

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

UNIVERSITÉ BADJI MOKHTAR - ANNABA
BADJI MOKHTAR – ANNABA UNIVERSITY



جامعة باجي مختار – عنابة

Faculté : Sciences de L'ingénierat
Département : Electronique
Domaine : Sciences et Techniques
Filière : Automatique
Spécialité : automatique et informatique industriel

Mémoire

Présenté en vue de l'obtention du Diplôme de Master

Thème: Commande de servomoteurs par une carte Arduino

Présenté par : *Regainia Djamel*

Encadrant : *Chaker Karima*

Maître assistante(B)

Bedji Mokhtar Annaba

Jury de Soutenance :

Gherbi Sofiane	MAA	Bedji Mokhtar Annaba	Président
Chaker Karima	MAB	Bedji Mokhtar Annaba	Encadrant
Benmoussa Samir	MAA	Bedji Mokhtar Annaba	Examineur

Année Universitaire : 2020/2021

Remerciements

Elhamdo Lileh, qui nous a donné la volonté, la santé et la patience de réaliser ce mémoire.

Je tiens à remercier mes parents, mon frère, ma sœur et toute ma famille pour leur soutien moral tout au long de cette période.

Je voudrais adresser toute ma gratitude à mon encadreur, Madame Phaker Karima, pour sa patience, sa disponibilité et surtout ses conseils.

Je désire aussi remercier mon ami Ahmed Gaid Othman pour ses conseils

Je remercie aussi mes amis et tous ceux qui m'ont aidé dans mon travail, même avec un mot gentil.

Dédicaces

Je dédie ce modeste travail qui a été fait avec l'aide de Dieu

À mes chers parents,

Ma sœur,

Mon frère,

Et toute ma famille

À tous mes amis et À tous ceux qui me sont chers

ملخص

العمل الذي تم تنفيذه في مشروع نهاية الدراسة هذا هو جزء من التحكم في المحركات الموازنة بواسطة بطاقة أردوينو، تم استغلال طرق التحكم في المحركات الموازنة القياسية في مجال الروبوتات بشكل عام وذراع المناور بثلاث درجات من الحرية على وجه الخصوص حيث سيظهر التحكم في المحركات الموازنة عند مفاصل هذا الذراع . وبالتالي، فإن المساهمة الرئيسية لهذا العمل هي في الأساس تصنيع ذراع مناور، يمر بجميع المراحل الأساسية لهذا الغرض، من الشكل ثلاثي الأبعاد إلى الحصول على الذراع المعنية جاهزة للاختبار.

RESUME

Le travail effectué dans ce projet de fin d'études s'intègre dans le cadre de la commande de servomoteurs par Arduino. Les méthodes de contrôle des servomoteurs standards ont été exploitées dans le domaine de la robotique en général et le bras manipulateur à trois degrés de liberté en particulier ou le contrôle des servomoteurs apparaîtra au niveau des articulations de ce bras. Ainsi, la principale contribution de ce travail est essentiellement la fabrication d'un bras manipulateur en passant par toutes les étapes indispensable à cet effet depuis la forme 3D jusqu'à l'obtention du bras en question prêt à être expérimenté.

ABSTRACT

The work carried out in this end-of-study project is part of the control of servomotors by Arduino. The standard servomotor control methods have been exploited in the field of robotics in general and the manipulator arm with three degrees of freedom in particular where the control of the servomotors will appear at the joints of this arm. Thus, the main contribution of this work is essentially the manufacture of a manipulator arm, going through all the stages essential for this purpose, from the 3D shape to obtaining the arm in question ready to be tested.

TABLE DES MATIERES

Remerciements	1
Dédicaces	2
ملخص.....	2
RESUME.....	4
ABSTRACT.....	5
Liste des Tableaux.....	10
Liste des figures	11
Liste des abréviations	14

INTRODUCTION GENERALE

Introduction générale.....	16
----------------------------	----

Chapitre 01 LES SERVOMOTEURS

1.1 Introduction	18
1.2 Définitions	18
1.3. Les type de servomoteurs	18
1.3.1. Servomoteur standard (à rotation de position)	18
1.3.2. Servomoteur à rotation continu	19
1.3.3. Servomoteur à rotation linéaire	19
1.3.4. Servomoteur analogique avec signal de feedback	20
1.4. Les constituants du servomoteur	20
1.4.1 Moteur électrique à courant continu (MCC)	20
1.4.2 Train d'engrenages	21
1.4.3 Le potentiomètre.....	22
1.4.4. Carte électronique.....	22
1.4.5. Butée mécanique (servomoteur standard)	22
1.4.6. Palonnier.....	23
1.5. Principe de fonctionnement d'un servomoteur standard	23
1.6. Le couple d'un servomoteur.....	24
1.7. Conclusion.....	25

Chapitre 02 L'ARDUINO

2.1. Introduction	27
2.2 Définitions	27
2.2.1 Arduino	27
2.2.2 La carte Arduino	28
2.3 Les applications de la carte Arduino	28
2.4 Les types de cartes Arduino :	29
2.5. Caractéristiques principales de la carte Arduino	29
2.5 .1 Version.....	29
2.5.2 Alimentation	29
2.5.3 Le microcontrôleur	30
2.5.4 La mémoire.....	30
2.5.5 Les ports d'Arduino	31
2.6. Différentes cartes Arduinos.....	32
2.7. Comparaison des différentes cartes Arduinos	33
2.8. L'Arduino IDE	34
2.8.1 Téléchargement et installation du logiciel Arduino IDE.....	34
2.8.2 Description de l'Arduino IDE	35
2.9 Structure d'un programme Arduino	36
2.10 Conclusion.....	38

Chapitre 03 LA ROBOTIQUE

3.1 Introduction	40
3.2 Aperçu sur l'origine des robots	40
3.2.1 Les types de robots	41
a- Les robots manipulateurs	41
b- Les humanoïdes.....	41
c- Les robots mobiles	42
3.3 Constituants mécaniques des robots manipulateurs	43
3.3.1 La structure mécanique articulée	43
3.3 Les types d'articulations.....	43
3.3.2 Espace de travail des robots manipulateurs	44
3.4 Modélisation du robot manipulateur	44

3.4.1 Description des attributs géométriques.....	45
a- La position	45
b- L'orientation	45
c- Matrice de transformation homogène.....	46
3.5 Modèle géométrique.....	46
3.5.1 Modèle géométrique directe	46
3.5.1.1 Description de la transformation de D-H.....	47
3.5.2 Modèle géométrique inverse	48
3.6 Conclusion.....	49

Chapitre 04 REALISATION PRATIQUE DU BRAS MANIPULATEUR

4.1 Introduction.....	51
Partie I Réalisation mécanique du robot manipulateur	52
4. I.1 Anatomie fonctionnelle.....	52
4. I. 2 Fabrication mécanique	52
4. I. 3 Etapes de fabrication du bras manipulateur	53
4. I. 3. 1 La modélisation par ordinateur	53
a. Présentation de SolidWorks	53
b. Création des pièces du robot manipulateur par SolidWorks.....	53
c. Assemblage des pièces du robot manipulateur par SolidWorks	62
4. I. 3. 2 Choix des outils d'impression et de la matière des pièces du robot.....	65
a. La Fraiseuse	65
b. Programmation de la machine	66
c. Choix du matériau d'impression	67
4. I. 3. 2 Assemblage des pièces	68
Partie II Modélisation du robot manipulateur	70
4. II.1 Introduction	70
4. II.2 Modèle géométrique directe de Denavit-Hartenberg	70
4. II. 3 Modèle géométrique inverse	73
Partie III Sélection des outils et composants de commande	75
4. III.1 Servomoteurs.....	75
4. III.2 Exemple de commande d'un servomoteur	77
4. III.3 Afficheur à cristaux liquides (LCD).....	78

4. III.3 Carte Arduino UNO	82
Partie IV Réalisation du circuit électronique du bras manipulateur.....	83
4. IV. 1 Circuit électronique complet de la commande du bras manipulateur avec Arduino ..	83
4. IV.2 Assemblage des composants dans le socle du robot manipulateur	85
4.2 Conclusion.....	87
Conclusion générale	88
Perspectives	89
RÉFÉRENCES.....	90

Liste des Tableaux

Tableau 2.1 : Tableau comparatif des différentes cartes Arduinos	33
Tableau 4.1 Tableau des paramètres de DH.....	70
Tableau 4.2 Description des broches du LCD 16*2.....	79
Tableau 1.3 Exigences des entrées/sorties.....	82

Liste des figures

Figure 1.1. Servomoteur standard mg996 de 180° [2]	19
Figure 1.2. Servomoteur SR4200 à rotation continu (360°)	19
Figure 1.3 Servomoteur à rotation linéaire.....	19
Figure 1.4 Servomoteur analogique avec signal de feedback.	20
Figure 1.5 Moteur à courant continu	20
Figure 1.6 Train d'engrenages de servomoteur sg90	21
Figure 1.7 Potentiomètre du servomoteur sg90	22
Figure 1.8 La carte électronique du servomoteur sg90	22
Figure 1.9 Les constituants élémentaires d'un servomoteur	23
Figure 1.10 La boucle d'asservissement générale de servomoteur.	23
Figure 1.11 Illustration du couple d'un servomoteur	24
Figure 2.1 Logo de la marque Arduino	27
Figure 2.2 Carte Arduino UNO	28
Figure 2.3 Alimentation de l'Arduino Uno.....	30
Figure 2.4 Microcontrôleur de la carte Arduino UNO	30
Figure 2.5 Les entrées/sorties de la carte Arduino	31
Figure 2.6 La famille des cartes Arduinos	32
Figure 2.7 Interface IDE d'Arduino	34
Figure 2.9 Liaison de la carte Arduino à l'ordinateur via le câble USB.	35
Figure 2.10 Icône du logiciel Arduino IDE.....	35
Figure 2.11 La barre de menu de l'IDE Arduino	36
Figure 2.12 Interface de l'Arduino 1.8.13.....	36
Figure 2.13 Structure d'un programme Arduino.....	37
Figure 3.1 Pièce théâtrale créée par Karel Čapek en1920 [2]	40
Figure 3.2 Robot manipulateur Staubli TX90. [1]	41
Figure 3.3 Robot humanoïde Nao	41
Figure 3.4 Le robot de manipulation mobile Momaro	42
Figure 3.5 Constituants du robot manipulateur	42
Figure 3.6 Symbole d'articulation rotoïde [5].....	43
Figure 3.7 Symbole d'articulation prismatique.....	43
Figure 3.8 Espace de travail des robots.....	44
Figure 3.9 Représentation spatiale d'un point.....	45
Figure 3.10 Représentation de l'orientation.	45
Figure 3.11 Illustration du modèle géométrique inverse.....	48
Figure 3.12 Illustration de la structure du robot manipulateur.....	52
Figure 11 Illustration de la structure du robot manipulateur.....	52
Figure 4.2 Icône de SolidWorks 2018.....	52
Figure 4.3 L'interface de SolidWorks.	54
Figure 4.4 Les différentes parties de l'interface de SolidWorks.	54
Figure 4.5 L'arbre de création.....	55
Figure 4.6 Outils de dessin d'esquisse	55
Figure 4.7 Barre d'outils.	55
Figure 4.8 Les mesures de la première pièce du robot (La face arrière du socle).....	55

Figure 4.9 Outils d'ajout de matière.....	56
Figure 4.10 Ajout de matière.....	56
Figure 4.11 Outil d'enlèvement de matière.....	56
Figure 4.12 Outil de dessin d'esquisse d'une ligne.....	57
Figure 4.13 Outil de choix de la police.	57
Figure 4.14 Exemple d'écriture avec Solidworks.	57
Figure 4.15 La base du socle.....	57
Figure 4.16 La pièce du coté gauche du socle.....	58
Figure 4.17 La pièce de l'interface du socle.	58
Figure 4.18 La pièce supérieure du socle.....	58
Figure 4.19 Coté droit du socle.	59
Figure 4.20 Face arrière du socle.	59
Figure 4.21 La pièce de la petite base.	59
Figure 4.22 support du servomoteur.	60
Figure 4.23 La pièce reliant la 10ème pièce à la base.....	60
Figure 4.24 Un segment du robot.....	60
Figure 4.25 Segment du support du 3ème servomoteur.....	61
Figure 4.26 Segment de fixation de l'organe terminal.....	61
Figure 4.27 Raccord des segments.....	61
Figure 4.28 : pièce d'Organe terminal du robot manipulateur.....	62
Figure 4.29 Les 3 modules élémentaires d'accueil.....	62
Figure 4.30 Outils d'insertion et de contrainte entre les composants.	62
Figure 4.31 Forme finale du robot sans l'effecteur.....	63
Figure 4.32 Les différentes pièces de l'effecteur.....	63
Figure 4.33 Assemblage des pièces de l'effecteur.....	64
Figure 4.34 La forme finale du robot manipulateur avec effecteur.....	64
Figure 4.35 Machine CNC SK-N1325.....	66
Figure 4.36 Fraise du Fraiseuse.....	66
Figure 4.37 10ème pièce sous forme Fisher source « DXF ».	67
Figure 4.38 DSP.....	67
Figure 4.39 Les matériaux testés dans la fabrication du robot.....	68
Figure 4.34 Accessoires d'assemblage.....	68
Figure 4.41 Assemblage final du bras manipulateur.....	69
Figure 4.42 Repères en termes des paramètres D-H.....	70
Figure 4.43 Programme de calcul sou Matlab.....	72
Figure 4.44 Projection du bras manipulateur sur un repère x, y, z. et x,y.....	73
Figure 4.45 Couples nécessaires pour chaque servomoteur.....	76
Figure 4.46 Servomoteur MG996.	76
Figure 4.47 Servomoteur SG90.....	77
Figure 4.48 Circuit de commande du servomoteur.....	77
Figure 4.49 LCD 16*2 et son brochage	79
Figure 4.50 Branchement du LCD avec Arduino Uno.....	80
Figure 4.51 Circuit de l'afficheur LCD.....	81
Figure 4.52 Test d'affichage sur LCD.	81

Figure 4.53 Schéma électrique de commande du bras manipulateur avec Tinkercad	84
Figure 4.54 Câblage électrique du le robot manipulateur	84
Figure 4.55 Robot manipulateur avec ses accessoires.	86

Liste des abréviations

2ddl : Deux degré de liberté.

MGD : Modèle géométrique direct.

MGI : Modèle cinématique inverse.

DH : Convention de Denavit-Hartenberg.

3D : Trois dimensions

Kg : kilogramme.

m : mètre.

S : seconde.

E/S : Entré/Sortie.

Pic : Périphérie Interface Controller

Api : Automate programmable industriel

CNC: Computerized Numerical Control

DXF: Drawing eXchange Format

DSP : Digital Signal Processors

MCC : Moteur a courant continu

P : Puissance

Ω_m : Vitesse angulaire de rotation de l'axe du moteur

Ω_p : Vitesse angulaire de rotation de l'axe du palonnier

PWM : Pulse Width Modulation

MLI : Modulation de largeur d'impulsion

F: force

R: Distance

C: Couple

USB: Universal Serial Bus

θ : Angle d'articulation

INTRODUCTION GENERALE

Introduction générale

IL est connu que l'utilisation des moteurs électriques est incontournable dans le monde de l'industrie, aucun concepteur ou constructeur de machines ne peut se passer de moteurs quel que soit son type à savoir servomoteurs, moteur pas à pas, moteur asynchrone...etc.

La commande des moteurs électriques a elle aussi connu un très grand engouement à travers l'histoire de l'industrie conduisant à une réelle concurrence à l'échelle mondiale et aboutissant à la création d'une multitude d'outils de commandes comme l'API, l'Arduino, le Raspberry Pi, le PIC, ...etc. Dans le cadre de notre projet de fin d'études, il est justement question de la **Commande de Servomoteurs par Arduino**, et ce à travers son application au domaine de la robotique comme moyen d'illustration.

Pour atteindre cet objectif, le travail a été divisé en quatre chapitres. Dans le premier chapitre, une étude générale sur les servomoteurs est présentée en se basant sur leurs caractéristiques électromécaniques et leurs principes de fonctionnements.

En raison de l'existence de plusieurs types de cartes Arduinos, cela nous a conduit à présenter dans le deuxième chapitre les différents types de cartes existantes en les regroupant dans un tableau comparatif pour mieux apprécier les différentes caractéristiques de chaque carte Arduinos connues dans le marché de l'industrie. Une présentation du logiciel de programmation employé pour les cartes Arduinos a clôturé ce chapitre.

Le troisième chapitre, renferme un bref historique sur la robotique, les différents types de robots ainsi que la constitution mécaniques des robots manipulateurs et leurs modélisation étant donné que ça représente l'application à concevoir.

Dans le quatrième chapitre, nous présentons les différentes étapes de fabrication mécanique du bras manipulateur de notre projet. Puis, on s'intéresse à la modélisation de notre bras manipulateur avec Solidwork. Dans la troisième partie, on a sélectionné des matériaux compatibles et nécessaires pour la commande du robot manipulateur. Enfin, dans cette dernière partie, la réalisation est relative au circuit et programme de commande des articulations du robot par une carte Arduino.

Nous clôturons le travail par une conclusion générale qui contient une récapitulation des étapes indispensables à la fabrication d'un bras manipulateur à trois articulations et à leurs commandes par une carte Arduino UNO pour contrôler les servomoteurs.

Chapitre 01

LES SERVOMOTEURS

1.1 Introduction

La technologie de pointe est en développement permanent et emploie inévitablement les servomoteurs comme système exécutif ou encore opératoire.

Les servomoteurs représentent donc un organe fondamental, notamment exploités dans divers applications, telles que dans le modélisme, la direction des voitures télécommandées, la commande des gouvernes des avions, et voire même dans l'industrie par exemple pour commander les vannes afin de réguler le flux de liquides.

Dans notre projet, le servomoteur est un élément de base pour la conception d'un bras manipulateur qui sert à l'exécution de tâches. Ainsi, on présente dans ce chapitre un aperçu sur le domaine d'application des servomoteurs, puis sur l'étude de ses caractéristiques électromécaniques, ensuite on présente différents types de servomoteurs ainsi que leurs principes de fonctionnement.

1.2 Définitions

Les servomoteurs sont des systèmes peu compliqués qui se composent d'éléments électroniques et mécaniques assemblés dans un petit boîtier, donc c'est un dispositif électromécanique.

Les servomoteurs peuvent actionner des parties mobiles et tournent ou poussent des pièces mécaniques avec une grande précision et à une certaine vitesse. Habituellement abrégé en servo, qui vient du latin et signifiant esclave, c'est un moteur qui vérifié et corrige continuellement sa position. Donc c'est un système asservi. [2]

1.3. Les type de servomoteurs

Il existe de nombreux types de servomoteurs, dont les suivants :

1.3.1. Servomoteur standard (à rotation de position)

C'est le type de servomoteurs le plus courant, l'arbre de sorti tourne environ la moitié d'un cercle (180°), Il comporte des butées placées dans le mécanisme d'engrenage pour éviter de dépasser ses limites, afin de protéger le capteur de rotation (potentiomètre).

Ces servomoteur sont les plus communs et se trouvent dans les voitures, les avions, les robots et bien d'autres applications.

1.3.4. Servomoteur analogique avec signal de feedback

Un signal analogique dépendant de la position du servomoteur permet de connaître la position réelle de celui-ci. Ce signal permet d'améliorer la précision et la stabilité. Applications: robotique, apprentissage de mouvement, enregistrement de position, etc. Le signal de retour analogique est disponible sur un quatrième fil.

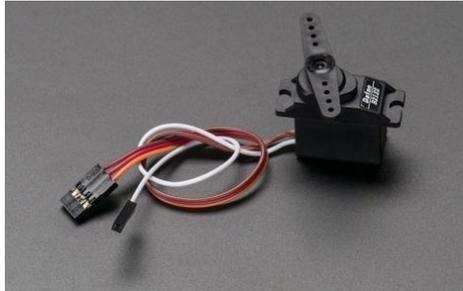


Figure 1.4 Servomoteur analogique avec signal de feedback.

