Robot Roulant 2021

Pour Toulouse Robot Race



| Revision | Date | Change |
| --- | --- | --- |
| -- | 16/08/2021 | Creation |
| -A | 29/08/2021 | Essentiellement sur les liaisons radio de télémétrie |
|  |  |  |
|  |  |  |
|  |  |  |

Table des matières

[1. Contexte 3](#_Toc80024054)

[2. Les grandes lignes du projet 4](#_Toc80024055)

[3. Conception détaillée 5](#_Toc80024056)

[3.1 Le châssis roulant 5](#_Toc80024057)

[3.2 Les capteurs 6](#_Toc80024058)

[3.3 La télémétrie 8](#_Toc80024059)

[3.3.1 Le type de liaison 8](#_Toc80024060)

[3.3.2 Le logiciel distant 9](#_Toc80024061)

[3.4 Le calculateur 10](#_Toc80024062)

[3.5 IVQ 12](#_Toc80024063)

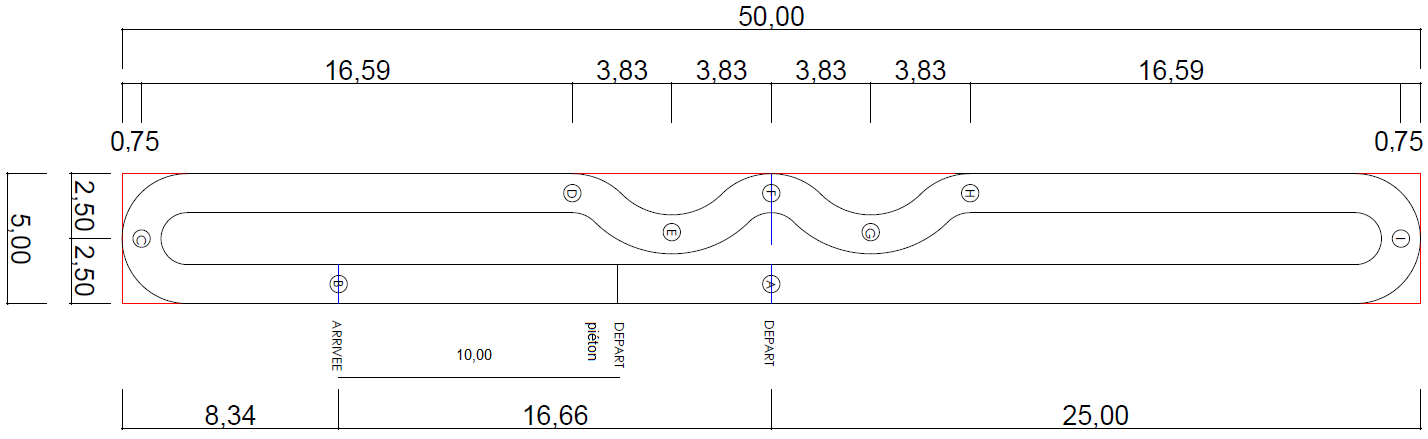
[4. Conclusion 13](#_Toc80024064)

# Contexte

La section propose de concevoir un roulant pour la prochaine TRR (Toulouse Robot Race). Vu le contexte sanitaire, pour l’instant nous ne nous engagerons pas sur une date, mais ça pourrait être mi Décembre 2021.

Pour ceux que veulent plus de détail, voici le site Internet : [Toulouse Robot Race - #TRR2021 - La course de robots (toulouse-robot-race.org)](http://www.toulouse-robot-race.org/" \l "presta)

L’idée est de faire un robot roulant électrique, qui évolue sur un parcours de ce type :



La piste fait environ 1,5m de large. Il y a au milieu une ligne blanche de 5cm de large, sur fond noir de 15 cm.



La ligne de départ est matérialisée par une ligne verte. La ligne d’arrivée est matérialisée par une ligne rouge et une arche.

Le but est de faire un (ou deux) tours le plus rapidement possible.

