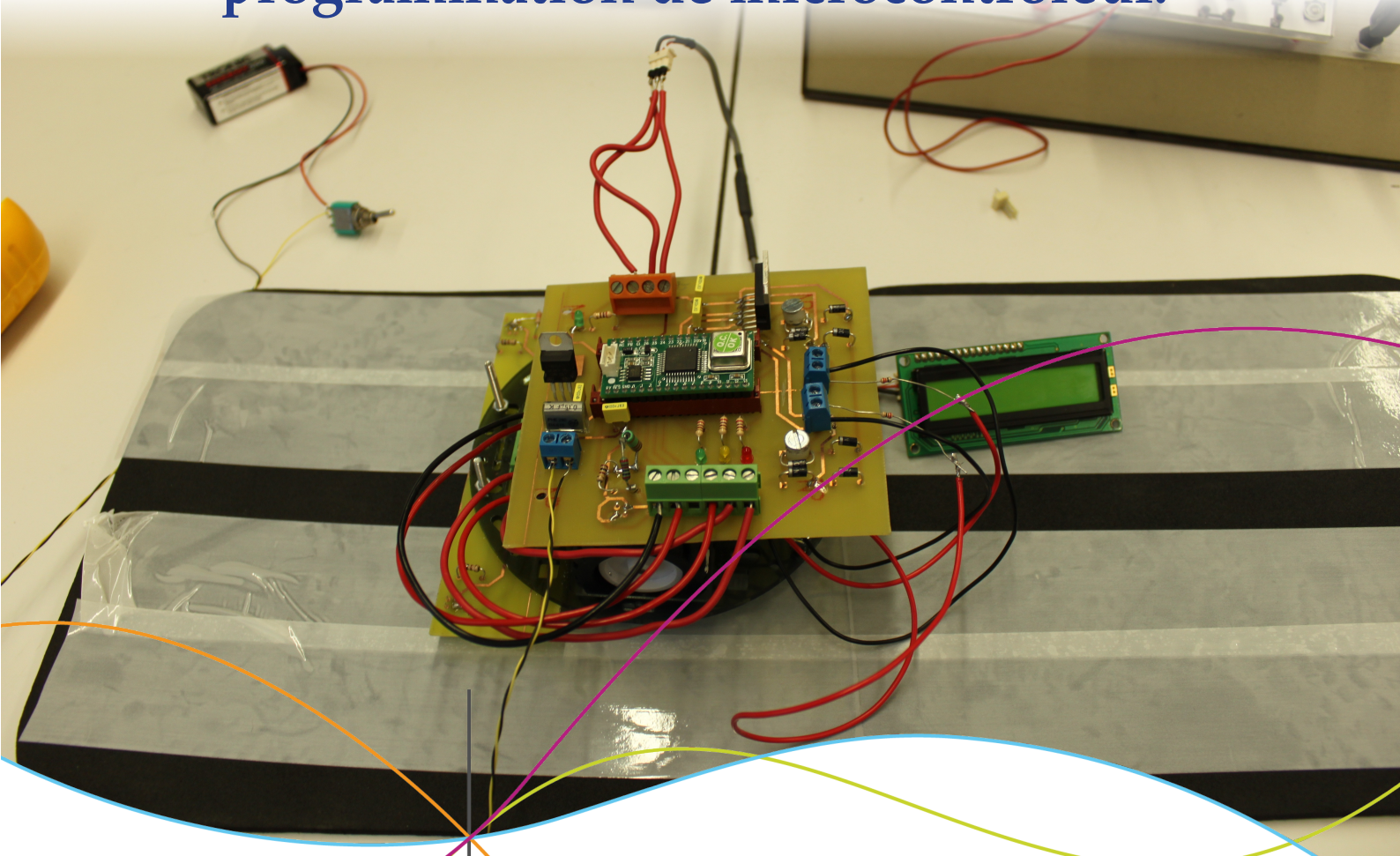


Développement de robot à base de modules Lextronic mettant en œuvre la programmation de microcontrôleur.



Enseignant responsable
Mounir Messedi

Étudiants :
Jean-Baptiste LOUVET
Adrien COULIER
Dong YAN

Mélanie ALVES
Violaine PIGNOL
Nicolas MILLER

Date de remise du rapport : 17/06/12

Référence du projet : STPI¹/P6/2012-5

Intitulé du projet : Développement de robot à base de modules Lextronic mettant en œuvre la programmation de microcontrôleur.

Type de projet : expérimental

Objectifs du projet :

Notre objectif global est de développer un robot correspondant à nos exigences : c'est à dire capable de suivre une ligne et d'afficher sur un écran la distance parcourue. Le robot doit être capable de faire demi-tour au bout de la ligne et l'utilisateur doit savoir à tout moment dans quel état se trouve robot, c'est-à-dire si le robot suit la ligne, la cherche ou fait demi tour.

Ce projet doit également nous donner l'opportunité d'appliquer nos connaissances théoriques dans une situation concrète.

Mots-clefs du projet : développement, robot, programmation, automatisme.

Remerciements :

Nous tenons à remercier chaleureusement monsieur Messedi pour l'aide qu'il nous a apporté durant ce projet ainsi que toutes les personnes ayant participé de près ou de loin à la réalisation de ce projet et à notre formation scientifique.

Table des matières

Notations	4
Introduction	5
1 Méthodologie, organisation du travail	6
1.1 Nos attentes	6
1.2 Description du projet	6
1.2.1 Intitulé, description du projet	6
1.2.2 Définition des besoins, objectif	7
1.2.3 Contraintes	7
1.3 Planification	8
1.4 Organisation et répartition des tâches	9
2 Travail réalisé et résultats	10
2.1 Travail réalisé par l'équipe programmation	10
2.1.1 Installation du logiciel PicBasic Studio	10
2.1.2 Rédaction du programme gérant la direction du robot	10
2.1.3 Rédaction du programme gérant la direction du robot	11
2.2 Travail réalisé par l'équipe électronique	12
2.2.1 Description des composants	12
2.2.2 Création du circuit à l'aide d'un logiciel de CAO (kicad)	14
2.2.3 Problèmes rencontrés	16
Conclusion et perspectives	18
3.1 Ressentis dans le groupe	18
3.2 Conclusion générale	19
3.3 Pistes d'amélioration du robot	20
Bibliographie	21
A Listings des programmes réalisés	22
A.1 Programme général	22
B Schémas de montages	24
C Propositions de sujets de projets (en lien ou pas avec le projet réalisé)	26

Notations et Acronymes

Polulu : nom donné à notre robot, en raison du nom du châssis sur lequel nous l'avons assemblé (à savoir Pololu).

PIC : PicBasic.

Introduction

Durant le dernier semestre de cycle préparatoire de l'INSA de Rouen, il est demandé aux élèves de réaliser un projet complet de physique. Cela représente un nouveau challenge car c'est la première fois pour nous qu'il faut réaliser un projet de A à Z, du cahier des charges à la réalisation concrète d'un objet fini. C'est une nouvelle occasion de pratiquer le travail en équipe. Cela s'avère à la fois effrayant, car nous serons en charge de tous le projet, et excitant car c'est un nouveau défi à relever pour notre équipe et un pas de plus vers le métier d'ingénieur. Le défi est d'autant plus intéressant à relever que nous aurons à nous auto-former dans certains domaines qui nous sont pour l'instant totalement inconnus. Cela représente un premier pas vers le monde professionnel où il est important de gérer une dynamique de groupe, et de savoir composer avec des contraintes techniques et temporelles.

