

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique



Université Dr. Tahar Moulay de Saïda
Faculté de la Technologie
Département d'Electrotechnique



Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme de

Master (LMD)

Spécialité : AUTOMATIQUE ET SYSTEME

Filière : AUTOMATIQUE

Intitulé :

**Évaluation pratique des dynamiques d'un robot mobile à direction
différentielle en Navigation dans un milieu sous contraintes**

Présenté par :

➤ MORSLI khayra

Devant le jury composé de :

Dr. MOSTEFAI lotfi

Encadreur

Dr .MOSTEFAI Mohammed

Examinateur

Dr. Mekkaoui Mohamed

Examinateur

Promotion 2019-2020

Dédicace

*Je dédie ce travail à :
ma chère mère et la mémoire de mon père
toute la famille et tous mes amis
tous mes collègues de promotion 2020*

Remerciement

Louange à Allah le très grand et miséricordieux, le seul et unique qui m'a donné la force et le courage pour terminer ma étude et élaborer ce travail.

je tenons à remercier très chaleureusement, en premier lieu, mon encadrant

Dr. MOSTEFAI Lotfi pour son encadrement et tous les efforts qu'il a fournis en vue de me diriger et me aider à m'améliorer, ainsi que pour ses précieux conseils et surtout pour m'avoir laissé une grande liberté dans la conception et la rédaction de ce travail., ses remarques pertinentes et son encouragement.

Je remercie vivement les maîtres de conférences, Monsieur Dr. MOSTEFAI Mohammed , Monsieur Dr. MERAH Abdelkader pour avoir accepté d'examiner mon travail et l'intérêt qu'ils y ont porté.

Enfin, Mes remerciements vont aussi à tous professeurs, enseignants et toutes les personnes qui m'ont soutenus jusqu'au bout, et qui ne cessent de donner des conseils importants

Résumé:

Ce travail consiste à concevoir, réaliser et commander un robot mobile à base différentiel

notre objectif était de rendre le robot plus autonome et interactive, pour cela, nous avons inséré un capteur à ultrasons, qui vise à donner au robot la possibilité d'éviter les obstacles qui lui rencontre dans sa trajectoire et un capteur infrarouge pour suivre la ligne

Mots clés : Robot mobile, capture, ultrasons, infrarouge, suivez la ligne, base différentiel

Abstract:

This work, involves designing, constructing and controlling a differential drive mobile robot.

our aim was to make it more autonomous and interactive, for that, we inserted an ultrasonic sensor, that aims at giving the robot the possibility of avoiding the obstacles that face him in his trajectory and an infrared sensor to follow the line

Key words: Robot mobile, sensor, ultrasonic, infrared, follow the line, differential drive

ملخص:

يتكون هذا العمل من تصميم وبناء والتحكم في روبوت متحرك قائم على التفاضل كان هدفنا هو جعل الروبوت أكثر استقلالية وتفاعلية ، ولهذا قمنا بإدخال جهاز استشعار فوق صوتي ، والذي يهدف إلى منح الروبوت إمكانية تجنب العوائق التي تواجهها في طريقها وجهاز استشعار الأشعة تحت الحمراء لمتابعة الخط

الكلمات المفتاحية : روبوت متنقل ، التقاط ، الموجات فوق الصوتية ، الأشعة تحت الحمراء ، اتبع الخط ، القاعدة التفاضلية

SOMMAIRE

Liste des figures	
Glossaire	
Introduction générale.....	1
Chapitre01: Généralités sur la robotique	
I.1.Introduction.....	3
I.2 la robotique	3
I.2.1 Définition de la robotique	3
I.2.2 Disciplines de la robotique.....	3
I.2.2.1 Mécanique.....	3
I.2.2.2 Electronique.....	3
I.2.2.3 Informatique.....	3
I.2.2.4 Automatique.....	3
I.2.2.5 Traitement du signal.....	3
I.2.2.6 Mathématiques.....	3
I.2.2.7 Sciences cognitives.....	3
I.3 Définition d'un robot	3
I.4.1 Historique.....	4
I.4.2 Les robots mobiles.....	5
I.4.3 Entre 2010 et 2019	5
I.5 Domaine d'utilisation des robots	6
I.5.1 Domaine industriel.....	6
I.5.2 Domaine spatial.....	6
I.5.3 Domaine agriculture.....	7
I.5.4 Robots domestiques ou ménagers	7
I.5.5 Domaine militaire.....	8
I.5.6 Domaine médical	8
I.6 Caractéristiques d'un robot.....	9
I.7 Classifications des robots.....	9
I.7.1 Par génération.....	9
I.7.1.a Le robot passif	9
I.7.1.b Le robot actif.....	9
I.7.1.c Le robot intelligent	9
I.7.2 Par type.....	9
I.7.2.a Les manipulateurs	9
I.7.2.b Télémanipulateurs	10
I.7.2.c Robots mobiles.....	10
I.7.2.c.1 Définition	10
I.7.2.c.2 Présentation générale des robots mobiles	11
I.7.2.c.3 Classification.....	11
I.7.2.c.4 Architecture d'un robot mobile.....	12
I.7.2.c.5 La structure mécanique et motricité.....	13
La structure mécanique.....	13

I.7.2.c.6 Traitement des informations et gestion des taches.....	16
I.7.2.c.7 Les avantages dans l'utilisation des robots mobiles.....	16
I.8 Conclusion.....	17

Chapitre02: Modélisation d'un robot mobile

II.1 Introduction	19
II.2 Les robots mobiles non-holonomes.....	19
II.3 Définition	19
II.4 Modélisation du robot unicycle	19
II.4.1 Modélisation géométrique.....	20
II.4.2 Modélisation cinématique	22
II.4.2.1 Le Modèle d'actionnement.....	24
II.4.2.2 modèle cinématique en posture de véhicule.....	24
II.4.3 modèle cinématique inverse.....	26
II.4.4 cinématique du robot différentiel (differential drive).....	26
II.4.4 .1 Localisation odométrique	28
II.4.5 Modélisation d'actionneur.....	29
II.4.6 Le modèle complet	31
II.4.6.1 Identification des paramètres.....	32
II.4.6.2 Résultat de simulation	33
II.5 Modélisation dynamique	34
II.5.1 L'approche dynamique de Lagrange.....	35
II.5.2 La simulation de modèle dynamique.....	39
Le modèle dynamique directe.....	39
Le modèle global.....	39
II.5.3 Résultat de simulation.....	40
Conclusion.....	40

Chapitre03: Description matérielle et logiciel

Introduction.....	42
III.1 description matériel	43
III.1.1 Arduino	43
III.1.2 Raspberry Pi.....	44
Les composantes du Raspberry Pi.....	44
III.1.3 Moteurs à courant continu.....	45
III 1.4 Le Driver L298	45
Description L298xx (H-Bridge Motor Driver.....	46
Principe de fonctionnement du pilote (Command du L298N).....	46
III.1.5 Servomoteur	47
La commande du servomoteur	48
Caractéristique du servomoteur utilisé dans ce projet.....	48
III 1.6 Capteur ultrason.....	49
Le fonctionnement du module est le suivant	49
III.1.7 Les capteurs à infrarouge	50
III.2 description logiciel.....	51
III .2.1 Arduino IDE.....	51

L'interface de logiciel.....	51
III.2.2 Langage de programmation Python.....	53

Chapitre04:La réalisation de robot

IIV.1 Introduction	55
IIV.2 Cahier de charge.....	55
IIV.3 MODE 01 : suiveur de ligne	56
IIV.3 .1La détection de ligne et mouvement de robot	56
IIV.3 .2 Organigramme pour suivre de trajectoire	57
IIV.3 .3 test de capteur infrarouge.....	58
circuit d'infrarouge.....	58
Le code Arduino pour test e capteur infrarouge et la résulta.....	59
Test final	59
IIV.3 .4 Organigramme pour évitement des obstacles	60
IIV.3 .5 test de capteur ultrason.....	61
Circuit d'évitement d'obstacle.....	61
IIV.3 .6 Le code Arduino pour test e capteur ultrason et la résulta.....	61
IIV.4 MODE 02 : navigation avec GPS.....	62
IIV.4 .1 La communication entre le robot mobile et application andriod.....	62
Que signifie MQTT.....	63
Les codes python pour MQTT	63
1 code python pour les donnas de la localisation	63
2. code python pour les donnas de trajectoire	64
IIV.4 .2 Le robot mobile et serveur web.....	65
1.La création d'une page Web de suivi de la géolocalisation en temps réel.....	65
2.Code python pour le Raspberry pi.....	66
Conclusion.....	66
Conclusion générale.....	68
REFERENCES BIBLIOGRAPHIE	69

Liste des figures

- Figure I.1 :** Robot en 1920 "Rossum's Universel Robots".
- Figure I.2 :** Unimate
- Figure I.3.** Puma
- Figure I.4:** Premier robots mobiles
- Figure I.5:** RoboCup.
- Figure I.5:** Robot kengoro
- Figure I.6:** Les robots industriels.
- Figure I.6:** Robot Rover Martien.
- Figure I.7:** Robot utilisé en agricole
- Figure I.7:** Robots domestiques ou ménagers
- Figure I.8:** Robot militaire
- Figure I.9:** Robot utilisé dans le domaine militaire
- Figure I.10:** Constitution d'un robot manipulateur
- Figure I.11:** télémanipulateur- A1000[6].
- Figure I.12:** Structure d'un robot mobile.
- Figure I.13:** Véhicule autonome
- Figure I.14:** L'architecture fonctionnelle du robot mobile
- Figure I.15:** Robot à roue
- Figure I.16:** Robot à chenille
- Figure I.17 :** Robot marcheur
- Figure I.18:** robot mobile rampant
- Figure II.1 :** Véhicule unicycle(Robot Pioneer-3DX)
- Figure II.2 :** Paramétrage du Robot mobile à entraînement différentiel
- Figure II.3:** Roulement sans glissement
- Figure II.4:** Relation entre les coordonnées du point A et les point de contacts des roues.
- Figure II.4:** Modèle cinématique directe
- Figure II.5:** Modèle cinématique en posture de véhicule
- Figure II.6:** modèle cinématique inverse
- Figure II.7:** Vecteurs d'erreur de la position et de l'orientation du robot
- Figure II.8:** Bloc moteur
- Figure II.9:** Le modèle du robot unicycle
- Figure II.10:** Le modèle global du robot unicycle
- Figure II.11:** le tableau de paramètre de moteur
- Figure II.12:** vitesses linéaire V et la vitesse angulaire ω du robot
- Figure II.13:** vitesses angulaires des roues $\dot{\phi}_R$ et $\dot{\phi}_L$
- Figure II.14:** trajectoire de robot
- Figure II.15:** le modèle dynamique directe
- Figure II.16:** le modèle global
- Figure II.17:** la vitesse linéaire V et la vitesse angulaire ω du robot
- Figure II.16:** la vitesse linéaire V et la vitesse angulaire $\dot{\phi}_L$ $\dot{\phi}_R$
- Figure III.1:** Schéma des différentes unités du robot
- Figure III.2:** Types de cartes Arduino
- Figure III.3:** Raspberry Pi
- Figure III.4:** Emplacement des ports et interfaces matérielles du Raspberry pi
- Figure III.5:** les quatre moteur DC
- Figure III.6:** Module a base de circuit L298
- Figure III.7:** Circuit d'un pont en H.

Figure III.8:Principe de fonctionnement du pilote (Command du L298N).
Figure III.9: le servomoteur SG-90
Figure III.10 : Signal modulé en code d'impulsion
Figure III.11: Capteur Ultrason HC-SR04
Figure III.12: Fonctionnement du capteur ultrason
Figure III.13: capteur à infrarouge
Figure III.14: disposition des capteurs
Figure II.15: Interface de logiciel Arduino.
Figure III.16: Détail de barre de bouton
Figure II.1:La disposition du robot sur la ligne.
Figure II.2: Organigramme de suivi de ligne.
Figure II.3: Schéma du capteur infrarouge avec Arduino
Figure II.4: Le code Arduino pour test e capteur infrarouge et la résulta
Figure II.4: les test final pour un suivre de trajectoire
Figure II.5: Schéma du capteur ultrason avec Arduino
Figure II.6: Le code Arduino pour test e capteur ultrason et la résulta
Figure II.7: schéma synoptique de notre robot mobile.
Figure II.8: schéma synoptique de MQTT
Figure II.9:code python pour les donnas de la localisation
Figure II.10:code python pour les donnas de trajectoire
Figure II.11: l' application mobile
Figure II.12: les donnas de trajectoire
Figure II.13: publish key , subscribe key et channel name en code HTML
Figure II.14: serveur web
Figure II.15: publish key , subscribe key et channel name en code python

Introduction générale

L'homme a toujours souhaité se libérer du travail dans ce qu'il a d'astreignant, de fatigant, d'inintéressant. Les progrès qu'on connue les technologies depuis un demis siècle sont a même de répondre en bonne parti à son désir par l'intermédiaire de deux outils générique: l'ordinateur pour les travaux mentaux et le robot pour les travaux physiques. Le premier a pris une certaine avance sur le second grâce à la découverte des semi-conducteurs, ces matériaux solides qui peuvent trier des informations sans mobilité mécanique. Le second est toujours tributaire de la vieille mécanique aux pièces fragiles et mobiles et au cout significatif.

Malgré son aspect récent, la robotique tire ses origines des civilisations les plus antiques. Il est peut être même vrai que toutes les autres sciences ont existé juste pour permettre à la robotique de se développer afin d'aider l'Homme à créer ce « esclave » qui appliquera ses ordres au doigt et à l'œil et qui le libérera à jamais du travail.

De l'Homme préhistorique qui s'empessa d'inventer des extensions technologiques de ses membres afin de faciliter son travail, comme la massue par exemple, jusqu'au robot mobile autonome le plus performant, la robotique mobile n'a cessé de connaître des progrès spectaculaires. En effet, les générations de robots mobiles se sont vite succédé offrant à chaque fois plus de performances en matière de perception, d'autonomie et de pouvoir décisionnel. Ces progrès étaient un résultat inévitable des développements qu'ont connu la mécanique, la microélectronique et l'informatique. C'est pour cela qu'en très peu de temps, les générations de robots se succèdent pour voir naître une branche de la robotique visant de plus amples horizons : c'est l'ère de la robotique mobile.

Après cette introduction, le travail est organisé comme suit :

Dans le chapitre 1 : On présente des généralités sur la robotique (historique de la robotique), les différents types des robots et des concepts plutôt spécifiques à la robotique mobile

Dans la partie suivante, on se focalise sur la modélisation géométrique, cinématique et dynamique d'un robot mobile de type uni cycle.

La troisième partie sera consacrer à la description des outils softwares et hardwares utilisés lors de la réalisation du projet de notre robot mobile. Et finalement, on expliquera les différentes étapes de la mise en œuvre et la réalisation du robot mobile connecté en navigation dans un environnement sous contraintes.

chapitre 01

I.1 Introduction:

L'objet de la robotique est l'automatisation de systèmes mécaniques. En dotant le système de capacités de perception, d'action et de décision, l'objectif est de lui permettre d'interagir rationnellement avec son environnement, et de façon autonome. La robotique est un domaine de recherche qui se situe au carrefour de l'intelligence artificielle, de l'automatique, de l'informatique et de la perception par ordinateur ; cette interdisciplinarité est à l'origine d'une certaine complexité. Des applications dans des domaines aussi variés que l'industrie manufacturière, le spatial, l'automobile ou plus récemment les loisirs et le secteur médical, démontrent aujourd'hui l'intérêt économique et sociale. Dans ce chapitre on présentera une généralité sur la robotique (historique de la robotique), ensuite, nous citerons les différents types des robots. et on intéresserons sur les notions de base de la robotique mobile.

I.2 la robotique :

I.2.1 Définition de la robotique :

La branche de la technologie qui traite la conception, la construction, l'exploitation l'application des robots [1].

I.2.2 Disciplines de la robotique:

Mécanique

Conception, réalisation, modélisation des robots.

Electronique

Mise en place de composants pour les robots, télécommunications, communications, commande, acquisition des données.

Informatique

Gestion des fonctionnalités des robots et de transfert d'information.

Automatique

Commande, calibrage des capteurs, des effecteurs, identification des paramètres

Traitement du signal

Analyse des informations enregistrées par les capteurs du robot.

Mathématiques

Modèles mathématiques pour la prise de décision ou/et l'apprentissage, calcul de trajectoires, localisation, planification.

Sciences cognitives

Interactions homme-machine, machine-machine, prise de décision.

1.3 Définition d'un robot :

étymologie : le mot tchèque robota (travail).

Définition : un robot est un système mécanique poly articulé mû par des actionneurs et commandé par un ordinateur qui est destiné à effectuer une grande variété de tâches.[2]

1.4.1 Historique:

- XVIII^{ème} siècle : création d'automates. En 1738, Jacques Vaucanson crée un canard articulé en cuivre capable de boire, manger, cancaner et digérer comme un véritable animal.
- 1921 : Karel Capek , RUR (Rossum's Universal Robots).[2]



Figure I.1 : Robot en 1921 "Rossum's Universel Robots".

- 1947 : premier manipulateur électrique téléopéré.
- 1954 : premier robot programmable

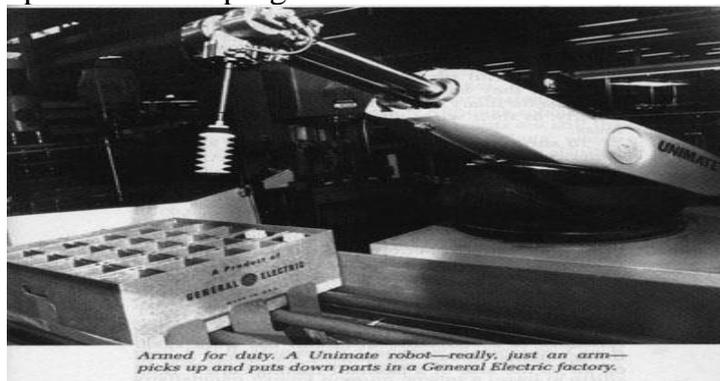


Figure I.2 :Robot en 1961 "Unimate"

- 1963 : utilisation de la vision pour commander un robot.
- 1972 : Nissan ouvre la première chaîne de production complètement robotisée.[3]

I.4.2 Les robots mobiles:

- Fin des années 60 : Mise en place de "Shakey" premier robot mobile intégrant perception, planification et exécution.
- 1977 : premier robot mobile français HILARE au LAAS (CNRS Toulouse)



Figure I.4: Premier robots mobiles.

- 1995 : Mise en place de la RoboCup



Figure I.5: RoboCup

I.4.3 Entre 2010 et 2019 :

Cette période est riche invention et on ne peut pas mentionner tous en raison des différents domaines et le nombre important, il y avait des inventions et un développement de nombreux types de robots et dans différent domaine, les plus récents sont des robots humanoïde basée sur l'intelligence artificielle telle que le robot kenshiro et kengoro, ce derniers est développer par l'université de Tokyo, capable de transpirer quand il fait du sport un moyen pour refroidir ces circuits.[4]

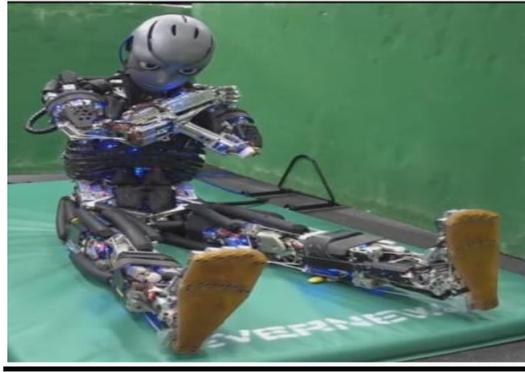


Figure I.5: Robot kengoro

1.5 Domaine d'utilisation des robots :

La robotique est un domaine en plein essor depuis quelques années, utilisée dans des domaines extrêmement rigoureux et exigeants. Nous allons explorer ces différents domaines:

1.5.1 Domaine industriel: sont des robots utilisés dans un environnement de fabrication industrielle. Ils sont utilisés dans la fabrication des automobiles, des composants et des pièces électroniques, des médicaments et de nombreux produits [5].



Figure I.6: Les robots industriels

1.5.2 Domaine spatial: Les manipulateurs mobiles spatiaux sont destinés à explorer des environnements où l'homme ne peut pas se rendre. C'est-à-dire des environnements souvent mortels pour l'homme.



Figure I.6: Robot Rover Martien

I.5.3Domaine agriculture: Après des décennies d'expérimentation et de tâtonnements, les robots ont enfin fait leur entrée à la ferme. Cette machine totalement autonome fonctionne grâce à l'énergie solaire et circule dans les rangées de plantations pour surveiller et analyser les plants

Ce robot a déjà passé avec succès de nombreux tests réalisés dans des champs de légumes mais se contente de surveiller la « bonne santé » et plantations. Grâce à ces nombreux capteurs, senseurs et caméras, il détecte rapidement d'éventuelles anomalies (présence de mauvaises herbes, animaux nuisibles, croissance trop faible) et avertit l'exploitant agricole qui peut ainsi prendre immédiatement les mesures appropriées.



