

REX Minotaure

2019



MINOTAURE

Table des matières

I	Introduction	3
0.1	Histoire	4
0.2	Rôle de Minotaure	4
0.3	Les différentes équipes au cours du temps	4
0.3.1	2017-2018	4
0.3.2	2018-2019	4
II	Composants	5
1	Actionneurs	6
1.1	Moteurs pas-à-pas	6
1.1.1	Utilisation	7
1.2	Moteurs à courant continu	8
1.2.1	Utilisation	9
1.3	Moteurs brushless	9
1.3.1	Utilisation	9
1.4	Servomoteurs	9
1.4.1	Cablages	9
1.4.2	Utilisation	9
1.5	Les Dynamixels	9
2	Système pneumatique	10
2.1	Pompe	10
2.1.1	Pneumatique	10
2.1.2	Hydrolique	10
2.2	Ventouse	10
2.3	Accessoires et autres	10
3	Communication	11
3.1	Radio	11
3.2	Bluetooth	11
3.3	Module Xbee	11
4	Cartes électroniques	12
4.1	Arduino	12
4.2	Raspberry Pi	12
4.2.1	Ports USB	12
4.3	Teensy	12
5	Capteurs	13
5.1	GNSS	13
5.2	LIDAR	13
5.2.1	Matériel	13
5.2.2	Utilisation	13

5.3	Ultrason	14
5.4	Infrarouge	14
5.5	Capteurs de fin de course	14
5.6	Codeurs incrémentaux	14
5.7	Caméra	14
5.7.1	Matériel	14
5.7.2	Utilisation	14
6	Alimentation	15
6.1	Batterie	15
6.1.1	Lipo	15
6.1.2	Nimh	15
6.1.3	Chargeur	15
6.2	Adapter l'alimentation	15
6.2.1	BEC	15
6.2.2	Régulateurs de tension linéaire	15
6.2.3	Convertisseur Buck	15
III	Programmation	16
7	Raspbian	17
7.1	Le Set up	17
7.2	Le système d'exploitation	17
7.2.1	Les commandes utiles	17
7.2.2	Utiliser une clef USB	18
8	Le code	19
8.1	Python	19
8.2	C/C++	19
8.2.1	Cross-compilation	19
9	Traitement de signaux GNSS : RTK Lib	20
10	Traitement d'images : OpenCV	21
11	ROS	22
IV	Conception	23
12	Général	24
13	CAO	25
V	La Coupe de France de Robotique	26
14	Pendant l'année	27
15	Le départ	28
16	La compétition	29
16.1	Les matchs	29

Première partie

Introduction

Ce REX a pour rôle d'accumuler les connaissances de générations en générations. Souvent, la coupe de france de robotique demandera l'utilisation d'outils spéciaux comme des ventouses mais celles-ci ne seront pas utilisées les années suivantes et les connaissances dessus seront ainsi perdues.

Dès que vous rencontrez quelque chose qui n'est pas dans ce REX, que vous voulez faire part d'une expérience enrichissante, d'une astuce ou d'un problème récurrent, il ne faut hésiter à l'ajouter à ce REX (Objectif : il faut que ce REX soit plus épais que le poly de MMC).

Minotaure ne pourra s'améliorer que si on accumule l'expérience au fur et à mesure des années (chose difficile avec les césures).

0.1 Histoire

Minotaure existe depuis 2004. C'est un P03 qui a créé l'association pour pouvoir continuer à participer à la coupe de france de robotique hors de la mécatronique (il a d'ailleurs créé sa propre entreprise qui sponsorise des équipes de la coupe de france de robotique).