La majorité des concurrents mettent en œuvre de l’IA, mais des essais rapides lors de notre dernière participation ont montrés qu’un simple suiveur de ligne (ou de bord) peut suffire.

Nous avons énormément travaillé sur l’IA lors de la dernière participation. Nous vous proposons cette année de faire un suiveur simple, mais efficace ! 😊.

La conception du robot est découpée dans ce document en plusieurs blocs, ce qui permet à tout à chacun de participer à tout ou partie de l’aventure, suivant son temps et ses envies.

# Les grandes lignes du projet

L’idée est de partir sur un châssis comme ceux de 2019. Ce sont des voitures modèle réduit au dixième, capable d’aller vite (ndlr bonne motorisation, bonnes suspensions, bonne stabilité dans les virages …).

La compétition nous impose moteur électrique.

Sur ce châssis à motoriser, application d’une plaque support sur laquelle l’électronique est installée (proc, miniLidars, gyro, boutons, leds, moyens de com pour mise au point …).

L’idée est de ne pas embarquée d’IA, donc le proc sera simplement type STM32 ou équivalent.

Suivant l’avancement du projet, nous pourrions :

- imaginer une carrosserie personnalisée,

- faire une version du robot avec lidar à l’avant pour participer à la catégorie DLVV.

# Conception détaillée

Afin de mener le projet à terme et de permettre à chacun de participer sur une partie seulement si il le souhaite, la conception est découpée par bloc.

## Le châssis roulant

Le châssis 1/10ième doit permettre d’atteindre 10m/s sur un circuit comme présenté en introduction de la compétition.

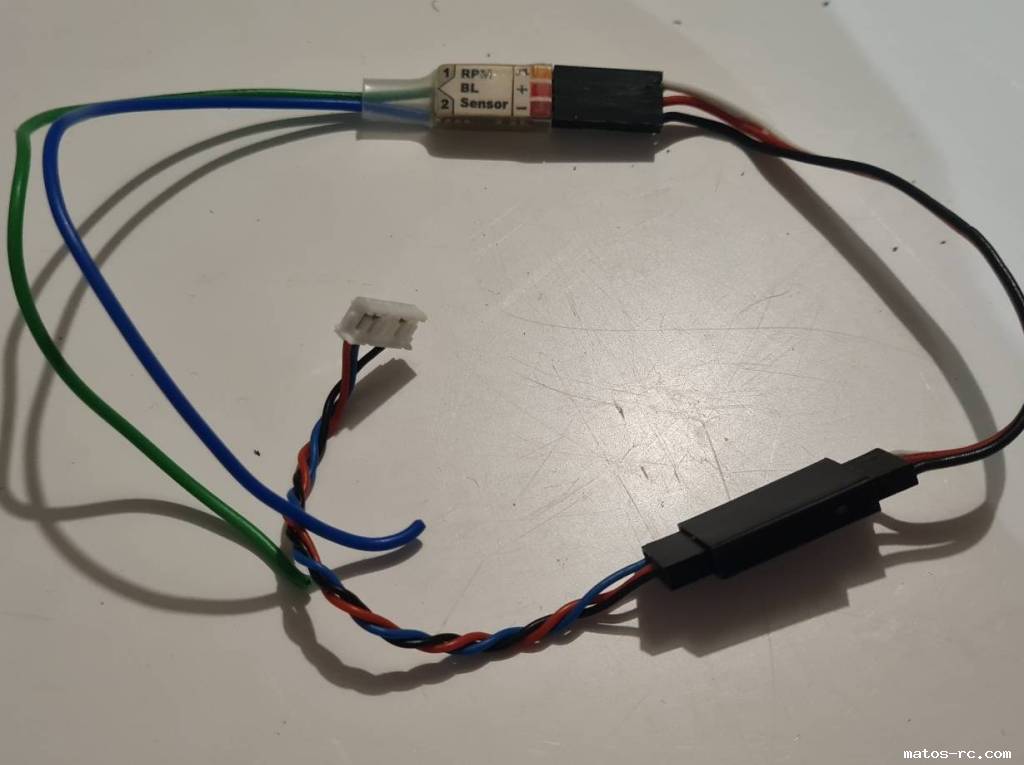
Il comprend principalement la mécanique, la motorisation, une transmission 4x4, le pack batterie, un capteur de vitesse, idéalement un capteur de direction.

