

Ordre...../F.S.S.A/UAMOB/2019

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE AKLI MOAND OULHADJE-BOUIRA



Faculté des Sciences et des Sciences Appliquées
Département : Génie Electrique

Mémoire de fin d'étude

Présenté par :

BOUZIR Rami

KACIMI Athman

En vue de l'obtention du diplôme de **Master 02** en :

Filière : Electronique

Option : Electronique des Systèmes Embarqués

Thème :

Réalisation d'un bras manipulateur à base d'Arduino

Devant le jury composé de :

Y. TOUAFEK	MAA	UAMOB	Président
K. SAUDI	MCA	UAMOB	Encadreur
M. BENZIANE	MAA	UAMOB	Co-Encadreur
M. DJEBIRI	MAA	UAMOB	Examineur

Année Universitaire 2018/2019

Remerciements

Nous rendons nos profondes gratitudees à dieu qui nous a aidés à réaliser ce modeste travail.

Nous exprimons nos profondes gratitudees à nos parents pour leurs encouragements, leurs soutiens et pour les sacrifices qu'ils ont enduré.

Nous remercions nos encadreurs SAOUDI Kamel et BENZIANE Mourad pour les efforts qu'ils ont déployés, pour nous aider, conseiller, encourager et corriger.

Nous remercions aussi tout le corps enseignant et administratif qui ont contribué à notre formation universitaire.

Dédicaces

Je dédie ce modeste travail

À l'esprit de mon père, À très chère ma mère, mes frères et ma sœur, que ce travail soit témoin de ma reconnaissance éternelle, de mon respect, de ma gratitude et surtout de mon amour pour eux

À toute ma famille qui ont été toujours là pour moi et qui m'ont tant soutenu

À mon binôme KACIMI ATHMAN avec qui j'ai tant appris et avec qui j'ai passé une inoubliable année

À celle qui m'est très chère

À mes amis qui m'ont aidé tout le long de ce projet.

Rami

Je dédie ce modeste travail

À mes très chers parents, mes sœurs, que ce travail soit témoin de ma reconnaissance éternelle, de mon respect, de ma gratitude et surtout de mon amour pour eux

A toute ma famille qui ont été toujours là pour moi et qui m'ont tant soutenu

A mon binôme BOUZIR RAMI avec qui j'ai tant appris et avec qui j'ai passé une inoubliable année

A mes amis qui m'ont aidé tout le long de ce projet

Athman

Sommaire

REMERCEIMENT.....	i
Dédicace	ii
Sommaire.....	iv
Liste des figures.....	vii
Liste des tableaux.....	viii
Introduction Générale.....	1

Chapitre I : généralité sur les robots industriels

I.1. Introduction.....	3
I.2. Définitions	3
I.2.1. Robotique	3
I.2.2. Robot	3
I.3. Historique	3
I.4. Les types des robots	5
I.4.1. Robots mobiles	5
I.4.2. Robots manipulateurs	6
I.5. Les éléments constitutifs d'un robot	6
I.5.1. Unité informationnelle	6
I.5.2. Unité opérationnelle	6
I.6. Architecture des robots manipulateurs.....	7
I.6.1. La base	7
I.6.2. Le porteur	7
I.6.3. L'actionneur	8
I.6.4. L'organe terminal(Effecteur)	9
I.7. Classification des robots	9
I.7.1. Classification fonctionnelle.....	9
I.7.2. Classification géométrique	11
I.8. Critère de choix d'un robot	14
I.9. Domaines d'application des robots	15
I.10. Conclusion	16

Chapitre II : conception matérielle et logiciel

II.1. Introduction	17
II.2. Conception matérielle.....	17
II.2.1. Carte de commande Arduino	17
II.2.1.1. Présentation.....	17
II.2.1.2. Principe de fonctionnement.....	17
II.2.1.3. Les familles de la carte Arduino.....	18
II.2.1.4. Les type de la carte Arduino.....	18
II.2.1.5. Applications.....	19
II.2.1.6. Bonnes raisons de choisir Arduino.....	20
II.2.2. La carte Arduino UNO.....	21
II.2.2.1. Présentation.....	21
II.2.2.1. Constitution de la carte Arduino UNO.....	21
II.2.3. Bluetooth.....	24
II.2.3.1. Présentation.....	24
II.2.3.2. Le module Bluetooth Arduino	25
II.2.4. Servomoteur	26
II.2.4.1. Présentation.....	26
II.2.4.2. Fonctionnement des servomoteurs.....	27
II.2.4.3. Les avantages et les inconvénients.....	29
II.3. Conception logiciel	29
II.3.1. IDE Arduino	29
II.3.1.1. Description de l'IDE	29
II.3.1.2. Structure d'un programme Arduino.....	30
II.3.2. MIT APP INVENTOR.....	31
II.3.2.1. Le système Android.....	31
II.3.2.2. Présentation de MIT APP INVENTOR	31
II.3.2.3. La conception d'APP INVENTOR.....	32
II.4. Conclusion.....	33

Chapitre III : Réalisation de bras manipulateur

III.1. Introduction	34
III.2. Elaboration de prototype.....	34
III.3. Etapes de réalisation et d'assemblage.....	34

III.3.1. Fabrication de la structure mécanique de bras.....	34
III.3.2. Montage électrique.....	37
III.3.2.1. Configuration de module Bluetooth	37
III.3.2.2. Câblage totale.....	38
III.3.3. Développement de l'application Android.....	39
III.3.4. Programmation de la carte Arduino	45
III.4. Conclusion	46
Conclusion générale.....	47
Références bibliographiques.....	49
Résumé	

Liste des figures

Chapitre I

Figure I.1. Le PUMA UNIMATION.....	4
Figure I.2. Robot mobile.....	4
Figure I.3. Robot manipulateur.....	5
Figure I.4. Architecture d'un Robot manipulateur.....	6
Figure I.5. Représentation d'une articulation rotoïde.....	7
Figure I.6. Représentation d'une articulation prismatique.....	7
Figure I.7. Manipulation de bobines de papier ou de film plastique.....	8
Figure I.8. Manipulateur automatique (kuka-robot de soudage à l'arc)	9
Figure I.9. Robot mobile programmable.....	9
Figure I.10. Robot avec système de vision intelligent.....	10
Figure I.11. (a) Structure cartésienne, (b) Robot cartésien(SEPRO)	10
Figure I.12. (a) Structure cylindrique, (b) Robot cylindrique (Plate Crane).....	11
Figure I.13. (a) Structure sphérique, (b) Robot sphérique (FANUC L1000+A).....	11
Figure I.14. (a) Structure SCARA, (b) Robot type SCARA (ADTECH)	12
Figure I.15. (a) Structure anthropomorphique, (b) Robot anthropomorphique (Stäubli)	12
Figure I.16. (a) Structure parallèles DELTA, (b) Robot type DELTA.....	13

Chapitre II

Figure II.1. Composition de carte Arduino UNO.....	20
Figure II.2. Microcontrôleur ATmega328P.....	21
Figure II.3. Alimentation de la carte Arduino.....	22
Figure II.4. Logo de Bluetooth.....	23
Figure II.5. Modules Bluetooth pour l'Arduino.....	24
Figure II.6. Composition d'un servomoteur.....	25
Figure II.7. Fonctionnement d'un servomoteur.....	26
Figure II.8. Illustration du signal de contrôle.....	27
Figure II.9. Interface de logiciel IDE Arduino.....	29
Figure II.10. Logo d'APP INVENTOR.....	31
Figure II.11. Création d'une application Android par APP INVENTOR.....	31

Chapitre III

Figure III.1. Schéma global de bras manipulateur.....	33
Figure III.2. Les mouvements de bras : (a) l'axe1 (90°) (b) l'axe2 (180°).....	34
Figure III.3. Les mouvements de bras : (a) la poignée (45°) (b) la base (180°).....	34
Figure III.4. Image réelle de la base.....	35
Figure III.5. Image réelle de l'axe 1 et l'axe 2.....	35
Figure III.6. Image réelle de la poignée.....	36
Figure III.7. L'assemblage du Smartphone avec le HC-06.....	37
Figure III.8. Montage électrique de bras manipulateur.....	37
Figure III.9. L'interface d'un nouveau projet.....	38
Figure III.10. Remplissage de « Screen 1 »	39
Figure III.11. L'interface de l'application sous App Inventor.....	39
Figure III.12. Block d'initialisation de l'application.....	40
Figure III.13. Blocks de la liste des périphériques disponibles.....	40
Figure III.14. Block de l'état du Bluetooth.....	40
Figure III.15. Block du bouton « déconnecter »	41
Figure III.16. Blocks de contrôle des servomoteurs.....	42
Figure III.17. Block du Bouton Automatique.....	42
Figure III.18. Blocks globale de l'application Android.....	43
Figure III.19. Création du fichier avec l'extension Android.....	44
Figure III.20. L'interface de l'application sous Smartphone.....	45
Figure III.21. Organigramme de fonctionnement global.....	46
Figure III.22. Organigramme de fonctionnement automatique.....	47

Liste des Tableaux

Tableau II.1.Synthèse des caractéristiques.....	20
Tableau II.2. Paramètres techniques des servomoteurs	27
Tableau III.1. Les paramètres de l'application.....	43

Introduction générale

L'homme a toujours chercher tout ce qui est nouveau et innovant pour faciliter sa vie. Notre temps actuel a connu un développement dans divers domaines de la science et de la technologie, en particulier l'informatique, les mathématiques, l'électronique et la mécanique, qui a permis de construire des machines et des robots hautement perfectionnés capables de remplacer l'humain dans certaines fonctions. [1]

Les robots existent dans différents domaines tels que : la santé, la domestique, l'agriculture, le domaine militaire, et l'industrie où ils sont généralement utilisés pour assurer des fonctions répétitives et précises. Dans ce domaine et comme il est connu actuellement dans les usines le « bras manipulateur » qui est un robot de type manipulateur, est implantés dans les procédés de fabrication modernes et est employé pour augmenter le volume de production et améliorer la qualité des produits dans les chaînes de montage, l'industrie automobile, etc. Il remplace les ouvriers dans les tâches pénibles, répétitives ou aventureuses (peinture, soudure...etc.)

Notre projet de fin d'études a pour objectif la conception et la réalisation d'un bras manipulateur à trois degrés de liberté commandé à distance via une application Android. La commande est assurée grâce à une carte Arduino munie de son module Bluetooth permettant de communiquer à distance avec le smartphone à des distances allant jusqu'à 10 mètres, notre principale motivation est l'intérêt qui s'accroît plus en plus de ces bras manipulateurs qui sont capables d'assurer des tâches pénibles, répétitives et dangereuses ou encore travailler dans les milieux hostiles, le choix des composants et du smartphone est justifié par la disponibilité et la fiabilité.

Ce mémoire est structuré comme suit :

- **Le premier chapitre**, présente des généralités sur les robots industriels et en particulier les robots manipulateurs, leur architecture et leur classification, ainsi que les domaines d'applications, sont brièvement discutés.
- **Le deuxième chapitre**, traite les éléments nécessaires à la fabrication du bras manipulateur en particulier la carte programmable assurant la commande et les servomoteurs ainsi que les différents logiciels nécessaires à la programmation de la carte Arduino et au développement de l'application Android.
- **Au troisième chapitre**, représentant le noyau du mémoire et dans lequel nous allons détailler les étapes de la réalisation de notre bras manipulateur.

- Finalement, nous terminerons ce mémoire par une **conclusion générale** qui résume les résultats de notre étude dans la partie théorique et la partie réalisation.

Chapitre I :

Généralités sur les robots industriels

1. Introduction

Depuis la naissance de l'industrie robotisée, les robots industriels ont été conçus afin de remplacer les humains dans des tâches répétitives, fatigantes et qui impliquent souvent un travail manuel dangereux. Actuellement, grâce à l'augmentation de la production et à la diminution des prix, les robots industriels sont utilisés dans une infinité d'applications.

Pour comprendre le développement des robots et le monde robotique qui est le but de ce chapitre, nous allons donner dans un premier lieu un bref aperçu de l'histoire des robots avec quelques définitions de base, dans un deuxième lieu une description des constituants technologiques d'un robot sera donnée ensuite une présentation de types et d'architectures de ces derniers et en fin les modèles les plus importants qui sont classés selon leurs fonctionnements et leurs structures. Ainsi que le critère de choix d'un robot et nous mentionnerons les différents domaines qui utilise aujourd'hui les robots pour répondre à leurs besoins.

2. Définitions

2.1 Robotique

La robotique est une science qui s'intéresse aux robots. En fait, il s'agit d'un ensemble de disciplines techniques (mécanique, électronique, automatique, informatique) articulées autour d'un objectif et d'un objet communs. Cet objectif est l'automatisation flexible de nombreux secteurs de l'activité humaine réputés jusqu'à très récemment comme ne pouvant se passer de la présence de l'homme, et l'objet est le robot, sorte de machine universelle, dont l'homme rêve depuis toujours pour le remplacer dans les tâches difficiles. [2]

2.2 Robot

Un robot est un système mécanique poly-articulé mû par des actionneurs et commandé par un calculateur qui est destiné à effectuer une grande variété de tâches. Il est également défini comme étant une machine capable d'effectuer des tâches et de manipuler des objets selon un programme de façon automatique. Ils sont généralement utilisés pour remplacer les humains dans des situations où ces derniers sont incapables d'effectuer le travail, des situations plus dangereuses, de haute précision ou répétitive. [1]

3. Historique

- Le mot robot a été introduit en 1921 par le dramaturge tchèque Karel Capek dans sa pièce satirique R. U. R (*Universal Robots de Rossum*).

- Un manipulateur de broyage de récipients en verre a également été ajouté aux téléopérateurs en 1949.
- Les premiers robots, développés par « George Devol » en 1954, ont remplacé le manipulateur maître du téléopérateur avec la possibilité de programmation avec CNC (un utile de contrôle de périphérique).
- Les droits de brevet ont été achetés par « Joseph Engelberger », étudiant à l'Université de Columbia, qui a par la suite fondé la société UNIMATION dans le Connecticut en 1956.
- UNIMATION a installé son premier robot en une usine de General Motors en 1961. L'innovation clé ici était « la programmabilité » de la machine.
- En 1962, Ernst développa un robot à détection de force qui lui permettait d'empiler des blocs, ce système était le premier qui impliquait un robot en interaction avec un environnement non structuré et a conduit à la création du projet MAC (*Man And Computer*) au MIT (Institut de technologie du Massachusetts).
- Un système de vision robot binaire qui a permis au robot de réagir à la présence des obstacles dans son environnement a été développé par McCarthy et ses collègues en 1963.
- Un autre robot novateur de l'époque était un robot marcheur développé par General Electric pour l'armée en 1969. Les robots qui répondent aux commandes vocales et des blocs empilés au hasard ont été développés à Stanford et dans d'autres lieux. Les robots font leur apparition au Japon par l'acquisition de Kawasaki d'une licence d'UNIMATION en 1968.
- En 1973, le premier langage de programmation de mouvement de robot, appelé WAVE, a été développé à Stanford pour permettre de commander un robot par des commandes de haut niveau.
- En 1974, le fabricant de machines à outil « Cincinnati Milacron, Inc. » a présenté son premier manipulateur commandé par ordinateur, appelé The Tomorrow Tool (T³), qui pourrait soulever une charge de 100 livres (45.35 kg) ainsi que de suivre les objets en mouvement sur une chaîne de montage.
- En 1978, UNIMATION présente un robot appelé PUMA (*Programmable Universal Machine for Assembly*), basé sur une étude de General Motors.

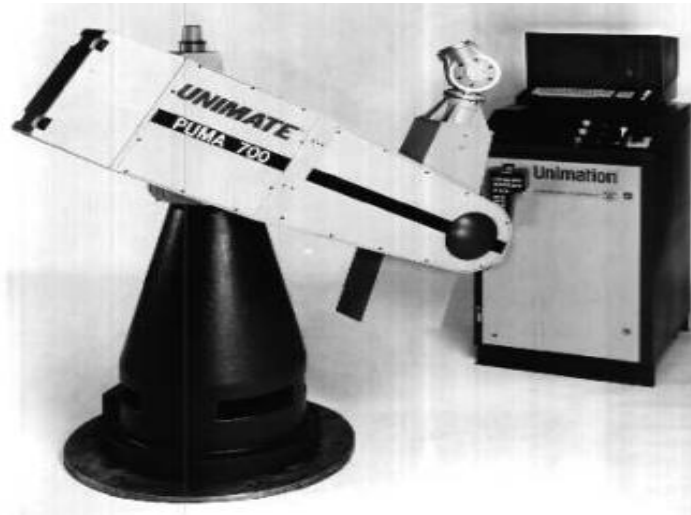


Figure I.1. Le PUMA UNIMATION. [3]

- En 1979, le SCARA (Robot articulé conforme sélectif pour montage) a été introduit au Japon puis en États-Unis.
- Un robot à entraînement direct a été développé à Carnegie Mellon par Asada en 1981. [3]

4. Les types des robots

Il existe deux grandes familles de robots : robots mobiles et robots manipulateurs.

4.1 Robots mobiles

Ce sont des robots qui peuvent se déplacer indépendamment dans leur environnement. Ils sont équipés ou non de manipulateurs suivant leur utilisation, (les robots explorateurs, les robots de services....).



Figure I.2. Robot mobile. [4]

4.2 Robots manipulateurs

Des robots ancrés physiquement à leur place de travail et généralement mis en place pour réaliser une tâche précise répétitive, tels que les robots industriels, médicaux...etc.



Figure I.3. Robot manipulateur. [5]

5. Les éléments constitutifs d'un robot

On peut distinguer deux ensembles interactifs dans un robot en fonctionnement : l'unité informationnelle et l'unité opérationnelle.

