

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد حطاب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Projet de Fin d'Études

présenté par

Benabdi fateh

&

Haoua yahia

pour l'obtention du diplôme de master en Électronique option système de vision et robotique

Thème

**Localisation d'un robot mobile par l'odométrie avec
affichage sur PC**

Proposé par : boualem Kazed

Année Universitaire 2012-2013

ملخص

هذا المشروع يتضمن انجاز الروبوت المتحرك , وفقا لتعليمية أوروبوت 2013 , و لصعوبة تحديد موقع الـروبوت داخل طاولة اللعبة . قمنا بدراسة و تحديد إحداثيات المستوي (y,x) و زاوية التوجيه θ للروبوت المتحرك و ذلك بإستعمال طريقة قياس المسار .تنص هذه الطريقة على حساب المسافة المقطوعة من طرف العجلتين المستقلتين المركبة مع الـروبوت . اللتان تقومان بتدوير عجلتي الرامزة للرامز مما يؤدي الى إعطاء معلومات على شكل ذبذبات للمعالج. ثم عرضها في جهاز الكمبيوتر.

résumé

Ce projet comprend à réalisation du robot mobile, selon la règlement Eurobot 2013, pour la difficulté de localisation de robot à la table de jeu. on à étudier et déterminer les coordonnées cartésiennes du plan $(x;y)$ et l'angle θ d'un robot mobile par la méthode de odométrie , cette méthode consiste à calculer la distance parcourue par les roues libres monté sur le robot, cette distance est transformer en impulsions grâce à une roue codeuse montée dans un encodeur , ce dernier envoie ces information vers un calculateur qui traite ces information , ensuite affichée dans l'ordinateur.

Abstract

This project includes the completion of the mobile robot, according to the instruction of eurorobot 2013 ,due to the difficulty of locating robot inside game table .we have studied and determined the coordinates of the plane (x,y) and the steering angle θ of the mobile robot with using the method of measuring path .

This method provides of calculating the distance traveled by the two independent wheels vehicle with a robot which rotate the encoder wheels to the encoder so that gives informations in the form of vibrations to the processor ;then displayed in the computer .

Sommaire

Introduction générale	1
Chapitre 1 : généralités sur la robotique	
1.1. Introduction	2
1.2. L'histoire et l'évolution de la robotique	3
1.3. La Définition	7
1.4 Les robots holonomes :	8
<i>Degrés de liberté</i>	8
<i>Robot non-holonyme</i>	8
<i>Robot holonyme</i>	9
1.5 Fonctionnement d'un robot :	10
1.6. Conclusion	12
Chapitre 2 : La partie mécanique	
2.1. Introduction	13
2.2. Description mécanique du robot.....	13
2.2.1. La plate-forme	14
2.2.2 Système de locomotion	15
2.4.1. Les moteurs	15
2.4.2. Les roue	16
2.4.3. La roue-bille	17
2.2.3 Bouton d'arrêt d'urgence	18
2.3. Conclusion	18
Chapitre 3 : Conception et réalisation de la partie électronique	
3.1. Introduction	19
3.2. Schéma synoptique de la partie électronique	19

3.3. Source d'énergie (la batterie)	20
3.4. La carte de puissance	21
3.4.1. Le pont en H	22
3.4.2. Rôle du pont en H	23
3.4.3. Fonctionnement du pont en H	23
3.4.4. Le circuit intégré L298	24
3.5. La carte de commande (Arduino)	26
3.5.1. Alimentation	27
3.5.2. Mémoire	28
3.5.3. Entrées et sorties numériques	28
3.5.4. Broches analogiques	29
3.5.5. Programmation	30
3.5.6. La régulation de la vitesse	30
3.6. Protocole I2C	31
3.6.1. Historique	31
3.6.2. Caractéristiques	31
3.6.3. La prise contrôle du bus	32
3.6.4. La transmission d'un octet	32
3.6.5. La transmission d'une adresse	33
3.6.6. Lecture d'une donnée	33
3.7. Le circuit intégré PCF8574	34
3.7.1. Schéma de la carte i2c	37
3.7.2. Circuit imprimé I2C :	37
3.8. Le servomoteur	37
3.9. Les capteurs	39
3.9.1. Définition d'un capteur	39
3.9.2. Le Télémètres à infrarouge	39

3.9.3. Capteur fin de course	40
3.9.4. Principe de fonctionnement	41
3.10. Méthode de décodage	41
3.11. Les encodeurs	42
3.12. Conclusion	44

Chapitre 4 : L'odométrie

4.1 introduction	45
4.2. La Définition	45
4.3. Les odomètres (encodeurs)	46
4.4. Positionnement du robot	47
4.5. Approximation par segmentation de droites	48
4.6 Les erreurs de l'odométrie	50
4.7 La réduction de l'erreur de l'odométrie	50
4.8 conclusion	52

Chapitre 5 : Analyse et résultat

5.1 Introduction	53
5.2 Construction de robot	53
5.3 Calcule et l'envoi de distance parcourue par le robot	53
5.3 L'implémentation de l'algorithme de l'odométrie sur Matlab	54
5.4 Analyse du résultat	56
5.5 Conclusion	57

Conclusion général	58
---------------------------------	-----------

Annexe

Bibliographie

Listes des figures

Figure 1.1 : Le robot mobile « Séjourner » utilisé pour la mission pathfinder de la NASA.....	2
Figure 1.2: A gauche : Robot "Beast" de l'université John Hopkins dans les années 1960.	
A droite : Le robot Shakey de Stanford en 1969, il a été une plate-forme de démonstration des recherches en intelligence artificiel.....	5
Figure 1.3: Genghis, développé par Rodney Brooks au MIT au début des années 1990.....	6
Figure 1.4 : Les degrés de liberté dans un plan a deux dimensions.....	8
Figure 1.5 : Architecture d'un robot chariot.....	9
Figure 1.6 : Architecture d'un robot holonome.....	9
Figure 1.7 : Exemple d'une roue omnidirectionnelle.....	10
Figure 1.8 : Exemple de schéma d'un robot avec des capteurs reliés à un cerveau qui envoie des commandes à des moteurs au travers d'une carte de contrôle des moteurs.....	11
Figure 2.1: Le robot mobile.....	14
Figure 2.2 : La plateforme du robot mobile.....	15
Figure 2.3 : Moteur MFA940D.....	16
Figure 2.4: La dimension du moteur 940D.....	16
Figure 2.5: Les roues du robot mobile.....	17
Figure 2.6: Roue bille en avant pour la stabilité de robot mobile.....	17
Figure 2.7 : Bouton d'arrêt d'urgence avec indication du sens de rotation pour son déverrouillage après actionnement.....	18
Figure 3.1 : Schémas synoptique de la partie électronique.....	20
Figure 3.2: Batterie pour l'alimentation 12 volt.....	21
Figure 3.3: Schémas développé de la carte de puissance avec le circuit L298.....	21
Figure 3.3.1 : Le circuit imprimé de la carte puissance.....	22
Figure 3.3.2 : La carte de puissance.....	22
Figure 3.4.1 : Schéma d'un pont en H.....	23
Figure 3.4.2: Le moteur tourne à droite.....	24
Figure 3.4.3 : Le moteur tourne à gauche.....	24
Figure 3.5 : Circuit intégré L298.....	25
Figure 3.5.1: schémas interne du circuit L298.....	25
Figure 3.6 : La carte Arduino.....	26
Figure 3.7 : Le Microcontrôleur ATMEGA 328p de la carte Arduino.....	26
Figure 3.8: Chronograms (Pulse Width Modulation).....	31
Figure 3.9 : Exemple de condition de départ et d'arrêt.....	32
Figure 3.9.1: Exemple de transmission réussie.....	33

Figure 3.9.2 : Exemple d'octet d'adresse.....	33
Figure 3.9.3 : Exemple de lecture d'une donnée.....	34
Figure 3.9.4: Schéma interne du PCF8574.....	34
Figure 3.9.5 : circuit pcf8574.....	35
Figure 3.10 : Schémas électrique de circuit I2C.....	37
Figure 3.10.1 : Circuit imprimé d'I2c	37
Figure 3.11 : Servomoteur (futaba s3003).....	38
Figure 3.12 : Chronogramme servomoteur par la durée d'implosion.....	38
Figure 3.13: Principe du télémètre infrarouge à triangulation et exemple de télémètre réel.....	40
Figure 3.14: Les Interrupteurs de fin de course.....	41
Figure 3.15: Schémas électrique de la bascule D.....	42
Figure 3.15.1: Schémas du circuit imprimé de la bascule D.....	42
Figure 3.16 : Les encodeurs du robot.....	43
Figure 3.17 : La forme du signal délivré par les sorties A, B, I	43
Figure 3.18 : Structure interne de l'encodeur.....	44
Figure4.1 : Approximation par des segmentations de droites.....	48
Figure 4.2 : Approximation par des arcs de cercles.....	50
Figure 5.1 : l'organigramme du calcule de la distance dan le microcontrôleur.....	54
Figure 5.2 : l'organigramme de calcule des cordonnées.....	55
Figure 5.3 : Une tour a gauche de robot.....	56
Figure 5.4 : Une tour a droite de robot.....	56

Introduction général

L'odométrie est une technique très utilisable pour estimer la position d'un robot en mouvement, cette position est mesurée à travers des capteurs spécialisés (les encodeurs).

Afin d'assurer la mesure de position de robot, une plateforme d'un robot mobile à base de deux moteurs à courant continu a été réalisée. Cette plateforme est équipée de deux encodeurs pour mesurer la position de robot sur une table de Eurobot pour faire cette mesure, un microcontrôleur (Arduino) est chargé de calculer la distance parcourue par les deux roues à travers un circuit I2C qui sert à augmenter le nombre d'entrées/sorties numériques dans le microcontrôleur ;

Après le calcul de la distance, on va envoyer les données via une porte série vers le Matlab pour calculer les coordonnées de ce robot.

Chapitre 1: Généralités sur la robotique

1.1 Introduction:

Depuis 1970 le monde scientifique et industriel porte des efforts sans cesse croissants sur l'étude et la réalisation d'engins mobiles. Ces derniers sont capables d'effectuer des déplacements sans aucune liaison mécanique avec un point fixe de l'espace dans lequel ils évoluent d'où le nom de robots mobiles. Nous pouvons considérer que « robot mobile » est synonyme de « véhicule » puisque ce dernier est composé de la structure matérielle et mécanique qui permet au robot d'effectuer un déplacement. Cet intérêt nouveau porté à la robotique mobile s'explique par les applications qu'elle peut apporter dans de nombreux domaines. Comme les robots manipulateurs les robots mobiles sont destinés à remplacer ou à soulager l'homme des tâches fatigantes dangereuses inaccessibles ou encore vitales.

Les robots mobiles sont largement utilisés dans les environnements industriels pour le transport de produits par exemple. Le plus souvent ces tâches sont répétitives et suivent un chemin bien défini, parfois même bien matérialisé comme des lignes sur le sol ou des amers artificiels.



Figure 1.1 : Le robot mobile « Séjourner » utilisé pour la mission pathfinder de la NASA

Actuellement, il y a un intérêt croissant vers les systèmes multi-robots mobiles en raison de leur utilisation aux diverses tâches telles que les missions spatiales, les opérations dans des environnements dangereux, et les applications militaires.

Nous donnons dans ce chapitre un aperçu sur les notions de base de la robotique mobile. Certains points seront mentionnés plus en détail que d'autres, car ils sont plus étroitement liés à notre travail de recherche.

La robotique est la branche de l'intelligence artificielle concernée par l'étude de systèmes automatiques capables d'une interaction directe avec le monde physique. On distingue deux groupes de robots : les robots fixes et les robots mobiles. Les robots fixes sont utilisés dans l'industrie pour réaliser des tâches dangereuses (soudure du châssis ou peinture de la carrosserie dans une usine automobile etc.), les robots mobiles pour transporter des charges (depuis les chaînes de fabrication jusqu'aux zones de stockage) ou encore pour transporter le courrier dans les bureaux, ou pour intervenir dans des milieux hostiles. [1]

1.2 L'histoire et l'évolution de la robotique:

La notion de robot, ou d'automate, remonte à l'époque médiévale. Même s'il n'existait pas de terme pour décrire ce que nous appelons aujourd'hui des robots, les gens de cette époque ont tout de même imaginé des mécanismes capables d'exécuter des tâches humaines.

Les automates, machines aux formes humaines dirigées par un mécanisme intérieur, ont été créés pour impressionner les paysans qui fréquentaient les églises. Le jacquemart, créé l'illusion du mouvement automatique. Les jacquemarts étaient des personnages mécaniques qui étaient capable de marquer le temps en frappant sur une cloche avec leur hache.

Au XVIIIe siècle, les automates miniatures sont devenus des jouets populaires chez les gens riches. Ces automates étaient à l'image des humains et en 1818, Mary Shelly a écrit Frankenstein, un récit qui relate la fabrication d'une créature d'apparence humaine. Le robot imaginé par cette auteure ressemblait à un homme, mais fonctionnait comme une machine. Il était composé d'éléments aux formes humaines maintenus ensemble à l'aide de boulons et d'écrous. On remarquera qu'il y a même des pinces pour retenir ensemble les parties de sa tête. Selon Shelly, un robot devait être plus costaud que la moyenne d'entre nous et être doté d'une force surhumaine.

En 1921, Karel Capek, un dramaturge tchèque, a imaginé un personnage créé artificiellement et doté d'intelligence, auquel il a donné le nom de « robota ». Ce mot, qui signifie « travail forcé » en tchèque, est graduellement entré tel quel dans la langue française. Même s'il a été créé cent ans après la publication du roman Frankenstein de Shelly, le robot de Capek a lui aussi des formes humaines. Ainsi, si le concept de robot remonte à l'époque médiévale, il a fallu attendre les années 1940 et l'arrivée des ordinateurs pour que les robots des temps modernes fassent leur apparition.

Le terme « robotique » désigne l'étude et l'utilisation des robots. Ce terme a été utilisé pour la 1ère fois en 1941, par le chercheur et écrivain Isaac Asimov dans un de ces livres, appelé "Cycle fermé".

Isaac Asimov a écrit les 3 lois fondamentales de la robotique. Celles-ci permettent d'éviter que cette science amène l'apocalypse avec elle :

- ✓ **1ère loi** : *Un robot ne peut porter atteinte à un être humain ni, en restant passif, laisser cet être humain exposé au danger.*
- ✓ **2ième loi** : *Un robot doit obéir aux ordres donnés par les êtres humains, sauf si de tels ordres sont en contradiction avec la première loi.*
- ✓ **3ième loi** : *Un robot doit protéger son existence dans la mesure où cette protection n'est pas en contradiction avec la première ou la deuxième loi. [2]*

Le concept du robot s'est étendu à la fin des années 1950 et au début des années 1960. C'est l'industrie automobile, en pleine expansion, qui allie les robots industriels aux travailleurs d'usines. Ces robots sont des mains ou des bras commandés par un ordinateur, qui sont capables de réaliser une tâche sur une chaîne de montage (soudage, pulvérisation de peinture etc.) et ceci avec une très haute précision. [3]

Dans les années 1960, les recherches en électronique vont conduire, avec l'apparition du transistor, à des robots plus complexes mais qui vont réaliser des tâches similaires. Ainsi le robot "Beast" (**Figure 1.2**) de l'université John Hopkins est capable de se déplacer au centre des couloirs en utilisant des capteurs ultrason, de chercher des prises électriques (noires sur des murs blancs) en utilisant des photodiodes et de s'y recharger.



Figures 1.2: *A gauche : Robot "Beast" de l'université John Hopkins dans les années 1960.*

A droite : Le robot Shakey de Stanford en 1969, il a été une plate-forme de démonstration des recherches en intelligence artificielle.

Les premiers liens entre la recherche en intelligence artificielle et la robotique apparaissent à Stanford en 1969 avec Shakey (**Figure 1.2**). Ce robot utilise des télémètres à ultrason et une caméra et sert de plate-forme pour la recherche en intelligence artificielle, qui à l'époque travaille essentiellement sur des approches symboliques de la planification. La perception de l'environnement, qui est considérée comme un problème séparé, voire secondaire, se révèle particulièrement complexe et conduit là aussi à de fortes contraintes sur l'environnement. Ces développements se poursuivent avec le Stanford Cart dans les années 1980, avec notamment les premières utilisations de la stéréovision pour la détection d'obstacles et la modélisation de l'environnement.

Une étape importante est à signaler au début des années 1990 avec l'apparition de la robotique réactive, représentée notamment par Rodney Brooks. Cette nouvelle approche de la robotique, qui met la perception au centre de la problématique, a permis de passer de gros robots très lents à de petits robots (**Figure 1.3**), beaucoup plus réactifs et adaptés à leur environnement.

Ces robots n'utilisent pas ou peu de modélisation du monde, problématique qui s'est avérée être extrêmement complexe.

Ces développements ont continué depuis l'arrivée sur le marché au cours des années 1990, de plates-formes intégrées qui ont permis à de très nombreux laboratoires de travailler sur la robotique mobile et ont conduit à une explosion de la diversité des thèmes de recherche.



Figure 1.3: Genghis, développé par Rodney Brooks au MIT au début des années 1990.

Ainsi, même si les problèmes de déplacement dans l'espace restent difficiles et cruciaux, des laboratoires ont pu par exemple travailler sur des approches multi-robot, la problématique de l'apprentissage ou sur les problèmes d'interactions entre les hommes et les robots. [4]

La robotique mobile a connu un essor considérable. Les progrès techniques ont permis la construction de robots de plus en plus perfectionnés :

De nombreux périphériques, une puissance de calcul accrue et des moyens de communication performants. La mobilité autonome des robots est devenue un sujet de recherche développé par tous les pays industrialisés. Qu'il s'agisse de robots mobiles à pattes, à roues ou même sous-marins et aériens, les applications sont vastes et multiples: robots de services, surveillance, construction, nettoyage, manipulation de charges, automobile intelligente, robots d'intervention, robots d'exploration planétaire ou de fonds marins, satellites, robots militaires, etc.... Le marché potentiel de la robotique est considérable, même s'il faut pour cela résoudre des problèmes plus importants et plus fondamentaux que prévus initialement dans la quête vers la machine intelligente [5].

1.3 Définition :

Le Petit Larousse définit un robot comme étant un appareil automatique capable de manipuler des objets, ou d'exécuter des opérations selon un programme fixe ou modifiable.

Souvent un robot est défini comme un manipulateur automatique à cycles programmables.

Pour « mériter » le nom de robot, un système doit posséder une certaine flexibilité, caractérisée par les propriétés suivantes :

- **La versatilité** : (caractère versatile (changeant)) Un robot doit avoir la capacité de pouvoir exécuter une variété de tâches, ou la même tâche de différente manière ;

- **L'auto-adaptabilité** : Un robot doit pouvoir s'adapter à un environnement changeant au cours de l'exécution de ses tâches.

L'Association Française de Normalisation (**A.F.N.O.R**) définit un robot comme étant un système mécanique de type manipulateur commandé en position, reprogrammable, polyvalent (i.e., à usages multiples), à plusieurs degrés de liberté, capable de manipuler des matériaux, des pièces, des outils et des dispositifs spécialisés, au cours de mouvements variables et programmés pour l'exécution d'une variété de tâches. Il a souvent l'apparence d'un, ou plusieurs, bras se terminant par un poignet. Son unité de commande utilise, notamment, un dispositif de mémoire et éventuellement de perception et d'adaptation à l'environnement et aux circonstances. Ces machines polyvalentes sont généralement étudiées pour effectuer la même fonction de façon cyclique et peuvent être adaptées à d'autres fonctions sans modification permanente du matériel. [6]

Un robot mobile est un système mécanique, électronique et informatique agissant physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné. Cette machine est polyvalente et capable de s'adapter à certaines variations de ses conditions de fonctionnement. Elle est dotée de fonctions de perception, de décision et d'action. Ainsi, le robot devrait être capable d'effectuer des tâches diverses, de plusieurs manières, et accomplir correctement sa tâche, même s'il rencontre de nouvelles situations inattendues. [7]

1.4 Les robots holonomes :

Il existe enfin une catégorie de robots mobiles qui sont dit "holonomes". Mais avant de voir ce type de robot, il nous faut définir ce que sont les degrés de liberté d'une plate-forme mobile ainsi que les robots non holonomes.

a.1 Degrés de liberté :

Le nombre de degrés de liberté d'un robot mobile est défini comme le nombre de mouvements indépendants que ce robot peut faire par rapport à un système de coordonnées déterminé.

