

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 OULHADJ-BOUIRA



Faculté des Sciences et Sciences Appliquées  
Département GENIE ELECTRIQUE

**Mémoire de fin d'étude**

Présenté par :

**SAIDANI SIHAM  
MECHEKAK CYLIA**

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

Filière : ELECTRONIQUE

Option : ELECTRONIQUE DES SYSTEMES EMBARQUER

**Thème :**

**Réalisation et commande d'un robot mobile à deux roues auto balancé.**

**Devant le jury composé de :**

<i>Mr. Asradj Zahir</i>	MCA	UAMOB	Président
<i>Mr. Haroun Smail</i>	MCB	UAMOB	Encadreur
<i>Mr. Ait Abass Hamou</i>	MCB	UAMOB	Cau encadreur
<i>Mr. Bouzida Ahcene</i>	MCA	UAMOB	Examineur

Année Universitaire 2019/2020

## REMERCIEMENT

---

# REMERCIEMENT

*Nous remercions avant tout DIEU le tout puissant pour la volonté, la santé, et la patience qu'il nous a donné pour terminer ce travail.*

*En seconde lieu nous tenons à adresser nos remerciements à notre encadreur Dr. Haroun Smail pour son aide et ses Conseils pendant toute la période de notre travail.*

*Nous exprimons également nos remerciements a Mr Ait Abass Hamou et aux membres de jury Mr Asradj Zahir et Mr Bouzida Ahcene qui ont accepté d'examiner notre travail,*

*Nos sincères remerciements s'adressent enfin à tout le personnel de l'ETS Bensalem service après-vente Condor de Bouira, en particulier Djadi Slimane, pour son aide durant ce projet de fin d'étude.*

## Dédicace

---

# **DÉDICACE DE CYLLIA**

### *JE DÉDIE CE TRAVAIL....*

*A mes très chers parents qui ont toujours étaient là, pour moi et qui m'ont donné l'amour, la confiance, soutiens et sacrifices. Que Dieu vous préserve et vous accorde santé, longue vie et bonheur.*

*A mes chers frères Mourad et Yacine et mes sœurs Samira, Lynda, Nadia, Mames, pour leur soutien moral et leurs conseils précieuses tout au long de mes études.*

*A mes neveux Ali, Kiki et Lounes et mes nièces Dina, Racha et Eline.*

*A mon binôme Saidani Siham et sa famille et à ma chère amie Leila en témoignage de l'amitié qui nous uni et les souvenirs de tous les moments que nous avons passés ensemble.*

## Dédicace

---

# **DÉDICACE DE SIHAM**

### *JE DÉDIE CE TRAVAIL....*

*A mes très chers parents qui ont toujours étaient là pour moi et qui m'ont donné l'amour, la confiance, soutiens et sacrifices. Que Dieu vous préserve et vous accorde santé, longue vie et bonheur.*

*A mon chère frère Ziane en témoignage de l'attachement, de l'amour et de l'affection que je porte pour lui. Pour son soutien moral et ses conseils précieuses tout au long de mes études*

*A mes tantes Na3ima, Nassira et Houria et à mes petits cousins Redha et Ghilas ; à mes petites cousines Anias, Darine sans oublier ma chère Lina qui nous a quittés trop tôt que dieu l'accueil dans son vaste paradis.*

*A mon binôme Mechekak Cylia et sa famille et a mes chère amis Tina, kenza, Thavxuxth ,Siham,Massi en témoignage de l'amitié qui nous uni et les souvenirs de tous les moments que nous avons passé ensemble.*

## Listes des Acronymes et Symboles

---

### Acronyme et Symbole

LAAS	<i>Laboratoire d'Analyses d'Architecture des Systèmes a Toulouse (France).</i>
ASIMO	<i>Advanced Step in Innovative Mobility (Etape avancée dans la mobilité innovante)</i>
IDU	<i>In Door Unit</i>
LFD	<i>Loi Fondamentale de Dynamique</i>
L	<i>le Lagrangien</i>
$E_c$	<i>Energie cinétique</i>
$E_p$	<i>Energie potentielle</i>
$E_M$	<i>Energie cinétique du chariot en mouvement</i>
$E_{cm}$	<i>Energie cinétique du pendule</i>
$v_c$	<i>La vitesse de centre de gravité du pendule</i>
$\partial \dot{q}_i$	<i>La <math>i^{\text{ème}}</math> composante du vecteur de coordonnées généralisées</i>
$\partial q_i$	<i>La <math>i^{\text{ème}}</math> composante de la vitesse généralisée</i>
$F_i$	<i>La force généralisée associée a <math>q_i</math></i>
IMU	<i>Inertial Measurement Unit</i>
BMS 3S	<i>Battery Management System (Système de gestion de batterie a 3 cellules)</i>
SDK	<i>Software Development Kit</i>
DMP	<i>Digital Motion Processor</i>
I2C	<i>Inter-Integrated Circuit</i>
PID	<i>Régulateur Proportionnel Intégral et Dérivé</i>
SCL	<i>Serial Clock Line</i>
SDA	<i>Serial Data Line</i>

---

# Table de Matières

---

## Table de Matières

Introduction Générale.....	1
----------------------------	---

### ***CHAPITRE I***

#### ***Généralités sur les robots mobiles***

I.1. Introduction.....	3
I.2. La robotique mobile .....	3
I.3. Historique sur la robotique mobile .....	3
I.4. Définition d'un robot.....	6
I.5. Domaines d'applications des robots.....	6
I.5.1. Exemple d'application de la robotique .....	7
a. Les robots industriels .....	7
b. Les robots chirurgicaux.....	7
c. Les robots domestiques .....	7
d. Les robots mobiles .....	7
I.5.2. Les différentes catégories de robots .....	8
I.5.2.1. Les Robots Fixés .....	8
a. Robot Cartésien.....	8
b. Robot Cylindrique.....	9
c. Robot Sphérique.....	9
d. Robot SCARA .....	9
e. Robot Parallèle.....	9
I.5.2.2. Les robots mobiles .....	10
I.5.3. Les classes de robots à roues (mobiles) .....	11
a. Robot unicycle .....	11
b. Robot tricycle .....	11
c. Robot voiture .....	11
d. Robot omnidirectionnel.....	12
I.6. Le robot mobile auto-balancé.....	13
I.7. Principe de robot auto balance (pendules inversés) .....	13
I.7.1. Le pendule inversé sur deux roues .....	14
I.8. Conclusion .....	16

---

## **CHAPITRE II**

### ***Modélisations d'un robot auto balancé***

II.1. Introduction .....	17
II. 2. Principe de fonctionnement du pendule inversé .....	17
II. 3. Modélisation mathématique du pendule inversé .....	17
II. 4. Modélisation du système chariot-pendule à deux degrés de liberté .....	19
II. 4.1. Énergie cinétique du système en mouvement.....	19
II. 4.2. Énergie potentielle du système .....	20
II. 5. Modélisation dynamique .....	20
II. 5.1. Équation de Lagrange .....	20
II. 5.2. L'équation de Lagrange pour le degré de liberté $q(t) = x(t)$ . .....	21
II. 5.3. L'équation de Lagrange pour le degré de liberté $q(t) = v(t)$ .....	21
II. 6. La régulation PID.....	22
II. 7. Implémentations de PID numérique sur Arduino .....	22
II. 7.1. Le régulateur proportionnel (P : première règle) .....	22
II. 7.2. Le régulateur proportionnel intégral (PI : première et seconde règle) .....	23
II. 7.3. Le régulateur proportionnel dérivé (PD : première et troisième règle) .....	23
II. 7.4. Le régulateur proportionnel intégral dérivé (PID : première, seconde et troisième règle). .....	23
II. 7.5. Régler les coefficients d'un PID .....	24
II. 8. Conclusion .....	24

## **CHAPITRE III**

### ***Conception matérielle du robot auto balancé***

III. 1. Introduction .....	25
III. 2. Schéma synoptique et fonctionnement du robot .....	25
III. 3. Unité de commande .....	26
III. 3.1. La carte Arduino .....	27
III. 3.2. La carte Arduino NANO .....	27
III. 4. L'unité de mesure inertielle MPU6050 .....	29
III. 5. Unité de puissance .....	30

---

III. 5.1. Les moteurs pas à pas.....	30
III.5.2. Le pilote (Driver) A4988.....	32
III. 6. Module Bluetooth HC-05 .....	34
III. 7. Batterie d'alimentation électrique.....	35
III. 8. Branchement globale du système .....	35
III. 9. Conclusion .....	35

## **CHAPITRE IV**

### ***Programmation et finalisation du projet***

IV. 1. Introduction.....	36
IV. 2. Organigramme de l'algorithme du système .....	36
IV. 3. Mesure de l'angle d'inclinaison .....	37
IV. 3.1. Le raccordement du MPU6050 avec l'Arduino .....	37
IV. 3.2. Mesure d'angle par l'accéléromètre.....	38
IV. 3.3. Mesure d'angle par le gyroscope .....	40
IV. 3.4. Mesure d'angle par fusion de donnée .....	41
IV. 4. La régulation PID.....	42
IV. 5. La réalisation pratique du robot .....	43
IV. 6. Raccordements et soudage .....	44
IV. 7. Réalisation et positionnement de la batterie.....	45
IV. 8. La forme finale du robot.....	46
IV. 9. Conclusion.....	47

<b><i>Conclusion Générale</i></b> .....	48
---	----

<b><i>Bibliographie</i></b> .....	49
-----------------------------------	----

---



# Liste de figure

---

## *Chapitre I*

<b>Figure I.1</b>	La tortue de Grey Walter « machina speculatrix » et une illustration de sa trajectoire pour rejoindre sa niche	3
<b>Figure I.2</b>	À gauche : le robot « Beast » de l'université John Hopkins À droite : le robot Shakey de Stanford	4
<b>Figure I.3</b>	Le robot Hilare du LAAS a été construit en 1977	5
<b>Figure I.4</b>	Gengis développé par Rodney Brooks	5
<b>Figure I.5</b>	Exemples de robots commerciaux ou de recherche	8
<b>Figure I.6</b>	Exemples des différentes classes de robots	10
<b>Figure I.7</b>	Différents types de robots mobiles à roues	12
<b>Figure I.8</b>	robot auto balancé	13
<b>Figure I.9</b>	un pendule inversé constitué d'un chariot de masse M	14
<b>Figure I.10</b>	Le pendule inversé sur deux roues avec son schéma de principe	15
<b>Figure I.11</b>	Photo du Segway	15

## *Chapitre II*

<b>Figure II.1</b>	Exemple sur l'équilibre à base du pendule inversé	17
<b>Figure II.2</b>	Schéma de l'ensemble chariot et pendule inversé	18

## *Chapitre III*

<b>Figure III.1</b>	schéma synoptique et architecture générale de robot	26
<b>Figure III.2</b>	Vue générale et branchement de la carte Arduino NANO	28

## Liste de figure

---

<b>Figure III.3</b>	Une carte GY-521 dotée de la puce MPU6050 et son branchement	29
<b>Figure III.4</b>	Moteur pas à pas	30
<b>Figure III.5</b>	Principe de rotation pas-à-pas du moteur	31
<b>Figure III.6</b>	Séquence à pas complet	32
<b>Figure III.7</b>	Chronogramme du pilotage à pas entier	32
<b>Figure III.8</b>	Le pilote (Driver) A4988 et son branchement	33
<b>Figure III.9</b>	Module Bluetooth HC-05 et son branchement	34
<b>Figure III.10</b>	Schéma électronique globale du robot auto balancé	35

### *Chapitre IV*

<b>Figure IV.1</b>	Organigramme de fonctionnement de robot auto balance	37
<b>Figure IV.2</b>	Mesure des accélérations sur les trois axes X, Y et Z	39
<b>Figure IV.3</b>	Mesure des angles par les données d'accéléromètre	39
<b>Figure IV.4</b>	signaux des vitesses angulaires sur les axes X et Y.	40
<b>Figure IV.5</b>	angles calculés à partir des données gyroscopiques.	40
<b>Figure IV.6</b>	l'angle mesuré par fusion de données	42
<b>Figure IV.7</b>	Composant et structure du robot réalisé.	44
<b>Figure IV.8</b>	raccordements des composants sur la plaque perforée	45
<b>Figure IV.9</b>	Test de la batterie réalisée	46
<b>Figure IV.10</b>	photo finale de robot réalisé	47

## Introduction générale

De nos jours, la robotique, les systèmes de commande électronique et numérique ont connu une évolution très importante, ils occupent une place majeure dans notre quotidien. On les rencontre dans différentes applications domestique, industriel, médicale, et même militaire.

En fait, la réalisation d'un système de commande numérique couvre plusieurs aspects : la mécanique, l'informatique, l'électronique, l'automatique et bien d'autre. Mais avec l'évolution des systèmes de commande, de communication et des technologies de l'information, réaliser un système de commande numérique est devenu facile en se basant sur un minimum de composant et la programmation informatique en utilisant les microcontrôleurs ou les cartes Arduino [1].

Le robot auto balancé est un objet ou un véhicule (robot mobile) miniaturisé à deux roues pouvant se déplacer horizontalement et se maintenir à la verticale même à l'arrêt sans action de l'utilisateur, Ce type de robot semble défier les lois de l'équilibre [2]. On en trouve des exemples d'applications dans de nombreux domaines mais plus particulièrement dans les transports. Ainsi, les trains à grandes vitesses, les Segway, le système de stabilisation d'une fusée ou encore d'un missile (lors du décollage ou du début de la propulsion) représentent tous plus ou moins des pendules inversés [3]. Enfin, il existe d'autres éléments que nous retrouvons dans la vie de tous les jours, tel que le système tonique postural (c'est à dire le système qui contrôle l'équilibre de l'être humain). Plus généralement nous pouvons généraliser ce modèle aux corps qui subissent un état d'équilibre instable [4].

Les robots mobiles à deux roues dont le centre de gravité est situé au-dessus de l'axe des roues reposent sur le concept du pendule inversé, sont des systèmes dynamiquement stables mais statiquement instables. Pour défier les lois de l'équilibre et maintenir ce genre de robot stable à la verticale, diverses théories de la physique et de contrôle sont impliquées [5].

L'objectif principal de ce projet de fin d'étude, est de concevoir, réaliser et commander un robot mobile auto balancé. Ce robot sera basé sur carte à microcontrôleur ARDUINO NANO comme unité de commande, et une unité de mesure inertielle (IMU) MPU6050. Les roues de robot seront actionnées par deux moteurs pas à pas NEMA 17 à travers deux drivers A4988. Le robot sera doté d'un module de communication Bluetooth afin de pouvoir le contrôler à distance.

L'algorithme à implémenter sur la carte ARDUINO NANO récupère les données d'accélération et de gyroscope à partir de l'unité de mesure inertielle, puis réalise une combinaison de deux mesures afin d'estimer l'angle d'inclinaison de robot mobile. Par la suite un régulateur PID est utilisé pour corriger l'erreur entre le point de consigne souhaité

## Introduction générale

---

(Position verticale) et l'angle d'inclinaison réel du robot en commandant la rotation et la vitesse des deux roues pour équilibrer le robot. Le programme doit aussi avoir la capacité de récupérer les commandes venant du module Bluetooth pour contrôler le robot en termes de mouvements à réaliser (avancer, reculer, tourner...etc.).

Ce manuscrit intitulé « Réalisation et commande d'un robot mobile à deux roues auto balancé » récapitule l'essentiel de notre projet de fin d'étude, il est organisé en quatre chapitres comme suit :

Dans le premier chapitre, nous présentons une vue générale sur les robots, leurs classifications, leurs domaines d'application, et leurs différentes structures et constitutions. Dans ce chapitre, nous allons aussi présenter quelques notions fondamentales et définitions liées à la robotique mobile en focalisant principalement sur les robots auto balancés et le fonctionnement d'un pendule inversé.

Le deuxième chapitre sera consacré à la modélisation mathématique du robot auto-balancé et la présentation des équations de mouvement du robot, ainsi que la commande par un régulateur PID.

Dans le troisième chapitre, on présente la configuration et la conception matérielle de notre robot mobile auto balancé à réaliser en termes de choix et description des composants et circuits électroniques.

Le dernier chapitre, concerne la conception logiciel du système à réaliser, de ce fait ce chapitre traitera les différents programmes mis au point pour que le robot mobile auto balancé fonctionne correctement. La réalisation pratique et finale de notre robot est aussi abordée dans ce chapitre.

A la fin, on termine notre travail avec une conclusion générale et des perspectives pour ouvrir une porte à de futures recherches afin d'améliorer le système réalisé dans ce travail.



CHAPITER 01  
**Généralités sur les robots  
mobile**

## I.1. Introduction

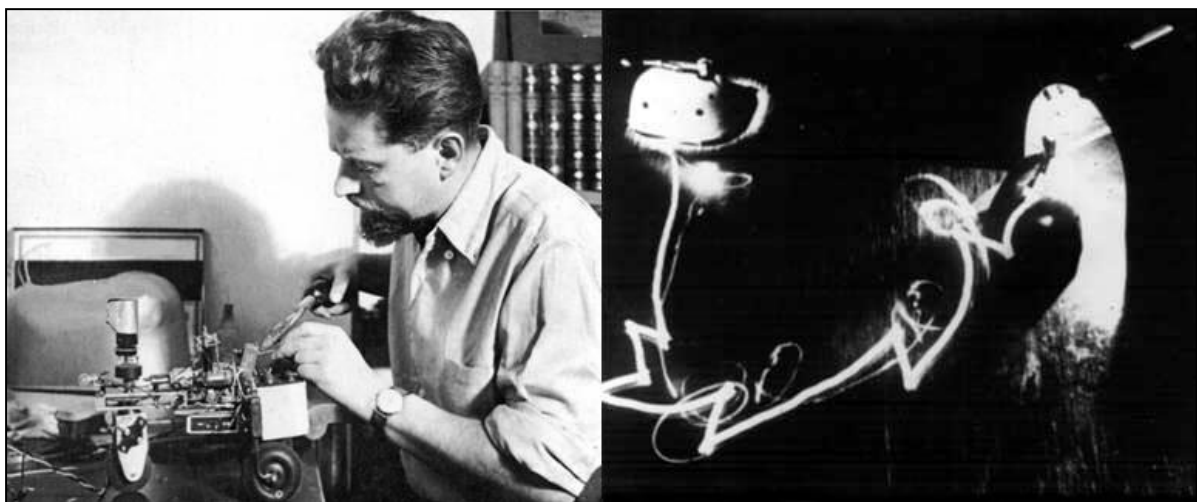
Dans ce premier chapitre, nous exposons le contexte général dans lequel s'inscrit ce travail de fin d'étude. Nous présentons un état de l'art de la robotique mobile. D'abord nous allons présenter une définition sur les robots ainsi les robots mobiles. Ensuite on donne l'objectif principale des robots mobiles, puis les diverses catégories des robots ainsi leurs classes ou types. On aborde en suite les caractéristiques et les différents domaines d'application des robots mobiles. A la fin de ce chapitre nous allons focaliser sur les robots auto balancé et le principe de pendule inversé.

## I.2. La robotique mobile

La robotique mobile est la discipline qui s'intéresse à la conception de robots mobiles. Elle s'appuie sur d'autres disciplines comme l'automatique, le traitement du signal, la mécanique, l'informatique et l'électronique [6].

## I.3. Historique sur la robotique mobile

Le terme de « robot » apparaît pour la première fois dans une pièce de Karel Capek en 1920 : Rossum's Universal Robots. Il vient du tchèque « robota » (qui signifie à peu près « servitude ») et présente une vision des robots comme serviteurs dociles et efficaces pour réaliser les tâches pénibles, mais qui déjà vont se rebeller contre leurs créateurs. La Tortue, construite par Grey Walter dans les années 1950 (Figure I.1), est l'un des premiers robots mobiles autonomes [7].