La motorisation pilotable est pilotable par PWM (ou équivalent) afin de permettre au calculateur de modifier la commande de vitesse rapidement (Rapidement est subjectif, certes. Disons une fois toutes les 10 ms).

Le modèle envisagé actuellement (à discuter/confirmer) est le châssis Tamiya TA-08 ([Tamiya TA-08 Pro KIT 58693 - RC Team](https://www.rcteam.fr/voitures-de-piste-kit/42733-tamiya-ta-08-pro-kit-58693-4950344586936.html?gclid=CjwKCAjw9uKIBhA8EiwAYPUS3BjsBxavYgXLpvEVAvvGBnl56Bed3MOVYIvUmHSoJYi8isYmjAIIHRoC9IEQAvD_BwE)).



Le capteur de vitesse est le RPM BL Sensor de BeastTX.com, car déjà utilisé sur le robot de 2019, donc connu. Son utilisation implique un moteur brushless, ce qui de toute façon est déjà la solution envisagée.



Le capteur de direction est aujourd’hui un gyro sur la carte processeur. Etudier si il est possible de mettre un capteur sur la direction, et si il y a un intérêt (en terme de précision/Rapidité de mesure, etc …).

Suivant l’avancement du projet, nous pourrions imaginer une carrosserie.

## Les capteurs

Comme évoqué dans le paragraphe précédent, l’odométrie est assurée par :

* Un capteur de vitesse sur le moteur brushless,
* Un capteur de direction qui pour l’instant est le gyro.

Pour la position sur la piste, le robot de 2019 avait des miniLidars qui donnaient de bon résultat. Il en faut au moins 3 :

* Un vers le haut pour détecter l’arche, donc la ligne d’arrivée,
* Deux avec un angle à déterminer pointant à gauche et à droite,
* Eventuellement deux autres à 90° mais l’utilité reste à discuter.

[Capteur de distance LIDAR TF-MINI-PLUS Benewake - Distance | GO TRONIC](https://www.gotronic.fr/art-capteur-de-distance-lidar-tf-mini-plus-28966.htm)



De [www.benewake.com](http://www.benewake.com) :

Our CE30 series solid state LiDAR is widely used for AGV/UGV/Mobile Robotics/Robots/Vehicles/Safety, etc.  and TF series single point LiDAR is widely used for IoT/ITS/smart city/traffic statistics/vehicles counting/people counting/level sensing/security detection,etc. Below are the most popular models FYI:

**TF350** ranging **350m**, FOV 0.7°, 1~1000Hz, IP67. UART/CAN/RS485/RS232/4-20mA.

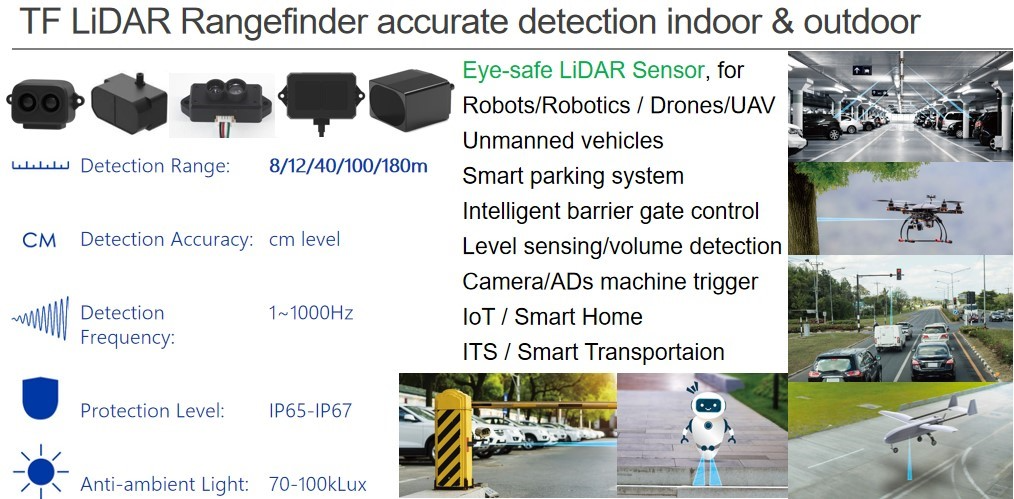
**TF03-180** ranging **180m**, FOV 0.5°, 1~1000Hz, IP67. UART/CAN/RS485/RS232/4-20mA.

**TF02-Pro/TF02-i** ranging 0.1~40m, FOV3°, 100Hz, IP65.  UART/I2C or CAN/RS485.

**TFmini-S/TFmini-i**, ranging 0.1-12m, FOV2°, 1~1000Hz. UART/I2C or CAN/RS485.

**TFmini Plus** ranging 0.1-12m, FOV 3.6°, 1-1000Hz, IP65. UART/I2C/IO. US$43.9/PCs.

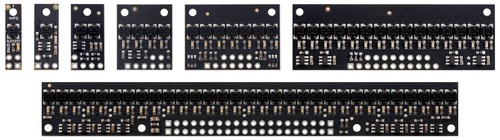
**TF-Luna,**ranging 0.2-8m, FOV2°, UART/I2C. US$24/PCs.



Par rapport à 2019, ajoutons un capteur de la ligne blanche sur fond noir, qui devrait nous permettre d’utiliser les algos des suiveurs de ligne de la compétition de Nîmes. Les tailles des robots et des lignes sont différentes, mais le principe reste le même.

Ce capteur reste à trouver ou concevoir en utilisant le principe des cartes POLOLU.

[Pololu QTR Reflectance Sensors](https://www.pololu.com/category/123/pololu-qtr-reflectance-sensors)



Son utilisation est à discuter car pour faire un tour de circuit le plus rapide possible, il ne faut pas rester au milieu de la piste, mais calculer les trajectoires pour la prise des virages. A méditer.

Si nous participons au DLVV, il faudra prévoir un lidar tournant comme le UST-10LX que nous avons déjà utilisé, ou un plus simple genre [Télémètre laser YDLidar X4 (generationrobots.com)](https://www.generationrobots.com/fr/403022-telemetre-laser-360-yd-lidar-x4.html) .

Tous les capteurs remontent vers la carte processeur.

## La télémétrie

### Le type de liaison

Lors de la TRR 2019 nous avions remarqués des problèmes de liaisons entre le robot et le PC, dus sans doute à des moments où la liaison n’était pas directe. Pour rappel, la liaison Wifi était entre la Jetson (2 antennes) et un dongle USB [Archer T2U Plus | Adaptateur USB WiFi bi-bande AC 600 Mbps à gain élevé | TP-Link France](https://www.tp-link.com/fr/home-networking/adapter/archer-t2u-plus/) sur PC (pour avoir plus de gain que l’antenne intégrée dans le portable).



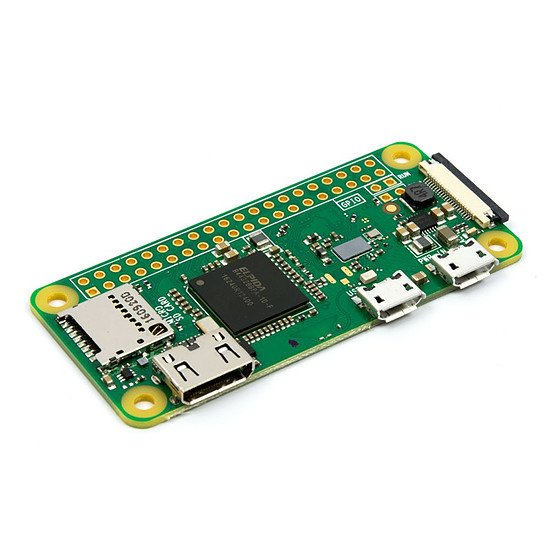
2 types de liaisons s’offrent à nous, le Wifi et le XBee (433 MHz / 868 MHz / 2.4 GHz).