0.2 Rôle de Minotaure

0.3 Les différentes équipes au cours du temps

0.3.1 2017-2018

0.3.2 2018-2019

Deuxième partie

Composants

Chapitre 1

Actionneurs

1.1 Moteurs pas-à-pas

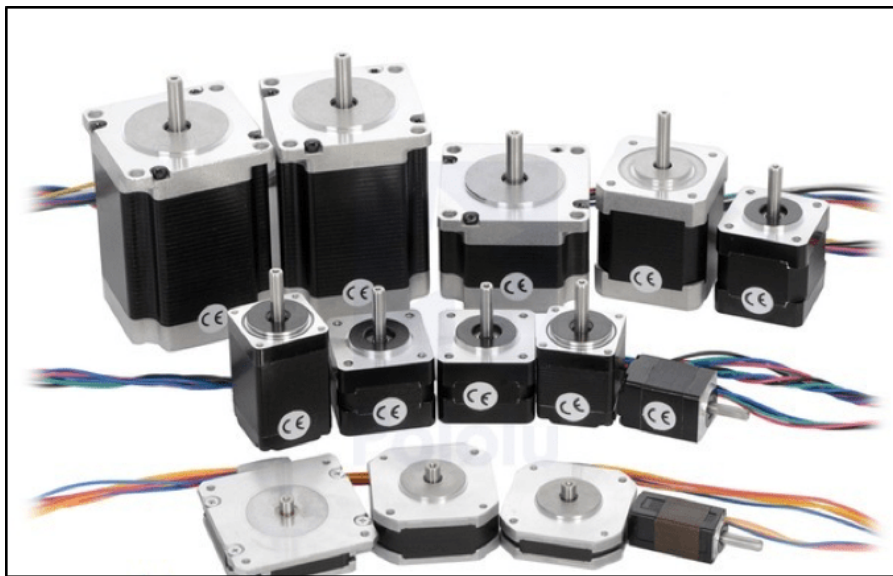


FIGURE 1.1: Moteur pas-à-pas

Les moteurs pas-à-pas sont des moteurs électriques brushless à courant continu. Il en existe 3 types :

Moteurs pas-à-pas à aimants permanents Un aimant permanent est solidaire de l'axe du moteur (rotor). Des bobines excitatrices sont placées sur la paroi du moteur (stator) et sont alimentées chronologiquement. Le rotor s'oriente suivant le champ magnétique créé par les bobines.

Moteurs pas-à-pas à reluctance variable Il s'agit d'un moteur qui comporte un rotor à encoches se positionnant dans la direction de la plus faible reluctance. Ce rotor, en fer doux, comporte moins de dents qu'il n'y a de pôles au stator.

Moteurs pas-à-pas hybrides synchrones Ils combinent les 2 technologies précédentes, et sont pluschers. Leur intérêt réside dans un meilleur couple, une vitesse plus élevée.

<https://eskimon.fr/tuto-arduino-603-a-petits-pas-le-moteur-pas-%C3%A0-pas#se-servir-du-mot>

Le nombre de phase d'un moteur correspond au nombre de bobines indépendantes. Ce nombre est proportionnel à la précision du moteur. Parmi les moteurs pas-à-pas à 2 phases, il existe 2 sous-groupes :

Unipolaire : Les plus simples d'utilisation et moins chers. Leur connecteur est constitué de 6 fils. Chaque pôle est constituée de deux bobines enroulées en sens inverse sur les pôles du stator.

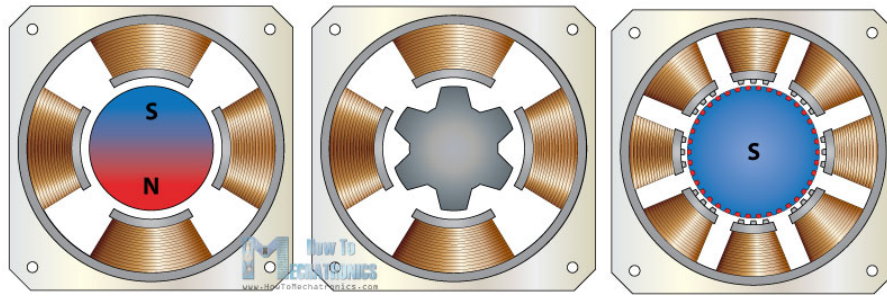


FIGURE 1.2: De gauche à droite : Aiment permanent, Reluctance Variable, Hybride

Pour changer le sens du champ magnétique dans un pôle, il faut alimenter l'une ou l'autre de ces deux bobines. Le sens du courant est constant.

Bipolaire : Les plus puissants. Leur connecteur est constitué de 4 fils. Chaque pôle du stator est constitué d'une seule bobine, et nécessite donc deux fils d'alimentation. Le sens du courant change tout le temps.

