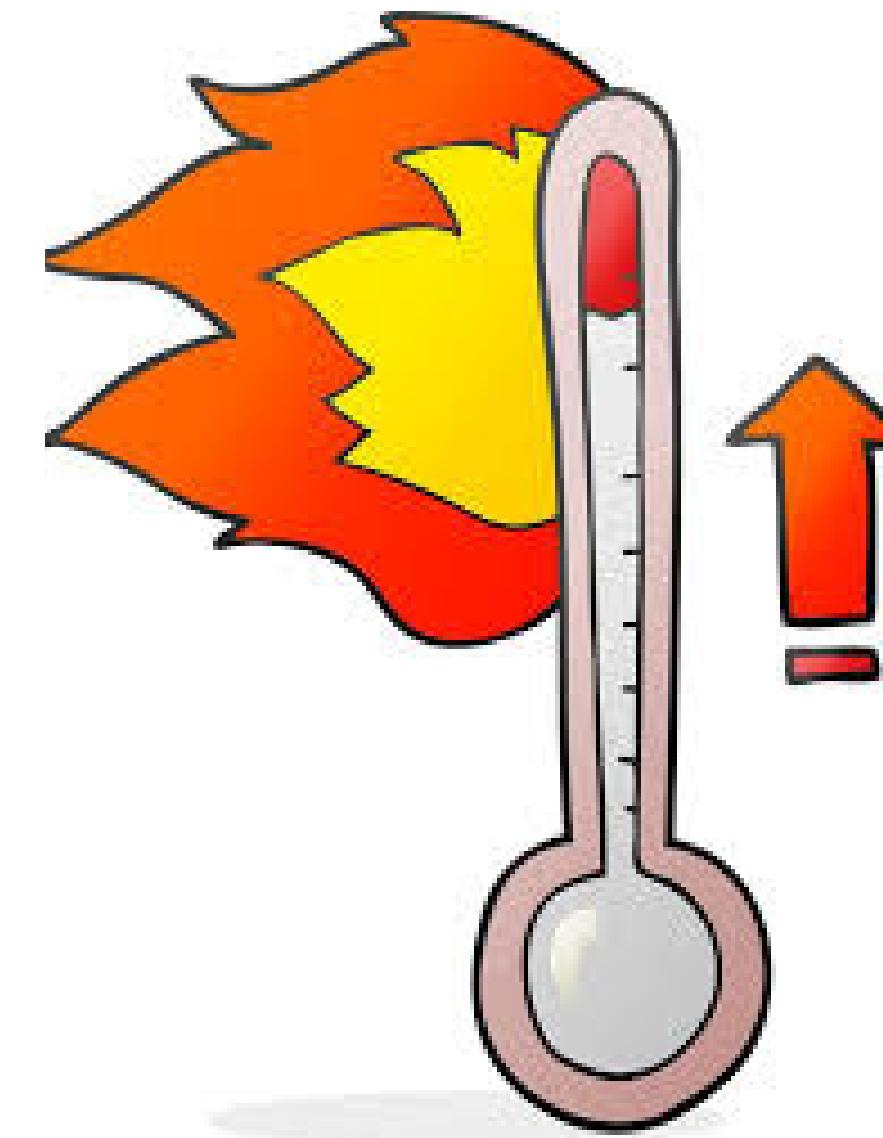
- Afficher les données en temps réel via une interface Qt.

- Gérer des alertes en cas de surchauffe.

Introduction

POURQUOI SURVEILLER LA TEMPÉRATURE D'UN VÉHICULE ?

Dans le monde de l'automobile intelligente, chaque degré compte. Une surchauffe peut endommager le moteur, réduire les performances ou mettre en danger le conducteur. Les systèmes embarqués doivent réagir vite, avec précision et fiabilité.



Introduction

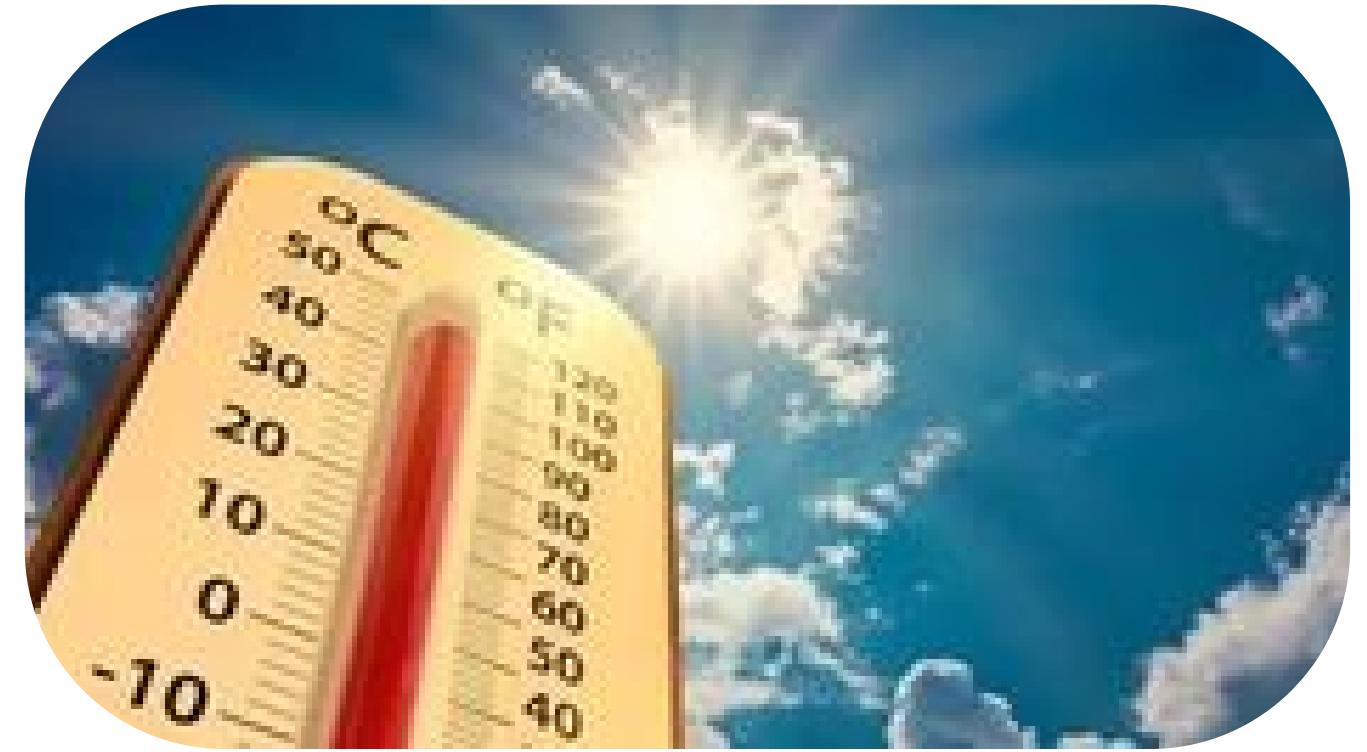
POURQUOI SURVEILLER LA TEMPÉRATURE D'UN VÉHICULE ?