Le projet est réalisé par six étudiants, qui travaillent au développement d'un robot à l'aide d'un micro-contrôleur. Le robot à réaliser doit posséder plusieurs fonctionnalités : notre robot suivra une ligne et fera demi-tour au bout de celle-ci. Il affichera également la distance parcourue. L'objectif d'un tel projet est d'appliquer nos connaissances dans le domaine informatique et d'approfondir nos connaissances en électronique.

Ce rapport fait le bilan de notre projet, en expliquant notre méthodologie et les problèmes rencontrés.

Chapitre 1

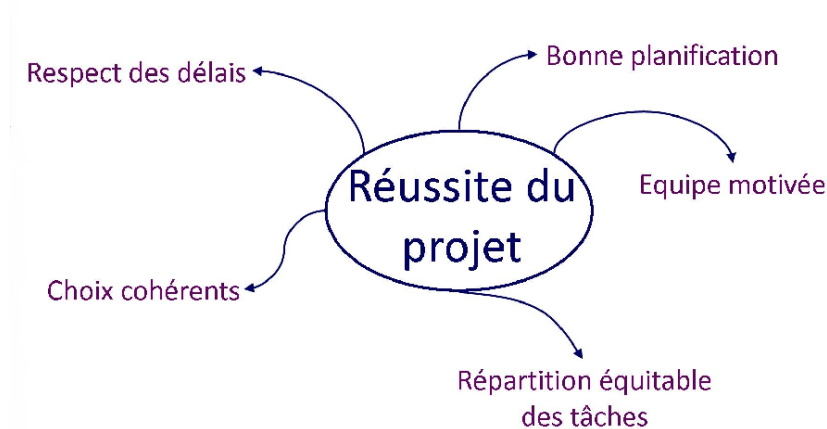
Méthodologie, organisation du travail

1.1 Nos attentes

Lors du premier cours de projet physique, nous avons pris connaissance de notre groupe, un groupe de six personnes. Très vite, nous avons discuté de nos attentes vis-à-vis du projet, et discuté des différentes possibilités qui s'offraient à nous. Nos étions en accord sur certains points : nous voulions réaliser un robot intelligent, qui possède plusieurs fonctionnalités, certaines simples mais également d'autres un peu plus complexe. Nous devons tenir compte du budget ainsi que des connaissances dont nous disposions.

Mais la principale attente vis-à-vis de l'EC projet physique était d'appliquer nos connaissances scientifiques concrètement, certains en programmation, d'autres en électricité et électronique.

La réussite d'un projet peut être résumée ainsi :



Il nous appartient donc de réunir tous les éléments nécessaires pour réussir notre projet.

1.2 Description du projet

1.2.1 Intitulé, description du projet

Pour mener à bien un projet, il est impératif de définir clairement les objectifs . Il faudrait ensuite définir les ressources requises, définir la planification de la mise en œuvre, les outils d'évaluation et enfin les méthodes de contrôle.

Le cahier des charges constitue une étape dans le processus suivant :

- Exploration de possibilités techniques
- Obtention d'un accord sur le projet au sein de l'équipe

L'UV « projet physique » permet aux étudiants de 2eme année de sortir du système éducatif classique. Elle leur offre l'opportunité de faire un « projet », de travailler en groupe tout en exploitant leurs connaissances dans le domaine scientifique.

Notre groupe a travaillé sur le développement d'un robot. Notre projet est le projet n°5 et s'intitule « Développement d'un robot à base de modules Lextronic mettant en œuvre la programmation de micro-contrôleur ». Ce projet, de nature expérimentale, demande des connaissances en informatique ainsi qu'en électricité. L'objectif global est de créer un robot intelligent, capable de se déplacer.

1.2.2 Définition des besoins, objectif

Suite à de nombreuses discussions, il est nécessaire d'explicitier les différents objectifs à accomplir durant la création du robot et les contraintes que celui nous impose :

Objectif 1 : le robot suit une ligne.

- Nécessité d'intégrer au robot des capteurs infrarouges pour pouvoir détecter la ligne.
- Le mouvement du robot doit être fluide.
- En cas de perte du signal, le robot doit retrouver la ligne rapidement.

NB : Le robot comportera trois capteurs à l'avant. Le premier capteur, au milieu, doit détecter la ligne. Lorsque le capteur du milieu perdra le signal, l'un des deux autres capteurs, situé à gauche et à droite du premier capteur, détectera la ligne. Le robot corrigera ainsi sa trajectoire intelligemment.

Objectif 2 : le robot affiche la distance parcourue tout au long de son parcours.

- Nécessité d'intégrer un quatrième capteur infrarouge, à placer devant l'une des deux roues.
- Nécessité d'intégrer un écran LCD qui affichera la distance parcourue.

NB : Le capteur sera placé devant une des deux roues. Il comptera le nombre de tours de roues grâce à une bande blanche autocollante placée sur la roue. Pour plus de précision, nous avons choisi de placer 4 bandes blanches sur la roue. La distance sera affichée en centimètres grâce à l'écran LCD.

Objectif 3 : le robot fait demi-tour au bout de la ligne.

NB : Une ligne « obstacle » sera placée perpendiculairement au bout de la ligne de départ du robot. Lorsque les capteurs recevront en même temps un signal, le robot fera face à la ligne « obstacle » et entreprendra donc un demi-tour.

Objectif 4 : l'observateur doit savoir dans quel état se trouve le robot.

NB : Le robot possède trois LED de couleurs différentes. Une LED verte s'allumera lorsque le robot suivra la ligne, une LED orange s'allumera lorsque le robot corrigera sa trajectoire, et la dernière, une LED rouge, s'allumera lorsque le robot fera demi-tour.

1.2.3 Contraintes

Délais : Afin de garantir la satisfaction des exigences apportées dans le planning établi plus loin, le rapport final doit être rendu le lundi 18 juin 2012 et la présentation orale le vendredi 22 juin. La soutenance est fixée le lundi 25 juin 2012.

Langage : Le micro-contrôleur, ou PicBasic, doit être programmé en Basic.

Utilisation : La distance affichée par le robot doit être fiable. Le robot devra fonctionner sans fil et devra avoir une assez longue autonomie afin de permettre plusieurs utilisations.

1.3 Planification

Les objectifs du planning sont les suivants

- déterminer si les objectifs sont réalisés ou dépassés.
- Suivre et communiquer l'avancement du projet.
- Affecter les ressources aux tâches.

Le planning constitue un élément essentiel dans l'organisation d'un projet. Comme nous avons un créneau réservé spécialement pour l'EC projet physique, le mardi de 16h45 à 18h15, il fut plus aisé pour nous de suivre l'avancement du projet que si nous n'avions pas de créneau réservé.

Semaine	Date	Tâches réalisées	Commentaires
n°2	07/02/12	Définition des objectifs de l'EC	Commande des premières pièces
n°3	14/02/12	Réalisation du cahier des charges	Commande des dernières pièces
n°4	21/02/12	Montage du châssis et apprentissage du Basic	Réception des pièces
n°5	13/03/12	Premier algorithme en Basic, première ébauche du circuit imprimé	
n°6	20/03/12	Installation du logiciel de programmation, dessin du circuit sur PC	Quelques problèmes avec l'installation du logiciel de programmation
n°7	27/03/12	Fin d'installation du logiciel, finalisation du dessin du circuit	
n°8	03/04/12	Premier tests du programme sur un banc d'essai, finalisation de l'algorithme concernant la gestion de la trajectoire	
n°9	17/04/12	Cours rattrapé à la fin de chaque séance	
n°10	15/05/12	Écriture de l'algorithme concernant le calcul de la distance parcourue	
n°11	22/05/12	Écriture du rapport, soudure des premiers composants sur le circuit imprimé	Manque écran LCD
n°12	29/05/12	Écriture du rapport, finalisation de la soudure du circuit imprimé	Pas de banc de test disponible
n°13	05/06/12	Premier test du robot	Problème de résistance au niveau des capteurs.
n°14	12/06/12	Deuxième test du robot, en situation réelle	Problème avec le câble de connexion du LCD, problème de vitesse des moteurs et problème de taille de ligne

1.4 Organisation et répartition des tâches

Le projet est à réaliser par un groupe de six étudiants. Les équipes sont constituées par les étudiants eux-mêmes. Après la détermination des équipes, il faut s'assurer que le projet est réalisable par le groupe : les compétences de chacun doivent permettre de réaliser chaque tâche à effectuer. Les ressources doivent permettre de répondre aux besoins dans le délai fixé. Le but est toujours d'optimiser l'affectation des ressources pour perdre le moins de temps possible et respecter le délai imposé.

