

BRUANT Jean
jean.bruant@telecom-bretagne.eu

CADOT Arnaud
arnaud.cadot@telecom-bretagne.eu

JOUANNY Théo
theo.jouanny@telecom-bretagne.eu

NERON François
francois.neron@telecom-bretagne.eu



Rapport projet 66

Projet Tourelle

Rapport technique

Version 2

Formation d'ingénieur généraliste

Projet développement

Année scolaire 2014-2015

Tuteur : KEROUEDAN Sylvie



Sommaire

1. RÉDACTEURS ET RELECTEURS.....	5
RÉSUMÉ.....	6
ABSTRACT.....	6
INTRODUCTION.....	7
2. PRÉSENTATION DU PROBLÈME.....	8
2.1 CONTRAINTES.....	8
1.2 SOLUTIONS ENVISAGÉES.....	8
1.2.1. Faisceau direct vs. faisceau réfléchi.....	8
1.2.2. Capteur tournant : conception.....	9
1.2.3. Support de la plate-forme supérieure.....	9
1.2.4. Électronique embarquée.....	10
1.3. SOLUTION RETENUE.....	10
3. RÉALISATION DU SYSTÈME.....	11
3.1 PARTIE MÉCANIQUE.....	11
3.1.1 Dimensionnement du moteur.....	11
3.1.2 Fabrication du support.....	11
3.1.3 Transmission, support capteur et compte-tour.....	12
3.1.4 Modélisation Solidworks.....	12
3.2 PARTIE INFORMATIQUE.....	12
3.2.1 Protocole.....	12
3.2.2 Arduino.....	13
3.2.3 Robot.....	14
3.3 PARTIE ÉLECTRONIQUE.....	15
3.3.1 Contrôle du moteur.....	16
3.3.2 Compte tour.....	16
3.3.3 Détecteur de balise.....	16
3.3.4 Carte Arduino.....	17
3.3.5 Prototypage.....	17
3.3.6 Résumé des interfaces avec le reste du robot.....	17
4. PRÉSENTATION DU PRODUIT FINI.....	17
4.1 PHASE DE TEST.....	17
4.1.1 Programmation Processing.....	18
4.1.2 Fonctionnalité principale.....	18
4.1.3 Contraintes réglementaires.....	18
4.1.4 Distances limites.....	19
4.1.5 Précision angulaire.....	19
4.1.6 Résistance au bruit.....	19
4.1.7 Puissance requise.....	19
4.2 PERSPECTIVES.....	19
5. CONCLUSION.....	20
6. RÉFÉRENCES BIBLIOGRAPHIQUES.....	21
6.1 RÉGLEMENT DE LA COUPE DE FRANCE DE LA ROBOTIQUE 2015.....	21

6.2 DATASHEETS..... 21
6.3 DOCUMENTATIONS..... 21
7. GLOSSAIRE.....21
ANNEXES.....23

1. RÉDACTEURS ET RELECTEURS

RÉDACTEUR

<u>Résumé</u>	François
<u>Abstract</u>	François
<u>Introduction</u>	François/Théo
<u>1. Présentation du problème</u>	
<u>1.1 Contraintes</u>	Théo
<u>1.2 Solutions envisagées</u>	
1.2.1. Faisceau direct vs. faisceau réfléchi	Arnaud
1.2.2. Capteur tournant : conception	Arnaud
1.2.3. Support de la plate-forme supérieure	Arnaud
1.2.4. Électronique embarquée	Arnaud
<u>1.3. Solution retenue</u>	Arnaud
<u>2. Réalisation du système</u>	
<u>2.1 Partie mécanique</u>	
2.1.1 Dimensionnement du moteur	Théo
2.1.2 Fabrication du support	Arnaud
2.1.3 Transmission, support capteur et compte-tour	Arnaud
2.1.4 Modélisation Solidworks	Théo
<u>2.2 Partie informatique</u>	
2.2.1 Protocole	François
2.2.2 Arduino	Jean
2.2.4 Robot	François
<u>2.3 Partie électronique</u>	
2.3.1 Contrôle du moteur	Jean
2.3.2 Compte tour	Jean
2.3.3 Détecteur de balise	Jean
2.3.4 Carte Arduino	Jean
2.3.5 Prototypage	Jean
2.3.6 Résumé des interfaces avec le reste du robot	Jean
<u>3. Présentation du produit fini</u>	
3.1 Tests effectués	
3.1.1 Programmation Processing	Jean
3.1.2 Fonctionnalité principale	Théo
3.1.3 Contraintes réglementaires	Arnaud
3.1.4 Distances limites	Théo
3.1.5 Précision angulaire	Théo
3.1.6 Résistance au bruit	Théo
3.1.7 Puissance requise	Théo
3.2 Perspectives	Théo
<u>Conclusion</u>	Théo

RELECTEURS :

Tous les membres du projet

RÉSUMÉ

Il s'agit de la première version du rapport technique du Projet Développement n°66, autrement appelé Projet Tourelle. Ce projet consiste à la conception d'un système de détection de robots à moyenne distance dans le cadre de la Coupe de France de robotique. On y retrouve l'ensemble des contraintes et des objectifs du projet, ainsi que la réalisation technique du projet avec la conception du module de détection et de son interfaçage avec le robot. De plus, les perspectives d'utilisation seront abordées.

On retrouve dans les annexes le Cahier des Charges fonctionnel ainsi que le planning du projet.

ABSTRACT

This is the first version of the technical report of the 66th development project, also called Turret Project. This project aims to design a detection system for robots at medium distance for the French robotic Cup. We find in it all the constraints and the goals of the project, and also the technical conception of the Turret and its interface with the robot. Furthermore, the future of this project is studied.

The annex includes the functional specifications and the project's planning.

INTRODUCTION

Ce document est le rapport technique du projet développement n°66, qui correspond au projet Tourelle.

Le but de ce projet est de fournir un module supplémentaire au robot principal du club robotique de l'école. Ce club a des activités qui se recoupent avec le projet développement n°61, ou projet Robotique, ainsi que le projet Ingénieur associé. Ces projets ont tous deux pour objectif commun de concevoir intégralement deux robots afin de participer à la Coupe de France de robotique, qui se déroule mi-mai, à la Ferté-Bernard dans la Sarthe. Le club robotique participant tous les ans à cette coupe, il s'agit chaque année de tirer profit des enseignements du travail de l'année précédente, tout en innovant, car les règles changent chaque année. Cependant, les délais courts et la quantité de travail à fournir d'une année sur l'autre pour les objectifs de l'année en cours ne permettent pas aux membres de ces projets de concevoir certaines parties complexes qui pourraient être utiles aux robots tous les ans. Ils peuvent donc faire appel à d'autres groupes pour concevoir des modules supplémentaires au robot.

Une des problématiques de la compétition est l'évitement des robots adverses, car une collision se solde par une défaite automatique. C'est ici que la Tourelle intervient. Le projet Tourelle consiste à développer une solution de détection à distance des robots adverses adaptée au contexte de la coupe de robotique. En effet, même si le robot est actuellement équipé de capteurs de contact évitant les collisions, le module de ce projet permettrait de connaître la position des autres robots présents à moyenne distance (et non seulement à courte distance). Cela rendrait donc possible des déplacements du robot plus économiques, s'adaptant à la position des autres robots. Par exemple, le robot pourra décider d'éviter une certaine zone déjà encombrée par d'autres robots, au lieu de s'approcher de cette zone pour l'éviter ensuite une fois que les capteurs de contact détectent un risque de collision imminente. Il s'agit d'une tourelle car c'est au sommet des robots qu'est prévu l'emplacement pour de tels dispositifs dans le règlement de la coupe.

Le projet consiste donc à la réalisation complète du module de détection de robots à moyenne distance, en suivant les contraintes liées à la coupe, et implique une forte communication avec le reste des activités du club robotique, afin de rendre ce module intégrable au robot.

Ce projet nécessitant un travail sur les plans mécaniques, électroniques, et informatiques, c'est ainsi que s'est séparé le travail.

Ce document présente en premier temps l'ensemble des objectifs et contraintes de ce projet. Par la suite, la partie technique de la réalisation est abordée. Enfin, nous analysons les résultats de ce projet avant de conclure quand à ses futures utilisations.

2. PRÉSENTATION DU PROBLÈME

2.1 CONTRAINTES

La majorité des contraintes sont liées au règlement de la coupe^[1], ainsi qu'à des contraintes de coûts et d'équipements déjà présents.

La première contrainte concerne la taille de l'équipement posé sur le robot. En effet, notre balise doit pouvoir tenir dans une cube de 80 mm de côté : si notre balise dépassait, cela serait éliminatoire (côté du robot ou au dessus du robot). Nous avons cependant la possibilité de dépasser légèrement sur la face inférieure du robot si nécessaire (cela n'augmentant pas le volume total du robot).

La deuxième contrainte concerne le choix de capteur de position utilisé. Celle-ci est tout d'abord liée aux conditions d'utilisation du robot. En effet, la zone de la compétition étant bruyante, et de luminosité très variable, certaines technologies (caméras, système sonar, ...) ne sont pas envisageables. De plus, le système ne devra jamais repérer plus de trois robots en même temps, et ceux-ci ne seront jamais à une distance supérieure à trois mètres. Enfin, dans le cas d'un système optique, les technologies à base de laser ne doivent pas dépasser la catégorie I.

Il était aussi nécessaire d'avoir un temps de rafraîchissement sur la position des autres robots assez élevé, sachant que les distances entre robots varient entre 0,2 et 3m et que les robots peuvent atteindre la vitesse de 1 m/s.