5.1 Unité informationnelle

Cette partie reçoit les instructions décrivant la tâche à accomplir, les mesures relatives à l'état interne de la structure mécanique qui constitue le bras manipulateur et les observations concernant son environnement. Elle élabore en conséquence les commandes de ses différentes articulations en vue de l'exécution de ses tâches. Les systèmes actuels fonctionnent en interaction permanente selon le cycle information-décision-action. [6]

5.2 Unité opérationnelle

Cette unité représente les actions commandées en empruntant la puissance nécessaire à la source d'énergie. En effet, ça constitue le robot physique, qui intègre la structure mécanique (segments, articulations, architecture,...), les modules d'énergie (amplificateurs, variateurs, servovalves...), les convertisseurs d'énergie (moteurs, vérins...), les chaînes cinématiques de transmission mécanique (réducteurs, vis à billes, courroies crantées....), les capteurs proprioceptifs placés sur chaque axe pour mesurer en permanence leur position et leur vitesse, et enfin l'effecteur, ou organe terminal, qui est en interaction avec l'environnement. [6]

6. Architecture des robots manipulateurs

Un robot manipulateur est composé principalement de la base, des segments, des articulations, des actionneurs et de l'organe terminal. L'architecture générale d'un robot manipulateur est illustrée dans la figure I.4.

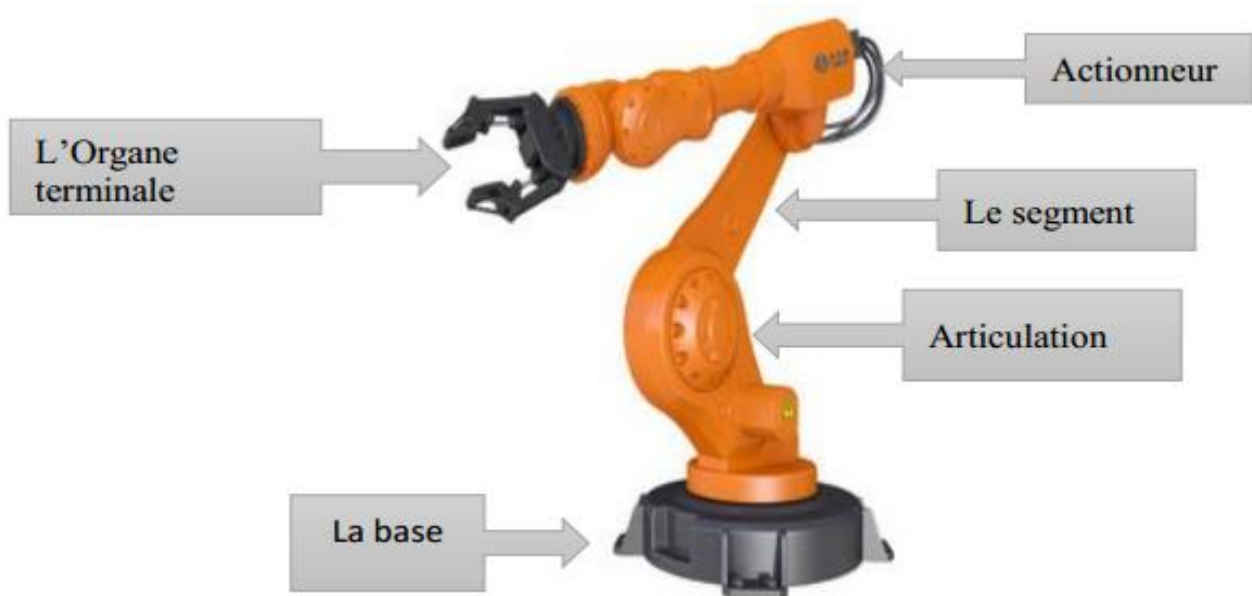


Figure I.4. Architecture d'un Robot manipulateur. [7]

6.1 La base

La base du manipulateur est fixée sur le lieu du travail. Ceci est le cas de la quasi-totalité des robots industriels. [8]

6.2 Le porteur

Le porteur représente l'essentiel du système mécanique articulé, il a pour rôle d'amener l'organe terminal dans une situation donnée imposée par la tâche (la situation d'un corps peut être définie comme la position et l'orientation d'un repère attaché à ce corps par rapport à un repère de référence). Il est constitué de : [8]

6.2.1 Segment

Corps solides rigides susceptibles d'être en mouvement par rapport à la base du porteur, et les uns par rapport aux autres. [8]

6.2.2 Articulation

Une articulation lie deux corps successifs en limitant le nombre de degré de liberté, de l'un par rapport à l'autre. Soit m le nombre de degré de liberté résultant, encore appelé mobilité de

l'articulation. La mobilité d'une articulation est telle que $0 \leq m \leq 6$, lorsque $m=1$, ce qui est fréquemment le cas en robotique, l'articulation est dite simple : rotoïde (R) ou prismatique (P). [8]

➤ **Articulation rotoïde (R)**

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

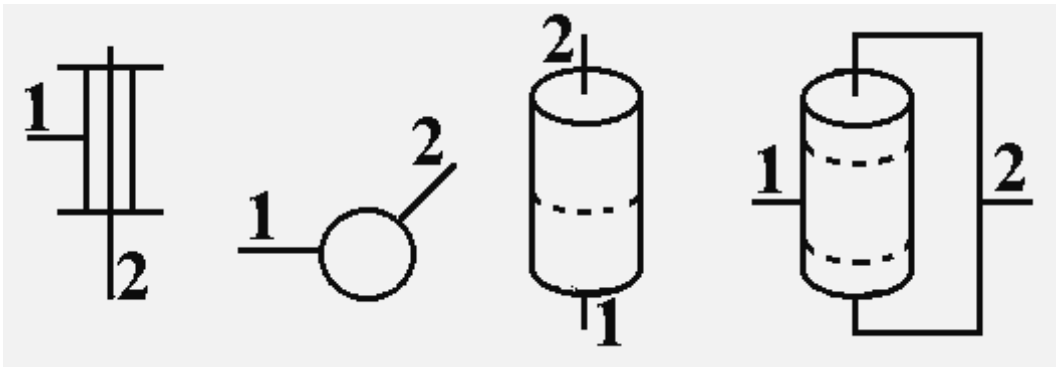


Figure I.5. Représentation d'une articulation rotoïde.

➤ **Articulation prismatique (P)**

Il s'agit d'une articulation de type glissière, notée P, réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe. [8]

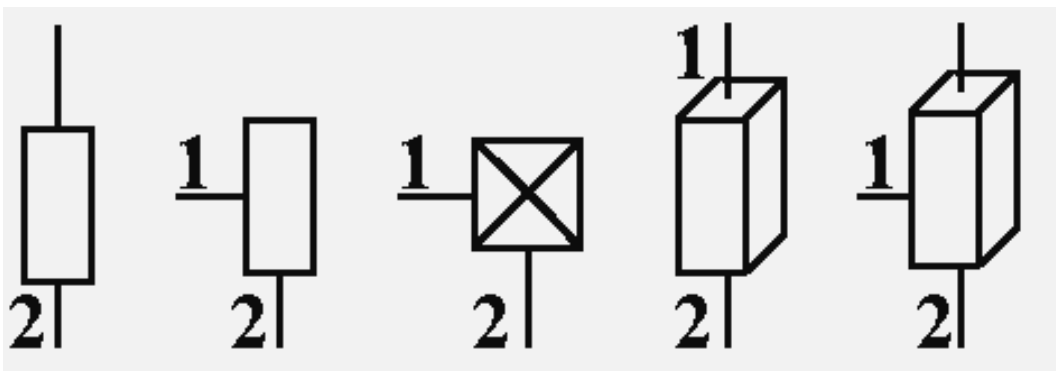


Figure I.6. Représentation d'une articulation prismatique.

6.3 L'actionneur

Les actionneurs utilisent fréquemment des moteurs électriques à aimant permanent, à courant continu, à commande par l'induit. On trouve de plus en plus de moteurs à commutation électronique (sans balais), ou, pour de petits robots, des moteurs pas à pas.

Pour les robots devant manipuler de très lourdes charges (par exemple, une pelle mécanique), les actionneurs sont le plus souvent hydrauliques, agissant en translation (vérin hydraulique) ou en rotation (moteur hydraulique). [8]

6.4 L'organe terminal (Effecteur)

Ça regroupe tout dispositif destiné à manipuler des objets (dispositifs de serrage, dispositifs magnétiques, à dépression, ...), ou à les transformer (outils, torche de soudage, pistolet de peinture, ...). En d'autres termes, il s'agit d'une interface permettant au robot d'interagir avec son environnement. [8]

7. Classification des robots

On peut classer les robots d'un point de vue fonctionnel ou d'après leur structure géométrique :

7.1 Classification fonctionnelle

7.1.1 Manipulateur à commande manuelle

Le manipulateur à commande manuelle conçu et réalisé pour les manipulations particulièrement difficiles, répond spécifiquement aux exigences liées à la manutention de charges élevées, dans tous les points du volume de travail sans effort et dans des conditions optimales d'ergonomie et de sécurité. [9]



Figure I.7. Manipulation de bobines de papier ou de film plastique. [9]

7.1.2 Manipulateur automatique

La commande peut se faire par automate programmable on peut distinguer entre manipulateurs à cycle fixe et manipulateurs à cycle programmable.

La figure I.8 montre un bras manipulateur qui exerce des mouvements de soudure sans l'intervention de l'homme. [10]



Figure I.8. Manipulateur automatique (kuka-robot de soudage à l'arc). [10]

7.1.3 Robots programmables

Ces robots répètent les mouvements qu'on leur a appris ou programmés sans informations sur l'environnement ou la tâche effectuée. On peut aussi faire la distinction entre robots « Playback » qui reproduit la tâche apprise et robots à commande numérique qui peuvent être programmés hors-ligne. [11]



Figure I.9. Robot mobile programmable. [12]

7.1.4 Robots intelligents

On trouve actuellement des robots de seconde génération qui sont capables d'acquérir et d'utiliser certaines informations sur leur environnement (systèmes de vision, détecteurs de proximité, capteurs d'efforts, ...). Les robots de troisième génération sont capables de comprendre un langage oral proche du langage naturel et de se débrouiller de façon autonome dans un environnement complexe grâce à l'utilisation de l'intelligence artificielle. [11]



Figure I.10. Robot avec système de vision intelligent. [13]

7.2 Classification géométrique

On peut aussi classer les robots suivant leur configuration géométrique, autrement dit l'architecture de leur porteur. Ci-dessous, nous soulignerons les caractéristiques les plus importantes pour chaque structure.

7.2.1 La structure cartésienne (PPP)

- 3 axes \perp 2 à 2, série, PPP, 3 DDL.
- Très bonne précision.
- Lent. [14]

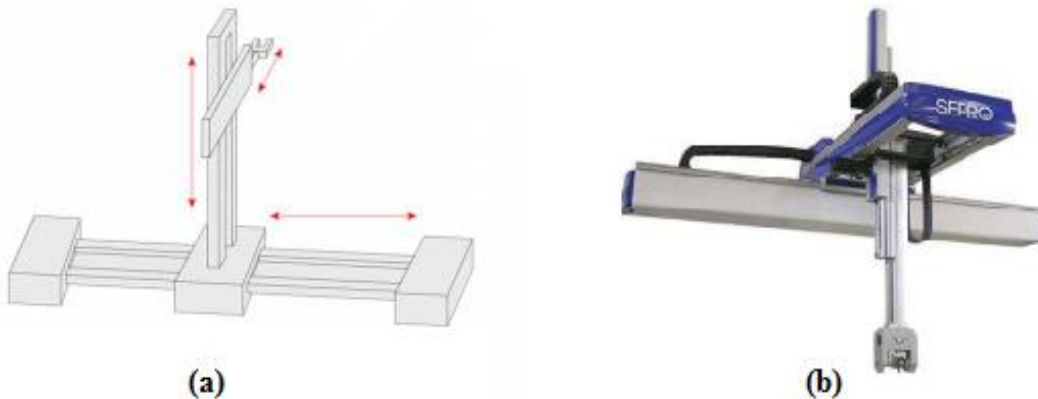


Figure I.11. (a) Structure cartésienne, (b) Robot cartésien(SEPRO). [15]

7.2.2 La structure cylindrique (RPP) ou (PRP)

- 3 axes, série, RPP/ PRP, 3 DDL.
- Espace de travail cylindrique.
- Très rapide. [14]



Figure I.12. (a) Structure cylindrique, (b) Robot cylindrique (Plate Crane). [15]

7.2.3 La structure sphérique ou polaire

- 3 axes, série, RRP, 3 DDL.
- Espace de travail sphérique.
- Grande charge utile. [14]

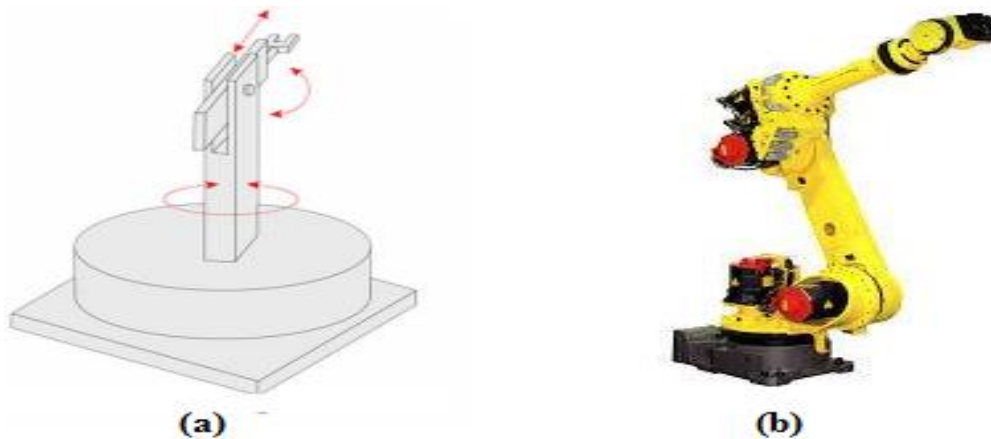


Figure I.13. (a) Structure sphérique, (b) Robot sphérique (FANUC L1000+A). [15]

7.2.4 La structure dite SCARA

- 3 axes, série, RRP, 3 DDL.
- Espace de travail cylindrique.
- Précis, Très rapide. [14]

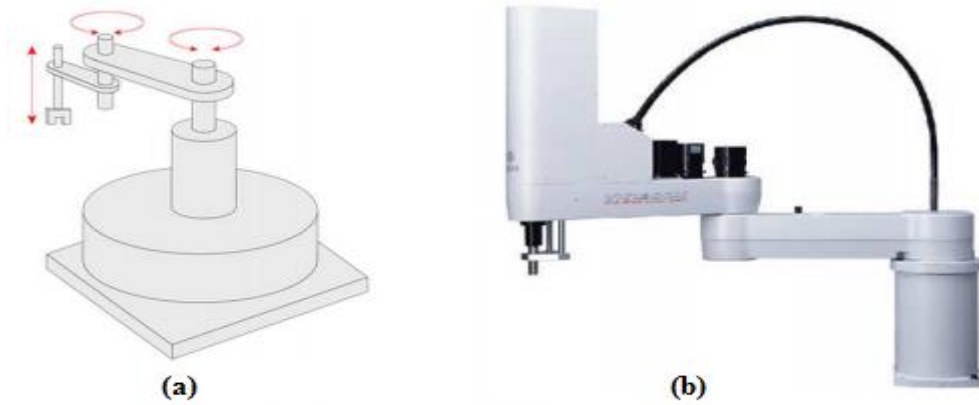


Figure I.14. (a) Structure SCARA, (b) Robot type SCARA (ADTECH). [15]

7.2.5 La structure 3R (anthropomorphe)

- Reproduit la structure d'un bras humain.
- 3 axes, série, 3R, 3DDL. [14]



Figure I.15. (a) Structure anthropomorphic, (b) Robot anthropomorphic (Stäubli). [15]

7.2.6 La structure parallèle

- Plusieurs architectures parallèles sont possibles (structure Delta, structure planes, structure Hexapode)
- Plusieurs chaînes cinématiques en parallèle.
- Espace de travail réduit.
- Précis (grande rigidité de la structure).
- Rapide. [14]



Figure I.6. (a) Structure parallèles DELTA, (b) Robot type DELTA. [15]

8. Critère de choix d'un robot

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

- La charge maximum transportable (de quelques kilos à quelques tonnes), à déterminer dans les conditions les plus défavorables (en élongation maximum).
- L'architecture de la structure mécanique articulée, le choix est guidé par la tâche à réaliser.
- Le volume de travail, est accessible par l'outil du robot qui peut balayer l'outil, le volume dépend de la géométrie du robot, de la longueur des segments et du débattement des articulations.
- Le positionnement absolu, correspondant à l'erreur entre un point souhaité (réel) – défini par une position et une orientation dans l'espace cartésien – et le point atteint et calculé via le modèle géométrique inverse du robot. Cette erreur est due au modèle utilisé, à la quantification de la mesure de position, à la flexibilité du système mécanique. En général, l'erreur de positionnement absolu, également appelée précision, est de l'ordre de 1 mm.
- La répétitivité d'un robot est l'erreur maximale de positionnement répété de l'outil en tout point de son espace de travail. En général, la répétitivité < 0.1 mm.
- La vitesse de déplacement (vitesse maximum en élongation maximum), accélération.
- La masse du robot.
- Le coût du robot.
- La maintenance.

