

D.R.U.M.S

Le Robot Suiveur de Ligne n°2

Enseignant responsable
Fabrice DELAMARE

Étudiants :

Etienne BATISE

Lucas ERARD

Guillaume MINETTE

DE SAINT MARTIN

Jean-Claude

BERNARD

Romain LE CORRE

Nathan MALO

Date de remise du rapport : 17/06/2013

Référence du projet : STPI¹/P6/2013 – 4

Intitulé du projet : Robot Suiveur de Ligne n°2

Type de projet : *experimental*

Objectifs du projet : (10 lignes maximum)

Ce projet a pour but de nous faire construire un robot capable de suivre une ligne en mettant en oeuvre des techniques liées à la programmation, à la mécanique et à l'électronique. Au delà de cet objectif, le but est aussi d'apprendre à travailler en groupe dans des domaines fondamentalement différents. Ainsi nous avons donc dû faire preuve d'autonomie et apprendre par nous même avec peu d'aide extérieure.

1. INSTITUT NATIONAL DES SCIENCES APPLIQUÉES DE ROUEN
DÉPARTEMENT SCIENCES ET TECHNIQUES POUR L'INGÉNIEUR
685 AVENUE DE L'UNIVERSITÉ BP 08- 76801 SAINT-ETIENNE-DU-ROUVRAY
TÉL : 33 2 32 95 66 21 - FAX : 33 2 32 95 66 31

Table des matières

Notations	5
Introduction	6
1 Méthodologie, organisation du travail	7
1.1 Méthodologie	7
1.2 Organisation et répartition du travail	8
2 Travail réalisé et résultats	9
2.1 Partie Mécanique	9
2.1.1 Explication du pôle	9
2.1.2 Approche de l'objectif	9
2.1.3 Conception du robot	9
2.1.4 Alimentation	10
2.1.5 Difficultés rencontrées	10
2.2 Partie Electronique	11
2.2.1 Explication du pôle	11
2.2.2 Méthode de travail	11
2.2.3 Approche de l'objectif	11
2.2.4 Conception de la carte électronique	11
2.3 Partie Programmation	13
2.3.1 Explication du pôle	13
2.3.2 Approche de l'objectif	13
2.3.3 Cahier des charges	14
2.3.4 Conception du code	14
Conclusion et perspectives	16
2.4 Conclusion sur le travail réalisé	16
2.5 Conclusions sur l'apport personnel de cette E.C. projet	16
2.6 Perspectives pour la poursuite de ce projet	18
Bibliographie	19
A Documentation technique	20
B Listings des programmes réalisés	22
C Schémas de montages, plans de conception	25
C.1 Partie électronique	25
C.2 Partie Mécanique	26

Notations et Acronymes

D.R.U.M.S : Signifie Digitally Robotized Unit Manually Started.

Introduction

Le semestre 4 de cycle préparatoire de l'INSA de Rouen est la dernière étape avant le cycle ingénieur. Pour nous former à ce cycle, de nombreux cours de pré-spécialisation nous sont enseignés. Néanmoins, pour correspondre au mieux avec l'esprit ingénieur, il nous est demandé de réaliser un projet complet de physique. Contrairement à d'autres projets imposés durant le cycle préparatoire, nous allons devoir partir de zéro et arriver à un résultat concret : un objet fini. Ce projet s'inscrit parfaitement dans notre formation car c'est aussi la première fois que nous travaillons dans des groupes aussi grand et dans des domaines très différents. Le défi majeur imposé par cette matière, outre la gestion et le travail de groupe, sera principalement l'auto-formation dans ces domaines.

Le groupe est composé de six étudiants qui auront pour but de réaliser un robot à l'aide de la technologie Arduino. L'unique objectif à atteindre sera de le rendre capable de suivre un ligne. Ce projet nous permettra donc d'utiliser et d'enrichir nos connaissances en programmation et en électronique.

Le bilan de notre projet est rédigé dans ce rapport. Il explique notre méthodologie ainsi que les problèmes rencontrés tout au long du semestre.

Chapitre 1

Méthodologie, organisation du travail

1.1 Méthodologie

Les pôles Mécanique, Electronique et Programmation étaient à distribuer. Nous avons eu la chance dans notre groupe d'être composé d'étudiants aux compétences multiples. En effet, certains d'entre nous faisaient la thématique ASI-GM, ASI-Meca, Meca-EP. Ainsi, il nous a paru évident de répartir les gens dans les catégories correspondants à leurs compétences dans un premier temps, puis en fonction en de leurs envies. En effet, en répartissant 2 personnes par catégorie, on se retrouvait en surnombre, spécialement au niveau de la programmation, ayant dans notre groupe 4 ASI-GM. Le choix s'est donc fait au niveau de l'expérience, car dans ce domaine, la pratique prône sur la théorie. Ainsi dans la partie Mécanique, se trouvent Guillaume Minette de Saint Martin et Romain Le Corre, en effet ceux-ci font respectivement la thématique ASI-Meca et EP-Meca. Ensuite concernant la programmation, le choix a été rude, du fait du nombre de prétendants. Le choix a été donc fait comme dit précédemment, sur la base de l'expérience. En effet Etienne et Nathan avaient déjà programmé en quantité par le passé, en dépit du bon niveau de Lucas et de Jean-Claude. Ces deux derniers présentaient des capacités en électronique, et ont tout de même dû rafraichir leurs connaissances avant de commencer. Ainsi on a donc Guillaume et Romain en mécaniciens, Jean-Claude et Lucas et électroniciens et Etienne et Nathan en développeurs.

Afin d'être le plus clair dans notre travail, nous nous sommes fixés des règles, comme la rédaction de minis rapports à chaque séance. En outre, afin d'adopter une démarche de clareté, nous nous sommes créés un espace de travail collaboratif en ligne, sur Dropbox, afin que chacun puisse disposer des fichiers des différents pôles sans avoir à quémander sans cesse. Enfin, un groupe confidentiel à été créé sur Facebook, afin d'assurer une communication efficace au sein de notre groupe.

