N° Ordre……../FHC/UMBB/2024

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE M'HAMED BOUGARA-BOUMERDES

Faculté des Hydrocarbures et de la Chimie

Mémoire de Fin d'Etudes
En vue de l'obtention du diplôme :

MASTER

Présenté par
Bennour Abderrahim

Filière : Commande automatique
Spécialité : Commande automatique

Thème

Conception et Mise en Œuvre d'une Carte de Commande pour le Bras Manipulateur TR5

Devant le jury :

HABBI	Hacene	Pr	UMBB	Président
KESRAOUI	Mohammed	Pr	UMBB	Examinateur
CHAIB	Ahmed	Pr	UMBB	Encadrant

Année Universitaire : 2023/2024

Remerciement

Au terme de ce travail de mémoire de master, les mots justes sont difficiles à trouver pour exprimer mes remerciements à **Allah**, le Tout-Puissant, qui m'a accordé le courage pour élaborer ce modeste travail. Merci pour tous ces bienfaits autour de moi et pour la direction de ma vie.

Un très grand merci à mon promoteur, **Mr. Chaib Ahmed** pour ses orientations, sa disponibilité, et pour tout ce qu'il m'a appris. Mes plus sincères remerciements s'étendent également au corps administratif de la faculté des sciences et sciences appliquées pour la richesse et la qualité de leur enseignement et pour les grands efforts qu'ils déploient pour assurer à leurs étudiants une formation actualisée.

Je remercie mes très chers parents, qui ont toujours été là pour moi. "Vous avez tout sacrifié pour vos enfants, n'épargnant ni santé ni efforts. Vous avez donné un magnifique modèle de labeur et de persévérance, et je suis redevable d'une éducation dont je suis fier"

Je tiens à remercier le merveilleux M. **Berkane Aimen**, qui a toujours été là pour moi, pour son soutien inconditionnel et son encouragement.

Enfin, je tiens à remercier tous ceux qui, d'une manière ou d'une autre, ont participé à la réussite de ce travail.

Dédicace

A ma très chère maman

Affables, honorable, aimable : Vous représentez pour moi le symbole de la bonté par excellence, la source de tendresse, et l'exemple de dévouement. Vous n'avez pas cessé de m'encourager et de prier pour moi.

Je vous dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, vous préserve et vous accorde santé, langue vie et bonheur.

A mon cher père

A ma chère famille

En témoignage de l'attachement, de l'amour et de l'affection que je porte pour vous.

Je dédie ce travail à ma chère et unique sœur sans oublier mes neveux Zaid et Majd.

À mes frères Ilyes et Riad

Pour votre soutien constant, vos encouragements et votre présence rassurante. Vous m'avez toujours aidé et conseillé quand j'en avais besoin. Je vous dédie ce travail en reconnaissance et gratitude pour tout ce que vous avez fait pour moi. Je vous souhaite santé, succès et bonheur.

Je vous dédie ce travail avec tous mes vœux de bonheur, de santé et de réussite.

Enfin à mes futures femmes, je dédie ce travail. Où que vous soyez, je veux que vous sachiez que chaque effort consenti dans cette aventure académique est aussi pour vous, en espérant partager avec vous un futur riche en amour, en soutien mutuel et en accomplissements communs.

Résumé

Ce projet de fin d'études se concentre sur la conception et la mise en œuvre d'une carte de commande en puissance pour le bras manipulateur TR5. En utilisant la carte Arduino UNO et le driver L298, le système est capable de contrôler précisément les mouvements du bras. Le projet vise à faciliter l'apprentissage des étudiants en programmation et en commande de machines robotisées, tout en explorant les concepts d'électronique et d'automatique. Des simulations sont réalisées à l'aide du logiciel Proteus pour valider le fonctionnement du système.

Abstract

This end-of-studies project focuses on the design and implementation of a power control board for the TR5 manipulator arm. Using the Arduino UNO board and the L298 driver, the system is able to precisely control the movements of the arm. The project aims to facilitate students' learning in programming and controlling robotic machines, while exploring concepts of electronics and automation. Simulations are carried out using Proteus software to validate the system's operation.

الملخص

يتركز هذا المشروع الختامي للدراسات على تصميم وتنفيذ لوحة تحكم في الطاقة لذراع TR5. باستخدام لوحة Arduino UNO ومشغل L298، يمكن للنظام التحكم بدقة في حركات الذراع. يهدف المشروع إلى تسهيل تعلم الطلاب في البرمجة والتحكم في الآلات الروبوتية، بينما يتم استكشاف مفاهيم الإلكترونيات والتحكم التلقائي. يتم إجراء محاكاة باستخدام برنامج Proteus للتحقق من عمل النظام

Les mots clés :TR5, bras manipulateur, Arduino UNO, L298N, Proteus

Table des matières

Introduction générale **1**

I. Chapitre 01 : Généralités sur la robotique ***2***

I.1. Introduction **3**

I.2. Définition d'un robot **3**

I.3. Historique **4**

I.3.1 Histoire ancienne de la robotique 4

I.3.2 Le 20ème siècle et les débuts de la robotique moderne 4

I.3.3 21ème siècle : l'ère de l'intelligence artificielle 5

I.4. Les différentes catégories des robots [2] **5**

I.4.1 Robot mobile 5

I.4.2 Robot sous- marins 5

I.4.3 Robots volants 6

I.4.4 Robots humanoïdes 6

I.4.5 Robot intelligent 7

I.5. Concepts de base [4] **7**

I.5.1 Le système mécanique articulé (S.M.A.) 8

I.5.2 Organe terminal 9

I.5.3 Les actionneurs 9

I.5.4 Les Capteurs 10

I.5.5 Le Processeur 10

I.5.6 Logiciel 11

I.6. Caractéristiques des robots **11**

I.6.1 Description de sa géométrie 11

I.6.2 Caractéristiques géométriques 11

I.6.3 Espace de travaille 12

I.6.4 Précision/Répétabilité 12

I.6.5 Performances dynamiques 13

I.6.6 Charge utile 13

I.7. Les types de robots **13**

I.7.1 Les Robots manipulateurs 13

I.7.2 Les robots mobiles 15

I.8. Utilisation des Robots **18**

I.8.1 Le domaine militaire 18

I.8.2 Le domaine de santé : 18

I.8.3 Domain civil 19

I.8.4 Domaine de l'usage domestique 20

I.8.5 Domaine de l'usage industriel : 21

I.9. Conclusion **21**

II. Chapitre 02 : Présentation du robot « TR5 » ***22***

II.1. Introduction **23**

II.2. La structure du bras manipulateur TR5 ..... **23**

II.2.1 Propriétés mécaniques 23

II.2.2 Propriétés électriques 25
II.3. Représentation géométrique du bras manipulateur TR5 30
II.4. Espace de travail du TR5 31
II.5. Les modes de fonctionnement 31
II.5.1 Mode SAISIE 31
II.5.2 Mode POSITION 31
II.5.3 Mode course 31
II.5.4 Mode PAUSE 31
II.5.5 Mode ÉTAPE 31
II.5.6 Mode d'affichage 32
II.6. Conclusion 32
III. Chapitre 03 : Software et hardware 33
III.1. Introduction 34
III.2. Présentation de l'Arduino 34
III.2.1 La partie Hardware 34
III.2.2 La partie software (Logiciel Arduino) 39
III.3. Joystick (organe de commande) 42
III.4. Le Driver L298 43
III.4.1 Description L298xx (H-Bridge Motor Driver) 43
III.4.2 Les Caractéristiques du L298N 44
III.4.3 Principe de fonctionnement du pilote (Command du L298N) 44
III.5. Présentation générale de Proteus (ISIS et ARES) : 45
III.5.1 L'éditeur de schéma ISIS 46
III.5.2 Ares 49
III.6. Conclusion 49
IV. Chapitre 04 : La commande et la simulation de projet 50
IV.1. Introduction 51
IV.2. Présentation de projet 51
IV.2.1 Limite en commande 51
IV.3. Description de Schéma proposé : 51
IV.4. Câblage d'un module de pilote de moteur L298N à un Arduino 53
(3 moteurs) 53
IV.4.1 Pourquoi la carte de puissance 53
IV.4.2 La simulation de câblage de 3 moteurs avec L298N et Arduino 54
IV.4.3 Code Arduino : 54
IV.5. Câblage de 6 moteurs dc avec 2 Arduino et 3 modules L298N 56
IV.5.1 Résumé des connexions : 56
IV.5.2 Fonctionnement global : 57
IV.5.3 Le code Arduino de chaque Arduino 57
IV.5.4 Le 2 -ème Arduino 60
IV.5.5 La simulation de notre câblage 61
IV.6. La partie expérimentale 62
IV.6.1 La partie finale de commande 62
IV.7. Conclusion 64

Conclusion générale 65

Bibliographie 66

Table des figures

Figure I-1:Parties commande-contrôle et opérative d'un robot 3
Figure I-2:Parties commande- contrôle et opérative d'un robot 3
Figure I-3:Robot sous- marin 6
Figure I-4: Robot volant 6
Figure I-5:Robot humanoïde 6
Figure I-6:Robots intelligents. (Exemple : robot Sophia) 7
Figure I-7:Concepts de base d'un robot 7
Figure I-8 : Parties principales d'un robot 8
Figure I-9:Articulation prismatique et Articulation rotoïde 9
Figure I-10:Actionneurs de robot 10
Figure I-11:Capteurs de robot 10
Figure I-12:Processeur de robot. 11
Figure I-13:volume de travail d'un robot. 12
Figure I-14:Structure et espace de travail cylindrique 14
Figure I-15:Structure et espace de travail sphérique 14
Figure I-16: Structure et espace de travail cartésien 14
Figure I-17:Structure et espace de travail anthropomorphe 15
Figure I-18:Robot parallèle 15
Figure I-19:Robot de type unicycle 16
Figure I-20:Robot de type tricycle 16
Figure I-21:Robot de type voiture 17
Figure I-22:Robot de type omnidirectionnel 17
Figure I-23:Usage des robots dans le domaine militaire 18
Figure I-24:Robots éclaireur tout-terrain 18
Figure I-25:Usage des robots dans le domaine médical 19
Figure I-26: Robot à usage civile 19
Figure I-27:Robot spirit et opportunité 20
Figure I-28:Usage des robots dans le domaine domestique 20
Figure I-29:Robot à usage industriel 21
Figure II-1:Le TR5 23
Figure II-2:Les axes du robot TR5 24
Figure II-3:Les dimensions du robot TR5 (vue de haut) 24
Figure II-4:Les dimensions du robot TR5 (vue de face) 24
Figure II-5:La pince de TR5 25
Figure II-6:Contrôleur de TR5 25
Figure II-7:La Boîte d'apprentissage de TR5 26
Figure II-8:Unité d'alimentation de TR5 27
Figure II-9:Le capteur de robot TR5 27
Figure II-10Architecteur d'un MCC à aiment permanent 29
Figure II-11:transmissions de mouvement avec pignons 29
Figure II-12:Câble d'interface 29
Figure II-13:Boîte de test d'E/S 30
Figure II-14:Carte E/S pour PC 30
Figure II-15:Représentation géométrique du TR5 30
Figure II-16:Espace de travail de bras manipulateur TR5 31

Figure III-1:Constitution de la carte Arduino UNO.....35
Figure III-2:Brochage de la carte Arduino Uno 38
Figure III-3:Microcontrôleur ATMega328 39
Figure III-4:Vue d'ensemble et interface du logiciel Arduino 40
Figure III-5:Message d'Erreur.....41
Figure III-III-6:Joystick de contrôle.....42
Figure III-III-7:schéma interne d'une Joystick.....42
Figure III-8:Circuit d'un pont en H......43
Figure III-9:Module à base de circuit L298 44
Figure III-10:Fonctionnement du double Pont en H. 45
Figure III-11:Le logiciel proteus 46
Figure III-12:Interface utilisateur ISIS. 46
Figure III-13:Barre de menu.....46
Figure III-14 : Barre d'outils de sélection de mode 47
Figure III-15 : Barre d'outils d'orientation.....47
Figure III-16:Zone de travail ou d'édition des schémas 48
Figure III-17: Vue d'ensemble du schéma et de positionnement 48
Figure III-18: Sélecteur d'objets 48
Figure III-19:Organigramme des Etapes de la saisie de schéma. 49
Figure IV-1:Schéma de notre montage.....52
Figure IV-2:Câblage d'un module de pilote de moteur avec Arduino 53
Figure IV-3:Câblage et simulation de 3 moteurs avec Proteus.....54
Figure IV-4:Cablage de notre projet 56
Figure IV-5:la simulation de notre projet 62

Table des tableaux

Tableau I-1:Les avantages et des inconvénients des différents types de robots à roues.. 17
Tableau II-1:Les angles de rotation du robot TR5 24
Tableau II-2:Caractéristiques des capteurs du robot TR5 28
Tableau III-1:Les différentes broches de la carte ARDUINO UNO 35
Tableau III-2:Caractéristiques de la carte Arduino UNO 38
Tableau III-3:Les caractéristiques d'atmega328 39
Tableau III-4:Barre d'Action...... 41
Tableau III-5:Barres d'outils de commande..... 47

Introduction générale

Le développement de la science et de la technologie s'étend dans divers domaines et disciplines, notamment l'électronique et la robotique, qui sont devenues des composantes essentielles de notre monde. La robotique est aujourd'hui une discipline extrêmement populaire, alliant un fort intérêt pédagogique et industriel. Elle exige une grande créativité et des connaissances pluridisciplinaires.

Un robot est généralement constitué d'un système mécanique articulé et d'une partie électronique permettant sa commande. Il peut également d'intégrer divers capteurs selon les tâches à accomplir. La programmation du système de commande requiert des compétences en informatique, en automatique, en traitement temps-réel, ainsi que dans des domaines avancés tels que la vision artificielle et l'intelligence artificielle.

Ce manuscrit est structuré en quatre chapitres principaux. Le premier chapitre introduit les concepts de base des robots, en illustrant les différentes catégories et usages. Le deuxième chapitre fournit une description détaillée du bras manipulateur TR5, en couvrant ses aspects mécaniques et électriques. Le troisième chapitre se concentre sur la partie logicielle, en particulier sur la programmation du microcontrôleur ATMEGA 2560 utilisé dans le projet. Enfin, le quatrième chapitre traite de la réalisation de la carte de commande en puissance des moteurs du TR5 ainsi que de l'unité de commande et de contrôle du mouvement de ce bras manipulateur.

I. Chapitre 01 : Généralités sur la robotique

I.1. Introduction

A l'origine des robots ont été conçus pour remplacer les travailleurs humains, qui souffraient de problèmes industriels à l'époque, on pensait que les robots seraient des machines universelles capables avec une reprogrammation rapide pour une grande variété de tâches, ce sont ces idées qui ont motivé leur développement.

La robotique est un ensemble de disciplines techniques (mécanique, électronique, automatique, informatique) articulées autour d'un objectif et d'un objet commun. Cet objectif est l'automatisation flexible de nombreux secteurs de l'activité humaine réputés jusqu'à très récemment comme ne pouvant se passer de la présence de l'homme, et l'objet est le robot, sorte de machine universelle dont l'homme rêve depuis toujours (mais qui n'a pas encore la forme de ce rêve).

I.2. Définition d'un robot

Un robot est un dispositif mécanique poly-articulé mus par des actionneurs et commandé par un contrôleur accomplissant automatiquement une grande variété des tâches qui sont généralement considérées comme dangereuses, pénibles, répétitives et impossible pour les humains ou dans un but d'une plus grande efficacité.

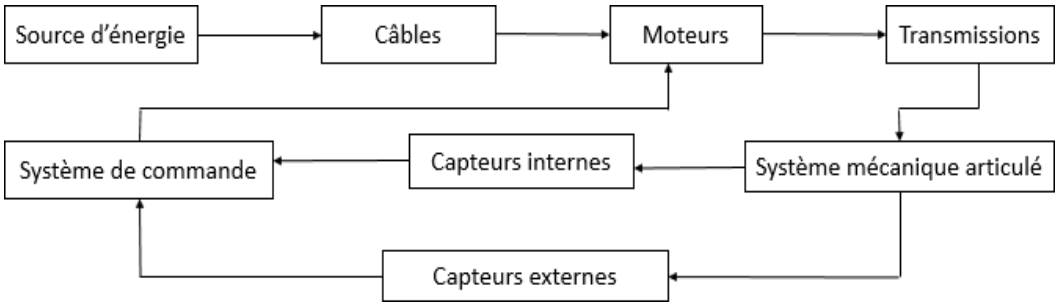


Figure I-1:Parties commande-contrôle et opérative d'un robot

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

- **La versatilité** : Un robot doit avoir la capacité de pouvoir exécuter une variété de tâches, ou la même tâche de différentes manières ;
- **L'auto-adaptabilité** : Un robot doit pouvoir s'adapter à un environnement Changeant au cours de l'exécution de ses tâches.

L'association française de normalisation (A.F.N.O.R) définit un robot comme étant un système mécanique de type manipulateur commandé en position, reprogrammable, polyvalent (i.e., usages multiples), à plusieurs degrés de liberté, capable de manipuler des matériaux, des pièces, des outils et des dispositifs spécialisés, au cours de mouvements variables et programmés pour l'exécution d'une variété de tâches [1].

I.3. Historique

I.3.1 Histoire ancienne de la robotique

Il y a des milliers d'années, les anciens Égyptiens développaient déjà des modèles mathématiques très avancés et construisaient des automatisations très sophistiquées, comme l'horloge à eau. Les montres peuvent être considérées comme les machines antiques les plus parfaites, très proches du concept d'automatisation et donc de la robotique.

Des siècles plus tard, dans la Grèce antique, l'ingénieur et mathématicien Archimède découvrit la puissance des leviers, des poulies et des plans inclinés au 3ème siècle avant J.-C. Avec ces mécanismes, capables d'agir comme des bras et des mains humaines, il conçut un prototype capable de tirer de l'eau un navire coulé.

Au Moyen Âge, les premiers automates sont créés : des machines qui imitent le mouvement d'un être vivant. L'un des plus célèbres était l'homme de fer d'Alberto Magno, créé au 13ème siècle : c'est le premier androïde de l'histoire, construit en fer, verre et cuir ; il savait marcher, se rendait à la porte du monastère et s'occupait de recevoir les visiteurs.

L'Automa Cavaliere était un automate à forme humaine conçu par Léonard de Vinci vers 1495. Développé en armure complète, il pouvait s'asseoir et se tenir debout, bouger les bras et les jambes, et était le plus grand divertissement des cours royales italiennes.

Une étape déterminante pour le développement de la robotique et de la programmation en général s'est produite au 17ème siècle. Le philosophe et mathématicien allemand Gottfried Von Leibniz a développé le système binaire comme base du calcul automatique, jetant définitivement les bases de l'informatique et du langage machine des zéros et des uns.

I.3.2 Le 20ème siècle et les débuts de la robotique moderne

Le 20ème siècle a marqué un avant et un après en robotique : en un seul siècle, nous sommes passés de la fabrication de robots mécaniques animés par le mouvement à la création de tout un univers de robots numériques et d'intelligence artificielle.

En 1937, Bill Taylor, un étudiant britannique de 21 ans, a conçu un robot inspiré des jouets mécaniques qui a fini par être utilisé pour prendre et placer des marchandises dans l'industrie. Le robot a été baptisé Gargantua et a été le premier de nombreux autres robots à usage industriel qui ont été utilisés pour peindre, couper et séparer des produits, etc. Lors de la foire industrielle de New York de 1939, plus de vingt prototypes de robots à usage industriel et domestique sont présentés.

Coïncidant avec l'apogée de la robotique moderne, l'écrivain de science-fiction Isaac Asimov a publié Robot (1950), un recueil de nouvelles mettant en scène des robots intelligents. Dans son travail, Asimov développe les soi-disant "lois de la robotique", qui limitent le fonctionnement des robots afin qu'ils ne constituent pas une menace pour l'homme.

Au début des années 1960, la société américaine Unimation développe le premier robot de transfert programmable : il est considéré comme l'embryon du robot industriel tel que nous le connaissons aujourd'hui. Cette conception inspire General Motors à créer les premières chaînes de montage automatisées pour fabriquer des automobiles. Ce sont les robots dits collaboratifs : ceux qui collaborent avec les humains pour réaliser une même tâche.