Tout mouvement effectué dans un plan peut être décomposé en un maximum de deux mouvements de translation purs effectués respectivement par rapport aux axes **X** et **Y**, plus un mouvement de rotation pur autour de l'axe perpendiculaire au plan.

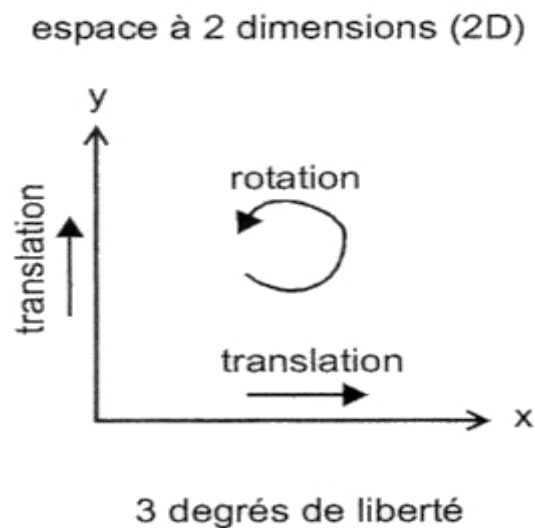


Figure 1.4 : Les degrés de liberté dans un plan à deux dimensions

L'espace à deux dimensions possède donc un maximum de 3 degrés de liberté.

a.2 Robot non-holonyme :

Une plate-forme mobile de type chariot ne dispose que de 2 degrés de liberté sur un plan puisque les translations latérales sont impossibles à réaliser :

- **Une translation** : avance ou recule.
- **Une rotation** : tourne vers la droite ou vers la gauche.

On dit donc qu'un robot de type chariot a une mobilité d'ordre 2.

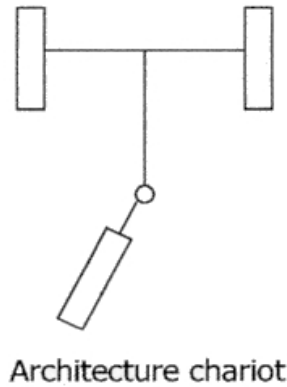


Figure 1.5 : Architecture d'un robot chariot

Il est donc considéré comme **non holonomes**.

a.3 Robot holonome :

En opposition aux robots à deux roues motrices différentielles qui possèdent 2 degrés de liberté sur un plan, un robot holonome ou robot omnidirectionnel est un robot qui possède 3 degrés de liberté sur un plan.

- **Deux translations** : avance ou recule, va vers la droite ou vers la gauche.
- **Une rotation** : tourne vers la droite ou vers la gauche.

Il existe plusieurs sortes de robot holonome :

- Les robots à pattes.
- Les robots à 3 ou 4 roues omnidirectionnelles.

S'il s'agit de faire un robot holonome à 3 roues omnidirectionnelles, il faut les disposer à 120 degrés les unes des autres.

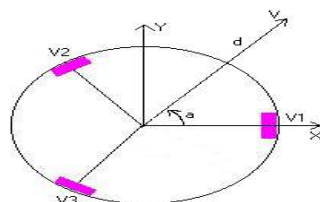


Figure 1.6 : Architecture d'un robot holonome



Figure 1.7 : Exemple d'une roue omnidirectionnelle

1.5 Fonctionnement d'un robot :

Un robot est une machine qui va évoluer en fonction de son environnement, soit de manière répétitive en effectuant une tâche programmée sans se tromper, soit de manière pseudo intelligente en évoluant dans un environnement aléatoire. Cela signifie qu'il va réagir en fonction de diverses contraintes qui lui seront soumises comme des obstacles, un objet, un trou, un changement de luminosité... Face à ces contraintes, le robot va devoir réagir, mais avant de réagir, il faut aussi qu'il reconnaisse la contrainte. Pour faire une approche plus précise, examinons la façon de fonctionner d'un être humain. L'être humain vit, se déplace et évolue dans un environnement. Il est capable de reconnaître de se déplacer sans tomber où se cogner, de communiquer,... Si l'homme est capable de répondre aux exigences que l'environnement lui soumet, c'est parce qu'il est capable de les sentir, de les voir... Exemple, l'être humain est capable de repérer un mur par la vision, il est capable de sentir une odeur de gaz par le nez, de sentir un sol instable par le toucher... Toutes les informations que l'homme va découvrir sur son environnement, vont permettre de le faire évoluer. Il saura qu'en voyant un mur il ne pourra pas passer, quand il sent une odeur de gaz, qu'il doit s'enfuir en courant... S'il est capable de réagir, c'est parce que les informations captées de l'environnement ont été analysées par le cerveau et celui-ci en a ressorti une solution. Cette solution a été mise en application à travers le corps: un déplacement avec les jambes, un son avec la bouche, un geste de la main...

Dans un travail répétitif, par exemple de remplir des conteneurs Ax avec des pièces B, l'homme perçoit l'arrivée d'un conteneur A1. Lorsque celui-ci est à sa portée, il sait qu'un mouvement de bras lui permettra de lâcher dans ce conteneur un nombre X de pièces B qu'il

pourra compter à la fois en les touchant et en les voyant. Quand le conteneur est rempli, l'homme sait qu'il peut le déplacer et attendre l'arrivée d'un conteneur A2. Et ainsi de suite.

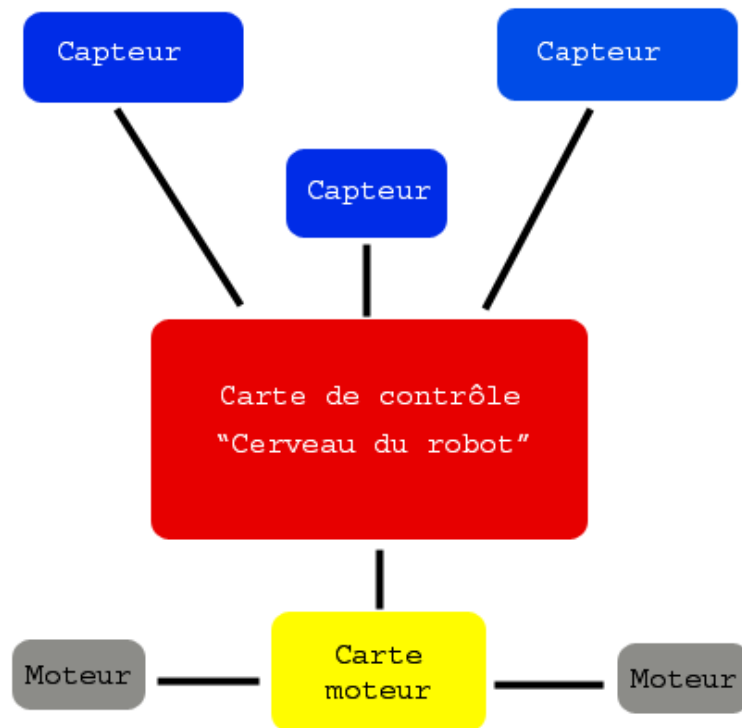


Figure 1.8 : Exemple de schéma d'un robot avec des capteurs reliés à un cerveau qui envoie des commandes à des moteurs au travers d'une carte de contrôle des moteurs

Donc, l'homme reçoit des informations par ses différents sens, ces informations sont analysées par le cerveau qui envoie la solution à appliquer aux muscles.

Pour un robot, c'est la même chose. La perception des sens est résolue par des capteurs. Ces capteurs peuvent être de tout type, du simple interrupteur à la caméra. Toutes les informations en provenance de ces capteurs sont envoyées à un « cerveau » qui peut-être constitué d'un montage électronique ou d'un microcontrôleur programmable (voire un PC). Ce cerveau, va analyser les informations reçues et va envoyer la réaction à prendre à un module permettant de contrôler les différentes actions qui agissent sur l'environnement, comme des moteurs, des bruits, des lumières... Le robot est dès lors capable de réagir, selon sa programmation et ses capacités de mouvement, à un environnement aléatoire ou d'exercer une tâche répétitive ; comme l'exemple des conteneurs ; sans mettre les pièces à côté ; car il reçoit en continu des informations provenant de l'environnement via les capteurs qui l'informent de l'état actuel. [8]

1.6 Conclusion :

Les robots prennent actuellement une place importante dans notre vie, on les trouve dans toutes les entreprises pour accélérer la production ou pour agir là où l'homme ne peut travailler à cause du danger, à la maison pour aider à la cuisine et aux tâches ménagères, à la guerre (un missile, ça peut-être considéré comme un robot), dans les services publics: hôpitaux, casernes de pompiers, la police, etc.....

La robotique est devenue en quelques années une science importante qui ne cesse d'évoluer. Les chercheurs parviennent petit à petit à donner à des machines une intelligence artificielle. Les robots envahissent littéralement notre vie, on les trouve dans des concours, dans les films de science-fiction et dans tous les autres domaines.

Dans ce chapitre nous a permis d'évoquer quelques généralités sur les robots mobiles. La connaissance de ces informations est importante pour réaliser ce projet.

chapitre 2: La partie mécanique

2.1 Introduction :

Pour concevoir, simuler ou commander un robot, il est nécessaire de disposer des modèles du mécanisme. Plusieurs niveaux de modélisation sont possibles. Ils dépendent des spécifications du cahier des charges de l'application envisagée : il en découle des modèles géométriques, cinématiques et dynamiques à partir desquels peuvent être générés les mouvements du robot, ou bien des modèles statiques qui décrivent les interactions du mécanisme avec son environnement.

Pour la construction du châssis il faut commencer par estimer le volume et la masse du robot pour en déterminer les dimensions et la puissance motrice nécessaire.

En effet, il est important de bien calibrer ses moteurs car, trop faible le robot ne sera pas en mesure de bouger et trop puissant il consommera plus d'énergie que prévu ce qui aura un impact direct sur l'autonomie des batteries et l'électronique de puissance.

Dans cette partie, Nous allons détailler la structure mécanique d'un robot mobile, qui représente la structure métallique et les actionneurs (le châssis, plate-forme, les roues ...).

2.2 Description mécanique du robot :

Le châssis du robot a une importance autant esthétique que fonctionnelle. L'étage inférieur comprendra souvent la propulsion (moteurs, réducteurs, batteries), puis viendra l'étage intermédiaire pour les cartes de périphériques : carte moteurs (la carte de puissance), la carte d'acquisition des capteurs, et la carte des encodeurs).

Enfin, on terminera souvent par le cerveau du robot (le microcontrôleur) et éventuellement la télémétrie ou la caméra.

Bien sûr, selon l'application, le robot pourra avoir une forme toute autre pour s'adapter à sa mission.

Le matériau constituant le châssis devra être choisi selon plusieurs critères. Il devra être :

- ✓ solide pour résister aux chocs avec son environnement,
- ✓ léger pour utiliser de petits moteurs consommant peu,
- ✓ facile à travailler pour pouvoir ajouter un perçage ou un orifice
- ✓ isolant (pour éviter les courts-circuits)

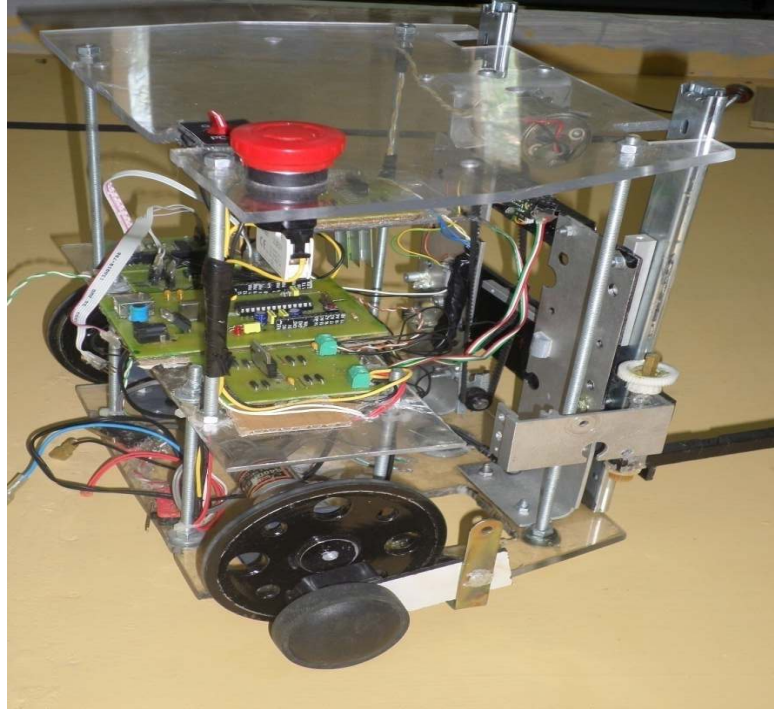


Figure 2.1: Le robot mobile

Le châssis devra être conçu de manière à pouvoir être démonté rapidement et devra permettre l'accès à tous les modules du robot. Dans ce but, les fixations devront être de préférence des vis ou des boulons.

Dans ce projet, le châssis est réalisé grâce à des plaques de plexiglas de 3 mm et 7 mm d'épaisseur. Ce matériau présente de nombreux avantages, La structure de base du robot est constituée de trois étages, reliées entre eux par des tiges filetées.

2.2.1 La plate-forme :

Cette plate-forme est très simple à commander, puisqu'il suffit de spécifier les vitesses des deux roues, cette possibilité permet de traiter le robot comme un robot non holonome, c'est à dire un robot pouvant se déplacer dans toutes les directions depuis sa position courante, cette structure simplifie énormément la planification de déplacement et la commande du robot.

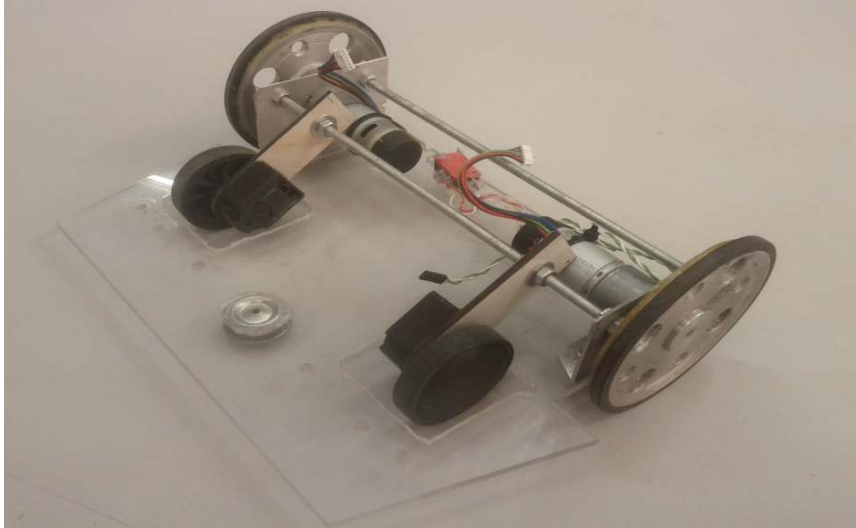


Figure 2.2 : La plateforme du robot mobile

2.2.2 Système de locomotion :

L'un des éléments essentiels pour garantir une bonne performance d'un robot mobile en extérieur est la structure de locomotion de la plate-forme utilisée.

Les systèmes de locomotion terrestre robotisés peuvent être de trois différents types principaux : à roues ou chenilles, à pattes (bien souvent sur le modèle d'animaux ou de l'humain) ou encore apodes (utilisation d'éléments de la structure du robot pour générer et appliquer au sol les efforts de propulsion).

a- Les moteurs :

Les moteurs utilisés de manière classique en robotique sont assez diversifiés. Les plus répandus sont les moteurs "**pas à pas**", les moteurs "**à courant continu**" et enfin "**les servos moteurs**" bien connus dans le milieu du modélisme.

On a utilisé dans ce robot mobile deux moteur MFA 940D, qui sont reliés avec les deux roues droite et gauche pour le déplacement du robot mobile.



Figure 2.3 : Moteur MFA940D

Les caractéristiques du Moteur :

- Model/REF : QUMFA940D1001
- Série 940D avec moteur 4,5 à 15V de la série RE385
- Tension nominale : 4,5V / 6V / 9V / 12V / 15V
- Courant de service : 0,99A
- Couple max : 30000 g.cm
- Ø axe sortie : 6 mm - Ø extérieur : 32 mm- Poids : 214 g

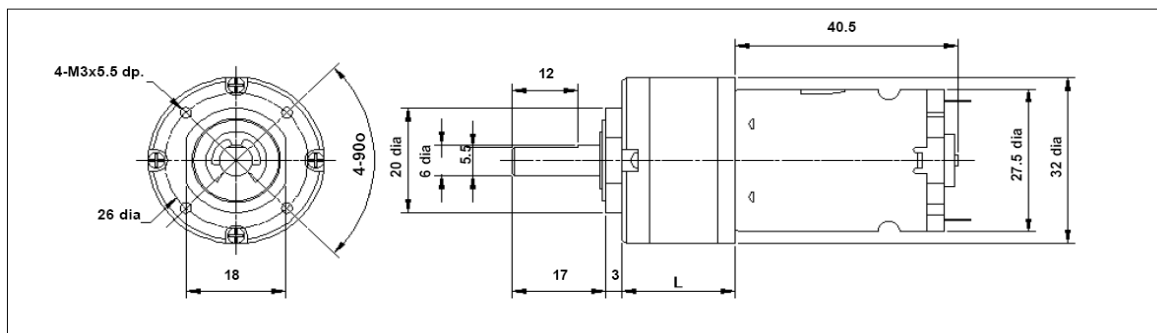


Figure 2.4:La dimension du moteur 940D

b- Les roues :

Les roues sont le moyen de locomotion le plus répandu en matière de robotique mobile. [9]

En fait, les robots mobiles à roues sont faciles à réaliser et présentent de grandes possibilités de déplacement et de manœuvrabilité avec une vitesse et une accélération importantes. Bien évidemment, pour un ensemble donné de roues, toute disposition ne conduit pas à une solution viable. Un mauvais choix peut limiter la mobilité du robot ou occasionner

d'éventuels blocages. Par exemple, un robot équipé de deux roues fixes non parallèles ne pourrait pas aller en ligne droite. Pour qu'une disposition de roues soit viable et n'entraîne pas de glissement des roues sur le sol, il faut qu'il existe pour toutes ces roues un unique point de vitesse nulle autour duquel tourne le robot de façon instantanée.



Figure2.5: Les roues du robot mobile

En général, les robots mobiles à roues ont des structures à locomotion différentielle. Le déplacement est assuré par le biais de deux moteurs d'entraînement indépendants, chacun d'eux est lié à une roue motrice. ^[10]

C- La roue-bille :

Le point d'appui des roues n'étant pas suffisant pour assurer la stabilité et l'équilibre de notre robot. Nous avons utilisé une roue-bille placée en avant.

En ce qui concerne la commande des moteurs, cette indépendance permet au robot mobile de changer de direction en jouant uniquement sur les vitesses des deux roues motrices. Les roues folles sont libres, leur rôle essentiel est d'assurer la stabilité de la plate-forme du robot.



Figure2.6: Roue bille en avant pour la stabilité de robot mobile

2.2.3 Bouton d'arrêt d'urgence :

Notre robot a un bouton d'arrêt d'urgence de 20mm de diamètre ; il est placé dans une position visible et dans une zone qui n'est pas dangereuse (sur le sommet du robot) et immédiatement accessible à tout moment. L'appui sur ce bouton provoquera l'arrêt immédiat des moteurs.



Figure2.7 : Bouton d'arrêt d'urgence avec indication du sens de rotation pour son déverrouillage après actionnement

2.3 Conclusion :

Dans ce chapitre nous nous sommes intéressés à la partie mécanique du robot mobile, nous avons présenté les différents pièces utilisés, leurs principes de fonctionnement ainsi que le rôle que joue chacun deux pour le bon fonctionnement du robot mobile.

Chapitre 3 : conception et réalisation de la partie électronique

3.1 Introduction :

La partie électronique du robot représente son système sanguin et son système cérébral. Le choix des cartes électroniques à fabriquer repose aussi sur les attentes du cahier de charge. Pour simplifier, la partie électronique est le lien entre la partie mécanique et la partie programmation.

