Plateforme véhiculaire embarquée multitâche basée sur les protocoles de communication I2C (Maître/Multi-Esclaves) et MQTT (Nuage/Noeud)

BADRY ZAKARIA
Filière : Genie informatique
Kénitra
email: zakaria.badry@uit.ac.ma

MEZIANE ZERHOUNI HASSAN
Filière Genie informatique
Kénitra
email:hassan.mezianezerhouni@uit.ac.ma

CHAGRI ANASS
Filière Genie informatique
Kénitra
email : anass.chagri@uit.ac.ma

hamza mouhawire
Filière : Genie informatique
Kénitra email:
hamza.mouhawire@uit.ac.ma

I. INTRODUCTION

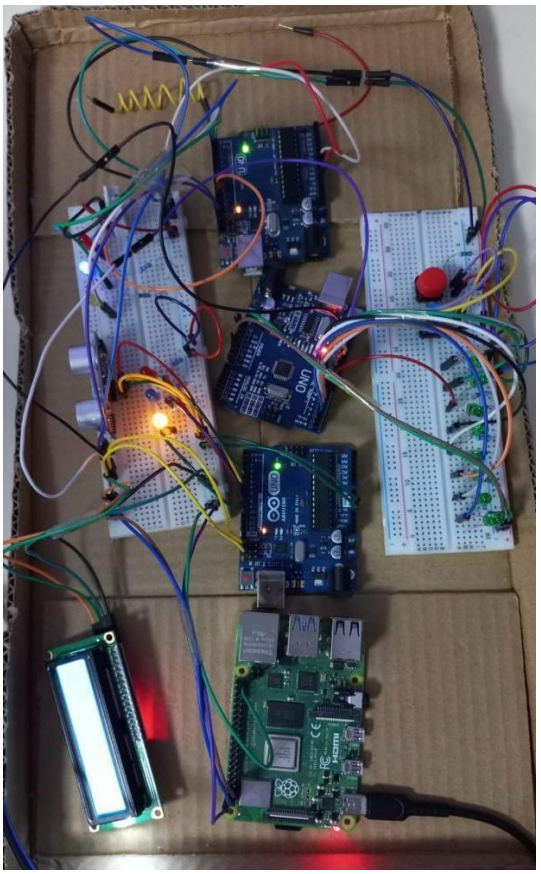
Avec l'essor des véhicules intelligents et connectés, il devient essentiel de concevoir des plateformes embarquées capables de gérer efficacement les données et les actions liées à la surveillance et au contrôle du véhicule. Le présent rapport décrit la conception et l'implémentation d'une plateforme véhiculaire embarquée multitâche basée sur les protocoles de communication I2C (Maître/Multi-Esclaves) et MQTT (Nuage/Noeud). Cette plateforme est composée d'un maître, un Raspberry Pi, et trois esclaves Arduino, chacun responsable de tâches bien spécifiques .

II. OBJECTIF

L'objectif principal de ce projet est de créer une plateforme capable de collecter et de traiter des données provenant de différents capteurs embarqués . Les capteurs sont capables de fournir des informations pertinentes , son environnement, . La plateforme est chargée de recevoir, de stocker, et de traiter ces données de manière efficace .

III. L'ARCHITECTURE DE LA PLATFORME

La structure fondamentale de cette plateforme repose sur une hiérarchie maître-esclave, où le Raspberry Pi assume le rôle de maître tandis que trois Arduino jouent le rôle d'esclaves. Les échanges d'informations entre le maître et les esclaves sont orchestrés au moyen du protocole I2C, garantissant une communication efficace et bidirectionnelle. Parallèlement, la connectivité entre le nuage (cloud) et le nœud (la plateforme Raspberry Pi et ses esclaves Arduino) est établie grâce au protocole MQTT, permettant une communication fluide et instantanée, enrichissant ainsi la plateforme de fonctionnalités de surveillance et de contrôle avancées.