Figure I.7: Robot utilisé en agricole

I.5.4Robots domestiques ou ménagers : Robots utilisés à la maison. Ce type de robots comprend de nombreux appareils très différents, tels que les aspirateurs robotiques, robots nettoyeurs de piscines, balayeuses, nettoyeurs gouttières et autres robots qui peuvent faire différentes tâches. En outre, certains robots de surveillance et de télé présence pouvaient être considérées comme des robots ménagers se il est utilisé dans cet environnement[5]



Figure I.7: Robots domestiques ou ménagers

I.5.5 Domaine militaire: Les robots sont de plus en plus utilisés dans le domaine militaire. En effet, la miniaturisation permet aujourd'hui de créer des robots discrets mais dotés de nombreux capteurs, ce qui est idéal pour des missions d'espionnage ou d'éclairage[5]



Figure I.8: Robot militaire

I.5.6 Domaine médical : Les robots semblent avoir de l'avenir à l'hôpital. Robodoc aide à réaliser certaines opérations de chirurgie. Le robot infirmier est encore en projet. Le cyber squelette HAL aide les personnes à se déplacer. Et le robot patient permet aux futurs chirurgiens dentistes d'apprendre à soigner sans faire de dégâts...[5]



Figure I.9: Robot utilisé dans le domaine militaire

I.6 Caractéristiques d'un robot:

Un robot doit être choisi en fonction de l'application qu'on lui réserve. Voici quelques paramètres à prendre en compte [11] :

- La charge maximale transportable (de quelques kilos à quelques tonnes), à déterminer dans les conditions les plus défavorables .
- L'architecture du système mécanique articulé, le choix est guidé par la tâche à réaliser .
- Le volume de travail, définit comme l'ensemble des points atteignables par OT. Tous

les mouvements ne sont pas possibles en tout point du volume de travail .

- Le type du robot et la mission à effectuer .
- La vitesse de déplacement .
- La masse du robot .
- Le coût du robot.

I.7 Classifications des robots:

I.7.1 Par génération:

A l'heure actuelle, on peut distinguer trois générations de robots :

I.7.1.a Le robot passif :

Il est capable d'exécuter une tâche qui peut être complexe, mais de manière répétitive, il ne doit pas y avoir de modifications intempestives de l'environnement. L'auto-adaptativité est très faible, de nombreux robots sont encore de cette génération.

I.7.1.b Le robot actif:

Il devient capable d'avoir une image de son environnement, et donc de choisir le bon comportement (sachant que les différentes configurations ont été prévues). Le robot peut se calibrer tout seul.

I.7.1.c Le robot intelligent :

Le robot est capable d'établir des stratégies, ce qui fait appel à des capteurs sophistiqués, et souvent à l'intelligence artificielle.

I.7.2 Par type:

I.7.2.a Les manipulateurs:

Ce sont des robots industriels destinés à effectuer des tâches bien déterminées et de manière répétitive. A noter que pour ce type de robot :

- Les trajectoires sont bien déterminées dans l'espace ;
- Les positions sont discrètes avec 2 ou 3 valeurs par axe ;
- La commande est séquentielle.

La figure suivante illustre les différents éléments qui constituent un robot manipulateur

:

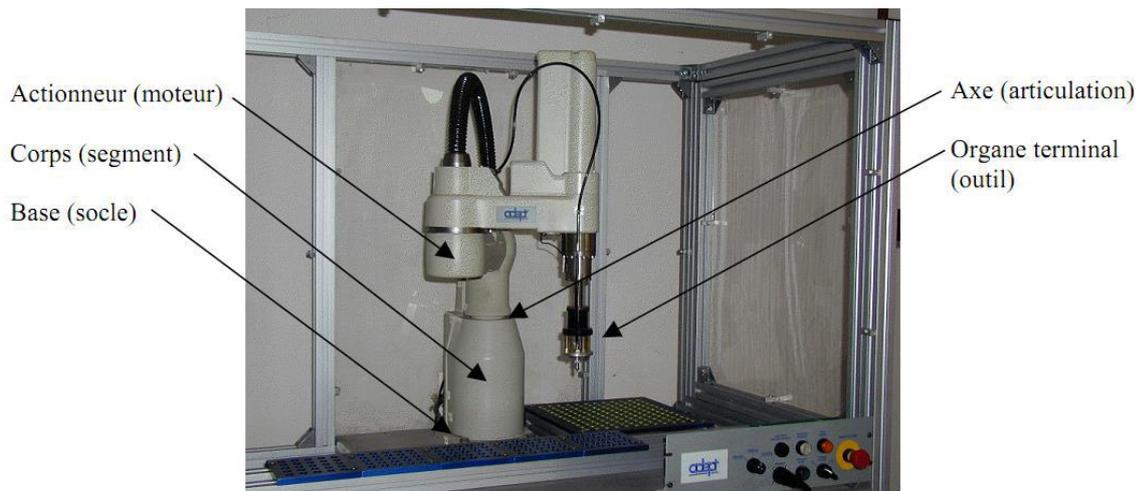


Figure 1.10: Constitution d'un robot manipulateur

1.7.2.b Télémanipulateurs :

Appareils de manipulation à distance (pelle mécanique, pont roulant, etc.), apparus vers 1945 aux USA. Dans ce cas :

- Les trajectoires peuvent être quelconques dans l'espace ;
- Les trajectoires sont définies de manière instantanée par l'opérateur.



Figure 1.11: télémanipulateur- A1000[6]

1.7.2.c Robots mobiles:

1.7.2.c.1 Définition : En général, on peut définir un robot mobile comme étant une machine équipée de capacités de perception, de décision et d'action qui lui permettent d'agir de manière autonome dans son environnement en fonction de la perception qu'il en a.[7]

1.7.2.c.2 Présentation générale des robots mobiles :

Contrairement au robot industriel qui est généralement fixé, le robot mobile est doté de moyens qui lui permettent de se déplacer dans son espace de travail. Suivant son degré d'autonomie ou degré d'intelligence, il peut être doté de moyens de perception et de raisonnement. Certains sont capables, sous contrôle humain réduit, de modéliser leur

espace de travail et de planifier un chemin dans un environnement qu'ils ne connaissent pas forcément d'avance.

Actuellement, les robots mobiles les plus sophistiqués sont essentiellement orientés vers des applications dans des environnements variables ou incertains, souvent peuplés d'obstacles, nécessitant une adaptabilité à la tâche.

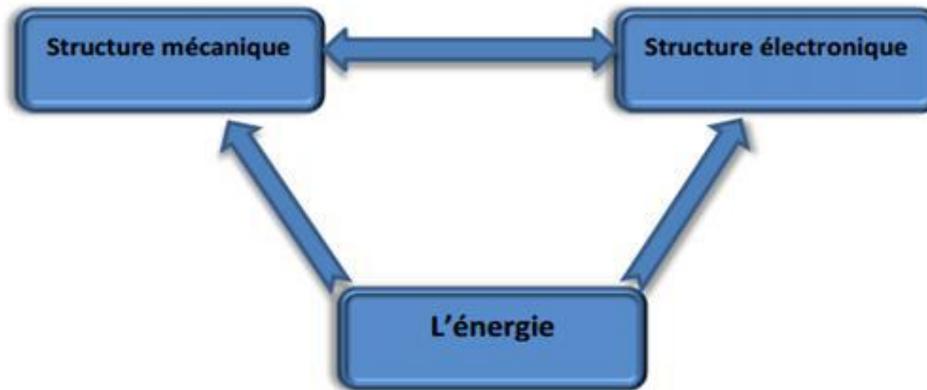


Figure I.12: Structure d'un robot mobile

Les robots mobiles ont une place particulière en robotique. 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 tâches pénibles (transport de charges lourdes), monotones ou en ambiance hostile (nucléaire, marine, spatiale, lutte contre l'incendie, surveillance...).[8]

1.7.2.c.3Classification: [9]

Une classification est proposée dans la littérature qui définit le degré d'autonomie du robot mobile.

- Véhicule télécommandé par un opérateur qui lui impose chaque tâche élémentaire à réaliser.
- Véhicule télécommandé au sens de la tâche à réaliser. Le véhicule contrôle automatiquement ses actions.
- Véhicule semi-autonome réalisant sans l'aide de l'opérateur des tâches prédéfinies
- Véhicule autonome qui réalise des tâches semi-définies. Ce type de véhicule pose des problèmes d'un niveau de complexité élevé de représentation des connaissances, de capacité décisionnelle et de génération de plans qui sont résolus à bord dans la mesure du possible.

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

1. La conception mécanique liée à la mobilité ;
2. La détermination de la position et de l'orientation ;
3. La détermination du chemin optimal.



Figure I.13: Véhicule autonome

1.7.2.c.4 Architecture d'un robot mobile:

Un robot mobile se compose de trois grands modules fonctionnels : la locomotion, la perception et la décision.[10]

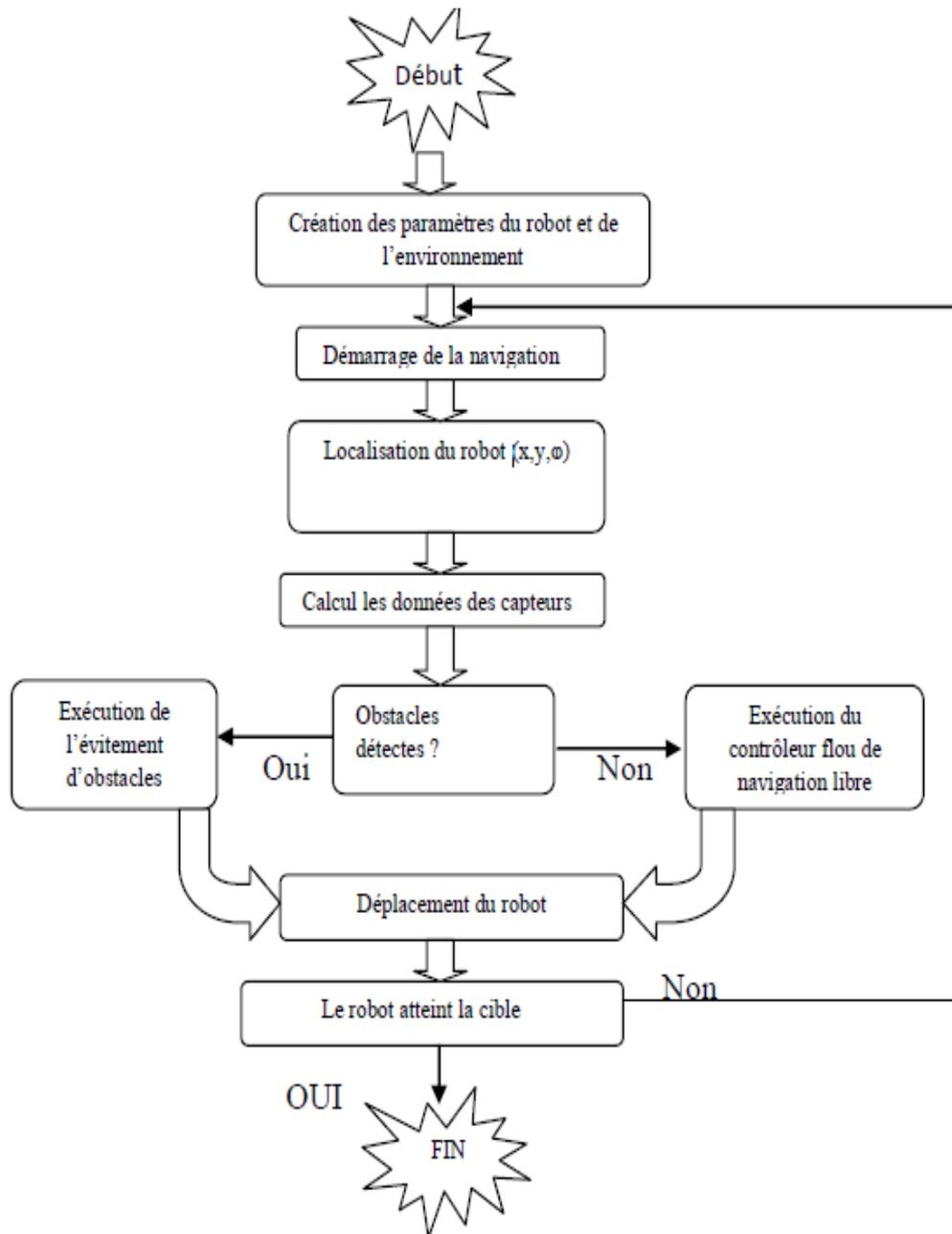


Figure I.14: L'architecture fonctionnelle du robot mobile

I.7.2.c.5 La structure mécanique et motricité:

❖ La structure mécanique:

On peut dénombrer quatre types de structures mécaniques assurant la motricité.[11]

- **Les mobiles à roues :** La mobilité par roues est la structure mécanique la plus communément appliquée. Cette technique assure selon l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importante. Le

franchissement d'obstacles ou l'escalade de marches d'escalier est possible dans une certaine mesure. Toutes les configurations (nombre, agencement, fonction) des roues sont appliquées.



Figure I.15: Robot à roue

- **Les 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 I.16: Robot à chenille

- **Les 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 autonomie 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.



Figure I.17 : Robot marcheur

- **Les robots rampants :** La reptation est une solution de locomotion pour un environnement de type "tunnel" qui conduit à réaliser des structures filiformes. Le système est composé d'un ensemble de modules ayant chacun plusieurs mobilités. Les techniques utilisées découlent des méthodes de locomotion des animaux. Le type scolopendre constitue une structure inextensible articulée selon deux axes orthogonaux. Le type lombric comprend trois articulations, deux rotations orthogonales et une translation dans le sens du mouvement principal. Le type péristaltique consiste à réaliser un déplacement relatif d'un module par rapport aux voisins.

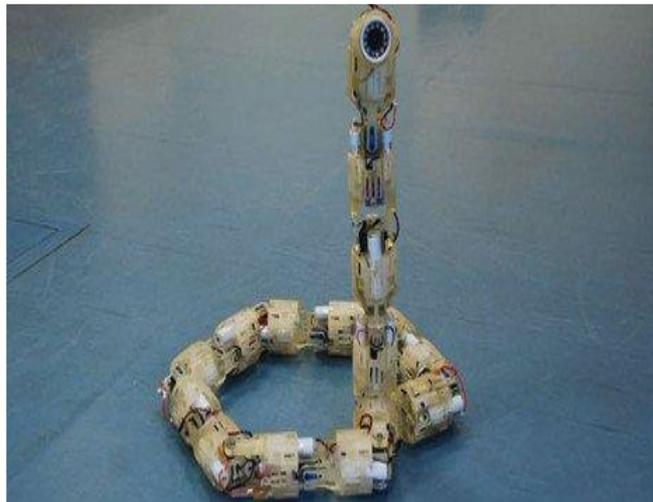


Figure I.18:robot mobile rampant

❖ La motricité et l'énergie:

Le déplacement des robots est réalisé par des moteurs de types: électrique, thermique et hydraulique. L'énergie électrique la plus fréquemment employée offre l'avantage d'une commande aisée. Par contre le transport et la généralisation 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.

1.7.2.c.6 Traitement des informations et gestion des tâches:

L'ensemble de traitement des informations et gestion des tâches constitue le noyau du module d'informatique central qui établit les commandes permettant au robot mobile de réaliser un déplacement et d'activer les divers organes en accord avec l'objectif. A ce niveau, le problème qui se pose est le problème de génération de plan qui consiste à établir la manière dont le robot se déplace par rapport à des connaissances aprioriques «Statiques» ou obtenues en cours d'évolution « dynamiques ». La génération des plans repose sur trois concepts :

1. La stratégie de navigation .
2. La modélisation de l'espace .
3. La planification.

1.7.2.c.7 Les avantages dans l'utilisation des robots mobiles

Les divers avantages des robots mobiles se résument ainsi :

- Accroissement de la capacité de production
- Remplacement de l'homme dans l'exécution des tâches pénibles ou dangereuses
- Manutentions.

1.8 Conclusion:

A travers ce chapitre on peut dire que dans le cadre de la robotique, la robotique mobile joue un rôle à part. Contrairement aux robots industriels manipulateurs qui travaillent de façon autonome dans un grand nombre d'usines automatisées, les robots mobiles sont très peu répandus. Cette situation n'est pas due au manque d'applications possibles, mais dès qu'on dispose de la mobilité, on peut imaginer des robots facteurs, nettoyeurs, gardiens, démineurs, explorateurs, jardiniers et beaucoup d'autres. La faible diffusion est surtout due au fait que ces tâches ont une complexité bien supérieure à celles effectuées par des robots manipulateurs industriels. Le monde dans lequel un robot mobile doit se déplacer est souvent très vaste, partiellement ou totalement inconnu, difficilement caractérisable géométriquement et ayant une dynamique propre.

chapitre 02

II.1 Introduction

La modélisation mathématique est une étape très importante pour la commande des robots. Deux types de modèles sont généralement utilisés lors de la commande, à savoir : le modèle cinématique et le modèle dynamique. D'après la littérature, on rencontre plusieurs types de robots à savoir : les robots de type unicycle, les robots de type tricycle, et les robots de type voiture. Dans le cadre de notre travail, nous avons fixé les roues d'un manière similaire au unicycle .

Les robots mobiles non holonomes ont une structure et un mouvement très similaire aux véhicules de notre vie courante : voiture particulière, les bus, les camions...

En générale leur structure mécanique est simple. Ils ont des roues motrices, des roues directrices et des roues porteuses, même on peut trouver des roues avec les deux ou trois fonctions combinées.

Un robot non-holonyme ne dispose que de deux degrés de libertés sur un plan :

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

Il est difficile de faire revenir un robot non-holonyme après son démarrage à sa position initiale sans manœuvres.

Ce chapitre consiste à représenter le comportement du robot par des équations algébriques, nous faisons appel à des notions mathématiques de point de vue position (*Modèle géométrique*) ou de point de vue vitesse (*modèle cinématique*) ou de point de vue des efforts misent en jeu (*modèle dynamique*). Les coordonnées articulaires et opérationnelles sont les paramètres qui caractérisent le modèle du robot. Les premières permettent au mécanisme de modifier sa géométrie et les secondes déterminent la position et l'orientation de l'organe terminal.

II.2 Définition :

Le robot Unicycle est un véhicule non holonyme à conduite différentielles (Différentiel Drive WMR). Il possède 2 roues motrices indépendantes non orientables sur le même axe et une roue décentrée orientable libre « roue folle » qui assure la stabilité du véhicule.

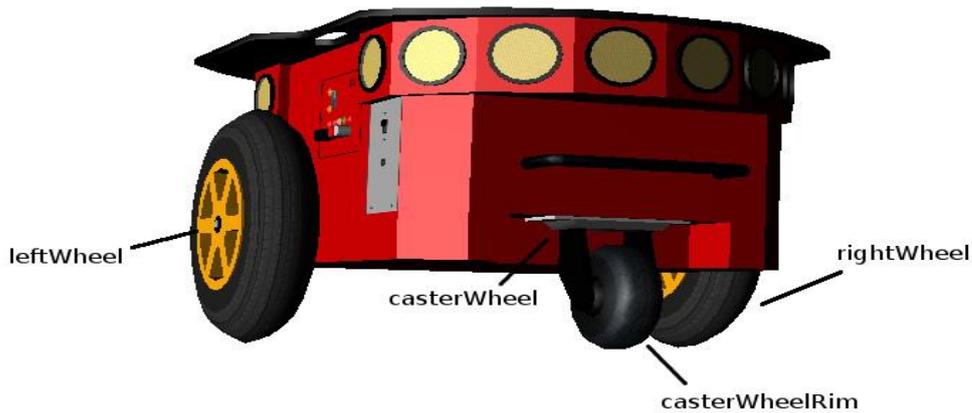


Figure II.1 : Véhicule unicycle(Robot Pioneer-3DX)

II.4 Modélisation du robot unicycle [12]

II.3.1 Modélisation géométrique

La modélisation géométrique permet de situer les différentes parties mobiles d'un robot les unes par rapport aux autres.

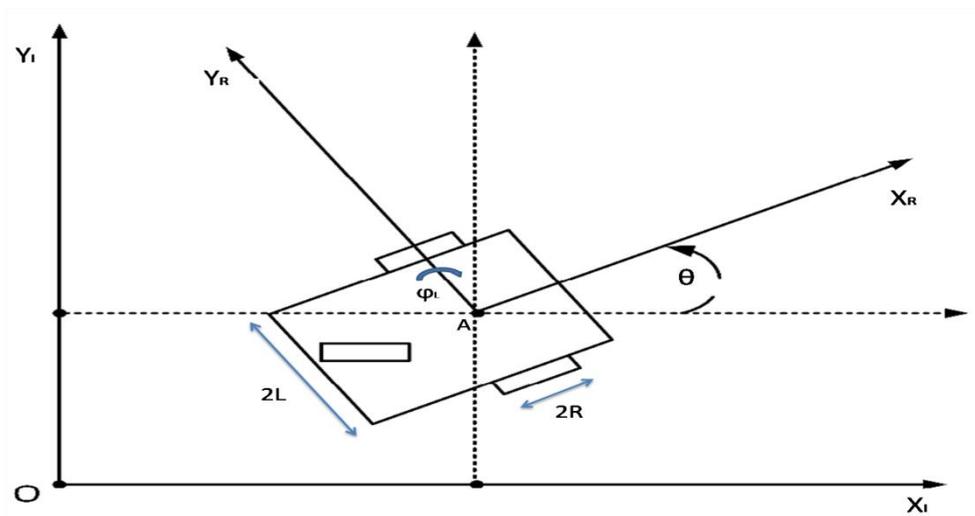


Figure II.2 : Paramétrage du Robot mobile à entraînement différentiel

A : est le point milieu de l'axe des roues.

$2R$: représente le diamètre de la roue du robot

$2L$: représente la largeur du robot

$\dot{\phi}_R$ et $\dot{\phi}_L$: représentent respectivement la vitesse de rotation de la roue droite et de la roue gauche

θ : est l'angle d'orientation du robot

- Notons $\{x_i, y_i\}$ un repère fixe quelconque et $\{x_r, y_r\}$ un repère mobile lié au

robot. Soient $q^i = \begin{bmatrix} x^i \\ y^i \\ \theta^i \end{bmatrix}$ un point du repère $\{x_i, y_i\}$ et $q^r = \begin{bmatrix} x^r \\ y^r \\ \theta^r \end{bmatrix}$ point du repère $\{x_r, y_r\}$.