Les moteurs les plus courants sont ceux à aimants permanents et les hybrides.

Vidéos explicatives des moteurs pas-à-pas hybrides : <https://www.youtube.com/watch?v=eyqwLiowZiU>

Utilisation : Imprimantes (3D ou simple), photocopieurs, robotique, pousse-seringue,...

Avantages :

Contrôle en position possible donc en vitesse sans avoir besoin d'une boucle fermée

Précision

Existence d'un couple d'arrêt

Inconvénients :

- Le couple maximal est inversement proportionnel à la vitesse maximale, c'est-à-dire qu'un moteur pas-à-pas sera capable de fournir le plus de couple pour de faible vitesse de rotation.
- Lent, ils ne dépassent pas les 3000 tr/min en général.
- Volumineux et lourd
- Fonctionnement plus complexe qu'un moteur à courant continu
- Résonance du moteur

Précautions :

- Le bobinage des moteurs pas-à-pas est très fragile. Si les câbles sont mal branchés, il peut y avoir un court-circuit entre les 2 bobinages ce qui "cassera" le moteur pas-à-pas. Il faut utiliser un ohmmètre pour distinguer les fils. 2 fils reliés à la même bobine auront une résistance tandis que d'autres connections afficheront soit rien soit une très grande résistance.
- Toujours manipuler les fils quand il n'y a pas de courant. Un court-circuit peut facilement détruire les moteurs.

Pour distinguer les différents moteurs pas-à-pas, la norme NEMA est utilisée. Cette norme indique la taille du moteur et donc sa puissance. Le nombre représente la surface de la face de l'axe du moteur en pouce carré. Ainsi, un moteur NEMA 11 aura une face d'une surface de 1,1 pouce carrés.

Le couple du moteur est proportionnel au courant.

Une bonne documentation fournira la vitesse et la puissance d'un moteur pas-à-pas. Cependant, il se peut que cela ne soit pas le cas. Dans ce genre de situation, il est possible de calculer des indicateurs sur la vitesse :

Une version en ligne permet de faire le calcul directement. <https://www.daycounter.com/Calculators/Stepper-Motor-Calculator.phtml>

Fournisseurs : Stepper Online

Pour plus d'information techniques, regarder : <https://www.mdp.fr/documentation/lexique/pas-a-pas/notions-techniques.html>

1.1.1 Utilisation

Un tr

FIGURE 1.3: Calcul de la vitesse d'un moteur pas-à-pas

$$\text{Speed} = \left\lceil \frac{\text{Voltage}}{((\text{Inductance}/1000) \times 2 \times \text{Imax} \times \text{Steps})} \right\rceil$$

$$\text{Time/Step} = \text{Inductance} \times \text{Imax} \times \left\lceil \frac{2}{\text{Voltage}} \right\rceil$$

$$\text{Power} = \text{Voltage} \times \text{Imax}$$

Unipolaire

<https://fr.rs-online.com/web/p/kits-de-developpement-pour-commande-de-moteur/1646982>

Code source pour l'utiliser sur le git

<http://www.f-legrand.fr/scidoc/docimg/sciphys/arduino/paspasunipol/paspasunipol.html>

Bipolaire

<http://www.f-legrand.fr/scidoc/docimg/sciphys/arduino/paspas/paspas.html>

1.2 Moteurs à courant continu

Les moteurs à courant continu sont les plus courants. Ils ne sont cependant utilisés que dans des cas basiques.

Avantages :

- Très simple d'utilisation
- Taille faible pour un couple puissant

Inconvénients :

- La loi entre la tension aux bornes du moteur et la vitesse de rotation ne sera pas la même pour des modèles identiques de moteur.
- Aucun retour d'information et contrôle en position ou en vitesse impossible en boucle ouverte.

