

Portage architecture Arduino vers ESP32

— Épita —

Équipe Robotique d'Exploration



Tony BERNIS

Valentin DUMOUSSET
Merlin VOTAT

Julien LE QUANG

12 décembre 2022

Résumé

Micro ROS (Robot Operating System) est un framework pour faciliter le développement robotique et qui sert d'interface entre les micro contrôleurs. Il permet ainsi l'abstraction du matériel, la transmission de message etc. Micro ROS n'est pas disponible sur toutes les cartes Arduino. Dans notre projet, nous avons choisi une carte Nano RP2040. L'objectif de notre projet est dans un premier temps de faire bouger l'araignée et dans un second temps travailler sur la synchronisation entre une autre araignée.

Exemple gras et italique :

Sed ut **perspiciatis** unde omnis iste **natus** error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore *veritatis* et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non *numquam eius modi tempora incidunt ut* labore et dolore magnam aliquam quaerat voluptatem. **Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis** suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur?

Exemple texte centré :

Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur? Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

Ut enim ad minim veniam, **quis nostrud exercitation ullamco** laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non **proident, sunt in culpa** qui officia deserunt mollit anim id est laborum.

Table des matières

Résumé	1
1 Introduction	3
2 État de l'Art	6
2.1 Les mouvements	6
2.2 Augmentation vitesse mouvements	8
2.3 Synchronisation arraignées	9
3 Implémentation	10
3.1 Implémentation technique	10
3.2 Limites et problèmes rencontrés	10
3.2.1 Limites	10
3.2.2 Problèmes rencontrés	10
4 Conclusion	12
Glossaire	12
Bibliographie	12

Chapitre 1

Introduction

Introduction du document - Nous avons qu'une personne a installé Micro ROS sur une Arduino Portenta et une carte Arduino RP2040 (<https://www.youtube.com/watch?v=mq1uFGsYqeU>). Nous avons aussi pensé à utiliser une carte ESP qui transmette les commandes à la carte Arduino fournie mais l'alimentation ne serait probablement pas suffisante.

// mettre le schéma de la carte <https://docs.arduino.cc/static/56034f29d9e2bd28f4fd3c90268d0557/datasheet.pdf>

Nous avons téléchargé ROS pour prendre en main le framework. Malheureusement, il y avait des soucis de compatibilité sur différents OS :

- sur Mac, il y avait un problème de dépendance python (graphviz) bien que la dépendance était installée
- sur Ubuntu 18.04, le framework ne fonctionnait pas (problème pour afficher une fenêtre graphique)

Il existe plusieurs versions de ROS, chacune étant destinée pour une version Linux spécifique. La dernière est la version humble et qui est compatible avec la version 22.04 d'Ubuntu. On retrouvera la documentation à cette adresse : <https://docs.ros.org/en/humble/Installation/Ubuntu-Install-Debian.html>

On peut également utiliser une image Docker pour ROS 2.

L'étape suivante était de générer l'agent Micro ROS et le lancer. On peut retrouver à cet effet un tutoriel à cette adresse : <https://gist.github.com/Redstone-RM/0ca459c32ec5ead8700284ff56a136f7>

Nous avons testé la connexion wifi entre l'agent sur l'ordinateur et la carte en utilisant le code exemple de la librairie micro ROS : https://github.com/micro-ROS/micro_ros_arduino/blob/galactic/examples/micro-ros_publisher_wifi/micro-ros_publisher_wifi.ino L'exemple initie la connexion wifi et envoie un message à la carte. Lorsque la connexion wifi échoue, la diode sur la carte se met à clignoter de manière intempestive.

Nous avons testé ensuite le code de l'araignée SEALK pour tester les mouvements mais le code fourni ne marchait pas sur la carte. En effet, nous avons le problème de compilation suivant :

```
src/armcontroller.h:52:13: error: there are no arguments to 'sei'
that depend on a template parameter, so a declaration of 'sei'
```

case 00	case 01
case 10	case 11

TABLE 1.1 – Légende du tableau

must be available `[-fpermissive]`
`sei()`;

Cette interruption permet d'activer les interruptions de timer.

