

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 دحلب البلدية
Université SAAD DAHLAB de BLIDA
كلية التكنولوجيا
Faculté de Technologie
قسم الإلكترونيك
Département d'Électronique



Mémoire de projet de fin d'études

Filière Automatique
Spécialité Automatique et Systèmes

Conception et Réalisation d'un robot mobile transporteur de gobelets

Présenté par :

TAIHI Mohamed Abd El Hafih

proposé par :

Mr.KAZED Boualem

Année Universitaire 2019-2020

Remerciements

*Louange à « **Dieu** » le tout puissant de m'avoir permis de réaliser ce modeste travail dans de très bonnes conditions, m'accordant toute l'énergie qu'il me fallait.*

*Je tiens à exprimer ma profonde gratitude à mon promoteur Mr. **B. Kazed**, pour m'avoir orienté et prodigué ses précieux conseils lors de la réalisation de ce projet.*

*De même, je remercie chaleureusement le **jury** qui a dû consacrer une partie de son temps à examiner ce projet, ainsi que toutes les personnes ayant collaboré de près ou de loin à cet accomplissement.*

Enfin, je saisis l'occasion pour exprimer ma reconnaissance à tous les enseignants qui ont assuré ma formation durant mon parcours universitaire.

Dédicace

Je dédie ce modeste travail en témoignage de ma reconnaissance. De mon cœur, et de mon profond respect :

A ma très chère mère qui a sacrifié sa noble existence pour bâtir la mienne, et qui est pour moi le symbole du courage et du sacrifice.

A mon cher père pour avoir mis tous les moyens à ma disposition et d'avoir été à mes côtés tout au long de ces années d'études.

A mes chers sœurs Amel, Amina et Fatima.

A ma grand mère mon grand père, mes tantes et mes oncles et leur familles et à tout la famille de proche et de loin.

Je le dédie enfin à tous mes amis et ceux qui me sont cher

Sommaire

1.1 Introduction général	1
--------------------------------	---

Chapitre1

Généralité sur la robotique mobile

1.1 Introduction	3
1.1.1 Apparition des robots (historique)	3
1.1.2 Définitions	6
1.2 Les robots mobiles	7
1.2.1 Présentation général sur les robots mobiles	7
1.2.2 Classification des Robots Mobile	8
1.2.3 Applications	9
1.2.4 Les avantages et les inconvénients de robot	11
A. Les avantages.....	11
B. les inconvénients	11
1.2.5 Architecture et les composants des robots mobiles	11
A. Architecture des robots mobiles	11
A .1 La structure mécanique et la motricité	12
A.1.1 Les robots mobiles à roues	12
A.1.2 Les robots mobiles à chenilles	14
A.1.3 Les robots mobiles marcheurs	14
A.1.4 Les robots mobiles rampants	16
A.1.5 La motricité et l'énergie	16
A.2 Les organes de sécurité	17
A.3 Traitement des informations et gestion des tâches	18
A.4 Le système de localisation	18
B. Les composants fonctionnels de robot	19
1.2.6 La robotique et l'intelligence artificielle	20
A. Intelligence artificielle	20
B. Les outils d'exécution de l'IA dans le robot	20
C. Types de programmes d'IA utilisées dans la robotique	20
1.3 Conclusion	21

Chapitre2

Description de la partie matérielle du robot

2.1 Introduction	22
2.2 Partie mécanique et mobile du robot.....	22
2.2.1 Structure	22
2.2.2 Partie motrice	23
A. Les roues	23
B. Roue à bille.....	24
C. Moteurs DC (MCC)	24
2.3 Système de ramassage de gobelet (gripper)	25
2.3.1 Fonctionnement du mécanisme	25
2.4 Partie électronique utilisé.....	27
2.4.1 Source d'alimentation	27
2.4.2 Carte de puissance	27
A. Pont en H	28
2.4.3 Codeur incrémental à effet Hall	31
2.4.4 La carte de commande Arduino méga.....	32
2.5 Transmission des données	32
2.5.1 la liaison entre les deux cartes via des Pins :(liaison série)	32
2.6 Asservissement du robot mobile.....	34
2.6.1 Contrôle PID	34
2.7 Conclusion.....	37

Chapitre 3

Modèle cinématique et régulation

3.1 Introduction.....	37
-----------------------	----

3.2 Modèle du robot mobile	38
3.2.1 Commande.....	39
3.4 Conclusion.....	42

Chapitre 4

Tests et résultats

4.1 Introduction.....	43
4.2 Explication de notre stratégie	43
4.2.1 Organigramme du programme utilisé sur Matlab	43
4.3 Solution des problèmes rencontrés.....	45
4.3.1 Simulation sur Matlab	45
4.3.2 Test réalisé.....	49
4.4 La variation de vitesse (commande du pont en H par PWM)	50
4.5 Intérêts de notre circuit électronique	51
4.6 La commande de la vitesse.....	52
4.7 La traduction de la simulation à la pratique	52
4.6 Conclusion.....	55

Table des figures

Figure 1.1 Dr. Grey Walter et sa tortue électronique	4
Figure 1.2 Premiers Robotique industrielle	4
Figure 1.3 Premiers robot mobiles	5
Figure 1.4 développent de la robotique mobile	6
Figure 1.5 Quelques exemples d'application des robots	9
Figure 1.6 Architecture d'un robot mobile	12
Figure 1.7 Robots à roues différentielles	12
Figure 1.8 Robot tricycle	13
Figure 1.9 Robot Voiture	13
Figure 1.10 Robot mobile omnidirectionnelles	14
Figure 1.11 Robot mobile à Chenilles	14
Figure 1.12 Robot hexapode (Asimo)	15
Figure 1.13 Robot hexapode	16
Figure 1.14 Robot Rampant	16
Figure 1.15 Synoptique de la sécurité	18
Figure 1.16 Architecture générale d'un système robotique	19
Figure 2.1 : Plateforme (châssis)	23
Figure 2.2 : Image en position verticale de trois roues du robot	23
Figure 2.3 : Roue à bille	24
Figure 2.4 : Schéma d'un moteur type EMG30	25
Figure 2.5 : La forme de notre gripper	26
Figure 2.6 : Les différentes positions de notre gripper	26
Figure 2.7 : Schéma électrique des composants de la carte de puissance	27
Figure 2.8 : Schéma d'implantation des composants de la carte de puissance.....	28

Figure 2.9 : Le pont en H.....	28
Figure 2.10 : Sens du courant en fonction de l'état des interrupteurs	29
Figure 2.11 : Pont en H avec transistors NPN, diodes et entrées Arduino.....	30
Figure 2.12 : L6203 H bridge	30
Figure 2.13 : Codeur incrémental	31
Figure 2.14 : Codeur incrémental à effet Hall	31
Figure 2.15 : La carte Arduino Méga associée à la carte de puissance	32
Figure 2.16 : Liaison série entre deux Arduino	33
Figure 2.17 : Le protocole d'échange asynchrone.....	33
Figure 2.18 : La boucle PID	34
Figure 2.19 : Réponse moteur asservi par la commande PID utilisé	35
Figure 2.20 : la vitesse des roues droite et gauche en fonction d'un échelon unité	35
Figure 2.21 : la vitesse des roues sur l'intervalle de temps [0 :18]	36
Figure 2.22 : La boucle d'asservissement du robot mobile	37
Figure 3.1 : La configuration cartésienne du robot	38
Figure 3.2 : Rayons de la roue droite et gauche dans une trajectoire de la vitesse instantanée.....	39
Figure 4.1 : Organigramme du programme utilisé sur matlab	44
Figure 4.2 : la rotation d'un angle θ_1 de la première étape.....	45
Figure 4.3 : le mouvement du robot de la 2ème étape	46
Figure 4.4 : la rotation d'un angle θ_2 de la troisième étape	47
Figure 4.5 : le mouvement final de notre robot	48
Figure 4.6 : le mouvement final de notre robot avec une comparaison entre la simulation et le réel	49

Figure 4.7 : Schéma développé du circuit L6203.....	51
Figure 4.8 : Rôle de la DRL	52
Figure 4.9 : Organigramme du programme utilisé sur arduino	53
Figure 4.10 : La vitesse des roues droite et gauche en fonction d'un échelon unité.....	54

Table des Tableaux

Tableau 1.1 Domaines d'utilisation des robots mobiles.....	10
Tableau 2.1 Caractéristique du moteur DC de type EMG30.....	48
Tableau 2.2 Connecteurs du codeur.....	31

ملخص:

يتمثل العمل الذي نناقشه في هذه المذكرة، في إنجاز روبوت متحرك من النوع التفاضلي مزود بألية مخصصة خصيصاً لتجميع أكواب مخروطية الشكل. من أجل أن ينتقل الروبوت إلى المكان المحدد حيث يوجد الشيء المراد التقاطه، يجب على الروبوت التحكم في موضعه، يتطلب هذا الجزء أولاً التحكم في المحركين المسؤولين عن حركات هذا الروبوت. سيتم حساب موقع الروبوت واتجاهه بعد النمذجة الرياضية للمنصة المتحركة. من خلال استغلال المشفرات المرتبة على جانبي مركز الروبوت وقياس المسافات. سيتم تنفيذ الخوارزمية على متحكم من نوع اردوينو. عندما يتم وضع الروبوت في الموقع المطلوب بالقرب من الكوب، سيتم تنشيط آلية الالتقاط لتحميل ونقل الشيء إلى موضع مبرمج مسبقاً من قبل المستخدم.

كلمات المفاتيح: الروبوت المحمول، التفاضلي، النمذجة الرياضية، المشفرات، قياس المسافات، اردوينو.

Résumé :

Le travail qu'on rapporte dans ce mémoire consiste à réaliser un robot mobile de type différentiel muni d'un mécanisme spécialement dédié au ramassage de gobelets de forme conique. Afin que le robot puisse se déplacer à l'endroit précis où se trouve l'objet à ramasser, le robot doit être asservi en position, cette partie exige d'abord l'asservissement des deux moteurs responsables des mouvements de ce robot. Le calcul de la position et de l'orientation du robot sera réalisé après une modélisation mathématique de la plateforme mobile. En exploitant les encodeurs disposés de parts et d'autres du centre du robot et de l'odométrie. L'implémentation de l'algorithme sera effectué sur un microcontrôleur de type ARDUINO. Lorsque le robot sera positionné à l'endroit désiré à proximité du gobelet le mécanisme de ramassage sera actionné pour charger et transporter l'objet à une position préprogrammée par l'utilisateur.

Mots clés : Robot mobile, différentiel, modélisation mathématique, encodeurs, odométrie, ARDUINO.

Abstract :

The work that we report in this thesis, consists of the realization a differential-type mobile robot equipped with a mechanism specially dedicated to collecting conical-shaped cups. In order for the robot to move to the exact location where the object is located to be picked up, the robot must be controlled in position; this part first requires the control of the two motors responsible for the movements of this robot. The calculation of the robot's position and orientation will be carried out after mathematical modeling of the mobile platform. By exploiting the encoders arranged on either side of the robot's center and odometry. The algorithm will be implemented on a microcontroller type ARDUINO. When the robot is positioned at the desired location near the cup the pickup mechanism will be activated to load and transport the object to a position preprogrammed by the user.

Keywords: Mobile robot, differential, mathematical modeling, encoders, odometry, ARDUINO.

Introduction générale

Les robots font partie intégrante de notre monde. Ils ont également conquis notre imaginaire, hantant nos univers de science-fiction jusqu'à se matérialiser sur nos écrans de par des capacités extraordinaires. Cependant, la réalité est toujours ordinaire.

Le but du travail que nous avons réalisé au cours de ce projet était, la conception et la réalisation d'un robot mobile capable de déplacer, naviguer et ramasser des objets sur la table du jeu d'une manière autonome afin d'atteindre un objectif prédéfini selon coordonnées x, y voulu.

Après une étude détaillée et profité de l'expérience que j'ai acquise lors de ma formation universitaire et professionnelle, nous avons donné une importance particulière quant à la conception des parties mécaniques de notre robot. Afin de gérer tous les mouvements nécessaires à réaliser les actions citées précédemment nous avons exploité un module intégré (Arduino) permettant de contrôler ces opérations en fonction des données acquises. La communication entre ces différents éléments est réalisée par un circuit électronique que nous aurons conçus à cet effet. Le coté <<intelligent>> du robot dépend des programmes implémentés dans le processeur utilisé, ce dernier est chargé de décider du comportement du robot durant tout le cycle de fonctionnement.

En ce qui concerne ce projet nous allons concevoir une pince (gripper) qui sera montée sur une plateforme mobile afin de lui permettre de manoeuvrer un objet (gobelet). Le robot proposé sera muni d'une carte Arduino avec un codeur incrémental qui sera utilisée pour calculer les coordonnées de l'objet à atteindre, la vitesse angulaire et linéaire du robot. Dans un premier temps les coordonnées de la position de l'objet seront utilisées pour diriger la partie mobile du robot et lui permettre de s'approcher de l'objet afin que, par la suite, la pince va fermer ses mâchoires pour faire appréhender le gobelet. Après ça, le mécanisme de ramassage va monter. L'opération suivante sera de calculer l'angle de rotation du robot ainsi que son déplacement linéaire, ceci sera réalisé en utilisant la cinématique dont les détails seront expliqués dans le chapitre 3. La dernière étape que devra accomplir notre robot sera de faire une rotation et une translation vers le deuxième point ciblé et déposer le gobelet à un point, dans le plan, dont les coordonnées seront préprogrammées.

Introduction Générale

Dans ce mémoire nous allons donner tous les détails relatifs à la conception de ce projet et qui s'articule autour de quatre chapitres :

- ✓ Le premier chapitre, on a parlé sur la robotique en général, on a aussi défini les robots ainsi on a classifié les robots mobiles, leurs domaines d'application, les avantages et les inconvénients de la robotique, les composants et l'architecture d'un robot mobile et l'intelligence artificielle dans le domaine de la robotique.
- ✓ Le deuxième chapitre est consacré à la description et la conception de la mécanique du robot avec de ses différentes éléments et l'électronique utilisé.
- ✓ Dans Le troisième chapitre nous avons présenté la modélisation d'un robot mobile, les méthodes de calculs des différentes trajectoires du robot ainsi leur vitesse angulaire et longitudinale.
- ✓ Le dernier chapitre contient les tests effectués et les résultats obtenus.

Au final, on a clôturé avec une conclusion générale dans laquelle on porte un regard critique sur l'ensemble des travaux réalisés et on finit cet exposé par quelques propositions de travaux.

Chapitre1

Généralités sur la robotique mobile

1.1 Introduction

La robotique peut être définie comme l'ensemble des techniques et études tendant à concevoir des systèmes mécaniques, informatiques ou mixtes, capables de se substituer à l'homme dans ses fonctions motrices, sensorielles et intellectuelles.

Dans ce chapitre, on va présenter le domaine de la robotique, historique, les définitions d'un robot, les classifications des robots mobiles, domaines d'application, Les avantages et les inconvénients de la robotique, les composants et l'architecture d'un robot mobile, l'intelligence artificielle dans le domaine de la robotique.

1.1.1 Apparition des robots (historique)

Depuis des millénaires, l'homme a toujours été fasciné par l'idée de créatures intelligentes qui lui ressembleraient et serait à son service. Le mot robot a été utilisé pour la première fois en 1921 par Karel Capek dans sa pièce R.U.R «Rossums Universal Robots » [1].

Il provient du tchèque "robota" qui signifie corvée, travail obligatoire.

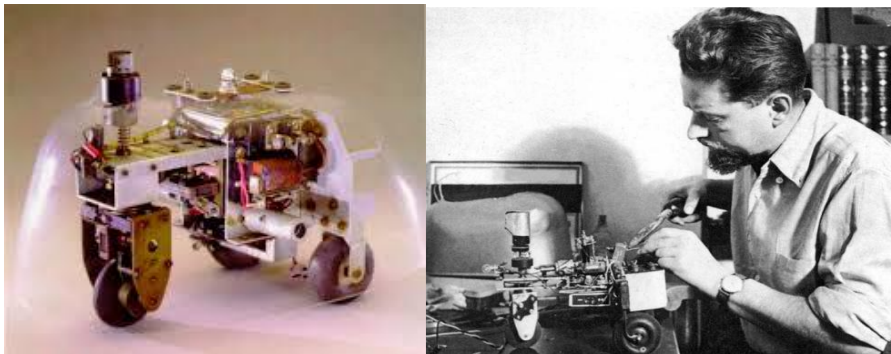
Le terme robotique a été employé pour la première fois par un écrivain russe Isaac Asimov en 1941 dans ces écrits, dans lesquels il invoque la mentalité de l'interaction entre les robots et les humains, les robots dans l'imagination d'Asimov seraient équipés de cerveaux dans lesquels les trois lois qui suivent seraient programmées :

1. Un robot ne doit pas porter atteinte à un être humain ou bien, sans intervenir, laisser porter atteinte à un humain.
2. Un robot doit obéir aux ordres d'un humain, sauf si de tels ordres entrent en conflit avec la Première loi.
3. Un robot doit protéger son existence tant que cela ne viole pas les deux premières lois.

Dans les années 1940 et 1950, les progrès de l'électronique permettent de miniaturiser les circuits électriques (inventions du transistor et du circuit intégré), ouvrant ainsi de nouvelles

voies à la fabrication des robots. Dans les premiers temps de la robotique, le robot est considéré comme une imitation de l'homme, aussi bien fonctionnelle que physique.

Grey Walter invente le premier robot mobile autonome en 1948, une tortue se dirigeant vers les sources de lumière qu'elle percevait (figure 1.1). Cependant, ce robot n'est pas programmable. La mise en place de robots n'a été possible que dans les années 50 avec la création des transistors et circuits intégrés.



1. Tortue électronique (1969)

2. Dr Grey Walter (1948)

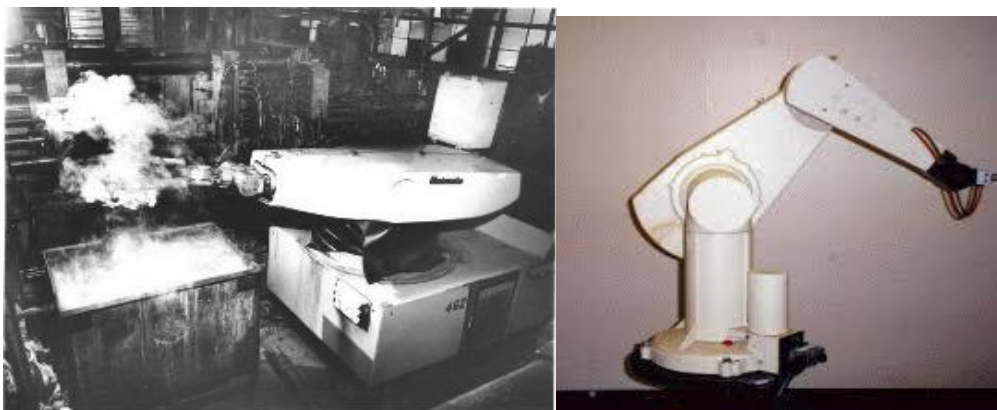
Figure 1.1 : Dr. Grey Walter et sa tortue électronique

➤ Les premiers robots industriels :

1961 : Premier robot industrielle mis en place dans une usine de General Motors : UNIMATE chez GM en fonderie sous pression (figure 1.2).

1972 : Nissan ouvre la première chaîne de production complètement robotisée.

1978 : PUMA (Programmable Universal Machine for Assembly) développé par General Motors (toujours utilisé) (figure 1.2).



1. Robot Unimate

2. Robot Puma

Figure 1.2 : Premiers Robots industrielle

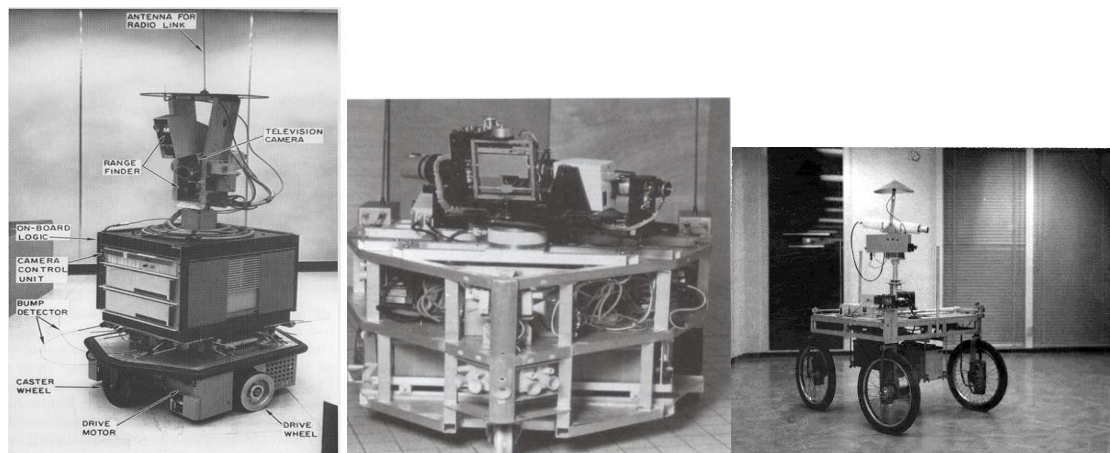
➤ Les premiers robots mobiles :

1960-64 : Ouverture des laboratoires d'Intelligence Artificielle au M.I.T., Stanford Research Institute (SRI), Stanford University, University of Edinburgh.

1967 : Mise en place de « Shakey » premier robot mobile intégrant perception, planification et exécution (figure 1.3.1).

1970 : Stanford Cart (figure 1.3.3).

1977 : premier robot mobile français HILARE au LAAS (CNRS Toulouse) (figure 1.3.2).



1. shakey

2. Hilare

3. Stanford cart

Figure 1.3: Premiers robot mobiles

Le monde a connu un développement important dans l'intelligence artificielle et de la robotique : de nouveaux robots apparaissent avec une amélioration dans tous les domaines.

1995 : Mise en place de la RoboCup.

1997 : premier robot mobile extra planétaire sur Mars.

1999 : Lancement du robot chien Aibo, par SONY.

2003 : Projet "Mars Exploration Rover" (Spirit & Opportunity).

2009 : Projet "Mars Science Laboratory" succédant au projet Rover, envoi prévu de Curiosity fin 2011.

2000 : Lancement d'Asimo, par HONDA.

2009 : robot Nao utilisé dans la RoboCup Soccer.



a. Robocup

b. Chien Aibo

c. Robot Spirit (image d'artiste)



d. Robot Curiosity



e. Asimo



f. robot Nao

Figure 1.4 : développement de la robotique mobile

1.1.2 Définitions

Le mot robot vient d'une pièce de théâtre tchèque dans laquelle apparaissait un travailleur artificiel employé pour le (servage), désigné en tchèque par (robota) [1].

Il existe plusieurs définitions pour d'écrire un robot, parmi celles-ci Webster définit robot comme « étant un équipement au appareil qui accomplit des fonctions normalement réservé à des humains ou bien qui opère avec une intelligence proche de celle des humains ».

L'institut Américain de Robotique définit un robot comme « Robot est un manipulateur multifonctions reprogrammables, conçu pour déplacer des matériel, des outils, des pièces, ou bien des dispositifs spécialisés à travers des mouvements programmables afin d'accomplir différents types de tâches ».

Le petit la rousse définit un robot comme « Robot est un appareil automatique capable de manipuler des objets au d'exécuter des opérations selon un programme fixe ou modifiable », souvent un robot est défini comme un manipulateur automatique à cycles programmable. Pour mériter le nom robot, le système doit posséder une certaine flexibilité, caractérisée par les propriétés suivantes :

Auto adaptabilité :

L'auto adaptabilité veut dire que le robot devrait être capable d'accomplir correctement sa tâche, même s'il rencontre de nouvelles situations inattendues, sans intervention humain.

Versatilité :

La versatilité signifie que le robot devrait être capable d'effectuer une variété des tâches, ou la même tâche de différente manière.

1.2 Les robots mobiles

1.2.1 Présentation générale sur les robots mobiles

Un robot est un dispositif mécanique électronique informatique. Leur intérêt réside dans leur mobilité qui ouvre des applications dans de nombreux domaines. Comme les robots manipulateurs, ils sont destinés à assister l'homme dans les tache pénibles (transport de charges lourdes), monotones ou en ambiance hostile (nucléaire, marine, surveillance, spatiale, lutte contre l'incendie) [1].

L'aspect particulier de la mobilité impose une complexité technologique et méthodologique qui s'ajoute en général aux problèmes rencontrés par les robots manipulateurs. La résolution de ces problèmes passe par l'emploi de toutes les ressources disponibles tant au niveau technologique (capteurs, motricité, énergie) qu'à celui du traitement des informations par l'utilisation des techniques de l'intelligence artificielle ou de processeurs particuliers (vectoriel, cellulaires).

L'autonomie du robot mobile est une faculté qui lui permet de s'adapter ou de prendre une décision dans le but de réaliser une tâche malgré un manque d'informations préliminaires ou

éventuellement erronées. Dans d'autres cas d'utilisation, comme celui des véhicules d'exploration de planètes, l'autonomie est un point fondamental puisque la télécommande est alors impossible par le fait de la durée du temps de transmission des informations.

1.2.2 Classification des Robots Mobiles

La classification des robots mobiles se fait suivant plusieurs critères (degré d'autonomie, système de locomotion, énergie utilisée...) [1].

La classification la plus intéressante et la plus utilisée est selon leur degré d'autonomie. Un robot mobile autonome est un système automoteur capable d'adapter son comportement à l'environnement. On dit qu'un robot mobile est autonome s'il vérifie ces deux conditions : la versatilité et l'auto adaptabilité.

La versatilité signifie que le robot devrait être capable d'effectuer une variété des tâches, ou la même tâche de différente manière.

L'auto adaptabilité veut dire que le robot devrait être capable d'accomplir correctement sa tâche, même s'il rencontre de nouvelles situations inattendues, sans intervention humaine.

On peut citer quelques types des robots mobiles autonomes :

- Robot télécommande par un opérateur Ces robots sont commandés par un opérateur qui leur impose chaque tâche élémentaire à réaliser.
- Robot télécommandé au sens de la tâche à réaliser. Le robot contrôle automatiquement ses actions.
- Robot semi-autonome Ce type de Robot réalise des tâches prédéfinies sans l'aide de l'opérateur.
- Robot autonome Ces derniers réalisent des tâches semi-définies.

Les principaux problèmes particuliers liés à la conception de tels robots sont :

- . La conception mécanique liée à la mobilité.
- . La détermination de la position et de l'orientation.
- . La détermination du chemin optimal.

1.2.3 Applications

Le domaine d'application des robots mobiles est vaste, nous présentons quelques applications dans le tableau suivant (Figure1.5) et sur (Tableu1.1).

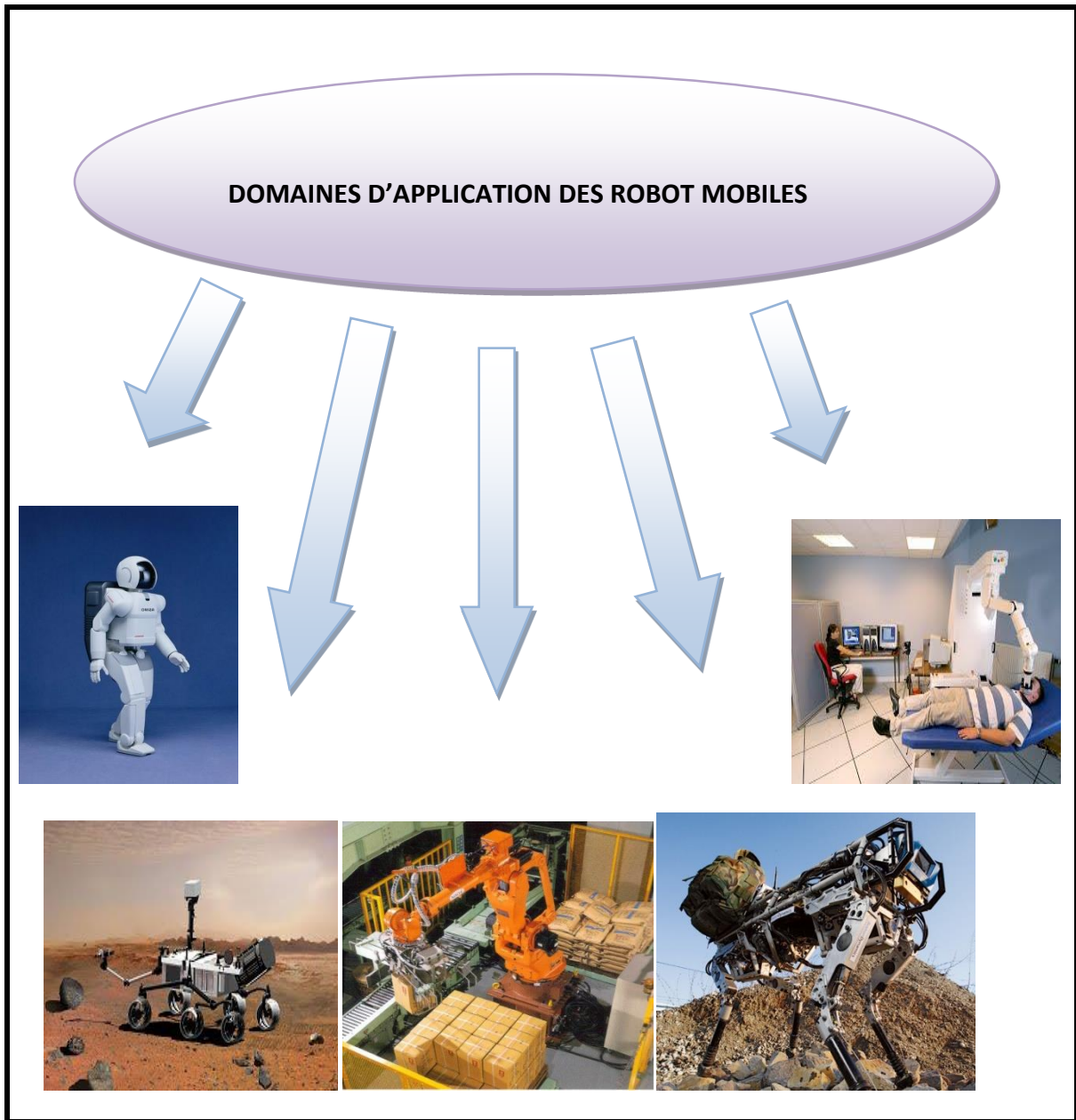


Figure 1.5 : Quelques exemples d'application des robots mobiles.