1.2 Organisation et répartition du travail

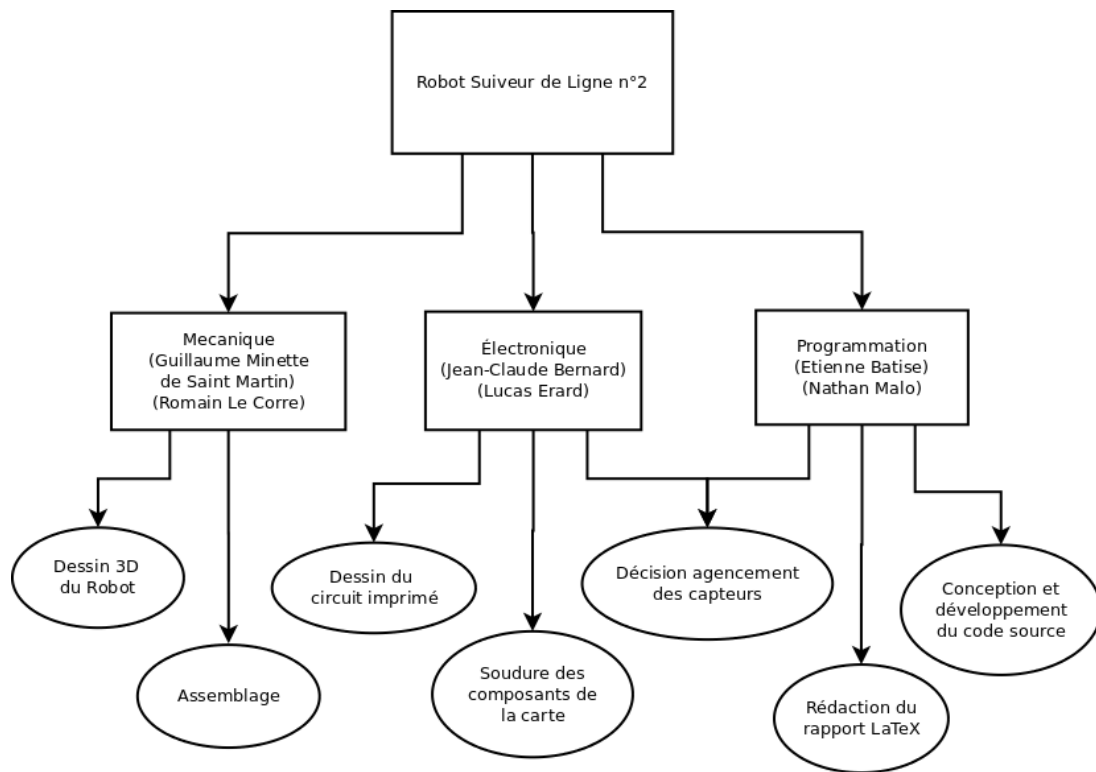


FIGURE 1.1 – Organigramme des tâches réalisées et des étudiants concernés

Chapitre 2

Travail réalisé et résultats

2.1 Partie Mécanique

2.1.1 Explication du pôle

Cette partie du projet a pour but d'élaborer la partie mécanique du robot, c'est à dire dans un premier temps de réfléchir à la forme, le fonctionnement global de la machine, ensuite créer un modèle numérique complet en 3 dimensions et enfin d'adapter et monter ensemble les différents composants du robot après les avoir usinés.

2.1.2 Approche de l'objectif

Formation

Pour réaliser ce robot, il a fallu créer un modèle 3D de chaque pièce et ensuite les assembler pour former la machine, pour cela nous avons utilisé le logiciel Solid Works. Nous avons alors dû apprendre à l'utiliser, pour cela Romain suivant les cours de CTI3 savait se servir de cet outil, il n'a donc pas eu besoin de formation supplémentaire. Guillaume ne suivant pas ce cours s'est servi d'un tutoriel trouvé sur internet pour acquérir les bases de l'utilisation de ce logiciel. Nous avons par la suite entrepris la création des modèles 3D de tous les composants du robot en respectant toutes les cotes et les forme. Voir annexe [C.5](#)

Cahier des charges

En premier lieu, il convient de définir le fonctionnement global du robot, c'est à dire si il sera tracté ou propulsé (roues motrices avant ou arrière), ensuite lui donner une première forme le tout en respectant les conditions préalablement établies, c'est à dire un périmètre de 80cm maximum. Grace à ces données et aux composants disponibles, il faut choisir les matériaux utilisés et concevoir un model 3D du robot une fois assemblé. Une fois toutes ces étapes réalisées, l'usinage des composants s'impose, puis l'assemblage final.

2.1.3 Conception du robot

Avant de commencer à monter le robot, il nous a d'abord fallu le concevoir à l'aide d'un logiciel de CAO, solidworks. Ensuite, une fois conçu, nous avons pu passer à la phase de fabrication, respectant scrupuleusement le modèle 3D de solidworks. Nous avons ainsi dû scier une base pour le robot, limer ensuite les bords afin d'obtenir une côte nette, puis installer tous les composants du robot, soit les moteurs, le circuit imprimé et la carte arduino, tout en laissant de la place pour les batteries. En raison d'un problème de place sur le robot,

nous avons été dans l'obligation de construire un étage en pvc au-dessus du circuit imprimé pour poser la carte arduino ainsi que le motor shield.

2.1.4 Alimentation

Romain possédant des batteries de meilleure qualité que celles proposées par l'INSA, il s'est proposé de les mettre à disposition pour le robot. Afin d'obtenir une tension de plus de 12V, nous avons donc décidé de coupler deux batteries li-po de Romain, montant ainsi à 14,8V à pleine charge. Pour cela, nous avons donc dû créer un adaptateur afin de pouvoir les brancher sur la carte arduino.

2.1.5 Difficultés rencontrées

Ayant eu la suppression de plusieurs séances de projet, on s'est vus prendre un peu de retard dans tous les parties, notamment en Electronique. Cependant, étant donné que la partie mécanique ne pouvait continuer sans le circuit imprimé conçu par le groupe s'occupant de l'électronique, elle a été dans l'obligation de patienter la réception de cette pièce. Le retard s'est donc accumulé sur la fin du semestre, ce qui nous a valu de travailler pendant tous nos temps libres en salle de robotique, quand celle-ci était ouverte.