1.4. Les constituants du servomoteur

1.4.1 Moteur électrique à courant continu (MCC)

Le moteur à courant continu est alimenté par une tension constante et tourne à vitesse angulaire proportionnelle à sa tension d'alimentation, par exemple dans un servomoteur standard la tension est de 5 volts et donc le moteur tourne à vitesse élevée. Le problème c'est que comme le moteur est de petite taille, il est beaucoup moins efficace pour la production de couples. C'est la raison pour laquelle il est judicieux de placer un réducteur (train d'engrenages) sur l'axe du moteur pour fournir un bon couple.[1]



Figure 1.5 Moteur à courant continu [1]

1.4.2 Train d'engrenages

Il s'agit d'une cascade d'étages constitués chacun d'un couple de roues dentées. Chaque étage possède son rapport de réduction qui est égal au rapport du nombre de dents des deux roues ou pignons. Au total, le rapport de réduction est le produit de tous les rapports et peut atteindre typiquement une valeur de l'ordre de 400 ou 500.

Sur un plan électromécanique, la puissance de sortie du moteur (P) est donnée par le produit :

$$P=C_m.\Omega_m \quad (1)$$

Où

C_m représente le couple sur l'axe du moteur

Ω_m est la vitesse angulaire de rotation de l'axe du moteur.

Dans le cas où la puissance du moteur est entièrement transférée au palonnier (mais en réalité, il y a toujours des pertes) alors :

$$P=C_p.\Omega_p \quad (2)$$

C_p représente le couple sur l'axe du palonnier

Ω_p est la vitesse angulaire de rotation de l'axe du palonnier.

Aux pertes près, le produit $P=C.\Omega$ est constant. La réduction de vitesse est ainsi associée à la démultiplication du couple dans des rapports identiques. [2]

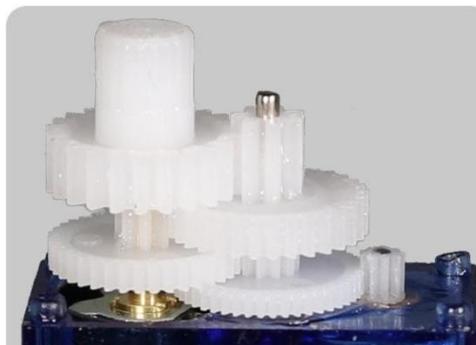


Figure 1.6 Train d'engrenages de servomoteur sg90 [4]

1.4.3 Le potentiomètre

Le potentiomètre est relié au train d'engrenages par l'intermédiaire d'une roue dentée, pour tirer l'angle et la vitesse réelle du servomoteur c'est à dire la position réelle instantanée du servomoteur, il joue le rôle de capteur de position du palonnier. Il délivre une tension proportionnelle à la position angulaire, la tension exploitée par la carte électronique dans la boucle d'asservissement. [2]

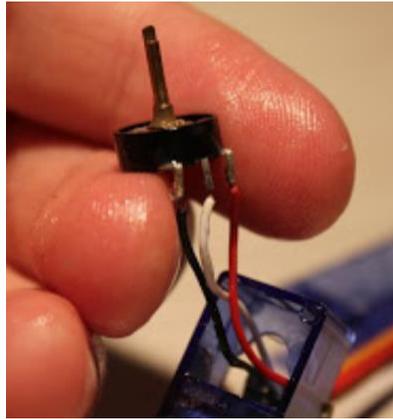


Figure 1.7 Potentiomètre du servomoteur sg90 [4]

1.4.4. Carte électronique

La carte électronique a pour rôle de lire les impulsions électriques et lire aussi la position du moteur pour contrôler et corriger continuellement la position angulaire de l'axe de sortie. Ceci permet de fournir une loi de commande convenable et bien gérer l'asservissement. [2]

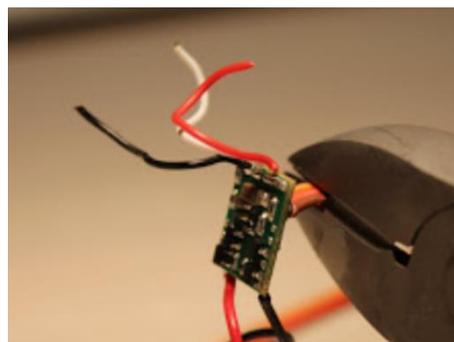


Figure 1.8 La carte électronique du servomoteur sg90 [4]

1.4.5. Butée mécanique (servomoteur standard)

Elle sert à limiter l'excursion en angle de -90° à $+90^\circ$ par rapport à la position dite neutre, soit une excursion de 180° . [2]

1.4.6. Palonnier

Il permet la liaison avec la partie mobile. [2]

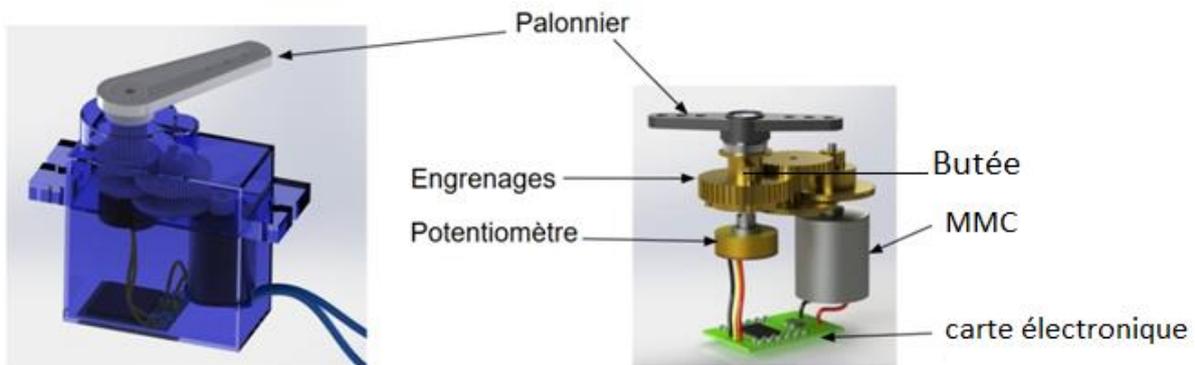


Figure 1.9 Les constituants élémentaires d'un servomoteur

1.5. Principe de fonctionnement d'un servomoteur standard

Le mot servo sous entend un contrôle en boucle fermée et donc un asservissement, Pour décrire le principe très général d'une boucle d'asservissement ou de régulation, on peut se référer au schéma-bloc l ci-dessous. [2]

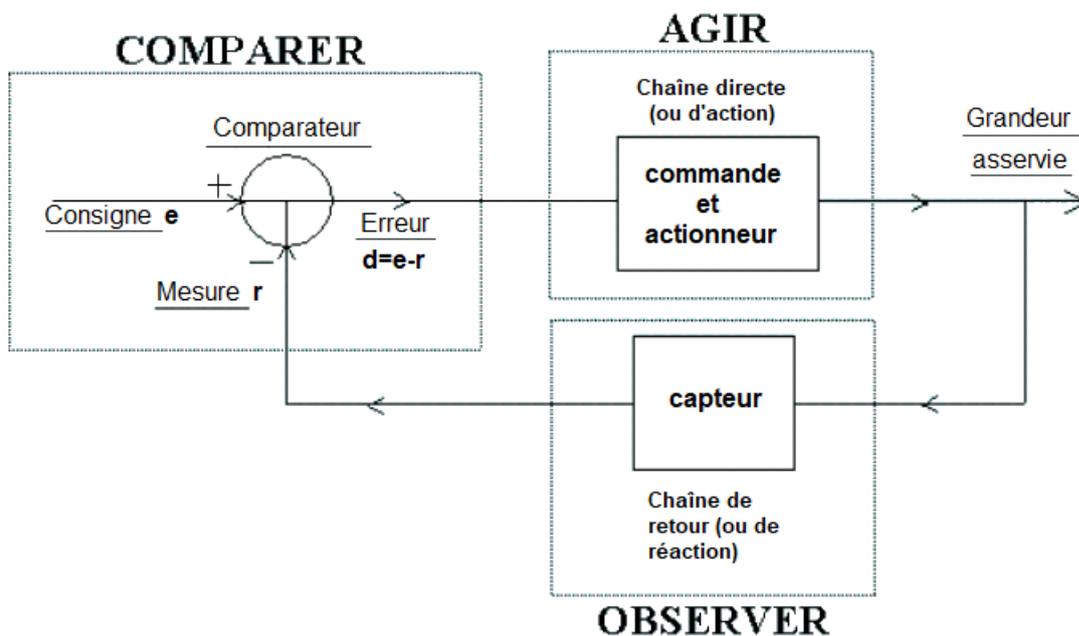


Figure 1.10 La boucle d'asservissement générale de servomoteur. [2]

Mécaniquement, il s'agit d'un moteur avec réducteur. Le mouvement de sortie est une rotation. Lorsque le moteur tourne, l'axe du servomoteur change de position, ce qui modifie la

résistance du potentiomètre. Le rôle de l'électronique est alors de commander le moteur pour que la position de l'axe de sortie soit conforme à la consigne reçue : c'est un asservissement. Ainsi, la commande se fait via une impulsion, ce qui représente la consigne de position codé en largeur d'impulsion appelée MLI (PWM) et délivrée au servomoteur.

Puisque le signal du servomoteur standard est codé en largeur d'impulsion (PWM), cela signifie que c'est la durée des impulsions (en général entre 1 et 2 millisecondes) qui détermine l'angle absolu de l'axe de sortie et donc la position du bras de commande du servomoteur. Le signal est répété périodiquement (en général toutes les 20 millisecondes), ce qui permet à l'électronique de contrôler et de corriger continuellement la position angulaire de l'axe de sortie, cette dernière étant mesurée par le potentiomètre [2], [2].

1.6. Le couple d'un servomoteur

Le couple exprimé en kg.cm, indique la relation entre la longueur du bras du servomoteur et la force exercée par ce dernier.

La formule du couple est la suivante :

$$C=F*R \quad (3)$$

Où

C représente le couple du servomoteur.

F est la force exercée sur le bras du servomoteur en kilogramme.

R est la distance en mètre par rapport à l'axe de rotation du servomoteur, à partir de laquelle s'exerce la force.[5]

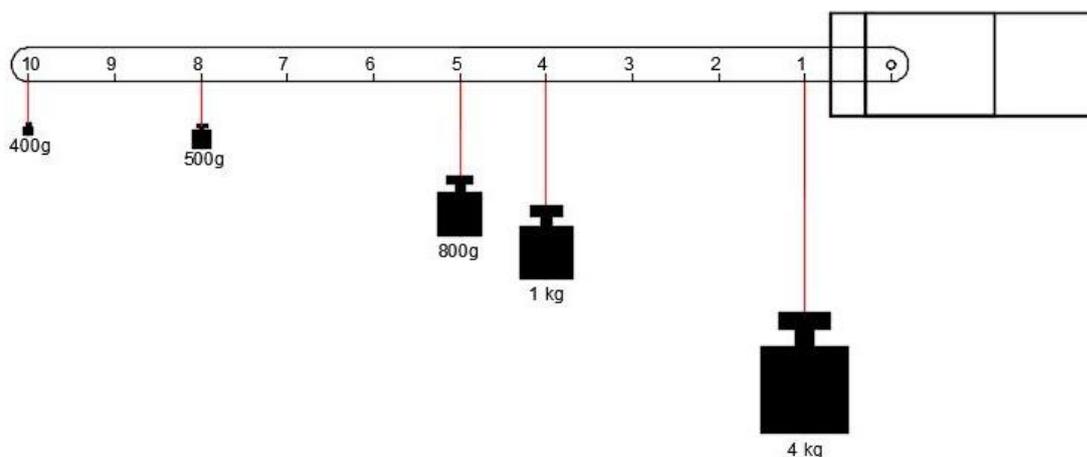


Figure 1.11 Illustration du couple d'un servomoteur. [5]

1.7. Conclusion

Le servomoteur est un dispositif électromécanique, ayant divers domaines d'utilisation à la fois dans l'industrie et dans le modélisme. Dans notre projet, nous allons l'utiliser comme actionneur pour commander les articulations d'un bras manipulateur.

Chapitre 02

L'ARDUINO

2.1. Introduction

Durant des siècles, le développement de moyens reliant les deux mondes celui de la commande et de la programmation n'a cessé de croître considérablement ouvrant la voie à une réelle course mondiale et donnant naissance à une variété de cartes électroniques munies de processeurs à la disposition des chercheurs et des ingénieurs. Dans notre travail, on propose d'utiliser la carte Arduino pour la commande des servomoteurs responsables du mouvement des articulations du bras manipulateur à concevoir. L'Arduino possède une multitude d'avantages aussi bien techniques que scientifiques disposées par ses constructeurs à la portée de tout utilisateur selon la complexité de son application. .

Ainsi, ce chapitre renferme essentiellement la description des cartes de commande des servomoteurs développés par Arduino. Etant donné qu'Arduino fournit une grande variété de cartes électroniques à notre disposition, il est utile de faire une étude comparative des différents types de cartes existantes pour mieux orienter notre choix pour la conception du bras manipulateur de notre projet. Puis on clôture ce chapitre avec un aperçu global sur la programmation de ces cartes.

2.2 Définitions

2.2.1 Arduino

L'Arduino est une plateforme de prototypage électronique open-source d'objets interactifs à usage créatif. Permet aux utilisateurs de créer des objets électroniques interactifs à partir de cartes électroniques et d'un environnement de programmation.



Figure 2.1 Logo de la marque Arduino

2.2.2 La carte Arduino

La carte Arduino est une carte électronique programmable, utilisée pour réaliser des montages électroniques, elle est facile d'utilisation, elle peut être contrôlée à partir de l'ordinateur et programmée par logiciel Arduino IDE qui y est installé, puis déconnectée et autorisée à fonctionner indépendamment.



Figure 2.2 Carte Arduino UNO

2.3 Les applications de la carte Arduino

La carte Arduino peut parfaitement convenir aussi bien à la réalisation d'applications simples, telle que faire clignoter une Led que des applications complexes comme la fabrication de robots proprement dits.

En fait Arduino permet la réalisation d'un grand nombre d'applications et dans tous les domaines, nous pouvons citer quelques unes à titre d'exemples :

- Télécommander des appareils mobiles.
- Commander des actionneurs.
- Réaliser un jeu de lumières
- Surveillance de la température
- Communiquer avec l'ordinateur
- Commander des robots,...etc.

De façon générale, il n'est une application pour laquelle une carte Arduino ne peut être mise à profit pour y venir à bout, il suffit de faire le bon choix de cartes qui va répondre à nos attentes. Dans notre projet, nous visons à utiliser ce type de carte pour contrôler des servomoteurs et à exploiter ce contrôle dans la conception d'un bras manipulateur qui sera largement discuté dans les deux prochains chapitres. [1]

2.4 Les types de cartes Arduino :

Il existe trois types :

- a- Les cartes Arduino officielles, qui sont fabriquées en Italie par le fabricant officiel : Smart Projects.
- b- Les cartes Arduino compatibles, qui ne sont pas fabriquées par Smart Projects, mais qui sont totalement compatibles avec les Arduino officielles.
- c- Les autres cartes, fabriquées par diverses entreprises et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, ...). [1]

2.5. Caractéristiques principales de la carte Arduino

2.5.1 Version

Il existe plusieurs versions de cartes Arduino sur le marché, par exemple, nous entendons toujours parler des appareils iPhone 6, iPhone 12 Max, Windows 7 et Windows 10, il s'agit simplement de différentes versions du même produit. Arduino ne déroge pas à cette règle, seulement il donne généralement des noms après Arduino pour chaque nouvelle version au lieu des nombres, exemples Arduino UNO, Arduino MINI, Arduino NANO,...etc.

2.5.2 Alimentation

Il existe trois façons d'alimenter l'Arduino, et elles sont les suivantes :

- a- Alimentation par USB mais sans ordinateur.
- b- Alimentation par connecteur Jack.
- c- Alimentation par 2 broches de l'Arduino : VIN et GND.

La spécification des cartes Arduinos est qu'elles utilisent toutes du courant continu. Trois broches sont possibles, nous trouverons généralement une broche marquée 5 volts, une broche marquée 3,3 volts, et une autre broche (VIN).

Le microcontrôleur fonctionnant directement sous 5V ou il est alimenté via un régulateur 3,3V dont l'entrée est connectée sur le 5V, la carte peut être alimentée en 5V par le port USB ou bien par une alimentation externe. VIN correspond à l'entrée d'un régulateur, qui se charge ensuite de réduire la tension à 5V pour le bon fonctionnement de la carte. [2]

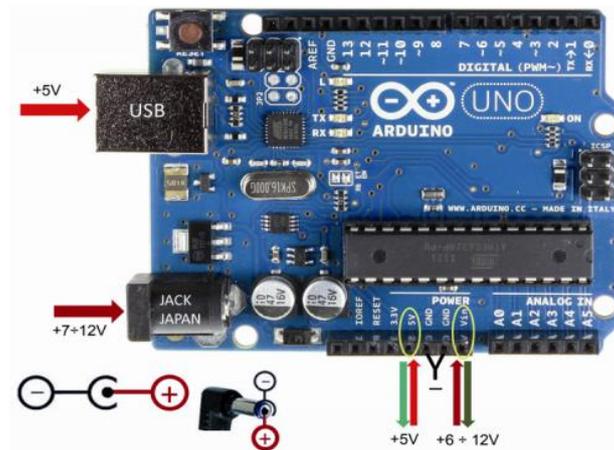


Figure 2.3 Alimentation de l'Arduino Uno

2.5.3 Le microcontrôleur

Le microcontrôleur est le cerveau de la carte Arduino, il reçoit le programme que l'on crée et le stocke dans sa mémoire avant de l'exécuter. Grâce à ce programme, il est capable de prendre des décisions, faire des calculs, compter le temps, comme l'envoi de données à un ordinateur ou la commande des moteurs, etc. [2]

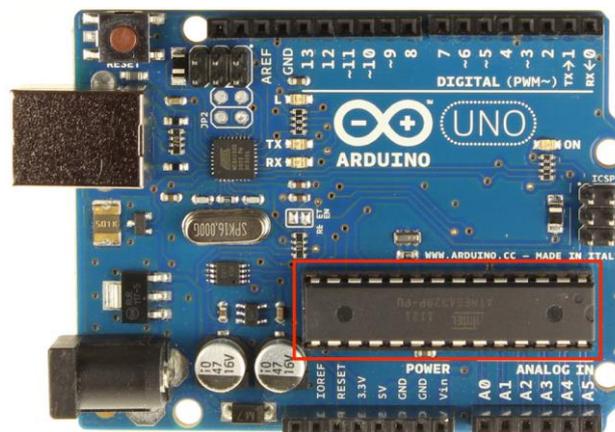


Figure 2.4 Microcontrôleur de la carte Arduino UNO [2]

2.5.4 La mémoire

La mémoire est l'un des principaux composants d'un microcontrôleur, par exemple l'Atmega, cœur de l'Arduino. Les principales caractéristiques d'une mémoire sont:

- La capacité de stockage,
- La vitesse de lecture et d'écriture (mégaoctet par seconde: Mb/s),
- La capacité à conserver les données dans le temps avec ou sans alimentation électrique,
- Le nombre de cycle d'écriture avant obsolescence. [1]

2.5.5 Les ports d'Arduino

À part la led à côté du numéro 13, la carte Arduino ne contient aucun composant pouvant être utilisé pour programmer comme des led et des moteurs. Il faut l'ajouter en les connectant à la carte. C'est là que les connecteurs, également appelés broches.

Sur de nombreuses cartes Arduino, les broches sont au même endroit. En effet, des cartes d'extension peuvent être connectées.