Afin de réaliser efficacement le programme, il est important de répartir efficacement les tâches. La répartition des tâches se fait en tenant compte des compétences de chacun. L'organisation du projet découle également du planning, c'est pourquoi il est important de respecter les délais imposés. Le projet est réalisé par six membres, divisé en deux équipes : une équipe programmation (Adrien Coulier, Mélanie Alves, Violaine Pignol) qui est chargée de l'écriture du programme pour le PicBasic, et une équipe électronique (Jean-Baptiste Louvet, Nicolas Miller, Dong Yan) chargée de mettre au point le circuit imprimé du robot.

Le diagramme de Gantt est fait pour décrire le déroulement des tâches en fonction des semaines. Voici celui de notre projet :

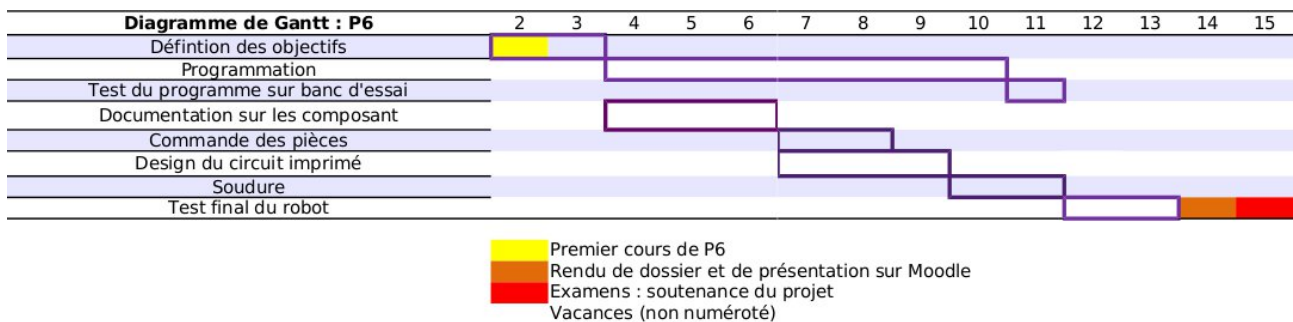


FIGURE 1.1 – diagramme de Gantt de notre projet

Chapitre 2

Travail réalisé et résultats

2.1 Travail réalisé par l'équipe programmation

2.1.1 Installation du logiciel PicBasic Studio

Nous avons décidé d'utiliser le logiciel recommandé par Lextronik , c'est à dire le logiciel PicBasic Studio, L'installation de ce logiciel n'a pas très facile : le port USB qui devait recevoir le banc d'essai n'était pas reconnu par l'ordinateur. Après plusieurs essais infructueux, nous sommes parvenu à résoudre ce problème et à installer le logiciel correctement. Ce problème a tout de même entraîné un léger retard sur le planning prévu, puisque deux séances nous ont été nécessaires pour parvenir à transférer un programme dans le PicBasic.

Étant donné que nous ne connaissions pas tous le langage Basic, nous avons du nous familiariser au langage grâce à des programmes simples rencontrés sur Internet. Nous avons pu ainsi écrire de courts programmes, par exemple faire clignoter une diode ou afficher un message sur l'écran LCD. Nous avons pu tester ces programmes grâce à PicBasic Studio.

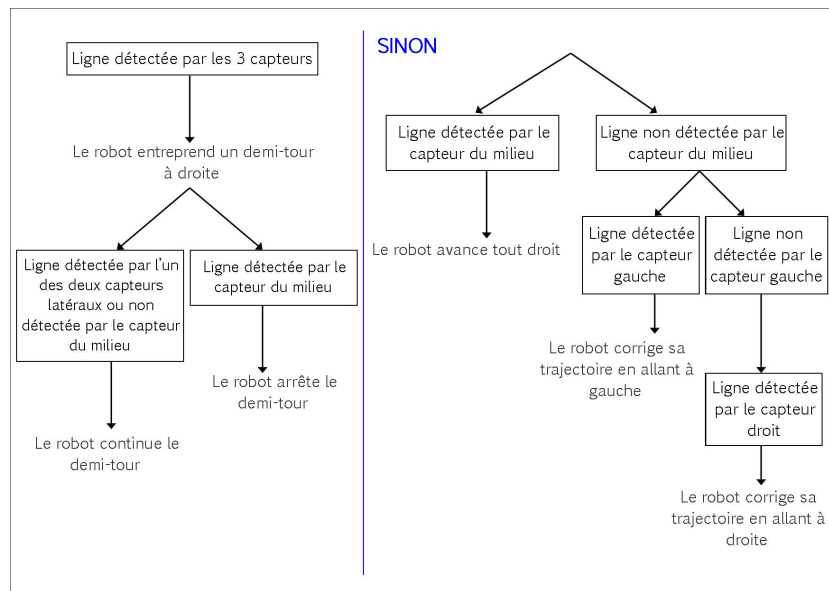
2.1.2 Rédaction du programme gérant la direction du robot

Nous nous sommes ensuite attaqué au programme concernant la direction du robot. Ce programme peut être résumé de la façon suivante. Le robot est équipé de trois capteurs infrarouge. Si le capteur du milieu capte la ligne le robot reçoit l'ordre d'aller tout droit. Lorsque le capteur central ne capte plus la ligne plusieurs situation sont possibles :

- La ligne est capté par le capteur de gauche ,dans ce cas la trajectoire du robot est modifiée et le robot reçoit l'ordre de se décaler vers la gauche afin que le capteur du milieu puisse retrouver la ligne.
- La ligne est capté par le capteur de droite ,dans ce cas le robot reçoit l'ordre de se décaler vers la droite jusqu'à ce que le capteur central retrouve la ligne.
- La ligne n'est capté par aucun des trois capteurs, dans ce cas, nous sommes partis du principe que la ligne devaient se trouver entre le capteur central et un des deux capteurs de coté, nous avons donc initialisé le déplacement du robot de façon à ce qu'il continue à avancer.

Nous avons aussi décidé que le robot doit faire demi tour lorsque il croise une «ligne d'arrêt» ,Cette ligne est une ligne perpendiculaire à la ligne que suit le robot, Ainsi nous avons ajouté une instruction à notre programme qui ordonne au robot de faire demi tour lorsque les trois capteurs capte en même temps une ligne.