Nous n'avons pas de contrainte majeure concernant la consommation de notre système, dans la limite du raisonnable, le système utilisant l'alimentation du robot principal.

De plus, nous n'avons pas non plus de contraintes de temps autres que celles imposées par le projet, étant donné que notre système doit être prêt pour le coupe de robotique 2016 pour laisser le temps à l'équipe s'occupant de gérer le robot d'intégrer notre système (entre autres : traiter les informations que nous envoyons au robot principal).

Enfin, nous devons nous servir le plus possible des ressources déjà existantes au Fablab, étant donné que nous n'avons aucun budget à notre disposition.

1.2 SOLUTIONS ENVISAGÉES

Il existe plusieurs solutions techniques à ce projet. Cependant, les contraintes temporelles et matérielles ne permettaient pas d'en explorer plus d'une.

Par ailleurs, les ressources utilisables ont *de facto* limité la portée de cette phase de recherche. Nous disposions pour ce projet de deux capteurs **SICK WL-100-P 3239**^[2] et notre budget ne nous permettait pas de nous en passer. De plus, aux dimensions de capteur doivent être ajoutées celles du connecteur type M3, qui rajoutent environ 40% en longueur de manière parfaitement superflue.

1.2.1. Faisceau direct vs. faisceau réfléchi

La plus importante distinction entre les variantes est la suivante : faut-il faire tourner le capteur, ou faut-il utiliser un miroir rotatif pour diriger le rayon dans la direction que l'on souhaite ?

La seconde solution a l'avantage de la simplicité mécanique. Il suffit de fixer le miroir à l'axe d'un moteur situé en position haute, et de placer le capteur en-dessous.

A l'inverse, la première solution implique une conception mécanique plus astucieuse. Le rotor du produit serait bien plus lourd et plus difficile à équilibrer. En outre, il faudrait résoudre le problème de l'interface électronique entre rotor et stator.

Par ailleurs, des tests et une modélisation plus avancée ont montré que :

- L'utilisation d'un miroir diminue la portée du capteur en-deçà des limites acceptables
- Le placement du moteur dans la partie haute force à passer des câbles verticalement le long de l'ensemble, avec tout ce que cela implique (angles morts, esthétique, etc.)
- Le placement du moteur est particulièrement problématique avec la première solution (cf. 1.2.2. Capteur tournant : conception)

1.2.2. Capteur tournant : conception

Dans l'hypothèse d'une solution de type 'capteur tournant', la place prise par les différents composants devient rapidement problématique.

En effet, il n'est pas désirable de placer le moteur en position haute, car le couple requis empêche l'utilisation d'un moteur 'plat' (exemple : petit ventilateur). De même, il devient nécessaire d'utiliser un collecteur tournant (3 fils minimum) avant de faire l'interface électronique rotor/stator. Dès lors, une solution technique s'impose, composée de :

- Un axe, long, solidement maintenu à ses deux extrémités par des roulements à bille
- Un support pour le capteur, fixé à l'axe
- Un collecteur tournant, relié d'une part au capteur, d'autre part à l'électronique
- Un moteur, parallèle à l'axe, qui l'entraîne à l'aide d'engrenages

Toujours pour des questions de place occupée, le collecteur tournant doit nécessairement se trouver sous l'axe, afin de maximiser l'élévation du capteur.

Il existe au moins deux possibilités de placement de l'axe, du moteur et du capteur :

- Si l'on place le capteur de manière verticale, alors il devient impossible de placer un moteur en parallèle si l'axe est centré. Une solution serait de décentrer l'axe autant que possible, afin d'avoir la place dans le coin opposé pour le moteur
- Si l'on place le capteur de manière horizontale, il devient alors possible de placer un petit moteur parallèlement à l'axe.

La première solution introduit de nombreux problèmes d'organisation sur l'axe vertical (moteur tête-en-bas, etc.). De plus, elle pose un risque de mauvaise interaction avec une solution utilisant un support en Plexiglass pour des raisons de diffraction^[4].

La seconde solution est plus simple mécaniquement. Elle libère une place non négligeable sur l'axe vertical. Malgré tout, elle nécessite un moteur de dimensions réduites mais qui possède le couple et la vitesse nécessaires à l'entraînement du rotor. Par ailleurs, la longueur de l'ensemble capteur+connecteur peut poser des problèmes sur le plan horizontal.

1.2.3. Support de la plate-forme supérieure

Le cahier des charges stipule que le haut de la tourelle doit être une plate-forme carrée horizontale, plate et de dimensions 80x80mm. Elle doit par ailleurs être recouverte de Velcro et être capable de supporter un poids de 400g maximum.

Une solution triviale serait de placer quatre colonnes (tiges, vis...) aux quatre coins du carré. Cependant, si cette solution possède l'avantage d'être très simple à mettre en place et très robuste, elle a l'inconvénient de créer des angles morts.

Une seconde solution serait d'utiliser un support cylindrique sur l'extérieur de la tourelle (i.e. son diamètre extérieur serait 80mm). Ce cylindre serait constitué d'un matériau complètement transparent à la longueur d'onde utilisée par le capteur* - par exemple l'acrylique type Plexiglass^[4]. Cette solution n'est pas sans inconvénients :

- Risque de perte de portée due à la transmittance du matériau, qui ne peut pas être 100%
- Risque de déviation du rayon dû aux changements de milieux (perte de précision des mesures...)
- Diminution de l'espace horizontal disponible autour du capteur (perte des coins, pertes dues à l'épaisseur du cylindre...)

Rajoutons également que l'accès au rotor devient compliqué. Si cela complexifie la maintenance et la fabrication, cela permet aussi de protéger les parties les plus sensibles en cas de problème.

* 680 nm, non précisé dans la datasheet mais donné sur le site constructeur

1.2.4. Électronique embarquée

La solution la plus simple et la plus pratique pour l'électronique embarquée est l'utilisation d'un microcontrôleur. Nous avons déjà des cartes Arduino Uno et Mini à notre disposition^[5], il n'a donc pas été nécessaire d'envisager d'autres solutions.

Il fallut ensuite répondre au besoin de la position angulaire du capteur. Il n'était pas possible d'utiliser des roues codeuses à cause de leur prix excessif et de notre budget réduit. Nous nous sommes résolu à utiliser un simple système de compte-tour et d'extrapoler la position du capteur à partir de là.

Au moins deux solutions pour le compte-tour ont été imaginées :

- Utiliser un interrupteur optique
- Utiliser un interrupteur magnétique

Dans le premier cas, une lame opaque est fixée sur le rotor et interrompt le faisceau de l'interrupteur une fois par tour. Dans le second cas, la lame est remplacée par un aimant.

Les deux solutions sont techniquement comparables.

1.3. SOLUTION RETENUE

Au final, la solution retenue est la suivante :

- Système à capteur tournant
- Axe centré
- Support cylindrique en Plexiglass sur la partie supérieure
- Interrupteur optique pour le compte-tour
- Arduino Mini pour le microcontrôleur

Ces choix ont été fait en fonction des avantages et inconvénients de chaque élément de solution, mis dans le contexte des outils et des composants à notre disposition.

3. RÉALISATION DU SYSTÈME

3.1 PARTIE MÉCANIQUE

3.1.1 Dimensionnement du moteur

Suite au choix fait de faire tourner le capteur, et non un miroir réfléchissant, il a tout d'abord été nécessaire de dimensionner le moteur. Nous avons fait le choix d'avoir une vitesse de rotation du moteur de 5 tours/s, il fallait donc trouver son couple minimum. Le système est tout d'abord équipé de deux roulements à billes **6902RS**, qui apportent chacun un couple résistant de 1,5 N.mm. Nous avons considéré que le temps acceptable pour atteindre la vitesse de rotation de 5 tours/s était de 2 secondes, ce qui nous donne une accélération angulaire de 31 rad/s^2 (dans le cas d'une accélération constante). Ces valeurs nous permettent de trouver un couple moteur minimum nécessaire de 4,4 N.mm. En sachant que notre système réel ne sera pas mécaniquement parfait, et qu'il y aura des couples de frottement parasites, nous avons décidé de prendre une marge de sécurité et de choisir un couple moteur minimum de 10 N.mm (ce qui reste quand même une valeur facilement atteignable sur des moteurs de petites tailles).

Le moteur choisi est un **N20 12V** avec un réducteur 1:30. Il possède un couple de 0,40 kg.cm (3,20 kg.cm bloqué), soit environ 39,2 N.mm (resp. 310 N.mm) à 600 rpm (500 en charge) pour une intensité d'environ 100mA. Ces caractéristiques et ses faibles dimensions (cf. Figure 1) sont idéales pour ce projet.