Dès lors, de nouveaux robots sont apparus dans d'autres domaines industriels : robots spécialisés dans le soudage, l'injection plastique, le mélange alimentaire… L'évolution de la technologie a permis d'automatiser des domaines et des secteurs auparavant inimaginables.

Les années 1980 marquent le début de la robotique intelligente : la recherche commence à donner aux robots une plus grande capacité de performance. Une décennie plus tard, avec l'apparition du Big Data et de l'Internet des Objets (IoT), l'intelligence artificielle fait son apparition.

I.3.3 21ème siècle : l'ère de l'intelligence artificielle

Aujourd'hui, nous sommes entourés de robots dans tous les domaines de notre vie : aspirateurs autonomes, livraison de colis par drones, véhicules sans conducteur, robots éducatifs dans les salles de classe, robots chirurgicaux dans les hôpitaux…

On dit que nous sommes dans la Quatrième Révolution Industrielle, basée sur le développement de l'Intelligence Artificielle (IA) et le fonctionnement autonome des machines. Comme toute révolution, elle transformera notre mode de vie et signifiera une numérisation de tous les processus de l'activité humaine.

Il est encore trop tôt pour savoir jusqu'où nous irons avec cette nouvelle révolution. Ce qui est certain, c'est que l'avenir est lié au développement et à l'évolution de la robotique : une relation de collaboration entre les robots et les humains pour renforcer l'humanité du futur.

I.4. Les différentes catégories des robots [2]

I.4.1 Robot mobile

Un robot mobile est un système mécanique, électrique et informatique agissent physiquement sur son environnement en vue d'atteindre un objectif qui lui a été assigné.

Cette machine est polyvalente et capable de s'adapter à certaines variations de ses conditionsde fonctionnement. Elle est dotée de fonction de perception, de décision et d'actions.

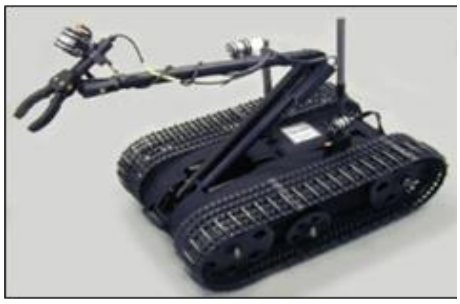


Figure I-2 : Robot mobile

I.4.2 Robot sous- marins

Les robots sous-marins (aussi appelés drones sous-marins) sont des véhicules qui sont capables de fonctionner sous l'eau sans un occupant humain. Ils peuvent être contrôlées par un opérateur humain directement ou être autonome et fonctionner indépendamment de commande humaine [2]

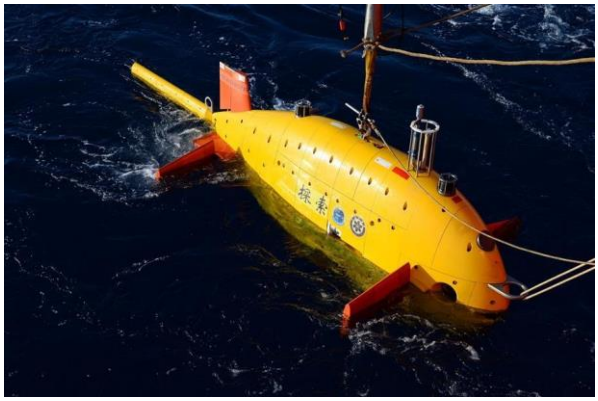


Figure I-3:Robot sous- marin

I.4.3 Robots volants

Un drone désigne un aéronef sans pilote à bord, il peut avoir un usage civil ou militaire**.** Les drones sont utilisés au profit des forces armées ou de sécurité.

Figure I-4: Robot volant

I.4.4 Robots humanoïdes

Ces robots ont la morphologie se rapprochant le plus de l'humain en tout ou en une partie : Deux bras, un tronc, des jambes et une tête. C'est aujourd'hui les robots les plus complexes à concevoir, ils nécessitent des connaissances variées et très pointues.[2]

Figure I-5:Robot humanoïde

I.4.5 Robot intelligent

On trouve actuellement des robots de seconde génération qui sont capables d'acquérir et d'utiliser certaines informations sur leur environnement (systèmes de vision, détecteurs de proximité, capteurs d'efforts, ...), Les robots de troisième génération sont capables de comprendre un langage oral proche du langage naturel et de se débrouiller de façon autonome dans un environnement complexe grâce à l'utilisation de l'intelligence artificielle. [3]

Figure I-6:Robots intelligents. (Exemple : robot Sophia)

I.5. Concepts de base [4]

Un robot est constitué des éléments suivants :

- Organe terminal
- Axe = Articulation
- Corps = Segment
- Effecteur = outil
- Actionneur = Moteur
- Base

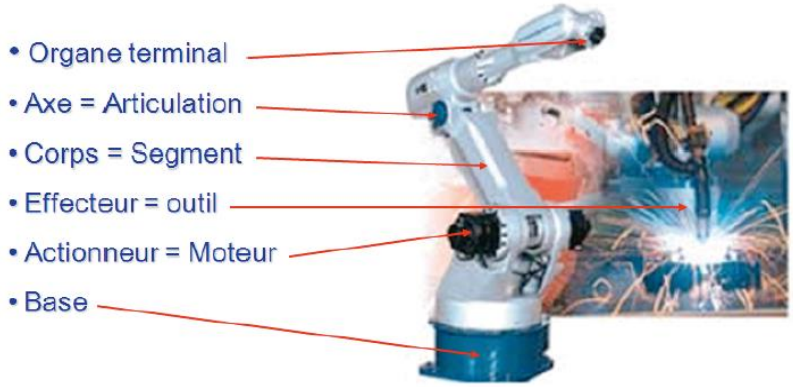


Figure I-7:Concepts de base d'un robot

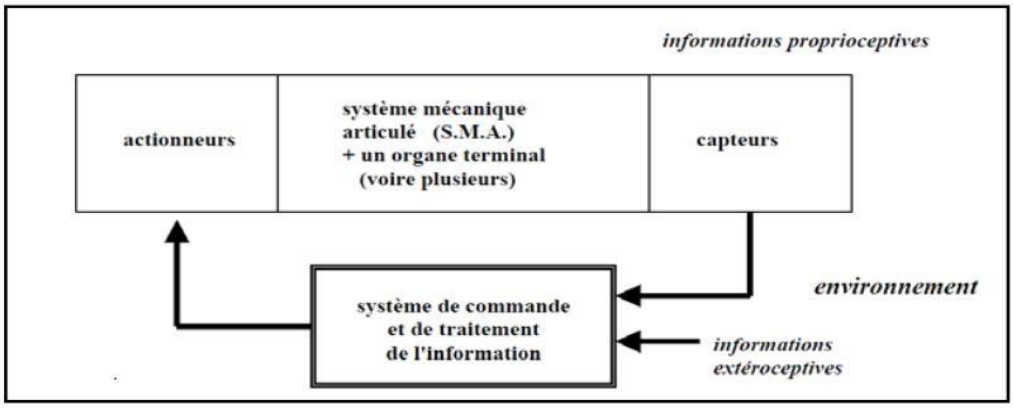


Figure I-8 : Parties principales d'un robot

On distingue classiquement 5 parties principales dans un robot :

- Système mécanique articulé (S.M.A.) + un organe terminal (voir plusieurs)
- Organe terminal
- Actionneurs
- Capteurs
- Système de commande

I.5.1 Le système mécanique articulé (S.M.A.)

Le système mécanique articulé (S.M.A.) est un mécanisme ayant une structure plus ou moins proche de celle du bras humain dans le cas des robots manipulateurs. Il permet de remplacer ou de prolonger son action. Son rôle est d'amener l'organe terminal dans une situation (position et orientation) donnée, selon des caractéristiques de vitesse et d'accélération données. Son architecture est une chaîne cinématique de corps, généralement rigides (ou supposés comme tels), assemblés par des liaisons appelées articulations.

Sa motorisation est réalisée par des actionneurs électriques, pneumatiques ou hydrauliques qui transmettent leurs mouvements aux articulations par des systèmes appropriés [5].

I.5.1.1 Articulation

Une articulation lie deux corps successifs en limitant le nombre de degré de liberté de l'un par rapport à l'autre. Soit m le nombre de degré de liberté résultant, encore appelé mobilité de l'articulation. La mobilité d'une articulation est telle que : m se situe entre 0 et 6 lorsque m = 1 ; ce qui est fréquemment le cas en robotique, l'articulation est dite simple : soit rotoïde, soit prismatique.

I.5.1.1.1 Articulation rotoïde

Il s'agit d'une articulation de type pivot, notée R, réduisant le mouvement entre deux corps à une rotation autour d'un axe qui leur est commun. La situation relative entre les deux corps est donnée par l'angle autour de cet axe (voir la figure suivante).

I.5.1.1.2 Articulation prismatique

Il s'agit d'une articulation de type glissière, notée P, réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe (voir la figure suivante) [5].

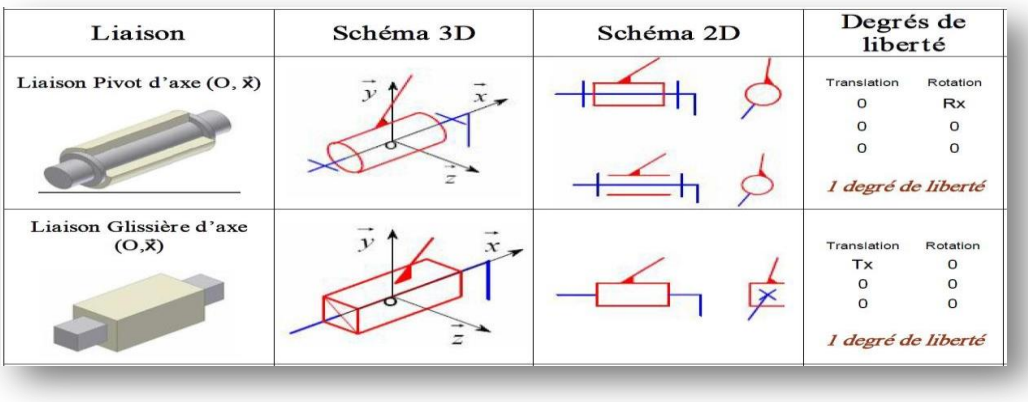


Figure I-9:Articulation prismatique et Articulation rotoïde

I.5.2 Organe terminal

Désigne tout dispositif destiné soit à manipuler des objets comme les dispositifs de serrage (pinces à deux ou trois doigts), les dispositifs magnétiques ou à dépression (ventouse), soit à transformer (outils de découpe, torche de peinture, torche de soudage). Il s'agit d'une interface permettant au robot d'interagir avec son environnement.

I.5.3 Les actionneurs

Pour être animé, le S.M.A comporte des moteurs plus souvent associés à des transmissions (courroies crantées), l'ensemble constitue les actionneurs. Les actionneurs utilisent fréquemment des moteurs électriques à aimant permanent, à courant continu, à commande par l'induit (la tension n'est continue qu'en moyenne car en général l'alimentation est un hacheur de tension à fréquence élevée, bien souvent la vitesse de régime élevée du moteur fait qu'il est suivi d'un réducteur, ce qui permet d'amplifier le couple moteur). On trouve de plus en plus de moteurs à commutation électronique (sans balais), ou pour des petits robots, des moteurs pas à pas. Pour les robots devant manipuler de très lourdes charges (par exemple, une pelle mécanique), les actionneurs sont le plus souvent hydrauliques, agissant en translation (vérin hydraulique) ou en rotation (moteur hydraulique).

Les actionneurs pneumatiques sont d'un usage général pour les manipulateurs à cycles robots tout ou une succession de mouvements contrôlés uniquement par des capteurs de fin de course réglables manuellement à la course désirée asservissement en position difficile dû à la compressibilité de l'air.

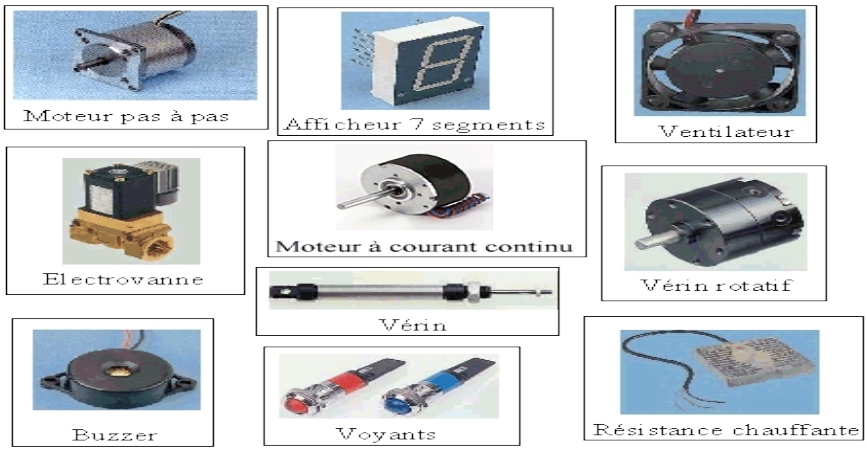


Figure I-10:Actionneurs de robot

I.5.4 Les Capteurs

La perception permet de gérer les relations entre le robot et son environnement. Les organes de perception sont des capteurs dits proprioceptifs lorsqu'ils mesurent l'état interne du robot (positions et vitesses des articulations) et extéroceptifs lorsqu'ils recueillent des informations sur l'environnement (détection de présence, de contact, mesure de distance, vision artificielle) [5].

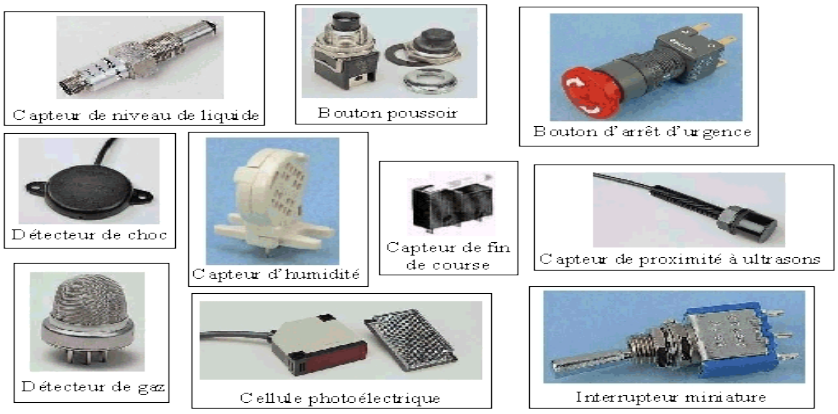


Figure I-11:Capteurs de robot

I.5.5 Le Processeur

Le processeur est le cerveau du robot. Il calcule les mouvements des articulations du robot, détermine combien et à quelle vitesse chaque articulation doit se déplacer pour atteindre l'emplacement et la vitesse souhaitée, et supervise les actions coordonnées du contrôleur et les capteurs. Dans certains systèmes, le contrôleur et le processeur sont intégrés ensemble en une seule unité, et dans d'autres cas, ce sont des unités séparées

Figure I-12:Processeur de robot.

I.5.6 Logiciel

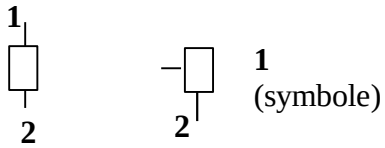
Trois groupes de logiciels sont utilisés dans un robot. L'un est le système d'exploitation qui exploite le processeur. Le second est le logiciel robotique qui calcule la motion nécessaire de chaque articulation du robot basée sur des équations cinématiques. Ces informations sont envoyées au dispositif de commande. Ce logiciel peut être à différents niveaux, de la langue de la machine aux langues sophistiqués utilisés par les robots modernes. Le troisième groupe est la collection d'application - orientée les routines et les programmes développés pour utiliser le robot ou ses périphériques pour des tâches spécifiques telles que l'assemblage, le chargement de machines, la manutention et les routines de vision.

I.6. Caractéristiques des robots

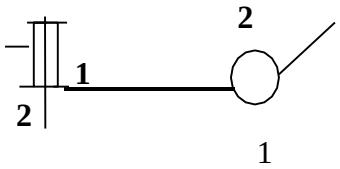
I.6.1 Description de sa géométrie

Robot=système mécanique poly-articulé :

Articulation prismatique :



Articulation rotoïde :

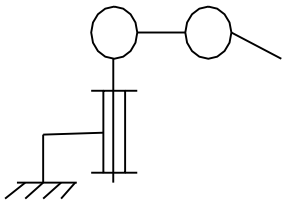


(Symbole)

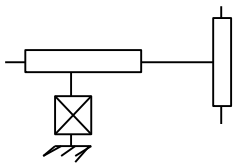
I.6.2 Caractéristiques géométriques

- Nombre d'axes (mus par actionneur).
- Architecture (série ou parallèle).
- Chaînage des articulations.
- Nombre de degré de liberté.

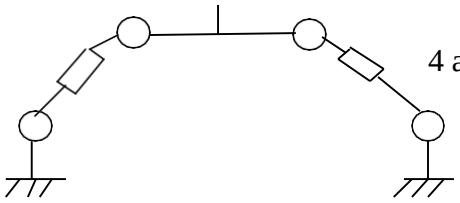
- ✓ P : Prismatique (translation)
- ✓ R : Rotation
- ✓ DL : Degré de liberté



3 axes, série, RR, 3DL



3 axes, séri**e,** PPP, 3DL



4 axes, parallèle, RP+RP,3DL

I.6.3 Espace de travaille

Le volume de travail accessible par l'outil du robot est le volume que peut balayer l'outil. Le volume dépend de la géométrie du robot, de la longueur des segments et du débattement des articulations [6].

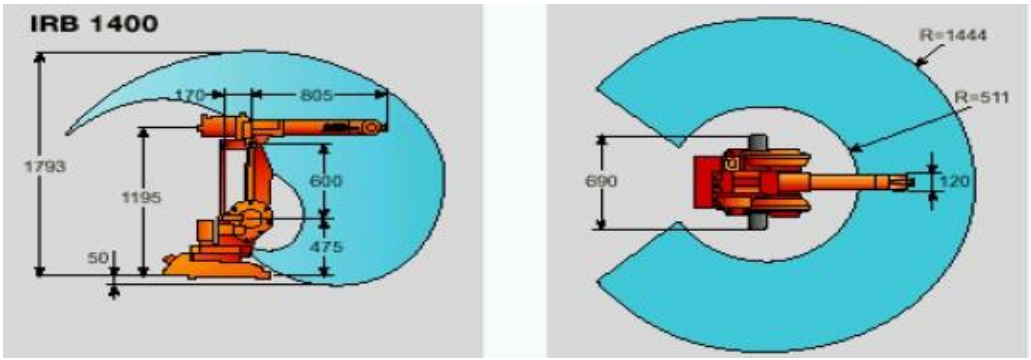


Figure I-13:volume de travail d'un robot.

I.6.4 Précision/Répétabilité

On dit que le positionnement absolu est imprécis lorsqu'il dépasse 1mm, alors on à l'une des erreurs suivantes :

- Erreurs du modèle géométrique.
- Erreurs de quantification de mesure de position.
- Flexibilité.

Répétabilité : la répétabilité d'un robot est l'erreur maximale de position répétée de l'outil en chaque point de son espace de travail. En générale, la répétabilité ne doit pas dépasser 0.1mm.

I.6.5 Performances dynamiques

I.6.5.1 Vitesse maximale

C'est la vitesse de translation ou de rotation de chaque axe du robot. Souvent les constructeurs donnent une vitesse maximale de l'outil ou de l'organe terminale.

I.6.5.2 L'accélération maximale

Dépend de l'inertie, donc de la position du robot. Pour chaque axe, cette accélération est donnée dans la configuration la plus défavorable (inertie maximale, charge maximale).

