

REX Minotaure

2019



MINOTAURE

Table des matières

I	Introduction	4
0.1	Histoire	5
0.2	Rôle de Minotaure	5
0.3	Les différentes équipes au cours du temps	5
0.3.1	2017-2018	5
0.3.2	2018-2019	5
II	Composants	6
1	Actionneurs	7
1.1	Moteurs pas-à-pas	7
1.1.1	Utilisation	8
1.2	Moteurs à courant continu	9
1.2.1	Utilisation	9
1.3	Moteurs brushless	9
1.3.1	Utilisation	9
1.4	Servomoteurs	10
1.4.1	Utilisation	10
1.5	Les Dynamixels	11
2	Système pneumatique	15
2.1	Pompe	15
2.1.1	Pneumatique	15
2.1.2	Hydraulique	15
2.2	Ventouse	15
2.3	Accessoires et autres	15
3	Communication	16
3.1	Radio	16
3.2	Bluetooth	16
3.3	Module Xbee	17
3.3.1	Architecture réseau	17
4	Cartes électroniques	19
4.1	Arduino	19
4.2	Raspberry Pi	19
4.2.1	Ports USB	19
4.3	Teensy	19
4.4	Shield	20
5	Capteurs	21
5.1	GNSS	21
5.1.1	NMEA	21
5.1.2	Mode de positionnement	21

5.2	LIDAR	21
5.2.1	Matériel	22
5.2.2	Utilisation	22
5.3	Ultrason	22
5.4	Infrarouge	22
5.4.1	Utilisation	23
5.5	Capteurs de fin de course	23
5.6	Codeurs incrémentaux	24
5.7	Caméra	24
5.7.1	CDD	24
5.7.2	CMOS	24
5.7.3	DVS	24
5.7.4	Matériel	24
5.7.5	Utilisation	24
6	Alimentation	26
6.1	Batterie	26
6.1.1	Lipo	26
6.1.2	Nimh	26
6.1.3	Chargeur	26
6.2	Adapter l'alimentation	26
6.2.1	BEC	26
6.2.2	Régulateurs de tension linéaire	27
6.2.3	Convertisseur Buck	27
7	Composants électroniques	28
7.1	Fusible	28
7.1.1	PTC	28
7.1.2	Fusible réinitialisable	28
7.1.3	Thermistor	28
III	Programmation	29
8	Raspbian	30
8.1	Le Set up	30
8.2	Le système d'exploitation	30
8.2.1	Les commandes utiles	30
8.2.2	Utiliser une clef USB	31
9	Le code	32
9.1	Python	32
9.2	C/C++	32
9.2.1	Cross-compilation	32
10	Traitement de signaux GNSS : RTK Lib	33
11	Traitement d'images : OpenCV	34
11.1	Systèmes de couleur	34
11.2	Odométrie visuel	34
11.3	Vision stéréoscopique	34
12	ROS	37

13 Utilisation d'un Git	38
13.1 Méthodologie	38
14 Protocole de communication	39
14.1 Série	39
14.1.1 UART	39
14.1.2 TTL	39
14.2 I2C	39
14.3 SPI	39
14.4 CAN	39
IV Automatique	40
15 Estimation de l'attitude	42
15.1 Différentes manières de représenter l'attitude	42
15.1.1 Angles d'Euler	42
15.1.2 Vecteur rotation	42
15.1.3 Quaternion	42
15.2 Filtre complémentaire	42
15.3 Filtre de Mahony	42
15.4 Filtre de Madgwick	42
15.5 Filtre de Kalman	42
V Machines de la menuiserie	43
16 Découpeuse Laser	44
17 Tour	45
18 Imprimante 3D	46
VI Conception	47
19 Général	48
20 CAO	49
VII La Coupe de France de Robotique	50
21 Pendant l'année	51
22 Le départ	52
23 La compétition	53
23.1 Les matchs	53

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 réluctance. 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.

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. 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.

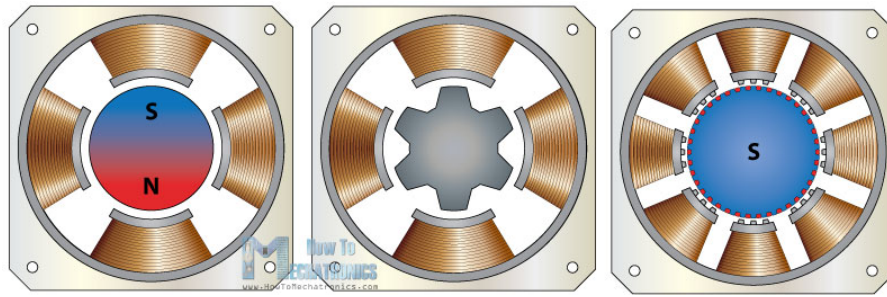


FIGURE 1.2: De gauche à droite : Aiment permanent, Reluctance Variable, Hybride

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 pouces 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, Gotronic,...