Industrie nucléaire :	<ul style="list-style-type: none">- surveillance de sites- manipulation de matériaux radioactifs- démantèlement de centrales
Sécurité civile	<ul style="list-style-type: none">- neutralisation d'activité terroriste- déminage- pose d'explosif- surveillance de munitions
Militaire	<ul style="list-style-type: none">- surveillance, patrouille- pose d'explosifs- manipulation de munitions
Chimique	<ul style="list-style-type: none">- surveillance de site- manipulation de matériaux toxiques
Médecine	<ul style="list-style-type: none">- assistance d'urgence- aide aux handicapés physiques, aux aveugles
Lutte contre l'incendie	<ul style="list-style-type: none">- assistance d'urgence- aide aux handicapés physiques, aux aveugles
Sous-marine	<ul style="list-style-type: none">- pose de câbles- recherche de nodules- recherche de navires immergés- inspection des fonds marins
Agricole	<ul style="list-style-type: none">- cueillette de fruits- traite, moisson, traitement des vignes...
Construction BTP	<ul style="list-style-type: none">- projection mortier- lissage du béton
Espace	<ul style="list-style-type: none">- exploration
Nettoyage	<ul style="list-style-type: none">- coque de navire- nettoyage industriel
Industriel	<ul style="list-style-type: none">- convoyage- surveillance

Tableau1.1 : Les domaines d'utilisation des robots mobiles.

1.2.4 Les avantages et les inconvénients de robot

A. Les avantages

- ✓ Les robots ont de nombreux avantages. Exemple : productivité, sécurité, fiabilité et rentabilité.
- ✓ Pour les entreprises, les robots industriels sont à l'image du salarié parfait. Ils travaillent rapidement et peuvent réaliser des tâches répétitives tout au long de la journée, sans manger, sans pause, sans aucune fatigue, sans erreur, sans maladie, mais aussi pendant la nuit, dans n'importe quelle condition de travail.
- ✓ Les robots ils n'ont pas besoin de formation, les mises à jour logicielles ou matérielles sont faciles.
- ✓ Les robots ils n'auront pas besoin du salaire ni d'une retraite, ni invalides.
- ✓ Les robots ont la possibilité d'embarquer des charges très lourdes et la possibilité de déplacements à très haute vitesse.

B. Les inconvénients

- ✓ Un robot peut également être mal programmé et peut aussi tomber en panne ce qui peut également avoir des conséquences dans la production et ralentir le fonctionnement de l'entreprise.
- ✓ Le prix d'un robot, la plupart des robots coutent plusieurs milliers de dollars, donc beaucoup d'entreprises n'ont pas les moyens d'en acheter.
- ✓ Acheter un robot revient à licencier un ou plusieurs ouvriers et diminue le nombre de postes disponibles dans une entreprise se qui cause de chômage en quelque sorte.

1.2.5 Architecture et les composants des robots mobiles

A. Architecteur des robots mobiles

L'architecture des robots mobiles se structure en quatre éléments :

- ✓ La structure mécanique et la motricité
- ✓ Les organes de sécurité
- ✓ Le système de traitement des informations et gestion des tâches.
- ✓ Le système de localisation.

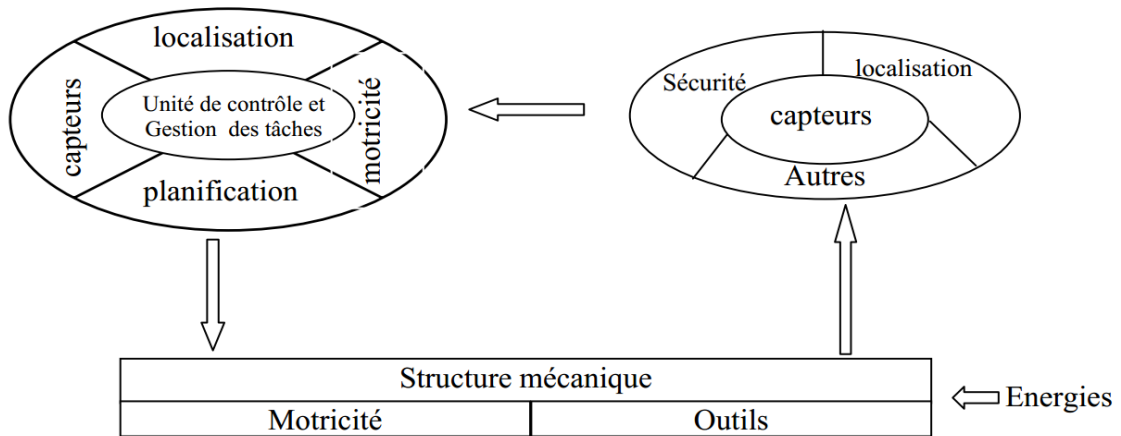


Figure 1.6 : Architecture d'un robot mobile

A.1 La structure mécanique et la motricité

Locomotion est le nom collectif pour les différentes méthodes que les robots utilisent pour se transporter d'un endroit à l'autre, objectif majeur dans locomotion des robots est dans le développement des capacités pour les robots de décider comment, quand et où se déplacer de manière autonome.

A.1.1 Les robots mobiles à roues

La mobilité par roues est la structure mécanique la plus communément appliquée. Cette technique assure l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importantes. Le franchissement d'obstacles ou l'escalade de marches d'escaliers est possible. On trouve plusieurs types des roues :

- 1) Robots à roues différentielles : Un robot à roues différentiel est un robot mobile dont le mouvement est basé sur deux roues entraînées séparément placées de part et d'autre du corps du robot. On peut ainsi changer de direction en faisant varier la vitesse relative de rotation des roues et donc ne nécessite pas de mouvement de braquage supplémentaire.

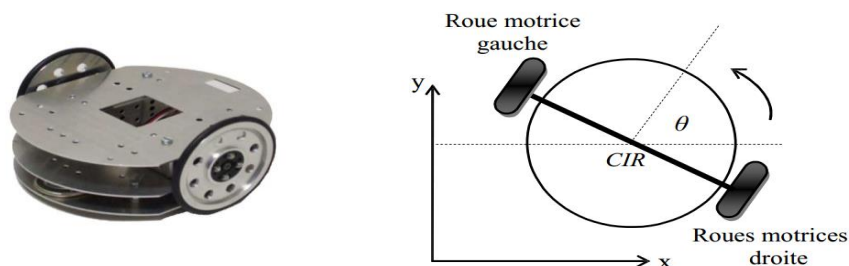


Figure 1.7 : Robots à roues différentielles

- 2) Robots de type « tricycle » : équipés d'un essieu arrière fixe muni de deux roues non orientables et d'une roue avant centrée orientable.

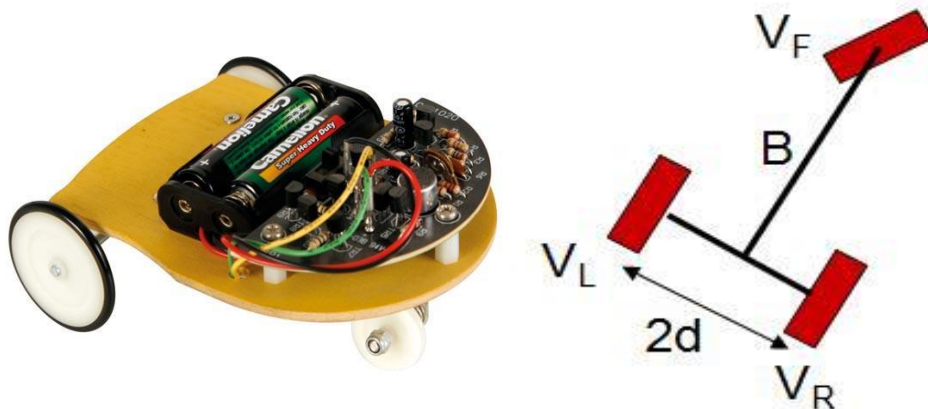


Figure 1.8 : Robot tricycle

- 3) Robots de type « voiture » : essieu arrière non orientable muni de deux roues non orientables et libres en rotation et deux roues avant centrées orientables.

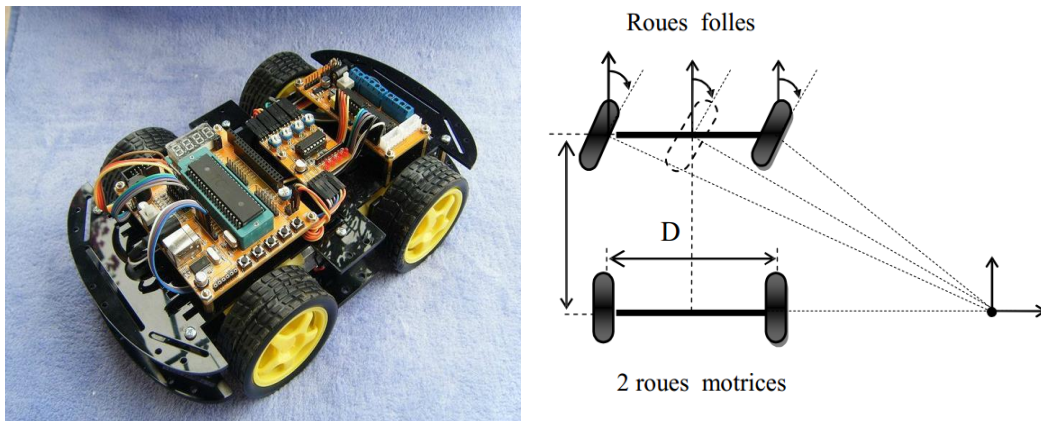


Figure 1.9 : Robot Voiture

- 4) Robots à roues omnidirectionnelles : c'est un robot qui peut se déplacer librement dans toutes les directions. Il est en général constitué de trois roues décentrées orientables placées en triangle équilatérale

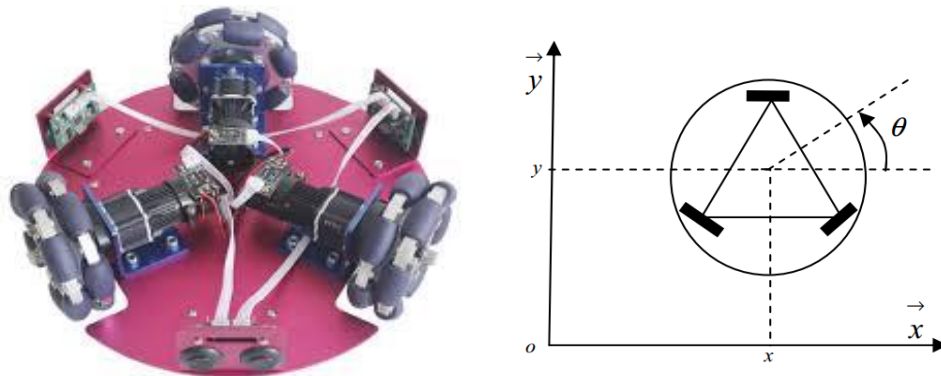


Figure 1.10 : Robot mobile omnidirectionnelles

A.1.2 Les robots mobiles à chenilles

L'utilisation des chenilles présente l'avantage d'une bonne adhérence au sol et d'une faculté de franchissement d'obstacles. L'utilisation est orientée vers l'emploi sur sol accidenté ou de mauvaise qualité au niveau de l'adhérence (présence de boue, herbe...).



Figure 1.11 : Robot mobile à Chenilles

A.1.3 Les robots mobiles marcheurs

Les robots mobiles marcheurs sont destinés à réaliser des tâches variées dont l'accès au site est difficile, dangereux ou impossible à l'homme. Leur anatomie à nombreux degrés de liberté permet un rapprochement avec les robots manipulateurs. La locomotion est commandée en termes de coordonnées articulaires. Les méthodes de commande des articulations définissent le concept d'allure qui assure le déplacement stable de l'ensemble. Les différentes techniques étudiées se rapprochent de la marche des animaux et notamment de celle des insectes.

L'adaptation au support est un problème spécifique aux marcheurs. Il consiste à choisir le meilleur emplacement de contact en alliant l'avance et la stabilité avec l'aide de capteurs de proximité, de contact ou de vision. Parmi les robots marcheurs on a :

1) Robots humanoïdes :

Un robot humanoïde ou androïde est un robot dont l'apparence générale rappelle celle d'un corp humain. Généralement les robots humanoïdes ont un torse avec une tête, deux bras et deux jambes, bien que certains modèles ne représentent qu'une partie du corps, par exemple à partir de la taille certains robots humanoïdes peuvent avoir un visage avec des yeux et une bouche.

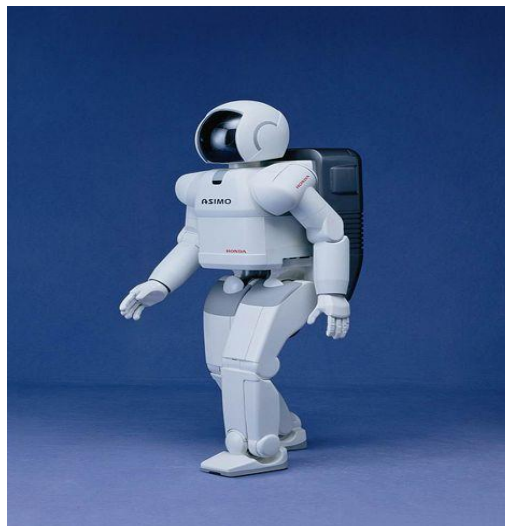


Figure 1.12: Robot hexapode (Asimo)

2) Robot hexapode :

Un robot hexapode est un dispositif mécatronique à base fixe ou mobile dont la locomotion est fondée sur trois paires de pattes, les hexapodes sont considérés plus stables que les robots bipèdes du fait que dans la plupart des cas, les hexapodes sont statiquement stables. De ce fait, ils ne dépendent pas de contrôleurs en temps réel pour rester debout ou pour marcher.

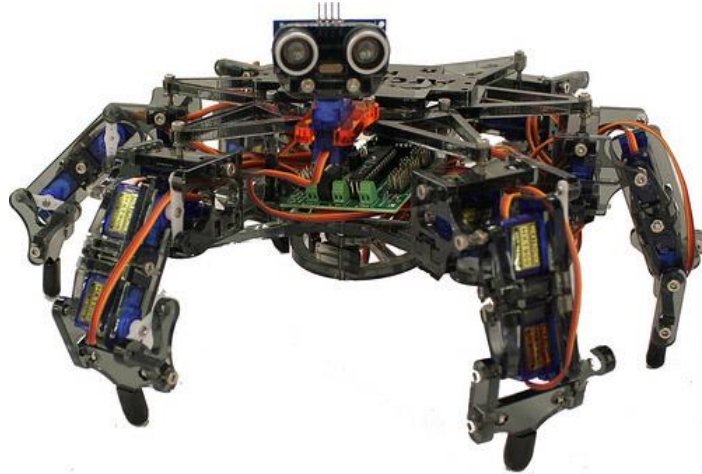


Figure 1.13 : Robot hexapode

A.1.4 Les robots mobiles rampants

Les robots rampants utilisent des méthodes de locomotion des animaux rampants comme les serpents, ils sont utilisés dans des environnements de type tunnel ou des endroits restreints, leur système est composé d'un ensemble de modules ayant plusieurs mobilités.



Figure 1.14 : Robot Rampant

A.1.5 La motricité et l'énergie

Les déplacements des robots sont réalisés par des moteurs de types électrique, thermique ou hydraulique.

Chapitre1 Généralités sur la robotique mobile

L'énergie électrique la plus fréquemment employée offre l'avantage d'une commande aisée. Par contre le transport et la génération présentent des difficultés. Plusieurs méthodes sont employées :

- ✓ Par batteries qui sont soit rechargées périodiquement de manière automatique ou manuelle, soit par un échange avec d'autres lorsqu'elles sont déchargées.
- ✓ Par groupe électrogène embarqué dont l'inconvénient constitue la masse élevée. L'énergie de base est alors thermique.
- ✓ Par cordon ombilical qui réduit l'autonomie du robot.

L'énergie thermique est essentiellement employée par des véhicules de forte puissance comme énergie de base pour la traction ou pour activer un compresseur hydraulique.

A.2 Les organes de sécurité

Il est mieux de sécuriser le robot pour éviter la collision ou prendre des chocs dans l'environnement où il navigue. Donc il est obligatoire qu'il soit doté d'organes garantissant la sécurité de robot. Des capteurs sont disponibles tout autour du mobile afin de détecter un obstacle sur un domaine le plus étendu possible. Deux types de capteurs sont employés :

- ✓ les capteurs proxy-métriques assurant la détection avant collision (ultra-son, infrarouge...)
- ✓ les capteurs de contact détectant une collision ou un choc avec l'environnement (contact électrique sur pare-chocs, résistance variable, fibre optique...).

L'organisation de la sécurité est représentée sur le schéma de la figure 1.15

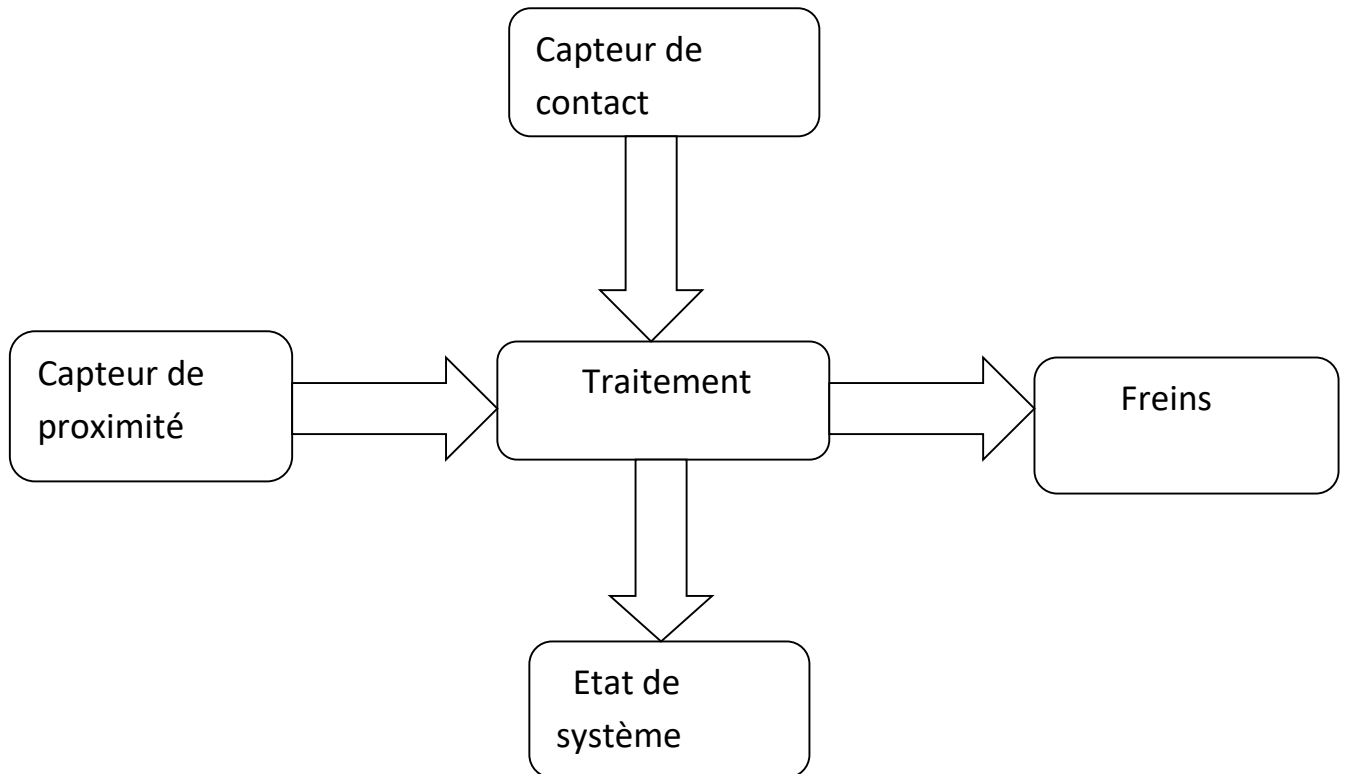


Figure 1.15 : Synoptique de la sécurité

Si le robot détecte un obstacle, la stratégie consiste soit à immobiliser le robot jusque l'obstacle s'éloigne, soit à ralentir le mouvement si l'obstacle n'est pas très proche, soit à choisir un autre chemin pour éviter l'obstacle.

A.3 Traitement des informations et gestion des tâches

L'ensemble de traitement des informations et gestion des tâches constitue le module information central qui établit les commandes permettant au mobile de réaliser un déplacement et d'activer les divers organes en accord avec l'objectif.

A.4 Le système de localisation

Le système de localisation est l'ensemble constitué par les capteurs et les logiciels de traitement de données utilisé par le robot pour estimer de manière autonome son déplacement ou sa situation dans l'espace.

Les techniques de localisation employées sont de deux types :

- ✓ La localisation relative, basée sur l'utilisation de capteurs proprioceptifs.
- ✓ La localisation absolue, basée sur l'utilisation de capteurs extéroceptifs.

B. Les composants fonctionnels de robot

Le fonctionnement de système d'un robot est basé sur les composants fonctionnels qui lui constituent. Les composants fonctionnels sont [5] :

Mécanismes : Système de corps articulés disposant d'une haute mobilité.

Actionneurs : Générateurs d'actions mécaniques pour mettre le mécanisme en mouvement.

Un convertisseur d'énergie primaire en énergie mécanique (actionneurs électriques, pneumatiques, hydrauliques, piézoélectriques, électrostatiques...).

Capteurs : Mesure de l'état du système mécanique et extraction d'informations

Relatives à la tâche et l'environnement dans lequel elle est réalisée. On a des capteurs proprioceptifs (position, vitesse, accélération), capteurs extéroceptifs (force, vision, télémétrie, tactile, sonores . . .).

Système de contrôle : Ensemble électronique pour la commande des actions à partir D'informations capteurs.

Interface de programmation : Système de communication entre l'opérateur et la Machine ou entre les machines (Interface graphique, interfaces de télé-opération, Langage structure, etc. ...).

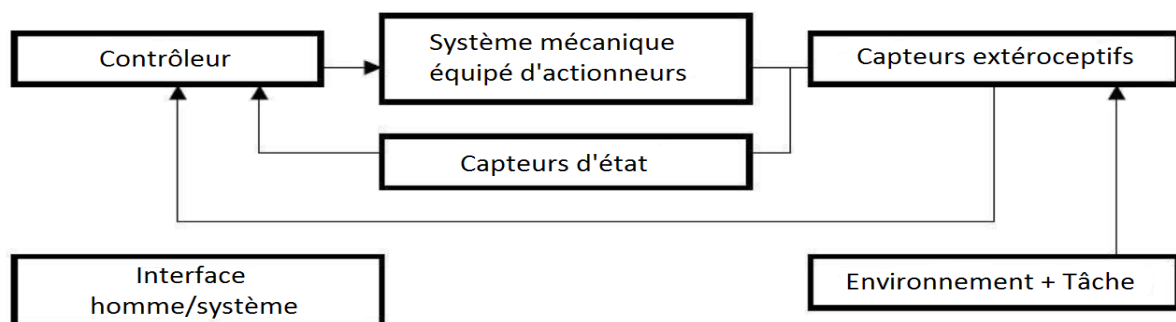


Figure 1.16 : Architecture générale d'un système robotique

1.2.6 La robotique et l'intelligence artificielle

La robotique est une branche de l'intelligence artificielle concernée par l'étude des systèmes automatiques capables d'interagir directement avec le monde physique. C'est une automatisation de ses machines, ou l'objectif est d'augmenter les capacités de localisation et de navigation dans son espace de travail. [3]

A. Intelligence Artificielle (IA)

L'intelligence artificielle est la reproduction, par des moyens artificiels de toutes les formes de l'intelligence humaine pour un objectif final, qui s'intéresse à la conception des systèmes intelligents. L'IA est la capacité à raisonner, à apprendre, et à s'adapter face à de nouveaux changements qu'ils sont des éléments principaux que nous recherchons.

B. Les outils d'exécution de l'IA dans le robot

L'intelligence Artificielle d'un robot se résume à un ensemble de programmes préalablement écrits avec un ordinateur [4] :

- ✓ les programmes s'exécutant sur les robots sont écrits avec un langage de programmation (exemples : C++, Java, python, ...).
- ✓ ils s'exécutent grâce au contrôleur et à la mémoire du robot.
- ✓ ils prennent en entrée les informations obtenues par les capteurs et en sortie envoient des ordres aux effecteurs.

C. Types de programmes d'IA utilisées dans la robotique

L'intelligence Artificielle d'un robot permet par exemple :

- ✓ Analyse d'images.
- ✓ Localisation et navigation.
- ✓ Gestion des interactions : interfaces, communication...
- ✓ Planification et prise de décision. Contrôle d'exécution des tâches.

1.3 Conclusion

Dans ce chapitre, on a donné un aperçu général sur la robotique : l'histoire de la robotique, les définitions d'un robot. On a également parlé sur les robots mobiles, leurs classifications, le domaine d'application, les avantages et l'inconvénient des robots mobiles. Ensuite, on a cité l'architecture et les composants fonctionnels de système pour le robot mobile, on a parlé à la fin sur l'utilisation de l'intelligence artificielle en robotique, ce qui va nous servir à la construction de notre robot mobile.

Chapitre 2

Description de la partie matérielle du robot

2.1 Introduction

La conception est une étape cruciale dans notre travail qui consiste à la création de notre propre robot mobile, il est composé de deux parties essentielles, la partie mécanique et la partie électronique qui assure les mouvements du robot ainsi que d'autres tâches. La partie électronique c'est la partie de la commande qui donne des ordres et fait le traitement d'information, et la partie mécanique c'est la partie qui recevoir les ordres et passe on action pour réaliser des tâches prédéfinie.

Dans ce chapitre, on va présenter les différents outils utilisés pour la réalisation du robot.

2.2 Partie mécanique et mobile du robot :

2.2.1 Structure :

La structure du robot est entièrement composée d'une plateforme plane en matière de la lico qui permet une stabilité mécanique grâce à la simplicité du calcul de centre de gravité du système. Le robot possède un corps de hauteur 30 cm et de largeur 30 cm et longueur 25 cm (*Voir la figure 2.1 ci-dessous*). Elle support tous les composants du robot : la carte de puissance, 2 cartes Arduino méga, 3 roues, 2 roues en arrière et 1 roue en avant, 2 encodeurs optiques intégrés dans les moteurs pour la mesure de vitesse. Le système de ramassage de gobelet compte sur deux servomoteurs qui vont pincer ou lâcher, soulever ou abaisser le gobelet.

Chapitre2 Description de la partie matérielle du robot

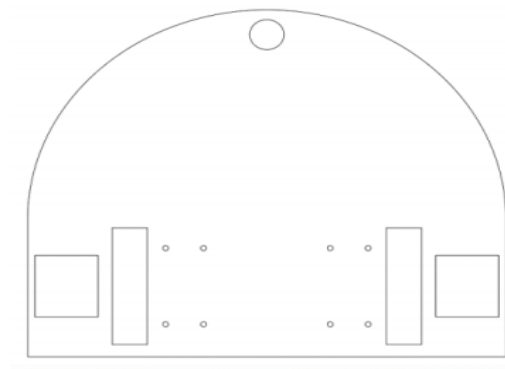


Figure 2.1 : Plateforme (châssis)

2.2.2 Partie motrice

A. Les roues

Pour assurer le mouvement du robot sur le terrain, on a utilisé trois roues, deux en arrière sont identiques, elles ont été fabriquées en aluminium de 10cm de diamètre et 1.5cm largeur couvertes d'une couche procurant une certaine adhérence et facilitant le déplacement et pour éviter le glissement, ces dernières sont reliées directement sur l'axe du moteur qui va assurer une liaison rigide qui permet de minimiser les erreurs entre le mouvement du moteur et de la roue. (Figure 2.2).

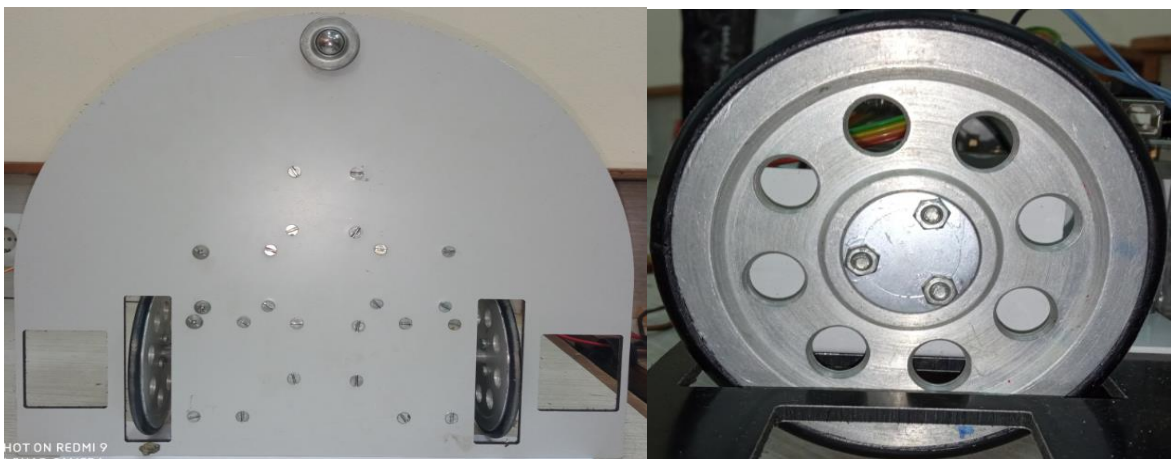


Figure 2.2 : Image en position vertical de trois roues du robot.

Chapitre2 Description de la partie matérielle du robot

B. Roue à bille

Pour améliorer le mouvement du robot et minimiser l'effet du frottement, nous avons opté pour l'utilisation des billes roulantes (*Figure 2.3*), fabriquée d'une manière à assurer un mouvement en 360° sur un plan Oxy. Cette caractéristique permet au robot de se déplacer librement sur toutes les directions sans aucune difficulté.



Figure 2.3 : Roue à bille

C. Moteurs DC (MCC)

Un moteur à courant continu est constitué de deux parties électriques : le stator et le rotor. Lorsqu'on alimente le moteur, il se crée une interaction magnétique qui met le moteur en mouvement. Lorsqu'on inverse le sens de la tension qui alimente le moteur, il tourne en sens inverse. Pour les deux roues du robot mobile, on a utilisé des moteurs DC de type EMG30 (*Figure 2.4*).

L'EMG30 (moteur, codeur, boîte de vitesse 30 :1) est un moteur 12v entièrement équipé de codeur et d'un réducteur de vitesse 30 :1. Il est idéal pour les applications robotique petites ou moyennes, offrant un entraînement et un retour d'informations économiques pour l'utilisateur.

Il comprend également un condensateur de suppression de bruit standard sur les enroulements du moteur.