1.2.1 Utilisation

Pont H : Utiliser le L298N

1.3 Moteurs brushless

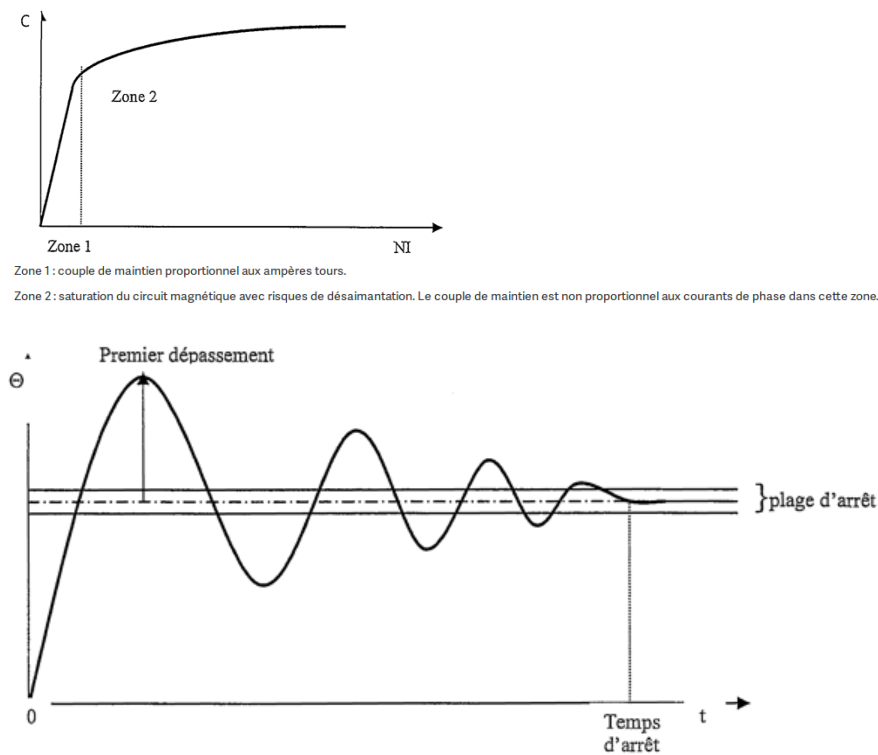
1.3.1 Utilisation

Pour utiliser un moteur brushless, il est nécessaire d'utiliser un ESC (Electronic Speed Controller)

1.4 Servomoteurs

C'est un moteur à courant continu asservi. Il est souvent utilisé pour effectuer de petits mouvements. On le commande en angle et il maintient ensuite sa position peu importe l'effort exercé dessus (bien sûr il casse si on force trop).

FIGURE 1.4: Gauche : Graphique du couple de maintien en fonction du courant consommé - Droite : Réponse en angle à un échelon



1.4.1 Cablages

Image des différents cables

Il faut 3 cables pour utiliser un servomoteur :

- Masse
- V+ : cela dépend des servomoteurs mais ce sera en général 5V
- Signal : un signal PWM est utilisé pour commander

Image PWM

Avantages :

- Simple d'utilisation
- Contrôle en position

Inconvénients :

- Aucun retour sur la position
- Limité à 160-180 ° de rotation (Il existe cependant des servos à rotation continu)

1.4.2 Utilisation

1.5 Les Dynamixels

Ce sont les actionneurs le plus souvent utilisé en robotique. Ils sont produits exclusivement par la société Robotis.

Avantages :

- Retour de beaucoup d'informations : courant consommé, couple appliqué, vitesse et position,...
- Configurable

Inconvénients :

- Très difficile d'utilisation. Il faut utiliser un connecteur spécial et coûteux pour les déboguer.
- Plus fragiles que les servomoteurs.