Nous avons utilisé le code que nous avons fait en cours et celui-ci marchait. Pour reset la mémoire de la carte, il faut soit tambouriner le bouton, soit appuyer longtemps. (à confirmer)

Exemple tableau & table :

case 00	case 01	case 03
case 10	case 11	case 13
case 20	case 21	case 23
case 20	case 21	case 33

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem.

Test des listes :

- Item 1
- Item 2
- Item 3
- ...

Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

- ⇒ Item 1
- Item 2
- Item 3
- × Item 4
- ✓ ...

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio.

1. Item 1
2. Item 2
3. ...

Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor

repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae.

- a) Item 1
- b) Item 2
- c) ...

Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat.

Chapitre 2

État de l'Art

Introduction de l'état de l'art - Nous utilisons une manette Xbox pour donner les ordres à notre araignée. Voici ce que nous avons réussi à faire :

Utilisation d'une référence [?].

Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur ? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur ?

2.1 Les mouvements

// Mettre une photo pour chaque mvt

sit : faire asseoir l'araignée

stand : met debout l'araignée

walk forward, walk back, walk left, walk right : bouge l'araignée dans les 4 directions

turn left, turn right : tourne l'araignée à gauche et à droite

rave (vague)

flex

hello

applaud sim

applaud wave

applaud

spoutnik

cross

pls : position par défaut

bolting

wink

Utilisation d'une autre référence [?]. Et encore une autre [?]. Ou bien plusieurs [?, ?, ?]. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Figure :



FIGURE 2.1 – Légende de l'image



Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem.

Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat.

2.2 Augmentation vitesse mouvements

Pour augmenter ou diminuer la vitesse des mouvements, il faut appuyer sur les joysticks de la manette.

Celui de gauche permet d'augmenter le délai entre les instructions et celui de droite permet de le diminuer.

Référence interne, voir la section 2.1 page 6. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

```
#include <stdio.h>

int main(void) {
    printf("Hello World!\n");
    return 0;
}
```

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

Code `#include <stdio.h>` en ligne.

Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

Code en provenance d'un fichier :

```
#include <stdio.h>

int main(void) {
    printf("Hello World!\n");
    return 0;
}
```

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus.

```

Une instruction
tant que La condition est vraie faire
|   si Une autre condition est vraie alors
|   |   Une instruction
|   sinon
|   |   Une autre instruction
|   fin
fin
si Une condition est vraie alors
|   Une instruction
|   Une autre instruction
fin
répéter
|   Une instruction
tant que La condition est vraie;

```

Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat.

Considérez par exemple l'équation [2.1](#) page [9](#) :

2.3 Synchronisation arraignées

Les arraignées sont synchros. Elles sont connectés par wifi à l'ordinateur qui envoient les messages en UDP.

$$\left(\frac{x^2}{y^3} \right) \quad (2.1)$$

Implémentation

Introduction de l'étude de faisabilité - Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem.

3.1 Implémentation technique

// A confirmer / compléter par notre expert Merlin

Les ordres sont envoyées dans une queue et il y a un publisher et un récepteur.

3.2 Limites et problèmes rencontrés

3.2.1 Limites

Alimentation : cartes besoin d'une batterie externe supplémentaire

Nous avons pensé à faire un support mais nous n'avons pas eu le temps (le support n'étant pas un élément essentiel).

3.2.2 Problèmes rencontrés

Servomoteur : on a besoin de la lib RP2040_ISR_Servo (https://github.com/khoih-prog/RP2040_ISR_Servo) car les mouvements envoient une instruction en dehors de la loop principale

Impossible d'utiliser l'Arduino IDE (car il manque un time.h).

Nous avons utilisé VSCode avec PlatformIO qui est une extension de VSCode. Il est compatible Windows, Mac et Linux.

PlatformIO est designé pour faire marcher énormément de cartes et il faut donc spécifier la carte sur laquelle on travaille. Il se charge ensuite de récupérer les libs.

Chapitre 4

Conclusion

Conclusion du document - Ce projet nous a beaucoup apportés.

Beaucoup de connaissances : la modélisation 3d, des notions en électronique et robotique

Nous avons atteint nos objectifs : mvt des araignées + synchro

Voici une phrase qui utilise un [mot en entier](#) du glossaire et encore un autre [another-word](#).