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

1.1.1 Utilisation

Ce site donne une bonne présentation des moteurs pas-à-pas et comment les utiliser : <https://eskimon.fr/tuto-arduino-603-a-petits-pas-le-moteur-pas-%C3%A0-pas#se-servir-du-moteur>
Sinon, le contrôleur de moteurs pas-à-pas utilisé pour l'édition 2019 était celui-ci :

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

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

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

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

<https://fr.rs-online.com/web/p/kits-de-developpement-pour-commande-de-moteur/1646982>
Un driver en C a été écrit par Matthieu Vignes et est disponible sur le Git.

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

Pour un moteur à courant continu, la vitesse de rotation est proportionnelle à la tension à ses bornes. Cependant, étant donné qu'il est difficile de générer une tension variable, le moteur DC est le plus souvent alimenté à travers un hacheur avec une tension fixe. https://wiki.mchobby.be/index.php?title=Pont-H_L298N

Le pont L298N est la meilleure méthode pour contrôler des moteurs à courant continu.

Il est aussi possible de choisir la tension à fournir pour contrôler le moteur avec des transistors :

<https://openclassrooms.com/fr/courses/2778161-programmez-vos-premiers-montages-avec-arduino/3285333-le-moteur-a-courant-continu-partie-1-transistors-et-sorties-pwm>

1.3 Moteurs brushless

Les moteurs brushless comme leur nom l'indique sont des moteurs à courant continu sans balais. De ce fait, ils s'usent moins rapidement que les moteurs. Ils sont notamment utilisés en aéromodélisme pour leur endurance et la puissance qu'ils peuvent délivrer. A la place des balais, 3 câbles d'alimentations sont nécessaires pour les phases des différentes bobines (alimentation triphasée).

Pour inverser le sens de rotation d'un moteur brushless, il suffit d'inverser la connexion de 2 de ses 3 câbles.

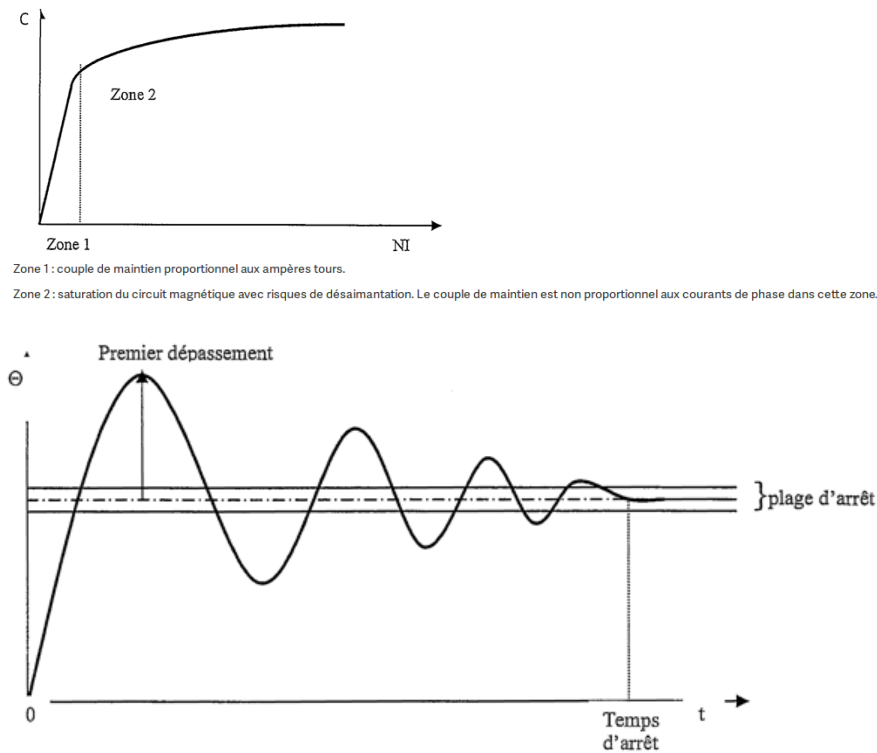
Avantages :

- Vitesse asservie
- Fabriqué pour tourner à de grandes vitesses

Inconvénients :

- Nécessité d'un ESC

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



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).