I.6.6 Charge utile

C'est la charge qui garantit une durée de vie la plus longe possible. Donc, c'est la charge maximale que peut porter le robot sans perturber ni la répétabilité ni les performances dynamiques du robot. Cette charge utile est nettement inférieure à la charge maximale que peut porter le robot et directement dépendante des actionneurs [6].

I.7. Les types de robots

Il existe deux grandes familles de robots sont :

- ✓ Les robots manipulateurs.
- ✓ Les robots mobiles.

I.7.1 Les Robots manipulateurs

Un robot manipulateur est en forme d'un bras et se composent d'un certain nombre de segments qui est conçu pour manipuler ou déplacer des matériaux, outils et pièces sans contact humain direct. Ils sont des dispositifs qui permettent aux humains d'interagir avec des objets dans un environnement en toute sécurité. Les robots manipulateurs sont utilisés dans des applications industrielles pour s'effectuer efficacement des tâches telles que l'assemblage, soudage, traitement de surface, et le forage.

Les robots manipulateurs viennent sous plusieurs formes. Les formes se répartissent en cinq grandes catégories :

I.7.1.1 Les robots cylindriques

Ce type de robots est caractérisé par une rapidité et un espace de travail cylindrique. C'est un robot à 3 axes, série (RTT) et 3 degrés de liberté, son espace de travail est cylindrique [7].

Figure I-14:Structure et espace de travail cylindrique

I.7.1.2 Les robots sphériques

Ce sont des robots à 3 axes, série RTT, et 3 degrés de liberté, leur espace de travail est sphérique, ils sont caractérisés par leur grande charge utile [7].

Exemple :

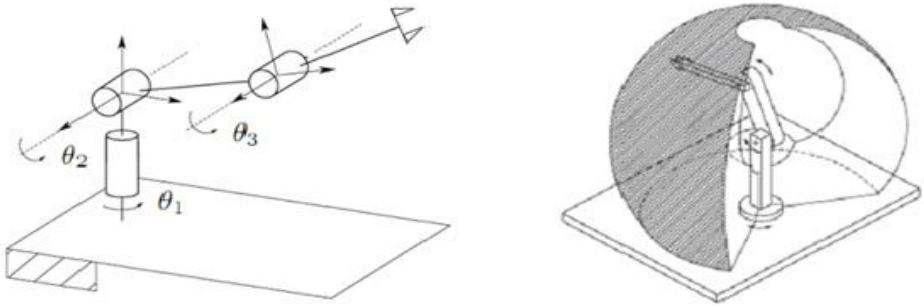


Figure I-15:Structure et espace de travail sphérique

I.7.1.3 Les robots cartésiens

Ce sont des robots à 3 axes perpendiculaire 2 à 2, série, TTT, et 3 degrés de liberté et caractérisés par leur très bonne précision.[7]

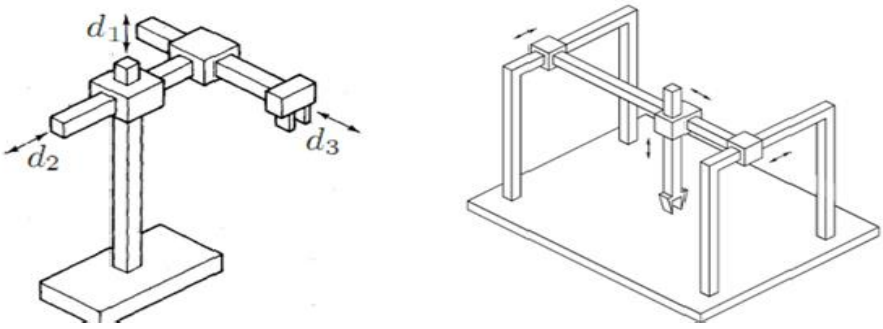


Figure I-16: Structure et espace de travail cartésien

I.7.1.4 Les robots anthropomorphes

Ce sont des robots à 9 axes, séries, 6 rotations, et 6 degrés de liberté, ils reproduisent la structure d'un bras humain [7].

Exemple :

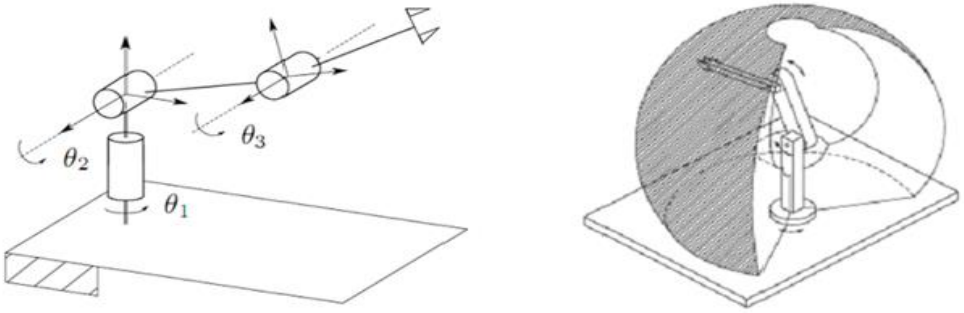


Figure I-17:Structure et espace de travail anthropomorphe

I.7.1.5 Les robots parallèles

Ce sont des robots à plusieurs chaines cinématique en parallèle et un espace de travail réduit. Ils sont caractérisés par la rapidité et la précision (grande rigidité de la structure).

Figure I-18:Robot parallèle

I.7.2 Les robots mobiles

Un robot mobile est celui qui peut se déplacer dans son environnement de façon indépendante. Pour ce faire, le robot doit pouvoir naviguer, la portée et la précision de navigation requise varient en fonction de la taille du robot et du type de sa tâche. [8]

I.7.2.1 L'architecteur des robots mobiles

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

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

I.7.2.2 Les robots mobiles à roues

Il existe 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 à roues.

I.7.2.2.1 Robot unicycle

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-holonome, 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. [8]

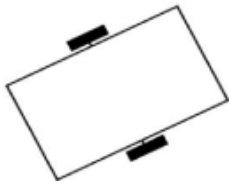


Figure I-19:Robot de type unicycle

I.7.2.2.2 Robot tricycle

Un robot de type tricycle est constitué de deux roues fixes 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 fixes et par l'orientation de la roue orientable. Son centre de rotation est situé à l'intersection de l'axe contenant les roues fixes et de l'axe de la roue orientable. C'est un robot non-holonome. En effet, il est impossible de le déplacer dans une direction perpendiculaire aux roues fixes. 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. [8]

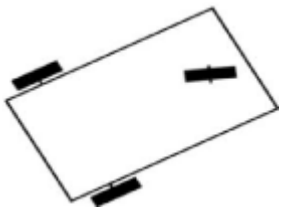


Figure I-20:Robot de type tricycle

I.7.2.2.3 Robot voiture

Un robot de type voiture est semblable au tricycle, il est constitué de deux roues fixes 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. [8]

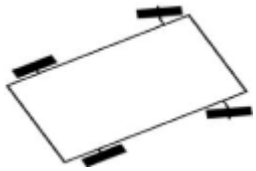


Figure I-21:Robot de type voiture

I.7.2.2.4 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.[8]

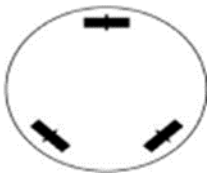


Figure I-22:Robot de type omnidirectionnel

I.7.2.2.5 Comparaison des différents types

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:Les avantages et des inconvénients des différents types de robots à roues

Type du robot	Avantage	Inconvénient
Unicycle	Stable Rotation sur soi-même Complexité mécanique faible	Non-holonome
Tricycle	Complexité mécanique Modérée	Non-holonome Peu stable Pas de rotation sur soi-même
Voiture	Stable Complexité mécanique modérée	Non-holonome Pas de rotation sur soi-même
Omnidirectionnel	Holonome Stable Rotation sur soi-même	Complexité mécanique Importante

I.8. Utilisation des Robots

La robotique est un domaine en plein développement depuis quelques années, l'évolution technologique, dépassant sans cesse nos espérances, permettent maintenant de réaliser des solutions technologiques s'adaptant au moindre problème. Par conséquent, la robotique est utilisée dans des domaines extrêmement rigoureux et exigeants. Nous allons explorer ces différents domaines [9]

I.8.1 Le domaine militaire

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

Figure I-23:Usage des robots dans le domaine militaire

Robots éclaireur tout-terrain : l'image captée par la caméra est transmise par une liaison fil aux fantassins au poste de commandement, de plus certains robots sont équipés d'un armement pour évoluer en milieu hostile, dans le but de remplacer les soldats pour limiter les pertes humaines, autre qu'une caméra infrarouge et une liaison Sans fil, ce robot est équipé d'un pistolet mitrailleur à but offensif (s'il en reçoit l'ordre à distance) ou défensif s'il est lui-même pris pour cible [8].

Figure I-24:Robots éclaireur tout-terrain

I.8.2 Le domaine de santé :

Les robots commencent à être de plus en plus dans le domaine médical, qu'il s'agisse de simples échographies ou d'opérations chirurgicales plus délicates. En fait ces robots ne sont pas complètement autonomes mais ils assistent les médecins ou chirurgiens, jusqu'à permettre des opérations médicales à distance (télémédecine).

On parle de surgé tique (mot né de l'anglais « SURGERY » chirurgie) c'est –à-dire tout ce qui consiste à introduire les derniers outils des technologies informatique et robotique dans la pratique médico-chirurgicale. Cette pratique de « chirurgie assistée » est émergente donc bien que peu répandue, elle est en train de devenir la chirurgie du futur .

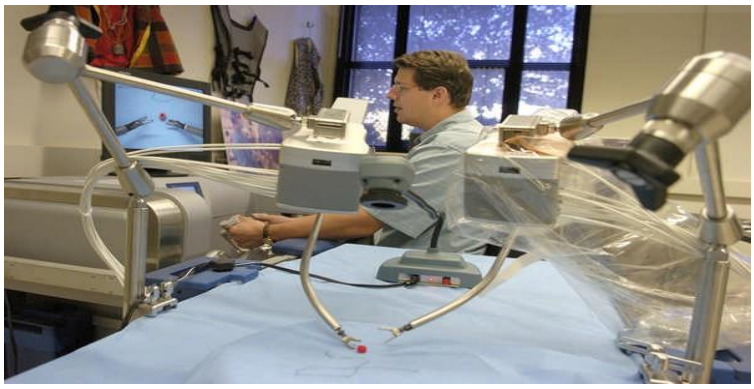


Figure I-25:Usage des robots dans le domaine médical

I.8.3 Domain civil

Par rapport à leur polyvalence, de plus en plus de tâches sont confiées aux robots, voici par exemple YUKITARO, un robot japonais conçu pour déneiger, il est équipé de deux caméras et d'un GPS. Il circule dans la rue de maniéré autonome en engouffrant la neige avec sa « bouche » pelleteuse, puis il la rejette sous forme de blocs de glace [8].

Figure I-26: Robot à usage civile

Les robots sont également utilisés à des fins scientifiques. En témoignent les robots de la NASA Spirit et opportunité, destinés à explorer la planète Mars et à récolter toutes sortes d'informations sur cette planète

Figure I-27:Robot spirit et opportunité

Ces deux robots sont équipés de caméra dont une microscopique, de spectromètres pour étudier la composition minéralogique des roches martiennes, d'un équipement informatique Assimilable à un micro-ordinateur, et d'un outil de perforation des roches pour effectuer des prélèvements. Ces robots peuvent être reprogrammés à distance, ce qui les rend opérationnels à long terme et d'autant plus polyvalents [8].

I.8.4 Domaine de l'usage domestique

La démocratisation de la robotique a conduit, ces dernières années, à voir de nombreux robots s'installer chez les particuliers pour effectuer des tâches à la place de leur possesseur. En effet ceux-ci sont capables de faire le ménage, tondre la pelouse, nettoyer la piscine etc… ce qui conduit certains clients (aisés) à se procurer ces domestiques contemporains**.** Robot aspirateur, fonctionnant de manière autonome Sur tous types de sol. Grâce à ses capteurs, il évite les escaliers, et retourne à ses capteurs, il évite de recharger sans assistance. Il est possible de délimiter une surface précise à nettoyer avec les accessoires fournis.

Il y'a aussi les Robots tondeuse, certes moins efficace que ses homologues manuels, cette tendeuse permet quand même de rafraîchir la pelouse pendant que l'on désherbe les rosiers, ou mieux, pendant que l'on profite du beau temps ou de la piscine [8].

Figure I-28:Usage des robots dans le domaine domestique

I.8.5 Domaine de l'usage industriel :

Un robot industriel est un dispositif poly articulé à l'image d'un bras humain fréquemment composé de 6 degrés de liberté, 3 axes destinés au positionnement et 3 axes à l'orientation servant à déplacer et d'orienter un outil (organe effecteur) dans un espace de travail donné. On peut distinguer trois types de robot industriel, les robots de peinture ou soudure beaucoup utilisés dans l'industrie automobile, les robots de montage de dimension plus fréquemment plus réduite, les robots mobiles conçus pour l'inspection fréquemment associé à de l'intelligence artificielle et capables dans certain cas de prendre en compte l'environnement [8].

Robot de peinture ou soudure

Robot de montage

Robot mobile

Figure I-29:Robot à usage industriel

I.9. Conclusion

Dans ce chapitre, nous avons fait une présentation générale de la robotique, l'historique des robots, leurs structures, leurs domaines d'utilisation ainsi que leurs classification et Type de robot et cela va nous servir pour la présentation de notre bras manipulateur qui va respecter les classifications et les définitions données dans ce chapitre.

II. Chapitre 02 : Présentation du robot « TR5 »

II.1. Introduction

Le TR5 est un robot de haute qualité de formation qui a démontré sa fiabilité depuis plusieurs années dans tous les domaines de l'éducation et de la formation. Tout en faisant l'objet d'un processus continu d'amélioration de la fiabilité du modèle, ce robot à bras de flambage à faible coût n'a rien perdu de son attractivité. Le bras articulé du TR 5 dispose de 5 axes rotatifs et d'une pince parallèle offrant plusieurs possibilités d'applications. Comme le contrôleur est intégré dans la base du robot, une installation et une mise en œuvre faciles et rapides peuvent être assurées. Avec la boîte à thé facile à utiliser et le logiciel correspondant TBPS, le débutant se familiarise rapidement avec ce système et obtient des résultats remarquables. Mais ce n'est pas tout. Pour les utilisateurs déjà plus avancés, il existe le PSI, un système de programmation qui a été testé avec succès dans l'industrie. PSI ne laisse vraiment aucun souhait ouvert.

Figure II-1:Le TR5

II.2. La structure du bras manipulateur TR5

II.2.1 Propriétés mécaniques

La structure TR5 est composée de pièces en aluminium vissées entre elles. Chacun des cinq axes indépendants est lié par un servomoteur à courant continu. La puissance de transmission est assurée par des engrenages. Pour la transmission des axes 3 et 4, des chaines dentées sont utilisées en plus. Les positions absolues de tous les axes sont déterminées par des capteurs potentiométriques. Cette caractéristique-là plus marquante du TR5 garantie que, même après le passage à une autre position, le robot repère la position exacte de chaque axe. Comme il est représenté dans la Figure II.2, le bras manipulateur TR5 possède en total 5 axes de rotations. Le tableau II.1 permet de représenter les différentes limites des angles du bras manipulateur. Les Figures II.3 et II.4 montrent les différentes longueurs des tiges selon 2 angles différents.[9]

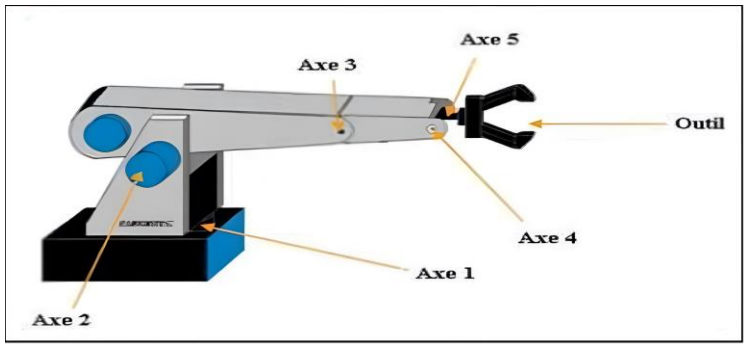


Figure II-2:Les axes du robot TR5

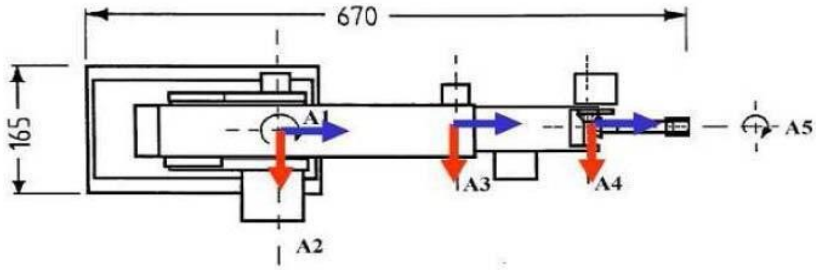


Figure II-3:Les dimensions du robot TR5 (vue de haut)

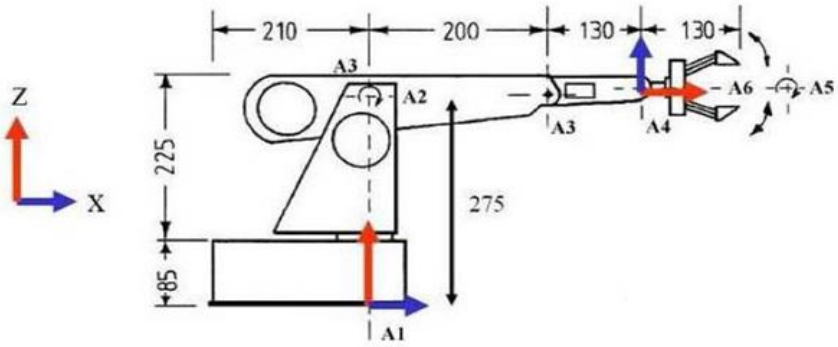


Figure II-4:Les dimensions du robot TR5 (vue de face)

Tableau II-1:Les angles de rotation du robot TR5

Les Articulations	Axes	Limites articulaires	Résolution
Axe 1 : Base	Q0	+80° …0° …-80°	0 … 255
Axe 2 : Epaule	Q1	+50°…0°…-50°	0 … 255
Axe 3 : Coude	Q2	0°… +100°	0 … 255
Axe 4 : poignet	Q3	+100°…0°…-100°	0 … 255
Axe 5 : Orientation	Q4	+100°…0°…-100°	0 … 255
Outil (Pince)		0 … 60 mm	0 … 255

II.2.1.1 La Pince

La pince est alimentée par un servomoteur CC via un câble monté sur ressort. Grâce à cette structure, la position de la pince ainsi que la force de préhension sont librement programmables. En outre, la course extrêmement large de 60 mm permet la manipulation d'une grande variété de pièces sans apporter de modifications aux doigts de la pince. Pour les utilisateurs qui souhaitent manipuler des pièces très fines ayant une surface lisse comme par exemple des feuilles de papier, la pince standard peut être équipée d'une pince à vide.

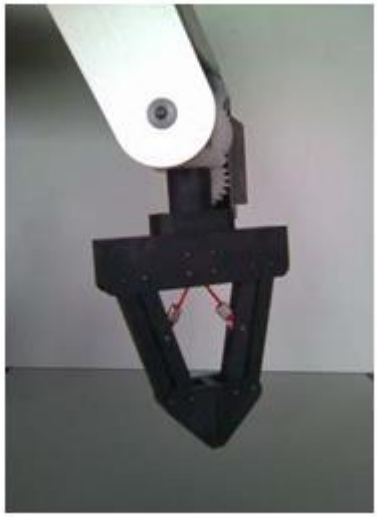


Figure II-5:La pince de TR5

II.2.2 Propriétés électriques

II.2.2.1 Le Contrôleur

Le contrôleur est intégré à la base du robot. Un processeur 8031 est le cœur du contrôleur. En mode PtP (point-à-point), des mouvements d'axes simples ou même de plusieurs axes en même temps peuvent être réalisés. Les programmes utilisateur peuvent être stockés dans une mémoire RAM sauvegardée sur batterie. La capacité de mémoire est de 8 Ko, ce qui correspond à environ 1000 étapes de programme. De plus, le robot peut être connecté à un ordinateur PC ou ATARI via une interface RS 232. 8 entrées et sorties numériques compatibles TTL sont à la disposition de l'utilisateur via un connecteur de port parallèle.