3.2 Schéma synoptique de la partie électronique

Nous intéressons dans le synoptique suivant (schéma synoptique) de la partie électrique qui est constitué de :

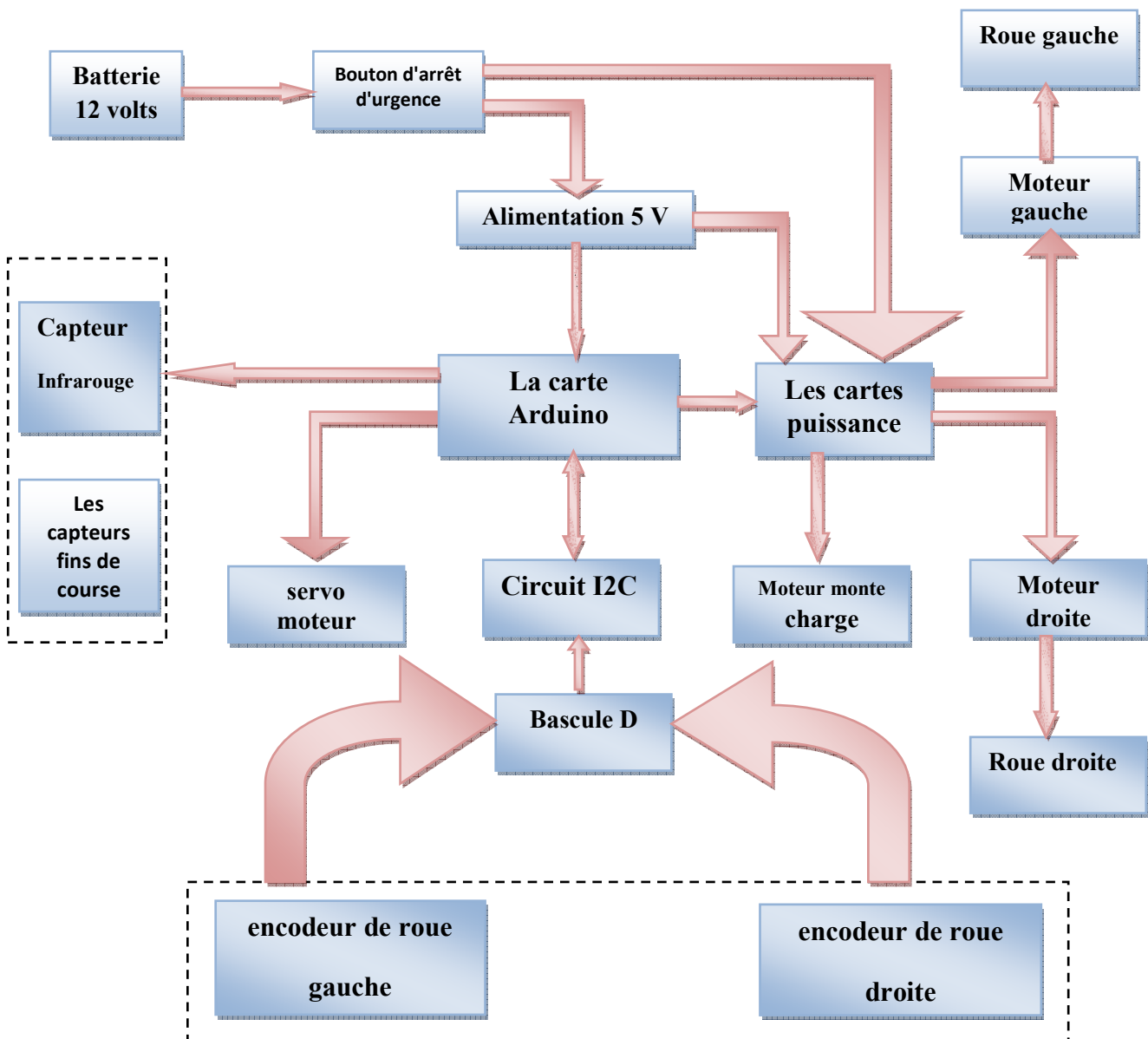


Figure 3.1 : Schémas synoptique de la partie électronique

3.3 Source d'énergie (la batterie):

Pour l'alimentation du robot mobile on a choisi une batterie de 12 volts, et de courant de 4.5 Ampère, placée à la base de la plate-forme, entre les deux moteurs droite et gauche.



Figure 3.2: Batterie pour l'alimentation 12 volt

3.4 La carte de puissance :

La carte de puissance joue le rôle d'une carte d'interface entre la carte de commande de la partie motrice comportant des composants mécaniques, elle nous permet de commander les moteurs à courant continu suivant les ordres fournis par la carte de commande.

Pour notre projet on a utilisé le circuit intégré L298 qui est le composant le plus utilisé pour la commande des moteurs à courant continu. Il permet en effet de commander n'importe quel type de moteur qu'il soit unipolaire ou bipolaire. La figure II représente la carte de puissance réalisée

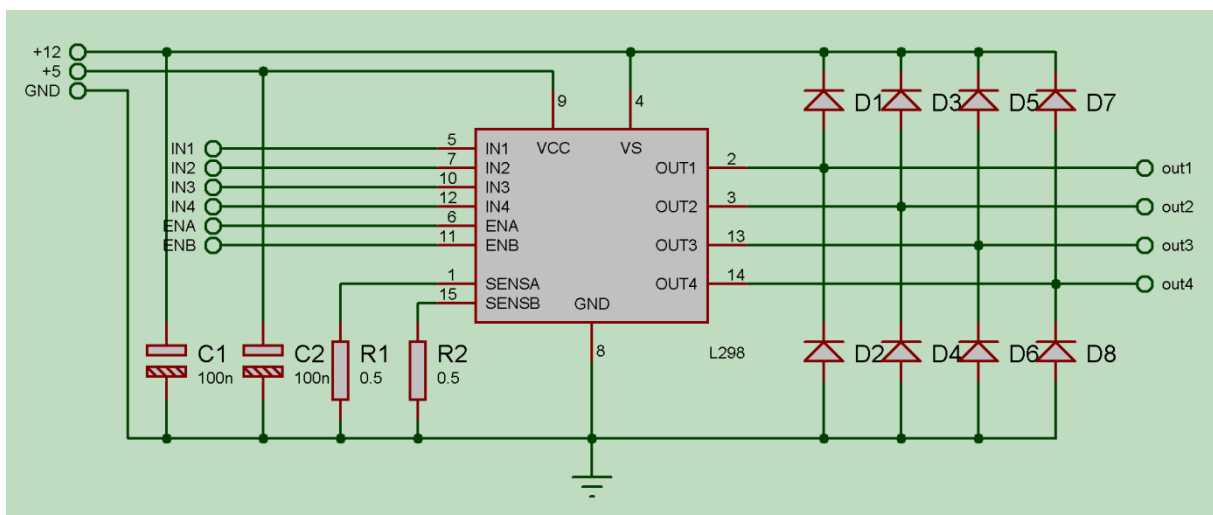


Figure 3.3: Schémas développé de la carte de puissance avec le circuit L298

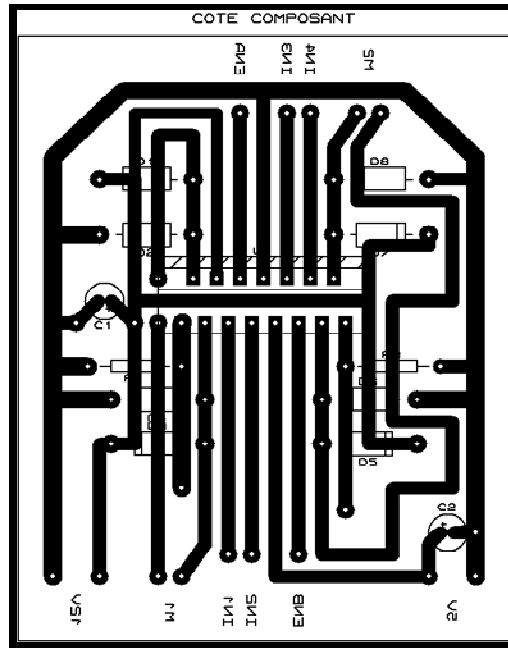


Figure 3.3.1 : Le circuit imprimé de la carte puissance

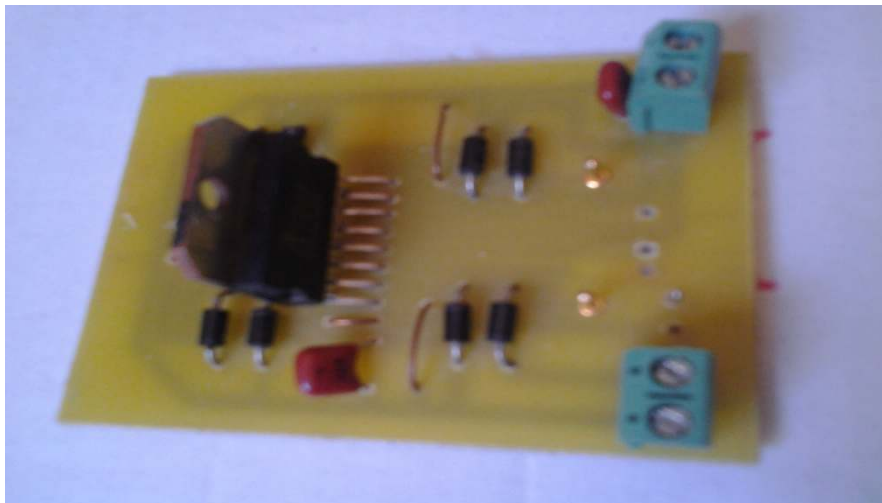


Figure 3.3.2 : La carte de puissance

3.4.1 Le pont en H :

Un Pont en H est un circuit électronique qui permet d'appliquer une tension à une charge dans deux directions différentes. Ce circuit est couramment utilisé en robotique et d'autres applications pour permettre aux moteurs DC de tourner dans les deux sens.

Le nom "pont en H", vient de la position des éléments de commutation dans le circuit, placés dans les quatre branches d'un H.

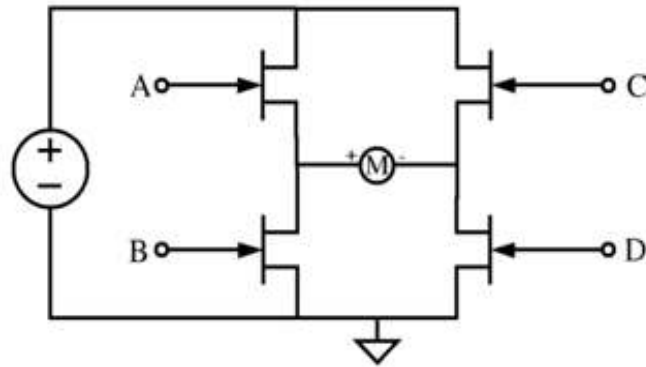


Figure 3.4.1 : Schéma d'un pont en H

3.4.2 Rôle du pont en H

Les moteurs électriques ont besoin de courants importants par rapport à la sortie maximale d'un microcontrôleur. Un moteur ne marcherait pas si on le branchait directement à la sortie d'un microcontrôleur. La première fonction du pont en H est de fournir de forts courants aux moteurs en permettant leur contrôle par le microcontrôleur. Une autre fonction est de pouvoir inverser la tension aux bornes du moteur pour inverser le sens de rotation. Inverser le sens d'un courant est une tâche simple dans les circuits électriques, il suffit de bien placer quelques interrupteurs. On utilise des transistors à la place d'interrupteurs parce qu'un transistor peut aussi amplifier un courant et, est plus facile à contrôler par des signaux numériques ou analogiques.

3.4.3 Fonctionnement du pont en H

Le sens de rotation du moteur est obtenu selon l'état des quatre transistors, le tableau ci-dessous présente les différents états du moteur.

A	B	C	D	
1	0	0	1	Le moteur tourne à droite
0	1	1	0	Le moteur tourne à gauche
0	0	0	0	N'est pas actionner
0	1	0	1	Le moteur freine
1	0	1	0	Le moteur freine

Tableau: Sens de rotation du moteur selon l'état des transistors

Les autres configurations sont interdites. ^[11]

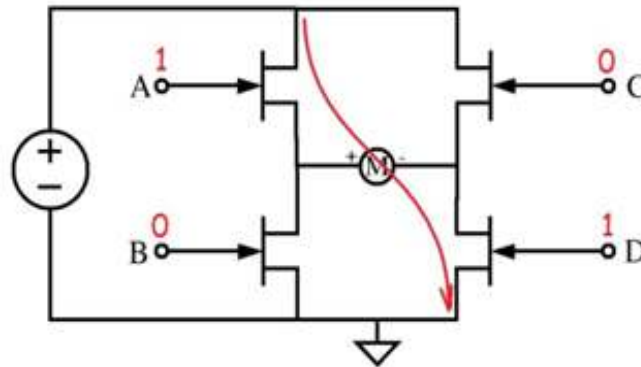


Figure 3.4.2: Le moteur tourne à droite

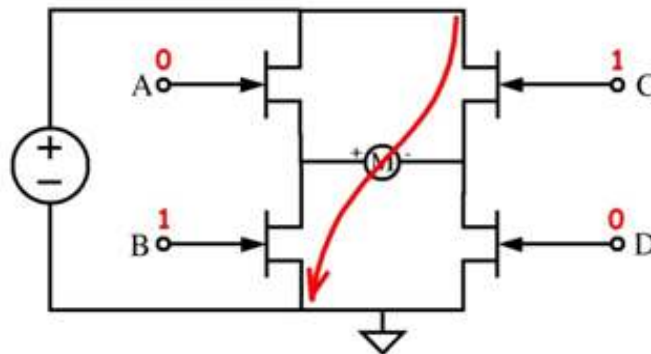


Figure 3.4.3 : Le moteur tourne à gauche

Il existe différents circuits intégrés réalisant les fonctions d'un **pont en H**, que nous choisissons selon la puissance nécessaire, la facilité de commande et les fonctions spéciales souhaitées.

3.4.4 Le circuit intégré L298 :

La carte de puissance est constituée du circuit intégré L298; Il s'agit d'un double Pont en H pouvant supporter 2A maximum par moteur et jusqu'à une tension de 46V, qui contrôle la puissance dans les moteurs pas à pas, à courant continu, dans les bobines etc....,

La puissance obtenue peut ainsi atteindre environ 200W, ce qui permet l'alimentation des moteurs puissants présentant des résistances de bobinage faibles. Il est donc conseillé dans notre cas, de fixer le circuit sur un dissipateur thermique de dimension adapté.

Le circuit intégré L298 (**figure 3.5.**) 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 (ENA et ENB).

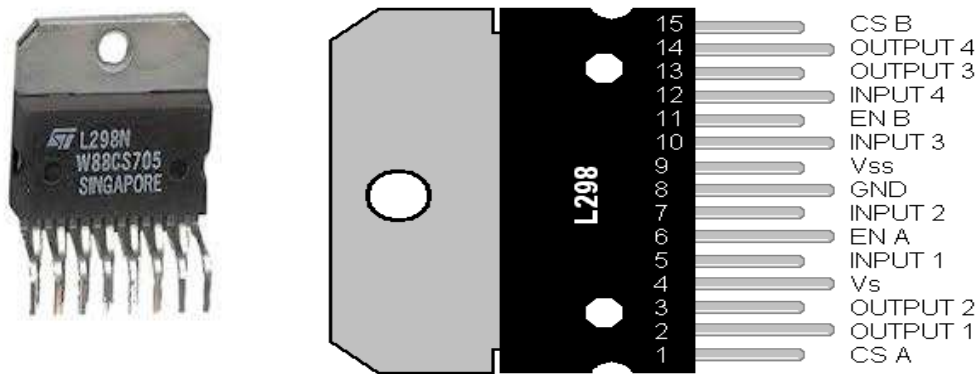


Figure 3.5 : Circuit intégré L298

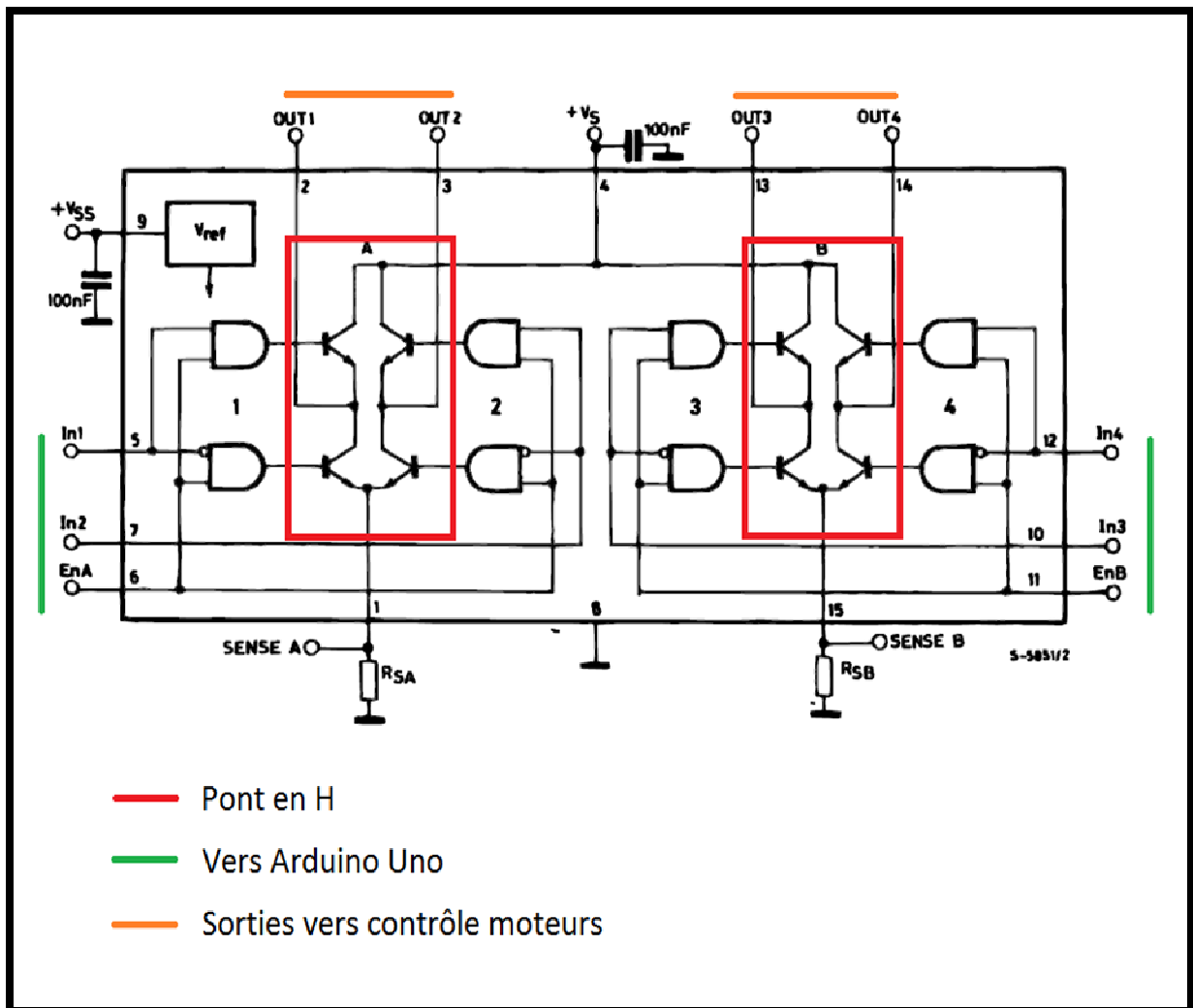


Figure 3.5.1: schémas interne du circuit L298

3.5 La carte de commande (Arduino) :

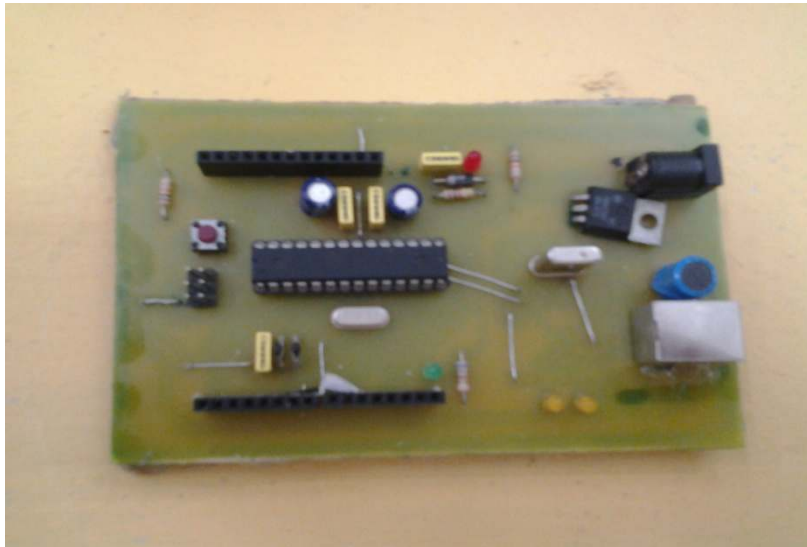


Figure 3.6 : La carte Arduino

Arduino est une plate-forme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique et d'un environnement de programmation.