2.2 Partie Electronique

2.2.1 Explication du pôle

Le rôle du pôle électronique du projet était principalement de faire le lien entre le programme de l'Arduino, et la mécanique du robot : déterminer la position optimale des capteurs, faire en sorte qu'ils puissent accomplir leur tâche au mieux, et assurer la communication entre eux et l'Arduino, puis entre l'Arduino et les moteurs.

2.2.2 Méthode de travail

Le pôle électronique se composant de deux personnes, il était impératif que nous puissions travailler en équipe. Nous avons donc tous deux téléchargé le logiciel KiCad, servant pour la réalisation de la carte, et tout en avançant chacun de notre côté, nous avons pu travailler de concert. Nous avons utilisé les technologies de communication qui s'offraient à nous pour pouvoir coopérer en tout temps : conférences Skype, mise à jour de notre travail sur Dropbox, messages pour le groupe sur Facebook, etc. Nous avons même pu travailler en même temps sur le même document KiCad à distance, grâce au logiciel de partage d'écran TeamViewer.

Toutes ces technologies, nous ont permis de bien avancer la conception de la carte, que ce soit pendant les sessions de travail individuelles ou collectives.

2.2.3 Approche de l'objectif

Formation

La première étape de notre travail fut une grande période d'auto-formation : les cours d'électricité que nous avons suivis en première année étaient bien trop théoriques pour nous permettre de réaliser quoi que ce soit. Nous avons donc trouvé des cours d'électronique appliquée sur le net, qui nous ont permis de mieux cerner l'ensemble du problème. La phase de formation théorique fut vite achevée, et nous avons pu passer assez tôt à ce qui fut la partie majeure de notre travail : l'élaboration de la carte sur ordinateur.

Prise en main du logiciel

Lors de la deuxième séance, Fabrice Delamare, nous a montré les fonctionnalités générales du logiciel KiCad : de la conception du schéma électrique jusqu'à l'édition du circuit imprimé. Nous avons complété les connaissances apportées pendant ce cours avec des tutoriels disponibles sur internet. Nous avons appliqué directement ces quelques notions sur le logiciel, et le reste est venu avec la pratique.

2.2.4 Conception de la carte électronique

Conception assistée par ordinateur

Nous avons donc utilisé le logiciel libre KiCad, qui prend en charge l'élaboration d'un circuit imprimé de la phase schématique au modèle final en trois dimensions. Nous avons commencé par dessiner le schéma électrique de notre montage. Le robot nécessitait l'emploi de quatre capteurs à infrarouge TCRT5000, chacun d'eux lié à deux résistances pour adapter la tension, et d'un connecteur 6 pins assurant la communication avec le MotorShield, et donc

l'Arduino. Ensuite vient le dessin de la carte, où sont définis le placement et l'établissement des connexions physiques entre les composants. Voir figure C.1 et C.2.

Nous avons choisi une configuration minimaliste. Nous avons disposé les capteurs en cerf-volant, ceci afin d'optimiser le déplacement du robot, avec le connecteur au centre. Les résistances sont ensuite placées en carré tout autour. Après quelques tentatives et échanges avec le professeur et nos camarades du groupe 3, en ce qui concerne la méthode pour tracer les pistes, notre carte fut finalement prête à être imprimée ! Voir figure C.3

Soudure

La partie pratique commença lorsque nous avons reçu notre circuit imprimé fini. Nous devons souder tous les composants selon la configuration définie lors du dessin de la carte, avec les résistances et le connecteur d'un côté de la carte, et les capteurs de l'autre. La carte possédant des pistes des deux côtés (double face), nous avons dû demander des conseils sur comment souder les composants.

L'étape finale consistait à relier notre carte au MotorShield pour finaliser le montage électronique. Nous avons eu l'aide précieuse du technicien Mr.Duhamel, il nous avait préparé les fils qui partent du connecteur du circuit imprimé. Il nous restait donc plus qu'à dénuder les fils, souder des pins, pour enfin connecter notre circuit imprimé au connecteur du MotorShield. La dernière étape avant les premiers tests consistait à connecter les moteurs sur le MotorShield ainsi que les batteries. Voir annexe C.4.

Difficultés rencontrées

La principale difficulté s'est trouvée comme prévu dans l'édition du circuit imprimé. En effet, nous avons dû réaliser plusieurs versions de celle-ci car elle ne remplissait pas certains critères. Nous avons dû changer plusieurs fois la position des composants jusqu'à arriver à une disposition optimale.

Le problème suivant résidait dans le dessin des pistes reliant le connecteur aux capteurs, celles-ci devant toutes se trouver du même côté.

Lors de la phase de soudure, une complication imprévue nous attendait : nous avons dû effectuer des doubles soudures sur les TCRT5000, dont les pattes étaient très courtes, ce qui impliquait le risque de brûler les composants.

Nous avons su surmonter tous les obstacles qui se sont dressés sur notre chemin, pour finalement arriver à un résultat plus que satisfaisant.

2.3 Partie Programmation

2.3.1 Explication du pôle

Notre mission est donc d'élaborer la partie "intelligence" du robot. Celui-ci doit être implémenté pour rendre le robot capable de suivre une ligne. Elle se met en oeuvre grâce au micro-contrôleur programmable Arduino Uno à l'aide d'un langage dérivé du C.

2.3.2 Approche de l'objectif

Formation

Afin de pouvoir pleinement maîtriser l'arduino, il nous a été nécessaire de nous auto-former. Pour cela nous nous sommes attelés à plusieurs recherches sur l'internet, afin de déterminer le tutoriel le plus pertinent. Nous nous sommes donc mis d'accord sur l'excellent tutoriel du site de zéro, concernant le langage. Une remise à en niveau en électronique et électricité nous a été nécessaire afin de nous constituer les bases de connaissances les plus solides possibles. En effet, notre formation en électronique bien que bien fournie n'était que très théorique et ne nous permettait pas d'aborder sereinement l'arduino. Fort heureusement, le tutoriel du site du zéro comportait un rappel sur l'électronique. Les vacances de février nous ont permis de nous remettre à niveau.