Chapitre2 Description de la partie matérielle du robot

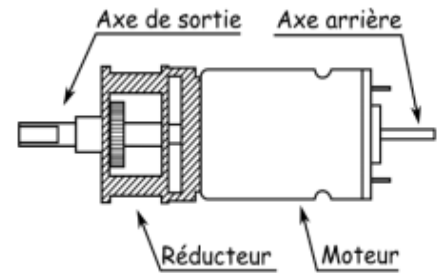
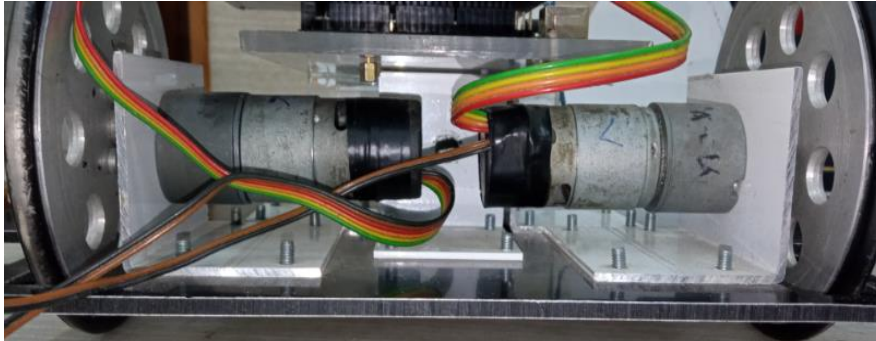


Figure 2.4 : Schéma d'un moteur type EMG30

Tension nominale	12 V
Couple nominale	1.5 Kg/cm
Vitesse nominale	170 tr/min
Courant nominal	530 mA
Pas de vitesse de chargement	216
Courant à vide	150 mA
Courant de décrochage	2.5 A
Puissance nominale	4.22 W
Nombre de codeurs par tour de sortie	360

Tableau 2.1 : Caractéristiques du moteur DC de type EMG30

2.3 Système de ramassage de gobelet (gripper)

2.3.1 Fonctionnement du mécanisme :

Avant d'entamer le principe de notre mécanisme, nous tenons à préciser que, pour réaliser n'importe quelle pièce ou élément, nous devons tout d'abord définir le matériau que nous utiliserons, quelles sont ses propriétés et est ce qu'il est réalisable et fonctionnelle aux même temps !

Le choix de matériau étant fait, on calcule ses dimensions et choisissant leur forme qui ressemble à deux pignons, l'un appelé pignon motrice fixé avec l'arbre servomoteur, c'est celle qui entraîne le deuxième pignon réceptrice, qui va générer un mouvement de rotation pour l'ouverture et la fermeture.

Chapitre2 Description de la partie matérielle du robot

La simulation de ce mécanisme est faite dans un programme appelé 'Builder' et imprimé à l'aide d'une imprimante 3 dimensions.

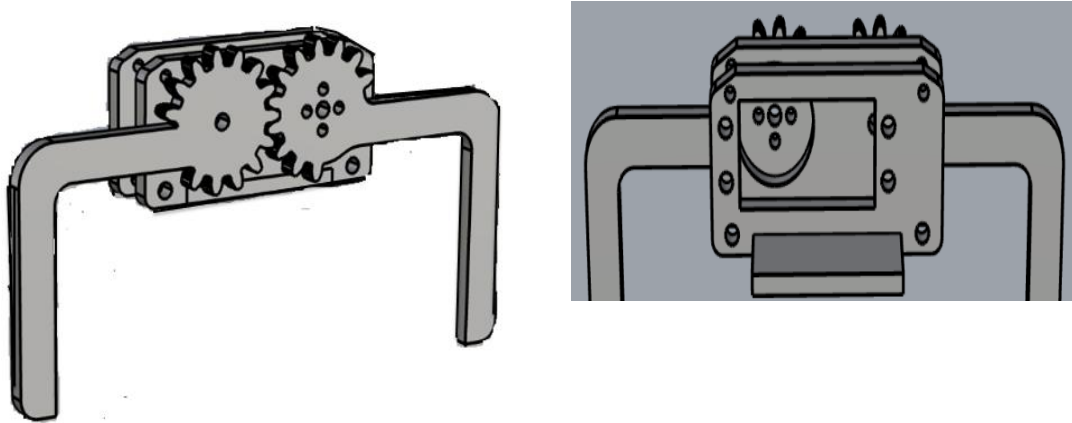


Figure 2.5 : La forme de notre gripper

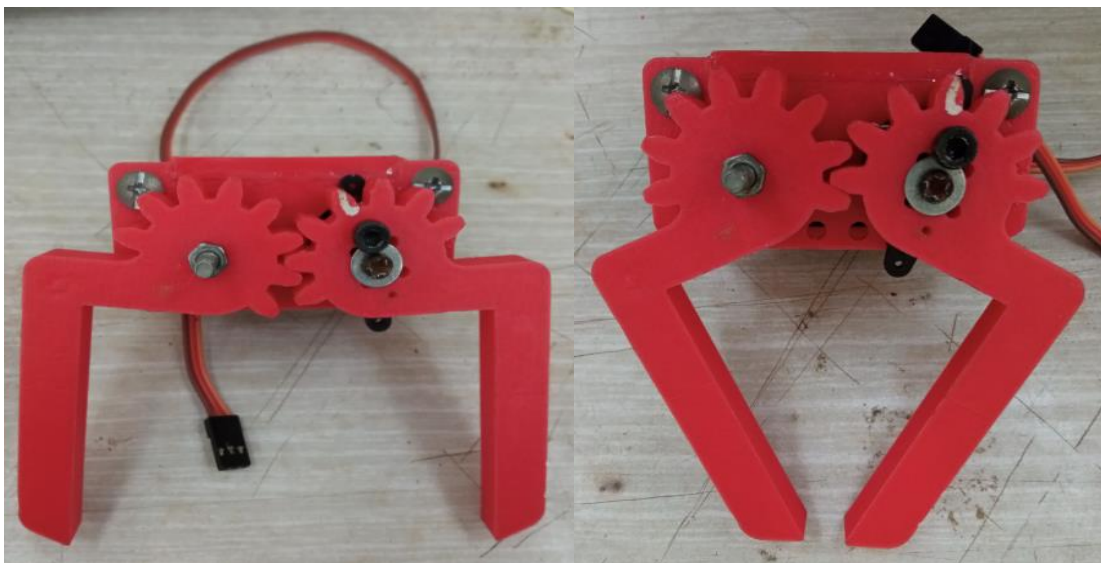


Figure 2.6 : les différentes positions de notre gripper

A la fin, notre robot doit ramasser le gobelet, pour ce faire, on a ajouté le mécanisme vue précédemment (figure 2.6). Le mécanisme est fixé sur une plaque de lico verticale, cette dernière est liée à une autre plaque horizontale placée à l'avant du robot. Notre but est de garantir le mouvement de la montée, la descente, l'ouverture et la fermeture.

Chapitre2 Description de la partie matérielle du robot

2.4 Partie électronique utilisé

Pour la mise en marche tous les systèmes mécaniques vus dans la partie précédente, notre robot doit disposer des certains éléments électroniques qui permettent de commander ces systèmes.

2.4.1 Source d'alimentation

Dans notre robot la source d'alimentation est une batterie de 12 Volts et 5.4 AH.

2.4.2 Carte de puissance

C'est la carte dans laquelle on trouve tous les composants électroniques utilisés pour cette réalisation. Parmi les plus importants, on trouve ce que l'on appelle un « pont en H ».

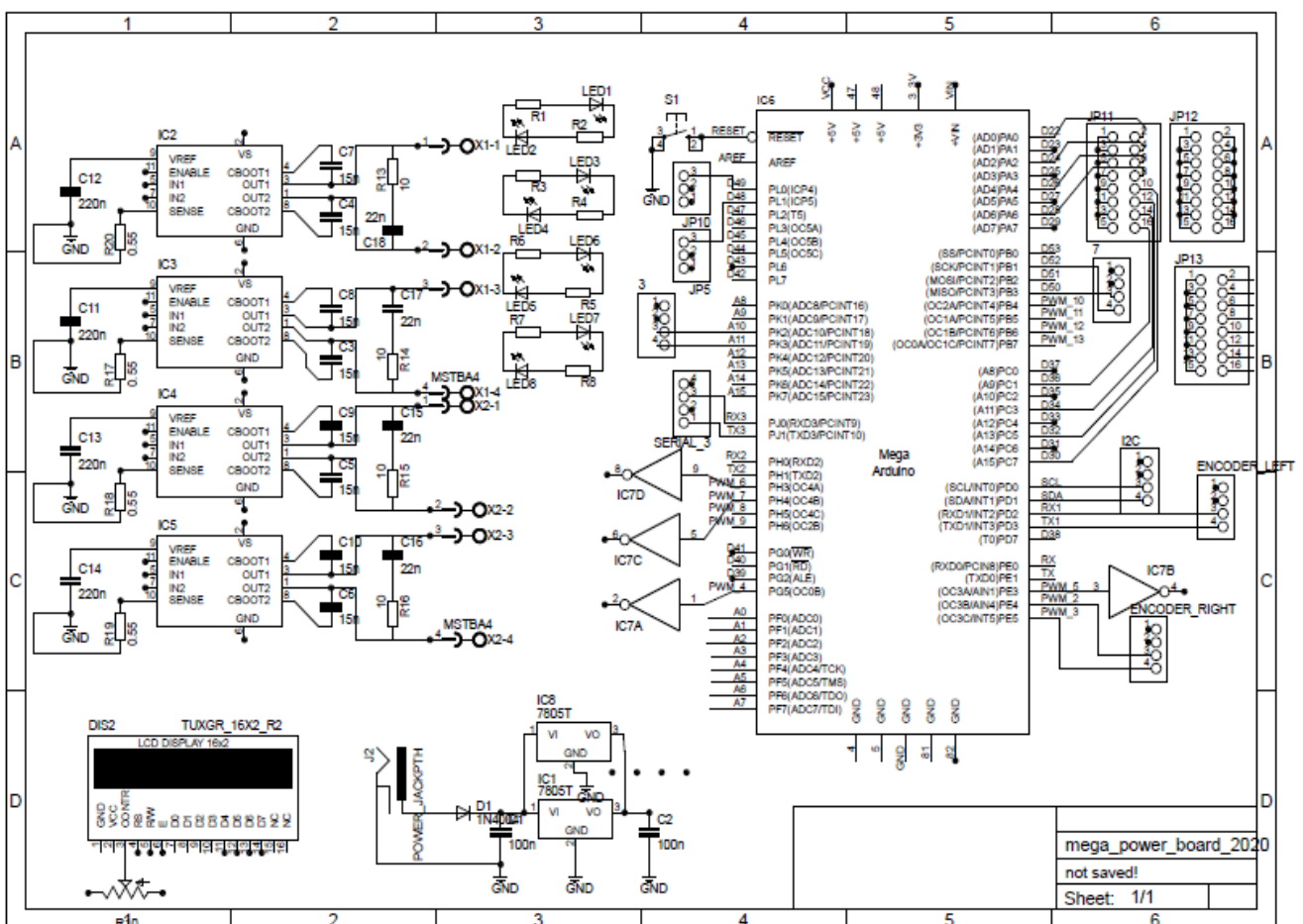


Figure 2.7 : Schéma électrique des composants de la carte de puissance

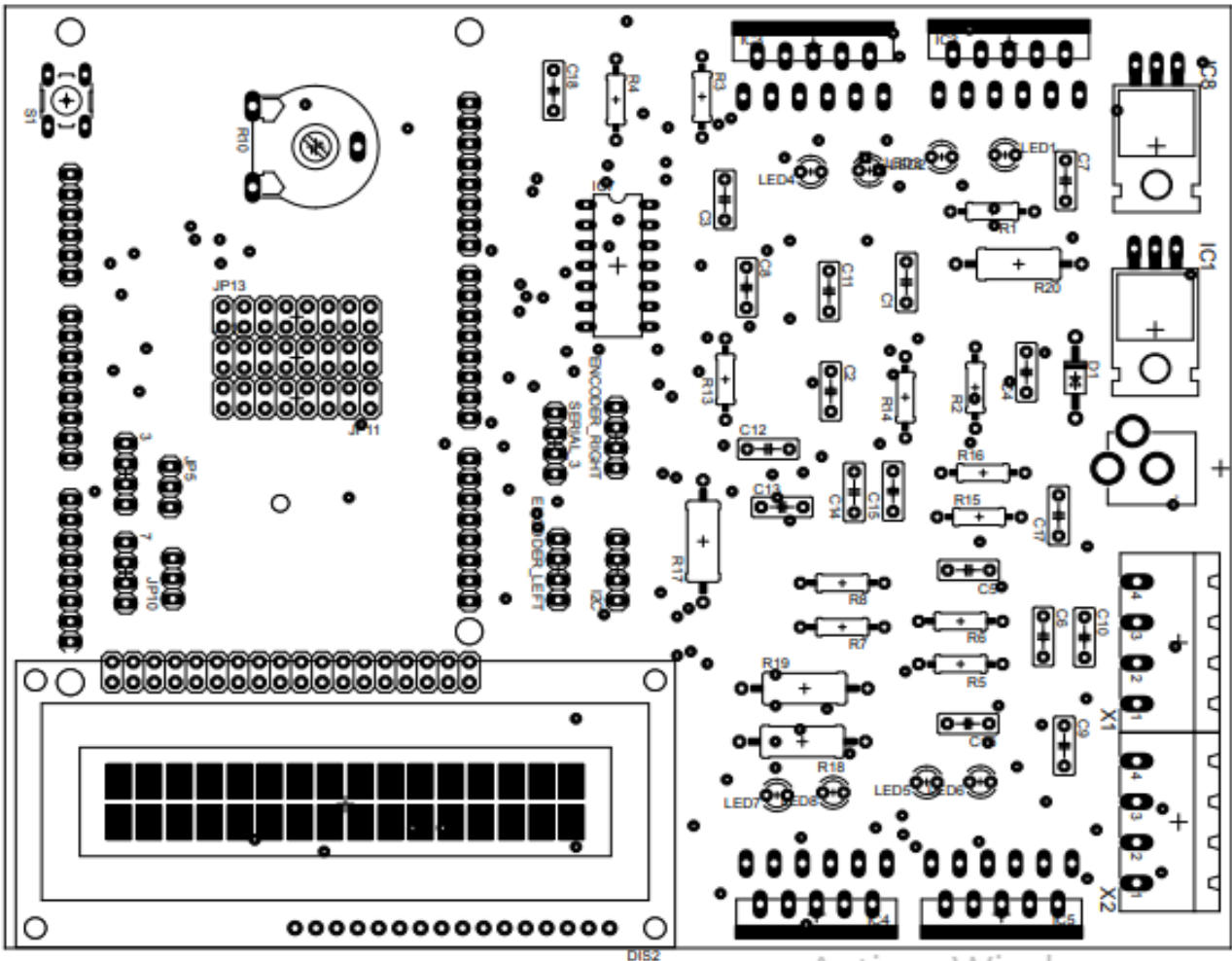


Figure 2.8 : Schéma d'implantation des composants de la carte de puissance

A. Pont en H

Grâce au pont en H, on arrive à faire tourner le moteur dans les deux sens (en continuant de gérer sa vitesse).

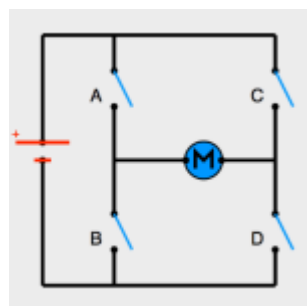


Figure 2.9 : Le pont en H

Chapitre2 Description de la partie matérielle du robot

Voici ce qu'on appelle un « pont en H », il tient son nom de la forme en H du circuit autour du moteur.

- **Principe de fonctionnement** : Des interrupteurs fonctionnent deux par deux. Le A est associé au D et le B est associé au C. Dans le schéma ci-dessus, rien ne se passe car tous les interrupteurs sont ouverts (ils ne laissent pas passer le courant). Le moteur est arrêté. Voyons maintenant ce qui arrive lorsqu'on actionne en même temps les interrupteurs A et D (schéma de gauche), ou les interrupteurs B et C (schéma de droite) :

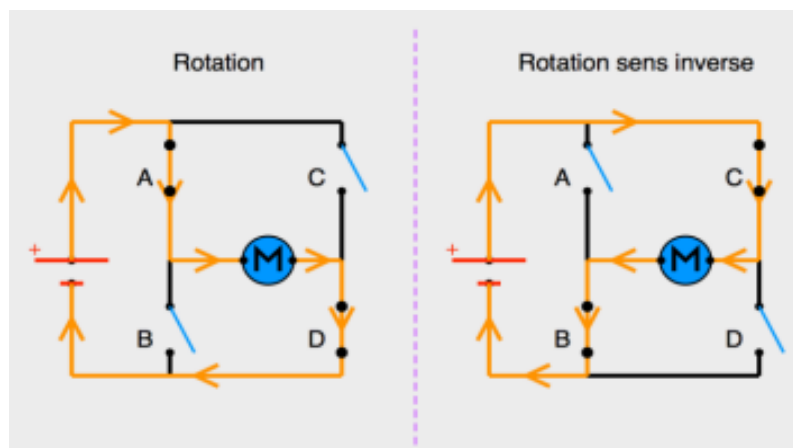


Figure 2.10 : Sens du courant en fonction de l'état des interrupteurs

On peut voir donc que :

- Sur le schéma de gauche : les interrupteurs A et D sont fermés. Le courant entre par la patte gauche du moteur et sort par sa droite. Le moteur tourne.
- Sur le schéma de droite : les interrupteurs B et C sont fermés. Le courant entre par la patte droite du moteur et sort par sa gauche. Le moteur tourne donc dans le sens inverse ! Un moteur lorsqu'il est en roue libre (c'est-à-dire qu'il tourne à cause de sa force d'inertie mais pas à cause du courant), il génère un courant. Ce courant peut être utilisé dans le pont en H comme frein électromagnétique. Le moteur s'envoie son propre courant à l'envers. Cela permet de contrôler l'arrêt du moteur plutôt que de le

Chapitre2 Description de la partie matérielle du robot

laisser en roue libre. On peut réaliser un pont en H avec des transistors (NPN) et des diodes, comme suit :

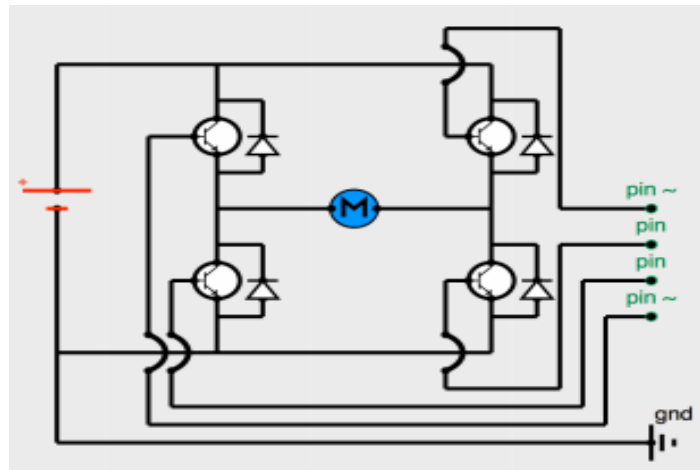


Figure 2.11 : Pont en H avec transistors NPN, diodes et entrées Arduino

Mais ce n'est pas la meilleure solution car cela fait beaucoup de connexions pour commander un seul moteur DC au final. Les moteurs DC sont très utilisés en robotique, en particulier pour les déplacements. Du coup, pour nous faciliter la tâche, une puce a vu le jour dans le monde de l'électronique. Dans ce travail, on a utilisé le L6203



Figure 2.12 : L6203 H bridge

Chapitre2 Description de la partie matérielle du robot

2.4.3 Codeur incrémental à effet Hall

Il est constitué de :

- Un aimant solidaire du rotor du moteur
- Capteurs magnétiques à effet Hall produisant au total 360 impulsions pour chaque rotation complète du rotor. Ils sont situés sur le stator du moteur et détectent le champ magnétique tournant de l'aimant fixe sur le rotor.
- Un circuit électronique d'alimentation des constituants et de traitement (le signal est amplifié puis converti en signal carré).



Figure 2.13 : Codeur incrémental

Couleur de fils	Connexion
Violet (1)	Capteur Hall B V_{out}
Bleu (2)	Capteur Hall A V_{out}
Vert (3)	Capteur Hall terre
Marron (4)	Capteur Hall V_{cc}
Rouge (5)	(+) Moteur
Noire (6)	(-) Moteur

Tableau 2.2 : Connecteurs du codeur

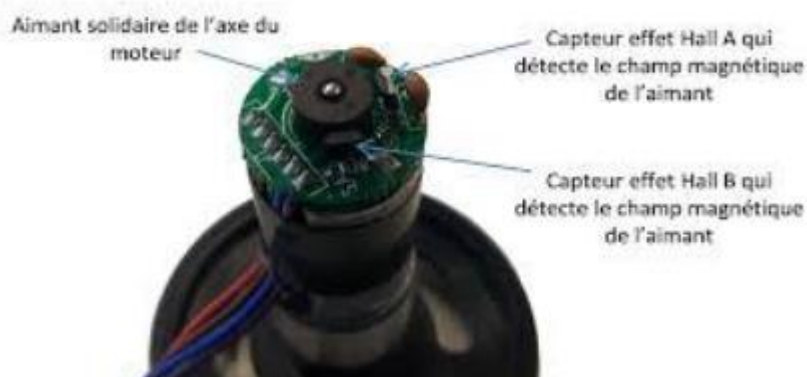


Figure 2.14 : Codeur incrémental à effet Hall

Chapitre2 Description de la partie matérielle du robot

2.4.4 La carte de commande Arduino méga

La carte Arduino Méga 2560 est basée sur un microprocesseur (ATMega2560) cadencé à 16 MHz. Elle dispose de 54 E/S dont 14 PWM, 16 analogiques et 4 UARTs (Universal Asynchronous Receiver Transmitter). Elle est idéale pour des applications exigeant des caractéristiques plus complètes que l'Uno.

Dans ce cas on a utilisé deux cartes Arduino méga (master, slave) pour le contrôle du robot mobile à l'aide d'une carte de puissance. (Comme montré sur la *Figure 2.15* ci-dessous)

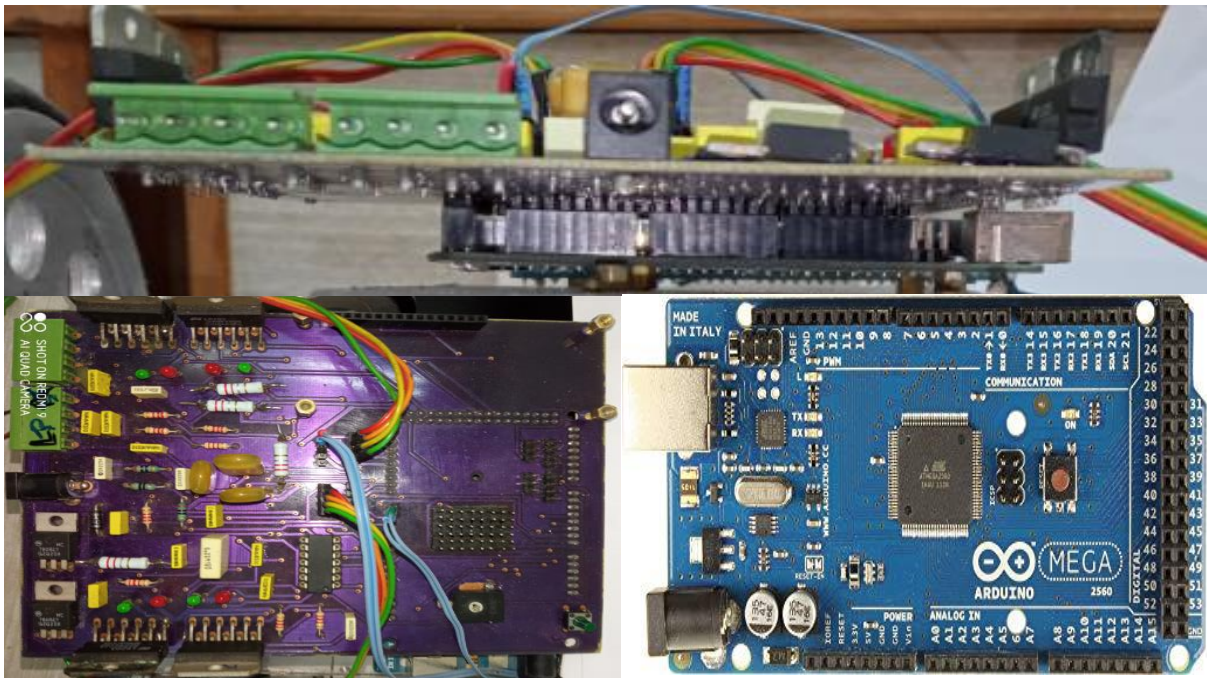


Figure 2.15 : La carte Arduino Méga associée à la carte de puissance

2.5 Transmission des données

2.5.1 la liaison entre les deux cartes via des Pins :(liaison série)

Le but est de faire s'échanger des informations ou des variables (int, float et long) d'une carte Arduino mega dite maitre à une autre carte Arduino mega dite esclave. Bien entendu la communication peut être bidirectionnelle.

Pour la configuration matérielle, il n'y a que 3 fils à connecter RX/TX et la masse dont la carte maitre Mega 2560 (pin 14(TX) et 15(RX)) raccordée en croisant les fils au port serial de la carte

Chapitre2 Description de la partie matérielle du robot

esclave Mega 2560 (pin 14(TX) et 15(RX)), la carte maitre est raccordée au PC par le port série standard pour contrôler la réception des valeurs sur le PC.

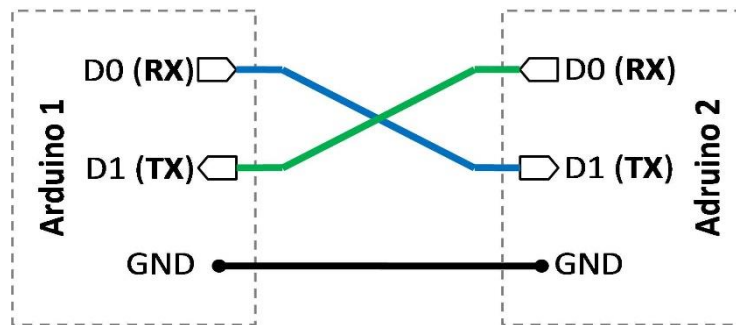


Figure 2.16 : Liaison série entre deux Arduino

Pour la liaison série, le protocole d'échange asynchrone est défini par l'envoi grâce à un UART, pour chaque caractère émis, de :

- Un bit de Start.
- Les 5 à 8 bits de données (bit de poids faible en premier).
- Eventuellement, un bit de vérification de parité qui permet de détecter des erreurs de transmission des 8 bits de données sur la ligne.
- Un, un et demi, ou deux bits de Stop après.



Figure 2.17 : Le protocole d'échange asynchrone

Chapitre2 Description de la partie matérielle du robot

2.6 Asservissement du robot mobile

2.6.1 Contrôle PID

L'algorithme de contrôle PID est utilisé pour le contrôle de presque toutes les boucles dans les industries de transformation, et il est également la base pour de nombreux algorithmes de contrôle et des stratégies avancées. L'équation d'un régulateur PID est donnée comme suit :

$$PID = K_p \times \text{erreur} + K_i \times \Sigma(\text{erreur} \times \Delta t) + K_d \times \frac{(\text{erreur} - \text{erreur}_{\text{précédente}})}{\Delta t}$$

- Le gain proportionnel assure une transmission instantanée du signal d'erreur (différence entre la consigne et la variable mesurée).
- Le comportement intégrateur à basse fréquence permet d'annuler l'erreur en régime permanent.
- Le comportement différentiel à haute fréquence permet d'améliorer la rapidité de la réponse du régime transitoire et limite les dépassements. Cependant, en cas de mesures bruitées, il peut contribuer à déstabiliser le système. Pour un fonctionnement correct des boucles de contrôle, la boucle PID doit être correctement réglée.

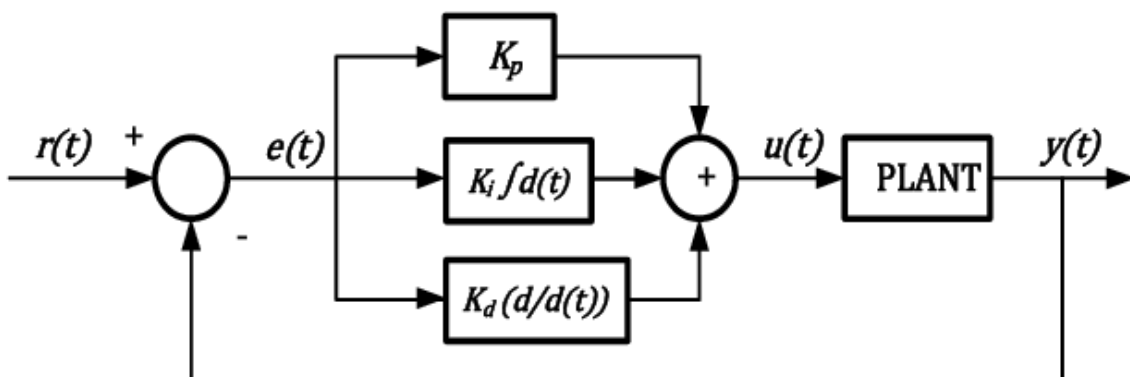


Figure 2.18 : La boucle PID

D'après les mesures obtenues, on a constatés que les valeurs des paramètres K_p , K_i , K_d de notre régulateur PID du moteur sont : $K_p= 0.073$, $K_i= 0.98$ et $K_d= 0.0004$

Chapitre2 Description de la partie matérielle du robot

- On a implémenté les résultats obtenue (K_p , K_i , K_d) dans notre programme Simulink (matlab) pour confirmer le comportement du model asservi, avec la vitesse de la consigne est de 1500 impulsion/seconde.

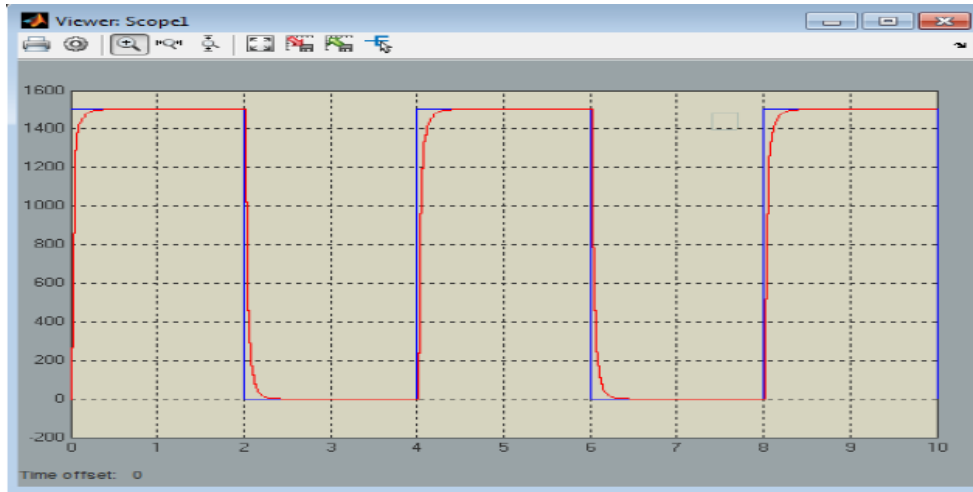


Figure 2.19 : Réponse moteur asservi par la commande PID utilisé

- La figure ci-dessus représente la réponse du moteur asservi par la commande PID utilisé, avec la vitesse de moteur en rouge et la consigne en bleu. On observe que la vitesse obtenue suit d'une manière très claire celle choisie comme consigne.
- Pour mieux vérifier nos résultats de simulation (K_p , K_i , K_d), on a les implémenté dans notre programme Arduino et on a téléversé ce programme vers la carte arduino méga pour visualise la stabilité et la robustesse de notre régulateur PID en réel.

