

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique

Université Akli Mohand Oulhadj - Bouira -

Tasdawit Akli Muḥend Ulḥağ - Tubirett -



وزارة التعليم العالي والبحث العلمي

جامعة أكلي محمد أولحاج

- البويرة -

Faculté des Sciences et des Sciences Appliquées

كلية العلوم والعلوم التطبيقية

Référence :/MM/2021

المرجع:/م/م / 2021

Mémoire de Master

Présenté au

Département : Génie Électrique

Domaine : Sciences et Technologies

Filière : Electronique

Spécialité : Electronique des systèmes embarqués

Réalisé par :

OUCHFOUNE BRAHIM

Et

ZINAFI FARHAT

Thème

CONCEPTION ET REALISATION D'UN ROBOT DE NETTOYAGE

Soutenu le: **30/10/2021**

Devant la commission composée de :

Mr	MOUDACHE .Said	Prof.	Univ. Bouira
	Président		
	BENSAFIA . Yassine	M.C.A	Univ. Bouira
	Rapporteur		
	NOURIN . Mourad	M.C.B	Univ. Bouira
	Examineur		

Remerciements

Ce travail a été effectué au sein du Département des Sciences et sciences appliquées de l'Université de Brouira.

En premier lieu nous avons remercié le Dieu tout puissant de m'avoir octroyé la patience, la force et le courage de traiter ce travail à terme.

Aussi nous tenons à dédicacer mes sincères salutations et remerciements à mon encadreur de thème monsieur **Bensafia Yassine** à l'Université Akli Mohand Oulhadj de Bouira pour sa disponibilité et maintenabilité, pour sa lecture, suggestion et remarques et surtout pour sa confiance sans limite mise en moi tout au long de ce projet de préparation.

Nous vous prie de bien vouloir agréer le témoignage de ma plus vive reconnaissance et mon profond respect.

Nous remercions particulièrement et sincèrement tout les membres de famille pour leur soutien moral et physique durant mon étude, merci de m'avoir encouragé, et cru en moi.

Nous remercions également mes amis mes collègues de l'université de Bouira, sans oublier mes collègues de la section électronique et électrotechnique qui m'ont encouragé tout au long de ce projet et m'ont beaucoup aidé. Enfin, nous remercions les membres de jury qui m'ont fait l'honneur d'accepter de juger ce modeste travail de fin d'étude.

Dédicaces

Je dédie ce mémoire à :

À Mes très chers parents pour leur bienveillance et leur soutien tout au long de mon parcours scolaire et universitaire.

À tout la famille.

À tous mes amis et proches.

Et à toute la promotion.

Ferhat.

Dédicace

*Je dédie ce modeste travail à ma mère et mon père symboles
d'affection et d'amour, qui ont sacrifié tout leur temps
pour mon bonheur et ma réussite qu'ils trouvent ici le
témoignage de ma profonde reconnaissance. Ma très
grande affection également à tous les membres de ma
famille Mes cousins et cousines qui m'ont soutenues
durant toute la période de mes études. À tous mes amis et
mes collègues et à tous ceux que J'aime*

BRAHM.

Table des matières

Liste des figures.....	I
Liste des tableaux	III
Liste des Acronymes	IV
Introduction générale	1
Chapitre I : Généralités sur les robots mobiles.....	3
I.1. Introduction	3
I.3. Définitions	3
I.3.1. Définition de la robotique.....	3
I.3.2 Définition de robot	3
I.2. Historique	4
I.2.1. Origines de la robotique	4
I.2.2. Origine de mot robot.....	4
I.2.3. Dates marquants de la Robotique	4
I.4. Les types de robots	5
I.5. Caractéristiques d'un robot	6
I.6. Les générations de robot.....	7
I.7. L'architecture des robots	8
I.7.1. La structure mécanique et la motricité	8
I.7.1.1. Les robots mobiles à roues	8
I.7.1.1.1. Un robot de type unicycle.....	9
I.7.1.1.2. Robot tricycle	9
I.7.1.1.3. Robot voiture.....	10
I.7.1.1.4. Robot omnidirectionnel	10
I.7.1.1.5. Comparaison des différents types	11
I.7.1.2. Les robots mobiles à chenilles	11
I.7.1.3. Les robots mobiles marcheurs	11
I.7.1.4. Les robots mobiles rampants	12
I.7.2. Le système de localisation	13
I.7.2.1. La localisation relative.....	13
I.7.2.1. 1. L'odométrie.....	13
I.7.2.1.2. La localisation par capteurs inertiels.....	13
I.7.2.2. La localisation absolue	13
I.7.3. Organes de sécurité	13

I.7.4.Le système de traitement des informations et gestion de tâches	15
I.7.4.1.Structure d'un robot mobile	15
I.7.4.2.Architecture	15
I.7.4.2.1.Une unité de traitement ou microprocesseur	15
I.7.4.2.1.1. Type de processeur	15
I.7.4.2.1.2. La zone mémoire.....	16
I.7.4.2.3. Les interfaces Entrées/Sorties	20
I.8. Mode de fonctionnement de robot	20
I.9. Classification des Robots Mobiles.....	21
I.10. Quelques exemples de robots mobiles	21
I.11.Architectures des planificateurs de trajectoires	22
I.11.1 Planification Globale	22
I.11.2. Planification locale (réactive)	23
I.11.2.1. Architecture de Brooks	23
I.12.Applications	24
I.13. Avantages et inconvénients des robots	25
I.14. Conclusion.....	26
Chapitre 2 : Conception du robot mobile	27
II.1. Introduction:.....	27
II.2. Les composants de robot.....	28
II.3.Les aspects fondamentaux de notre projet	29
II.3.1. Châssis.....	29
II.3.2.Roues	30
II.3.3.Moteur	30
II.3.4.Arduino	30
II.3.4.1.Présentation Arduino	30
Arduino est-il un microcontrôleur?.....	33
II.3.4.2.Types de carte Arduino :	34
II.3.4.3.Alimentation de la carte Arduino	35
3.4.3.1. Les pins (ou broches) d'alimentation.....	36
II.3.4.4. Entrées et sorties	36
II.3.5.La puce L298N	37
II.3.6.Bluetooth HC-05.....	43
II.3.7. Alimentation	44
II.4.Les Logiciels utilisés.....	45
II.4.1.Logiciel l'Arduino	45

II.4.1.1. Les étapes de téléchargement du programme	45
II.4.2. ISIS PROTEUS	46
II.5. Conclusion	46
Chapitre 3 : Réalisation du robot de Nettoyage	47
III.1. Introduction	47
III.2. Architecture du robot mobile	47
III.3. Principe de fonctionnement du robot mobile	47
III.3.1. Algorithme de commande par Bluetooth	48
III.3.2. Organigramme du robot :	49
III.4. Montage matériel et logiciel	50
III.4.1. Brochage sur l'Arduino	51
III.4.2. Configuration des moteurs	51
III.5. Logiciels utilisés	53
III.5.1. Circuit du robot sous Proteus ISIS	53
III.5.2. Logiciel Fritzing	55
III.5.3. Logiciel de programmation Arduino	56
III.6. Télécommander notre robot mobile	57
III.7. Conclusion	58
Conclusion Générale	59
Références bibliographiques	61
Références Webiographie	62
Annexe	63
Résumé	65

Liste des figures

CHAPITRE I : Généralités sur les robots mobiles.

Fig. I. 1. représente un exemple sur le premiere type.....	5
Fig. I. 2. représente un exemple sur le second type	6
Fig. I.3. Représente un exemple sur le dernier type.....	6
Fig. I. 4. Robot de type unicycle	9
Fig. I. 5. Robot de type tricycle.....	9
Fig. I. 6. Robot de type voiture	10
Fig. I. 7. Robot de type omnidirectionnel	10
Fig. I. 8. Robots mobiles à chenilles	11
Fig. I. 9. Robots mobiles marcheurs.	12
Fig. I. 10. Robot mobile rampant.	12
Fig. I. 11. Les organes de robot.....	14
Fig. I. 12. Robot mobile de NASA	14
Fig. I. 13. Structure de base.....	15
Fig. I. 14. Robot mobile Hilare	21
Fig. I. 15. Robot mobile Pioneer P3-DX	22
Fig. I. 16. Planification globale	22
Fig. I. 17. Hiérarchie des niveaux dans l’architecture de Brooks.	24
Fig. I. 18. Différents types de robots mobiles selon le champ d’application	25

CHAPITRE II : Conception du robot mobile

Fig.II. 1. La structure générale d’un robot mobile	28
Fig.II. 2.Châssis.....	29
Fig.II. 3.Roues	30
Fig.II. 4.Moteur DC Fig.II. 5.Architecture interne de moteur DC	30
Fig.II. 6. Interagir de la carte Arduino avec le monde réel.....	Error! Bookmark not defined.
Fig.II. 7.Microcontrôleur ATmega328	33
Fig.II. 8.Les éléments de la carte Arduino	34
Fig.II. 9.Schéma des « ports »	37
Fig.II. 10.Schéma d’un pont en H	39
Fig.II. 11.Les deux configurations possibles de pont en H	39
Fig.II. 12.Module L298N (tout équipé)	40
Fig.II. 13.Entrés /Sorties du module L298N	41
Fig.II. 14.Bluetooth HC-05	44
Fig.II. 15.Batteries	44
Fig.II. 16.Interface IDE Arduino.....	45
Fig.II. 17.Interface d’utilisateur du logiciel.....	46

CHAPITRE III : Réalisation du robot de nettoyage.

Fig. III. 1. Architecture de notre robot mobile.	47
Fig. III. 2. Organigramme du robot.....	49

Fig. III. 3. Algorithme de téléguidage via Bluetooth.....	48
Fig. III. 4. Les composants de robot avant le brochage.....	50
Fig. III. 5. Les composants après brochage.....	51
Fig. III. 6. Les différents sens de robot.....	53
Fig. III. 7. Le Circuit de montage avec Isisproteuse.....	53
Fig. III. 8. brochage de hc-5 avec arduino.....	55
Fig. III. 9. Le montage final de notre projet par logiciel Fritzing.....	55
Fig. III. 10. Programmation sur Arduino.....	56
Fig. III. 11. Téléversement de programme vers la carte Arduino.....	56
Fig. III. 12. L'application android Bluetooth RC Contrôler.....	57

Liste des tableaux

CHAPITRE I : Généralités sur les robots mobiles.

Tableau. I. 1:comparaison entre les types des robots.....	11
Tableau. I. 2:Applications des robots mobiles	24

CHAPITRE II : Conception du robot mobile.

Tableau. II. 1:Caractéristiques de la carte Arduino Uno	35
Tableau. II. 2:Alimentation des composants	44

CHAPITRE III : Réalisation du robot de nettoyage.

Tableau. III. 1Connexion des composants avec Arduino.	51
Tableau. III. 2.Configuration des moteurs.....	52

Liste des Acronymes

AAAI	Association for the Advancemt of Artificial Inteligence.
ASIC	Application Specific Integrated Circuit.
ASIP	Application Specific Instruction Processor
ASSP	Application Specific System Processor
EEPROM	Eletrically-Erasable-and-Programmable Read Only Memory
EPROM	Erasable-and-Programmable Read Only Memory
GND	Ground
GPP	General Purpose Processor
LED	Light emitting Diode
MCC	Moteur à courant continue
NVRAM	No-Volatile Random Access Memory
Pin	Broche
PROM	Programmable Read Only Memory
PWM	Pulse Widh Modulation
RAM	Random Access Memory
ROM	Read Only Memory
RX	Receiver
TX	Transmmitter
USB	Universal Serial Bus
VCC	Common Collector Voltage
VLSI	Very Large Scale Integration

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.

L'objectif principal de notre travail est de réaliser un robot mobile pour nettoyage on utilisant une carte électronique Arduino pour assuré la tache principale du robot est de détection et assurer les signes du nettoyage.

Ce travail est alors organisé en trois chapitres de la manière suivante :

*Le premier chapitre présente certaines généralités concernant la robotique et les différents types des robots et robots mobiles.

Introduction générale

*Le second chapitre introduit quelques composants du robot comme les capteurs, les actionneurs et d'autres concepts liés à leurs utilisations, langages de programmation et en particulier l'IDE de l'Arduino.

*Le troisième chapitre représente le diagramme fonctionnel du robot, ainsi les étapes suivies à réaliser notre prototype robot mobile.

*Nous terminerons ce manuscrit par une conclusion générale.

Chapitre I : Généralités sur les robots mobiles

I.1. Introduction

Dans le domaine industriel, la robotique produit des automates réalisant des fonctions précises sur des chaînes de montage. La robotique produit aussi des engins capables de se mouvoir dans différents milieux : dangereux (pollués, radioactifs...), aérien, sous-marins, spatiaux... Outre l'industrie, la robotique concerne ainsi aujourd'hui la recherche scientifique, l'exploration spatiale et les activités de défense militaire ou de maintien de l'ordre. Elle intéresse également le secteur médical, pour les prothèses, les assistantes aux chirurgiens ou aux infirmiers.

Dans le but de clarifier l'utilité des robots, ce chapitre sera consacré à la présentation des généralités sur la robotique.

I.3. Définitions

I.3.1. Définition de la robotique

La robotique est la science qui s'intéresse aux robots. En fait, il s'agit d'un domaine multi disciplinaire : on y trouve des aspects concernant la mécanique, l'informatique, l'électronique.... [2].

I.3.2 Définition de robot

Un robot est un appareil dirige a l'aide d'un système de commande automatique a basée de microprocesseurs et qui été conçu pour accomplir une tâche précise, dans le domaine industriel, scientifique ou encore domestique.

OU : c'est un dispositif électronique programmable destiné à la commande de processus par un traitement séquentiel [sw1]. Il envoie des ordres Vers les préactionneurs (partie opérative ou PO côté actionneur)à partir de données d'entrées (capteurs) (partie commande ou PC côté capteur), de consignes et d'un programme informatique.

Il existe deux grandes familles de robots :

- Les robots manipulateurs (fixe).
- Les robots mobiles.

I.2. Historique

Pendant plusieurs années, laboratoires, industriels, informaticiens et mécaniciens vont continuer leurs travaux en parallèle. On accède ainsi côté industriel à la télé-opération et à une partie de la robotique classique, tandis que du côté informatique, on assiste à de grands progrès dans le domaine de l'intelligence artificielle. Ainsi, vers la fin des années soixante-dix, trois pôles géographiques principaux se distinguent (France, Japon, Etats-Unis). La synthèse de tous les travaux réalisés jusqu'alors donne enfin naissance aux robots mobiles autonomes (du robot domestique au robot militaire)[sw2].

C'est dans cet environnement de plus en plus automatisé que se fait sentir le besoin d'outils capables, non seulement d'effectuer des tâches répétitives ou encore impossibles à l'homme (porter des charges lourdes, découpage ultra précis, ...), mais aussi de manifester une certaine autonomie de déplacement dans des milieux hostiles à l'homme [sw2].

I.2.1. Origines de la robotique

Les ancêtres des robots sont les automates. Un automate très évolué fut présenté par Jacques de Vaucanson en 1738 : il représentait un homme jouant d'un instrument de musique à vent. Jacques de Vaucanson créa également un automate représentant un canard mangeant et refoulant sa nourriture après ingestion de cette dernière [sw3].

I.2.2. Origine de mot robot

Le mot robot qui apparaît pour la première fois en 1920 dans la pièce de théâtre de Karel Capek de science-fiction R.U.R. (Rossum's Universal Robots) a été inventé par son frère Josef à partir du mot tchèque « Robota » qui signifie « travail, besogne, corvée » [sw3].

I.2.3. Dates marquants de la Robotique [1]

1947 : premier manipulateur électrique télé opéré.

1948 : Grey Walter invente le premier robot mobile autonome, une tortue se dirigeant vers les sources de lumière qu'elle perçoit. Cependant, ce robot n'est pas programmable.

1954 : premier robot programmable.

1961 : Premier robot industriel mis en place dans une usine de General Motors.

Qui a fait premier robot avec contrôle en effort.

1963 : utilisation de la vision pour commander un robot.

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

1973 : premier robot mobile à roues.

1977 : Premier robot mobile français HILARE au LAAS (CNRS Toulouse)

1992 : Mise en place de la compétition annuelle AAAI sur la robotique mobile.

1995 : Mise en place de la Robo Cup.

1997 : Premier robot mobile extra planétaire sur Mars.

Depuis 2000 : Exploration,

2003 : Projet "Mars Exploration Rover" (Spirit & Opportunity). 2009 : Projet "Mars Science Laboratory" succédant au projet Rover, envoi prévu de Curiosity fin 2011.

I.4. Les types de robots [3]

Au cours de l'histoire on peut distinguer 3 types de robots correspondant en quelques sortes à l'évolution de cette "espèce" créée par l'Homme.

- Le **premier type** de machine que l'on peut appeler robot correspond aux "Automates". Ceux-ci sont généralement programmés à l'avance et permettent d'effectuer des actions répétitives.



Fig. I. 1. Robot articulé - KR 1000 titan

- Le **second type** de robot correspond à ceux qui sont équipés de capteurs (en fait les sens du robot). On trouve des capteurs de température, photoélectronique, à ultrasons pour par exemple éviter les obstacles et/ou suivre une trajectoire. Ces capteurs vont permettre au robot une relative adaptation à son environnement afin de prendre en compte des paramètres aléatoires qui n'aurait pu être envisagés lors de leur programmation initiale. Ces robots sont donc bien plus autonomes que les automates

mais nécessitent un investissement en temps de conception et en argent plus conséquent.



Fig. I. 2. Robot aspirateur trilobite

- Enfin le **dernier type** de robot existant correspond à ceux disposant d'une intelligence dite "artificielle" et reposant sur des modèles mathématiques complexes tels que les réseaux de neurones. En plus de capteurs physiques comme leurs prédécesseurs, ces robots peuvent prendre des décisions beaucoup plus complexes et s'appuient également sur un apprentissage de leurs erreurs comme peut le faire l'être humain. Bien sûr il faudra attendre encore longtemps avant que le plus "intelligent" des robots ne soit égal, tant par sa faculté d'adaptation que par sa prise de décisions, à l'Homme.