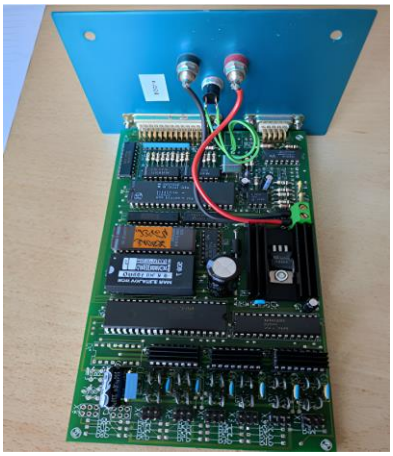


Figure II-6:Contrôleur de TR5

II.2.2.2 L'unité Teachbox

Le Teachbox permet de contrôler et programmer le TR5 sans utiliser un ordinateur, en utilisant un clavier digital avec 25 boutons, répartis en trois groupes :

- Boutons numériques qui sont utilisés pour introduire le numéro de l'axe désigné et la valeur de la position sous forme de valeur décimale.
- Boutons de commandes : pour indiquer la commande à exécuter directement (RUN, STOP, INS, DEL, POS, OUT).
- Boutons d'instructions pour implanter le programme d'instructions de commande, (POS, TIM, OUT, MARK, IF, GOTO, NOP). Une fois l'instruction est introduite, elle est sauvegardée dans la mémoire du robot par ordre d'entrée (FIFO) et elles sont exécutées en mode RUN ou STEP.[9]

Pour programmer le robot, il vous suffit de connecter la Teachbox au port E/S du TR5 ; l'unité peut

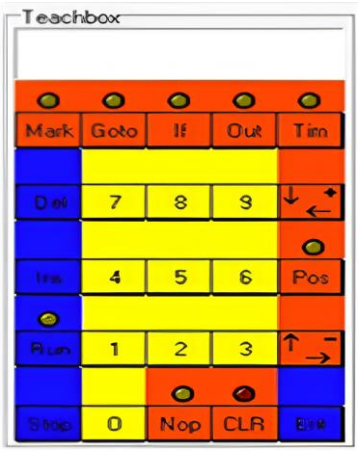


Figure II-7:La Boîte d'apprentissage de TR5

alors être facilement déconnectée une fois le programme démarré. Le connecteur femelle de la Teachbox fournit toutes les entrées et sorties nécessaires à l'interfaçage avec le port E/S du TR5. Lorsque le TR5 est utilisé avec la Teachbox connectée, seules 3 sorties numériques (DO 1-3) et 5 entrées numériques (DI 1-5) sont disponibles. Chaque clé d'instruction est équipée d'une LED qui s'allume pour confirmer la bonne entrée. Si une entrée de commande ou d'instruction est invalide, le voyant ERR s'allume jusqu'à ce que l'entrée incorrecte soit supprimer en appuyant sur CLR. Toutes les instructions et commandes doivent être confirmées avec la touche ENT.[9]

II.2.2.2.1 SPTB

Le système de programmation Teachbox est un logiciel de programmation guidé par menu qui contient essentiellement les fonctions de la Teachbox. Une caractéristique intéressante de SPTB est sa capacité à éditer ultérieurement des programmes écrits via la Teachbox sur l'écran et à les charger ensuite dans la mémoire du robot.

SPTB est disponible pour les PC compatibles IBM et les ordinateurs ATARI. Pour les utilisateurs qui souhaitent développer leurs propres programmes de contrôle de robot, des interfaces logicielles sont disponibles sous forme de fichiers Inclue pour les langages de programmation BASIC, TURBO-PASCAL et C pour les PC compatibles IBM.

II.2.2.2.2 Psi

Le système de programmation pour robots industriels est un langage de programmation extrêmement convivial pour l'utilisateur ambitieux. Les caractéristiques suivantes ne représentent qu'un extrait du jeu d'instructions complet :

- Système de coordonnées d'axe, d'univers et d'outil
- Sous-programmes pratiques
- Variables de programmation
- Fonctions de palettisation
- Diverses fonctions de test du programme

Psi peut être exécuté hors ligne avec toutes les fonctionnalités opérationnelles. Cette alternative peu coûteuse permet de programmer le robot sur plusieurs postes de travail pour un test ultérieur en ligne avec un robot.

Le système d'aide en ligne contextuel permet à l'utilisateur de récupérer à tout moment toutes les informations nécessaires dans la mémoire.

Psi fonctionne sur des PC compatibles IBM**.**

II.2.2.3 Alimentation

L'alimentation de l'énergie pour notre bras robot est assurée par une unité séparée qui fonctionne sous une tension de 220v et fournit une tension maximale égale à 9v.

Figure II-8:Unité d'alimentation de TR5

II.2.2.4 Les capteurs

Plusieurs technologies de construction des capteurs existent. On rencontre dans ce contexte des capteurs incrémentaux et potentiométrique. Pour notre plateforme, les capteurs sont de type proprioceptif, ce sont des capteurs potentiométriques rotatifs placés sur les axes des articulations. Ils détectent la position angulaire des actionneurs. Donc le potentiomètre tourne en même temps que l'axe de sortie.

Figure II-9:Le capteur de robot TR5

Le tableau II.2 donne la relation entre les angles et les tensions correspondante délivrées par le potentiomètre.

Tableau II-2:Caractéristiques des capteurs du robot TR5

Les Articulations	Axes	Limites articulaires	Résolution (volts)
Axe 1 : Base	Q0	+80° …0° …-80°	0…1.37
Axe 2 : Epaule	Q1	+50°…0°…-50°	0.39…5
Axe 3 : Coude	Q2	0°… +100°	2.35…4.31
Axe 4 : poignet	Q3	+100°…0°…-100°	0.98…4.90
Axe 5 : Orientation	Q4	+100°…0°…-100°	0…5
Outil (Pince)		0 … 60 mm	3.62…1.76

II.2.2.5 Les actionneurs

Le bras est muni de six petits moteurs à courant continu, trois moteurs de marque « Bühler » et les trois autres sont des petits servomoteurs « Multiplex ». Cinq d'entre eux sont destinés chacun à faire mouvoir une articulation. En revanche, le dernier est adapté pour ouvrir et fermer la pince.[10]

Les moteurs utilisés sont des moteurs à aimants permanents. Ils fonctionnent avec une tension de 9V générée par une alimentation continue stabilisée et un courant max de 3A. Le moteur électrique à courant continu est le muscle du bras de robot, qui est un dispositif électromécanique servant à convertir de l'énergie électrique d'entrée en énergie mécanique. En effet, il existe plusieurs types de moteurs électriques, on a étudié un seul cas, il s'agit d'un moteur à courant continu à aimants permanents. Ces moteurs qui présentent une très faible inertie mécanique, sont très bien adaptés aux applications nécessitant des variations rapides de vitesse de rotation ou commande en position (périphériques d'ordinateurs, robotique, etc.…). Ils sont toujours commandés par un ensemble électronique comportant une alimentation de puissance avec une électronique de commande réalisant un asservissement.

Classiquement, un moteur à courant continu est constitué d'un stator fixe et d'un rotor mobile. Le courant continu arrive par les cosses à l'extérieur de la carcasse sur des balais souples. Le collecteur est la partie du rotor qui touche les balais. Le rotor est constitué, quant à lui, de plusieurs bobines, toujours en nombres impairs, reliées au collecteur. Lorsque l'une des bobines reçoit le courant continu, elle crée un champ magnétique, mais la carcasse du stator abrite aussi deux aimants de pôles opposés. Le champ magnétique créé attire les pôles des aimants, créant par cela une rotation de l'axe du moteur.[9]

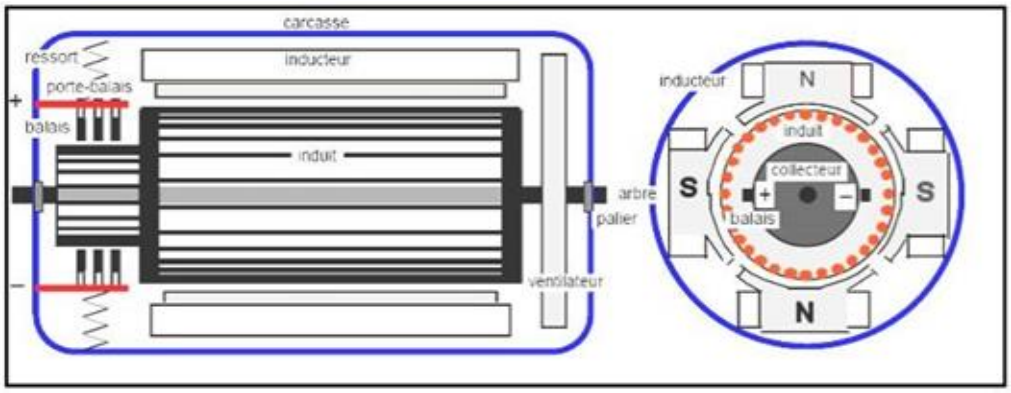


Figure II-10Architecteur d'un MCC à aiment permanent

II.2.2.6 Transmissions et systèmes réducteurs de vitesse

La transmission de mouvement dans notre bras de robot est assurée à l'aide des courroies et des engrenages qui assurent une liaison entre les actionneurs et les articulations surtout au niveau du poignet. Suite aux vitesses élevées des moteurs à courant continu, le système réducteur de vitesse sert tout simplement à réduire la vitesse de rotation de l'axe de sortie.[10]

Figure II-11:transmissions de mouvement avec pignons

II.2.2.7 Câble d'interface

Un câble d'interface série à 9 ou 25 broches permet TR5 à connecter à un PC.

Figure II-12:Câble d'interface

II.2.2.8 Boîte de test d'E/S

Le Test Box permet la simulation et le test des 8 entrées et sorties numériques.

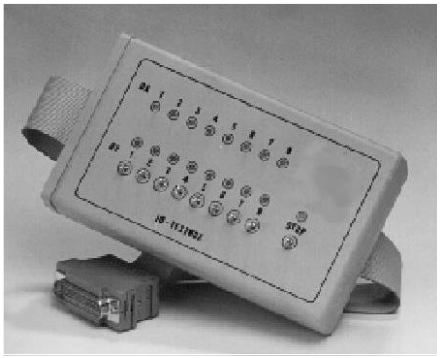


Figure II-13:Boîte de test d'E/S

II.2.2.9 Carte E/S pour PC

Avec 16 entrées numériques opte-isolées et 16 sorties relais sans potentiel (max. 24V/1A).

La carte E/S est conçue uniquement pour être utilisée avec SPTB pour Windows, psi ou la bibliothèque logicielle.

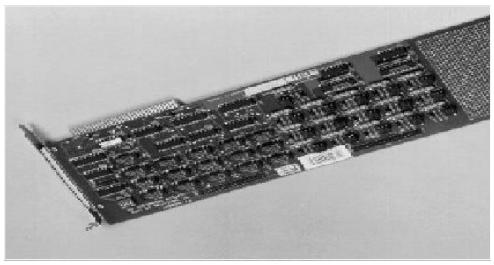


Figure II-14:Carte E/S pour PC

II.3. Représentation géométrique du bras manipulateur TR5

La figure II.9 présente une représentation géométrique du bras manipulateur TR5

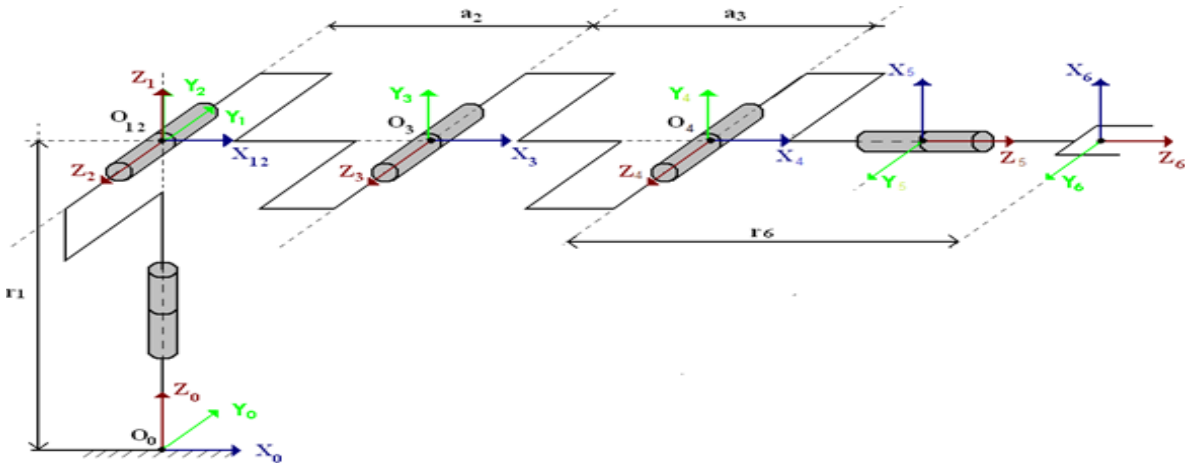


Figure II-15:Représentation géométrique du TR5

II.4. Espace de travail du TR5

La figure II.10 ci-dessous montre l'espace de travail du bras robot. Pour qu'une position soit atteinte par la pince, elle doit être comprise dans cette région figurée.

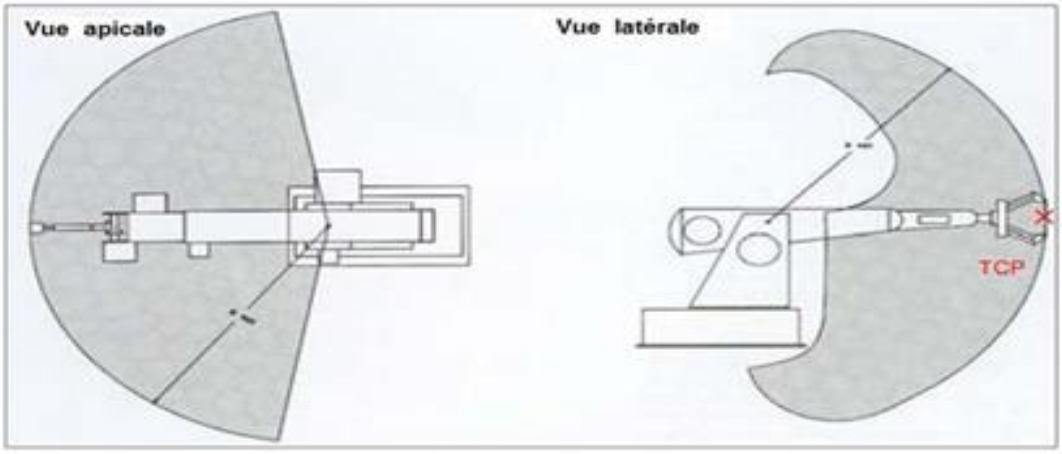


Figure II-16:Espace de travail de bras manipulateur TR5

II.5. Les modes de fonctionnement

Le système de contrôle intégré du robot dispose de six modes de fonctionnement différents :

II.5.1 Mode SAISIE

Pour créer des programmes de contrôle en saisissant une séquence d'instructions. Il s'agit du mode par défaut qui est activé à chaque mise sous tension du système.

II.5.2 Mode POSITION

Ce mode est sélectionné en appuyant sur l'une des touches numériques 1-6, qui sélectionne également l'axe correspondant. L'axe peut, ensuite, être déplacé vers l'avant ou vers l'arrière avec les touches + et -. Dans ce mode, les axes peuvent également être directement positionnés avec la séquence POS (axe a) NOP (position n) ENT. En appuyant sur CLR on revient au mode INPUT.

II.5.3 Mode course

Lorsqu'on actionne la touche RUN (étiquette m) ENT, TR5 commence à exécuter la séquence d'instructions dans sa mémoire en commençant à l'étiquette m.

II.5.4 Mode PAUSE

Lorsque le robot est en mode RUN, toutes les touches sauf la touche STOP sont désactivées. Un appui sur STOP CLR active le mode BREAK. On peut, alors, revenir au mode RUN avec RUN ENT (=CONTINUE). Un appui sur STOP ENT permet de sélectionner l'un des autres modes.

II.5.5 Mode ÉTAPE

Entrée dans EXÉCUTER. (Étiquette m) ENT active le mode STEP à partir de l'étiquette m (à condition que l'étiquette m existe, bien sûr). On peut ensuite parcourir le programme une étape à la fois en appuyant sur la touche +. L'appui sur CLR quitte le mode STEP.

II.5.6 Mode d'affichage

L'entrée de STOP NOP (étiquette m) ENT fait passer le système en mode AFFICHAGE, en commençant par l'étiquette m. Lorsqu'on appuie sur la touche +, l'instruction suivante stockée dans la séquence en mémoire s'affiche (la LED correspondante sur la Teachbox s'allume), mais l'instruction n'est pas exécutée sur la TR5. appui sur CLR quitte le mode DISPLAY.

II.6. Conclusion

En conclusion, le robot TR5 se distingue par ses caractéristiques mécaniques, électriques et une programmation de haute qualité. Doté de 5 axes rotatifs, d'une pince programmable, d'un contrôleur intégré et de divers modes de fonctionnement, il constitue une solution complète pour l'apprentissage et la formation dans le domaine de la robotique. Sa facilité d'utilisation, la précision de ses capteurs potentiométriques et les fonctionnalités avancées de programmation, telles que le PSI, en font de lui un outil polyvalent et performant. En somme, le robot TR5 est une solution fiable et efficace pour répondre aux besoins en matière de formation et de programmation robotique.

III. Chapitre 03 : Software et hardware

III.1. Introduction

Ce chapitre se concentre sur la présentation détaillée du matériel et du logiciel utilisés dans le cadre de la réalisation d'un projet basé sur l'Arduino. Ce chapitre explore en profondeur les composants matériels de la carte Arduino Uno, les caractéristiques du microcontrôleur ATMega328, ainsi que les fonctionnalités du logiciel Arduino. En outre, il aborde également l'utilisation d'un joystick comme organe de commande et la description du driver L298 pour le contrôle des moteurs. Ce chapitre vise à fournir une compréhension approfondie du matériel et du logiciel nécessaires pour la mise en œuvre de projets électroniques et robotiques.

III.2. Présentation de l'Arduino

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

L'Arduino est une plateforme de contrôle, elle est constituée de deux parties :

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

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

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

etc.…

III.2.1 La partie Hardware

III.2.1.1 Constitution d'une carte Arduino UNO

La carte Arduino Uno est basée sur un ATMega328 cadencé à 16 MHz. Des connecteurs situés sur les bords extérieurs du circuit imprimé permettent d'enficher une série de modules complémentaires. Elle est une carte électronique de prototypage rapide sur laquelle on peut raccorder des capteurs (entrées) et des actionneurs (sorties). Elle est constituée de plusieurs composants électroniques dont le principal est un microcontrôleur permettant de stocker et d'exécuter un programme informatique.

Pour pouvoir le programmer, il suffit de la connecter à un ordinateur à l'aide d'un câble USB, puis vérifiez le port, après cela, le code peut être envoyé à l'Arduino.