9. Domaines d'application des robots

La robotique permet de réaliser des tâches répétitives, pénible, dangereuse et dans des milieux hostiles, avec précision et fiabilité, dans tout environnement, à des cadences élevées et continue,

tout en palliant au manque de main-d'œuvre, favorisant la haute productivité, et donnant ainsi à la robotique un fort intérêt humain, technique et économique. Ce qui lui a permis de se diversifier dans de nombreux domaines d'utilisation qui sont les suivants. [15]

➤ **Domaines industriels**

- La transformation de pièces : moulage, usinage, perçage, rectification.
- La préparation de surface : peinture, enduction, projection thermique/plasma, grenailage.
- L'assemblage : rivetage, vissage, soudage, insertion, collage.
- Les manutentions : manipulation de pièces, chargement/déchargement de machine.
- La distribution.
- La palettisation et le conditionnement.
- L'inspection et le test.
- Le nettoyage.

➤ **Domaine médical**

- Opérations.
- Diagnostics.
- Radiothérapie.
- Prothèses.

➤ **Domaine militaire**

- Désamorçage.
- Déminage.

➤ **Autres domaines**

- Nucléaire : Manipulation des produits radioactifs.
- Spatial : Exploration des planètes.
- Sous-marin : Forage.
- Dessin : Robographe (imitation de gestes humain).

10. Conclusion

Nous avons donnée dans ce chapitre une vue générale sur le domaine de la robotique en passant par la définition de la robotique et le robot ainsi que leur historique, les types des robots, les

éléments constitutifs d'un robot, l'architecture des robots manipulateurs, la classification des robots tel que la classification fonctionnel et géométrique, nous avons cité également les domaines d'applications les plus réponde.

L'utilisation de la robotique devient donc plus en plus importante et les avancées technologiques et l'intelligence artificielle permettent toujours des améliorations que ce soit dans les architectures et la composition ou encore dans les fonctionnalités et l'autonomie.

Chapitre II :

Conception matérielle et logicielle

1. Introduction

Avant d'entamer le troisième chapitre qui est le cœur de notre projet, nous allons présenter au cours de ce chapitre un ensemble d'éléments nécessaires à la fabrication de notre bras robotique. Dans la partie conception matérielle nous fournirons un ensemble d'informations sur la carte de commande Arduino telles que sa définition, son principe de fonctionnement, les types et les familles de carte Arduino...etc. ainsi que le module Bluetooth Arduino qui est l'outil de communications avec notre bras manipulateur et ses caractéristiques, finalement les servomoteurs et leur fonctionnement. Dans la partie logicielle, nous allons décrire les logiciels utilisés, tels que « l'IDE », pour programmer Arduino et « App Inventor » afin de créer une application Android.

2. Conception matérielle

Le circuit de commande de notre système est basé complètement sur un circuit programmable dit « Arduino », l'avantage de ce dernier est de minimiser la surface des cartes électroniques et par conséquent minimiser la quantité de composants électroniques utilisés.

2.1 Carte de commande Arduino

2.1.1 Présentation

Arduino est un projet créé par une équipe de développeurs, composée de six individus. Cette équipe a créé le "système Arduino" [16]. L'Arduino est une petite carte électronique, équipé non pas d'un microprocesseur, mais d'un microcontrôleur. Ce n'est donc pas un ordinateur, et elle ne possède donc pas de système d'exploitation mais elle permet de connecter un bon nombre de composants (LEDs, moteurs, servomoteurs, capteurs, écran LCD, Wifi, Bluetooth...) et de les faire fonctionner de manière autonome en la programmant via son logiciel Arduino IDE (ou d'autre logiciel comme Scratch, Arduiblock, ou encore Processing...) avec un langage de programmation assez simple (dérivé du C++) idéal donc pour réaliser des projets de Domotiques, de Robotiques ou d'Informatique Embarqué.

On parle de carte de prototypage, car cette petite merveille permet de créer des prototypes assez simplement. En effet, celle-ci possède des broches appelées PIN qui permettent de brancher très facilement divers composants afin de tester les branchements et les programmes, une fois que l'on obtient ce que l'on veut, il ne reste plus qu'à souder les composants pour obtenir un prototype [17].

2.1.2 Principe de fonctionnement

Le fonctionnement de tel système à Arduino se passe par ces étapes :

- ✓ On conçoit ou on ouvre un programme existant avec le logiciel Arduino.
- ✓ On vérifie ce programme avec le logiciel Arduino (compilation).
- ✓ Si des erreurs sont signalées, on modifie le programme.
- ✓ On charge le programme sur la carte.
- ✓ On câble le montage électronique.
- ✓ L'exécution de programme est automatique après quelques secondes.
- ✓ On alimente la carte soit par le port USB, soit par une source d'alimentation autonome (pile 9 volts par exemple).
- ✓ On vérifie que notre montage fonctionne. [17]

2.1.3 Les familles de carte Arduino

Il existe trois familles de cartes Arduino selon le fabricant :

- Les dites « officielles » qui sont fabriquées en Italie par le fabricant officiel : Smart Projects.
- Les dits « compatibles » qui n'est pas fabriqués par Smart Projects, mais qui sont totalement compatibles avec les Arduino officielles.
- Les « autres » fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, ...). [18]

2.1.4 Les types de carte Arduino

Actuellement, il existe plus de 20 versions de module Arduino, nous citons quelques un afin d'éclaircir l'évaluation de ce produit scientifique et académique :

- ✓ Le NG d'Arduino, avec une interface d'USB pour programmer et usage d'un ATmega8.
- ✓ L'Arduino Mini, une version miniature de l'Arduino en utilisant un microcontrôleur ATmega168.
- ✓ L'Arduino Nano, une petite carte programme à l'aide porte USB cette version utilisant un microcontrôleur ATmega168 (ATmega328 pour une plus nouvelle version).
- ✓ Le LilyPad Arduino, une conception de minimaliste pour l'application *wearable* en utilisant un microcontrôleur ATmega168.
- ✓ Le NG d'Arduino plus, avec une interface d'USB pour programmer et usage d'un ATmega168.
- ✓ L'Arduino Bluetooth, avec une interface de Bluetooth pour programmer en utilisant un microcontrôleur ATmega168.
- ✓ L'Arduino Diecimila, avec une interface d'USB et utilise un microcontrôleur ATmega168.

- ✓ L'Arduino Duemilanove ("2009"), en utilisant un microcontrôleur l'ATmega168 (ATmega328 pour une plus nouvelle version) et actionné par l'intermédiaire de la puissance d'USB/DC.
- ✓ L'Arduino Mega, en utilisant un microcontrôleur ATmega1280 pour I/O additionnel et mémoire.
- ✓ L'Arduino UNO, utilisations microcontrôleur ATmega328.
- ✓ L'Arduino Mega2560, utilisations un microcontrôleur ATmega2560, et possède toute la mémoire à 256 KBS. Elle incorpore également le nouvel ATmega8U2 (ATmega16U2 dans le jeu de puces d'USB de révision 3).
- ✓ L'Arduino Leonardo, avec un morceau ATmega32U4 qui élimine le besoin de raccordement d'USB et peut être employé comme clavier.
- ✓ L'Arduino Esplora : ressemblant à un contrôleur visuel de jeu, avec un manche et des sondes intégrées pour le bruit, la lumière, la température, et l'accélération. **[19]**

Parmi ces types, nous avons choisi une carte Arduino UNO (carte Basique). L'intérêt principal de cette carte est de faciliter la mise en œuvre d'une telle commande qui sera détaillée par la suite.

2.1.5 Applications

Plate-forme logicielle et matérielle de création d'objets numériques, Arduino permet de commander des circuits électroniques qui interagissent avec le milieu qui les entoure. Connectés notamment à des capteurs sonores, thermiques, de mouvement, ces circuits électroniques peu coûteux, dénommés microcontrôleurs, peuvent en retour générer des images, actionner un bras articulé, envoyer des messages sur Internet, etc. Des dizaines de milliers d'artistes, de designers, d'ingénieurs, de chercheurs, d'enseignants et même d'entreprises l'utilisent pour réaliser des projets dans de multiples domaines :

- prototypage rapide de projets innovants utilisant l'électronique, Arduino facilitant l'expérimentation en amont de la phase d'industrialisation.
- production artisanale d'objets numériques et de machines-outils à faible coût dans la perspective d'une culture d'appropriation technologique favorisant le bricolage et la débrouille.
- captation et analyse de données scientifiques (environnement, énergie, etc.) à des fins éducatives, de recherche ou d'appropriation citoyenne.

- spectacle vivant, grâce aux nombreuses fonctions d'interaction offertes par Arduino, il est possible de créer des performances de VJing, utiliser le mouvement des danseurs pour générer en temps réel des effets sonores et visuels dans un spectacle.
- installations d'arts numériques, Arduino permettant de réaliser des œuvres d'art interagissant de manière autonome avec le public.
- Mode et design textile, plusieurs stylistes et designers investissant ce domaine créatif en exploitant les possibilités offertes par l'intégration de l'électronique notamment dans des vêtements (e-textile).
- projets pédagogiques à destination d'étudiants, de professionnels ou du grand public selon les porteurs de ces initiatives : écoles supérieures, centres de formation spécialisée ou des Media Labs. [20]

2.1.6 Bonnes raisons de choisir Arduino

En fait, il existe de nombreux microcontrôleurs disponibles sur le marché tels que BX-24 Phidgets, Basic Stamp et Parallax de Netmedia Raspberry Pi et tous ont de puissantes capacités et ont la capacité de contrôler différents composants électroniques. Bien sûr, avec un taux de préférence différent, mais que distinguer l'Arduino est un ensemble de choses qui font la différence entre lui et les autres le plus important :

- **Simplicité** : la pièce Arduino est conçue pour répondre aux besoins de tous les pros, professeurs, étudiants et amateurs d'électronique interactive.
- **Prix**: L'Arduino est moins cher que d'autres panneaux du même type.
- **Auto assemblage** : on peut télécharger la fiche technique privée Arduino gratuitement à partir du site officiel et acheter les morceaux et l'installer nous-mêmes.
- **Multiplateforme** : Le programme Arduino a la capacité de fonctionner sous Windows, Mac OS, Linux et la plupart des autres commandes électroniques seulement sous Windows.
- **Un environnement logiciel simple et convivial** : l'environnement de programmation conçu pour être facile pour les débutants, fixes et puissants pros.
- **Open Source** :
 - ✓ Le logiciel Open Source est écrit en C ++ et il est de plus téléchargeable. Les programmeurs peuvent le modifier en fonction de leurs besoins.
 - ✓ le matériel Open Source est principalement constitué de microcontrôleurs ATmega8 et ATmega168 et les schémas sont publiés sous une licence Créative Commons, permettant aux concepteurs de circuits électroniques de concevoir leurs propres circuits. [21]

2.2 La carte Arduino UNO

2.2.1 Présentation

Version de base du système basé sur un microcontrôleur Atmega328 cette carte délivre un maximum de 14 points d'entrée / sortie digitaux et de 6 entrées analogiques. Existe avec un processeur en version DIL sur support ou SMD soudée. Outre le processeur, la principale évolution consiste en un remplacement du chipset de l'interface de programmation USB de la puce d'interface USB / Série FTDI en une puce Atmel8U2 ou sur les dernières versions 16U2 .Voici les principales caractéristiques de la carte Arduino Uno: [21]

Catégorie	Valeur
Microcontrôleur	AT méga 328
Fréquence d'horloge	16 Mhz
Tension de service	5V
Tension d'entrée (recommandée)	7-12V
Tension d'entée (limites)	6-20V
Ports numériques	14 entrées/sorties
Ports analogiques	6 entrées
Courant maxi par broche d'E/S (cc)	40mA
Courant maxi par broche 3.3V	50mA
Mémoire	32Ko Flash,2Ko SRAM,1Ko EEPROM
Interface	USB

Tableau II.1.Synthèse des caractéristiques. [21]

2.2.2 Constitution de la carte Arduino UNO

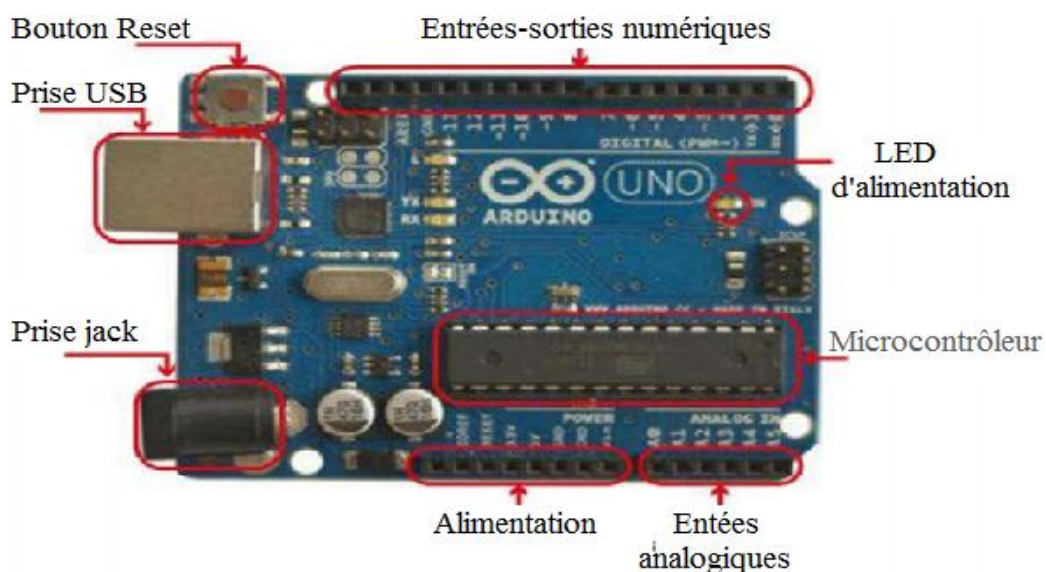


Figure II.1. Composition de carte Arduino UNO.

➤ Le Microcontrôleur ATmega328

Un microcontrôleur ATmega328 est un circuit intégré qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit au temps des pionniers de l'électronique. Aujourd'hui, en soudant un grand nombre de composants encombrants, tels que les transistors, les résistances et les condensateurs tout peut être logé dans un petit boîtier en plastique noir muni d'un certain nombre de broches dont la programmation peut être réalisée en langage C.



Figure II.2. Microcontrôleur ATmega328P.

Le microcontrôleur ATmega328 est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère d'un ordinateur. Globalement, l'architecture interne de ce circuit programmable se compose essentiellement sur :

- **La mémoire Flash**: C'est celle qui contiendra le programme à exécuter. Cette mémoire est effaçable et réinscriptible mémoire programme de 32Ko (dont bootloader de 0.5 ko).
- **RAM** : c'est la mémoire dite "vive", elle va contenir les variables du programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur. Sa capacité est 2 ko.
- **EEPROM** : C'est le disque dur du microcontrôleur. On y enregistre des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée. Cette mémoire ne s'efface pas lorsque l'on éteint le microcontrôleur ou lorsqu'on le reprogramme. [21]

➤ L'alimentation électrique:

La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

L'alimentation externe (non-USB) peut être soit un adapter secteur (pouvant fournir typiquement de 3V à 12V sous 500mA) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise 2.1mm positif au centre dans le connecteur jack de la carte. Les fils en

provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées GND (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V et la carte pourrait être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V. [21]

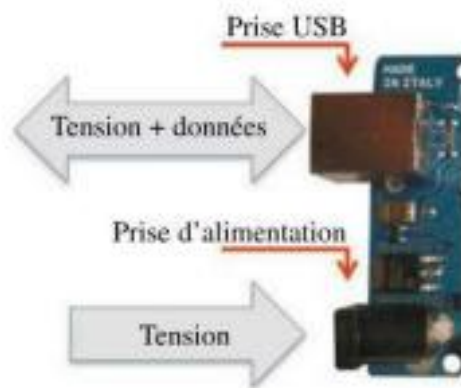


Figure II.3. Alimentation de la carte Arduino. [21]

➤ Entrées et sorties numériques

Chacune des 14 broches numériques de la carte UNO (numérotées des 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions « `pinMode`, `digitalWrite` et `digitalRead` » du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de "rappel au plus" (pull-up) (déconnectée par défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction « `digitalWrite(broche, HIGH)` ». De plus, certaines broches ont des fonctions spécialisées :

- **Communication Série:** Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données séries de niveau TTL. Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U programmé en convertisseur USB-vers-série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.
- **Interruptions Externes:** Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. Voir l'instruction « `attachInterrupt()` » pour plus de détails.
- **Impulsion PWM (largeur d'impulsion modulée):** Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction « `analogWrite()` ».

- **SPI (Interface Série Périphérique):** Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.
- **I2C:** Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (TwoWire Interface - Interface "2 fils"), disponible en utilisant la librairie «Wire/I2C » (ou TWI - Two-Wire interface - interface "2 fils").
- **LED :** Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte. [19]

➤ Les ports de communications

- La carte Arduino UNO a de nombreuses possibilités de communications avec l'extérieur : Communication série UART TTL (5V), grâce aux broches numériques 0 (RX) et 1 (TX). On utilise (RX) pour recevoir et (TX) transmettre (les données séries de niveau TTL). Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega328.
- Programmé en convertisseur USB – vers – série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.

Comme un port de communication virtuel pour le logiciel sur l'ordinateur, La connexion série de l'Arduino est très pratique pour communiquer avec un PC. [19]

2.3 Bluetooth

2.3.1 Définition

Bluetooth est un type de communications sans fil permettant l'échange bidirectionnel (maître/esclave) de données à très courte distance (10 m) en utilisant des ondes radio UHF sur une bande de fréquence de 2,4 GHz. Son objectif est de simplifier les connexions entre les appareils électroniques en supprimant des liaisons filaires. Elle peut remplacer par exemple les câbles entre ordinateurs, tablettes, haut-parleurs, téléphones mobiles entre eux ou avec des imprimantes etc.