Les points q^i et q^r sont liés par la matrice orthogonale $R(\theta)$.

$$q^i = R(\theta) q^r \quad \text{avec } R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{bmatrix} x^i \\ y^i \\ \theta^i \end{bmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} x^r \\ y^r \\ \theta^r \end{bmatrix} \quad (II.1)$$

Hypothèse I :

Aucun glissement latéral. Cela signifie simplement que le robot ne peut pas se déplacer latéralement dans le repère local, ce qui se traduit mathématiquement par l'équation :

$$\dot{y}_A^r = 0 \quad (II.2)$$

En utilisant la matrice de rotation $R(\theta)$, l'expression de la vitesse du robot associée au point A dans le repère fixe est :

$$\begin{bmatrix} \dot{x}_A^i \\ \dot{y}_A^i \\ \dot{\theta}_A^i \end{bmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} \dot{x}_A^r \\ 0 \\ \dot{\theta}_A^r \end{bmatrix} \rightarrow \begin{cases} \dot{x}_A^i = \dot{x}_A^r \cos\theta \\ \dot{y}_A^i = \dot{x}_A^r \sin\theta \\ \dot{\theta}_A^i = \dot{\theta}_A^r \end{cases} \quad (II.3)$$

Ainsi on obtient :

$$-\dot{x}_A^i \sin\theta + \dot{y}_A^i \cos\theta = 0 \quad (II.4)$$

Hypothèse II :

La contrainte de roulement pure représente le fait que chaque roue maintient un point de contact P avec le sol tel que représenté sur la figure II.3 Il n'y a pas de glissement de la roue dans son axe longitudinal (y^r) ni de dérapage dans son axe orthogonal (x^r). Les vitesses des points de contact dans le cadre du robot sont liées aux vitesses de roue :

$$v_R = R\dot{\phi}_R \quad v_L = R\dot{\phi}_L \quad (II.3)$$

avec:

v_R : la vitesse de la roue droite

v_L : la vitesse de la roue gauche

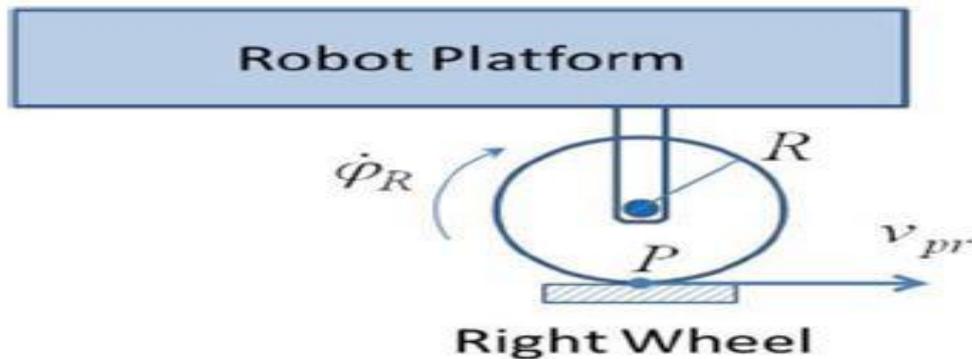


Figure II.3: Roulement sans glissement

II.4.2 Modélisation cinématique

La modélisation cinématique est l'étude du mouvement d'un system mécanique sans considération des forces qui influent sur son mouvement. À ce stade on s'intéresse qu'aux vecteurs de vitesse.

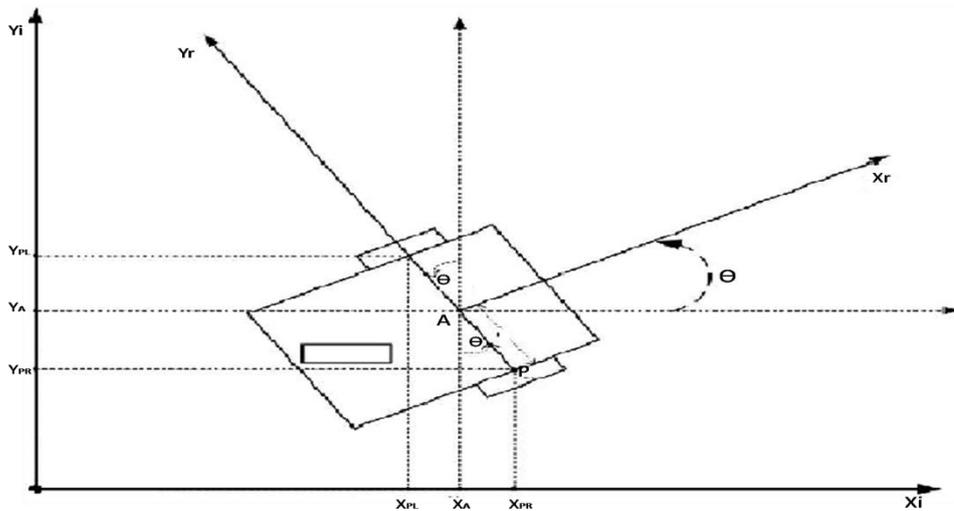


Figure II.4: Relation entre les coordonnées du point A et les point de contacts des roues.

Dans le repère monde, ces vitesses peuvent être calculées en fonction des vitesses du point centre A. Suivant la figure II.4 on a:

$$\text{Roue droit} \begin{cases} X_{PR}^i = X_A^i + L \sin \theta \\ Y_{PR}^i = Y_A^i - L \cos \theta \end{cases} \Rightarrow \begin{cases} \dot{x}_{PR}^i = \dot{x}_A^i + L \dot{\theta} \cos \theta \\ \dot{y}_{PR}^i = \dot{y}_A^i + L \dot{\theta} \sin \theta \end{cases} \quad (\text{II.4})$$

$$\text{Roue gauche} \begin{cases} X_{PL}^i = X_A^i - L \sin \theta \\ Y_{PL}^i = Y_A^i + L \cos \theta \end{cases} \Rightarrow \begin{cases} \dot{x}_{PL}^i = \dot{x}_A^i - L \dot{\theta} \cos \theta \\ \dot{y}_{PL}^i = \dot{y}_A^i - L \dot{\theta} \sin \theta \end{cases} \quad (\text{II.5})$$

En utilisant la matrice de rotation $R(\theta)$ et en l'appliquant à la roue droite on a :

$$\begin{pmatrix} \dot{x}_{PR}^i \\ \dot{y}_{PR}^i \\ \dot{\theta}^i \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{x}_{PR}^r \\ \dot{y}_{PR}^r \\ \dot{\theta}^r \end{pmatrix}$$

avec: $\dot{y}_{PR}^r = 0$ ainsi:

$$\begin{pmatrix} \dot{x}_{PR}^i \\ \dot{y}_{PR}^i \\ \dot{\theta}^i \end{pmatrix} = \begin{pmatrix} \dot{x}_{PR}^r \cdot \cos\theta \\ \dot{x}_{PR}^r \sin\theta \\ \dot{\theta}^r \end{pmatrix} \quad (II.6)$$

ainsi que:

$$\vec{V}_{PR}^r = \dot{x}_{PR}^r \vec{i}^r + \dot{y}_{PR}^r \vec{j}^r = \mathbf{R}\dot{\phi}_R \vec{i}^r \Rightarrow v_{PR}^r = \dot{x}_{PR}^r = \mathbf{R}\dot{\phi}_R \quad (II.7)$$

L'équation (II.6) dévient :

$$\begin{cases} \dot{x}_{PR}^i \cos\theta = \dot{x}_{PR}^r \cdot \cos^2\theta & (a) \\ \dot{y}_{PR}^i \sin\theta = \dot{x}_{PR}^r \sin^2\theta & (b) \end{cases}$$

En sommant (a) et (b), on obtient:

$$\dot{x}_{PR}^i \cos\theta + \dot{y}_{PR}^i \sin\theta = \mathbf{R}\dot{\phi}_R$$

En suivant les mêmes étapes pour la roue gauche on trouve la contrainte de roulement pour la roue gauche:

$$\dot{x}_{PL}^i \cos\theta + \dot{y}_{PL}^i \sin\theta = \mathbf{R}\dot{\phi}_L$$

On peut former le système d'équation des deux roues

$$\begin{cases} \dot{x}_{PR}^i \cos\theta + \dot{y}_{PR}^i \sin\theta = \mathbf{R}\dot{\phi}_R \\ \dot{x}_{PL}^i \cos\theta + \dot{y}_{PL}^i \sin\theta = \mathbf{R}\dot{\phi}_L \end{cases} \quad (II.8)$$

En introduisant les équations (II.4) et (II.5) dans (II.8) on obtient

$$\begin{cases} (\dot{x}_A^i + L\dot{\theta}\cos\theta)\cos\theta + (\dot{y}_A^i + L\dot{\theta}\sin\theta)\sin\theta = \mathbf{R}\dot{\phi}_R \\ (\dot{x}_A^i - L\dot{\theta}\cos\theta)\cos\theta + (\dot{y}_A^i - L\dot{\theta}\sin\theta)\sin\theta = \mathbf{R}\dot{\phi}_L \end{cases}$$

Les hypothèses 1 et 2 et les équations précédentes produisent les contraintes suivantes :

$$\begin{cases} -\dot{x}_A^i \sin\theta + \dot{y}_A^i \cos\theta = 0 \\ \dot{x}_A^i \cos\theta + \dot{y}_A^i \sin\theta + L\dot{\theta} = \mathbf{R}\dot{\phi}_R \\ \dot{x}_A^i \cos\theta + \dot{y}_A^i \sin\theta - L\dot{\theta} = \mathbf{R}\dot{\phi}_L \end{cases}$$

Dans le cas d'un roulement pur sans glissement on peut écrire les trois équations des contraintes sous la forme matricielle :

$$A(q)\dot{q} = 0 \quad (II.9)$$

$A(q)$ est la matrice de contraintes non-holonomes donnée par:

$$A(q) = \begin{pmatrix} -\sin\theta & \cos\theta & 0 & 0 & 0 \\ \cos\theta & \sin\theta & L & -R & 0 \\ \cos\theta & \sin\theta & -L & 0 & -R \end{pmatrix}$$

\dot{q} représente la dérivé de la coordonnée généralisée q , donnée par :

$$\dot{q} = [\dot{x}_A \quad \dot{y}_A \quad \dot{\theta} \quad \dot{\phi}_R \quad \dot{\phi}_L]^T$$

II.4.2.1 Le Modèle d'actionnement [13]:

La modélisation d'actionnement est l'étude du mouvement des systèmes mécaniques sans considérer les forces qui affectent le mouvement. Le but principal de la modélisation d'actionnement est de représenter les vitesses du robot en fonction des vitesses des roues motrices ainsi que des paramètres géométriques du robot.

A partir de ces deux hypothèses on peut alors déduire le modèle d'actionnement de notre robot :

$$\begin{cases} v = \frac{v_R+v_L}{2} = \frac{R(\dot{\phi}_R+\dot{\phi}_L)}{2} \\ \omega = \frac{v_R-v_L}{2L} = \frac{R(\dot{\phi}_R-\dot{\phi}_L)}{2L} \end{cases} \quad (II.10)$$

avec :

- v :la vitesse linéaire du robot
- ω :la vitesse angulaire du robot

A partir de ces équations nous pouvons construit notre modèle cinématique directe (MCD) sous Simulink/Matlab

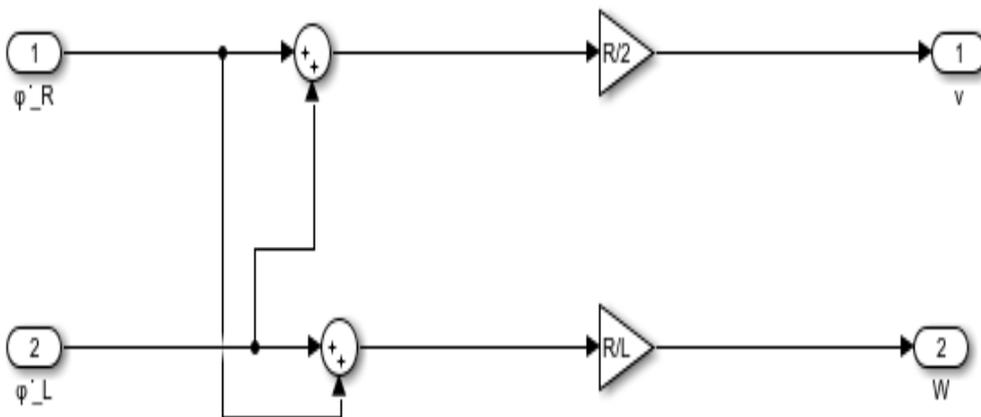


Figure II.4: Modèle cinématique directe

II.4.2.2 modèle cinématique en posture de véhicule[12]:

Dans le repère mobile les coordonnées du point A sont :

$$\begin{cases} \dot{x}_A^r = v = \frac{R(\dot{\phi}_R + \dot{\phi}_L)}{2} \\ \dot{y}_A^r = 0 \\ \dot{\theta}_A^r = \omega = \frac{R(\dot{\phi}_R - \dot{\phi}_L)}{2L} \end{cases} \quad (II.11)$$

En remplaçant l'équation (II.11) dans (II.3), on obtient:

$$\begin{bmatrix} \dot{x}_A^i \\ \dot{y}_A^i \\ \dot{\theta}_A^i \end{bmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} \frac{R(\dot{\phi}_R + \dot{\phi}_L)}{2} \\ 0 \\ \frac{R(\dot{\phi}_R - \dot{\phi}_L)}{2L} \end{bmatrix} \quad (II.12)$$

$$\begin{bmatrix} \dot{x}_A^i \\ \dot{y}_A^i \\ \dot{\theta}_A^i \end{bmatrix} = \begin{pmatrix} \frac{R}{2} \cos\theta & \frac{R}{2} \cos\theta \\ \frac{R}{2} \sin\theta & \frac{R}{2} \sin\theta \\ \frac{R}{2L} & -\frac{R}{2L} \end{pmatrix} \begin{pmatrix} \dot{\phi}_R \\ \dot{\phi}_L \end{pmatrix} \quad (II.13)$$

on obtient le modèle cinématique du robot mobile unicycle :

$$\dot{q}^i = \begin{bmatrix} \dot{x}_A^i \\ \dot{y}_A^i \\ \dot{\theta}_A^i \end{bmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$$\begin{aligned} \dot{x}_A^i &= v * \cos(\theta) \\ \dot{y}_A^i &= v * \sin(\theta) \\ \dot{\theta}_A^i &= \omega \end{aligned} \quad (II.14)$$

Nous avons modéliser ces équations sous Simulink qui nous permet d'avoir les vitesses \dot{x}_A^i, \dot{y}_A^i à partir de v et $\dot{\theta}_A^i$

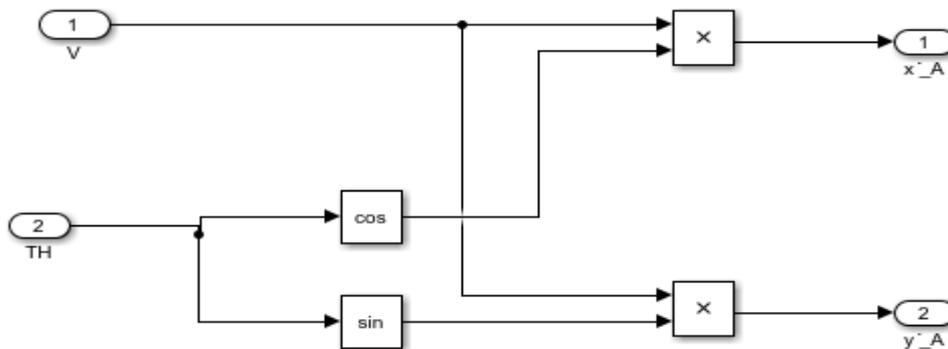


Figure II.5: Modèle cinématique en posture de véhicule

TH=θ

II.4.3 modèle cinématique inverse [13]:

Le modèle cinématique inverse permet quant à lui, de passer des vitesses opérationnelles V et θ aux vitesses de chaque roue. On admis les équations suivantes :

$$\dot{\phi}R = \frac{V+L\omega}{R} \quad (\text{II.15a})$$

$$\dot{\phi}L = \frac{V-L\omega}{R} \quad (\text{II.15b})$$

A partir de ces relations nous pouvons construit notre modèle cinématique inverse sous Simulink .

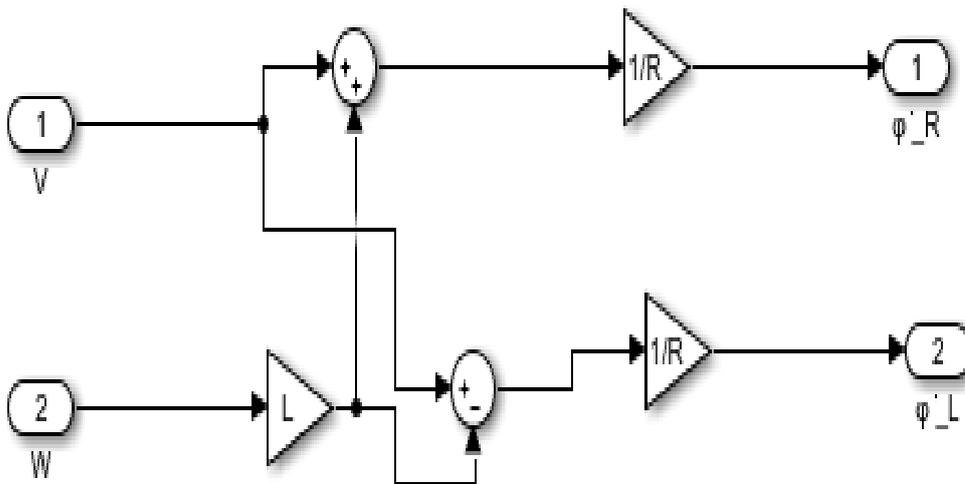


Figure II.6: _modèle cinématique inverse

II.4.4 cinématique du robot différentiel (differential drive)[14]

La cinématique est l'étude la plus fondamentale du comportement des systèmes mécaniques. Dans la robotique mobile, nous devons comprendre le comportement mécanique du robot à la fois pour concevoir des robots mobiles appropriés pour les tâches souhaitées et pour comprendre comment construire un logiciel de contrôle.

Considérez un robot d'entraînement différentiel à une position arbitraire $P(q_c)$ [x_c, y_c, θ_c] qui a une distance non nulle avec la position de but $R(q_d)$ [x_d, y_d, β] définie par rapport au cadre inertiel global.

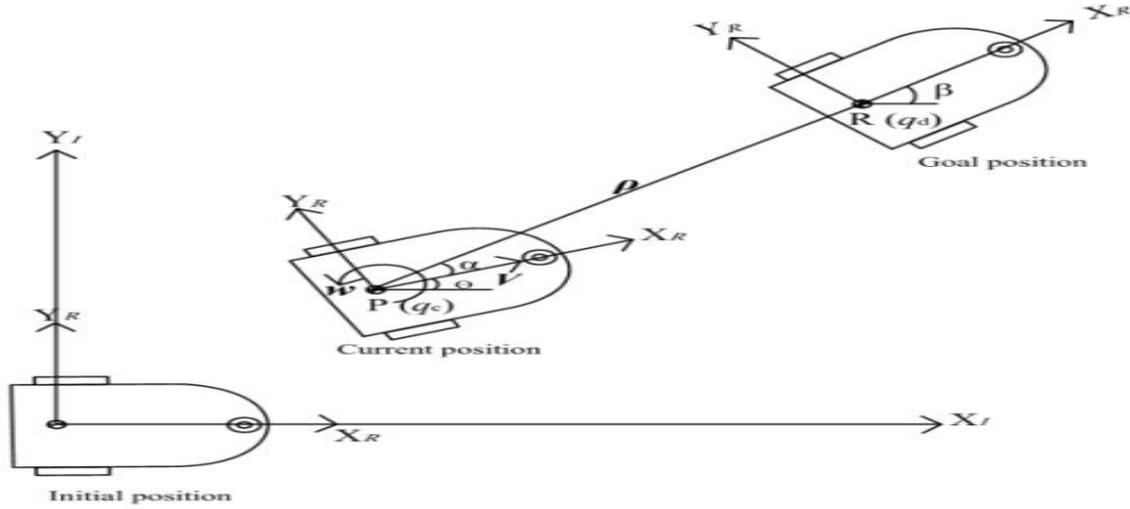


Figure II.6: Vecteurs d'erreur de la position et de l'orientation du robot

Le système de coordonnées du robot est régi par l'action combinée de la vitesse linéaire v et de la vitesse angulaire ω . En utilisant l'hypothèse géométrique, le robot cartésien est donné par :

$$\begin{aligned} \dot{x}_A^i &= v \cdot \cos(\theta) \\ \dot{y}_A^i &= v \cdot \sin(\theta) \\ \dot{\theta}_A^i &= \omega \end{aligned} \quad (\text{II.16})$$

Lorsque $v \cdot \cos(\theta)$ et $v \cdot \sin(\theta)$ sont les composants de v le long de ses axes X et Y et x, y et l'orientation θ sont mesurés par rapport au cadre inertiel de référence. La position du robot peut également être représentée en termes de coordonnées polaires impliquant une distance d'erreur $\rho > 0$ comme suit :

$$\dot{\rho} = -v \cos(\beta - \theta) = -v \cos(\alpha) \quad (\text{II.17})$$

$$\dot{\beta} = v \frac{\sin(\alpha)}{\rho} \quad (\text{II.18})$$

$$\dot{\theta} = \omega \quad (\text{II.19})$$

Maintenant, $\alpha = \beta - \theta$ est l'angle mesuré entre le cadre des axes du robot $[X_R, Y_R]$ et le cadre du vecteur de distance ρ également $\rho = \sqrt{(x_d - x_c)^2 + (y_d - y_c)^2}$ selon la formule de distance entre deux coordonnées système. Enfin, nous concluons

$$\dot{\rho} = -v \cos(\alpha) \quad (\text{II.20})$$

$$\dot{\alpha} = -\omega + \frac{v \sin(\alpha)}{\rho} \quad (\text{II.21})$$

$$\dot{\beta} = \frac{v \sin(\alpha)}{\rho} \quad (\text{II.22})$$

II.4.4.1 Localisation odométrique

La localisation est l'une des plus fondamentales aspects d'un robot mobile. Tout le système de robot mobile a pour répondre à la question fondamentale, qui est «Où suis-je», c'est-à-dire l'emplacement actuel et l'orientation du robot doit être obtenu pour que le robot puisse se déplacer facilement de la source à la destination. Il y a un certain nombre de techniques de localisation par rapport au robot mobile, cependant dans le présent travail nous avons utilisé des méthode de calcul pour la localisation. Dead Reckoning utilise l'odométrie pour mesurer le mouvement du robot. Dans le présent travail, nous obtenons les données d'un codeur incrémental (odométrie), qui est monté le long avec un moteur du robot mobile .Codeurs incrémentaux mesurer la rotation des roues, qui à leur tour, calcule la position et l'orientation du robot en utilisant

approches d'intégration du modèle cinématique sur $[t_k, t_{k+1}]$. En supposant que la configuration du robot $q_k[x_k \ y_k \ \theta_k]$ et les entrées à vitesse constante v_k et ω_k sont connu à un instant discret t_k , puis en utilisant l'intégration Euler

$$x_{k+1} = x_k + v_k T_s \cos(\theta_k) \tag{II.23}$$

$$y_{k+1} = y_k + v_k T_s \sin(\theta_k) \tag{II.24}$$

$$\theta_{k+1} = \theta_k + \omega_k T_s \tag{II.25}$$

où

$$v_k T_s = \Delta s \quad \text{et} \quad \omega_k T_s = \Delta \theta \tag{II.26}$$

$$T_s = t_{k+1} - t_k \tag{II.27}$$

La reconstruction du robot actuel la configuration est basée sur les données du codeur incrémental (odométrie). Soit $\Delta\phi_R$ et $\Delta\phi_L$ le non. de roue rotations mesurées pendant le temps d'échantillonnage T_s par le encodeurs. Déplacements linéaires et angulaires du robot est donné comme

$$\Delta s = \frac{R}{2} (\Delta\phi_R + \Delta\phi_L), \Delta \theta = \frac{R}{d} (\Delta\phi_R - \Delta\phi_L) \tag{II.28}$$

Où

R= rayon de la roue et d = axial distance entre les roues.