Appréhension du matériel

L'Arduino dans son ensemble n'est pas trop difficile à appréhender. Son concept de micro-contrôleur programmable avec un code qui tourne en boucle et d'entrées sorties digitales/analogiques ne nous a pas été trop dur à comprendre. Nous avons étudié l'Arduino en profondeur tout de même, comme par exemple l'élément permettant de déterminer la tension analogique en entrée. Nous avons ensuite abordé la partie langage de l'appareil, intégrant la notion d'états bas et hauts, de boucles principales et de types de variable pour économiser au mieux la mémoire vive de cet appareil présente en très basse quantité. Le MotorShield a été plus compliqué à aborder. Son utilisation requiert une configuration non évidente vis à vis du micro-contrôleur. En effet, il ne fallait pas son tromper dans les câblages des moteurs sur celui-ci et bien gérer le sens de rotation des moteurs ainsi que leur vitesse.

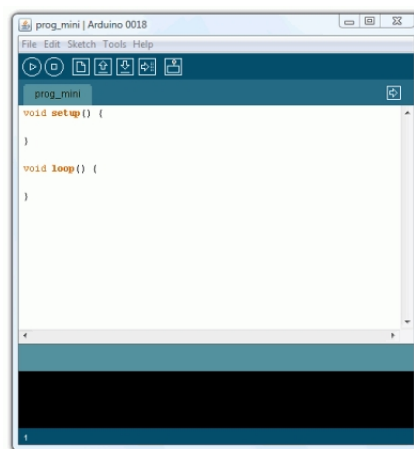


FIGURE 2.1 – Environnement de programmation Arduino

2.3.3 Cahier des charges

Tout d'abord, le programme doit analyser constamment les valeurs renvoyées par les capteurs. Ces valeurs sont en Volts. À partir de ces valeurs le programme doit être capable de situer le robot par rapport à la ligne à suivre. En fonction de cette analyse, le programme doit déterminer la marche à suivre, à savoir : aller tout droit, tourner à gauche, tourner à droite. Le programme doit être léger. Pour des raisons évidentes de lisibilité, surtout dans un programme où nous travaillons à deux, le respect des conventions de rédactions de code est obligatoire. Par exemple bien commenter nos instructions. Etant donné du contrate élevé entre la piste (ruban adhésif noir) et le reste (feuilles blanches), l'instruction `digitalRead`, qui permet dans notre cas de connaître l'état des capteurs, est suffisante. En effet, celle-ci détermine des états hauts et bas en fonction de la tension qu'elle reçoit. Ces deux états sont bien distincts du fait du contrate élevé.

2.3.4 Conception du code

Analyse

Arduino implique l'utilisation de fonctions indispensables. Ce sont des fonctions dites d'initialisation "`setup()`" et de programme principal "`loop()`". Dans l'initialisation nous avons défini les valeurs analogiques renvoyées par les capteurs comme des INPUT et les valeurs digitales envoyées au MotorShield comme des OUTPUT. Dans le programme principal, nous avons d'abord configuré l'interaction avec le MotorShield. Ensuite le programme récupère les valeurs renvoyées par les capteurs. Ce sont les lignes de codes suivantes qui permettent les changements de directions : grâce à deux itérations indéterministes dont les conditions reposent sur les valeurs des capteurs récupérées à chaque itération. Voir annexe B.

Concept d'innovation

De manière à rendre le robot plus performant, nous avons réfléchi à plusieurs manières d'optimiser le programme par des procédures automatiques de calcul d'angle de virage permettant de garantir une vitesse maximale pour chaque type de virage rencontré. Ces procédures n'ont été faisables étant donné le niveau de complexité mathématique d'un tel algorithme.

Difficultés rencontrées

La plus grosse difficulté de la programmation de micro-contrôleur est de ne pas pouvoir tester l'efficacité de son code. En effet, même si notre code source nous semble juste et compile, il nous est impossible de savoir s'il n'y pas d'erreur de conception et s'il sera efficace. Pour ne pas se retrouver avec un robot non fonctionnel sur les bras, il nous a paru évident de prévoir des alternatives. Le but a donc été d'obtenir le même résultat avec différentes techniques.

Une seconde difficulté a été le changement de structure de programmation. En effet contrairement à un programme classique composé de plusieurs boucles qui s'exécutera d'une traite, l'arduino s'exécute sur un boucle pseudo-infinie. Cela nous a donc destabilisé un certain temps et nous a conduit à quelques erreurs de conception au début.

Un des éléments le plus délicat concerne le test du code sur le robot. En effet, lors de la première expérimentation, le robot ne semblait rien détecter et nous avons dû effectuer

plusieurs tests pour éliminer la source du problème (Composants...) à l'aide du pôle électronique . Enfin, après plusieurs heures de test, nous avons pu déterminer une vitesse de croisière et une vitesse de virage adéquate, afin que la détection soit optimale.

Conclusion et perspectives

2.4 Conclusion sur le travail réalisé

Ce projet est dans son ensemble une réussite. En effet l'objectif escompté est atteint, le robot suit la ligne noire. Ainsi notre organisation des pôles et notre méthode de travail ont porté leurs fruits. La grande partie d'autonomie, en particulier l'auto-formation, a nécessité beaucoup de temps, ce qui a requis de travailler sur le projet en dehors du créneau prévu. Ainsi malgré un retard certain, le projet a pu être rendu dans les temps. L'aboutissement du projet nous a tous beaucoup apporté, tant sur le plan personnel, que sur le plan technique. La réussite de ce premier projet pluridisciplinaire nous a tous conforté dans nos choix d'orientation plus tard.

2.5 Conclusions sur l'apport personnel de cette E.C. projet

Etienne : Depuis le début de notre formation à l'INSA, nous avons presque toujours été formé très théoriquement sur de nombreux domaines. Cela a souvent été une source de frustration pour moi car j'ai toujours eu du mal à apprendre sans avoir des perspectives d'application. C'est pourquoi la réalisation complète d'un robot, a été pour moi une grande satisfaction, car j'ai pu utiliser mes connaissances. Travailler à plusieurs sur un projet dans différents domaines très distincts a aussi été une grande nouveauté en comparaison d'autres projets. C'est justement sur ce point que ce projet m'a été le plus bénéfique car il m'a permis d'acquérir de l'expérience dans la gestion d'un groupe dans des domaines hétéroclites. Selon moi, cette façon de travailler est celle qui se rapproche le plus de celle d'un ingénieur.