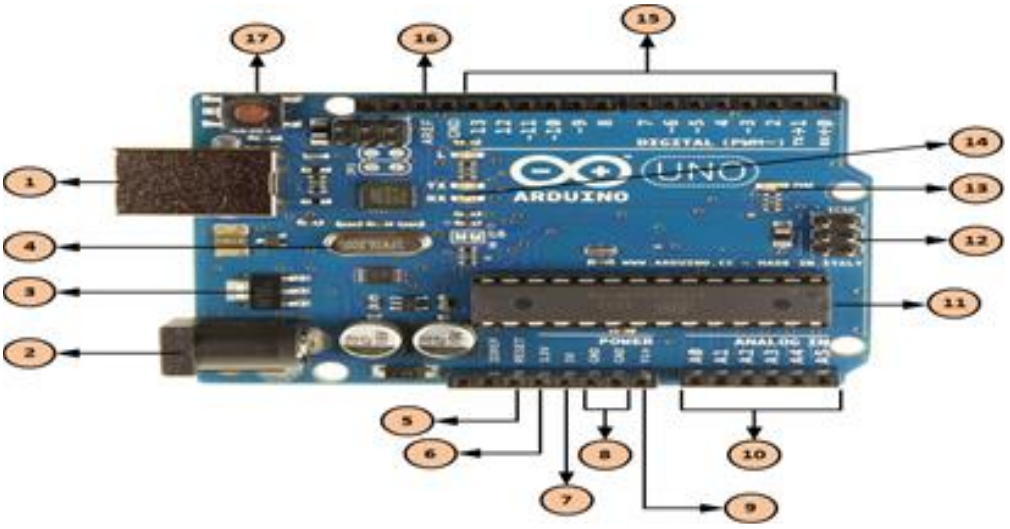
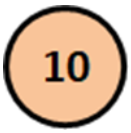


Figure III-1:Constitution de la carte Arduino UNO

Tableau III-1:Les différentes broches de la carte ARDUINO UNO

1	**Alimentation / Programmation par USB** La carte Arduino peut être alimentée avec un câble USB relié à l'ordinateur. Tout ce qu'il faut, c'est de connecter la carte Arduino à l'ordinateur avec le câble USB type A/B.
2	**Alimentation via connecteur Jack DC** **Diamètre interne 2.1mm, externe 5.5mm** La carte Arduino peut être directement alimentée par ce connecteur Jack DC. Ce connecteur (2) est relié au régulateur de tension intégré à la carte. L'alimentation via ce connecteur (2) doit être comprise entre 5 et 12 V
3	**Régulateur de tension** La fonction du régulateur de tension (3) est de contrôler la tension d'alimentation de l'Arduino pour la stabiliser à la bonne tension du microcontrôleur et de chaque élément de la carte. La tension de stabilisation est de 5 Volts sur les cartes UNO.
4	**Oscillateur à quartz** Un oscillateur à quartz est un élément électronique qui a la particularité de posséder un quartz à l'intérieur qui vibre sous l'effet piézoélectrique. Les propriétés électromécaniques du quartz sont telles qu'on arrive à fairevibrer le quartz à une fréquence très précise. Cet élément aide l'Arduino UNO à calculer les données de temps. Sur le dessus du composant, onpeut lire 16.000H9H. Cela signifie que la fréquence est de 16, 000,000 Hertz, soit 16 MHz.

	Arduino Reset L'Arduino peut être redémarré avec un "Reset". Cela aura pour effet de relancer le programme depuis le début. Le redémarrage de l'Arduino UNO peut se faire de deux manières : soit en utilisant le bouton "Reset" (17), soit en connectant un bouton externe sur la broche de lacarte Arduino mentionnée "RESET" (5).
	Broches (3.3, 5, GND, Vin) 3.3V (6) – Broche d'alimentation de tension 3.3 Volts 5V (7) – Broche d'alimentation de tension 5 Volts La plupart des composants destinés à fonctionner avec Arduino fonctionnent bien en 3.3 Volts ou 5 Volts. GND (8) (Ground / Masse) – Il y a plusieurs broches de ce type présentes sur la carte Arduino, elles sont toutes communes et peuvent être utilisées comme masse (potentiel 0 Volts) pour les circuits. Vin (9) – Cette broche permet d'alimenter l'Arduino depuis une source de tension extérieure. Elle est reliée au circuit d'alimentation principale de la Carte Arduino.
	**Broches analogiques** L'Arduino UNO possède 5 broches d'entrées analogiques numérotée de A0 jusqu'à A5. Ces broches permettent de lire un signal analogique d'un capteur comme un capteur d'humidité ou de température. La carte Arduino utilise un convertisseur analogique/numérique (convertisseur CAN) pour permettre la lecture du signal par le microcontrôleur. Un signal sera converti sur 10 bits. La valeur pourra être lue sur une echelle 1024 points.
	Microcontrôleur principal Chaque carte Arduino possède son propre microcontrôleur (11). Il peut être considéré comme le cerveau de la carte Arduino. Le microcontrôleur sur l'Arduino est légèrement différent d'une carte à l'autre. Les microcontrôleurs sont généralement de la société ATMEL. Il faut savoir quel est le microcontrôleur de la carte utilisée, avant de charger un nouveau programme depuis l'IDE Arduino. Cette information est disponible directement sur le composant.

12	**Connecteur ICSP** Avant tout, le connecteur ICSP (*In-Circuit Serial Programming*) est une connectique AVR comprenant les broches MOSI, MISO, SCK, RESET, VCC et GND. Il s'agit d'un connecteur de programmation. Ce connecteur permet entre autres de programmer directement le microcontrôleur sur les couches les plus basses (boot loader, code ASM…). C'est aussi un port appelé port SPI (Serial Peripheral Interface), qui permet de dialoguer avec d'autres composants SPI (écrans, capteurs, etc.…).
13	**Indicateur LED d'alimentation** Ce voyant doit s'allumer lorsque l'Arduino est connecté à une source d'alimentation pour indiquer que la carte est correctement alimentée. Si cette lumière ne s'allume pas, il y a un problème avec l'alimentation.
14	**LEDs TX et RX** Sur la carte se trouve deux indicateurs : TX (émission) et RX (réception). Ils apparaissent à deux endroits sur la carte Arduino UNO. Tout d'abord, sur les broches numériques 0 et 1, pour indiquer les brochesresponsables de la communication série. Deuxièmement, les Leeds TX et RX (13). Le voyant TX clignote à une vitesse variable lors de l'envoi des données série. La vitesse de clignotement dépend de la vitesse de transmission utilisée par la carte. RX clignote pendant le processus de réception. La vitesse de transmission s'exprime en bauds, soit l'équivalent du bits/seconde si le signal est binaire.
15	**Entrées/Sorties numériques** La carte Arduino UNO possède 14 broches d'Entrées / Sorties numériques (15), dont 6 peuvent fournir une sortie PWM (Pulse Width Modulation). Ces broches peuvent être configurées pour fonctionner comme des broches numériques d'entrée pour lire des valeurs logiques (0 ou 1) ou numériques. Elles peuvent également être utilisées comme des broches de sortie pour piloter différents modules comme des Leeds, des relais, etc. Les broches Étiquetées "~" peuvent être utilisées pour générer des PWM.
16	**Broche AREF** AREF est l'acronyme anglais de "référence analogique". Cette broche est parfois utilisée pour définir une tension de référence externe (entre 0 et 5 Volts) comme limite supérieure pour les broches d'entrée analogiques.

III.2.1.2 Caractéristiques de la carte Arduino UNO

Dans le tableau III.2, les caractéristiques de la carte Arduino UNO sont présentées.

Tableau III-2:Caractéristiques de la carte Arduino UNO

Microcontrôleur	ATmega328
Tension d'alimentation interne	5V
Tension d'alimentation (recommandée)	6-20V
Entrées/sorties numériques	14 dont 6 sorties PWM
Entrées analogiques	6
Courant max par broches E/S	40 mA
Courant max sur sortie 3,3V	50mA
Mémoire Flash	32 KB
Mémoire SRAM	2 KB
Mémoire EEPROM	1 KB
Fréquence horloge	16 MHz
Dimensions	68.6mm x 53.3mm

III.2.1.3 Brochage de la carte Uno

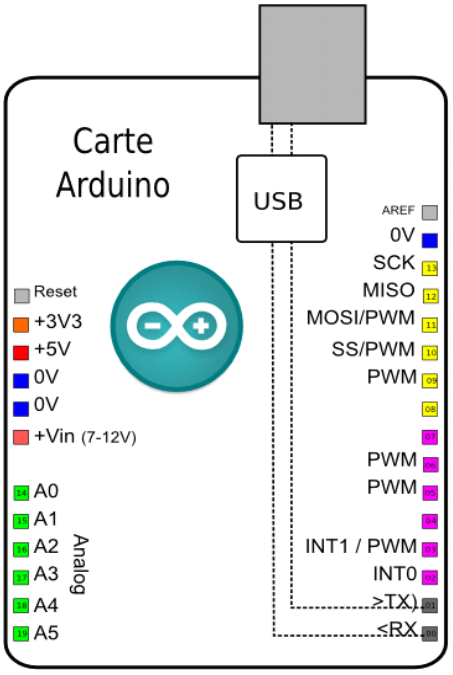


Figure III-2:Brochage de la carte Arduino Uno

III.2.1.4 Microcontrôleur ATMEL ATMega328

Le microcontrôleur de la carte Arduino UNO est un ATMega328. C'est un microcontrôleur ATMEL de la famille AVR 8bits.

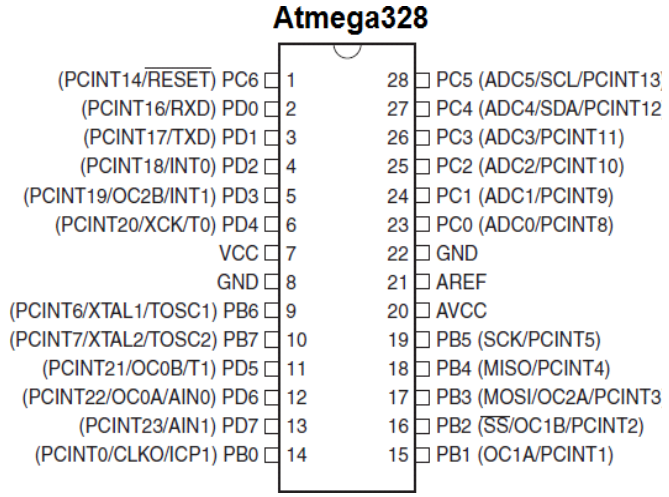


Figure III-3:Microcontrôleur ATMega328

III.2.1.4.1 Les principales caractéristiques d'atmega328

Tableau III-3:Les caractéristiques d'atmega328

Mémoire FLASH	Mémoire programme de 32Ko
Mémoire SRAM	Données 2Ko
Digital I/O (entrées-sorties Tout Ou Rien)	3 ports Port B, Port C, Port D
Mémoire EEPROM	Données 1Ko
Timers	Timer0 et Timer1Timer2
PWM	6 broches OC0A(PD6), OC0B(PD5), 0C1A(PB1), OC1B(PB2), OC2A(PB3), OC2B(PD3)
Analog to Digital Converter	6 entrées multiplexes ADC0(PC0) à ADC5(PC5)
Gestion bus I2C	SDA(PC5) /SCL(PC4).
Port série (USART)	TXD(PD1) /RXD(PD0)
Comparateur Analogique	Broches AIN0(PD6) et AIN1 (PD7)

III.2.2 La partie software (Logiciel Arduino)

Pour commander Arduino Méga 2560 sur PC on installe le logiciel du même nom Arduino (version 1.8.5 dans notre cas) qui est une application Java et qui a l'interface présentée dans la figure II.4, ci-dessous :

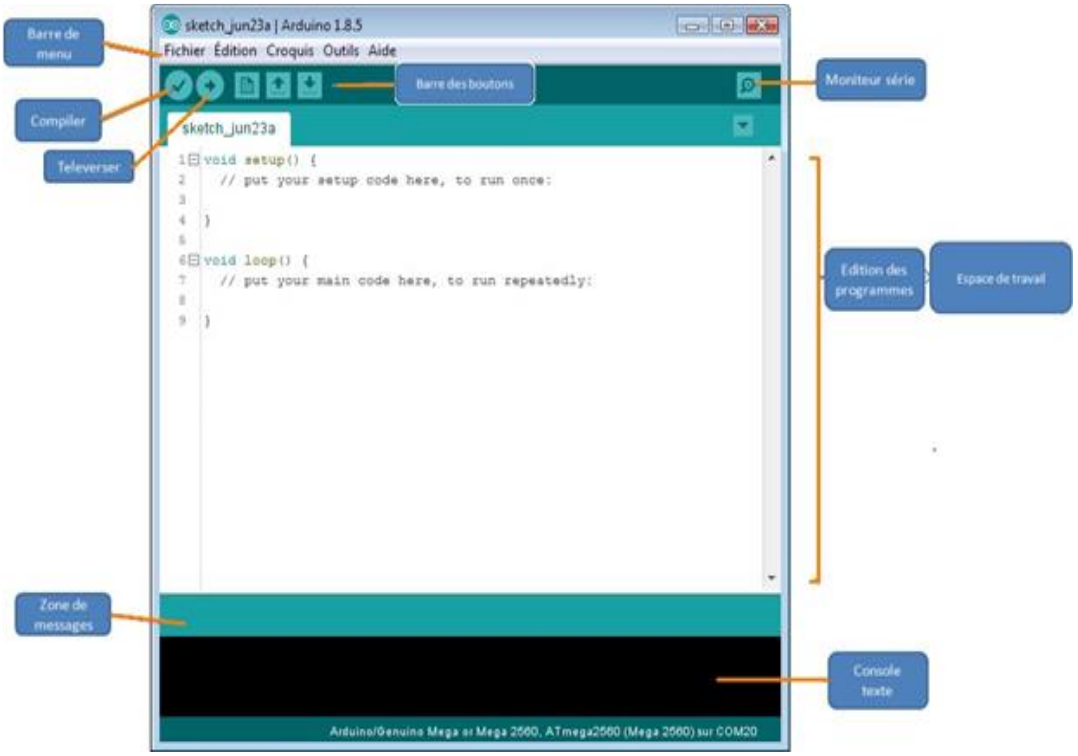


Figure III-4:Vue d'ensemble et interface du logiciel Arduino

Le logiciel Arduino a pour fonctions principales :

- De pouvoir écrire et compiler des programmes pour la carte Arduino.
- De se connecter avec la carte Arduino pour y transférer les programmes.

L'interface du logiciel contient une BARRE DE MENU comme tous les logiciel, Un MONITEUR SERIE pour afficher sur l'ordinateur les états des variables, des résultats de calculs ou de conversions analogique-numérique. Un EDITEUR (l'espace de travail) pour écrire le code (programme), le logiciel obéit à quelques notions pour pouvoir bien structurer le programme afin de le compiler et d'éviter les erreurs de syntaxe et autres. Une ZONE DE MESSAGES qui affiche l'état des actions en cours, et une CONSOLE TEXTE qui affiche les messages concernant le résultat de la compilation du programme.

Cet espace de développement intégré (EDI) dédié au langage Arduino et à la programmation des cartes Arduino comportent aussi :

Une BARRE DE BOUTONS qui donne l'accès direct aux fonctions essentielles du logiciel tel que :

Tableau III-4:Barre d'Action.

	Bouton Vérifié (Compiler) : il permet de vérifier le programme pour trouver d'éventuelles erreurs. Cette procédure prend un certain temps d'exécution et lorsque elle est terminée, elle affiche un message de type « Binary sketch size: »
	Bouton Upload (Téléverser) : ce bouton permet de compiler et charger leprogramme sur la carte Arduino.
	Bouton New (Nouveau) : ce bouton permet d'ouvrir une nouvelle fenêtre de programmation.
	Bouton Open (Ouvrir) : il fait apparaître un menu qui permet d'ouvrir un programme qui figure dans le dossier de travail ou des exemples de programmes intégrés au logiciel.
	Bouton Save (Sauvegarder) : il permet de sauvegarder le programme.
	Bouton Serial Monitor (Moniteur série) : ce bouton fait apparaître le moniteursérie.

Une BARRE DES ERREURS : pour afficher les erreurs faites au cours du programme, comme l'oubli d'un point-virgule, le manque d'une accolade ou toute autre erreur dans les instructions. Un exemple de ce genre s'affiche en cas d'erreurs :

Figure III-5:Message d'Erreur

III.2.2.1 Structure du programme

Le programme comporte trois phases consécutives :

III.1.2.1.1 La Définition des constantes et des variables : Cette partie est optionnelle uniquement pour la déclaration des variables et des constantes chaque entrée et sortie est définie et déclarée, en lui donnant un nom arbitraire et en lui affectant le numéro de l'entrée ou celui de la sortie voulue, sans oublier de préciser le type de la variable.

III.1.2.1.2 Configuration des entrées/sorties : Les instructions viennent après le void setup (), après avoir ouvert une accolade, on peut manipuler les broches de la carte en les configurant comme étant des entrées ou des sorties, selon les besoins. Les entrées analogiques pour les capteurs par exemple, ne sont soumises à aucune configuration, car la carte possède 16 entrées analogiques qui

ne font que cela.

III.1.2.1.3 Programmation des interactions et comportements : Celles-ci viennent après le void loop (), c'est la partie principale, ou on rédige les instructions et les opérations comme la lecture des données, les boucles, les affectations...etc. Chacune d'elles doit obligatoirement se terminer par un point-virgule.

III.1.2.1.4 Les commentaires : Comme chaque IDE, des commentaires peuvent être ajoutés au programme. Dans la configuration des entrées/sorties, les commentaires doivent être écrits après un slash « / » ou une étoile « * » ou les deux, tandis que sur une ligne de code, on les écrit après deux slash « // ». [11]

III.3. Joystick (organe de commande)

Un joystick est un périphérique d'entrée constitué d'un manche posé sur un socle. Des boutons-pressoirs sont également accessibles sur le manche ou le socle. L'utilisateur peut agir sur le périphérique soit en bougeant le manche dans une direction, soit en pressant les boutons. C'est un appareil à commande manuelle, qui offre un moyen pratique d'obtenir l'entrée de l'opérateur. Il se compose fondamentalement de deux potentiomètres de 10kΩ et d'un bouton-poussoir. Ce périphérique d'entrée s'interface avec l'Arduino via cinq broches. Trois d'entre elles sont des entrées pour l'Arduino, tandis que les deux autres sont pour la tension d'alimentation et la masse.

Dans notre réalisation on a utilisé le type Break out bouclier. Comme illustré dans la figure II.6, ci-dessous :

Figure III-III-6:Joystick de contrôle

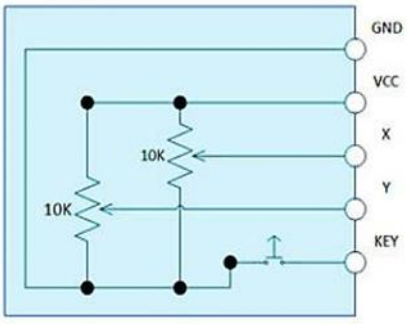


Figure III-III-7:schéma interne d'une Joystick

III.4. Le Driver L298

III.4.1 Description L298xx (H-Bridge Motor Driver)

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

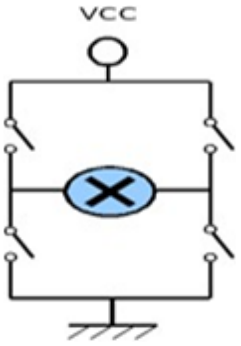


Figure III-8:Circuit d'un pont en H.

Ce circuit qui est un double Pont-H, très populaire et moins couteux, Il peut contrôler deux moteurs à courant continu ou un moteur pas-à-pas 4 fils 2 phases. Il est conçu pour supporter des tensions plus élevées (6 à 30 volts), des courants importants (environ 2A) tout en proposant une commande logique TTL (basse tension, courant faibles, idéal donc pour un microcontrôleur). [12]

Les deux types de moteurs peuvent être contrôlés en vitesse constante et aussi bien en vitesse variable qu'en direction. Toutes les sorties en puissance sont déjà protégées par des diodes anti-retour (selon la version et le modèle choisi). Il peut aussi piloter des charges inductives comme des relais et des bobines ou pour contrôler la luminosité de certains projets utilisant des LED par exemple.

De plus, ce pilote L298xx peut se trouver en module prêt à l'emploi, il inclut des diodes pour protéger le circuit, des résistances de rappel et un dissipateur de chaleur en cas de forte charge. Ce module est extrêmement utile, il Offre un bon moyen de piloter et contrôler des robots et un ensemble mécanisé.