Le programme peut être résumer, de façon globale, grâce au schéma suivant :



2.1.3 Rédaction du programme gérant la direction du robot

Afin de mesurer la distance parcourue par le robot, il nous a fallu réfléchir à un moyen de compter le nombre de tours de roues, sans avoir une imprécision trop élevée. Le meilleur moyen de mesurer cette distance est de placer un quatrième capteur infrarouge devant la roue et de tracer une bande blanche sur la roue pour compter le nombre de tours. Il nous est alors très simple de calculer le périmètre de la roue :

$$\pi * d = 3,14 * 4,2 = 13,195cm$$

Afin d'obtenir une meilleure précision pour le calcul et une mise à jour plus rapide sur l'écran LCD, nous avons décidé de placer quatre bandes blanches sur la roue afin de séparer la roue en quart de cercle. Le périmètre du quart de roue est alors $13.195/4 = 3.299$ cm. Nous avons cependant été confrontés à un problème particulièrement handicapant lors de la compilation du programme : les variables réelles n'étaient pas reconnues par PicBasic Studio. Nous avons donc trouvé un moyen de combler l'imprécision due à l'arrondi de certaines valeurs.

Ici le programme complet du calcul de la distance parcourue, avec quelques explications pour bien comprendre le rôle de certaines variables :

```

1 front=0
2 {on initialise front à 0, ce qui correspond à la non-détection d'une bande
   blanche sur la roue}
3 coeff=3
4 {correspond à l'arrondi de la longueur du quart de roue entre chaque bande
   blanche}
5
6 IF IN(capteurroue)=1 THEN
7 front=1
8 END IF
9 {si le capteur de la roue détecte une bande blanche, front prend 1}
10
11 IF IN(capteurroue)=0 AND front=1 THEN
12 front=0
13 distance=distance+coeff
14 compteur=compteur+1
    
```

```

15 {dès que le capteur ne détecte plus la bande blanche, front prend 0, la distance
    parcourue prend la longueur dun quart de roue en plus ; la variable compteur
    est crée pour combler limprécision de larrondi de 3.299 à 3 ; à chaque
    détection dune bande blanche, sa valeur est incrémentée}
16
17 IF compteur > 2 THEN
18     compteur=0
19     distance=distance+1
20 {puis dès que lon effectue  $\frac{3}{4}$  de tours, compteur reprend 0 et la distance prend
    sa valeur +1 ; en effet, cela permet de corriger lerreur commise ( $0.299*3 =$ 
     $0.897\text{cm}$  en moins pour  $\frac{3}{4}$  de tours) lors de larrondi de 3.299 à 3}
21 END IF
22 END IF
    
```

2.2 Travail réalisé par l'équipe électronique

2.2.1 Description des composants

Voici le détail des différents composants de Polulu :

- Le PicBasic « PB-2H »
 - 16 Ko de mémoire programme
 - 96 octets de RAM
 - 34 broches
 - 27 entrées/sorties

Le PicBasic s'alimente en 5V et il est important de noter que la tension des signaux de sortie du Pic est de 5V.

- Afficheur 2x16 caractères :
 - La liaison I2C se connecte à la sortie PicBus (patte 32) du pic.
 - Alimentation en 5 V

- Hacheur L298
 - 4 entrées
 - 4 sorties

Pour protéger le L298 et les moteurs, il faut mettre en place des diodes de protection entre ces deux composants, comme on peut le voir sur le schéma de montage final.

- Capteur CNY70
 - s'alimente en 5V
 - Composé d'une photodiode et d'un phototransistor.
 - Distance optimale d'utilisation : 0,3 mm.

Fonctionnement : la photodiode émet un rayonnement infrarouge sur le spectre de réception du phototransistor. Le rayonnement est renvoyé (il faut être assez près du sol) lorsque la surface du sol est d'une couleur assez claire, ce qui « active » le phototransistor qui applique un tension en sortie du capteur.

- Moteurs Polulu 993 :
 - Réduction 30 :1
 - Plage d'alimentation : 3-7 V
 - Courant à vide : 40 mA
 - Courant en charge : 360 mA

Étant donné que tous les composants de Polulu ont besoin d'être alimentés le seront en 5V, il suffit de mettre une unique puce de conversion de tension (7805). Pour stabiliser la tension, et donc le signal en sortie de certains composants (l'alimentation et le L298 en particulier) nous avons ajouté des condensateurs sur la patte à stabiliser. En effet, il peut y avoir des interférences sur les pistes, qui peuvent brouiller le signal et à terme endommager les différents composants.

Fonctionnement du L298

Le circuit intégré L298 contient deux étages de puissance configurés en pont, chacun commandé par deux entrées logiques (A, B et C, D) ainsi que deux entrées de validation (INH1 et INH2). De plus deux broches sont connectées en interne aux émetteurs des transistors (paires inférieures) qui permettent la connexion de résistances palpeuses de courant. Le schéma interne du L298 est donné en figure 2.1.

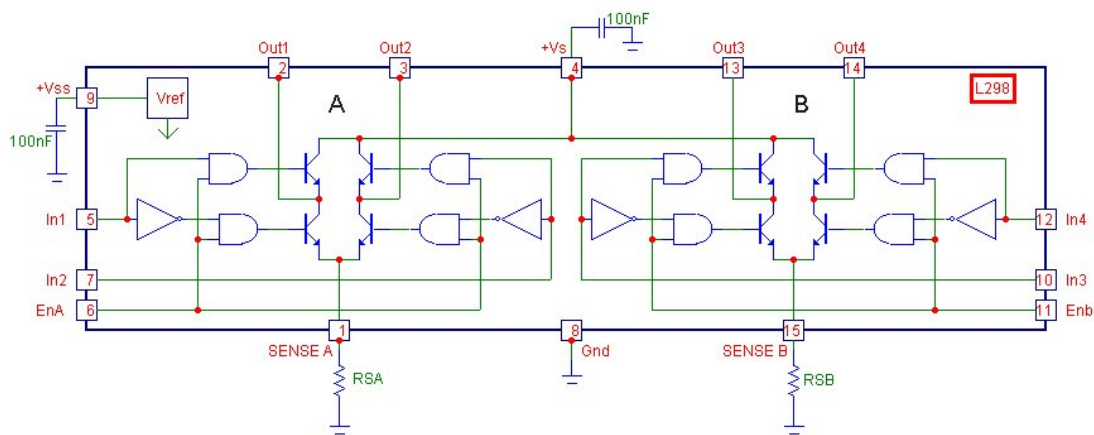


FIGURE 2.1 – Schéma interne du L298

Celui-ci permet de disposer d'un courant important (2,5A) sous une tension élevée (46V). la puissance obtenue peut ainsi atteindre environ 200W, ce qui permet l'alimentation de moteurs puissants présentant des résistances de bobinage faibles. Il va sans dire que dans ce cas, le circuit devra être fixé sur un dissipateur thermique de dimensions convenables.

Comme on le constate sur le schéma, le L298 possède deux broches d'alimentation : l'une pour le moteur (Vs) et l'autre pour la logique interne (Vss, +5V) ce qui permet de limiter la dissipation du boîtier.

Il est à noter que, contrairement à la majorité des circuits, le L298 ne contient pas les diodes de protections des transistors internes. Il conviendra donc de les câbler à l'extérieur du boîtier. Ces diodes devront présenter des temps de commutation faible et devront être capable de laisser passer un courant important.

Le L298 est présenté en boîtier MULTIWATT à 15 broches, ce qui simplifie sa fixation sur un refroidisseur.

Fonctionnement du régulateur

Le 78xx (parfois noté LM78xx) est une famille de régulateurs de tension linéaire. Les 78xx sont couramment utilisés dans les circuits électroniques nécessitant une alimentation régulée en raison de leur facilité d'utilisation et de leur faible coût. Pour ce type de régulateur, le

xx est remplacé par deux chiffres, indiquant la tension de sortie (par exemple, le 7805 dispose d'une sortie de 5 volts, tandis que le 7812 dispose d'une sortie de 12 volts). Les 78xx sont des régulateurs de tension positive : ils produisent une tension qui est positive par rapport à un terrain d'entente. À l'inverse, les régulateurs 79xx permettent de générer une tension négative. 78xx et 79xx peuvent être combinés pour fournir des tensions d'alimentation positive et négative dans le même circuit.