Pour un robot différentiel, la position peut être estimée à partir d'une position connue en intégrant le mouvement (additionnant les distances de déplacement incrémentales).

L'estimation de la configuration du robot au temps t_k est calculé comme:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \end{bmatrix} + \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta s \\ \Delta \theta \end{bmatrix} \quad (\text{II.29})$$

Localisation du robot à l'aide de l'odométrie ci-dessus la prédiction (communément appelée Dead Reckoning) est suffisamment précis en l'absence de patinage des roues et contrecoup.

II.4.5 Modélisation d'actionneur

Les moteurs à courant continue sont utilisés pour entraîner les roues du notre robot unicycle qui sont considérés comme étant les actionneurs. Les équations d'un moteur à courant continue sont :

$$Va(t) = R_a * i_a(t) + L_a \frac{di_a(t)}{dt} + e_a(t) \quad (\text{II.30})$$

$$e_a(t) = K_b * \omega_m(t) \quad (\text{II.31})$$

$$C_m(t) = K_t * i_a(t) \quad (\text{II.32})$$

$$J \frac{d\omega(t)}{dt} - f\omega(t) = C_m(t) - C_r(t) \quad (\text{II.33})$$

$$\omega_m = N * \omega_r \quad (\text{II.34})$$

V_a : la tension est utilisée comme entrée du moteur[V]

i_a : est le courant d'induit[A].

R_a : résistance des armatures du moteur[Ω]

L_a : l'inductance des armatures du moteur[H]

e_a : la force électromotrice[V]

ω_m : la vitesse angulaire du rotor[rad/s]

C_m : le couple du moteur[H]

C_r : le couple résistant [N.m]

K_t : constante de couple moteur[N.m/A]

K_b constante de la force électromotrice[V.s/red]

N: le rapport de réduction

J: l'inertie du moteur[Kg.m²]

f : coefficient de frottement[N.m.s]

Transformée de Laplace:

Pour construire le modèle causal du MCC , on applique d'abord la transformée de Laplace aux équations du MCC.

$$(1): Va(t) = R_a \cdot i_a(t) + L_a \frac{di_a(t)}{dt} + e_a(t) \quad (II.35)$$

La transformée de Laplace:

$$V=R_a \cdot I + S \cdot L_a \cdot I + E \quad (II.36)$$

$$V-E=(R_a + S \cdot L_a)I \quad (II.37)$$

donc:

$$(V-E) \frac{1}{R_a+S.L_a} = I \quad (II.38)$$

$$(2): J \frac{d\omega(t)}{dt} - f\omega(t) = C_m(t) - C_r(t)$$

La transformée de Laplace:

$$S.J.w - f.w=F(C_m)-F(C_r)$$

$$w(S.J- f)= F(C_m)-F(C_r)$$

$$(F(C_m)-F(C_r)) \frac{1}{S.J- f} = w$$

$$F(C_m)= K_t \cdot I \quad (II.39)$$

Les grandeurs et les transformées de Laplace associées sont résumées dans le tableau ci-dessous :

Grandeur	$Va(t)$	$i_a(t)$	$e_a(t)$	$\omega(t)$	$C_m(t)$	$C_r(t)$
Transformée de Laplace	V	I	E	w	F(C _m)	F(C _r)

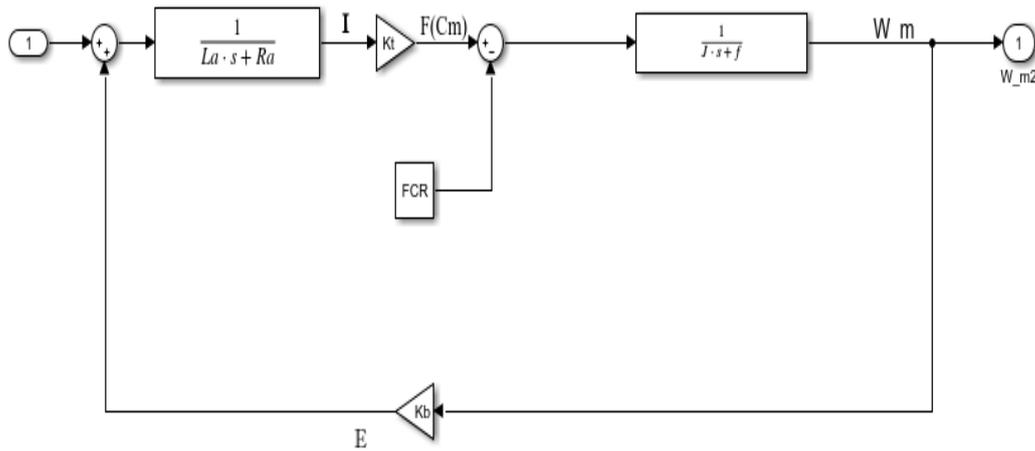


Figure II.7: Bloc moteur

II.4.6 Le modèle complet :

Maintenant nous pouvons déterminer le modèle complet de notre robot comme nous montre la Figure II.8:

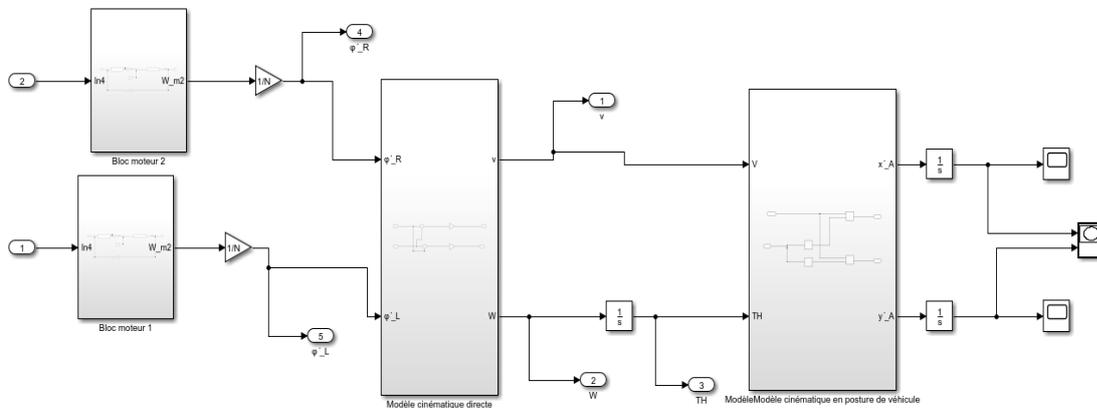


Figure II.8: Le modèle du robot unicycle

Donc il s'agit de deux blocs (bloc moteur1 et bloc moteur2) contenant chacun les deux moteurs du robot avec la boîte vitesse (réducteur), ces deux blocs nous permet d'avoir les vitesses angulaires appliqués à chaque roue qui seront par la suite les entrées de notre modèle cinématique direct. La vitesse linéaire (V) et angulaire (w) du robot unicycle seront les sorties du modèle cinématique direct, qui seront les entrées du modèle cinématique en posture de véhicule nous permet de déterminer la trajectoire du robot et sa position dans le plan (x, y). Tous ces blocs sont à l'intérieur d'un bloc (Subsystem) qui s'appelle « modèle Robot unicycle » dans laquelle on peut contrôler et simuler notre robot. La figure II.8 présente le schéma global de notre système.

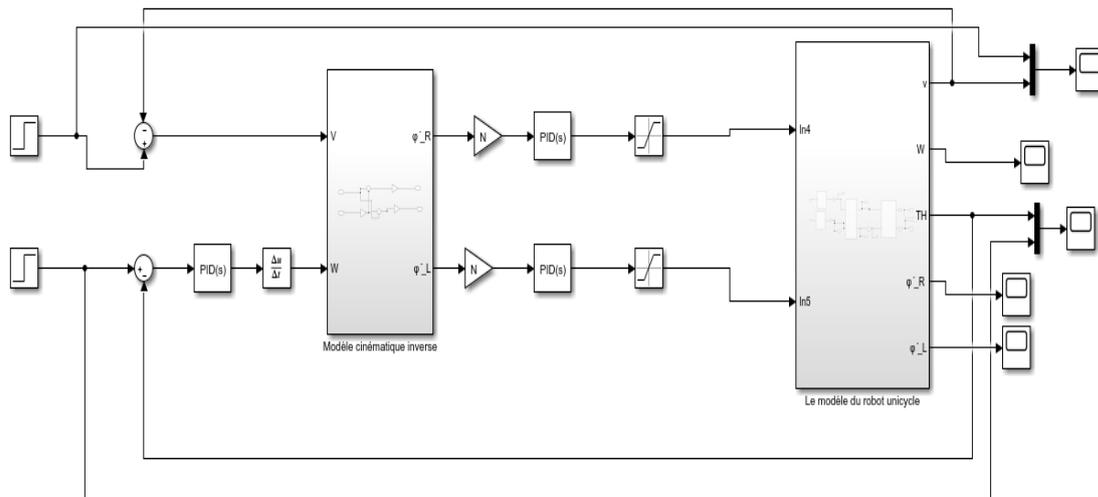


Figure II.9: Le modèle global du robot unicycle

Tout d'abord on commande notre robot unicycle en donnant une vitesse linéaire (V) et un Cap (θ) qui seront l'entrée du modèle cinématique inverse, ce dernier nous permet de passer des vitesses opérationnelles V et ω aux vitesses angulaires de chaque roue du robot. Le rapport de réduction N nous permet de convertir la vitesse angulaire de la roue à une vitesse du moteur qui est contrôlé par un correcteur PID et enfin une saturation de la tension pour protéger notre moteur.

II.4.6.1 Identification des paramètres

Afin de simuler notre modèle nous avons besoin de déterminer les paramètres du moteur, Pour cela, nous avons pris comme référence la documentation du moteur FIT0520.

paramètre	Valeur
R	0.065m
L	0.1m
J	$3 * 10^{-5} \text{ Kg.m}^2$
Kt	0.35 N.m/A
Kb	$5 * 10^{-5} \text{ V.s/red}$
La	0.5 H
Ra	4 Ω
N	38.3
f	$1 * 10^{-4} \text{ N.m.s}$
Fcr	0
V	4

Figure II.10: le tableau de paramètre de moteur

les paramètres PID étaient donc $P = 0$, $I = 1$ et $D=0$.

II.4.6.2 Résultat de simulation

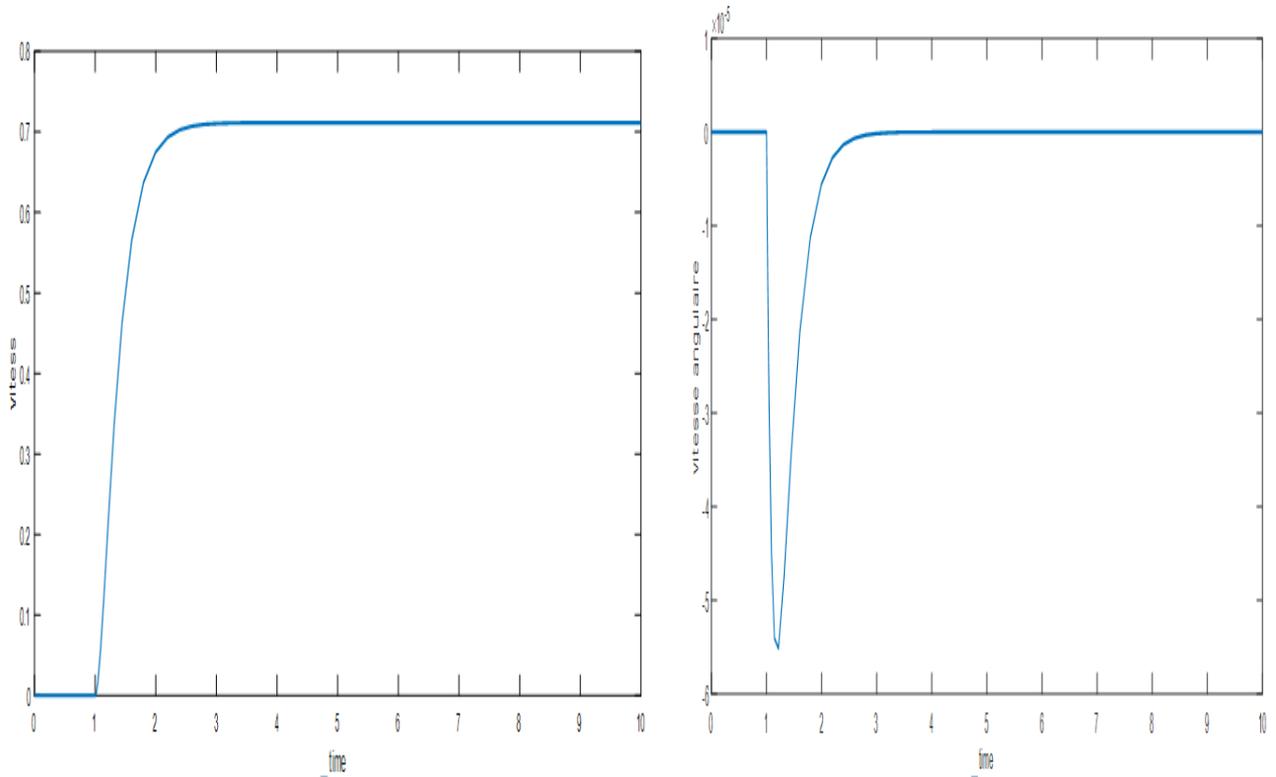


Figure II.11: vitesses linéaire V et la vitesse angulaire ω du robot

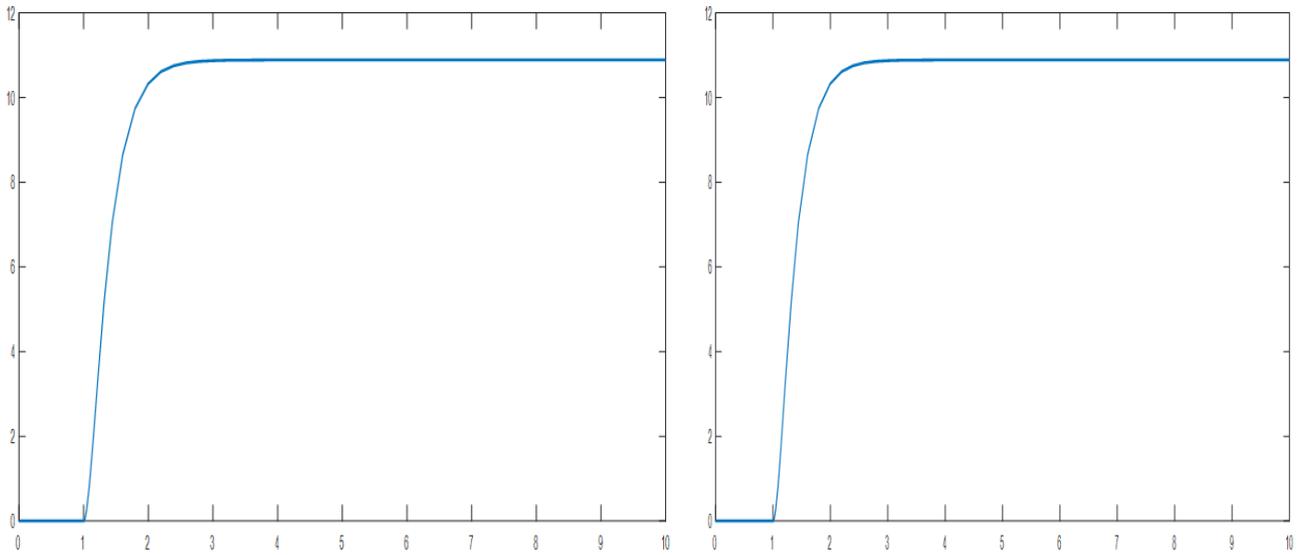


Figure II.12: vitesses angulaires des roues $\dot{\phi}_R$ et $\dot{\phi}_L$

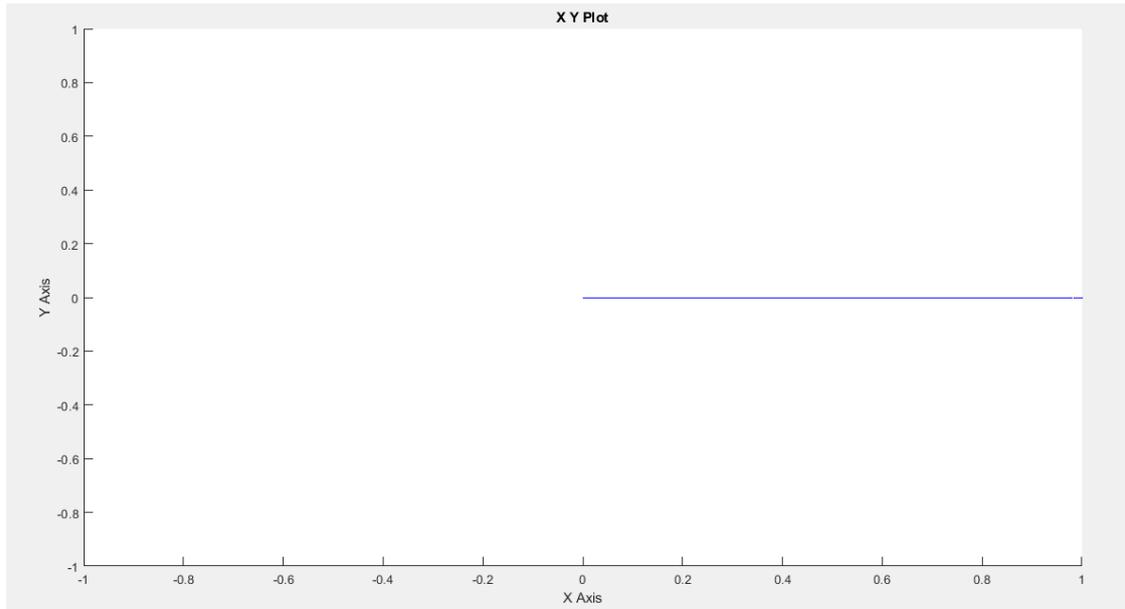


Figure II.13: trajectoire de robot

II.5 Modélisation dynamique[12]

La dynamique est l'étude du mouvement d'un système mécanique en tenant compte les différentes forces qui affectent son mouvement. Le modèle dynamique est essentiel pour la simulation et l'analyse du mouvement du robot, et pour la conception des variétés des algorithmes de commande.

Un robot non-holonyme avec un vecteur de coordonnées généralisées q , et soumis à des contraintes cinématiques peut être décrit par l'équation de mouvement suivante [12]

$$M(q)\ddot{q} + v(q, \dot{q})\dot{q} = B(q)u - A^T(q)\lambda \quad (\text{II.40})$$

Où $M(q)$ est une matrice d'inertie symétrique définie positive. $v(q, \dot{q})$ est la matrice des forces centrifuge et de Coriolis. $A^T(q)\lambda$ est la matrice des contraintes non-holonomes. λ le vecteur des multiplicateurs de Lagrange. $B(q)$ est la matrice de transformation d'entrée u : est le vecteur d'entrée, $u = \begin{bmatrix} \tau_R \\ \tau_L \end{bmatrix}$

Il y a deux méthodologies pour obtenir la dynamique d'un système :

- La méthode de Newton-Euler
- La méthode de Lagrange-Euler.

Dans ce travail on va voir la méthode de Lagrange-Euler

II.5.1 L'approche dynamique de Lagrange

Le formalisme d'Euler-Lagrange est décrit par l'équation suivant:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) + \frac{\partial L}{\partial q_i} = F - A^T(q)\lambda \quad (\text{II.41})$$

avec:

$$L(q, \dot{q}) = T - P$$

T: L'énergie cinétique du système

P: L'énergie potentielle du système

F :Le vecteur de force généralisée

q_i :la coordonnée généralisée et $q = [X_A, Y_A, \theta_A, \varphi_R, \varphi_L]^T$ de dimension $n = 5$.