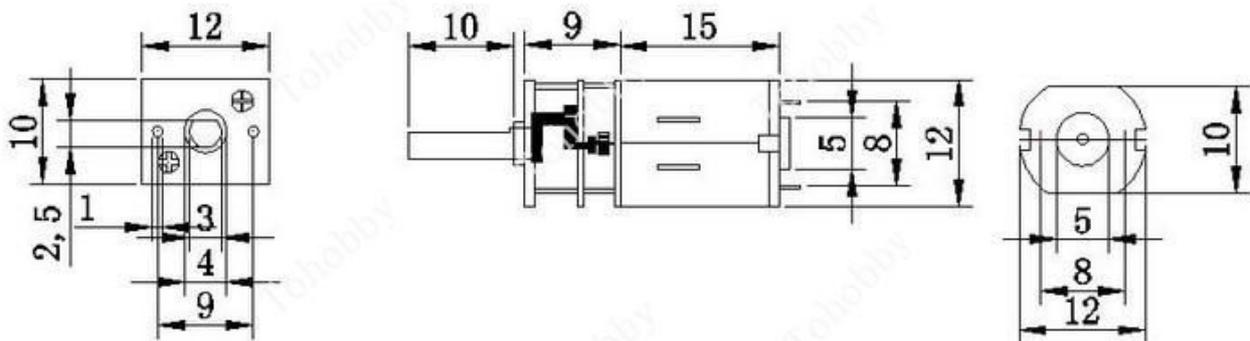


Figure 1: Schéma technique moteur N20 (côtes en mm)

3.1.2 Fabrication du support

Le support a été modélisé sous Solidworks et consiste de trois parties disponibles en annexe 2:

- Le support inférieur
- Le support médian
- Le support supérieur

L'axe est maintenu au centre des supports médian et supérieur par les roulements à bille. Le collecteur tournant est fixé au support inférieur. Les deux supports inférieurs sont maintenus ensemble par des vis, les deux supports supérieurs sont maintenus ensemble par le cylindre de Plexiglass.

Ces trois pièces ont ensuite été imprimées à l'imprimante 3D en PLA (cf glossaire).

3.1.3 Transmission, support capteur et compte-tour

La transmission est assurée par deux engrenages (module 1, 22 dents) fixés respectivement sur l'axe et sur le moteur.

Juste au-dessus sur l'axe, le capteur est maintenu en place horizontalement par une pièce modélisée sur Solidworks et imprimée en PLA (cf annexe 2).

L'interrupteur optique du compte-tour est fixé sur le support médian, juste en-dessous de l'engrenage. La languette de plastique servant d'interruption est collée au dessous de l'engrenage.

3.1.4 Modélisation Solidworks

Il a été nécessaire de faire une modélisation du système sous Solidworks pour vérifier que notre système puisse effectivement être contenu dans un cube de 80 mm de côté.

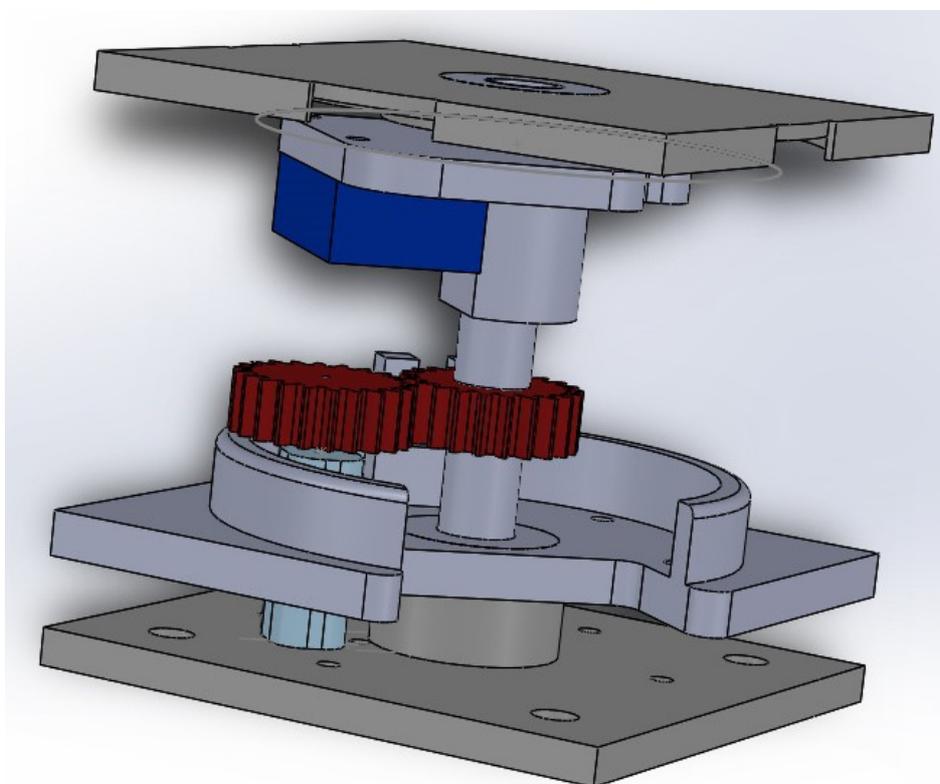


Illustration 1: Modélisation Solidworks du système

3.2 PARTIE INFORMATIQUE

3.2.1 Protocole

Le microcontrôleur du robot, un **STM32-H107**, est codé en C++, tandis que nous utilisons, pour des raisons de limitation spatiale et de matériel disponible, un Arduino mini pour gérer le capteur de la tourelle. L'Arduino récupère le signal en sortie du capteur tournant, ainsi que celui en sortie du compte-tour. Il utilise ces données pour conclure à quelle distance et à quel angle sont les objets qui ont été détectés par le capteur.

Il est important de considérer la taille des données à envoyer pour définir un protocole de communication, étant donné que le mode de communication Série UART que nous avons sélectionné pour la présence de bibliothèques adaptées sur les deux supports permet d'envoyer des données sous forme d'octets, qui sont aisément transcrits en entiers compris entre 0 et 255.

Nous devons transmettre la distance et l'angle. Soit une distance en centimètres qui du fait de la distance maximale d'observation ne dépasserait pas 255, et peut donc être codée sur un octet, et un angle en degrés dont la précision au degré près ne nous permet pas de le coder sur moins de 2 octets. Nous avons donc retranscrit l'angle en deux octets : le dividende et le reste de sa division euclidienne par 256.

Pour assurer la transmission correcte des informations, nous avons décidé d'ajouter un drapeau de début de message et un drapeau de fin de message, des octets valant respectivement 250 et 255.

De plus, afin de savoir quelles sont les objets détectés lors du dernier tour et de les différencier des objets repérés plus tôt, nous avons ajouté un message envoyé par l'Arduino à chaque fin de tour. Il prend la forme d'un octet à 254, qui permet au robot de savoir qu'un tour de tourelle s'est fini.

3.2.2 Arduino

La carte Arduino réalise à partir des signaux qu'elle reçoit du compte-tour et du détecteur de balise, le calcul de l'angle et de distance des objets détectés.

Les différentes entrées et sorties ainsi que les variables globales sont instanciées dans la partie setup. La boucle principale ne contient que la définition de la PWM (cf glossaire) associée à la vitesse.

Les fonctions destinées au calcul de l'angle et de la distance sont appelées grâce à des interruptions associées au changement d'état (RISING pour un front montant, ou FALLING pour un front descendant) sur les pin du compte-tour et du capteur. Deux pins sont utilisées pour l'entrée du capteur de manière à pouvoir déclencher les deux types d'interruptions.

La vitesse de rotation est calculée à chaque passage à '1' de l'entrée associée au compte-tour, comme étant inversement proportionnelle à la durée séparant deux passages à '1'. Dans la suite *timeSpeed* est l'heure¹ de la dernière interruption du compte-tour et *deltaSpeed* la durée séparant les deux dernière interruptions.

Par défaut l'entrée associée au capteur est à '0'. Lorsque le capteur croise un objet réfléchissant, le signal passe à '1' puis retourne à '0', ce qui fournit deux informations.

L'heure de chacun des deux changements est notée respectivement *timeTop* et *timeStop* pour la montée et la descente.

Les angles respectifs *angleTop* et *angleStop* sont alors calculés (en radian) de la façon suivante :

$$angleTop = \frac{timeTop - timeSpeed}{deltaSpeed} * 2\pi$$

équation 1

L'angle (en °) transmis sur la liaison série est alors donné à partir de la moyenne des deux heures :

$$angle = \frac{\frac{timeTop + timeStop}{2} - timeSpeed}{deltaSpeed} * 360$$

équation 2

La distance (en cm) est ensuite donnée par la formule suivante :

$$dist = \frac{r}{|angleStop - angleTop|}$$

équation 3

¹ On appelle ici « heure » la durée en millisecondes écoulée depuis le démarrage du programme.

La liaison série de la Arduino utilise un UART. Le débit utilisé pour la liaison avec le STM sera de 9600 baud/s. Les données seront envoyées octets par octets. La distance est comprise entre 0 et 200 cm donc naturellement codable sur un octet. Cependant l'angle est lui compris entre 0 et 360°, il nécessite donc 2 octets distincts. Le premier représente les bits de point fort (angle/256) et le second les bits de poids faible (angle modulo 256).

À chaque interruption descendante sur le capteur, une trame de données est émise sur la liaison série à destination du STM. Elle contient 5 octets, dont un drapeau de début (250) et un drapeau de fin (255) suivant le format de trame suivant :

250	anglePFort	anglePFaible	distance	255
-----	------------	--------------	----------	-----

De plus à chaque événement sur le compte tour, un octet dit de « reset » à 254 est envoyé sur la liaison série. L'utilité de cet octet de « reset » est détaillée dans la partie programmation du STM.

3.2.3 Robot

Afin que les informations produites par la tourelle puissent être utilisées par le robot, il fallait réaliser une interface informatique à insérer dans le code de celui-ci.

Nous avons donc créé une classe Tourelle. La stratégie du robot pourra accéder à la dernière liste d'objets détectés, stockée à l'intérieur de cette classe.