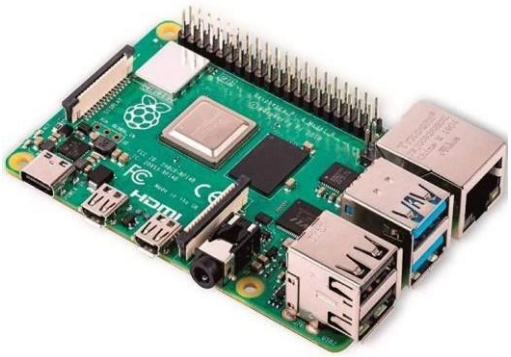


IV. LE MAITRE (RASPBERRY PI)

Le maître Raspberry Pi a plusieurs fonctions dans la plateforme. Il est responsable de :

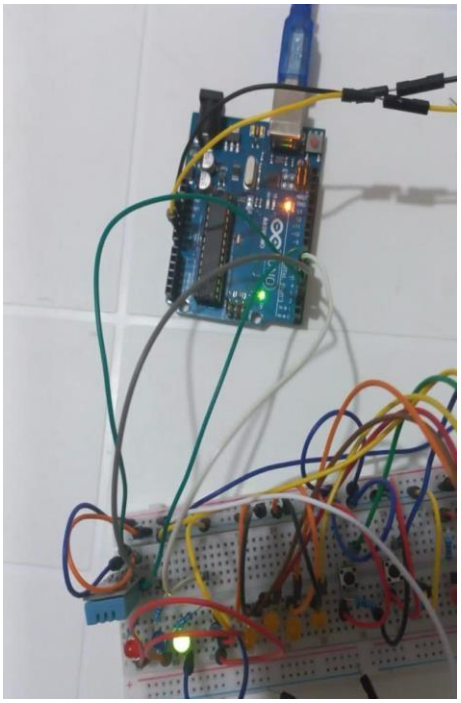
- Gérer la communication I2C avec les esclaves Arduino, en leur envoyant des requêtes ou des commandes, et en recevant leurs données.
- Gérer la communication MQTT avec le nuage, en publiant les données des esclaves sur des topics spécifiques, et en s'abonnant à des topics pour recevoir des instructions ou des alertes du nuage.

- Exécuter le script principal qui coordonne les différentes tâches de la plateforme, en fonction des données reçues et des décisions prises.
- Afficher les données et les actions sur un écran LCD connecté au Raspberry Pi, pour permettre une visualisation locale des données.

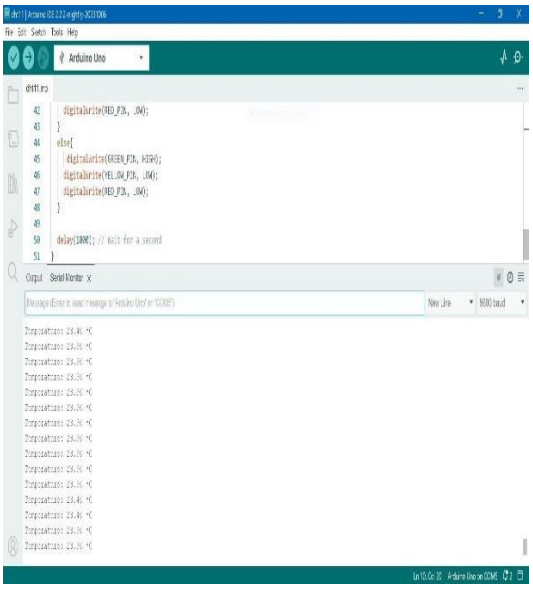


V. L'ESCLAVE 1

Le premier esclave est équipé d'un capteur DHT11 permettant de mesurer la température et l'humidité. Ces données sont transmises au maître via le protocole I2C.