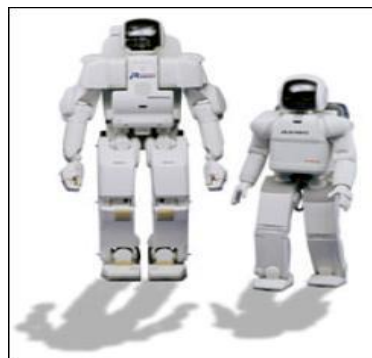


Fig. I.3. Robot marchant

I.5. Caractéristiques d'un robot [4]

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

- La charge maximum transportable (de quelques kilos à quelques tonnes), à déterminer dans les conditions les plus défavorables (en élongation maximum).

- L'architecture du S.M.A., le choix est guidé par la tâche à réaliser (quelle est la rigidité de la structure ?).

- Le volume de travail, défini comme l'ensemble des points atteignables par l'organe terminal. Tous les mouvements ne sont pas possibles en tout point du volume de travail.

L'espace de travail (*reachable workspace*), également appelé espace de travail maximal, est le volume de l'espace que le robot peut atteindre *via* au moins une orientation. L'espace de travail dextre (*dextrous workspace*) est le volume de l'espace que le robot peut atteindre avec toutes les orientations possibles de l'effecteur (organe terminal). Cet espace de travail est un sous-ensemble de l'espace de travail maximal.

- Le positionnement absolu, correspondant à l'erreur entre un point souhaité (réel) – défini par une position et une orientation dans l'espace cartésien – et le point atteint et calculé *via* le modèle géométrique inverse du robot.

Cette erreur est due au modèle utilisé, à la quantification de la mesure de position, à la flexibilité du système mécanique. En général, l'erreur de positionnement absolu, également appelée précision, est de l'ordre de 1 *mm*.

- La répétabilité, ce paramètre caractérise la capacité que le robot a à retourner vers un point (position, orientation) donné. La répétabilité correspond à l'erreur maximum de positionnement sur un point prédéfini dans le cas de trajectoires répétitives. En général, la répétabilité est de l'ordre de 0,1 *mm*.

- La vitesse de déplacement (vitesse maximum en élongation maximum), accélération.

- La masse du robot.

- Le coût du robot.

- La maintenance, ...

I.6. Les générations de robot [5]

Des progressions s'opèrent dans tous les domaines :

- Mécanique,

- Micro-informatique,

- Energétique,

- Capteurs – actionneurs.

A l'heure actuelle, on peut distinguer 3 générations de robots :

1. Le robot est *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.

2. Le robot devient *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.

3. Le robot devient « *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.L'architecture des robots [6]

Les robots mobiles se composent de quatre parties essentielles:

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

I.7.1.La structure mécanique et la motricité

Il existe quatre types de structures mécaniques assurant la motricité:

1. Les robots à roues.
2. Les robots à chenilles.
3. Les robots marcheurs.
4. Les robots rampants.

I.7.1.1. Les robots mobiles à roues

La mobilité par roues est la structure mécanique la plus utilisée. Ce type de robot assure un déplacement avec une accélération et une vitesse rapide mais nécessite un sol relativement plat. On distingue plusieurs classes de robots à roues, déterminées principalement par la position et le nombre de roues utilisées.

Nous citerons ici les quatre classes principales de robots mobiles à roues : [4]

I.7.1.1.1. Un robot de type unicycle

Est actionné par deux roues indépendantes, il possède éventuellement des roues folles pour assurer sa stabilité. Son centre de rotation est situé sur l'axe reliant les deux roues motrices.

C'est un robot non-holonyme¹, en effet il est impossible de le déplacer dans une direction perpendiculaire aux roues de locomotion.

Sa commande peut être très simple, il est en effet assez facile de le déplacer d'un point à un autre par une suite de rotations simples et de lignes droites.

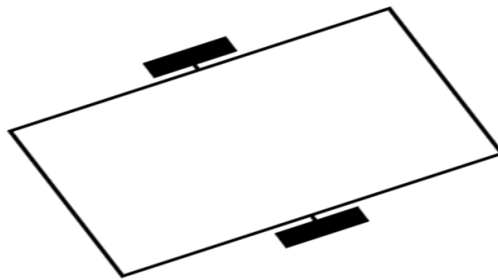


Fig. I. 4. Robot de type unicycle

I.7.1.1.2. Robot tricycle

Un robot de type tricycle est constitué de deux roues axes placées sur un même axe et d'une roue centrée orientable placée sur l'axe longitudinal.

Le mouvement du robot est donné par la vitesse des deux roues axes et par l'orientation de la roue orientable. Son centre de rotation est situé à l'intersection de l'axe contenant les roues axes et de l'axe de la roue orientable.

C'est un robot non-holonyme. En effet, il est impossible de le déplacer dans une direction perpendiculaire aux roues axes. Sa commande est plus compliquée. Il est en général impossible d'effectuer des rotations simples à cause d'un rayon de braquage limité de la roue orientable.

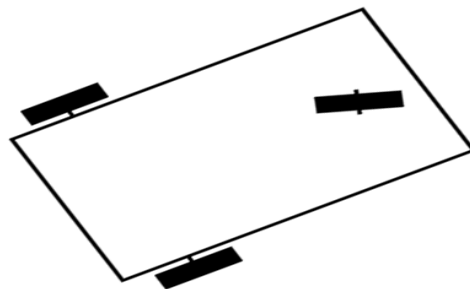


Fig. I. 5. Robot de type tricycle.

I.7.1.1.3. Robot voiture

Un robot de type voiture est semblable au tricycle, il est constitué de deux roues axes placées sur un même axe et de deux roues centrées orientables placées elles aussi sur un même axe.

Le robot de type voiture est cependant plus stable puisqu'il possède un point d'appui supplémentaire.

Toutes les autres propriétés du robot voiture sont identiques au robot tricycle, le deuxième pouvant être ramené au premier en remplaçant les deux roues avant par une seule placée au centre de l'axe, et ceci de manière à laisser le centre de rotation inchangé.

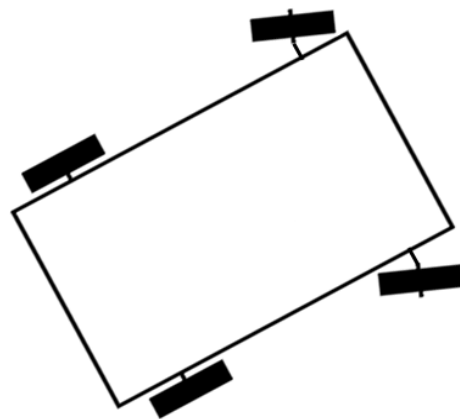


Fig. I. 6. Robot de type voiture

I.7.1.1.4. Robot omnidirectionnel

Un robot omnidirectionnel est un robot qui peut se déplacer librement dans toutes les directions. Il est en général constitué de trois roues décentrées orientables placées en triangle équilatéral.

L'énorme avantage du robot omnidirectionnel est qu'il est holonome puisqu'il peut se déplacer dans toutes les directions. Mais ceci se fait au dépend d'une complexité mécanique bien plus grande.

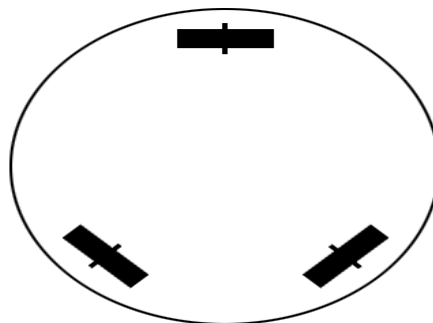


Fig. I. 7. Robot de type omnidirectionnel

I.7.1.1.5. Comparaison des différents types [7]

Nous pouvons observer dans le tableau ci-dessous un récapitulatif des avantages et des inconvénients des différents types de robots à roues.

Tableau. I. 1: comparaison entre les types des robots

Robot unicycle	+ stable + rotation sur soi-même + complexité mécanique faible
Robot unicycle	- peu stable - pas de rotation sur soi-même + complexité mécanique moderne
Robot voiture	+ stable - pas de rotation sur soi-même + complexité mécanique moderne
Robot omnidirectionnel	+ stable + rotation sur soi-même - complexité mécanique importante

I.7.1.2. Les robots mobiles à chenilles

Les robots mobiles à chenilles illustrés par la figure I.8, présentent l'avantage d'une bonne adhérence au sol et d'une faculté de franchissement d'obstacles. L'utilisation de ces robots est orientée vers l'emploi sur sol accidenté ou de mauvaise qualité au niveau de l'adhérence (présence de boue, herbe...).



Fig. I. 8. Robots mobiles à chenilles

I.7.1.3. Les robots mobiles marcheurs

Les robots mobiles marcheurs donnés à la figure I.9 sont destinés à réaliser des tâches variées dont l'accès à un site est difficile et dangereux pour l'être humain. Leur structure est dotée de plusieurs degrés de liberté ce qui permet un rapprochement avec les robots manipulateurs. On

distingue les robots marcheurs (figure à deux jambes (humanoïdes), à quatre pattes (type cheval), et à six pattes (type araignée)).



Fig. I. 9. Robots mobiles marcheurs.

I.7.1.4. Les robots mobiles 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. Ici aussi les techniques utilisées découlent des méthodes de locomotion des animaux et des insectes comme la montre la figure I.10.

Les applications de ce type de robots sont très spécialisées et les architectures des robots sont en général spécifiques à l'application visée [8].

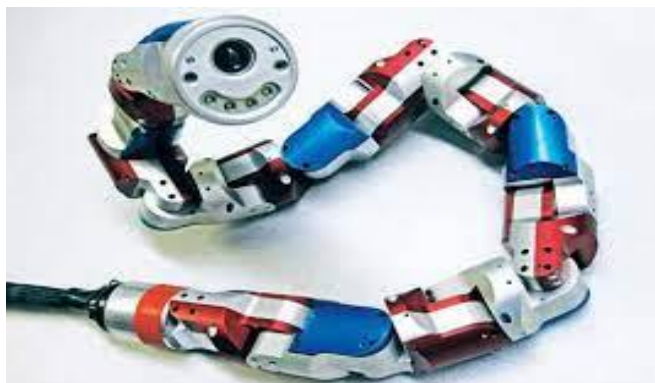


Fig. I. 10. Robot mobile rampant.

I.7.2. Le système de localisation [6]

I.7.2.1. La localisation relative

I.7.2.1. 1. L'odométrie

C'est la méthode de localisation relative la plus courante, elle nécessite une initialisation par rapport à l'environnement, des codeurs optiques montés sur les axes des moteurs permettant d'évaluer les rotations élémentaires des roues.

I.7.2.1.2. La localisation par capteurs inertiels

Les capteurs inertiels utilisés en robotique mobile, sont deux types

- ❖ les géomètres qui fournissent une mesure de la vitesse instantanée de la rotation
- ❖ les accéléromètres qui servent à mesurer l'accélération instantanée en translation ces systèmes inertiels sont simples mais coûteux

I.7.2.2. La localisation absolue

Qui exige une vision globale du milieu de travail comme (Global Positioning System) les caméras, les télémètres. Par exemple le télémètre à laser fait un balayage de 360° sur l'environnement et fournit les distances séparant le robot des différents objets

I.7.3. Organes de sécurité [6]

Il est dangereux de laisser le robot mobile complètement libre. Donc il est obligatoire qu'il soit doté d'organes garantissant la sécurité.

Deux types de capteurs sont employés:

- Les capteurs proximétriques assurent la détection avant collision

(Ultrasons, hyperfréquences, infrarouge...).

- Les capteurs à contact détectent une collision ou un choc avec l'environnement

(Contact électrique sur pare-chocs, résistance variable, fibre optique...).

L'organisation de la sécurité d'un robot mobile est représentée sur le schéma suivant:

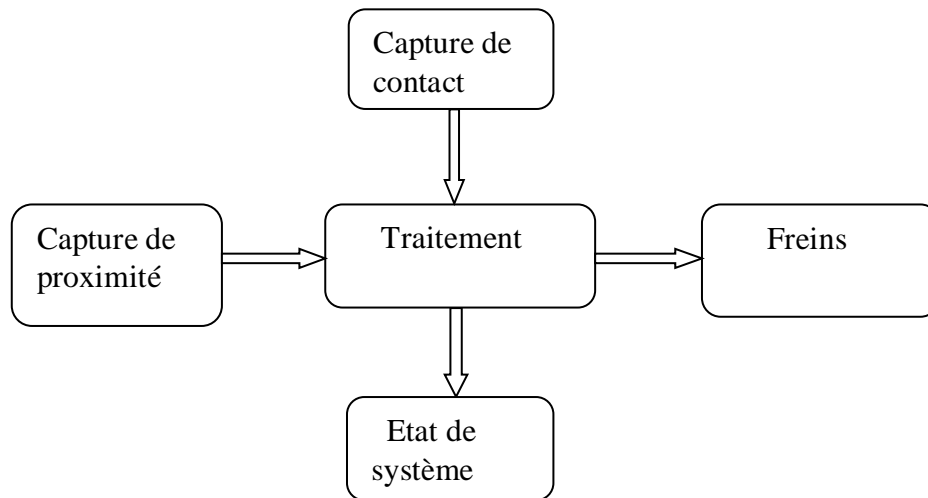


Fig. I. 11. Les organes de robot

Le comportement du robot mobile lors de la détection d'un obstacle s'effectue selon plusieurs cas:

- Si le capteur a contact est sollicité le robot s'immobilise soit définitivement soit tant que le contact persiste, ou il effectue un mouvement opposé au contact.
- Si on détecte une présence, la stratégie consiste soit à immobiliser le robot en attendant que la personne s'éloigne, soit à ralentir le mouvement si la personne n'est pas très proche, soit à choisir un autre chemin qui l'éloigne de la personne.



Fig. I. 12. Robot mobile

I.7.4. Le système de traitement des informations et gestion de tâches [10]

I.7.4.1. Structure d'un robot mobile

- ✓ **Capteur (Sensor)**- Il mesure la quantité physique et la convertit en un signal électrique qui peut être lu par un observateur ou par tout instrument électronique comme un convertisseur A-N. Un capteur stocke la quantité mesurée dans la mémoire.
- ✓ **Convertisseur A-N (A-D Converter)** - Un convertisseur analogique-numérique convertit le signal analogique envoyé par le capteur en un signal numérique.
- ✓ **Processeur et ASIC (Processors & ASIC)** - Les processeurs traitent les données pour mesurer la sortie et les stocker dans la mémoire.
- ✓ **Convertisseur D-A (D-A Converter)** - Un convertisseur numérique-analogique convertit les données numériques fournies par le processeur en données analogiques.
- ✓ **Actionneur (Actuator)**- Un actionneur compare la sortie fournie par le convertisseur N-A à la sortie réelle (attendue) qui y est stockée et stocke la sortie approuvée

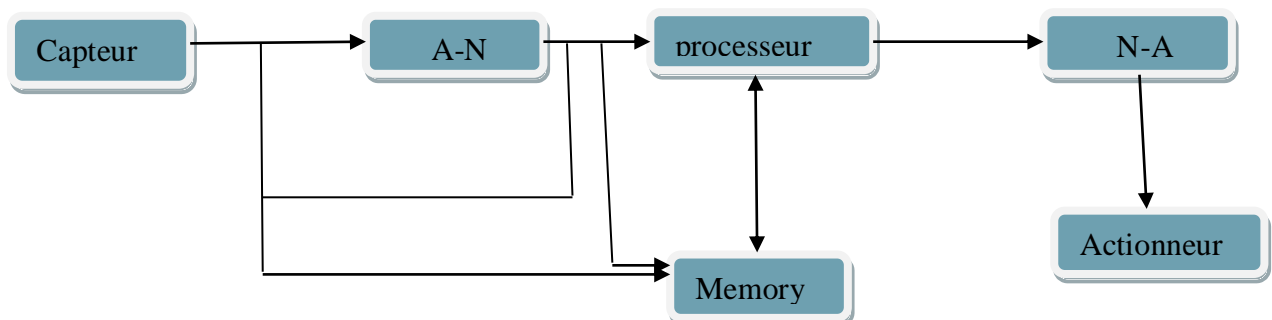


Fig. I. 13. Structure de base de robot

I.7.4.2 Architecture

I.7.4.2.1. Une unité de traitement ou microprocesseur

Des échanges informationnels en assurant:

- La lecture des informations d'entrée
- L'exécution des instructions du programme mis en mémoire
- La commande ou l'écriture des sorties

I.7.4.2.1.1. Type de processeur

Les processeurs peuvent appartenir aux catégories suivantes :

- Processeur à usage général (General Purpose Processor GPP).

- ✓ Microprocesseur (Microprocessor)
- ✓ Microcontrôleur (Microcontroller)
- ✓ Processeur intégré (Embedded Processor)
- ✓ Processeur de signal numérique (Digital Signal Processor)
- ✓ Processeur multimedia (Media Processor)
- Processeur système spécifique à l'application (Application Specific System Processor ASSP)
- Processeurs d'instructions spécifiques aux applications (Application Specific Instruction Processors ASIP)
- Cœur (s) GPP ou cœur (s) ASIP sur un circuit intégré à application spécifique (Application Specific Integrated Circuit ASIC) ou un circuit d'intégration à très grande échelle (Very Large Scale Integration VLSI).

I.7.4.2.1.2. La zone mémoire

-Type de mémoire

De nombreux types de périphériques de mémoire sont disponibles pour une utilisation dans les systèmes informatiques modernes. Il est très important de savoir les différences entre eux et comprendre comment utiliser efficacement chaque type.

Dans cette partie, nous aborderons ces appareils d'un point de vue logiciel. Le développement de ces appareils a pris plusieurs décennies et qu'il existe des différences physiques importantes dans le matériel sous-jacent. Les noms des types de mémoire reflètent souvent la nature historique du processus de développement et sont souvent plus déroutants que perspicaces.

La plupart des développeurs de logiciels considèrent la mémoire comme étant à accès aléatoire (RAM) ou en lecture seule (ROM). Mais, en fait, il existe des sous-types de chaque et même une troisième classe de mémoires hybrides.

Dans un dispositif RAM, les données stockées à chaque emplacement de mémoire peuvent être lues ou écrites, à volonté.

Dans un dispositif ROM, les données stockées à chaque emplacement de mémoire peuvent être lues à volonté, mais jamais écrites.

Dans certains cas, il est possible d'écraser les données dans un périphérique de type ROM.