Dégradations mécaniques possibles :

- Détérioration du joint de culasse
- Fonte partielle des pistons
- Dommages au radiateur ou à la pompe à eau

Risques liés à la sécurité :

- Panne soudaine sur la route
- Risque d'incendie moteur
- Altération du comportement du véhicule

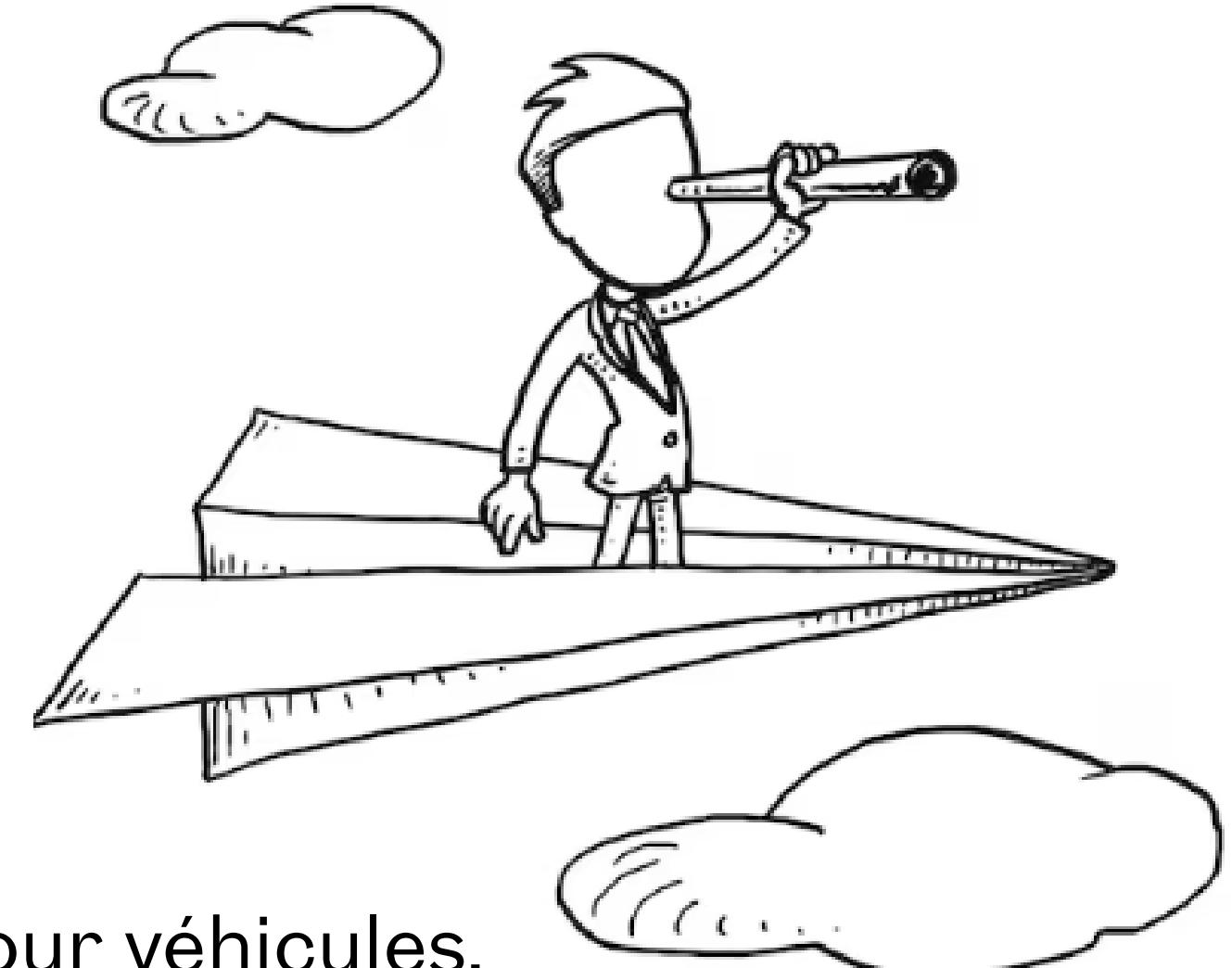


Introduction

NOTRE VISION :

Créer un système intelligent de surveillance thermique pour véhicules, combinant la puissance d'une carte Arduino avec une interface graphique Qt moderne et interactive, pour :

- 📈 Afficher la température en temps réel
- ⚠️ Déetecter les anomalies thermiques
- 💡 Commander des actionneurs automatiquement
- 📅 Enregistrer les données et les visualiser avec style



Matériel et Configuration du Projet :

 Liste du matériel :

- **Carte Arduino (Uno)**

→ **Cœur du système pour la lecture et le contrôle.**

- Capteurs de température : **LM35 (analogique)**

→ **Pour mesurer la température ambiante.**

- Actionneurs : **LED** :

→ **Indicateur d'état .**

- Composants divers :

- **Breadboard, câbles Dupont**

- **Résistances** (surtout pour la LED)



la carte Arduino Uno

Matériel et Configuration du Projet

QU'EST-CE QUE LA CARTE ARDUINO ?