Il est composé d'un moteur à courant continu, d'un réducteur, d'un potentiomètre et d'un circuit intégré. C'est le potentiomètre qui fournit le retour en angle (d'où l'impossibilité de positionner de manière absolue sur plus de 180°)

Avantages :

- Simple d'utilisation
- Contrôle en position

Inconvénients :

- Aucun retour sur la position
- Limité à $160-180^\circ$ de rotation (Il existe cependant des servos à rotation continue)

1.4.1 Utilisation

Cablage

Image des différents câbles

Il faut 3 câbles pour utiliser un servomoteur :

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



FIGURE 1.5: Moteurs à courant continu

Signaux PWM

La durée d'un signal PWM (Pulse Width Modulation) est proportionnelle à la commande. En général, la fréquence est de 50 Hz pour les servomoteurs mais elle peut être différente pour d'autres applications.

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 1.6: Moteur brushless

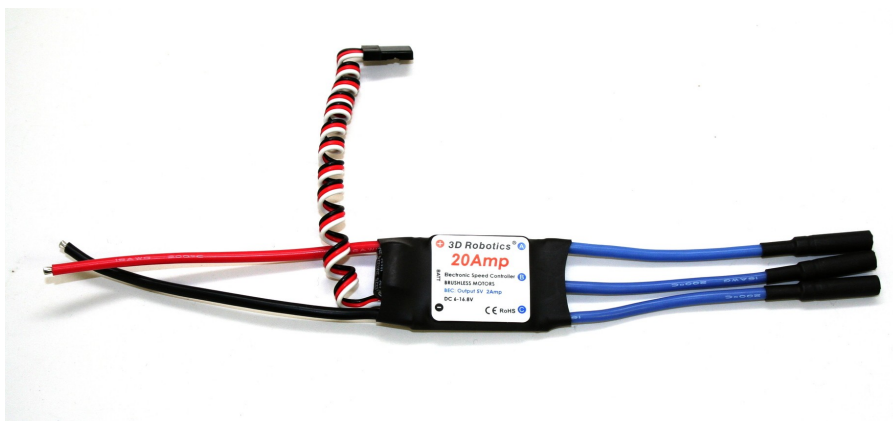


FIGURE 1.7: ESC

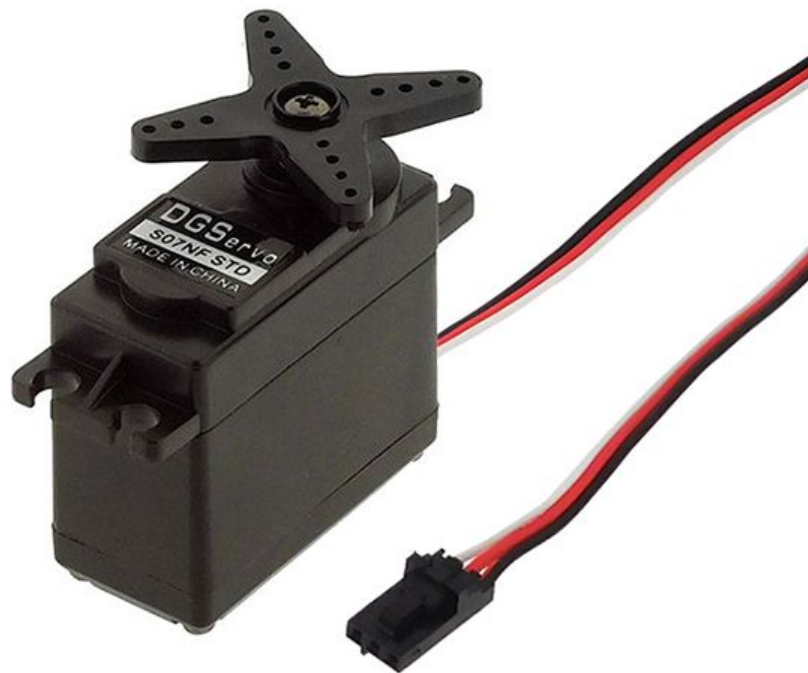


FIGURE 1.8: Servomoteurs

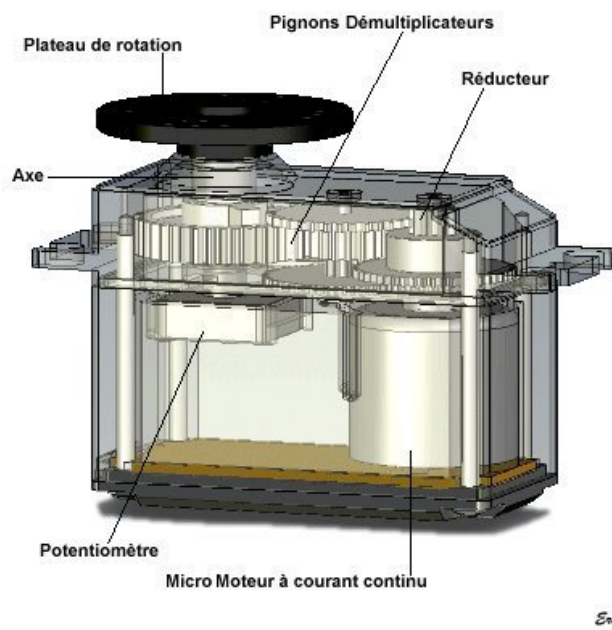


FIGURE 1.9: Schéma d'un servomoteur

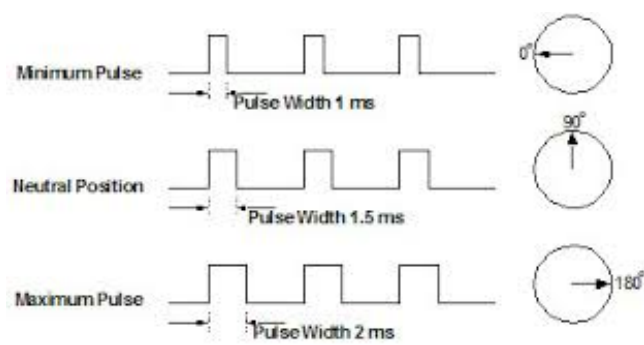


FIGURE 1.10: Signal PWM pour contrôler un servomoteur

Chapitre 2

Système pneumatique

2.1 Pompe

2.1.1 Pneumatique

2.1.2 Hydraulique

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