FIGURE 1.5: Moteurs à courant continu



FIGURE 1.6: Moteur brushless

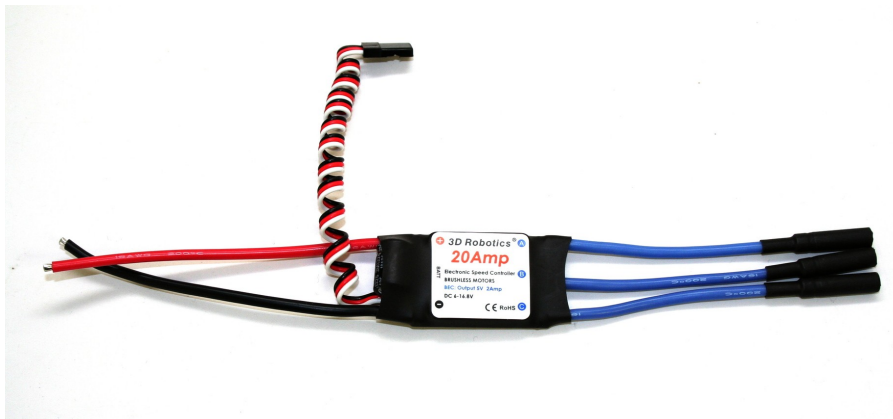


FIGURE 1.7: Moteurs à courant continu

Chapitre 2

Système pneumatique

2.1 Pompe

2.1.1 Pneumatique

2.1.2 Hydrolique

On peut ne pas pouvoir inverser le sens

2.2 Ventouse

Les ventouses sont très utiles pour se saisir d'objets ayant des surfaces relativement plane et lisse.

Ventouse avec et sans soufflet

Avantage :

Inconvénients :

Fournisseur : RS

2.3 Accessoires et autres

Chapitre 3

Communication

3.1 Radio

3.2 Bluetooth

3.3 Module Xbee

Chapitre 4

Cartes électroniques

4.1 Arduino

4.2 Raspberry Pi

C'est un micro-ordinateur. Elle est donc beaucoup plus puissante que la Teensy ou les cartes Arduino. Cependant, elle est aussi limitée dans ses capacités.

Avantages :

- Ports USB
- Wifi et bluetooth intégrés pour la version 3B+
- Plus de puissance de calcul
- Possibilités de programmer dans n'importe quel langage.

Inconvénients :

- Il est difficile de générer des signals PWM
- Seulement une vrai liaison série disponible et encore. Elle est normalement dédiée au bluetooth mais on peut l'utiliser pour autre chose si on utilise pas le Bluetooth. Il y a aussi une autre pseudo liaison série mais elle est très limitée.

4.2.1 Ports USB

Interface série

Courant

4.3 Teensy

La teensy est en résumé une version miniature d'une carte arduino.

Chapitre 5

Capteurs

5.1 GNSS

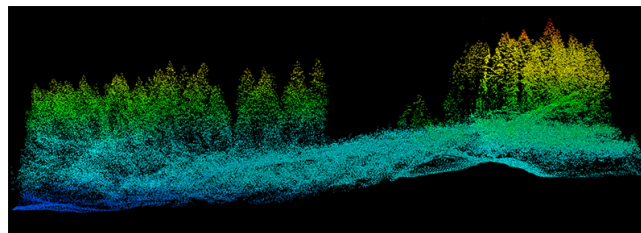
GNSS (Global Navigation Satellite System) designe un ensemble de composants reposant sur une constellation de satellites artificiels permettant de fournir à un utilisateur par l'intermédiaire d'un récepteur portable de petite taille sa position 3D, sa vitesse et l'heure.

Il est important de distinguer GPS et GNSS. GPS (Global Positioning System) correspond à la constellation de satellites des Etats-Unis qui appliquent le principe de GNSS. C'est un abus de langage.

5.2 LIDAR

Acronyme pour Light Detection And Ranging, ce capteur permet de générer un nuage de point de l'environnement qui l'entoure. Le LIDAR fait cela grâce à un laser qui mesure la distance avec un obstacle. En connaissant la configuration du LIDAR, on connaît la direction du laser et donc la direction de l'obstacle. Les LIDARs que nous utilisons sont font juste des mesures dans un plan mais il existe des LIDARs qui fournissent des nuages de points en 3D.

FIGURE 5.1: Un module Xbee (gauche) / Xbee Explorer Regulated (droite)



5.2.1 Matériel

Modèle : A2M8

Quantité : 2

Fabricant : Slamtech

Fournisseur : Robotshop

5.2.2 Utilisation

Positionnement : En utilisant des techniques de SLAM, il est possible d'utiliser des balises posées sur le bord du terrain pour obtenir la position et l'orientation du robot. Le problème est que la résolution des LIDAR que Minotaure possède n'est pas suffisante pour détecter les balises partout sur le terrain.

FIGURE 5.2: Un module Xbee (gauche) / Xbee Explorer Regulated (droite)



Détection d'obstacles : Il est possible de détecter les robots adverses à partir du nuage de points. Pour cela, il faudrait faire une détection de blobs, qui correspondraient aux robots.