- Arduino est une carte électronique programmable qui permet de contrôler des composants électroniques (capteurs, moteurs, LEDs, etc.).
- Très utilisée pour l'apprentissage, les projets DIY et l'innovation technologique.



- Ce microcontrôleur est idéal pour
- les applications en temps réel
 - Facile à apprendre même pour les débutants.
 - Communauté très active (des milliers de projets partagés).
 - Prix abordable et nombreux modèles adaptés aux besoins.

POURQUOI LA CARTE ARDUINO POUR NOTRE PROJET ?

Haute performance:

Exécution rapide des instructions, idéale pour la lecture fréquente des capteurs

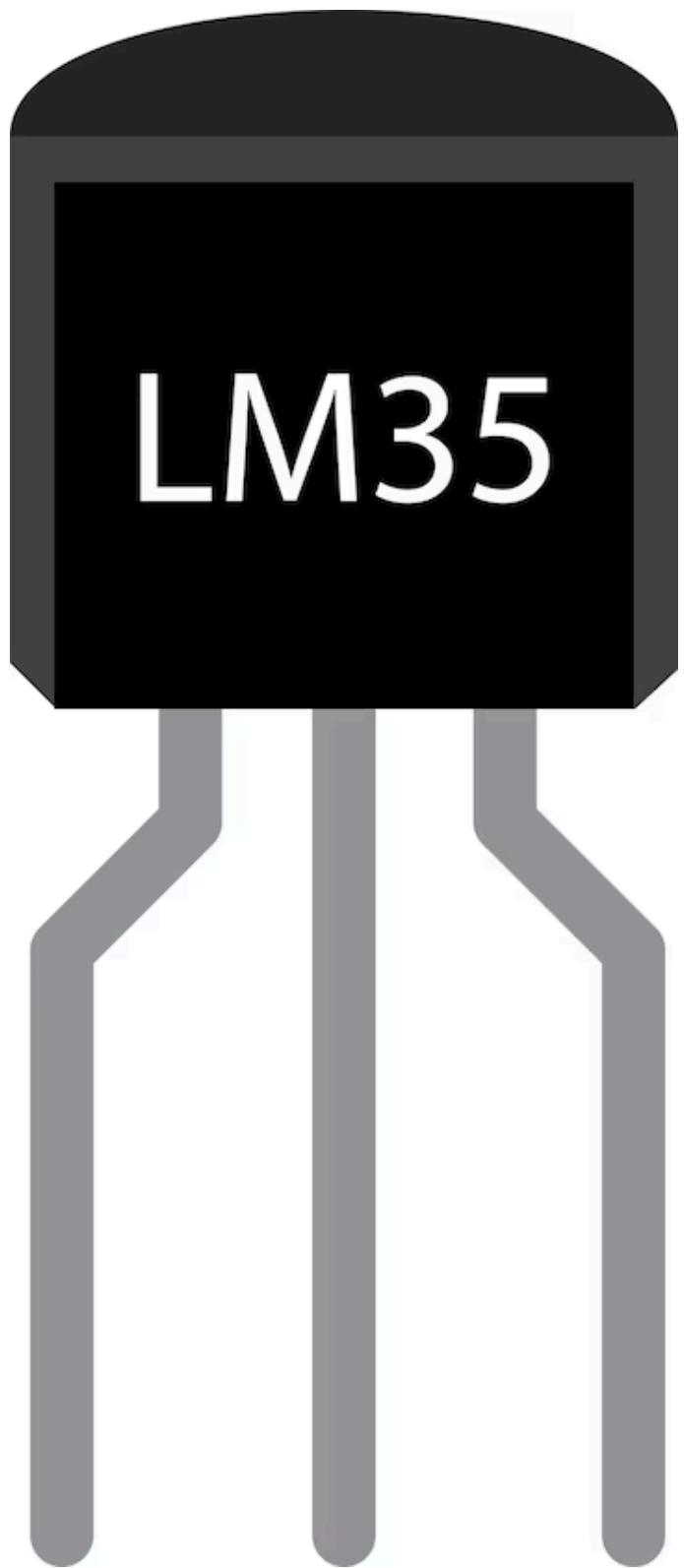
Temps réel :

Utilisation du noyau temps réel pour assurer des calculs rapides et fiables (idéal pour le contrôle HVAC et la télémétrie).

Communication avec l'interface Qt :

Transmission des données collectées via UART/USB vers l'interface Qt.





Capteur LM35

Présentation du capteur LM35

POURQUOI CAPTEUR DE TEMPÉRATURE LM35 POUR NOTRE PROJET ?