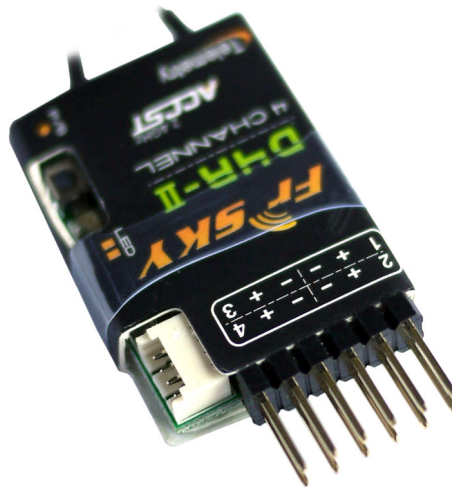


FIGURE 3.1: Récepteur radio

En modélisme, des radiocommandes sont utilisées pour commander avion, bateau ou voiture. De nos jours, la fréquence utilisée en majorité est la 2.4GHz car plusieurs personnes peuvent utiliser cette fréquence sans risquer d'interférences mais avant, des fréquences de l'ordre du MHz étaient utilisées. Ces fréquences dépendent du pays mais en France, on pouvait utiliser 42Mhz et 58Mhz. Ces fréquences ont l'avantage d'avoir une plus grande portée.

3.2 Bluetooth

On peut utiliser le Bluetooth pour faire communiquer deux appareils à courte distance. C'est en général ce qui est fait pour transmettre des informations entre 2 Raspberry Pi.

Le Bluetooth utilise 2.4GHz comme le Wifi mais il n'est pas utilisé pour les mêmes applications. Il permet un débit de données plus faibles et à une plus faible portée. En contrepartie, le Bluetooth consomme moins de courant que le Wifi. Les protocoles de communication ne sont aussi pas les mêmes, tout comme les organismes qui gèrent les normes pour le Bluetooth et le Wifi.



FIGURE 3.2: Radiocommande



FIGURE 3.3: Adaptateur bluetooth

3.3 Module Xbee

XBee est une marque qui produit des interfaces réseaux. Il en existe une grande diversité. On peut les distinguer par leur architecture réseau, la fréquence utilisée, le type d'antenne et la portée/puissance du signal.