Les 78xx fonctionnent avec une tension allant de quelques volts au dessus de la tension de sortie, jusqu'à 35 ou 40 volts au dessus de celle-ci et peuvent fournir jusqu'à 1 ou 1,5 ampère en courant.

Caractéristiques :

- Courant maximum en sortie : 1,5 A
- Tension en sortie : 5, 6, 8, 9, 10, 12, 15, 18 ou 24 V, suivant le modèle.
- Le régulateur est équipé d'une protection thermique ainsi que d'une protection contre les courts-circuits.

Avantages : Les 78xx ne nécessitent pas de composants supplémentaires pour fournir une tension constante et régulée, ce qui les rend facile à utiliser, économiques, et peu gourmands en place.

Les 78xx ont intégré une protection contre les circuits demandant trop de puissance, une protection contre les courts-circuits, ainsi qu'une protection thermique, ce qui les rend suffisamment robustes pour la plupart des applications. Dans certains cas, les caractéristiques de limitation de courant des appareils 78xx peuvent offrir une protection non seulement pour le 78xx lui-même, mais aussi pour d'autres parties du circuit.

Inconvénients : La tension d'entrée doit toujours être supérieure d'au moins 2 volts à la tension de sortie. Cela peut rendre ces dispositifs inadaptés pour alimenter certains appareils avec certains types de sources d'énergie (par exemple, alimenter un circuit qui nécessite 5 volts à l'aide de 6 volts ne fonctionnera pas avec un 7805). Comme ils sont basés sur un modèle de régulation linéaire, le courant d'entrée nécessaire est toujours le même que le courant de sortie. Comme la tension d'entrée doit toujours être supérieure à la tension de sortie, cela signifie que la puissance totale (tension multipliée par le courant) qui entre dans le 78xx sera plus grande que la puissance de sortie. La puissance d'entrée supplémentaire sera donc dissipée sous forme de chaleur. Cela signifie à la fois que pour certaines applications un dissipateur adéquat doit être prévu, et aussi qu'une partie (souvent importante) de la puissance d'entrée est gaspillée au cours du processus, les rendant moins efficaces que d'autres types d'alimentations. Lorsque la tension d'entrée est nettement supérieure à la tension de sortie régulée (par exemple, alimenter un 7805 en utilisant une source d'alimentation 24 volts), cette inefficacité peut être une question importante.

2.2.2 Création du circuit à l'aide d'un logiciel de CAO (kicad)

Pour notre projet, nous avons eu besoin de réaliser deux circuits imprimés. En effet, nous avons besoin d'un circuit principal sur lequel nous avons mis l'alimentation, la commande des moteurs ainsi que le PIC entre autres. Et un circuit secondaire qui est en fait qu'un circuit avec les trois capteurs qui nous permettent de suivre la ligne. Ce deuxième circuit est nécessaire puisque les capteurs doivent être placé le plus près possible du sol, il est donc bien plus facile de les placer correctement si ils sont sur un circuit séparé.

Ainsi pour la réalisation de ces deux circuits, nous avons eu besoin de choisir un logiciel de CAO. Nous avons choisi de réaliser ces circuits à l'aide du logiciel Kicad, qui est un

logiciel très complet pour la réalisation de circuit imprimé. De plus il s'agit d'un logiciel open source et multiplateforme installé sur les machines de l'INSA, cela nous a donc permis d'avoir accès à ce logiciel aussi bien à l'INSA que chez nous et donc d'avancer plus vite et facilement la conception du circuit imprimé.

La conception d'un circuit imprimé avec kicad se découpe en quatre parties que je vais vous exposer. Il a fallu d'abord dessiner le schéma de nos circuit, cette étape a été assez rapide puisque nous avons déjà réalisé les schémas sur papier, néanmoins nous avons rencontré quelques problèmes de prise en main du logiciel. En effet bien que très complet, ce logiciel n'est pas toujours très intuitif et ergonomique, il nécessite donc un certain temps d'adaptation. L'autre problème que nous avons rencontré se situe au niveau des bibliothèques de composants, celles par défaut de kicad ne comprenait pas le PIC que nous avons choisi, et quelques autres composants. Heureusement nous avons réussi à trouver les bibliothèques kicad avec tout les composants dont nous avons besoin sur le site : http://per.launay.free.fr/kicad/kicad_php/composant.php. Une fois l'étape de prise en main passée et les bibliothèques de composants trouvées, nous avons put passer à l'étape suivante.

Après la réalisation du schéma électrique, il faut associer chaque composant que nous avons utilisé dans le schéma à une «footprint» qui signifie empreinte de pas. Plus concrètement nous avons du associer chaque composant du schéma au composant physique qui lui correspondait, cela permet de fixer par exemple la bonne distance entre les différentes pattes d'un même composant, ainsi que sa taille dans l'espace, pour que tous ces paramètres soient pris en compte dans le circuit final.

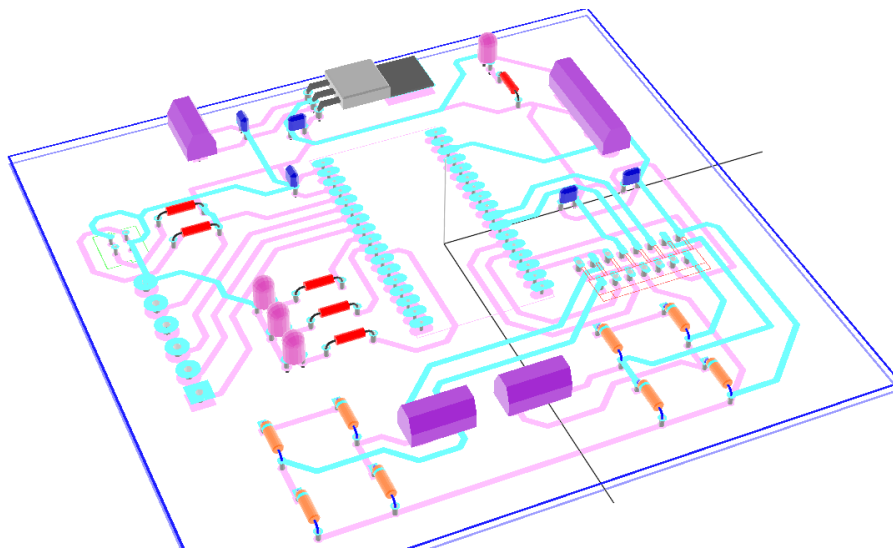


FIGURE 2.2 – Rendu de la carte à l'état final

Une fois que chaque composant est associé à la bonne «footprint». On peut passer à l'étape de réalisation du circuit à proprement parler. Dans cette étape, il s'agit de définir les contours de notre circuit, ainsi que de placer les composants dessus et d'en dessiner les pistes. Kicad a des fonctions qui permettent de faire ça automatiquement, mais nous l'avons tout de même fait manuellement car les circuits réalisés automatiquement étaient très compact et parfois même aberrants. Nous avons tout de même utilisé ces fonctions pour réaliser les pistes sur certains modules compliqués comme le L298, tout en rectifiant légèrement les tracés de kicad sur certains points, pour éviter certains tracés aberrants, comme par exemple une piste qui passe entre les deux pattes d'une LED, ou encore l'usage intensifs de via, pour passer d'une face à l'autre du circuit. Arrivé à ce stade, de la réalisation du circuit,

notre professeur nous a fait remarqué que la taille des pistes par défaut sur kicad était très petite. Nous avons donc dû élargir les pistes, cela n'a pas été très compliqué pour la plus grande partie du circuit. Malheureusement, dans le routage des pistes autour du L298, nous faisons passer certaines pistes entre les pattes du composant, nous avons donc du refaire entièrement le routage de ce module. Cela a pris un certain temps compte tenu du fait que kicad n'arrivait pas à faire le routage automatique du composant avec la nouvelle taille des pistes, nous avons donc été obligé de le faire manuellement, ce qui a d'ailleurs demandé de modifier une bonne partie du circuit, en effet le L298 est relié à un bon nombre de composant. Mais une fois cette difficulté passée, notre circuit était désormais prêt à être réalisé.