Figure II.4. Logo de Bluetooth. [22]

Un esclave pourra communiquer avec un seul maître, mais un maître pourra dialoguer avec plusieurs esclaves. Pour son utilisation, elle se passe en plusieurs étapes :

- ✓ Le maître se met en mode « reconnaissable ».
- ✓ L'esclave trouve le maître et demande à s'y connecter.
- ✓ Le maître accepte la connexion.
- ✓ Les périphériques sont alors appariés (ou associés).
- ✓ La communication peut commencer. [23]

2.3.2 Le module Bluetooth Arduino

Le Module Bluetooth Arduino est la plateforme populaire Arduino avec une connexion série, Bluetooth à la place d'une connexion USB, très faible consommation d'énergie, très faible portée (sur un rayon de l'ordre d'une dizaine de mètres), faible débit, très bon marché et peu encombrant. Il existe deux sortes de module Bluetooth (HC-05 et HC-06), tous les deux sont compatibles avec Arduino et les deux modules peuvent être utilisés en mode COMMANDE, pour les programmer avec des « commandes AT », ou en mode DATA, pour échanger des données. [23]

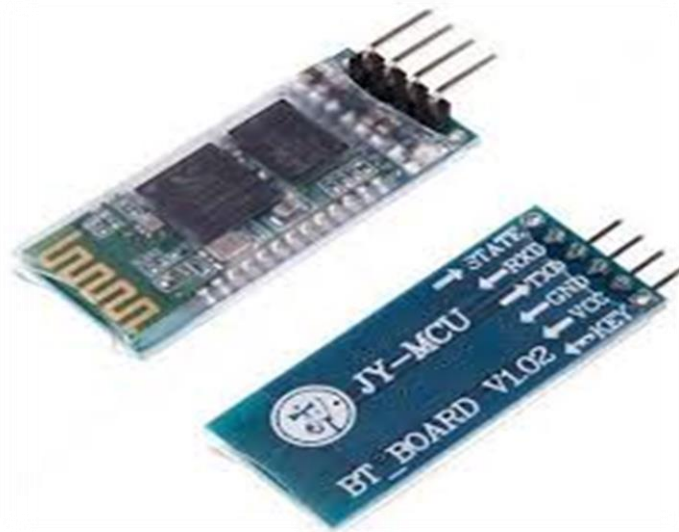


Figure II.5. Modules Bluetooth pour l'Arduino. [24]

La différence entre le HC-05 et HC-06 est que le HC-05 est doté du micro logiciel « complet » : de nombreuses commandes AT, et peut être à la fois maître et esclave. D'autre part, le micro logiciel HC-06 ne peut être qu'esclave, avec des commandes AT très limitées. En d'autres termes :

- Le module HC-05 peut établir une connexion avec d'autres modules. Par exemple, un robot étant un maître et connecte au module Bluetooth esclave, ou en mode esclave pour créer une connexion sans fil avec un ordinateur portable.

- Le module HC-06 peut uniquement être un esclave. Cela fait qu'il ne sert que pour connecter un ordinateur portable en tant que maître à un robot avec un module esclave, par exemple, pour un pont série sans fil. [19]

Pour notre travail nous avons utilisé le module Bluetooth HC-06.

2.4 Servomoteur

2.4.1 Présentation

Un servomoteur est un système qui a pour but de produire un mouvement précis en réponse à une commande externe, C'est un actionneur (système produisant une action) qui intègre l'électronique, la mécanique et l'automatique. Un servomoteur est capable d'attendre des positions prédéterminées dans les instructions qui lui ont été données, puis de les maintenir. [25]

Un servomoteur est composé de :

- ✓ Un micro moteur à courant continu.
- ✓ Un réducteur en sortie de ce moteur diminuant la vitesse et augmentant le couple.
- ✓ Un potentiomètre (faisant fonction de diviseur résistif) qui génère une tension variable, proportionnelle à l'angle de l'axe de sortie.
- ✓ Un axe dépassant hors du boîtier avec différents bras ou roues de fixation.
- ✓ Un dispositif électronique d'asservissement.

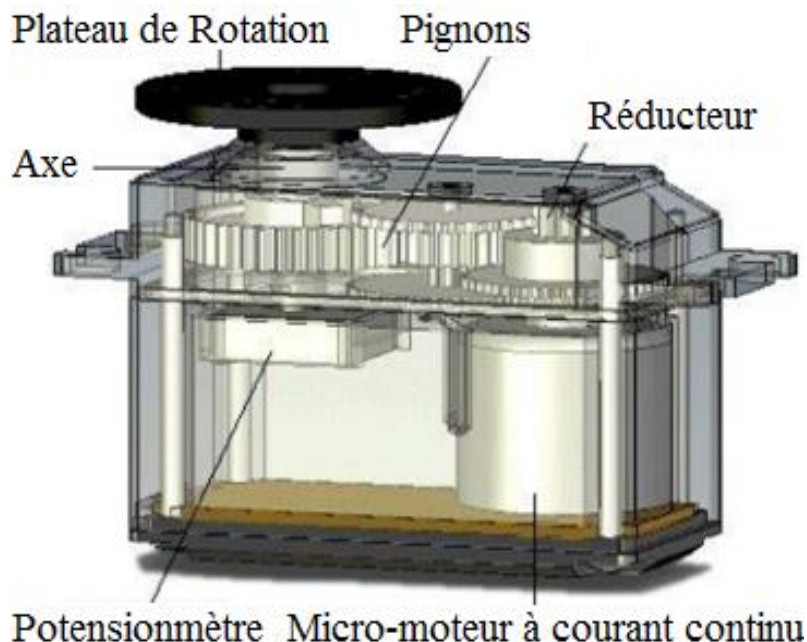


Figure II.6. Composition d'un servomoteur. [26]

Dans l'univers Arduino, le servomoteur est essentiellement utilisé dans les applications robotiques. Le servomoteur possède trois fils de connexion pour pouvoir fonctionner, deux fils servent à son alimentation, le dernier étant celui qui reçoit le signal de commande :

- Rouge : pour l'alimentation positive (4.5V à 6V en général).
- Noir ou marron : pour la masse (0V).
- Orange, jaune, blanc, ... : entrée du signal de commande. [26]

2.4.2 Fonctionnements des servomoteurs

Le fonctionnement interne d'un servomoteur se déroule comme suit :

Un petit circuit électronique permet de contrôler un moteur à courant continu en fonction de la position d'un potentiomètre intégré au servomoteur. La sortie du moteur à courant continu est reliée mécaniquement à une série d'engrenages qui augmente la force (le couple) du servomoteur en réduisant la vitesse de rotation de celui-ci. Quand le moteur tourne, les engrenages s'animent, le bras bouge et entraîne avec lui le potentiomètre. Le circuit électronique ajuste continuellement la vitesse du moteur pour que le potentiomètre (et par extension le bras) reste toujours au même endroit.

Le servomoteur a l'avantage d'être asservi en position angulaire, cela signifie que l'axe de sortie du servomoteur respectera la consigne d'instruction envoyée à son entrée, même si un obstacle intervient pour faire changer l'orientation de sa trajectoire, le servomoteur essaiera de conserver la position. Pour un ajustement précis de la position, le moteur et son réglage sont équipés d'un système de mesures qui détermine la position courante.

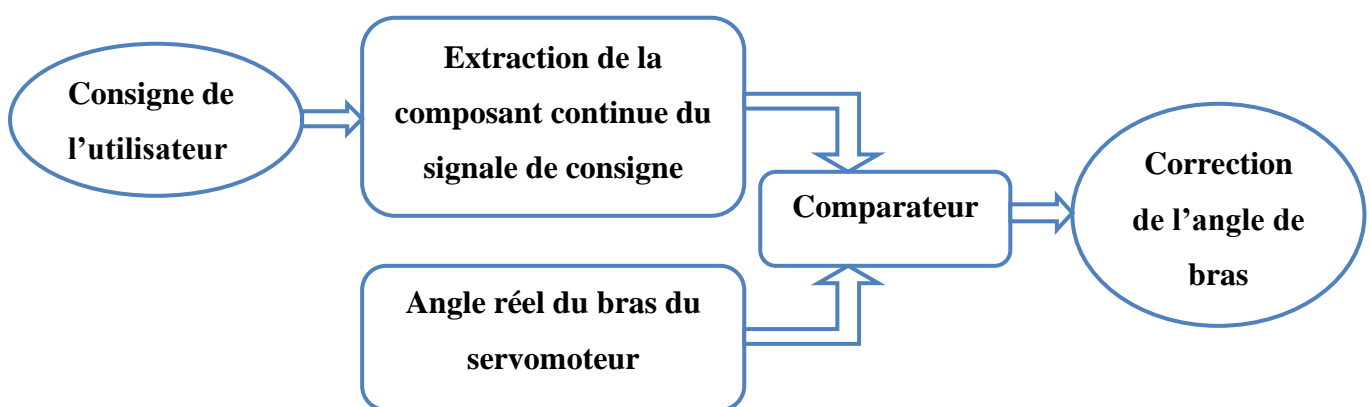


Figure II.7. Fonctionnement d'un servomoteur. [27]

Il suffit de donner une consigne au servomoteur ("reste à 45°" par exemple) et le servomoteur fera son maximum pour rester au plus près de cette consigne.

Cette consigne est transmise au moyen d'un signal numérique, d'une impulsion pour être précis. Pour que le servomoteur reste à une position donnée, il faut transmettre toutes les 20 millisecondes (soit à une fréquence de 50Hz) une impulsion d'une longueur comprise entre 1 et 2 millisecondes.

- Une impulsion de 1 ms correspond à un angle de 0° .
- Une impulsion de 2 ms correspond à un angle de 180° .
- Une impulsion de 1,5 ms correspond à un angle de 90° .

La plupart des servomoteurs fonctionnent en 5 volts, mais certains fonctionnent en 3.3 volts. [28]

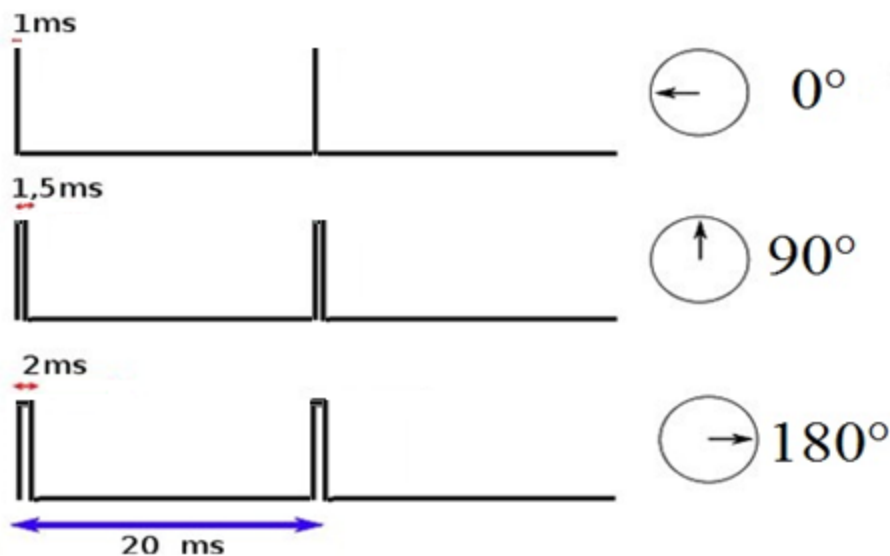


Figure II.8. Illustration du signal de contrôle. [28]

Parmi les types des servomoteurs nous avons choisir pour notre réalisation les types suivants :

Spécifications / Références	MG996R	MG90	SG90
Poids	55g	13,4g	9g
Couple d'arrêt	9.4 kg·cm (4.8 V) 11 kg·cm (6 V)	1,8 kg·cm (4.8 V) 2,2 kg·cm (6 V)	1.8 kg·cm
Tension de fonctionnement	4,8V à 7,2V	4,8V à 6V	4,8V à 5V
Vitesse de rotation	0.17 60°/s (4.8 V) 0.14 60°/s (6 V)	0.1 60°/s (4.8 V) 0.08 60°/s (6 V)	0.1 60°/s
L'angle de rotation	180°	180°	180°

Tableau II.2. Paramètres techniques des servomoteurs utilisés.

2.4.3 Les avantages et les inconvénients

➤ Les Avantages

- Aucune connaissance en programmation n'est nécessaire dans la plupart des cas.
- Vous disposez d'un petit budget pour votre projet.
- Projet de base dans lequel la précision n'est pas importante.
- Votre projet ne nécessite pas de rétroaction.
- Robot de petite taille (les servomoteurs intelligents ne sont pas encore disponibles en micro format).
- Le signal MLI est universel et que tous les servomoteurs, quels que soient la marque et le modèle, répondent au même signal.

➤ Les inconvénients

- La communication est unidirectionnelle.
- Il n'est pas possible d'obtenir de rétroaction sur la position réelle.
- Les servomoteurs standards connectés à un contrôleur nécessitent beaucoup de câblage. [15]

3. Conception logicielle

3.1 IDE Arduino

3.1.1 Description de l'IDE

L'IDE est un logiciel de programmation qui permet d'écrire, de modifier un programme et de le convertir en une série d'instructions compréhensibles pour la carte. Il programme par code, contenant une cinquantaine de commandes différentes. Le langage de l'IDE Arduino est un mélange entre le C et le C++, il possède un jeu d'instruction très riche. A l'ouverture, L'interface du logiciel Arduino se présente de la façon suivante : [20]

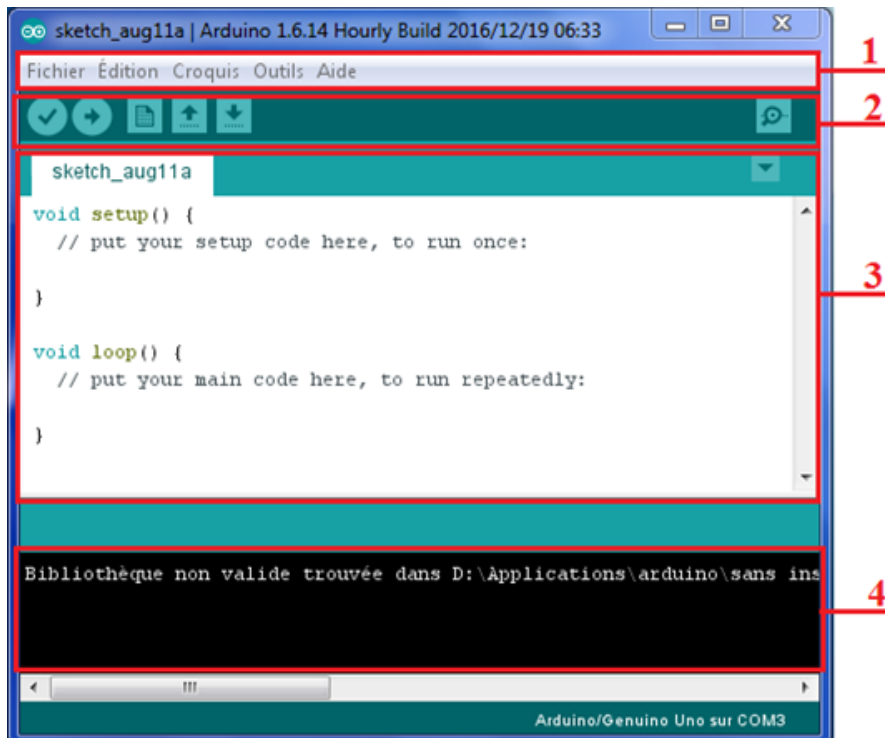


Figure II.9. Interface de logiciel IDE Arduino.

- **Barre1** : Options de configuration du programme tel que :
 - Créer un nouveau programme ou ouvrir un programme existant.
 - Enregistre/demande où enregistrer le document en cours.
 - Toute une liste se déroule pour afficher les noms d'exemples de programmes existant...etc.
- **Barre2** : boutons pour la programmation des cartes
 - Vérifier (*Verify*) : vérifier les erreurs dans le code.
 - Charge (*Upload*) : compiler le code et charge le programme sur la carte Arduino.
 - Nouveau (*New*) : créer un nouveau sketch.
 - Ouvrir (*Open*) : ouvrir un des sketches déjà présents.
 - Sauvegarder (*Save*) : sauvegarder le sketch.
 - Serial Monitor : permet d'accéder au port série (en RX/TX).
- **Barre3** : programme à créer.
- **Barre4** : débogueur (affichage des erreurs de programmation).

3.1.2 Structure d'un programme Arduino

Un programme Arduino est constitué de trois parties :

- La partie où l'on déclare les variables, affecte les broches, appelle les bibliothèques.

- L'initialisation (*Setup*) où l'on configure les broches en entrée ou en sortie, initialise les différents objets (comme le moniteur série, les composants ayant leur propre bibliothèque. Cette fonction est exécutée une seule fois après la mise sous tension de la carte ou après un reset.
- Le programme principal (*Loop*). Cette fonction est répétée indéfiniment.