3.3.1 Architecture réseau

Différentes architectures réseaux ont été développées pour les modules Xbee en fonction des besoins.

-
-



FIGURE 3.4: Module XBee

Chapitre 4

Cartes électroniques

4.1 Arduino



FIGURE 4.1: Différentes cartes Arduino

Les cartes Arduino sont les cartes de prototypage les plus populaires de part leur simplicité d'utilisation et leur rapport qualité-prix. La plus répandue est la carte Arduino Uno.

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 vraie 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, en plus puissante. Elle possède notamment plus de sortie série. Elle se programme exactement de la même manière qu'une carte arduino.

4.4 Shield

Les différentes cartes électroniques ne possèdent pas tous les capteurs ou sorties nécessaires pour un projet. Pour compenser ces problèmes, des Shield ont été créés. Ce sont des extensions des cartes électroniques très faciles d'utilisation. Cela coûte bien sûr moins cher de les faire soit même mais on est certain au moins que c'est fiable avec un shield.

Différents shields :

- GNSS
- Servo

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.

Avantages :

- Position absolue
- Donne aussi la vitesse du véhicule (dans)

Inconvénients :

- La précision des mesures dépend fortement de la qualité du signal. En milieu urbain, les immeubles bloquent certains satellites et peuvent générer des multipaths.

Fournisseurs : Ublox, Emlid,...

Il existe différents modes de positionnement par GNSS aujourd'hui dont certains peuvent donner une précision allant jusqu'au centimètre.

5.1.1 NMEA

5.1.2 Mode de positionnement

Single

PPP

Différentiel

5.2 LIDAR

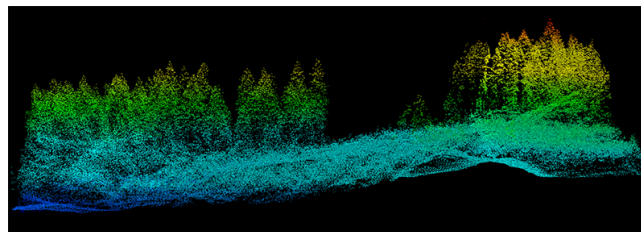


FIGURE 5.1: LIDAR de SLAMTEC

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. Contrairement à un capteur infrarouge actif, le laser du LIDAR est très directif et permet

des mesures plus précises. 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.



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.

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

On distingue 2 types de capteurs infrarouges, les capteurs passifs qui mesurent juste la lumière ambiante et les capteurs actifs qui génèrent aussi des ondes infrarouges. On pourrait aussi parler des caméras infrarouges, mais ce sont simplement des caméras qui mesurent captent les infrarouges en plus des ondes du domaine visible.

Avantages :

- Peu cher
- Simple à mettre en oeuvre

Inconvénients :

- Très sensible à la lumière ambiante, les éclairages traditionnels émettent aussi des infrarouges qui peuvent induire le capteur en erreur.



FIGURE 5.2: Capteurs infrarouges Actif (gauche) et passif (droite)

5.4.1 Utilisation

Le capteur infrarouge actif est en général utilisé en tant que capteur d'obstacle. En fonction de la quantité d'ondes infrarouges reçues, on peut estimer la distance de l'obstacle. Le problème est que cette quantité varie en fonction de l'éclairage de l'environnement (notamment avec des projecteurs). Pour compenser ce problème, il est nécessaire de calibrer les capteurs infrarouges à chaque lancement.

La phase de calibration peut consister définissant le taux moyen d'infrarouge dans l'environnement en mesurant sur une certaine période ou en redéfinissant la valeur limite.

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

5.6 Codeurs incrémentaux



FIGURE 5.3: Codeur Incrémental

Les capteurs incrémentaux mesurent une rotation.

Il existe des codeurs incrémentaux absolus et relatifs. Cependant, les codeurs absolus sont très chers. En effet, chaque position doit avoir sa m

5.7 Caméra

Il existe 2 types majoritaires de capteurs photographiques : CDD et CMOS. Cependant, un nouveau type de capteur fait se popularise dans le milieu de la robotique : DVS.