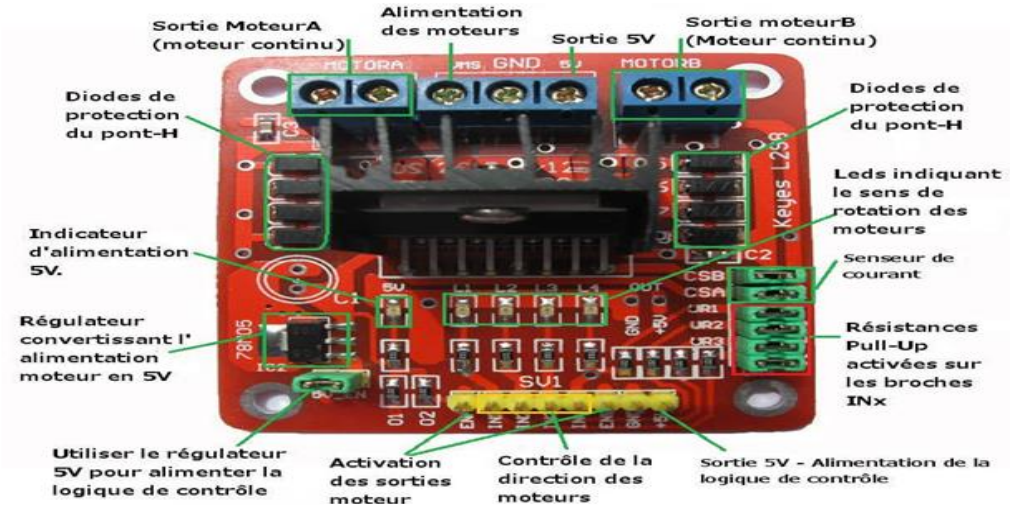


Figure III-9:Module à base de circuit L298

III.4.2 Les Caractéristiques du L298N

- ✓ Tension d'entrée : 3,2 V~ 40 Vdc.
- ✓ Pilote : Pilote de moteur CC à double pont-H L298N.
- ✓ Alimentation : CC 5 V - 35 V.
- ✓ Courant de crête : 2 A.
- ✓ Plage de courant de fonctionnement : 0 ~ 36 mA.
- ✓ Plage de tension d'entrée du signal de commande :
- Faible : -0,3 V ≤ Vin ≤ 1,5 V.
- Élevé : 2,3 V ≤ Vin ≤Vss.
- ✓ Plage de tension d'entrée du signal d'activation :
- Faible : -0,3 ≤ Vin ≤ 1,5 V (le signal de commande est invalide).
- Haut : 2,3 V ≤ Vin ≤Vss (signal de commande actif).
- ✓ Consommation électrique maximale : 20W (lorsque la température T = 75 ℃).
- ✓ Température de stockage : -25 ~ +130 ℃.
- ✓ Alimentation de sortie régulée +5 V intégrée (alimentation à la carte contrôleur, c'est- à-dire Arduino).
- ✓ Taille : 3,4 cm x 4,3 cm x 2,7 cm.
- ✓ Compatible TTL (peut donc être commandé directement avec Arduino).
- ✓ Dispose d'un dispositif de mesure du courant (sens or/sens pins).
- ✓ A raccorder à la masse si on ne l'utilise pas.

Le seul inconvénient avec ce circuit de puissance c'est qu'il ne contient pas de protection incorporée, donc il va falloir qu'on lui ajoute des diodes rapides entre le L298 et les bobines du moteur, ces diodes vont récupérer le courant et ainsi assurer la protection de notre étage de puissance.

III.4.3 Principe de fonctionnement du pilote (Command du L298N)

Le L298N est un double pont en H, c'est à dire qu'il permet de faire tourner les moteurs dans un sens ou dans l'autre sans avoir à modifier les branchements, grâce à sa forme de H, d'où il tient

son nom et qui lui permet de faire passer le courant soit dans un sens ou dans l'autre.

Les interrupteurs fonctionnent deux par deux. Le A est associé au D et le B est associé au C.

Dans le cas où tous les interrupteurs sont ouverts (ils ne laissent pas passer le courant), donc le moteur est arrêté.

Voyons maintenant ce qui arrive lorsque nous actionnons en même temps les interrupteurs A et D (schéma de gauche), ou les interrupteurs B et C (schéma de droite) selon la figure III.10**.** Sur le schéma de gauche, les interrupteurs A et D sont fermés, donc le courant entre par la patte gauche du moteur et sort par sa droite et par conséquent le moteur tourne.

Sur le schéma de droite, les interrupteurs B et C sont fermés, donc le courant entre par la patte droite du moteur et sort par sa gauche ce qui fait tourner le moteur dans le sens inverse.

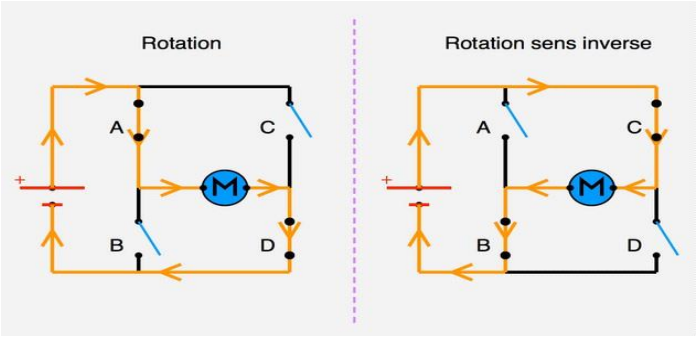


Figure III-10:Fonctionnement du double Pont en H.

On remarque que les transistors fonctionnent deux par deux, en effet, si on ferme juste un seul et laisse ouvert les trois autres le courant n'a nulle part où aller et rien ne se passe, le moteur est en roue libre. Maintenant, si on ferme A et C en laissant B et D ouverts, cette action va créer ce que l'on appelé un frein magnétique. Le moteur se retrouve alors court-circuité. En tournant à cause de son inertie, le courant généré va revenir dans le moteur et va le freiner.

III.5. Présentation générale de Proteus (ISIS et ARES) :

Proteus est une suite logicielle destinée à l'électronique. Développé par la société Labcenter Electronics, les logiciels incluent dans Proteus permettent la CAO dans le domaine électronique. Deux logiciels principaux composent cette suite logicielle : ISIS, ARES, PROSPICE et VSM.

Cette suite logicielle est très connue dans le domaine de l'électronique. De nombreuses entreprises et organismes de formation (incluant lycée et université) utilisent cette suite logicielle. Outre la popularité de l'outil, Proteus possède d'autres avantages

- ✓ Pack contenant des logiciels faciles et rapides à comprendre et à utiliser
- ✓ Le support technique est performant
- ✓ L'outil de création de prototype virtuel permet de réduire les coûts matériel et logiciel lors de la conception d'un projet. [13]

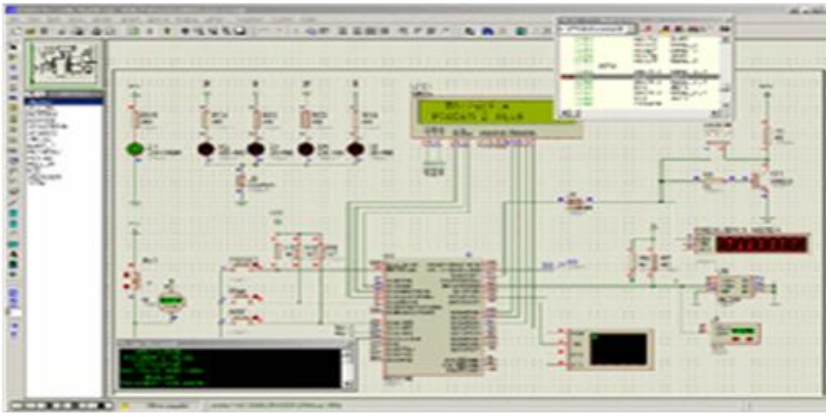


Figure III-11:Le logiciel proteus

III.5.1 L'éditeur de schéma ISIS

Le logiciel ISIS de Proteus Professional est principalement connue pour éditer des schémas électriques. Par ailleurs, le logiciel permet également de simuler ces schémas ce qui permet de déceler certaines erreurs dès l'étape de conception. Indirectement, les circuits électriques conçus grâce à ce logiciel peuvent être utilisés dans des documentations car le logiciel permet de contrôler la majorité de l'aspect graphique des circuits.

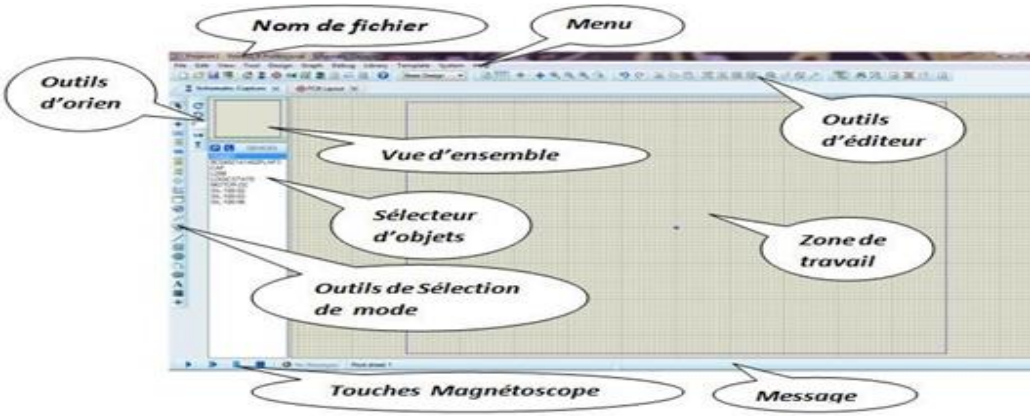


Figure III-12:Interface utilisateur ISIS.

III.5.1.1 Interface utilisateur :

III.5.1.1.1 Barre de menus

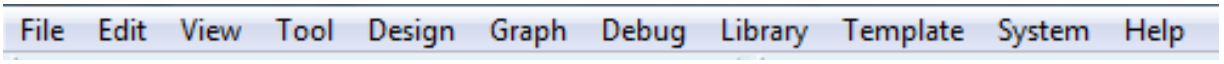


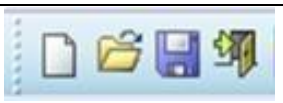
Figure III-13:Barre de menu

Cette barre permet de gérer le travail (ouverture, sauvegarde, impression, mode d'affichage, etc.).

III.5.1.1.2 Barres d'outils de commande

Ces barres fournissent un accès équivalent aux commandes des menus. Elles peuvent être masquées par la commande "Barre d'outils" du menu "Affichage".

Tableau III-5:Barres d'outils de commande

Commandes Fichier / Projet	
Commande d'affichage	
Commandes Édition / Bibliothèque	
Commandes Outils	

III.5.1.1.3 Barre d'outils de sélection de mode

Cette barre permet de sélectionner un outil parmi les 3 modes d'édition disponibles.

Mode Principal		Mode Gadgets		Mode Graphique	
	Composants		Terminal		Ligne
	Point de jonction		Patte de composant		Rectangle
	Label de fil		Graphe		Cercle
	Script de texte		Cassette		Arc
	Bus		Générateurs		Chemin
	Sous circuit		Sonde de tension		Texte
	Édition		Sonde de courant		Symbole
			Appareils		Marqueur origine

Figure III-14 : Barre d'outils de sélection de mode

III.5.1.1.4 Barre d'outils d'orientation

Cette barre permet d'afficher et de contrôler la rotation et la réflexion d'un objet placé ou à placer.

Figure III-15 : Barre d'outils d'orientation

III.5.1.1.5 Zone de travail ou d'édition des schémas

Cet espace rectangulaire correspond à la zone de travail effectif. Tous les schémas apparaîtront dans cette zone et seront visualisés avec le coefficient d'agrandissement ou de réduction choisi.

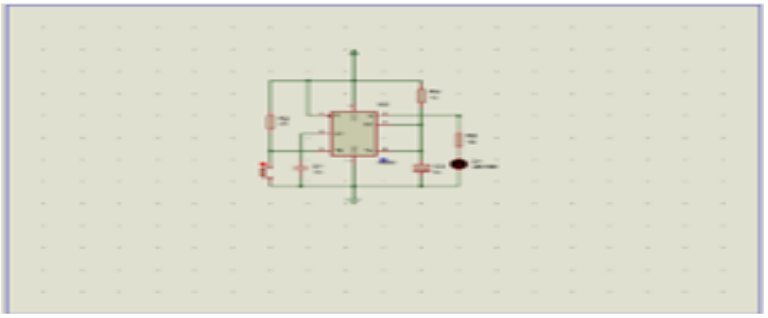


Figure III-16:Zone de travail ou d'édition des schémas

III.5.1.1.6 Vue d'ensemble du schéma (cadre extérieur) et de positionnement (cadre intérieur)

Elle montre une représentation simplifiée de la totalité du dessin. Le cadre bleu marque le contour de la feuille, alors que le cadre vert montre la zone du schéma actuellement visible dans la fenêtre d'édition. Dans cette fenêtre, apparaît également l'aperçu d'un objet sélectionné pour un placement.

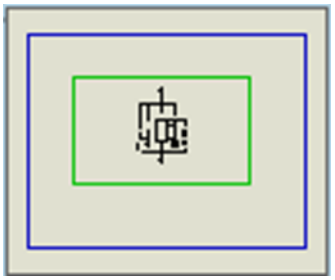


Figure III-17: Vue d'ensemble du schéma et de positionnement

III.5.1.1.7 Sélecteur d'objets

Le sélecteur d'objets liste les différents éléments, selon le mode de travail choisi. Les types d'objets qui peuvent y apparaître sont les composants, les terminaux, les pattes, les symboles graphiques, les marqueurs.

Figure III-18:
Sélecteur d'objets

III.5.1.2 Etapes de la saisie de schéma :

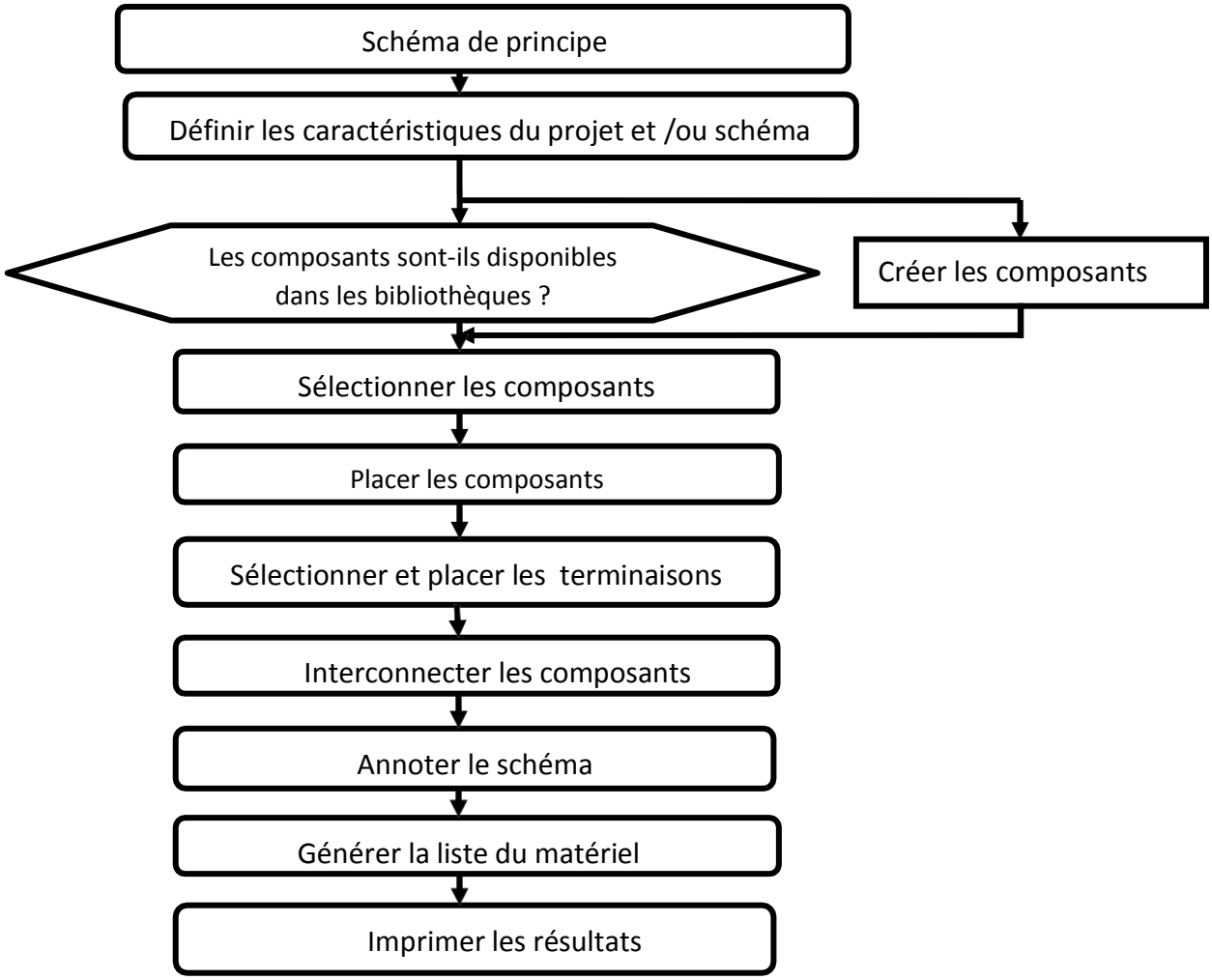


Figure III-19:Organigramme des Etapes de la saisie de schéma.

III.5.2 Ares

Le logiciel ARES est un outil d'édition et de routage qui complète parfaitement ISIS. Un Schéma électrique réalisé sur ISIS peut alors être importé facilement sur ARES pour réaliser le PCB (Printed circuit board) de la carte électronique. Bien que l'édition d'un circuit imprimé soit plus efficiente lorsqu'elle est réalisée manuellement, ce logiciel permet de placer automatiquement les composants et de réaliser le routage automatiquement.

III.6. Conclusion

Ce chapitre a permis d'explorer en détail les aspects matériels et logiciels essentiels à la réalisation de projets basés sur l'Arduino. En mettant en lumière les caractéristiques de la carte Arduino Uno, du microcontrôleur ATMega328, du joystick comme organe de commande, et du driver L298 pour le contrôle des moteurs, ce chapitre a fourni une base solide pour la compréhension et la manipulation de ces composants clés. Grâce à une analyse approfondie du matériel et du logiciel présentés, les lecteurs sont désormais mieux équipés pour se lancer dans la conception et la mise en œuvre de projets électroniques et robotiques utilisant l'Arduino.

IV. Chapitre 04 : La commande et la simulation de projet

IV.1. Introduction

Dans ce chapitre, nous expliquons la raison d'être et les objectifs de notre projet, ainsi que le choix des composants et la conception de la carte de commande. Cette réalisation vise à créer une carte de commande en puissance basée sur un driver L298, contrôlé par la carte Arduino UNO, pour piloter les déplacements du bras manipulateur TR5. Un programme a été développé pour faire fonctionner l'ensemble des équipements connectés à ce système.

IV.2. Présentation de projet

Le robot sur lequel nous allons travailler est un bras manipulateur à 6 degrés de liberté TR5. Ces modèles de bras robotisés sont largement utilisés dans l'industrie, la recherche et l'enseignement. Il est composé d'une carte de commande et d'un logiciel. Le robot disponible au niveau du laboratoire de recherche d'automatique appliquée est actuellement en panne (absence du logiciel de commande et de la documentation).

Ce projet vise essentiellement à réhabiliter le bras manipulateur TR5 et le rendre opérationnelle pour réaliser les travaux pratique de la robotique ; cette réhabilitation concerne la partie hardware et la partie software du robot. Pour la partie hardware, nous allons développer et réaliser une carte de commande basée sur la carte de prototypage open source Arduino. En ce qui concerne la partie software, nous allons proposer une interface de commande qui permettra le développement des programmes de commande du robot sous Matlab.

IV.2.1 Limite en commande

Le robot TR5 a un système de contrôle intégré à base d'un microprocesseur Intel 8031, ce dernier permet de commander le manipulateur uniquement en boucle ouverte. Dans le cadre de notre travail, nous nous sommes rendu compte que la carte de commande originale était défectueuse, outre l'absence des références des différents circuits électroniques composants, cette situation nous a rendu la compréhension du fonctionnement de la carte originale impossible.