#### Le Wifi

Tests sur Raspberry 3B+ et la Raspberry Zero qui existent déjà à la section.

Sur la Raspberry Pi 3B+, mise en place faite d’un noyau à jour avec le Wifi en AP (Carre92Roulant1), connexion d’un adaptateur USB to Serial FTDI, et activation d’un service ser2net.

Avec ce service, le PC connecté à la raspberry peut lancer un telnet sur le port 4001 et piloter la liaison série.



Raspberry Zero

Des tests sont en cours sur une Raspberry Zero par François. L’avantage est la consommation réduite par rapport à la 3B+, donc ESC moins gros.

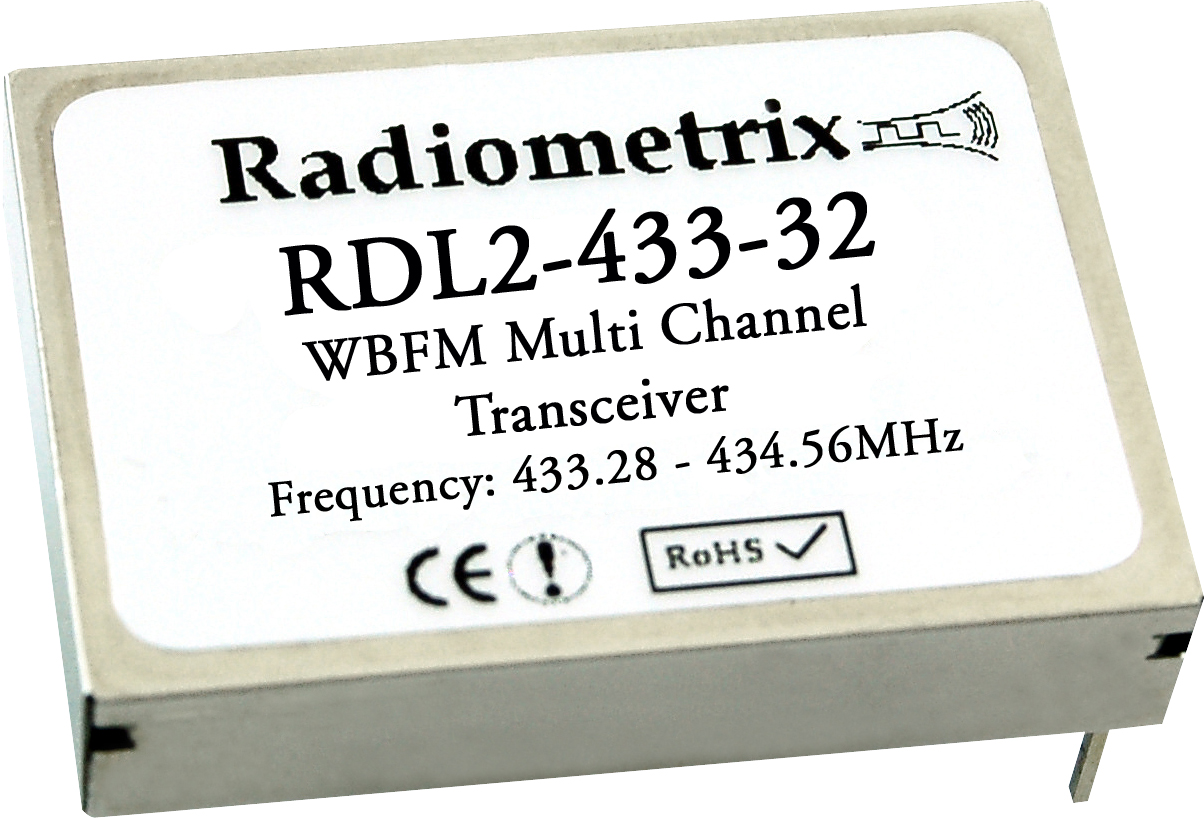
#### Les solutions alternatives

D’autres radios sont disponibles pour la robotique :

* Le 433 MHz : cette bande est initialement prévue pour la domotique. Les principaux modules disponibles sont soit émetteurs soit récepteurs. (Il est rare que les volets électriques aient quelque chose à dire à la télécommande 😊). Le débit est de quelques kbauds ce qui est limité pour remonter des fichiers de télémétrie.

Voici quelques exemples :

* [Radiometrix - Radio Modules - RF Modules - Wireless Modules | RDL2](http://www.radiometrix.com/content/rdl2-1)



* Un exemple chinois moins cher : [1W Full Duplex TCXO 433MHz rf Module ebyte E62 433T30D Long Range Wireless Transceiver iot Transmitter and Receiver|Fixed Wireless Terminals| - AliExpress](https://www.aliexpress.com/i/32977311505.html)



* Le XBEE/Zigbee (868 MHz ou 2.4 GHz) : ces modules ont été développés pour la robotique. Ils sont moins volumineux et moins consommant qu’un module Wifi. Ceci dit, un ESP-01 ne consomme pas beaucoup, avec une portée sans doute moindre.

Voici quelques exemples :