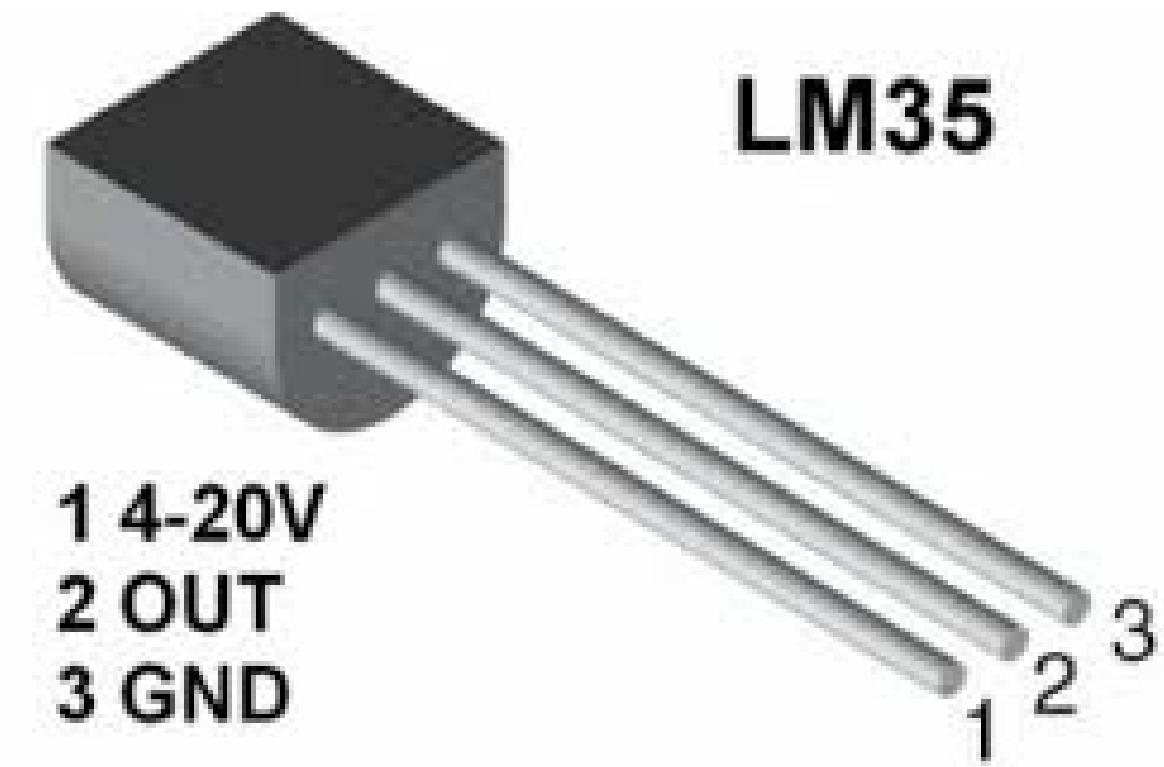
Type de capteur :

Capteur analogique de température

Temps de réponse:

Rapide (moins de 1 seconde pour petits changements)

- Simple à utiliser
- Précis : $\pm 0.5^\circ\text{C}$ typique
- Étalonné en $^\circ\text{C}$
- Ne nécessite aucun composant externe pour fonctionner



Présentation du capteur LM35

🔍 PRINCIPE DE FONCTIONNEMENT :

- Le LM35 est un capteur de température analogique.
- Il fournit une tension de sortie proportionnelle à la température.
- Tension de sortie : 10 mV par °C.

📊 Formule de conversion :

$$\text{Température } (^{\circ}\text{C}) = \left(\frac{\text{valeur_analogique} \times 5.0}{1023} \right) \div 0.01$$

👉 Cela revient à :

$$\text{Température } (^{\circ}\text{C}) = \left(\frac{\text{valeur_analogique} \times 5.0}{1023} \right) \times 100$$



LES LEDS:

Dans notre système, deux LED sont utilisées pour indiquer l'état de la transmission des données : une **LED verte** et une **LED rouge**.



- la LED rouge s'allume lorsque la température dépasse la température seuil .
- La LED verte s'allume lorsque la température est normale .

LES Résistances:

Dans notre système, on a utilisés deux résistances de **220 ohm** .
Pour protéger les leds.



Afficheur LCD

Historique des températures avec le temps et l'état.