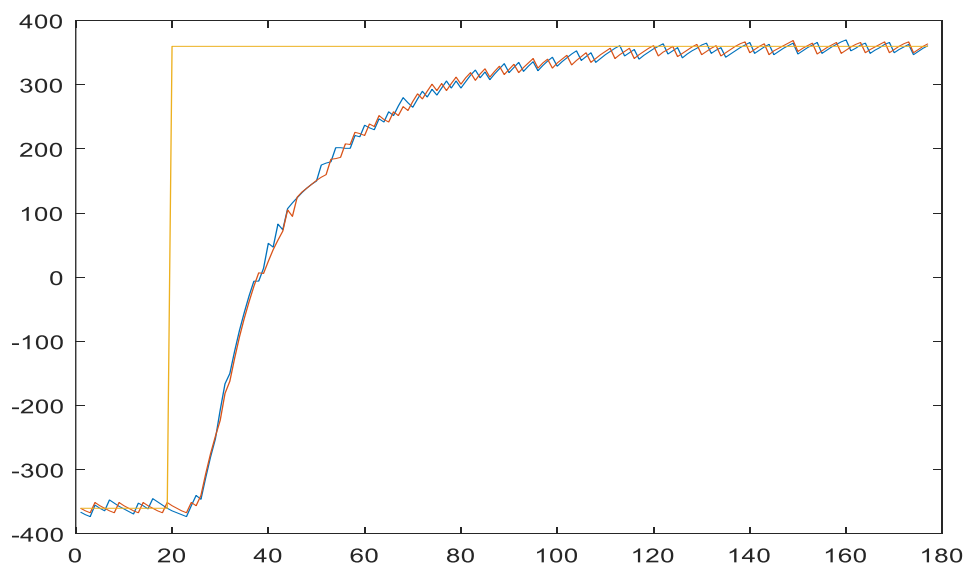


Figure 2.20 : la vitesse des roues droite et gauche en fonction d'un échelon unité

Chapitre2 Description de la partie matérielle du robot

A partir des figures précédentes nous remarquons que :

- Le temps de réponse est $\tau=58\text{ms}$, pratiquement on peut dire qu'il est tellement petit, donc notre système réagit rapidement.
- Il y a une bonne poursuite de la vitesse dans le régime permanent et transitoire avec des petits bruit (dépassements) non contrôlés observé de la figure (2.6.1.d) de faible amplitude causé par les forces d'inertie (chassie, moteur ect....), ce qui peut jouer sur la précision, mais il est tellement négligeable (2.77% d'écart).

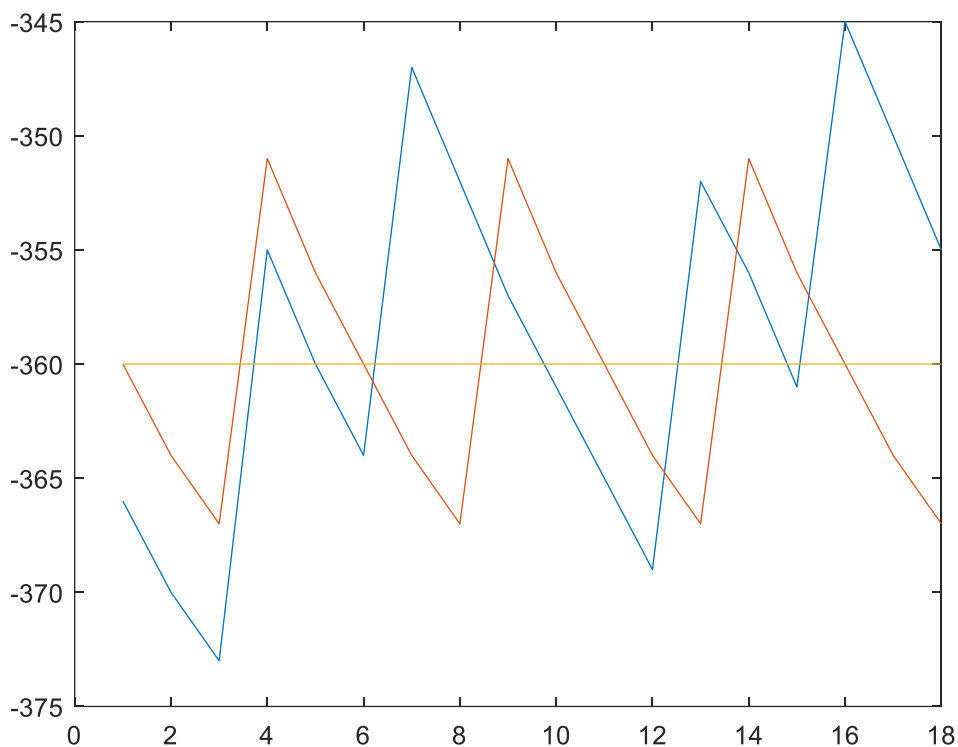


Figure 2.21 : la vitesse des roues sur l'intervalle de temps [0 :18].

C'est exactement ce qui a été implémenté dans le programme Arduino. Après ce résultat, on confirme que les valeurs obtenues (K_p , K_i , K_d) de notre modèle de commande PID programmé sur Arduino ont été effectués correctement.

Chapitre2 Description de la partie matérielle du robot

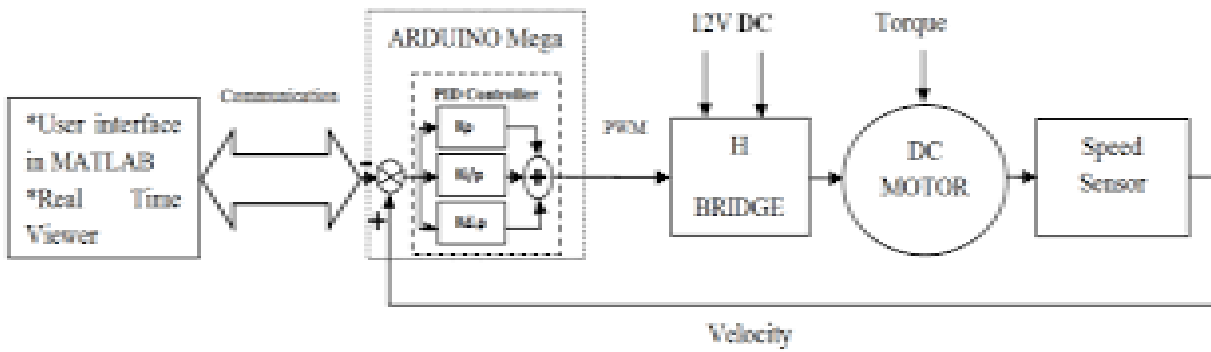


Figure 2.22 : La boucle d’asservissement du robot mobile

2.7 Conclusion

Dans ce chapitre, on a présenté les différentes parties (mécanique et électronique) de notre robot en montrant les outils utilisés dont le choix a été fait au fur et à mesure avec les tests. Le matériel choisi répond au besoin (cahier de charge), à la disponibilité et aux coûts.

Chapitre 3**Modèle cinématique et régulation****3.1 Introduction**

Dans le domaine de la robotique mobile, l'étude de la locomotion possède une place prépondérante. De nombreuses approches et solutions peuvent être envisagées et il convient d'apporter un soin particulier quand à leur sélection afin de garantir les performances du système final. Ce chapitre a pour objectif la conception complète du système de locomotion d'un robot mobile. La réalisation des cartes électroniques de commande pour les moteurs, ainsi que la synthèse des lois de contrôle sont étudiées en détails. Ces lois seront principalement divisées en deux phases : d'une part, la génération d'une trajectoire réalisable, et d'autre part, une régulation sur cette trajectoire grâce, entre autres, à un contrôle local de la vitesse des roues motrices. Ce travail se termine, enfin, par une implémentation pratique du système étudié et par une évaluation de ses performances.

L'objectif final sera la possibilité de déplacer le robot d'une configuration A à une configuration B tout en restant dans un périmètre fixé, et ceci, de la manière la plus rapide et la plus précise possible tout en respectant le roulement sans glissement des roues et tout en restant dans un certain périmètre fixé.

Plusieurs solutions sont envisagées : une suite de rotations et de lignes droites, de l'asservissement en position, du suivi de chemin ou encore du suivi de trajectoire avec profil de vitesse.

La dernière solution est celle qui permet de réaliser au mieux nos objectifs et est étudiée en détail. Elle consiste globalement en un asservissement à deux niveaux :

- un asservissement local PID en vitesse de chaque moteur pouvant être réalisé par une carte électronique.
- une régulation sur une trajectoire pré-calculée supposant le contrôle parfait des moteurs pouvant être réalisée sur l'ordinateur embarqué.

Le calcul des trajectoires et des profils de vitesse associés est étudié en détail, il fait appel à la génération d'ensembles de lignes droites, d'arcs de cercle et permet d'obtenir des trajectoires de durée minimale, réalisables pour notre robot. Nous définissons, ensuite, une loi de contrôle permettant de réguler le robot à partir d'une trajectoire réalisable calculée. Pour ce faire, nous essayons de minimiser l'erreur en orientation et l'erreur en distance du robot par rapport à la trajectoire.

Une fois la carte électronique réalisée et testée, nous en venons à l'étape d'implémentation du contrôle. Tout d'abord, le contrôle de bas niveau PID est codé dans le microcontrôleur et permet de contrôler localement en vitesse un moteur. Ensuite, un programme de haut niveau est établi en Arduino pour aider au paramétrage de la boucle PID et permet d'évaluer les performances de notre contrôleur.

3.2 Modèle du robot mobile

Le robot mobile utilisé dans ce projet est de type différentiel, il possède deux roues motrices à braquage différentiel et une roue folle assurant la stabilité du robot.

Pour modéliser le déplacement du robot mobile en tenant compte de ces contraintes, un système de repères convenable doit être judicieusement choisi. On commence par définir un repère fixe (X O Y) dans lequel le centre du robot a pour coordonnées x et y et on définit aussi un repère (X_r O_r Y_r) lié au robot, son centre est fixé au centre du robot. L'axe X_r est défini par l'axe (Avant/Arrière) du robot et l'axe Y_r est en quadrature directe par rapport à l'axe X_r .

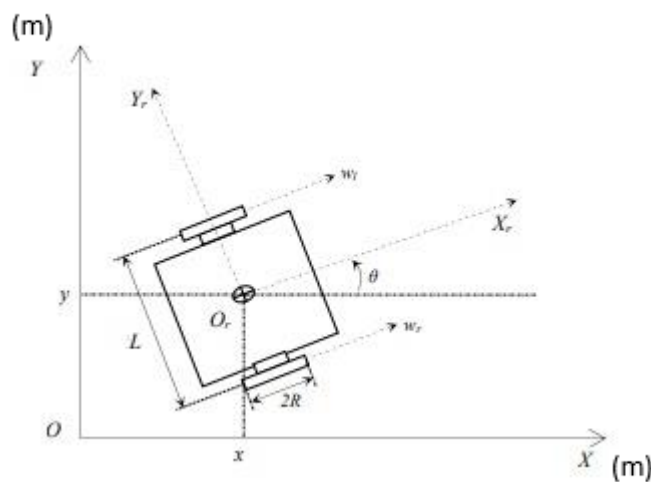


Figure 3.1 : La configuration cartésienne du robot

Le robot est un objet rigide se déplaçant sur un plan horizontal avec deux degrés de liberté : le premier correspond à un mouvement de déplacement à l'avant ou à l'arrière, tandis que le deuxième est celui de la rotation autour d'un axe vertical.

La position du robot est définie par le vecteur $[x \ y \ \theta]^T$, avec x et y sont les coordonnées cartésiennes du centre du robot et θ est l'angle d'orientation du robot par rapport au repère fixe.

Nous allons décrire ici plus en détail les propriétés de commande et de localisation particulières du robot (modèle unicycle).

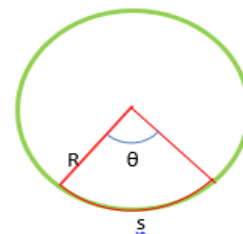
3.2.1 Commande

Dans des conditions de roulement sans glissement, la vitesse longitudinale et de rotation du robot sont liées aux vitesses de rotation angulaire des deux roues motrices (w_g et w_d) par les relations suivantes :

➤ Pour la vitesse longitudinale du robot V , on a :

$$s = R\theta \quad (3.1)$$

$$\frac{s}{t} = \frac{R\theta}{t} \quad (3.2)$$



Lorsqu'on dérive l'équation (3.2), on trouve l'équation suivante :

$$V = R\omega \quad (3.3)$$

Avec :

$$V_c = \frac{V_g + V_d}{2} \quad (3.4)$$

On remplace l'expression (3.3) dans (3.4), on obtient :

$$V_c = \frac{R\omega_g + R\omega_d}{2} \quad (3.5)$$

$$V_c = \frac{R(\omega_g + \omega_d)}{2} \quad (3.6)$$

Avec :

- s : La surface parcourue par la roue.
- R : Le rayon de la roue.
- θ : Angle de rotation de la roue.
- ω : la vitesse de rotation de la roue.
- ω_d, ω_g : les vitesses de rotation des roues droite et gauche respectivement.

➤ Pour la vitesse de rotation du robot Ω , on a :

A partir de la relation (3.3) :

$$\omega_d = R\Omega \quad (3.7)$$

$$\omega_g = R\Omega \quad (3.8)$$

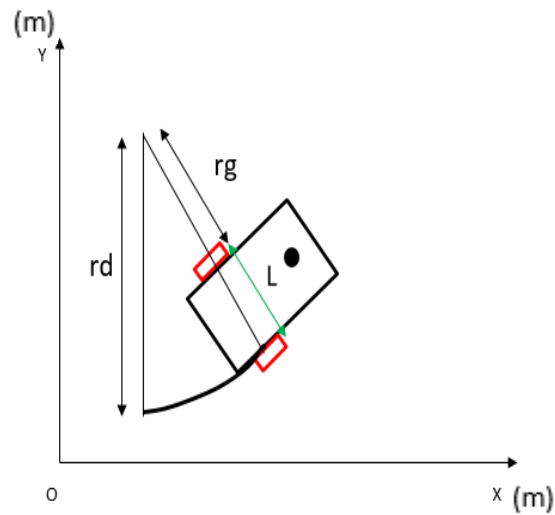


Figure 3.2 : Rayons de la roue droite et gauche dans une trajectoire de la vitesse instantanée

D'après la figure 3.2 :

$$rd = rc + \frac{L}{2} \quad (3.9)$$

$$rg = rc - \frac{L}{2} \quad (3.10)$$

Alors :

$$Vd = \left(rc + \frac{L}{2} \right) \Omega \quad (3.11)$$

$$Vg = \left(rc - \frac{L}{2} \right) \Omega \quad (3.12)$$

Avec :

- rc : Rayon de rotation du centre de robot par rapport à la trajectoire instantanée.
- rd, rg : Rayons de rotation des roues par rapport à la trajectoire instantanée.
- L : Distance entre les deux roues.

La soustraction de l'équation (3.11) et (3.12), nous donne :

$$Vd - Vg = L\Omega \quad (3.13)$$

Ce qui implique :

$$\Omega = \frac{R(Vd - Vg)}{RL} \quad (3.14)$$

Donc :

$$\Omega = \frac{R}{L}(wd - wg) \quad (3.15)$$

➤ Pour le modèle cinématique du robot utilisé, on a :

$$\begin{pmatrix} Xt \\ Yt \\ \theta t \end{pmatrix} = \begin{pmatrix} Xt - 1 \\ Yt - 1 \\ \theta t - 1 \end{pmatrix} + \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta \theta \end{pmatrix} \quad (3.16)$$

$$\begin{pmatrix} Xt \\ Yt \\ \theta t \end{pmatrix} = \begin{pmatrix} Xt - 1 \\ Yt - 1 \\ \theta t - 1 \end{pmatrix} + \begin{pmatrix} \cos\theta \cdot \frac{R(wg+wd)}{2} \cdot \Delta t \\ \sin\theta \cdot \frac{R(wg+wd)}{2} \cdot \Delta t \\ \Omega \cdot \Delta t \end{pmatrix} \quad (3.17)$$

$$\begin{bmatrix} X' \\ Y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \Omega \end{bmatrix} \quad (3.18)$$

Avec : V est la vitesse linéaire du centre du robot, Ω est la vitesse de rotation du robot autour de son centre.

3.3 Conclusion

Dans ce chapitre nous avons présenté la modélisation d'un robot mobile, les méthodes de calculs des différentes trajectoires du robot ainsi leur vitesse angulaire et longitudinale.

Chapitre 4**Tests et résultats****4.1 Introduction**

La partie la plus importante de notre projet sera discutée dans ce chapitre. Cela inclura les résultats expérimentaux et les problèmes auxquels nous nous sommes confrontés pour faire fonctionner le système global d'une manière satisfaisante.

4.2 Explication de notre stratégie

Le robot doit être positionné aux coins de notre table, le mouvement contient quatre étapes :

- ✓ 1ere étape : le robot fait une rotation avec un angle θ_1 calculé d'après les coordonnées x_1, y_1 donnée.
- ✓ 2eme étape : le robot quitte sa zone de départ en ligne droite dont le chemin le plus court vers la cible, une fois arrivée au gobelet, il va le ramasser et le soulevé avec un Gripper.
- ✓ 3eme étape : le robot fait une 2eme rotation avec un angle θ_2 calculé d'après les coordonnées x_2, y_2 donnée.
- ✓ 4eme étape : le robot quitte le point d'arrivé de la zone précédente et poursuivre sa trajectoire vers la 2eme zone voulus.

4.2.1 Organigramme du programme utilisé sur Matlab

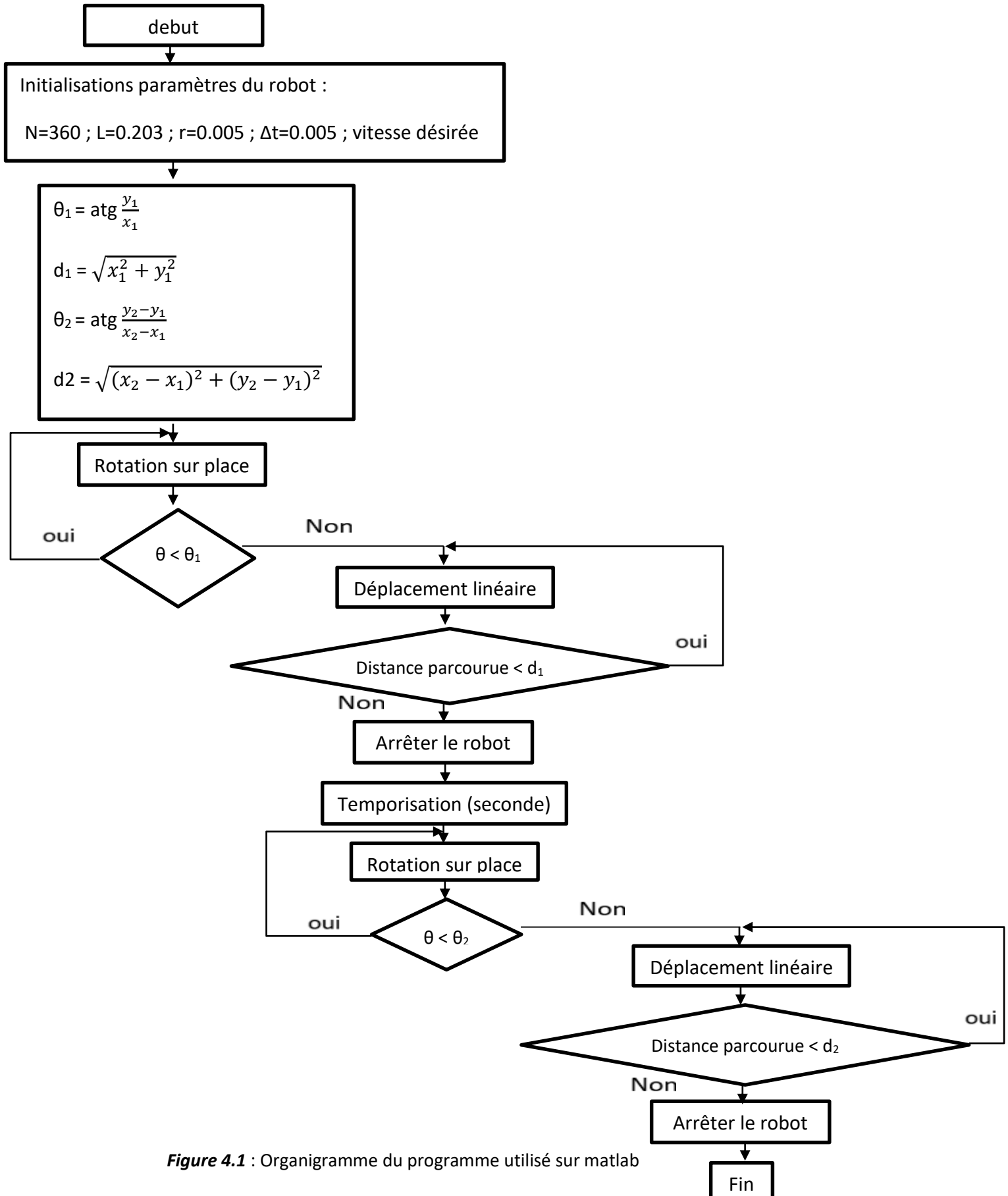


Figure 4.1 : Organigramme du programme utilisé sur matlab

4.3 Solution des problèmes rencontrés

D’après nos essai, la roue gauche fais la rotation d’un angle $\theta_1 - \frac{\pi}{12}$ au lieu d’un angle θ_1 , car on a pas pu traiter les informations de rotation et translation on ligne droite ; lors de l’exécution de notre programme aux même temps que le robot roule, ce qui va accumulé l’erreur de translation dx, dy et rendue le déplacement vers la cible très difficile, car il devait s’arrêté à chaque fois après la première étape, pour ce positionné et laissais la roue gauche terminé sa rotation.

Ce qui fait perde du temps.

On a voulu tester la précision des mouvements de notre robot mobile et pour cela, on a effectué quelques modifications dans le même programme utilisé précédemment ; on a décidé d’augmenté l’angle est devient $\theta_1 + \frac{\pi}{12}$ dont le but de compensé le retard de la roue gauche, (même résonnement a la rotation de la 3eme étape, la seul différence au niveau de l’angle $\theta_2 - \frac{\pi}{18}$ alors on a décidé d’augmenté l’angle est devient $\theta_1 + \frac{\pi}{18}$).

4.3.1 Simulation sur Matlab

Résultat obtenue lors de la simulation de la trajectoire des deux roues et le centre du robot après avoir une rotation puis une translation :

- La figure ci-dessous représente la coordonnée x en fonction de la coordonnée y, on observe que l’angle obtenue suit d’une manière très claire le résultat de $\text{atg}(\frac{y_1}{x_1})$; celle choisie comme consigne dans le programme matlab ($\theta_1 = 45^\circ$).

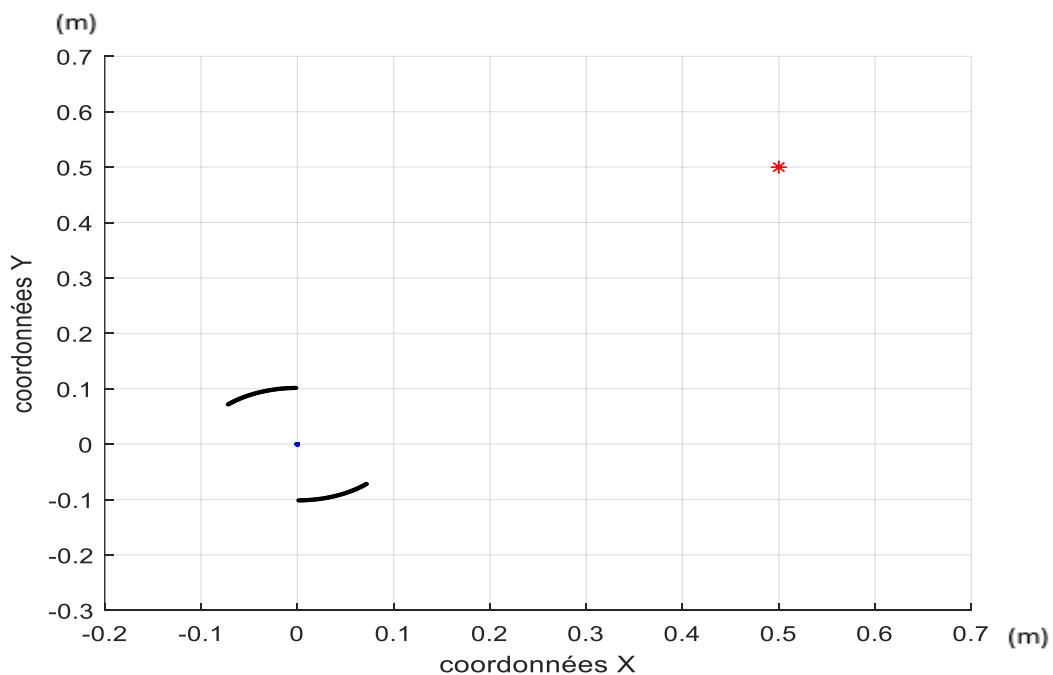


Figure 4.2 : la rotation d’un angle θ_1 de la première étape

- La figure ci-dessous représente la coordonnée x en fonction de coordonnée y, là on peut voir le changement de position sur l'axe x et y; on observe que la trajectoire obtenue suit d'une manière très claire celle choisie comme consigne dans le programme matlab ($x_1=0.5$; $y_1=0.5$).

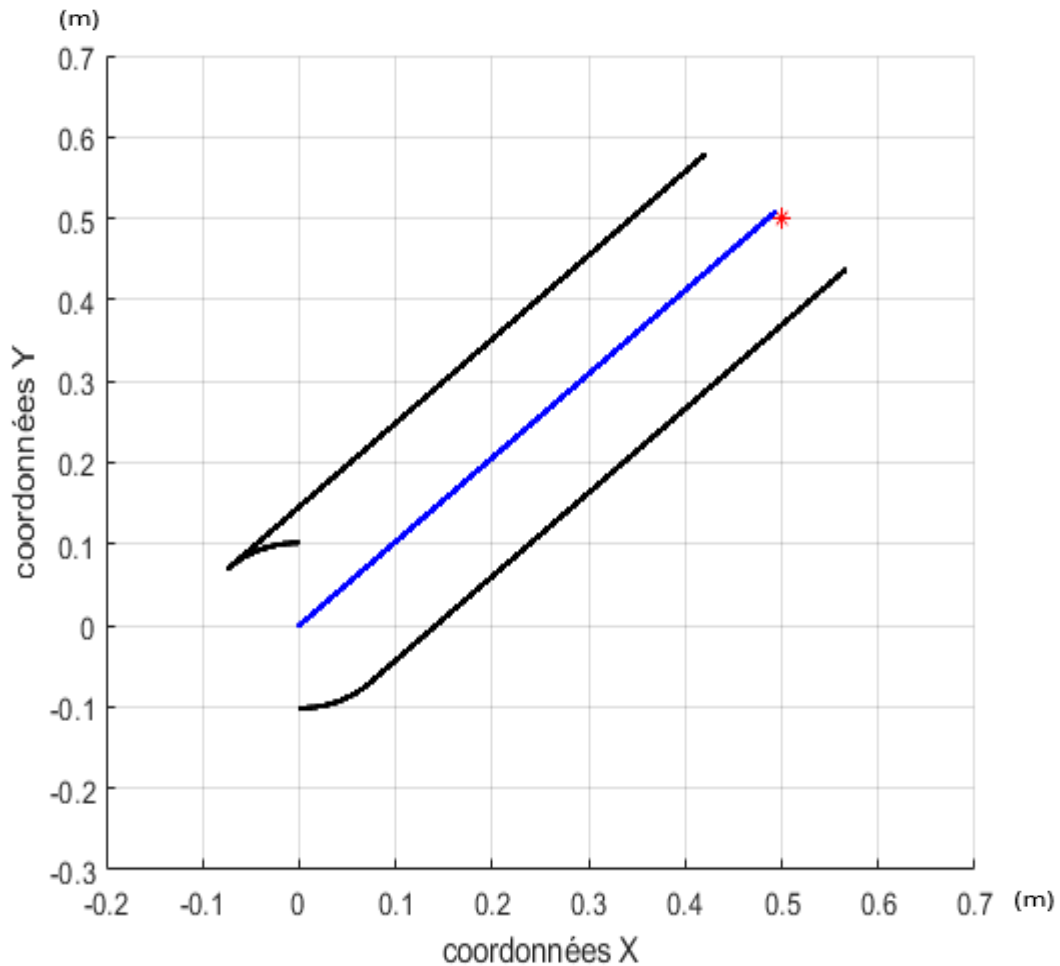


Figure 4.3 : le mouvement du robot de la 2eme étape

- La figure ci-dessous représente la coordonnée x en fonction de la coordonnée y, on observe que l'angle obtenue suit d'une manière très claire le résultat de $\text{atg} \left(\frac{y_2 - y_1}{x_2 - x_1} \right)$; celle choisie comme consigne dans le programme matlab ($\theta_2 = 26^\circ.57$).