5.7.1 CDD

5.7.2 CMOS

5.7.3 DVS

Un nouveau type de caméra fait son apparition : DVS (Dynamic Vision Sensor). Ce type de caméra n'enregistre pas une image à chaque pas de temps. Elle suit le même principe que les récepteurs lumineux dans nos yeux et enregistrement seulement les variations de luminosité. Ces caméras ont beaucoup de potentiel. En effet, puisque seulement la variation est enregistrée, il n'y a pas d'informations redondantes et les programmes peuvent être plus rapides. De plus, ces caméras sont plus aptes à enregistrer les mouvements rapides de part leur nature. Contrairement aux capteurs CDD et CMOS, il n'y a pas " de perte d'inforamtion" entre 2 images. Problème : les DVS coûtent très chères et ont encore une très faible résolution pour le prix demandé.

5.7.4 Matériel

Modèle :

Fabricant :

Fournisseur :

5.7.5 Utilisation

Lecture de codes couleur :

Détection d'obstacle :

Positionnement d'objets sur le terrain :

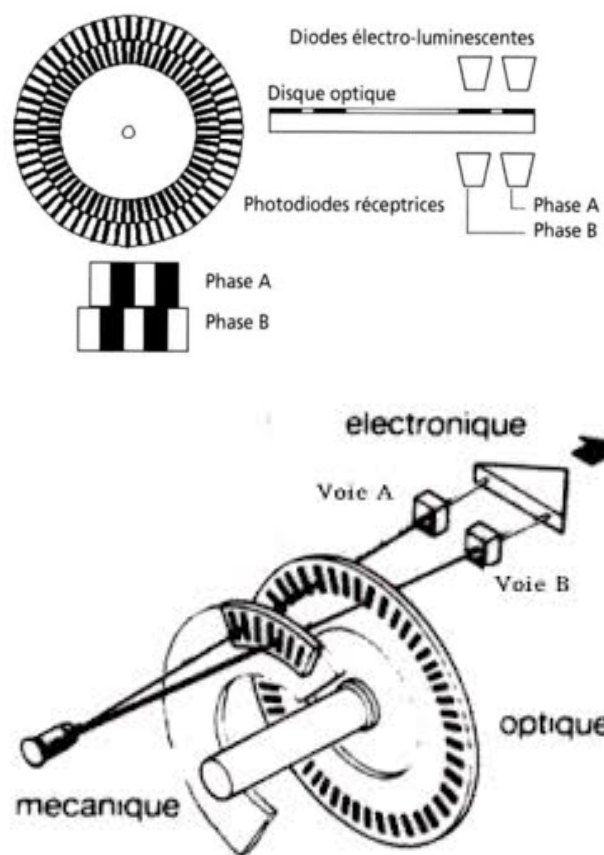


FIGURE 5.4: Schema de codeurs incrémentaux

Chapitre 6

Alimentation

6.1 Batterie

Il existe 2 types majoritaires de batterie chimique. Les Lithium Polymère (LiPo) et les Nickel Mercure (Nimh).

6.1.1 Lipo

Ces batteries présentent la meilleure densité énergétique. Cependant, elles sont aussi très dangereuses. Elles peuvent en effet exploser et provoquer un feu chimique difficilement arrêtable.

Les choses suivantes peuvent causer l'explosion d'une LiPo.

- Mouillée
- Percée
- Recoit un choc
- Laissée au soleil
- Court-circuit

Si jamais la LiPo est bombée ou a des bosses, il faut absolument s'en débarrasser. En attendant, la meilleure solution est de la placer dans un seau en métal ou sur une surface non-inflammable (pierre, métal, ...). Ce type de batterie doit être jeté à la déchèterie dans la section réservée pour.

6.1.2 Nimh

Les batteries Nimh sont moins fragiles que les batteries Lipo. Cependant, elles ont une plus faible densité énergétique.

6.1.3 Chargeur

Les différents types de batterie ne se chargent pas de la même façon. Il y a un mode pour chaque sur le chargeur. Il faut aussi surveiller le taux de charge.

Pour les Lipo, il faut aussi brancher la prise d'équilibrage. Ce n'est pas obligatoire mais cela permet de prolonger la durée de vie de la batterie. En effet, au fur et à mesure des différents cycles de charge/décharge, les éléments de la Lipo ne vont pas avoir tous la même tension. La prise d'équilibrage permet d'empêcher cela.

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

Chapitre 7

Composants électroniques

7.1 Fusible

7.1.1 PTC

7.1.2 Fusible réinitialisable

7.1.3 Thermistor

Troisième partie

Programmation

Chapitre 8

Raspbian

8.1 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.

8.2 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.