On en arrive donc à la dernière étape qui consiste à imprimer chacun des deux couches de cuivre sur une feuille de papier calque. Pour la réalisation physique de la carte, notre professeur s'est chargé de nous la faire réaliser à partir des fichiers et impressions que nous lui avons fourni. Nous avons donc pu passer à l'étape de soudure des composant sur la carte, dans le but d'obtenir un résultat tel que celui présenté dans l'image ci-dessous :

2.2.3 Problèmes rencontrés

Lors de la soudure du circuits, nous nous somme heurté à quelques erreurs de conception :

- Problème de réception des signaux de la carte capteurs par le PICbasic :
Lors du premier essai du robot, nous avons diagnostiqué un problème avec la carte capteurs : quel que soit la surface devant laquelle les capteurs étaient positionnés (surface blanche ou noire, ou même rien), le robot ne changeait pas d'état, les DEL restaient dans le même état (soit la rouge, soit l'orange allumée, en fonction de la version du programme chargée sur le PIC). Suite au test des capteurs, nous avons conclu à une erreur de conception. En effet nous étions partis du principe que lorsque le capteur était devant une surface noire ou devant aucune surface, sa tension de sortie était proche de 0V et lorsqu'il était devant une surface blanche sa tension de sortie était proche de 5V. Mais cela ne s'est pas vérifié par la pratique : nous avons mesuré une tension aux alentours de 4,8V lorsque le capteur était devant une surface noire ou n'ayant rien devant lui et environ 2,4V devant une surface blanche. Or les capteurs étaient reliés aux ports avec niveau d'entrée TTL du PIC, ce qui correspond à une tension entre 1,3V et 1,4V pour le seuil entre les niveaux haut et bas du PIC. Le signal était donc toujours haut, quel que soit l'état des capteurs. Trois solutions ont alors été envisagées :
 - Relier les capteurs à d'autres pattes du PIC, présentant une entrée en Trigger de Schmidt, avec une valeur de seuil entre 3,3V et 3,4V. Cela aurait sûrement fonctionné au vu des valeurs extrêmes observées lors du test des capteurs. La réalisation n'aurait pas été trop complexe, car il aurait simplement fallu souder trois fils entre différentes pattes du PIC.
 - Utiliser la fonction convertisseur Analogique/Numérique des pattes auxquelles sont reliés les capteurs, ce qui permettrait de fixer une valeur de seuil à la main. Cependant son implémentation algorithmique nous est parue trop complexe pour être réalisable simplement.
 - Remplacer certaines résistances de la carte capteurs pour obtenir une valeur de seuil bas plus faible que celle du TTL.

Nous avons retenu cette dernière solution.

Lors de la dernière séance, nous avons testé les capteurs pour vérifier, alors que nous effectuions les derniers tests pour voir si le robot marchait. Nous avons alors mesuré une tension d'environ 0,1V pour un capteur sur une surface blanche et d'environ 4V

sur une surface noire.

– Oubli d'alimenter la patte « reset » :

La patte reset du PIC doit être alimentée en +5V lors de son fonctionnement. Or nous n'avions pas pensé à cela lors du design de la carte. Cependant, par chance, la patte d'alimentation et la patte reset du PIC sont placées côte à côte sur la puce. Un point de soudure un peu épais a permis de relier les deux pattes, résolvant le problème de manière assez simple.

– Hauteur de la carte capteurs :

Pour que les CNY70 de la carte capteurs ne viennent pas reposer sur le sol, nous avons vérifié les dimensions de chaque composant pour que la hauteur totale de la carte ne soit pas excessive. Nous nous sommes rendus compte lors de la soudure qu'il était plus judicieux de souder les composants (les résistances en particulier) sur le dessous de la carte, pour permettre un réglage plus précis de la hauteur des vis de fixation.

– Vitesse du robot :

Après les premiers tests, nous avons rapidement compris que la vitesse de déplacement du robot était trop élevée. En effet, nous avons remarqué une petite latence entre le changement de surface d'un capteur et la modification de trajectoire du robot. À cela s'ajoutait l'inertie du robot, ainsi que le manque d'adhérence sur certaines surfaces des roues. Nous avons alors cherché un moyen de diminuer la vitesse de rotation des roues. La solution que nous avons retenu est de placer des résistances sur les fils entre le circuit imprimé et les moteurs. Cela nous a permis de tester différentes valeurs de résistance, pour avoir une vitesse convenable et en évitant que les moteurs ne se bloquent. En effet en rajoutant des résistances sur le circuit d'alimentation des moteurs, on diminue non seulement la vitesse de rotation mais aussi le couple déployé par ceux-ci.

– Surfaces pour poser le robot :

Après de nombreux essais sur différentes surfaces, nous avons trouvé les matériaux idéaux pour faire fonctionner Polulu : le blanc un matériau brillant (nous avons fait nos premiers essais avec du scotch) et pour le noir un matériau mat (de la mousse ou du caoutchouc par exemple), pour respectivement refléter et absorber les rayonnements de la photodiode.

– Affichage :

Lors du premier essai du robot, nous avons remarqué que rien ne s'affichait sur l'écran. Après quelques vérifications, nous avons compris que le problème venait d'une soudure oubliée isolant toute une partie du circuit. Mais même après rectification de ce détail, rien ne s'affichait à l'écran, pas même le curseur. Nous avons alors entrepris de vérifier, plusieurs fois, les branchements et routages (l'écran doit être alimenté en 5V, relié à la patte Picbus du PIC et à la masse). Lors de la dernière séance, nous avons fait des tests encore plus approfondis et avons fini par trouver le défaut : dans la gaine reliant les câbles de l'écran au circuit, le câble de la masse est sectionné, ouvrant ainsi le circuit de l'écran. Ce problème devrait être résolu avant la présentation orale.

Conclusion et perspectives

3.1 Ressentis dans le groupe

Jean-Baptiste : Ce projet m'a apporté, sur de nombreux plans, des atouts certains pour un futur ingénieur. En plus d'augmenter mes connaissances dans différents domaines scientifiques, j'ai appris à établir un planning de travail, gérer un groupe et prendre des décisions pour trouver des solutions en ayant très peu de temps. C'est, selon moi, l'enseignement le plus proche de la vraie vie d'ingénieur que l'on ait eu jusqu'à maintenant. Pendant toute la durée de ce projet, j'ai compris qu'être ingénieur ne signifie pas seulement réunir des compétences sur le plan scientifique, mais aussi maîtriser l'aspect humain du travail en groupe et savoir rapidement prendre des décisions, sans forcément avoir toutes les cartes en main, quand la situation l'impose. Je remercie chaleureusement toutes les personnes ayant contribué à la réalisation de ce projet et plus particulièrement Monsieur Messedi pour son soutien et ses conseils à chaque étape de la conception de Polulu.

Adrien : Ce projet a été le premier choix vers lequel je me suis porté en raison de son lien avec le domaine de l'informatique et de l'électronique. Le fait qu'il soit essentiellement tourné vers l'expérimentation a aussi été un facteur important.