Sans tout connaître ni tout comprendre de l'électronique, cet environnement matériel et logiciel permet à l'utilisateur de formuler ses projets par l'expérimentation directe avec l'aide de nombreuses ressources disponibles en ligne.

Pont tendu entre le monde réel et le monde numérique, Arduino permet d'étendre les capacités de relations humain/machine ou environnement/machine.

La carte Arduino repose sur un circuit intégré

La carte Arduino Uno est une carte à microcontrôleur basée sur l'ATmega328p



Figure 3.7 : Le Microcontrôleur ATMEGA 328p de la carte Arduino

Elle dispose :

- de 14 broches numériques d'entrées/sorties (dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée)),
- de 6 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques),
- d'un quartz 16Mhz,
- d'une connexion USB,
- d'un connecteur d'alimentation jack,
- d'un connecteur ICSP (programmation "in-circuit"),
- et d'un bouton de réinitialisation (reset).

Elle contient tout ce qui est nécessaire pour le fonctionnement du microcontrôleur; Pour pouvoir l'utiliser et se lancer, il suffit simplement de la connecter à un ordinateur à l'aide d'un câble USB (ou de l'alimenter avec un adaptateur secteur ou une pile, mais ceci n'est pas indispensable, l'alimentation étant fournie par le port USB).

La carte Arduino Uno diffère de toutes les cartes précédentes car elle n'utilise pas le circuit intégré FTDI usb-vers-série. A la place, elle utilise un Atmega8U2 programmé en convertisseur usb-vers-série.

3.5.1 Alimentation :

La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500mA) ou des piles. L'adaptateur secteur peut être connecté en branchant une prise 2.1mm positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accumulateurs peuvent être insérés dans les connecteurs des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V et la carte

pourrait être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. **Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V.**

Les broches d'alimentation sont les suivantes :

- VIN. La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5V de la connexion USB ou autre source 5V régulée). Nous pouvons alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- 5V. La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite "tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.
- GND. Broche de masse (ou 0V).

3.5.2 Mémoire :

L'ATmega 328 a 32Ko de mémoire FLASH pour stocker le programme (dont 0.5Ko également utilisés par le bootloader). L'ATmega 328 a également 2ko de mémoire SRAM (volatile) et 1Ko d'EEPROM (non volatile – mémoire). Le bootloader est un programme préprogrammé une fois pour toute dans l'ATméga et qui permet la communication entre l'ATmega et le logiciel Arduino via le port USB, notamment lors de chaque programmation de la carte.

3.5.3 Entrées et sorties numériques :

Chacune des 14 broches numériques de la carte UNO (numérotées des 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions `pinMode()`, `digitalWrite()` et `digitalRead()` du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de "rappel au plus" (pull-up) (déconnectée par

défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite(broche, HIGH)`.

De plus, certaines broches ont des fonctions spécialisées :

- **Communication Serie:** Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données série de niveau TTL. Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U2 programmé en convertisseur USB-vers-série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.
- **Interruptions Externes:** Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement d'état. Voir l'instruction `attachInterrupt()` pour plus de détails.
- **Impulsion PWM (largeur d'impulsion modulée):** Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction `analogWrite()`.
- **I2C:** Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils"), disponible en utilisant la librairie `Wire/I2C` (ou `TWI - Two-Wire interface - interface "2 fils"`) .
- **LED:** Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

3.5.4 Broches analogiques :

La carte Arduino Uno dispose de 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (c-à-d sur 1024 niveaux soit de 0 à 1023) à l'aide de la très utile fonction `analogRead()` du langage Arduino. Par défaut, ces broches mesurent entre le 0V (valeur 0) et le 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference()` du langage Arduino.

Note : les broches analogiques peuvent être utilisées en tant que broches numériques : elles sont numérotées en tant que broches numériques de 14 à 19.

3.5.5 Programmation :

La carte Arduino Uno peut être programmée avec le logiciel Arduino . Il suffit de sélectionner "Arduino Uno" dans le menu **Tools > Board** (en fonction du microcontrôleur présent sur la carte).

3.5.6 La régulation de la vitesse :

Nous avons besoin de commander la vitesse du robot, et pour ce faire nous utilisons la méthode du PWM (Pulse Width Modulation).

(Pulse Width Modulation) ou PWM, est une technique pour obtenir des résultats analogiques avec des moyens numériques. La commande numérique est utilisée pour créer une onde carrée, un signal de commutation entre on et off. Ce modèle on-off peut simuler des tensions entre le plein (5 Volts) ou désactiver (0 volt) en changeant la partie du moment où le signal passe sur en fonction du temps que le signal passe au large. La durée du «à temps» est appelé la largeur d'impulsion. Pour obtenir des variations analogiques, on agit sur la largeur d'impulsion. Si nous répétons ce modèle on-off assez vite avec une LED par exemple, le résultat est comme si le signal est une tension constante entre 0 et 5V contrôlait la luminosité de la LED.

Dans le graphique ci-dessous, les lignes vertes représentent un intervalle de temps régulier. Cette durée ou la période est l'inverse de la fréquence PWM. En d'autres termes, avec la fréquence PWM de Arduino à environ 500 Hz, les lignes vertes permettraient de mesurer 2 millisecondes chacun. Un appel à `analogWrite ()` est sur une échelle de 0 à 255, de sorte que `analogWrite (255)` demande un rapport cyclique de 100% (toujours), et `analogWrite (127)` est un cycle de service de 50% (sur la moitié du temps) pour exemple.

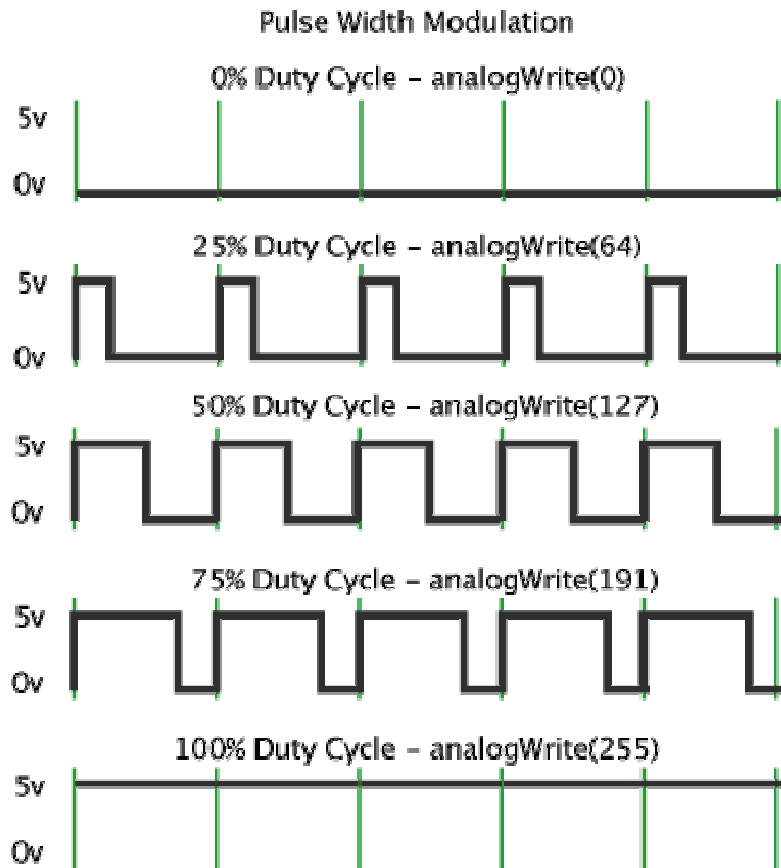


Figure 3.8: Chronograms (Pulse Width Modulation)

3.6 Protocole I2C :

3.6.1 Historique :

Le bus I2C (Inter Integrated Circuit) a été développé au début des années 80 par (Philips semi-conducteurs) pour permettre de relier facilement à un microprocesseur les différents circuits d'un téléviseur moderne.

3.6.2 Caractéristiques :

Le bus I2C permet de faire communiquer entre eux des composants électroniques très divers grâce à seulement trois fils : Un signal de donnée (SDA), un signal d'horloge (SCL), et un signal de référence électrique (Masse).

Ceci permet de réaliser des équipements ayant des fonctionnalités très puissantes (en apportant toute la puissance des systèmes microprogrammes) et conservant un circuit imprimé très simple, par rapport un schéma classique (8bits de données, 16 bits d'adresse + les bits de contrôle).

Les données sont transmises en série à 100Kbits/s en mode standard et jusqu'à 400Kbits/s en mode rapide. Ce qui ouvre la porte de cette technologie à toutes les applications où la vitesse n'est pas primordiale.

De nombreux fabricants ayant adopté le système, la variété des circuits disponibles disposant d'un port I2C est énorme : Ports d'E/S bidirectionnels, Convertisseurs A/N et N/A, Mémoires (RAM, EPROM, EEPROM, etc.....), Circuits Audio (Egaliseur, Contrôle de volume, ...) et autre drivers (LED, LCD, ...).

3.6.3 La prise contrôle du bus :

Pour prendre le contrôle du bus, il faut que celui-ci soit au repos (SDA et SCL à '1').

Pour transmettre des données sur le bus, il faut donc surveiller deux conditions particulières :

- La condition de départ. (SDA passe à '0' alors que SCL reste à '1')
- La condition d'arrêt. (SDA passe à '1' alors que SCL reste à '0')

Lorsqu'un circuit, après avoir vérifié que le bus est libre, prend le contrôle de celui-ci, il en devient le **maître**. C'est lui qui génère le signal d'horloge.

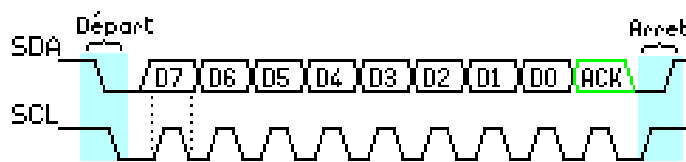


Figure 3.9 : Exemple de condition de départ et d'arrêt

3.6.4 La transmission d'un octet :

Après avoir imposé la condition de départ, le maître applique sur SDA le bit de poids fort D7. Il valide ensuite la donnée en appliquant pendant un instant un niveau '1' sur la ligne SCL. Lorsque SCL revient à '0', il recommence l'opération jusqu'à ce que l'octet complet soit transmis. Il envoie alors un bit ACK à '1' tout en scrutant l'état réel de SDA. L'esclave doit alors imposer un niveau '0' pour signaler au maître que la transmission s'est effectuée correctement. Les sorties de chacune étant à collecteurs ouverts, le maître voit le '0' et peut alors passer à la suite.

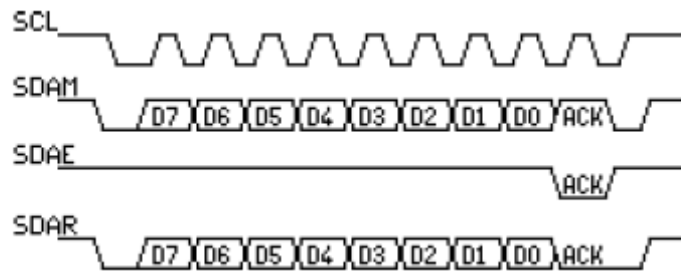


Figure 3.9.1: Exemple de transmission réussie

Dans cet exemple :

- * SCL : Horloge imposée par le maître.
- * SDAM : Niveaux de SDA imposés par le maître.
- * SDAE : Niveaux de SDA imposés par l'esclave.
- * SDAR : Niveaux de SDA réels résultants.

3.6.5 La transmission d'une adresse :

Le nombre de composants qu'il est possible de connecter sur un bus I2C étant largement supérieur à deux, il est nécessaire de définir pour chacun une adresse unique. L'adresse d'un circuit, codée sur sept bits, est définie d'une part par son type et d'autre part par l'état appliqué à un certain nombre de ces broches. Cette adresse est transmise sous la forme d'un octet au format particulier.

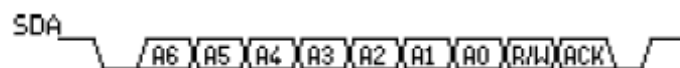


Figure 3.9.2 : Exemple d'octet d'adresse

On remarque ici que les bits D7 à D1 représentent les adresses A6 à A0, et que le bit D0 est remplacé par le bit de R/W qui permet au maître de signaler s'il veut lire ou écrire une donnée. Le bit d'acquiescement ACK fonctionne comme pour une donnée, ceci permet au maître de vérifier si l'esclave est disponible.

3.6.6 Lecture d'une donnée :

La lecture d'une donnée par le maître se caractérise par l'utilisation spéciale qui est faite du bit ACK. Après la lecture d'un octet, le maître positionne ACK à '0' s'il veut lire la donnée suivante (cas d'une mémoire par exemple) ou à '1' le cas échéant. Il envoie alors la condition d'arrêt.

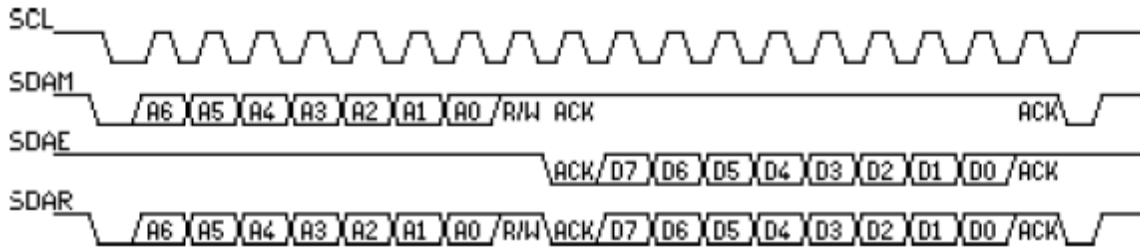


Figure 3.9.3 : Exemple de lecture d'une donnée

3.7 Le circuit intégré PCF8574 :

Le PCF8574 est un circuit intégré I2C (**Inter Integrated Circuit**) permettant de contrôler 8 d'entrées/sorties digitales en utilisant 2 broches de l'Arduino (SCL, SDA). En utilisant un PCF8574A, nous allons donc pouvoir ajouter un nombre important d'entrées/sorties à notre Arduino.

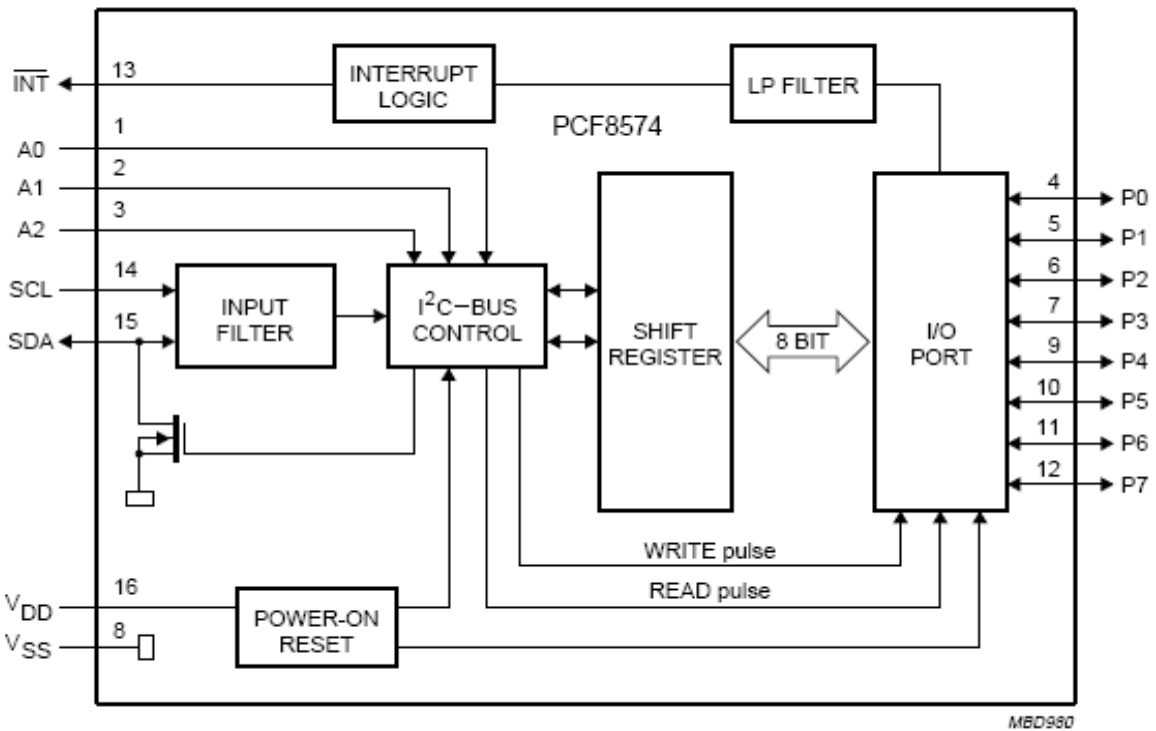


Figure 3.9.4: Schéma interne du PCF8574

Le principe de son fonctionnement un peu délicat est décrit dans le paragraphe suivant:

La figure ci dessus représente un schéma équivalent du port d'entrées/sorties du PCF8574. Le circuit permet de recevoir (ou d'émettre) des trames via la ligne SDA, cadencées par la ligne SCL. Lorsque l'adresse reçue concerne le composant et que le bit R/W est positionné à 0, alors l'octet suivant de la trame est mémorisé dans le registre d'écriture. Chacun des bits positionne l'interrupteur qui le concerne : l'interrupteur est fermé si le bit est à 0, ce qui fait passer le potentiel de la broche correspondante au niveau 0 volt. De même, si le bit est au niveau logique 1, alors l'interrupteur est ouvert, ce qui fait passer la sortie correspondante au niveau 5 Volts. Le raisonnement n'est valable que si la broche n'est pas reliée à un organe extérieur, par exemple un bouton poussoir qui forcerait le potentiel au niveau bas.

Les interrupteurs restent en l'état tant qu'un nouveau motif contenu dans une trame n'est pas parvenu au composant.

Lorsque l'adresse reçue concerne le composant et que le bit R/W est positionné à 1, alors le pcf8574 vient lire son registre de lecture. Celui ci reflète l'état des broches du composant, à condition que les transistors internes (ceux fixant les sorties) ne soient pas fermés, car sinon, ils forcent les broches au niveau bas.

On connecte dessus les signaux **SDA** et **SCL** de la carte Arduino UNO (vers les pins A4 (SDA) et A5 (SCL)) dont voici les pins :

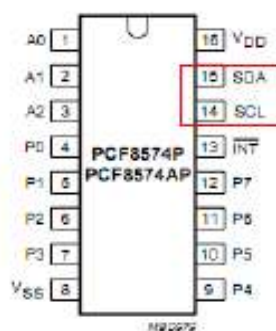
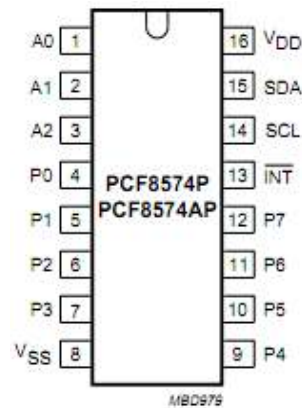


Figure 3.9.5 : circuit pcf8574

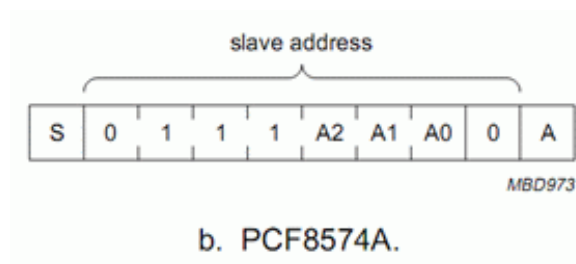
Les pins d'adressage sont les pins A0, A1, A2 :



INPUTS			I ² C BUS SLAVE ADDRESS
A2	A1	A0	
L	L	L	56 (decimal), 38 (hexadecimal)
L	L	H	57 (decimal), 39 (hexadecimal)
L	H	L	58 (decimal), 3A (hexadecimal)
L	H	H	59 (decimal), 3B (hexadecimal)
H	L	L	60 (decimal), 3C (hexadecimal)
H	L	H	61 (decimal), 3D (hexadecimal)
H	H	L	62 (decimal), 3E (hexadecimal)
H	H	H	63 (decimal), 3F (hexadecimal)

Tableaux d'adressage

Les valeurs sont obtenues à partir des bits d'adressage du PCF8574A donnés sur la datasheet du composant :



le PCF8574A a une plage allant de 70 à 7E ce qui nous permet d'utiliser 8 adresses différentes sans que les interfaces se gênent entre eux.