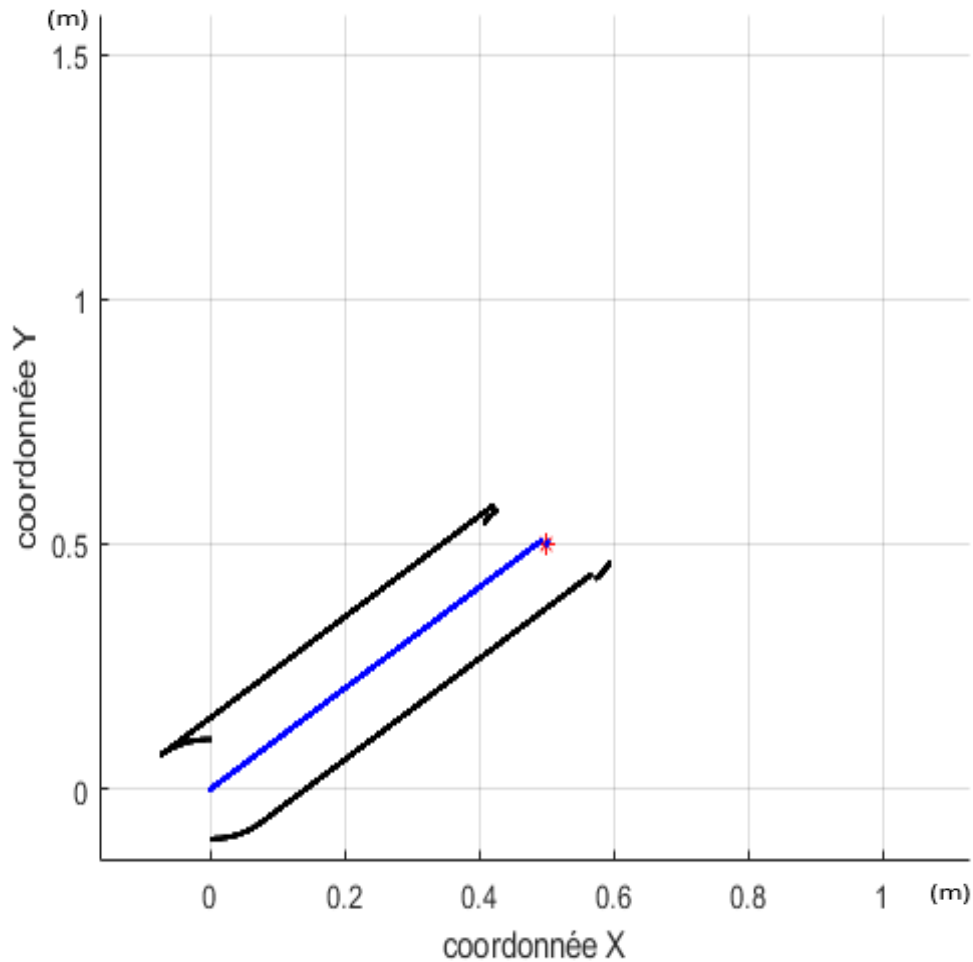


Figure 4.4 : la rotation d'un angle θ_2 de la troisième étape

- La figure ci-dessous représente la coordonnée x en fonction de coordonnée y , là on remarque que le robot arrive aux point final voulu ($x_2 = 1$; $y_2 = 0.75$).

Donc on a réussi à mettre le robot sur la trajectoire programmée.

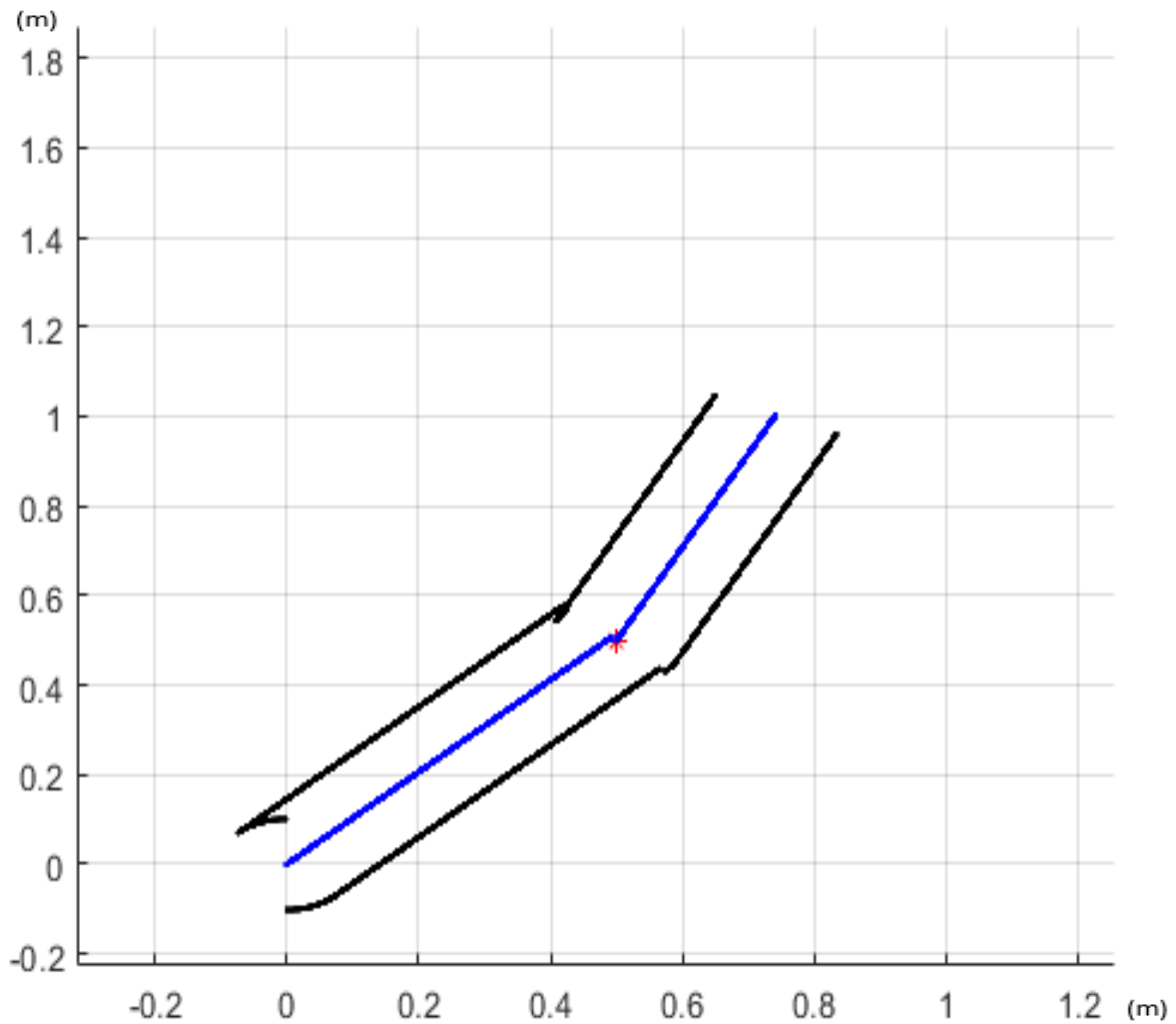


Figure 4.5 : le mouvement final de notre robot

4.3.2 Test réalisé

Même si la programmation de notre robot mobile se fait sur Arduino (programme téléversé à la carte), on a quand même créé un programme MATLAB (script) dans le but d'établir une comparaison entre les résultats de la simulation et les résultats des tests réalisés concrètement.

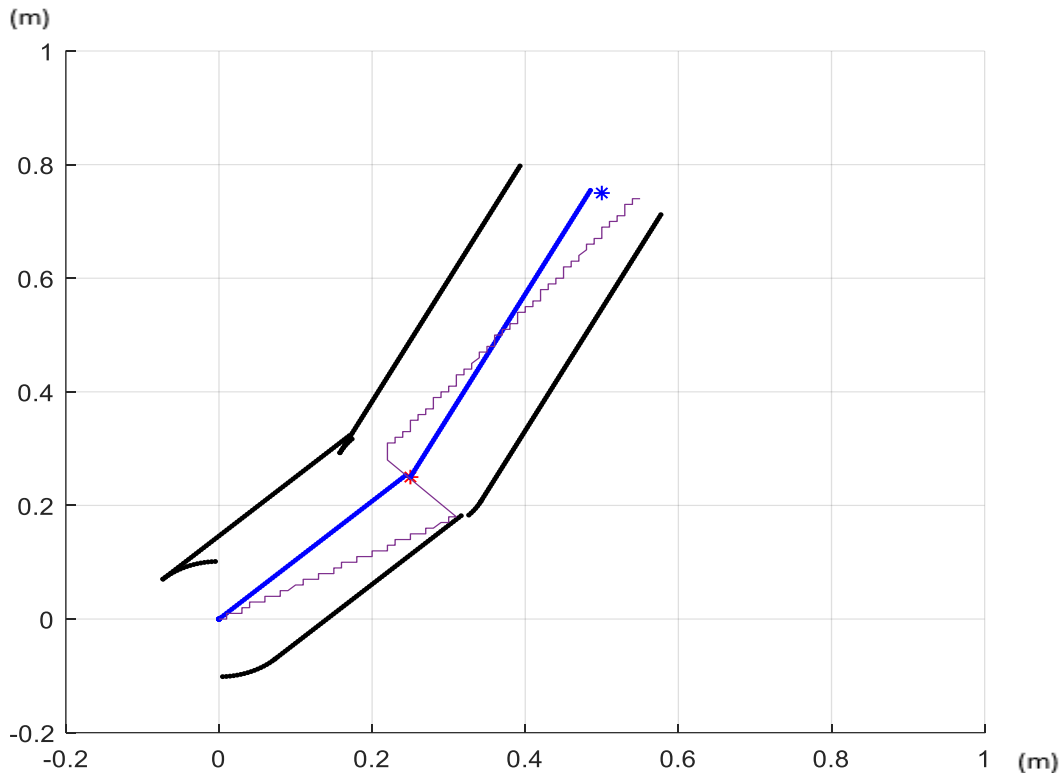


Figure 4.6 : le mouvement final de notre robot avec une comparaison entre la simulation et le réel

- D'après le graph ci-dessus et suivant le déplacement réel de notre robot (violet), on remarque que le robot suit la trajectoire désiré mais une petite coupure apparaitre au niveau de la première zone ciblé. Cette coupure due à l'absence de l'instruction delay() dans le programme arduino et qui a été déclaré dans le programme de simulation matlab. Pratiquement on peut pas le déclaré dans la boucle du programme arduino parce que sa vas être exécuté à chaque itération, qui vas influencer sur la vitesse de notre robot, ce qui fait un perd du temps, mais en général le robot fonctionne très bien.

4.4. La variation de vitesse (commande du pont en H par PWM)

Quand on veut faire varier la vitesse d'un moteur, la première idée qui vient à l'esprit est de faire varier la tension aux bornes du moteur, mais on constate que pour des valeurs faibles de la tension, le moteur ne tourne pas. Le moteur demande une tension minimale assez importante pour démarrer. Si cette dernière est trop basse, les forces électromagnétiques ne sont pas suffisantes pour combattre le frottement. Il devient donc difficile d'ajuster la vitesse de façon précise.

La solution à ce problème est astucieuse. Il suffit de fournir au moteur une tension qui est toujours la même soit la tension maximale ! Par contre, cette tension ne sera appliquée que par très courtes périodes de temps. En ajustant la longueur de ces périodes de temps, on arrive à faire avancer plus ou moins vite les moteurs.

Mieux, on remarque que la vitesse des moteurs devient proportionnelle à la longueur des périodes de temps. Contrôler la longueur des périodes passées à la tension maximale par rapport au temps passé sans application de tension (tension nulle) est donc le cœur de la solution.

En réalité, cette solution est fort connue en contrôle des systèmes et en électronique et elle porte le nom de PWM (Pulse Width Modulation) ou MLI pour modulation en Largeur d'impulsion.

Le principe général d'une MLI consiste à appliquer un signal de commande logique et périodique dont on fera varier le rapport cyclique afin de modifier sa valeur moyenne :



Lorsqu'un tel signal est appliqué à un des commutateurs inférieurs tandis que le commutateur supérieur opposé est en conduction, le pont devient un hacheur permettant de faire varier la vitesse de rotation du moteur.

La commande d'actionneurs de puissance par PWM est très liée à la notation de fréquence.

Pour que l'impression d'une valeur moyenne constante d'allumage apparaisse, il faut que l'alternance d'allumage/extinction soit suffisamment rapide pour qu'elle ne se remarque pas. Selon les utilisations la fréquence de PWM va de 100 Hz à 200 kHz.

4.5 Intérêts de notre circuit électronique

Nous avons réussi de construire un circuit électrique dont l'avantage d'envoyer un seul signal PWM dans une entrée INT1 pour faire tourner le moteur et inverser le sens de rotation en utilisant un inverseur logique, ce dernier joue un rôle essentiel au freinage.

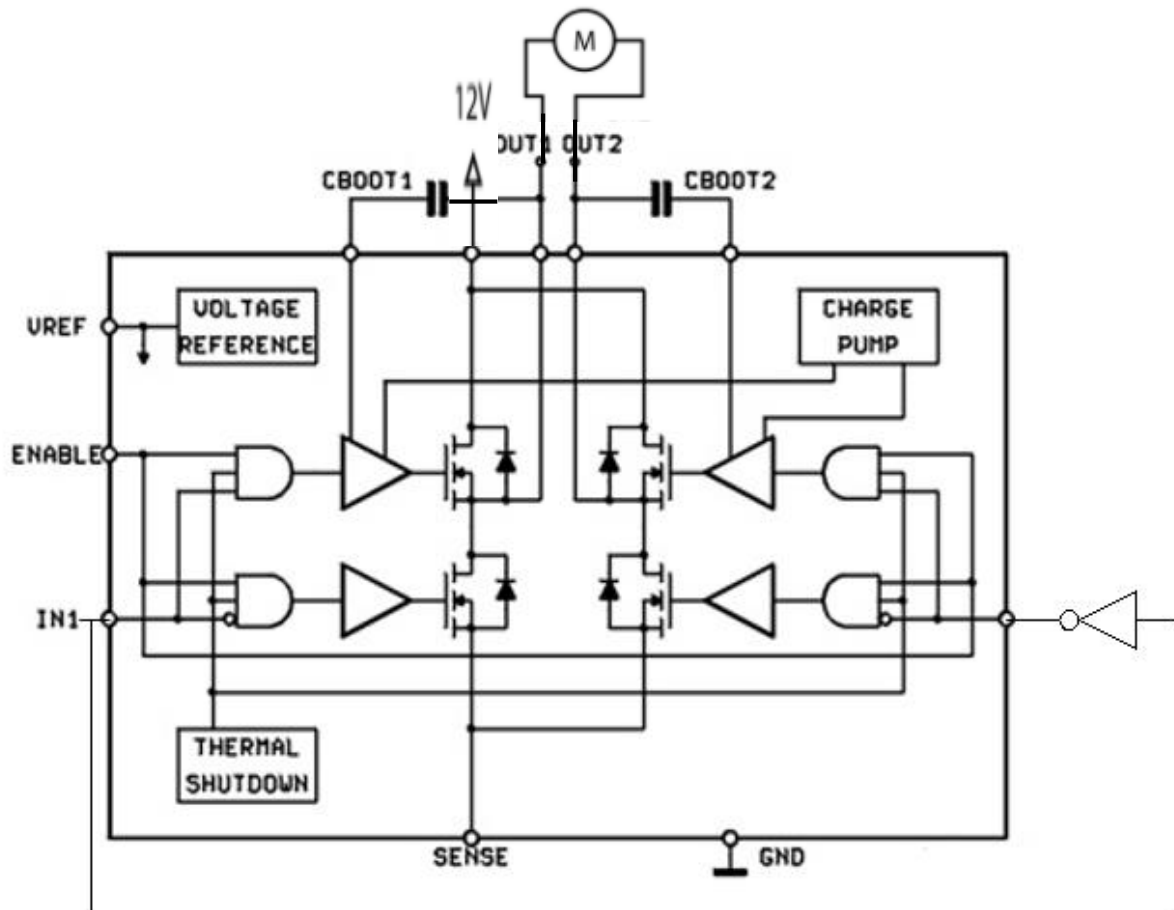


Figure 4.7 : Schéma développé du circuit L6203.

Lorsqu'on fournit un signal PWM à l'entrée INT1, l'inverseur va inverser le signal dont le but est d'annuler la vitesse d'un seul coup, la tension V s'annule, le transistor se bloque ; le courant ne peut donc plus s'évacuer par celui-ci.

Exemple : La diode D, appelée diode de roue libre, permet au courant maintenu par la bobine L de circuler.

Si on ne met pas cette diode, la tension induite générée par la bobine B risque de provoquer le claquage du transistor.

Lorsque le transistor conduit, la diode est bloquée et n'intervient pas dans le fonctionnement.

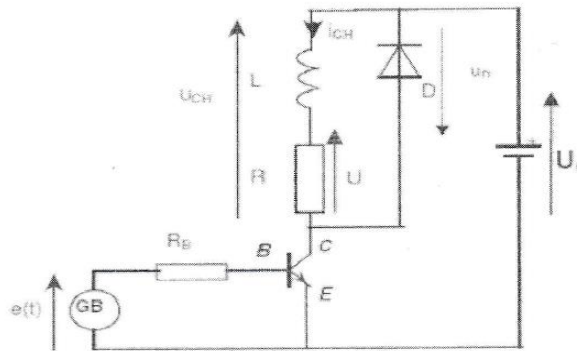


Figure 4.8 : Rôle de la DRL

4.6 La commande de la vitesse

Pour commander la vitesse du robot, on a envoyé la donnée du signal de la vitesse à travers Arduino master (maitre) avec un protocole de communication serial (série), l'avantage de notre programme est de faire augmenter la capacité de transfert des bits de la donnée (8 bits à l'état normal) vers 16 bits.

L'astuce et de récupérer les 8 bits premiers du poids faible de notre signal dans le but est de faire décalé pour rajouté le deuxième signal à 8 bits du poids fort.

A la fin on obtient les 16 bits dont la vitesse est de $2^{16}-1=65535$ au lieu de $2^8-1=255$ (la vitesse maximale réel du moteur aller jusqu'au 1000).

4.7 La traduction de la simulation à la pratique

En pratique, notre robot doit déplacer est ramassé un gobelet, pour ce fait on a traduit le programme sous matlab à un programme compatible au langage arduino et on a ajouté des petites modifications concernant le mouvement de ramassage, pour ce faire, on a proposé l'organigramme suivant :

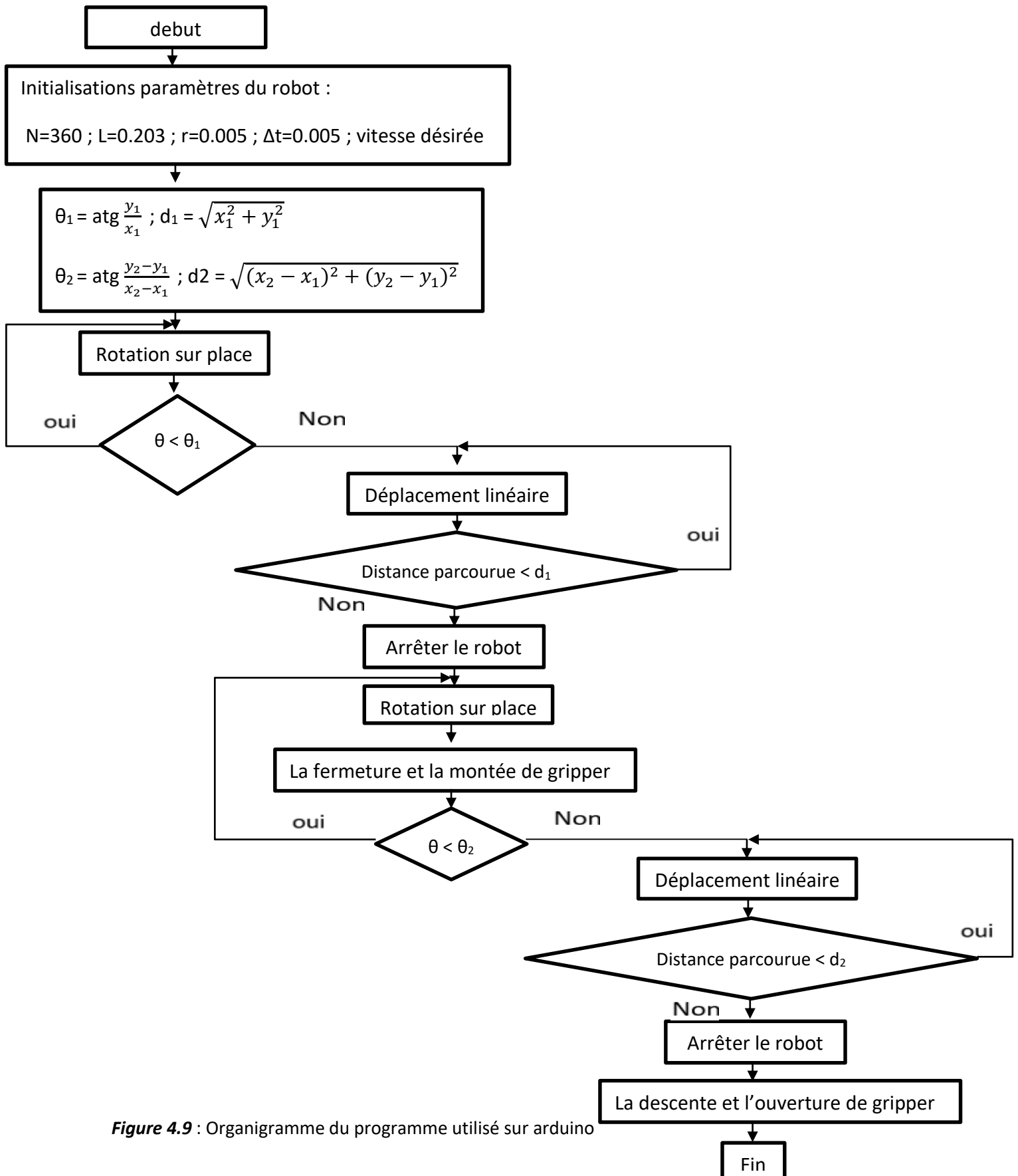


Figure 4.9 : Organigramme du programme utilisé sur arduino

La figure ci-dessous représente la vitesse réel des deux roues (roue droite en orange, roue gauche en bleu) qui vont suivre la vitesse programmé $V=200$ impulsion/s.

Le graphe contient 4 étapes essentielles au mouvement total de notre robot :(avec $\Delta t=0.005s$)

- Lorsque $t \in [0 : 700] * 0.005s$: une augmentation de vitesse des deux roues dans le sens opposées pour faire tourner le robot d'un angle θ_1 .
- La 2eme étape : la roue gauche change le sens de rotation pour suivre son mouvement avec la roue droite dans une trajectoire linéaire et se stabilise dans a une vitesse programmé $V=200$ impulsion/s.
- Lorsque $t=3250*0.005s$: une augmentation de vitesse des deux roues dans le sens opposées pour faire tourner le robot d'un angle θ_1 .
- La vitesse des deux roues diminue, dont l'une atteint 100 impulsion/s, tandis que l'autre s'annule pour tourner d'un angle θ_2 sans faire arrêter le robot au cours du ramassage, puis une croissance de vitesse des deux roues jusqu'elles stabilises leurs vitesse $V=200$ impulsion/s et arrivé au 2eme point ciblé pour que le robot dévier avec un angle de θ_2
- Puis il change la vitesse avec une croissance de la vitesse des deux roues dans le même sens pour atteindre la même vitesse dans le but de suivre la 2eme trajectoire linéaire.
- A la fin une décroissante de la vitesse jusqu'elle s'annule pour arrêter le robot au dernier point souhaité.

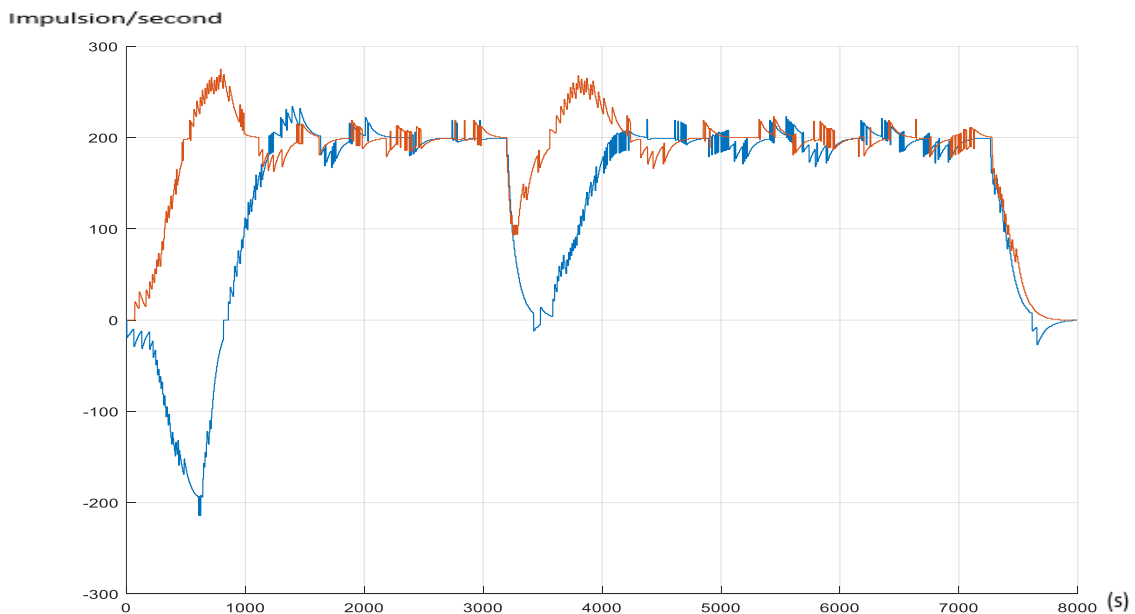


Figure 4.10 : La vitesse des roues droite et gauche en fonction d'un échelon unité

Remarque : Il faut aussi tenir compte des petits bruit non contrôlés observé de la figure 4.10, ce qui peut jouer sur la précision, mais il est tellement négligeable (faible amplitude)

On trouve aussi un dépassement dans le régime transitoire car notre travail est basé sur le modèle cinématique, pas sur le modèle dynamique, mais en générale le robot déplace très bien.

C'est exactement ce qui a été implémenté dans le programme Arduino.

Après ce résultat, on confirme que les calculs de notre modèle cinématique programmé sur Arduino ont été effectués correctement.

4.8 Conclusion

Dans ce chapitre, les résultats expérimentaux et de simulation sont présentés. On a démontré les différentes méthodes utilisées pour le calcul d'angles de rotation, position des axes x , y du robot mobile ainsi leur vitesse. La phase de réalisation a été décrite à travers les différentes captures et figures présentées.

Conclusion général

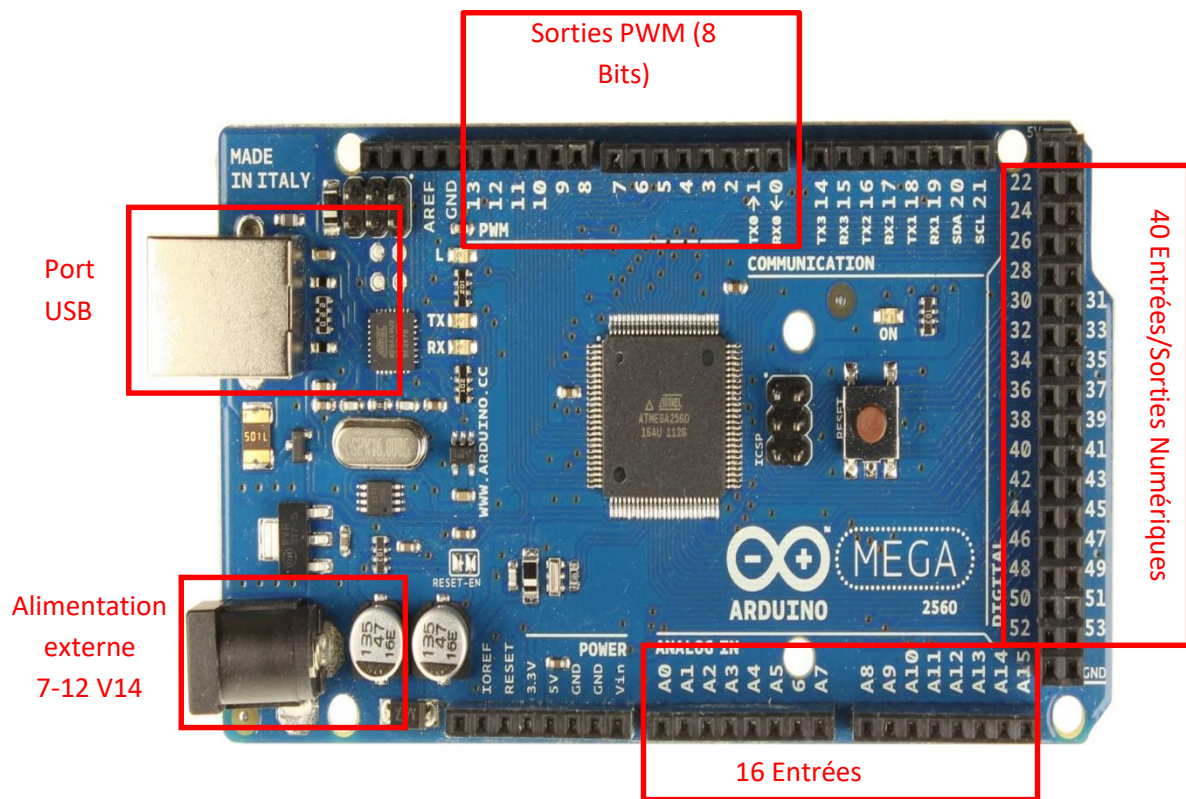
Durant ce projet j'ai pu acquérir une certaine expérience pour la conception et la réalisation des différentes parties d'un robot mobile. J'ai avant tout, appris à bâtir un projet en respectant un cahier des charges stricte, et ce par la construction d'une stratégie idée par idée jusqu'à maturité.

La partie réalisation nous a été d'un profit inestimable, notamment pour la partie mécanique, qui a nécessité l'usinage des pièces et différentes fonctions de tournage. De même pour la partie électronique et informatique, dans lesquelles j'ai pu mettre en pratique mes connaissances théoriques acquises tout au long de mon cursus de formation.

Le principal problème auquel mon robot ainsi que bien d'autre est celui de la fiabilité. Cette dernière remarque nous a permis de corriger ma vision sur la manière de concevoir le système devant manipuler des objets en évitant de réaliser ce dernier en fonction des caractéristiques physiques très particulières qui peuvent être différentes d'un objet à un autre.

Je souhaite que ce projet connaitra une suite dans son perfectionnement et que d'autres étudiants reprennent le travail déjà accompli en travaillant surtout sur le plan d'éviter les obstacles et l'asservissement du déplacement et le suivie d'une trajectoire de telle sorte que le robot tourne on même temps qu'il avance en réalisant un module spécialement dédié à cette fonction.