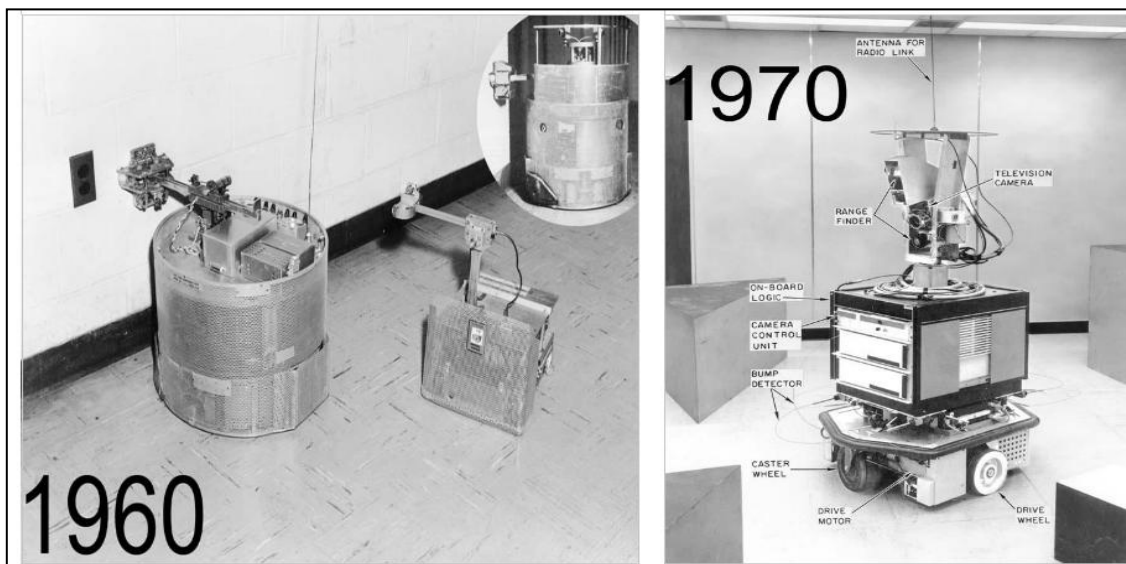


**Figure I.1** : La tortue de Grey Walter « machina speculatrix » et une illustration de sa trajectoire pour rejoindre sa niche.

Grey Walter n'utilise que quelques composants analogiques, dont des tubes à vide, mais son robot est capable de se diriger vers une lumière qui marque un but, de s'arrêter face à des obstacles et de recharger ses batteries lorsqu'il arrive dans sa niche.

Toutes ces fonctions sont réalisées dans un environnement entièrement préparé, mais restent des fonctions de base qui sont toujours des sujets de recherche et de développement technologiques pour les rendre de plus en plus génériques et robustes.

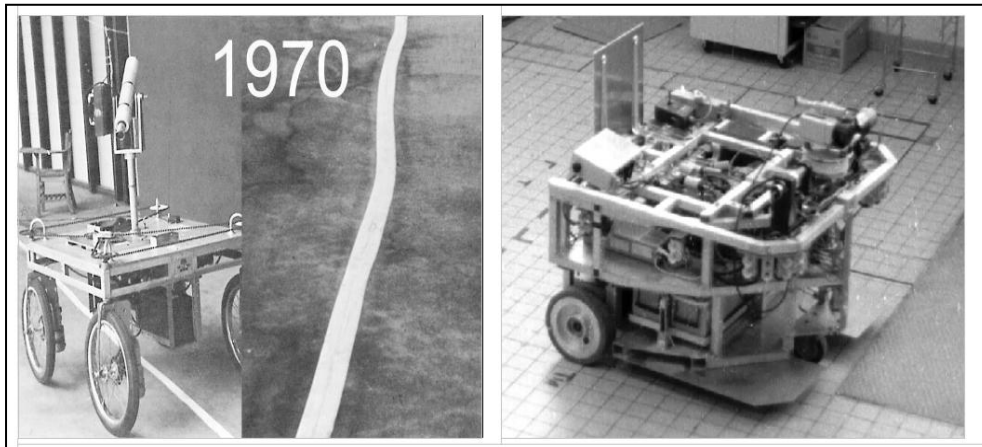
Dans les années 1960, les recherches en électronique vont conduire, avec l'apparition du transistor, à des robots plus complexes, mais qui vont réaliser des tâches similaires. Ainsi, le robot « Beast » (Figure I.2) de l'université John Hopkins est capable de se déplacer au centre des couloirs en utilisant des capteurs ultrason, de chercher des prises électriques (noires sur des murs blancs) en utilisant des photodiodes et de s'y recharger [7].



**Figure I.2 :** À gauche : le robot « Beast » de l'université John Hopkins  
À droite : le robot Shakey de Stanford.

Les premiers liens entre la recherche en intelligence artificielle et la robotique apparaissent à Stanford en 1969 avec Shakey. Ce robot utilise des télémètres à ultrason et une caméra et sert de plate-forme pour la recherche en intelligence artificielle, qui à l'époque travaille essentiellement sur des approches symboliques de la planification. La perception de l'environnement, qui à l'époque est considérée comme un problème séparé, voire secondaire, se révèle particulièrement complexe et conduit là aussi à de fortes contraintes sur l'environnement. Ces développements se poursuivent avec le Stanford Cart dans la fin des années 1970, avec notamment les premières utilisations de la stéréo-vision pour la détection

d'obstacles et la modélisation de l'environnement. En France, le robot Hilare est le premier robot construit au LAAS à Toulouse (Figure I.3) [7].



**Figure I.3 :** Le robot Hilare du LAAS a été construit en 1977.

Une étape importante est à signaler au début des années 1990 avec la mise en avant de la robotique réactive, représentée notamment par Rodney Brooks. Cette nouvelle approche de la robotique, qui met la perception au centre de la problématique, a permis de passer de gros robots très lents à de petits robots (Figure 4) , beaucoup plus réactifs et adaptés à leur environnement. Ces robots n'utilisent pas ou peu de modélisation du monde, problématique qui s'est révélée être extrêmement complexe [7].



**Figure 1.4 :** Genghis, développé par Rodney Brooks

Ces développements ont continué et l'arrivée sur le marché depuis les années 1990 de plates-formes intégrées telles que le pionnier de la société Mobile Robots, a permis à de très nombreux laboratoires de travailler sur la robotique mobile et à conduit à une explosion de la diversité des thèmes de recherche. Ainsi, même si les problèmes de déplacement dans l'espace



et de modélisation de l'environnement restent difficiles et cruciaux, des laboratoires ont pu par exemple travailler sur des approches multi robot, la problématique de l'apprentissage ou sur les problèmes d'interactions entre les hommes et les robots [7].

#### **I.4. Définition d'un robot**

Un robot est un dispositif mécatronique (alliant mécanique, électronique et informatique) accomplissant automatiquement soit des tâches qui sont généralement dangereuses, pénibles, répétitives ou impossibles pour les humains, soit des tâches plus simples mais en les réalisant mieux que ce que ferait un être humain.

Les robots les plus évolués sont capables de se déplacer et de se recharger par eux-mêmes, à l'image du robot ASIMO fabriqué par Honda ou du robot NAO fabriqué par Aldebaran Robotics. En dépit de leur coût élevé à l'époque (faute de microprocesseurs puissants produits en masse), les robots se sont imposés dès le début des années 1970, pour certaines tâches comme la peinture des carrosseries automobiles, en atmosphère de vapeurs toxiques. Depuis, l'évolution de l'électronique et de l'informatique permet aux robots de réaliser des tâches de plus en plus complexes, avec de plus en plus d'autonomie, et de plus en plus rapidement.

Le terme « robot » est également utilisé pour désigner un dispositif qui n'est pas automatique, pour évoquer la haute technicité du dispositif, à l'image du robot chirurgical DaVinci [8].

#### **1.5. Domaines d'applications des robots**

Aujourd'hui, le marché commercial de la robotique mobile est toujours relativement restreint en dehors des robots aspirateurs vendus à plusieurs millions d'exemplaires. Cependant, il existe de nombreuses perspectives de développement qui en feront probablement un domaine important dans le futur. Les applications des robots peuvent se trouver dans de nombreuses activités « ennuyeuses, salissantes ou dangereuses » (3 D en anglais pour dull, dirty, dangerous), mais également pour des applications ludiques ou de service, comme l'assistance aux personnes âgées ou handicapées [5].

### **I.5.1. Exemples d'application de la robotique**

Parmi les domaines d'application possibles de la robotique, on peut citer :

- La robotique de service (hôpital, bureaux, maison) ;
- La robotique de loisir (jouets, robot « compagnon ») ;
- La robotique industrielle ou agricole (entrepôts logistiques, récolte de productions agricoles, mines) ;
- La robotique en environnement dangereux (spatial, industriel, militaire, catastrophes naturelles).

#### **a. Les robots industriels**

Les robots industriels sont les premiers à avoir été produits en grand nombre surtout pour souder, démolir, nettoyer et emballer... On les utilise surtout dans les industries, dans les travaux à la chaîne ou très précis. Ils se trouvent plus particulièrement sur les chaînes de montage, et le plus souvent dans l'industrie automobile [5].

#### **b. Les robots chirurgicaux :**

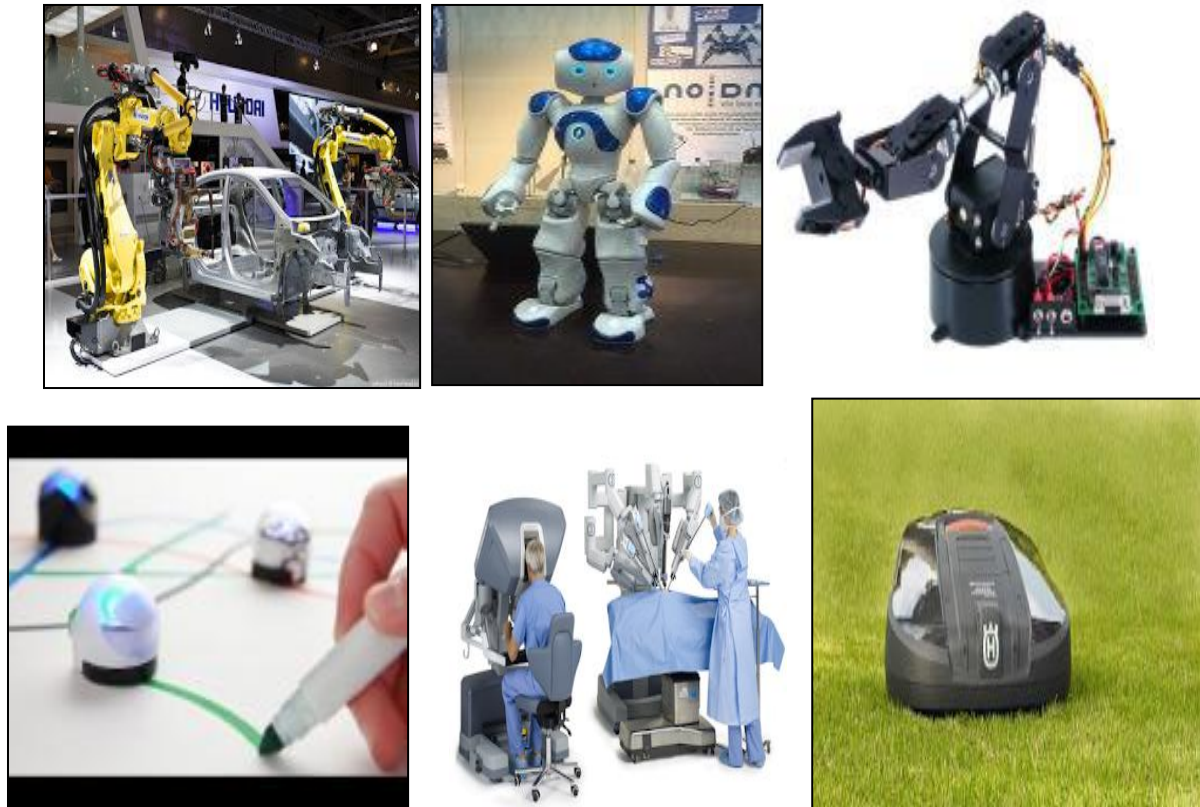
Ils sont en fait une autre branche de robots industriels. Ils fonctionnent soit par bras mécaniques reliés à un ordinateur, soit par bras articulés dirigés par le chirurgien et dont les mouvements sont reproduits en même temps [5].

#### **c. Les robots domestiques**

Ils peuvent accomplir de multiples tâches ou simplement nous divertir. Plusieurs sociétés ont créé une tondeuse à gazon automatique qui sur un périmètre donné peut éviter les obstacles tout en coupant le gazon [5].

#### **d. Les robots mobiles**

On regroupe sous l'appellation ' robots mobiles l'ensemble des robots à base mobile, par opposition notamment aux robots manipulateurs. Un robot mobile est celui qui peut se déplacer dans son environnement de façon indépendante. Pour ce faire, le robot doit pouvoir naviguer, et la portée et la précision de navigation requise varie en fonction de la taille du robot et du type de sa tâche [5].



**Figure I.5:** Exemples d'applications de la robotique.

## I.5.2. Les différentes catégories de robots

### I.5.2.1. Les Robots Fixés

Les robots fixes sont principalement des manipulateurs robotiques industriels qui fonctionnent dans des environnements bien définis adaptés aux robots. Les robots industriels effectuent des tâches répétitives spécifiques telles que des pièces à souder ou à peindre dans des Usines de fabrication automobile. Avec l'amélioration des capteurs et des dispositifs d'interaction Homme-robot, des manipulateurs robotiques sont de plus en plus utilisés dans un environnement moins contrôlé comme la haute précision en chirurgie [9].

Il existe différentes structures de robots fixés :

#### a. Robot Cartésien

Les robots cartésiens sont utilisés dans de nombreux processus industriels et notamment dans l'industrie plastique pour des opérations de décharge de presses à injection [9].

## **b. Robot Cylindrique**

Robot manipulateur dont le porteur de robot possède une articulation toroïde et deux articulations prismatiques, ou deux articulations toroïdes et une articulation prismatique, qui permettent d'effectuer des translations et des rotations selon des axes de mouvement définis par un système de coordonnées cylindriques, donnant ainsi la forme géométrique d'un cylindre à l'enveloppe de travail [9].

## **c. Robot Sphérique**

Un robot sphérique, également appelé robot mobile sphérique, est un robot mobile à la forme externe sphérique. Un robot sphérique est généralement constitué d'une coque sphérique servant de corps au robot et d'une unité de pilotage interne (IDU) permettant au robot de se déplacer. Les robots mobiles sphériques se déplacent généralement en roulant sur des surfaces. La source d'alimentation de ces robots est principalement une batterie située à l'intérieur du robot, mais il existe des robots sphériques qui utilisent des cellules solaires [9].

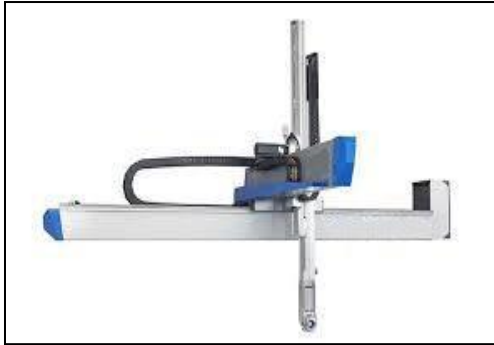
## **d. Robot SCARA**

Un des grands classiques de la robotique, le robot SCARA signifie « Selective Compliant Assembly Robot Arm », ce qui peut se traduire par Bras robotisé d'assemblage à mobilité sélective. C'est un robot d'origine japonaise aujourd'hui, la majeure partie des fabricants de robots industriels, si ce n'est l'intégralité, propose le SCARA dans leur gamme.

Son espace de travail est cylindrique. Il est généralement choisi pour sa grande précision, sa rapidité d'action, son faible encombrement et son prix. Le SCARA est généralement utilisé pour faire de la palettisation, du chargement ou du positionnement. Il est très souvent utilisé dans l'industrie agroalimentaire et l'industrie pharmaceutique [9].

## **e. Robot Parallèle**

Un robot parallèle est un mécanisme dont l'architecture lui confère des propriétés remarquables. Mécanisme en chaîne cinématique fermée dont l'organe terminal est relié à la base par plusieurs chaînes cinématiques indépendantes [9].



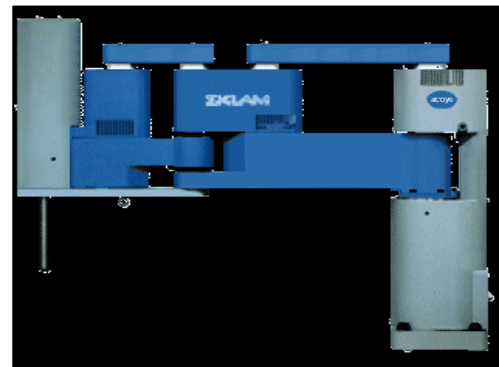
a . Robot Cartésien Toshiba



b .Robot Seiko



c .Robot FANUC



d. Robot Sankyo



e. Robot COMAU

**Figure I.6 :** Exemples de différents robots fixés.

### I.5.2.2. Les robots mobiles

Un robot mobile se caractérise par l'absence de lien mécanique avec un objet de référence, ce qui ouvre magistralement la porte aux applications potentielles. La liberté de mouvements lui confère une autonomie qui lui permet de trouver d'autres utilisations que manufacturières [9].

Un robot mobile consiste à réaliser un mouvement en reliant un point source à un point destination. L'exécution de cette tâche avec un certain degré d'autonomie nécessite l'utilisation d'un ensemble de ressources et d'une structure assurant une coopération efficace entre elles. Sur le terrain, plusieurs structures sont utilisées, le choix d'une structure est directement lié à la nature de l'environnement dans lequel le robot doit évoluer, et de la nature de sa tâche ainsi que des performances recherchées. Sachant que celles-ci dépendent à la fois de la qualité des ressources et de leur gestion. Ayant pour mission de concevoir un robot dans un cadre bien défini ; qui est la compétition de robotique Eurobot. La tâche globale du robot a été bien déterminée ; le robot devait jouer au golf en respectant le cahier des charges élaboré par les organisateurs. Ce cahier des charges définit l'environnement dans lequel se déroulent les matches, les règles du jeu et les dimensions des robots [9].

### **I.5.3. Classes de robots à roues (mobiles)**

#### **a. Robot unicycle**

Un robot mobile de type unicycle est actionné par deux roues indépendantes, il possède éventuellement des roues folles pour assurer sa stabilité. Son centre de rotation est situé sur l'axe reliant les deux roues motrices. C'est un robot non-holonyme, en effet il est impossible de le déplacer dans une direction perpendiculaire aux roues de locomotion. Sa commande peut être très simple, il est en effet assez facile de le déplacer d'un point à un autre par une suite de rotations simples et de lignes droites [10].

#### **b. robot tricycle**

Un robot mobile de type tricycle est constitué de deux roues fixes placées sur un même axe et d'une roue centrée orientable placée sur l'axe longitudinal. Le mouvement du robot est donné par la survitesses des deux roues fixes et par l'orientation de la roue orientable. Son centre de rotation est situé à l'intersection de l'axe contenant les roues fixes et de l'axe de la roue orientable. C'est un robot non-holonyme. En effet, il est impossible de le déplacer dans une direction perpendiculaire aux roues fixes. Sa commande est plus compliquée. Il est en général impossible d'effectuer des rotations simples à cause d'un rayon de braquage limité de la roue orientable [10].

#### **c. Robot voiture**

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

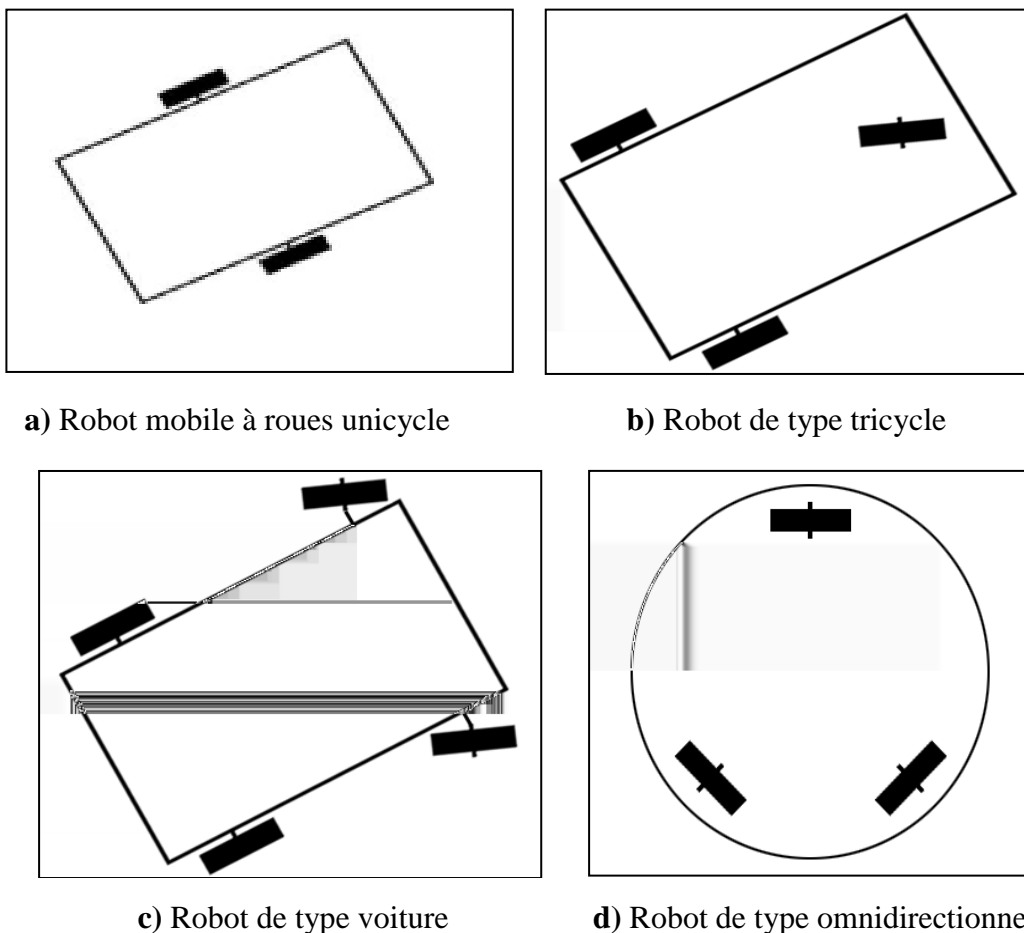
même axe. Le robot mobile de type voiture est cependant plus stable puisqu'il possède un point d'appui supplémentaire.