De tels dispositifs sont appelés mémoires hybrides car ils présentent certaines des caractéristiques à la fois de la RAM et de la ROM.

1. Types de RAM

Il existe deux périphériques de mémoire importants dans la famille RAM: SRAM et DRAM. La principale différence entre eux est la durée de vie des données stockées.

- **La SRAM** (RAM statique) conserve son contenu tant que l'alimentation électrique est appliquée à la puce. Cependant, si l'alimentation est coupée ou perdue temporairement, son contenu sera perdu à jamais.

- **La DRAM** (RAM dynamique), a une durée de vie des données extrêmement courte, généralement inférieure à un quart de seconde. Cela est vrai même lorsque l'alimentation est appliquée en permanence.

Donc la SRAM a toutes les propriétés de la mémoire qu'un système embarqué a besoin. Comparé à cela, DRAM semble un peu inutile.

À quoi sert un périphérique de mémoire qui ne conserve son contenu que pendant une fraction de seconde? Une mémoire aussi volatile est en effet sans valeur.

Cependant, un simple élément matériel appelé contrôleur DRAM peut être utilisé pour faire en sorte que la DRAM se comporte davantage comme SRAM.

Le travail du contrôleur DRAM consiste à actualiser périodiquement les données stockées dans la DRAM. En actualisant les données plusieurs fois par seconde, le contrôleur DRAM maintient le contenu de la mémoire en vie aussi longtemps qu'il est nécessaire. Donc, DRAM est aussi utile que SRAM après tout. Lorsqu'il décide du type de RAM à utiliser, un concepteur de système doit tenir compte du temps d'accès et du coût. Les dispositifs SRAM offrent des temps d'accès extrêmement rapides (environ quatre fois plus rapides que la DRAM) mais sont beaucoup plus chers à produire. Généralement, la SRAM n'est utilisée que lorsque la vitesse d'accès est extrêmement importante.

Un coût par octet inférieur la DRAM attrayante chaque fois que de grandes quantités de RAM sont nécessaires.

De nombreux systèmes embarqués incluent les deux types: un petit bloc de SRAM (quelques centaines de kilo-octets) le long d'un chemin de données critique et un bloc beaucoup plus grand de DRAM (en méga-octets) pour tout le reste.

2. Types de ROM

Les mémoires de la famille ROM se distinguent par les méthodes utilisées pour leur écrire de nouvelles données (généralement appelées programmation) et par le nombre de fois qu'elles peuvent être réécrites.

Une caractéristique commune à tous ces appareils est leur capacité à conserver les données et les programmes pour toujours, même en cas de panne de courant.

a) Les toutes premières **ROM** étaient des dispositifs câblés contenant un ensemble préprogrammé de données ou d'instructions. Le contenu de la ROM devait être spécifié avant la production de la puce, afin que les données réelles puissent être utilisées pour organiser les transistors à l'intérieur de la puce! Les mémoires câblées sont toujours utilisées, bien qu'elles soient maintenant appelées "ROM masquées" pour les distinguer des autres types de ROM. Le principal avantage d'une ROM masquée est un faible coût de production.

b) Un pas en avant par rapport à la ROM masquée est la **programmable ROM** **PROM (Programmable ROM)**, qui est achetée dans un état non programmé.

Le contenu des données d'une PROM est entièrement constitué de 1. Le processus d'écriture de ces données dans la PROM implique un équipement spécial appelé un programmeur de périphérique. Le programmeur de l'appareil écrit les données sur l'appareil un mot à la fois, en appliquant une charge électrique aux broches d'entrée de la puce. Une fois qu'une PROM a été programmée de cette manière, son contenu ne peut jamais être modifié. Si le code ou les données stockées dans la PROM doivent être modifiés, l'appareil actuel doit être mis au rebut. En conséquence, les PROM sont également connus sous le nom de dispositifs programmables à usage unique (one-time programmable OTP).

c) **Une ROM effaçable et programmable EPROM** (erasable-and-programmable ROM) est programmée exactement de la même manière qu'une PROM.

Cependant, les EPROM peuvent être effacées et reprogrammées à plusieurs reprises. Pour effacer une EPROM, il suffit d'exposer l'appareil à une forte source de lumière ultraviolette. En faisant cela, une réinitialisation de la puce entière à son état initial non programmé. Bien que plus chères que les PROM, leur capacité à être reprogrammées fait des EPROM une partie essentielle du processus de développement et de test des logiciels.

3. Types hybrides

Au fur et à mesure que la technologie de la mémoire a mûri ces dernières années, la frontière entre les périphériques RAM et ROM s'est estompée. Il existe désormais plusieurs types de mémoire qui combinent les meilleures fonctionnalités des deux.

Ces périphériques n'appartiennent à aucun des groupes et peuvent être collectivement appelés périphériques de mémoire hybrides.

Les mémoires hybrides peuvent être lues et écrites à volonté, comme la RAM, mais conservent leur contenu sans alimentation électrique, tout comme la ROM.

Deux des périphériques hybrides, EEPROM et Flash, sont des descendants de périphériques ROM; le troisième, NVRAM, est une version modifiée de SRAM.

a) **Les EEPROM** sont effaçables et programmables électriquement (Electrically-Erasable-and-Programmable ROM).

En interne, elles sont similaires aux EPROM, mais l'opération d'effacement est effectuée électriquement plutôt que par exposition à la lumière ultraviolette. Tout octet dans une EEPROM peut être effacé et réécrit.

Une fois écrites, les nouvelles données resteront dans l'appareil pour toujours - ou du moins jusqu'à ce qu'elles soient effacées électriquement.

Le compromis pour cette fonctionnalité améliorée est principalement un coût plus élevé.

Les cycles d'écriture sont également beaucoup plus longs que les écritures dans une RAM.

Une EEPROM ne peut pas être utilisée pour la mémoire principale d'un système embarqué.

b) **La mémoire flash** est l'avancement le plus récent de la technologie de la mémoire. Elle combine toutes les meilleures fonctionnalités des périphériques de mémoire décrites jusqu'à présent. Les dispositifs de mémoire flash sont de haute densité, peu coûteux, non volatils, rapides (à lire, mais pas à écrire) être programmables électriquement. Ces avantages sont énormes et l'utilisation de la mémoire Flash a considérablement augmenté dans les systèmes embarqués en conséquence directe. D'un point de vue logiciel, les technologies Flash et EEPROM sont très similaires.

La principale différence est que les périphériques Flash ne peuvent être effacés qu'un secteur à la fois, et non octet par octet. Les tailles de secteur typiques sont comprises entre 256 octets et 16 kilo-octets. Malgré cet inconvénient, Flash est beaucoup plus populaire que l'EEPROM et remplace également rapidement de nombreux périphériques ROM.

c) **La RAM non volatile NVRAM (non-volatile RAM)**

La NVRAM est également une caractéristique de la mémoire ROM et des mémoires hybride s'évoquées précédemment.

Cependant, une NVRAM est physiquement très différente de ces appareils.

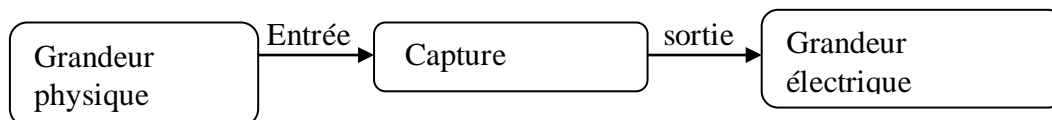
Une NVRAM est généralement juste une SRAM avec une batterie de secours.

À la mise sous tension, la NVRAM fonctionne comme n'importe quelle autre SRAM. Mais lorsque l'alimentation est coupée, la NVRAM tire juste assez d'énergie électrique de la batterie pour conserver son contenu actuel. La NVRAM est assez courante dans les systèmes embarqués. Cependant, il est très coûteux - encore plus cher que la SRAM – de sorte que ses applications sont généralement limitées au stockage de seulement quelques centaines d'octets d'informations critiques pour le système qui ne peuvent pas être mieux stockées.

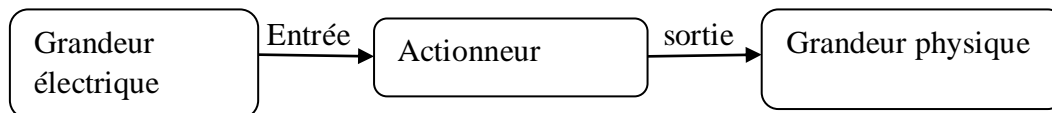
I.7.4.2.3. Les interfaces Entrées/Sorties

Généralement constitue par des capture et actionneurs.

➤ capture



➤ actionneurs



I.8. Mode de fonctionnement de robot

Il existe deux principaux mode de fonctionnement pour un robot mobile : télé-opéré et autonome.

-En mode télé-opéré, le terme robot (esclave) est amplement justifié, du fait que, une personne donne des ordres et pilote le robot à distance via une interface de commande (clavier, souris...), dans ce domaine les efforts de recherche sont basés sur l'amélioration de la perception de l'environnement pour l'opérateur, et les problèmes liés à la télécommunication.

-Mode autonome dont le robot doit prendre ses propres décisions ; de cette effet, il doit percevoir correctement son environnement, et agir en séquence suivant le niveau d'autonomie. C'est à lui de planifier son parcours et faire de sorte à éviter les obstacles afin d'atteindre un objectif et avec quel mouvement réaliser cela. Dans ce modes l'étude sa base

essentiellement sur la structure de contrôle, stratégies de commande, planification et la notion d'autonomie décisionnelle [11].

I.9. Classification des Robots Mobiles

La classification des robots mobiles se fait selon plusieurs critères :

- Degré d'autonomie ;
- Système de locomotion ;
- Énergie utilisée.

I.10. Quelques exemples de robots mobiles [12]

-Robot SHAKEY :

C'est l'un des premiers robots mobiles autonomes doté des capacités de perception, d'analyse et de planification et équipé de deux roues motrices. Ce robot est développé au Stanford Research Institute (USA, en 1967) et se déplace à la vitesse de 2m/heure.

-Robot HILARE : développé au laboratoire LAAS de Toulouse (France, en 1979) pour la planification de trajectoires d'un robot mobile ponctuel.

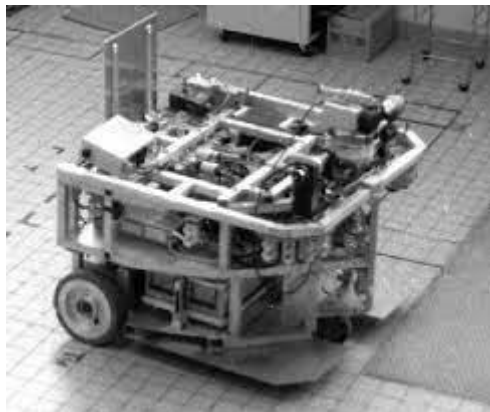


Fig. I. 14. Robot mobile Hilare

-Robot Pioneer : Conçu au 'Stanford Research Institute' (USA, en 1993) à un prix abordable possédant un ensemble de capteurs intéroceptifs et extéroceptifs qui servent à la planification de trajectoires, figure I.15.



Fig. I. 15. Robot mobile Pioneer P3-DX

I.11. Architectures des planificateurs de trajectoires

Parce que le robot mobile se déplace dans des environnements qui sont généralement encombrés d'obstacles statiques et dynamiques, alors cela nécessite la planification de sa trajectoire pour le mener à son but. Lorsqu'on peut avoir une information globale sur la scène sujette du mouvement du robot mobile, une planification globale est possible ; dans le cas contraire une planification locale (réactive) doit être réalisée [13].

I.11.1 Planification Globale [13]

On vise par cette architecture à reproduire le mode de raisonnement humain. La planification est donc décomposée en série d'opérations successives décrites par la figure I.16.

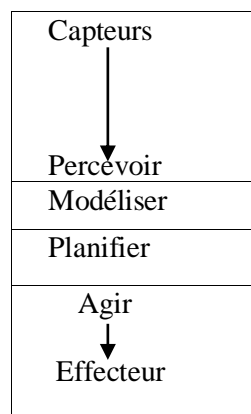


Fig. I. 16. Planification globale

La première opération consiste à traiter les données sensorielles qui fournissent au robot des informations sur son environnement. La deuxième étape consiste en la construction (ou la mise à jours) à partir des informations capteurs, du modèle de l'environnement tel que la position et le type des obstacles. L'étape suivante est la planification du chemin à suivre par le robot mobile pour mener à bien sa tâche. L'étape finale à pour but de calculer puis d'exécuter

les actions permettant au robot mobile de suivre le plan généré par l'étape précédente. Les étapes modélisation de l'environnement et planification de la trajectoire sont gourmandes en calcul et ceci notamment en milieu peu structuré. Alors le temps restant pour l'action devient plus petit.

On peut dire sur les architectures globales que la vitesse de réaction est très faible, donc on ne peut pas prendre en considération les obstacles dynamiques. Un autre inconvénient fait que l'opération de planification devient plus complexe lorsqu'on veut de la robustesse vis-à-vis des incertitudes de perception. L'utilisation de cette approche globale est limitée aux robots mobiles évoluant dans des environnements statiques et dont la structure est fortement contrainte et connue à priori. Les architectures délibératives possèdent un point fort important, qui est leur capacité à prendre en compte des raisonnements de haut niveau lors de la phase de planification. Il leur est possible en effet de gérer des missions complexes enchaînant plusieurs tâches et buts successifs.

I.11.2. Planification locale (réactive)

Dans le cas où on ne possède pas une vue globale de l'environnement ; parce que les obstacles et les buts à atteindre sont dynamiques; alors on passe à une planification locale réactive où le robot réagit continûment à son environnement. En partant de cette constatation on peut définir une architecture de commande basée sur la composition de plusieurs modules implémentant des comportements simples. Le comportement global du robot est alors le résultat de la composition des divers comportements élémentaires. Chacun des comportements, qui peuvent être par exemple l'évitement d'obstacle ou se diriger vers le but, réalise un couplage direct entre les capteurs et les effecteurs du robot ; ce qui permette une rapidité d'exécution. La trajectoire que le robot doit suivre ne peut être connue d'avance. Parmi les architectures comportementales celle proposée par Brooks [13].

I.11.2.1. Architecture de Brooks [13]

C'est l'architecture la plus connue qui a donné naissance à plusieurs travaux. La méthode de Brooks [13], donnée en figure I.17, s'appuie sur une décomposition verticale en niveaux de compétences. Chacun de ces niveaux correspond à un comportement indépendant recevant des données capteurs et agissant sur les effecteurs. Chaque module est capable d'inhiber ceux de niveau inférieur au sien ou de les utiliser pour son propre traitement. Cette architecture ne possède aucun arbitre central permettant de choisir l'un ou l'autre des comportements. Ce choix se fait par la réalisation des différents niveaux et le pré câblage des communications possible entre eux.

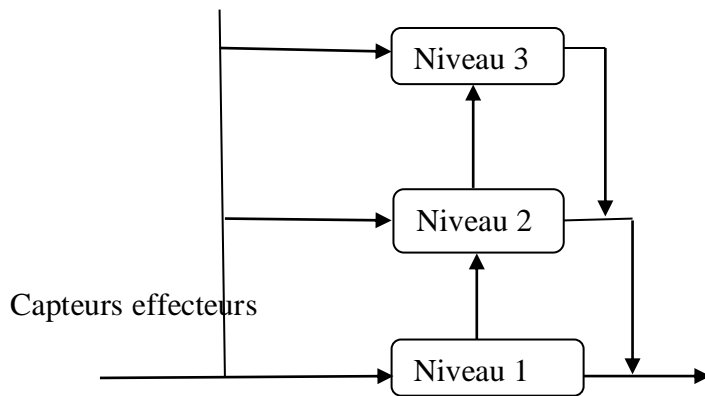


Fig. I. 17. Hiérarchie des niveaux dans l'architecture de Brooks.

Cette architecture possède les points forts suivants :

- La rapidité des réponses lui permet d'évoluer dans des environnements fortement dynamiques.
- Sa robustesse qui découle du parallélisme et de l'indépendance relative des comportements, permet au robot de continuer à fonctionner en cas de panne de l'un d'entre eux.
- Possibilité d'ajouter aisément des niveaux supérieurs.

L'architecture de Brooks possède bien l'inconvénient des architectures locales, qui est le manque du raisonnement haut niveau.

I.12.Applications

Le tableau ci-après résume de manière non exhaustive les diverses applications des robots mobiles.

Tableau. I. 2: Applications des robots mobiles

Industrie nucléaire	- surveillance de sites - manipulation de matériaux radio-actifs - dismantlement de centrals
Sécurité civile	- neutralisation d'activité terroriste - déminage - pose d'explosif - surveillance de munitions
Militaire	- surveillance, patrouille - pose d'explosifs - manipulation de munitions
Chimique	- surveillance de site - manipulation de matériaux toxiques

Médecine	- assistance d'urgence - aide aux handicapés physiques, aux aveugles
Sous-marine	- pose de câbles - recherche de nodules - recherche de navires immergés - inspection des fonds marins
Nettoyage	- coque de navire - nettoyage industriel
Espace	- exploration



Fig. I. 18. Différents types de robots mobiles selon le champ d'application

I.13. Avantages et inconvénients des robots [14]

Un système robotique consiste non seulement en des robots mais aussi d'autres dispositifs et systèmes qui sont utilisés avec le robot pour effectuer la tâche nécessaire.

Les avantages des robots sont :

- La robotique et l'automatisation peut dans de nombreuses situations accroître la productivité, la sécurité, l'efficacité, la qualité et la cohérence des produits.
- Les robots peuvent travailler dans un environnement dangereux, sans besoin de soutien de vie, ou de préoccupations concernant la sécurité.

- Les Robots n'ont pas besoin d'éclairage, de climatisation, de ventilation ou de protection contre le bruit.
- Les robots travaillent continuellement, sans ressentir une fatigue ou l'ennui, et ne nécessitent pas une assurance médicale ou de vacances.
- Les robots sont de précision répétable à tous les moments.
- Les robots peuvent être beaucoup plus précis que les êtres humains. Précision linéaire d'un robot typiquement est de 10 à 20 microns.
- Ils se perfectionnent sans cesse.