Les broches d'entrée et de sortie de l'Arduino sont les suivantes :

- Entrées/sorties numériques
- Entrées/sorties analogiques
- Terre ou masse (0V)
- Alimentation +5V
- Alimentation +3.3V
- Alimentation non stabilisée Vin (le même voltage que celui à l'entrée de la carte).

Il existe également d'autres broches tel que le montre la Figure 2.3 [2]

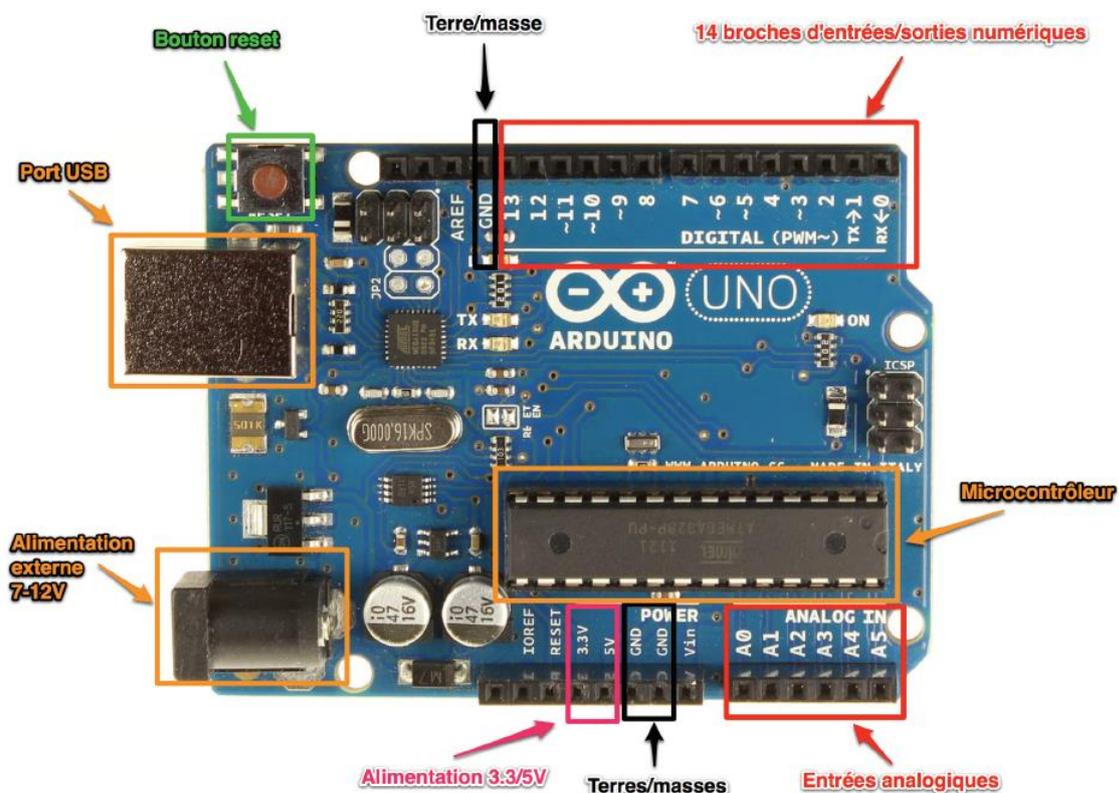


Figure 2.5 Les entrées/sorties de la carte Arduino [2]

2.6. Différentes cartes Arduinos

L'Arduino n'est pas un type spécifique de carte ou de microcontrôleur. Au lieu de cela, l'Arduino désigne plutôt toute une famille comme indiqué dans la Figure 2.6, et choisir la bonne carte Arduino pour notre projet est une étape primordiale et très importante.



Figure 2.6 La famille des cartes Arduinos [2]

2.7. Comparaison des différentes cartes Arduinos

Comme nous l'avons dit, le choix de la carte Arduino est étroitement lié aux attentes que l'on aspire de l'application à commander. Toutes les caractéristiques des cartes Arduinos sont réunies dans un tableau comparatif suivant pour faciliter le choix, voir le Tableau 2.1

Cartes Arduino	Microcontrôleur	Entrées/sorties Numérique	Entrées digitale	Tension de fonctionnement	Spécificité
Arduino Uno	Atmega328p	14/6	6	5V	
Arduino Mini	Atmega328p	14/6	8	5V	
Arduino Nano	Atmega328p	14/6	8	5V	
Arduino Mega2560	Atmega2560	54/15	16	5V	
Arduino 101	Intel Curie	14/4	6	7-12V	Central inertielle & bluetooth
Arduino Lipade	Atmega328p	14/6	6	5V	
Arduino Leonardo	Atmega32u4	14/6	12	5V	
Arduino Due	AT91SAM3X8E	54/12	12	3.3V	
Arduino mico	Atmega32u4	20/7	12	5V	
Arduinio Yun	Atmega32u4	20/7	12	5V	Wiffi&lunix
Arduino Mkr100	SAMD21Cortex	8/12	7	3.3V	
Arduino Mega Akd	Atmega32u4	54/15	16	5V	

TABLEAU 2.1 : Tableau comparatif des différentes cartes Arduinos

2.8. L'Arduino IDE

Après avoir passé en revue les propriétés de l'Arduino et de ses composants, nous verrons comment créer un programme qui contrôle un moteur ou fait clignoter une lumière. Nous suivrons toujours les étapes suivantes, qui sont les mêmes pour chaque projet dans lequel l'Arduino est utilisé. [2]

2.8.1 Téléchargement et installation du logiciel Arduino IDE

Ce logiciel s'appelle l'Environnement Intégré Arduino, ou en anglais Integrated Development Environment (IDE) que nous librement téléchargeable sur le site web suivant <https://www.arduino.cc/en/software> (Figure 2.7).

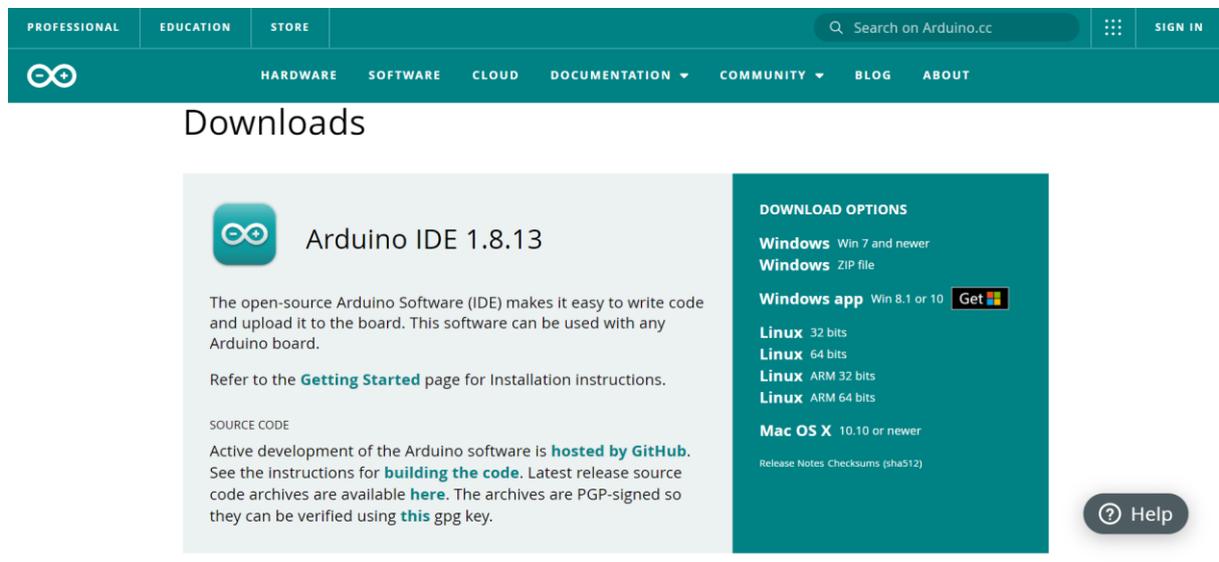


Figure 2.7 Interface IDE d'Arduino

On va choisir notre système d'exploitation Windows 10 (Windows apps) et installer la dernière version du logiciel Arduino IDE (voir Figure 2.8).

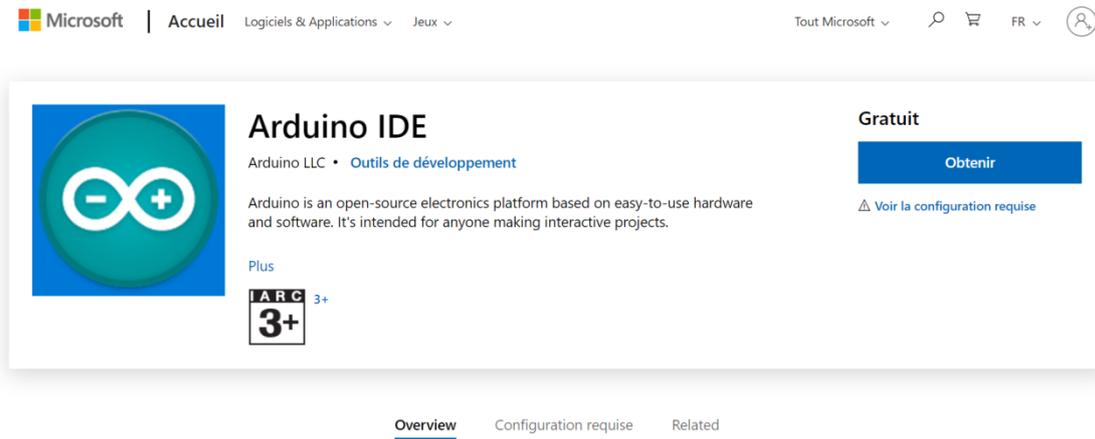


Figure 2.8 Application de l'Arduino IDE

Après l'installation du logiciel, nous relierons le câble USB à notre carte Arduino puis l'autre extrémité à l'ordinateur. Un message indiquant qu'un nouveau périphérique réseau est détecté pourra alors apparaître à l'écran.



Figure 2.9 Liaison de la carte Arduino à l'ordinateur via le câble USB.

2.8.2 Description de l'Arduino IDE

Lorsque nous cliquons sur l'icône de l'Arduino IDE (Figure 2.10), une interface apparaît (Figure 2.12) qui comprend les éléments suivants :



Figure 2.10 Icône du logiciel Arduino IDE

Comme tout autre programme d'interface graphique, il dispose d'une barre de menus. La partie importante de l'IDE Arduino, la barre de menu car elle donne un accès direct aux fonctions essentielles du logiciel. Les différents boutons sont clairement représentés sur la Figure (2.11).

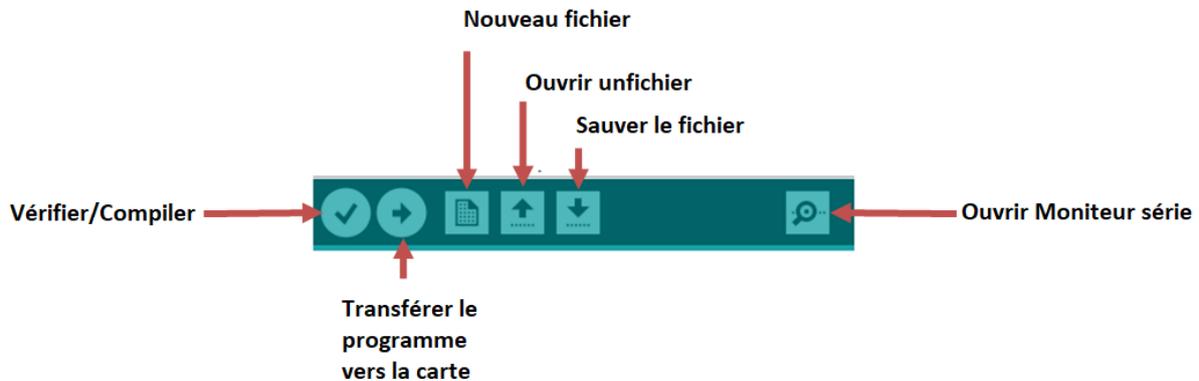


Figure 2.11 La barre de menu de l'IDE Arduino

- Fenêtre d'édition pour écrire le code de notre programme
- Zone de messages qui affiche indique l'état des actions en cours,
- Console texte qui affiche les messages d'état et les erreurs du programme.

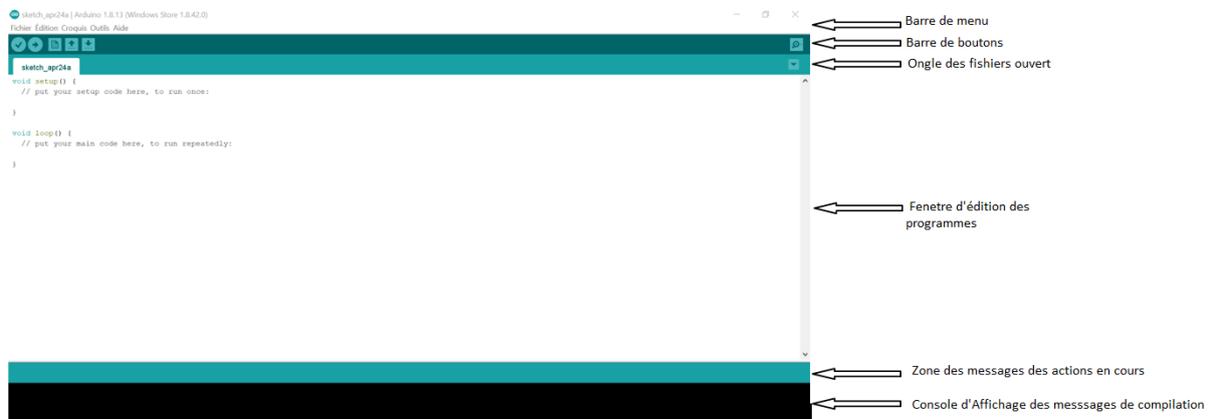


Figure 2.12 Interface de l'Arduino 1.8.13

2.9 Structure d'un programme Arduino

Le programme Arduino utilise une suite d'instructions sous forme textuel, lit ce programme lignes par ligne, puis effectue l'exécution des instructions dans l'ordre défini.

Un programme Arduino est divisé en 3 parties (Figure 2.11)

. Partie déclarations

Cette partie permet des déclarations et initialisation des variables, définir le type des constantes et les variables globales (des nombres, des chaînes des caractèresetc.), elle permet aussi d'intégrer des bibliothèques utilisées dans le programme.

. Partie initialisation et configuration

Il s'agit de la fonction d'initialisation des entrées/sorties, la fonction `setup()`. C'est la première fonction qui va être exécutée au démarrage de l'Arduino et permet de paramétrer les broches en entrée ou sorties, elle s'exécute une seule fois après le démarrage.

. Partie principale et exécution en boucle

La fonction `loop()` est la 2ème fonction et la principale à être lancée par Arduino après la fonction `setup()`. Cette fonction sera répétée indéfiniment, le code de cette dernière est exécuté en boucle, une fois terminé, il recommence [10].

Zone de déclaration et initialisation des variables	{	<code>#include <Servo.h></code>
		<code>int ;</code>
		<code>float ;</code>
initialisation et configuration des entrées/sorties	{	<code>void setup() {</code>
		<code>}</code>
partie principale qui s'exécute en boucle	{	<code>void loop() {</code>
		<code>}</code>

Figure 2.13 Structure d'un programme Arduino

2.10 Conclusion

Dans ce chapitre, nous avons une explication détaillée de la carte Arduino, de la partie matérielle et logicielle, puis nous avons cités les différents types de cette dernière, puis nous avons expliqué le principe de fonctionnement, et au final nous avons déduit un tableau de comparaison pour les différentes cartes Arduino a cause de faciliter le processus de sélection de la carte appropriée à la nature de tout projet.

Chapitre 03

LA ROBOTIQUE

3.1 Introduction

Depuis toujours, les robots ont alimenté l'imagination humaine et ont occupé une grande place dans les centres d'intérêts des scientifiques très tôt déjà. D'ailleurs, la science de la robotique continue son développement à ce jour, car elle ouvre des horizons inimaginables pour l'humanité. Grâce à l'utilisation de la nouvelle technologie mise au service de la robotique, nous avons assisté à l'introduction de robots dans l'industrie de sorte que les robots manipulateurs sont devenus capables de souder, pulvériser, traiter, assembler des pièces conduisant le progrès dans certaines industries à se passer graduellement de l'intervention de l'homme.

Ainsi, les robots manipulateurs ont été dédié à de multiples utilisations, notamment la prédiction des mouvements, le conditionnement des moteurs d'articulation, la planification des tâches, etc.

Dans ce chapitre, nous allons voir qu'est ce qu'un robot, les types de robots de l'industrie et de ce fait présenter les constituants mécaniques des robots manipulateurs en passant par leur modélisation mathématique.

3.2 Aperçu sur l'origine des robots

La première apparition du mot «robot» fut dans "Rossumovi Univerzální Roboti », une pièce théâtrale tchèque créée par Karel Čapek en 1920, et cela signifie «les robots globaux de Rossum». [1]

En fait, un robot n'est autre qu'un système mécanique articulé et actionné, composé de corps et d'articulations et contrôlé par un ordinateur. [1]

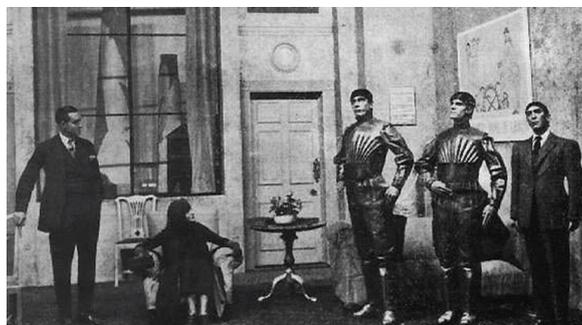


Figure 3.1 Pièce théâtrale créée par Karel Čapek en 1920 [2]

3.2.1 Les types de robots

Il existe plusieurs types des robots, chaque un a son rôle de travail, a savoir :

a- Les robots manipulateurs

Appelés aussi robots industriels, ils sont utilisés pour effectuer des tâches complexes telles que la peinture, le soudage, l'assemblage, etc. leur zone de travail est limitée car ils ont la forme d'un bras manipulateur avec peu de degrés de liberté.

Il existe deux types de robots manipulateurs, qui sont des robots parallèles et séries [1].



Figure 3.2 Robot manipulateur Staubli TX90. [1]

b- Les humanoïdes

On peut dire que les robots humanoïdes sont les types des robots les plus compliqués du point de vue mécanique. Ils sont similaires aux humains dans leurs structures articulées, ils sont beaucoup plus utilisés dans les films ou pour les divertissements dans les supermarchés, etc.



Figure 3.3 Robot humanoïde Nao [3]

c- Les robots mobiles

On peut distinguer deux types de mouvements: les robots marcheurs qui imitent la démarche humaine, et les robots mobiles qui ressemblent davantage à des véhicules. Utilisés dans les zones présentant un risque pour l'être humain, (nucléaire, incendie, sécurité civil, déminage . . . etc.).