8.2.1 Les commandes utiles

cd :

ls :

mkdir :

nano :

vim :

chmod :

apt install :

apt upgrade :

apt update :

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

8.2.2 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 9

Le code

Quel langage utilisé ?

9.1 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++).

9.2 C/C++

9.2.1 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 10

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 11

Traitement d'images : OpenCV

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

11.1 Systèmes de couleur

De part la manière dont nous voyons le monde, nous utilisons en général le système RGB pour décrire les couleurs. Cependant, ce n'est pas toujours le système de couleur le plus efficace pour étudier des images. En effet, certains éléments ressortiront plus dans un certain système de couleur. Par exemple, les feux de forêt sont plus distinct avec la luminescence du système LUV qu'avec le rouge de RGB.

11.2 Odométrie visuel

<https://link.springer.com/article/10.1186/s40064-016-3573-7>

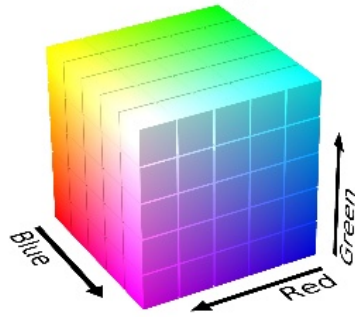
11.3 Vision stéréoscopique

A partir de deux caméras dont on connaît la position relative, il est possible d'obtenir la profondeur de l'image. Cependant, cette méthode pour obtenir la profondeur est très sensible à différents paramètres tels que la calibration des caméras et la distance entre les caméras. Par ailleurs, la carte de profondeur générée est très approximative. Si on veut vraiment mesurer la profondeur d'une image, on utilisera plutôt une caméra infrarouge telle que la Kinect ou la Intel Realsense D435.

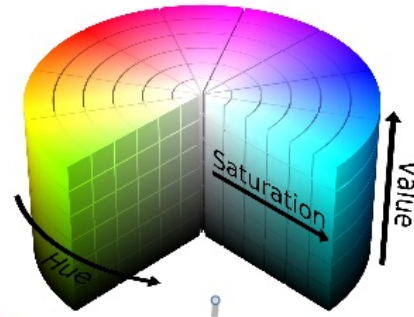
RGB Vs HSV (Color Spaces - a common mistake...)

*Slide produced using POV-Ray (<http://11b.povray.org/>)

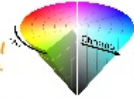
RGB Color Space



HSV Color Space

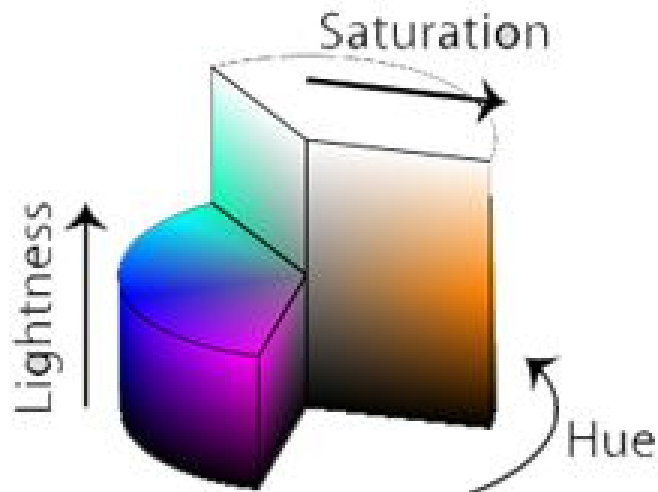


A cone is commonly adopted but cylindrical representation is more mathematically accurate.



Michel Alves - Five Minute Speech (FMS) - Overview of Activities Developed in Disciplines and Guided Studies - Graduate Program in Systems Engineering and Computing