3.7.1 Schéma de la carte i2c :

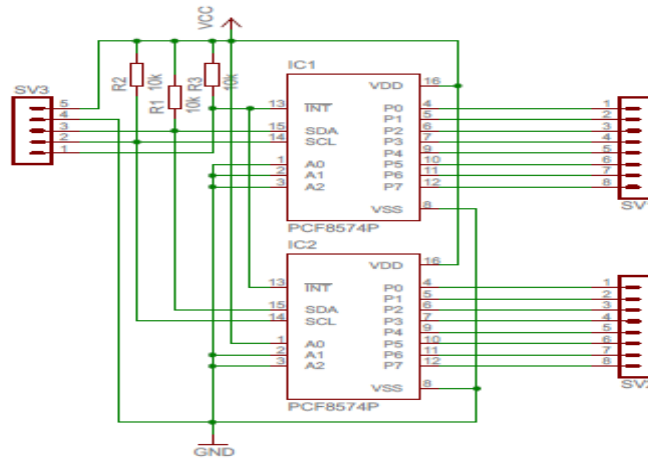


Figure 3.10 : schémas électrique de circuit I2C

3.7.2 Circuit imprimé I2C :

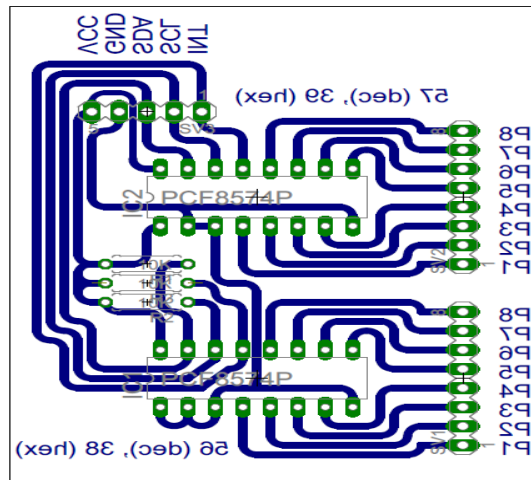


Figure 3.10.1 : Circuit imprimé d'I2c

3.8 Le servomoteur :

Un **servomoteur** (couramment appelé "servo" du latin "servus" signifiant "esclave") est un moteur capable de maintenir une opposition à un effort statique et dont la position est vérifiée en continu et corrigée en fonction de la mesure. C'est donc un système asservi.

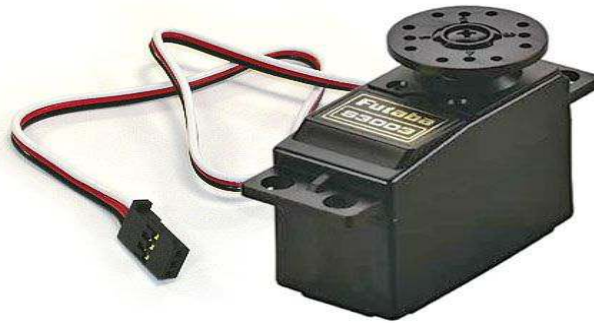


Figure 3.11 : Servomoteur (futaba s3003)

Pour faire tourner l'axe de sortie il faut lui envoyer un signal sur sa connexion de donnée. Ce signal est un train d'impulsions. C'est la largeur de cette impulsion qui détermine l'angle de rotation de l'axe de sortie. La durée de l'impulsion peut être comprise entre 0.9 et 2ms, 1.5ms correspond au centre. Cette impulsion doit être répétée toutes les 20ms.

Exemple: en envoyant une impulsion de 0.9ms l'axe va tourner jusqu'à la position extrême gauche. Avec 2ms il va tourner jusqu'à l'extrême droite. Et avec 1.5ms il revient au centre. Le potentiomètre sert à l'asservissement de position. L'axe du potentiomètre est solidaire de l'axe de sortie. Donc le potentiomètre tourne en même temps que l'axe de sortie. La résistance aux bornes du potentiomètre varie donc en fonction de la position de l'axe.

La résistance du potentiomètre est en rapport avec la durée du signal. ^[12]

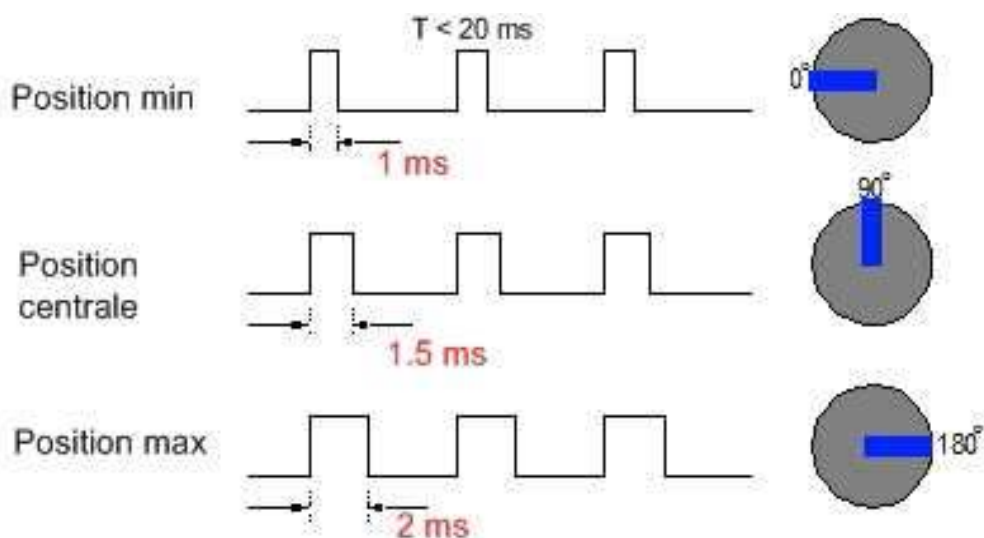


Figure 3.12 : Chronogramme servomoteur par la durée d'impulsion

Le signal de commande d'un servomoteur est un signal PWM de fréquence 50Hz et dont le niveau haut doit être entre 1 et 2ms :

- 1ms : -90° en angle de sortie
- 1.5ms : 0° en angle de sortie
- 2ms : $+90^\circ$ en angle de sortie

Le déplacement angulaire total est de 180° .

Dans ce projet, Le servomoteur permet de déplacer un bras jusqu'à une certaine position (ou angle).

3.9 Les capteurs :

3.9.1 Définition d'un capteur :

Un **capteur** est un dispositif qui transforme l'état d'une grandeur physique observée en une grandeur utilisable, exemple : une tension électrique, une hauteur de mercure, une intensité de courant, la déviation d'une aiguille.... On fait souvent la confusion entre capteur et transducteur : le capteur est au minimum constitué d'un transducteur (Un **transducteur** est un dispositif convertissant une grandeur physique en une autre).

3.9.2 Le Télémètres à infrarouge

On appelle télémétrie toute technique de mesure de distance par des procédés acoustiques, optiques ou radioélectriques. L'appareil permettant de mesurer les distances est lui appelé télémètre. De même qu'il existe différentes techniques de mesure de distance (mesure du temps de vol d'une onde, triangulation), il existe différentes technologies pour réaliser des télémètres. Nous présentons ici les plus répandues en robotique mobile, en donnant une idée de leur gamme de mesure et d'application. Tous les capteurs télémétriques, basés sur des mesures de l'environnement, sont bien évidemment actifs et extéroceptifs.

Ces télémètres possèdent l'avantage d'avoir un cône de détection beaucoup plus restreint.

Ils utilisent une lumière infrarouge au lieu d'une onde sonore pour la détection et peuvent être basés sur différentes techniques qui permettent de recueillir plus ou moins d'information.

Il est possible de mesurer simplement le retour ou le non-retour d'une impulsion codée, ce qui permet de détecter la présence ou l'absence d'un obstacle dans une certaine portion de l'espace.

Il est également possible de réaliser une triangulation sur le faisceau de retour de l'onde lumineuse, ce qui permet d'avoir une mesure de la distance de l'obstacle.

Les inconvénients de ces télémètres sont liés à leur portée, en général relativement restreinte, et à leur sensibilité aux fortes sources de lumières qui contiennent un fort rayonnement infrarouge.

Un projecteur du type de ceux utilisés pour la télévision pointé sur le robot, par exemple, sature en général complètement le récepteur et empêche toute détection d'obstacle. Ils sont également très sensibles à la couleur et à la nature de la surface de l'obstacle (par exemple, ils détectent difficilement les vitres). [13]



Figure 3.13: Principe du télémètre infrarouge à triangulation et exemple de télémètre réel

Le capteur GP2D120 de Sharp a l'avantage de se déclencher à n'importe quelle distance inférieure à 15 cm s'il rencontre un obstacle.

3.9.3 Capteur fin de course :

La détection d'un contact est certainement l'une des tâches les plus rencontrées dans la robotique mobile. Pour détecter un contact, nous pouvons utiliser un bouton-poussoir, un micro rupteur à levier (micro-Switch ou micro rupteur) ou tout simplement 2 fils dénudés.

Le micro-Switch, appelés aussi "Interrupteur de fin de course", sont des boutons poussoirs simples à utiliser à leur déclenchement, ils passent à l'état bas (0 volt) et en position de repos restent à l'état haut (5V). Ils coupent ou établissent un circuit lorsqu'ils sont actionnés par un mobile ou par un obstacle.

La détection s'effectue par contact d'un objet extérieur sur le levier ou un galet. Ce capteur peut prendre alors deux états :

- Enfoncé (en logique positive l'interrupteur est fermé).
- Relâché (en logique positive l'interrupteur est ouvert).

3.9.4 Principe de fonctionnement :

Le fonctionnement est simple, Le port du microcontrôleur (Interruption 0) est configuré en entrée, c'est-à-dire qu'il va lire une information. Tant que le bouton poussoir n'est pas actionné, le port est relié e a l'état Haut (5V) et le microcontrôleur génère une interruption a chaque fois ou cette pin change d'état vers l'état bas ce qui signifie qu'il y'a un obstacle et qu'il faut arrêter les moteurs,



Figure 3.14: Les Interrupteurs de fin de course

3.10 Méthode de décodage :

Méthode de décodage : afin d'obtenir le nombre de rotation du rotor ainsi que sa vitesse, nous pouvons adopter deux approches :

La première approche : consiste à utiliser le circuit de la figure. On a utilisé deux bascules D dont les sorties, qui seront utilisés pour décoder le sens de rotation, ainsi que la vitesse du rotor.

La deuxième approche du décodage sera de réaliser ces fonctions entièrement dans le programme de gestion du robot avec le reste des opérations.

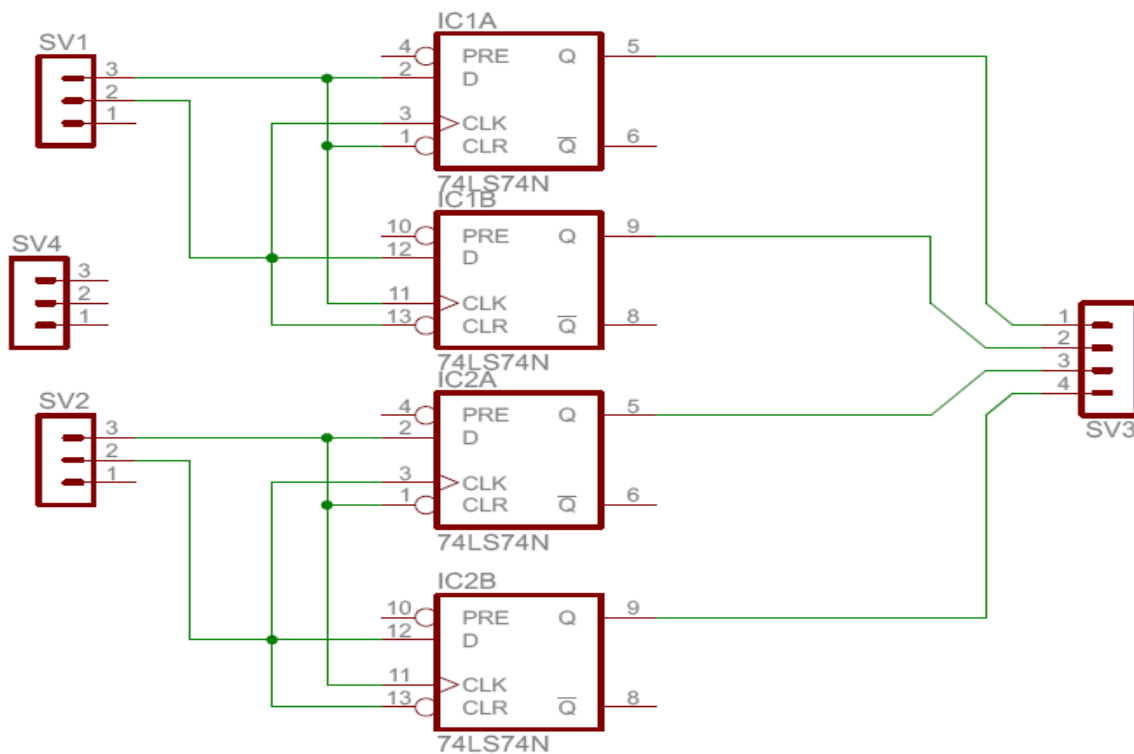


Figure 3.15: Schémas électrique de la bascule D

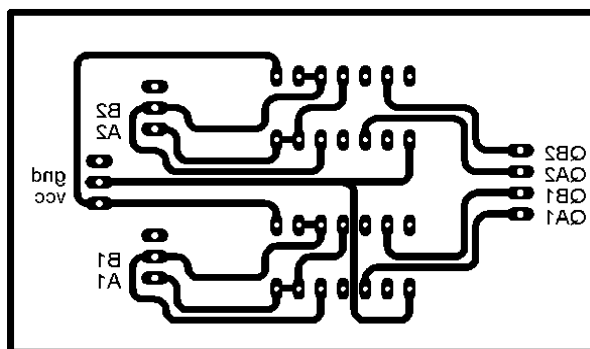


Figure 3.15.1: Schémas du circuit imprimé de la bascule D

3.11 Les encodeurs :

Un encodeur est un dispositif électromécanique qui permet de mesurer le mouvement ou la position. La plupart des encodeurs utilisent des capteurs optiques pour fournir des signaux électriques sous la forme de trains d'impulsions, qui peuvent à leur tour être traduits en mouvement, la direction ou la position. ^[19]

Pour calculer la position du robot, nous avons utilisé un encodeur d'une résolution de 500 cpr (500 impulsion par tour) (voir l'annexe).

Il délivre trois signaux, les canaux A et B donnant 500 impulsion /tour avec un déphasage d'un quart de période, et la canal I (index) donne une impulsion /tour.

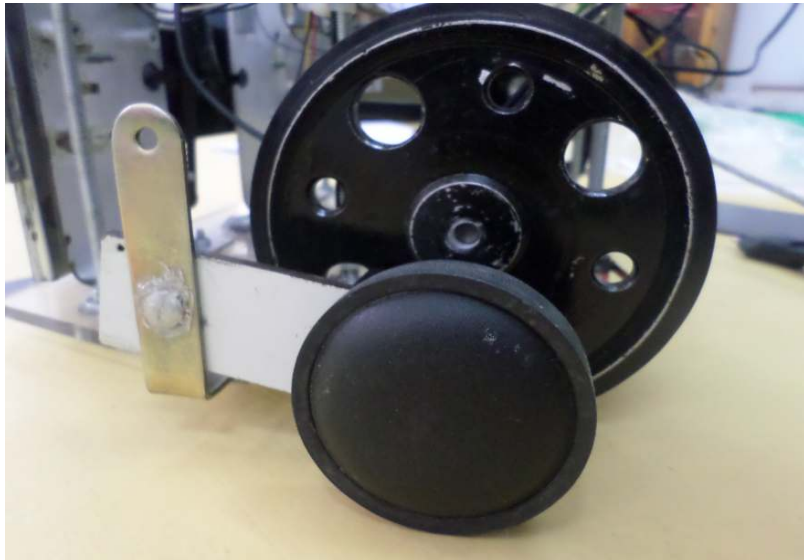


Figure 3.16 : Les encodeurs du robot

La roue codeuse est fabrique en métal, elle est formé de rayons vide/plain, elle tourne entre la LED et les photodiodes, les rayons lumineux émis par la LED dont stoppé ou laissé passer par le motif de la roue codeuse (rayons plein/ vide). Entre la LED et la roue codeuse en trouve une lentille en polycarbonate qui intensité la densité lumineuse émise par la LED.

Chaque canal (A,B et I) lui est attribué une paire de photodiode , ces dernière détectent les interruption causées par le motif de la roue codeuse en rotation, les envoies vers un circuit de mise en forme pour donner à sa sortie un signal et son inverse pour chaque canal (A, B et I), le signal et son inverse sont envoyées vers un comparateur, qui a son tour délivre les deux canaux final en quadrature (A,B) par leur comparateurs respectives, ces dernier sont envoyé avec la troisième signal délivré par le comparateur " I " vers un circuit spécialisé pour délivré à sortie le canal I final qui donne une impulsion/tour.

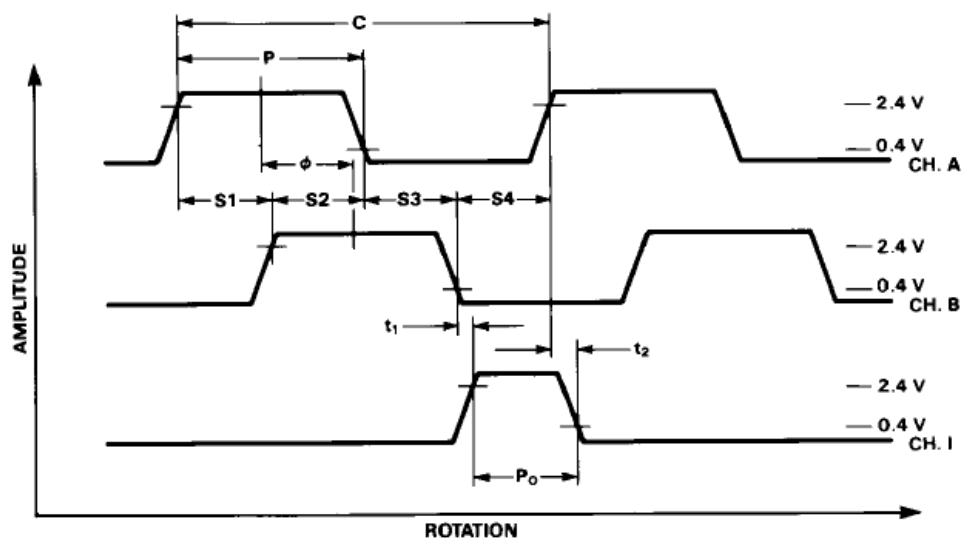


Figure 3.17 : La forme du signal délivré par les sorties A, B, I

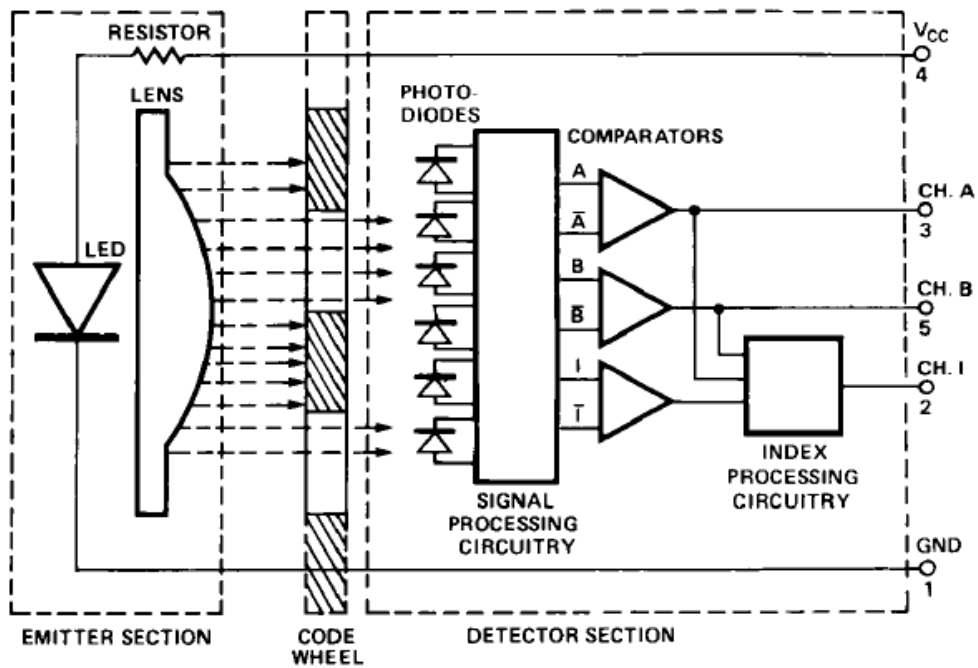


Figure 3.18 : Structure interne de l'encodeur

3.12 Conclusion

Dans ce chapitre nous nous sommes intéressés à la partie perception du robot mobile, nous avons présenté les différents capteurs utilisés, leurs principes de fonctionnement ainsi que le rôle que joue chacun d'eux pour le bon fonctionnement du robot mobile.