Coloration syntaxique : Lorsqu'un code est écrit dans l'interface de programmation, certains mots apparaissent en différentes couleurs qui clarifient le statut des différents éléments :

- **En orange**, apparaissent les mots-clés reconnus par le langage Arduino comme des fonctions existantes. Lorsqu'on sélectionne un mot coloré en orange et qu'on effectue un clic avec le bouton droit de la souris, on a la possibilité de choisir « Find in reference » : cette commande ouvre directement la documentation de la fonction sélectionnée.
- **En bleu**, apparaissent les mots-clés reconnus par le langage Arduino comme des constantes.
- **En gris**, apparaissent les commentaires qui ne seront pas exécutés dans le programme. Il est utile de bien commenter son code pour s'y retrouver facilement ou pour le transmettre à d'autres personnes. [20]

3.2 MIT APP INVENTOR

3.2.1 Le système Android

Android est un système d'exploitation développé initialement pour les Smart phones. Il utilise un noyau Linux qui est un système d'exploitation libre pour PC et intègre tous utilitaires et les périphériques nécessaires à un smart phone. Il est optimisé pour les outils *Gmail*. Aussi, l'Android est libre et gratuit et a été ainsi rapidement adopté par des fabricants.

La société Android a été rachetée en 2007 par *Google*. Mais aujourd'hui, l'Android est utilisé dans de nombreux appareils mobiles (smart phones). Les applications sont exécutées par un processeur de type ARM à travers un interpréteur JAVA. En plus de cela, l'Android concurrence l'opérateur système d'Apple qu'il tend à dépasser en nombre d'utilisateurs. Android évolue pour mieux gérer l'hétérogénéité des appareils qu'il utilise. [29]

3.2.2 Présentation de MIT APP INVENTOR

APP INVENTOR est un logiciel en ligne développé par *Google Labs*. Abandonnée par *Google*, cette application a été reprise par le MIT (*Massachusetts Institute of Technology*) et permet de développer des applications Android, basées sur l'assemblage de blocs (langage Scratch). La

programmation se réalise en ligne, à l'aide de son navigateur préféré, et sous l'environnement logiciel de son choix (Mac, Linux ou Windows). Seules contraintes : avoir un compte *Gmail* pour pouvoir y accéder, et un accès à internet évidemment. Les informations sont stockées sur des serveurs distants. Le site est en anglais, nous pouvons basculer sur une traduction française mais, question d'habitude certainement, les fonctions des blocs sont moins compréhensibles. [15]



Figure II.10. Logo d'APP INVENTOR. [15]

3.2.3 La conception d'APP INVENTOR

L'application est composée de trois fenêtres, une pour créer l'interface graphique, une autre pour assembler les blocs de programmation et l'émulateur :

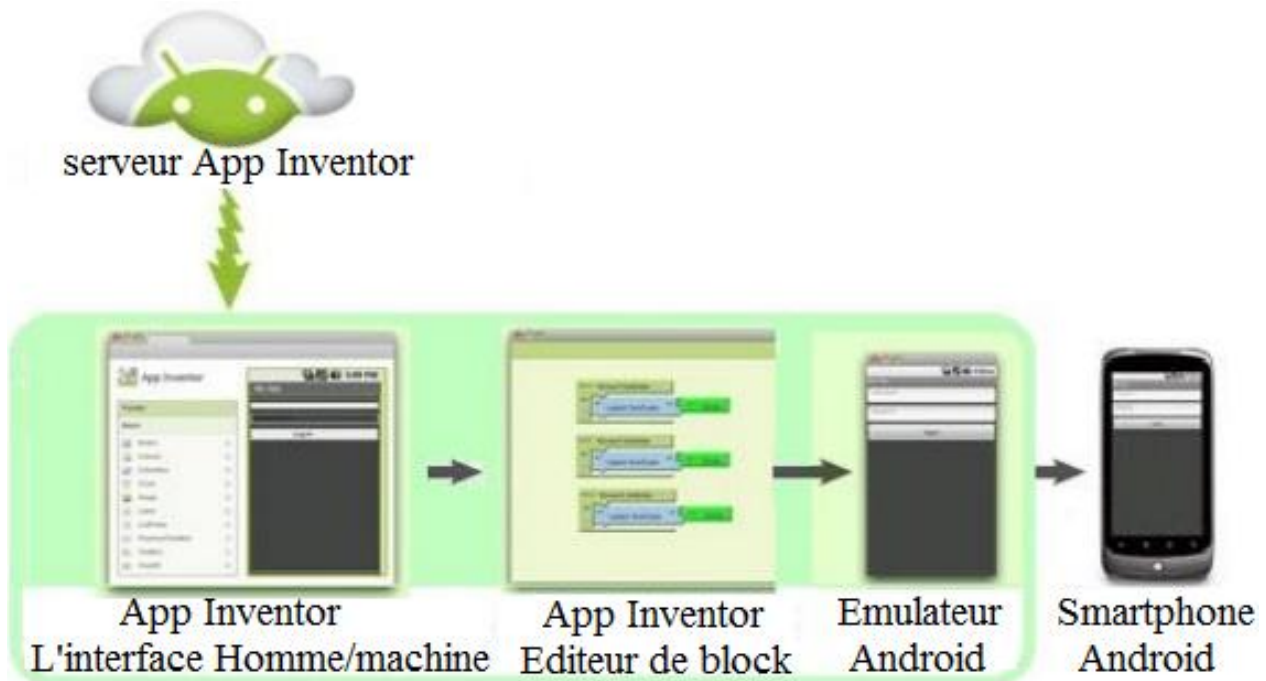


Figure II.11. Création d'une application Android par APP INVENTOR.

- **L'interface graphique** : Dans cette partie on dispose de divers blocks graphiques comme des bouton, des champs de texte, des images, des *sliders*, mais aussi des *layouts* qui servent à

délimiter les zones de notre application, des *senseurs* pour exploiter les capteurs du téléphone, et quelques autres fonctionnalités utiles de multimédia, de GPS, et de communication Bluetooth.

- **Blocs de programmation** : Dans la partie blocs nous retrouvons toutes les fonctions mathématique, logique, ou évènementiel, ce rapportant aux objets utilisés dans la partie graphique, ce qui nous permet de tirer profit d'une simplicité de programmation orienté objet sous la forme de blocs prêts programmé, qu'ils ne restent plus qu'à organiser en fonction du programme souhaité.
- **L'émulateur** : dans cette partie nous pouvons tester l'application. L'émulateur permet de remplacer un terminal réel pour vérifier le bon fonctionnement du programme. [15]

4. Conclusion

Dans ce chapitre, nous avons expliqué les deux parties essentielles de notre projet, en commençant par la partie matérielle, nous avons présenté la carte de commande qui est l'Arduino, nous avons cité les différentes familles et les différents types de cette dernière. Puis nous avons donné les raisons pour lesquelles on l'a choisie. Ensuite, la partie logicielle plus précisément l'IDE Arduino pour la programmation de la carte et le MIT APP INVENTOR pour développer l'application Android. Le chapitre suivant sera consacré à la réalisation de notre bras manipulateur, à sa mise en fonctionnement et à son évaluation.

Chapitre III :

Réalisation de bras manipulateur

1. Introduction

Dans ce chapitre, nous allons présenter les différentes étapes de réalisation de notre bras. La fabrication commence par une structure mécanique possédant trois degrés de libertés, ainsi que l'assemblage mécanique et électrique de la carte avec les servomoteurs et le module Bluetooth HC-06. Ensuite le développement d'une application Android basée sur le logiciel MIT APP INVENTOR, ce qui permet de connecter et de contrôler notre bras manipulateur via tous les téléphones Android. A la fin de ce chapitre, nous allons présenter sous forme d'organigramme le fonctionnement global du bras.

2. Elaboration de prototype

Le schéma suivant permet de mieux comprendre le fonctionnement de notre bras manipulateur qu'on désire réaliser pratiquement :

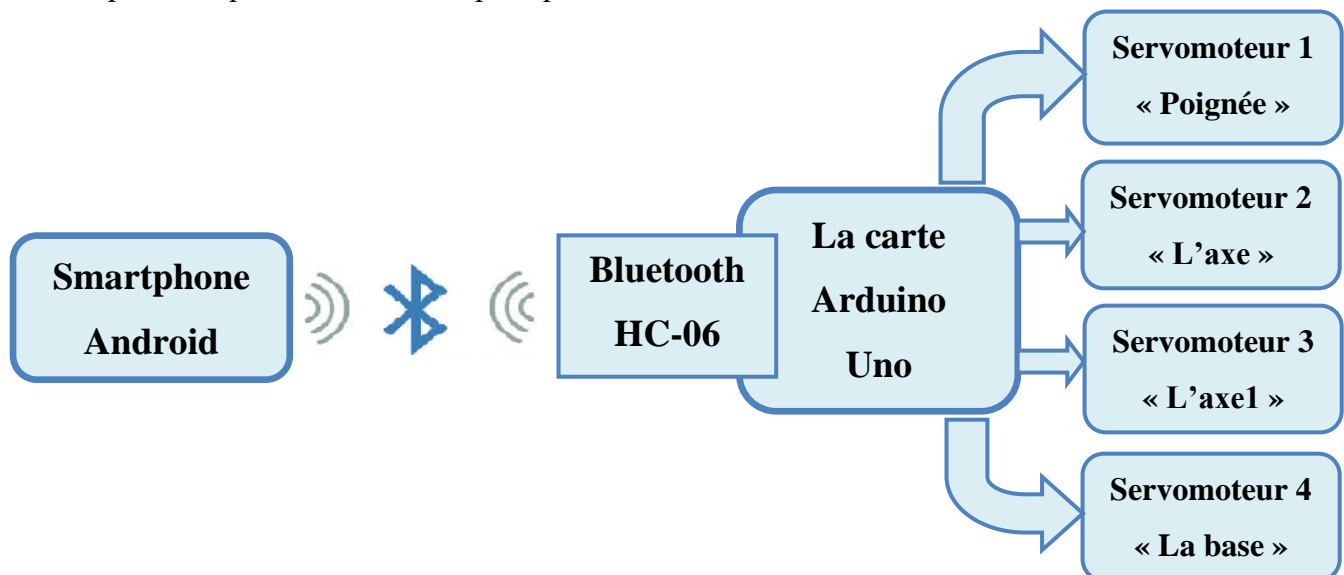


Figure III.1. Schéma global de bras manipulateur.

3. Etapes de réalisation et d'assemblage

3.1 Fabrication de la structure mécanique du bras

La structure mécanique du bras manipulateur est constituée de trois parties : la base, bras (axe 1 et axe 2) et la poignée et qui est construite en utilisant des bâtons en bois d'un diamètre de 2mm (Un poids léger pour les bras). La première étape de la conception de bras consiste à décider de la dimension et de l'espace de travail.

La Conception de la structure mécanique répond bien du cahier de charges suivant :

- ✓ Un bras manipulateur à trois degrés de liberté (trois liaisons rotondes « RRR ») possédant un volume de travail de demi-sphère.

- ✓ Les longueurs des segments : la base=7cm, l'axe 1=21cm, l'axe 2=19cm,
- ✓ Masse maximale de la charge de la pince égale à 10 g (forme sphérique).

Les mouvements exercés sont des mouvements programmés selon les besoins, dans notre cas nous avons programmé le bras manipulateur de telle façon qui effectué le maximum de positions et que ces positions soient différentes. Les figures III.2 et III.3 représentent ces mouvements :

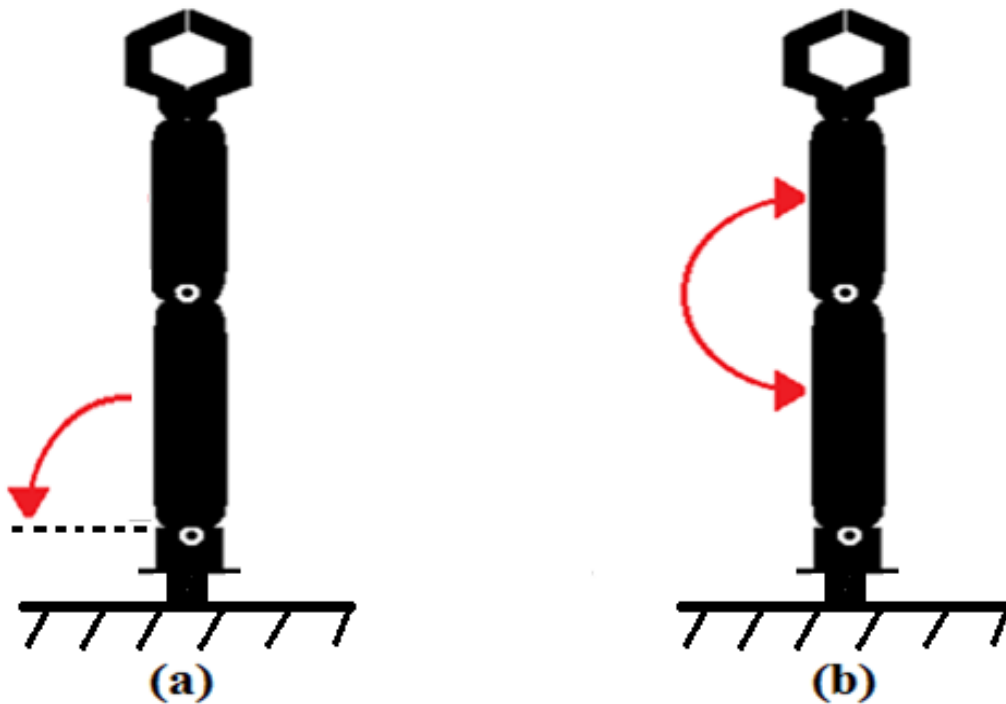


Figure III.2. Les mouvements de bras : (a) l'axe1 (90°) (b) l'axe2 (180°).

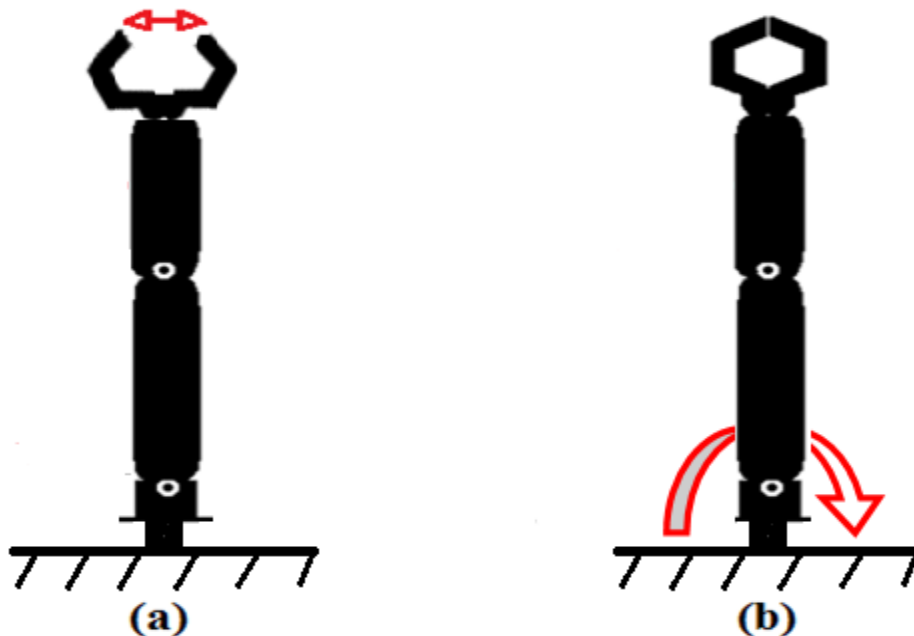


Figure III.3. Les mouvements de bras : (a) la poignée (45°) (b) la base (180°).

La base d'axe verticale doit faire un cercle lors du mouvement de l'outil dans le plan horizontal. Il est important de mentionner également que la base doit avoir un poids considérable pour qu'on puisse garder l'équilibre général du bras en cas de soulèvement d'un objet.

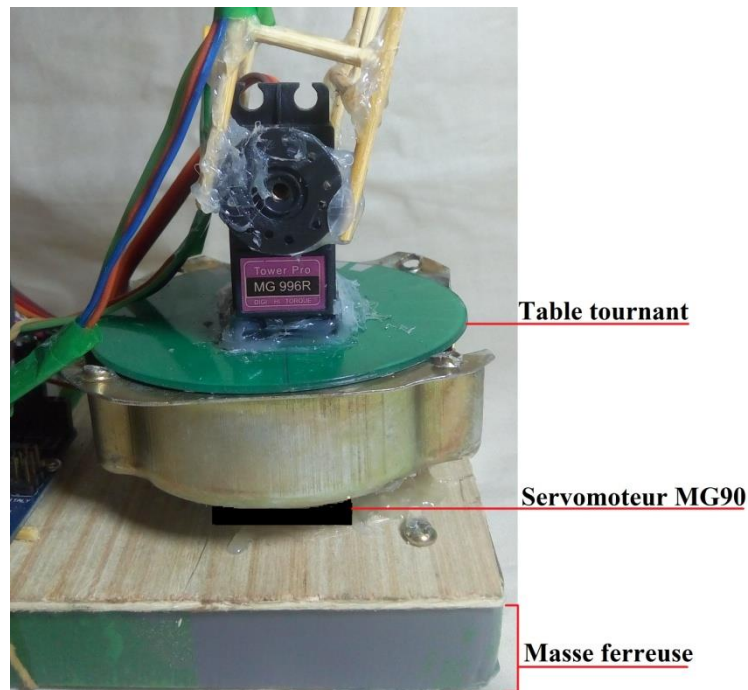


Figure III.4. Image réelle de la base.

Les deux axes (Axe 1, Axe 2), permettant un mouvement de l'outil dans le plan vertical.



Figure III.5. Image réelle de l'axe 1 et l'axe 2.