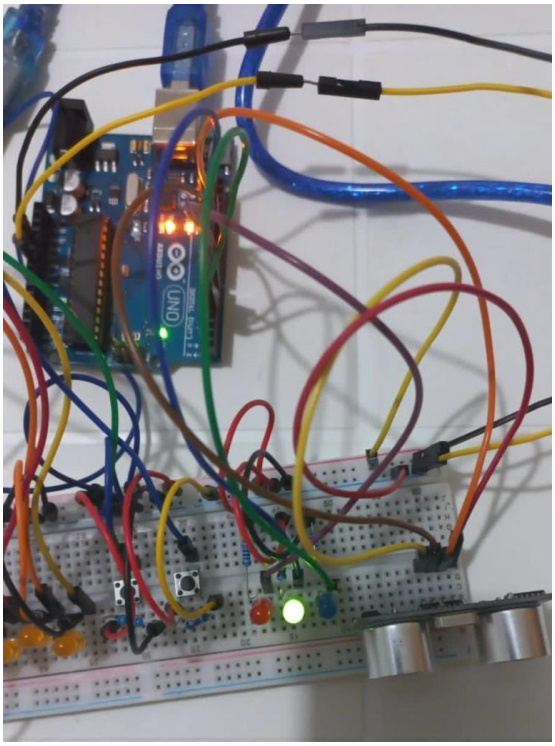


- Execution DHT11 :

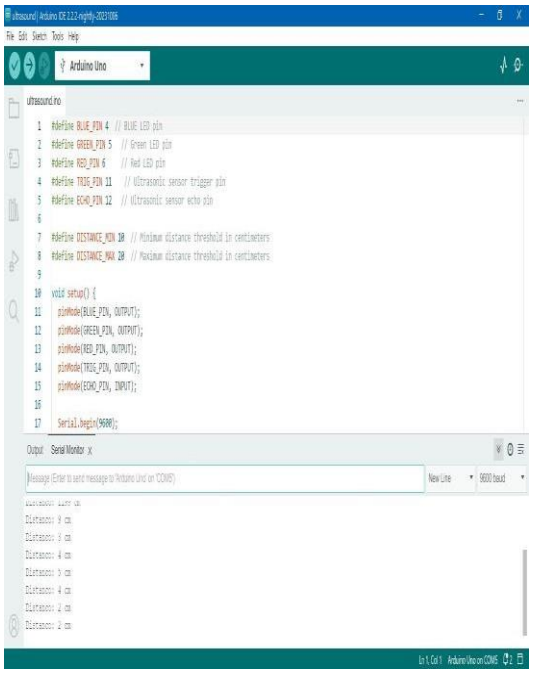


VI. L'ESCLAVE 2

Le deuxième esclave est équipé d'un capteur ultrasonique pour mesurer la distance et la vitesse du véhicule. Il contrôle également trois lampes de signalisation (bleue pour loin, verte pour moyenement proche , rouge pour tres proche) en fonction de la distance mesurée.



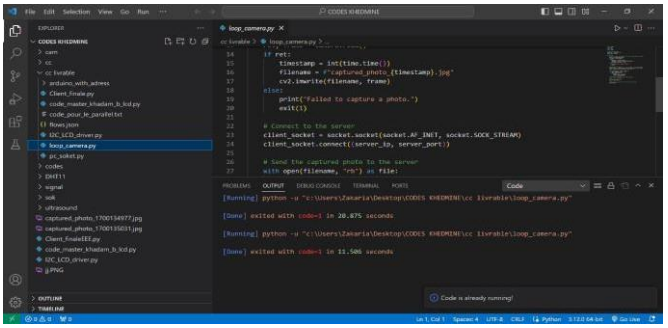
- Execution ULTRASON :

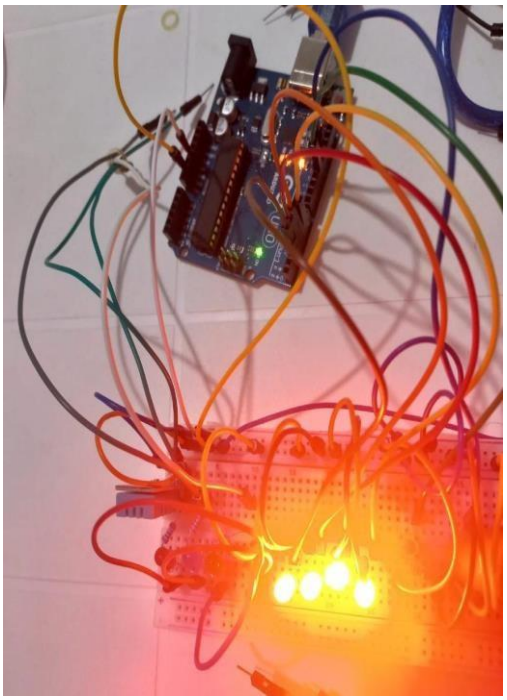


- Execution camera :