Toutes les autres propriétés du robot voiture sont identiques au robot tricycle, le deuxième pouvant être ramené au premier en remplaçant les deux roues avant par une seule placée au centre de l'axe, et ceci de manière à laisser le centre de rotation inchangé [10].

#### d. Robot omnidirectionnel

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

L'énorme avantage du robot omnidirectionnel est qu'il est holonome puisqu'il peut se déplacer dans toutes les directions. Mais ceci se fait au dépend d'une complexité mécanique bien plus grande [10].

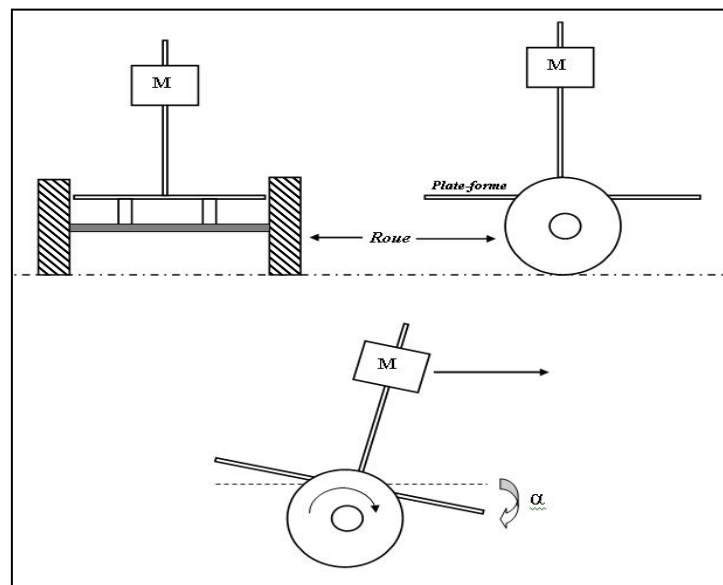


**Figure I.7 :** Différents types de robots mobiles à roues.

## I.6. Les robots mobiles auto-balancés

Le robot auto balancé est un objet ou un véhicule (robot mobile) miniaturisé à deux roues pouvant se déplacer horizontalement et se maintenir à la verticale à l'arrêt sans action de l'utilisateur, Ce type de robot semblent défier les lois de l'équilibre.

Le principe du contrôle de position est basé sur le pendule inversé. L'inclinaison de la plate-forme d'un angle  $\alpha$ , va provoquer le déplacement du robot, de telle sorte que l'angle redevienne nul. D'où le nom de " Robot Auto-Balancé".



**Figure I.8** : Structures de robot auto balancé.

Si le dispositif réagit suffisamment vite, l'angle est en permanence proche de zéro. La plate-forme est donc toujours horizontale et l'utilisateur n'est pas obligé de maintenir son équilibre. Un déplacement du centre de gravité vers l'avant obligera le robot à se déplacer dans le même sens pour maintenir l'équilibre. Ceci est valable dans les deux sens de déplacement [11].

## I.7. Principe de robot auto balancé (pendules inversés)

Un pendule inversé est une tige posée en position d'équilibre instable (Verticale à  $180^\circ$ ) sur une base qui peut être fixe ou mobile. La dynamique non linéaire et Instable de ce système physique a fait de lui un système très intéressant et largement étudié Dans la communauté des automaticiens, ce qui explique l'existence de plusieurs aspects et Formes de pendules inversés On peut trouver divers types de pendules inversés tels que : le pendule inversé simple, le



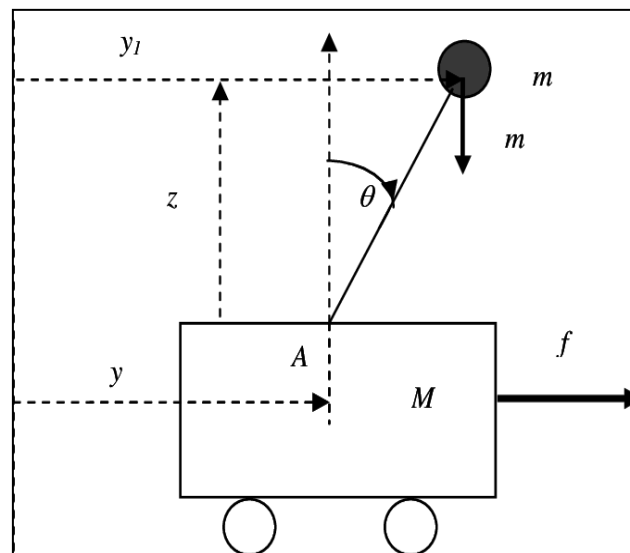
Pendule inversé double, le pendule inversé de Furuta, etc. Dans la suite ces différents types seront présentés. L'étude du pendule inversé a plusieurs intérêts :

- La robotique utilise le principe du pendule inversé, en particulier dans de nouveaux moyens de transports à 2 roues comme le Segway qui permet d'avancer en se penchant en avant.
- On voit apparaître des moyens de locomotion dotés de deux roues montées sur un même axe sur lequel on est en position debout. On accélère en se penchant en avant, et on ralentit en se penchant en arrière. Le système est le même que le pendule inversé [12].

$m$  : Masse du pendule

$M$  : Masse du chariot

$\theta$  : L'angle du pendule

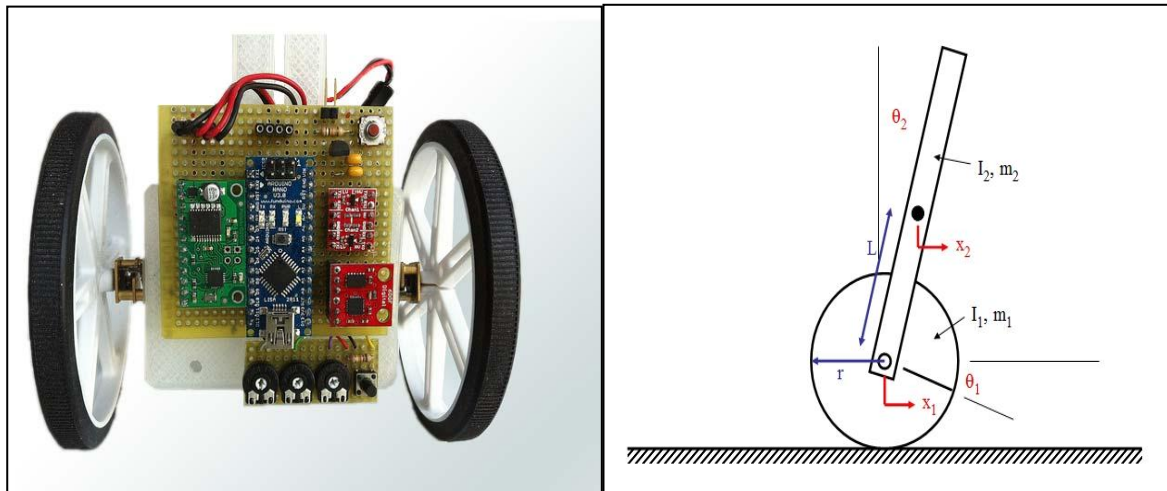


**Figure I.9** : Un pendule inversé constitué d'un chariot de masse  $M$ .

### I.7.1. Le pendule inversé sur deux roues

Le pendule inversé sur deux roues est considéré comme un prototype académique largement étudié dans le domaine de l'automatique. Comme son nom l'indique, il est constitué d'une base mobile (l'essieu et les deux roues) surmontée d'un pendule inversé en rotation libre autour d'un pivot (articulation passive) entre l'essieu et la tige du pendule. L'angle d'inclinaison du pendule par rapport à la verticale. (Figure I.10). Le principe de fonctionnement du système est très simple en théorie : quand le pendule penche vers l'avant, la partie mobile doit le rattraper en effectuant un mouvement vers l'avant et vice-versa. La

difficulté réside dans le réglage de l'intensité de réaction des roues afin d'agir face à l'angle que fait le pendule avec la verticale [10].



**Figure I.10:** Le pendule inversé sur deux roues avec son schéma de principe.

Comme une application réussite du principe de ce pendule, on peut citer le Segway (figure I.11). C'est un moyen de transport électrique et individuel inventé par Dean Kamen [10].



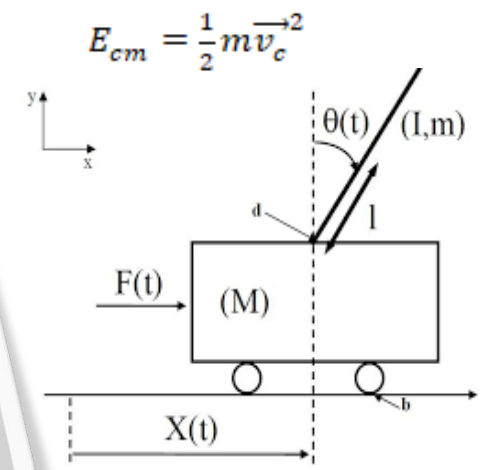
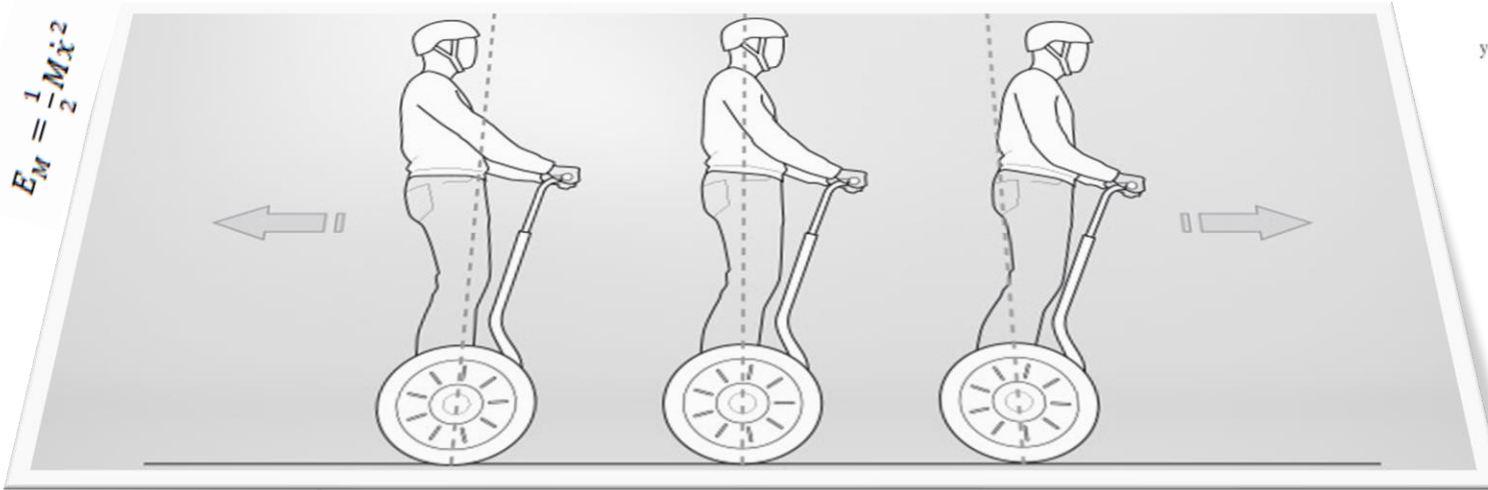
**Figure I.11 :** Photo du Segway

## **I.8. Conclusion**

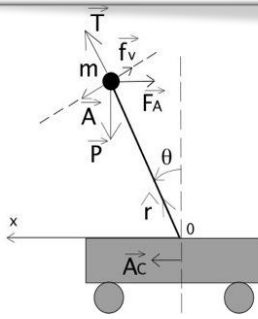
Les robots ont avancé technologiquement. Ils se perfectionnent avec le temps et leurs capacités vont certainement être au-dessus de nos espérances. Nous constatons que les robots auront une place importante dans un futur proche et que les liens entre Homme-Machine se tisseront de plus en plus facilement. Les robots possèdent de nombreux avantages, ils améliorent réellement la vie de l'homme mais il y a encore des tâches que les robots ne peuvent pas faire grâce à leur impossibilité de s'adapter aux différentes situations et il faut bien prendre en compte que de telles avancées devront être contrôlées, car il existe toujours des abus donc Nous devons prendre garde à toujours rester maître de nos inventions. Dans le prochain chapitre, nous allons aborder la modalisation de robot auto balance.

$$\vec{v}_c = \frac{d\vec{r}_c}{dt} = (\dot{x} + l \cos(\theta) \dot{\theta})\vec{i} - l \dot{\theta} \sin \theta \vec{j}$$

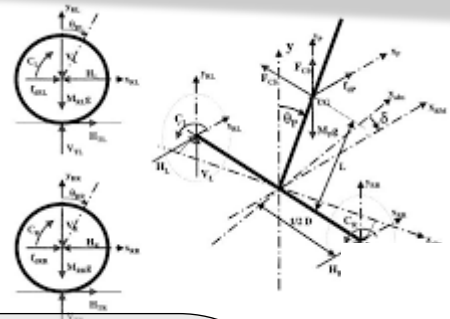
$$E_M = \frac{1}{2} M \dot{x}^2$$



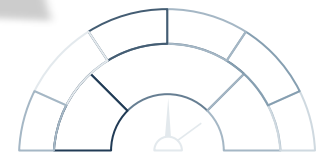
$$E_{cm} = \frac{1}{2} m \vec{v}_c^2$$



$$E_{cm} = \frac{1}{2} m \vec{v}_c^2$$



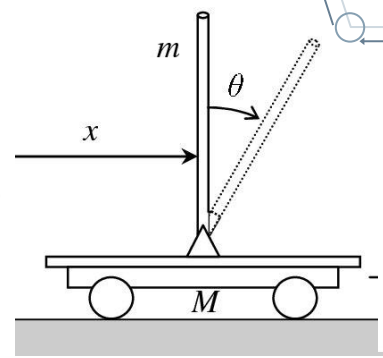
$$E_M = \frac{1}{2} M \dot{x}^2$$



$$E_{cm} = \frac{1}{2} m (\dot{x}^2 + 2\dot{x}l \cos(\theta) \dot{\theta} + l^2 \cos^2(\theta) \dot{\theta}^2 + l^2 \sin^2(\theta) \dot{\theta}^2)$$

# CHAPITER 02

## Modélisations d'un robot auto balancé



## II.1. Introduction

Dans ce chapitre nous allons d'abord aborder la présentation de principe de fonctionnement du pendule inversé ainsi sa modélisation mathématique puis, à partir du formalisme d'Euler-Lagrange nous développons son modèle dynamique non linéaire qui pose un problème d'instabilité au point d'équilibre instable  $\theta = 0$  que nous utiliserons pour illustrer les commandes que nous développerons dans ce mémoire.

## II.2. Principe de fonctionnement du pendule inversé

Le principe est simple théoriquement : quand le pendule penche vers l'un des côtés (gauche ou droit), le chariot doit le rattraper en effectuant un mouvement vers ce côté.

Le Problème se pose dans le réglage de l'intensité et la forme de réaction que doit subir le chariot en fonction de l'angle que le pendule fait avec la verticale. L'ensemble du chariot-pendule à deux degrés de liberté qui sont représentés par deux coordonnées généralisées, pour le déplacement horizontal du chariot, pour la rotation du pendule. La direction positive d'est le sens à droite en mètre et celui de l'angle est le sens des aiguilles d'une montre en radian [13].

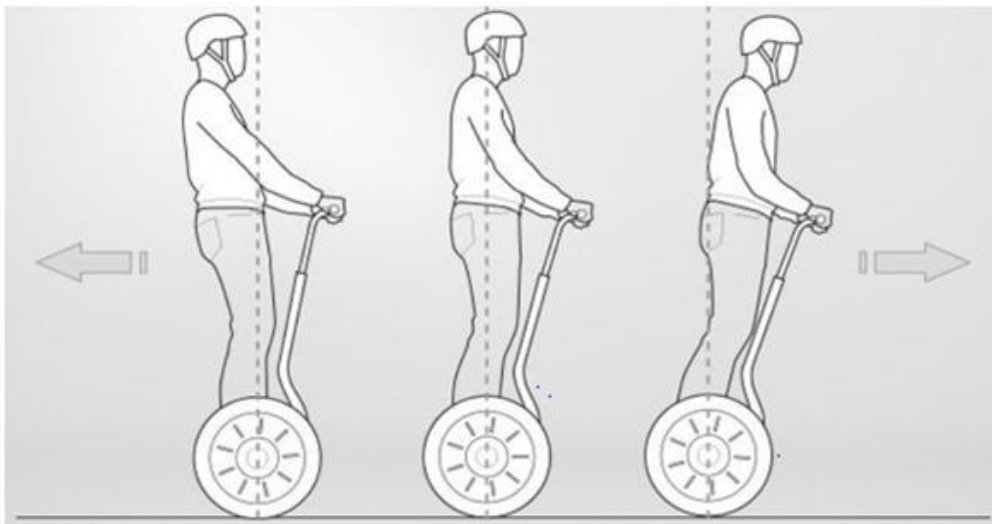


Figure II.1 : Exemple sur l'équilibre à base du pendule inversé

## II.3. Modélisation mathématique du pendule inversé

Le système du pendule inversé est un exemple couramment utilisé dans les manuels de systèmes de contrôle et la littérature de recherche. Sa popularité découle en partie du fait qu'elle est instable sans contrôle, c'est-à-dire que le pendule tombera simplement si le chariot

n'est pas déplacé pour l'équilibrer. En outre, la dynamique du système est non-linéaire. L'objectif de la commande du système est d'équilibrer le pendule inversé en appliquant une force au chariot sur lequel le pendule est fixé. Dans ce cas, nous considérons un problème bidimensionnel où le pendule est forcé de se déplacer dans le plan vertical représenté sur la figure ci-dessous. Pour ce système, l'entrée de commande est la force  $F$  qui déplace le chariot horizontalement et les sorties sont la position angulaire du pendule  $\theta$  et la position horizontale du chariot  $x$  [14].

Soit :

$m$  : Masse du pendule

$b$  : Frottements de déplacement du chariot

$M$  : Masse du chariot

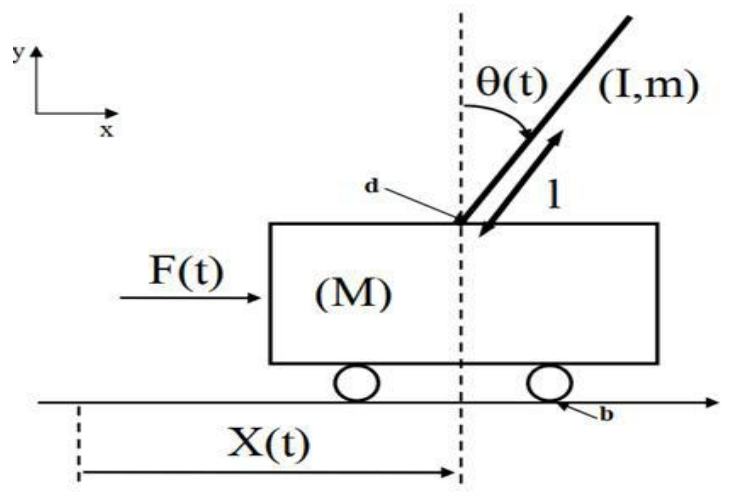
$x(t)$  : Position du chariot

$l$  : Demi-longueur du pendule

$\theta(t)$  : L'angle du pendule

$F(t)$  : Force exercée sur le chariot

$g$  : Intensité de pesanteur



**Figure II.2:** Schéma de l'ensemble chariot et pendule inversé.

Il existe deux méthodes de détermination des équations du mouvement d'un pendule inversé : Celle de la loi fondamentale de la dynamique (LFD) de Newton qui est basée sur le concept de force, et celle de formalisme d'Euler-Lagrange qui est basée sur le principe de la conservation d'énergie mécanique. Dans ce travail on s'intéresse au formalisme d'Euler-Lagrange dans lequel le lagrangien ( $L$ ) est défini comme une différence entre l'énergie cinétique  $E_c$  et l'énergie potentielle  $E_p$  du système :

$$L = Ec - Ep \quad (\text{II.1})$$

## II.4. Modélisation du système chariot-pendule à deux degrés de liberté

Les équations du mouvement du pendule sont déterminées par le formalisme D'Euler-Lagrange, l'avantage de ce formalisme réside dans l'élimination des efforts D'interaction.

