

1 Introduction

Le projet "Contrôleur et télémétrie pour Asseto Corsa" permettra de contrôler une voiture dans le jeu Asseto Corsa ainsi que d'afficher la télémétrie renvoyée par celui-ci.

Ce projet sera composé de deux périphériques :

- Le volant, utilisant une Mylab2 pour la direction, l'accélération et freinage de la voiture.
- Le dashboard communiquant avec l'ordinateur et le volant permettant l'affichage de données télémétriques et de boutons pour certains contrôles spécifiques (ABS, TC, etc.). Il servira alors de relai entre le volant et l'ordinateur.

2 Prérequis

2.1 Matériel

- 2 Mylab2 + LPC1769
- 1 câble CAN (+ 50 cm)
- 1 câble UART (USB A -> Micro USB)
- 1 câble HID (USB A -> Micro USB)

2.2 Logiciel

- Asseto Corsa
- Script python pour l'acquisition de données télémétriques et envoi au dashboard via UART.

3 Implémentation

3.1 Schéma de montage

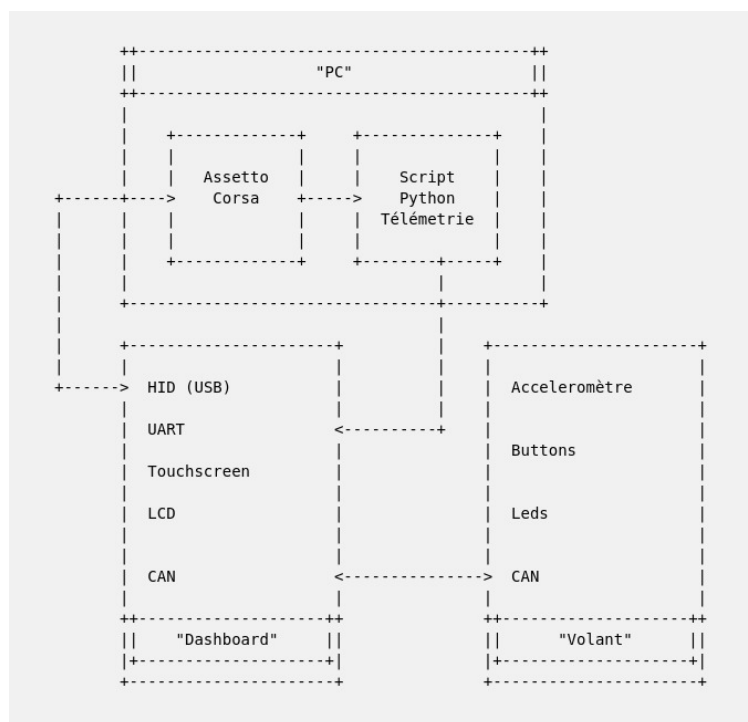


FIGURE 1 – Schéma de montage

3.2 Volant

Le volant sera une Mylab2 utilisant l'accéléromètre pour la direction, les boutons pour l'accélération / freinage. Il enverra périodiquement les données de l'accéléromètre et le status des boutons au dashboard via le protocole CAN. De plus, les leds seront utilisées comme indication si on s'approche du nombre de tours par minute maximum du moteur du véhicule.

3.3 Dashboard

Le dashboard sera utilisé pour afficher différentes données récupérées par le jeu (liste complète à déterminer) ainsi qu'un ensemble de boutons cliquables sur l'écran permettant le réglage du véhicule (ABS, Traction Control, etc.). Il communiquera avec le volant via CAN, avec le PC via HID pour l'envoi des données et UART pour la réception de la télémétrie.

3.4 PC

Le pc utilisé devra faire tourner le jeu Asseto Corsa ainsi qu'un script python intermédiaire permettant de transmettre les données télémétriques envoyées (via UDP) automatiquement par le jeu et ainsi les retransmettre via UART. Le script python sera fortement inspiré d'un exemple fonctionnel en JavaScript : <https://github.com/rickwest/ac-remote-telemetry-client?tab=readme-ov-file>.

3.5 Interruptions

Voici la liste des interruptions que nous allons utiliser :

- CAN
- UART
- Dalle tactile
- USB
- GPIO
- Timer

3.6 Bonus

Si le temps le permet, il pourrait être intéressant d'intégrer un second contrôleur (en plus du volant) permettant par exemple de gérer le frein à main.