L'inconvénient des robots :

- Ils manquent de capacité de réagir en cas d'imprévu, à moins que les situations soient comprises et les réponses soient incluses dans le système. Les mesures de sécurité sont nécessaires pour s'assurer qu'ils ne blessent pas les opérateurs et n'endommagent pas les machines. On peut également citer d'autres inconvénients comme
- Réponse inadéquate.
- Le manque de prise de décision,
- Consommation de l'énergie.
- Ils peuvent causer des dommages à d'autres appareils, et la blessure à l'homme.
- Les robots sont coûteux en raison du coût initial de l'équipement, d'installation, le besoin de périphériques, le besoin de formation et la nécessité de la programmation (les robots ont besoin des humains).
- Peur de voir les robots voler nos emplois
- Les capacités des robots sont limitées

I.14. Conclusion

A travers ce chapitre nous avons présenté une introduction aux catégories principales des robots, ainsi que d'autres spécificités de la robotique. Nous avons ensuite abordé le domaine d'utilisation des robots, ainsi que leurs avantages et inconvénients. Nous avons pu constater l'évolution actuelle des robots proposés par divers constructeurs permettant de présenter une alternative aux machines-outils en utilisant une commande adéquate.

Chapitre 2 : Conception du robot mobile

II.1. Introduction:

La planification des actions et le contrôle des déplacements se situent dans trois référentiels différents.

Du plus abstrait au plus concret, nous les avons repérés : monde idéal, monde perçu et monde réel. Cette décomposition est classique pour les systèmes automatiques de commande.

L'élément nouveau, s'agissant des robots mobiles, est que le monde idéal a un champ très étendu, car se référant, éventuellement, non seulement à des représentations de type intégral-différentiel, mais aussi à des représentations et classifications de type sémantique (par exemple des notions de corridor, de pièces, etc.). Ainsi, le monde idéal dans lequel évolue le robot sera l'objet d'au moins trois modélisations, correspondant à des niveaux de planification différents. Il s'agit de la modélisation sémantique, de la modélisation topologique et de la modélisation géométrique [4].

Dans la vie moderne, on utilise pas mal d'outils et d'accessoires de commande à distance afin de simplifier notre contrôle, donc nous chercherons toujours à se concentrer sur la souplesse de la commande et de contrôler sur une zone bien définie le plus grand nombre possible d'accessoires.

Le système de commande occupe la première place d'objets qui ne nous quittent pas donc notre travail se concentre sur l'utilisation de ce dernier avec bien sûr sa liaison avec un système ou une carte de commande (carte d'interface) telle que l'Arduino, dans le thème le plus courant, la communication avec l'ordinateur se fait par un port USB, et dans ce chapitre nous étudions différentes matérielles de notre projet comme l'Arduino. Il existe plusieurs versions de l'Arduino (UNO, MEGA..) et L298N et Bluetooth HC-5....etc.

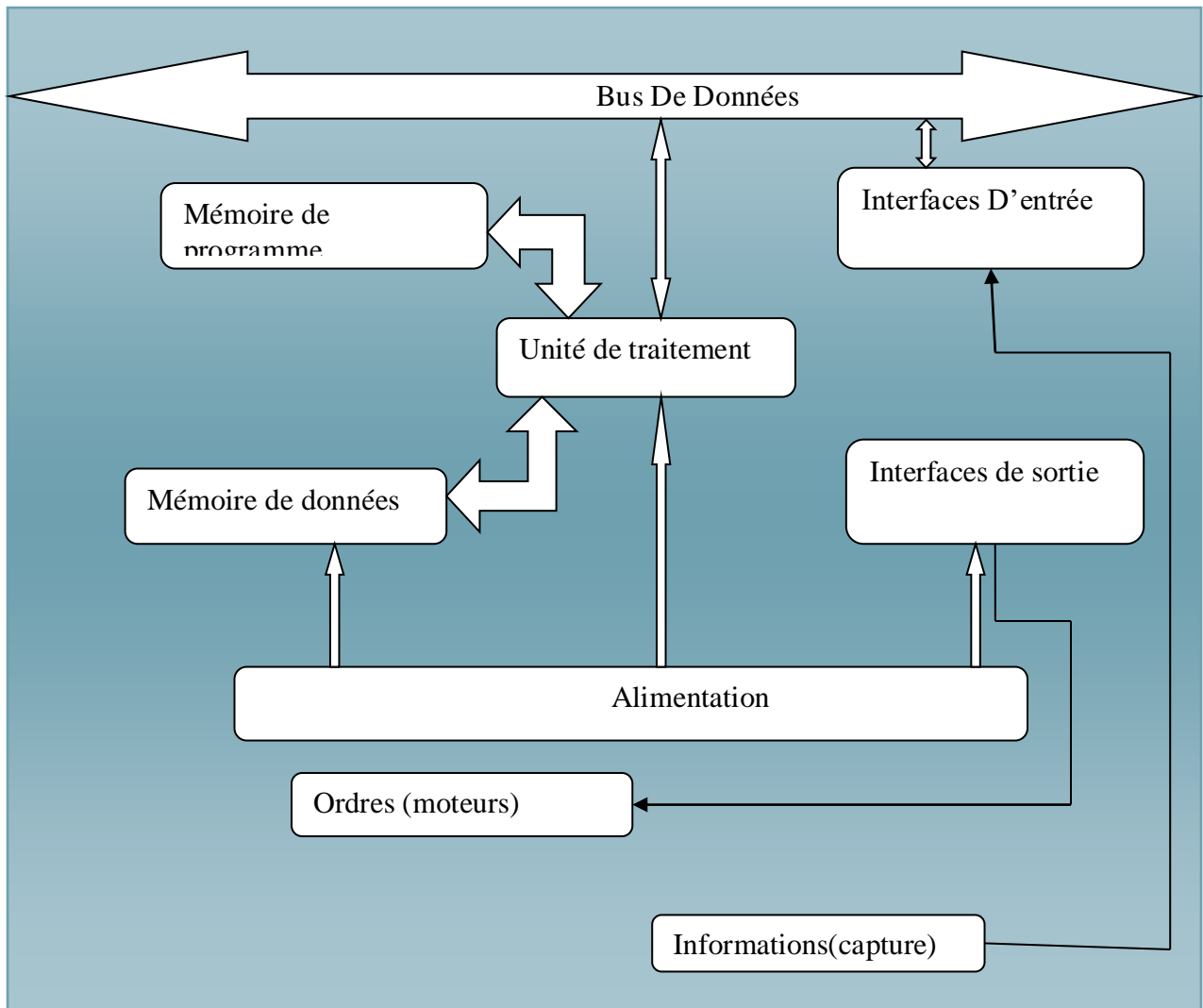


Fig.II. 1. La structure générale d'un robot mobile

II.2. Les composants de robot

On peut distinguer les différentes parties fonctionnelles de robot :

➤ Mécanique :

- Moteurs
- Roues

➤ Electronique :

- Unité de traitement
- interfaces de d'entrée
- interfaces de sortie

➤ Informatique :

- unité de traitement

II.3. Les aspects fondamentaux de notre projet

-L'aspect matériel composé essentiellement de :

- Carte Arduino Uno ;
- Les moteurs (MCC) ;
- Liaison Bluetooth (HC-05) ;
- de véhicule (châssis et roues) ;
- Source d'alimentation (batteries) ;

-L'aspect logiciel comprenant :

- Plateforme de programmation Arduino ;
- Utilisation de programme.

II.3.1. Châssis

Un châssis est une plaque sur laquelle sont posés tous les composants du robot et permettant de le faire tenir en équilibre.

Parmi les différents matériaux que l'on peut utiliser pour réaliser le châssis d'un robot, l'aluminium et plastique les plus appréciés. Cette matière présente de nombreux avantages, elle existe en plaque, en tube, carre ou rond, ou sous d'autres formes. L'aluminium et plastique léger, facile à travailler, à découper et à percer. De plus, il permet de réaliser un châssis très solide [sw4].



Fig.II. 2. Châssis

II.3.2.Roues

Pour mettre les roues en mouvement on a besoin d'un système de rotation qui est le moteur, Chaque roue sera entraînée par un moteur.



Fig.II. 3. Roues

II.3.3.Moteur

Il convertie l'énergie électrique absorbée en énergie mécanique, lorsqu'ils sont capables de fournir une puissance mécanique suffisante pour démarrer puis entraîner une charge mobile



Fig.II. 4.Moteur DC

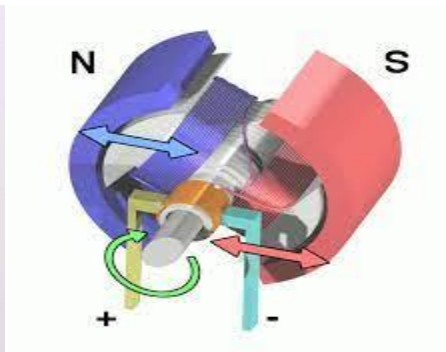


Fig.II. 5.Architecture interne de moteur DC

II.3.4.Arduino

II.3.4.1.Présentation Arduino [sw5]

Arduino est une plate-forme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique et d'un environnement de programmation. Sans tout ne connaître ni tout comprendre de l'électronique, cet environnement matériel et logiciel permet à l'utilisateur de formuler ses projets par l'expérimentation directe avec l'aide de nombreuses ressources disponibles en ligne.

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

Arduino est un projet en source ouverte, la communauté importante d'utilisateurs et de concepteurs permet à chacun de trouver les réponses à ces questions

- Ça sert à quoi?
- C'EST pour qui?
- Qu'est-ce que c'est?
- Arduino est-il un microcontrôleur ?
- Comment ça marche ?
- Faut-il des connaissances en électronique?

Arduino : ça sert à quoi ?

Arduino est un circuit imprimé open source. Vous connaissez déjà sûrement ce qualificatif pour des logiciels gratuits, où chacun peut prendre part au développement du projet et y apporter sa contribution. Ce mode de collaboration, réunissant beaucoup de personnes intéressées et engagées, possède un fort potentiel et fait clairement avancer les choses. Les résultats parlent d'eux-mêmes et n'ont rien à envier à ceux des projets commerciaux.

Sous ce nom Arduino se cachent non seulement du matériel mais aussi un logiciel. On parle alors de Physical Computing, qui désigne la construction de systèmes interactifs permettant de connecter le monde physique à celui des ordinateurs. Le monde dans lequel nous vivons est considéré comme un système analogique, alors que les ordinateurs agissent dans un environnement numérique ne connaissant que les états logiques O et I. C'est à nous, individus créatifs, qu'il appartient d'établir une liaison entre ces deux mondes et de montrer par des actions et des faits de quoi nous sommes capables.

Ce chapitre traite de deux thématiques fondamentales, dont nous ne pourrions-nous affranchir:

- L'électronique (composants et fonctions).
- Le microcontrôleur (la carte Arduino).

Toutes les expériences sont réalisées avec une tension d'alimentation de 5 ou 12v.

Arduino : c'est pour qui ?

Le projet Arduino est né en hiver 2005. Massimo Banzi enseigne dans une école de Design à Ivrea en Italie, et souvent ses étudiants se plaignent de ne pas avoir accès à des solutions bas prix pour accomplir leurs projets de robotique. Banzi en discute avec David Cuartielles, un ingénieur Espagnol spécialisé sur les microcontrôleurs...

Ils décident de créer leur propre carte en embarquant dans leur histoire un des étudiants de Banzi, David Mellis qui sera chargé de créer le langage de programmation allant avec la carte. En deux jours David écrira le code, Trois jours de plus et la carte était créé...Ils décidèrent de l'appeler Arduino (un bar fréquenté par les élèves à proximité de l'école) ...

Ça devient un hit tout de suite auprès des étudiants. Tout le monde arrive à en faire quelque chose très rapidement sans même avoir de connaissances particulières ni en électronique ni en informatique, réponse à des capteurs, faire clignoter des leds, contrôler des moteurs... Ils publient les schémas, investissent 3000 euros pour créer le premier lot de carte.

Les 50 premières partent directement à des élèves de l'école. En 2006 5000 cartes vendues...En 2007 plus de 30 000, en 2011 >120 000, sans compter les clones.

Arduino : qu'est-ce que c'est ?

La carte Arduino repose sur un circuit intégré (un mini-ordinateur appelé également microcontrôleur) associée à des entrées et sorties qui permettent à l'utilisateur de brancher différents types d'éléments externes. Les caractéristiques de la carte Arduino sont les suivantes :

- Une plate-forme de développement et de prototypage Open Source.
- Le rôle de la carte Arduino est de stocker un programme et de le faire fonctionner.
- Shields (cartes d'extension) avec des fonctions diverses qui s'enfichent sur la carte Arduino:
 - Relais, commande de moteurs, lecteur carte SD, ...
 - Ethernet, WIFI, GSM, GPS, ...
 - Afficheurs LCD, Écran TFT, ...
 - IDE (Environnement de Développement Intégré) multi OS :
 - Édition du programme
 - Compilation du programme

Transfert du programme dans la carte via le port USB

Arduino est-il un microcontrôleur?

La réponse est oui, sans aucun doute. Il possède bien tous les composants dont nous avons parlé et les réunit en son sein. Mais il cohabite aussi avec d'autres composants sur une carte compacte.

C'est quoi un microcontrôleur ?

Un microcontrôleur est un circuit intégré (ou IC, Integrated Circuit), qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit. Au temps des pionniers de l'électronique, on soudait un grand nombre de composants encombrants, tels que les transistors, les résistances ou les condensateurs, sur des cartes plus ou moins grandes. Aujourd'hui, tout peut loger dans un petit boîtier en plastique noir muni d'un certain nombre de broches. Ces dernières sont les connexions du circuit intégré au moyen des quelles s'effectue la communication. La figure II.7 montre un microcontrôleur ATmega328, qu'on trouve sur la carte Arduino. Avec ses dimensions réduites, il dispose pourtant d'une grande puissance de calcul. En fait, il suffit de le souder sur une carte et de le mettre sous tension pour pouvoir l'utiliser. Certes, il manque encore quelques composants (par exemple, des stabilisateurs de tension, des connexions pour la programmation, et d'autres. Mais il est cependant sous cette forme déjà (presque) prêt à l'emploi[15].



Fig.II. 6.Microcontrôleur ATmega328

-Arduino : comment ça marche ?

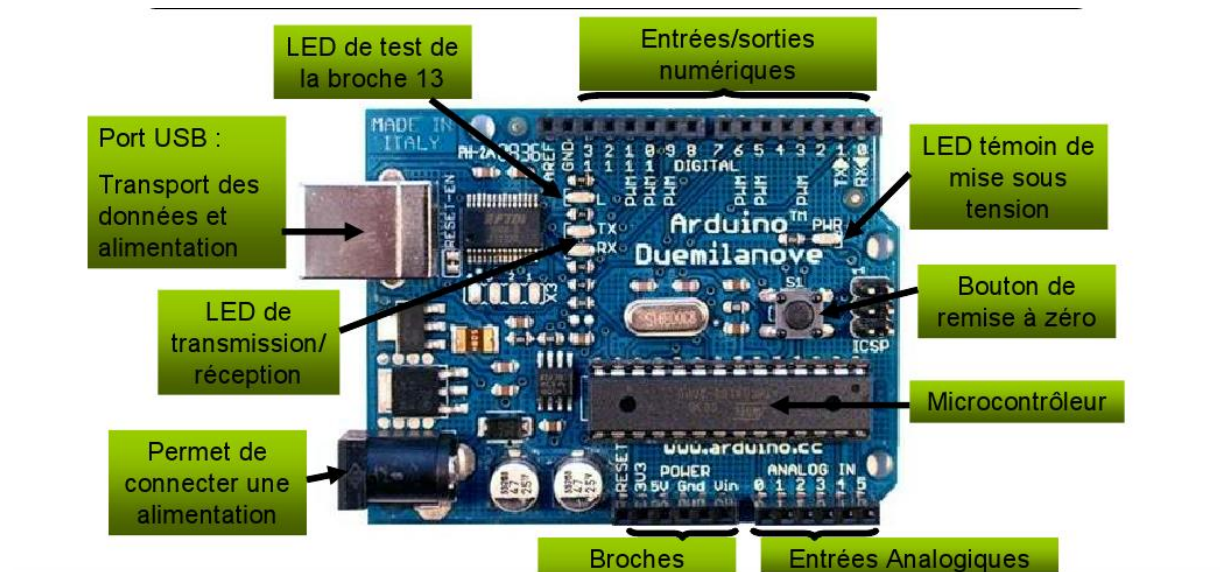


Fig.II. 7. Les éléments de la carte Arduino

Ces éléments sont les plus importants de la carte Arduino mais, bien entendu, cela ne veut pas dire que les autres sont à négliger.

-Arduino : faut-il des connaissances en électronique ?

- Pas ou peu si on utilise des cartes et des modules tout faits.
- La communauté francophone est très active sur le forum.
- Entraide, tutoriels, exemples de réalisations, ...
- Il faut des connaissances en électronique si on veut optimiser ou faire du sur-mesure.

II.3.4.2.Types de carte Arduino :

Il existe trois types des cartes :

- a. Les « officielles » qui sont fabriquées en Italie par le fabricant officiel (Smart Project).
- b. Les « compatibles » qui n'est pas fabriqués par (Smart Project), mais qui sont totalement compatibles avec les Arduino officielles.
- c. Les « autres » fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, ...).

Il existe beaucoup de cartes **Arduino** différentes, mais elles possèdent toutes des éléments en commun.

Pour notre application, nous avons utilisé Arduino Uno à microcontrôleur basée sur l'ATmega328. Elle possède 14 broches d'entrée / sortie numériques (dont 6 peuvent être utilisées en tant que sorties PWM), 6 entrées analogiques, un oscillateur à quartz de 16 MHz, une connexion USB, une prise d'alimentation, un en-tête ICSP et un bouton de réinitialisation, dont le tableau. II. 1 présente une synthèse des caractéristiques [16].

Tableau. II. 1:Caractéristiques de la carte Arduino Uno

Tension d'entrée (limites)	6-20V
Broches d'E / S numériques	14 (dont 6 fournissent la sortie PWM)
Broches d'entrée analogiques	6
Courant CC par broche I / O	40 mA
Courant DC pour Pin 3.3V	50 mA
Mémoire flash	32 Ko (ATmega328) dont 0,5 Ko utilisé par boot loader)
Mémoire SRAM	2 Ko (ATmega328)
Mémoire EEPROM	1 Ko (ATmega328)
Vitesse de l'horloge	16MHz
Longueur	68,6 mm
Largeur	53,4 mm
Poids	25 g

II.3.4.3.Alimentation de la carte Arduino

La carte **Arduino Uno** peut être alimentée via la connexion USB ou avec une alimentation externe. La source d'alimentation est automatiquement sélectionnée.