Nathan : Ces projets de P6 permettent de concrétiser les différentes connaissances acquises depuis mon arrivée à l'INSA. Malgré cela, ces connaissances restent très minimes par rapport à ce qui doit être mise en oeuvre pour réaliser un tel robot. Voulant faire ASI, ce projet s'inscrivait parfaitement dans mon cursus, en effet on trouve de la robotique dans ce département. La mise en relation de la programmation, de l'électronique ainsi qu'un domaine que je connais pas, la mécanique, m'a permis d'avoir une vision globale sur comment faire un robot. Même si cette vue d'ensemble n'est pas très développée en raison de la "simplicité" du robot. La partie programmation était relativement simple. Je me suis également intéressé à la partie électronique, étant directement liée à mon domaine de compétence. Concernant notre travail, l'autonomie étant quasi totale, nous avons eu des difficultés, surtout dans la partie électronique, que nous devons intégrer.

Jean-Claude : J'ai appris énormément de choses grâce à ce projet P6. Celui-ci m'a permis de mettre en pratique mes connaissances et m'a poussé à apprendre de nouvelles notions et de pouvoir les utiliser instantanément. Cette matière m'a fait connaître le travail en

groupe. Ce n'est pas du tout pareil que de travailler en binôme, on a beaucoup plus de responsabilités. De plus, on peut apprendre les uns des autres. Souhaitant intégrer le département ASI, j'ai trouvé ce projet très intéressant, puisque c'est pour la robotique que j'ai choisi ce département. Avec Lucas, je m'occupais de la partie électronique, mais toutefois j'ai été très attentif à ce que faisait les autres membres de mon groupe. J'ai notamment appris de nouvelles notions concernant la programmation et la mécanique. Notre partie du projet n'a pas été simple puisque l'on devait maîtriser un nouveau logiciel et aussi se rappeler de connaissances vu en première année. L'agencement optimal des différents composants sur le circuit imprimé nous a pris beaucoup de temps mais nous sommes finalement arrivés à un résultat convenable. En somme, ce projet a été pour moi très intéressant et nous délivre une vision de ce que sera notre futur métier d'ingénieur. Enfin, ayant eu peur des robots étant petit, c'est une satisfaction personnelle d'avoir pu dompter ma peur et d'avoir créé un robot.

Lucas : Ce projet fut pour essentiellement moi une occasion de tester mes facultés d'adaptation dans une configuration peu familière : un projet expérimental, s'appuyant sur des notions majoritairement nouvelles, à réaliser en collaboration avec des partenaires inconnus. Alors qu'il arrive à son terme, je peux dire que l'expérience m'a été très bénéfique, en terme de méthode de travail en groupe, ainsi qu'en terme de compétences pures. Je me suis inscrit pour ce projet avec l'intention d'en assurer la partie code, mais la discussion du groupe sur la répartition des tâches en a décidé autrement. Cependant, je ne regrette absolument pas d'avoir été affecté à l'électronique, considérant que cela m'a appris énormément de choses, bien plus que si je m'étais "contenté" d'apprendre le langage Arduino. De plus, le côté "livré à nous même", si il a été difficile à gérer au début, nous a permis de vraiment mettre "notre touche personnelle" dans notre robot. En résumé, un ressenti très positif sur la matière P6 et le projet Robot Suiveur de ligne.

Romain : J'avais choisi ce projet car je le trouvais largement plus intéressant que les autres, regroupant plusieurs domaines de compétences tels que la programmation, la mécanique ou l'électronique. Bien que j'aie dû me concentrer sur la partie mécanique sans trop pouvoir toucher aux autres domaines, ce projet s'est montré formateur. Tout d'abord, ce projet nous permet d'apprendre à travailler en groupe. En effet, le projet étant plutôt conséquent, une répartition des tâches efficace est obligatoire, et chacun doit se tenir à son objectif, sans gêner le travail des autres. Ensuite, en partie mécanique, nous avons dû monter le robot avec les outils disponibles, et parfois trouver des alternatives à ce qui était prévu à la base, afin que ce soit réalisable.

Guillaume : Ce projet m'a permis de travailler en groupe sur un projet plus conséquent que les précédents et présentant plusieurs aspects (mécanique, électronique et informatique), et ensuite de concrétiser notre travail, ce que j'ai vraiment apprécié. Grâce au projet j'ai pu me rendre compte de ce qu'est faire partie d'un pôle et le réaliser de A à Z, étape par étape. Ce robot s'inscrit parfaitement dans ma thématique ASI-MECA et ayant fait partie du pôle meca j'ai pu voir ce qu'est un projet dans ce domaine. J'ai longtemps hésité entre ASI et MECA pour la 3^e année et le fait de faire ce projet m'a même aidé dans mon choix. Je suis donc très content d'avoir pu participer à ce projet de P6.

2.6 Perspectives pour la poursuite de ce projet

La conception et la réalisation d'un robot autonome de A à Z nous a tous beaucoup plu. Néanmoins, par manque de temps et de connaissances nous n'avons pas pu mettre en oeuvre toutes les idées que nous avons eu tout au long du projet. Il nous semble que pour la poursuite du projet irait dans les direction suivantes :

- Un algorithme de calcul de vitesse optimale des trajectoires
- Une meilleure esthétique générale
- Un contrôle manuel du démarrage

Bibliographie

- [1] <https://www.sparkfun.com/tutorials/195> (Valide à la date du 16/06/13)
- [2] <http://www.siteduzero.com/sciences/tutoriels/arduino-pour-bien-commencer-en-electronique-et-en-programmation> (Valide à la date du 16/06/13)
- [3] <http://www.ferdinandpiette.com/blog/2012/04/kicad-creer-un-nouveau-composant/> (Valide à la date du 16/06/13)
- [4] <http://www.ferdinandpiette.com/blog/2012/04/kicad-creer-une-nouvelle-empreinte/> (Valide à la date du 16/06/13)
- [5] <http://www.ferdinandpiette.com/blog/2012/04/kicad-dessiner-le-schema-electronique-du-montage/> (Valide à la date du 16/06/13)
- [6] <http://www.ferdinandpiette.com/blog/2012/04/kicad-concevez-le-circuit-imprime/> (Valide à la date du 16/06/13)