HSL



HSV

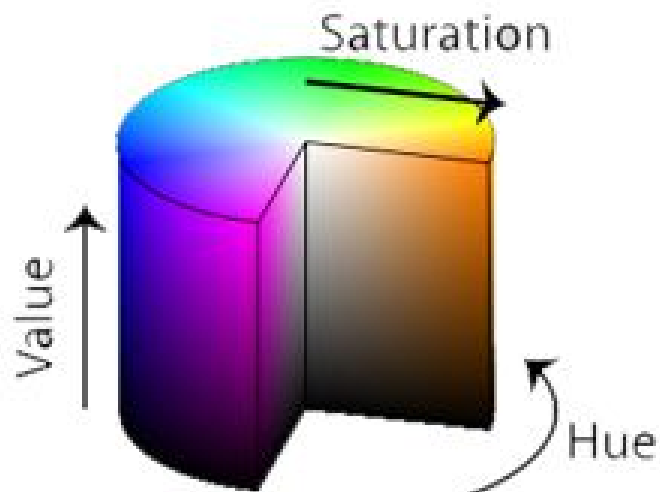


FIGURE 11.1: Systèmes de couleur



FIGURE 11.2: Profondeur d'image avec des stéréos caméras

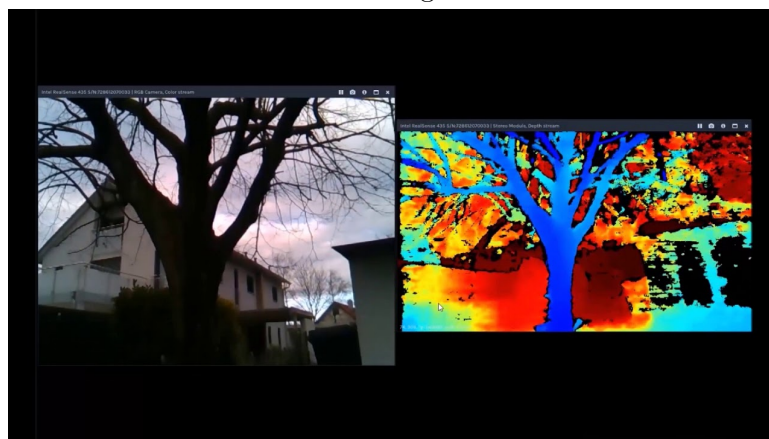


FIGURE 11.3: Profondeur d'image avec RealSense D435 utilisant une caméra infrarouge

Chapitre 12

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)
- Il est très souvent utilisé dans le milieu de la robotique

Cependant, ROS demande une certaine prise en main. Pour apprendre à l'utiliser, il a plusieurs solutions. Personnellement, je conseille ce livre disponible en version PDF "ROS Robot Programming Book" . Il rassemble tout ce qui est nécessaire de savoir.

Cependant, pour avoir de la pratique, investir dans la plateforme IgniteAcademy est une très bonne idée. Un abonnement au mois coûte 39 euros. Je l'ai essayé et ça vaut vraiment le coût.

Chapitre 13

Utilisation d'un Git

Un Git est très pratique pour coder. Il permet travailler à plusieurs sur un même projet et surtout il permet de gérer l'évolution du code.

Il existe plusieurs services qui propose hébergements en ligne : Github, Gitlab,...

13.1 Méthodologie

1. Avant de commencer, il faut récupérer la dernière version disponible du code.
2. Une fois satisfait avec les modifications réalisées sur un fichier, on enregistre la nouvelle version (vérifier que le code marche). Il faut mettre un commentaire au commit afin de garder une trace des modifications réalisées.
3. Quand on a finit de travailler sur le code, on sauvegarde les modifications sur le serveur.

Chapitre 14

Protocole de communication

Pour faire communiquer différents appareils électroniques entre eux, il existe plusieurs méthodes.

14.1 Série

14.1.1 UART

14.1.2 TTL

14.2 I2C

14.3 SPI

14.4 CAN

Ce protocole de communication a été développé par Bosch pour réduire la quantité de câble dans les voitures. Il permet en effet de faire communiquer plusieurs appareils avec seulement 2 fils.

Quatrième partie

Automatique

Le livre "Probabilistic Robotics" est excellent pour découvrir les différents problèmes de robotique. Il sert de très bonne introduction aux algorithmes de fusion de données pour estimer la position, au contrôle de robots et au SLAM.

Chapitre 15

Estimation de l'attitude

15.1 Différentes manières de représenter l'attitude

15.1.1 Angles d'Euler

15.1.2 Vecteur rotation

15.1.3 Quaternion

15.2 Filtre complémentaire

15.3 Filtre de Mahony

15.4 Filtre de Madgwick

15.5 Filtre de Kalman

Cinquième partie

Machines de la menuiserie

Chapitre 16

Découpeuse Laser

Chapitre 17

Tour

Chapitre 18

Imprimante 3D

Sixième partie

Conception

Chapitre 19

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 20

CAO

Septième partie

La Coupe de France de Robotique

Chapitre 21

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 22

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 23

La compétition

23.1 Les matches

- 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.