La poignée à une structure qui permet d'emballée des objets :

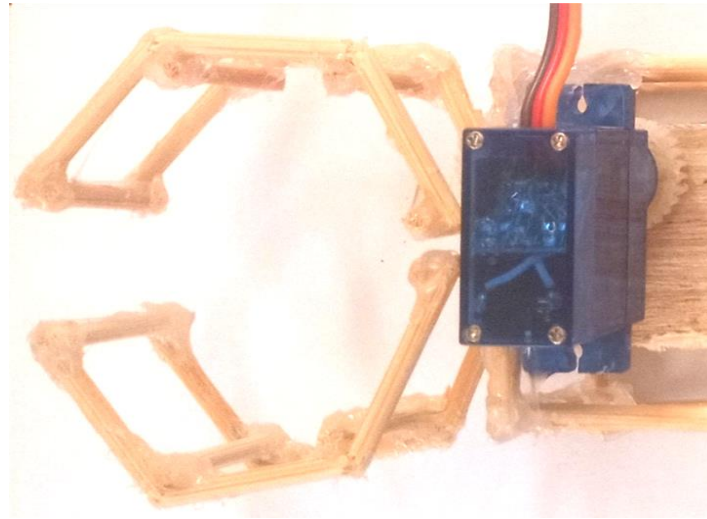


Figure III.6. Image réelle de la poignée.

3.2 Montage électrique

3.2.1 Configuration du module Bluetooth

Le module Bluetooth HC-06 possède quatre broches qui se connectent à la carte Arduino comme suit :

Les broches Tx et Rx du Bluetooth aux broches numérique 0 et 1 respectivement de la carte Arduino UNO et pour l'alimentation de module Bluetooth nous avons utilisé un transformateur 220v AC/5v DC.

Le Smartphone est connecté au Bluetooth comme suit :

- Recherche des adresses MAC des appareils disponibles depuis le menu Bluetooth du Smartphone.
- Sélection de l'adresse MAC correspondant au module Bluetooth HC-06.
- Saisie du code PIN du module Bluetooth HC-06 au niveau du Smartphone.
- Avant de connecter le Vcc, on doit d'abord allumer la carte Arduino, la LED du module Bluetooth clignotera à l'intervalle avec une cadence de 2 secondes.
- Une fois que le Smartphone est connecté au HC-06 la LED s'arrête de clignote et reste allumée.

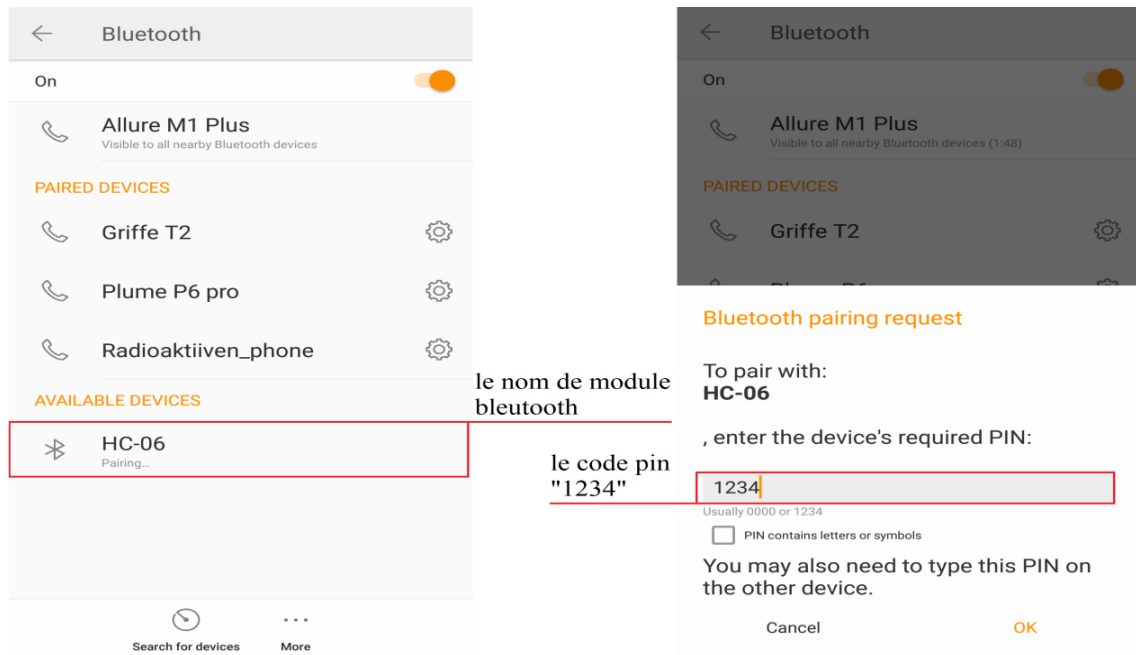


Figure III.7. L'assemblage du Smartphone avec le HC-06.

3.2.2 Câblage totale

Dans la construction du bras, nous avons utilisé quatre servomoteurs (y compris la pince), car notre structure permet un mouvement dans les trois dimensions. Un servomoteur de type MG90 est utilisé dans la base, ce qui permet un mouvement angulaire de la structure entière, deux autres aux axes 1 et 2 sont de type MG996R et MG90 respectivement pour permettre le mouvement du bras vers le haut et vers le bas, tandis que le dernier servomoteur de type SG90 est monté au bout de l'effecteur et qui permet de soulever des objets.

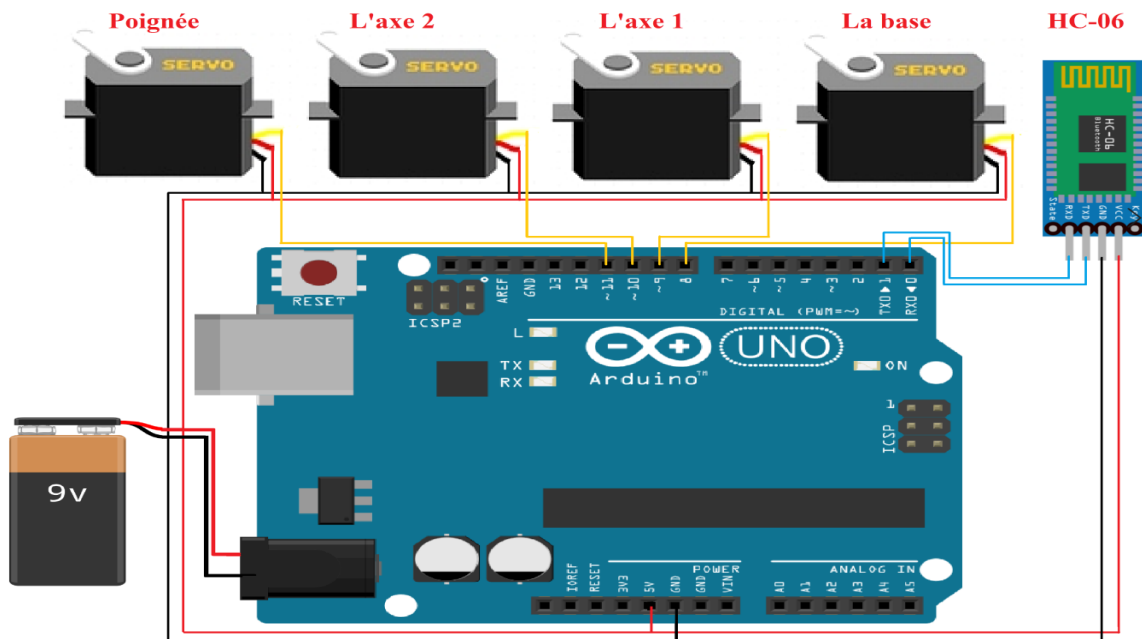


Figure III.8. Montage électrique de bras manipulateur.

3.3 Développement de l'application Android

Comme nous l'avons déjà souligné, le bras sera équipé d'un module Bluetooth pour permettre un pilotage à distance. Pour cela nous avons utilisé le MIT App Inventor (en ligne) pour créer une application Android facilitant cette commande à distance.

Avant l'utilisation de cette application un compte Google Play personnel doit être créé. Les étapes du développement de l'application sont les suivantes :

- Commencer un nouveau projet.

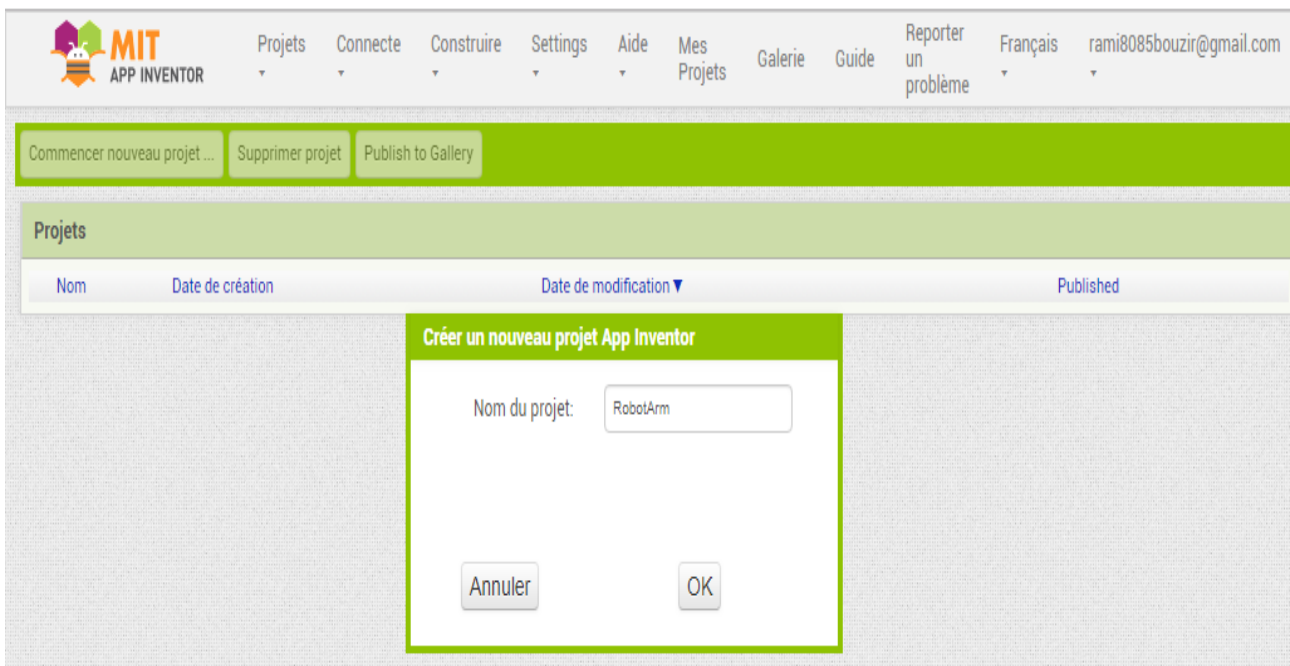


Figure III.9. L'interface d'un nouveau projet.

- Au début du projet et à partir de la section « Designer » du logiciel App Inventor on commence par la conception graphique désiré pour notre application. Les objets qu'on souhaite utiliser (les boutons, les tableaux, les ascenseurs, les zones de textes...etc.) doivent être glissés dans la « Screen1 ».

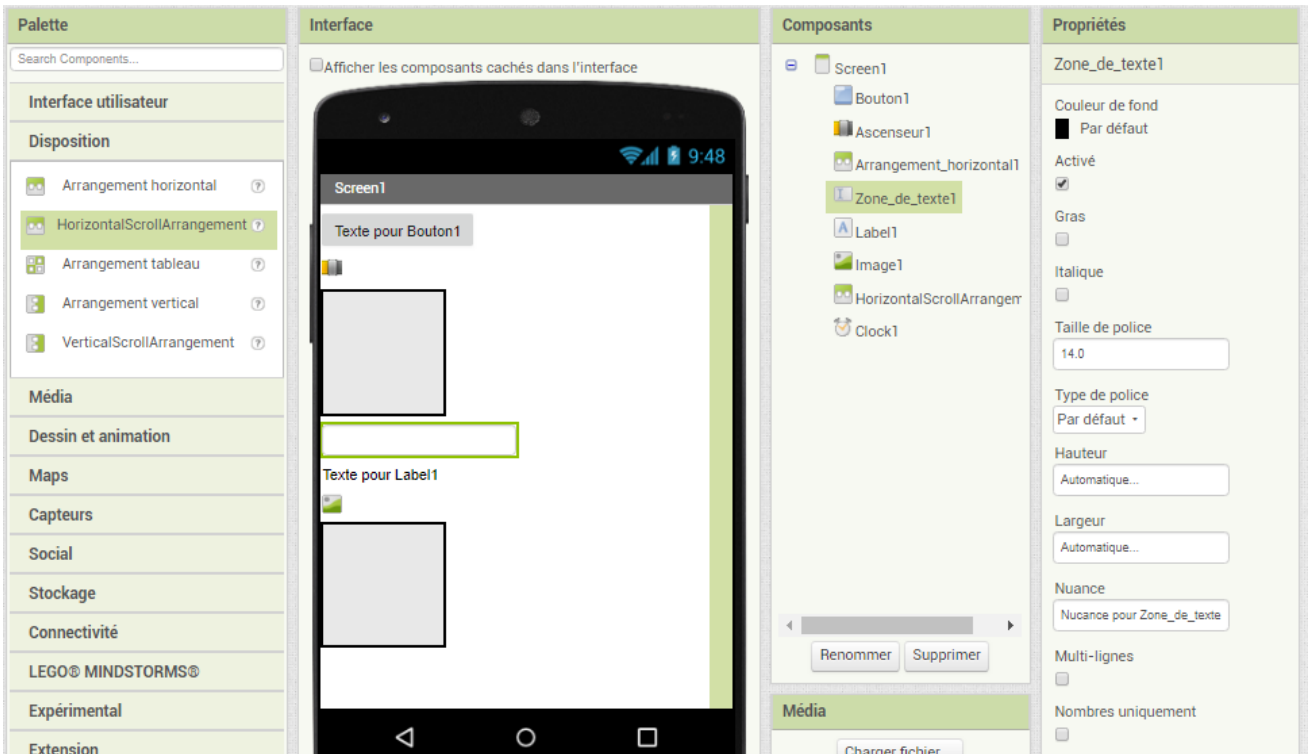


Figure III.10. Remplissage de « Screen 1 ».

- L'étape suivante consister à modifier les propriétés des différents objets utilisés (boutons, ascenseurs...) pour obtenir l'interface voulue.

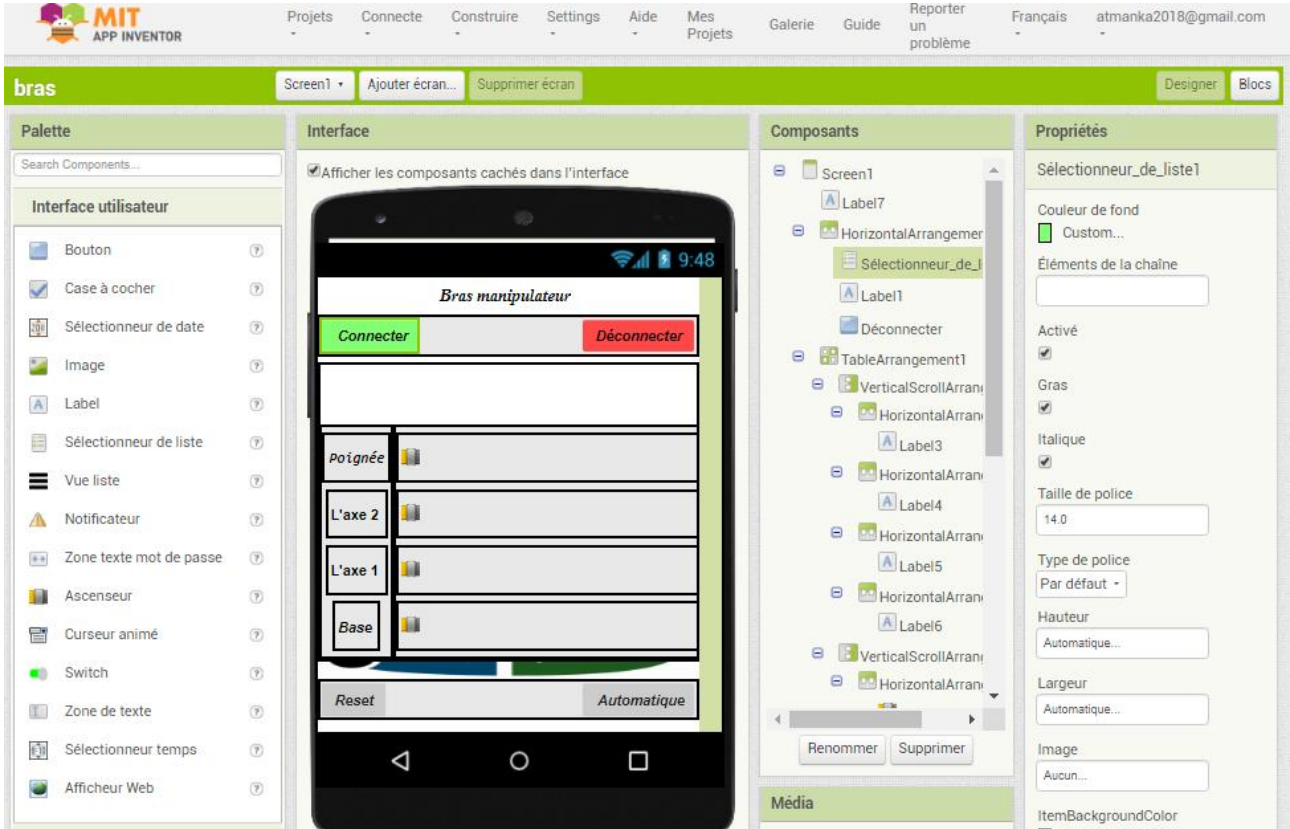


Figure III.11. L'interface de l'application sous App Inventor.