* [Module XBee série 2C XB24CASIT-001 Digi - Xbee | GO TRONIC](https://www.gotronic.fr/art-module-xbee-serie-2c-xb24casit-001-25785.htm)

Le 802.15.4 est à la base du Zigbee et autres. Contrairement au Wifi, elle permet un module plus petit et moins consommant Je ne vois pas de redondance accrue des messages, et de durcissement de la liaison par rapport au Wifi. C’est présenté comme étant juste moins gros/moins consommant.

* [Modem radio 869.85 MHz "RPM3B-869-17" Radiometrix - Lextronic](https://www.lextronic.fr/modem-radio-rpm3b-869-17-760.html)

Nous avons besoin de débit pour que la remontée des infos de télémétrie ne prenne pas trop de temps.

#### Sélection

Dans un premier temps, implémentation de la télémétrie par Wifi avec une Raspberry Pi Zero. Ce qui laisse le temps d’analyser d’autres modules ensuite.

### Le logiciel distant

Il serait intéressant que le logiciel distant puisse s’exécuter sur PC et sur tablette. Pour les essais d’intégration et le jour avant la compétition, la tablette est moins volumineuse à déplacer.

L’IHM pourrait être implémentée sur le distant (PC ou tablette), ou sur un serveur http sur une Raspberry qui sert à la liaison radio.

Voici quelques idées de ce que pourrait contenir le logiciel :

* 1 fenêtre d’affichage et de mise à jour des paramètres de conf,
* 1 fenêtre de pilotage (Start, Stop, Auto, Manuel),
* 1 fenêtre d’affichage des données brutes de télémétrie avec possibilité de stockage et chargement dans un fichier,
* 1 fenêtre de courbes. L’utilisateur choisi dans une liste des champs de télémétrie, ce qu’il souhaite afficher,
* 1 fenêtre pour le parcours :
  + Affichage sur la base de l’odométrie,
  + Calcul d’une trajectoire optimale et chargement dans le robot.