Pour simplifier la modélisation, on suppose tout au long de notre étude que la masse du pendule est concentrée au bout de la tige. Afin de pouvoir écrire les équations dynamiques régissant le mouvement du pendule inversé, on considère les coordonnées  $q_1 = x$  et  $q_2 = \theta$  qui représente les deux degrés de libertés du pendule. On prend comme direction positive de  $x$  le sens de gauche à droite et celui de l'angle est le sens des aiguilles d'une montre [15].

### II.4.1. Énergie cinétique du système en mouvement

Le système en mouvement comporte le chariot qui se déplace linéairement sur les rails et le pendule qui se balance sur son axe de rotation. L'énergie cinétique du chariot en mouvement est donnée par l'équation :

$$E_M = \frac{1}{2} M \dot{x}^2 \quad (\text{II.2})$$

L'énergie cinétique du pendule est exprimée par l'équation :

$$E_{cm} = \frac{1}{2} m \vec{v}_c^2 \quad (\text{II.3})$$

$v_c$  : La vitesse de centre de gravité du pendule

La position du centre de gravité du pendule, notée  $r_c$  à partir de ces coordonnées est donnée

Par :

$$\vec{r}_c = (x + l \sin \theta) \vec{i} + l \cos \theta \vec{j} \quad (\text{II.4})$$

D'où la vitesse du centre de gravité du pendule est :

$$\vec{v}_c = \frac{d\vec{r}_c}{dt} = (\dot{x} + l \cos(\theta) \dot{\theta}) \vec{i} - l \dot{\theta} \sin \theta \vec{j} \quad (\text{II.5})$$

En substituant les équations (II.3) dans l'équation (II.5) on trouve :

$$E_{cm} = \frac{1}{2}m(\dot{x}^2 + 2\dot{x}l \cos(\theta)\dot{\theta} + l^2 \cos^2(\theta)\dot{\theta}^2 + l^2 \sin^2(\theta)\dot{\theta}^2) \quad (\text{II.6})$$

Avec simplification, l'équation (II.6) de l'énergie cinétique du pendule prend alors la forme :

$$E_{cm} = \frac{1}{2}m(\dot{x}^2 + 2l\dot{x} \cos(\theta) \dot{\theta} + l^2\dot{\theta}^2) \quad (\text{II.7})$$

L'énergie cinétique de l'ensemble chariot (II.2) et pendule (II.7) est exprimée par :

$$E_c = E_{cM} + E_{cm} = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}^2 + 2l\dot{x} \cos(\theta)\dot{\theta} + l^2\dot{\theta}^2) \quad (\text{II.8})$$

## II.4.2. Énergie potentielle du système

Le chariot étant en mouvement sur un rail horizontal, seul le pendule en mouvement possède une énergie potentielle. L'énergie potentielle du centre de gravité du pendule est donnée par :

$$E_p = mgl \cos \theta \quad (\text{II.9})$$

## II.5. Modélisation dynamique [16]

### II.5.1. Équation de Lagrange

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \left( \frac{\partial L}{\partial q_i} \right) = F_i \quad (\text{II.10})$$

Avec :

L : Le lagrangien.

$\partial \dot{q}_i$ : La  $i^{\text{ème}}$  composante du vecteur de coordonnées généralisées.

$\partial q_i$ : La  $i^{\text{ème}}$  composante de la vitesse généralisée.

$F_i$ : La force généralisée associée a  $q_i$ .

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{x}} \right) - \left( \frac{\partial L}{\partial x} \right) = F \quad (\text{II.11})$$



### II.5.2. L'équation de Lagrange pour le degré de liberté $q(t) = x(t)$

La dérivée partielle du lagrangien suivant  $\dot{x}$  et  $t$  s'écrit :

$$\frac{d}{dt} (M\dot{x} + m\dot{x} + ml \cos(\theta) \dot{\theta}) = F \quad (\text{II.12})$$

D'où la première équation de Lagrange :

$$(M + m)\ddot{x} + ml \cos(\theta) \ddot{\theta} - ml \sin(\theta) \dot{\theta}^2 = F \quad (\text{II.13})$$

### II.5.3. L'équation de Lagrange pour le degré de liberté $q(t) = \theta(t)$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \left( \frac{\partial L}{\partial \theta} \right) = 0 \quad (\text{II.14})$$

La dérivée partielle du lagrangien suivant  $\dot{\theta}$  et  $t$  s'écrit :

$$\frac{d}{dt} (ml\dot{x} \cos(\theta) + ml^2\dot{\theta}) - (-ml\dot{x} \sin(\theta)\dot{\theta} + mgl \sin(\theta)) = 0 \quad (\text{II.15})$$

Donc la deuxième équation de Lagrange est :

$$ml^2\ddot{\theta} + ml\ddot{x} \cos(\theta) - ml\dot{x} \sin(\theta) \dot{\theta} + ml\dot{x} \sin(\theta) \dot{\theta} - mgl \sin(\theta) = 0 \quad (\text{II.16})$$

Après la simplification, on obtient :

$$ml^2\ddot{\theta} + ml\ddot{x} \cos(\theta) - mgl \sin(\theta) = 0 \quad (\text{II.17})$$

Vers la fin, le modèle du système chariot-pendule est donné par :

$$\begin{cases} (M + m)\ddot{x} + ml \cos(\theta) \ddot{\theta} - ml \sin(\theta) \dot{\theta}^2 = f \\ ml^2\ddot{\theta} + ml\ddot{x} \cos(\theta) - mgl \sin(\theta) = 0 \end{cases} \quad (\text{II.18})$$

Ce modèle peut être écrit sous la forme de Gauchy (modèle d'état non linéaire). Pour cela, on définit les variables d'état comme suit :

$X_1 = x$  (Position rectiligne du chariot).

$X_2 = \dot{x}$  (vitesse du chariot).

$X_3 = \theta$  (Position angulaire de la tige).

$X_4 = \dot{\theta}$  (vitesse angulaire de la tige).

Le modèle d'état non linéaire s'écrit sous la forme suivante :

$$\dot{X}(t) = f(X(t)) + g(X(t))f(t) \quad (\text{II.19})$$

## II.6. La régulation PID

L'erreur observée est la différence entre la consigne et la mesure. Le PID permet 3 actions en fonction de cette erreur.

La commande PID est dite aussi (correcteur, régulateur, contrôleur), se compose de trois termes **P**, **I** et **D**, d'où le "P" correspond au terme proportionnel, "I" pour terme intégral et "D" pour le terme dérivé. Les régulateurs PID sont probablement les plus largement utilisés dans le contrôle robotique. Même les plus complexes systèmes de contrôle industriel peut comporter un réseau de contrôle dont le principal élément de contrôle est un module de contrôle PID [17].

Le régulateur PID est une simple implémentation de retour d'information. Il a la capacité d'éliminer la compensation de l'état d'équilibre grâce à l'action intégrale, et il peut anticiper le futur grâce à une action dérivée. L'équation d'un régulateur PID est donnée comme suit :

$$R(t) = K_p \cdot e(t) + K_i \int e(t) dt + K_d \cdot \frac{de}{dt} \quad (\text{II.20})$$

Le développement extraordinaire des microprocesseurs et microcontrôleurs a provoqué des changements importants dans la conception des systèmes de commande régulation des procédés industriels. Leur puissance de calcul et leur faible coût les rendent aptes à prendre également en charge les aspects commande/régulation des procédés en question avec des performances nettement supérieures à celles des régulateurs analogiques. Autant dire que la régulation numérique va se substituer progressivement à la régulation analogique. Cependant la mise en œuvre de ces nouveaux systèmes numériques de contrôle commande nécessite de nouvelles compétences pour l'utilisateur [18].

## II.7. Implémentations de PID numérique sur Arduino

Le but de cette partie est justement d'essayer d'expliquer comment implémenter une régulation PID sans entrer dans les calculs ni sans utiliser les fonctions de transfert du système et autres équations plus compliqué [19].

### II.7.1. Le régulateur proportionnel (P : première règle)

La commande de ce régulateur est proportionnelle à l'erreur.

$$\text{Commande} = K_p * \text{erreur}$$

$K_p$  est le coefficient de proportionnalité de l'erreur à régler de façon manuelle.

### II.7.2. Le régulateur proportionnel intégral (PI : première et seconde règle)

La commande de ce régulateur est proportionnelle à l'erreur, mais aussi proportionnelle à l'intégrale de l'erreur. On rajoute donc à la commande générée par le régulateur proportionnel, la somme des erreurs commises au cours du temps.

$$\text{Commande} = K_p * \text{erreur} + K_i * \text{somme\_erreurs}$$

$K_i$  est le coefficient de proportionnalité de la somme des erreurs. Il faut aussi le régler de façon manuelle.

### II.7.3. Le régulateur proportionnel dérivé (PD : première et troisième règle)

La commande de ce régulateur est proportionnelle à l'erreur, mais aussi proportionnelle à la dérivée de l'erreur. La dérivée de l'erreur correspond à la variation de l'erreur d'un échantillon à l'autre et se calcule simplement en faisant la différence entre l'erreur courante et l'erreur précédente (c'est une approximation linéaire et locale de la dérivée).

$$\text{Commande} = K_p * \text{erreur} + K_d * (\text{erreur} - \text{erreur précédente})$$

$K_d$  est le coefficient de proportionnalité de la variation de l'erreur. Il faut régler ce coefficient expérimentalement.

### II.7.4. Le régulateur proportionnel intégral dérivé (PID : première, seconde et troisième règle)

La commande est à la fois proportionnelle à l'erreur, proportionnelle à la somme des Erreurs et proportionnelle à la variation de l'erreur.

$$\text{Commande} = K_p * \text{erreur} + K_i * \text{somme\_erreurs} + K_d * (\text{erreur} - \text{erreur précédente})$$

Nous allons donc faire une mesure sur notre système pour pouvoir calculer l'erreur et ainsi Appliquer le PID. Cette mesure est à faire régulièrement à une certaine fréquence d'échantillonnage.

Tous les x millisecondes, faire :

*Erreur = consigne - mesure;*

*Somme\_erreurs += erreur;*

*Variation- erreur = erreur - erreur précédente;*

*Commande =  $K_p * erreur + K_i * somme\_erreurs + K_d * variation\_erreur$ ;*

*Erreur précédente = erreur*

### II.7.5. Régler les coefficients d'un PID

Le réglage des coefficients  $K_p$ ,  $K_i$  et  $K_d$  d'un PID peut se faire expérimentalement par essais/erreurs. Tout d'abord, sachons qu'il ne sert à rien de vouloir régler les trois coefficients En même temps. Il y a trop combinaisons possibles et trouver un triplet performant relèverait De l'exploit. Il vaut mieux y aller par étape.

Tout d'abord, il faut mettre en place un simple régulateur proportionnel (les Coefficients  $K_i$  et  $K_d$  sont donc nuls). Par essais/erreurs, il faut régler le coefficient  $K_p$  afin d'améliorer le temps de réponse du système. C'est-à-dire qu'il faut trouver un  $K_p$  qui permette au système de se rapprocher très vite de la consigne tout en faisant attention de garder la stabilité du système : il ne faut pas que le système réponde très vite tout en oscillant beaucoup une fois ce coefficient réglé, on peut passer au coefficient  $K_i$ . Celui-là va permettre

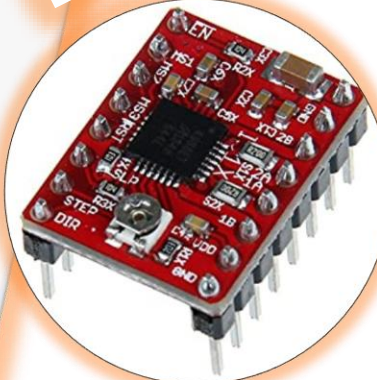
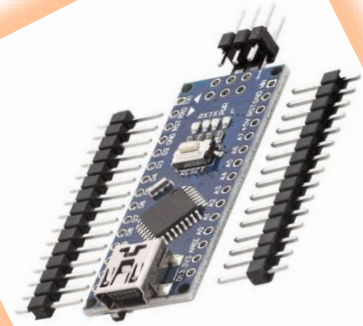
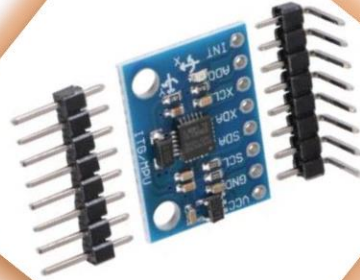
D'annuler l'erreur finale du système afin que celui-ci respecte exactement la consigne. Il faut Donc régler  $K_i$  pour avoir une réponse exacte en peu de temps tout en essayant de minimiser les oscillations apportées par l'intégrateur.

Enfin, on peut passer au dernier coefficient  $K_d$  qui permet de rendre le système plus Stable. Son réglage permet donc de diminuer les oscillations [20].

### II.8. Conclusion

Dans ce chapitre, on a présenté le principe de fonctionnement du pendule inversé. Nous avons proposé aussi un modèle mathématique simplifié du système global : chariot-pendule, qui est le système du pendule inversé sur deux roues.

Pour notre projet nous utilisons une méthode plus facile pour le faire fonctionner avec une implémentation de régulation PID sans faire des calculs compliqués. Dans le prochain chapitre nous allons aborder le choix et les caractéristiques et différents composants utilisés dans ce projet.



## CHAPTER 03

# Conception matérielle du robot auto-balancé

### **III.1. Introduction**

Nous intéresserons dans ce chapitre aux composants électronique et dispositifs nécessaires pour la réalisation et la conception de notre projet (robot auto balancé). Nous allons commencer par un schéma synoptique pour décrire l'architecture matérielle et le fonctionnement du robot. On présente par la suite le choix des composants, leurs principes de fonctionnement et leurs branchements. En fin nous terminons par le schéma électronique général du robot à réaliser.

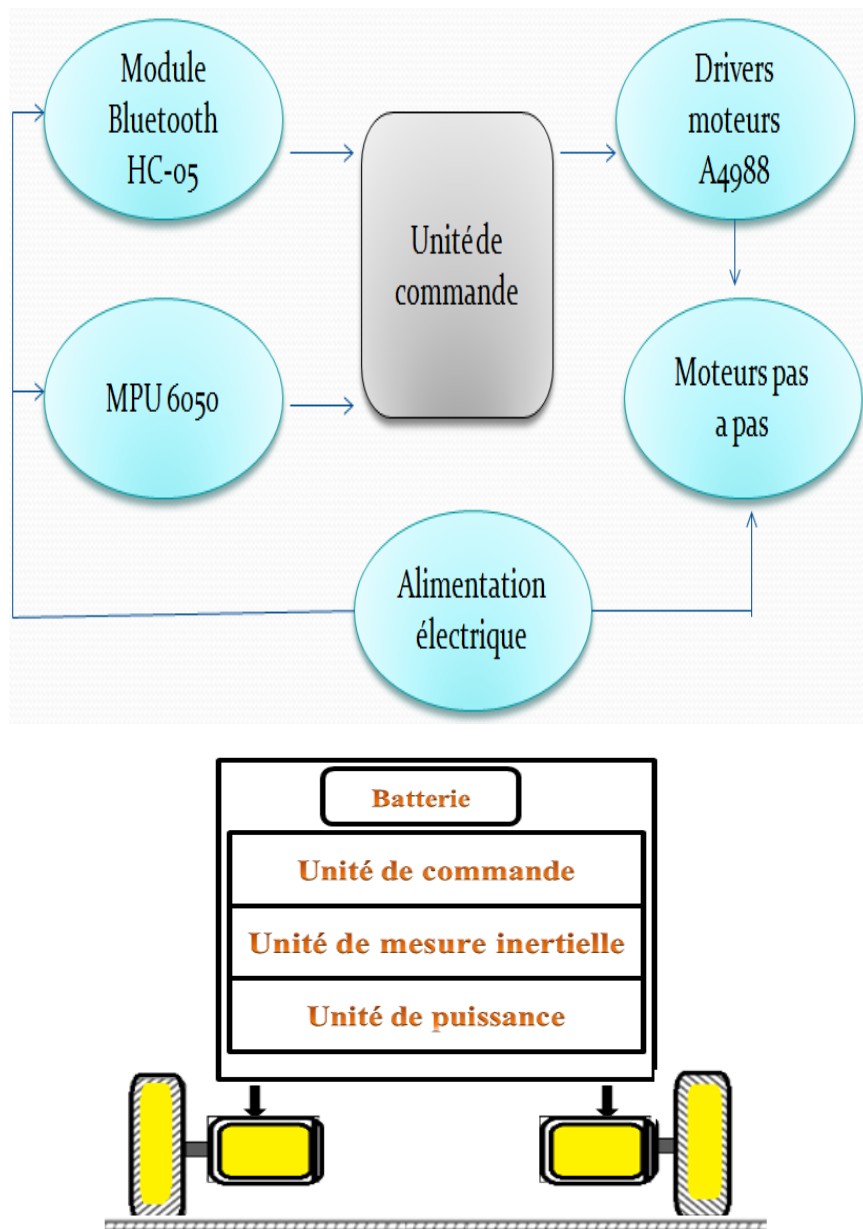
### **III.2. Schéma synoptique et fonctionnement du robot**

Le principe du pendule inversé a pour but de garder les roues sous le centre de la masse du châssis du robot. Si le robot commence à s'incliner vers l'avant, alors pour maintenir la stabilité, la roue devra avancer pour revenir sous la masse du châssis. Si cela n'est pas maintenu, le robot va simplement tomber.

Il est nécessaire de généraliser les effets des roues gauche et droite et de les incorporer sous le terme combiné «roues». Cela simplifie les calculs car les deux roues travailleront à l'unisson pour maintenir la stabilité.

La Figure III.1 présente l'architecture générale de notre robot et ces différentes unités, Le robot mobile est constitué de :

- Une unité de commande (carte Arduino NANO) : c'est le cerveau de notre système.
- Une unité de mesure : basée sur une puce de mesures inertielles nommée MPU6050 qui intègre un accéléromètre et un gyroscope qui permettent de calculer l'angle d'inclinaison du robot.
- Une unité de puissance constitué de deux moteurs pas à pas avec leurs drivers de type A4988.
- Un Module Bluetooth qui permettra d'établir une communication avec le robot afin de pouvoir le commandé à distance par une application mobile nommée EZGUI.
- Une Batterie électrique de 12V : réalisée à partir des cellules Lithium de récupération (batteries de Laptop) et un module BMS 3S de protection et régulation de charge.



**Figure III.1** : schéma synoptique et architecture générale de robot

### III.3. Unité de commande

L'unité de commande qui utilise dans notre robot est la carte Arduino NANO, cette dernière est une carte électronique programmable qui permet de traiter les informations qui entre ou sorte vers la carte, Il est similaire au cerveau humain. Le cerveau reçoit des informations de l'environnement externe puis les traite dans des conditions Standard, puis envoie une dynamique aux membres pour remplir sa fonction.

### **III.3.1. Les cartes Arduino**

Le système Arduino est une plateforme open source basée sur une carte programmée par microcontrôleur et un environnement de développement (SDK) permettant d'écrire, compiler et tester un programme. Les cartes et modules Arduino sont pourvus d'entrées-sorties qui peuvent recevoir des signaux de capteurs ou interrupteurs et peuvent commander des moteurs, éclairages, etc.