L'énergie cinétique T du système est donnée par :

$$T = T_C + T_{\omega R} + T_{\omega L} \quad (\text{II.42})$$

Où

T_C est l'énergie cinétique de la plate forme

$$T_C = \frac{1}{2} m_c v_c^2 + \frac{1}{2} I_c \dot{\theta}^2 \quad (\text{II.43})$$

$T_{\omega R}$ est l'énergie cinétique de la roue droite

$$T_{\omega R} = \frac{1}{2} m_\omega V_{\omega R}^2 + \frac{1}{2} I_m \dot{\theta}^2 + \frac{1}{2} I_\omega \dot{\varphi}_R^2 \quad (\text{II.44})$$

$T_{\omega L}$ est l'énergie cinétique de la roue gauche

$$T_{\omega L} = \frac{1}{2} m_\omega V_{\omega L}^2 + \frac{1}{2} I_m \dot{\theta}^2 + \frac{1}{2} I_\omega \dot{\varphi}_L^2 \quad (\text{II.45})$$

m_c :masse de la plate-forme.

m_ω : masse de chaque roue plus la masse du moteur.

$V_{\omega R}$: vitesse linéaire de la roue droite.

$V_{\omega L}$: vitesse linéaire de la roue gauche.

I_m : Moment d'inertie de chaque roue avec le moteur par rapport au diamètre de la roue.

I_ω : Moment d'inertie de chaque roue avec le moteur par rapport à $\mathbf{1}'$ axe de la roue.

I_c : Moment d'inertie de la plate-forme du robot sans les roues, les moteurs, autour de l'axe vertical qui passe par le point C.

Le point C dans le repère fixe a pour coordonnées:

$$\begin{cases} X_c = x_a + d \cdot \cos(\theta) \\ Y_c = y_a + d \cdot \sin(\theta) \end{cases} \quad (\text{II.46})$$

$$\begin{cases} X_{WR} = x_a + L \cdot \sin(\theta) \\ Y_{WR} = y_a + L \cdot \cos(\theta) \end{cases} \quad (\text{II.47})$$

$$\begin{cases} X_{WL} = x_a - L \cdot \sin(\theta) \\ Y_{WL} = y_a + L \cdot \cos(\theta) \end{cases} \quad (\text{II.48})$$

L'énergie cinétique totale est donnée par :

$$T = \frac{1}{2}m(\dot{x}_a^2 + \dot{y}_a^2) - m_c \cdot d \cdot \dot{\theta}(\dot{y} \cdot \cos\theta - \dot{x} \cdot \sin\theta) + \frac{1}{2}I_w(\dot{\phi}_R^2 + \dot{\phi}_L^2) + \frac{1}{2} \cdot I \cdot \dot{\theta}^2 \quad (\text{II.49})$$

$$\text{Avec } \begin{cases} m = m_c + 2m_w \\ I = I_c + m_c \cdot d^2 + 2m_w L^2 + 2I_m \end{cases}$$

L'énergie potentielle étant nulle car le robot se déplace sur un plan horizontal. Alors on a donc le Lagrangien :

$$L=T$$

En se servant de l'équation (II.49), l'équation (II.41) devient:

$$\begin{cases} m \cdot \ddot{x}_A - m \cdot d\ddot{\theta}\sin\theta - m \cdot d \cdot \dot{\theta}^2 \cos\theta = C_1 & (\text{II.50a}) \\ m \cdot \ddot{y}_A - m \cdot d \cdot \ddot{\theta} \cdot \cos\theta - m \cdot d \cdot \dot{\theta}^2 \sin\theta = C_2 & (\text{II.50b}) \\ -m \cdot d \cdot \ddot{x}_A \sin\theta + m \cdot d \cdot \ddot{y}_A \cos\theta + I\ddot{\theta} = C_3 & (\text{II.50c}) \\ I_w \cdot \ddot{\phi}_R = \tau_R + C_4 & (\text{II.50d}) \\ I_w \cdot \ddot{\phi}_L = \tau_L + C_5 & (\text{II.50e}) \end{cases}$$

avec C1 , C2 , C3 , C4 et C5 sont les coefficients relatifs aux contraintes cinématiques

qui peuvent être exprimés en fonction du vecteur de multiplicateurs de Lagrange et de la matrice de contrainte cinématique $A^T(q)$, où

$$A^T(q)=[C_1 \ C_2 \ C_3 \ C_4 \ C_5]^T$$

Maintenant, les équations de mouvement obtenues (II.50a)- (II.50e) peuvent être représentées sous la forme générale donnée par l'équation (II.40)comme

$$M(q)\ddot{q} + v(q, \dot{q})\dot{q} = B(q)u - A^T(q)\lambda$$

Avec

$$M(q) = \begin{pmatrix} m & 0 & -m \cdot d \cdot \sin\theta & 0 & 0 \\ 0 & m & m \cdot d \cdot \cos\theta & 0 & 0 \\ -m \cdot d \cdot \sin\theta & m \cdot d \cdot \cos\theta & I & 0 & 0 \\ 0 & 0 & 0 & I_w & 0 \\ 0 & 0 & 0 & 0 & I_w \end{pmatrix}$$

$$v(q) = \begin{pmatrix} 0 & -m \cdot d \cdot \cos\theta & 0 & 0 & 0 \\ 0 & -m \cdot d \cdot \sin\theta & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad B(q) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$A^T(q)\lambda = \begin{pmatrix} -\sin\theta & \cos\theta & \cos\theta \\ \cos\theta & \sin\theta & \sin\theta \\ 0 & L & -L \\ 0 & -R & 0 \\ 0 & 0 & -R \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{pmatrix}$$

Il est possible d'obtenir l'expression du modèle dynamique du robot en éliminant le terme $A^T(q) \cdot \lambda$ qui correspond aux forces de contraintes liées aux contraintes cinématiques.

En définissant $\eta = \begin{pmatrix} \dot{\phi}_R \\ \dot{\phi}_L \end{pmatrix}$ comme étant le vecteur de vitesses auxiliaires, on peut écrire que (en utilisant II.12) :

$$\begin{pmatrix} \dot{x}_A^i \\ \dot{y}_A^i \\ \dot{\theta} \\ \dot{\phi}_R \\ \dot{\phi}_L \end{pmatrix} = \begin{pmatrix} R/2 \cos\theta & R/2 \cos\theta \\ R/2 \sin\theta & R/2 \sin\theta \\ \frac{R}{2L} & -\frac{R}{2L} \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{\phi}_R \\ \dot{\phi}_L \end{pmatrix} \Rightarrow \dot{q} = S(q)\eta \quad (II.51a)$$

et

$$\ddot{q} = \dot{s}(q)\eta + s(q)\dot{\eta} \quad (II.51b)$$

$S(q)$ est une matrice de rang complet qui satisfait à la condition suivante :
 $S^T(q)A^T(q)=0 \quad (II.52)$

En remplaçant (II.51) dans (II.40), on obtient :

$$M(q)[\dot{s}(q)\eta + s(q)\dot{\eta}] + v(q, \dot{q})[S(q)\eta] = B(q)u - A^T(q)\lambda \quad (II.53)$$

En multipliant l'équation (II.52) par $S^T(q)$ on a :

$$S^T(q) \cdot M(q) \cdot \dot{s}(q)\eta + S^T(q) \cdot M(q) \cdot s(q)\dot{\eta} + S^T(q)v(q, \dot{q}) \cdot S(q)\eta = S^T(q) \cdot B(q)u - S^T(q) \cdot A^T(q)\lambda$$

en posant:

$$\begin{cases} \bar{M}(q) = S^T(q) \cdot M(q) s(q) \\ \bar{V}(q, \dot{q}) = S^T(q) \cdot M(q) \dot{s}(q) + S^T(q) v(q, \dot{q}) \cdot S(q) \\ \bar{B}(q) = S^T(q) \cdot B(q) \end{cases} \quad (II.54)$$

L'équation (II.30) sous la forme réduite est

$$\bar{M}(q)\dot{\eta} + \bar{V}(q, \dot{q})\eta = \bar{B}(q)u \quad (II.55)$$

Avec

$$\bar{M}(q) = \begin{bmatrix} I_w + \frac{R^2}{4L^2}(mL^2 + I) & \frac{R^2}{4L^2}(mL^2 - I) \\ \frac{R^2}{4L^2}(mL^2 - I) & I_w + \frac{R^2}{4L^2}(mL^2 + I) \end{bmatrix}$$

$$\bar{V}(q, \dot{q}) = \begin{bmatrix} 0 & \frac{R^2}{2L^2}(m_c d \dot{\theta}) \\ -\frac{R^2}{2L^2}(m_c d \dot{\theta}) & 0 \end{bmatrix} \quad \bar{B}(q) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

En insérant (II.10) dans (II.32) on obtient:

$$\begin{bmatrix} (m + \frac{2I_w}{R^2}) & 0 \\ 0 & (I + \frac{2L^2}{R^2} I_w) \end{bmatrix} \begin{pmatrix} \dot{v} \\ \dot{\omega} \end{pmatrix} + \begin{bmatrix} 0 & -m_c d \omega \\ m_c d \omega & 0 \end{bmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{bmatrix} \frac{1}{R} & 0 \\ 0 & \frac{L}{R} \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (II.56)$$

$$u = \begin{cases} u_1 = \tau_R + \tau_L \\ u_2 = \tau_R - \tau_L \end{cases}$$

où

(τ_R, τ_L) est le couple d'entrée exprimé en Newton mètres (N.m).

L'équation (II.56) est la forme non linéaire du modèle dynamique du robot mobile soumis à notre étude.

L'expression linéarisée de l'équation (II.56) est

$$\begin{bmatrix} (m + \frac{2I_w}{R^2}) & 0 \\ 0 & (I + \frac{2L^2}{R^2} I_w) \end{bmatrix} \begin{pmatrix} \dot{v} \\ \dot{\omega} \end{pmatrix} = \begin{bmatrix} \frac{1}{R} & 0 \\ 0 & \frac{L}{R} \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (II.57)$$

où

m : est la masse totale du robot

I : est l'inertie totale équivalente du système

R : est le rayon de la roue

$2L$: est la largeur du robot.

En remplaçant $m + \frac{2I_w}{R^2}$ par m_0 et $(I + \frac{2L^2}{R^2} I_w)$ par I_0 , on obtient une expression simplifiée de l'équation (II.34) donnée par:

$$\begin{bmatrix} m_0 & \mathbf{0} \\ \mathbf{0} & I_0 \end{bmatrix} \begin{pmatrix} \dot{v} \\ \dot{\omega} \end{pmatrix} = \begin{bmatrix} \frac{1}{R} & 0 \\ 0 & \frac{L}{R} \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (II.58)$$

II.5.2 La simulation de modèle dynamique

Dans ce qui suit, nous présenterons les résultats de simulation obtenues après plusieurs tests.

En faisant varier les valeurs des paramètres P, I et D, afin de trouver les valeurs convenables pour assurer un système stable.

Le modèle dynamique directe

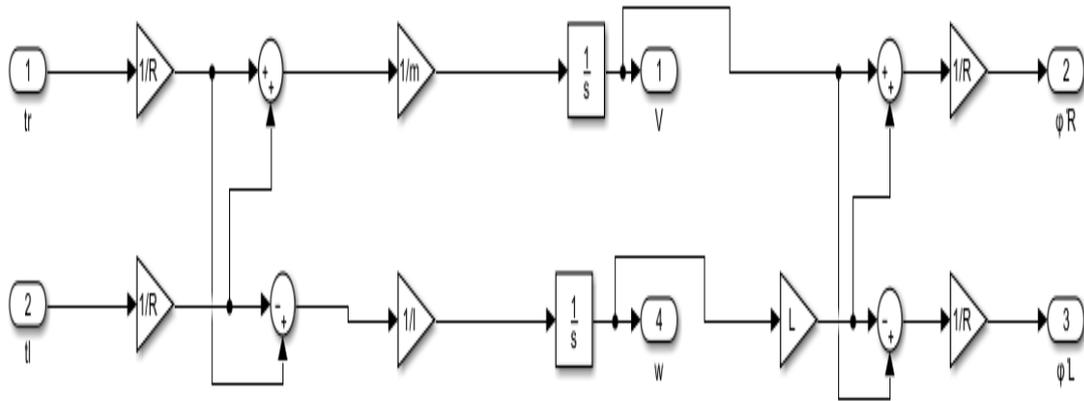


Figure II.14: le modèle dynamique directe

Le modèle global:

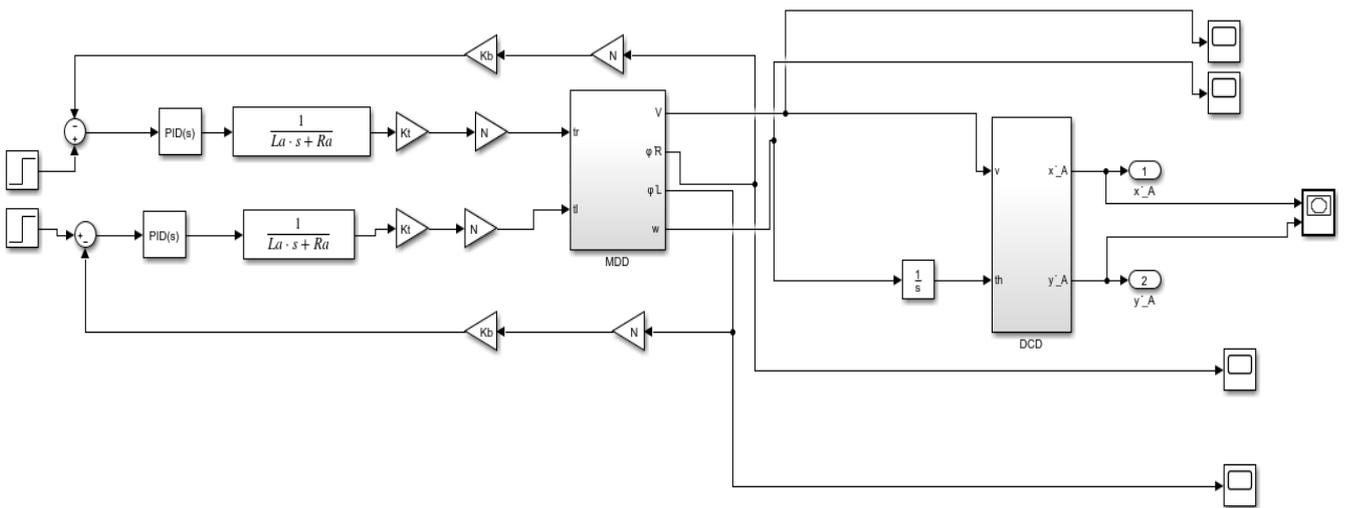


Figure II.15: le modèle global

II.5.3 Résultat de simulation

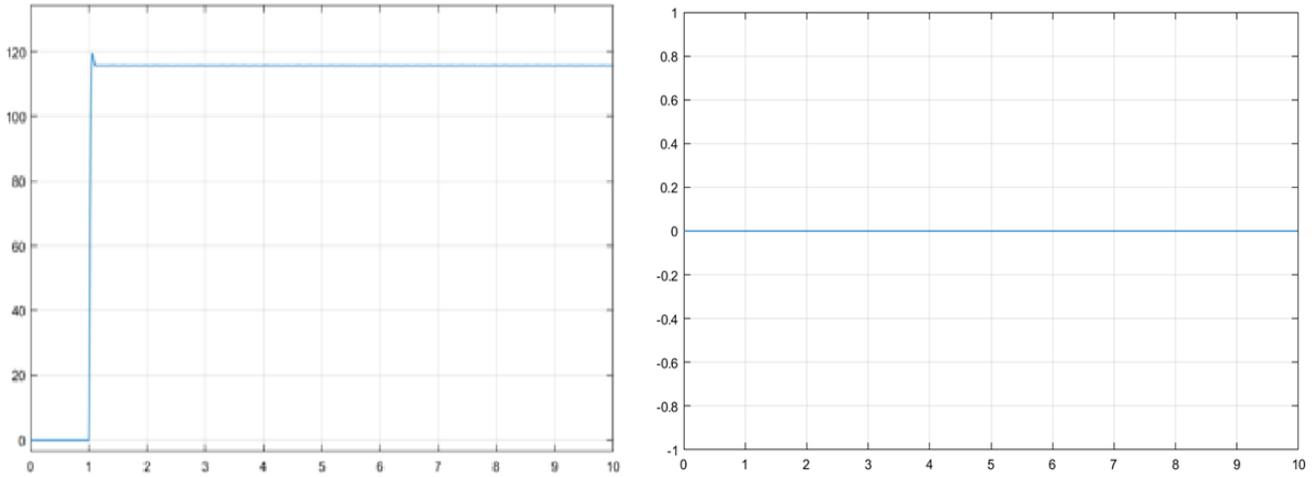


Figure II.15: la vitesse linéaire V et la vitesse angulaire ω du robot

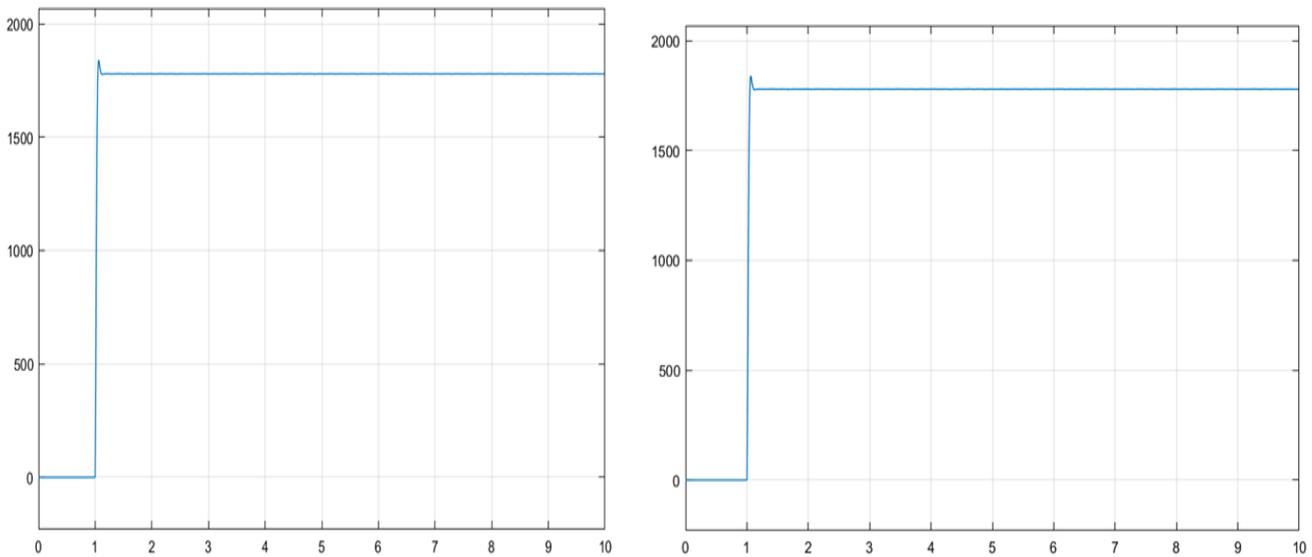


Figure II.16: la vitesse linéaire V et la vitesse angulaire $\dot{\phi}_L \dot{\phi}_R$

Conclusion

Dans ce chapitre, la classification générale, les constituants et la modélisation du robot mobile ont été présentés. Nous avons également développé des équations cinématiques et dynamiques du robot mobile unicycle. Le modèle dynamique, tout comme le modèle cinématique sont deux modèles essentiels sur lesquels est basée la commande. La modélisation mathématique de ce robot à roues différentielles tient compte des hypothèses 1 et 2 ci-dessus. Ce travail nous a permis de mettre en évidence la non-linéarité des systèmes réels.

chapitre 03

Introduction:

Un robot est un assemblage complexe de pièces mécaniques, électromécanique ou pièces électroniques. L'ensemble est piloté par une [unité centrale](#) appelée « [système embarqué](#) » : une simple séquence d'automatisme, un [logiciel informatique](#) ou une [intelligence artificielle](#) suivant le degré de complexité des tâches à accomplir. Lorsque les robots autonomes sont mobiles, ils possèdent également une [source d'énergie](#) embarquée : généralement une [batterie d'accumulateurs électriques](#) ou un [générateur électrique](#) couplé à un moteur à essence pour les plus énergivore.