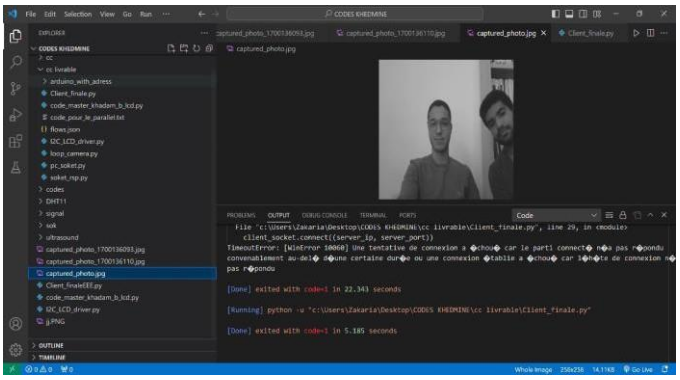
VII. L'ESCLAVE 3 ET CAMERA

Le troisième esclave assume la responsabilité de la gestion de la direction . Cette direction est visuellement signalée par des indicateurs lumineux qui défilent de manière dynamique, s'ajustant en temps réel en fonction de la direction sélectionnée par l'utilisateur à l'aide d'un bouton dédié.il y a aussi une camera qui capture en continue des photos et eux aussi sont envoyés au maitre .



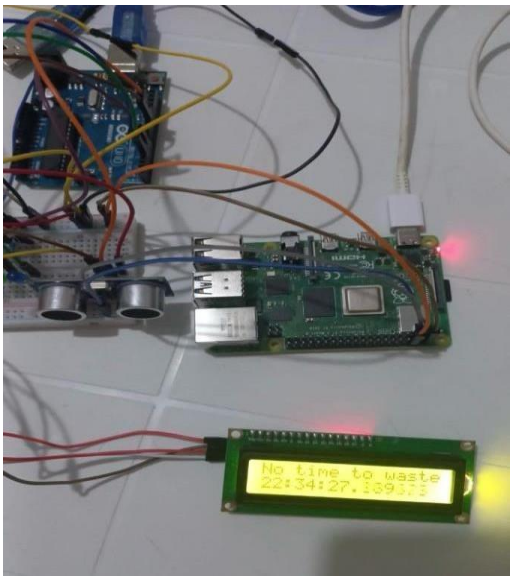


- Execution SIGNAL :