Annexe A

Documentation technique

TCRT5000, TCRT5000L



Vishay Semiconductors

Reflective Optical Sensor with Transistor Output

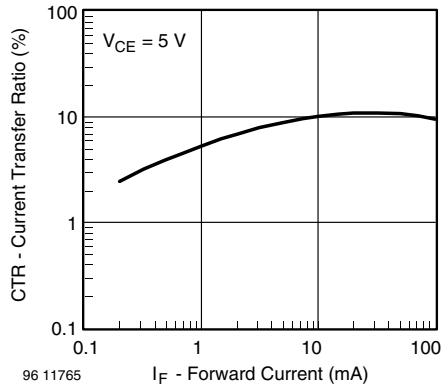


Fig. 8 - Current Transfer Ratio vs. Forward Current

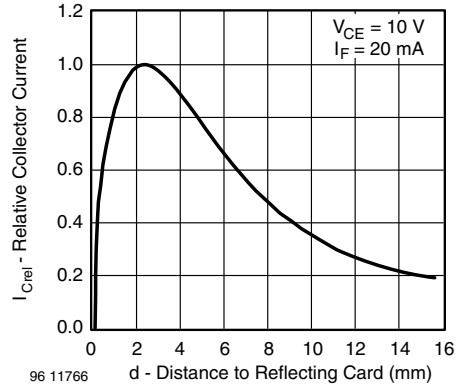
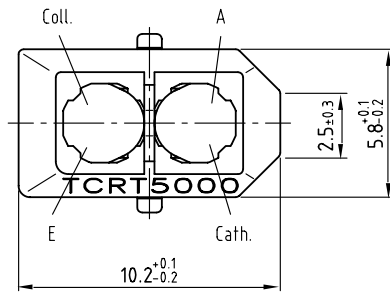
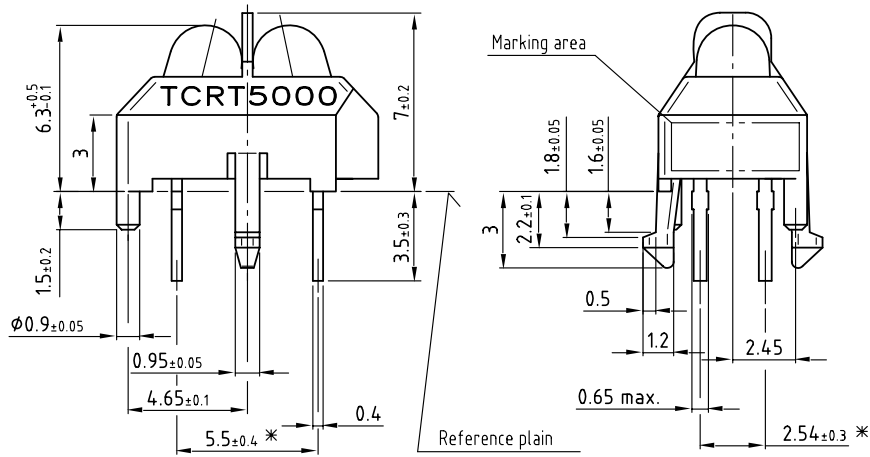