Une alimentation externe peut provenir soit d'un adaptateur AC-DC ou d'une batterie.

L'adaptateur peut être connecté en branchant une prise 2.1 mm dans la prise d'alimentation de la carte ou à partir d'une batterie connectée dans le pin (ou broche) GND et V-IN (alimentation externe). Le processeur peut fonctionner sur une alimentation externe de 6 à 20 volts.

Cependant, si la tension est inférieure à 7V, le pin 5V peut fournir moins de cinq volts et le processeur peut devenir instable. Si la tension est supérieure à 12V, le régulateur de tension peut surchauffer et endommager la carte. La plage recommandée est de 7 à 12 volts [16].

3.4.3.1. Les pins (ou broches) d'alimentation : sont les suivantes [16]

- ✚ **V-IN.** Tension d'entrée à la carte Arduino à l'aide d'une source d'alimentation externe (par opposition à 5 volts de la connexion USB ou une autre source d'alimentation régulée). Si l'alimentation en tension est faite par l'intermédiaire de la prise d'alimentation, on pourra y accéder via ce pin.
- ✚ **5V.** Cette pin délivre un 5V régulé par la carte. Le processeur peut être alimenté soit à partir de la prise d'alimentation DC (7-12V), le connecteur USB (5V), ou le pin V-IN de la carte (7-12V). La fourniture d'une tension via les 5V ou 3,3V contourne le régulateur, et peut endommager votre processeur. A déconseiller !
- ✚ **3.3 V** Une alimentation de 3,3 volts générée par le régulateur. La consommation de courant maximale est de 50 mA
- ✚ **GND.** Masse
- ✚ **IOREF.** Ce pin sur la carte Arduino fournit la référence de tension avec laquelle le microcontrôleur fonctionne.

Certaines broches (les pins) ont des fonctions spécialisées :

- ✚ **Série : 0 (RX) et 1 (TX).** Permet de recevoir (RX) et transmettre (TX) TTL données série. Ces pins sont connectés aux pins correspondants de l'USB-TTL puce Serial ATmega8U2.
- ✚ **Interruptions externes : 2 et 3.** Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, un front montant ou descendant ou un changement de valeur.
- ✚ **Impulsion PWM : Broches 5, 6, 9, 10,11et 12.** Fournissent une impulsion PWM 8bits avec la fonction analogWrite
- ✚ **LED : 13.** il est équipé d'un conduit par la broche numérique 13. LED Lorsque la broche est à la valeur HIGH, la LED est allumée, lorsque la broche est faible, il est hors tension.

II.3.4.4. Entrées et sorties

Chacune des 14 broches numériques sur la carte Uno peut être utilisée comme une entrée ou une sortie, en utilisant les fonctions (**pin Mode**), (**digital Write**), et (**digital Read**) fonctionnant à 5 V. Chaque broche peut fournir ou recevoir 20 mA en état de fonctionnement recommandée et a une résistance de pull-up interne (déconnecté par défaut) de 20-50k ohm. Un maximum de 40mA est la valeur qui ne doit pas être dépassée sur toutes les broches d'Entrée/Sorties pour éviter des dommages permanents au microcontrôleur [16]

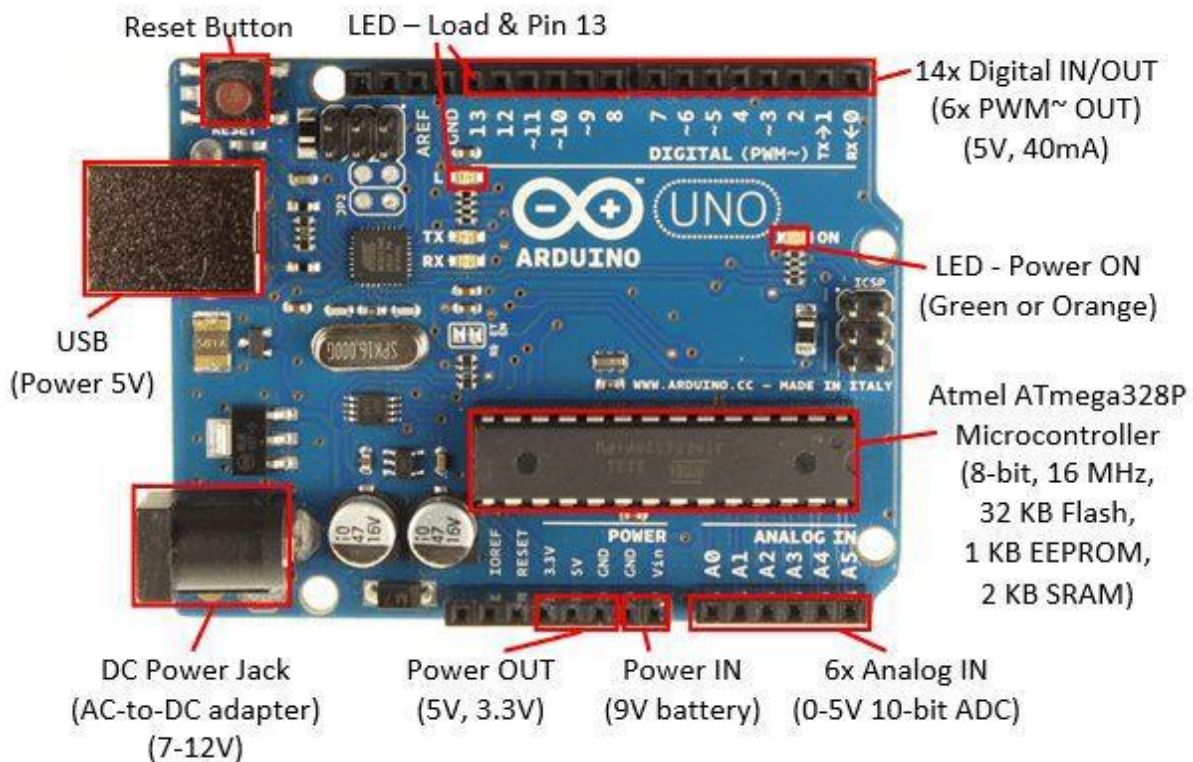


Fig.II. 8. Schéma des « ports »

II.3.5. La puce L298N [sw6]

C'est un circuit intégré de chez STMicroelectronics. Elle contient essentiellement :

- 2 ponts en H, permettant de piloter chacun 1 moteur électrique DC (dans un sens, ou dans l'autre)
- Et une logique de commande à « faible courant », pour piloter ces ponts à « fort courant »

En somme, le L298N est simplement un « double étage de puissance », permettant de commander directement deux moteurs électriques, via des commandes logiques « basse puissance ». C'est donc le genre de puce idéale pour s'interfacer avec un microcontrôleur, tel que l'ATmega328 équipant la plupart des Arduino. Ainsi, on peut aisément contrôler le sens et la vitesse de rotation de « gros » moteurs à courant continu, et ce, sans se forcer .

Le L298N requiert 2 alimentations distinctes pour fonctionner :

- Une tension pour la partie puissance (nommée Vs), qui servira à alimenter les moteurs, au travers de transistors de puissance

- Une tension pour la partie commande (nommée V_{ss}), qui servira à alimenter toute la partie logique de commande, dont ces transistors de puissance

Au niveau logique de pilotage, on distingue :

- Des broches d'activation des ponts (ENA et ENB), qui permettent de lancer ou arrêter les moteurs. À noter qu'on peut alimenter ces entrées en tout ou rien (alors les moteurs « tourneront » à fond), ou en PWM, pour contrôler leur vitesse de rotation.
- Des broches de sélection des ponts (IN1, IN2, IN3, et IN4), qui permettent de sélectionner les sens de rotation des moteurs (et comme il y a deux moteurs pilotables ici, on retrouve 4 entrées, correspondant aux 4 possibilités de sens de rotation, à raison de 2 par moteurs).

À noter que le L298N se trouve principalement sous 2 formats, dans le commerce :

- L298N nu (la puce électronique seule, à intégrer à ses montages)
- L298N « tout équipé », équipé de diodes de protection moteur, dissipateur thermique, borniers à visser, ... et bien d'autres choses encore.

Principe de fonctionnement du L298N (pilotage PWM)

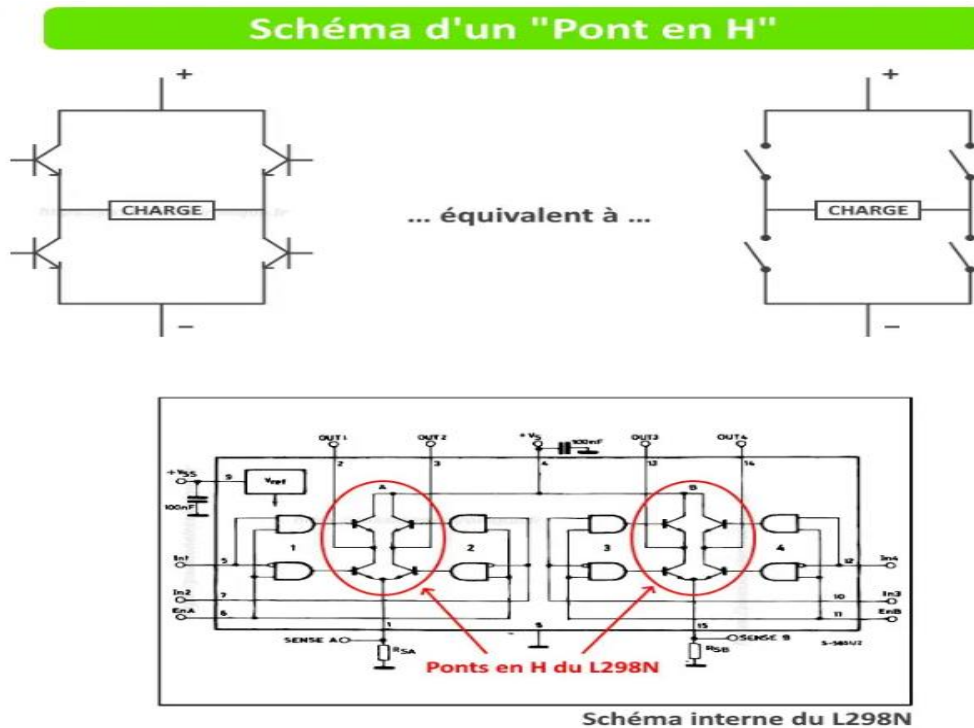
Pour comprendre le fonctionnement du L298N, il faut tout d'abord bien comprendre qu'il y a 2 aspects à considérer :

D'une part le fait qu'on peut faire tourner les moteurs dans un sens, ou dans l'autre (ce qui est permis par les « Ponts en H »)

Et d'autre part, le fait qu'on peut contrôler la vitesse de rotation d'un moteur, grâce à un signal PWM envoyé sur sa broche de commande

Ce sont ces 2 aspects qui font que le L298N peut à la fois contrôler le sens de rotation des moteurs, ET, faire varier leur vitesse. Nous allons donc les voir en détail, dès à présent.

Les Ponts en H ne sont en fait qu'un nom donné à une configuration particulière, de composants électroniques. Classiquement, ceux-ci sont composés de 4 transistors, disposés d'une certaine manière, formant visuellement la lettre « H » (ou presque !). En voici un exemple, avec quatre transistors bipolaires, comme ceux qu'on trouve en interne, dans le L298N :[sw6]



PassionElectronique.fr

Fig.II. 9. Schéma d'un pont en H

Le principe d'un pont en H est super simple : **en activant les transistors deux par deux (ceux de sens opposés), on peut contrôler le sens du passage du courant dans la charge, branchée « au milieu du H »**. Et c'est ce changement de sens de courant, qui permet aux moteurs à courant continu de pouvoir tourner « à l'endroit », ou « à l'envers ». Visuellement, voici comment cela se passé [sw6]:

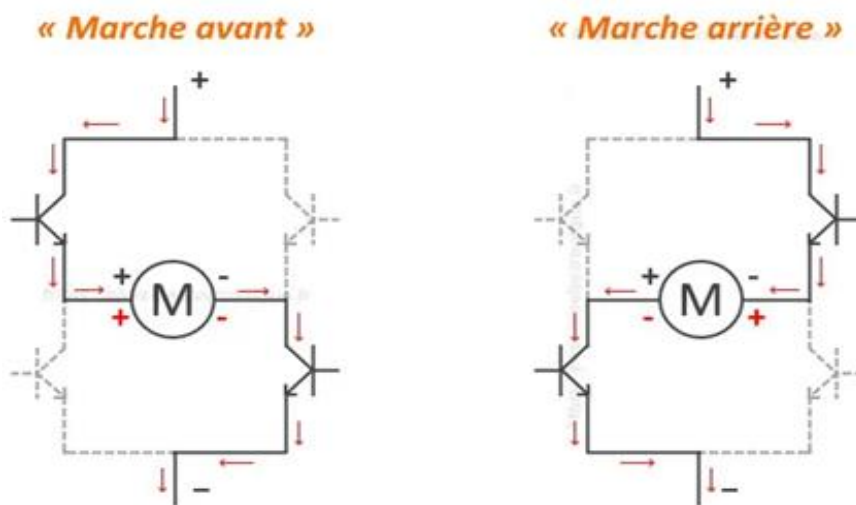


Fig.II. 10. Les deux configurations possibles de pont en H

Cette configuration à 4 transistors permet :

Dans le 1er cas : le « + » du moteur sur le « + » de l'alimentation, et le « - » du moteur sur le « - » de l'alimentation

Dans le 2ème cas : le « + » du moteur sur le « - » de l'alimentation, et le « - » du moteur sur le « + » de l'alimentation (faisant ainsi tourner le moteur en sens inverse)

En bref, en commandant simplement 2 des 4 transistors de manière symétrique, on peut facilement faire passer le courant dans un sens, ou bien dans l'autre. Et c'est d'ailleurs exactement le principe utilisé dans le L298N, pour faire changer le sens de rotation des moteurs électriques.

Remarque : le module L298N comporte 2 ponts en H, totalement indépendants. On peut utiliser un seul de ces ponts, si on le souhaite, tout comme les coupler ensemble, pour pouvoir délivrer deux fois plus de courant.

Module L298N (carte PCB toute assemblée) : comment le brancher à l'Arduino ?

Maintenant que nous avons vu la puce L298N en détail, il faut savoir que celle-ci est très souvent vendue « toute équipée », sous forme de mini plaquette électronique. C'est-à-dire, avec borniers, condensateurs de découplage, diodes de protection, et régulateur de tension embarqué. Voici à quoi cela ressemble, la plupart du temps :



Fig.II. 11. Module L298N (tout équipé)

Le détail de toutes les entrées/sorties de cette plaquette électronique :[sw7]

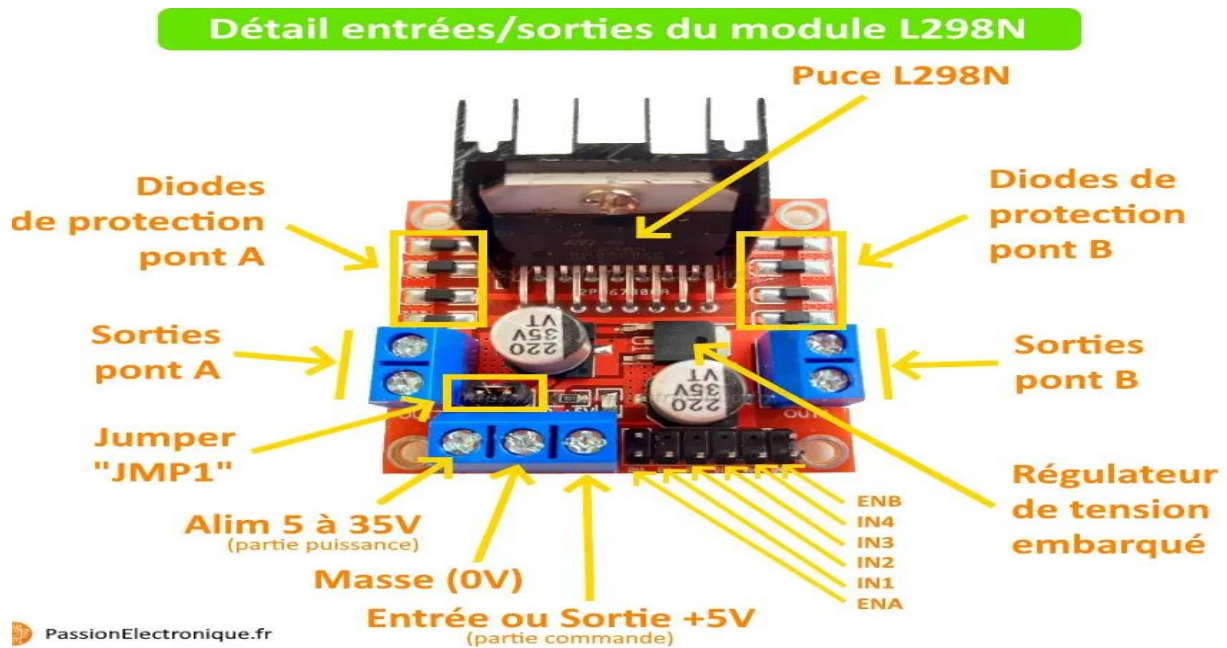


Fig.II. 12.Entrés /Sorties du module L298N

Borne « +12V » : il s'agit en fait de l'alim de puissance du L298N. Cette tension devra être comprise entre 5 et 35 volts, comme indiqué par la plupart des fabricants de ces modules, bien que le L298N en lui-même, supporte jusqu'à 46 volts. Toutefois, je vous recommande de ne jamais dépasser 35 volts, et même de rester bien en dessous, à 30V maxi par exemple. Car si vous jetez un œil sur la carte présentée en photo ci-dessus, vous verrez que le condensateur de filtrage de cette alim admet une tension maximale de 35V. Donc autant se prendre une petite marge de sécurité, afin d'éviter tout claquage de condensateur !

Borne « +5V » : il s'agit là d'une broche pouvant, selon les cas, soit être une entrée, soit une sortie.