Caractéristiques d'Arduino MEGA :



Résumé des caractéristiques de la carte Arduino MEGA :

Microcontrôleur	ATmega328
Tension d'alimentation interne	5V
Tension d'alimentation externe	7 à 12V
Entrées/sorties numériques	54 dont 14 sorties PWM
Entrées analogiques	16
Courant max par broches E/S	40 mA
Courant max sur sortie 3,3V	50mA
Mémoire Flash	256 Ko
Mémoire SRAM	8 Ko
mémoire EEPROM	4 Ko
Fréquence horloge	16 MHz



L6201 L6202 - L6203

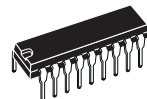
DMOS FULL BRIDGE DRIVER

- SUPPLY VOLTAGE UP TO 48V
- 5A MAX PEAK CURRENT (2A max. for L6201)
- TOTAL RMS CURRENT UP TO
L6201: 1A; L6202: 1.5A; L6203/L6201PS: 4A
- $R_{DS(ON)}$ 0.3 Ω (typical value at 25 °C)
- CROSS CONDUCTION PROTECTION
- TTL COMPATIBLE DRIVE
- OPERATING FREQUENCY UP TO 100 KHz
- THERMAL SHUTDOWN
- INTERNAL LOGIC SUPPLY
- HIGH EFFICIENCY

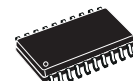
DESCRIPTION

The I.C. is a full bridge driver for motor control applications realized in Multipower-BCD technology which combines isolated DMOS power transistors with CMOS and Bipolar circuits on the same chip. By using mixed technology it has been possible to optimize the logic circuitry and the power stage to achieve the best possible performance. The DMOS output transistors can operate at supply voltages up to 42V and efficiently at high switch-

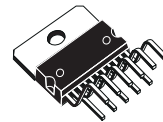
MULTIPOWER BCD TECHNOLOGY



Powerdip 12+3+3



SO20 (12+4+4)



Multiwatt11



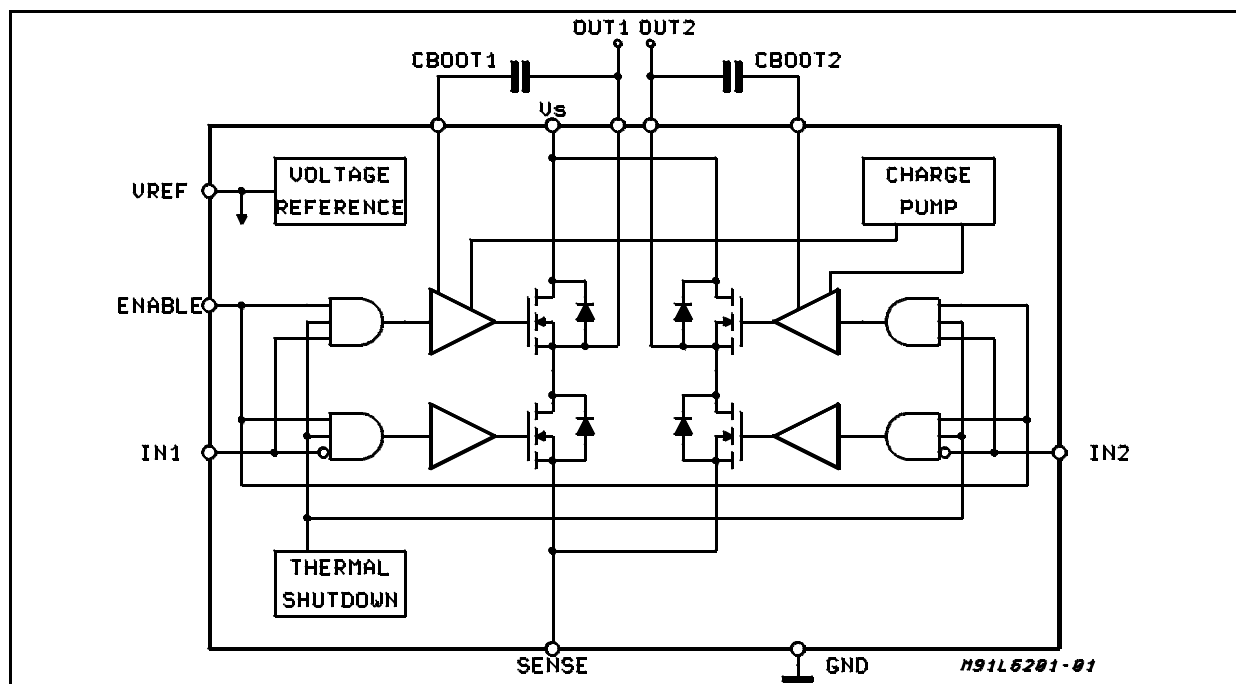
PowerSO20

ORDERING NUMBERS:

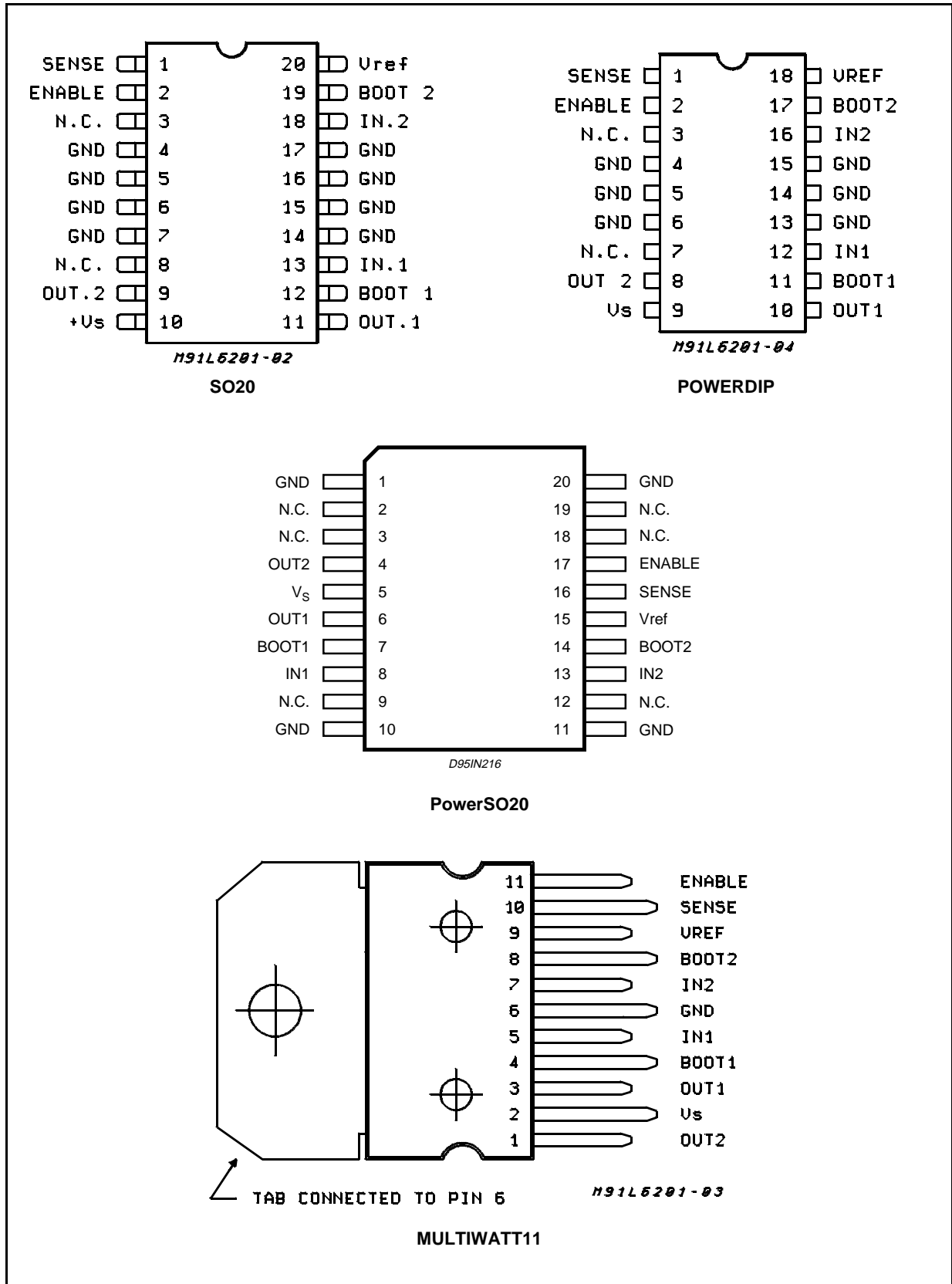
- L6201 (SO20)
- L6201PS (PowerSO20)
- L6202 (Powerdip18)
- L6203 (Multiwatt)

ing speeds. All the logic inputs are TTL, CMOS and μ C compatible. Each channel (half-bridge) of the device is controlled by a separate logic input, while a common enable controls both channels. The I.C. is mounted in three different packages.

BLOCK DIAGRAM



PIN CONNECTIONS (Top view)



PINS FUNCTIONS

Device				Name	Function
L6201	L6201PS	L6202	L6203		
1	16	1	10	SENSE	A resistor R_{sense} connected to this pin provides feedback for motor current control.
2	17	2	11	ENABLE	When a logic high is present on this pin the DMOS POWER transistors are enabled to be selectively driven by IN1 and IN2.
3	2,3,9,12,18,19	3		N.C.	Not Connected
4,5	–	4	6	GND	Common Ground Terminal
–	1, 10	5		GND	Common Ground Terminal
6,7	–	6		GND	Common Ground Terminal
8	–	7		N.C.	Not Connected
9	4	8	1	OUT2	Output of 2nd Half Bridge
10	5	9	2	V_s	Supply Voltage
11	6	10	3	OUT1	Output of first Half Bridge
12	7	11	4	BOOT1	A bootstrap capacitor connected to this pin ensures efficient driving of the upper POWER DMOS transistor.
13	8	12	5	IN1	Digital Input from the Motor Controller
14,15	–	13	6	GND	Common Ground Terminal
–	11, 20	14		GND	Common Ground Terminal
16,17	–	15		GND	Common Ground Terminal
18	13	16	7	IN2	Digital Input from the Motor Controller
19	14	17	8	BOOT2	A bootstrap capacitor connected to this pin ensures efficient driving of the upper POWER DMOS transistor.
20	15	18	9	V_{ref}	Internal voltage reference. A capacitor from this pin to GND is recommended. The internal Ref. Voltage can source out a current of 2mA max.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_s	Power Supply	52	V
V_{OD}	Differential Output Voltage (between Out1 and Out2)	60	V
V_{IN}, V_{EN}	Input or Enable Voltage	– 0.3 to + 7	V
I_o	Pulsed Output Current for L6201PS/L6202/L6203 (Note 1)	5	A
	– Non Repetitive (< 1 ms) for L6201	5	A
	DC Output Current for L6201PS/L6202/L6203 for L6201 (Note 1)	10 1	A A
V_{sense}	Sensing Voltage	– 1 to + 4	V
V_b	Bootstrap Peak Voltage	60	V
P_{tot}	Total Power Dissipation:		
	$T_{pins} = 90^\circ\text{C}$ for L6201	4	W
	for L6202	5	W
	$T_{case} = 90^\circ\text{C}$ for L6201PS/L6203	20	W
	$T_{amb} = 70^\circ\text{C}$ for L6201 (Note 2)	0.9	W
for L6202 (Note 2)	1.3	W	
for L6201PS/L6203 (Note 2)	2.3	W	
T_{stg}, T_j	Storage and Junction Temperature	– 40 to + 150	$^\circ\text{C}$

Note 1: Pulse width limited only by junction temperature and transient thermal impedance (see thermal characteristics)

Note 2: Mounted on board with minimized dissipating copper area.



L6201 - L6202 - L6203

THERMAL DATA

Symbol	Parameter		Value				Unit
			L6201	L6201PS	L6202	L6203	
R _{th j-pins}	Thermal Resistance Junction-pins	max	15	–	12	–	°C/W
R _{th j-case}	Thermal Resistance Junction Case	max.	–	–	–	3	
R _{th j-amb}	Thermal Resistance Junction-ambient	max.	85	13 (*)	60	35	

(*) Mounted on aluminium substrate.

ELECTRICAL CHARACTERISTICS (Refer to the Test Circuits; T_j = 25°C, V_S = 42V, V_{sens} = 0, unless otherwise specified).

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V _s	Supply Voltage		12	36	48	V
V _{ref}	Reference Voltage	I _{REF} = 2mA		13.5		V
I _{REF}	Output Current				2	mA
I _s	Quiescent Supply Current	EN = H V _{IN} = L EN = H V _{IN} = H EN = L (Fig. 1,2,3) I _L = 0		10 10 8	15 15 15	mA mA mA
f _c	Commutation Frequency (*)			30	100	KHz
T _j	Thermal Shutdown			150		°C
T _d	Dead Time Protection			100		ns

TRANSISTORS

OFF						
I _{DSS}	Leakage Current	Fig. 11 V _s = 52 V			1	mA
ON						
R _{DS}	On Resistance	Fig. 4,5		0.3	0.55	Ω
V _{DS(ON)}	Drain Source Voltage	Fig. 9 I _{DS} = 1A I _{DS} = 1.2A I _{DS} = 3A	L6201 L6202 L6201PS/0 3	0.3 0.36 0.9		V V V
V _{sens}	Sensing Voltage		– 1		4	V

SOURCE DRAIN DIODE

V _{sd}	Forward ON Voltage	Fig. 6a and b I _{SD} = 1A L6201 EN = L I _{SD} = 1.2A L6202 EN = L I _{SD} = 3A L6201PS/03 EN = L		0.9 (**) 0.9 (**) 1.35 (**)		V V V
t _{rr}	Reverse Recovery Time	$\frac{dif}{dt} = 25 \text{ A}/\mu\text{s}$ I _F = 1A L6201 I _F = 1.2A L6202 I _F = 3A L6203		300		ns
t _{fr}	Forward Recovery Time			200		ns

LOGIC LEVELS

V _{IN L} , V _{EN L}	Input Low Voltage		– 0.3		0.8	V
V _{IN H} , V _{EN H}	Input High Voltage		2		7	V
I _{IN L} , I _{EN L}	Input Low Current	V _{IN} , V _{EN} = L			–10	μA
I _{IN H} , I _{EN H}	Input High Current	V _{IN} , V _{EN} = H		30		μA

ELECTRICAL CHARACTERISTICS (Continued)
LOGIC CONTROL TO POWER DRIVE TIMING

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
t ₁ (V _i)	Source Current Turn-off Delay	Fig. 12		300		ns
t ₂ (V _i)	Source Current Fall Time	Fig. 12		200		ns
t ₃ (V _i)	Source Current Turn-on Delay	Fig. 12		400		ns
t ₄ (V _i)	Source Current Rise Time	Fig. 12		200		ns
t ₅ (V _i)	Sink Current Turn-off Delay	Fig. 13		300		ns
t ₆ (V _i)	Sink Current Fall Time	Fig. 13		200		ns
t ₇ (V _i)	Sink Current Turn-on Delay	Fig. 13		400		ns
t ₈ (V _i)	Sink Current Rise Time	Fig. 13		200		ns

(*) Limited by power dissipation

(**) In synchronous rectification the drain-source voltage drop V_{DS} is shown in fig. 4 (L6202/03); typical value for the L6201 is of 0.3V.

Figure 1: Typical Normalized I_s vs. T_j

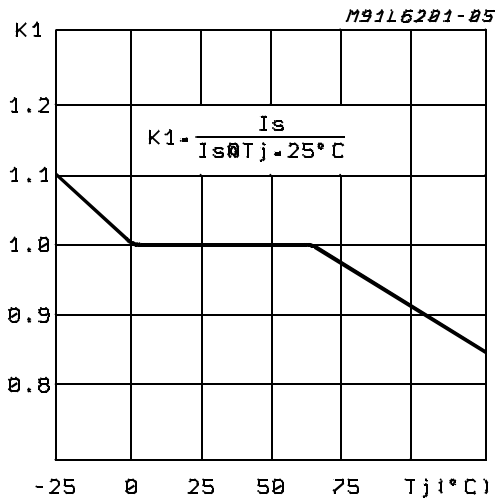


Figure 2: Typical Normalized Quiescent Current vs. Frequency

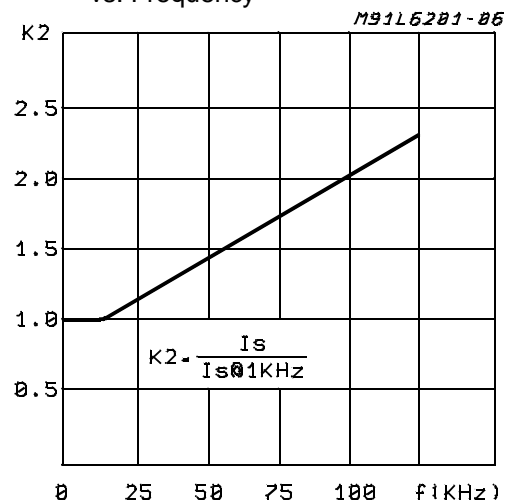


Figure 3: Typical Normalized I_s vs. V_s

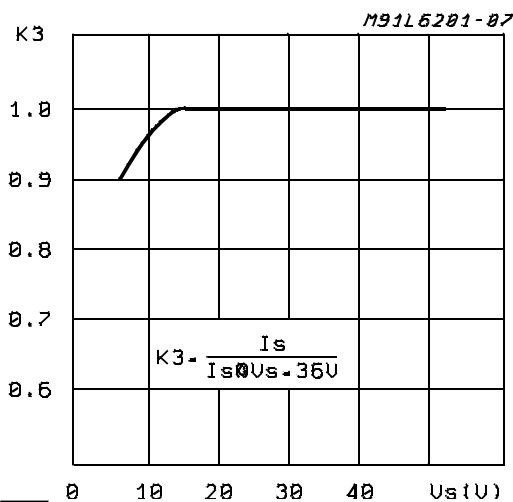


Figure 4: Typical R_{DS(ON)} vs. V_s ~ V_{ref}

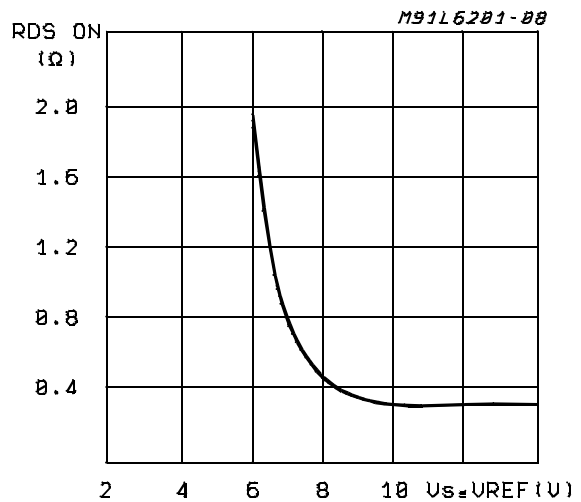


Figure 5: Normalized $R_{DS(ON)}$ at 25°C vs. Temperature Typical Values

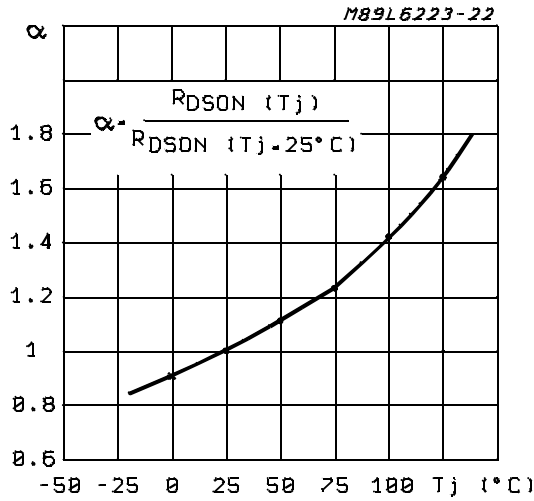


Figure 6a: Typical Diode Behaviour in Synchronous Rectification (L6201)

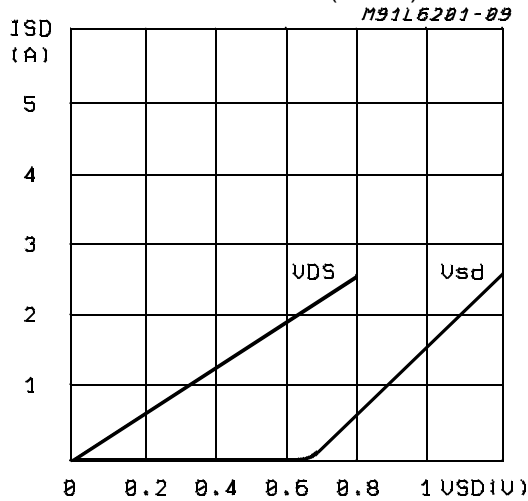


Figure 6b: Typical Diode Behaviour in Synchronous Rectification (L6201PS/02/03)

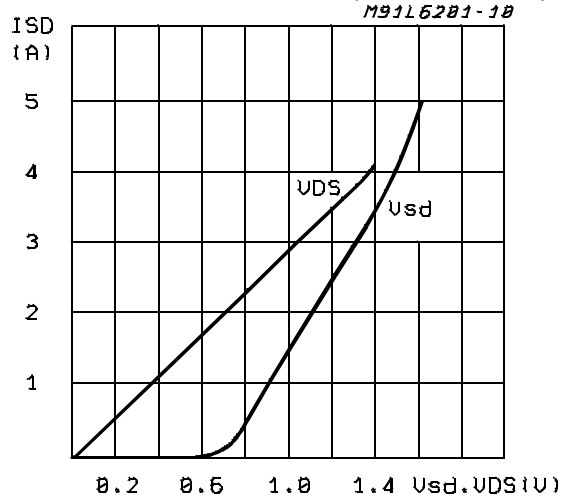


Figure 7a: Typical Power Dissipation vs I_L (L6201)

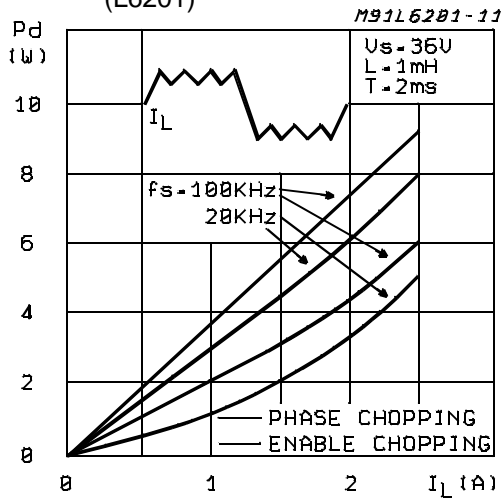


Figure 7b: Typical Power Dissipation vs I_L (L6201PS, L6202, L6203)

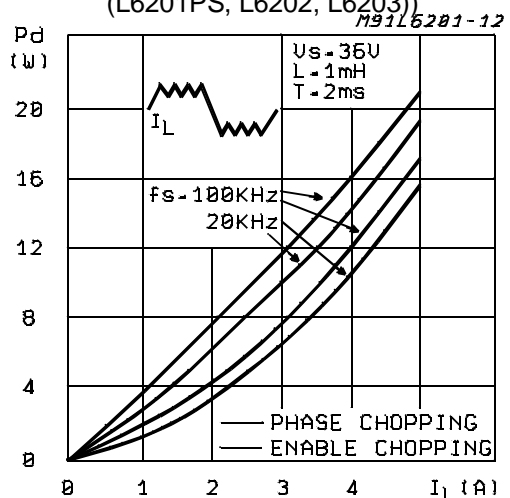


Figure 8a: Two Phase Chopping

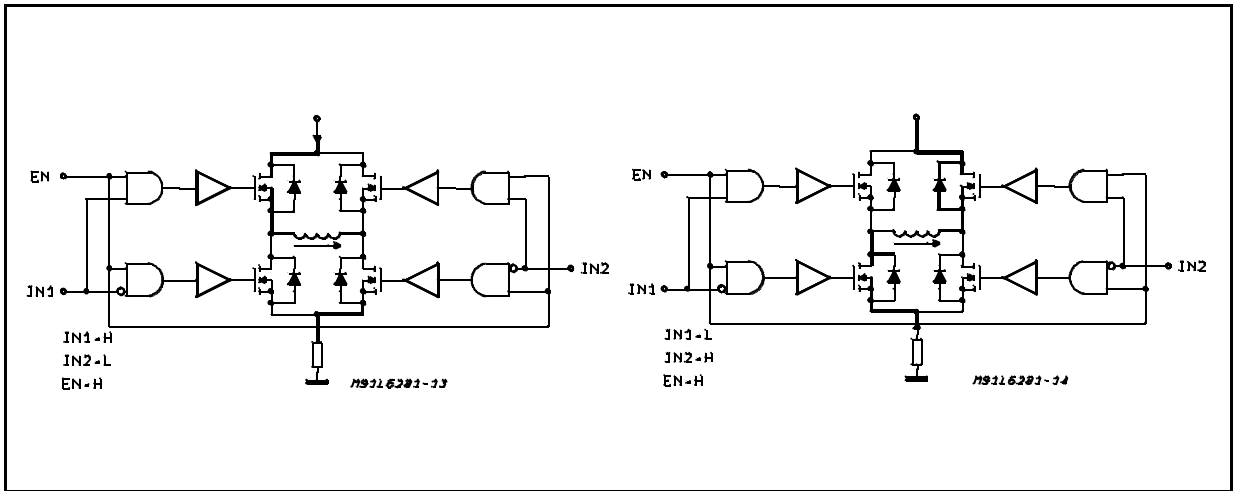


Figure 8b: One Phase Chopping

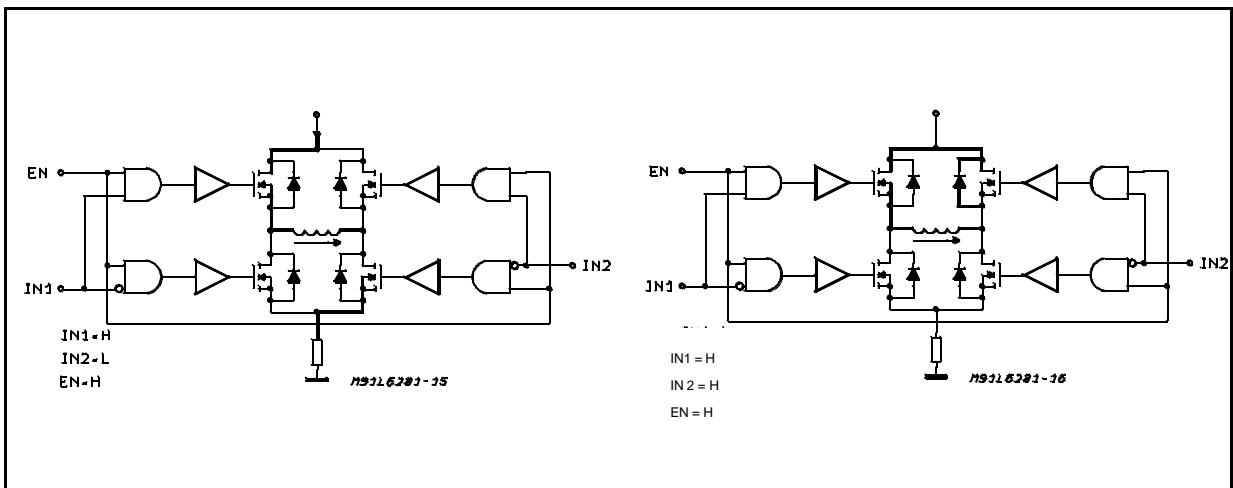
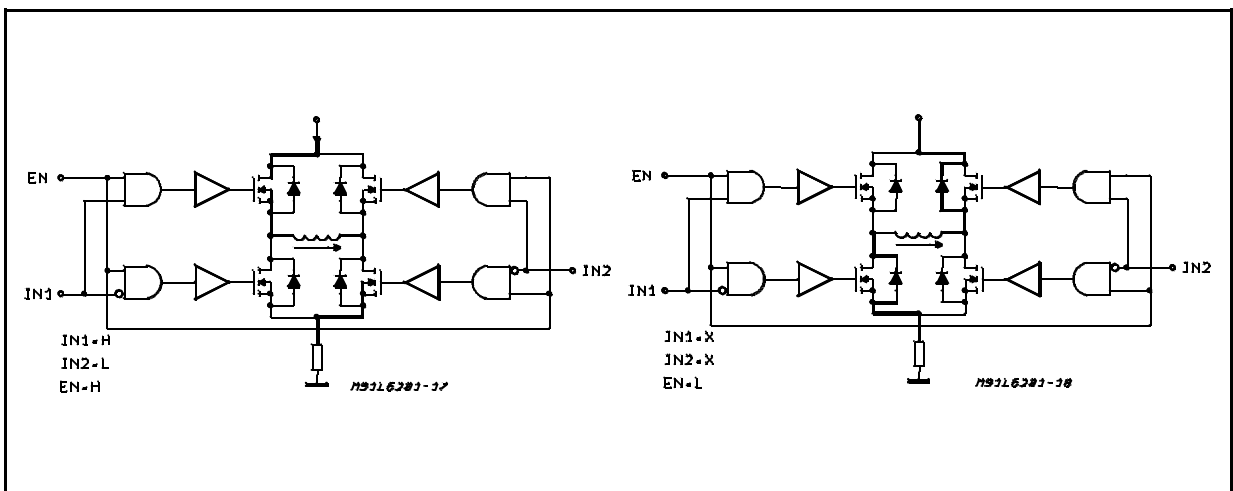