La classe contient une méthode *update*, qui devra être appelée par le robot à chaque fois qu'il souhaitera vérifier que des informations sont parvenues de la tourelle. Celle-ci récupère toutes les éventuelles données présentes dans le tampon du port série, et ajoute la position d'un objet détecté à une liste temporaire si elle reçoit bien les 5 octets du message. La liste des positions d'objets détectés ainsi construite est validée intégralement, et devient la liste accessible par le robot, à chaque fois que la classe lit dans le tampon la suite d'octets correspondant dits de « reset », exprimant la fin d'un tour de tourelle.

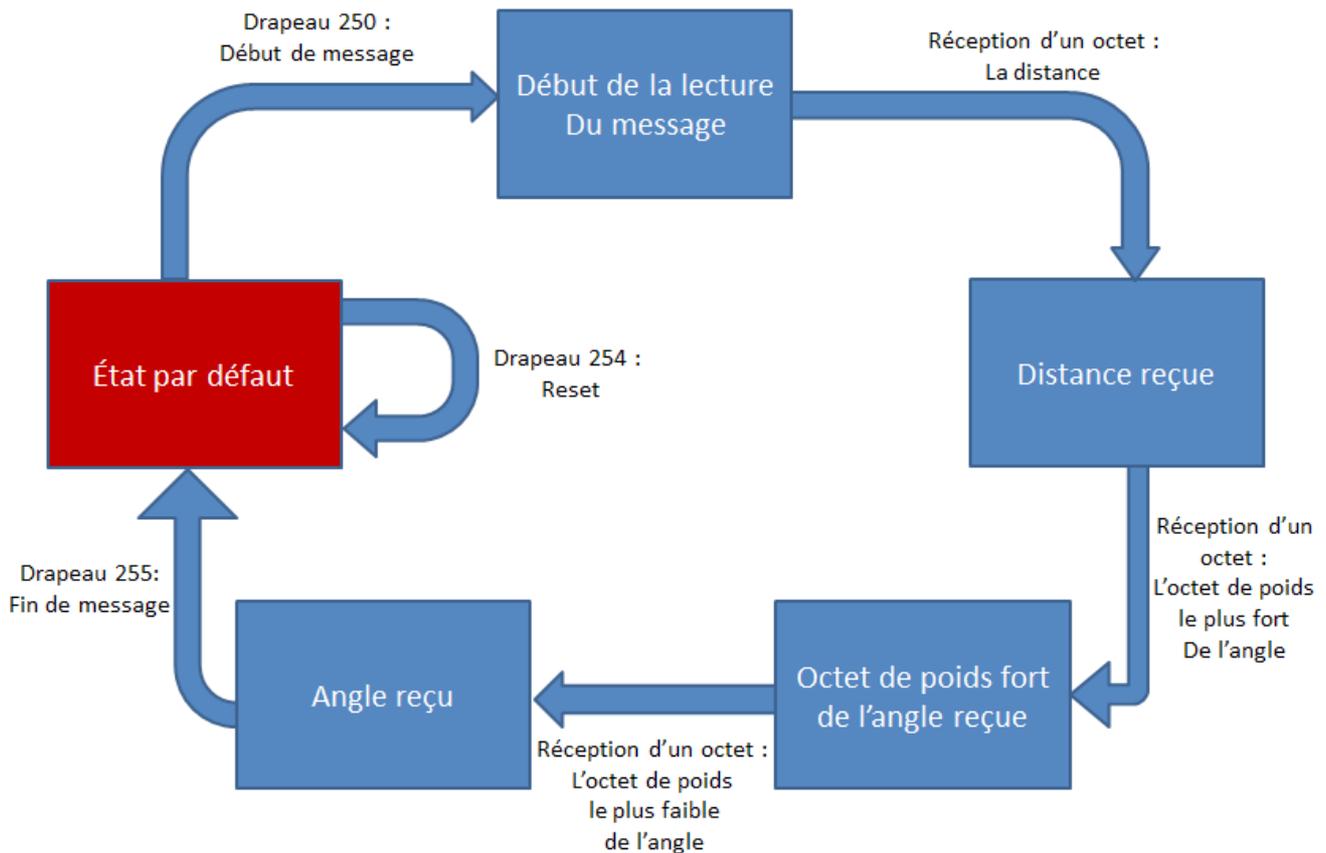


Figure 2: Schéma du protocole en réception

Le nombre d'objets à détecter est variable, car il peut y avoir un ou deux robots dans l'équipe adverse, et que ceux-ci peuvent disposer d'un système similaire et donc placer une balise sur notre deuxième robot. Ainsi, pour assurer que le système enregistre bien tous les objets détectés sans problème, nous avons codé une classe qui permet de gérer une liste dynamique de doublets, correspondant à la liste des positions, identifiées par l'angle et la distance.

La classe Tourelle, lors de l'appel de sa méthode créatrice, initialise tout ce qui est nécessaire à l'usage du port série. Cette initialisation varie selon le port utilisé, et elle est actuellement configurée pour se brancher sur un port utilisé par un autre module ajouté au robot : le module Bluetooth le reliant à une télécommande. Les deux modules ne seront pas utiles en même temps, car la tourelle est utile lorsque le robot se déplace librement, et non pas si il est télécommandé. Cependant, un autre emplacement serait préférable afin de ne pas avoir à refaire les branchements à chaque fois que l'on veut utiliser l'un ou l'autre. Cela n'était pas possible cette année car la carte électrique portant le microcontrôleur n'était pas adaptée pour cet usage.

Certains autres emplacements ont été prévus dans le code, mais sans moyen de s'assurer de leur fonctionnement pour le moment, les tests de ces configurations devront attendre la prochaine carte, l'année prochaine.

3.3 PARTIE ÉLECTRONIQUE

La partie électronique constitue l'interface entre les éléments matériels présents sur le robot et le code informatique exécuté par le processeur principal du robot. Le matériel nécessitant un interfaçage électronique est constitué d'un moteur, d'un capteur compte-tour et d'un détecteur de balise.

L'interfaçage de l'ensemble de ces éléments est réalisé par une carte programmable de type Arduino qui communique avec le microcontrôleur du robot (appelé « STM » dans la suite) via une liaison série.

3.3.1 Contrôle du moteur

Le moteur nécessite une alimentation nominale de 12V. Pour contrôler la vitesse de rotation du moteur, l'alimentation de celui-ci sera modulé par un signal de contrôle de type PWM (voir glossaire) généré par la carte Arduino et pilotant un transistor (voir montage en annexe).

La diode placée au borne du moteur est une diode dite de « roue libre », destinée à éviter que l'énergie emmagasinée par le moteur, à cause de son comportement inductif, ne traverse le transistor lorsque celui-ci vient de se bloquer.

3.3.2 Compte tour

Le compte-tour est un capteur composé d'une diode infrarouge en entrée et d'un phototransistor en sortie. Le fonctionnement est le suivant : lorsqu'un objet opaque est interposé entre la diode et le phototransistor, le transistor est bloqué et lorsque le champ est libre, le transistor est passant.

Nous utilisons ici un interrupteur optique **HY870P**^[3].

La diode infrarouge nécessite une tension d'au moins 1.6V pour une intensité de 10mA. En utilisant une tension d'alimentation de 5V, il faut donc placer une résistance en série avec l'alimentation de 330Ω (1/4W), ce qui est visible sur la figure 3 partie compte tour.

Le phototransistor est passant avec un courant d'environ 1 mA pour une tension VCE(sat) de 0.4V, il faut donc placer en série une résistance de pull-up de 4.6kΩ (1/4W).

3.3.3 Détecteur de balise

Le détecteur de balise est un capteur qui émet un rayonnement et qui détecte la réflexion de ce rayon sur un matériaux réfléchissant adapté. Dans l'ensemble du document, ce module de détection est plus simplement appelé « capteur ».

Le capteur nécessite une alimentation de 12V.

La sortie du capteur fournit une tension logique de 0V pour l'absence de réflexion et 12V pour la présence d'un élément réfléchissant. Les entrées logiques de la carte Arduino supportant les tensions logiques 0V et 5V, il est nécessaire de réaliser une division de tension.

Comme on peut le voir sur la figure 3 partie capteur, on utilise pour cela un pont diviseur avec une résistance de 1kΩ (1/4W) et une autre de 1k5Ω (1/4W), ce qui donne une tension résultante d'environ 4.8V qui sera bien détectée comme '1' logique par la carte Arduino.

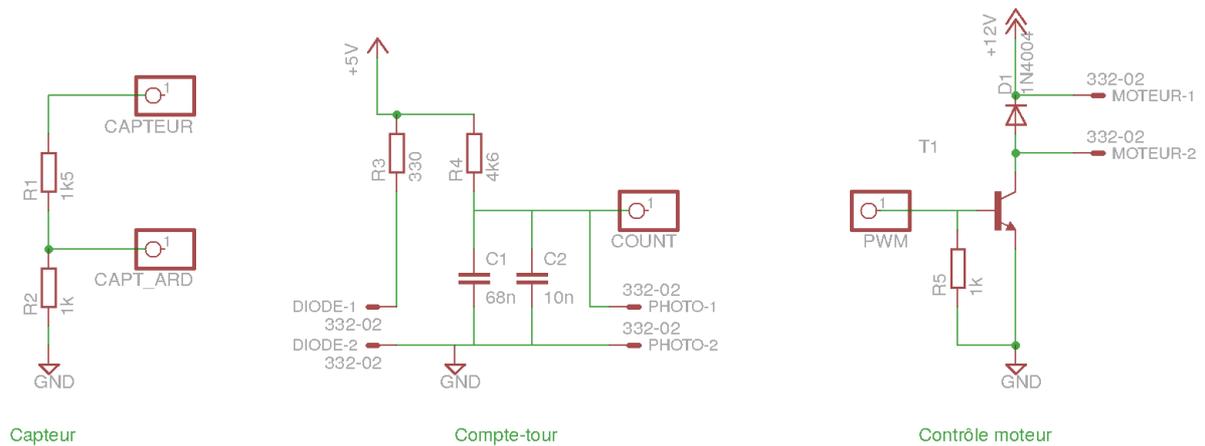


Figure 3: Schéma de l'électronique embarquée

3.3.4 Carte Arduino

La carte Arduino reçoit le signal logique du compte-tour (passage à '1' logique puis retour à '0' une fois par tour) et celui du capteur qui passe à '1' à chaque fois qu'il détecte la balise. Les calculs effectués par la carte Arduino sont détaillés dans la partie Informatique du module.