5.3 Ultrason

5.4 Infrarouge

5.5 Capteurs de fin de course

Comme son nom l'indique, le capteur de fin de course permet d'observer les contacts. Ils sont aussi appelés interrupteurs ou détecteurs de position. Ce sont en soit des interrupteurs poussoirs avec des Utilisations :

5.6 Codeurs incrémentaux

5.7 Caméra

5.7.1 Matériel

Modèle :

Fabricant :

Fournisseur :

5.7.2 Utilisation

Lecture de codes couleur :

Détection d'obstacle :

Positionnement d'objets sur le terrain :

Chapitre 6

Alimentation

6.1 Batterie

6.1.1 Lipo

6.1.2 Nimh

6.1.3 Chargeur

6.2 Adapter l'alimentation

6.2.1 BEC

Le BEC (Battery Eliminator Circuit) permet de délivrer une tension. Ce circuit appartient à l'électronique de puissance.

6.2.2 Régulateurs de tension linéaire

Les régulateurs permettent comme les BEC de fournir un
si pas de source de frais, dangereux

Inconvénients :

- Pas efficace, la tension a dissipé l'est sous la forme de chaleur
- Cesse de d'alimenter en tension si la température dépasse un certain seuil

Avantages :

- Simple à mettre en oeuvre
- Peu cher (de l'ordre de la dizaine de centimes)

Utilisation

Montage

6.2.3 Convertisseur Buck

C'est le type de circuit qui est utilisé dans

Prix de l'ordre de l'euro

Utilise une bobine donc peut être source d'interférences électromagnétiques

Troisième partie

Programmation

Chapitre 7

Raspbian

7.1 Le Set up

En règle général, il n'y aura pas d'écrans, de claviers ou de souris à disposition pour pouvoir utiliser directement la raspberry pi avec son ordinateur. Pour controurner ce problème, Matthieu Vignes(P14) a créé une configuration spéciale de l'image Raspbian. Elle fait en sorte que dès le premier démarrage, la raspberry pi émet son propre réseau wifi sur lequel il est possible de se connecter pour pouvoir utiliser la raspberry pi.

La méthodologie pour créer cette image est disponible sur son Github : https://github.com/matthieuvigne/MiAM_eurobot2019/tree/master/ConfigRPi

Windows : Il faut installer Putty pour avoir accès au terminal de la raspberry pi. Sinon, il est possible d'avoir accès à la GUI avec VNC viewer.

Linux : Il faut utiliser les lignes de commandes.

sshfs : Monter sur son système de fichier, un autre système de fichier distant, à travers une connexion SSH . En gros, il est possible d'avoir accès sur l'ordinateur à tous les fichiers de la raspberry pi. Les modifications sur l'ordinateur seront aussi réalisées sur la raspberry pi.

ssh : Créé une liaison SSH avec la raspberry pi, ce qui permet d'avoir accès à son terminal.

scp : Copie les fichiers de la raspberry pi à l'ordinateur et vice-versa par liaison SSH.

Matériel utile :

Clef USB Wifi : Permettra à l'ordinateur de se connecter à 2 réseaux Wifi en même temps : Celui de la raspberry pi et à une connection internet.

Cable ethernet : La raspberry Pi utilise déjà le wifi pour communiquer avec l'ordinateur. Il est alors nécessaire d'utiliser le cable ethernet pour pouvoir réaliser des mises-à-jours logiciel. Il faut pour cela faire un partage de connection par ethernet avec l'ordinateur.

7.2 Le système d'exploitation

Raspbian est un système d'exploitation basé sur Linux. Il est possible d'installer 2 versions sur la Raspberry Pi : la version lite (sans interface graphique), la version graphique (beaucoup plus lourde et gourmande en ressource). En règle générale, il est préférable d'installer la version Lite, cela laissera plus de puissance de calcul pour les programmes qui tourneront dessus.