Buzzeur

Donne une alerte sonore lorsque la température dépasse
la température seuil



Bouton poussoir

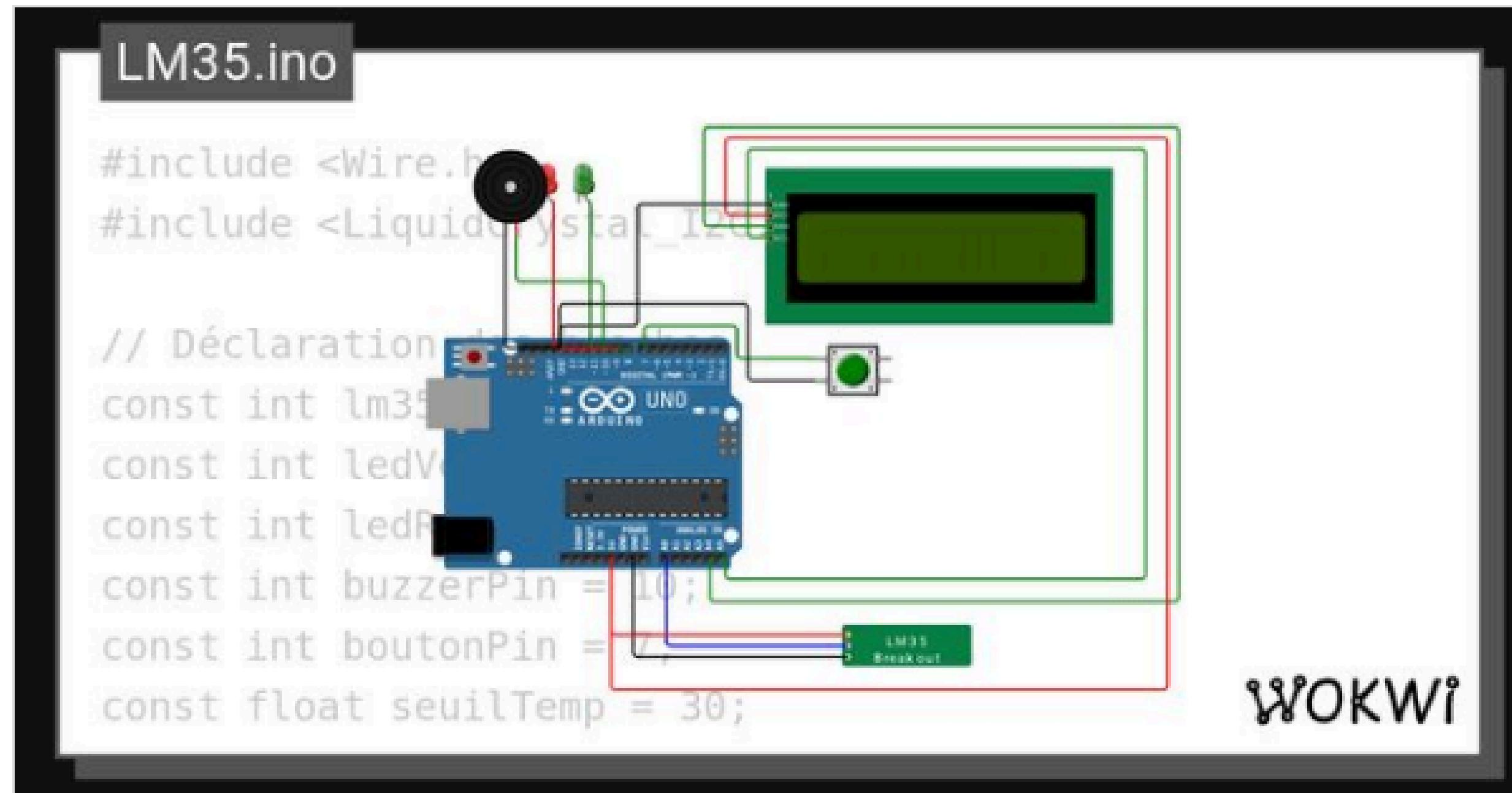
comme interrupteur maître :

- Initialisation sécurisée du système.



Simulation Numérique :

POUR RÉALISER LA SIMULATION NUMÉRIQUE DE NOTRE MONTAGE, NOUS AVONS UTILISÉ **LE LOGICIEL EN LIGNE WOKWI**.