Ces facteurs nous ont obligé à concevoir une nouvelle carte de commande qui permet de contrôler le système en introduisant le calcul dynamique de l'erreur c'est-à-dire la commande en boucle fermé.

IV.3. Description de Schéma proposé :

Le schéma représente un système de contrôle d'un bras robotisé utilisant joystick, une carte Arduino et un module de contrôle de moteur L298N. Voici une explication détaillée des différents composants et de leurs interconnexions :

1. **Ordinateur :**

L'ordinateur est utilisé pour programmer et contrôler la carte Arduino. Il envoie des instructions à l'Arduino via une connexion USB.

2. **Arduino :**

La carte Arduino reçoit les instructions de l'ordinateur et les exécute. Elle est le cerveau du système, contrôlant les mouvements du bras robotique en envoyant des signaux au module de commande de moteur.

3. **Joystick :**

Le joystick est utilisé comme dispositif d'entrée pour contrôler le mouvement du bras robotique. Il

envoie des signaux analogiques en fonction de la direction et de l'intensité du mouvement appliqué par l'utilisateur.

4. **Le module L298N :**

Le module de commande de moteur L298N est utilisé pour contrôler les moteurs du bras robotique. Il reçoit des signaux de l'Arduino et alimente les moteurs en conséquence. Ce module permet de contrôler la direction et la vitesse des moteurs.

5. **Alimentation :**

Une source d'alimentation externe est nécessaire pour fournir l'énergie requise par le module de commande de moteur L298N et les moteurs du bras robotique. Cette alimentation est connectée au module L298N.

6. **Bras robotique (TR5) :**

Le bras robotique est l'élément final du système. Il est équipé de moteurs qui permettent de déplacer ses différentes parties. Les moteurs sont contrôlés par le module L298N, qui reçoit des instructions de l'Arduino.

- **Flux de contrôle**

1. Le joystick envoie des signaux à l'Arduino.
2. L'Arduino traite ces signaux et envoie des commandes au module L298N.
3. Le module L298N contrôle les moteurs du bras robotique en fonction des signaux reçus de l'Arduino.
4. L'alimentation externe fournit l'énergie nécessaire au module L298N et aux moteurs du bras robotique.
5. L'ordinateur peut être utilisé pour programmer et surveiller l'Arduino.

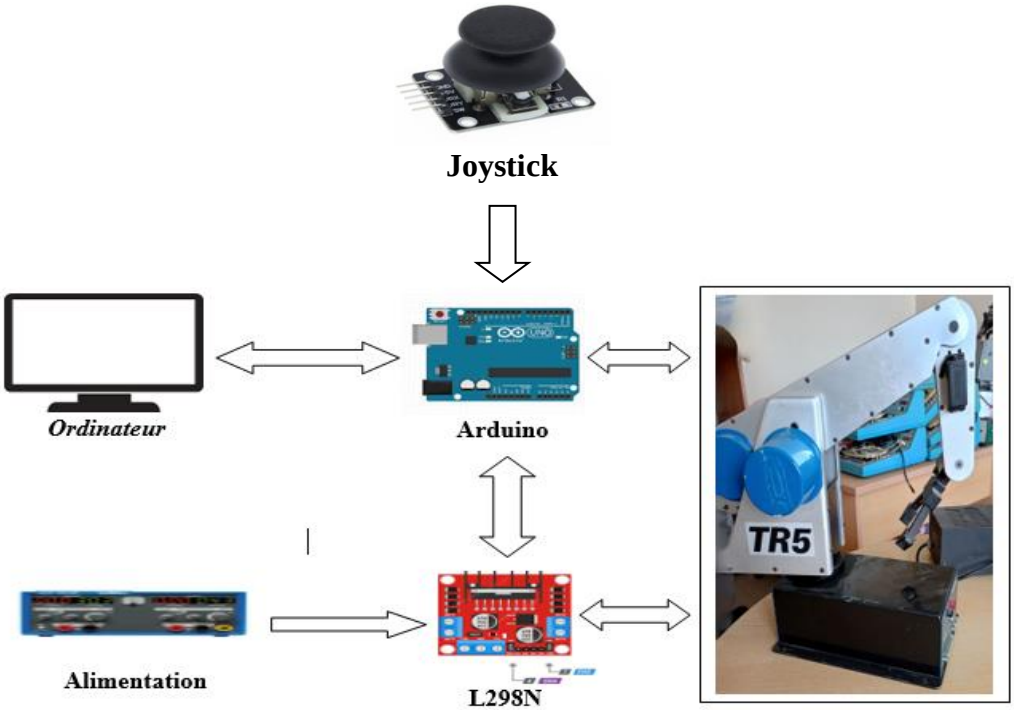


Figure IV-1:Schéma de notre montage

IV.4. Câblage d'un module de pilote de moteur L298N à un Arduino (3 moteurs)

1. On commence par connecter l'alimentation du moteur. Dans notre exemple, nous utilisons des moteurs avec réducteur de vitesses CC, également appelés moteurs "TT", que l'on retrouve souvent dans les robots à deux roues motrices. Ils sont conçus pour fonctionner sous une tension de 3 à 12V. Nous allons donc connecter une source d'alimentation externe de 12V à la borne VS. Étant donné que le L298N a une chute de tension d'environ 2V, les moteurs recevront 10V et tourneront légèrement moins rapidement. Mais c'est acceptable.
2. Ensuite, nous devons fournir 5V à la partie logique du L298N. Nous utiliserons le régulateur 5V intégré pour obtenir 5V à partir de l'alimentation du moteur, donc maintenir le cavalier 5V-EN en place.
3. Maintenant, on relie les broches de contrôle d'entrée et d'activation du premier module L298N (ENA, IN1, IN2, IN3, IN4 et ENB) aux six broches de sortie numériques de l'Arduino (13, 12, 11, 10, 9 et 8) et le 2ème module (ENA, IN1, IN2) avec 3 broches de sorties numériques de l'Arduino (4 ,3,2). Il faut notez que les broches de sortie 9,8 et 4 de l'Arduino prennent en charge la PWM (modulation de largeur d'impulsion).
4. Enfin, on relie un moteur à la borne A (OUT1 et OUT2) et l'autre à la borne B (OUT3 et OUT4). Les connexions du moteur peuvent être inversée.

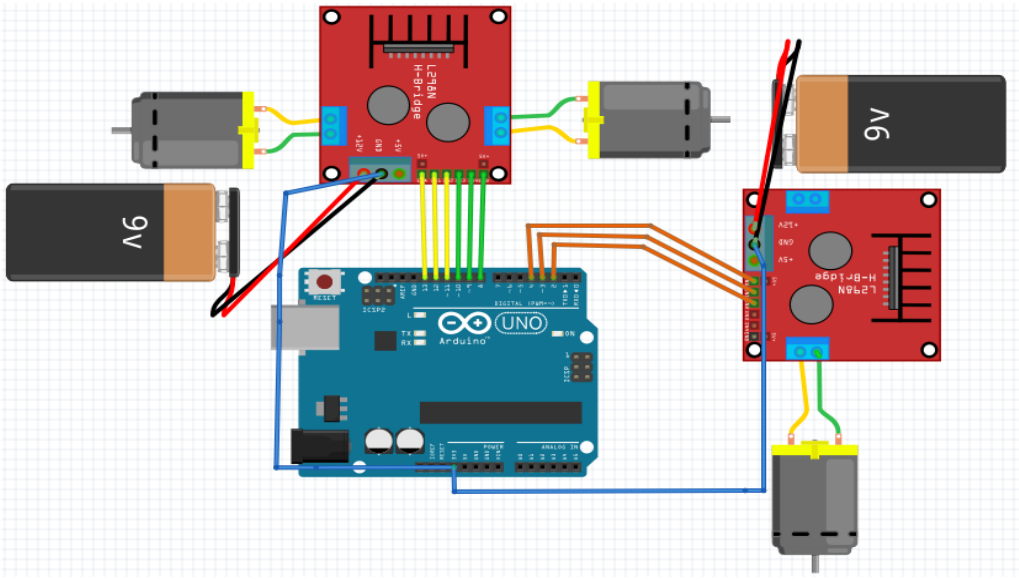


Figure IVIV-2:Câblage d'un module de pilote de moteur avec Arduino

IV.4.1 Pourquoi la carte de puissance

Le microcontrôleur (Arduino) a des limitations de courant qui ne peuvent pas alimenter un élément à haute puissance tel qu'un moteur. Le H-Bridge résout le problème en fournissant une alimentation différente pour conduire les moteurs. C'est pour cette raison que le L298 est utilisé.

Une tension d'alimentation stabilisée 9v 3A pour alimenter les moteurs, est offerte par une source externe.

IV.4.2 La simulation de câblage de 3 moteurs avec L298N et Arduino

1) On ajoute les composants nécessaires : l'Arduino Uno, les drivers L298N et trois moteurs DC
2) On Connecte les broches d'alimentation du L298N (Vcc et GND) à une source d'alimentation appropriée et au GND de l'Arduino.
3) Ensuite, on connecte les sorties des moteurs (OUT1-OUT2, OUT3-OUT4) du L298N aux bornes des trois moteurs DC. Il faut utiliser les broches d'entrée IN1, IN2, IN3 et IN4 du L298N pour contrôler les deux premiers moteurs en les reliant aux broches de l'Arduino. Pour le troisième moteur, on utilise une deuxième L298N, connectée de manière similaire.
4) Enfin, dans l'IDE Arduino, on écrit un programme pour contrôler la rotation des moteurs en définissant les broches comme sorties et en utilisant les fonctions « digitalWrite » et « analogWrite » pour gérer la vitesse et la direction des moteurs et on Importe ce programme dans Proteus pour la simulation et on s'assure que les connexions sont correctes et que les moteurs fonctionnent comme prévu.

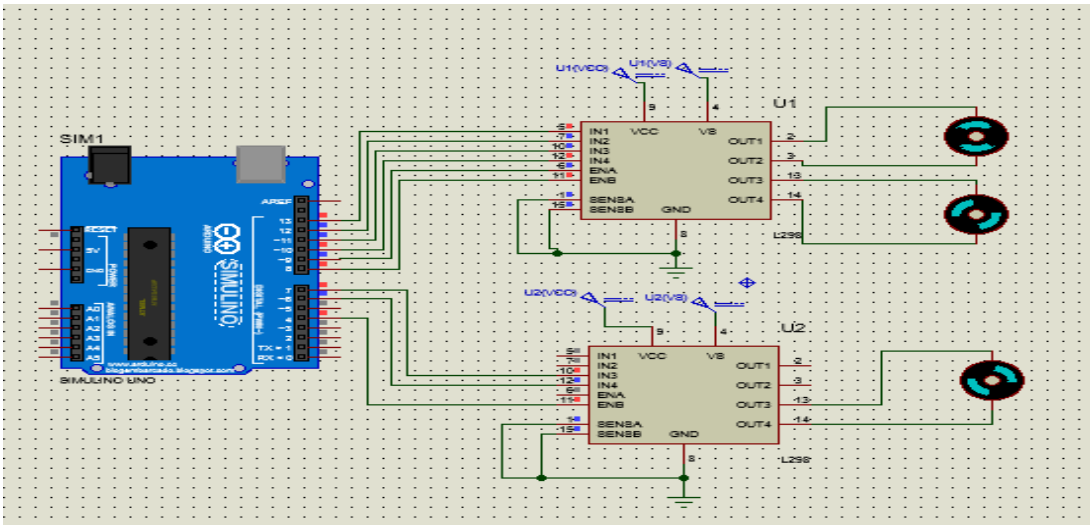


Figure IV-3:Câblage et simulation de 3 moteurs avec Proteus

IV.4.3 Code Arduino :

Voici une explication détaillée de chaque section du code :

1) **Déclaration des constantes :**

```
//Moteur 01
const int IN1 = 13;
const int IN2 = 12;
const int ENA = 9;
//moteur 02
const int IN3 = 11;
const int IN4 = 10;
const int ENB = 8;
//Moteur 03
const int IN5 = 7;
const int IN6 = 6;
const int ENC = 4;
```

1. **Fonction « Setup » :**

   Dans la fonction setup (), toutes les broches sont configurées comme des sorties car elles vont envoyer des signaux pour contrôler les moteurs.

```
void setup() {
//moteur 01
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(ENA, OUTPUT);
//Moteur 02
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  pinMode(ENB, OUTPUT);
//Moteur 03
  pinMode(IN5, OUTPUT);
  pinMode(IN6, OUTPUT);
  pinMode(ENC, OUTPUT);
}
```

2. **Fonction Loop :**

Dans la fonction loop(), le programme fait tourner les moteurs en continu :

- **Vitesse des moteurs**

La fonction analogWrite() utilise la modulation de largeur d'impulsion (PWM) pour contrôler la vitesse des moteurs. La valeur 250 (sur une échelle de 0 à 255) correspond à une vitesse élevée pour chaque moteur.

- **Direction des moteurs**

Les broches IN1 et IN2 contrôlent le moteur 1, IN3 et IN4 contrôlent le moteur 2, et IN5 et IN6 contrôlent le moteur 3. En mettant une broche en HIGH et l'autre en LOW, chaque moteur est configuré pour tourner dans une direction spécifique.

```
void loop() {
  analogWrite(ENA, 250);
  analogWrite(ENB, 250);
  analogWrite(ENC, 250);
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,HIGH);
  digitalWrite(IN5,HIGH);
  digitalWrite(IN6,LOW);
}
```

IV.5. Câblage de 6 moteurs dc avec 2 Arduino et 3 modules L298N

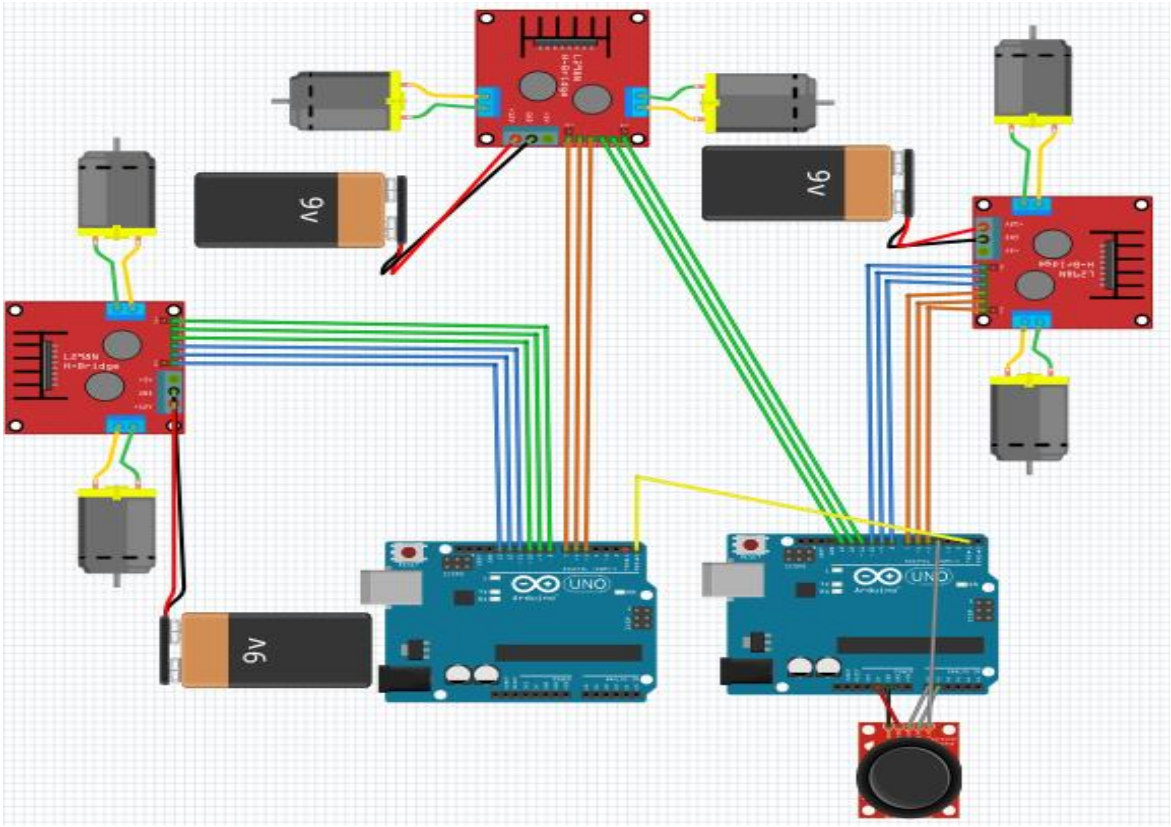


Figure IV-4:Cablage de notre projet

Le schéma montre un circuit avec deux cartes Arduino Uno, plusieurs modules de commande de moteur (probablement des modules L298N), des moteurs à courant continu, une manette (joystick) et plusieurs batteries de 9V

IV.5.1 Résumé des connexions :

1. **Modules L298N :**

- IN1, IN2, IN3, IN4 pour chaque module connecté aux broches numériques de l'Arduino.
- ENA et ENB connectés aux broches PWM de l'Arduino.
- VCC des modules connectés aux batteries de 9V.
- GND des modules connectés à la GND des batteries et des Arduino.

2. **Moteurs DC :**

Connectés aux sorties des modules L298N.

3. **Joystick :**

X et Y connectés aux entrées analogiques de l'Arduino.
SW (bouton) connecté à une entrée numérique de l'Arduino.
VCC et GND connectés aux 5V et GND de l'Arduino.

IV.5.2 Fonctionnement global :

Le joystick envoie des signaux analogiques et numériques à l'Arduino Uno de droite.
L'Arduino Uno de droite contrôle les moteurs via les modules L298N en fonction des signaux reçus du joystick.
Chaque module L298N contrôle deux moteurs et est alimenté par une batterie de 9V.
Les deux Arduinos travaillent ensemble pour contrôler les six moteurs DC de manière coordonnée.

IV.5.3 Le code Arduino de chaque Arduino

IV.5.3.1 Le premier Arduino

1. **Déclarations de broches et variables**

```
// Déclaration des broches pour les moteurs 1, 2, 3
#define ENA1 9
#define IN1A 8
#define IN2A 7
#define ENB1 6
#define IN3A 5
#define IN4A 4

int selectedPair = 0; // Variable pour stocker la paire de moteurs sélectionnée
```

- VRx, VRy et SW sont les broches du joystick pour les axes X et Y et le bouton du joystick.
- ENA1, IN1A, IN2A, ENB1, IN3A, IN4A sont les broches utilisées pour contrôler les moteurs 1, 2 et 3.
- selectedPair est une variable pour suivre quelle paire de moteurs est actuellement sélectionnée.

2. **Initialisation**

```
void setup() {
  // Initialisation des broches du joystick
  pinMode(VRx, INPUT);
  pinMode(VRy, INPUT);
  pinMode(SW, INPUT_PULLUP);

  // Initialisation des broches comme sorties pour les moteurs
  pinMode(ENA1, OUTPUT);
  pinMode(IN1A, OUTPUT);
  pinMode(IN2A, OUTPUT);
  pinMode(ENB1, OUTPUT);
  pinMode(IN3A, OUTPUT);
  pinMode(IN4A, OUTPUT);

  // Initialisation de la communication série
  Serial.begin(9600);
}
```

- Configure les broches du joystick comme entrées.
- Configure les broches des moteurs comme sorties.
- Initialise la communication série à 9600 bauds.

3. Boucle principale

```
void loop() {
  int xValue = analogRead(VRx);
  int yValue = analogRead(VRy);
  boolean buttonPressed = digitalRead(SW) == LOW;

  // Changement des paires de moteurs en appuyant sur le bouton du joystick
  if (buttonPressed) {
    selectedPair = (selectedPair + 1) % 3; // Alterne entre 0, 1, et 2
    delay(300); // Anti-rebond pour éviter des changements multiples sur un seul appui
  }

  // Envoyer les commandes de moteur en fonction de la position du joystick
  char command = getJoystickCommand(xValue, yValue, selectedPair);
  Serial.write(command);
  controlMotors(command);
  delay(100); // Petite pause pour éviter d'envoyer trop de commandes en même temps
}
```

- Lit les valeurs analogiques des axes X et Y du joystick.
- Lit l'état du bouton du joystick.
- Si le bouton est pressé, il change la paire de moteurs sélectionnée, alternant entre 0, 1 et 2 avec un délai pour éviter le rebond du bouton.
- Calcule la commande de moteur appropriée en fonction des valeurs du joystick et de la paire de moteurs sélectionnée.
- Envoie cette commande via la communication série.
- Appelle la fonction controlMotors pour exécuter la commande.
- Attends 100 millisecondes avant de lire à nouveau les valeurs du joystick.