Figure 3.4 Le robot de manipulation mobile Momaro [4]

- A. Actionneur = moteur, vérin...
- B. Axe = articulation.
- C. Corps = segment.
- D. Organe terminal.
- E. Effecteur = outil.
- F. Base

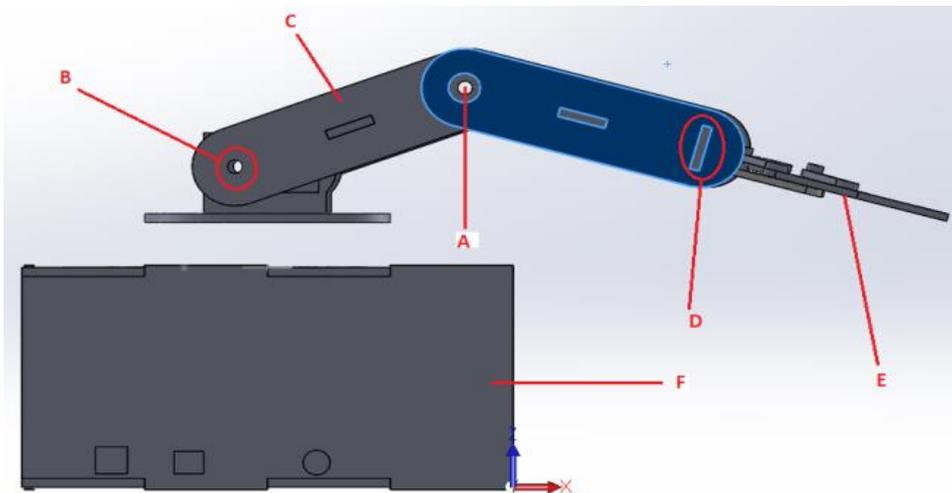


Figure 3.5 Constituants du robot manipulateur

3.3 Constituants mécaniques des robots manipulateurs

Les robots manipulateurs sont constitués de la structure mécanique articulée et l'organe terminal, ces derniers fournissent une interface permettant aux robots de se mettre en mouvement dans leur environnement. La tâche de la structure mécanique articulée est d'amener l'organe terminal à la position d'ingénierie imposée par la tâche en fonction des caractéristiques de vitesse et d'accélération spécifiées. [6]

3.3.1 La structure mécanique articulée

La structure mécanique articulée bénéficie de plusieurs degrés de libertés et équipé de moteurs également appelés actionneurs, elle est formé de lien physique entre le lieu de référence et du travail.

3.3 Les types d'articulations

Une articulation lie deux corps successifs, limitant le nombre de degré de liberté selon leur fonction. En général, le nombre des paramètres indépendants qui fixent la situation de l'organe terminal est égal au nombre de degré de liberté d'un robot manipulateur.

En robotique nous avons deux types d'articulations : rotoïde et prismatique [6], Il existe donc deux types de liaisons mécaniques :

- Liaison "Rotoïde" : Elle réduit le mouvement entre deux corps à une rotation autour d'un axe commun [1], cela se traduit par un angle de rotation.

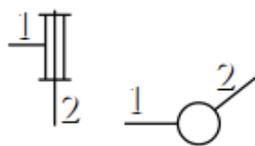


Figure 3.6 Symbole d'articulation rotoïde [5]

- Liaison "prismatique" : Elle réduit le mouvement entre deux corps à une translation le long d'un axe commun [1], cela se traduit par un déplacement.

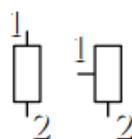


Figure 3.7 Symbole d'articulation prismatique. [5]

3.3.2 Espace de travail des robots manipulateurs

L'espace de travail du robot est tout les points sont espace, que l'Effecteur peut atteindre. Les robots manipulateurs peuvent être classés en fonction de leur espace de travail comme indiqué dans l'image (Figure 3.8). [1]

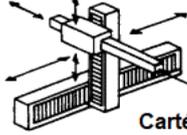
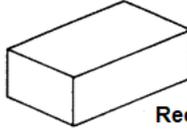
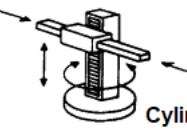
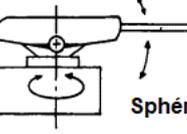
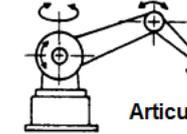
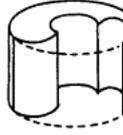
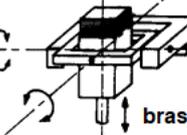
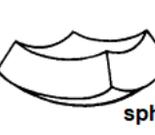
configuration	Espace de travail
 <p>Cartésiens</p>	 <p>Rectangulaire</p>
 <p>Cylindriques</p>	 <p>cylindrique</p>
 <p>Sphériques</p>	 <p>sphérique</p>
 <p>Articulé</p>	 <p>sphérique</p>
 <p>SCARA</p>	 <p>cylindrique</p>
 <p>bras pendulaire</p>	 <p>sphérique partielle</p>

Figure 3.8 Espace de travail des robots.

3.4 Modélisation du robot manipulateur

La conception et la commande des robots, Impose de calculer certains modèles mathématiques, Qui sont au nombre de trois comme suit :

- Modèle géométrique
- Modèle cinématique
- Modèle dynamique

Dans notre étude nous allons nous contenter de présenter le modèle géométrique, car il suffit à répondre aux objectifs fixés dans cette étude.

3.4.1 Description des attributs géométriques

Lorsque nous parlons du domaine de la robotique, nous utilisons la description mathématique des attributs géométriques pour déterminer la position et l'orientation des composants constituant un robot. Cela nécessite une étude détaillée de la structure du robot.

a- La position

Un point P dans l'espace 3D est repéré par trois coordonnées (A_x , B_y , C_z) par rapport à un repère orthonormé orthogonal. La représentation vectoriel du point P est donné par :

$$P = \begin{bmatrix} A_x \\ B_y \\ C_z \end{bmatrix}$$

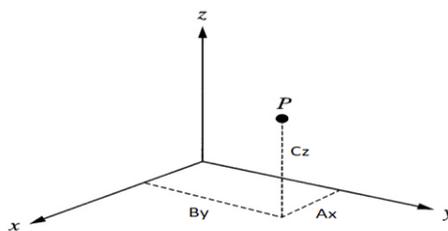


Figure 3.9 Représentation spatiale d'un point.

b- L'orientation

L'orientation de l'organe terminal doit être déterminée pour connaître la localisation de l'effecteur, c'est à dire l'orientation du repère {A} par rapport à {B} [1].

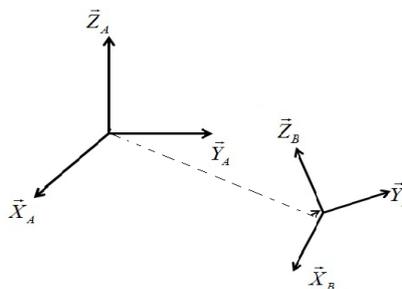


Figure 3.10 Représentation de l'orientation.

La matrice de rotation est définie comme suit :

$${}^A R_B = \begin{bmatrix} X_B X_A & Y_B X_A & Z_B X_A \\ X_B Y_A & Y_B Y_A & Z_B Y_A \\ X_B Z_A & Y_B Z_A & Z_B Z_A \end{bmatrix}$$

c- Matrice de transformation homogène

C'est une matrice carrée de dimension $[4, 4]$, elle représente la transformation par translation ou rotation des deux repères $\{A\}$ vers $\{B\}$ (Figure 34).

$${}^A T_B = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Elle est constituée de 4 sous matrices comme suit:

$$T = \begin{bmatrix} R_{3 \times 3} & P_{3 \times 1} \\ f_{1 \times 3} & 1 \times 1 \end{bmatrix} = \begin{bmatrix} \text{matrice de rotation} & \text{Vecteur de position} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.5 Modèle géométrique

3.5.1 Modèle géométrique directe

Le modèle géométrique directe d'un robot manipulateur permet de calculer à chaque instant t , la position et l'orientation de l'organe terminal d'un robot en fonction de ses variables d'articulations (les consignes des actionneurs) et ses paramètres géométriques. On peut utiliser le modèle géométrique directe pour définir l'espace de travail du robot. [1]

Par exemple, pour un robot de la Figure (3.11), la position P de l'organe terminal est exprimée en fonction des variables d'articulation $\theta_1, \theta_2, \theta_3$,

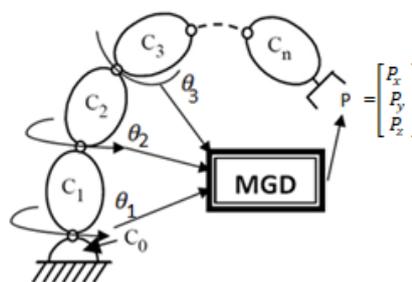


Figure 3.11 Illustration du modèle géométrique directe.

Il existe plusieurs méthodes pour le calcul de MGD d'un robot manipulateur. Denavit Hartenberg (D-H) (1955), est la méthode la plus utilisée pour les robots à chaîne ouverte simple (Figure 3.11).

3.5.1.1 Description de la transformation de D-H

a. Numéroté les articulations de 1 à n du robot.

b. Associer un repère orthogonal orthonormé $\{R_i\}$ au corps i avec :

- L'axe Z_i est porté l'articulation i
- L'axe X_i est porté à la normale commune à Z_i et Z_{i+1} . Si les axes se croisent, X_i est perpendiculaire au plan contenant les deux axes.
- L'axe Y_i est choisi de manière à former un trièdre direct avec l'axe Z_i et Z_i

c. Le passage du repère $\{R_{i-1}\}$ au repère $\{R_i\}$ est exprimé en fonction des paramètres suivants :

a_i : La distance entre Z_{i-1} et Z_i suivant l'axe X_i ;

α_i : L'angle formé par Z_{i-1} et Z_i suivant l'axe X_i ;

d_i : La distance entre X_{i-1} et X_i suivant Z_{i-1} ;

θ_i : L'angle formé par X_{i-1} et X_i suivant Z_{i-1} . [1]

La matrice de transformation homogène défini le $\{R_{i-1}\}$ dans $\{R_i\}$ et exprimé par :

$$Rot(X_i, \alpha_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot(Z_{i-1}, \theta_i) = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Tr(Z_{i-1}, d_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Tr(X, A) = \begin{bmatrix} 1 & 0 & 0 & \alpha_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{i-1}T_i = Rot(Z_{i-1}, \theta_i) * Tr(Z_{i-1}, d_i) * Tr(X_i, a_i) * Rot(X_i, \alpha_i) =$$

$${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\theta_i & \cos\theta_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

[5]

La matrice de transformation homogène du robot manipulateur est obtenue par l'équation suivante :

$${}^0T_n = {}^0T_1 * {}^{n-2}T_{n-1} * {}^{n-1}T_n = \begin{bmatrix} {}^0R_n & {}^0P_n \\ 0 & 1 \end{bmatrix}$$

A partir de la matrice de transformation homogène (0T_n), le MGD d'un robot est obtenu. Les coordonnées du vecteur 0P_n représentent la position et 0R_n l'orientation de l'effecteur ou de l'organe terminal.

3.5.2 Modèle géométrique inverse

Le modèle géométrique inverse (MGI) consiste à déterminer les variables articulaires ($\theta_1, \theta_2, \theta_3$), pour générer l'ensemble des consignes des actionneurs pour que l'effecteur atteigne une position et des orientations données $[P_x, P_y, P_z]$. [1]

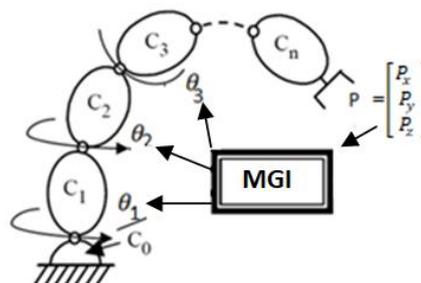


Figure 3.12 Illustration du modèle géométrique inverse.

Il existe trois méthodes de calcul du MGI :

- La méthode de Paul, qui convient pour la plupart des robots industriels.
- La méthode de Pieper, qui permet de résoudre le problème pour les robots à six degrés de liberté avec trois articulations rotoïdes ou trois articulations prismatiques.
- La méthode de Raghavan et Roth, donnant la solution générale des robots à six articulations à partir d'un polynôme de degré au plus égale à 16.

Pour le calcul du MGI, trois cas se présentent :

- Absence de solution, lorsque la position de l'organe terminal désirée est en dehors de la zone accessible du robot.
- Infinité de solutions.
- Solutions en nombre fini. [2]

3.6 Conclusion

Dans ce chapitre, nous avons présenté les différents types des robots en général, et les robots manipulateurs en particulier. Etant donné qu'e le robot manipulateur fait l'objet de notre étude nous avons également présenté une idée sur leurs caractéristiques, leurs fonctionnements et principalement leurs modélisations géométriques en vue de contrôler les articulations des robots manuellement ou automatiquement.

Chapitre 04

REALISATION PRATIQUE DU BRAS MANIPULATEUR

4.1 Introduction.

Le quatrième chapitre représente l'accomplissement effectif de notre travail car il est entièrement dédié à la conception du bras manipulateur en commençant par le croquis de ce type de robot et de sa modélisation géométrique puis à la conception 3D des différentes pièces qui le constituent avec le logiciel Solidwork. Ensuite on a procédé au choix de la matière pour sa fabrication et à l'impression 2D des pièces par une machine CNC à trois axes, l'étape finale étant l'assemblage des pièces. Dans ce quatrième chapitre le travail est divisé en quatre parties

Partie I Réalisation mécanique du robot manipulateur

Cette partie contient une description des étapes de la fabrication mécanique du robot manipulateur tout en passant en revue les différentes phases de sa réalisation.

Partie II Modélisation du robot manipulateur

La deuxième partie concerne la modélisation géométrique de notre robot manipulateur en question.

Partie III Sélection des outils et composants de commande

Dans la troisième partie, on a sélectionné tous les outils et matériels électroniques compatibles et nécessaires pour la réalisation du circuit de commande du robot manipulateur.

Partie IV Réalisation du circuit électronique du bras manipulateur

Enfin, la dernière partie est dédiée au développement du programme de commande des servomoteurs qui vont commander les articulations du robot ainsi qu'à la réalisation du circuit électronique autour de la carte Arduino UNO.

Partie I Réalisation mécanique du robot manipulateur

4. I.1 Anatomie fonctionnelle

Les robots manipulateurs sont d'origine des bras, d'où l'appellation de bras manipulateur. Il est donc question de créer un objet qui imite le bras humain. Le bras se compose de trois os principaux avec des connexions intermédiaires, en fait ces os sont actionnés par des articulations (épaule, coude, main).

Le schéma de la Figure 4.1, montre les différentes articulations du bras humain correspondant aux connexions entre les parties du robot manipulateur.

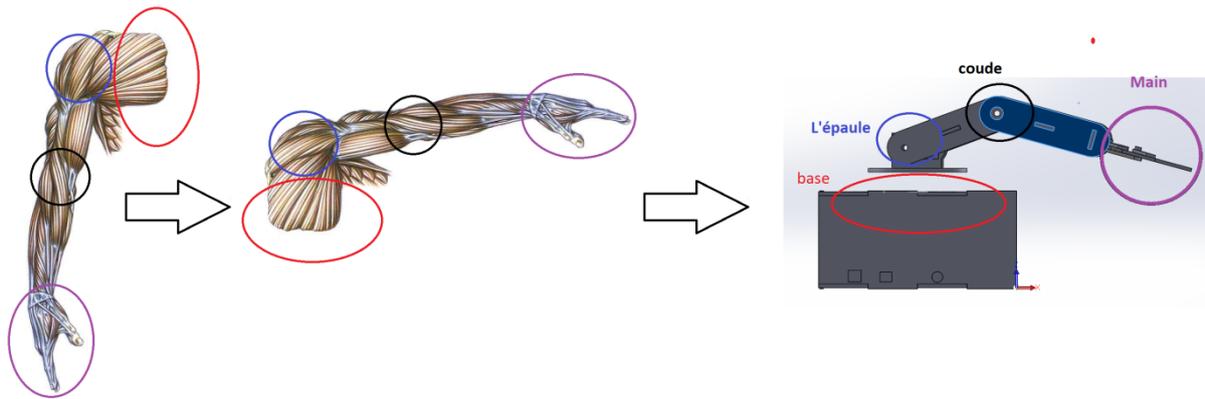


Figure 21 Illustration de la structure du robot manipulateur.

4. I. 2 Fabrication mécanique

La fabrication mécanique du robot est une industrie en soit, elle rassemble plusieurs disciplines et implique des connaissances polyvalentes telles que la conception, la fabrication, l'assemblage, l'ajustement et la réparation des pièces.

Le développement de la fabrication mécanique des robots manipulateurs a connu la modernisation de ses équipements qui sont devenus à la base de la fabrication et ce par l'introduction de la technologie de pointe, telles que les programmes de conception 3D, les machines de production mécanique, les machines d'impression CNC à six axes, etc.

Il est donc clair que la conception d'un bras manipulateur va nous nécessiter beaucoup de compétences et d'acquisitions préalables en la matière. C'est pourquoi nous nous sommes orientés vers les spécialistes afin d'obtenir l'aide indispensables essentiellement lors de la phase d'impression 2D des différentes pièces qui constituent le bras, grâce à la machine CNC à trois axes.

4. I. 3 Etapes de fabrication du bras manipulateur

La connaissance des étapes de conception du bras manipulateur nous permette de classer nos priorités lors de sa fabrication, les étapes sont donc les suivantes :

4. I. 3. 1 La modélisation par ordinateur

La modélisation informatique vient après la validation des dessins techniques, en l'absence de ces derniers, nous prenons comme référence les études précédentes et disponibles. Les pièces sont modélisées par des logiciels de conception 3D professionnels tels que:

- Fusion 360
- KeyCreator
- Inventor
- Solid Edge
- SolidWorks
- CATIA
- ProE

Nous avons utilisé SolidWorks, étant donné que nous l'avons déjà manipulé durant notre formation en robotique, et ça nous a permis de le maîtriser encore plus grâce à ce projet.

a. Présentation de SolidWorks

SolidWorks est un logiciel de modélisation 3D parfaitement adaptés aux ingénieurs, pour la conception des modèles 3D. L'interface de SolidWorks facilite le développement des prototypes les plus complexes, par exemple les pièces de machinerie et convient également aux étudiants souhaitant réaliser des schémas mécaniques. [1]

b. Création des pièces du robot manipulateur par SolidWorks

Nous allons détailler à titre d'exemple les étapes de modélisation d'une partie de la pièce représentant la base sur laquelle est monté le bras manipulateur, qui est la face arrière du socle, pour illustrer l'utilisation de SolidWorks. De la même manière, nous modélisons toutes les 13 autres pièces qui forment le bras, en représentant directement le résultat final des figures correspondantes sans s'attarder sur les détails. Ensuite nous réaliserons avec le même logiciel, l'assemblage de toutes les pièces pour obtenir le prototype complet du modèle du bras manipulateur que nous allons concevoir.

Après installation du logiciel SolidWorks, l'icône de la Figure (4.2) apparaît dans le bureau.



Figure 4.2 Icône de SolidWorks 2018.

Après un double-clic sur l'icône avec le bouton gauche de la souris, le logiciel affiche l'interface SolidWorks (Figure 4.3).

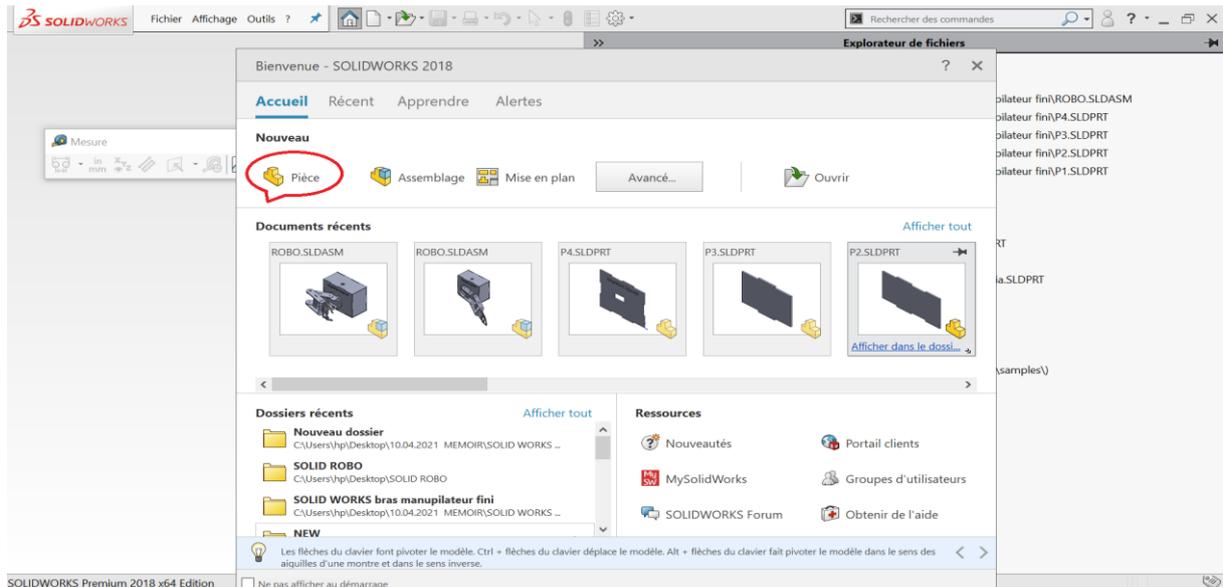


Figure 4.3 L'interface de SolidWorks.