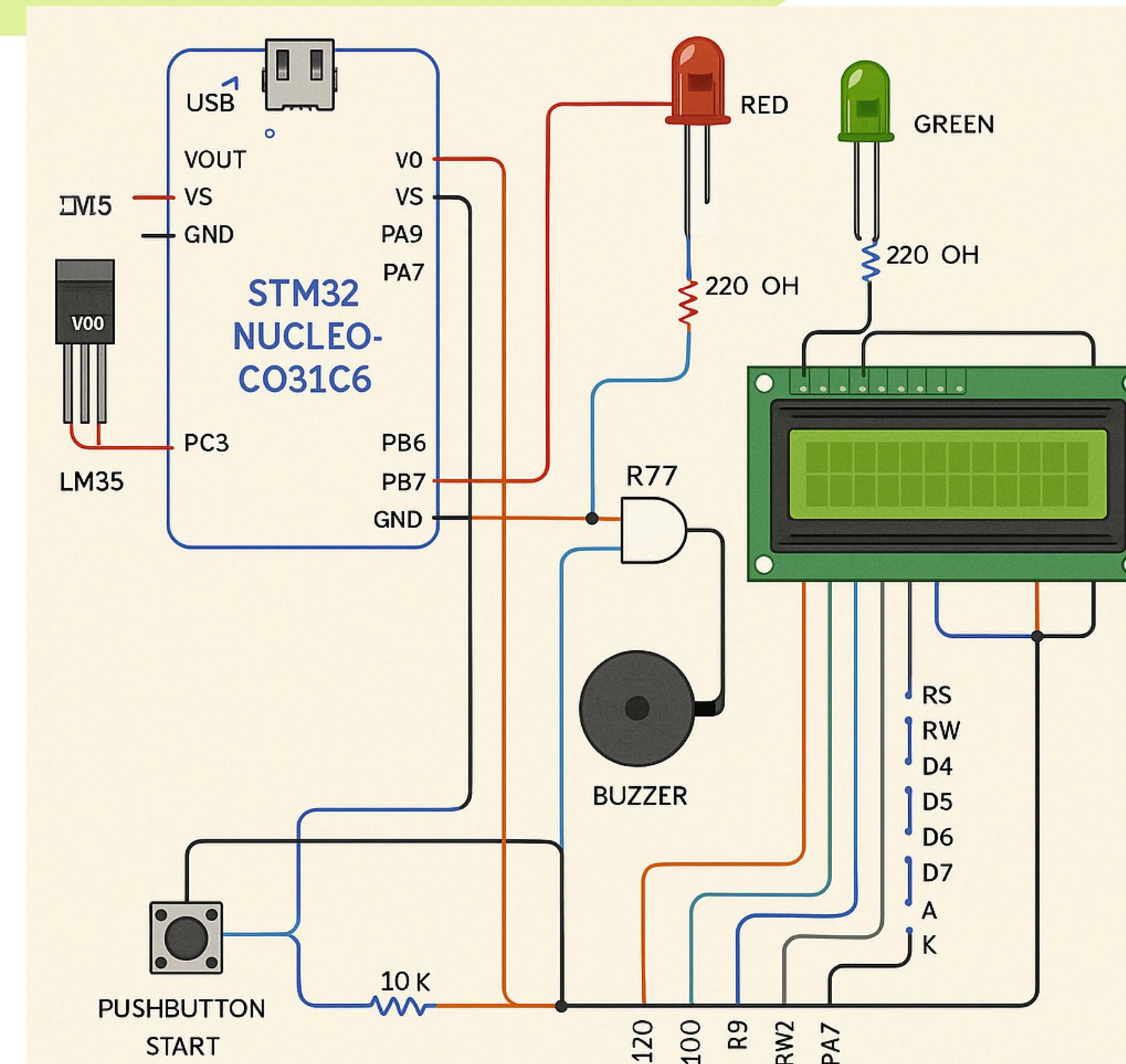


LM35 Copy (2) - Wokwi ESP32, STM32, Arduino Simulator

Run IoT and embedded projects in your browser: ESP32, STM32, Arduino

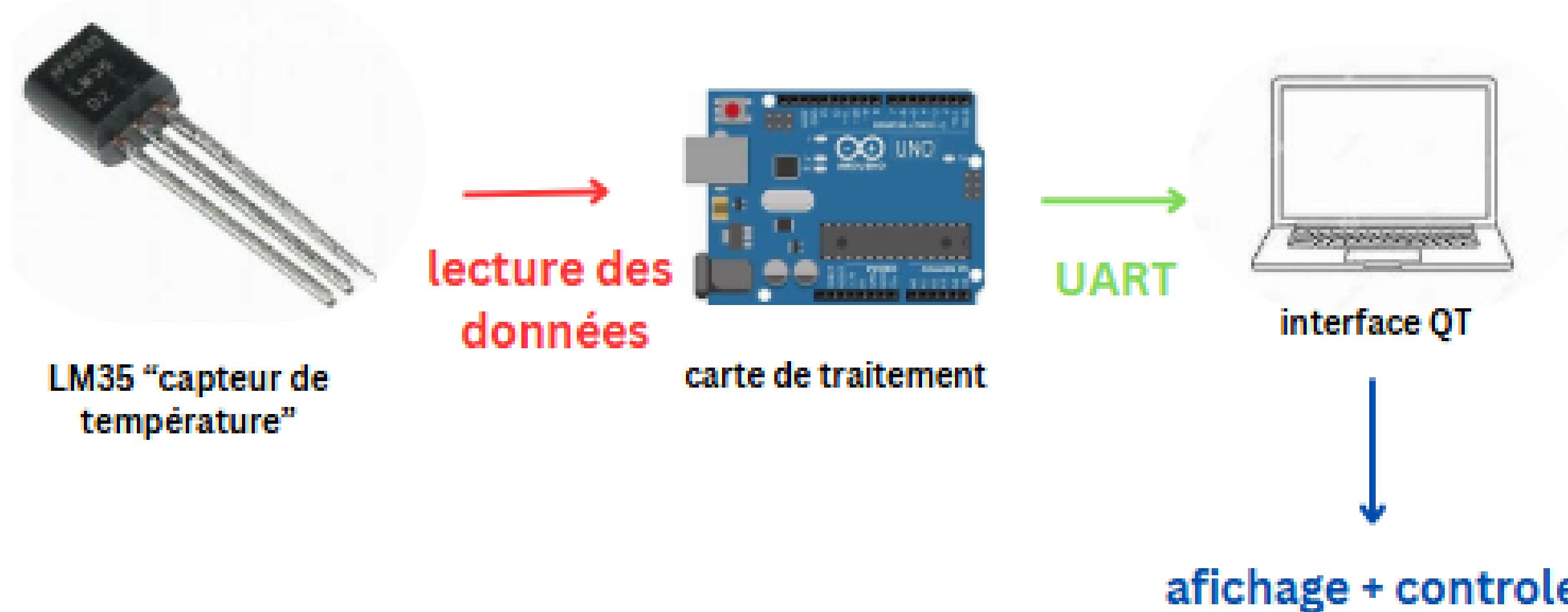
Simulation Numérique :

Dans le cas d'une implémentation utilisant un STM32, le câblage devrait être configuré comme suit :



Architecture du Système

INTÉGRATION ARDUINO ET INTERFACE QT



Interface Graphique Qt

NOTIONS ET FONDAMENTAUX DE L'INTERFACE GRAPHIQUE

Framework C++:

→ Qt est un framework open-source basé sur C++ pour créer des interfaces graphiques modernes.

Widgets et Interface Graphique (GUI):

→ Qt propose des widgets prêts à l'emploi : boutons, sliders, fenêtres...

Qt Designer:

→ Outil visuel pour concevoir des interfaces sans écrire de code.





Objectif de l'interface

Créer un tableau de bord pour afficher et contrôler la température à bord du véhicule.

Technologie utilisée

Qt pour l'interface utilisateur, avec des éléments de design

Composants clés

Affichage de la température en fait la gestion des zones avant et arrière.

Interface Graphique Qt

ACQUISITION ET TRANSMISSION DES DONNÉES DE TEMPÉRATURE

🔍 Acquisition des données:

- 🚗 Température via ADC / 1-Wire / I2C
- 📡 Envoi périodique des mesures
- 🔧 Traitement local

⚠️ Vérification des seuils

📝 Formatage des données en trames simples

📶 Transmission des données à l'interface utilisateur

🔌 UART/USB (câble)



Interface Graphique Qt

VUE GÉNÉRALE DE L'INTERFACE



48.44

 État: Normal

Min: 48.4°C

Max: 48.5°C

Moyenne: 48.4°C

	Heure	Température
1	00:32:03	48.4
2	00:32:10	48.5
3	00:32:10	48.4
4	00:32:18	48.5
5	00:32:18	48.4
6	00:32:18	48.4



Mode sombre



Exporter CSV

Interface Graphique Qt

DESCRIPTION DE L'INTERFACE

● Dashboard

Affichage en temps réel de la température via un compteur numérique (LCD)