Fig. 9 - Relative Collector Current vs. Distance

PACKAGE DIMENSIONS in millimeters, TCRT5000



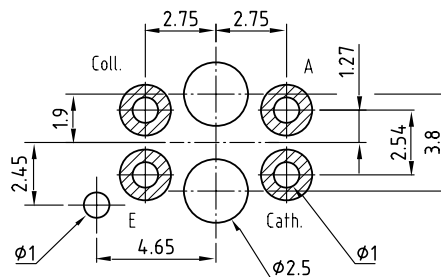
* Tolerances related to reference plain

weight: ca. 0.23g



Technical drawings according to DIN specifications

Footprint Top View



Drawing-No.: 6.550-5096.01-4

Issue: 4; 11.04.02

96 12073

Annexe B

Listings des programmes réalisés

```
const
  //Declaration Capteur Haut, Gauche, Droit, Bas
  int capteur_tete = 4;
  int capteur_gauche = 5;
  int capteur_droit = 6;
  int capteur_arriere = 7;

  //Declaration Vitesse roue gauche, Vitesse roue Droite
  int speed_gauche = 11; //OUTPUT 1 & 2
  int speed_droit = 10; //OUTPUT 3 & 4
  //Declaration direction roue gauche, direction roue droite
  int dir_gauche = 12;
  int dir_droit = 13;

  //Declaration valeur des capteurs Haut, Gauche, Droite, Bas
  int value_tete;
  int value_gauche;
  int value_droit;
  int value_arriere;
  int time;
  //Declaration vitesse des roues, exterieur et interieur, en cas de
  virage
  int vitesse_roue_ext = 70;
  int vitesse_roue_int = 30;

  //fonction qui (re)definit la vitesse des roues a 100
void vitesse_de_croisiere(void) {
  analogWrite(speed_gauche, 80);
  analogWrite(speed_droit, 80);
}

void setup()
{
  //Definit les capteurs en INPUT
  pinMode(capteur_tete, INPUT);
  pinMode(capteur_gauche, INPUT);
  pinMode(capteur_droit, INPUT);
  pinMode(capteur_arriere, INPUT);
```

```

40 //Definit la vitesse de roues et la direction des roues en OUTPUT
pinMode(speed_gauche, OUTPUT);
pinMode(speed_droit, OUTPUT);
pinMode(dir_gauche, OUTPUT);
45 pinMode(dir_droit, OUTPUT);

}

void loop()
50 { //Temps de boot au demarrage (5s)
time=millis();
if(time==0)
{
55 delay(5000);
}

value_tete = digitalRead(capteur_tete);
value_gauche = digitalRead(capteur_gauche);
60 value_droit = digitalRead(capteur_droit);
value_arriere = digitalRead(capteur_arriere);
digitalWrite(dir_gauche, HIGH); //Parametrage de la direction du moteur
gauche
digitalWrite(dir_droit, LOW); //Parametrage de la direction du moteur
droite

65 //Arret du robot en cas de perte de ligne
if ( (value_tete) && (value_gauche) && (value_droit) && (value_arriere)
)
{
analogWrite(speed_gauche, 0);
analogWrite(speed_droit, 0);
70 }
else
{
vitesse_de_croisiere();
}

75 //recuperation des valeurs renvoyees par les capteurs
value_tete = digitalRead(capteur_tete);
value_gauche = digitalRead(capteur_gauche);
value_droit = digitalRead(capteur_droit);
80 value_arriere = digitalRead(capteur_arriere);

//Cas d'un virage a droite
while ( (value_tete) && (value_gauche) && (!(value_droit)) )
{
85 digitalWrite(dir_droit, HIGH);
digitalWrite(dir_droit, LOW );
analogWrite(speed_gauche, vitesse_roue_ext); // Parametrage de la
vitesse du moteur gauche
    
```

```

    analogWrite(speed_droit,vitesse_roue_int);// Parametrage de la
        vitesse du moteur droit

    //recuperation des valeurs renvoyees par les capteurs
    value_tete = digitalRead(capteur_tete);
    value_gauche = digitalRead(capteur_gauche);
    value_droit = digitalRead(capteur_droit);
    value_arriere = digitalRead(capteur_arriere);
    }
    vitesse_de_croisiere();

//Cas d'un virage a gauche
while ( (value_tete) && (!(value_gauche)) && (value_droit) )
    {
        digitalWrite(dir_gauche,LOW);
        digitalWrite(dir_gauche,HIGH );
        analogWrite(speed_gauche,vitesse_roue_int);
        analogWrite(speed_droit,vitesse_roue_ext);

        //recuperation des valeurs renvoyees par les capteurs
        value_tete = digitalRead(capteur_tete);
        value_gauche = digitalRead(capteur_gauche);
        value_droit = digitalRead(capteur_droit);
        value_arriere = digitalRead(capteur_arriere);
    }
    vitesse_de_croisiere();
:}