4. Génération des commandes du joystick

```
char getJoystickCommand(int x, int y, int selectedPair) {
  if (x > 600) {
    return '1' + (selectedPair * 4); // Avancer les moteurs sélectionnés
  }
  if (x < 400) {
    return '2' + (selectedPair * 4); // Reculer les moteurs sélectionnés
  }
  if (y > 600) {
    return '3' + (selectedPair * 4); // Avancer les moteurs sélectionnés
  }
  if (y < 400) {
    return '4' + (selectedPair * 4); // Reculer les moteurs sélectionnés
  }
  return '0'; // Arrête tous les moteurs si le joystick est en position neutre
}
```

- En fonction des valeurs X et Y du joystick, détermine une commande pour avancer ou reculer les moteurs de la paire sélectionnée.
- Retourne '0' si le joystick est dans une position neutre (aucune action).

5. Contrôle des moteurs

```
void controlMotors(char command) {
  switch (command) {
    // Moteurs 1 et 2
    case '1':
      digitalWrite(IN1A, HIGH);
      digitalWrite(IN2A, LOW);
      analogWrite(ENA1, 255);
      digitalWrite(IN3A, HIGH);
      digitalWrite(IN4A, LOW);
      analogWrite(ENB1, 255);
      break;
    case '2':
      digitalWrite(IN1A, LOW);
      digitalWrite(IN2A, HIGH);
      analogWrite(ENA1, 255);
      digitalWrite(IN3A, LOW);
      digitalWrite(IN4A, HIGH);
      analogWrite(ENB1, 255);
      break;

    // Moteurs 3
    case '5':
      digitalWrite(IN1A, HIGH);
      digitalWrite(IN2A, LOW);
      analogWrite(ENA1, 255);
      break;
    case '6':
      digitalWrite(IN1A, LOW);
      digitalWrite(IN2A, HIGH);
      analogWrite(ENA1, 255);
      break;
      default:
        stopMotors();
        break;
    }
  }

  void stopMotors() {
    analogWrite(ENA1, 0);
    analogWrite(ENB1, 0);
  }
```

- controlMotors : en fonction de la commande reçue, contrôle les moteurs en définissant les broches appropriées. Les cas '1' et '2' contrôlent les moteurs 1 et 2, tandis que les cas '5' et '6' contrôlent le moteur 3.
- stopMotors : arrête tous les moteurs en mettant les broches PWM à 0.

Résumé :

Le code utilise un joystick pour contrôler trois moteurs DC. La position du joystick (axes X et Y) et le bouton du joystick déterminent quelles paires de moteurs sont sélectionnées et dans quelle direction elles doivent se déplacer. Les commandes sont envoyées via la communication série et les moteurs sont contrôlés en conséquence.

IV.5.4 Le 2 -ème Arduino

Ce code est utilisé sur une deuxième carte Arduino qui travaille en tandem avec la première pour contrôler les moteurs DC 4, 5 et 6. La communication série est utilisée pour recevoir les commandes envoyées par la première Arduino, qui lit les valeurs du joystick et décide des actions à entreprendre.

Les mêmes étapes de premier code sont effectuées mais il y a un changement dans Fonction de contrôle des moteurs

1. Fonction de contrôle des moteurs

```
void controlMotors(char command) {
  switch (command) {
    // Moteurs 4 et 5
    case '9':
      digitalWrite(IN1B, HIGH);
      digitalWrite(IN2B, LOW);
      analogWrite(ENA2, 255);
      digitalWrite(IN3B, HIGH);
      digitalWrite(IN4B, LOW);
      analogWrite(ENB2, 255);
      break;
    case 'A':
      digitalWrite(IN1B, LOW);
      digitalWrite(IN2B, HIGH);
      analogWrite(ENA2, 255);
      digitalWrite(IN3B, LOW);
      digitalWrite(IN4B, HIGH);
      analogWrite(ENB2, 255);
      break;

    // Moteur 6
    case 'B':
      digitalWrite(IN1B, HIGH);
      digitalWrite(IN2B, LOW);
      analogWrite(ENA2, 255);
      break;
    case 'C':
      digitalWrite(IN1B, LOW);
      digitalWrite(IN2B, HIGH);
      default:              255);
        stopMotors();
        break;
    }
  }

  void stopMotors() {
    analogWrite(ENA1, 0);
    analogWrite(ENB1, 0);
  }
```

- La fonction controlMotors prend en entrée une commande et exécute les actions correspondantes sur les moteurs.
- Les commandes '9' et 'A' contrôlent les moteurs 4 et 5.
- ✓ '9' : Avance les moteurs 4 et 5 en mettant IN1B et IN3B à HIGH et IN2B et IN4B à LOW.
- ✓ 'A' : Recule les moteurs 4 et 5 en mettant IN1B et IN3B à LOW et IN2B et IN4B à HIGH.
- Les commandes 'B' et 'C' contrôlent le moteur 6.
- ✓ 'B' : Avance le moteur 6 en mettant IN1B à HIGH et IN2B à LOW.
- ✓ 'C' : Recule le moteur 6 en mettant IN1B à LOW et IN2B à HIGH.

Si une commande non reconnue est reçue, la fonction stopMotors est appelée pour arrêter tous les moteurs.

Interaction entre les deux Arduino

Première Arduino :

- Lit les positions du joystick.
- Génère une commande en fonction des positions du joystick et de la paire de moteurs sélectionnée.
- Envoie cette commande à la deuxième Arduino via la communication série.
- Contrôle les moteurs 1, 2 et 3 directement.

Deuxième Arduino :

- Reçoit la commande via la communication série.
- Exécute la commande reçue pour contrôler les moteurs 4, 5 et 6 en fonction de la commande.

En résumé, ce code permet à la deuxième Arduino de recevoir des commandes de la première Arduino et de contrôler les moteurs 4, 5 et 6 en conséquence. La communication série permet aux deux Arduino de travailler ensemble pour gérer l'ensemble des six moteurs contrôlés par le joystick.

IV.5.5 La simulation de notre câblage

Ce schéma montre une configuration où deux cartes Arduino Uno (SIM1 et SIM2) contrôlent des moteurs à travers des pilotes de moteurs L298N. Chaque Arduino est connecté à deux pilotes de moteurs (U1 et U2 pour SIM1, et U2 et U3 pour SIM2) via leurs broches numériques. Les signaux de commande (IN1 à IN4) des pilotes de moteurs sont reliés aux broches numériques des Arduinos. Chaque pilote de moteur L298N peut contrôler deux moteurs, représentés par les bobines en rouge sur le schéma, ce qui permet de contrôler un total de six moteurs. Les broches ENA et ENB des pilotes L298N sont utilisées pour activer les moteurs et réguler leur vitesse via PWM. Les connexions d'alimentation (VCC, GND, et VS) sont également représentées, assurant l'alimentation des composants

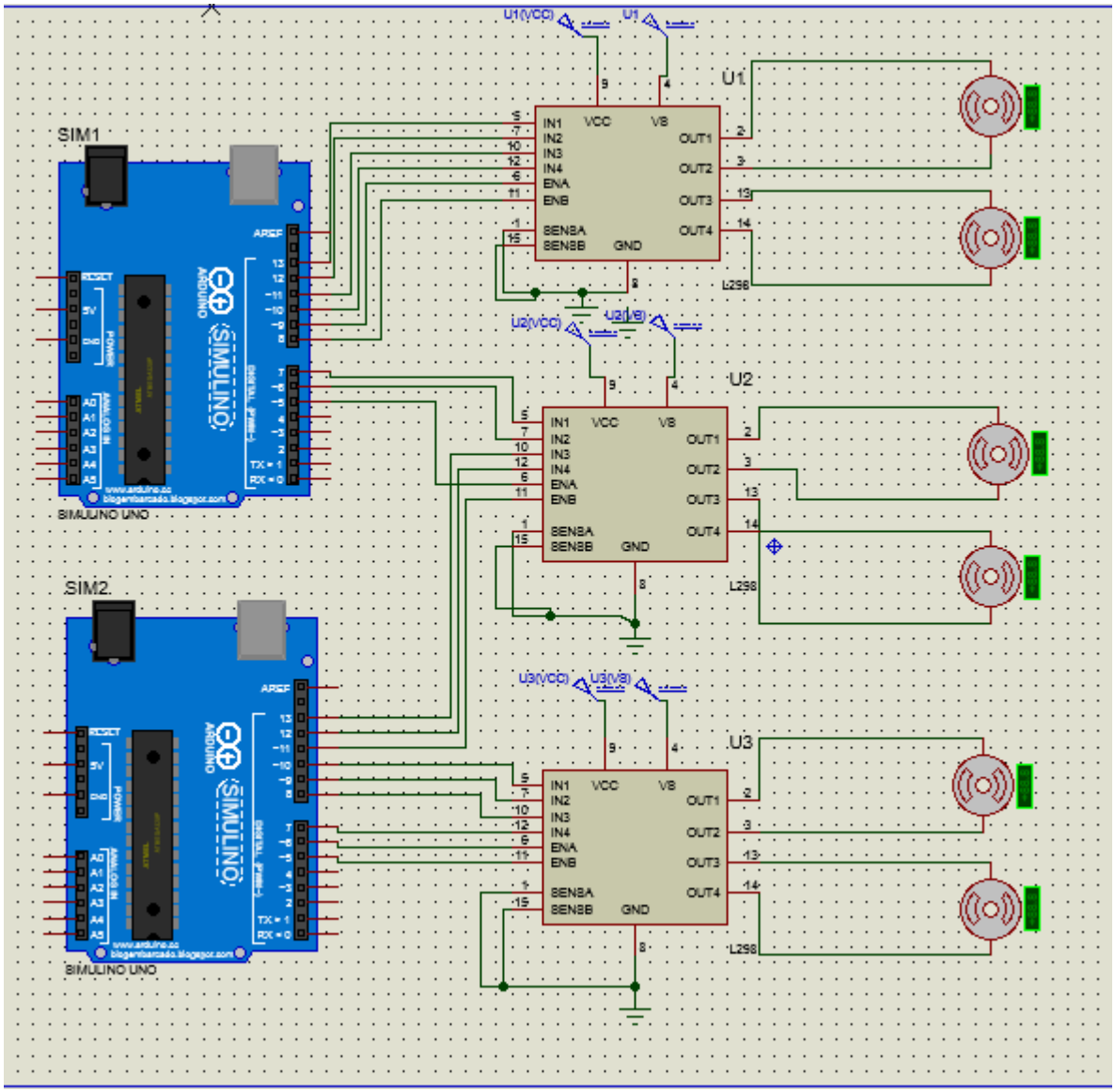


Figure IV-5:la simulation de notre projet

IV.6. La partie expérimentale

IV.6.1 La partie finale de commande

La figure IV.6 montre la nouvelle carte de commande du robot, celle-ci est composée de deux parties, la première partie comporte les cartes Arduino méga et Uno.
La deuxième partie, comporte les pilotes des moteurs. Elle permet de contrôler les moteurs.

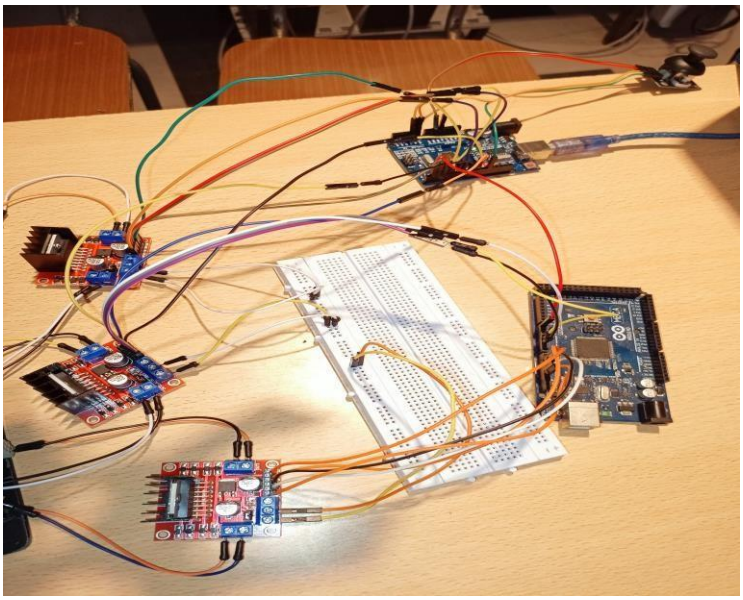


Figure IV-6 : La partie commande finale

Le système est configuré de manière à ce que l'Arduino puisse interpréter les commandes du joystick, puis envoyer les signaux appropriés aux drivers de moteur, qui à leur tour contrôlent les moteurs du bras robotique. Cette configuration permet de tester et de valider les algorithmes de contrôle dans un environnement de prototypage rapide, facilitant ainsi les ajustements et les améliorations nécessaires au développement du projet.

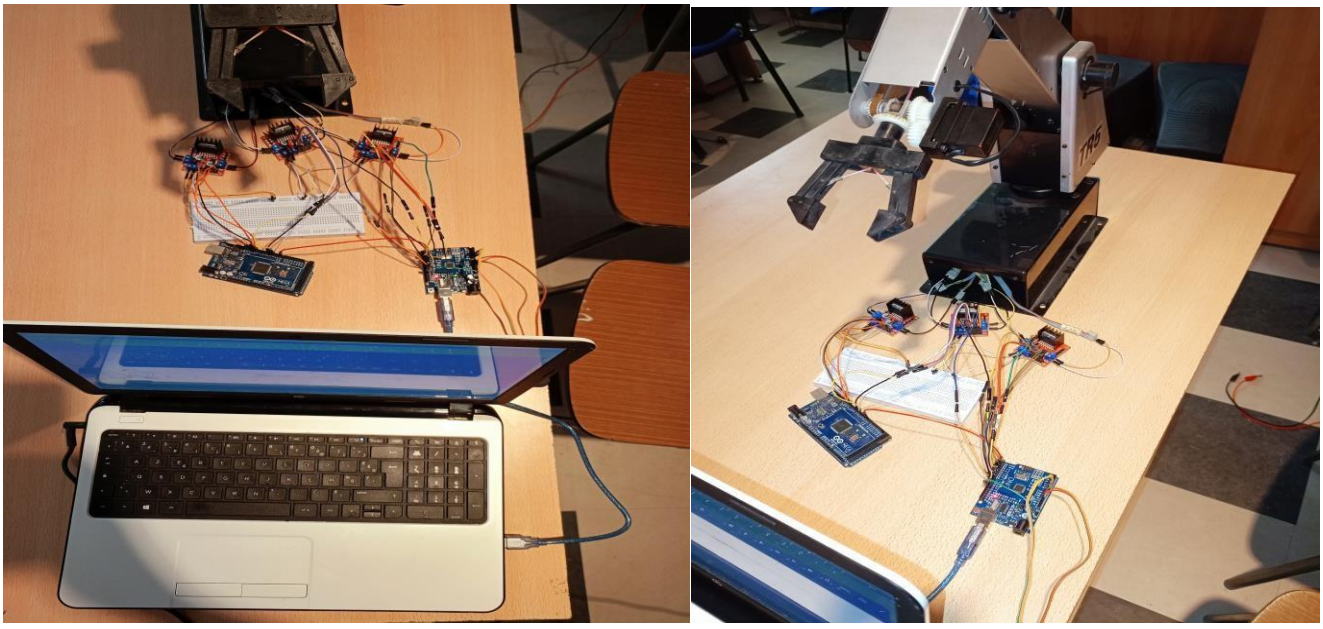


Figure IV-7:Le câblage de nouvelle commande avec le TR5

IV.7. Conclusion

Ce chapitre a permis d'explorer en détail les aspects matériels et logiciels essentiels à la réalisation de projets basés sur l'Arduino. En mettant en lumière les caractéristiques de la carte Arduino Uno du microcontrôleur ATMega328, et la carte Méga, du joystick comme organe de commande, et du driver L298 pour le contrôle des moteurs, ce chapitre a fourni une base solide pour la compréhension et la manipulation de ces composants clés. Grâce à une analyse approfondie du matériel et du logiciel présentés

Conclusion générale

Au cours de ce mémoire, nous avons exploré les différentes facettes de la robotique et de l'électronique, en mettant un accent particulier sur le développement et la mise en œuvre du bras manipulateur TR5. Le parcours à travers les quatre chapitres principaux nous a permis d'acquérir une compréhension globale et détaillée des composantes essentielles de ce domaine.

Dans le premier chapitre, nous avons introduit les concepts de base de la robotique, en présentant les différentes catégories de robots et leurs applications variées. Cette introduction a posé les fondations nécessaires pour aborder les aspects plus techniques des chapitres suivants.

Le deuxième chapitre a fourni une description exhaustive du bras manipulateur TR5, en couvrant ses propriétés mécaniques et électriques. Nous avons examiné la structure et le fonctionnement de ce robot, soulignant ses capacités et les défis associés à son utilisation.

Le troisième chapitre s'est concentré sur la partie logicielle, en particulier sur la programmation du microcontrôleur ATMEGA 2560 utilisé dans notre projet. Nous avons détaillé les étapes de codage et les configurations nécessaires pour assurer un fonctionnement optimal du bras manipulateur.

Enfin, le quatrième chapitre a traité de la réalisation de la carte de commande en puissance des moteurs du TR5 ainsi que de l'unité de commande et de contrôle du mouvement. Nous avons exploré les aspects pratiques de la mise en œuvre et de la simulation de notre projet, utilisant des outils comme Proteus pour vérifier et valider nos configurations.

À travers cette étude, nous avons démontré l'importance de combiner des connaissances théoriques approfondies avec des compétences pratiques pour réussir dans le domaine de la robotique. Le projet du bras manipulateur TR5 illustre comment des concepts complexes peuvent être transformés en applications pratiques, offrant ainsi des perspectives prometteuses pour l'avenir de la robotique et de l'automatisation.

Bibliographie

[1] https://www.ensta-bretage.fr/jaulin/mastersds_cours_robot_boimond.pdf

[2] https://www.univ-saida.dz/butec/index.php?lvl=notice_display&id=376131

[3] Mr. SAADI Ramzy, Mr. GUESBAYA Tahar, et Mr. SALHI Nassereddine; Réalisation de carte à microcontrôleur pour le contrôle de bras manipulateur via un pc ; Mémoire de fin d'études pour l'obtention du diplôme de MASTER université Mohamed Khider Biskra ; 2010.

[4] Jean-Louis Boimond « ROBOTIQUE », ISTIA, Université Angers.

[5] : https://elearn.univ-ouargla.dz (Cours de Dr. Belloufi Abderrahim)2015

[6] : Jacque Gangloff « Cours de robotique » Université de strasboug 2010

[7] : https://sites.google.com/site/avesnellestechno12c33ga/les-differents-types-de-robots

[8] : http://ia-2011tpe.e-monsite.com/pages/les-robots/differents-usages-des-robots.html

[9] : « TEACHBOX MANUAL IR 50 p », © EUROBTEC Gmbh, Zirndorf 1993, 1994.

[10] : Marzine Abd ElKarim « Commande du bras de robot ROB3 par microcontrôleur », Université Ibn Khaldoun – Tiaret-.

[11] : Mr. DJAFRI MENAD, Mr. CHELOUCHE Djalal, Etude et Réalisation d'une Carte Arduino ; Mémoire de fin d'études pour l'obtention du diplôme de MASTER université A. MIRA DE BEJAIA ; 2016.

[12] : http://www.elektronique.fr/logiciels/proteus.php

[13] Variateur de Vitesse Pont Hmax 10A – Technologue pro