Indication de l'état du système (✓ Normal / ⚠ Surchauffe)

Statistiques instantanées : Min, Max, Moyenne

Bouton pour changer de thème (🌙/☀ mode sombre ou clair)

Logo animé en haut de l'écran

Historique

Tableau listant toutes les températures enregistrées avec l'heure
Bouton Exporter CSV pour sauvegarder les données

	Heure	Température (°C)
1	21:40:00	26.3
2	21:40:05	26.5
3	21:40:17	30.7
4	21:40:20	23.8
5	21:40:30	33.3
6	21:40:35	29.1
7	21:40:42	30.0
8	21:40:45	25.3
9	21:40:50	26.7
10	21:40:55	25.4
11	21:41:00	20.3
12	21:41:05	28.5
13	21:41:10	20.4
14	21:41:15	24.6
15	21:41:34	32.7
16	21:41:40	34.0
17	21:41:40	24.9
18	21:47:13	30.0
19	21:47:15	21.6

Historique

Bouton Exporter CSV permettant de sauvegarder les données sous forme de fichier Excel

	A	B	C
1	Heure, Température		
2	21:40:00, 26.3		
3	21:40:05, 26.5		
4	21:40:17, 30.7		
5	21:40:20, 23.8		
6	21:40:30, 33.3		
7	21:40:35, 29.1		
8	21:40:42, 30.0		
9	21:40:45, 25.3		
10	21:40:50, 26.7		
11	21:40:55, 25.4		
12	21:41:00, 20.3		
13	21:41:05, 28.5		
14	21:41:10, 20.4		
15	21:41:15, 24.6		
16	21:41:34, 32.7		
17	21:41:40, 34.0		
18	21:41:40, 24.9		
19	21:47:13, 30.0		



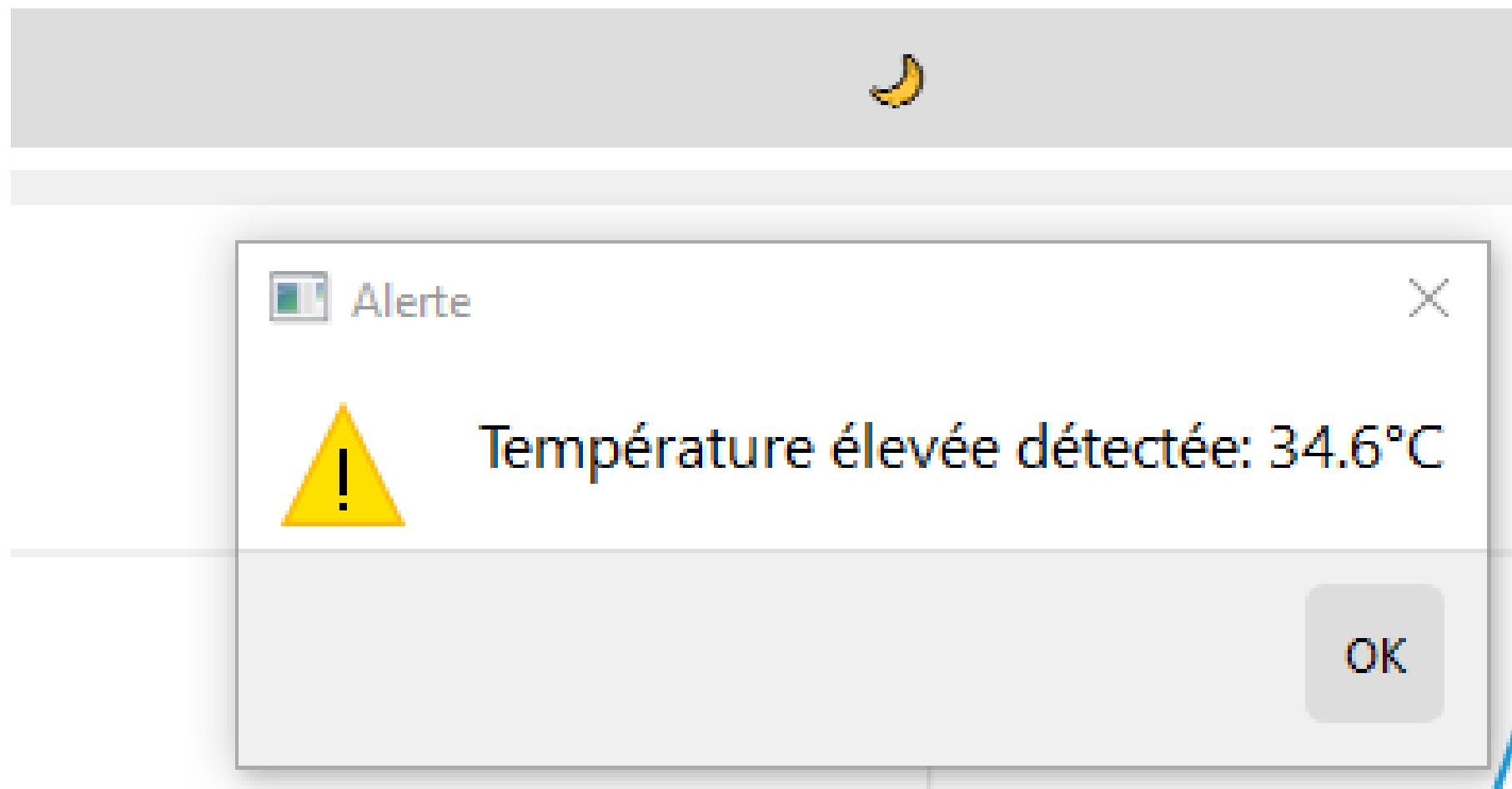
Alerte en cas de surchauffe

Notification immédiate si la température dépasse un seuil critique
Message d'avertissement affiché avec l'icône et la température mesurée