La carte Arduino réalise ensuite une connexion série avec le STM, ce standard de connexion nécessite trois fils, 2 fils de données (dénommés RX et TX) ainsi qu'un fil de masse (GND). À noter que la transmission de données du STM vers la carte Arduino n'est pas prévu dans la première version du projet.

3.3.5 Prototypage

La plateforme de test (doublant avec une preuve de concept) a été réalisée avec un **Arduino UNO** et une plaquette d'essai sur laquelle l'ensemble des composants ont été soudés.

L'ensemble de la partie électronique est dès à présent fonctionnelle mais la liaison série avec le STM n'a pas été testé.

Pour les prototypes et la version finale, un circuit imprimé conçu pour intégrer une Arduino mini est en cours de réalisation.

3.3.6 Résumé des interfaces avec le reste du robot

Alimentation : +12V, +5V, GND

Connexion série avec le STM : RX, TX, GND

Il y a donc au total 6 fils de connexion avec la partie principale du robot, la masse alimentation et la masse numérique se voulant au maximum séparée.

4. PRÉSENTATION DU PRODUIT FINI

4.1 PHASE DE TEST

Le prototype réalisé doit être soumis à une batterie de tests, afin de vérifier sa conformité avec le cahier des charges.

4.1.1 Programmation Processing

Le programme VisualPoint développé en Java avec l'interface Processing permet de lire les données émises par la carte Arduino sur la liaison série. Il a été développé dans le but de tester le comportement de la Arduino et du système indépendamment du STM. Cette phase de test est particulièrement importante car les test directs sur le STM ne sont pas aisés et nécessitent la disponibilité du robot, ce qui n'est pas possible lorsque l'équipe du projet robotique est à l'œuvre. Chaque minute de débogage ainsi gagné est cruciale.

L'interface de Visualpoint est des plus basiques mais permet d'afficher en temps réel les positions successives des objets détectés par le robot, et ainsi d'évaluer la précision donnée par le système. Le système est représenté par la croix au centre du cercle blanc qui lui-même représente la zone de détection du capteur avec un rayon de 200cm.

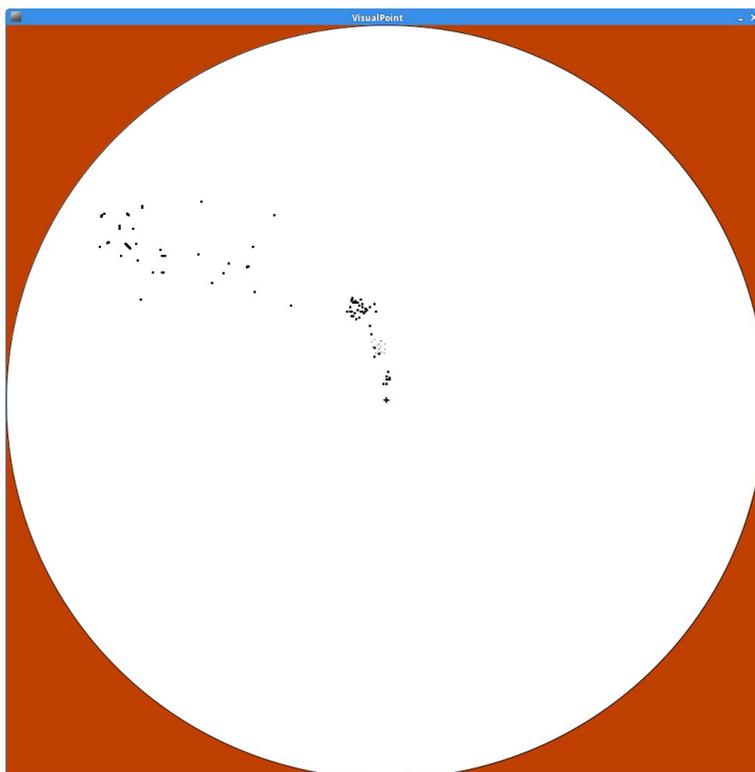


Figure 4: La fenêtre de VisualPoint permet d'observer la bonne précision du système lorsque l'objet est proche, mais qui se dégrade fortement lorsque l'objet s'éloigne à plus de 1m50.

4.1.2 Fonctionnalité principale

Il est nécessaire de vérifier que le système est bien capable de détecter au moins trois balises entre 0,5m et 1,5 m. Pour cela, le test de validation consiste à placer une, puis deux, puis trois balises entre 0,5 m et 1,5 m et de vérifier que le système les détecte bien puis transmet les données utiles à l'interface.

4.1.3 Contraintes réglementaires

Une des exigences principales sur ce produit est le respect du règlement de la coupe Eurobot^[1]. Elles se manifestent sur trois points :

- La tourelle est contenue dans un cube de 80x80x80mm, avec une tolérance plus importante pour la face inférieure
- La face supérieure est recouverte de Velcro
- L'ensemble de la tourelle peut supporter une masse supérieure à 400g sur sa face supérieure

Les tests requis pour la validation sont simples :

- Mesurer directement la tourelle avec un pied à coulisse
- Vérifier la présence du Velcro
- Poser une masse de 500g sur la tourelle et vérifier son bon fonctionnement

4.1.4 Distances limites

Il faut vérifier que les atteintes minimum concernant les distances de détection définies dans le cahier des charges soient bien réalisées. Pour cela, il faut placer une balise à 0,5 m du capteur, et vérifier que celui-ci la détecte bien, puis recommencer la procédure à des intervalles de distance réguliers pour trouver la portée maximale du système.

4.1.5 Précision angulaire

Il faut vérifier que la précision angulaire atteigne au minimum 10°. De plus, il faut vérifier le comportement du système lorsque le capteur et deux balises sont alignés : le système devrait voir une seule balise à une distance plus proche que la distance réelle, mais il est nécessaire de vérifier que ce comportement ne présente pas de gêne majeure.

4.1.6 Résistance au bruit

Étant donné que lors de la Coupe de France de robotique, l'éclairement de la table est très inégal, il est nécessaire d'étudier le comportement du système dans diverses conditions de luminosité. Il faut donc tester le système sous fort éclairage, ainsi que sans aucune source de lumière parasite. Il faut que notre système soit capable de continuer à détecter une balise même avec une ou plusieurs sources de lumière extérieure.

4.1.7 Puissance requise

Il faut vérifier que le système ne demande pas plus de 5 W en utilisation, vérifiable en le branchant sur deux alimentations adaptées de 12 V et 5 V (identiques à celle utilisées par le robot).

4.2 PERSPECTIVES

Ce système pourra être utilisé sur le robot participant à la coupe de la robotique 2016, soit en remplaçant les capteurs de contact évitant les collisions, soit en complément de ceux-ci. Il est cependant nécessaire que l'équipe s'occupant de la fabrication du robot de 2016 intègre la tourelle dans leur stratégie de gestion du robot.

5. CONCLUSION

Dans ce rapport, nous avons vu les différents aspects que nous avons dû considérer pour la conception du système de détection de robot à moyenne distance. Il a fallu réfléchir au type de capteur à utiliser pour satisfaire aux contraintes impératives de la coupe de la robotique, puis l'intégrer dans un espace réduit et le connecter avec le robot principal. Ce système pourra donc être intégré sur le robot présenté par Télécom Bretagne lors de la coupe 2016 pour améliorer ses performances et lui permettra d'envisager des nouvelles stratégies pour se déplacer sur la table de jeu. Il est cependant regrettable que le système final n'ait pas pu être testé en totalité avant la rédaction de ce rapport. Une partie du groupe sera disponible l'an prochain pour aider le projet robotique à intégrer la tourelle sur le robot principal.

A plus long terme, ce module peut servir de base à un système plus élaboré, qui permettrait d'avoir une mesure plus précise, ou pourrait apporter d'autres améliorations au robot.