7.2.1 Les commandes utiles

cd :

ls :

mkdir :

nano :

vim :

chmod :

apt install :

apt upgrade :

apt update :

Pour lancer un programme binaire, il faut `"/programme"`

7.2.2 Utiliser une clef USB

Une clef USB ne peut pas être utilisée directement avec Linux, il faut la monter. Dans les OS avec interface graphique, cette étape se fait automatiquement mais pas si on utilise la version Lite de Raspbian.

Trouver le périphérique : `lsblk`

Le périphérique SUB apparaîtra dans le dossier `/dev` en général avec le nom `sdb1`. Il faut alors créer un dossier dans `/media` qui correspondra à notre clef USB.

`sudo mkdir /media/usb`

`sudo mount /dev/sdb1 /media/usb`

`sudo umount /media/usb`

Chapitre 8

Le code

Quel langage utilisé ?

8.1 Python

Python, le langage que tout le monde maîtrise normalement après la prépa. Il est très facile de faire du prototypage avec. C'est d'ailleurs avec ce langage qu'il faudra tester de nouvelles choses. Cependant, en tant que langage interprété il est beaucoup plus lent que les langages compilés (C/C++).

8.2 C/C++

8.2.1 Cross-compilation

Il est possible de compiler les programmes en C et C++ sur la raspberry pi mais cela prends beaucoup de temps en raison du manque de puissance de calcul (de l'ordre des minutes). Cependant, une fois compiler une première fois, le compilateur ne devra recompiler que les fichiers modifiés. La compilation prendra alors moins de temps.

En raison de l'architecture différente des processeurs sur l'ordinateur et sur la Raspberry Pi (ARM), il n'est pas possible de compiler les programmes sur l'ordinateur puis de les copier sur la Raspberry Pi. Il est cependant possible de simuler l'environnement de la raspberry pi pour la compilation sur ordinateur en utilisant la cross-compilation.

Chapitre 9

Traitement de signaux GNSS : RTK Lib

RTK Lib est un ensemble de programmes créés pour pouvoir exploiter et analyser les signaux obtenus à partir de récepteurs GNSS. Il existe un ensemble avec une interface graphique utilisable sous Windows. Il existe aussi une version UI utilisable sous n'importe quel système d'exploitation après avoir compiler les programmes en C.

Une branche a été créé à partir de RTK Lib : RTK Lib Explorer. Cette branch a été optimisée pour certains récepteurs et fonctionne en général mieux(en tout cas dans les cas vus).

Chapitre 10

Traitement d'images : OpenCV

OpenCV est la bibliothèque par excellence pour tout ce qui concerne le traitement d'images.

Chapitre 11

ROS

ROS est une sorte de système d'exploitation pour la robotique.

Pourquoi l'utiliser :

- Une grande communauté
- Beaucoup de bibliothèques déjà développées
- Code beaucoup plus modulable
- Possibilité de programmer un C/C++ ou en python (autres langages supportés aussi de manière non officielle)

Quatrième partie

Conception

Chapitre 12

Général

La règle d'or doit toujours être la SIMPLICITÉ. Il faut un système robuste et très répétable. Il y a toujours des imprévus (dimension de la table ou des objets à manipuler notamment).

Chapitre 13

CAO

Cinquième partie

La Coupe de France de Robotique

Chapitre 14

Pendant l'année

- Dès que du matériel est nécessaire, le commander immédiatement afin de l'avoir pour la semaine d'après. C'est souvent le matériel qui empêche de progresser sur les robots et de faire des test.

Chapitre 15

Le départ

- Faire une liste des outils et matières premières une semaine à l'avance afin de ne rien oublier. Il faudra cette liste à Mazouz.
- Trouver un moyen pour déplacer les 2 robots et l'équipement nécessaire pour les préparer avec une seule personne. En effet, seulement 2 personnes peuvent aller jusqu'à la table de match et il peut être dur de tout porter à 2. Cela peut être un petit chariot comme le font d'autres équipes.

Chapitre 16

La compétition

16.1 Les matchs

- Faire une checklist de toutes les tâches à effectuer pour préparer le match. Avec le stress, il n'est pas rare d'oublier quelque chose et de faire rater le match.
- Éviter vraiment de faire des changements de dernière minute. En 2017, RCV a perdu en demi-finale car ils ont choisi une stratégie non-testée. Un bug a empêché les 2 robots de démarrer.