Il se compose à trois parties essentielles :

- la partie mécanique.
- l'électronique de commande qui se compose a son tour de trois unités :
 - Unité de commande et de décision (microcontrôleur).
 - Unité de Perception de L'environnement (les capteurs).
 - Unité de mouvement des moteurs (circuit de puissance).
- La partie de programmation

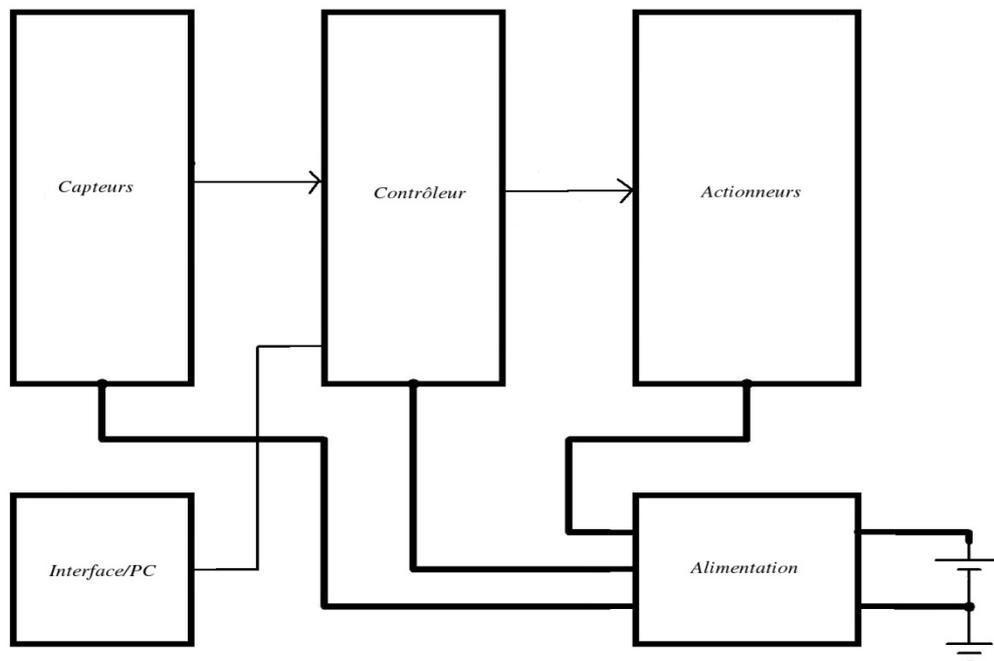


Figure III.1:Schéma des différentes unités du robot

L'objet de ce chapitre est de décrire tout d'abord les éléments principaux constitutifs de notre projet .la partie matérielle de la commande (le choix du matériel). Ensuite, nous aborderons les différents logiciels que nous utiliserons pour programmer la carte de commande

III.1 description matériel:

III.1.1 Arduino:

L'univers Arduino repose sur deux piliers, le premier s'agit de la carte électronique programmable (Hardware), composée de plusieurs composants semi-conducteurs, de circuits intégrés et des périphériques, le deuxième s'agit de l'interface de programmation (Software), qui possède un langage de programmation très spécifique, basé sur les langages C et C++, adapté aux possibilités de la carte. L'Arduino est une plateforme de contrôle, elle est constituée de deux choses:

- ❖ **Le logiciel (Software):** gratuit et open source, développé en Java, dont la simplicité d'utilisation relève du savoir cliquer sur la souris.
- ❖ **Partie Hardware (Le matériel):** cartes électroniques dont les schémas sont en libre circulation sur internet.

Le système Arduino permet à l'utilisateur de réaliser un grand nombre de projets puisque l'étendue de l'utilisation de l'Arduino est gigantesque. Voici quelques exemples:

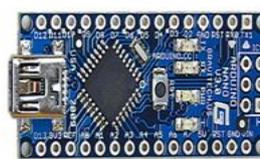
- Contrôler les appareils domestiques.
- Fabriquer son propre robot.
- Faire un jeu de lumières.
- Communiquer avec l'ordinateur.
- Télécommander un appareil mobile.



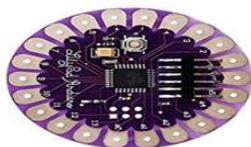
Arduino UNO



Arduino Méga



Arduino Nano



arduino LilyPad



arduino Leonardo



RedBoard

Figure III.2: Types de cartes Arduino

III.1.2 Raspberry Pi:

Raspberry Pi est un petit ordinateur sous le système d'exploitation Linux sur carte SD destine a des applications d'informatique embarquée. Le cœur de l'ordinateur est un FPGA (Broadcom 2835) intégrant un processeur ARM11 cadence a 700MHz et de nombreux périphériques.

Raspberry Pi peut être directement connecte a une IHM (Interface Homme Machine) classique, souris/clavier/écran HDMI ou vidéo composite, cependant comme tout ordinateur Linux.

Raspberry Pi peut intégrer ses propres outils de développement et une IHM reposant sur SSHcontrôlable depuis un autre ordinateur par Ethernet ou Wifi.

Le connecteur d'extension supporte les entrées/sorties parallèles ainsi que la plupart des busde communication. C'est un support particulièrement économique et puissant qui peut être facilement mis en œuvre dans de petits systèmes nécessitant un accès au monde physique par des capteurs/actionneurs disposants d'interfaces numériques



Figure III.3: Raspberry Pi

Les composants du Raspberry Pi

Vous serez peut-être tente de considérer le Raspberry Pi comme une carte a microcontrôleur

comme Arduino, ou comme un substitut a votre ordinateur portable. En réalité, la structure du

Raspberry Pi ressemble davantage a celle d'un appareil mobile, dote de plusieurs connecteurs

permettant d'accéder facilement a ses nombreux ports et fonctions. La (figure III.3)représente les différents composants de la carte.



Figure III.4: Emplacement des ports et interfaces matérielles du Raspberry Pi.

III.1.3 Moteurs à courant continu:

pour notre travail on utilise quatre moteurs à courant continu pour commander les roues . Évidemment, pour pouvoir valider un moteur, il faut connaître les spécifications que nous voulons atteindre.

On utilise les moteurs à CC, ils sont facile à utiliser car ils sont programmables depuis une très large gamme de langages de programmation (C#, Java,Python, Labview, Matlab...)

Ses spécifications :

- Tension : DC 3V-6V
- Courant : 100 MA-120MA
- RPM (avec pneu) : 100-240
- Diamètre des pneus : 65 mm
- Vitesse de la voiture (M/minute) : 20-48
- Poids du moteur (g) : 29/chacun
- Taille du moteur : 70mm X 22mm X 18mm
- Bruit : <65dB

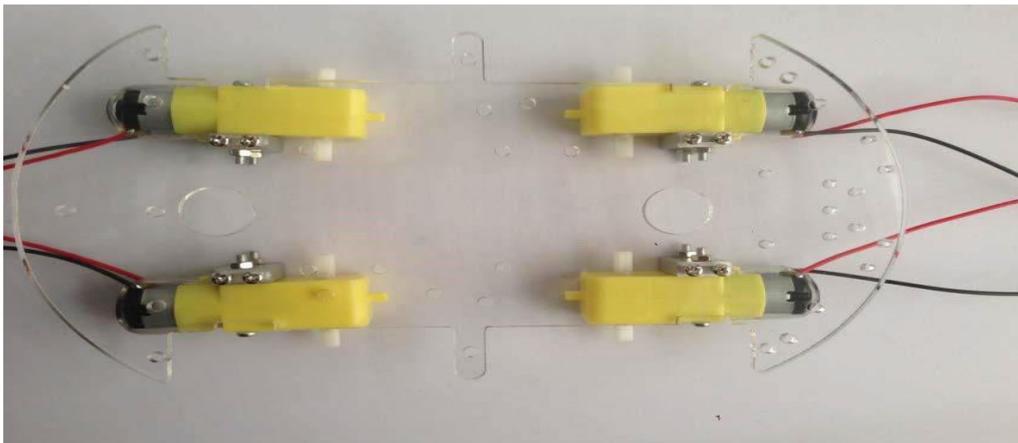


Figure III.5: les quatre moteur DC

III 1.4 Le Driver L298 :

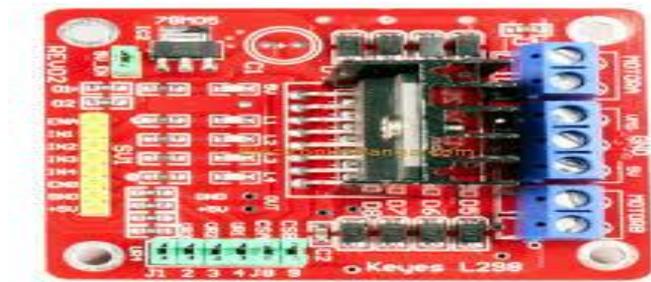


Figure III.6: Module a base de circuit L298

Description L298xx (H-Bridge Motor Driver):

Le Pont H, est une structure électronique servant à contrôler la polarité aux bornes d'un dipôle. Il est composé de quatre éléments de commutation généralement disposés schématiquement en une forme de H d'où il porte le nom. Les commutateurs peuvent être des relais, des transistors, ou autres éléments de commutation en fonction de l'application visée

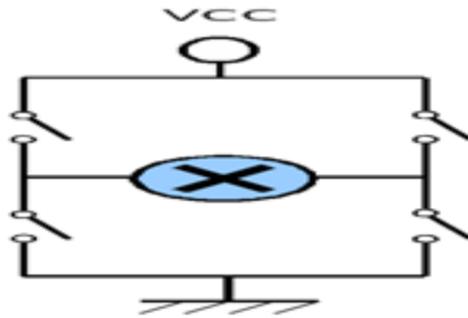


Figure III.7: Circuit d'un pont en H.

Principe de fonctionnement du pilote (Command du L298N):

Le L298N est un double pont en H, c'est à dire qu'il permet de faire tourner les moteurs dans un sens ou dans l'autre sans avoir à modifier les branchements, grâce à sa forme de H, d'où il tient son nom et qui lui permet de faire passer le courant soit dans un sens ou dans l'autre.

Les ports **ENA** et **ENB** permettent de gérer l'amplitude de la tension délivrée au moteur, grâce à un signal PWM. Les ports **In1**, **In2** pour le moteur A et **In3**, **In4** pour le moteur B, permettent de contrôler le pont en H et par conséquent le sens de rotation des moteurs. Comme expliqué sur le tableau suivant :

		Arrêt (moteur libre)	Sens +	Sens -	Arrêt (moteur freiné)		
Moteur A	Moteur B						
		In1	In3	LOW	HIGH	LOW	HIGH
		In2	In4	LOW	LOW	HIGH	HIGH

Tableau III.1:Principe de fonctionnement du pilote (Command du L298N).

III.1.5 Servomoteur :

Les servomoteurs sont très utilisés en robotique car ils sont à la base de nombreux mécanisme d'actionneurs, comme des pinces par exemple. Par contre, ils consomment souvent beaucoup de courant et demandent une tension souvent très bien régulée ce qui nécessite des dispositions particulières.



Figure III.8: le servomoteur SG-90

Il comprend :

- Un moteur électrique de petite taille.
- Un réducteur en sortie de ce moteur pour avoir moins de vitesse et plus de couple.

- Un capteur : un potentiomètre qui produit une tension variable en fonction de l'angle de l'axe de sortie.
- Un asservissement électronique pour contrôler la position de cet axe de sortie.

La commande du servomoteur :

Un connecteur à trois fils permet de commander un servomoteur. Bien sûr, il y a la masse et le +5V (à peu près). Ces deux fils sont en général noir et rouge et le 3ème fil est le fil de commande. Il est normalement blanc.

Pour commandé un servomoteur on parle de Pulse-Code Modulated Signal (Signal modulé en code d'impulsion).

En clair, il faut fournir au servomoteur une impulsion à 1 (suivie d'un retour à 0). Le servomoteur va prendre en compte la largeur temporelle de cette impulsion qu'il va convertir de façon linéaire en un angle. La durée de retour à 0 de l'impulsion n'est pas critique, en pratique il faut éviter de dépasser 20ms entre deux fronts montants.

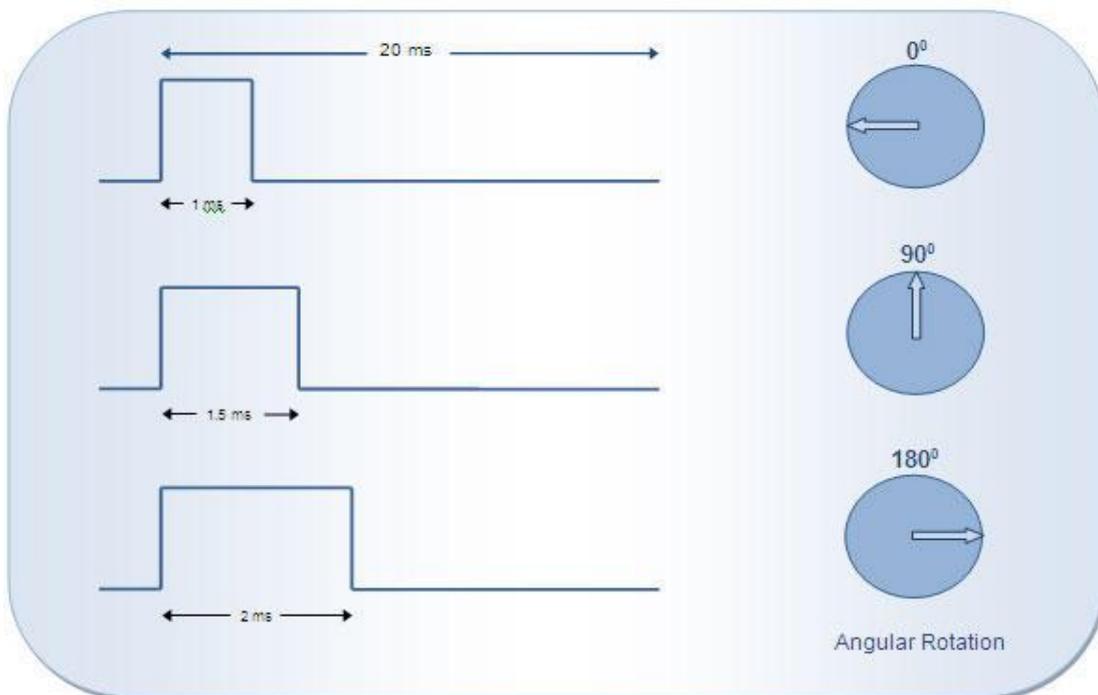


Figure III.9 : Signal modulé en code d'impulsion.

Caractéristique du servomoteur utilisé dans ce projet :

- Tension d'alimentation = 4.8v-6v.
- Dimension : 24 x 13 x 29 mm.

- Poids : 8gr.
- Couple : 1.6 kg/cm.
- Vitesse : 0.12sec/60°.
- Pignons : nylon.

III 1.6 Capteur ultrason:

On va utiliser ce dispositif pour empêcher le robot de frapper les murs ou d'autres obstacles sur son chemin.

Nous avons choisi le capteur ultrason **HC-SR04**, car il dispose simplement de 4 pins: VCC, TRIG, ECHO, GND. Il est donc très facile de l'interfacer à un microcontrôleur.

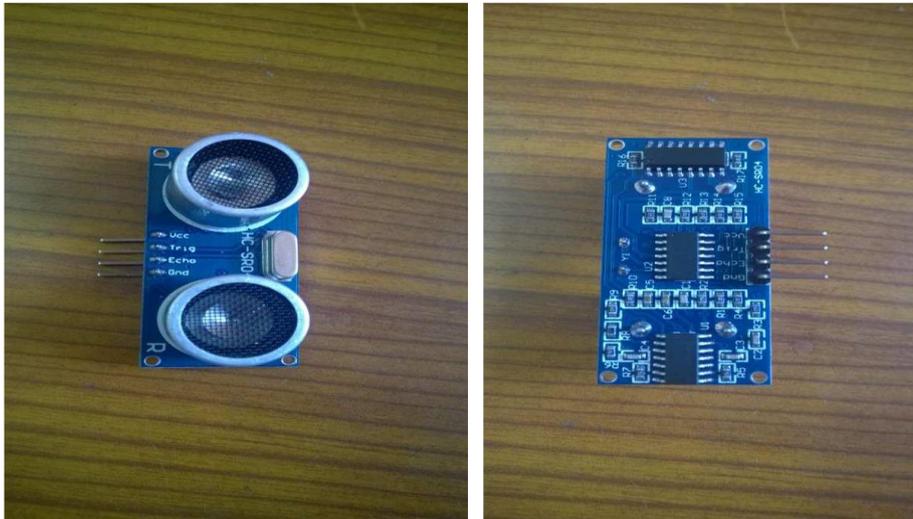


Figure III.10: Capteur Ultrason HC-SR04

Le fonctionnement du module est le suivant :

Il faut envoyer une impulsion niveau haut (5v) pendant au moins 10 μ s sur la broche '**Trig Input**' cela déclenche la mesure. En retour la sortie '**Output**' ou '**Echo**' va fournir une impulsion + 5v dont la durée est proportionnelle à la distance si le module détecte un objet.

Afin de pouvoir calculer la distance en cm, on utilisera la formule suivante

Distance = (durée de l'impulsion) (en μ s) / 58 (basée sur la formule de la vitesse du son)

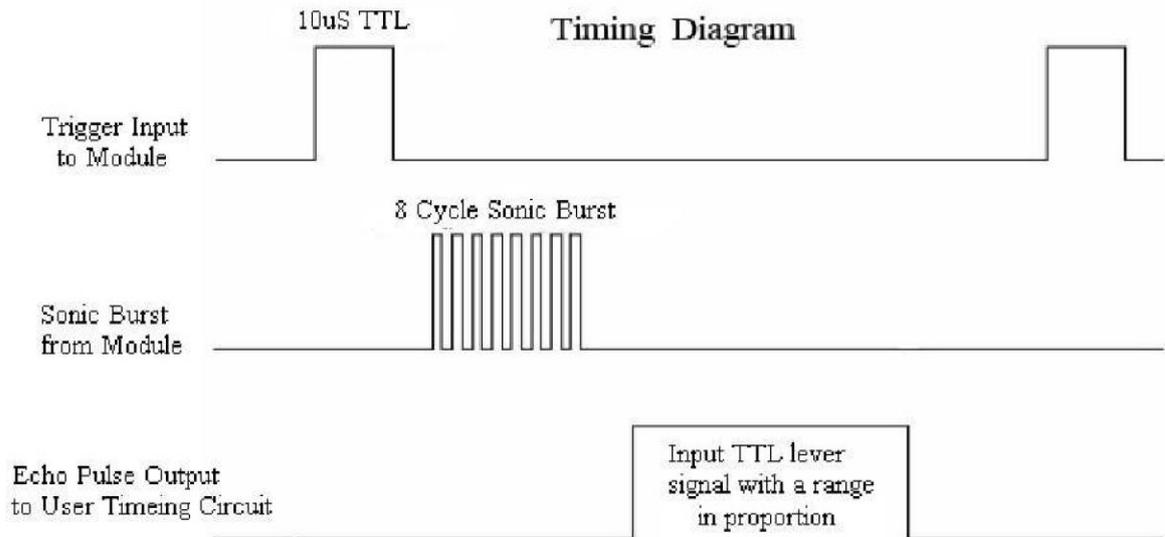


Figure III.11: Fonctionnement du capteur ultrason

III.1.7 Les capteurs à infrarouge

Le capteur détecte si une zone réfléchissant ou absorbant la lumière se trouve devant. Il montre laquelle des 2 zones il s'agit via une sortie numérique, comme vous pouvez le voir sur l'image ci-dessous. La sensibilité (plage minimale) du capteur peut être réglée par le contrôleur. Ce comportement peut être utilisé pour suivre automatiquement une ligne avec un robot.



Figure III.12: capteur à infrarouge

C'est le système que nous avons présenté pour nos TPE : il consiste à disposer 3 capteurs alignés sur une ligne perpendiculaire (et décalée par rapport au centre du robot) au sens de déplacement du robot.

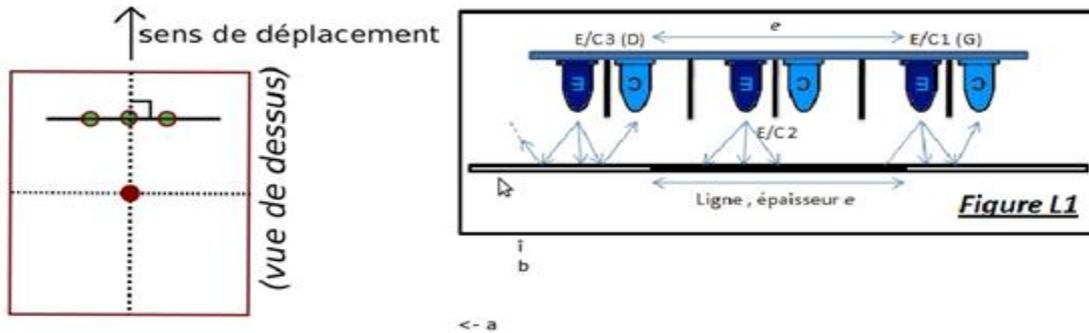


Figure III.12: disposition des capteurs

Ce système consiste à détecter les deux sorties de lignes possibles, en supposant qu'initialement le robot est correctement placé sur la ligne. Si on détecte que le robot sort de celle-ci, il tournera à l'instant même dans le sens favorisant le retour sur la ligne. Ici, nous utilisons le capteur central pour certifier si le robot est bien sur la ligne (sachant que son épaisseur d'est suffisamment grande pour ne pas être "cachée" entre deux capteurs).

III. 2 description logiciel:

III .2.1 Arduino IDE:

IDE (Integrated Development Environment) est un programme spécial exécutable sur votre ordinateur qui vous permet d'écrire des esquisses pour la carte Arduino dans un langage simple sur le modèle du langage de traitement. La magie se produit lorsque vous appuyez sur le bouton qui télécharge l'esquisse à la carte: le code que vous avez écrit est traduit dans la langue de C (qui est généralement assez difficile pour un débutant d'utiliser), et est passé au compilateur avr-gcc, une pièce importante du logiciel open source qui fait la traduction finale dans la langue comprise par le microcontrôleur. Cette dernière étape est très importante, parce que c'est là où Arduino rend votre vie simple en cachant autant que possible de la complexité de la programmation des microcontrôleurs.