6. RÉFÉRENCES BIBLIOGRAPHIQUES

6.1 RÉGLEMENT DE LA COUPE DE FRANCE DE LA ROBOTIQUE 2015

[1] http://www.planete-sciences.org/robot/data/file/coupe/2015/C2015_Reglement_FR_final.pdf
(03/06/15)

6.2 DATASHEETS

[2] Capteur optique **SICK WL100**, site fabricant (en)
<http://www.audin.fr/pdf/documentations/sick/detection/capteurs-photoelectriques/W100/W100.pdf> (11/05/2015)

[3] Interrupteur optique **HY870P**, site fabricant (en)
<http://hyzt.com/manager/upimg/hy870P.pdf> (11/05/2015)

[4] Caractéristiques Plexiglass (en)
<http://www.plexiglas.com/export/sites/plexiglas/content/medias/downloads/sheet-docs/plexiglas-optical-and-transmission-characteristics.pdf> (07/06/2015)

6.3 DOCUMENTATIONS

[5] Références **Arduino (Uno et Mini)**, site fabricant (en)
<http://www.arduino.cc> (11/05/2015)

- Bjarne Stroustrup, *The C++ Programming Language, 4th Edition*, 2013
- Recherches partie électronique :
http://en.wikipedia.org/wiki/Pulse-width_modulation (11/05/15)
http://fr.wikipedia.org/wiki/Modulation_de_largeur_d%27impulsion (11/05/15)
<http://en.wikipedia.org/wiki/Rx> (11/05/15)

7. GLOSSAIRE

- GND : Ground (ou masse en français) : équipotentielle 0 V.
- PLA : acide polylactique, polymère utilisé par de nombreuses imprimantes 3D dont celle du FabLab
- PWM : Pulse-width modulation, en français : Modulation de largeur d'impulsions
- RX : Abréviations Télégraphiques pour « recevoir », « récepteur » ou « réception ». Concerne ici le port sur lequel la Arduino va recevoir des données de type série.
- Sonar : appareil de détection utilisant des ondes sonores
- TX : Abréviations Télégraphiques pour « transmettre », « transmetteur » ou « transmission ». Concerne ici le port sur lequel la Arduino va émettre des données de type série.
- UART : Universal Asynchronous Receiver Transmitter, est un émetteur-récepteur asynchrone universel. Il sert à faire la liaison entre le processeur qui fonctionne en parallèle et le port série.
- VisualPoint : notre programme java développé à l'aide de l'interface processing