Pour créer une nouvelle pièce, nous cliquons sur "Pièce", une fenêtre apparaît (Figure 4.4).

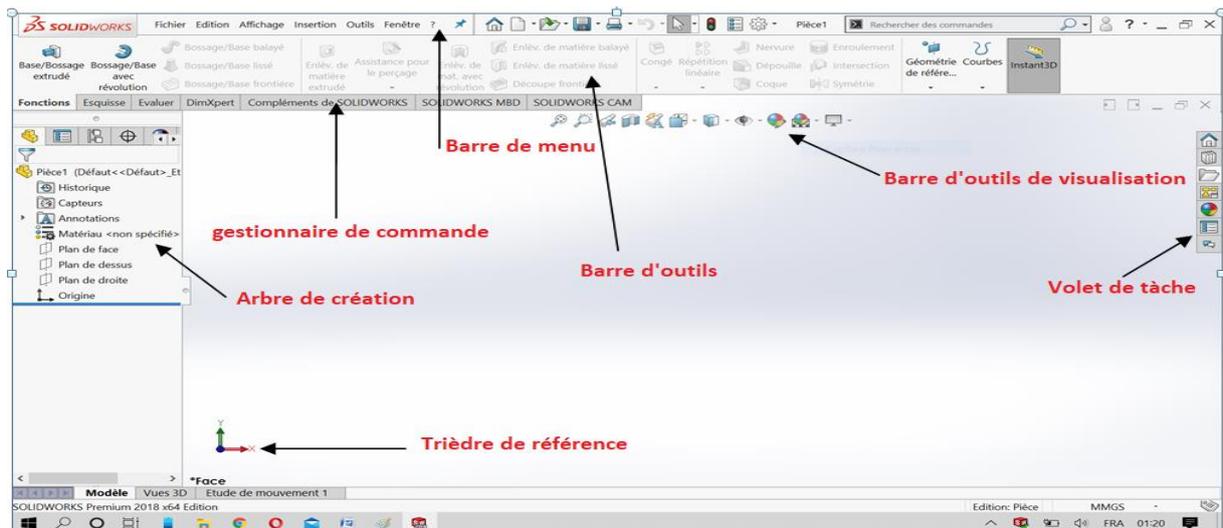


Figure 4.4 Les différentes parties de l'interface de SolidWorks.

Pour modéliser notre pièce, nous commençons par choisir le plan de face, ensuite dans « bandeau principal » on va commencer par esquisser un rectangle (Figure 4.6).

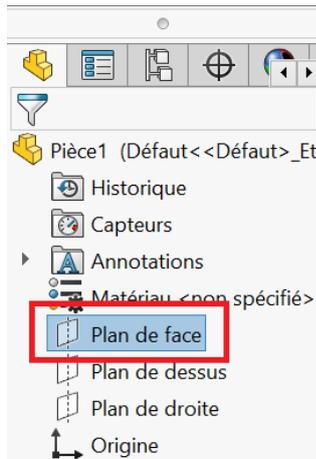


Figure 4.5L'arbre de création.

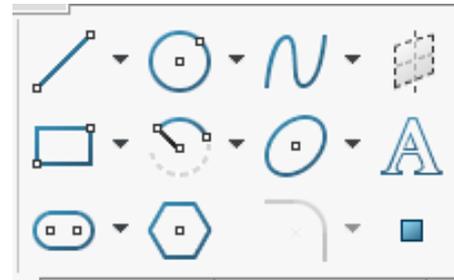


Figure 4.6 Outils de dessin d'esquisse.

Dans l'espace de travail au centre de l'écran, on dessine un rectangle et on donne des mesures précises suivant nos besoins avec la fonction « **Cotation intelligente** » (Figure 4.7).



Figure 4.7 Barre d'outils.

Ensuite, on donne toutes les mesures de la première pièce à concevoir et on obtient le schéma de la Figure (4.8).

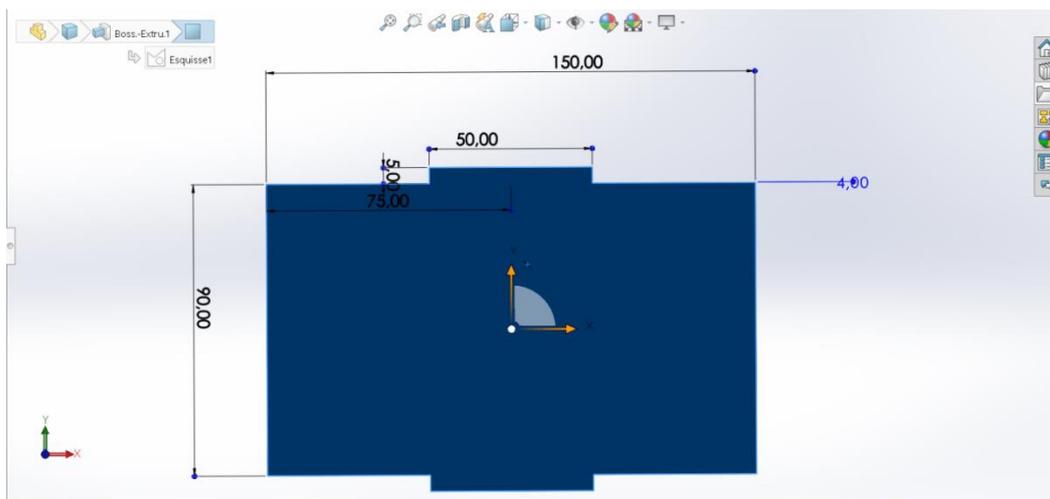


Figure 4.8 Les mesures de la première pièce du robot (La face arrière du socle).

Pour donner de l'épaisseur à notre pièce, on revient au bandeau principal, on clique sur la fonction Base/Bossage extrudé (Figure 4.9), elle prend l'épaisseur dont on a besoin, c'est-à-dire 4 mm (4.10).

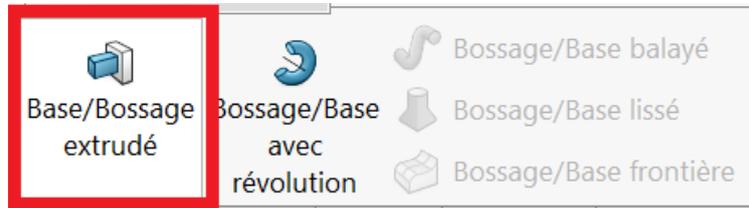


Figure 4.9 Outils d'ajout de matière.

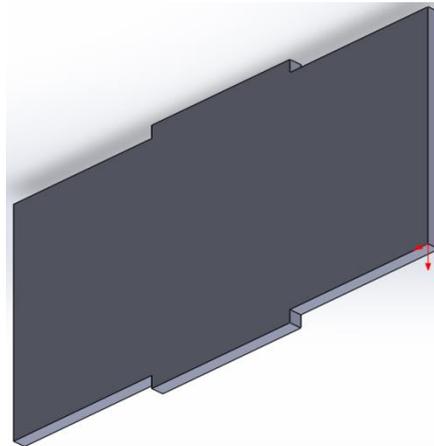


Figure 4.10 Ajout de matière.

Pour d'autres pièces, où il est nécessaire de retirer une partie ou encore de créer un trou, nous devons ajuster la forme géométrique sur l'endroit choisi puis on clique sur la fonction 'Enlèvement de matière extrudé' (Figure 47).

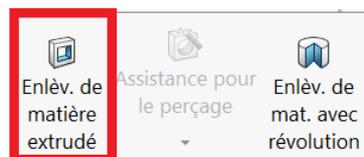


Figure 4.11 Outil d'enlèvement de matière.

Nous pouvons parfois avoir besoin de tracer une ligne, d'ajouter une phrase, un texte ou bien un motif alors on clique sur la fonction "Texte" et un panneau apparait sur le côté (Figure 12 et 13). La Figure 14, montre l'exemple de l'affichage de « université badji mokhtar annaba sisi amar ».



Figure 4.12 Outil de dessin d'esquisse d'une ligne.

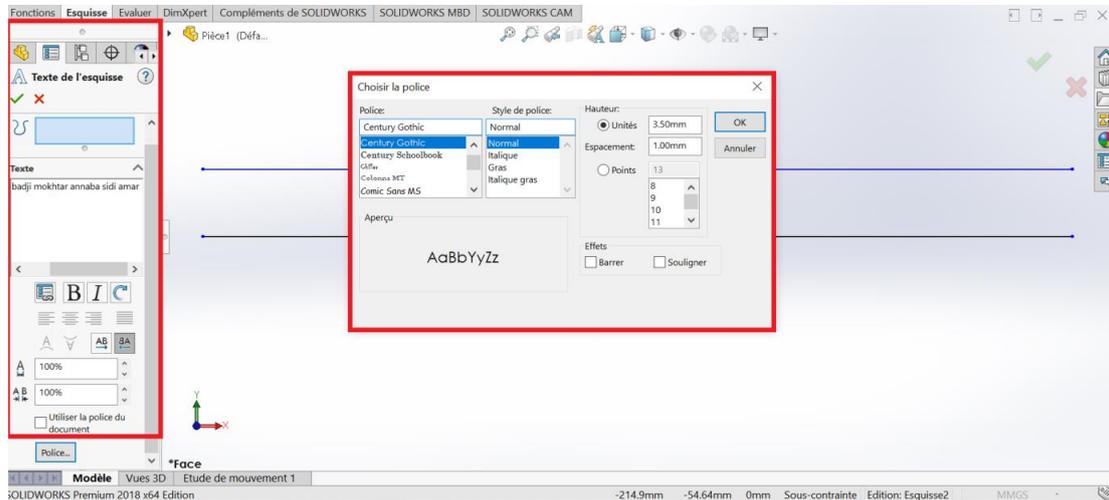


Figure 4.13 Outil de choix de la police.



Figure 4.14 Exemple d'écriture avec Solidworks.

Ainsi, grâce à SolidWorks on est en mesure de modéliser l'ensemble des 14 pièces qui constituent le robot manipulateur, à savoir :

Première pièce : c'est la base du socle (Figure 4.15), elle est censée supporter:

- l'Arduino Uno.
- la plaque d'essai.
- le support de pile.

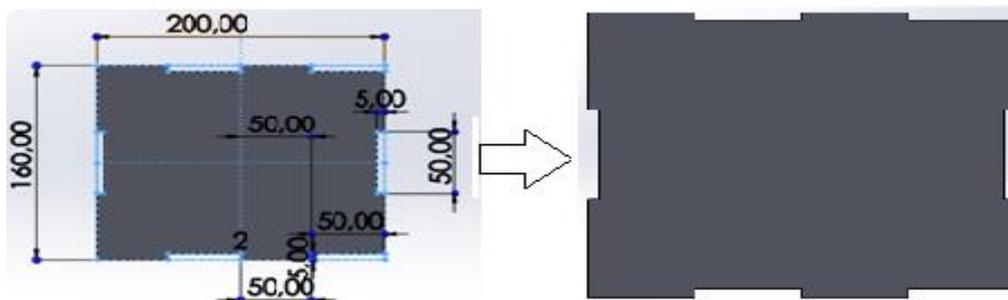


Figure 4.15 La base du socle.

Deuxième pièce : c'est le coté gauche du socle, (Figure 4.16), contient 3 trous :

- Connecteur d'alimentation de l'Arduino Uno.
- Port USB de l'Arduino Uno.
- Connecteur jack femelle.

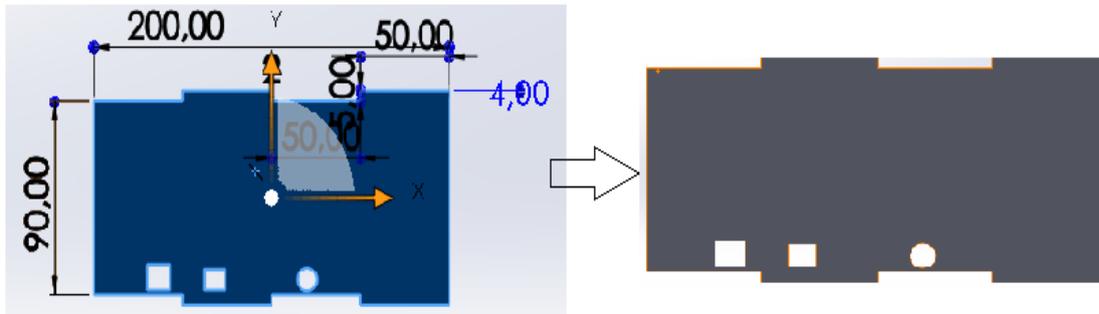


Figure 4.16 La pièce du coté gauche du socle.

Troisième pièce : c'est la face du socle, (Figure 4.17), contient l'emplacement de :

- 4 Leds.
- 4 boutons poussoirs.
- 4 Potentiomètres.
- Afficheur LCD.

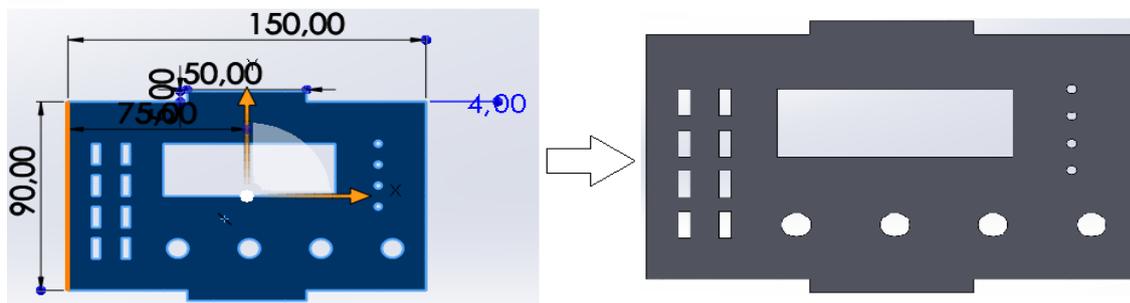


Figure 4.17 La pièce de l'interface du socle.

Quatrième pièce: c'est la pièce supérieure du socle (Figure 4.18), contient un trou rectangulaire qui devient lui-même le support du servomoteur.

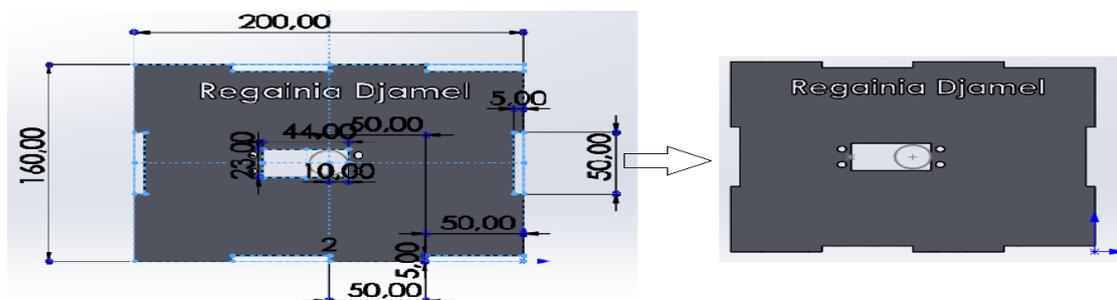


Figure 4.18 La pièce supérieure du socle.

Cinquième pièce: c'est le coté gauche du socle, Figure 4.19 , contient le trou de la traversée des câbles d'alimentation.

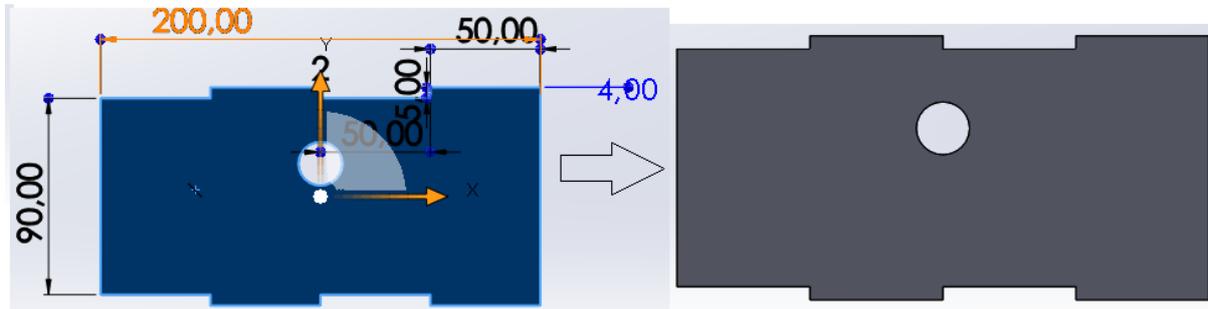


Figure 4.19 Coté droit du socle.

Sixième pièce : c'est la face arrière du socle (Figure 4.20).

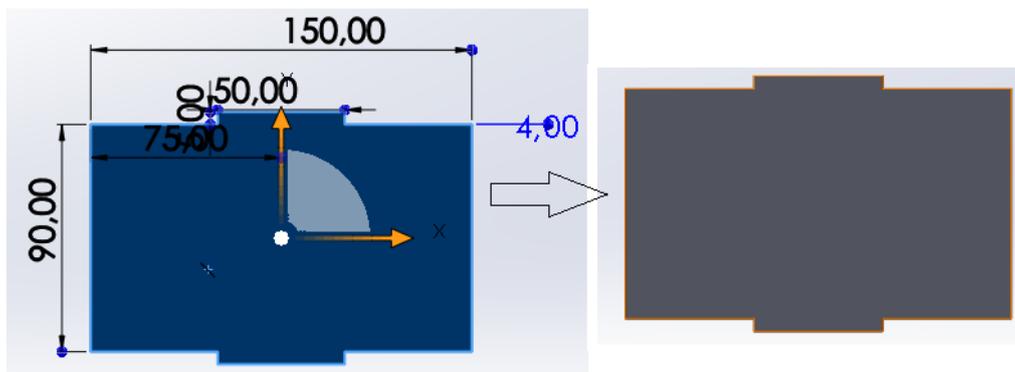


Figure 4.20 Face arrière du socle.

Septième pièce: c'est la petite pièce circulaire du robot manipulateur (Figure 4.21), qui a pour rôle les fixations de la 8^{ème} et 9^{ème} pièce. Elle est raccordée avec le servomoteur au niveau du trou central.

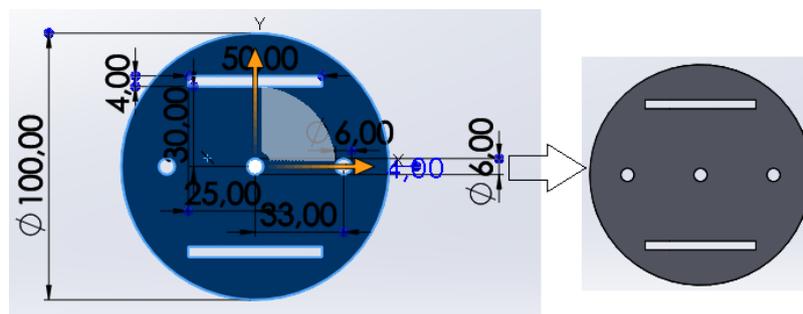


Figure 4.21 La pièce de la petite base.