En effet : Cette borne sera équivalente à une entrée, si le jumper d'alim (cavalier « JMP1 », visible ci-dessus) est retiré. Il faudra alors envoyer de 5 à 7 volts sur cette borne, afin de permettre l'alimentation de la partie « commande » du L298N.

Cette borne sera équivalente à une sortie fixe +5V, si le cavalier d'alim (jumper « JMP1 ») est mis en place. Celle-ci peut servir, dans certains cas, d'alimentation pour des appareils fonctionnant en 5 volts, branchés dessus (comme un Arduino, ou autre). Par contre, cette possibilité n'est valable que si la tension d'alimentation de puissance n'excède pas 12V.

Sinon, le régulateur de tension embarqué pourrait griller, du fait de la trop grande différence de potentiel, entre la tension qu'il reçoit, et la tension qu'il doit fournir.

Borne « GND » : c'est la masse commune (0 volt) aux deux alimentations vues précédemment (alim de puissance et alim de commande)

Bornes « OUT1 » et « OUT2 » : c'est ici que devra se brancher le 1er moteur

Bornes « OUT3 » et « OUT4 » : c'est ici que viendra se brancher le 2nd moteur, ou rien du tout, s'il n'y a qu'un seul moteur à piloter, via les sorties OUT1/OUT2

Bornes « ENA » et « ENB » : celles-ci commandent respectivement la mise sous tensions du 1er et 2ème moteur, via le pilotage respectif des ponts A et B (montages en H, internes au L298N)

Bornes « IN1 » et « IN2 » : ces entrées permettent de définir le sens de rotation du 1er moteur (selon table de vérité vue plus haut)

Bornes « IN3 » et « IN4 » : ces entrées permettent de définir le sens de rotation du 2ème moteur (voir table de vérité vue plus haut)

À noter que vous verrez également certains cavaliers (jumpers) présents sur cette plaquette électronique, pour court-circuiter (shunter) certaines choses. Ceci vous permettra de :

Faire le choix d'un « +5V » interne ou externe. Comme vu précédemment :

Si le cavalier JMP1 est présent : alors la borne +5V est une sortie fournissant 5 volts (dont vous pourrez vous servir pour alimenter d'autres choses avec, comme un Arduino ou autre)

Si le cavalier JMP1 est absent : alors la borne +5V est une entrée, où il faudra envoyer une tension continue, entre 5 et 7 volts (de préférence égale à +5V)

Faire le choix d'un « ENA » et « ENB » pour une vitesse fixe et maximale, ou d'une vitesse variable à souhait (via l'envoi d'un signal PWM sur ces broches) :

Si le cavalier sur ENA ou ENB est présent : alors chacun des moteurs sera soit totalement arrêté, soit tournera à vitesse maximale ce soit

II.3.6. Bluetooth HC-05 [sw7]

C'est un module Bluetooth SPP (protocole de port série) facile à utiliser, conçu pour la configuration d'une connexion série sans fil transparente. Sa communication s'effectue via une communication série, ce qui facilite l'interface avec le contrôleur ou le PC. Le module Bluetooth HC-05 fournit un mode de commutation entre les modes maître et esclave, ce qui signifie qu'il ne peut utiliser ni la réception ni la transmission de données.

Spécification:

- Modèle: HC-05
- Tension d'entrée: DC 5V
- Méthode de communication: communication série
- Les modes maîtres et esclave peuvent être changés

VCC + 5V Connecter à + 5V

GND Ground Connect to Ground.

Ce **module** communique via une liaison série avec une carte Arduino. Cette liaison s'établit sur deux broches RX et TX définies dans notre programme en tant que broches 11 et 10. La broche RX de la carte Arduino doit être raccordée à la broche TX du **module Bluetooth HC05**.

Le **module Bluetooth HC-05** permet de faire de la communication sans fil avec un ordinateur, smartphone ou un autre **module Bluetooth** Wifi et Bluetooth fournir une communication sans fil à l'aide de signaux radio. Cependant, la principale différence entre Bluetooth et Wifi réside dans l'objectif pour lequel ils sont conçus. En effet, Bluetooth est utilisé pour connecter des appareils à courte portée. Le Wifi offre cependant un accès Internet sans fil rapide. Avec Bluetooth, nous pouvons connecter notre appareil à un nombre limité d'autres appareils (Connexions Peer to Peer). Cependant, le Wi fi offre un accès Internet à plusieurs utilisateurs en même temps.

Si nous n'avons pas à nous soucier de la vitesse, Bluetooth est une excellente solution, car il y a très peu de bande passante nécessaire pour. Le Wifi offre plus de bande passante car la vitesse est un facteur important pour une bonne connexion Internet [sw8].



Fig.II. 13. Bluetooth HC-05

II.3.7. Alimentation

Pour assuré la fonctionnement de tous les composant, nous devant prendre en considération les conditions de chaque composant et fournir l'énergie nécessaire pour fonctionnement.

Les tensions d'alimentation des différents modules utilisée sont représentées dans le tableau suivant :

Tableau. II. 2:Alimentation des composants

Composants	Alimentation
Arduino UNO	5V/20mA
Moteur courant continu	5V/1A
L298N	5V/25mA
Bluetooth HC-05	3.3V



Fig.II. 14. Batteries

II.4. Les Logiciels utilisés

II.4.1. Logiciel l'Arduino [17]

Le logiciel va nous permettre de programmer la carte Arduino en langage C, il nous offre une multitude de fonctionnalités.

C'est un logiciel de programmation par code, qui contient une cinquantaine de commandes différentes. A l'ouverture, l'interface visuelle du logiciel ressemble à ceci :

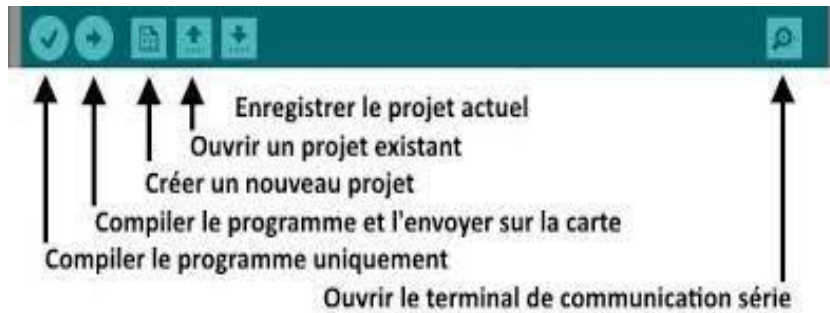


Fig.II. 15. Interface IDE Arduino

II.4.1.1. Les étapes de téléchargement du programme [17]

- Injecter un code vers la carte Arduino via le port USB.
- On conçoit ou on ouvre un programme existant avec le logiciel IDE Arduino.
- On vérifie ce programme avec le logiciel Arduino.
- Si des erreurs sont signalées, on modifie le programme.
- On charge le programme sur la carte.
- On câble le montage électronique.
- L'exécution du programme est automatique après quelques secondes.
- On alimente la carte soit par le port USB, soit par une source d'alimentation autonome. (Pile 9 volts par exemple).
- On vérifie que notre montage fonctionne.

II.4.2. ISIS PROTEUS [18]

La simulation en robotique mobile consiste tout d'abord à implémenter sur un système informatique, lui même embarqué sur un système mécanique, les méthodes théoriques résolvant les différentes fonctionnalités. Il s'agit ensuite de définir une mission que le robot doit exécuter, et évaluer notre commande.

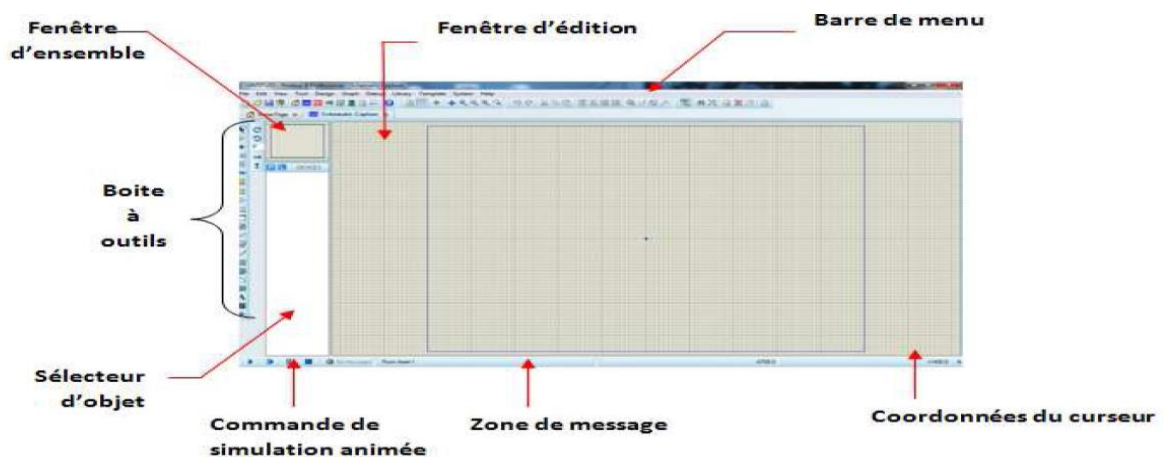


Fig.II. 16.Interface d'utilisateur du logiciel

II.5. Conclusion

Dans ce chapitre nous avons étudié les composants les plus connus et les plus utilisés actuellement en robotique mobile, ce que nous a permis de choisir tout ce qui est nécessaire pour réaliser notre robot mobile qui se compose de trois parties : mécanique, électronique et informatique.

Après avoir eu la liste de matériels et logiciels dont on a besoin pour le projet et comprendre leurs fonctionnements, il nous reste qu'à assembler ces composants de façon raisonnable pour réaliser un robot qui fonctionne selon nos besoins, c'est ce qu'on va voir dans le chapitre suivant.

Chapitre 3 : Réalisation du robot de Nettoyage

III.1. Introduction

Après avoir fait une étude théorique on passe maintenant à la réalisation pratique. En pratique il y a plusieurs défis à relever, on commence par choisir le bon châssis.

Dans le chapitre précédant on a établi une structure pour la conception de notre robot, maintenant il ne reste plus qu'à passer à la mise en oeuvre pratique. Pour cela il nous faut tout d'abord établir une méthode à suivre pour la réalisation matérielle et logicielle, par la suite nous procéderons aux tests ainsi qu'à leurs interprétation.

III.2. Architecture du robot mobile

Toute l'architecture de notre robot s'articule autour de la carte Arduino Uno qui define l'interfaçage entre les différents composant du robot. L'acquisition des données du monde extérieur étant assurées par l'application Bluetooth car RC .Ainsi que la prise de décision, la figure suivante représente les différentes interactions entre tous les constituants du robot :

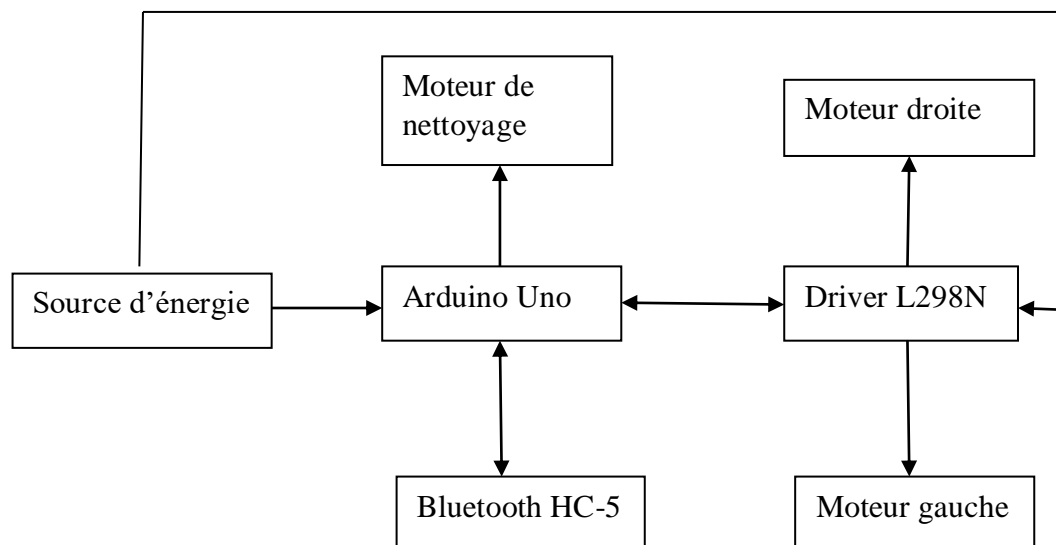


Fig. III. 1. Architecture de notre robot mobile.

III.3. Principe de fonctionnement

- La fonction du robot voiture est initialement gérée en mode infrarouge.
- Le Bluetooth peut également effectuer le même fonctionnement que le mode infrarouge.
- Le robot est mis en marche par l'utilisateur, il est guidé à distance.
- La structure commande robot est de type Maître- Esclav

Le Smartphone qui est le « maître », est équipé d'une interface logicielle de commande adaptée au processus à commander ;

Le robot qui est « l'esclave » exécute les ordres de pilotage donnés par le Smartphone.

Fonctionnement du robot peut être choisi à travers l'application Bluetooth à l'aide de quatre boutons intégrés.

Ses boutons vont activer un des modes de fonctionnement lorsque le Bluetooth est actif.

Bouton 'F' : marche en avant. Bouton 'B' : marche en arrière.

Bouton 'L' : marche à gauche. Bouton 'R' : marche à droite.

III.3.1. Algorithme de commande par Bluetooth

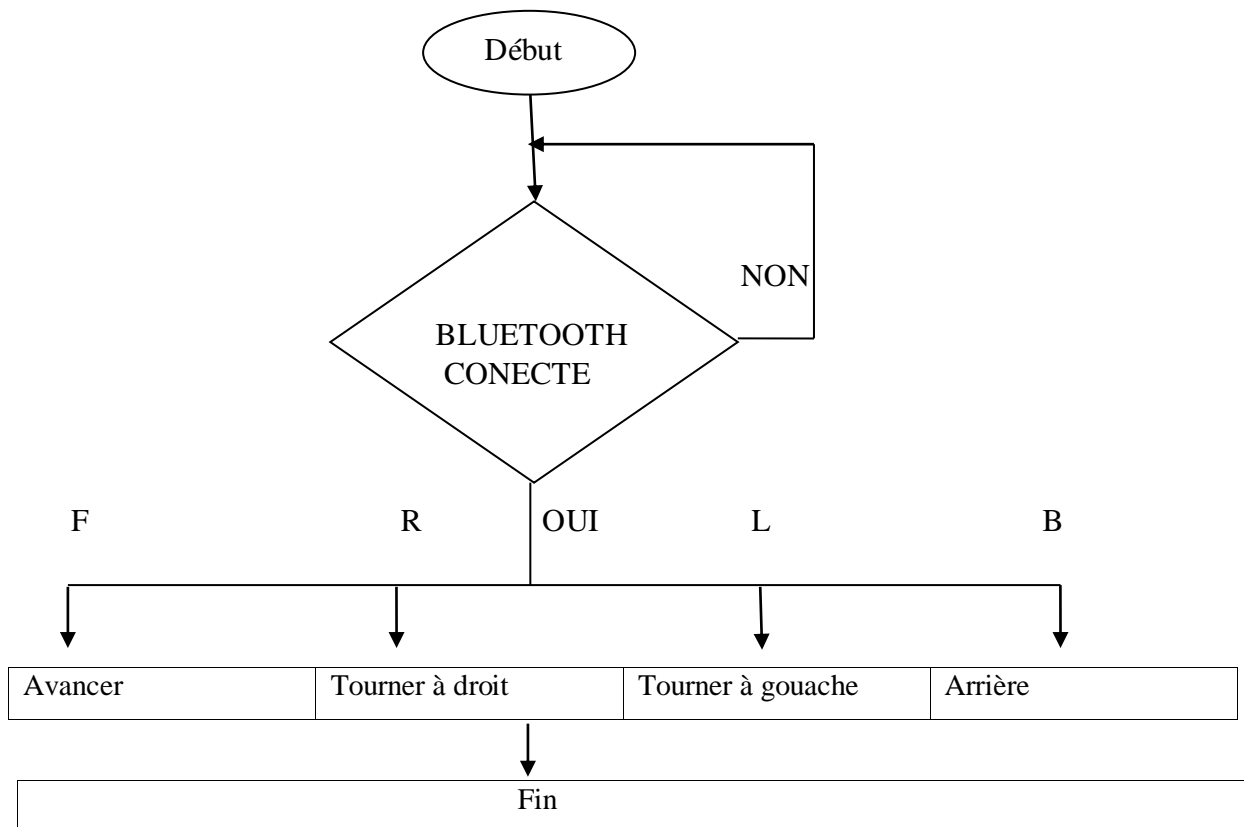


Fig. III. 2. Algorithme de téléguidage via Bluetooth.