Annexes

ANNEXE 1 - PLANIFICATION

1-A) PLANNING PRÉVISIONNEL

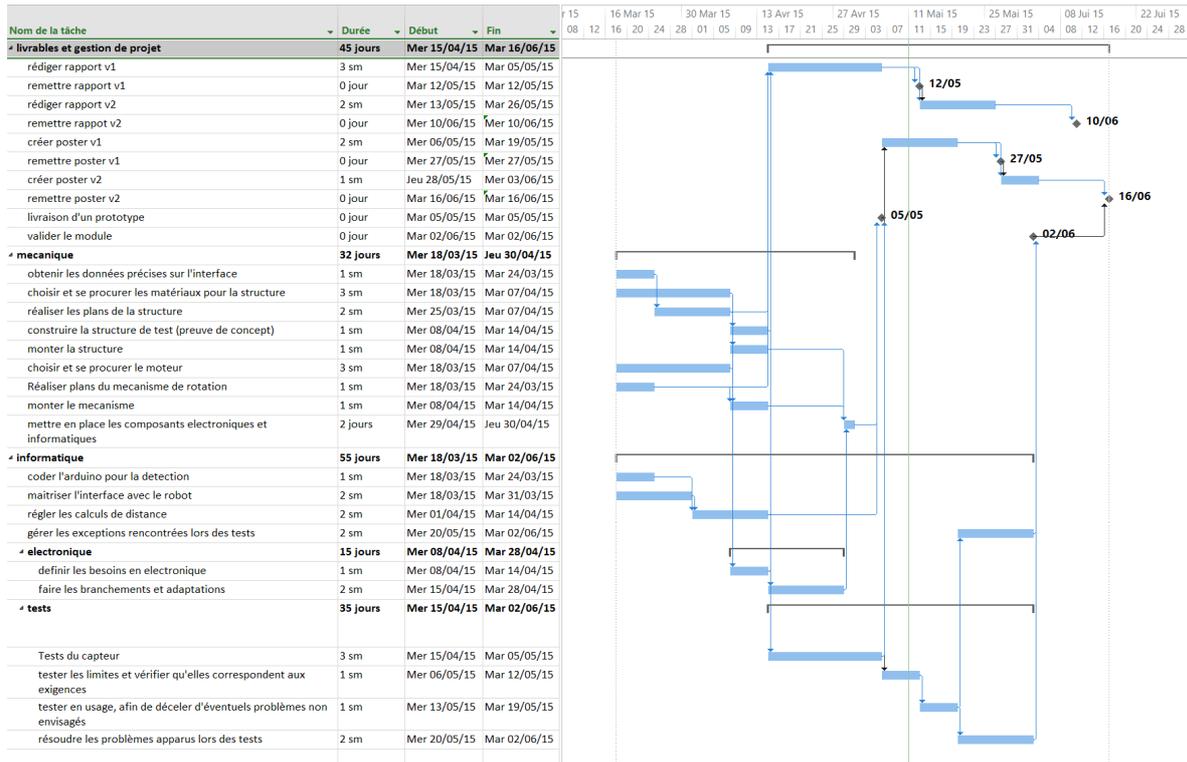


Illustration 2: Diagramme de Gantt du 16 mars

ANNEXE 2 - PLANS DES PIÈCES

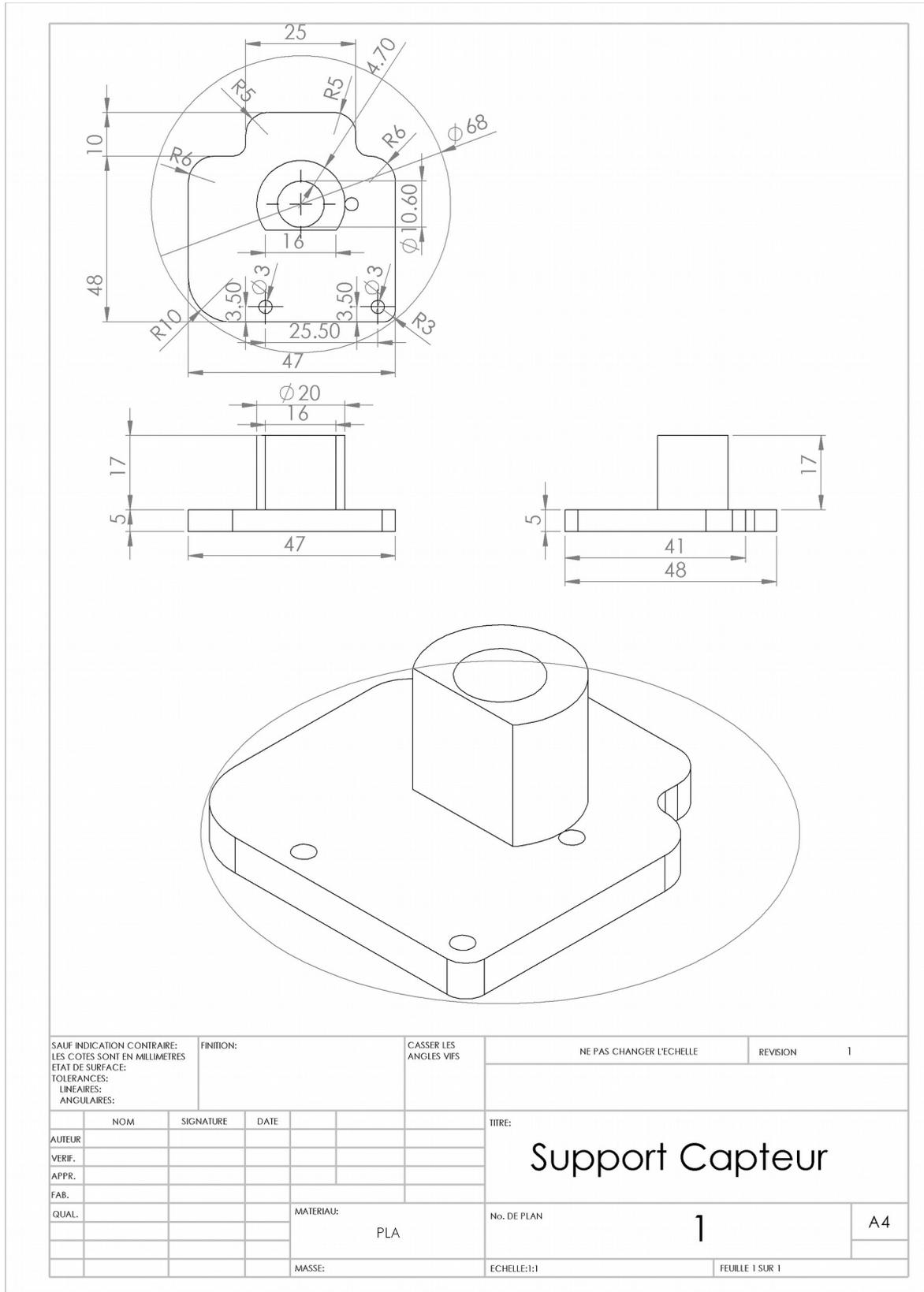
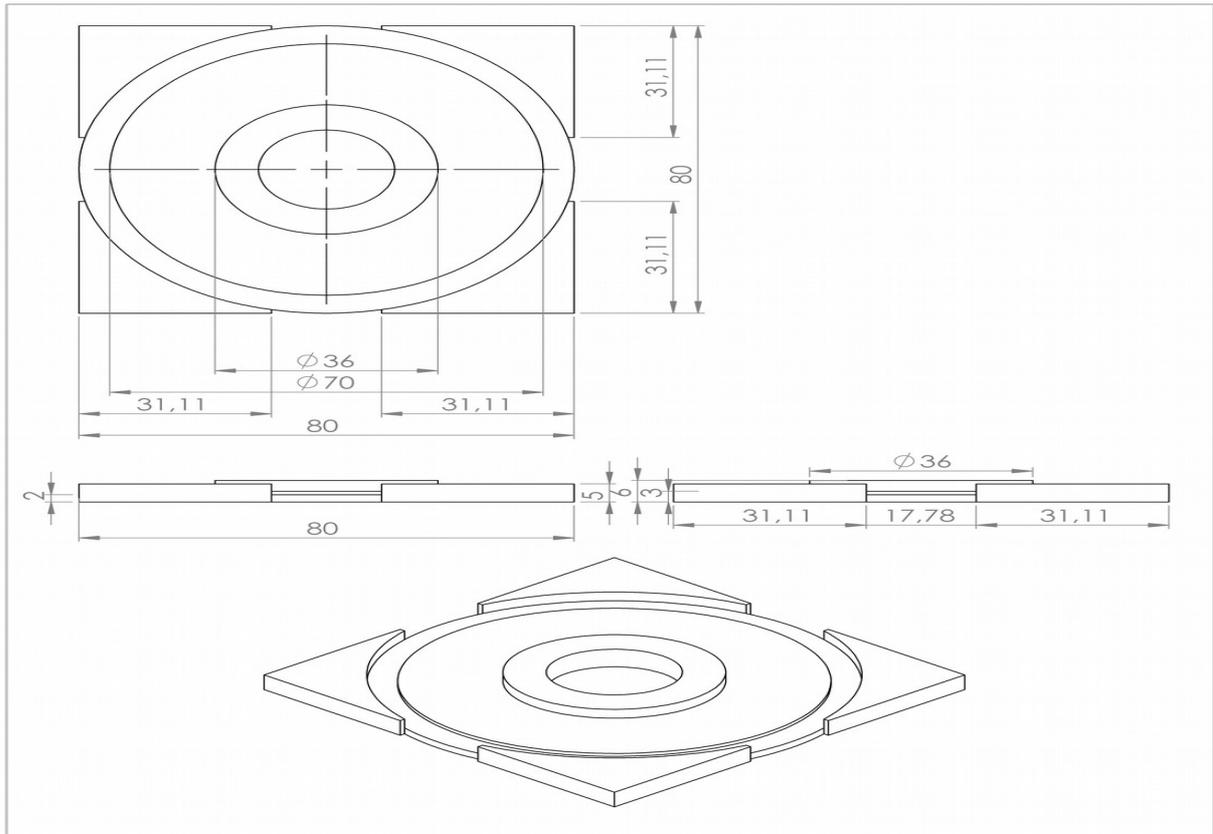
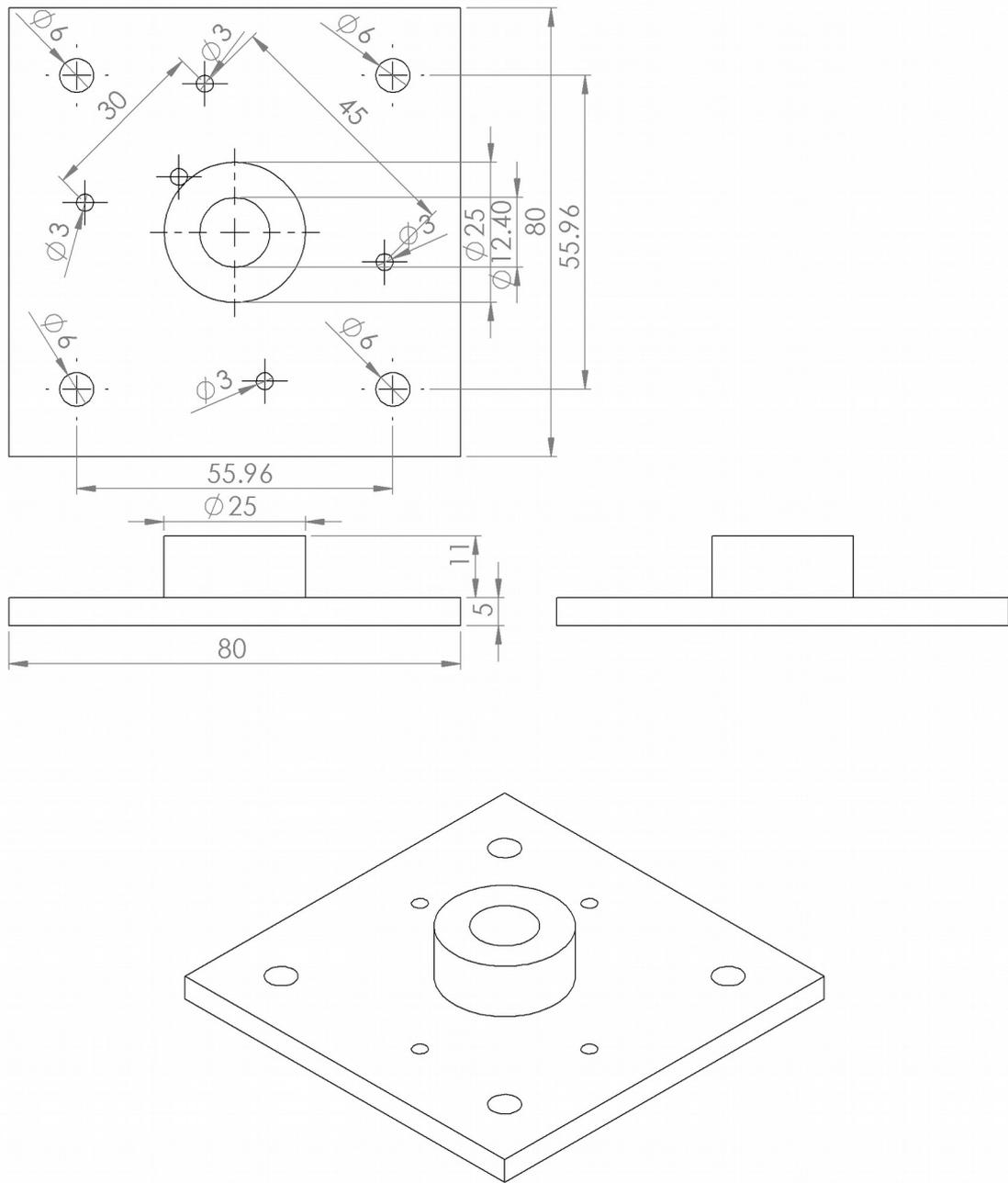


Figure 5: Schéma technique: support capteur

Figure 6: Schéma technique - Support supérieur



SAUF INDICATION CONTRAIRE: LES COTES SONT EN MILLIMETRES				FINITION:		CASSER LES ANGLES VIFS		NE PAS CHANGER L'ECHELLE		REVISION 1	
ETAT DE SURFACE:								TITRE: Support Supérieur			
TOLERANCES:								No. DE PLAN 2			
LINEAIRES:								A4			
ANGULAIRES:								Echelle: 1:1			
AUTEUR	NOM	SIGNATURE	DATE	MATERIAU: PLA		MASSE:		FEUILLE 1 SUR 1			
VERIF.											
APPR.											
FAB.											
QUAL.											



UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:	DEBUR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION	2
					TITLE:		
					Support inferieur		
DRAWN	NAME	SIGNATURE	DATE		DWG NO.		4
CHK'D					MATERIAL:		PLA
APP'VD					SCALE:1:1		A4
MFG					WEIGHT:		
Q.A					SHEET 1 OF 1		

Figure 7: Schéma technique - Support inferieur

ANNEXE 3 - CAHIER DES CHARGES

Table des matières

Table des matières.

I Introduction.

1/ Objet du présent document

2/ Considérations lexicales.

II Objectifs du projet

1/ Définition du produit

2/ Fonctionnalité principale.

3/ Contexte d'exploitation.

4/ Pérennité.

III Exigences sur le produit

1/ Capacités fonctionnelles.

1.1/ Fonctionnalité principale.

1.2/ Exigences techniques spatiales.

1.3/ Exigences techniques liées à l'environnement

1.4/ Sécurité.

1.5/ Rendement

1.6/ Maintenabilité et compatibilité.

2/ Exigences de développement

2.1/ Délais et coûts.

2.2/ Réalisation.

3/ Flexibilité des exigences.

3.1/ Catégories d'exigences.

3.2/ Classement des exigences.

IV Annexes.

I Introduction

1/ Objet du présent document

Le présent document cherche à établir le cahier des charges du projet n°66 (Balise du robot de la coupe robotique). Il établit :

- Les objectifs de ce projet
- Les contraintes techniques et logistiques de ce projet

Il ne vise cependant pas à proposer de solution technique mais sert de base à la conception et à la réalisation de prototypes.

2/ Considérations lexicales

Est appelé « notre robot » le robot sur lequel est placé le module du projet.

Sont appelés « robots adverses » les robots avec lesquels évoluera notre robot.

Le terme « balise » peut référer, selon le contexte et de manière abusive :

- Aux modules réflecteurs (passif) placés sur les robots adverses
- Au capteur (actif) placés sur notre robot

II Objectifs du projet

1/ Définition du produit

Notre produit est un module de localisation à moyenne distance de balises mobiles. Il existe sous la forme d'un module externe dépendant énergétiquement et ne prenant pas de consignes.

2/ Fonctionnalité principale

Ce module doit pouvoir détecter un petit nombre de balises mobiles et passives, et renvoyer leur position relative.

Les données chiffrées sont considérées dans la suite de ce document.