L'interface de logiciel:

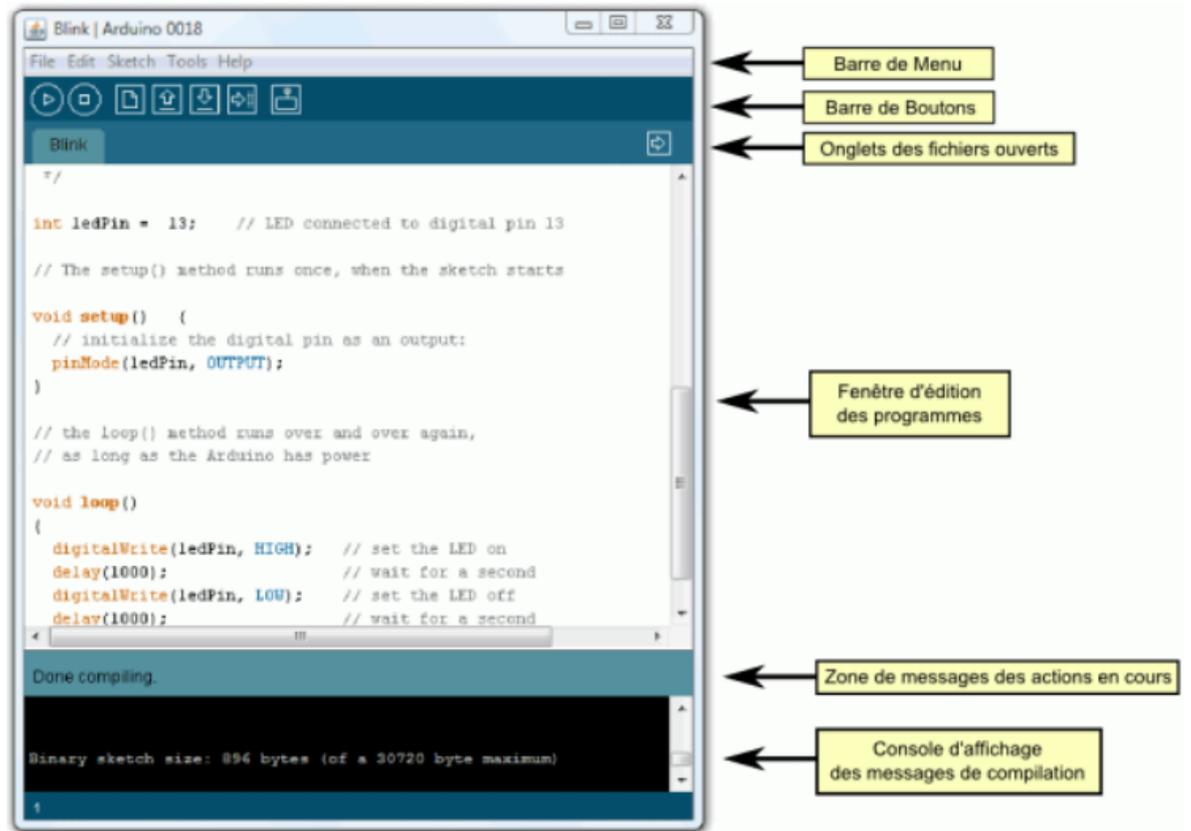


Figure II.13: Interface de logiciel arduino.

Cette fenêtre vide sera remplie de mots et de chiffres et d'autres textes. Cette fenêtre est comme tout autre logiciel que vous avez utilisé. Elle contient des menus, des boutons, des alertes spéciales, et toutes sortes de contrôles: Ouvrir, Enregistrer, et le bouton du moniteur de série à l'extrême droite. Nouveau: crée une nouvelle esquisse. Ouvert: présente un menu de tous les croquis dans votre dossier de croquis. Cliquez sur l'un va ouvrir dans la fenêtre actuelle. Enregistrer: enregistre votre croquis.

Vérifiez: ce bouton vous permet de vous assurer que votre logiciel est exempt d'erreurs de syntaxe. Envoyez: ce bouton peut à la fois vérifier et télécharger un croquis à l'Arduino si aucune erreur d'orthographe ou de mise en forme n'est trouvée.

Serial Monitor: ce bouton vous permet d'ouvrir le moniteur de série et d'afficher les informations en provenance du port série sur l'Arduino. Le moniteur de série est comme un outil pour "parler avec l'Arduino et vous l'utilisez pour voir les choses d'intérêt.

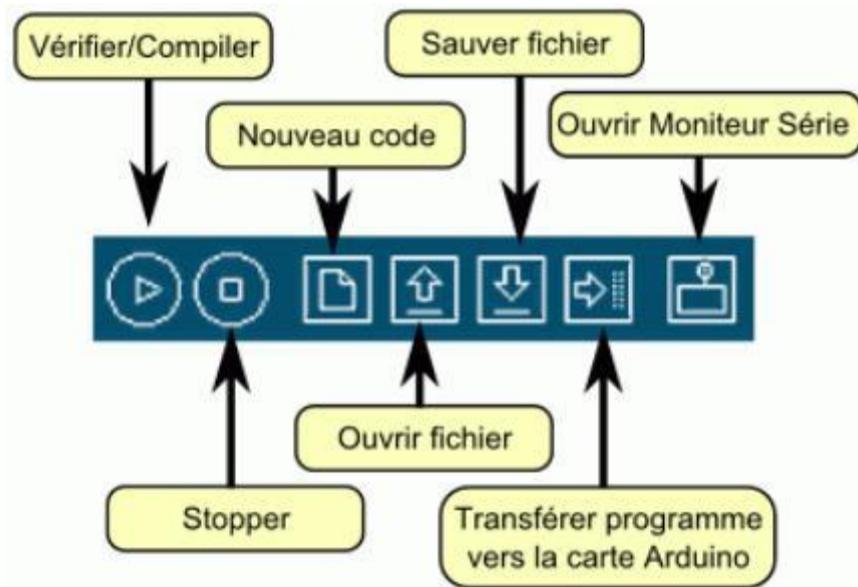


Figure III.14: Détail de barre de bouton

III.2.2 Langage de programmation Python

Python est un langage interprète, ce qui veut dire que vous pouvez écrire un programme ou un script et l'exécuter directement sans le compiler en code machine. Les langages interprètes sont souvent un peu plus rapides à écrire que d'autres, et ils fournissent quelques avantages supplémentaires.

Avec Python, par exemple, il n'est pas nécessaire d'indiquer à votre ordinateur si une variable est un nombre, une liste ou une chaîne de caractères. L'interpréteur devine les types de données lors de l'exécution d'un script. L'interpréteur de Python est capable de fonctionner de deux façons différentes : comme un Shell interactif pour exécuter les commandes individuelles, ou comme un programme en ligne de commande pour exécuter les scripts indépendamment.

L'environnement intégré de développement (IDE) pour Python fourni avec le Raspberry Pi s'appelle **IDLE**

chapitre 04

IIV.1 Introduction :

Un robot mobile se compose à trois parties essentielles :

- la partie mécanique.
- l'électronique de commande qui se compose à son tour de trois unités :
 - Unité de commande et de décision (microcontrôleur).
 - Unité de Perception de L'environnement (les capteurs).
 - Unité de mouvement des moteurs (circuit de puissance).
- La partie de programmation.

Dans ce chapitre présentera et expliquera les différentes étapes de la conception et la réalisation de notre robot mobile.

IIV.2 Cahier de charge :

Notre objectif est de réaliser un robot à vitesses différentielles qui a les caractéristiques suivantes :

- Il devrait être un véhicule qui peut aller en l'avant, en arrière et tourner.
- Il devrait être facile à monter et à démonter.
- Il devrait y avoir un mode de déplacement avec la capacité de détecter des obstacles avant et les éviter.
- Il devrait avoir un mode de suivi d'une ligne à l'aide des capteurs de réflecteur IR disposés dans le châssis.
- Il devrait être facile de manipuler, modifier et améliorer.
- Il devrait avoir un mode navigation avec GPS

On peut déjà remarquer l'émergence 02 modes :

- Mode suiveur de ligne (Line follower robot)
- Mode intelligent

IIV.3 MODE 01 : suiveur de ligne

Chapitre 04 : Réalisation et tests du robot mobile connecté en navigation

La fonctionnalité de ce mode est que le robot soit capable de suivre une ligne tracée au sol en utilisant ses capteurs de vision (vision sensors). Le modèle qu'on a choisi est constitué de 02 capteurs de infrarouge monté à l'avant du châssis (Leftsensor et Rightsensor). Lorsque le capteur est en face du sol (blanc), son émetteur reçoit de la lumière et transmet cette information au micro contrôleur. Dans le cas où le capteur est sur une surface noire, aucune information ne serait transmise. Les moteurs du robot vont être commandés en se basant des informations reçues des capteurs par le micro contrôleur.

Le guidage de notre robot est fait par une bande noire tracée sur une surface blanche. Pour que le robot fasse un mouvement tout droit l'OUTPUT des capteurs doit être en état normal. On a défini cet état par 101 où les chiffres 0 et 1 sont l'information transmise au micro contrôleur par les capteurs. L'état normal consiste à Leftsensor et Rightsensor sont face à la bande noire. Lorsque cet état est perturbé, dans le cas des virages ou courbe, une action corrective aura lieu par la procédure suivante : Si le Leftsensor commence à transmettre un 1 ce qui est lié à un virage à gauche, le moteur gauche s'arrête et le robot tourne à gauche jusqu'à l'état normal est atteinte.

Si le Rightsensor change son OUTPUT en 1 donc le robot doit tourner à droite en coupant le signal du moteur droit jusqu'à l'obtention de l'état normal.

IIV.3 .1La détection de ligne et mouvement de robot :

Ce tableau et le figure montre les valeurs possibles des détecteurs de ligne et le mouvement du robot :

état capteur à gauche	état capteur a droit	Mouvement De robot
1	1	Robot en avant
0	0	Robot en arrêtez
0	1	Robot à droit
1	0	Robot à gauche

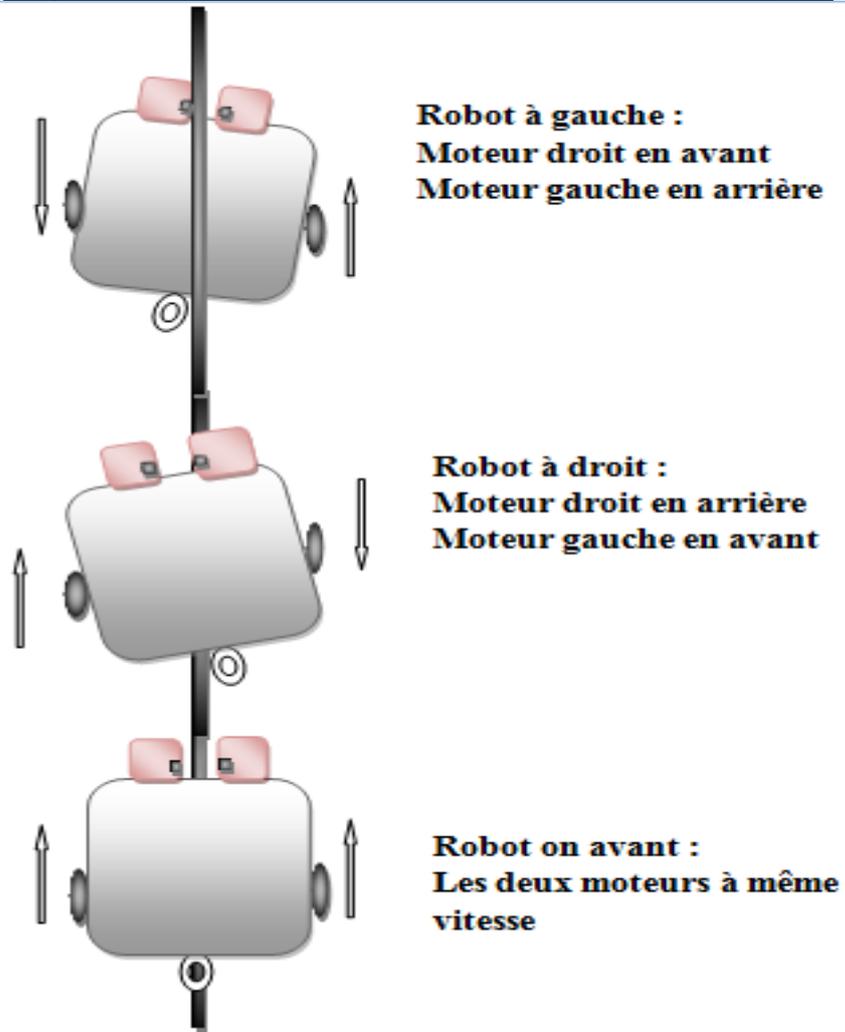


Figure IIV.1:La disposition du robot sur la ligne.

IIV.3 .2 Organigramme pour suivre de trajectoire :

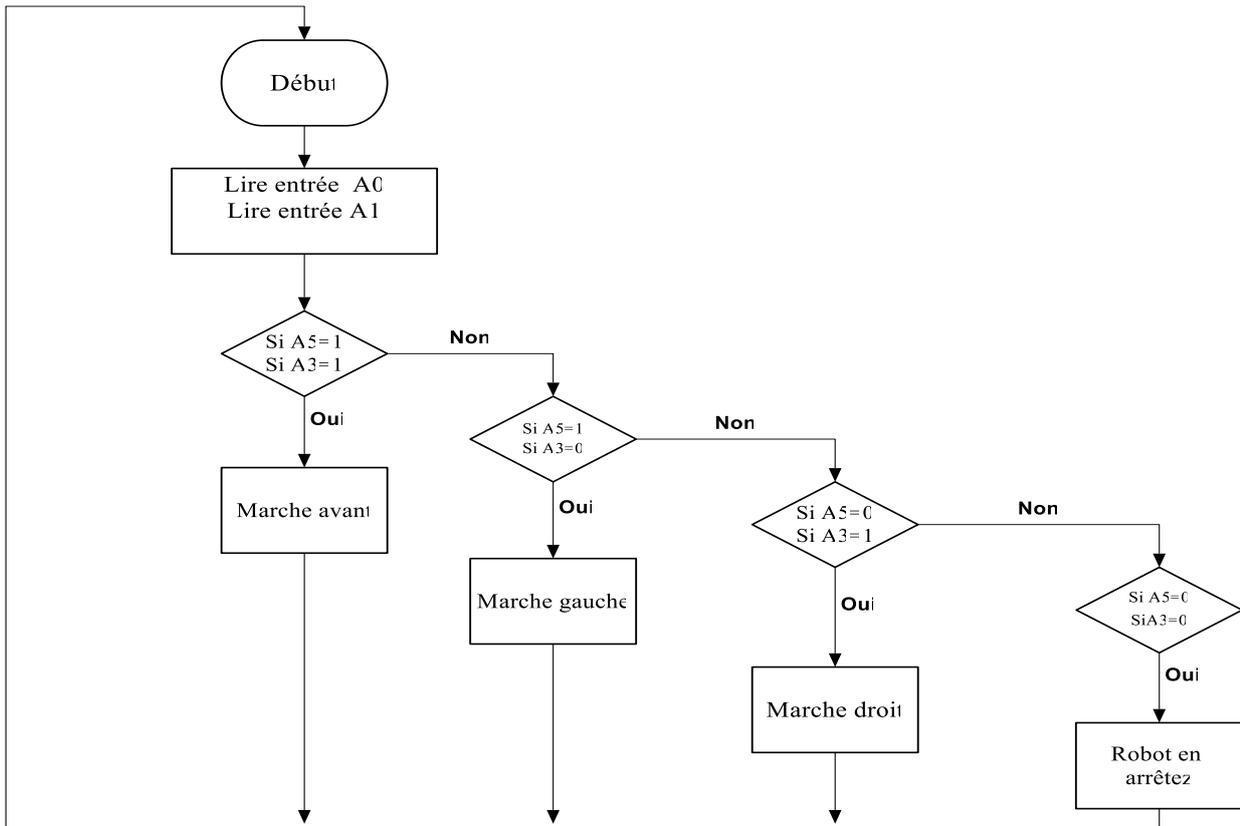


Figure IIV.2: Organigramme de suivi de ligne.

IIV.3 .3 test de capteur infrarouge:

circuit d'infrarouge

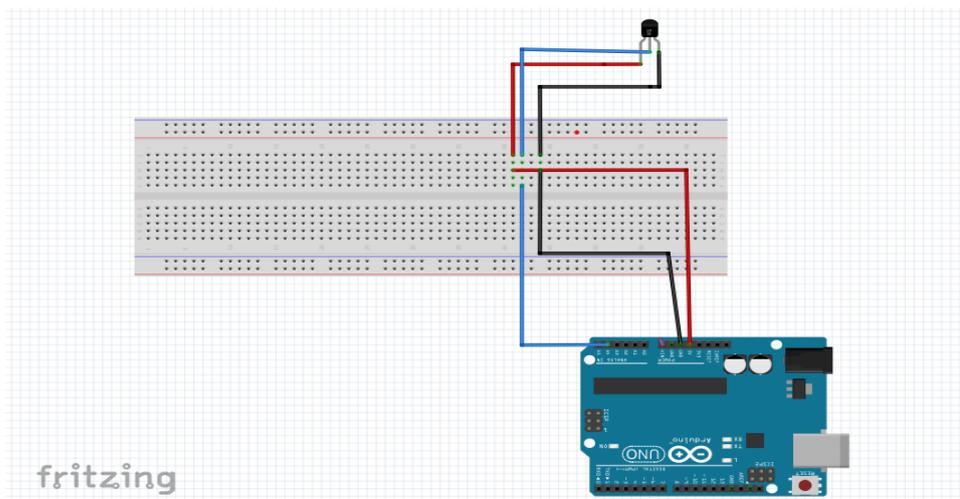


Figure IIV.3: Schéma du capteur infrarouge avec Arduino

La réalisation de robot

Le capteur doit être alimenté par une tension de 5V et son GROUND doit être connecté avec le GROUND d'Arduino et out connecté à un PIN digitales.

IIV.3.4 Le code Arduino pour test e capteur infrarouge et la resulta:

```
int Sensor=A4;
int outputsensor;
void setup() {

  Serial.begin(9600);
  pinMode(A4, INPUT);
}
void loop(){
  outputsensor=digitalRead(Sensor);
  Serial.print("outputsensor=");
  Serial.println(outputsensor);
  delay(1000);
  Serial.print("outputsensor=");
  Serial.println(outputsensor);
}
```

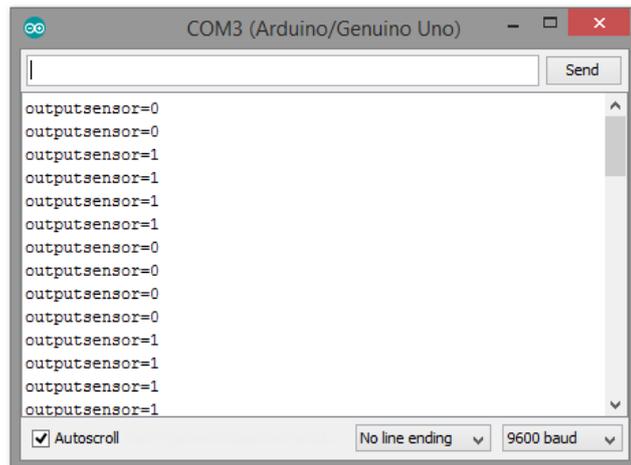


Figure IIV.4: Le code Arduino pour test e capteur infrarouge et la resulta

0:le capteur ont face à la sol

1: le capteur ont face à la bond noir

Test final :

Après finalisation du montage, nous avons testé notre robot dans un trajectoire

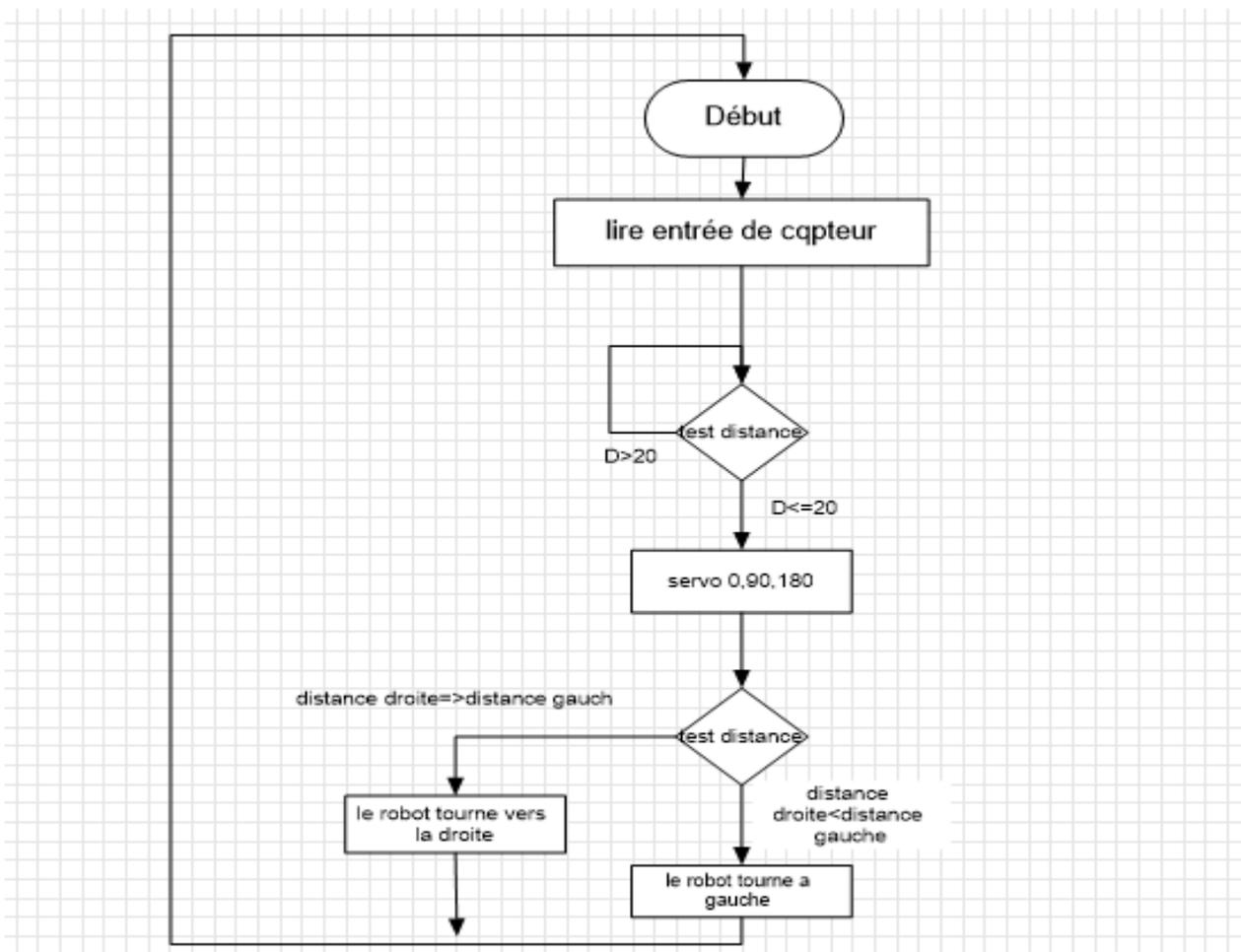


La réalisation de robot



Figure IIV.4: les test final pour un suivre de trajectoire

IIV.3 .4 Organigramme pour évitement des obstacles



IIV.3.5 test de capteur ultrason :

