

Notice télémétrie Matlab

La télémétrie permet la communication de données depuis la carte Arduino vers un laptop. Elle est essentielle pour vérifier que le code s'exécute correctement, que les capteurs fonctionnent, évaluer les performances de vos contrôleurs, etc. Il y a deux parties essentielles :

- les fonctions de télémétrie de la librairie MecatroUtils
- le code Matlab / Python à exécuter sur la machine qui reçoit les données. On explique ici comment se servir du code Matlab, pour la partie Python

Télémétrie dans MecatroUtils

Le fichier d'exemple TelemetryDemo contient déjà beaucoup de commentaires sur la façon dont fonctionne la télémétrie. Il y a en gros deux étapes : initialiser la télémétrie dans la fonction `setup()` et logger les variables dans la `controlLoop()` principale.

Initialisation

- on renseigne le nombre de variables à enregistrer et le nom (chaîne de caractère) qu'on veut qu'elles aient lors de leur enregistrement en Matlab. Pas besoin que ce nom corresponde à la variable Arduino qu'on va logger.
- On appelle la fonction `initTelemetry`, qui prend également comme argument le nom du réseau WiFi (donnez-lui un nom spécifique, pour que les autres groupes ne s'y connectent pas par erreur...), le mot de passe, et la `controlLoopPeriod`. **Important : cette fonction bloque l'exécution du code Arduino.** Celle-ci ne reprend que lorsque le caractère 's' (pour 'start') a été envoyé à l'Arduino. Les scripts Matlab et Python envoient automatiquement ce caractère quand vous les exécutez. Vous pouvez aussi tester l'envoi du caractère via le Serial Monitor (mais les infos envoyées ensuite sont illisibles). La carte est prête à recevoir le caractère 's' lorsque les LEDs rouges **s'éteignent**.
- Eventuellement, on peut recevoir des paramètres avec la fonction 'receiveGains'. Le fichier d'exemple `sendGains.ino` montre comment faire.

Logging

Dans la fonction `controlLoop()`, on appelle juste la fonction `mecatrou::log`, avec la position de la variable qu'on veut logger et la valeur.

Côté Matlab

Le fichier à utiliser est `main_telemetry`. Les autres fichiers (`get_data_Wifi` et `get_data_USB`) doivent être dans le même dossier. Vous pouvez choisir de logger en WiFi ou en USB en changeant simplement la variable 'method'. Aucun changement n'est nécessaire côté Arduino.

Le paramètre 'K' peut être laissé vide (`K = []`) ou même être supprimé des arguments de la fonction `get_data`, il sert à envoyer des paramètres à la carte Arduino (cf le code `sendGains`). Assurez-vous d'avoir le même nom de réseau, mot de passe, baudrate que le code Arduino.

Pour la communication en USB, le nom du port série sur lequel est branchée la carte doit être renseigné. Vous pouvez accéder à une liste des ports série en tapant juste 'serialportlist' dans la ligne de commande Matlab. L'IDE Arduino vous indiquera lequel de ces ports série est utilisé pour la carte, si besoin.

Pour la communication en WiFi, pensez à vous connecter au réseau WiFi créé par la carte Arduino avant de lancer le script. **Important : pour la télémétrie en WiFi, le baudrate doit être de 230400.**