3/ Contexte d'exploitation

Ce module est conçu dans le contexte de la coupe robotique **Eurobot**. De nombreuses contraintes sont imposées par des considérations réglementaires (règles...) et pratiques (environnement...). Ces contraintes sont détaillées dans le règlement officiel de la coupe².

4/ Pérennité

Notre produit vise à être réutilisable pour les itérations successives du robot. Les interfaces doivent donc être flexibles et non permanentes (i.e. pas de colle, etc.)

Les contraintes de taille ne sont pas susceptibles de changer d'une année sur l'autre. Cependant, si le prototype réalisé n'est plus adapté aux contraintes réglementaires d'une coupe future, les ressources utilisées et les techniques développées doivent être accessibles au prochain groupe de travail sur le sujet.

III Exigences sur le produit

1/ Capacités fonctionnelles

1.1/ Fonctionnalité principale

Ce module doit répondre à une seule fonctionnalité : la détection à distance de ses réflecteurs.

1.2/ Exigences techniques spatiales

Les contraintes réglementaires de la coupe **Eurobot** imposent de nombreuses contraintes techniques :

- Le capteur doit tenir dans un bloc de 80x80x80mm
- Idem pour le réflecteur
- Le réflecteur est placé juste au-dessus du capteur du robot adverse (i.e. sur un plan horizontal, juste au-dessus du capteur de notre robot)
- La zone de la compétition n'est jamais plus grande que 3.6m (rectangle de 3x2m)
- La technologie utilisée ne doit pas être dangereuse pour les spectateurs au bord de la zone

²http://www.planete-sciences.org/robot/data/file/coupe/2015/E2015_Rules_EU_EN_final.pdf

Cependant, des contraintes géométriques (liées à la position du couple capteur/réflecteur et à la taille réduite du réflecteur) imposent des limites à la portée théorique d'un tel système³.

Les contraintes mises à jour seraient :

- De 0.4m à 2.5m pour la détection
- Précision angulaire de moins de 0.8° sur le réglage du capteur

1.3/ Exigences techniques liées à l'environnement

L'environnement dans lequel évolue le robot est saturé en sons (environnement bruyant...) et en infrarouges (flashes d'appareils photo...). Les mauvaises conditions lumineuses limitent l'utilisation de solutions à base de caméras. Les distances en jeu et les natures diverses des obstacles préviennent l'utilisation de capteurs optiques sans réflecteurs.

Le module ne sera jamais à portée de plus de 3 balises en même temps.

1.4/ Sécurité

Les technologies à base de laser, etc. ne doivent pas dépasser la classe I.

1.5/ Rendement

Le temps de rafraîchissement du capteur doit être suffisant à son utilisation dans le contexte de la coupe **Eurobot**. La fréquence de rafraîchissement

minimum serait 5Hz (cinq tours par seconde), soit 300rpm. Une fréquence supérieure serait souhaitable.

La précision angulaire est imposée par les contraintes de dimensionnement. Elle est d'environ 1°, soit ~4cm à 2.5m.

La puissance nécessaire au fonctionnement du capteur doit être inférieure à 5W.

1.6/ Maintenabilité et compatibilité

Les plans, les processus de fabrications et une API documentée pour son utilisation seront livrés avec le produit.

2/ Exigences de développement

2.1/ Délais et coûts

Ce capteur devra être entré en phase de prototypage avant le 31 Mai 2015.

Les phases de productions et de déploiement seront ignorées.

Le budget pour ce projet est non existant. Il s'agira d'utiliser au maximum les ressources du club robotique et du Fablab et de faire preuve d'inventivité pour le reste.

2.2/ Réalisation

L'API devra être développée en C++ pour un microcontrôleur **STM32** sous licence MIT.

Les plans et les processus de fabrication seront placés sous la même licence.

³ CF Figures en annexe

3/ Flexibilité des exigences

3.1/ Catégories d'exigences

Les exigences de ce projet peuvent se répartir en trois catégories :

- **Impératives** : ces exigences doivent être remplies à tout prix. Elles n'admettent en général pas de marge de manœuvre.
- **Majeures** : ces exigences doivent être remplies comme annoncées sur le présent document. Il existe une certaine marge de manœuvre.
- **Mineures** : ces exigences doivent être remplies plus dans l'esprit du texte que dans sa formulation. Elles peuvent être soumises à révision lors du développement, si de nouvelles contraintes apparaissent. La marge de manœuvre est large.

3.2/ Classement des exigences

3.2.1/ Exigences impératives

- Sécurité

Aucun laser de classe supérieure à **I** ne sera admis. Le produit ne devra en aucun cas être dangereux dans des circonstances d'utilisation normales.

- Contraintes de taille

Le produit devra être contenu dans un volume de 80x80x80mm, avec une marge de moins d'1mm. Il est possible de faire plus petit, mais certainement pas plus grand.

Note : il est possible de légèrement dépasser au travers de la face inférieure. Les autres faces n'admettent aucune tolérance.

- Interfaces

La face supérieure de la balise doit impérativement être recouverte de Velcro.

L'API devra impérativement être développée pour le microcontrôleur indiqué, dans le langage indiqué.

3.2.2/ Exigences majeures

- Délais

Sauf cas de force majeure, le développement d'un prototype devra être bien avancé à la date indiquée.

- Fonctionnalité principale et rendement

Il est entendu que le produit doit être capable de réaliser la fonction pour laquelle il a été conçu. Une marge de manœuvre existe cependant sur les grandeurs caractéristiques de cette fonction. Sont considérées comme *préférables* les valeurs données en **1.1** et **1.5**. Sont considérées comme *minimales* les valeurs suivantes :

- Détection minimale : 0.5m
- Détection maximale : 1.5m
- Précision angulaire : 10°
- Vitesse de rafraîchissement : 1Hz

3.2.3/ Exigences mineures

- Coûts

Une hypothétique source de financement additionnelle ne serait pas refusée.

- Licence

Le projet et ses ressources devraient être fournis sous une licence libre. Les détails peuvent être facilement

révisés.

- Documentation

La documentation n'est pas soumise aux contraintes de délais. Elle pourra être rajoutée à posteriori.

- Rendement

La puissance indiquée en **1.5** est purement informative. Si il serait absurde de consommer plus pour un capteur de ce type, en pratique l'alimentation du robot nous permet d'ignorer cette contrainte.

- Interfaces

La face inférieure du produit doit fournir une interface mécanique avec le robot. En pratique, il plus simple de mettre cette contrainte de côté et d'aviser lors de la réalisation du prototype.

IV Annexes

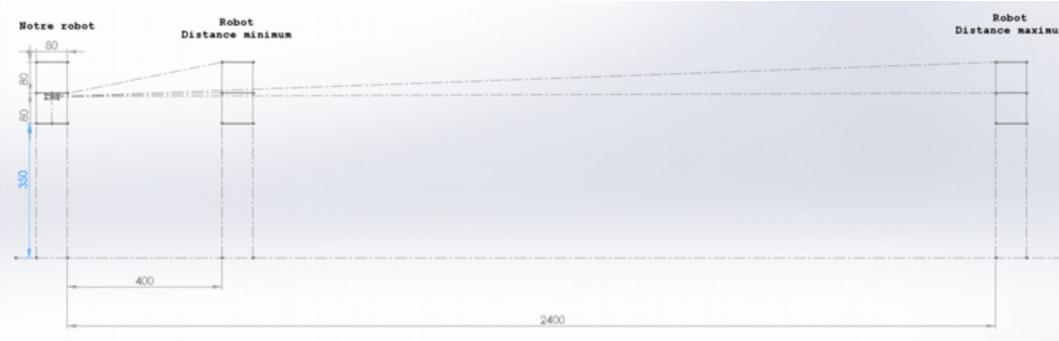


Figure 8

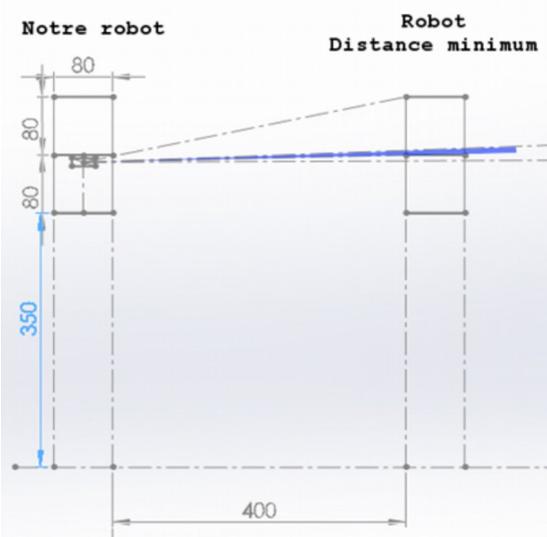


Figure 9

w w w . t e l e c o m - b r e t a g n e . e u

Campus de Brest
Technopôle Brest-Iroise
CS 83818
29238 Brest Cedex 3
France
Tél. : + 33 (0)2 29 00 11 11
Fax : + 33 (0)2 29 00 10 00

Campus de Rennes
2, rue de la Châtaigneraie
CS 17607
35576 Cesson Sévigné Cedex
France
Tél. : + 33 (0)2 99 12 70 00
Fax : + 33 (0)2 99 12 70 19

Campus de Toulouse
10, avenue Edouard Belin
BP 44004
31028 Toulouse Cedex 04
France
Tél. : +33 (0)5 61 33 83 65
Fax : +33 (0)5 61 33 83 75

TELECOM
Bretagne