34.62

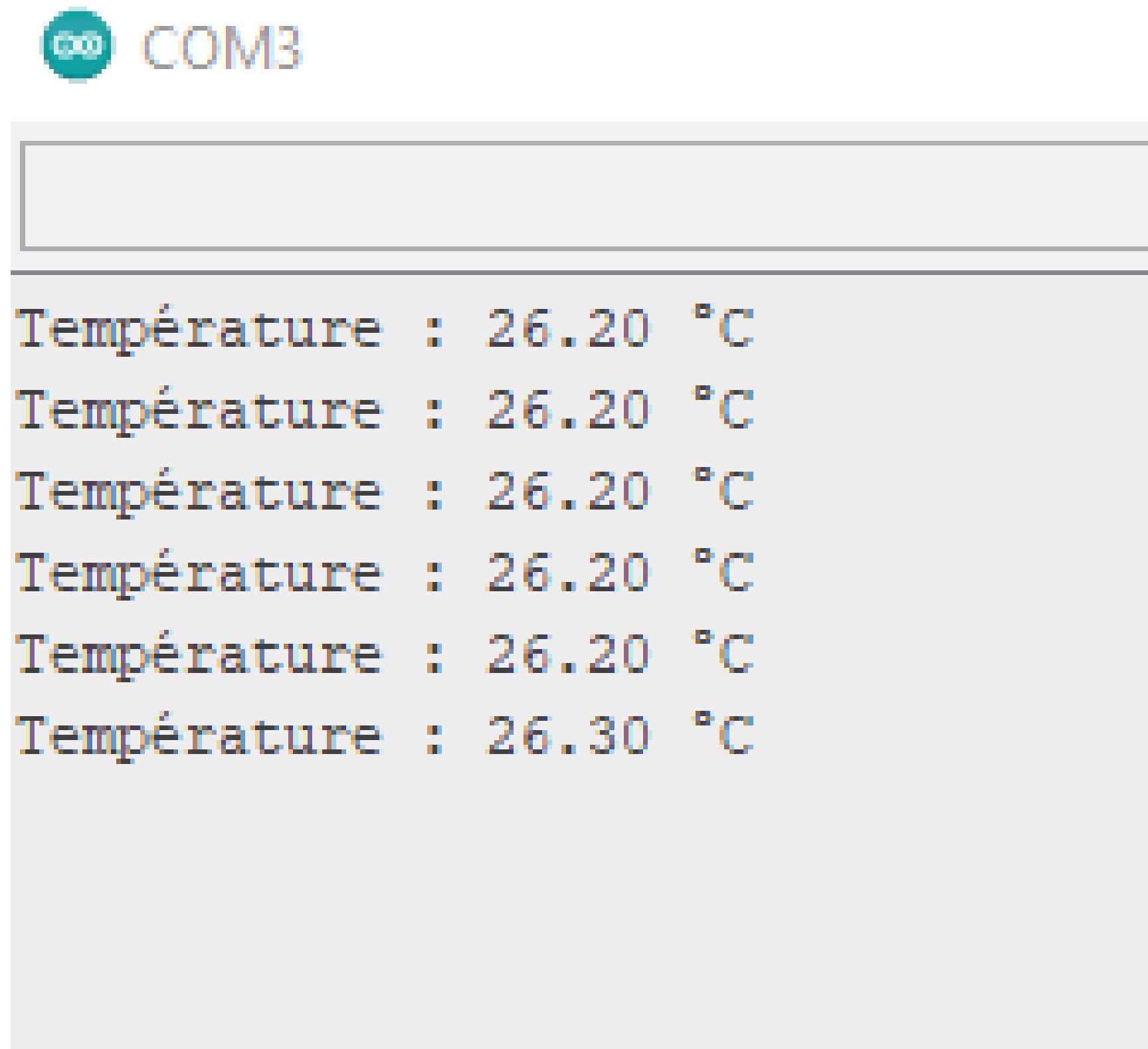
⚠ État: Surchauffe !

Max: 34.0°C

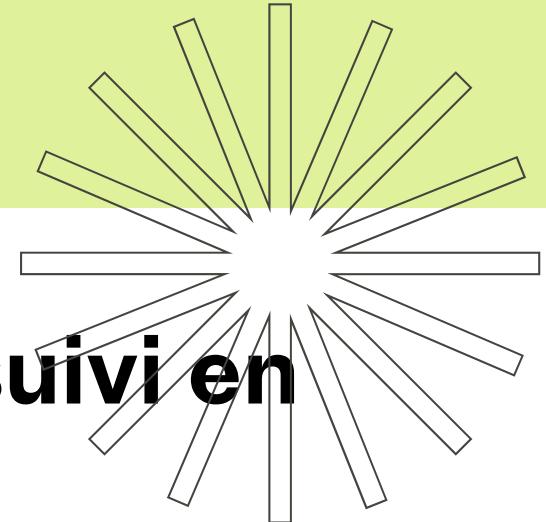


Moniteur Série

Affiche en direct les données reçues depuis l'ARDUINO UNO via la liaison série
Permet de visualiser les valeurs brutes transmises par le microcontrôleur



CONCLUSION



- **Interface intuitive, complète et interactive pour le suivi en temps réel**
- **Tableau de bord avec affichage clair des données**
- **Historique exportable au format CSV**
- **Graphique dynamique pour la visualisation des températures**
- **Moniteur série intégré pour communication avec l'ESP32**
- **Alertes automatiques en cas de dépassement de seuil**
- **Design moderne et personnalisable pour une meilleure expérience utilisateur**
- **Vers une supervision fiable et sécurisée des systèmes automobiles**

Merci pour votre attention 