La gestion des différents systèmes du robot mobile se fait par le module Arduino (carte de commande) qui commande le servomoteur, la commande des moteurs à courant continu « CC » se fait par les ponts en H (carte de puissance).

Nous avons choisi de décrire une partie de l'électronique du robot qui est en relation directe avec les capteurs, cela nous a permis de comprendre la partie perception de tous les côtés. Les tests ont montré que toutes les cartes électroniques réalisées fonctionnent convenablement.

Chapitre 4 : L'odométrie

4.1 Introduction :

Afin de pouvoir se localiser un robot mobile dans un environnement , on doit toujours connaître sa situation ,pour se déplacer d'un point à un autre, le robot mobile a besoin de capteurs lui donnant des informations sur sa position absolue dans l'espace. Dans un environnement d'intérieur, les robots mobiles utilisent principalement l'odométrie, basée sur des données proprioceptives ce qui permet par intégration du déplacement, de connaître les coordonnées (x, y, θ) d'un point du robot dans un repère . Pour cela on a utilisé dans ce projet un système de mesure relatif de position odométrique. dans ce chapitre nous allons parler de la localisation d'un robot mobile par odométrie.

4.2 Définition :

L'odométrie est une technique de navigation couramment employée pour les robots mobiles navigants sur un sol plan, disposant d'une structure de locomotion à roues, ou les détecteurs de position sont disposés sur l'axe de la roue, où sur le système de transmission (par exemple sur la sortie de la boîte de vitesse).

Son principe consiste à déterminer la position et le cap (x, y, θ) d'un robot par intégration de ses déplacements élémentaires, de façon incrémentale, par rapport à un repère lié à sa configuration il ne nécessite qu'une puissance de calcul très limitée .

L'algorithme de localisation est basé sur le comptage des impulsions générées par des encodeurs durant une période d'échantillonnage choisie, il est possible d'exprimer de manière récursive la position et le cap du robot.

Tout d'abord, elle nécessite une connaissance précise de la géométrie du robot, le diamètre des roues doit bien sûr être déterminé, mais d'autres dimensions, comme l'entraxe ou le point de contact avec le sol, doivent également être prises en compte. Ces paramètres sont généralement difficiles à obtenir de façon précise, dépendent fortement de la nature du sol. De

plus, les cas difficiles sont l'estimation de la position et ils ne peuvent en général pas être mesurés directement. Selon le type de terrain ils peuvent rendre quasiment inutile l'odométrie en tant que méthode de localisation. Celle-ci est pourtant inévitable car, outre sa fonction d'estimateur de position, elle permet de réaliser des déplacements très fins.

4.3 Les odomètres (encodeurs) :

Les odomètres sont des capteurs proprioceptifs, ils permettent de fournir une quantification des déplacements curvilignes du robot en mesurant la rotation de ses roues.

Les odomètres sont généralement composés de codeurs incrémentaux qui permettent de mesurer l'angle de déplacement, ceci nécessitera la connaissance du diamètre des roues, de l'entraxe des roues, de la structure mécanique et cinématique du robot. Pour réaliser le système d'odométrie, certaines contraintes sont prises en considération:

- La nécessité d'avoir un contact permanent entre la roue de l'odomètre et la table de jeu.
- Que l'on ne souhaite pas avoir une précision supérieure à celle admissible par les capteurs.
- L'axe des roues roue libre avec celui des roues motrices doivent être sur le même axe (sur la même ligne).

Parmi les différentes technologies de codeurs (optique, électronique, inductive, capacitive..), les systèmes optiques sont les plus répandus en robotique mobile. Le codeur optique rotatif fonctionne selon le principe de balayage photoélectrique d'un disque en Acier sur le quel à été déposé un réseau radial plus ou moins complexe de traits. Lorsque ce disque effectue une rotation, le flux lumineux émis par une diode électroluminescente est modulé et les cellules photoélectriques qui captent la lumière à la sortie du disque délivrent un signal sinusoïdal. Ce signal est ensuite numériser. Un codeur incrémental génère ainsi de signaux décalés d'un quart de résolution du capteur est fonction du nombre n de traits par tours ($50 < n < 1000$). Sa précision dépend de la clarté des traits du réseau, de la qualité du montage mécanique et des performances de l'électronique de traitement des signaux (numérisation).

Placés sur les arbres moteurs dont le nombre de tours/minutes est l'image de la distance parcourue par le robot. Pour cela, on peut employer des codeurs de résolution moyenne montés sur l'axe du moteur.

4.4 Positionnement du robot :

Afin de pouvoir contrôler assez précisément la position du robot dans son environnement, il faut d'abord s'assurer que l'on est en mesure de contrôler précisément chacune de ses deux roues, équipées d'un « encodeur optique » qui permet de savoir les coordonnées(x, y) et l'angle (θ). En allant lire régulièrement, et assez rapidement les impulsions de l'encodeur, on récupère une position instantanée de chaque roue.

On cherche à coordonner le mouvement des roues afin de faire conduire le robot d'un point de la table à un autre aussi sûrement et rapidement que possible. Mais avant cela, il faut être en mesure de savoir où se trouve le robot. En effet, lorsque l'une des roues effectue un déplacement caractérisé par une petite avancée et une petite rotation. Si l'on calcule les variations de positions assez souvent, on peut rétablir la position du robot.

Une fois la position connue, on peut faire adopter au robot des actions lui permettant de se déplacer sur la table. Là encore, la génération de trajectoire est l'une des méthodes les plus simples qu'on va aborder dans la suite de ce chapitre.

Il y'a en fait deux manières d'approximer la trajectoire élémentaire parcourue par le robot pendant le temps d'échantillonnage T_e :

- En utilisant des segments de droites. On considère alors que le robot va en ligne droite pendant le temps T_e gardant l'orientation qu'il avait au début de cet élément de trajectoire. Ceci revient à supposer que les deux roues ont une vitesse constante et identique sur cet élément de trajectoire.
- En utilisant des arcs de cercles. On considère alors que le robot se déplace et change d'orientation suivant un arc de cercle pendant le temps T_e . Ceci revient à considérer que chacune des roues à une vitesse constante sur la période T_e mais que les vitesses des deux roues ne sont pas identiques.

Sur le robot, il n'y a que l'approximation par des segments de droites qui a été implémentée et elle donnait de bons résultats. La façon d'aborder le problème revient à émettre des relations géométriques permettant de déterminer le déplacement élémentaire du robot.

On présente ici la couche mathématique qui permet au robot de connaître en permanence sa position entre deux instants d'échantillonnage « n-1 » et « n », on mesure ainsi des déplacements élémentaires de roues, notées Δd et Δg correspondant respectivement à la roue droite et à la roue gauche.

On présentera quand même ici les deux méthodes d'approximation pour voir en quoi elles diffèrent :

4.5 Approximation par segmentation de droites :

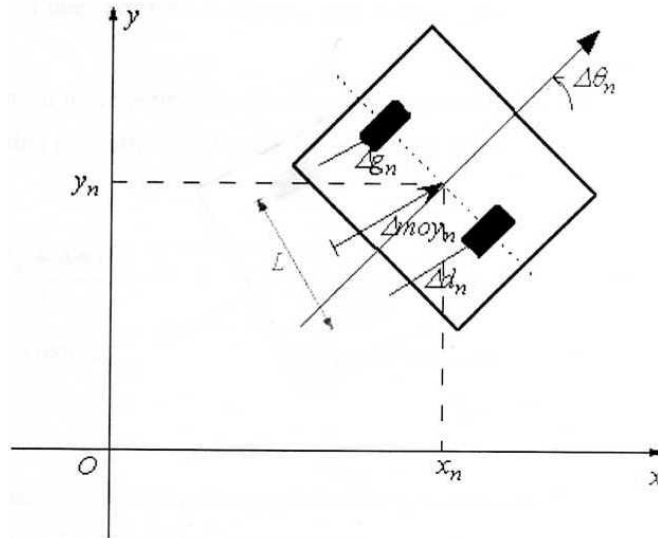


Figure 4.1 : Approximation par des segmentations de droites

Soit un intervalle de temps T_e . Connaissant la position du robot à l'instant $n-1$, on cherche la position à l'instant n .

Tandis que les erreurs non systématiques sont engendrés par :

- Des irrégularités sur le sol.
- Des glissements de roues codeuse dus à une surface glissante
- l'interaction d'une force interne (par exemple un défaut mécanique conduisant à une perte de données causée par des frottements).
- la roue n'est plus en contact permanent avec le sol.

La simple distinction entre les deux types d'erreur est d'une grande importance si l'on désire les réduire. Le problème avec les erreurs non systématiques est qu'il n'est par définition pas possible de les prédire, et qu'elles peuvent causer de fortes perturbations dans la précision du positionnement du robot.

Aucune de ces causes n'est supprimable, il faut donc trouver des solutions. L'ajout du système et des moyens repositionnant le robot était le moyen de limiter l'erreur grandissante.

Pour passer d'une position à l'autre, on réalise une trajectoire composée de deux mouvements :

- Une rotation sur place pour diriger le robot vers le point destination.
- Une translation pour aller du point de départ à l'arrivée.

On a donc :

$$\Delta moy_n = \frac{\Delta d_n + \Delta g_n}{2}$$

$$\Delta dif_n = \Delta d_n - \Delta g_n$$

$$\Delta x_n = \Delta moy_n \cos \theta_{n-1}$$

$$\Delta y_n = \Delta moy_n \sin \theta_{n-1}$$

$$\Delta \theta_n = \frac{\Delta dif_n}{L}$$

En effet, lorsque le robot fait un tour sur lui même ($2\pi \text{ rad}$), la différence de parcours entre les deux roues est de $2L$. Ce qui donne une différence de $L/2$ pour une rotation de π radian ainsi :

$$x_n = x_{n-1} + \Delta x_n$$

$$y_n = y_{n-1} + \Delta y_n$$

$$\theta_n = \theta_{n-1} + \Delta \theta_n$$

Or, l'orientation du robot doit toujours rester dans l'intervalle $[-\pi, +\pi]$, ce choix est totalement arbitraire mais il nous parait être le plus juste et le plus facile à utiliser, on aurait pu choisir $[0, 2\pi]$. On a donc du écrire une fonction réalisant un modulo dont le résultat est toujours dans l'intervalle $[-\pi, +\pi]$.

Approximation par des arcs de cercles:

En prenant la même notation que précédemment, on obtient la figure suivante :

On a toujours:

$$\Delta moy_n = \frac{\Delta d_n + \Delta g_n}{2}$$

$$\Delta \theta_n = \frac{\Delta dif_n}{L}$$

$$\Delta dif_n = \Delta d_n - \Delta g_n$$

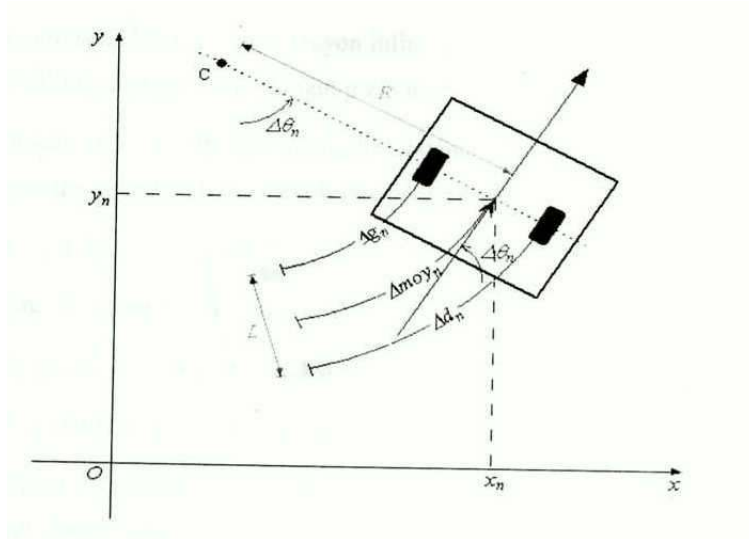


Figure 4.2 : Approximation par des arcs de cercles

4.6 Les erreurs de l'odométrie :

Basée sur les simples équations de trigonométrie facilement calculables dans le processeur, l'odométrie part de l'hypothèse que la révolution des roues codeuses peut se transformer en un déplacement linéaire par rapport au sol. Ceci n'est évidemment pas toujours vérifiable, puisque le moindre glissement d'une roue provoque une perte d'information et donc de précision. Toutefois, beaucoup d'autres sources d'imprécision peuvent intervenir dont on peut citer; l'imperfection de la mécanique, le glissement des roues au sol malgré les précautions prises, les erreurs se divisent en deux catégories, les erreurs systématiques aléatoires :

Des erreurs systématiques apparaissent lorsque :

- La moyenne des diamètres des roues diffère du diamètre nominal.
- La distance entre roues diffère de la distance nominale.
- Les roues ne sont pas correctement alignées.
- Le taux d'échantillonnage n'est pas défini.

4.7 La réduction de l'erreur de l'odométrie :

En ce qui concerne les erreurs systématiques, la précision d'un système d'odomètre est en grande partie liée à certaines dimensions critiques qui sont liées à la conception mécanique du robot (plate forme, moteurs, encodeurs...). Les erreurs non systématiques sont extrêmement difficiles à corriger puisqu'elles sont totalement imprévisibles et aléatoires. Si la perturbation

est limitée par les odomètres, alors une régulation software peut suffire à permettre au robot de retrouver sa trajectoire. En revanche, si les odomètres ont perçu une mauvaise information, il faut alors absolument un repère extérieur au robot pour qu'il soit capable de réinitialiser sa position. Nous pouvons alors imaginer plusieurs solutions qui peuvent être complémentaires, dont on cite les suivantes :

Les systèmes de positionnement par balises ont fait leurs preuves. Ils permettent de connaître la position du robot à chaque instant et sont indépendants de toutes erreurs non systématiques.

Toutefois, la précision maximale étant de l'ordre du centimètre, les balises ne permettent pas de découvrir les faibles perturbations. En revanche, couplées aux odomètres, elles permettent de diminuer fortement la sensibilité d'un robot aux majorités d'erreurs non systématiques.

La navigation inertielle utilise des « gyroscopes », (appareil, qui est animé d'un mouvement de rotation autour de l'un de ses axes, peut être placé d'une manière quelconque sans que la direction de son axe soit modifiée). Ou parfois des accéléromètres pour mesurer les vitesses de rotation ou les accélérations linéaires du robot. Ces mesures sont ensuite intégrées une ou deux fois pour obtenir les variations de position. Par ailleurs, il faut aussi souligner le coût élevé de ces systèmes lorsque l'on veut obtenir une bonne précision.

La réduction des erreurs d'odomètre peut être faite aussi par apprentissage et fusion de données ; le principe est basé sur l'apprentissage des comportements irréguliers lorsque le robot évolue dans l'environnement d'exploitation. La mise au point de cette méthode nécessite un grand nombre de manipulations. Les résultats obtenus peuvent théoriquement être améliorés par la prise en compte de capteurs inertiels.

Il n'existe pas, à l'heure actuelle, de méthode simple et suffisamment performante pour fournir une estimation précise et fiable de la position et du cap du robot pendant toute la durée de l'exécution d'une tâche. C'est la raison pour laquelle on fait généralement appel aux coopérations de plusieurs types de solutions.

4.8 conclusion :

L'odométrie est un mode de localisation d'une simplicité remarquable. Aussi, tout robot mobile un tant soit peu évolué en est équipé. En revanche, il s'agit aussi d'un mode de mesure particulièrement imparfait. Le calcul de la posture est fait en supposant qu'il n'y a pas de glissement et que les paramètres géométriques du robot sont parfaitement connus, notamment le diamètre des roues et la longueur de l'entraxe.

Chapitre 5 : Analyse et résultat

5.1 Introduction :

Notre tâche principale dans ce projet est de faire localiser un robot mobile sur un environnement (la table d'Eurobot), pour faire cela on doit d'abord concevoir une plateforme d'un robot à base de deux moteurs à courant continu, une roue bille, deux encodeurs pour faire calculer la distance parcourue par chaque roue, et ensuite en faire implémenter l'algorithme de localisation (Odométrie) sur une carte Arduino Uno suivi par un transfert des données vers Matlab pour faire calculer les coordonnées de robot sur la table.

5.2 Construction de robot :

On a construit le robot à base d'une plat forme en plexi glacé, ce robot peut déplacer (de manière non-holonome) on utilise les deux moteurs, le robot est équipé de deux encodeurs placés sur le même axe avec les moteurs pour éviter le glissement des roues des encodeurs, et il contient aussi des capteurs de fin de course et un télémètre infrarouge pour la détection des obstacles, le robot est équipé de trois étages le premier contient les moteurs, les encodeurs, la roue bille, les roues et la batterie, le deuxième étage contient la partie électronique, le microcontrôleur, la carte de puissance, le circuit I2C, et le circuit de décodage, et pour le troisième étage il contient le bouton d'arrêt d'urgence, le bouton on-off, le moteur de montage et un capteur infrarouge.

5.3 Calcul et l'envoi de la distance parcourue par le robot :

Pour faire ce calcul, l'organigramme suivant nous permettra de calculer la distance :

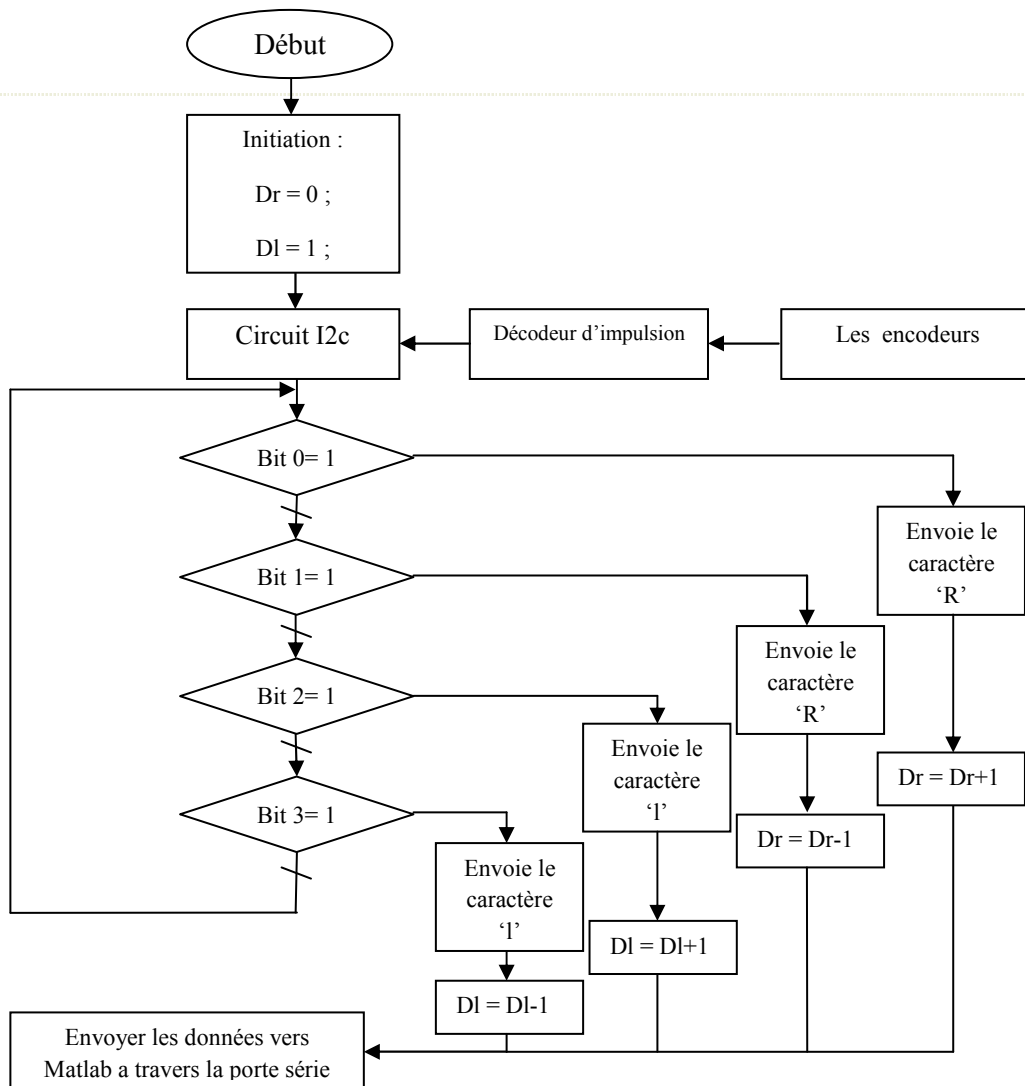


Figure 5.1 : l'organigramme du calcul de la distance dan le microcontrôleur

5.4 L'implémentation de l'algorithme de l'odométrie sur Matlab :

Après faire le calcul de distance sur le microcontrôleur, cette distance sera ensuite transmise vers le Matlab à travers une porte série, ce dernier fait le rôle de calculer les coordonnées (X, Y, θ) , afin de tracer la trajectoire du robot sur la table.