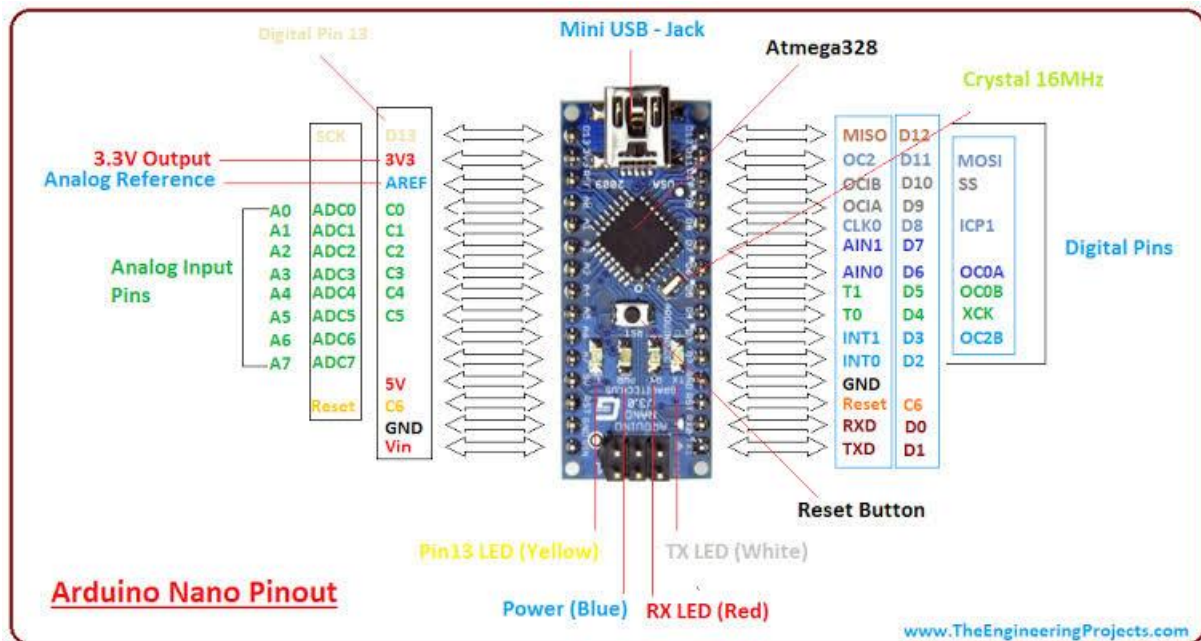
Le choix d'un microcontrôleur pour faire de la programmation dépend en réalité des situations. Autrement dit, il faut trouver ce qui convient idéalement au type de projet que l'on souhaite réaliser. Cet état de choses peut malheureusement limiter les travaux. Heureusement, il existe un microcontrôleur capable de s'adapter à toute situation. La carte Arduino est un élément fiable et capable d'exécuter un programme complet. Elle représente la solution adaptée pour contrôler l'électronique. Il existe plusieurs types de cartes Arduino qui ont été produites et vendues dans le commerce à ce jour et les cartes les plus utilisés sont : Arduino UNO, Arduino Nano, Arduino Due, Arduino Mega, et Arduino Leonardo.

Cependant, dans certains cas, il peut arriver que l'on soit confronté à un problème de format ou d'espacement irrégulier avec certaines cartes Arduino. C'est justement pour faciliter ce genre de situation que la carte Arduino Nano a été conçue. Notre choix est basé sur cette carte qui présente de nombreux avantages et s'avère une solution idéale pour apprendre à coder avec l'IDE Arduino [21].

### **III.3.2. La carte Arduino NANO**

L'Arduino Nano est une carte de dimensions 45mm x 18mm. Très légère, elle ne pèse que 5g, ce qui lui permet de s'adapter aux petits systèmes électroniques embarqués et aux petits robots. C'est la carte la moins chère du marché. Cependant, ce serait une erreur d'établir une corrélation entre son prix et sa performance. Certes, elle coûte moins cher, mais elle rend un service de qualité et affiche des performances aussi remarquables que celles d'une UNO. Elle est dotée de 14 pins digitaux à travers lesquels elle fournit 40 mA maximum. En outre, elle intègre 2 pins analogiques de plus que la carte UNO. Cela fait donc 8 pins analogiques pour la Nano. En ce qui concerne la capacité, elle est dotée d'une mémoire Flash de 32Kb et 2Kb de mémoire dynamique. La particularité de la carte est qu'elle est très minimaliste et se présente avec un empattement vraiment réduit. On note l'absence du connecteur DC de 2.5 mm chez l'Arduino Nano. Elle est adaptée à des projets où l'on est amené à intervenir sur un espace très restreint. Elle est aussi sollicitée pour réaliser des systèmes légers ou souples.





**Figure III.2** : Vue générale et branchement de la carte Arduino NANO

La Nano peut être reliée à un PC grâce à un câble micro-USB. À noter que la Nano est livrée sans le câble, mais il est possible de l'obtenir dans le commerce. Il existe une similitude entre la carte Arduino Nano et la carte Arduino Uno, en ce sens que les deux sont basées sur un ATmega328 couplé à un quartz de 16 MHz. On peut donc affirmer que la Nano est tout simplement le format réduit de l'Arduino UNO, qui correspond au format standard DIL30. Ainsi, lors des travaux, il devient facile d'insérer l'Arduino Nano sur une breadboard. Il est également possible de la souder sur des cartes électroniques. De ce fait, elle permet de travailler sans difficulté majeure, car elle devient facile à utiliser. La programmation de l'Arduino Nano se fait exactement comme les autres avec l'IDE Arduino ou avec des versions supérieures. L'Arduino Nano intègre toutes les fonctionnalités électroniques qui permettent de réaliser des travaux de programmation sans difficulté, mais aussi d'utiliser un microcontrôleur intégré. Pour cela, il est juste nécessaire de relier la carte à un PC à l'aide d'un câble USB.

Pour fonctionner, la carte Nano a besoin d'être alimentée par une prise mini-USB. Il est également possible d'utiliser une source d'énergie externe avec les caractéristiques suivantes 6-20V (pin 30). On peut également se servir d'une source d'énergie externe non régulée à 5 V (pin 27). Pour la programmer, on peut utiliser le logiciel de programmation libre de droits disponible en téléchargement et mis à disposition par Arduino. Au moment de la programmation, les cartes cibles à choisir sont les suivantes : Arduino Diecimila

Duemilanove ou Nano w/ ATmega16 Arduino Duemilanove ou Nano w/ ATmega328 » Il faut noter que le microcontrôleur ATmega328 de la carte Arduino Nano bénéficie d'un bootloader. Celui-ci assure le téléchargement de nouveaux codes sans pour autant solliciter un programmeur matériel supplémentaire. Toutefois, il existe une solution qui permet de se passer du boot loader afin de programmer le microcontrôleur grâce au connecteur In-Circuit Serial programming [22].

### III.4. L'unité de mesure inertielle MPU6050

Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable.

La puce est équipée d'un accéléromètre 3 axes et d'un gyroscope 3 axes. Outre l'acquisition des mesures, la puce est aussi équipée d'un processeur appelé DMP (Digital Motion Processor) se chargeant de convertir les mesures brutes en données compréhensibles pour un humain. Le processeur DMP réalise les calculs directement sur la puce allégeant ainsi le travail de l'Arduino. On peut aussi lire les mesures brutes sans passer par le traitement du DMP mais c'est très compliqué.

Le MPU-6050 est livré avec une bibliothèque qui fait tous les calculs et filtrations et fourni une information numérique [23].

Le MPU-6050 utilise I2C pour communiquer avec la carte Aduino, alors on a commencé en connectant les broches comme indiqué dans les schémas : la ligne SDA se connecte à la broche analogique 4, le SCL analogique broche 5, puissance d'entrée à la broche de 3.3v et le sol à la broche GND.



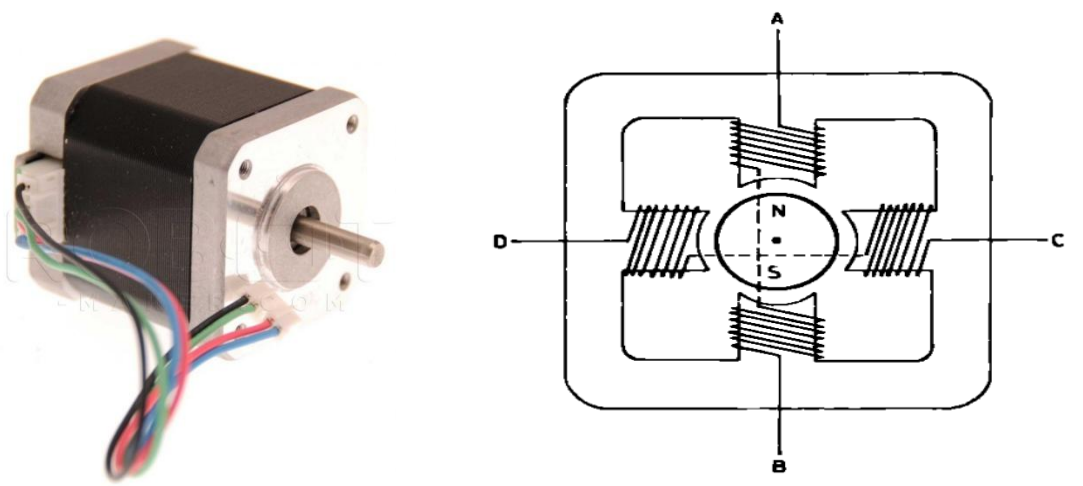
**Figure III.3 :** Une carte GY-521 dotée de la puce MPU6050 et son branchement.

### III.5. L'unité de Puissance

#### III.5.1. Les moteurs pas à pas

Les moteurs utilisés de manière classique en robotique sont assez diversifiés, les plus répandus sont les moteurs "pas à pas", les moteurs "à courant continu" et enfin les servomoteurs bien connus dans le milieu du modélisme. Dans notre projet nous avons approuvé les moteurs pas à pas.

Le moteur pas à pas est un convertisseur électromécanique qui assure la transformation d'un signal électrique impulsionnelle en un déplacement mécanique (angulaire ou linéaire). Sa structure de base se présente sous la forme de deux pièces séparées mécaniquement, le Stator et le Rotor. L'interaction électromagnétique entre ces deux parties assure la rotation [22].



**Figure III.4 :** moteur pas à pas et sa structure interne.

Le moteur pas à pas offre une grande précision de mouvement. Dans ce dernier, c'est sur le rotor (au centre) que l'on retrouve l'aimant permanent, et les bobines sont sur le stator (autour du rotor) dans le but est de "faire tourner un champ magnétique" pour faire tourner l'aimant fixé au rotor. Il existe cependant différents types de moteurs pas à pas dont le placement des bobinages diffère les uns des autres et la façon de les alimenter n'est pas non plus identique. Nous allons maintenant étudier le câblage ainsi le fonctionnement du moteur pas à pas bipolaire.

Le moteur pas à pas bipolaire à aimants permanents possède quatre fils d'alimentation pour piloter des bobines par paire. Comme le montre le schéma de la figure III.4.

Les bobines sont reliées deux à deux en série et sont donc pilotées ensemble. Il n'y a donc finalement que deux enroulements à commander puisque deux bobines montées en série n'en font plus qu'une.

- Lorsqu'il n'y a aucun courant traversant les bobines, le rotor (où l'axe de sortie est lié) n'est libre de tourner, rien ne cherche à le retenir dans sa course.
- Maintenant, si nous décidons de faire passer du courant entre les points C et D pour alimenter la bobine de gauche et celle de droite. Un courant va s'établir et deux champs électromagnétiques vont apparaître de part et d'autre du rotor. Que va-t-il alors se passer ? L'aimant du rotor va tourner sur lui-même pour se placer de façon à ce que son pôle Nord soit en face du pôle Sud du champ magnétique créé dans la première bobine et que son pôle Sud soit en face du pôle Nord créé dans la deuxième bobine.
- Si ensuite on alimente non plus les bobines entre C et D mais plutôt celles entre A et B, le rotor va alors tourner pour s'aligner à nouveau vers les pôles qui l'intéressent (Nord/Sud, Sud/Nord).
- Et c'est reparti, on va alors alimenter de nouveau les bobines entre D et C, donc avec un courant de signe opposé à la fois où l'on les a alimenté entre C et D (par exemple C était relié au "+" de l'alimentation tout à l'heure et là on le fait passer au "-", idem pour D que l'on fait passer du "-" au "+") et le moteur va encore faire un quart de tour.
- On peut continuer ainsi de suite pour faire tourner le moteur en faisant attention de ne pas se tromper dans les phases d'alimentation.

À chaque phase on va donc faire tourner le moteur d'un quart de tour (pas) :

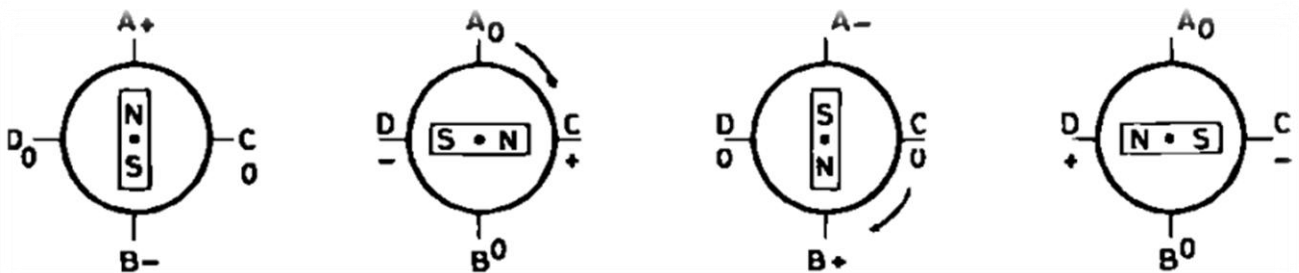


Figure III.5 : Principe de rotation pas-à-pas du moteur

Dans le cas illustré ci-dessus, on dit que le moteur fait 4 pas par tour. Il existe bien des moteurs qui font ce nombre de pas, mais il en existe qui ont un nombre de pas plus conséquent (24, 48, etc.).

### Rotation par pas complet :

Grâce à ce dernier, nous allons pouvoir faire des rotations "pas par pas". Pour cela, nous allons alternativement alimenter les bobines de droites et de gauche et inverser le sens du courant pour pouvoir faire une rotation complète du champ magnétique.

En considérant que la bobine A est connectée aux entrées IN1 et IN2 et que la bobine B est connectée aux commandes IN3 et IN4, on peut donc écrire la séquence de commande suivante :

Etape	In1	In2	In3	In4
Pas $n^0$	HIGH	LOW	-	-
Pas $n^0$	-	-	HIGH	LOW
Pas $n^0$	LOW	HIGH	-	-
Pas $n^0$	-	-	LOW	HIGH

Figure III.6 : Séquence à pas complet

On peut traduire cette activité par le chronogramme suivant :

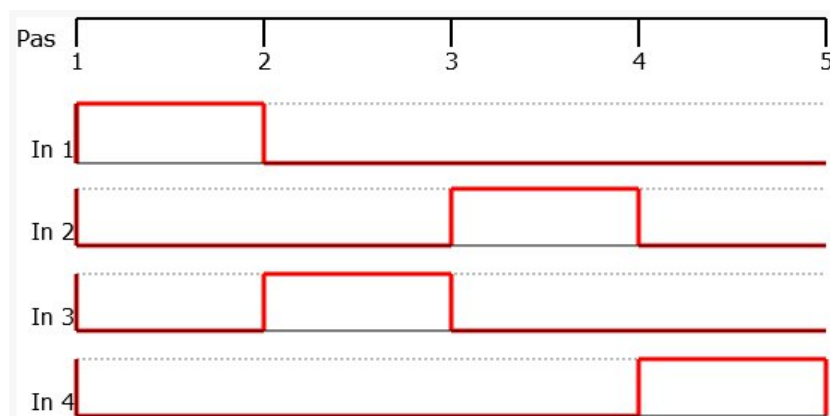


Figure III.7 : Chronogramme du pilotage à pas entier

### III.5.2. Le pilote (Driver) A4988

A4988 est un pilote de moteur micro-pas complet avec traducteur intégré pour une utilisation facile. Ce produit est disponible en modes full, half, 1/4, 1/8 et 1/16 pas actionnent des moteurs pas à pas bipolaires, capacité d'entraînement de sortie allant jusqu'à 35 V et  $\pm 2$  A. A4988 comprend un régulateur de courant fixe, le régulateur en modes de décroissance lente ou mixte. Le convertisseur A4988 est la clé de la mise en œuvre facile. Tant que l'entrée «étape» introduit une impulsion entraîne le moteur un microstep. L'interface A4988 est très appropriée pour le microprocesseur complexe est indisponible ou est surchargé. Dans le fonctionnement en micro-étape, le contrôle de hachage A4988 à l'intérieur sélectionne automatiquement le mode de décroissance du courant (lent ou mixte). En mode de

désintégration mixte, l'appareil est initialement réglé sur un temps d'arrêt fixe dans une certaine désintégration rapide, puis le reste du temps d'arrêt lent. Le schéma de contrôle du courant de décroissance mixte réduit le bruit sonore du moteur, augmente la précision des pas et réduit la consommation d'énergie.

Les caractéristiques et les avantages de ce driver sont résumés ci-dessous :

- ✓ Faible sortie RDS (On) ·
- ✓ Détection/sélection automatique du mode de décroissance du courant ·
- ✓ Mélanger avec les modes de décroissance à courant lent ·
- ✓ Rectification synchrone pour une faible dissipation de puissance ·
- ✓ Protection contre les courants croisés ·
- ✓ Alimentation logique compatible 3.3 et 5 V ·
- ✓ Circuit d'arrêt thermique ·
- ✓ Protection contre les défauts au sol ·
- ✓ Protection contre les courts-circuits ·
- ✓ Étape 5 en option : full, 1/2, 1/4, 1/8 et 1/16

Le module dispose d'un brochage et interface presque identique à notre contrôleur de moteur pas-à-pas A4988. Il peut donc être directement utiliser en remplacement sur les différentes cartes d'interface avec ce modèle haute performance [25].

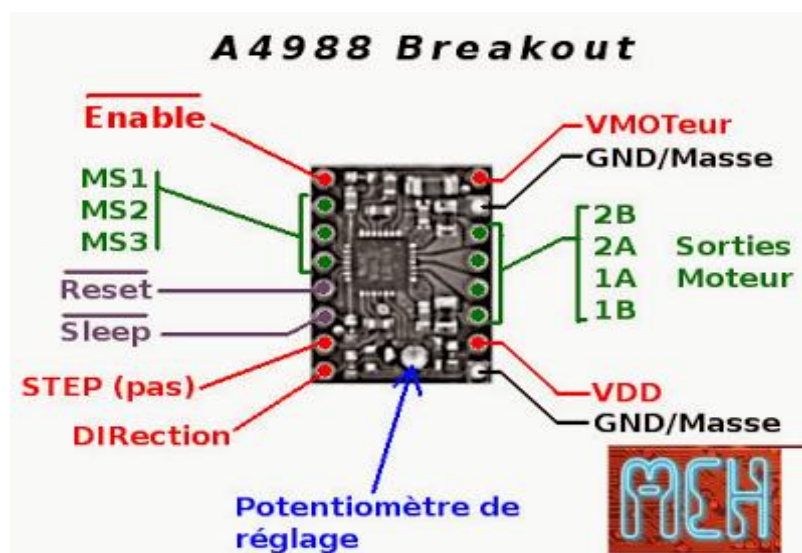


Figure III.8: Le pilote (Driver) A4988 et son branchement.

Il est nécessaire de régler le courant en fonction de votre moteur pas-à-pas, ici pour notre moteur il faut régler le courant sur 0,7 A (on règle généralement le courant à 70% du courant de bobine ici 1A). Chaque driver possède son propre système de réglage du courant, Il faut alimenter le driver à partir de la carte Arduino pour procéder au réglage du courant :

- VCC de l'Arduino sur VDD du driver
- GND de l'Arduino sur le GND du driver

### III.6. Module Bluetooth HC-05:

Module de communication série HC-05 Bluetooth intégré (peut être un raccourci pour module) Deux modes de fonctionnement : Mode de travail Réponse à la demande et numérotation automatique Système de travail. Il existe trois rôles professionnels (maître, esclave et bouclage) en mode de connexion automatique. Lorsque l'appareil est en connexion automatique en mode de travail, la méthode par défaut suivra finalement automatiquement le transfert de données spécifié. Lorsque l'unité est en mode d'action Réponse à la demande, l'utilisateur peut envoyer une commande à unité pour ajuster les paramètres de contrôle et envoyer la commande de contrôle. Méthode de travail L'unité peut être commutée en contrôlant le niveau d'entrée PIN (PIO11). PIO11 : est le commutateur de mode de travail. Lorsque ce port PIN est entré de haut niveau, le travail Le mode deviendra le mode de travail de réponse à la commande. Tant que ce port PIN est entré au niveau bas ou suspendu dans l'air, le mode de travail deviendra le mode de travail de connexion automatique. Le module de communication qui utilise dans notre projet est pour commander notre robot à distance [26].