III.3.2.Organigramme du robot :

L'organigramme de notre travail est présenté comme suit :

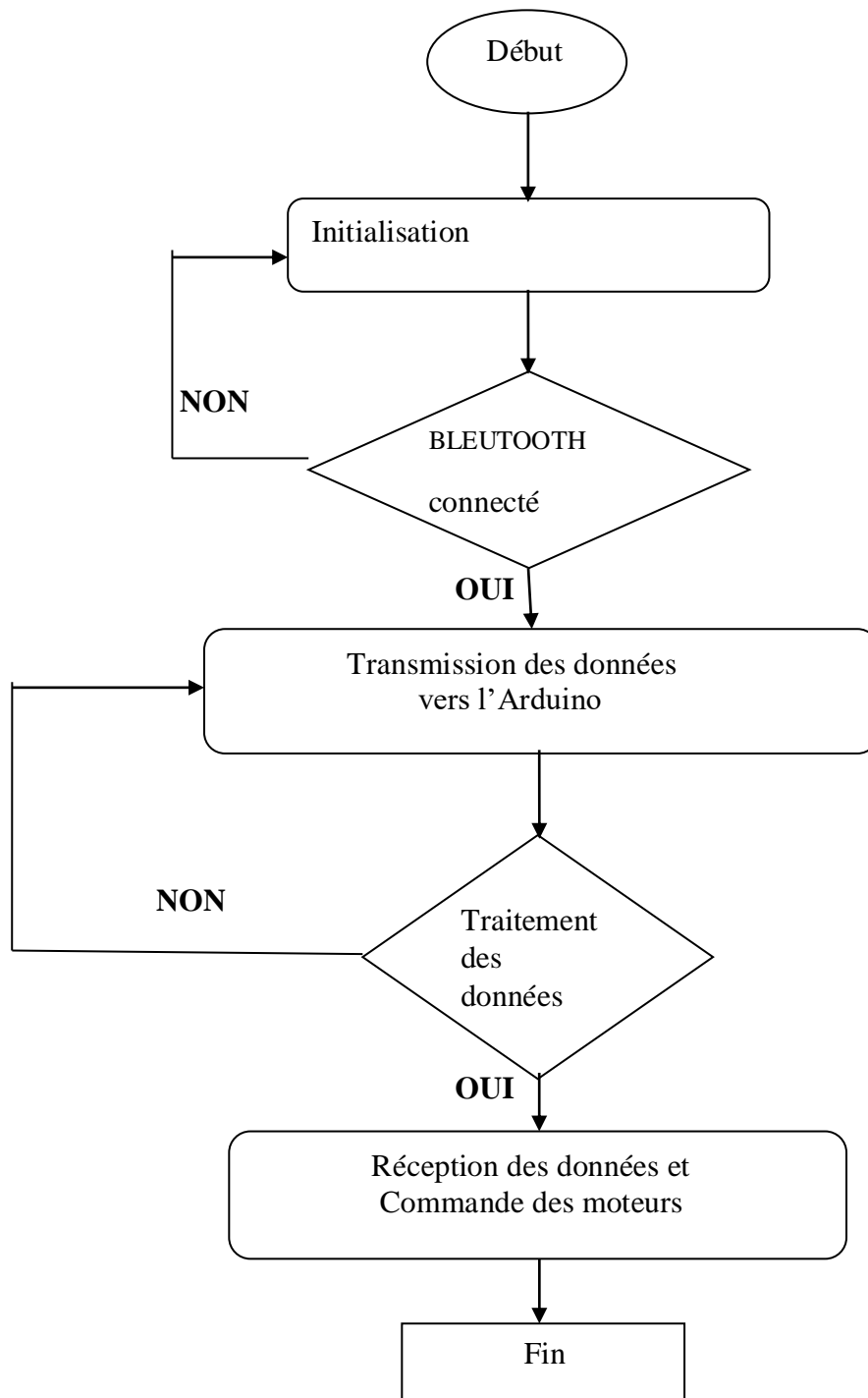


Fig. III. 3. Organigramme du robot.

III.4. Montage matériel et logiciel

Notre robot est constitué de :

4 x Moteur à engrenages (12v);

2x Moteur à engrenages (5v) ;

4 x Pneu ;

2x Ponge arrondi ;

1 x Pilote L298N ;

1 x Arduino Uno ;

1 x Télécommande (application android Bluetooth RC Contrôler) ;

1 x Adaptateur Bluetooth.



Fig. III. 4. Les composants de robot avant le brochage.

L'assemblage du robot se fait selon les étapes suivantes :

- Assembler le châssis qui permet de fixer les 4 moteurs, les 2 roues en avant et 2 roues en arrière par des vis et des écrous.
- Brochage L298N et Bluetooth HC-05 sur Arduino.
- Fixer les différents composants sur le châssis.
- Chargement du programme sur Arduino.

III.4.1. Brochage sur l'Arduino

Tableau. III. 1 Connexion des composants avec Arduino.

Composants	Broche	Connexion sur l'Arduino
L298n	ENA	5
	IN1	9
	IN2	10
	IN3	11
	IN4	12
	ENB	6
Bluetooth HC-05	RX	TX
	VSS	5V
	TX	RX
	GND	GND

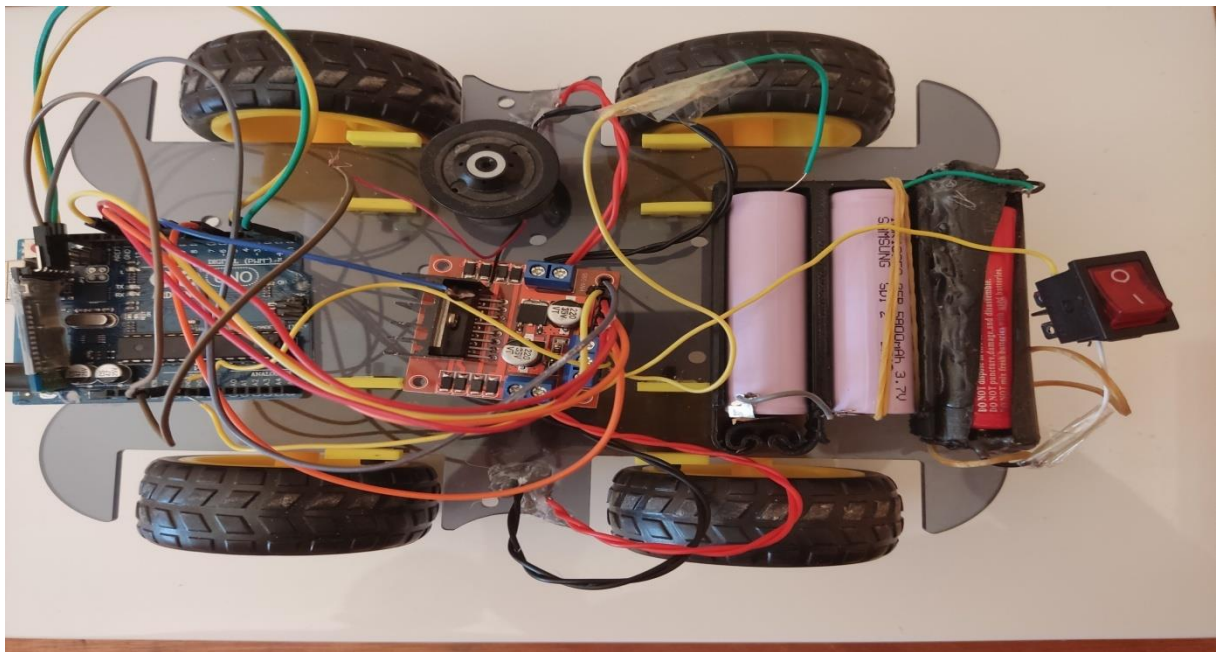


Fig. III. 5. Les composants après brochage.

III.4.2. Configuration des moteurs

Pour assurer le déplacement du robot, on a utilisé des moteurs .Ces dernier fonctionnent sous une tension de 12V.

Le circuit intégré L298N à été choisi pour commander les moteurs.

L298N qui permette de contrôle la direction du moteurs, la modification de directions des moteurs qui va imposer un changement de directions du mouvement du robot, le tableau suivant défini la configuration des défèrent moteur de notre projet.

Tableau. III. 2. Configuration des moteurs.

Moteur	Moteur gauche		Moteur droit	
	IN1	IN2	IN3	IN4
Movement	IN1	IN2	IN3	IN4
Marche en avant	LOW	HIGH	LOW	HIGH
Marche en arrière	HIGH	LOW	HIGH	LOW
Droit	LOW	HIGH	LOW	LOW
Gauche	LOW	LOW	HIGH	LOW
Stop	LOW	LOW	LOW	LOW

On a quatre modes de fonctionnement de moteurs ont été utilisés selon une logique indiquée. Ces modes ont été vérifiés et présentés sur la figure III. 6.

Il s'agit des modes : **Avancer - Arrière – Gauche - Droite**

Combinés, nous obtenons globalement :

–**Le mode Avancer/ Arrière**: Les deux moteurs tournent dans le sens des aiguilles d'une montre. En inversant la logique de commande ; les moteurs tourneront en sens contraire.

–**Le mode Gauche/Droite** : Les moteurs sont alimentés mais ne fonctionnent que d'un seul côté selon l'orientation que l'on veut donner

- **Variation de la vitesse du moteur à courant continu (MCC)**

Nous nous proposons de varier la vitesse des moteurs. Pour cela, nous utilisons les avantages de la carte Arduino permettant d'exploiter la technique PWM qui est présentée par des broches spécifiées 5, 6, 9, 10, 11 et 12 sur la carte ARDUINO et symbolisés par le symbole alternatif.

Sur le pont H, elle est utilisée sur les broches ENABLE (sur des drivers moteurs, elle est écrite PWM).

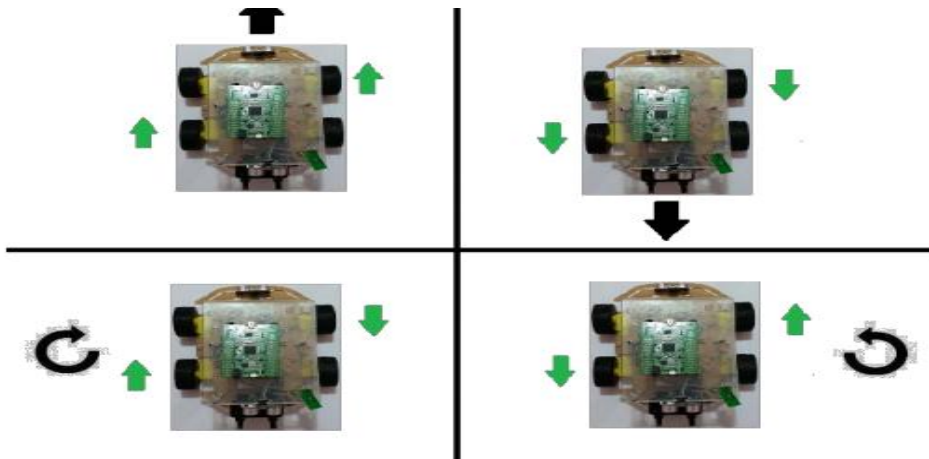


Fig. III. 6. Les différents sens de robot

III.5. Logiciels utilisés

III.5.1. Circuit du robot sous Proteus ISIS

Pour réaliser notre schéma électrique, on utilise Proteus Isis, ce logiciel qui permet de faire le test sur la commande de fonctionnement de moteur avant commencé la réalisation.

Avant de commencer la simulation, on doit ajouter au PROTEUS les bibliothèques Arduino, L298N (Driver Motor).

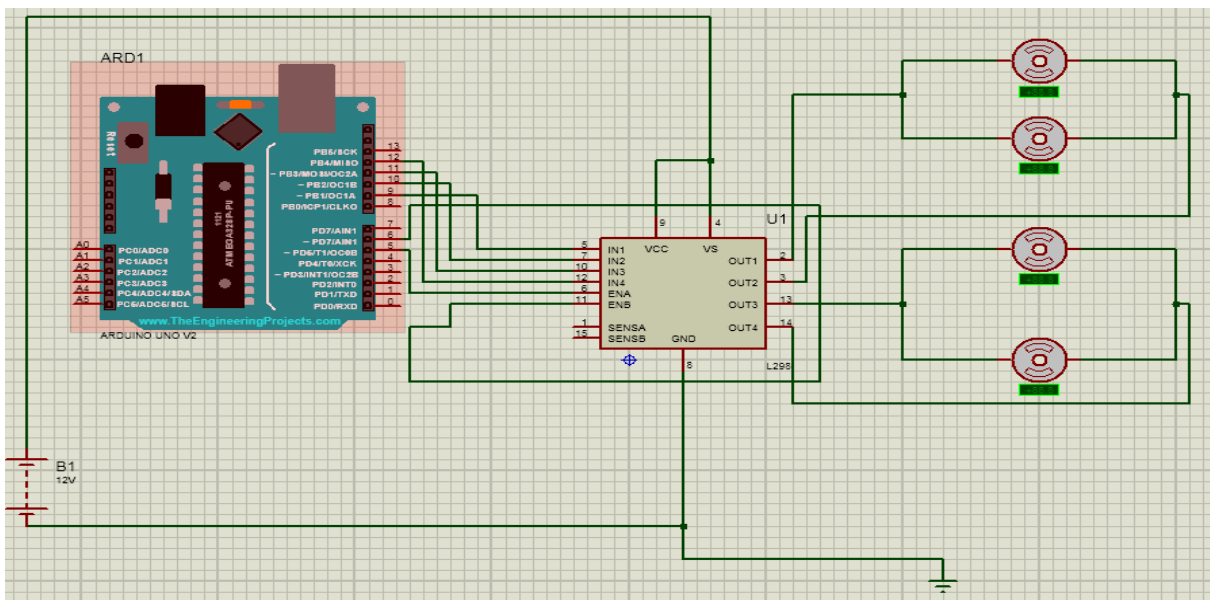


Fig. III. 7. Le Circuit de montage avec Isisproteuse.

Programme pour régler le fonctionnement des moteurs :

```
Int moteur LF = 9; Int moteurLR = 10;           //déclaration des ports utilisés

Int moteur RF = 11; Int moteurRR = 12; Int moteur D = 7; Int vitesse; Voidsetup () {

PinMode (moteur LF, OUTPUT);                   //mode des ports

PinMode (moteurLR, OUTPUT);

PinMode (moteurRF, OUTPUT);

PinMode (moteurRR, OUTPUT);

PinMode (moteurD, OUTPUT);

PinMode (5, OUTPUT);

PinMode (6, OUTPUT);

Serial.begin (9600);                           //vitesse de communication avec le pc

Vitesse=256;                                   //la vitesse de moteur

}

voidloop () {                                  // putyour main code here, to runrepeatedly:

digitalWrite (moteurLF,HIGH);

digitalWrite (moteurLR,HIGH);

digitalWrite (moteurRF,HIGH);

digitalWrite (moteurRR,HIGH);

digitalWrite(moteurD,HIGH);

analogWrite(5,vitesse);

analogWrite (6,vitesse);

}
```

III.5.2. Logiciel Fritzing

Ce logiciel permet de concevoir de façon entièrement graphique le circuit et d'en imprimer le typon.

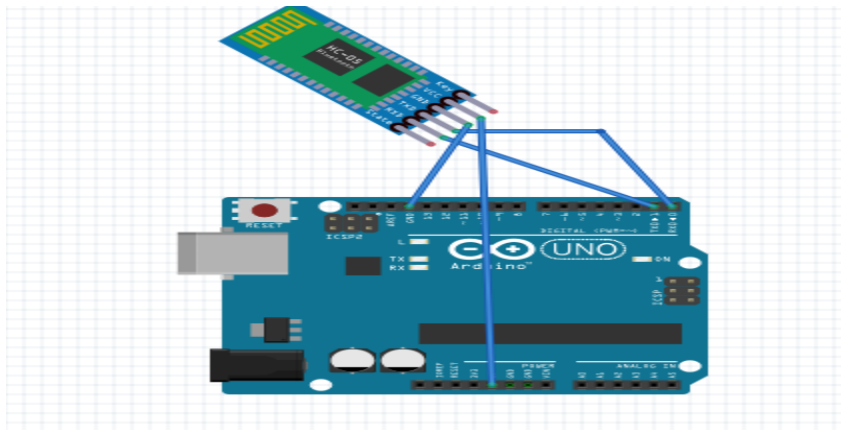


Fig. III. 8. brochage de hc-5 avec arduino

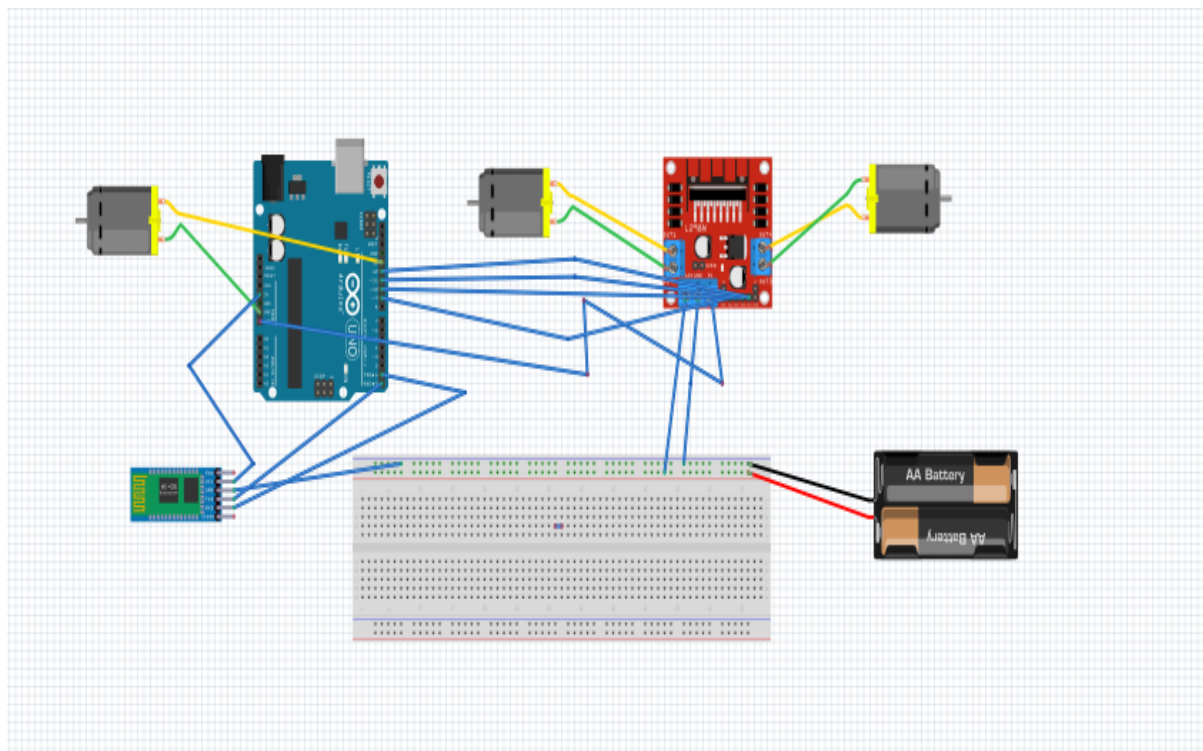


Fig. III. 9. Le montage final de notre projet par logiciel Fritzing

III.5.3. Logiciel de programmation Arduino

Ce logiciel permet de faire un programme sur la commande de moteurs et fonctionnement.



```
dd
vitesse=255;

}

void loop() {
  // put your main code here, to run repeatedly:
  char t;
  if(Serial.available()){
    t= Serial.read();
    Serial.println(t);
  }
  switch(t){
    case 'S' :
      digitalWrite(moteurLF, LOW);
      digitalWrite(moteurLR, LOW);
      digitalWrite(moteurRF, LOW);
      digitalWrite(moteurRR, LOW);
      digitalWrite(5, vitesse);
      digitalWrite(6, vitesse);
      break;
    case 'F' :
      digitalWrite(moteurLF, HIGH);
      digitalWrite(moteurLR, LOW);
      digitalWrite(moteurRF, HIGH);
      digitalWrite(moteurRR, LOW);
      digitalWrite(5, vitesse/2);
      digitalWrite(6, vitesse/2);
      break;
    case 'B' :

```

Compilation terminée.