Figure 8c: Enable Chopping



TEST CIRCUITS

Figure 9: Saturation Voltage

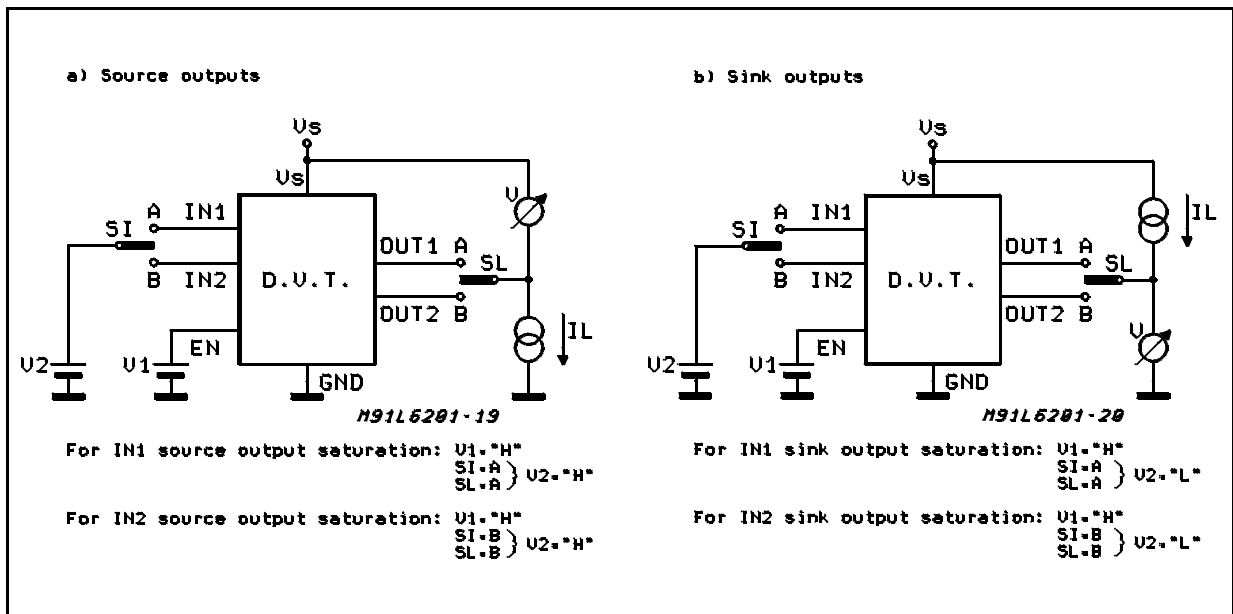


Figure 10: Quiescent Current

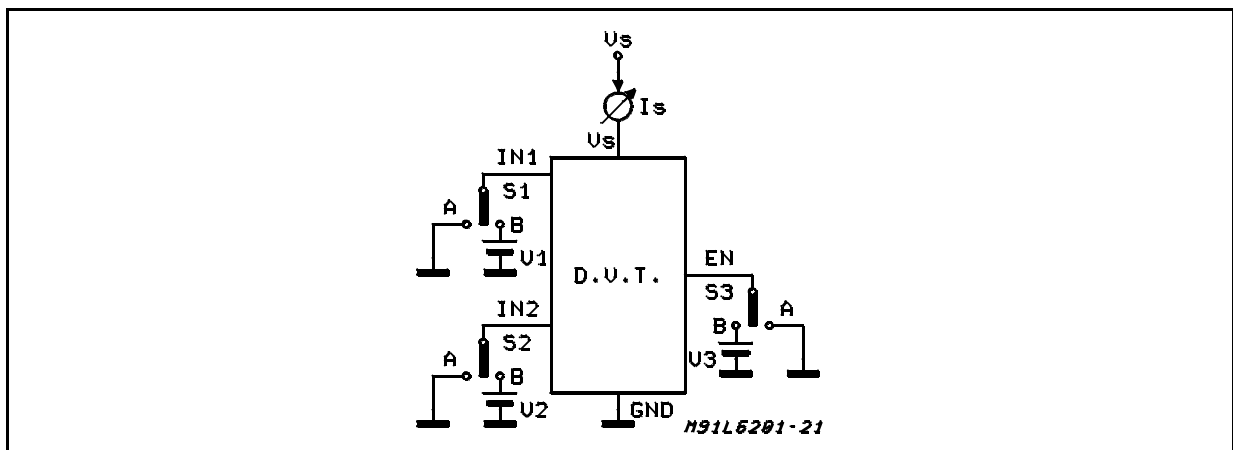


Figure 11: Leakage Current

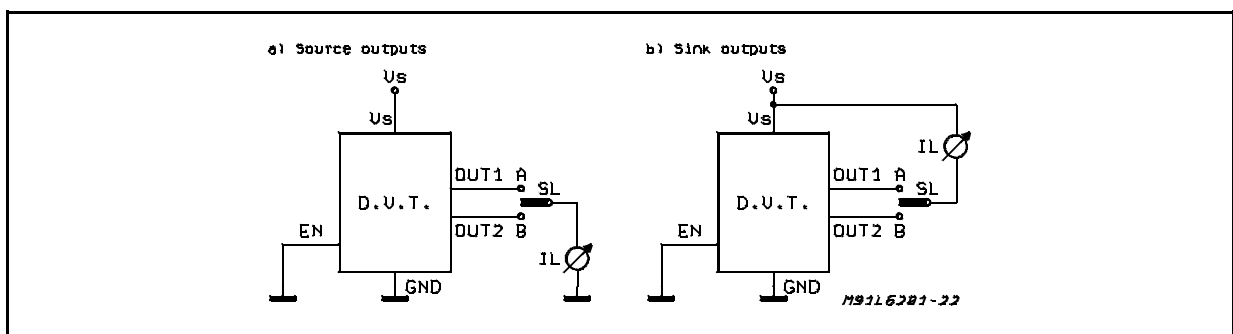


Figure 12: Source Current Delay Times vs. Input Chopper

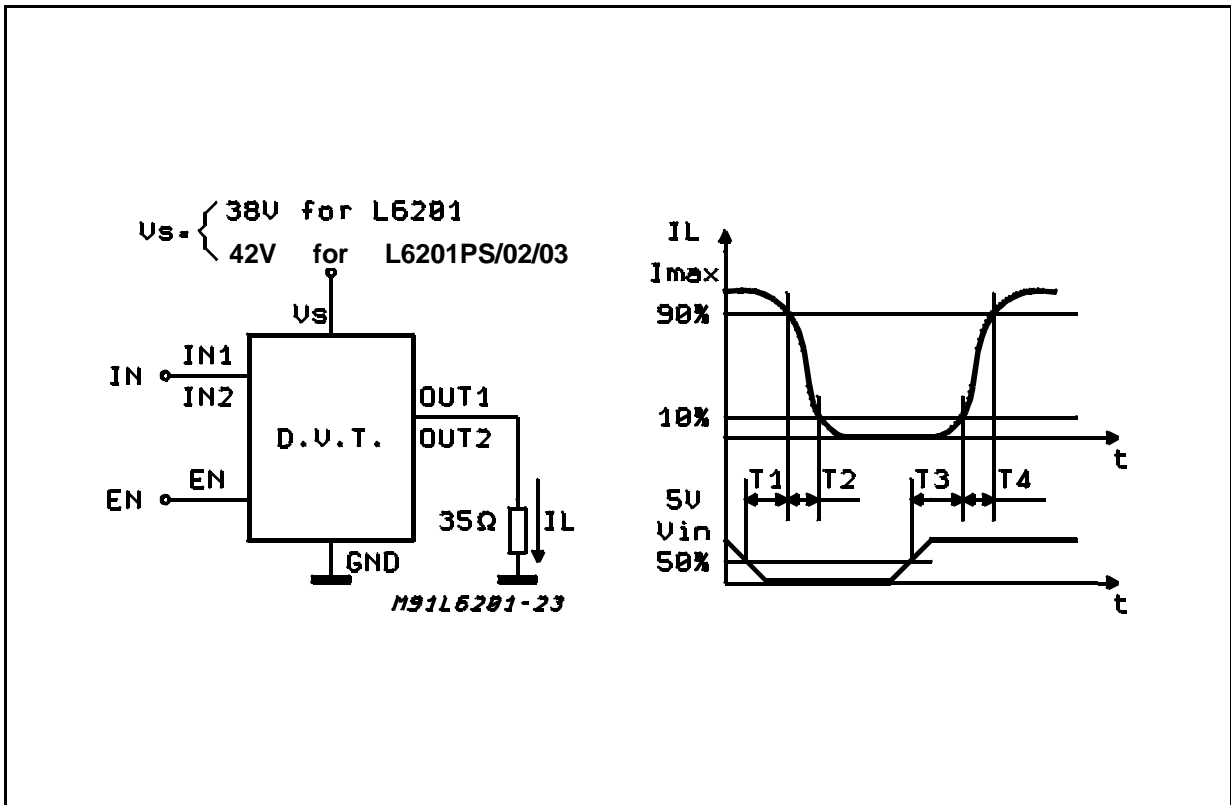
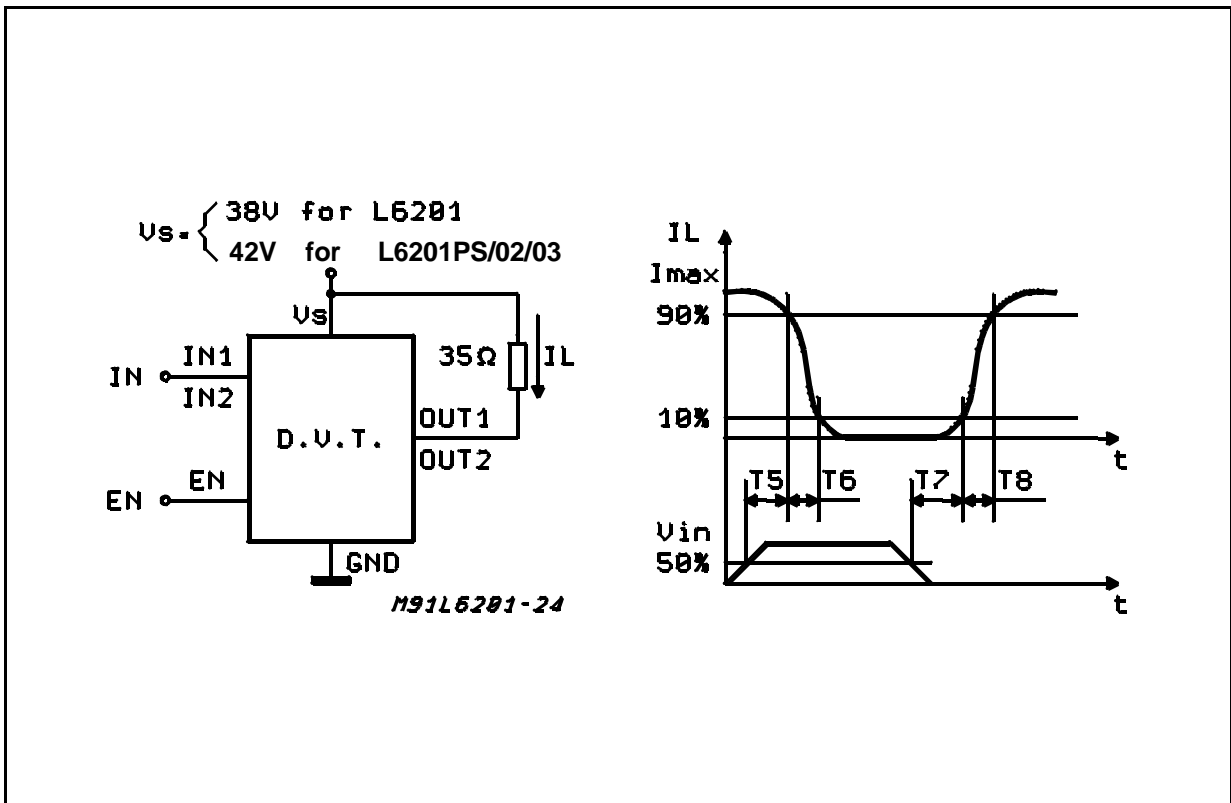


Figure 13: Sink Current Delay Times vs. Input Chopper



CIRCUIT DESCRIPTION

The L6201/1PS/2/3 is a monolithic full bridge switching motor driver realized in the new Multipower-BCD technology which allows the integration of multiple, isolated DMOS power transistors plus mixed CMOS/bipolar control circuits. In this way it has been possible to make all the control inputs TTL, CMOS and μC compatible and eliminate the necessity of external MOS drive components. The Logic Drive is shown in table 1.

Table 1

Inputs			Output Mosfets (*)
	IN1	IN2	
$V_{EN} = H$	L	L	Sink 1, Sink 2
	L	H	Sink 1, Source 2
	H	L	Source 1, Sink 2
	H	H	Source 1, Source 2
$V_{EN} = L$	X	X	All transistors turned OFF

L = Low H = High X = DON't care
 (*) Numbers referred to INPUT1 or INPUT2 controlled output stages

Although the device guarantees the absence of cross-conduction, the presence of the intrinsic diodes in the POWER DMOS structure causes the generation of current spikes on the sensing terminals. This is due to charge-discharge phenomena in the capacitors C1 & C2 associated with the drain source junctions (fig. 14). When the output switches from high to low, a current spike is generated associated with the capacitor C1. On the low-to-high transition a spike of the same polarity is generated by C2, preceded by a spike of the opposite polarity due to the charging of the input capacity of the lower POWER DMOS transistor (fig. 15).

Figure 14: Intrinsic Structures in the POWER DMOS Transistors

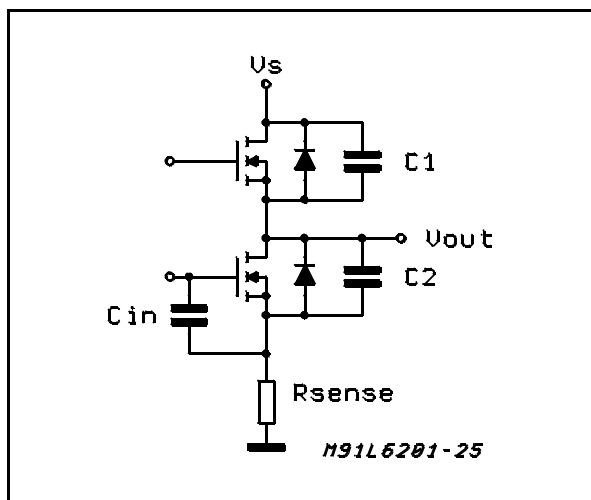
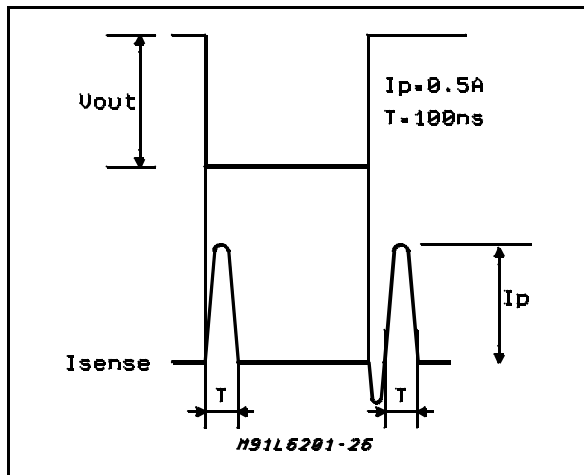


Figure 15: Current Typical Spikes on the Sensing Pin



TRANSISTOR OPERATION

ON State

When one of the POWER DMOS transistor is ON it can be considered as a resistor $R_{DS(ON)}$ throughout the recommended operating range. In this condition the dissipated power is given by :

$$P_{ON} = R_{DS(ON)} \cdot I_{DS}^2 (RMS)$$

The low $R_{DS(ON)}$ of the Multipower-BCD process can provide high currents with low power dissipation.

OFF State

When one of the POWER DMOS transistor is OFF the V_{DS} voltage is equal to the supply voltage and only the leakage current I_{DSS} flows. The power dissipation during this period is given by :

$$P_{OFF} = V_S \cdot I_{DSS}$$

The power dissipation is very low and is negligible in comparison to that dissipated in the ON STATE.

Transitions

As already seen above the transistors have an intrinsic diode between their source and drain that can operate as a fast freewheeling diode in switched mode applications. During recirculation with the ENABLE input high, the voltage drop across the transistor is $R_{DS(ON)} \cdot I_D$ and when it reaches the diode forward voltage it is clamped. When the ENABLE input is low, the POWER MOS is OFF and the diode carries all of the recirculation current. The power dissipated in the transitional times in the cycle depends upon the voltage-current waveforms and in the driving mode. (see Fig. 7ab and Fig. 8abc).

$$P_{trans.} = I_{DS}(t) \cdot V_{DS}(t)$$



Bootstrap Capacitors

To ensure that the POWER DMOS transistors are driven correctly gate to source voltage of typ. 10 V must be guaranteed for all of the N-channel DMOS transistors. This is easy to be provided for the lower POWER DMOS transistors as their sources are referred to ground but a gate voltage greater than the supply voltage is necessary to drive the upper transistors. This is achieved by an internal charge pump circuit that guarantees correct DC drive in combination with the bootstrap circuit. For efficient charging the value of the bootstrap capacitor should be greater than the input capacitance of the power transistor which is around 1 nF. It is recommended that a capacitance of at least 10 nF is used for the bootstrap. If a smaller capacitor is used there is a risk that the POWER transistors will not be fully turned on and they will show a higher RDS (ON). On the other hand if a elevated value is used it is possible that a current spike may be produced in the sense resistor.

Reference Voltage

To by-pass the internal Ref. Volt. circuit it is recommended that a capacitor be placed between its pin and ground. A value of 0.22 μF should be sufficient for most applications. This pin is also protected against a short circuit to ground: a max. current of 2mA max. can be sinked out.

Dead Time

To protect the device against simultaneous conduction in both arms of the bridge resulting in a rail to rail short circuit, the integrated logic control provides a dead time greater than 40 ns.

Thermal Protection

A thermal protection circuit has been included that will disable the device if the junction temperature reaches 150 °C. When the temperature has fallen to a safe level the device restarts the input and enable signals under control.

APPLICATION INFORMATION

Recirculation

During recirculation with the ENABLE input high, the voltage drop across the transistor is RDS (ON) · IL, clamped at a voltage depending on the characteristics of the source-drain diode. Although the device is protected against cross conduction, current spikes can appear on the current sense pin due to charge/discharge phenomena in the intrinsic source drain capacitances. In the application this does not cause any problem because the voltage spike generated on the sense resistor is masked by the current controller circuit.

Rise Time Tr (See Fig. 16)

When a diagonal of the bridge is turned on current begins to flow in the inductive load until the maximum current IL is reached after a time Tr. The dissipated energy EOFF/ON is in this case :

$$E_{OFF/ON} = [R_{DS (ON)} \cdot I_L^2 \cdot T_r] \cdot 2/3$$

Load Time TLD (See Fig.16)

During this time the energy dissipated is due to the ON resistance of the transistors (ELD) and due to commutation (ECOM). As two of the POWER DMOS transistors are ON, EON is given by :

$$E_{LD} = I_L^2 \cdot R_{DS (ON)} \cdot 2 \cdot T_{LD}$$

In the commutation the energy dissipated is :

$$E_{COM} = V_s \cdot I_L \cdot T_{COM} \cdot f_{SWITCH} \cdot T_{LD}$$

Where :

$$T_{COM} = T_{TURN-ON} = T_{TURN-OFF}$$

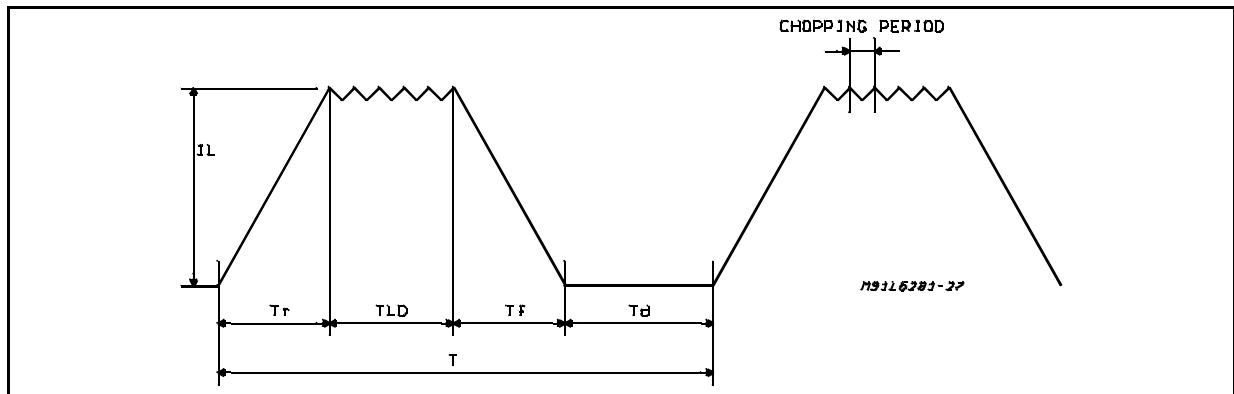
fSWITCH = Chopping frequency.

Fall Time Tf (See Fig. 16)

It is assumed that the energy dissipated in this part of the cycle takes the same form as that shown for the rise time :

$$E_{ON/OFF} = [R_{DS (ON)} \cdot I_L^2 \cdot T_f] \cdot 2/3$$

Figure 16.



Quiescent Energy

The last contribution to the energy dissipation is due to the quiescent supply current and is given by:

$$E_{QUIESCENT} = I_{QUIESCENT} \cdot V_s \cdot T$$

Total Energy Per Cycle

$$E_{TOT} = E_{OFF/ON} + E_{LD} + E_{COM} + E_{ON/OFF} + E_{QUIESCENT}$$

The Total Power Dissipation P_{DIS} is simply :

$$P_{DIS} = E_{TOT}/T$$

- T_r = Rise time
- T_{LD} = Load drive time
- T_f = Fall time
- T_d = Dead time
- T = Period
- $T = T_r + T_{LD} + T_f + T_d$

DC Motor Speed Control

Since the I.C. integrates a full H-Bridge in a single package it is ideally suited for controlling DC motors. When used for DC motor control it performs the power stage required for both speed and direction control. The device can be combined with a current regulator like the L6506 to implement a transconductance amplifier for speed control, as shown in figure 17. In this particular configuration only half of the L6506 is used and the other half of the device may be used to control a second

motor.

The L6506 senses the voltage across the sense resistor R_s to monitor the motor current: it compares the sensed voltage both to control the speed and during the brake of the motor.

Between the sense resistor and each sense input of the L6506 a resistor is recommended; if the connections between the outputs of the L6506 and the inputs of the L6203 need a long path, a resistor must be added between each input of the L6203 and ground.

A snubber network made by the series of R and C must be foreseen very near to the output pins of the I.C.; one diode (BYW98) is connected between each power output pin and ground as well.

The following formulas can be used to calculate the snubber values:

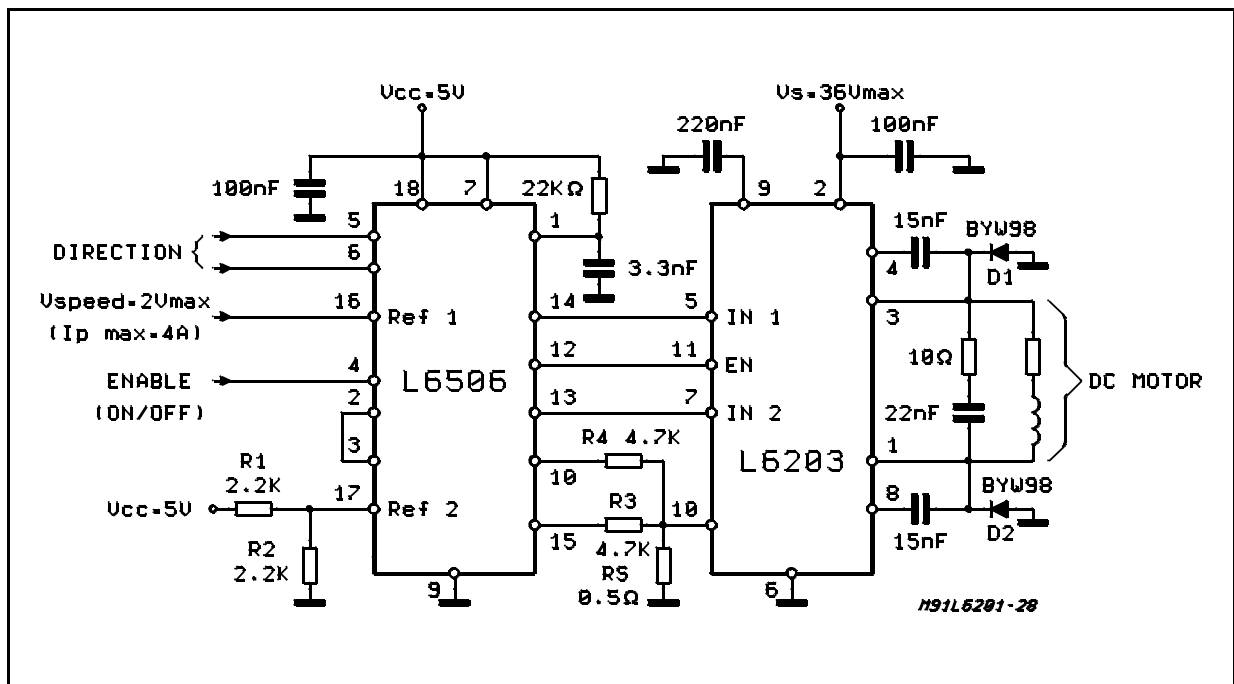
$$R \cong V_s/I_p$$

$$C = I_p/(dv/dt) \text{ where:}$$

V_s is the maximum Supply Voltage foreseen on the application;
 I_p is the peak of the load current;
 dv/dt is the limited rise time of the output voltage (200V/ μ s is generally used).

If the Power Supply Cannot Sink Current, a suitable large capacitor must be used and connected near the supply pin of the L6203. Sometimes a capacitor at pin 17 of the L6506 let the application better work. For motor current up to 2A max., the L6202 can be used in a similar circuit configuration for which a typical Supply Voltage of 24V is recommended.

Figure 17: Bidirectional DC Motor Control



BIPOLAR STEPPER MOTORS APPLICATIONS

Bipolar stepper motors can be driven with one L6506 or L297, two full bridge BCD drivers and very few external components. Together these three chips form a complete microprocessor-to-stepper motor interface is realized.

As shown in Fig. 18 and Fig. 19, the controller connect directly to the two bridge BCD drivers. External component are minimalized: an R.C. network to set the chopper frequency, a resistive divider (R1; R2) to establish the comparator reference voltage and a snubber network made by R and C in series (See DC Motor Speed Control).

Figure 18: Two Phase Bipolar Stepper Motor Control Circuit with Chopper Current Control

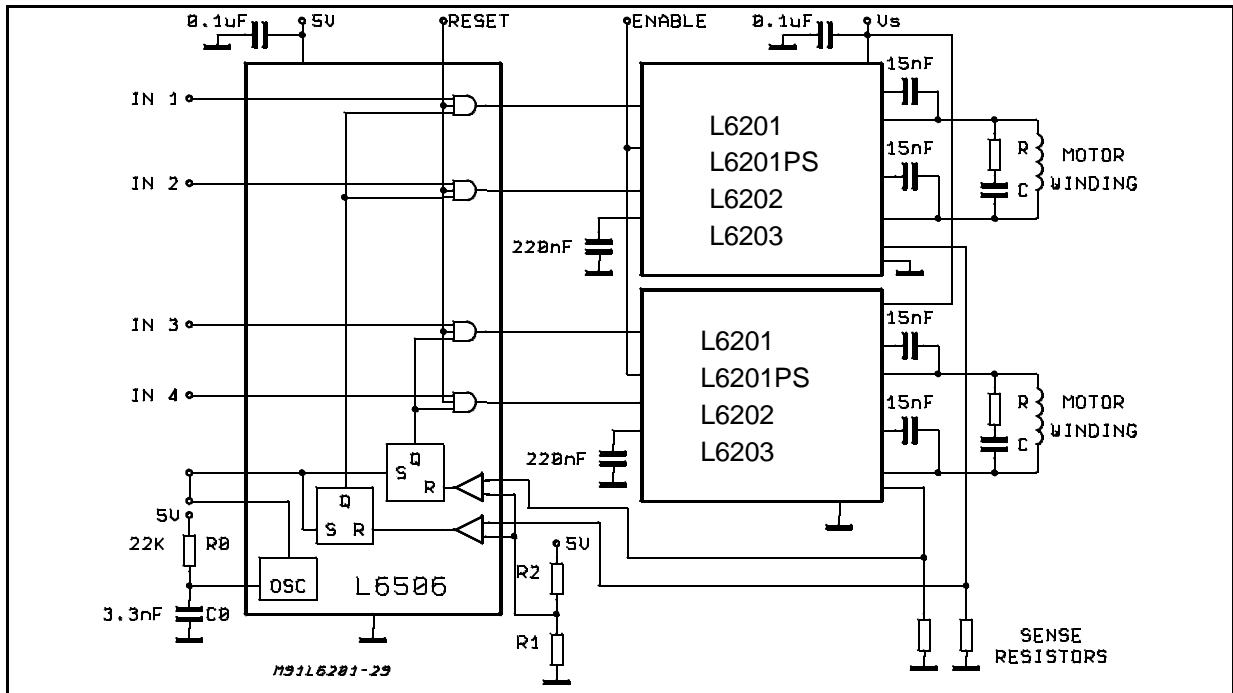
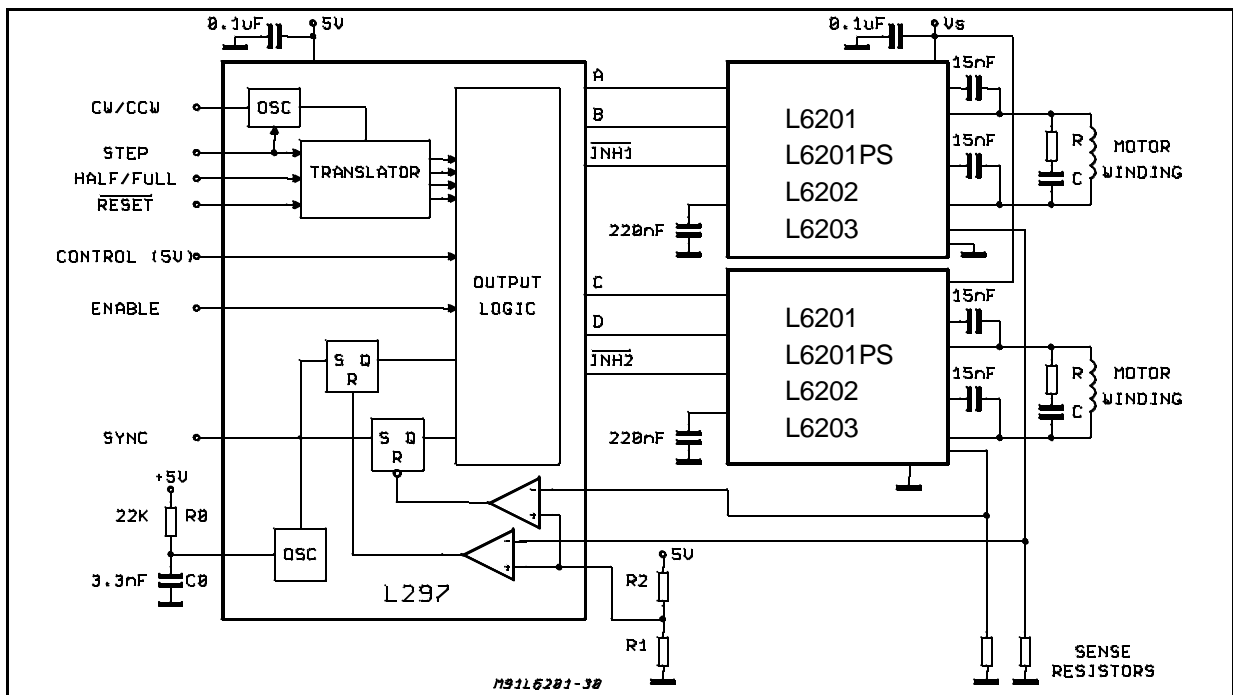
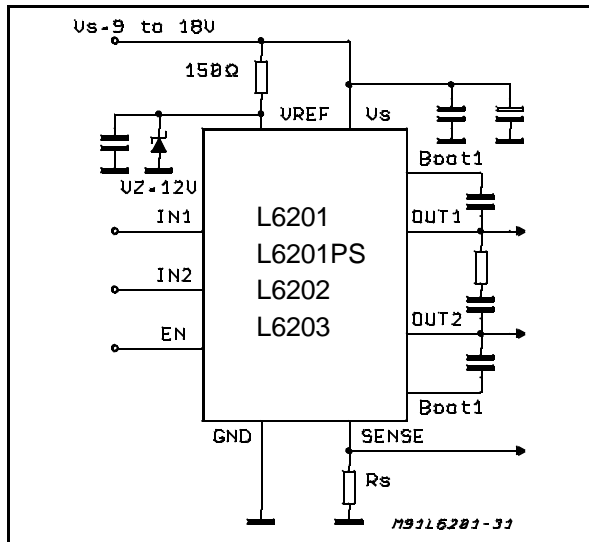


Figure 19: Two Phase Bipolar Stepper Motor Control Circuit with Chopper Current Control and Translator



It could be requested to drive a motor at V_s lower than the minimum recommended one of 12V (See Electrical Characteristics); in this case, by accepting a possible small increase in the $R_{DS(ON)}$ resistance of the power output transistors at the lowest Supply Voltage value, may be a good solution the one shown in Fig. 20.