Huitième pièce ; c'est le support du servomoteur (Figure 4.22).

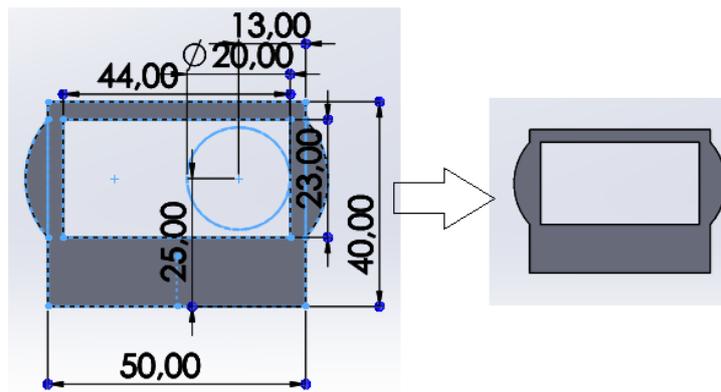


Figure 4.22 support du servomoteur.

Neuvième pièce : elle contient l'emplacement du trou du boulon (Figure 4.23), pour la fixation de la dixième pièce.

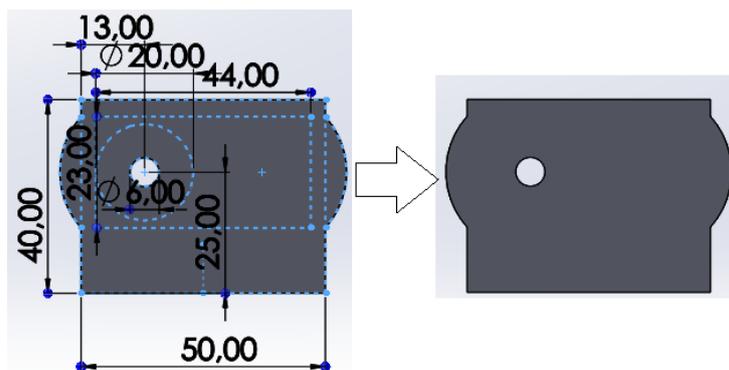


Figure 4.23 La pièce reliant la 10ème pièce à la base.

Dixième pièce : c'est la connexion entre la deuxième et la troisième articulation, un segment (Figure 4.24).

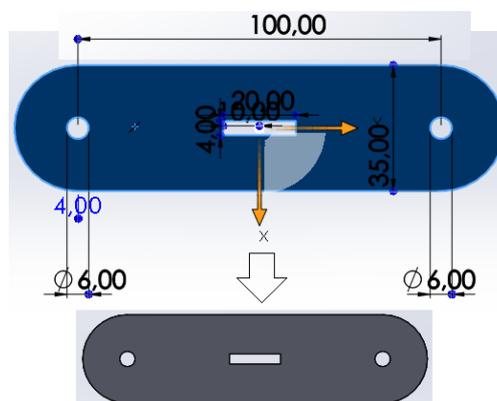


Figure 4.24 Un segment du robot.

Onzième pièce : c'est le segment qui contient le support du 3^{ème} servomoteur (Figure 4.25).

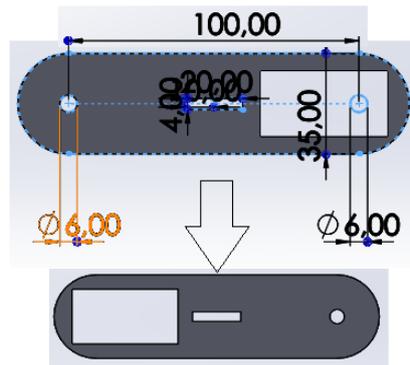


Figure 4.25 Segment du support du 3^{ème} servomoteur.

Douzième pièce : nous devons réaliser deux pièces jumelles pour permettre la fixation de l'organe terminal. Les dimensions de celles-ci sont identiques à la 10^{ème} pièce à l'exception du trou de l'organe terminal, (Figure 4.26).



Figure 4.26 Segment de fixation de l'organe terminal.

Treizième pièce : cette pièce sert à lier les segments pour assurer l'écartement et la stabilité de l'ossature (Figure 4.27).

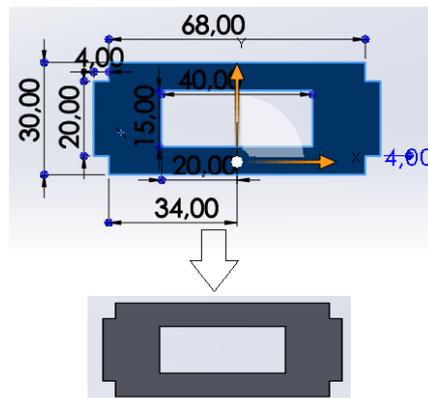


Figure 4.27 Raccord des segments

Quatorzième pièce : c'est l'organe terminal du robot manipulateur, il contient les quatre trous d'assemblage avec la pince (Figure 4.28).

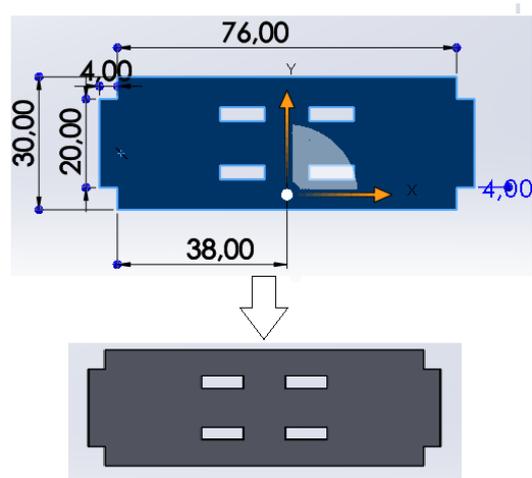


Figure 4.28 : pièce d'Organe terminal du robot manipulateur.

c. Assemblage des pièces du robot manipulateur par SolidWorks

Une fois que la première étape de la conception des pièces maîtresses étant achevée, nous pouvons entamer la deuxième étape qui est l'opération de l'assemblage. Avec SolidWorks, pour créer un nouvel assemblage, nous appuyons sur "Assemblage, (Figure 4.29).



Figure 4.29 Les 3 modules élémentaires d'accueil.

On va voir apparaître une nouvelle interface pour assembler nos pièces, nous allons suivre les étapes suivantes, (Figure 4.30) :

- Insérer les pièces déjà réalisées avec la fonction "insérer des composants".
 - Contraindre les pièces avec fonction " Contrainte ".
- Après avoir assemblé les pièces, nous obtenons ce prototype (Figure 4.31).



Figure 4.30 Outils d'insertion et de contrainte entre les composants.

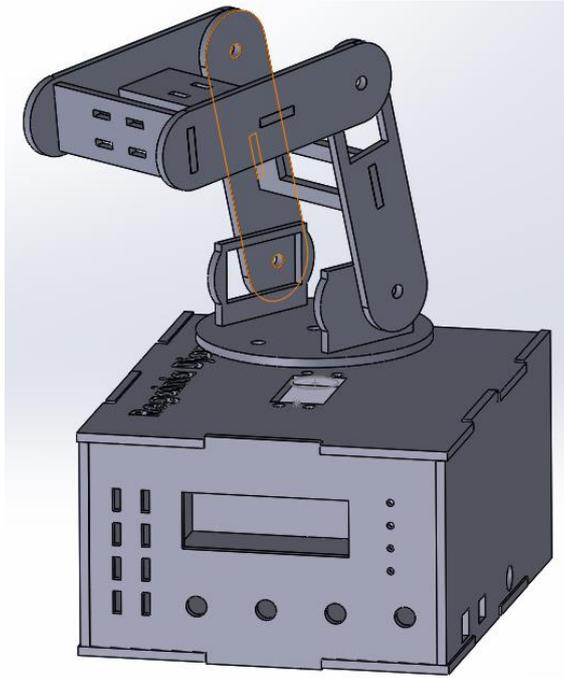


Figure 4.31 Forme finale du robot sans l'effecteur.

Après avoir obtenu la forme du prototype, nous passons à la conception des pièces de l'effecteur. Compte-tenu de la complexité de la conception des pièces de la pince et parce qu'elle nécessite beaucoup de temps, nous nous sommes contentés de reprendre le modèle de [Alvaro Rubio] d'une pince similaire à celle que l'on veut utiliser dans notre projet (Figure 32).

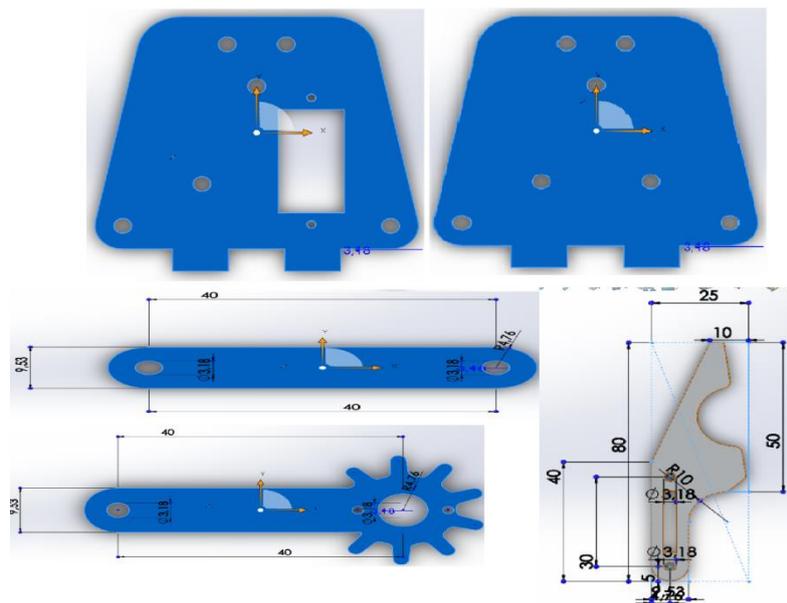


Figure 4.32 Les différentes pièces de l'effecteur.

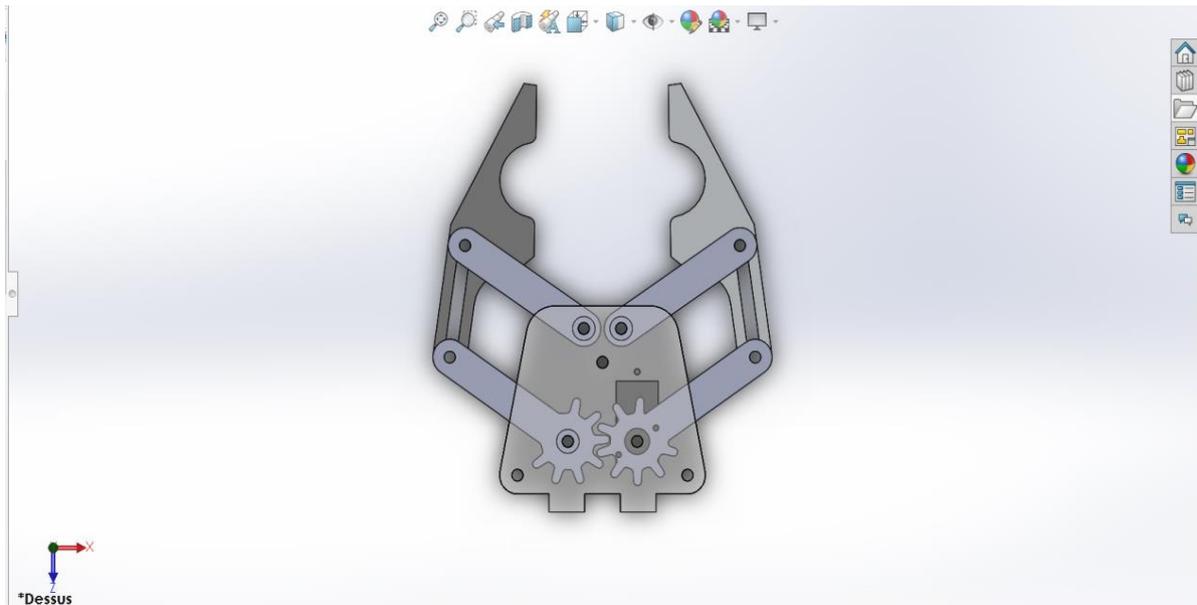


Figure 4.33 Assemblage des pièces de l'effecteur.

Finalement, nous raccordons les pièces de la pince avec le prototype du robot avec les différentes contraintes de SolidWorks, pour obtenir la forme finale de prototype de notre robot manipulateur (Figure 4.34).

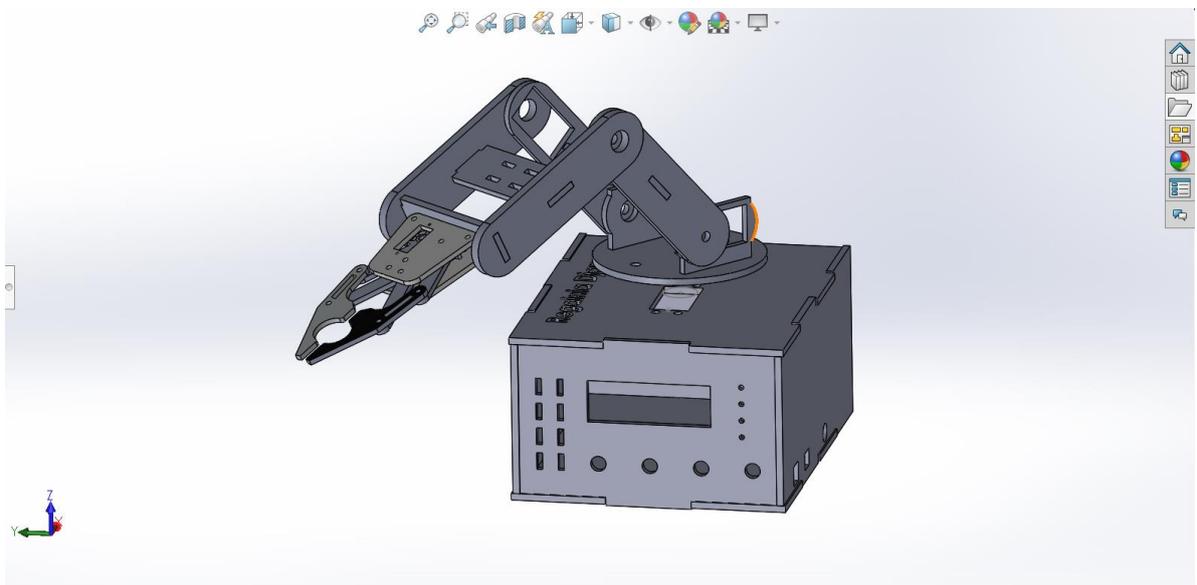


Figure 4.34 La forme finale du robot manipulateur avec effecteur.

4. I. 3. 2 Choix des outils d'impression et de la matière des pièces du robot

Il existe de nombreuses machines que nous pouvons utiliser pour fabriquer les pièces de notre robot manipulateur, ainsi l'une des machines suivantes peut être utilisée:

- Tours conventionnels
- Fraiseuses
- Poinçonneuses
- Rectifieuses
- Presses hydrauliques

Nous remarquons que toutes les pièces du robot sont plates, pour cela nous choisirons la Fraiseuse car elle répond à tous nos besoins pour aboutir au résultat souhaité.

a. La Fraiseuse

Le fraisage est un procédé d'usinage par enlèvement de matière. Il se caractérise par le recours à une machine-outil : la fraiseuse. En fraisage, l'enlèvement de matière, sous forme de copeaux, résulte de la combinaison de deux mouvements : rotation de l'outil de coupe d'une part, et avance de la pièce à usiner d'autre part.

La fraiseuse est particulièrement adaptée à l'usinage de pièces prismatiques et permet également, si la machine est équipée de commande numérique, de réaliser tout type de formes mêmes complexes.

Les fraiseuses actuelles sont fréquemment automatisées (fraiseuses CNC à Commande Numérique et Centres d'usinage). La programmation de commande numérique de ces machines nécessite le recours à des interfaces logicielles, d'une part embarquées sur la machine elle-même (Directeur de Code Numérique), et d'autre part extérieure à la machine, c'est de la fabrication assistée par ordinateur 2D et 3D (PC + logiciels). Dans l'industrie, les ouvriers fraiseurs qualifiés effectuent fréquemment des travaux de tournage complémentaires [2]. Dans notre cas, on utilise une fraiseuse à commande numérique à 3axes (CNC) de type SK-N1325 (Figure 4.35).



Figure 4.35 Machine CNC SK-N1325.



Figure 4.36 Fraise du Fraiseuse.

b. Programmation de la machine

A ce stade du travail, il est question de procéder à l'enregistrement des pièces modélisées avec SolidWorks, sous forme de fichiers sources avec l'extension « DXF ». Prenons l'exemple de la 10^{ème} pièce représentée dans la Figure (4.36), il convient maintenant de créer le programme de commande avec le logiciel ArtCam, puis de transférer ces fichiers pour effectuer la connexion avec la machine d'impression CNC à travers la DSP de cette dernière (Figure 4.37). La dernière étape étant le lancement de la machine pour la fabrication des pièces selon les données programmées.

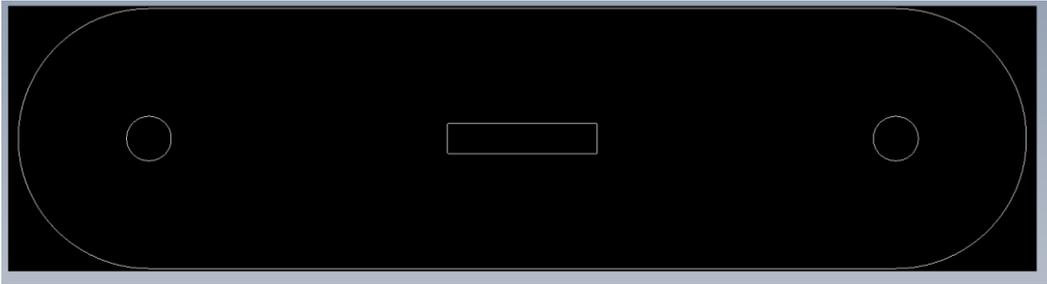


Figure 4.37 10ème pièce sous forme Fisher source « DXF ».



Figure 4.38 DSP.

c. Choix du matériau d'impression

Pour choisir la meilleure matière servant à la fabrication du robot manipulateur, on a testé 3 matières différentes (Figure 4.39).

- Le bois
- Le MDF (panneau de Fibre à Densité Moyenne).
- L'Alucobande

Après plusieurs tentatives il semble que le matériau le plus approprié soit l'Alucobande. Ce matériau est un panneau composite composé de deux tôles aluminium et d'un noyau de remplissage minéral difficilement inflammable. Ce matériau présente les avantages suivants :

- Facilité de mise en œuvre
- Large gamme de surface
- Difficilement inflammable
- Poids faible
- Haute formabilité et stabilité
- Diversité de couleurs brillantes



Figure 4.39 Les matériaux testés dans la fabrication du robot.

4. I. 3. 2 Assemblage des pièces

Après avoir terminé la fabrication de toutes les pièces du robot manipulateur par la machine CNC, on procède à l'assemblage des différentes pièces en utilisant les accessoires représentés dans la Figure (4.39), de la colle rapide "SEGAFIX 5000" et des boulons de différents diamètres (10 et 3). Enfin, nous atteignons la forme finale du robot représenté dans la Figure (4.41).



Figure 4.40 Accessoires d'assemblage.



Figure 4.41 Assemblage final du bras manipulateur.