```


Annexe C

Schémas de montages, plans de conception

C.1 Partie électronique

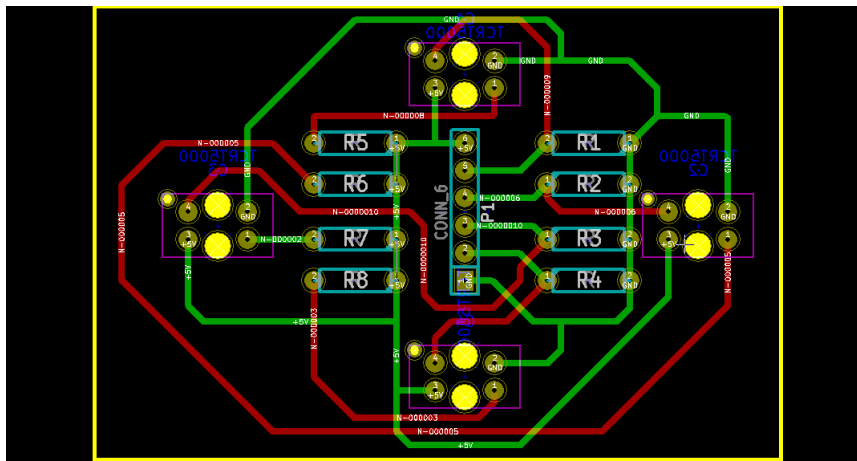


FIGURE C.1 – Visualisation du circuit imprimé

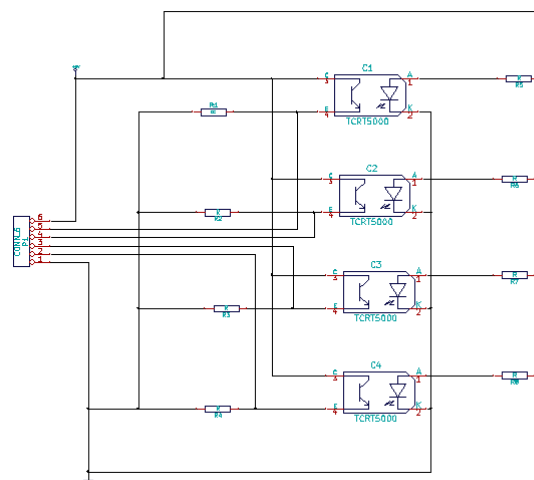


FIGURE C.2 – Visualisation du schéma du circuit imprimé

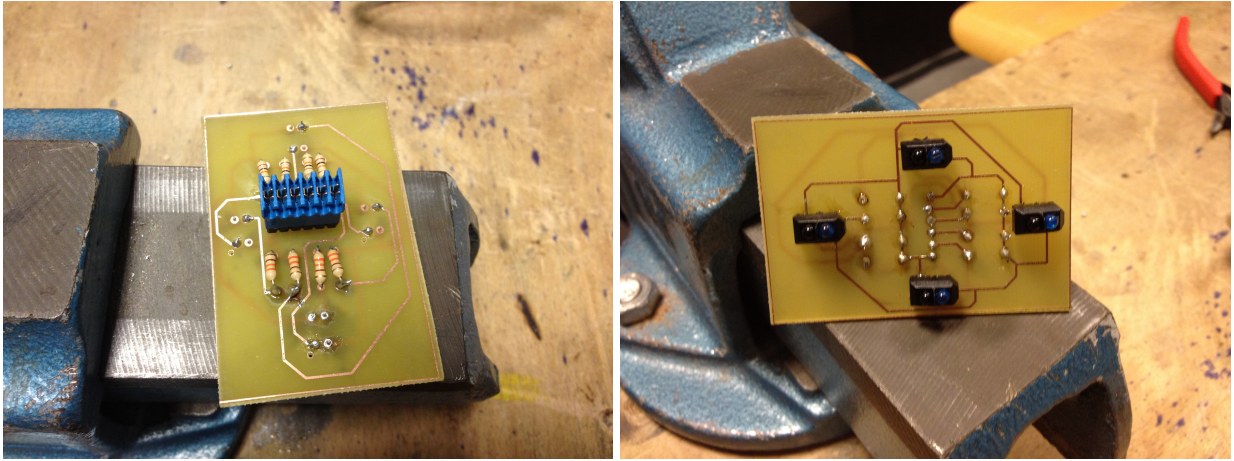


FIGURE C.3 – Photographies du circuit

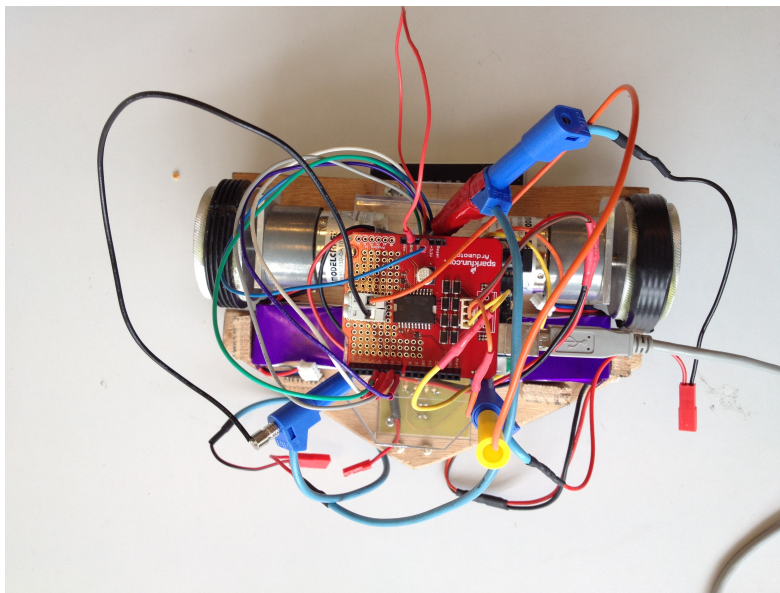


FIGURE C.4 – Photographie du Robot avec tous les branchements

C.2 Partie Mécanique

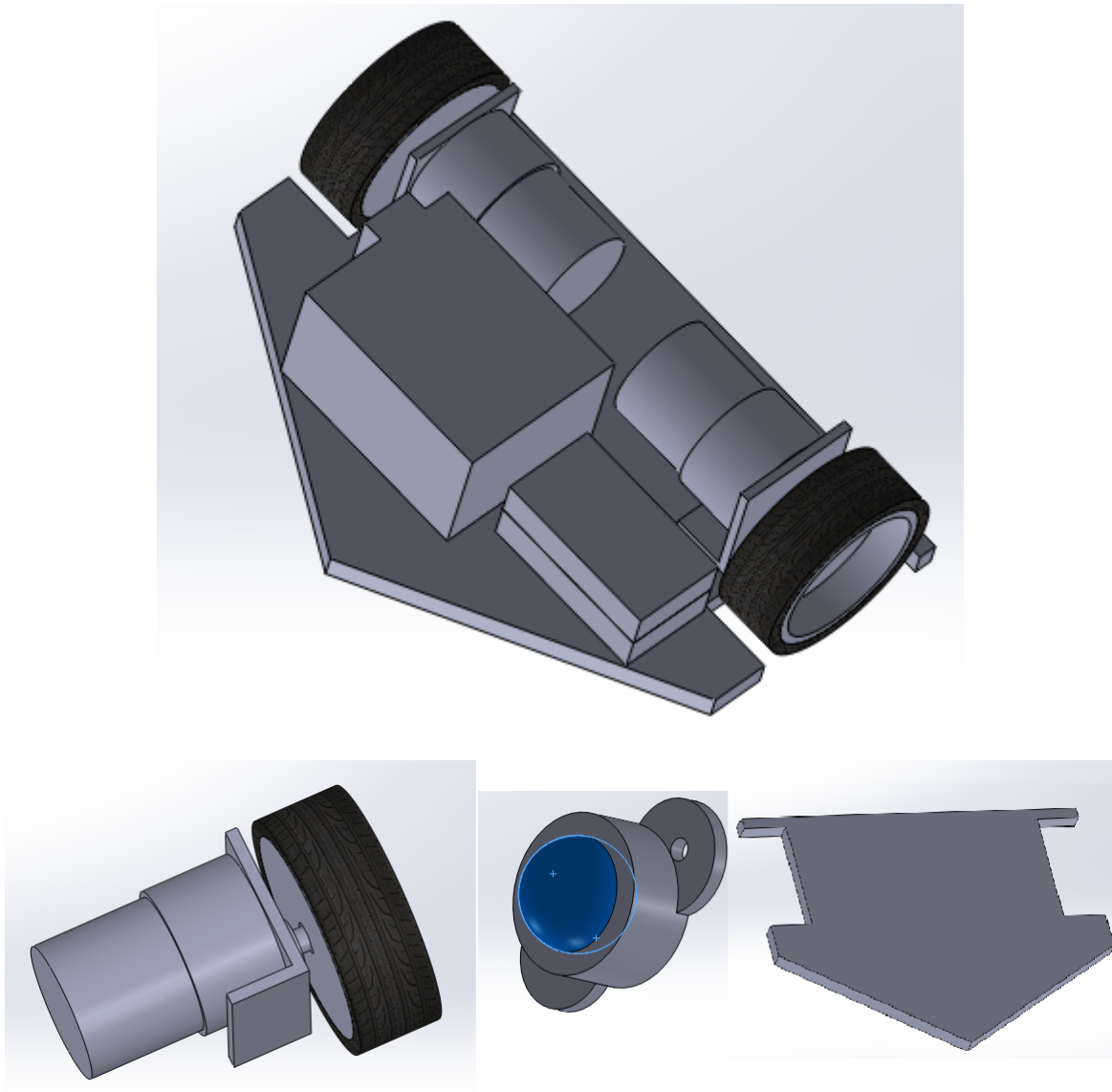


FIGURE C.5 – Schéma du robot réaliser sous SolidWorks