Figure 20: L6201/1P/2/3 Used at a Supply Voltage Range Between 9 and 18V



THERMAL CHARACTERISTICS

Thanks to the high efficiency of this device, often a true heatsink is not needed or it is simply obtained by means of a copper side on the P.C.B. (L6201/2).

Under heavy conditions, the L6203 needs a suitable cooling.

By using two square copper sides in a similar way as it shown in Fig. 23, Fig. 21 indicates how to choose the on board heatsink area when the L6201 total power dissipation is known since:

$$R_{Th\ j-amb} = (T_{j\ max.} - T_{amb\ max}) / P_{tot}$$

Figure 22 shows the Transient Thermal Resistance vs. a single pulse time width.

Figure 23 and 24 refer to the L6202.

For the Multiwatt L6203 addition information is given by Figure 25 (Thermal Resistance Junction-Ambient vs. Total Power Dissipation) and Figure 26 (Peak Transient Thermal Resistance vs. Repetitive Pulse Width) while Figure 27 refers to the single pulse Transient Thermal Resistance.

Figure 21: Typical $R_{Th\ J-amb}$ vs. "On Board" Heatsink Area (L6201)

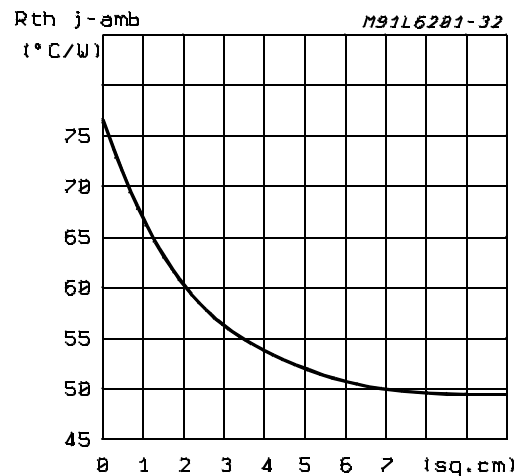


Figure 22: Typical Transient R_{Th} in Single Pulse Condition (L6201)

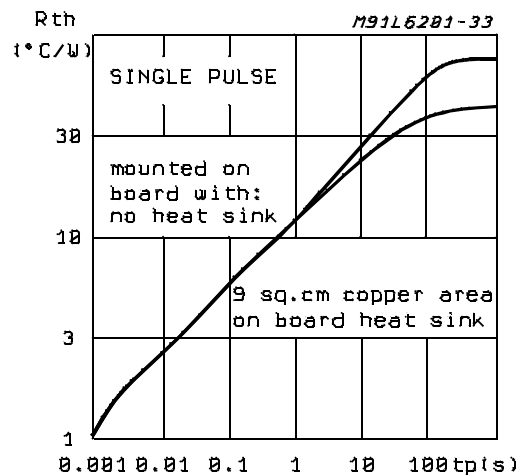


Figure 23: Typical $R_{Th\ J-amb}$ vs. Two "On Board" Square Heatsink (L6202)

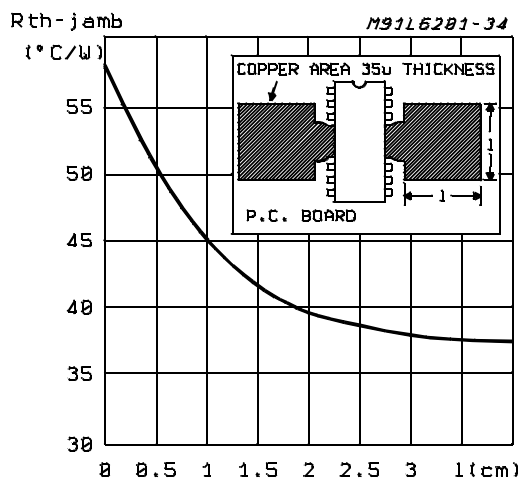


Figure 24: Typical Transient Thermal Resistance for Single Pulses (L6202)

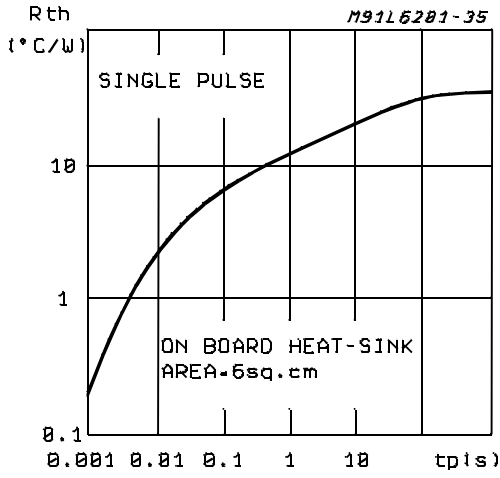


Figure 25: Typical $R_{Th J-amb}$ of Multiwatt Package vs. Total Power Dissipation

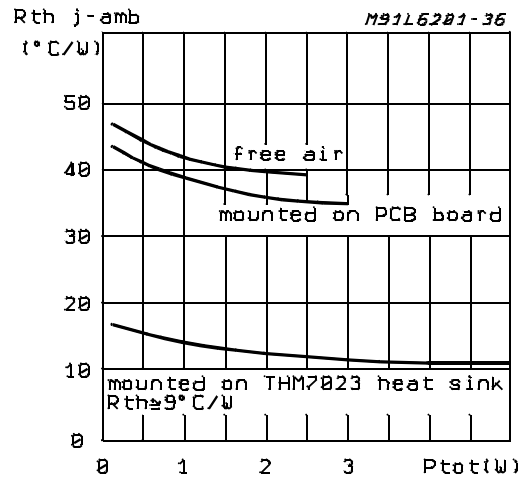


Figure 26: Typical Transient Thermal Resistance for Single Pulses with and without Heatsink (L6203)

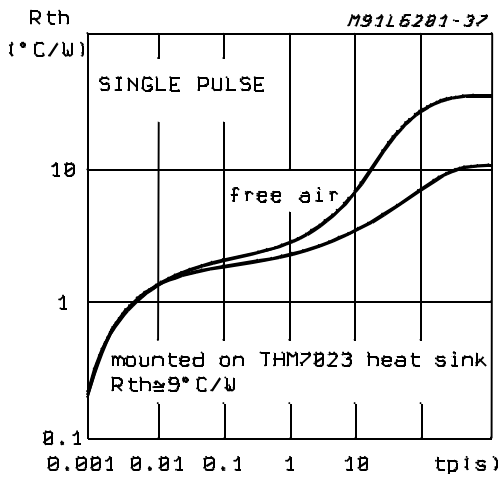
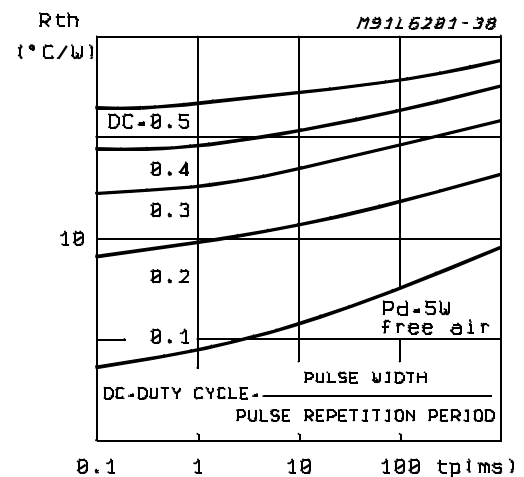
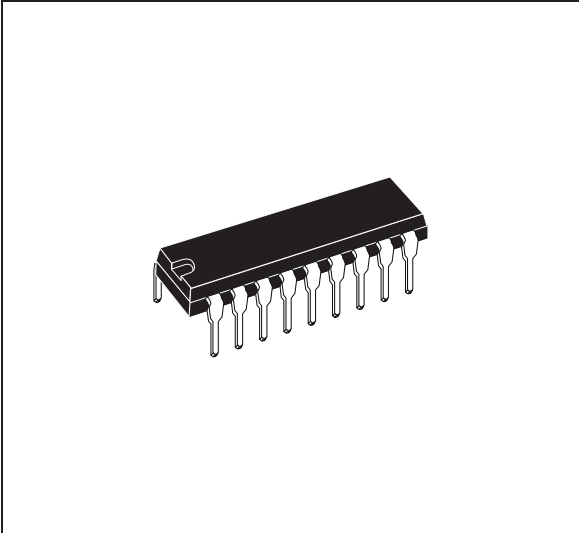


Figure 27: Typical Transient Thermal Resistance versus Pulse Width and Duty Cycle (L6203)

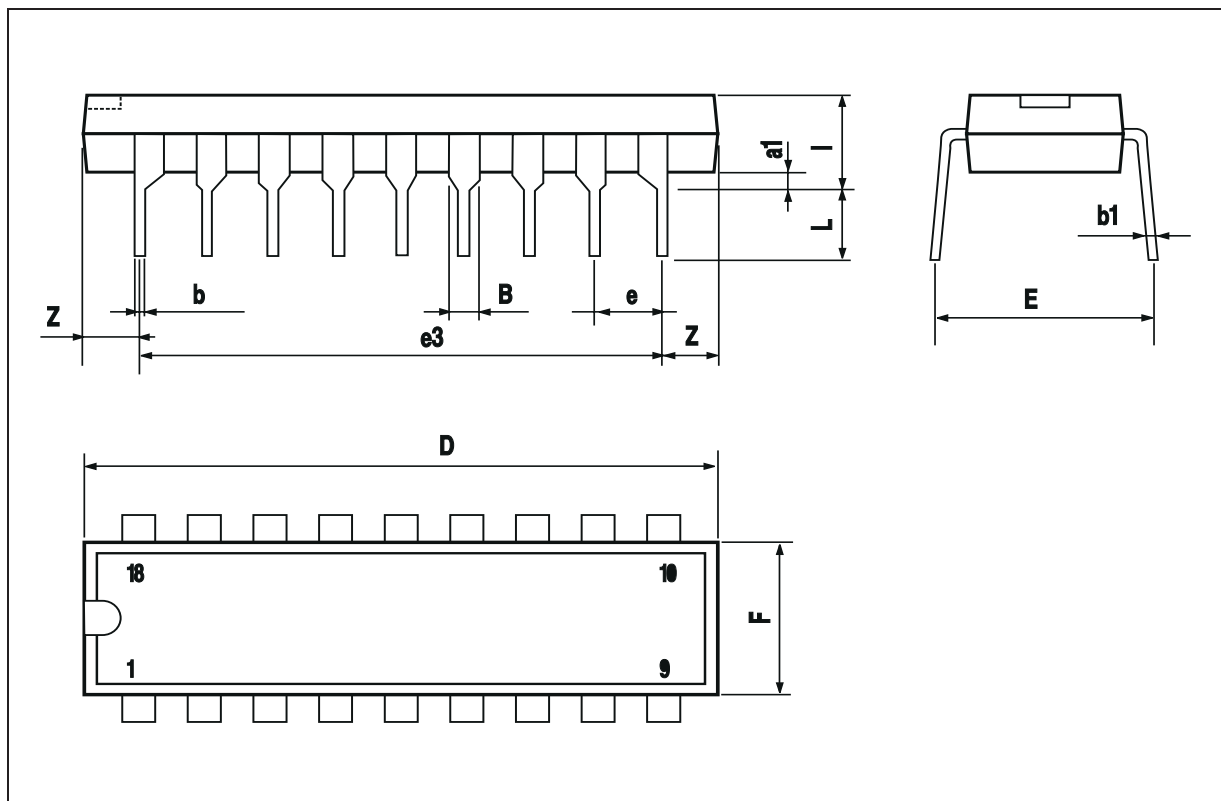


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
a1	0.51			0.020		
B	0.85		1.40	0.033		0.055
b		0.50			0.020	
b1	0.38		0.50	0.015		0.020
D			24.80			0.976
E		8.80			0.346	
e		2.54			0.100	
e3		20.32			0.800	
F			7.10			0.280
I			5.10			0.201
L		3.30			0.130	
Z			2.54			0.100

OUTLINE AND MECHANICAL DATA

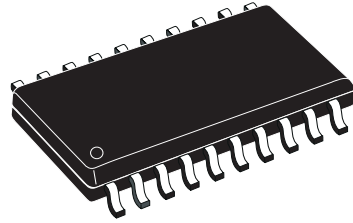


Powerdip 18

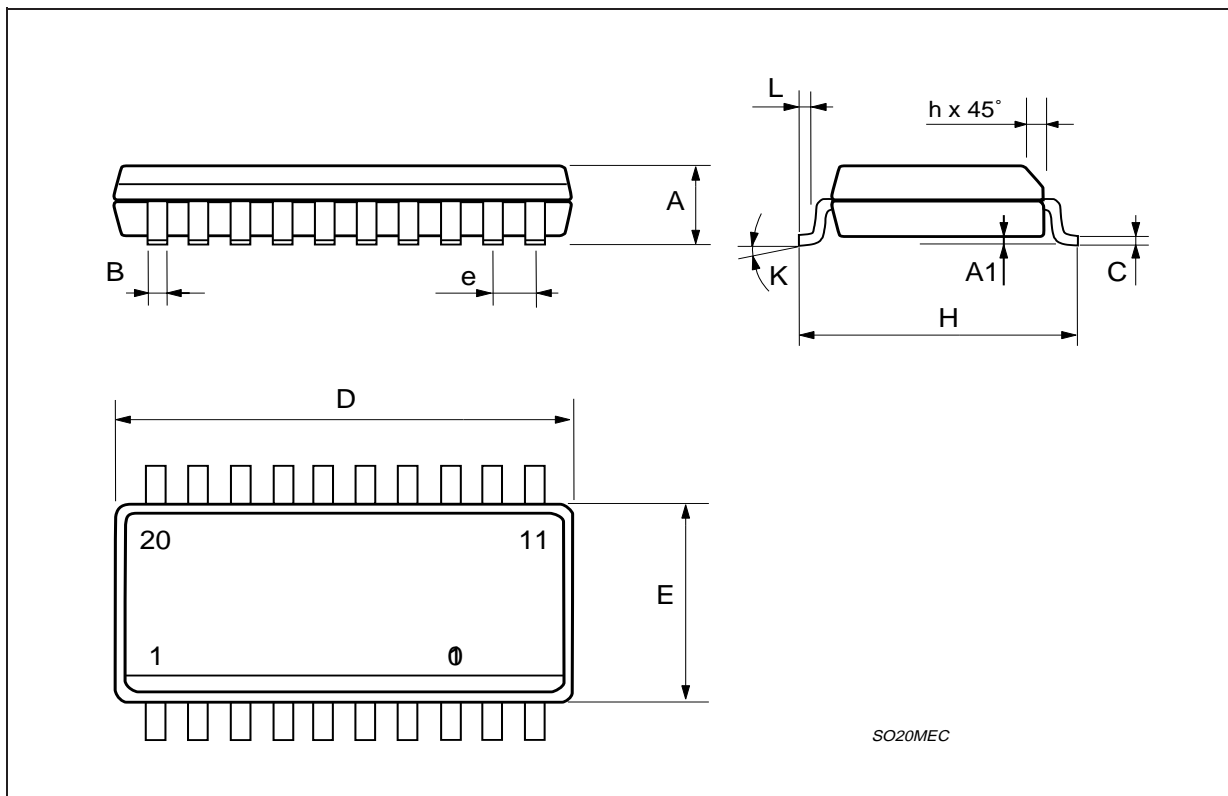


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A	2.35		2.65	0.093		0.104
A1	0.1		0.3	0.004		0.012
B	0.33		0.51	0.013		0.020
C	0.23		0.32	0.009		0.013
D	12.6		13	0.496		0.512
E	7.4		7.6	0.291		0.299
e		1.27			0.050	
H	10		10.65	0.394		0.419
h	0.25		0.75	0.010		0.030
L	0.4		1.27	0.016		0.050
K	0° (min.)8° (max.)					

OUTLINE AND MECHANICAL DATA



SO20



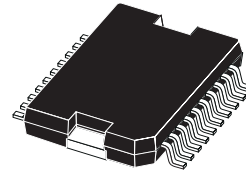
L6201 - L6202 - L6203

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			3.6			0.142
a1	0.1		0.3	0.004		0.012
a2			3.3			0.130
a3	0		0.1	0.000		0.004
b	0.4		0.53	0.016		0.021
c	0.23		0.32	0.009		0.013
D (1)	15.8		16	0.622		0.630
D1	9.4		9.8	0.370		0.386
E	13.9		14.5	0.547		0.570
e		1.27			0.050	
e3		11.43			0.450	
E1 (1)	10.9		11.1	0.429		0.437
E2			2.9			0.114
E3	5.8		6.2	0.228		0.244
G	0		0.1	0.000		0.004
H	15.5		15.9	0.610		0.626
h			1.1			0.043
L	0.8		1.1	0.031		0.043
N	8° (typ.)					
S	8° (max.)					
T		10			0.394	

(1) "D and E1" do not include mold flash or protusions.
 - Mold flash or protusions shall not exceed 0.15mm (0.006")
 - Critical dimensions: "E", "G" and "a3".

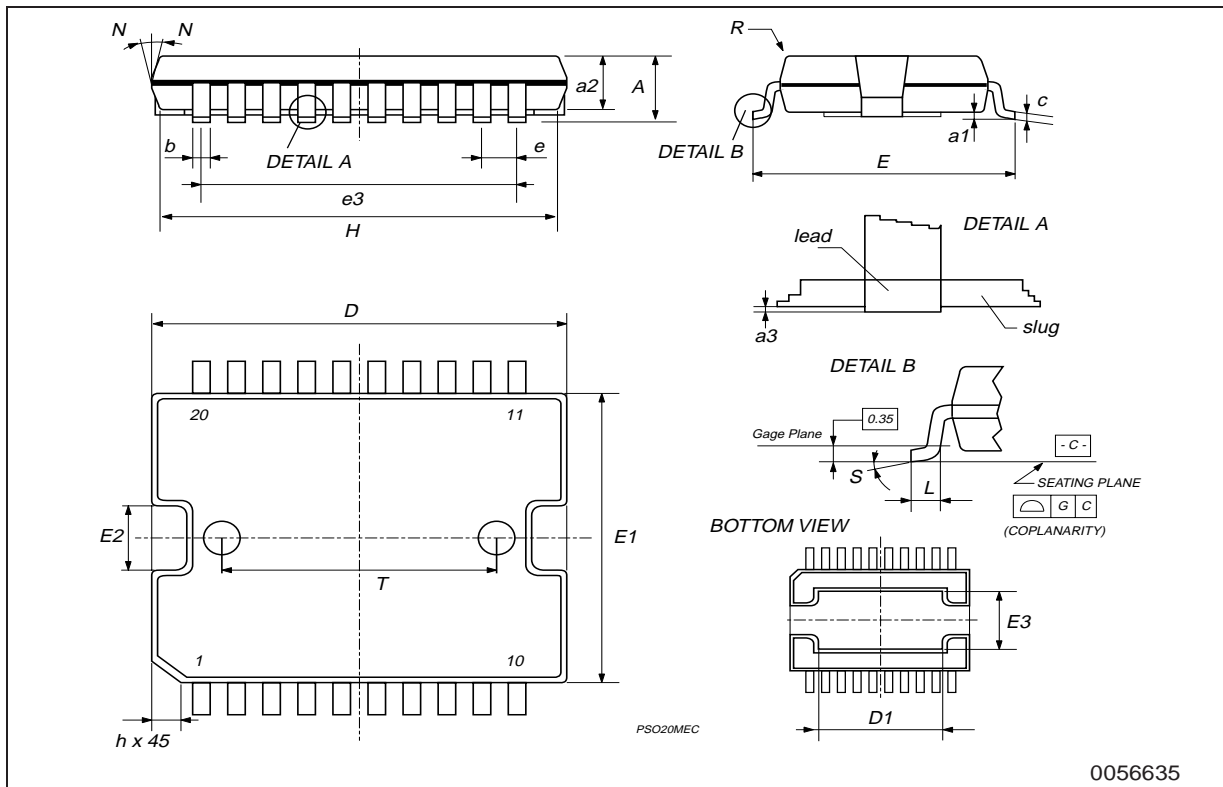
OUTLINE AND MECHANICAL DATA

Weight: 1.9gr



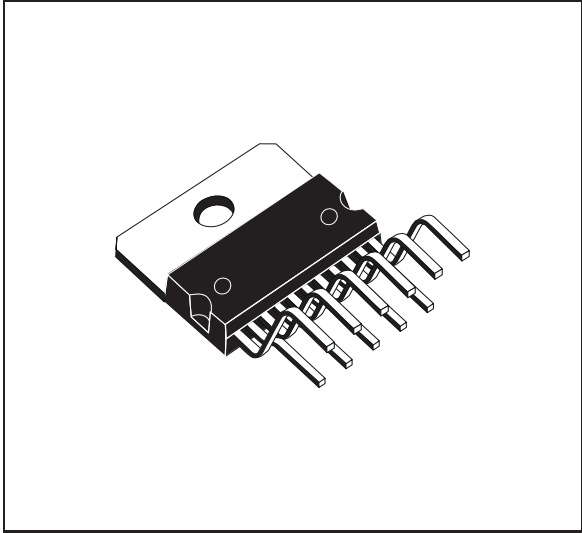
JEDEC MO-166

PowerSO20

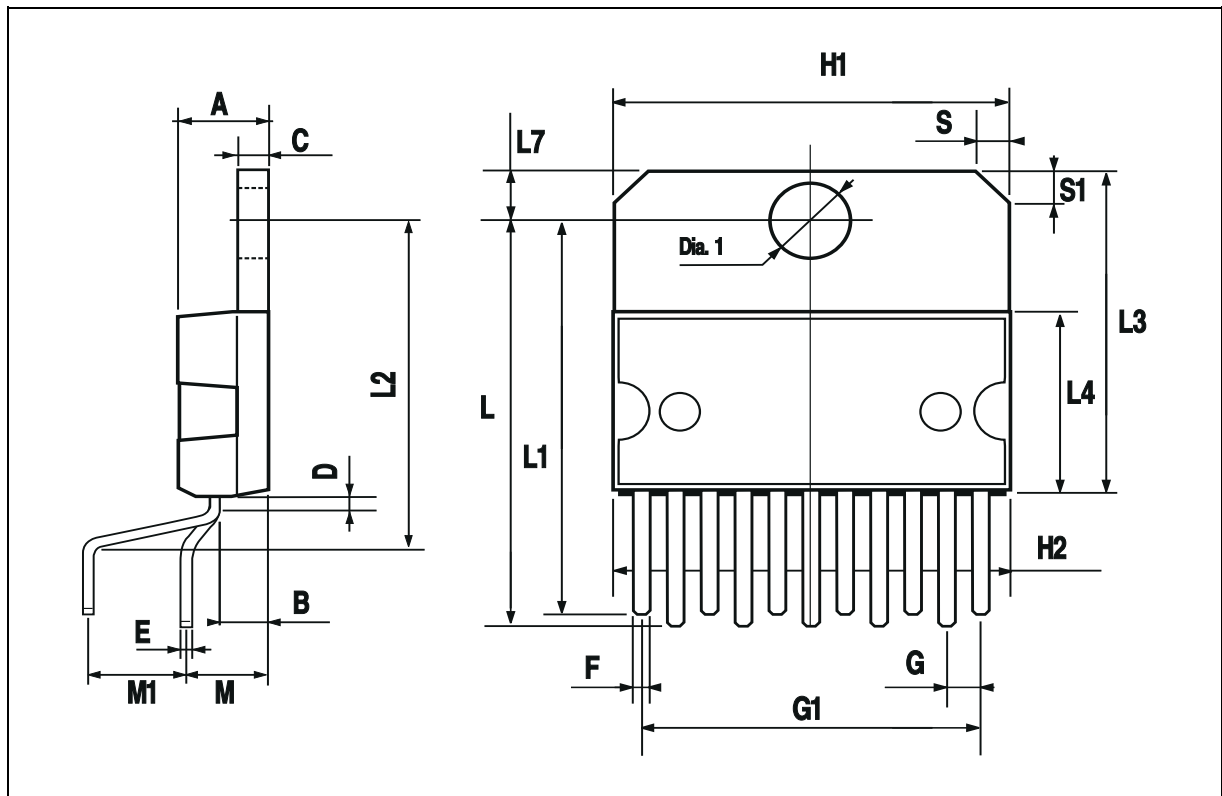


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
D		1			0.039	
E	0.49		0.55	0.019		0.022
F	0.88		0.95	0.035		0.037
G	1.45	1.7	1.95	0.057	0.067	0.077
G1	16.75	17	17.25	0.659	0.669	0.679
H1	19.6			0.772		
H2			20.2			0.795
L	21.9	22.2	22.5	0.862	0.874	0.886
L1	21.7	22.1	22.5	0.854	0.87	0.886
L2	17.4		18.1	0.685		0.713
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L7	2.65		2.9	0.104		0.114
M	4.25	4.55	4.85	0.167	0.179	0.191
M1	4.73	5.08	5.43	0.186	0.200	0.214
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

OUTLINE AND MECHANICAL DATA



Multiwatt11 V



Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specification mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics

© 2003 STMicroelectronics – Printed in Italy – All Rights Reserved

STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States.

<http://www.st.com>

ESPECIFICACIONES PARA LOS MOTORES EMG30 Y PLATAFORMA DE SUJECCIÓN

A.1. EMG30 and Mounting Bracket specification

The EGM30 (encoder, motor and gearbox 30:1) is a 12v motor fully equipped with encoders and a 30:1 reduction gearbox. It is ideal for small or medium robotic applications, providing cost effective drive and feedback for the user. It also includes a standard noise suppression capacitor across the motor windings.



Figura 1.1: Motor EMG30.

A.2. Measurements

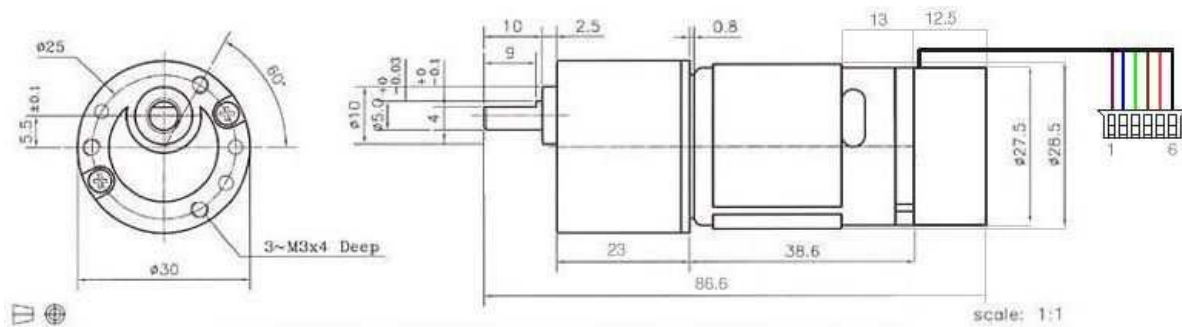


Figura 1.2: Esquema del motor EMG30.

A.3. Connector

The EGM30 is supplied with a 6 way JST connector (part no PHR-6) at the end of approx 90mm of cable as standard. The connections are:

- Wire colour: Connection
- Purple (1): Hall Sensor B Vout
- Blue (2): Hall sensor A Vout
- Green (3): Hall sensor ground
- Brown (4): Hall sensor Vcc
- Red (5): + Motor
- Black (6): - Motor

note that pull up resistors (we used 4k7) are required on the hall sensor outputs, and wires are the colours from the actual cable.

A.4. Specification

- Rated voltage: 12v
- Rated torque: 1.5kg/cm

- Rated speed: 170rpm
- Rated current: 530mA
- No load speed: 216
- No load current: 150mA
- Stall Current: 2.5A
- Rated output: 4.22W
- Encoder counts per output shaft turn: 360

Measured Shaft Speed when used off-load with MD23 and 12v supply:

- Minimum Speed: 1.5rpm
- Maximum Speed: 200rpm

A.5. EMG30 Mounting Bracket

Providing easy mounting of the EMG30 to the robot, the bracket is made from a 2mm thick strong aluminum and finished in blue enamel.



Figura 1.3: Esquema del motor EMG30.

A. ESPECIFICACIONES DEL MOTOR EMG30

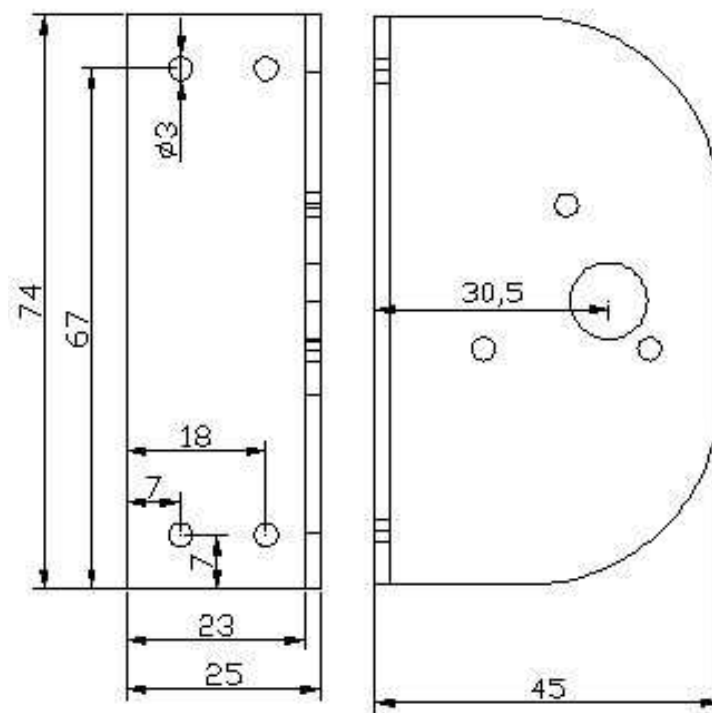


Figura 1.4: Esquema del motor EMG30.

La Bibliographie

[1] Site web : <http://fr.wikipedia.org/wiki/Robot#Historique>

[2] : Robotique mobile. Cours de robotique générale Ecole National supérieure de Physique de Strasbourg Université Louis Pasteur 75p

[3] CHERROUN Lakhmissi , « Navigation Autonome d'un Robot Mobile par des Techniques

[4] Laetitia Matignon, « Introduction a la robotique », Université de Caen, France, 2011/2012

[5] Ph. Bidaud, Cours de Modélisation des Systèmes Robotiques, Université Pierre et Marie Curie. Master Systèmes Intelligents et Robotiques.

[6] : http://www.planete-sciences.org/robot/data/file/trophees/2016/T2016_Rules.pdf

[7] : Christophe Le Lann2, Le PID utilisé en régulation de position et/ou de vitesse de moteurs électriques1