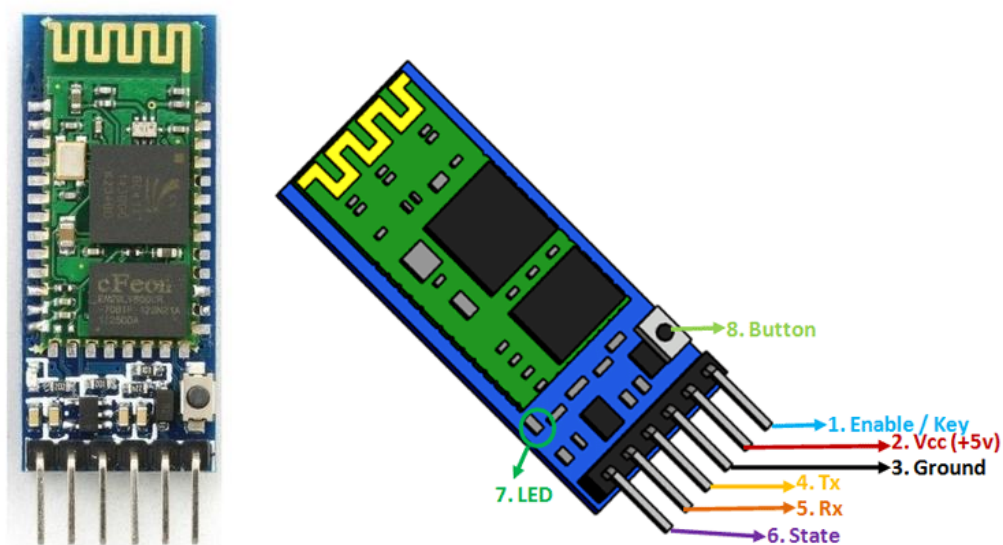


Figure III.9 : module Bluetooth HC-05 et son branchement

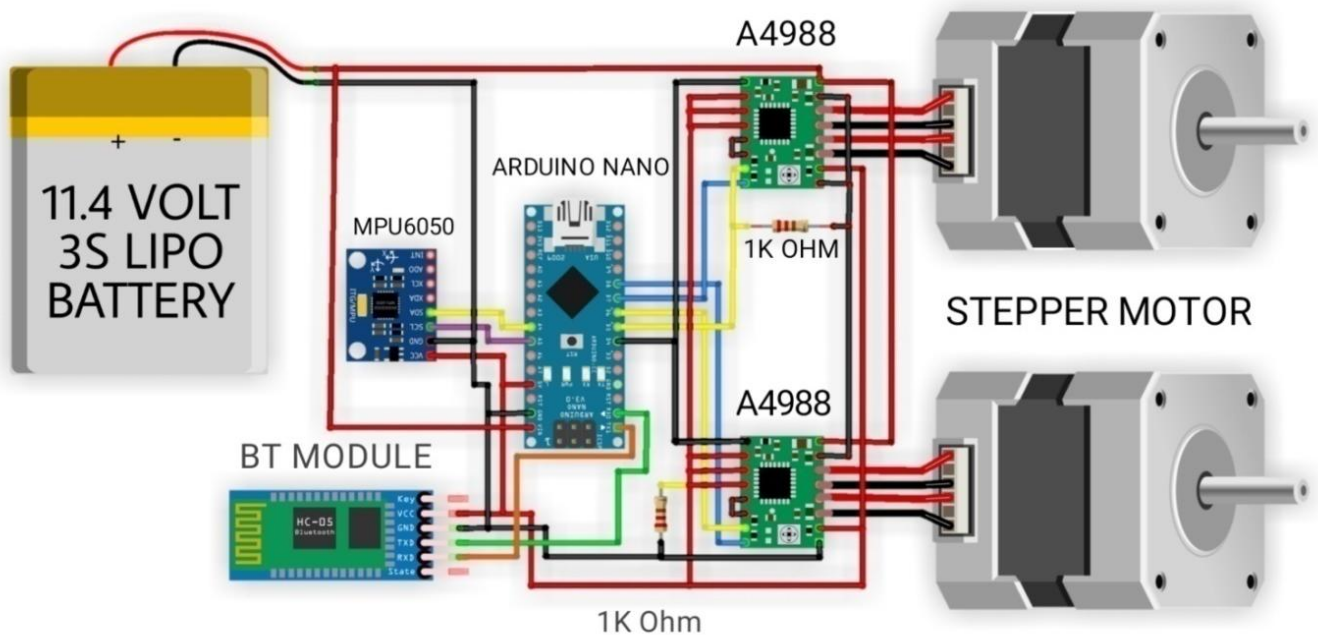
### III.7. Batterie d'alimentation électrique

Dans notre projet il fallait utiliser une alimentation électrique qui est une batterie Lipo à 12 v, mais on a préféré de construire notre propre batterie dans le but d'économiser le cout vue son prix élevé par rapport aux autres composants d'un côté et de l'autre pour avoir une expérience supplémentaire beaucoup plus pratique. Cette partie sera bien détaillée dans le prochain chapitre.

### III.8. Branchement globale du système

Comme la montre le schéma si dessus, on a utilisé deux moteurs pas à pas bipolaire à quatre phases qui sont branchés aux pines (2B ,2A, 1A, 1B) des deux drivers moteurs de type A4988 qui sont eux-mêmes reliés à partir des deux pines STEP et DIR a la carte Arduino NANO. Cette dernière est connectée à une puce MPU 6050 et un module Bluetooth.

L'ensemble des composants de ce système sont alimentés avec une batterie de 12V.

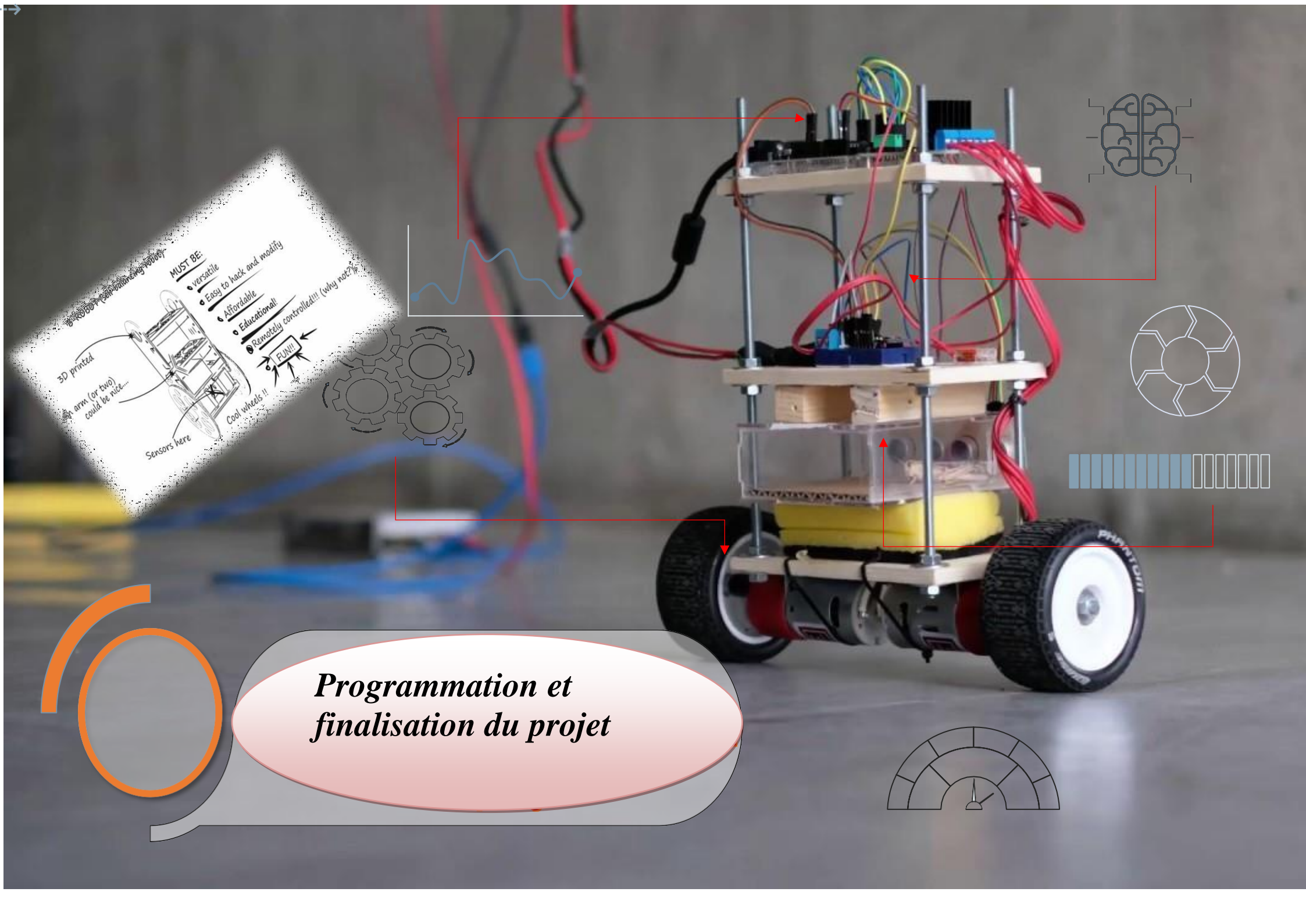


**Figure III.10** : schéma électronique global du robot auto balancé.

### III.9. Conclusion

Dans ce chapitre, nous avons présenté les différents composants de chaque unité qui constitue le robot auto balance à réaliser. Nous avons aussi présenté le fonctionnement et le branchement du circuit global du robot. Dans le chapitre suivant, nous allons parler sur la partie programme du projet et la construction finale du robot à réaliser.





3D printed  
An arm (or two)  
could be nice.....  
Sensors here

**MUST BE:**

- Versatile
- Easy to hack and modify
- Affordable
- Educational
- Remotely controlled!!! (why not?)

**FUN!!!**

Cool wheels !!

*Programmation et finalisation du projet*

### **IV.1. Introduction**

Dans le chapitre précédent, on s'est intéressé à l'architecture matérielle et les éléments nécessaires pour réaliser notre projet. Dans ce chapitre, nous allons présenter la partie software c.à.d. l'algorithme à implémenter sur l'Arduino. Nous allons entamer par la suite la réalisation pratique du robot. En outre nous allons citer les différentes contraintes rencontrées au cours de la réalisation. On termine ce chapitre par des essais pratiques de notre robot auto balancé.

### **IV.2. Organigramme de l'algorithme du système**

Cet organigramme représente le code à implémenter sur la carte du robot auto balancé. Il est constitué de deux étapes initiales. La première étape consiste en mesure de l'angle d'inclinaison de robot par l'IMU MPU6050. La deuxième étape est le calcul des paramètres PID par l'unité de commande. Si l'erreur de l'angle d'inclinaison est positive on tourne les roues vers l'arrière, et s'il est négatif, on tourne les roues vers l'avant pour maintenir le robot en état vertical. A la fin le code revient au début de l'algorithme pour refaire les mêmes instructions en boucle.

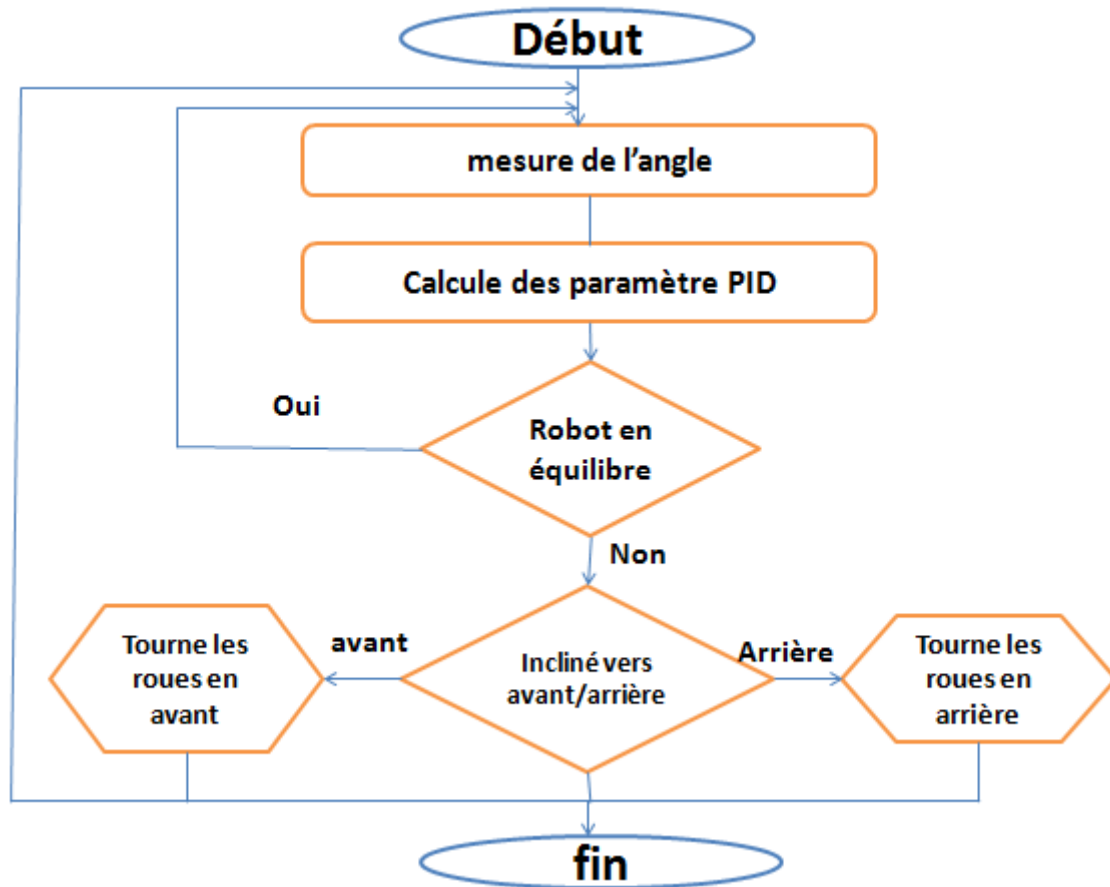


Figure IV.1 : Organigramme de fonctionnement de robot auto balance

### IV.3. Mesure de l'angle d'inclinaison

#### IV.3.1. Raccordement du MPU6050 avec l'Arduino

Le module MPU6050 est une centrale inertielle qui permet de mesurer l'évolution d'un objet dans l'espace. Il permet de mesurer les accélérations linéaires et angulaires dans les trois axes de l'espace. Le module MPU6050 est constitué d'un capteur de température et de deux capteurs micromécaniques : un gyroscope qui permet de mesurer les rotations et un accéléromètre qui permet de mesurer les accélérations dans l'espace. Ces capteurs sont généralement constitués d'une partie mécanique et d'une partie piézoélectrique. Lorsque l'objet est en mouvement la pièce mécanique se déplace contre la partie piézoélectrique qui envoie un signal électrique [27].

Le module peut être alimenté directement via la broche 5V de la carte Arduino. Le Module GY-521 utilise la communication I2C pour s'interfacer avec le microcontrôleur.

- SDA à la broche A4
- SCL à la broche A5
- VCC à la broche 5V
- GND à la broche GND

Le module **MPU6050** communique avec la carte Arduino via un bus de communication **I2C**, Ce bus porte parfois le nom de TWI (Two Wire Interface) chez certains constructeurs. Il existe d'innombrables périphériques exploitant ce bus, il est même implémenté par logiciel. Les données sont transmises à un microcontrôleur par liaison série I2C. Ce dernier est une collection d'interfaces entre divers microcontrôleurs et des périphériques I2C,

Le MPU6050 peut être programmé pour effectuer des numérisations échantillonnées à une fréquence donnée. Il possède une sortie INT sur laquelle un front montant est généré à chaque fois que les données demandées ont été numérisées (accélérations et vitesses angulaires). Cette sortie est branchée sur l'entrée D2 de l'arduino, afin de générer une interruption permettant de lire ces données. Le protocole I2C comprend deux signaux, une ligne de données série (SDA) et une ligne d'horloge série (SCL) (respectivement les bornes A4 et A5 sur l'arduino Nano) [28].

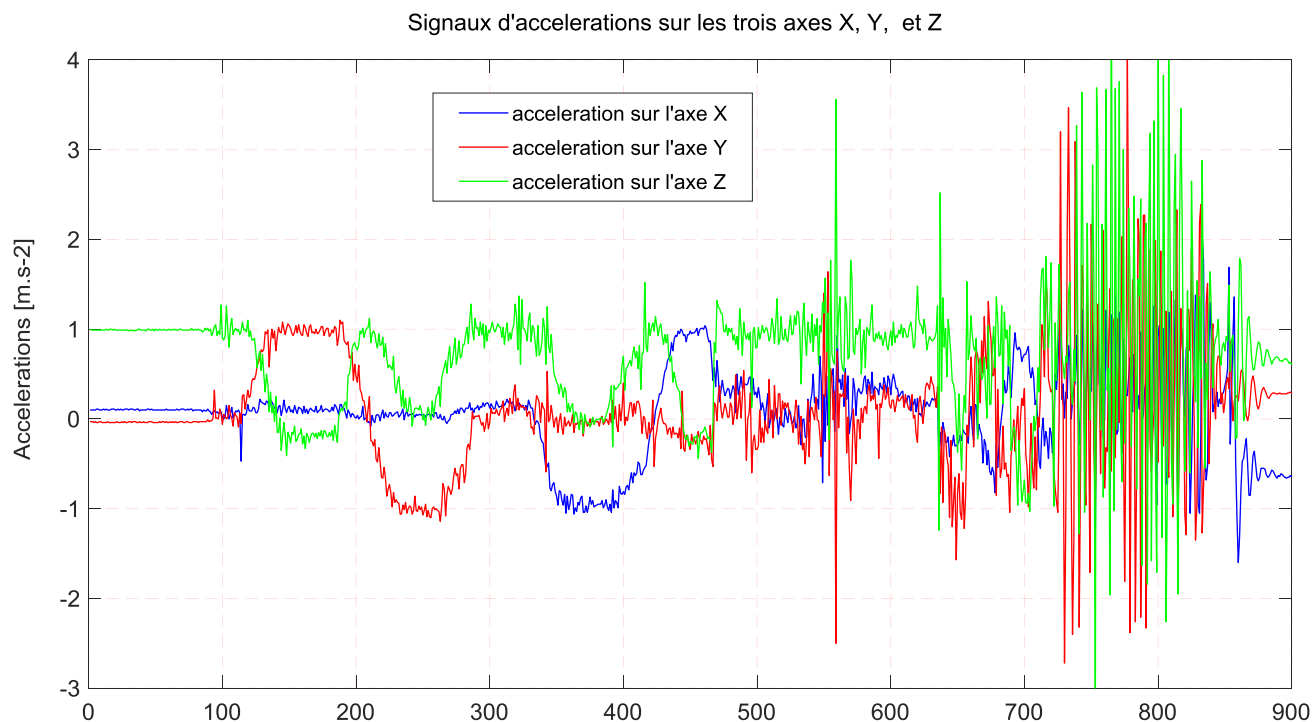
- SDA (Serial Data Line) : ligne de donnée bidirectionnelle.
- SCL (Serial Clock Line) : ligne d'horloge de synchronisation bidirectionnelle.

I2C est un bus série synchrone bidirectionnel half-duplex. Plusieurs équipements, soit Maîtres, soit esclaves, peuvent être connectés au bus. Les échanges ont toujours lieu entre un Seul maître et un (ou tous les) esclave(s), toujours à l'initiative du maître (jamais de maître à maître ou d'esclave à esclave). Cependant, rien n'empêche un composant de passer du statut de maître à esclave et réciproquement [28].