Circuit d'évitement d'obstacle :

Le capteur doit être alimenté par une tension de 5V et son GROUND doit être connecté avec le GROUND d'Arduino. Son ECHO et TRIG sont connectés à des PINs digitales

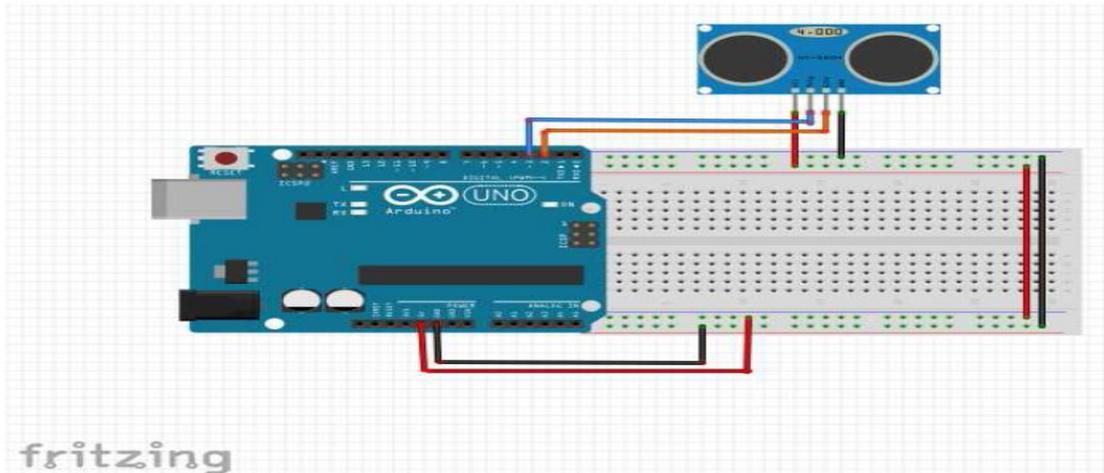


Figure IIV.5: Schéma du capteur ultrason avec Arduino

IIV.3.6 Le code Arduino pour test e capteur ultrason et la resulta:

```
int trig = 11;
int echo = 12;
long lecture_echo;
long cm;

void setup()
{
  pinMode(trig, OUTPUT);
  digitalWrite(trig, LOW);
  pinMode(echo, INPUT);
  Serial.begin(9600);
}

void loop()
{
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  lecture_echo = pulseIn(echo, HIGH);
  cm = lecture_echo / 58; // lecture_echo*340/(2*10000)
  Serial.print("Distance en cm : ");
  Serial.println(cm);
  delay(1000);
}
```

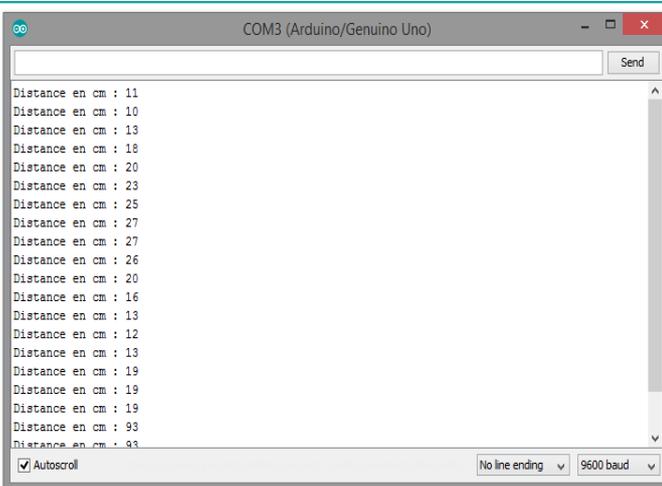


Figure IIV.6: Le code Arduino pour test e capteur ultrason et la resulta

IIV.4 MODE 02 : navigation avec GPS

La fonctionnalité de ce mode est que le robot soit capable de le robot Tracer et suivre une ligne en utilisant un GPS qui envoie la localisation de robot à l'application android. L'application trace la trajectoire et renvoie les données vers le robot qui peut suivre la trajectoire. Le robot peut être suivi en utilisant un serveur web conçu avec HTML

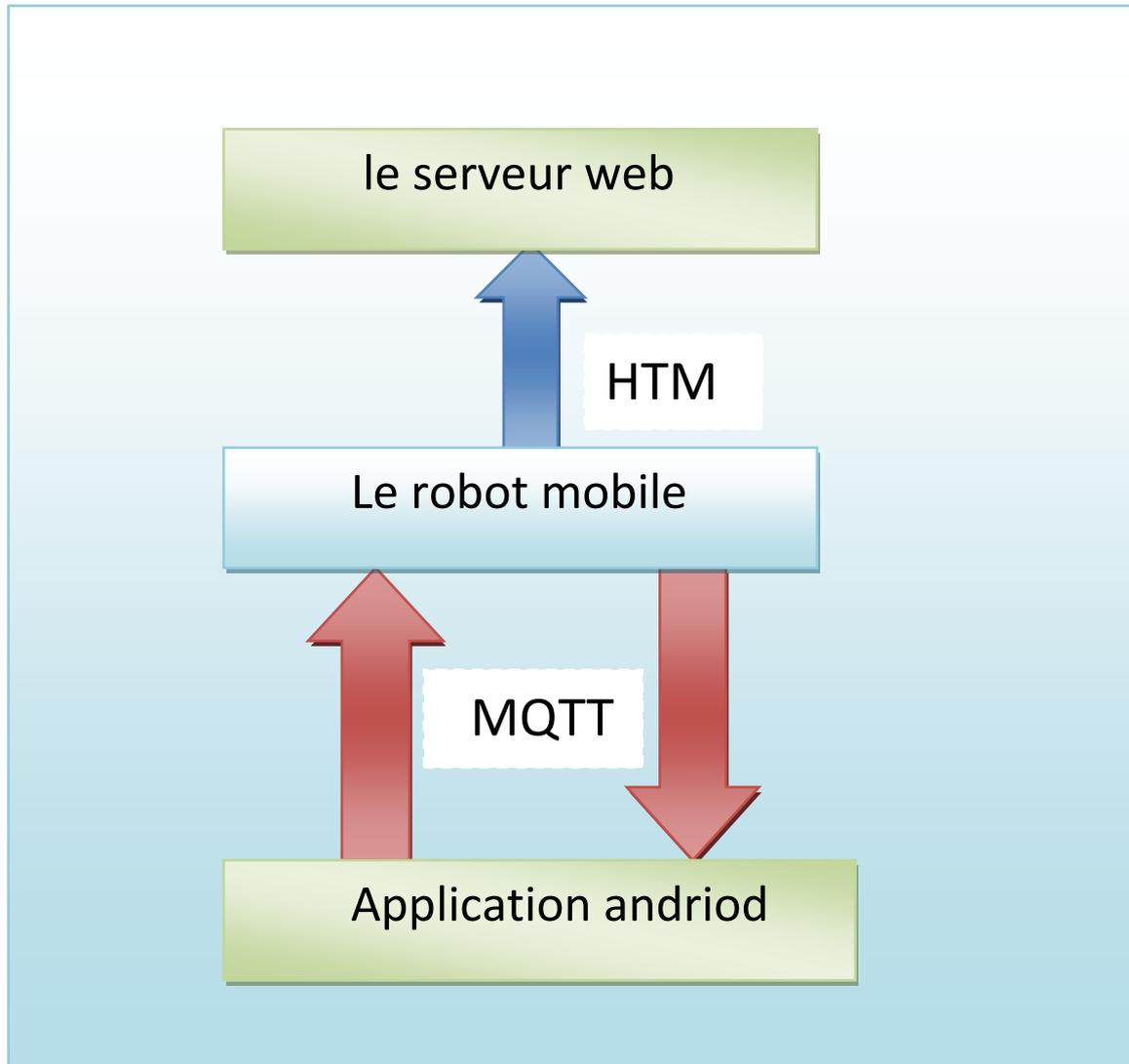


Figure IIV.7: schéma synoptique de notre robot mobile.

IIV.4 .1 La communication entre le robot mobile et application andriod :

pour communiquer le robot avec l'application. on utilisant mqtt

Que signifie MQTT

MQTT (MQ Telemetry Transport) est un [protocole](#) de messagerie léger qui facilite la transmission des données de télémétrie pour les clients [réseau](#) dont les ressources sont limitées. Ce protocole, qui suit un modèle de communication de type publication/abonnement via un broker de messages, est utilisé pour les échanges machine-à-machine ([M2M](#)). Il joue un rôle important dans [l'Internet des objets](#).

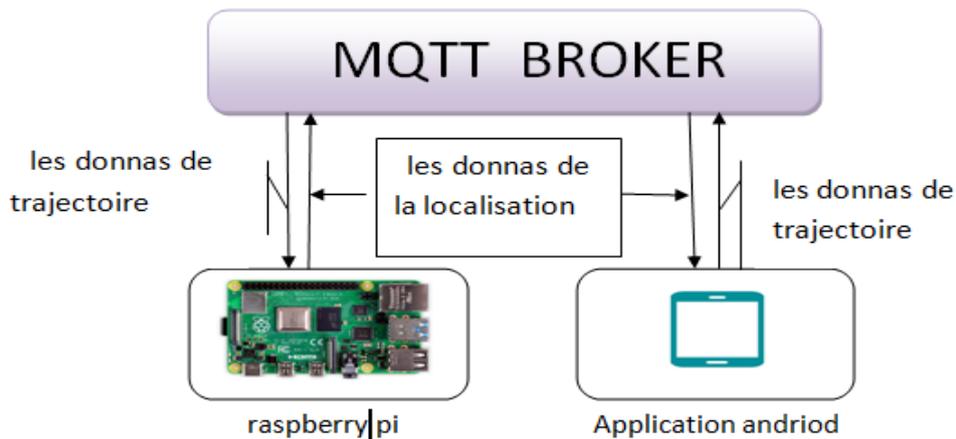


Figure IIV.8: schéma synoptique de MQTT

Les codes python pour MQTT

1 code python pour les donnas de la localisation

```
import paho.mqtt.client as mqtt
import ast
import json
# The callback for when the client receives a CONNACK response from the server.
def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))

    # Subscribing in on_connect() means that if we lose the connection and
    # reconnect then subscriptions will be renewed.
    client.subscribe("test_map_gps")

# The callback for when a PUBLISH message is received from the server.
def on_message(client, userdata, msg):
    message = ast.literal_eval(msg.payload.decode())
    print(message)
    print(len(message))

def on_publish(client, userdata, result):
    print("data published \n")

    # your code

client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message
client.on_publish = on_publish

client.connect("mqtt.eclipse.org", 1883, 60)

# your raspberry pi locations

longitude = 10
latitude = 11

# send data to the application
data = dict()
data['lon']=longitude
data['lat']=latitude
client.publish('send_locations_morsli',json.dumps(data))
client.loop_forever()
```

Figure IIV.9:code python pour les donnas de la localisation

2. code python pour les donnas de trajectoire

La réalisation de robot

```
import paho.mqtt.client as mqtt
import ast

# The callback for when the client receives a CONNACK response from the server.
def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))

    # Subscribing in on_connect() means that if we lose the connection and
    # reconnect then subscriptions will be renewed.
    client.subscribe("test_map")

# The callback for when a PUBLISH message is received from the server.
def on_message(client, userdata, msg):
    message = ast.literal_eval(msg.payload.decode())
    print(message)
    print(len(message))

    # your code

client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message

client.connect("165.232.70.175", 1883, 60)

# Blocking call that processes network traffic, dispatches callbacks and
# handles reconnecting.
# Other loop*() functions are available that give a threaded interface and a
# manual interface.
client.loop_forever()
```

*Figure IIV.10:*code python pour les donnas de trajectoire

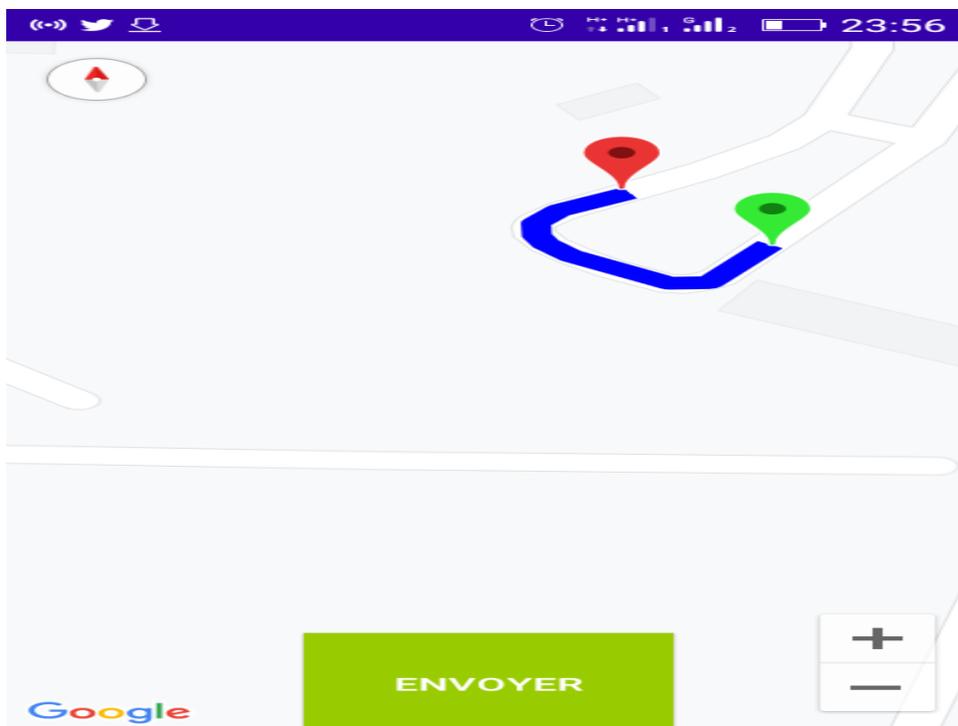


Figure IIV.11: l' application mobile

La réalisation de robot

```
type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\user\Desktop\df.py =====
Connected with result code 0
[{'lon': '0.15452', 'lat': '34.85974'}, {'lon': '0.15425', 'lat': '34.85941'}, {'lon': '0.15424', 'lat': '34.8594'}, {'lon': '0.15421', 'lat': '34.8594'}, {'lon': '0.15419', 'lat': '34.8594'}, {'lon': '0.15413', 'lat': '34.85943'}]
6
|
```

Figure IIV.12: les donnas de trajectoire

IIV.4 .2 Le robot mobile et serveur web

1.La création d'une page Web de suivi de la géolocalisation en temps réel

on utiliser API appelée [PubNub](#), qui est utilisée pour fournir des données en temps réel. Ici, on l'utiliser pour envoyer nos données GPS de Raspberry Pi à notre page Web. En vous inscrivant sur le site et en obtenant publish key , subscribe key et channel name .Aussi Nous avons besoin de la clé API. Nous l'avons obtenue sur [Google Cloud Console](#)

```
56 var pnChannel = "raspi-tracker";
57
58 var pubnub = new PubNub({
59   publishKey: 'pub-c-38a3a17f-76a3-47ed-a66d-63e10c79447d',
60   subscribeKey: 'pub-c-38a3a17f-76a3-47ed-a66d-63e10c79447d'
61 });
```

Figure IIV.13: publish key , subscribe key et channel name

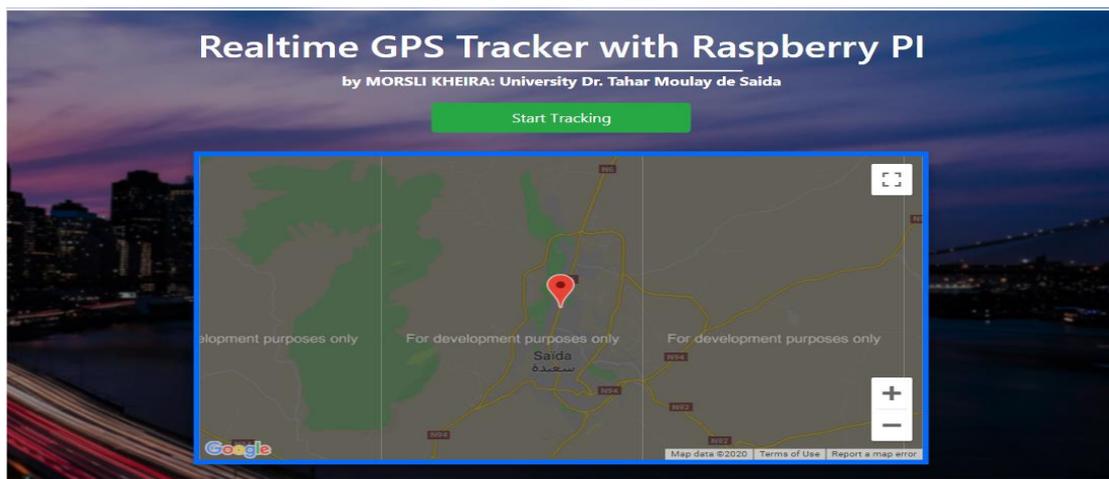


Figure IIV.13: serveur web

2.Code python pour le Raspberry pi:

La réalisation de robot

Ici encore, nous devons écrire la clé de publication et la clé d'abonnement. Et notez également que nous avons utilisé le nom de chaîne "raspi-tracker"

```
pnChannel = "raspi-tracker";

pnconfig = PNConfiguration()
pnconfig.subscribe_key = "pub-c-38a3a17f-76a3-47ed-a66d-63e10c79447d"
pnconfig.publish_key = "pub-c-38a3a17f-76a3-47ed-a66d-63e10c79447d"
pnconfig.ssl = False

publisher = Publisher(pnconfig)
```

Figure IIV.14: code python

Maintenant, si nous exécutons ce code python sur Raspberry Pi, nous verrons immédiatement que le marqueur sur la carte pointera vers l'emplacement actuel du robot . Et comme il se déplacera à différents endroits, le marqueur se déplacera également et dessinera également le chemin.

Conclusion :

Ce chapitre était le noyau de notre projet fin d'étude. On a commencé par la conception des circuits de chaque mode de notre robot, Puis nous avons fait une explication pour incarner chaque mettre, en finissant par la réalisation de ces trois modes.

Conclusion générale

ce projet de fin d'étude est un projet vaste qui regroupe plusieurs disciplines, et qui requiert des connaissances et des capacités multidisciplinaires

Il nous a permis de mettre en œuvre les connaissances requises le long de notre cursus, il nous a donné l'opportunité de construire et contrôler un robot mobile, il nous a permis aussi d'ouvrir l'esprit sur la robotique mobile et sur la programmation mobile,

A travers ce projet on a exploré la puissance du Arduino et Raspberry pi, ces dernier était le cœur du robot et du projet, sa programmation a permis de commander le robot qui dispose de quatre moteurs à courant continu, on est arrivé à lui donner la capacité de se mouvoir dans son environnement. L'association des capteurs infrarouges a donné au robot plus d'autonomie, en évitant des obstacles sans notre intervention.

Vu le manque du temps nous n'avions pas pu tout faire comme par exemple la commande du robot à travers l'application mobile, l'échange des données entre le robot et le smartphone.

Enfin, ce fut un immense plaisir de travailler sur ce genre de projet très intéressant, de découvrir de nouveaux aspects, on espère que ce robot mobile continuera à se développer avec de futures PFE.

REFERENCES

BIBLIOGRAPHIE

- [1]: « Robotics » , dans English Oxford living Dictionaire
- [2]: Introduction à la robotique par Laetitia Matignon Université de Caen, France
- [3]: John J. Craig, *Introduction to robotics - mechanics and control*, Addison-Wesley.
- [4]: <https://sciencepost.fr/2017/12/robot-humanoide-transpire-sport/>
- [5]: Nizar, Chelly. "Support de cours Électricité Générale." Nizar, C. Support de cours
- [6]: Roland Siegwart and Illah R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, the MIT press. ISBN: 0-262-19502-X.
- [7]: Etude et conception d'un Robot marchant par HEDDOUCHE Kamel Université Université Mohamed Khider Biskra, année 2014
- [8]: Introduction à la robotique par Laetitia Matignon Université de Caen, France
- [9]: Frédéric GIAMARCHI, «*Petits robots mobiles*», Edition Technique et Scientifiques Françaises (ETSF), 2000
- [10]: Alain BRUSKI, «*Robotique générale*», Edition Ellipses Paris, 1988
- [11]: https://www.ensta-bretagne.fr/jaulin/mastersds_cours_robot_boimond.pdf,
- [12] <https://www.hilarispublisher.com/open-access/dynamic-modelling-of-differentialdrive-mobile-robots-using-lagrange-and-newtoneuler-methodologies-a-unified-framework-2168-9695.1000107.pdf>
- [13] Commande cinématique en vitesse/Cap d'un robot unicycle ,El Mouderrib Iliasse , Belmiloud Jedjiga
- [14] https://globaljournals.org/GJRE_Volume14/1-Kinematics-Localization-and-Control.pdf