Partie II Modélisation du robot manipulateur

4. II.1 Introduction

Dans cette partie, on présente la modélisation géométrique par modèle directe et inverse du bras manipulateur, conçu et présenté dans la première partie. Cette étape est indispensable pour le

4. II.2 Modèle géométrique directe de Denavit-Hartenberg

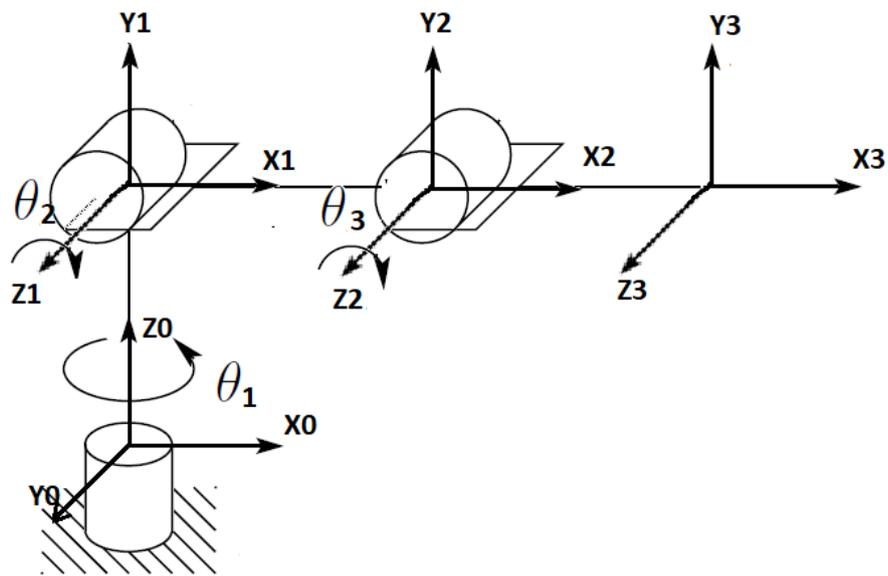


Figure 4.42 Repères en termes des paramètres D-H.

Les paramètres géométriques de DH du robot sont donnés dans le Tableau 2.

	a_i	α_i	d_i	θ_i
0_1T	0	$\pi/2$	L1	θ_1
1_2T	L2	0	0	θ_2
2_3T	L3	0	0	θ_3

Tableau 4.1 tableau des paramètres de DH.

où

a_i est la distance entre Z_{i-1} et Z_i suivant l'axe X_i

α_i est l'angle formé par Z_{i-1} et Z_i suivant l'axe X_i

d_i est la distance entre X_{i-1} et X_i suivant Z_{i-1}

θ_i est l'angle formé par X_{i-1} et X_i suivant Z_{i-1} .

Pour calculer le MGD de notre robot en question, nous calculons la matrice de transformation (1). Il faudrait d'abord calculer les matrices de transformation homogène associée à chaque articulation.

$${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\theta_i & \cos\theta_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Les matrices de transformations homogènes sont données comme suit :

$${}^0T_1 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots\dots\dots (2)$$

$${}^1T_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_2 \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & L_2 \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots\dots\dots (3)$$

$${}^2T_3 = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & L_3 \cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & L_3 \sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots\dots\dots (4)$$

D'après la matrice de transformation homogène associée à chaque articulation nous avons déduit la matrice de transformation homogène associée au robot qui est par la formule (5) :

$${}^0T_3 = {}^0T_1 * {}^1T_2 * {}^2T_3 \dots\dots\dots(5)$$

A l'aide de MATLAB on calcule la formule (5) : (Figure 4.43).

```

clear all
clc
% |-----+-----+-----+-----+-----+
% |           | a |alpha | d | theta |
% |-----+-----+-----+-----+-----+
% | T01 | 0 | 90 | 11 |theta1 |
% | T12 | 12 | 0 | 0 |theta2 |
% | T03 | 11 | 0 | 0 |theta3 |
% |-----+-----+-----+-----+
theta1=sym('theta1');
theta2=sym('theta2');
theta3=sym('theta3');
l1=sym('l1');
l2=sym('l2');
l3=sym('l3');
T0_1=[cos(theta1) 0 sin(theta1) 0;sin(theta1) 0 -cos(theta1) 0;0 1 0 l1;0 0 0 1]
T1_2=[cos(theta2) -sin(theta2) 0 l2*cos(theta2);sin(theta2) cos(theta2) 0 l2*sin(theta2);0 0 1 0 ;0 0 0 1]
T2_3=[cos(theta3) -sin(theta3) 0 l3*cos(theta3);sin(theta3) cos(theta3) 0 l3*sin(theta3);0 0 1 0 ;0 0 0 1]
%MGD
T0_3=simplify(T0_1*T1_2*T2_3)
Px=T0_3(1,4)
Py=T0_3(2,4)
Pz=T0_3(3,4)

```

Figure 4.43 Programme de calcul sous Matlab.

Après la simulation du programme nous obtiendrons les résultats des calculs suivants:

$${}^0_3T = {}^0_1T * {}^1_2T * {}^2_3T \rightarrow$$

$$\begin{bmatrix} \cos(\theta_2 + \theta_3) * \cos\theta_1 & -\sin(\theta_2 + \theta_3) * \cos\theta_1 & \sin\theta_1 & \cos\theta_1 * (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2) \\ \cos(\theta_2 + \theta_3) * \sin\theta_1 & -\sin(\theta_2 + \theta_3) * \sin\theta_1 & -\cos\theta_1 & \sin\theta_1 * (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2) \\ \sin(\theta_2 + \theta_3) & \cos(\theta_2 + \theta_3) & 0 & L_1 + L_3 * \sin(\theta_2 + \theta_3) + L_2 * \sin\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Alors :

$$P_x = \cos\theta_1 * (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2)$$

$$P_y = \sin\theta_1 * (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2)$$

$$P_z = L_1 + L_3 * \sin(\theta_2 + \theta_3) + L_2 * \sin\theta_2$$

Donc :

$$P = \begin{bmatrix} \cos\theta_1 * (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2) \\ \sin\theta_1 * (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2) \\ L_1 + L_3 * \sin(\theta_2 + \theta_3) + L_2 * \sin\theta_2 \end{bmatrix}$$

4. II. 3 Modèle géométrique inverse

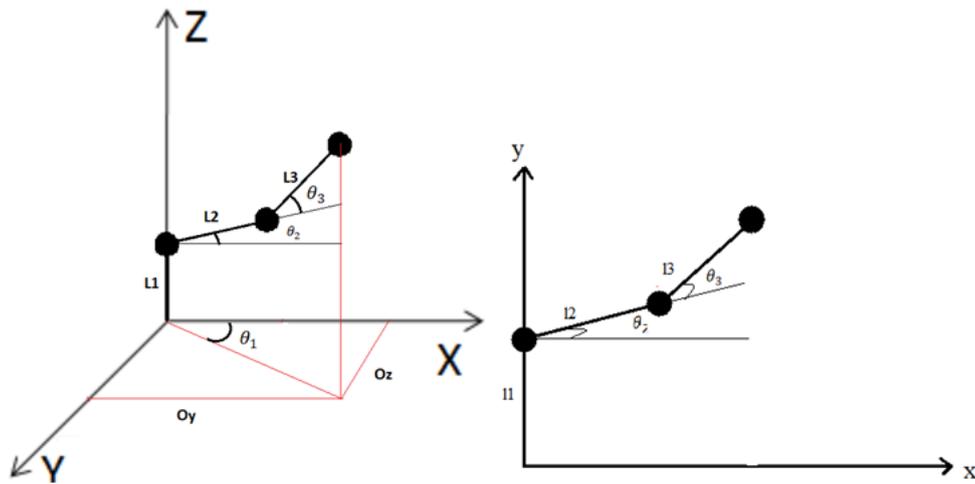


Figure 4.44 Projection du bras manipulateur sur un repère x, y, z. et x,y

- Calcul de θ_1

$$\theta_1 = \arctan2(Oy, Ox)$$

- Calcul de θ_3

$$\begin{cases} Ax = l_1 + l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \\ By = l_2 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3) \end{cases}$$

$$\begin{cases} (Ax - l_1) = l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \\ By = l_2 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3) \end{cases}$$

$$\begin{cases} (Ax - l_1)^2 = l_2^2 \cos^2\theta_2 + l_3^2 \cos^2(\theta_2 + \theta_3) + 2 l_2 l_3 \cos\theta_2 \cos(\theta_2 + \theta_3) \\ By^2 = l_2^2 \sin^2\theta_2 + l_3^2 \sin^2(\theta_2 + \theta_3) + 2 l_2 l_3 \sin\theta_2 \sin(\theta_2 + \theta_3) \end{cases}$$

$$\begin{cases} (Ax - l_1)^2 = l_2^2 \cos^2\theta_2 + l_3^2 \cos^2(\theta_2 + \theta_3) + 2 l_2 l_3 \cos\theta_2 \cos\theta_2 \cos\theta_3 - \sin\theta_2 \sin\theta_3 \\ By^2 = l_2^2 \sin^2\theta_2 + l_3^2 \sin^2(\theta_2 + \theta_3) + 2 l_2 l_3 \sin\theta_2 \sin\theta_2 \cos\theta_3 + \sin\theta_3 \cos\theta_2 \end{cases}$$

$$(Ax - l_1)^2 + By^2 = 2l_2^2 + 2l_3^2 + 4 l_2 l_3 + 2\cos\theta_3$$

$$(Ax - l_1)^2 + By^2 = 2l_2^2 + l_2^2 + l_3^2 + 2 l_2 l_3 \cos\theta_3$$

$$\cos\theta_3 = \frac{(Ax - l_1)^2 + By^2 - l_2^2 - l_3^2}{2 l_2 l_3}$$

$$\theta_3 = \cos^{-1}\left(\frac{(Ax - l_1)^2 + By^2 - l_2^2 - l_3^2}{2 l_2 l_3}\right)$$

- Calcul de θ_2

$$\begin{cases} (Ax - l_1) = l_2 \cos\theta_2 + l_3 \cos(\theta_2 + \theta_3) \\ By = l_2 \sin\theta_2 + l_3 \sin(\theta_2 + \theta_3) \end{cases}$$

$$\begin{cases} (Ax - l_1) = l_2 \cos\theta_2 + l_3 \cos\theta_2 \cos\theta_3 - l_3 \sin\theta_2 \sin\theta_3 \\ By = l_2 \sin\theta_2 + l_3 \sin\theta_2 \cos\theta_3 + l_3 \sin\theta_3 \cos\theta_2 \end{cases}$$

$$(Ax - l_1) = \underbrace{\cos\theta_2(l_2 + l_3 \cos\theta_3)}_{K1} - \underbrace{l_3 \sin\theta_3}_{K2} \sin\theta_2$$

$$By = \underbrace{\sin\theta_2(l_2 + l_3 \cos\theta_3)}_{K1} - \underbrace{l_3 \sin\theta_3}_{K2} \cos\theta_2$$

$$\begin{cases} (Ax - l_1) = \cos\theta_2 K1 - K2 \sin\theta_2 \\ By = \sin\theta_2 K1 - K2 \cos\theta_2 \end{cases} \Rightarrow \begin{cases} \sin\theta_2 = \frac{K1 By - K2(Ax - l_1)}{K1^2 + K2^2} \\ \cos\theta_2 = \frac{K2 By - K1(Ax - l_1)}{K1^2 + K2^2} \end{cases}$$

$$\theta_2 = \text{Arct2}(K1 By - K2(Ax - l_1), K2 By - K1(Ax - l_1))$$

Partie III Sélection des outils et composants de commande

4. III.1 Servomoteurs

Pour le bon choix du servomoteur, en général il faut calculer le couple fourni, voir Figure (45).

- **Servomoteur de la base**

$$C_B = m_B * l_B = [(30 * 10^{-3}) + 4(60 * 10^{-3}) + 4(10 * 10^{-3}) + (30 * 10^{-3}) + 10(30 * 10^{-3}) + 20(13 * 10^{-3}) + 20((16 + 60) * 10^{-3}) + 10(60 * 10^{-3})]$$

$$C_B = 3,07 \text{ kg.m}$$

- **Servomoteur de l'épaule**

$$C_P = m_P * l_P = [4(10 * 10^{-3}) + (30 * 10^{-3}) + 10(30 * 10^{-3}) + 20(13 * 10^{-3}) + 20((16 + 60) * 10^{-3}) + 10(60 * 10^{-3})]$$

$$C_P = 2,75 \text{ kg.cm}$$

- **Servomoteur du coude**

$$C_C = m_C * l_C = [(45 * 10^{-3}) + 4(13 * 10^{-3}) + 20((16 + 60) * 10^{-3})]$$

$$C_C = 1.62 \text{ kg.cm}$$

- **Servomoteur de la pince (la main)**

$$C_m = m_m * l_m = ((16 + 60) * 10^{-3})$$

$$C_m = 1.52 \text{ kg.cm}$$

Après les calculs des différents couples nécessaires pour le bon fonctionnement du bras manipulateur, nous trouvons que le servomoteur doit avoir un couple pas moins de 4 kg.cm. De ce fait, le servomoteur MG996 (Figure 4.46) et le TowerPro SG90 (Figure 4.47) correspondent aux exigences obtenus par nos calculs et répondent donc aux besoins de notre bras manipulateur.

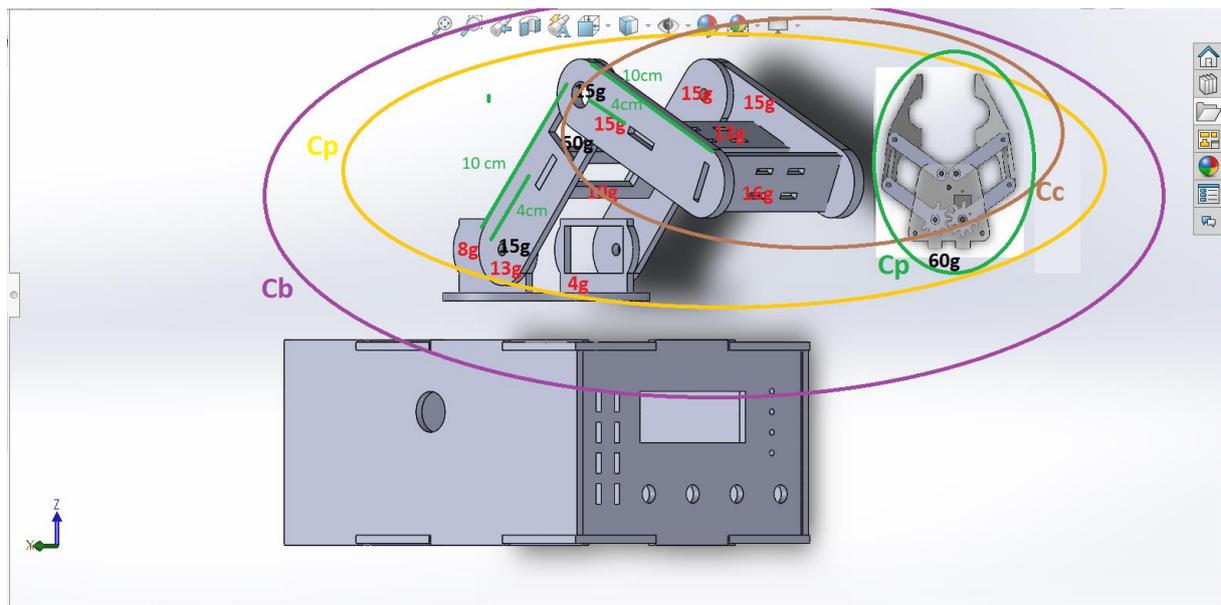
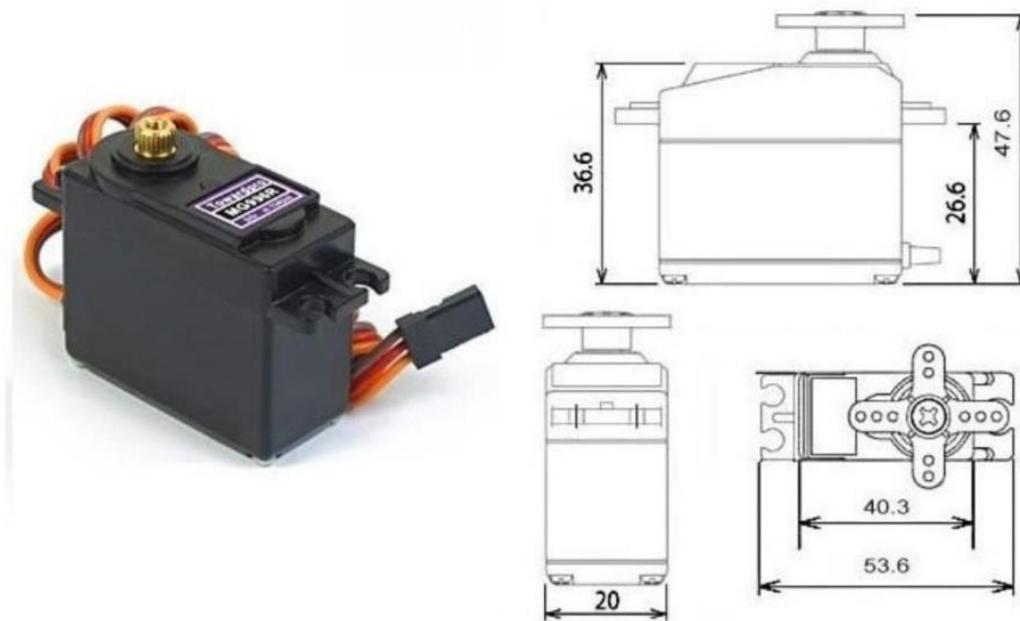
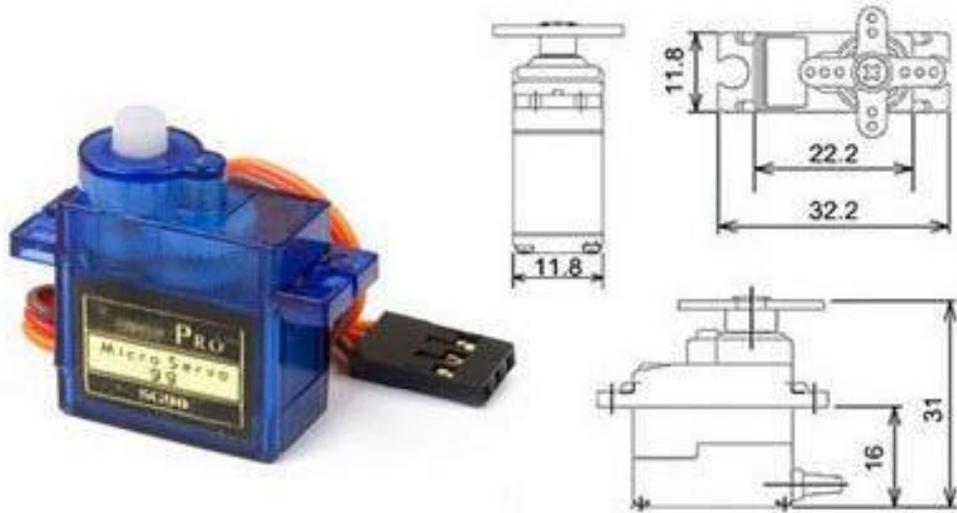


Figure 4.45 Couples nécessaires pour chaque servomoteur.



Poids: 55 g
 Couple de décrochage: 9,4 kgf · cm (4,8 V), 11 kgf · cm (6 V)
 Vitesse de fonctionnement: 0,17 s / 60 ° (4,8 V), 0,14 s / 60 ° (6 V)
 Tension de fonctionnement: 4,8 V à 7,2 V
 Courant de fonctionnement 500 mA -900 mA(6 V)

Figure 4.46 Servomoteur MG996. [1]



Poids : 9 g
Vitesse : 0.12 sec/60° sous 4.8v
Couple : 1.2kg/cm sous 4.8v
Tension : 4.8v à 6v

Figure 4.47 Servomoteur SG90.

4. III.2 Exemple de commande d'un servomoteur

Il existe plusieurs méthodes pour la commande des servomoteurs. Nous choisissons la commande du servomoteur par un potentiomètre, pour déterminer la position du palonnier de ce dernier, voir le schéma d'assemblage (Figure 4.48), et le programme de commande correspondant.