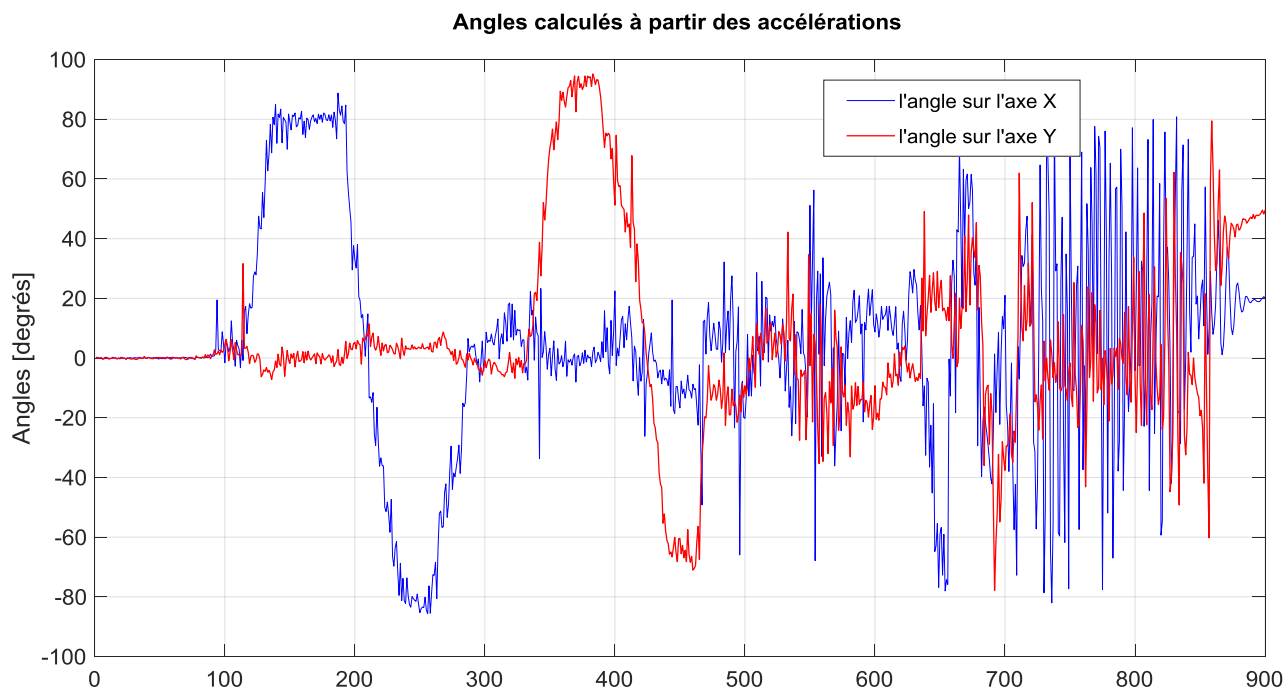
### IV.3.2. Mesure d'angle par l'accéléromètre

Comme cette accélération est toujours vers le bas et de module  $9.81 \text{ m/s}^2$ , si on incline un accéléromètre, une partie de l'accélération de pesanteur va être mesuré sur chaque axe. Ainsi, il est possible d'en déduire, par projection, les angles sur les axes X, Y et Z. Les angles à partir des données d'accélération sont calculés de la manière suivante [29].

$$Angle_x = \arctan\left(\frac{a_y}{\sqrt{a_y^2 + a_z^2}}\right) \quad angle_y = \arctan\left(\frac{a_x}{\sqrt{a_x^2 + a_z^2}}\right)$$



**Figure IV.2** : Mesure des accélérations sur les trois axes X, Y et Z.

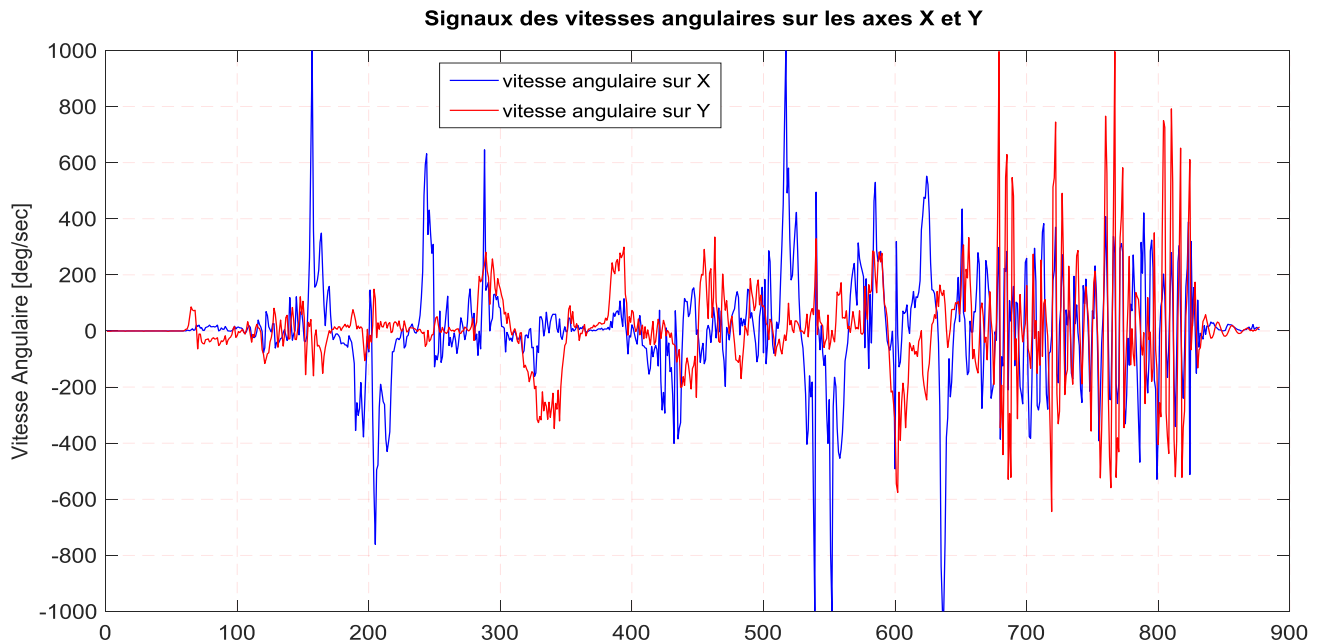


**Figure IV.3** : Mesure des angles par les données d'accéléromètre.

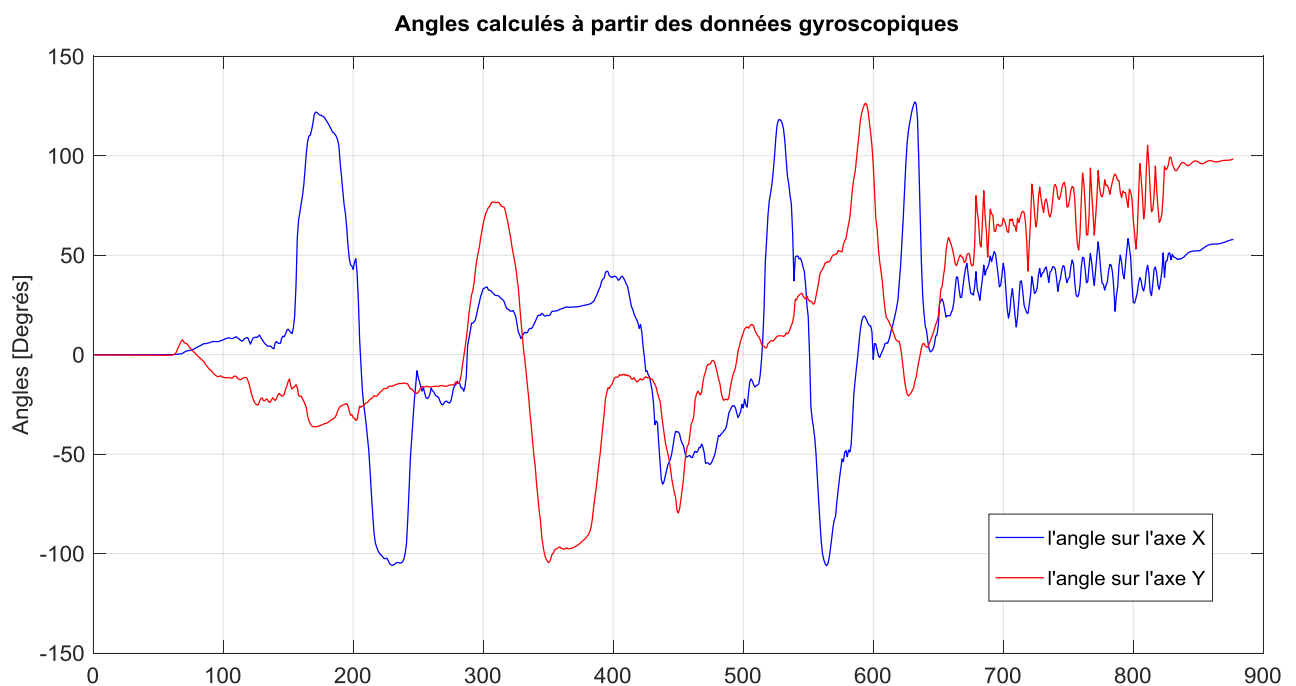
### IV.3.3. Mesure d'angle par le gyroscope

Le gyroscope mesure une vitesse angulaire. Donc on peut obtenir l'angle en intégrant la vitesse angulaire [28].

$$\text{Angle} = \theta_0 + \int w(t)dt$$



**Figure IV.4 :** signaux des vitesses angulaires sur les axes X et Y.



**Figure IV.5 :** angles calculés à partir des données gyroscopiques.

### IV.3.4. Mesure d'angle par fusion de donnée

L'angle mesuré par l'accéléromètre est juste mais bruitée, car le signal d'accélération est très sensible aux chocs. En effet, on considère que l'accélération mesurée est uniquement due à la pesanteur. Or ce n'est vrai que si la vitesse du corps est constante. D'autre part, On peut remarquer que l'angle calculée par intégration des données gyroscopiques est stable mais dérive au cours du temps. Ce phénomène est dû à une accumulation d'erreur au niveau du calcul de l'intégrale.

Afin de présenter et visualiser les données d'accéléromètres, et gyroscopiques ainsi que les angles calculés par les différentes méthodes, nous avons réalisé un programme sous Arduino qui permet de lire, calculer puis envoyer ces données vers un ordinateur à travers la communication série. Puis nous avons utilisé le logiciel Matlab afin de les tracer.

Les Figures IV.3 et IV.5 permettent d'observer le bruit des données de l'accéléromètre et la dérive des données du gyroscope.

Pour résoudre ce problème, un algorithme de fusion de données a été introduit en appliquant un filtre complémentaire aux données de l'accéléromètre et du gyroscope, cela permet de se débarrasser des variations "lentes" du gyroscope et des variations "rapides" de l'accéléromètre.

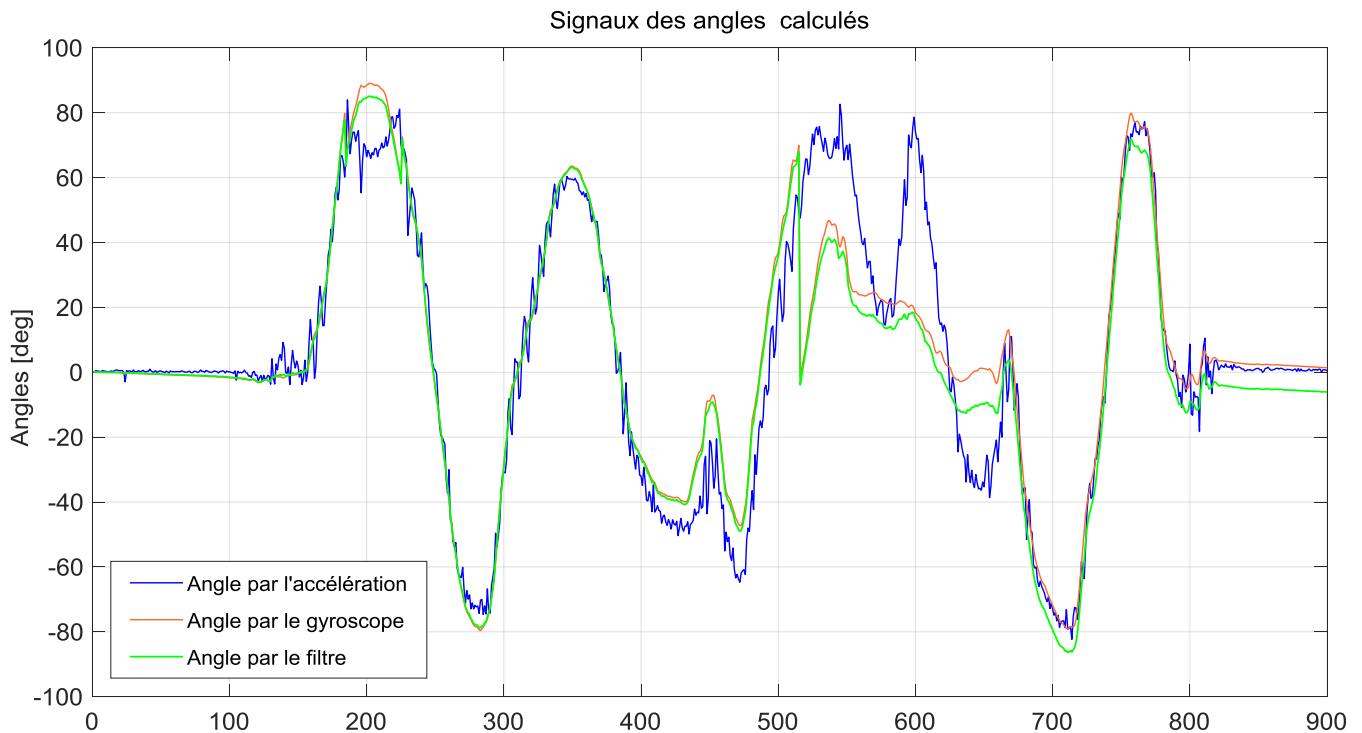
La fusion des données en provenance d'un gyroscope et d'un accéléromètre, se fait par la formule suivante :

$$Angle = 0.96 * (Angle + Gyro * \Delta t) + 0.04 * A_{Accelero}$$

Avec :

- ✓ Angle : l'angle résultant qui nous intéresse.
- ✓ Gyro : la vitesse mesurée par le gyroscope.
- ✓  $\Delta t$  : période d'échantillonnage du signal.
- ✓  $A_{Accelero}$  : angle mesuré par l'accéléromètre.

En mettant en œuvre cette équation de fusion, les données du gyroscope et de l'accéléromètre sont fusionnées en un seul signal en donnant aux signaux des poids différents, par exemple, le poids des données de l'accéléromètre était de 0,04 (4%) et le gyroscope de 0,96 (96%) pour obtenir un signal lisse et filtré, comme illustré dans la figure ci-dessous [29].



**Figure IV.6 :** l'angle mesuré par fusion de données

#### IV.4. La régulation PID

PID est l'acronyme de Proportionnel, Intégral, Dérivé et le mélange de ces trois actions dont le régulateur agit :

- Action Proportionnelle : l'erreur est multipliée par un gain  $K_p$
- Action Intégrale : l'erreur est intégrée et multipliée par un gain  $K_i$
- Action Dérivée : l'erreur est dérivée et multipliée par un gain  $K_d$

L'action proportionnelle est la plus intuitive des trois. Il s'agit d'appliquer un coefficient  $K_p$  à l'erreur. Le résumé de cette action est la diminution de la précision et la stabilité et l'augmentation de la rapidité.

$$K_p \cdot e(t)$$

L'action intégrale consiste à sommer dans le temps les erreurs mesurées et d'appliquer un coefficient  $K_i$  à l'ensemble. Cette action permet de réduire la stabilité et la rapidité et d'améliorer la précision du système.



$$K_i \int_0^t e(t') dt'$$

L'action dérivée consiste à calculer la différence entre l'erreur courante et l'erreur à t-1 et d'appliquer un coefficient  $K_d$  au résultat. Cette action augmente la stabilité du système et diminue la rapidité.

$$K_d \frac{de(t)}{dt}$$

Afin de faciliter l'intégration de la régulation PID dans notre robot, nous avons utilisé la bibliothèque PID\_V1 compatible avec la carte Arduino [30].

### IV.5. Réalisation Pratique du robot

Pour la réalisation la structure de notre robot auto balancé, nous avons utilisé les matériaux nécessaires en respectant la minimisation du poids pour son bon fonctionnement. L'ensemble des composants utilisés dans la réalisation sont :

- 3x plaques MDF sur lesquelles nous allons installer nos matériaux.
- Des tiges filetées et des écrous pour relier les plaques MDF.
- Des fils de raccordement.
- 2x moteurs pas à pas.
- 2x drivers moteurs A4988.
- 1x carte Arduino NANO.
- 1x gyroscope module MPU6050.
- Un module BMS 3S.
- 1x module Bluetooth HC-05
- Des piles extraites de batteries des pc utilisés pour la conception de la batterie.
- 2x roues réalisées manuellement en MDF.

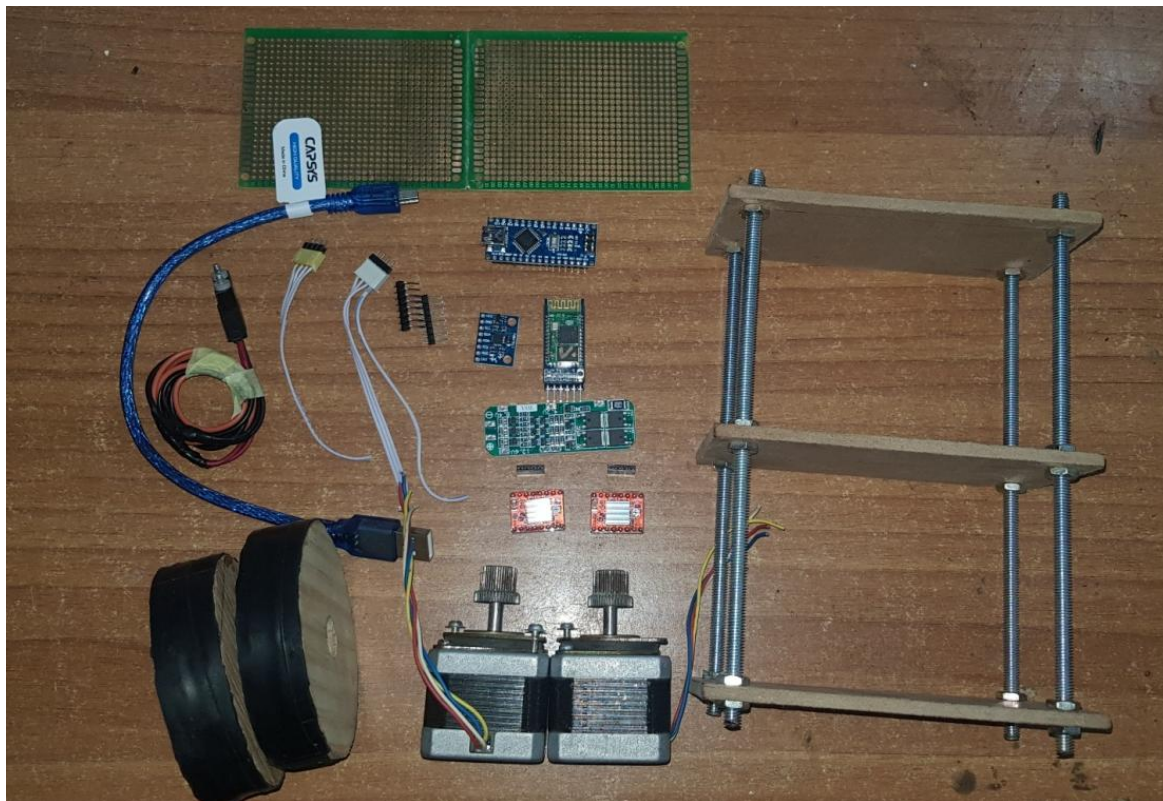
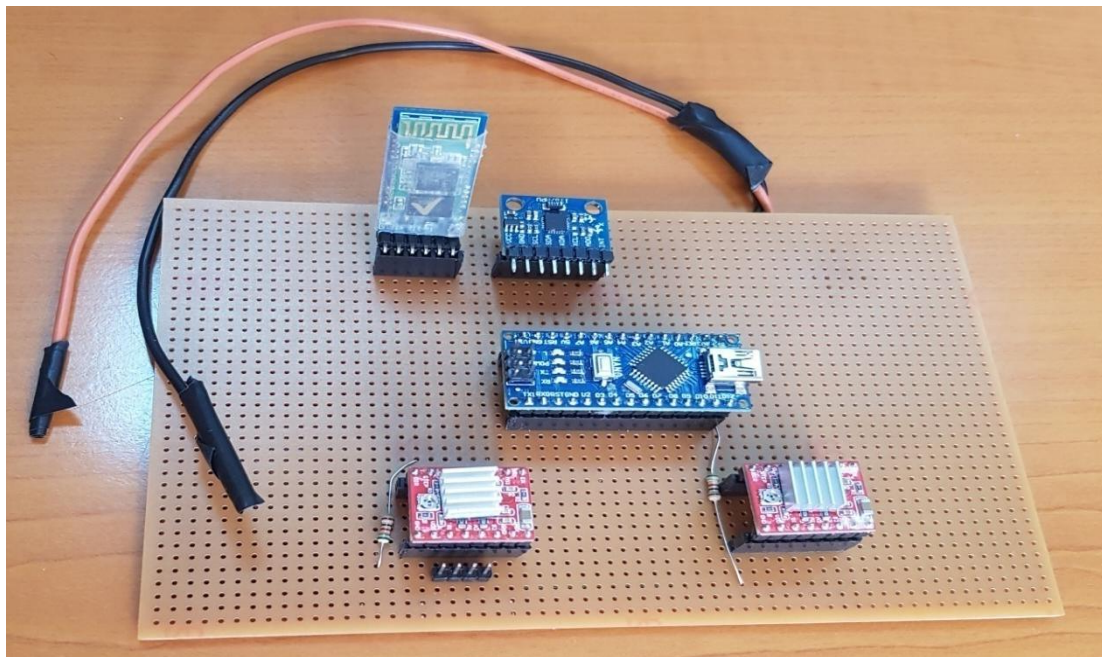
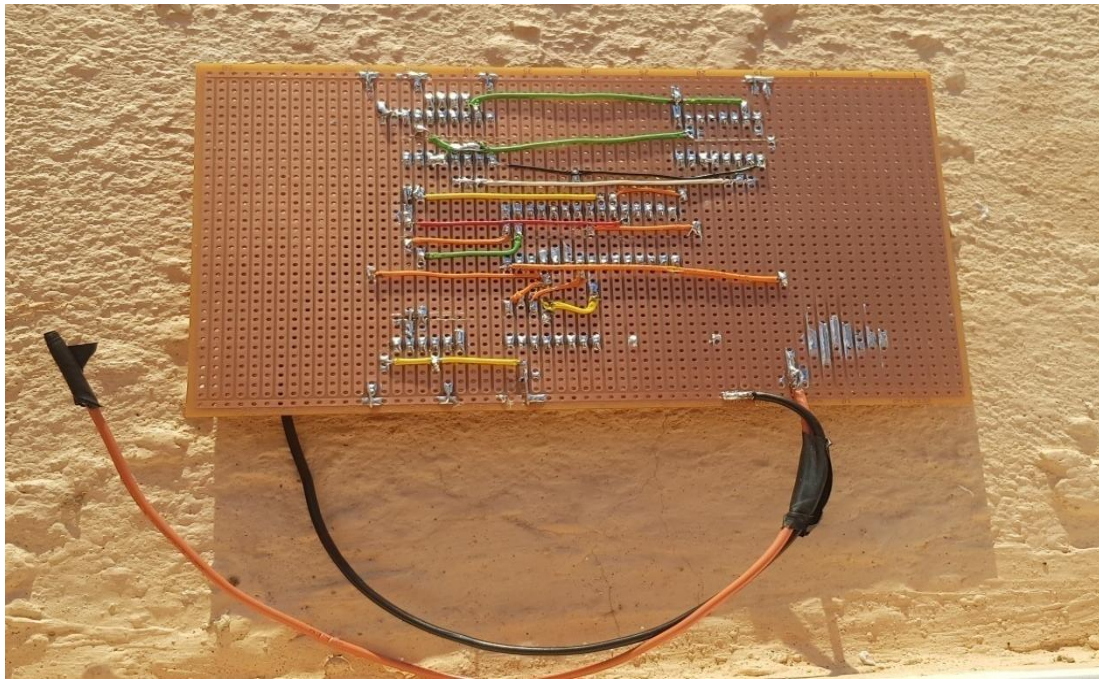


Figure IV.7 : Composant et structure du robot réalisé.