Malgré mes appréhensions concernant le travail de groupe, dues aux précédents projets, la réalisation de ce robot s'est merveilleusement bien déroulée, notamment grâce au travail fourni par chacun des membres du groupe.

En outre, ce projet m'a permis de fortifier mes connaissances en électronique en me donnant l'occasion de travailler sur une expérience concrète. Il m'a de plus donné l'opportunité d'apprendre à coder dans un langage de bas niveau, le Basic.

Nicolas : D'un point de vue personnel, ce projet de physique a été très intéressant. En effet, ce projet m'a permis de découvrir la réalisation de circuits imprimés notamment la création du circuit par ordinateur à l'aide d'un logiciel, ici kicad. Je me suis principalement chargé de la réalisation du circuit par ordinateur et j'ai aussi participé à la réalisation technique du robot. C'est à dire la soudure des composants sur la carte, ainsi que le positionnement du circuit sur le châssis. Il m'a aussi permis de réapprendre des bases d'électricité et d'électronique dans un cadre moins théorique. Ainsi que les contraintes liées à la réalisation technique du robot, celles liées aux soudures, ainsi que celles liées à la réalisation du circuit imprimé, par exemple la taille des pistes ou encore celle des pastilles. Nous avons eu pas mal de problèmes pour réaliser ce robot, mais nous avons plutôt bien réussi à les surmonter. Il nous a donc demandé un certain investissement personnel étant donné notre inexpérience dans ce domaine en particulier, il a donc été un bon moyen pour apprendre de nouvelles choses et acquérir de l'expérience. De plus, étant donné que je voudrais faire ASI l'année prochaine, cette expérience d'électronique cadre bien dans mon cursus. Du point de vue de l'organisation maintenant, ce projet m'a aussi permis d'acquérir une expérience supplémentaire dans le domaine du travail de groupe et de l'organisation d'un tel projet. Pour conclure, je

dirais donc que ce projet m'a beaucoup apporté, aussi bien sur le plan technique que sur le plan organisationnel.

Dong : Tout d'abord, je tiens à remercier tous les membres de l'équipe et le professeur. Pour être honnête, lors du premier cours, j'ai pensé que notre robot Pololu n'était pas très intéressant parce que je pensais que nous allions concevoir un robot humanoïde. De plus, les autres membres de l'équipe sont tous français, et mon français n'est pas très bon. Mais, ils se sont montrés passionnés et très patients. Ainsi ils m'ont expliqués chaque étape du projet et l'utilisation de certains équipements, la programmation, etc. J'ai appris beaucoup de choses. Je me suis occupée d'étudier les deux composants : L298 et 7805, mais pour moi, j'ai appris beaucoup plus. Comme son mouvement est basé sur le signal de rétroaction à la masse, le robot a besoin de l'informatique. Pour réaliser notre projet efficacement, nous nous sommes divisés en deux groupes. Pendant ces 4 mois, j'ai senti que le travail d'équipe était très important. Dans de nombreux cas, la force de l'équipe est très puissante. En fin, j'ai été très contente de travailler avec eux et je pense que cela restera un de mes meilleurs souvenirs à l'INSA.

Violaine : J'ai choisi de travailler sur ce projet car j'envisage de faire le département ASI et je suis très intéressée par l'informatique, Ce projet a été une merveilleuse opportunité pour moi d'appliquer mes connaissances en informatique et de parfaire ces connaissances, N'ayant appris que le langage Pascal en I2 en STPI1 et en I3 en STPI2 ,la réalisation du programme du robot m' a permis de m'initier au langage Basic.

La réalisation de ce robot m'a aussi permis de travailler en équipe, Le fait de travailler en équipe permet l'entraide et le partage mais peut aussi se révéler difficile en certaines occasions comme par exemple lors de la rédaction du cahier des charges de notre robot, Cette collaboration entre les membres du groupes m'a également donner de l'expérience du travail en équipe pour mes projets futurs.

Mélanie : L'EC projet physique est un EC assez atypique vis-à-vis des autres cours scientifiques. Le développement d'un robot m'a permis d'appliquer mes connaissances en programmation, acquises en STPI1 et STPI2, mais également d'enrichir nos connaissances en programmation par l'apprentissage d'un nouveau langage, le Basic. Les projets sont une part importante de la vie d'un ingénieur, c'est pourquoi il faut prendre contact dès maintenant, en cycle préparatoire.

Le travail de groupe est un bon moyen de travailler, mais il faut savoir parfois ne pas être trop exigeant envers les autres. Prendre en compte l'avis de chacun est primordial lorsque l'on travaille à plusieurs. L'organisation et la rigueur sont les deux qualités essentielles à avoir lorsque l'on veut mener à bien un projet.

3.2 Conclusion générale

Les principaux objectifs de l'EC Projet Physique ont été atteints : nous avons pris contact avec notre futur métier d'ingénieur au travers d'un réel projet expérimental. Le projet s'est bien déroulé, grâce à une bonne gestion du temps et des ressources et une bonne coordination et entente entre les membres. La réalisation d'un robot nous a permis d'enrichir nos connaissances en informatique, grâce à l'apprentissage d'un nouveau langage, le Basic, et en électronique grâce à la réalisation d'un circuit imprimé, entre autres. Nous sommes satisfaits de la réalisation de notre robot. Les exigences du cahier des charges ont été respectées. Nous sommes cependant bien conscients que ce robot pourrait être amélioré : par exemple, au niveau esthétique, rendre le design plus agréable, ou au niveau technique, lui faire éviter des obstacles ou le faire suivre un objet en mouvement, ... Mais cela relève de connaissances

que nous n'avons pas encore et le manque de temps ne nous permettait pas d'approfondir la question.

Nos premiers pas à tous dans le monde de la robotique se sont déroulés dans une très bonne ambiance de travail, à la fois sérieuse et détendue. La répartition des tâches a permis à chacun d'assumer ses responsabilités. Ce projet est le fruit d'un travail d'équipe durant les quatre derniers mois.

3.3 Pistes d'amélioration du robot

Nous espérons que ce projet sera à nouveau mis en place les prochaines années. En effet, nous avons beaucoup apprécié de réaliser un robot autonome de A à Z. Par manque de temps, nous n'avons pas pu développer toutes les idées que nous avons eu et qui nous tenaient à cœur. Nous pensons qu'elles pourraient constituer des pistes de recherches pour les prochains projet :

- détecteur d'obstacle avec un capteur à ultrasons.
- amélioration du châssis et de l'esthétique en général.
- développer un mode manuel pour placer le robot à l'aide d'une télécommande sur la ligne.
- développer un algorithme de résolution de labyrinthe.
- développer un deuxième robot traceur de ligne.
- envoyer la mesure de la distance parcourue par bluetooth.

Bibliographie

[1] *Fiche composant de chaque élément du circuit imprimé* : 7805, L298, CNY70, Pic, écran LCD.