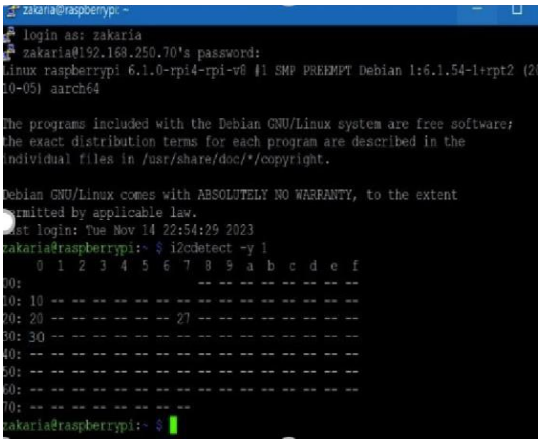


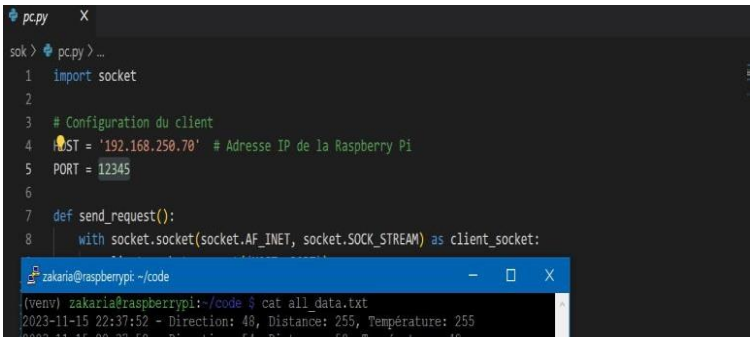
VIII. GESTION DE LA COMMUNICATION I2C

Le maître (Raspberry Pi) communique avec les esclaves (Arduino) via le protocole I2C, assurant une transmission de données bidirectionnelle. Des adresses I2C uniques sont attribuées à chaque esclave pour permettre une communication sélective.



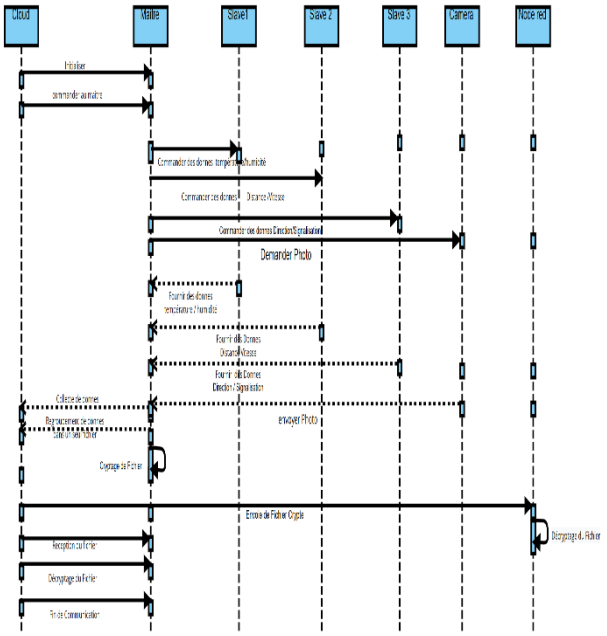
- Détection lcd i2c





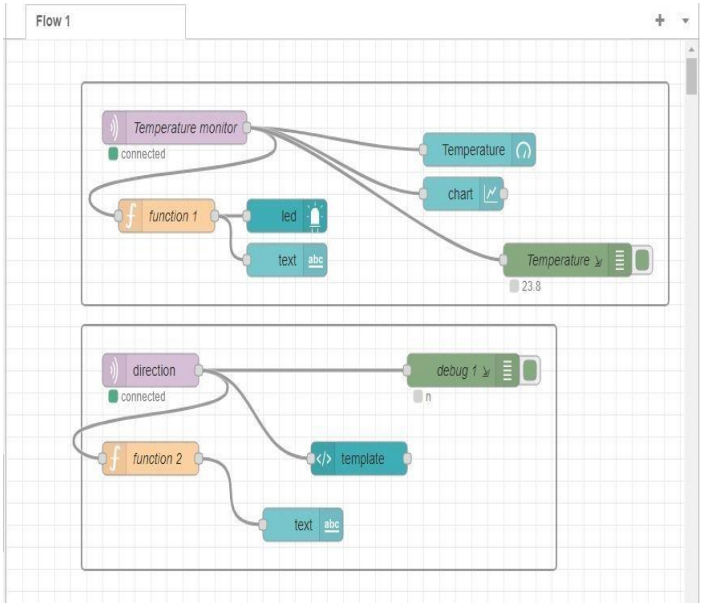
La plateforme utilise MQTT ou dans notre cas Socket pour la communication Cloud-Maitre. Le nuage envoie des ordres au maître via le broker MQTT, lui demandant de lui fournir les données des esclaves. Le maître reçoit les ordres , et les exécute en demandant les données aux esclaves via I2C, il collecte les données des esclaves en simultané, les compresse en un seul fichier, et le crypte . Le maître utilise ensuite un socket pour envoyer le fichier crypté au cloud via le broker MQTT. Le nuage reçoit le fichier crypté , le décrypte , et le décompresse pour extraire les données. Puis le cloud communique les données à NODE-RED pour la création d'un tableau de bord interactif affichant les données de manière conviviale.