- Une fois qu'on a terminé l'interface graphique de l'application. On passe à la section « Blocs » où on programme tous les objets qui se trouvent dans le « Screen 1 », chacun selon les tâches à exécuter.
- Le Block d'initialisation de l'application sur le Smartphone :

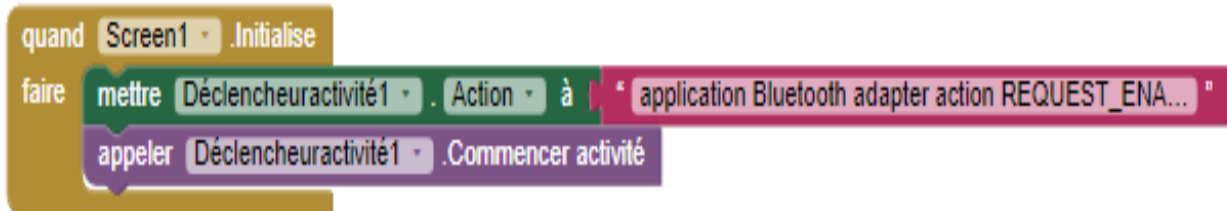


Figure III.12. Block d'initialisation de l'application.

- Les Blocks permettant d'afficher la liste des périphériques Bluetooth disponibles au Smartphone :

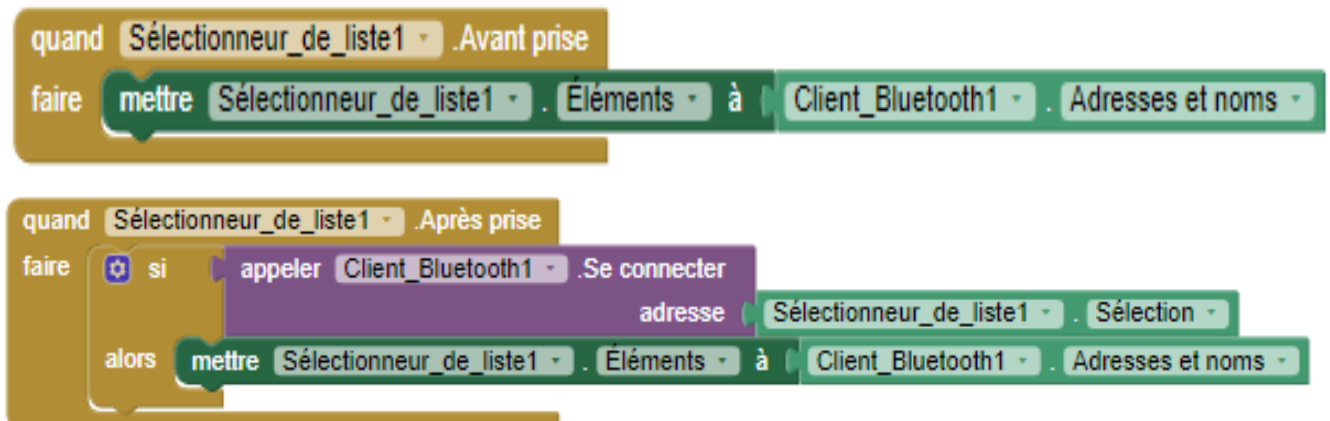


Figure III.13. Blocks de la liste des périphériques disponibles.

- Le Block qui montre l'état du Smartphone avec le module Bluetooth (connecté ou déconnecté) affiché avec une couleur différente :

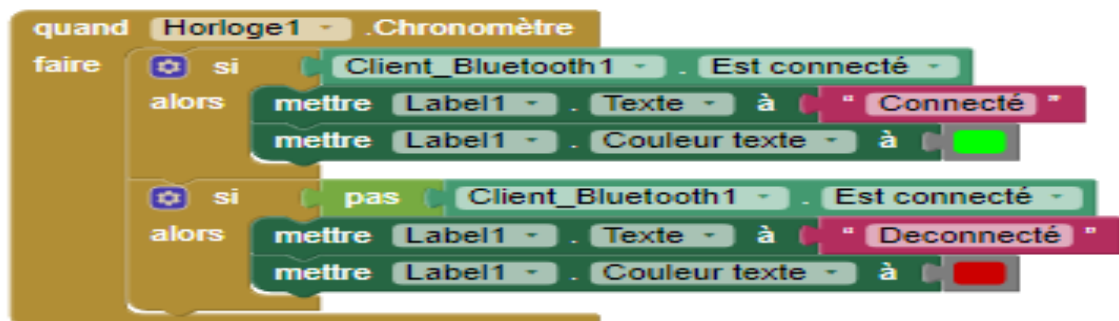


Figure III.14. Block de l'état du Bluetooth.

- Le Block du bouton « déconnecter » qui désactive les commandes de contrôle :

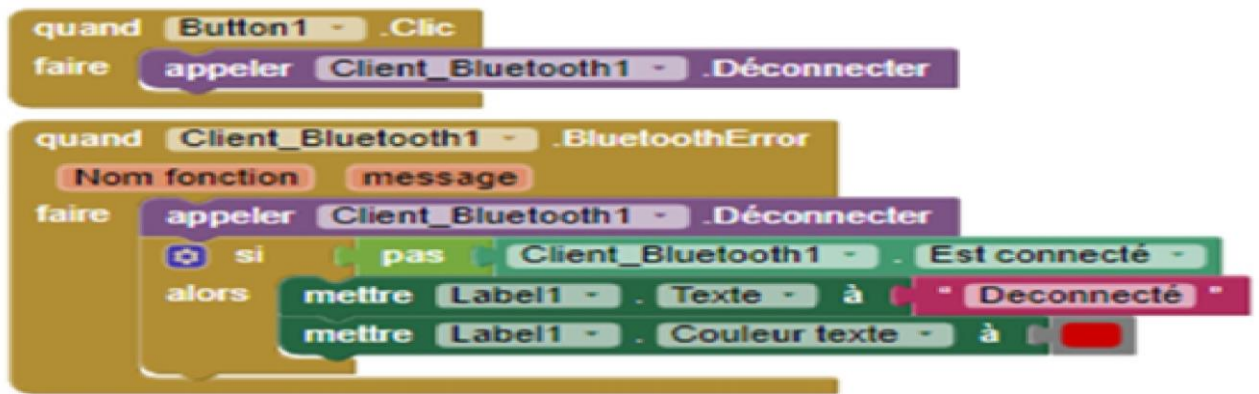


Figure III.15. Block du bouton « déconnecter ».

- Les Blocks de contrôle de position des servomoteurs. Chaque servomoteur a son propre bloc de contrôle qui est caractérisé par un angle de début et l’angle maximal à atteindre.

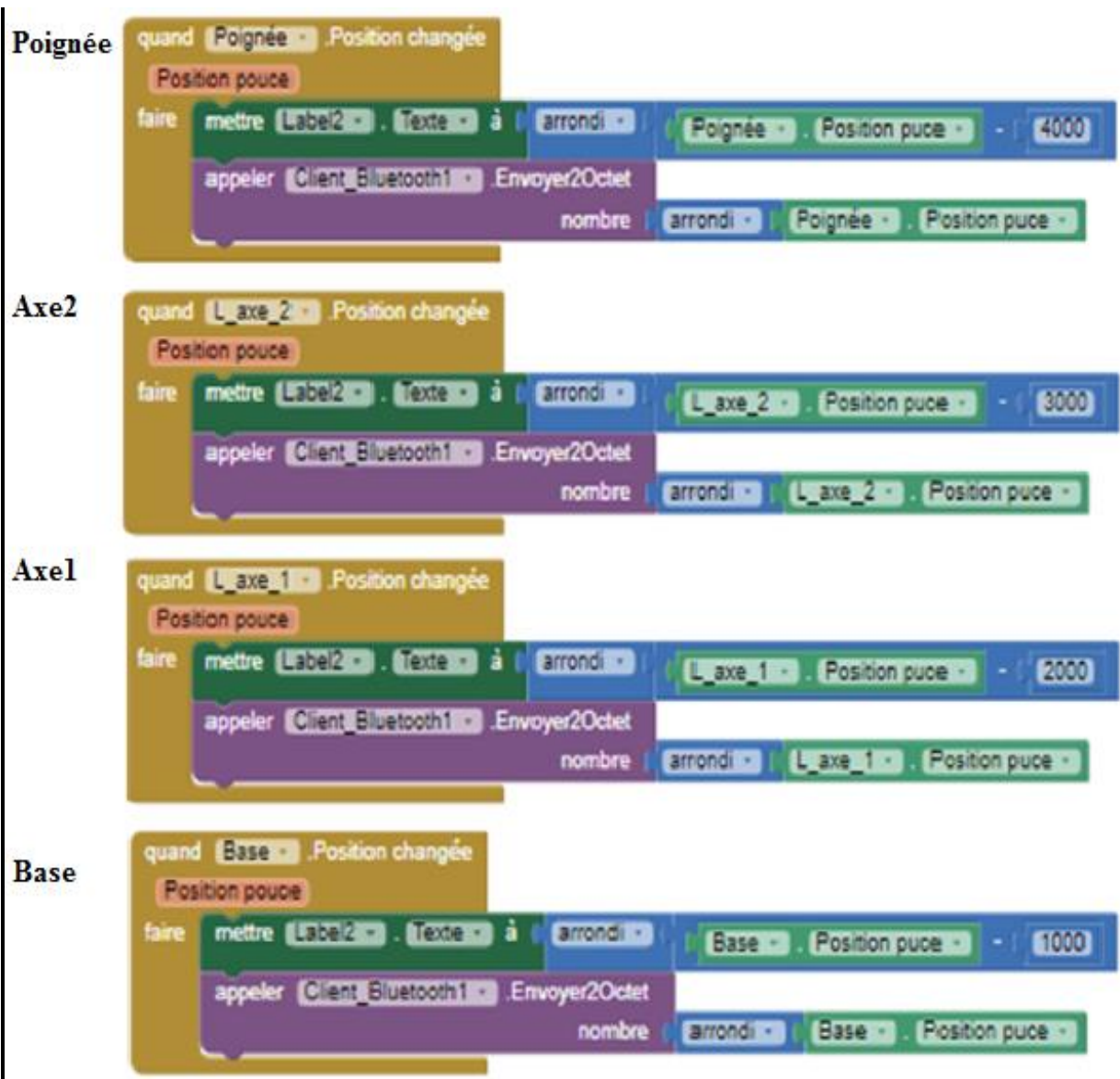


Figure III.16. Blocks de contrôle des servomoteurs.

- Block de Bouton Automatique :

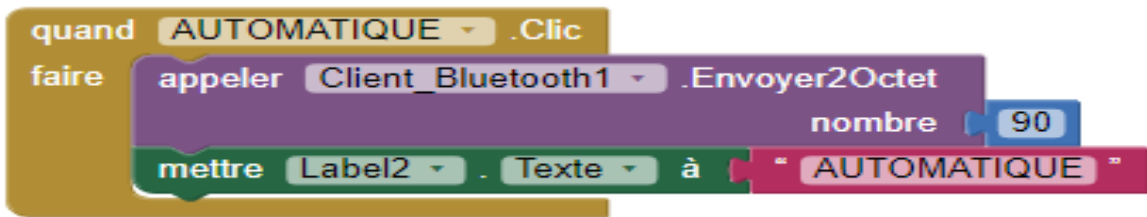


Figure III.17. Block du Bouton Automatique.

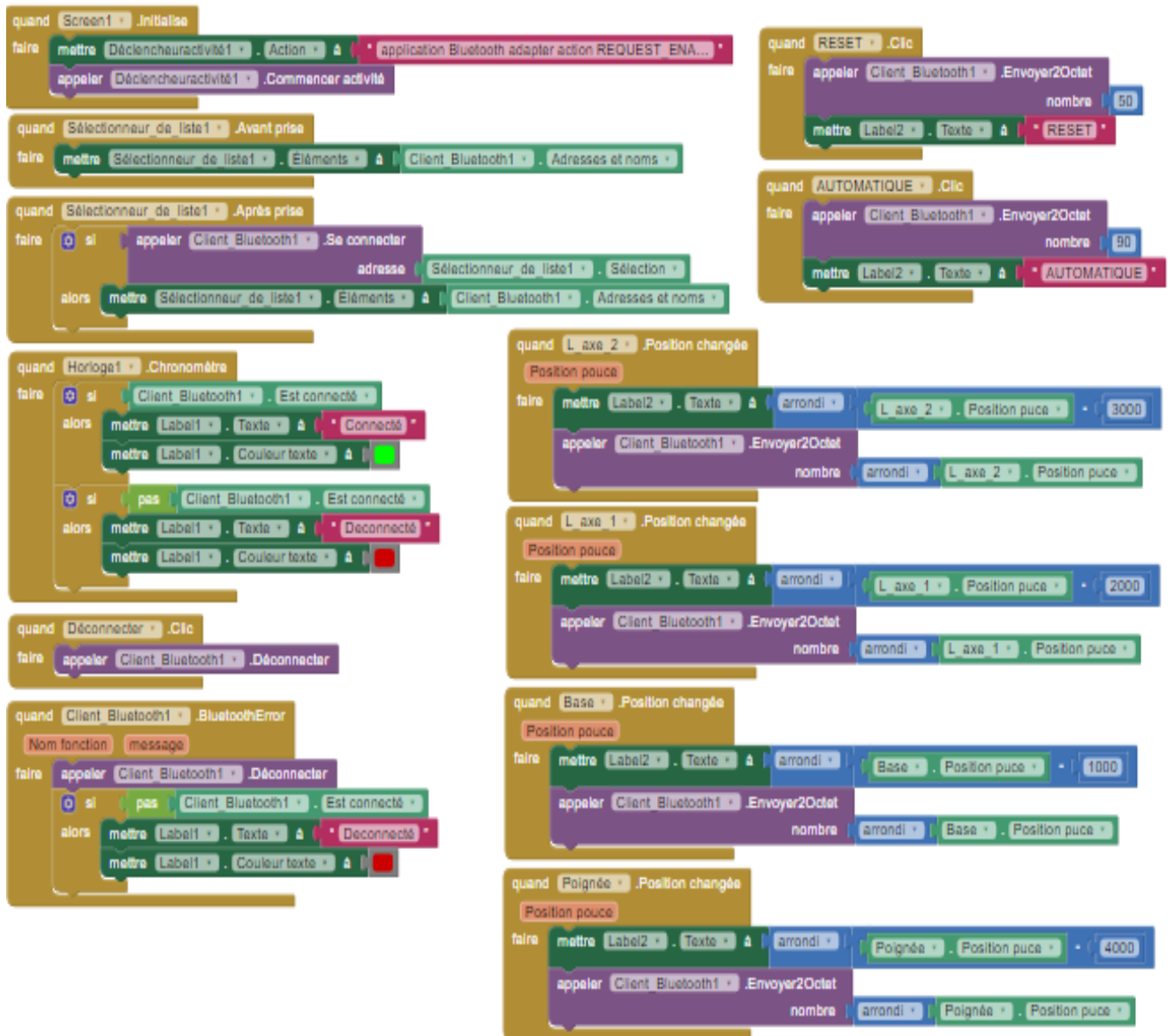


Figure III.18. Blocks globale de l'application Android.

- L'étape suivante consiste à terminer la création des blocs de l'application en cliquant sur "construire" pour créer un fichier avec une extension Android « .APK » pour que l'application soit compatible avec tous les téléphones Android.

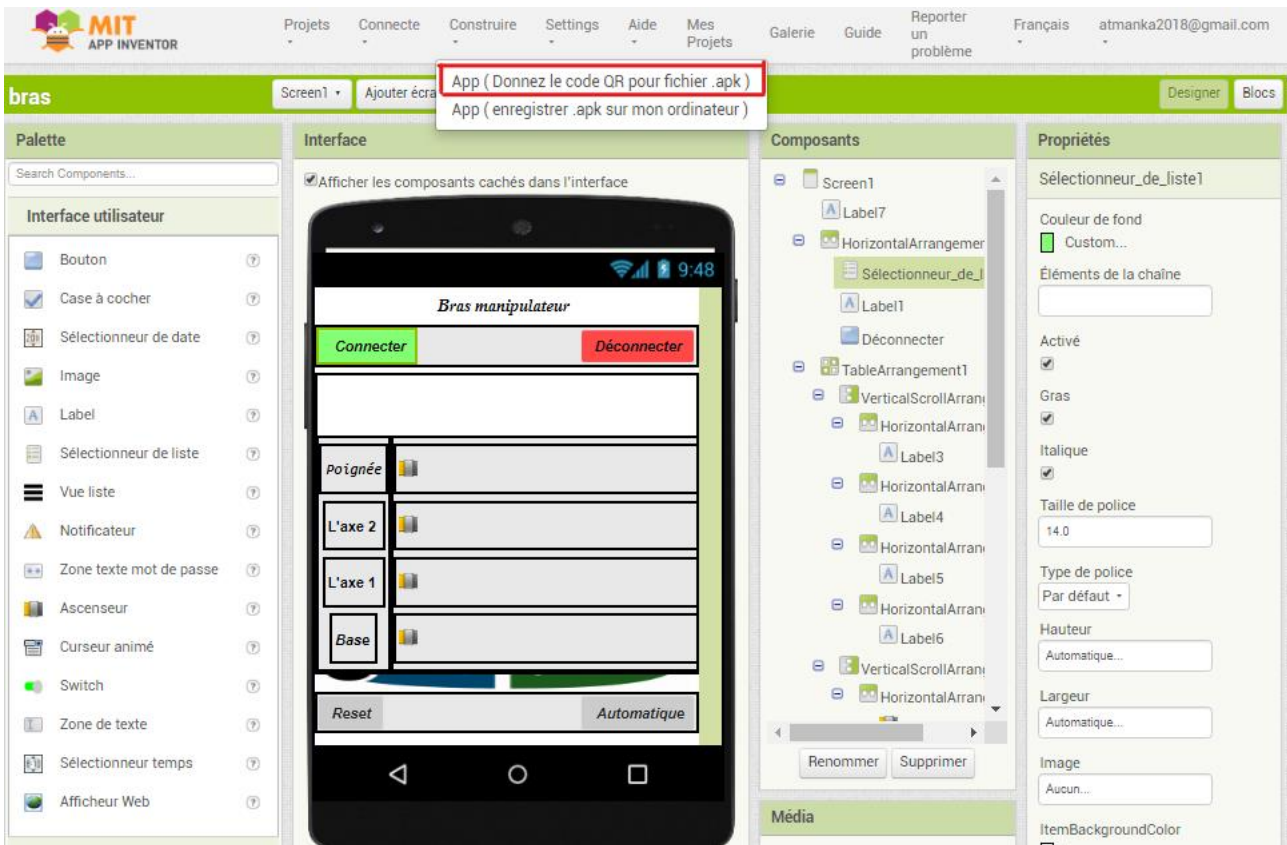


Figure III.19. Création du fichier avec l’extension Android.