#### IRS entre la STM32 et le logiciel de Télémétrie

## Le calculateur

Le calculateur :

* Récupère les informations des capteurs (miniLidars, capteur vitesse, capteur direction ou gyro, détecteur de ligne),
* Calcul les consignes de vitesse et direction,
* Applique les consignes sur les moteurs et servo moteurs,
* Dispose d’une IHM permettant de sélectionner les consignes de vitesses, départ, arrêt d’urgence (c’est dans le règlement, il faudra trouver qui court après le robot pour l’arrêter ☺).
  + Boutons et voyants,
  + Ecran et joystick (type robot labyrinthe).

Dans un premier temps, réutilisation de la carte existante de 2019 à base de STM32, puis design d’une nouvelle carte si besoin.

Nous avons aussi la possibilité de partir d’une carte standard (Nucléo ou Raspberry) pour faire des essais.

En cas de consignes reçues sur la voie Radio Télécommande, ces consignes sont prioritaires à celles calculées par l’algo (sécurité).

Un PID bas niveau gère le suivi de la consigne de vitesse. Un autre gère le suivi de la consigne de direction.

Au-dessus, quelques modes de fonctionnement pour comparaison :

* Mode 2019 :

Le robot calcul sa position vis-à-vis des murs fonction des lidars gauche et droite, et corrige sa direction. Vitesse constante.

* Mode suivi de ligne actuel :

Le robot suit la ligne en corrigeant la direction pour maintenir sa position sur la ligne. Les lidars gauche et droit servent à mettre des grands coups de volant pour se ramener au milieu quand on se rapproche trop près d’un mur.

Au premier tour, il apprend le tracé. Aux tours suivants, il accélère dans les lignes droites et anticipe son arrivée sur un virage en freinant, tout en maintenant sa position sur la ligne.

Les vitesses ligne et virage sont configurable par IHM/télémétrie.

* Mode suivi de trajectoire avec apprentissage :

Au premier tour, le robot suit la ligne en corrigeant la direction pour maintenir sa position sur la ligne. Les lidars gauche et droit servent à mettre des grands coups de volant pour se ramener au milieu quand on se rapproche trop près d’un mur. Il apprend le parcours.

Aux tours suivants, il optimise sa trajectoire et sa consigne de vitesse. La trajectoire optimisée ne passe pas forcément par la ligne du milieu.

Les vitesses ligne et virage sont configurable par IHM/télémétrie.

* Mode suivi de trajectoire :

Le circuit est découvert lors des phases de mise au point, le jour précédent la compétition. La trajectoire optimisée et la vitesse (en tout point ou par tronçon) sont pré calculées et chargées dans le robot.

Le robot suit la pré programmation en se recalant sur la base des informations des mini Lidars gauche et droit.

Si participation à la catégorie DLVV, il faudra prévoir une interface Ethernet ou USB sur le calculateur.

## IVQ

Nous avons la possibilité de réutiliser le deuxième robot de 2019 dès le début du projet, pour mettre au point :

* L’environnement de développement SW,
* La liaison de télémétrie,
* Développement de certaines briques logicielles,
* …

Les performances de vitesse et stabilité de ce proto seront limitées, mais ce n’est pas l’objectif au début.

Cela laisse le temps de développer le châssis final sans que le reste du développement soit bloqué.

L’ordre des tests me semble être :

* Test de la liaison de télémétrie, (radio et mise au point du logiciel). Une bonne télémétrie permet de gagner du temps ensuite.
* Test des capteurs. C’est sur leurs résultats que e base l’algorithme ensuite. Si les mesures ne sont pas fiables, la suite est compliquée.
* Test des algos. Patrick, est ce qu’un environnement de simulation type pyBullet pourrait nous aider ? Gain temps de mise en place de la simu par rapport à essai en direct sur châssis.
* Test du robot complet en réel. 2 possibilités :
  + Le couloir du CE vers 17H/18H.
  + Le bâtiment I en achetant des basting ou planches et équerres pour faire les murs.

Dans tous les cas, il faut retrouver les bandes de lignes blanches au local, voir continuer la fabrication pour avoir plus de longueur.

# Conclusion

On gagne le premier prix dans toutes les catégories !!!!!