Annexe A

Listings des programmes réalisés

A.1 Programme général

```
1  CONST DEVICE = 2H
2  CONST enable1 = 16
3  CONST enable2 = 17
4  CONST input1 = 21
5  CONST input2 = 20
6  CONST input3 = 19
7  CONST input4 = 18
8  CONST ledverte=4
9  CONST ledrouge=6
10 CONST ledorange=5
11
12 DIM distance AS INTEGER
13 DIM coeff AS INTEGER
14 DIM front AS BYTE
15 DIM compteur AS BYTE
16
17 CONST capteurmilieu=2
18 CONST capteurgauche = 3
19 CONST capteurdroite = 1
20 CONST capteurroue=0
21
22
23 OUT enable1,1
24 OUT enable2,1
25 OUT input1,0
26 OUT input2,1
27 OUT input3,0
28 OUT input4,1
29
30
31 SET PICBUS HIGH
32 LCDINIT
33 CLS
34 CSROFF
35 LOCATE 0,0
36
37 front=0
38 coeff=3
39 compteur=0
40 distance=0
41
42 debut:
```

```

43 IF IN (capteurmilieu) = 1 AND IN (capteurgauche) = 1 AND IN (capteurdroite) = 1
    THEN
44 REPEAT:
45 OUT input1,0
46 OUT input2,1
47 OUT input3,1
48 OUT input4,0
49 OUT ledverte,0
50 OUT ledorange,0
51 OUT ledrouge,1
52     IF IN (capteurmilieu) = 0 OR IN (capteurgauche) = 1 OR IN (capteurdroite
        ) = 1 THEN
53         GOTO REPEAT
54     END IF
55 OUT ledrouge,0
56
57 OUT input2,0
58 OUT input3,0
59 END IF
60
61
62 IF IN (capteurmilieu) = 1 THEN
63 OUT input2,1
64 OUT input4,1
65 OUT ledverte,1
66 OUT ledorange,0
67 ELSE
68 OUT ledverte,0
69 OUT ledorange,1
70     IF IN (capteurgauche) = 1 THEN
71     OUT input2,1
72     OUT input4,0
73     ELSE
74         IF IN (capteurdroite) = 1 THEN
75         OUT input4,1
76         OUT input2,0
77         END IF
78     END IF
79 END IF
80
81 IF IN(capteurrouge)=1 THEN
82 front=1
83 END IF
84
85 IF IN(capteurrouge)=0 AND front=1 THEN
86 front=0
87 distance=distance+coeff
88 compteur=compteur+1
89 IF compteur > 2 THEN
90     compteur=0
91     distance=distance+1
92 END IF
93 END IF
94
95 PRINT distance
96 GOTO debut
    
```

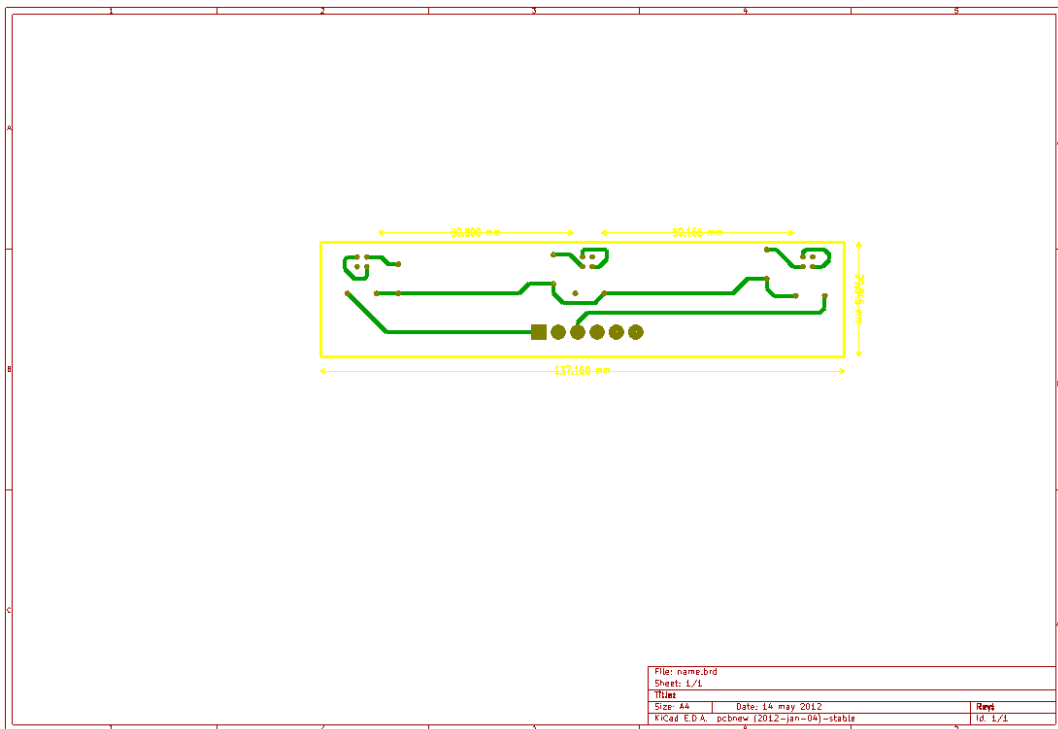
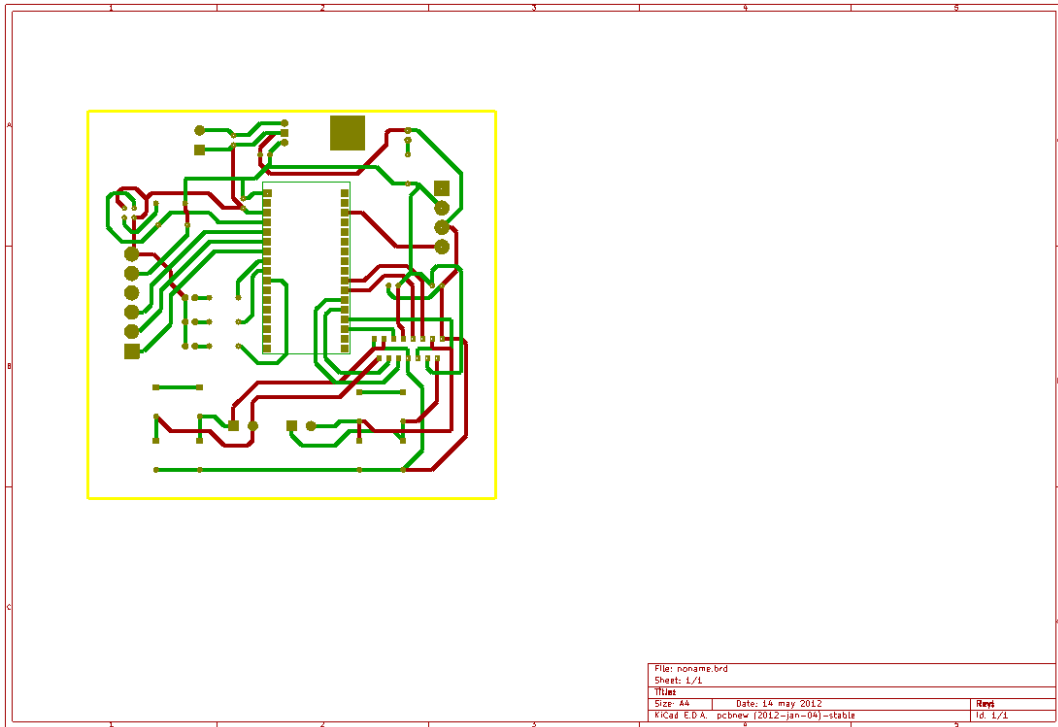



FIGURE B.2 – Représentation de la carte telle qu'elle sera imprimée

Annexe C

Propositions de sujets de projets (en lien ou pas avec le projet réalisé)

- développement d'un patator (chimique ou à explosion et à allumage électronique).
- développement d'un hélicoptère téléguidé.
- fabrication d'un moteur à explosion.
- expérimentation sur les feux d'artifice et autres explosifs.
- fabrication d'un mini-canon à poudre.
- conception d'une base de lancement de fusées à eau.
- fabrication d'une fronde géante et étude de la balistique (avec tir sur cible).
- étude de relevés sismiques en France.
- étude de l'assainissement des eaux.