La figure (5.2) illustre l'organigramme de calcul des coordonnées :

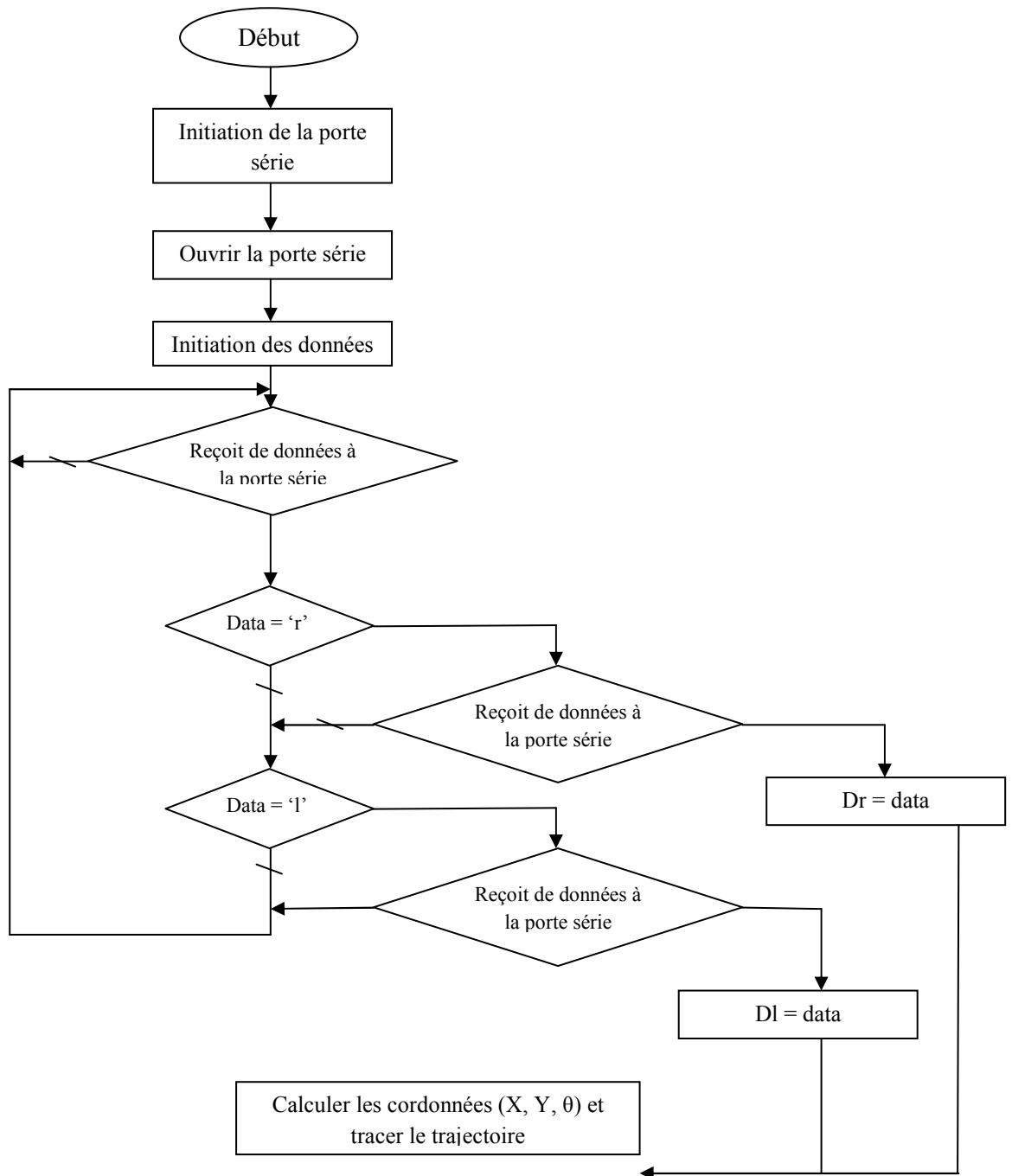


Figure 5.2 : l'organigramme de calcule des cordonnées

Quelque résultat obtenu dans le Matlab :

La figure (5.3) représente une tour a gauche de robot sur lui-même tel que le point tournant est le centre de gravité (point milieu) de robot, et la figure (5.4) représente une tour a droite de ce robot.

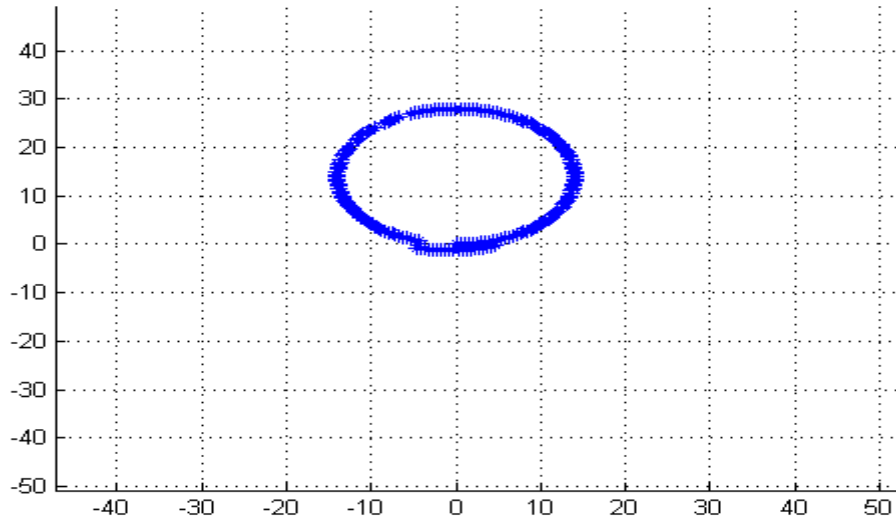


Figure 5.3 : Une tour a gauche de robot

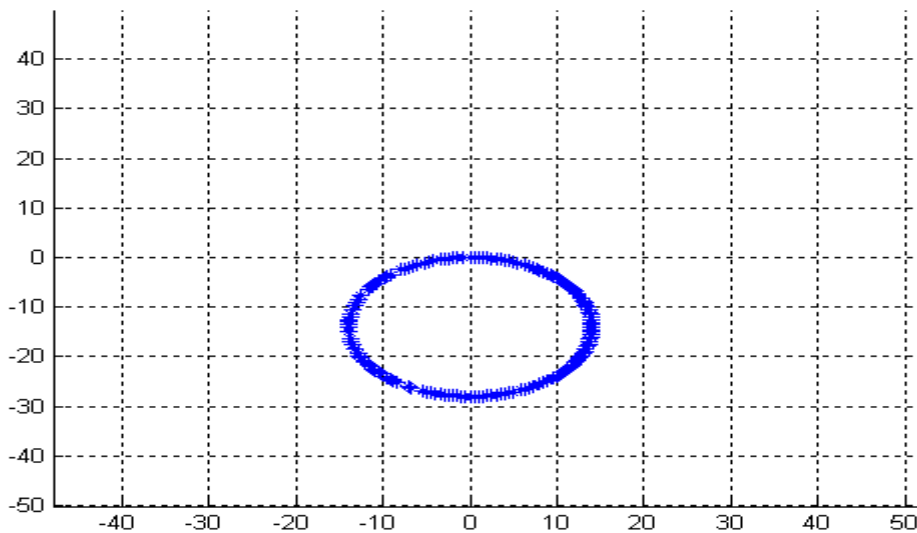


Figure 5.4 : Une tour a droite de robot

5.5 Analyse du résultat :

Lorsque on tourne les moteurs le robot commence à se déplacer, l'encodeur commence a données des impulsions, ces impulsion seront envoyer vers le décodeur d'impulsion, les sorties de ce décodeur sera transférer vers le circuit I2c, ce circuit contient des entrées/ sorties qui sont configurés a partir de microcontrôleur.

On a configurées ces portes comme des entrées pour lire ces impulsions (impulsion d'avance et de recule de chaque encodeurs), a chaque changement d'état de ces entrées, les données qui sont a l'entrées seront transmises vers le microcontrôleur.

Le microcontrôleur reçoit ces données et commence à faire calculer les impulsions et d'après ces impulsions on peut calculer la distance parcourue par chaque roue cette distance sera envoyée vers Matlab via une porte série.

Le Matlab reçoit ces données envoyées par le microcontrôleur et il traite ces données pour faire calculer les coordonnées et ensuite on les trace.

On a fait plusieurs tests pour cet algorithme, pour le premier test on a trouvé des difficultés pour l'envoi des données de l'Arduino vers Matlab a travers la porte série, on effectue la valeur reçue par Matlab n'est pas la même envoyée par l'Arduino, c'est-à-dire il y'a un chevauchement entre les données et cela est due à la vitesse de robot, mais lorsque on diminue la vitesse de robot au delà d'une certaine valeur ce dernier n'arrive pas a se déplacer, mais lorsque on utilise juste les encodeurs a une très lente vitesse l'algorithme fonctionne et donne des bons résultats.

5.6 Conclusion :

Afin d'assurer le bon fonctionnement de l'algorithme plusieurs conditions doivent être satisfaites tel que des moteurs a des couples importants et de vitesse minimale.

L'algorithme donne des résultats acceptables pour des marges de vitesse de l'ordre inférieur a 1/5 de la vitesse maximum de notre moteur.

Conclusion général

La tâche principale de notre projet était la détermination des coordonnées d'un robot mobile sur une table de Eurobot, dans le plan théorique l'implémentation de l'algorithme apparaît simple mais ce n'est pas le cas toujours on effectue plusieurs problèmes apparaissent surtout l'hors de implémentation de l'algorithme sur une plateforme réelle, on peut citer quelque problème tel que la vitesse des moteurs et son rôle dans le calcul de distance, la transmission des données vers matlab à travers la porte série et son influence sur le tracer de trajectoire du robot.

Malgré tout les problèmes cités précédemment, notre algorithme donne de bons résultats dans des marges de vitesse bien déterminées, ce projet nous a fourni une expérience considérable surtout sur l'approche pratique ce qui concerne la réalisation de la plateforme mobile et l'implémentation de l'algorithme.

Pour la continuité de ce projet nous devons insister sur la puissance des moteurs et son influence sur les calculs des coordonnées.

Bibliographie

Bibliographie

[1]: F. GAMARCHI. Petits robots mobiles, étude et construction. ETSF, N° ISBN 2 10 004708 6, 2001.

[2]: site web : <http://www.roboticus.org/robotique/29-introduction-a-la-robotique>

[3]: site internet : <http://tpe-robotique-industrielle.e-monsite.com/pages/1-1-histoire-et-l-evolution-de-la-robotique.html>

[4]: Robotique Mobile . David FILLIAT : Ecole Nationale Supérieure de Techniques Avancées

[5]: J-Y. TIGLI. Vers une Architecture de contrôle pour Robot Mobile orientée Comportement. Thèse de doctorat, Université de Nice-Sophia, Antipolis, 1996

[6]: cours robotique : Jean-Louis Boimond ; Université Angers

[7]: http://www.memoireonline.com/01/09/1842/m_Elaboration-ontroleuradaptatifrobot-mobile-base-sur-les-reseaux-immunitaires-artificiels3.html

[8]: site internet : <http://www.roboticus.org/robotique/30-principe-de-fonctionnement-des-robots>

[9]: règlement 2013: 20 eme édition de la coupe de France de robotique eurobot ; site Web : <http://www.eurobot.org/>

[10]: C. Grand (2004); Optimisation et commande des modes de déplacement des systèmes locomoteurs hybrides roue-patte. Application au robot Hylos. Thèse de doctorat, Université Paris 6; 2004 .

[11]: A. Louchene, système de navigation pour robot mobile, thèse de doctorat, de l'université de Batna 2004.

[12]: B. Abdeslam ; Contrôleur Flou Pour La Navigation D'un Robot Mobile D'intérieur ,
thèse de magister , de l'université batna .

[13]: site web : <http://rose.eu.org/2010/wp-content/uploads/2010/03/PWM-et-ponts-en-H.pdf>

[14]: site Web : <http://share.pdfonline.com>

[15]: site web : <http://fr.wikipedia.org/wiki/Servomoteur>

[16]: site Web : <http://www.robot-mobile-irbot.com/3-servomoteur-base-robot.htm>

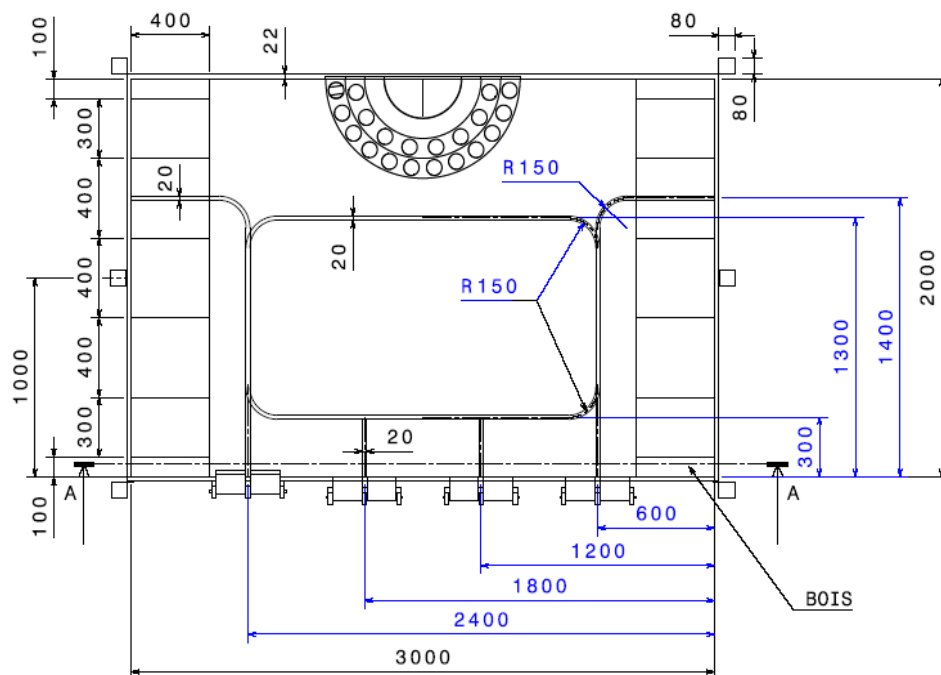
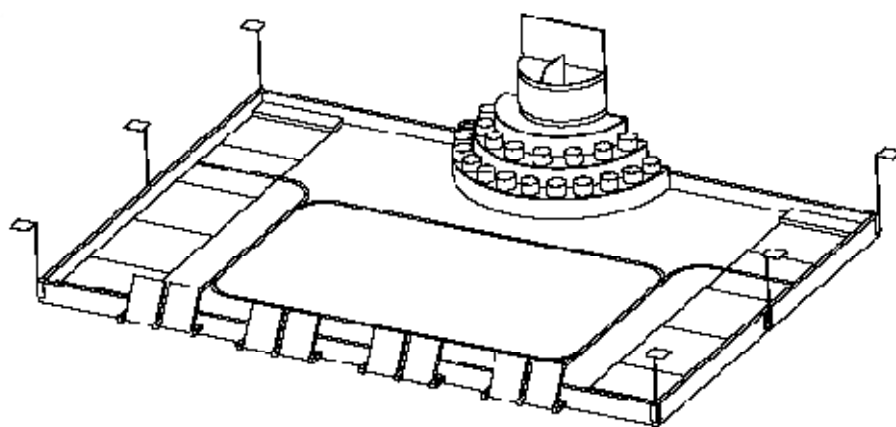
[17]: site Web : <http://share.pdfonline.com>

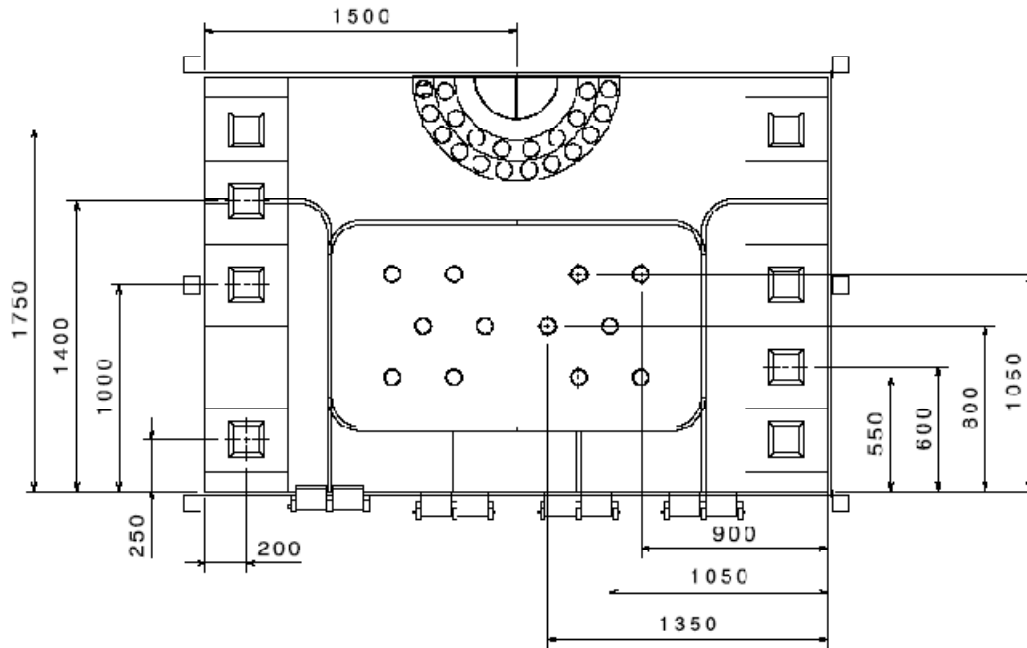
[18]: Robotique Mobile . David FILLIAT : Ecole Nationale Supérieure de Techniques
Avancées.