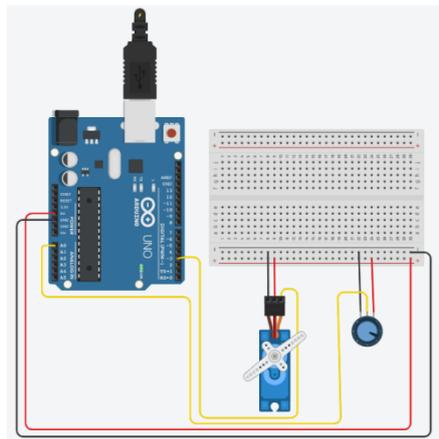


Figure 4.48 Circuit de commande du servomoteur.

Programme de commande du servomoteur :

```
#include <Servo.h> //accesses the Arduino Servo Library

Servo SERVO1;
int vall;
void setup()
{
  Serial.begin(9600);
  SERVO1.attach(3);
  SERVO1.write(90);
  delay(5);
}
void loop()
{
  vall = analogRead(0);
  vall = map(vall, 0, 1023, 0, 180);
  SERVO1.write(vall);
  delay(5);
}
```

4. III.3 Afficheur à cristaux liquides (LCD)

Dans notre application on a besoin d'afficher quelques résultats comme la position de l'effecteur, les angles des articulations, etc. Nous avons considéré que l'afficheur LCD 16*2 est le plus pratique et le adapté. En fait, LCD signifie "Liquid Crystal Display" et se traduit, en français, par "Écran à Cristaux Liquides" (Figure 4.49). Ce type d'écran est utilisé sur plusieurs appareils électroniques disposant d'afficheur : les calculatrices, les tableaux de bords des voitures, les montres, etc. Leur utilisation est courante en raison de leur faible coût.

L'afficheur LCD possède 2 types de fonctionnement suivant le type des données :

Permet d'initialiser le positionnement du curseur et l'effacement de l'écran, etc.

Permet l'affichage des données [2]

L'afficheur LCD contient 2 types de mémoires ;

- DDRAM mémorise les caractères affichés à l'écran.
- CGRAM contient la table des caractères affichables. [2]

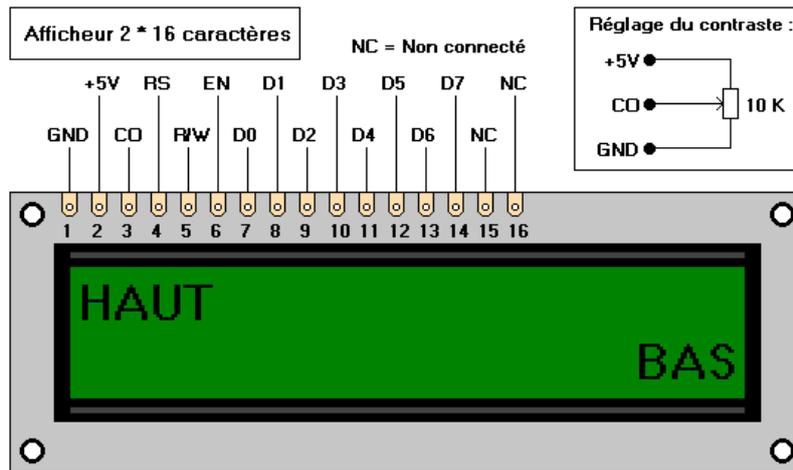


Figure. 4.49 LCD 16*2 et son brochage

Nom	Description
1	Vss Masse
2	Vdd Alimentation 5v
3	CO Variables de 0 à 5v permet de modifier le contraste de l'afficheur
4	RS Indique une commande ou une donnée à afficher (0: Commande / 1: Donnée)
5	R/W Indique une écriture ou une lecture (0: Écriture / 1: Lecture)
6	E Indique une validation (Le niveau Haut doit être maintenu 500µs)
7	D0
8	D1
9	D2
10	D3 Bus de données bidirectionnel.
11	D4
12	D5
13	D6
14	D7
15	A Anode rétro éclairage (+5V)
16	K Cathode rétro éclairage (masse)

Tableau 4.2 Description des broches du LCD 16*2.

A ce stade, on a besoin d'un logiciel ou d'une plateforme de simulation pour tester le bon fonctionnement de nos circuits avant le brochage final. Pour cela, nous avons fait l'usage d'un logiciel entièrement gratuit appelé Tinkercad, il nous permet de concevoir un prototype de notre carte Arduino avec tous les composants commandés avec un rendu final très ressemblants à la réalité. Ainsi, nous avons la main pour la création de nos circuits et de simuler leurs fonctionnements afin de réaliser les premiers essais et relever les éventuels dérèglements, ce qui permet d'effectuer les corrections nécessaires avant le branchement définitif.

Nous avons donc créé le circuit de branchement de l'afficheur LCD (voir Figure 50) et nous utilisons un programme de commande qui affiche mon nom et prénom dan l'écran LCD.

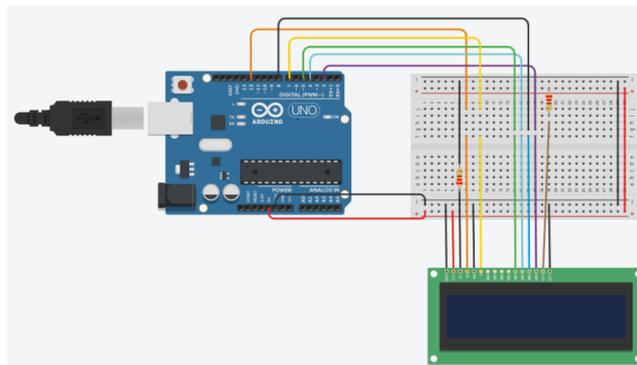


Figure. 4.50 Branchement du LCD avec Arduino Uno.

Programme de commande du LCD

```
// include the library code:
#include <LiquidCrystal.h>

//LiquidCrystal lcd(RS,enable, 5, 4, 3, 2);
LiquidCrystal lcd(12, 7, 5, 4, 8, 2);

void setup() {
  // nombre de (colonnes,lignes)
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("regainia djamel");
}

void loop() {
  //placement de curseur (colonne,ligne)
  lcd.setCursor(0, 1);
}
```

En lançant la simulation, nous obtenons l'affichage de mon nom et prénom sur l'écran du LCD comme le montre la Figure (4.51) en simulation, et sur la Figure (4.52) en vrai.

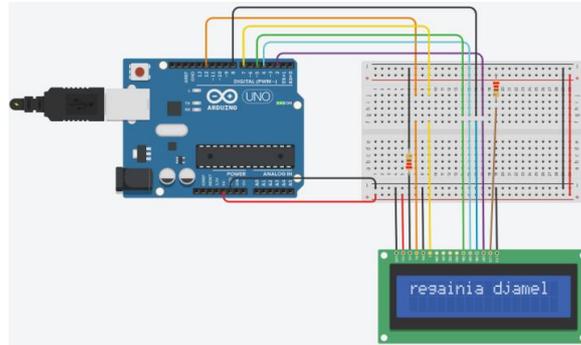


Figure 4.51 Circuit de l'afficheur LCD.



Figure 4.52 Test d'affichage sur LCD.

4. III.3 Carte Arduino UNO

Après avoir établi la liste des différents composants nécessaires à la commande de notre bras manipulateur (LCD, Servomoteur....etc.), nous pouvons dresser le tableau suivant pour nous aider à faire le meilleur choix de la carte Arduino qui répond à toutes les exigences du robot que nous avons conçu. Après comparaison des résultats du tableau 1 et du tableau 4, nous pouvons conclure que l'Arduino Uno couvre largement les besoins de tous nos composants électroniques.

	Entrées/sorties Numérique	Entrées Analogique	Tension (Volts)	Spécificité
Servomoteur	3(PWM)	0	5V	Rien
LCD	6	-	5V	Rien
Potentiomètre	0	4	-	Rien
Intercepteur	-	-	-	Rien
Led	-	-	-	Rien
Résultat	-	-	-	Rien

Tableau 2.3 Exigences des entrées/sorties

Partie IV Réalisation du circuit électronique du bras manipulateur

4. IV. 1 Circuit électronique complet de la commande du bras manipulateur avec

Arduino

Pour la réalisation de circuit électronique à base de carte Arduino dédié à la commande de notre bras manipulateur nous avons donc utilisé trois servomoteurs de modèle MG996 et un servomoteur de modèle SG90. Ces servomoteurs sont connectés aux entrées numériques de l'Arduino UNO de type PWM, à savoir les PWM 3, 5, 9, 10. Nous avons aussi connecté quatre potentiomètres aux entrées analogiques (A1, A2, A3, A4), par lesquelles passera le signal de la commande manuel des servomoteurs en convertissant sa valeur analogique en valeur numérique par un convertisseur analogique/numérique intégré au microcontrôleur d'Arduino. Nous avons également fourni au circuit un écran LCD que nous connectons comme nous l'avons montré dans le paragraphe précédent pour afficher les différents changements d'angles des servomoteurs. Puis nous avons muni le tout d'un bouton poussoir pour lancer le système et de deux lampes Led. La première est blanche et elle s'allume lors de l'affichage des angles et une lampe Led bleu qui s'allume pour en appuyant sur le bouton poussoir, ce qui activera le lancement de l'exécution de l'algorithme du modèle géométrique direct pour le calcul des angles. Nous avons également alimenté le circuit en 5v à partir de la broche Vdd 5V et GND de l'Arduino et ajouté une tension externe de 6 volts afin de renforcer l'alimentation du circuit pour protéger la carte Arduino UNO.

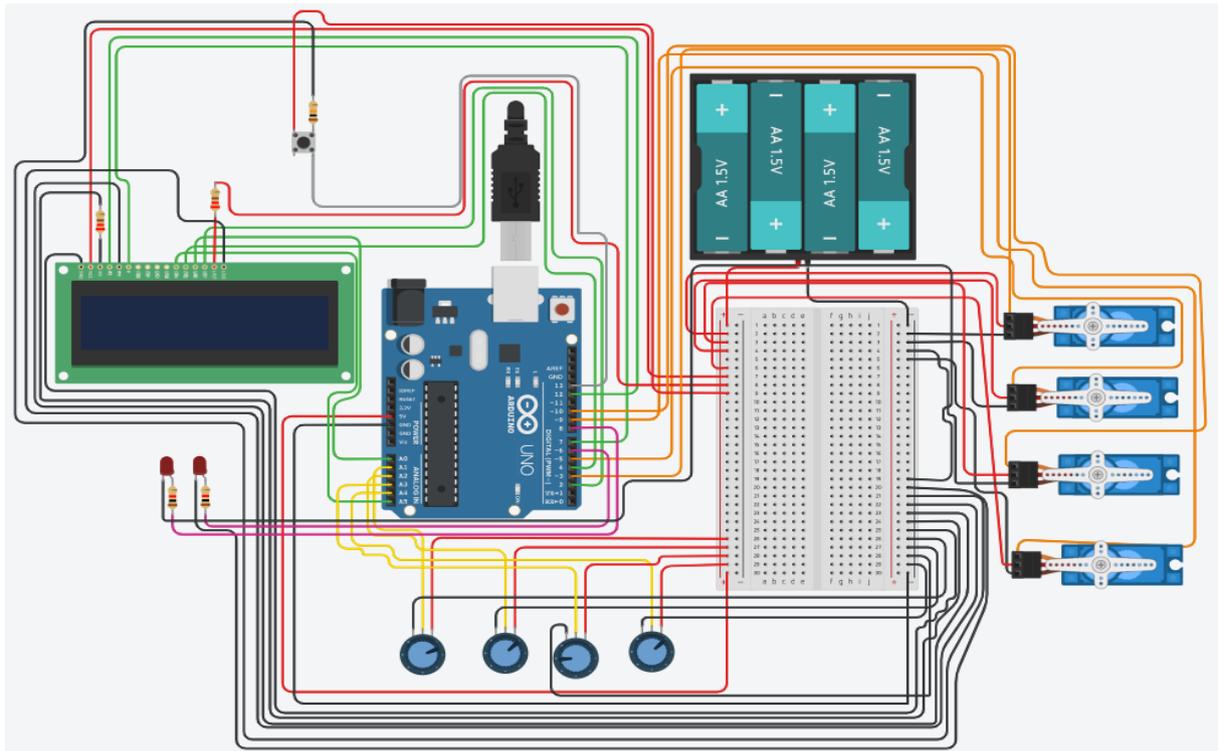


Figure 4.53 Schéma électrique de commande du bras manipulateur avec Tinkercad



Figure 4.54 Câblage électrique du le robot manipulateur

4. IV.2 Assemblage des composants dans le socle du robot manipulateur

Après avoir fait tourner le circuit du montage de la Figure (4.54) en simulation, via Tinkercad nous nous assurons que tous les composants du circuit (les servomoteurs, les potentiomètres, les leds, etc.) sont fonctionnels, donc maintenant nous allons passer au câblage réelle de tous les composants avec la carte Arduino qui est placée dans le socle du robot manipulateur, comme indiquée dans l'image de la Figure (4.55).



Figure 4.55 Robot manipulateur avec ses accessoires.

4.2 Conclusion

La fabrication du bras manipulateur n'est pas une tâche facile, mais après plusieurs tentatives et de longues heures de travail, nous sommes parvenus à trouver des solutions adaptées nous permettant d'aboutir à la conception qui répond à nos objectifs. La conception 3D du bras manipulateur avec Solidworks nous a permis d'obtenir un prototype du robot à concevoir. Une machine CNC à trois axes a été utilisée pour l'impression des pièces du robot, fabriqué en alucobande pour les très bonnes propriétés que présente cette matière. Pour la commande de ce bras, il a fallu le calcul du modèle géométrique direct et inverse, ensuite est venu le moment du choix du meilleur outil de commande qui répond à nos attentes et qui est à notre disposition c'est-à-dire les servomoteurs MG996 et SG90, en plus du choix de la carte et pleins d'autres composants qui serviront à construction le circuit de commande autour de la carte Arduino UNO.

Conclusion générale

Ce travail traite la commande des servomoteurs par carte Arduino, qui est le thème de notre projet de fin d'étude. Pour illustrer au mieux cet objectif, nous avons choisi de réaliser un bras manipulateur afin d'aborder par le biais de cette réalisation tous les problèmes aussi bien théoriques que pratiques qu'elle soulève.

Dans le premier chapitre, nous montrons que la robotique est parfaitement adaptée à l'illustration de la commande des servomoteurs par une carte à base de microcontrôleur. Dans le deuxième chapitre, nous avons orienté l'étude sur les composants dressé un tableau comparatif des différentes cartes Arduino pour faciliter le choix de la carte appropriée qui répond au plus près aux exigences de chaque application afin d'éviter les pertes de temps et d'argent. Quant au troisième chapitre, une solution parmi tant d'autres pour la modélisation des robots a été adoptée pour obtenir les angles de rotation des servomoteurs qui vont commander le bras manipulateur.

Toutes les solutions présentées dans les trois premiers chapitres traitent purement le côté théorique de la conception d'un bras manipulateur, vient alors le dernier chapitre pour illustrer la fabrication effective d'un robot manipulateur à travers toutes les étapes de conception en commençant par la modélisation 3D du bras manipulateur à trois degrés de liberté puis après plusieurs tentatives de conception échouées nous sommes parvenus à trouver une machine CNC à 3 axes pour la fabrication des pièces du robot. Enfin, après l'assemblage final de toutes les pièces et de la connexion de la carte électronique réalisée pour la commande du bras manipulateur obtenu, nous avons choisis les servomoteurs qui serviront à commander ce bras suivant la quantité d'effort à fournir relatif à la taille de notre robot. Nous avons également choisis la commande des servomoteurs par des potentiomètres afin de fixer pour chaque mouvement la position du palonnier de ces derniers.

On peut dire que nous avons réussi le challenge à la fois de la conception et de la commande d'un bras manipulateur avec une carte Arduino même si c'est manuellement à l'aide de potentiomètres. Nous avons également réussi d'afficher à l'aide d'un écran "LCD" tous les changements angulaires des servomoteurs et des coordonnées de l'organe terminal x, y, z calculées avec l'algorithme de MGD. Malheureusement, nous ne sommes pas arrivés à le commander automatiquement car on manquait cruellement de temps et de moyen.

Perspectives

Ce mémoire a présenté la fabrication d'un robot manipulateur et la réalisation d'un circuit de commande à base pour le contrôle de ses servomoteurs. Cependant, dans de prochaines applications plusieurs améliorations peuvent être apportées pour obtenir de meilleurs résultats. Il faut par exemple prévoir d'autres équipements pour la fabrication tels que la machine CNC à 5 axes pour obtenir une structure mécanique de meilleure qualité. Aussi, il est à prévoir de changer de type des servomoteurs pour avoir des couples élevés et donc une plus grande puissance. On peut même leurs associer un ensemble de capteurs pour une meilleure précision dans les mouvements. Aussi, on peut rajouter la commande automatique du robot en créant une interface homme-machine HMI à l'aide de logiciel tel que "Visual Studio" programmé en langage C++ qui contient l'emplacement d'écriture de l'ordre de commande automatique (les coordonnées x,y,z, l'activation et désactivation de la commande manuelle, etc). Il est également possible de créer une application Android pour le contrôle automatique et manuel à du bras manipulateur à l'aide du module HC-05 et Arduino.

RÉFÉRENCES

CHAPITRE 1

[1] M. Amira, "Conception et réalisation d'un bras manipulateur commandé par l'arduino 2560," Master's thesis, Université M'hamed Bougara de Boumerdes, Faculté des Sciences de l'ingénieur, Département de Génie Mécanique, Modélisation et Simulation en Mécanique, 2016/2017.

[2] F. G. . Jean-Pierre, Frédéric Genevey & Jean-Pierre Dulex, février, Arduino à l'école : Edition ressource éducatif libre, 2018.

[3] S. L. et Hippolyte Wesslinger, "Premiers pas en informatique embarquée," Le blog d'Eskimon, Jun. 2019.

[4] B. B. et Eddie Smigiel, "Découvrez la constitution des servomoteurs," Stras- bourg, 2019.

[5] B. M. et MARTINELLI Pascal, "Transmission de puissance engrenages," Yumpu, 2014.

CHAPITRE 2

[1] A. Bas, "Modified servo," Research Andbas, Jan. 2012.

[2] H. Marzelin, "Les servomoteurs," Redohm, Sep. 2015.

CHAPITRE 3

[1] D. Samir, Modrélisation et Commande De Robots Industriels, D. Samir, Ed. Université Badji Mokhtar Sidi Amar Annaba, 2017/2018.

[2] A. J. Castro-Tirado, "Robotic autonomous observatories : A historical perspective," Advances in Astronomy, vol. 2010, pp. 1–8, 2010.

[3] J. D. B. Latita, "De l'usage de nao (robot humanoide) dans l'apprentissage de l'informatique," Researchgate, 2011.

[4] M. B. M. S. Max Schwarz, David Droeschel, "Solving disaster-response tasks through mobile manipulation robot momaro," Field Rbotics, 2017.

[5] R. Mansouri, "Conception et réalisation d'un bras manipulateur à trois degré de liberté commandé par la carte arduino uno," UNIVERSIT'E MOU- LOUD MAMMERI TIZI- OUZOU, Jul. 2015.

[6] M. O. Bachir, Développement d'un Système Intelligent Distant Basé sur les Concepts d'Intelligence artificielle : Application aux Bras Manipulateurs, Apr. 2015.

[7] I. B. Abdallah Ghou, "Modélisation des bras manipulateurs," Ph.D. dissertation, Université Tahri Mohammed Béchar, Département de Mécanique, 2016.

CHAPITRE 4

[1] S. L. et Hippolyte Wesslinger, “Premiers pas en informatique embarquée :Arduino,” Le blog d’Eskimon, Jun. 2019.

[2] T. Science, “Fraisage-d’efinition et explications,” Techno Science, 2020.