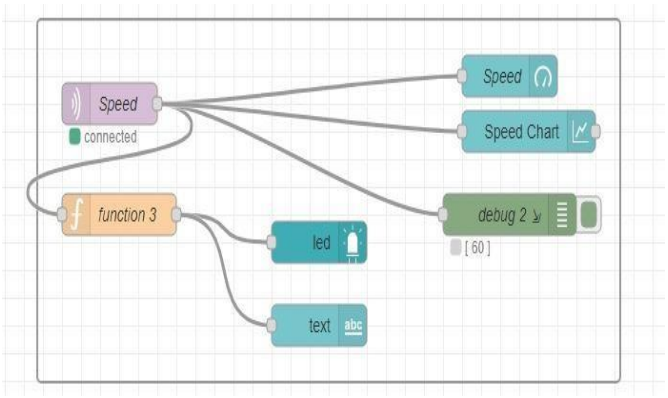
IX. Diagramme de séquence



X. MISE EN OEUVRE LOGICIELLE (INTEGRATION DU PROTOCOLE MQTT , SOCKET ET DE NODE-RED)

- Conception NODE-RED :

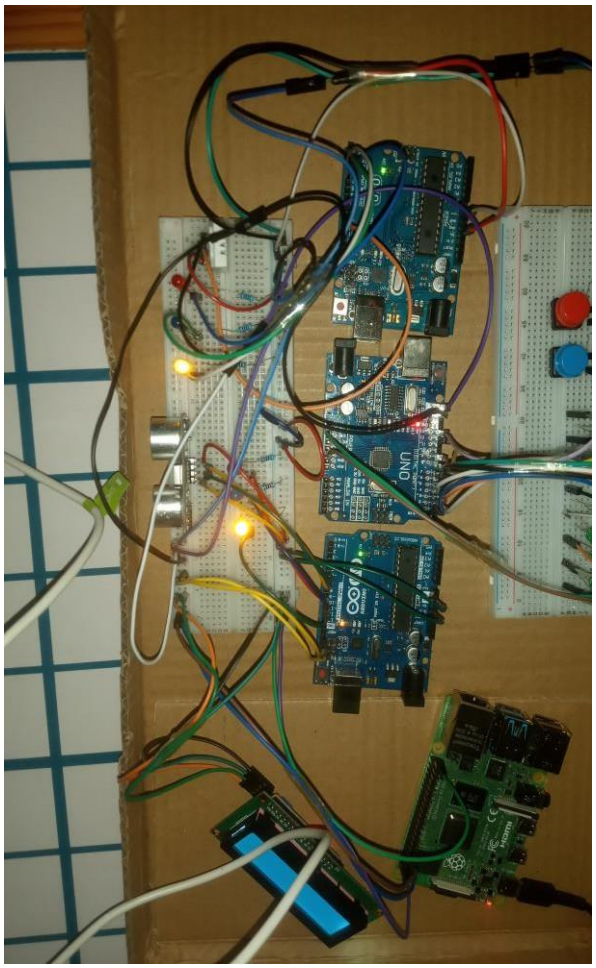




XI. RESULTATS ET TESTS

Les tests effectués ont validé la fiabilité de la plateforme dans des scénarios simulés. L'étalonnage des capteurs, la vérification de la communication I2C, et la réactivité du système ont été confirmés avec succès.

- LES ESCLAVES détectés :



XII. CONCLUSION

La plateforme véhiculaire embarquée multitâche, combinant I2C, MQTT, et Node-RED, offre une solution complète pour la surveillance et le contrôle du véhicule. L'utilisation de Node-RED facilite la visualisation des données, offrant une interface utilisateur intuitive pour une gestion optimale des informations. Cette intégration réussie ouvre la voie à des améliorations futures et à des applications étendues de la plateforme.