➤ Dans la dernière étape, on doit définir pour chaque curseur une valeur initiale, minimale et maximale, adaptées aux articulations des bras du bras comme montrer dans le tableau suivant :

Paramètre Bouton/curseur	Valeur Max de Bluetooth	Valeur Min de Bluetooth	L’angle Max de servomoteur (degrés)	L’angle Min de servomoteur (degrés)
Base	1180	1000	180	0
Axe 1	2076	2000	180	104
Axe 2	3165	3000	165	0
Poignée	4046	4000	170	124
RESET				50
Automatique				90

Tableau III.1. Les paramètres de l'application.

➤ En haut de l'application, nous avons deux boutons pour connecter ou déconnecter le Smartphone au module Bluetooth HC-06. En bas, nous avons deux boutons, un bouton « RESET » qui réinitialise le bras manipulateur et l’autre « automatique » pour le mode de fonctionnement automatique. En arrière-plan de l’application nous avons mis le logo de notre université.

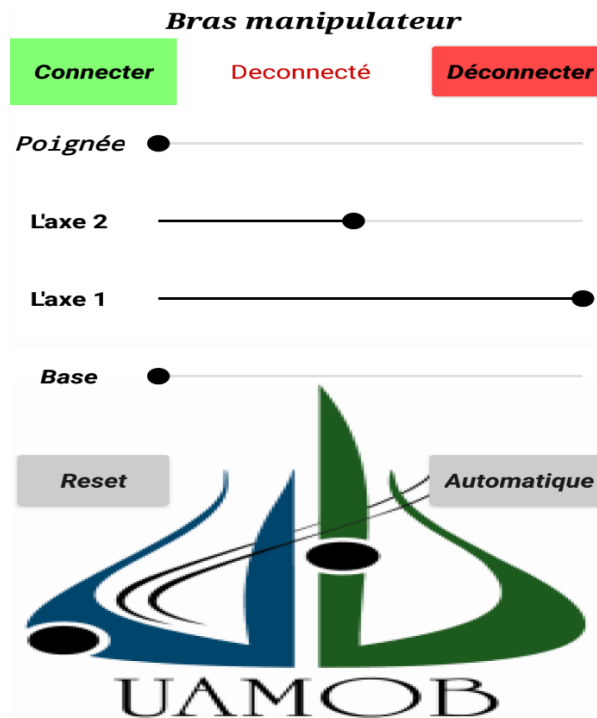


Figure III.20. L'interface de l'application sous Smartphone.

3.4 Programmation de la carte Arduino

Pour bien comprendre le programme de commande et pour des raisons de lisibilité, l'organigramme de la figure III.21 illustre le fonctionnement global du programme Arduino :

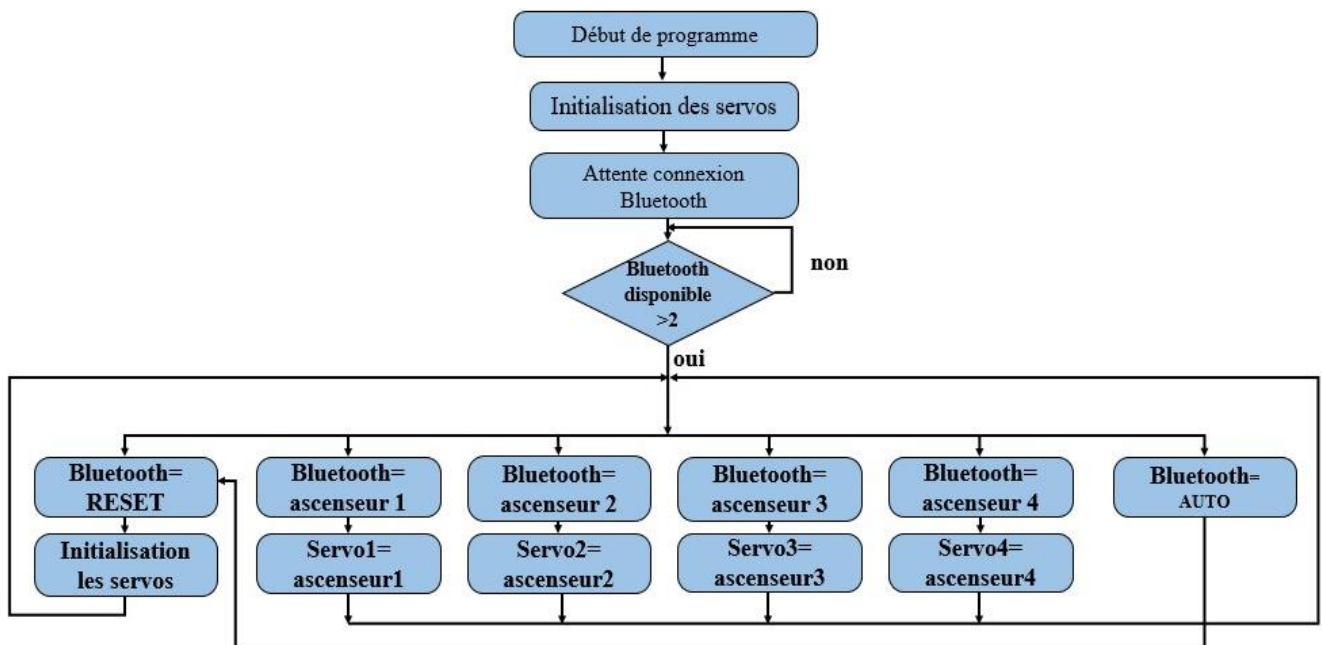


Figure III.21. Organigramme de fonctionnement global.

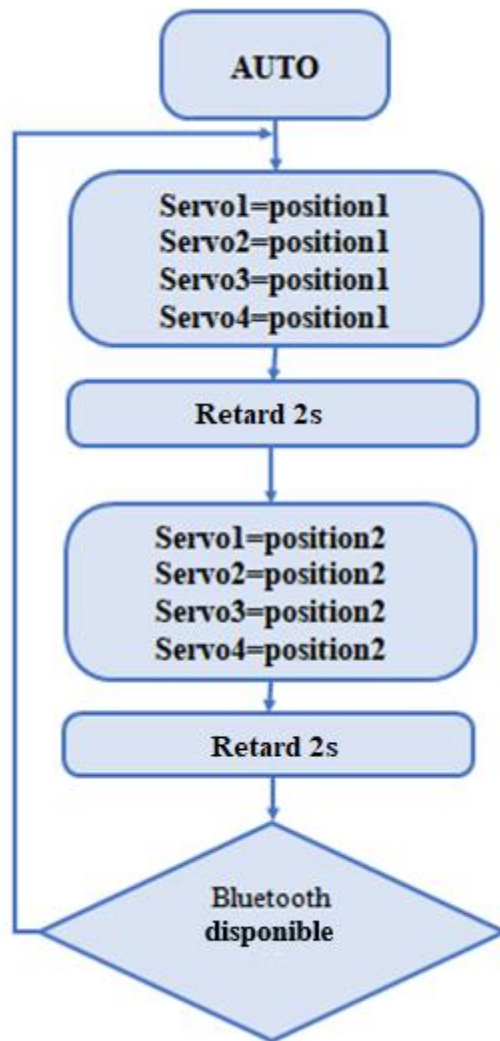


Figure III.22. Organigramme de fonctionnement automatique.

4. Conclusion

Dans ce dernier chapitre, nous avons expliqué toutes les étapes de réalisation de notre bras manipulateur telles que la structure mécanique et le montage électrique, ainsi que le programme Android qui commande le bras manipulateur à distance et le programme de la carte Arduino sous forme d'un organigramme pour bien comprendre le fonctionnement du bras.

Conclusion générale

Ce mémoire présente notre travail qui concerne la réalisation d'un bras manipulateur à base d'une carte Arduino. Le système actionneur utilisé pour notre bras est composé de quatre servomoteurs chacun a un angle de 180° et une puissance différente qui est fixée selon le mouvement et le poids de chaque articulation. Le circuit de commande du système est basé entièrement sur une carte programmable de type Arduino, l'avantage de cette dernière est la minimisation de la quantité de composants électroniques utilisés et par conséquent la minimisation de la surface des cartes électroniques, nous avons utilisé la carte Arduino UNO ayant des caractéristiques (le nombre de pin et la capacité de mémoire pour le stockage de programme...etc) convenant bien aux objectifs de notre réalisation. La programmation de la carte Arduino est effectuée par le logiciel IDE Arduino qui est un mélange entre le C et le C++. Nous avons opté également pour une technique de communication sans fil pour contrôler le bras manipulateur, qui est la technologie Bluetooth, ensuite nous avons développé une application Android « Bras manipulateur » qui permet à l'utilisateur d'avoir les diverses commandes pour déplacer des objets dans un espace de travail de demi sphère à l'aide de ces commandes envoyées depuis le Smartphone Android au bras qui possède un récepteur Bluetooth qui reçoit les commandes et les transmet au circuit du microcontrôleur Arduino qui transmet le signal aux circuits intégrés de commande de chaque servomoteur pour fonctionner.

En cours de la conception et du développement du projet, nous avons toujours essayé de mettre en place toute la documentation disponible ainsi que l'infrastructure nécessaire, par conséquent de nombreuses connaissances théoriques ont été transférées à la pratique et il a été garanti que cela convient aux objectifs du projet.

Ce travail nous a été d'un grand apport pour notre expérience car il nous a permis d'améliorer nos connaissances à la fois dans le langage de programmation et dans le domaine de robotique, il nous a aussi donné l'opportunité de toucher de près aux difficultés lors de la résolution d'un problème qu'on qualifie de complexe et d'exploiter nos connaissances théoriques dans le domaine pratique.

Bien que le bras manipulateur fabriqué dans ce projet soit de qualité prototype, il peut être amélioré davantage. En outre, le secteur des bras robotiques, ouvert au développement, conservera son importance à l'avenir.

En perspective, pour aller plus loin dans ce domaine :

- Conception d'un nouveau modèle de bras à six ou à sept degrés de liberté
- Utiliser des moteurs à courant continu, avec asservissement en vitesse et en position
- Utiliser les équations de rotation.
- Choix de composants électroniques plus adaptés
- Ajouter une caméra pour la visualisation.
- Utiliser l'apprentissage par intelligence artificielle

Références bibliographiques

- [1] RICHA Brahim et BOUYEKHF Mohamed Redah, « Etude et réalisation d'un bras robot à 2DDL », mémoire master, Université KHEMIS MILIANA (2017).
- [2] AIT DAHMANE Kahina et AIT ZIANE Meziane, « Conception et Réalisation d'un Bras Manipulateur Commandé par API », mémoire master, Université KHEMIS MILIANA (2015).
- [3] RICHARD M. Murray et ZEXIANG Li, «A Mathematical Introduction to Robotic Manipulation», (1994).
- [4] <https://www.gridam.com/2018/06/mars-le-rover-opportunity-de-la-nasa-ne-repond-plus>, (20-06-2019).
- [5] <https://www.istockphoto.com/be/vectoriel/bras-robotis%C3%A9-bras-m%C3%A9canique-manipulateur> (05-06-2019).
- [6] E. Dombre, « Analyse et Modélisation des Robots Manipulateurs », (22-01-2002).
- [7] BELAID Issam Eddine et AZZOUZ Youcef, « étude et réalisation d'un bras manipulateur de type r-p », Université TLEMCEN (2018).
- [8] J.L. Boimond, « Robotique », « <http://laris.univ-angers.fr/fr/vie-unite/liste-du-personnel-1/jean-louis.html> » Université ANGERS.
- [9] <https://www.indevagroup.com/solutions/handling-reels-paper-plastic> (06-06-2019).
- [10] <http://www.machine-outil.com/actualites/t157/a4676-robot-de-soudage-a-l-arc-kuka-tolexpo-2011> (08-06-2019).
- [11] SMAHI Mohammed, « Commande d'un bras manipulateur via une application Androïde », mémoire master, Université TLEMCEN (2018).
- [12] <https://www.amazon.de/Makeblock-Programmierbarer-Ultimate-Roboterbausatz> (14-06-2019).
- [13] <https://www.businesswire.com/news/home/20161208005422/en/ITRI-DemonstratesCutting-Edge-Technologies-CES-2017-Booth> (16-06-2019).
- [14] RAZIBAOUENE Ihab et BENYOUCEF Samir, « Automatisation d'un processus d'emballage à l'aide d'un robot série à quatre degrés de liberté », mémoire master, Université BOUMERDES (2017).
- [15] TAZEROUT Yacine, « Commande d'un bras robotise avec une application androïde », mémoire master, Université LORRAINE (2018).

Références bibliographiques

- [16] Simon Landrault et Hippolyte Weisslinger,"Arduino-Premiers pas en informatique embarquée", (19-06-2014).
- [17] <https://ardwinner.jimdo.com/arduino> , (12-07-2019).
- [18] Astalaseven, Eskimon et olyte,"Arduino pour bien commencer en électronique et en programmation », (04/08/2012).
- [19] Jean-Noël « Livret Arduino en français », Centre de Ressources Art Sensitif, novembre 2006.
- [20] DJAFRI Menad et CHELOUCHE Djalal, « Etude et Réalisation d'une Carte Arduino », mémoire master, Université Bejaia (2016).
- [21] Erik Bartmann, « LE GRAND LIVRE D'ARDUINO », (02-01-2014).
- [22] <https://fr.ubergizmo.com/2012/04/05/lhistoire-secrete-du-bluetooth.html> ,(06-08-2019).
- [23] <https://knowledge.parcours-performance.com/arduino-bluetooth-hc-05-hc-06> ,(19-08-2019).
- [24] <http://www.martyncurrey.com/hc-05-and-hc-06-zs-040-bluetooth-modules-first-look> , (23-08-2019).
- [25] MEKKI Walid et BELHADJ Mohammed, « Conception et réalisation d'un panneau solaire à base d'Arduino », mémoire master, Universitaire d'Ain-Temouchent (2018).
- [26] SAADI Naim, « Réalisation d'une Maquette Didactique de Robotique et Arduino », mémoire master, Université Bejaia (2017).
- [27] <http://users.polytech.unice.fr/~ferrero/TPelec2/arduino4> ,(26-08-2019).
- [28] LAIB Bachir et SAADANI Fathi, « Commande d'une Barrière Automatique à Base d'Arduino », mémoire master, Université El-Oued (2018).
- [29] MEKHALFIA Toufik et GHADBANE Toufik"Etude et réalisation d'un système de commande à distance des installations électriques pour la domotique",2018.

Résumé :

La robotique fait actuellement l'objet de nombreux développements. Dans le cadre du projet de fin d'études, nous nous intéressons au contrôle et à la réalisation d'un bras manipulateur à 3 degrés de liberté, il s'agit d'une structure mécanique articulée, actionnée par des servomoteurs et contrôlée par la carte Arduino. Nous avons contrôlé le bras du robot à l'aide d'une application Android dans un téléphone Android créé à l'aide du logiciel "MIT APP INVENTOR", qui facilite le contrôle du bras manipulateur.

Mots-clés Bras manipulateur, Arduino, Bluetooth, application Android, microcontrôleur.

Abstract :

Robotics is currently the subject of many developments. As part of the graduation project, we are interested in control and realization of a manipulator arm with 3 degrees freedom, it is an articulated mechanical structure, actuated by servomotors and controlled by the Arduino board. We controlled the robot arm by an Android application in an android phone that we created using the "MIT APP INVENTOR" a software that make it easier for us to control the manipulator arm.

Keywords Arm manipulator, Arduino, Bluetooth, Android application, microcontroller.

ملخص :

الروبوتات هي حاليا موضوع العديد من التطورات. كجزء من مشروع التخرج ، نحن مهتمون بالتحكم في ذراع مناوول مع حرية 3 درجات ، وهو عبارة عن هيكل ميكانيكي مفصل ، يتم تشغيله من قبل المحركات المؤازرة ويتم التحكم فيه بواسطة لوحة Arduino. لقد سيطرنا على ذراع الروبوت من خلال تطبيق Android في هاتف android أنشأناه باستخدام برنامج " MIT APP INVENTOR" الذي يسهل علينا التحكم في ذراع التحكم.

الكلمات المفتاحية الذراع مناوور ، اردوينو ، بلوتوث ، تطبيق أندرويد ، متحكم دقيق.