[19] : site web : <http://www.ni.com/white-paper/7109/en/>

Annexe

ANNEXES A: (CAHIER DES CHARGES)



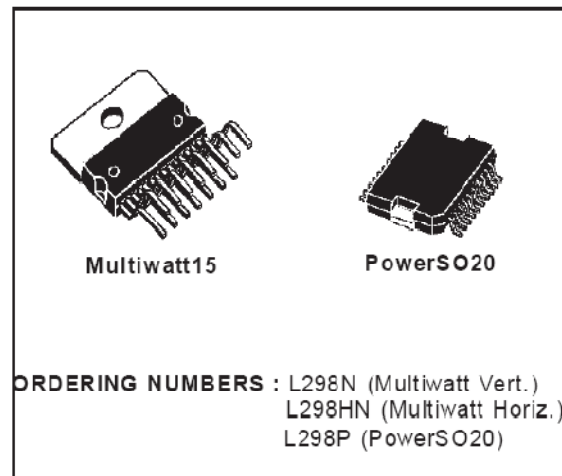


Annexe B: L298

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the con-

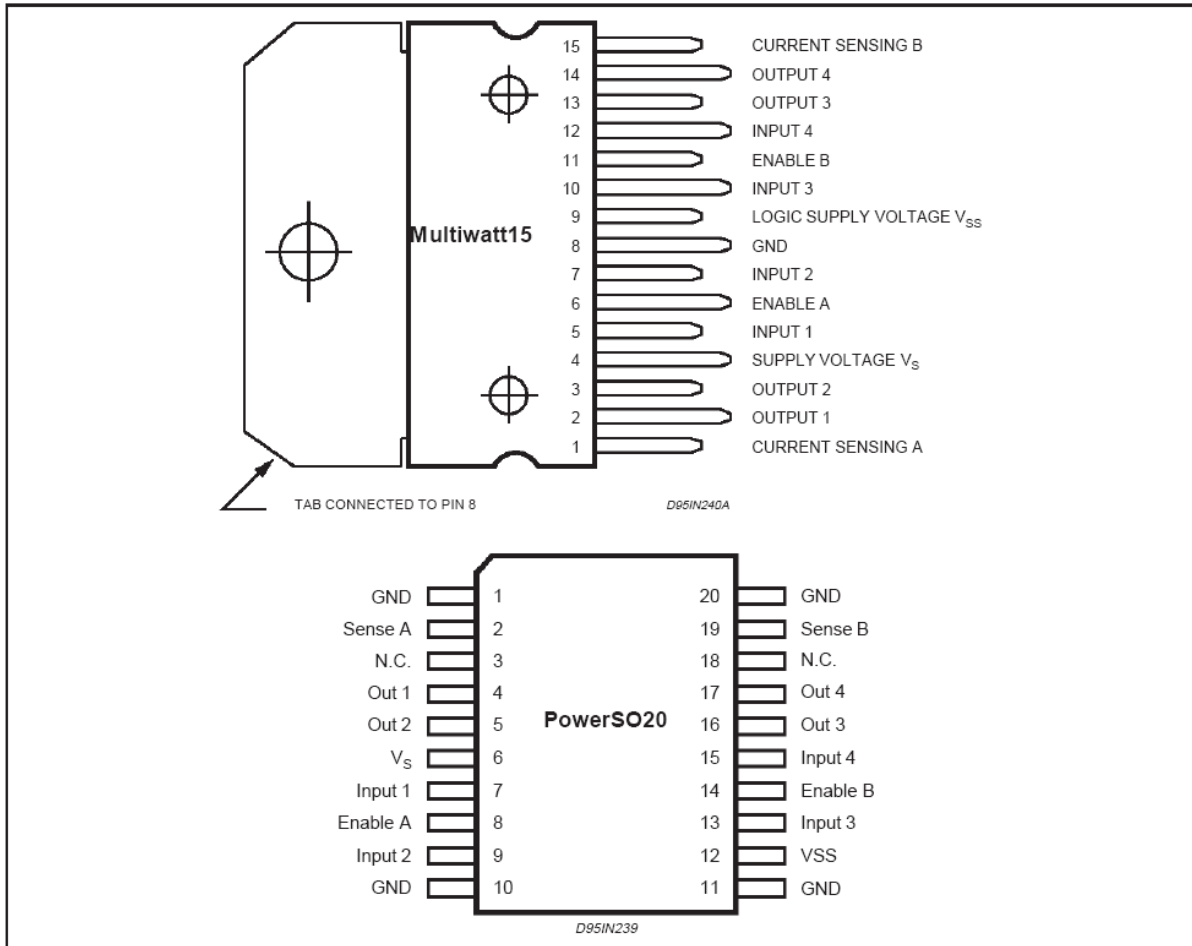


nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

ABSOLUTEMAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Power Supply	50	V
V_{SS}	Logic Supply Voltage	7	V
V_I, V_{en}	Input and Enable Voltage	-0.3 to 7	V
I_O	Peak Output Current (each Channel)		
	- Non Repetitive ($t = 100\mu s$)	3	A
	- Repetitive (80% on -20% off; $t_{on} = 10ms$)	2.5	A
	-DC Operation	2	A
V_{sens}	Sensing Voltage	-1 to 2.3	V
P_{tot}	Total Power Dissipation ($T_{case} = 75^\circ C$)	25	W
T_{op}	Junction Operating Temperature	-25 to 130	$^\circ C$
T_{stg}, T_j	Storage and Junction Temperature	-40 to 150	$^\circ C$

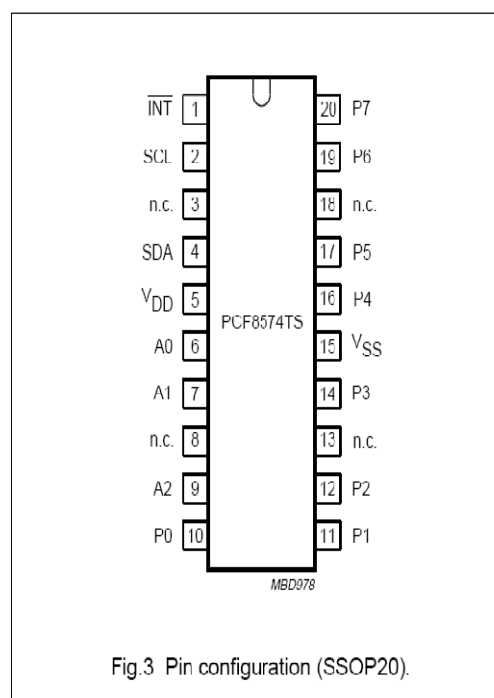
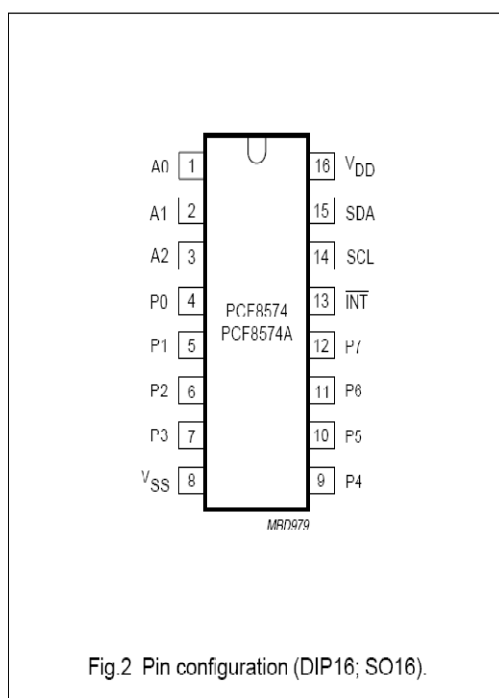
PIN CONNECTIONS (top view)



PIN FUNCTIONS (refer to the block diagram)

Annexe c : bus I²c (pcf8574A)

SYMBOL	PIN		DESCRIPTION
	DIP16; SO16	SSOP20	
A0	1	6	address input 0
A1	2	7	address input 1
A2	3	9	address input 2
P0	4	10	quasi-bidirectional I/O 0
P1	5	11	quasi-bidirectional I/O 1
P2	6	12	quasi-bidirectional I/O 2
P3	7	14	quasi-bidirectional I/O 3
V _{SS}	8	15	supply ground
P4	9	16	quasi-bidirectional I/O 4
P5	10	17	quasi-bidirectional I/O 5
P6	11	19	quasi-bidirectional I/O 6
P7	12	20	quasi-bidirectional I/O 7
$\overline{\text{INT}}$	13	1	interrupt output (active LOW)
SCL	14	2	serial clock line
SDA	15	4	serial data line
V _{DD}	16	5	supply voltage
n.c.	-	3	not connected
n.c.	-	8	not connected
n.c.	-	13	not connected
n.c.	-	18	not connected



Annexe : D

Quick Assembly Two and Three Channel Optical Encoders

Technical Data

Features

- **Two Channel Quadrature Output with Optional Index Pulse**
- **Quick and Easy Assembly**
- **No Signal Adjustment Required**
- **External Mounting Ears Available**
- **Low Cost**
- **Resolutions Up to 1024 Counts Per Revolution**
- **Small Size**
- **-40°C to 100°C Operating Temperature**
- **TTL Compatible**
- **Single 5 V Supply**

Description

The HEDS-5500/5540, HEDS-5600/5640, and HEDM-5500/5600 are high performance, low cost, two and three channel optical incremental encoders. These encoders emphasize high reliability, high resolution, and easy assembly.

Each encoder contains a lensed LED source, an integrated circuit with detectors and output circuitry, and a codewheel which rotates between the emitter and detector IC. The outputs of the

HEDS-5500/5600 and HEDM-5500/5600 are two square waves in quadrature. The HEDS-5540 and 5640 also have a third channel index output in addition to the two channel quadrature. This index output is a 90 electrical degree, high true index pulse which is generated once for each full rotation of the codewheel.

The HEDS series utilizes metal codewheels, while the HEDM series utilizes a film codewheel allowing for resolutions to 1024 CPR. The HEDM series is not available with a third channel index.

These encoders may be quickly and easily mounted to a motor. For larger diameter motors, the HEDM-5600, and HEDS-5600/5640 feature external mounting ears.

The quadrature signals and the index pulse are accessed through five 0.025 inch square pins located on 0.1 inch centers.

Standard resolutions between 96 and 1024 counts per revolution are presently available. Consult local Agilent sales representatives for other resolutions.

HEDM-550x/560x
HEDS-550x/554x
HEDS-560x/564x



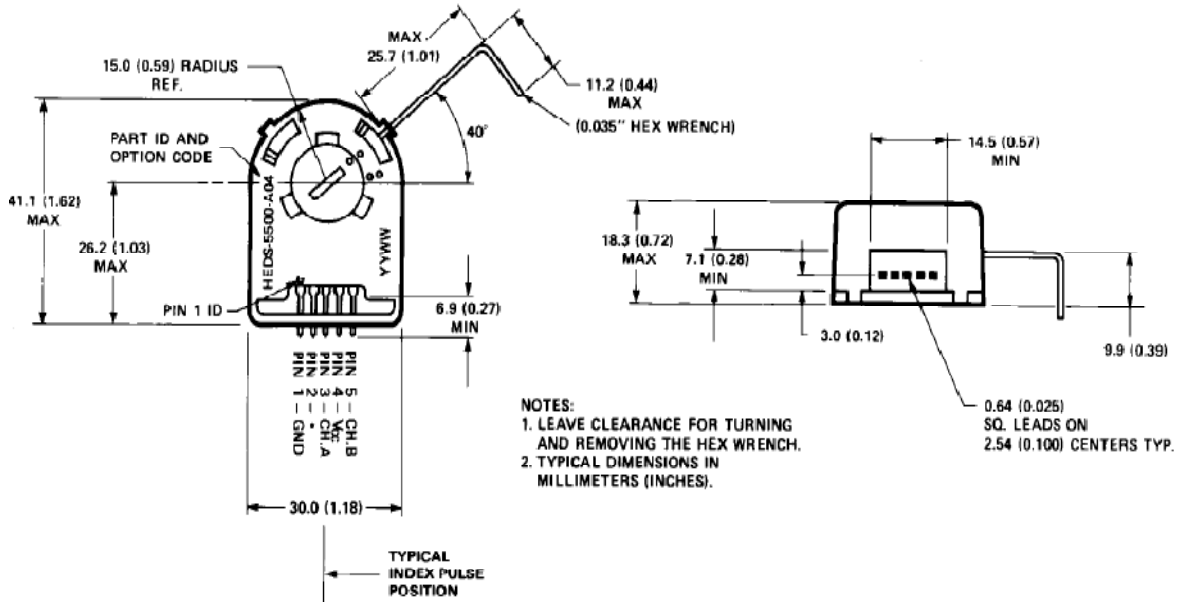
Applications

The HEDS-5500, 5540, 5600, 5640, and the HEDM-5500, 5600 provide motion detection at a low cost, making them ideal for high volume applications. Typical applications include printers, plotters, tape drives, positioning tables, and automatic handlers.

Note: Agilent Technologies encoders are not recommended for use in safety critical applications. Eg. ABS braking systems, power steering, life support systems and critical care medical equipment. Please contact sales representative if more clarification is needed.

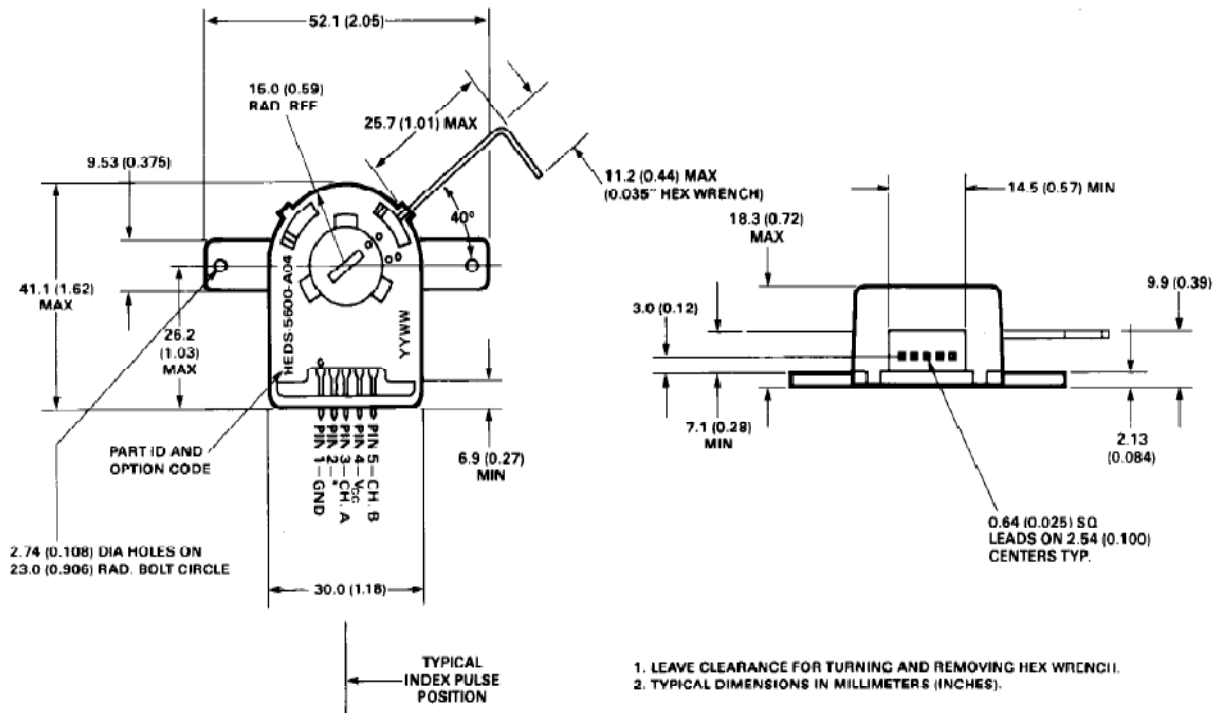
Package Dimensions

HEDS-5500/5540, HEDM-5500



*Note: For the HEDS-5500 and HEDM-5500, Pin #2 is a No Connect. For the HEDS-5540, Pin #2 is CH. I, the index output.

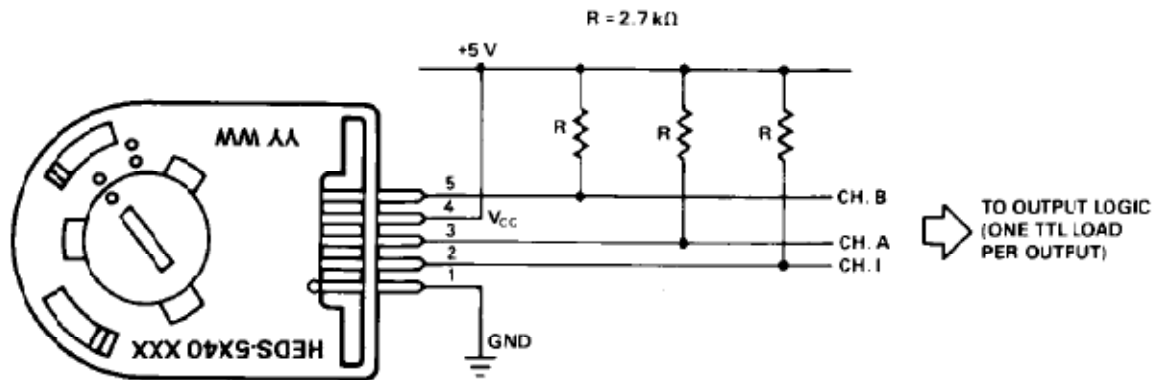
HEDS-5600/5640, HEDM-5600



*Note: For the HEDS-5600 and HEDM-5600, Pin #2 is a No Connect. For the HEDS-5640, Pin #2 is CH. I, the index output.

Absolute Maximum Ratings

Parameter	HEDS-55XX/56XX	HEDM-550X/560X
Storage Temperature, T_S	-40°C to 100°C	-40°C to +70°C
Operating Temperature, T_A	-40°C to 100°C	-40°C to +70°C
Supply Voltage, V_{CC}	-0.5 V to 7 V	-0.5 V to 7 V
Output Voltage, V_O	-0.5 V to V_{CC}	-0.5 V to V_{CC}
Output Current per Channel, I_{OUT}	-1.0 mA to 5 mA	-1.0 mA to 5 mA
Vibration	20 g, 5 to 1000 Hz	20 g, 5 to 1000 Hz
Shaft Axial Play	± 0.25 mm (± 0.010 in.)	± 0.175 mm (± 0.007 in.)
Shaft Eccentricity Plus Radial Play	0.1 mm (0.004 in.) TIR	0.04 mm (0.0015 in.) TIR
Velocity	30,000 RPM	30,000 RPM
Acceleration	250,000 rad/sec ²	250,000 rad/sec ²



Bibliographie

[1]: F. GAMARCHI. Petits robots mobiles, étude et construction. ETSF, N° ISBN 2 10 004708 6, 2001.

[2]: site web : <http://www.roboticus.org/robotique/29-introduction-a-la-robotique>

[3]: site internet : <http://tpe-robotique-industrielle.e-monsite.com/pages/1-1-histoire-et-l-evolution-de-la-robotique.html>

[4]: Robotique Mobile . David FILLIAT : Ecole Nationale Supérieure de Techniques Avancées

[5]: J-Y. TIGLI. Vers une Architecture de contrôle pour Robot Mobile orientée Comportement. Thèse de doctorat, Université de Nice-Sophia, Antipolis, 1996

[6]: cours robotique : Jean-Louis Boimond ; Université Angers

[7]: http://www.memoireonline.com/01/09/1842/m_Elaboration-ontroleuradaptatifrobot-mobile-base-sur-les-reseaux-immunitaires-artificiels3.html

[8]: site internet : <http://www.roboticus.org/robotique/30-principe-de-fonctionnement-des-robots>

[9]: règlement 2013: 20^{eme} édition de la coupe de France de robotique eurobot ; site Web : <http://www.eurobot.org/>

[10]: C. Grand (2004); Optimisation et commande des modes de déplacement des systèmes locomoteurs hybrides roue-patte. Application au robot Hylos. Thèse de doctorat, Université Paris 6; 2004 .

[11]: A. Louchene, système de navigation pour robot mobile, thèse de doctorat, de l'université de Batna 2004.

[12]: B. Abdeslam ; Contrôleur Flou Pour La Navigation D'un Robot Mobile D'intérieur , thèse de magister , de l'université batna .

[13]: site web : <http://rose.eu.org/2010/wp-content/uploads/2010/03/PWM-et-ponts-en-H.pdf>

[14]: site Web : <http://share.pdfonline.com>

[15]: site web : <http://fr.wikipedia.org/wiki/Servomoteur>

[16]: site Web : <http://www.robot-mobile-irbot.com/3-servomoteur-base-robot.htm>

[17]: site Web : <http://share.pdfonline.com>

[18]: Robotique Mobile . David FILLIAT : Ecole Nationale Supérieure de Techniques Avancées.

[19] : site web : <http://www.ni.com/white-paper/7109/en/>