Le croquis utilise 2404 octets (7%) de l'espace de stockage de programmes. Le maximum est de 32256 octets.
Les variables globales utilisent 190 octets (9%) de mémoire dynamique, ce qui laisse 1858 octets pour les variables locales. Le maximum est de

Fig. III. 10. Programmation sur Arduino.

-Chargement du programme

Cette étape consiste à téléverser le programme sur l'Arduino afin d'assurer son bon fonctionnement.

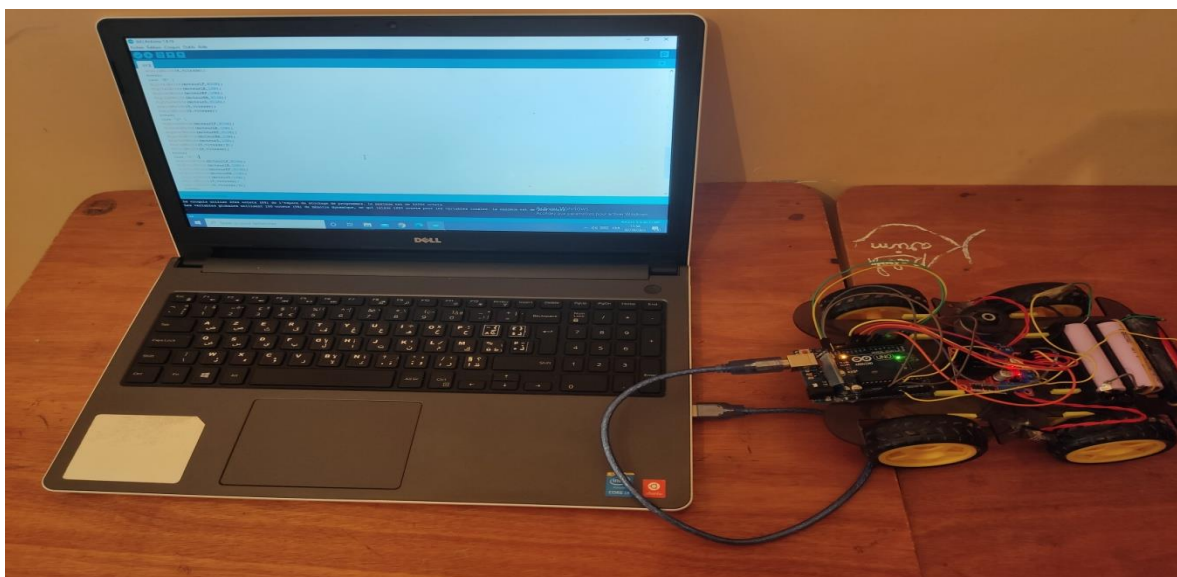


Fig. III. 11. Téléversement de programme vers la carte Arduino.

III.6. Télécommander notre robot mobile

Pour la communication avec le robot, il existe deux manières: soit filaire ou sans fil, mais il est choisi la sans-fil par rapport à la distance qui s'augmente et afin d'éviter le placement et le cote de fils[19].

La communication sans fil entre l'opérateur et le robot se fait par liaison radio fréquence, Il existe plusieurs systèmes qui fonctionnent par cette liaison, Bluetooth, wifi et xbee...etc[19].

Ils sont désormais des technologies courantes. Afin de commander notre robot à distance on a besoin d'utiliser une de ces technologies.

Pour notre travail, la commande sans fils, nous avons choisi d'utiliser le Bluetooth, et cela, en insérons le module hc-05 Arduino dans notre robot.

Une application android « Bluetooth RC Contrôler » doit être installée sur votre téléphone, qui permet automatiquement connecté le Bluetooth avec votre téléphone dont on pourra commander notre robot.

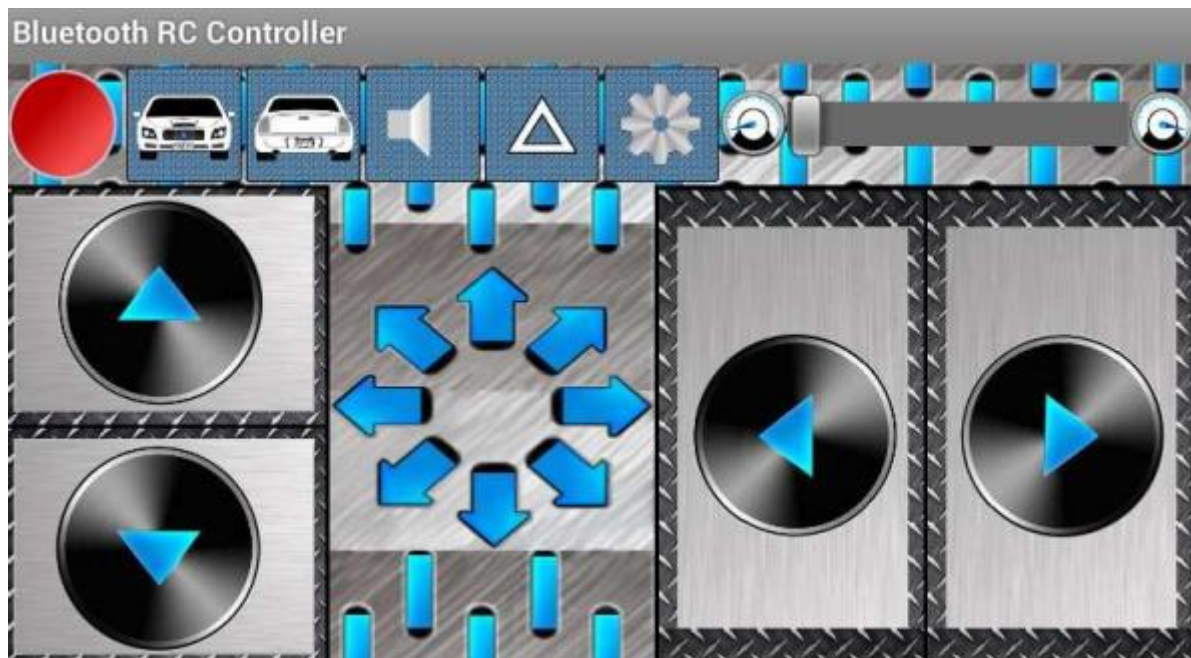


Fig. III. 12. L'application android Bluetooth RC Contrôler.

III.7. Conclusion

D'après les résultats obtenus dans ce dernier chapitre, on peut conclure que ce travail peut être divisé en deux grandes parties, la partie software qui est basée essentiellement sur la programmation en langage Arduino, et la partie hardware qui est la réalisation de la partie électronique et mécanique du robot.

Enfin, nous avons pu réaliser notre robot mobile avec sa télécommande (L'application android Bluetooth RC Contrôler) pour pouvoir communiquer avec lui à distance.

Conclusion Générale

Généralement l'image que l'on se donne d'un robot est celle d'un système mécanique articulé devant effectuer des opérations tels que le soudage, la peinture, la manutention dans des ateliers de l'industrie automobile ...etc.

Il en existe pour tant une autre, dont on parle le moins souvent : les robots mobiles à roues. Ces systèmes sont actuellement très répandus dans les applications industrielles, pourtant de nombreuses applications potentielles s'offrent à ce type de véhicules : applications de nettoyage, l'aide à la mobilité de personnes âgées ou handicapées, ...etc.

Un robot mobile est une machine qui a besoin pour sa mise en œuvre des spécialistes dans de nombreux domaines dont la mécanique, l'automatique, l'électronique et l'informatique. Ce projet, nous avons pu concevoir et développer un robot nettoyeur mobile, mettant en application nos connaissances acquises durant nos études et recherches, ainsi, il nous a permis d'améliorer nos connaissances en robotique et de maîtriser la nouvelle technologie de télécommande à distance sans fil par bluetooth et de l'intégrer dans notre application informatique.

En revanche, l'utilisation de la carte Arduino programmable en langage C++ nous a permis de gagner en temps et en simplicité du programme, nous avons choisi notre matériel en fonction de nos besoins puis essayé de différentes solutions afin d'arriver aux fonctionnalités souhaitées la réalisation de ce robot, nous a également donné l'opportunité de travailler sur plusieurs domaines à la fois, notamment la programmation et l'électronique, nous avons donc au final pu mener un vrai travail de conception de système, ce qui s'apparente le plus à un travail d'ingénieur.

De manière générale, ce projet nous a fait découvrir le monde de la robotique et toutes ses perspectives professionnelles.

Par ailleurs, nous avons pu acquérir des connaissances théoriques et pratiques pendant nos recherches scientifiques, particulièrement atteindre notre objectif qui consiste à réaliser un robot nettoyeur mobile à quatre roues, qui est un prototype qui peut être bien évidemment amélioré en divers points à l'aide de nouveaux composants afin d'augmenter ses performances.

Donc nous avons pu réaliser l'objectif de notre projet qui est l'utilisation de logiciel et carte électronique arduino pour manipuler un robot mobile à quatre roues.

Nous avons étudié tout d'abord les principales caractéristiques d'un robot et en particulier le robot mobile. On a ensuite essayé de parler des différents composants utilisés pour construire se robot a savoir le moteurs vers les capteurs jusqu' arduino et microcontrôleurs pour mieux comprendre son fonctionnement et pouvoir ensuite générer les signaux de commandes qui seront envoyés a partir de la carte électronique (commande a distance).

Ce projet pourra plutôt servir de base pour commander d'autres systèmes ou d'autres robots qu'ils ont des différents taches et applications selon les besoins à désigner.

Références bibliographiques

- [1] K. Oukherfallah & I.Fatiha, "Modélisation et conception d'une pince à deux doigts et à deux degrés de liberté d'un bras manipulateur. Diss". Université Mouloud Mammeri, 2015.
- [2] N. Robe, " l'International Federation of World robotics", France, Editions 2006 .
- [3] A. MMicipsa , B Mendil, RDerradji. "Exploration des facultés de reconnaissance du robot NAO. Diss", Université Abderahmane MIRA de Bejaia, Algeria. 2020.
- [4] B. Bernard, "Robotique mobile", Ecole Nationale Supérieure de Physique de Strasbourg", Université Louis Pasteur 2007.
- [5] J. L BOIMOND, "Cours de robotique. *ISTIA, Université Angers*". France, vol. 14, p. 15-16.
- [6] J. Laouedj & A. djeriani, "commande vocale d'un robot via bluetooth à base d'arduino" mémoire de master université Mouloud Mammeri, Tizi-ouzou, Algérie, 2018.
- [7] S. Lense "locomotion d'un robot mobile", mémoire présenté en vue de l'obtention du grade d'ingénieur Civil Informaticien "université de Liège faculté des sciences Appliquées Institut Montefiore. France. mai 2008.
- [8] A. PRUSKI , " Robotique générale " Edition Ellipse 1988.
- [9] N. SLIMANE , "Généralité sur la robotique mobile", Université BATNA, Algérie, 2008.
- [10] S. MADI, "cours électronique de système embarqué" Université Bouira, Algérie, 2020.
- [11] D. FILLIAT " Robotique Mobile ", Ecole Nationale Supérieure de Techniques avancées, Paris Tech, 2013.
- [12] T. Guesbaya, "Identification et commande de robot manipulateur rigide et flexible floue", Thèse de doctorat d'état, université Mohamad Khider, Biskra, Algérie, 2012.
- [13] R. Brooks, "A robust layered control system for a mobile robot", IEEE journal of robotics and automation 1986.
- [14] A. Kuttan , "Robotics ", USA , 2004
- [15] B. Erik. " Le grand livre d'Arduino ". Editions Eyrolles, 2015.

- [16] B.CottenceauProfessor, " Carte ArduinoUno microcontrôleur ATMega328 ", Université d'Angers France. 2017.
- [17] A.Krama,"Etude et réalisation d'une carte de contrôle par Arduino via le système Androïde", Texas A&M University at Qatar.2015
- [18]N.Lamara. "Réalisation d'une commande d'un robot mobil",Thèse de doctorat. Université Mouloud Mammeri.,algerie.2016.
- [19]M.Ayan, et al, "Android application based bluetooth controlled robotic car." *Intell. Inf.Syst* .2017

Références Webiographie

- [SW1] <https://www.automation-sense.com/quiz-test-automatisme-et-electricite/qcm-automatisme.html>
- [SW2] <https://www.clicours.com/generalites-sur-la-robotique-mobile/>
- [SW3] <http://www.jonathan-besson.fr/2018/la-programmation-robotique-aux-enfants-codinggouter/>
- [SW4] <https://www.frwiki.org/wiki/Chassis>
- [SW5] [https://arduino.technologiescollege.fr/IMG/PDF/cahier_\(\)_initialisation](https://arduino.technologiescollege.fr/IMG/PDF/cahier_()_initialisation)
- [SW6] <https://passionelectronique.fr/tutoriel-1298n/>
- [SW7] <https://www.cours-gratuit.com/cours-arduino/arduino-bluetooth-module-hc-05-pdf-eng>
- [SW8] <https://fr.itpedia.nl/2018/07/12/wifi-en-bluetooth-wat-is-het-verschil/>

Annexe

Programme Arduino pour notre robot

```

intmoteurRF = 9;intmoteurLR = 10;int moteurRF = 11;int moteurRR = 12;

intmoteurD = 7;intvitesse;void setup() {

pinMode(moteurLF,OUTPUT);pinMode(moteurLR,OUTPUT);

pinMode(moteurRF,OUTPUT); pinMode(moteurRR,OUTPUT);

pinMode(moteurD,OUTPUT); pinMode(5,OUTPUT); pinMode(6,OUTPUT);

Serial.begin(9600);vitesse=256;  }

voidloop() { // putyour main code here, to runrepeatedly:

char t;

if(Serial.available()){ t= Serial.read();Serial.println(t);}switch(t){

case 'S' :

digitalWrite(moteurLF,LOW);digitalWrite(moteurLR,LOW);digitalWrite(moteurRF,LOW);

digitalWrite(moteurRR,LOW);digitalWrite(moteurD,LOW);analogWrite(5,vitesse);

analogWrite(6,vitesse);break;

case 'F' :

digitalWrite(moteurLF,HIGH);digitalWrite(moteurLR,LOW);digitalWrite(moteurRF,HIGH);

digitalWrite(moteurRR,LOW);digitalWrite(moteurD,HIGH);analogWrite(5,vitesse);

analogWrite(6,vitesse);break;

case 'B' :

digitalWrite(moteurLF,LOW);digitalWrite(moteurLR,HIGH);digitalWrite(moteurRF,LOW);

digitalWrite(moteurRR,HIGH);digitalWrite(moteurD,LOW);analogWrite(5,vitesse);

```

```

analogWrite(6,vitesse);break;

case 'L' :

digitalWrite(moteurLF,LOW);digitalWrite(moteurLR,HIGH);digitalWrite(moteurRF,HIGH);

digitalWrite(moteurRR,LOW);digitalWrite(moteurD,LOW);analogWrite(5,vitesse);

analogWrite(6,vitesse);break;

case 'R' :

digitalWrite(moteurLF,HIGH);digitalWrite(moteurLR,LOW);digitalWrite(moteurRF,LOW);

digitalWrite(moteurRR,HIGH);digitalWrite(moteurD,HIGH);analogWrite(5,vitesse);

analogWrite(6,vitesse);break;

case 'G' :

digitalWrite(moteurLF,HIGH);digitalWrite(moteurLR,LOW);digitalWrite(moteurRF,HIGH);

digitalWrite(moteurRR,LOW);digitalWrite(moteurD,LOW);analogWrite(5,vitesse/3);

analogWrite(6,vitesse);break;

case 'I' :

digitalWrite(moteurLF,HIGH);digitalWrite(moteurLR,LOW);digitalWrite(moteurRF,HIGH);

digitalWrite(moteurRR,LOW);digitalWrite(moteurD,LOW);analogWrite(5,vitesse);

analogWrite(6,vitesse/3);break;

case'q': vitesse=256;break; case'9': vitesse=220;break ;case'8': vitesse=190;break;

case'7': vitesse=160;break; case'6': vitesse=130;break; case'5': vitesse=110;break;

case'4': vitesse=90;break;case'3': vitesse=70;break;case'2': vitesse=50;break;

case'1': vitesse=30;break;case'0': vitesse=10;break; } }

```

Résumé

Notre projet c'est un mémoire d'un robot mobile de nettoyage avec commande, est de type véhicule composé par un châssis de plastique et quatre roues.

Ce robot est basé sur une carte Arduino Uno et un moteur driver L298N, un module Bluetooth qui nous permet de le contrôler avec une application d'un téléphone android.

Son rôle est de faire les tâches de nettoyage (sols par exemple). Ce dernier permet d'économiser du temps et de l'énergie pour la personne.

Mots clés: Robot mobile, Arduino, mode Bluetooth ,moteur driver L298N.

Abstract

Our project is a robot mobile 4-wheel robot cleaner. This robot is based on an Arduino Uno board, L298N motor driver and a Bluetooth module that allows us to control it with an application from an android phone. Its role is to do cleaning tasks (floors for example). By doing this type of tasks it saves time and energy for the person.

Keywords: Robot mobile, Arduino, Bluetooth mode, L298N motor driver.

ملخص

مشروعنا هو عبارة روبات متحرك من نوع 4عجلات مزود بالية تنظيف للأرضية.

يعتمد هذا الروبوت على وحدة معالجة أرد وينو أونو, محرك سائق و بلوتوت الذي يتيح لنا التحكم بالروبوت عن بعد بواسطة تطبيق في الهاتف.

دوره هو القيام ببعض الأعمال وتوفير الجهد والوقت للإنسان.

الكلمات الافتتاحية روبات متحرك , اردوينو ,بلوتوت, محرك سائق