#### IV.6. Raccordements et soudage

- Relier des fils entre la carte Arduino NANO et le module MPU6050.
- Assembler les fils des deux drivers moteurs A4988 et l'Arduino NANO.
- Raccordements du module Bluetooth avec la carte Arduino NANO et l'MPU6050.
- Enfin connecter les fils des moteurs pas à pas avec les deux drivers moteurs et les fils de la batterie vers la carte mère ci- dessous.

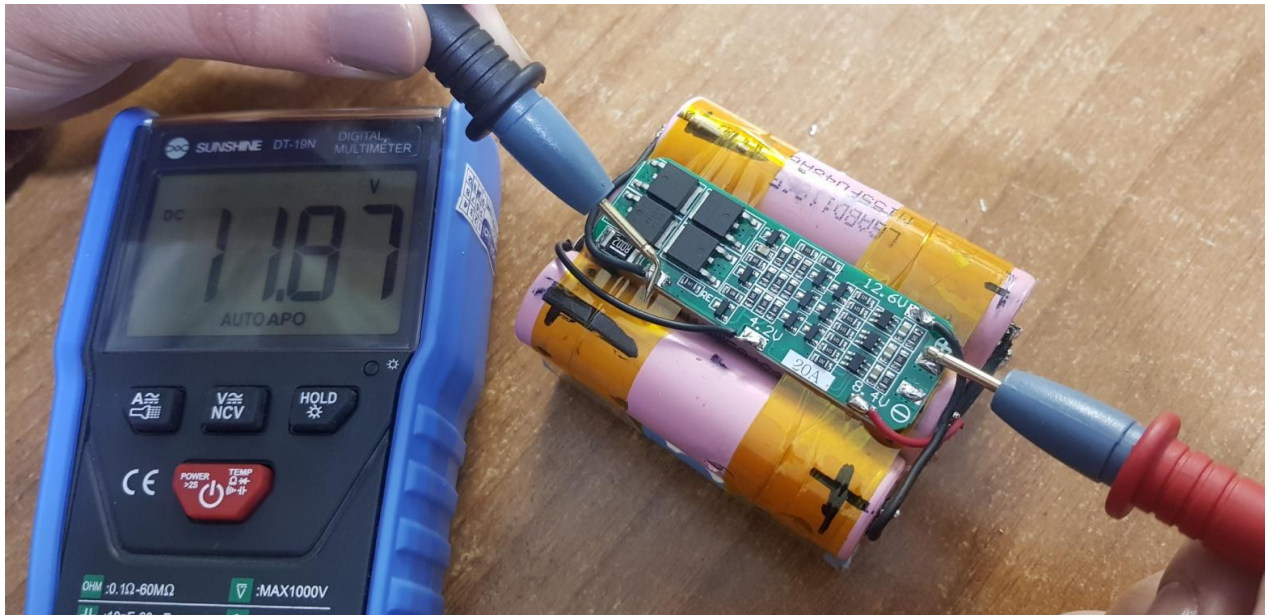


**Figure IV.8 :** raccordements des composants sur la plaque perforée

### **IV.7. Réalisation et positionnement de la batterie**

Pour la réalisation de la batterie, nous avons utilisé des piles (Cellules Lithium) récupérés à partir des batteries des ordinateurs portables. Premièrement Les cellules sont testées et chargées, puis nous avons vérifié leurs capacités. Par la suite, nous avons réalisé et connectée en parallèle deux séries de trois cellules. UN module BMS 3S a été intégrer à la batterie ce qui permet de la protéger et contrôler sa charge-décharge.

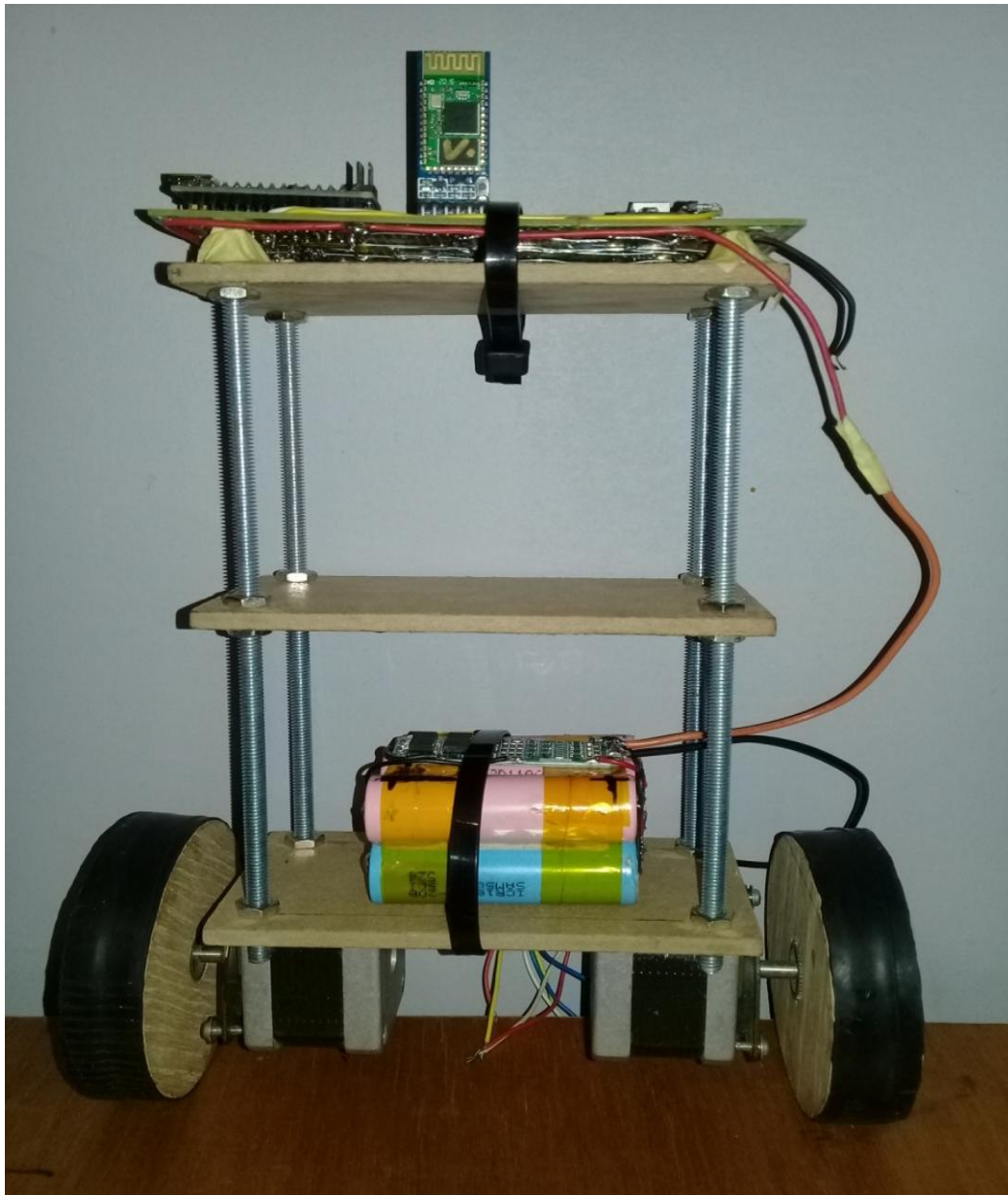
Finalement, la batterie finale a été testée afin de vérifier son voltage total comme la montre la figure si dessous.



**Figure IV.9** : Test de la batterie réalisée.

#### **IV.8. La forme finale du robot**

Le montage final de robot mobile auto balancé réalisé est présenté dans la figure IV.10. Durant la réalisation de projet nous avons rencontré des difficultés mais on a réussi à les surmonter. Notre robot est maintenant prêt à ce balancé après avoir été programmé à l'aide d'un logiciel de programmation nommé Arduino IDE, il fonctionne avec le module Bluetooth commander par une application installée sur le téléphone portable appelée **EZGUI [31]**.



**Figure IV.10** : Photo finale de robot réalisé.

## IV.9. Conclusion

Ce chapitre a été consacré pour la partie programmation et réalisation finale de robot mobile auto balancé. Nous avons présenté les différentes étapes de la réalisation du robot auto balance.

### Conclusion générale

Dans ce présent travail qui s'inscrit dans le cadre de projet de fin d'études, nous avons mis en valeur l'une des principales lignes de recherche actuelle pour la robotique mobile en environnement naturel et qui consiste en : « réalisation et commande d'un robot mobile à roues auto balancé ».

Conformément aux objectifs ciblés d'une part, et vus, les problèmes rencontrés, d'autre part, les résultats obtenus se résument dans la réalisation et la commande électronique, ainsi qu'une première série de tests a été effectuée et montrée le fonctionnement correct du système après une analyse critique et résolution des anomalies localisées. En expérimental, il nous a été plus simple d'obtenir des résultats similaires à l'aide d'un régulateur PID.

La technique du filtre complémentaire a été appliquée avec succès. La dérive du gyroscope a été effectivement éliminée permettant une estimation relativement correcte de l'angle d'inclinaison et de sa dérivée.

Par ailleurs, nous signalons que le travail abordé englobe plusieurs domaines d'étude (la programmation, l'électronique, la gestion de l'énergie, la mécanique). Il nous a permis d'approfondir nos connaissances théoriques et d'améliorer notre niveau et aussi de toucher le côté pratique grâce à la réalisation et la gestion de projet.

Dans la dernière partie qui nous a pris beaucoup de temps et d'efforts, nous avons réalisé notre prototype de robot auto balancier ou nous avons rencontré quelques contraintes mais nous avons réussi à les surmonter en effectuant des améliorations de balancement.

Finalement, nous souhaitons que ce travail soit repris par d'autres étudiants en vue d'améliorer encore plus le travail et de confirmer les résultats aussi bien sur le plan théorique que pratique. En perspective, nous recommandons la prise en compte des points suivants :

- Effectuer des améliorations en intégrant d'autres fonctionnalités de détection de traitement et de vision par exemple un détecteur d'obstacles ou suiveur de trajectoire.
- De nouveaux composants, très compacts et complets ont vu le jour, comme les IMU (Inertia Measurement Units) MEMS et pourront avantageusement remplacer des composants discrets, mal positionnées sur le corps du robot, afin d'augmenter la précision de la mesure de l'angle et des vitesses angulaires et linéaires.

### Références Bibliographiques

- [1] B. Siciliano, O. Khatib, " Springer Handbook of Robotics ", Springer-Verlag, Berlin, ISBN :978-3-540-23957-4, 2008.
- [2] S. Arramaz , H. Larhrir, O. Elfahimi, " Balancing Robot ", Ecole Nationale des Science Appliquees Khoribga, Maroc, 2016.
- [3] A. Kouider, M. Bellila, " Etude et réalisation expérimentale d'une commande destinée à contrôler un pendule inversé ", Mémoire de Master, Université Djilali Bounaama Khemis Miliana, 2017.
- [4] A. KHEMISSAT, " Plateforme de prototypage rapide pour la robotique mobile : Application à un robot Auto-balancé ", Mémoire de Master, Université M'hamed Bougara-Boumerdes, 2016.
- [5] S. Boussaad, S. Bouachrine, " Etude, réalisation et commande d'un robot auto balancé à deux roues", Mémoire de Master, Université Akli Mohand Oulhadj de Bouira, 2018.
- [6] D. Filliat, Robotique mobile, Article en ligne, Publié le 17-10-2016 -- Mis à jour le 15-09-2019 <https://david-filliat.developpez.com/tutoriel/robotique-mobile/.Consulter> le 10/06/2020.
- [7] M. Silanus, " ISN - Robotique", support de cours en ligne, lycée Alphonse Benoit à l'Isle sur la Sorgue (Vaucluse)  
[http://silanus.fr/sin/formationISN/Parcours/Robotique/co/module\\_Robotique\\_8.html](http://silanus.fr/sin/formationISN/Parcours/Robotique/co/module_Robotique_8.html)  
Consulter le 10/06/2020.
- [8] Y. Moualhi, K. Ammour, " Mise en oeuvre de la partie perception d'un Robot Mobile Autonome Joueur De Golf (Eurobot 2006), Memoir d'ingenieur, Universite Saad Dahleb De Blida, 2006.
- [9] S. Lens, "Locomotion d'un robot mobile". Memoir d'ingenieur, Université de liège, 2008.
- [10] B. Assogna, les différentes sortes de robots, Article en ligne, publié le 1-02-2018  
<https://savanturiersdunumerique.wordpress.com/2018/02/01/les-differentes-sortes-de-robots/> consulter le 16/06/2020.
- [11] G. Tulloue, " Le pendule inversé ", Article en ligne, Consulter le 16/06/2020  
[https://www.sciences.univ-nantes.fr/sites/genevieve\\_tulloue/Meca/non\\_galileen/pendule\\_inverse.php](https://www.sciences.univ-nantes.fr/sites/genevieve_tulloue/Meca/non_galileen/pendule_inverse.php)
- [12] G. Gastebois, " Pendule inversé et pendule elliptique", Article en ligne, Consulter le 16/06/2020  
[http://gilbert.gastebois.pagesperso-orange.fr/java/pend\\_inverse/theorie\\_inverse.pdf](http://gilbert.gastebois.pagesperso-orange.fr/java/pend_inverse/theorie_inverse.pdf)

## Références bibliographiques

---

- [13] F. Mudry, " Modélisation et régulation d'un pendule inversé ". support de cours, Ecole d'Ingénieurs du Canton de Vaud, Suisse, 2003.
- [14] A. Mikael, K. Jonas. "Design, construction and verification of a self-balancing vehicle", Chalmers University of Technology, Göteborg, Sweden, 2012.
- [15] F. Lahouazi, " Mise en œuvre d'une stratégie de commande neuro floue Application à un pendule inversé ", thèse de Magister, Université Mouloud Mammeri de Tizi Ouzou, 2011
- [16] D. Benali, " Commande adaptative floue d'un système non-linéaire (application : Pendule inversé) ", These de Magister, Université Mouloud Mammeri de Tizi Ouzou 2015.
- [17] F. Ayeb, "Conception et implémentation d'une commande PID numérique d'un moteur à courant continu ", mémoire de fin d'études, Université Larbi Ben M'hidi D'Oum El Bouagui, 2014
- [18] F. Piette, "Implémenter un PID sans faire de calculs !", Article en ligne, Consulter le 18/06/2020  
<http://www.ferdinandpiette.com/blog/2011/08/implémenter-un-pid-sans-faire-de-calculs/>
- [19] "Implémenter un PID sans calcul", cours en ligne, publier le 03/Oct/2019  
[http://projet.eu.org/pedago/sin/term/6-asservissement\\_arduino.pdf](http://projet.eu.org/pedago/sin/term/6-asservissement_arduino.pdf), Consulter le 14/07/2020
- [20] B. Atmani, "Optimisation des paramètres du Contrôleur PID par Algorithme Génétique Multi-objectifs", Mémoire de Master, Université Abderrahmane Mira de Béjaïa, 2011.
- [21] Cite officiel de la compagnie ARDUINO, Consulter le 05/07/2020  
<https://www.arduino.cc/>
- [22] Cite de ARDUINO France, Consulter le 08/07/2020  
<https://www.arduino-france.com/review/arduino-nano-avantages-inconvenients-utilisation-et-fonctionnement/>
- [23] "L'Arduino dans tous ces états", Article en ligne, publier le : 22 décembre 2015  
<http://arduinooo.blogspot.com/2015/12/gyroscope-pour-arduino-mpu6050.html>
- [24] B. Djedouani. M. L. Ould Amhed Haiballa, "Développement d'un Banc d'essai pour l'étude est le diagnostic des moteurs à pas Hybrides", Mémoire ce Master, Université de Tébessa, 2016.
- [25] Cite officel de la compagnie GOTRONIC  
<https://www.gotronic.fr/blog/articles/moteur-pap/>, Consulter le 15/07/2020
- [26] Y. Oumoussa, N. Rebiai, " Commande à distance d'un robot mobile", Mémoire de Fin d'Etudes, Université Mouloud Mammeri de Tizi Ouzou, 2016.



## Références bibliographiques

---

- [27] <https://www.f-legrand.fr/scidoc/docimg/sciphys/arduino/accelgyro/accelgyro.html>  
consulter le 17/11/2020.
- [28] A. Algoz, B. Asef Hasnain, "A Control System for A 3-Axis Camera Stabilizer",  
Universite Uppsala, 2018.
- [29] <https://Delta-Iot.Com/La-Theorie-Du-Filtre-Complementaire/>
- [30] "Asservissement PID - drone", Article en ligne, publier le 29/10/2019,  
<https://www.firediy.fr/article/asservissement-pid-drone-ch-8?fbclid=IwAR0bwg5mgGVfXDmmgXgNwVvbTWxRUYNm0uVmHOyqF-juLiL1ZS-jroXYjms>
- [31] <https://www.askvina.org/post/balancing-robot-arduino-ez-gui>

## ***Résumé***

Dans ce travail, nous nous intéressons à la réalisation et à la commande d'un robot mobile auto-balancé sur deux roues à base d'une carte **Arduino NANO** utilisée pour la combinaison et l'acquisition de données, et la génération des signaux de commande. Le robot auto balancé est un pendule inversé qui est un système non linéaire sous-actionné instable. Le but de notre travail est de stabiliser notre robot autour de son point d'équilibre et contrôler les roues pour maintenir le robot en position verticale. Pour cela, deux capteurs intégrés sur une puce nommée **MPU6050** sont mis en œuvre pour estimer l'angle d'inclinaison : un gyroscope et un accéléromètre. Pour pallier les inconvénients des capteurs, la technique du filtre complémentaire est adoptée pour donner une meilleure estimation de l'angle d'inclinaison du robot. Une régulation **PID** est utilisée pour améliorer la stabilité du robot. Les deux roues sont connectées avec deux moteurs pas à pas contrôlé par deux **drivers moteurs A4988**.

**Mots clés:** Carte Arduino NANO; pendule inversé; gyroscope MPU6050; Régulation PID; driver A4988; moteur pas à pas.

## ***Abstract***

In this work, we are interested in the realization and control of a self-balancing two wheels' robot based on an Arduino NANO controller for combining and acquiring data, and generating control signals. The self-balancing robot is an inverted pendulum which is an unstable under-actuated non-linear system. The goal of our work is to stabilize our robot around its point of balance and control the wheels to maintain the robot to a vertical position. For this, two sensors integrated on a chip called "MPU6050" are implemented to estimate the tilt angle: a gyroscope and an accelerometer. To overcome the drawbacks of the sensors, the complementary filter technique is adopted to give a better estimate of the tilt angle of the robot. PID regulation is used to improve the stability of the robot. The two wheels are connected with two stepper motors controlled by two A4988 motor drivers.

**Key words:** Arduino NANO board; inverted pendulum; MPU6050 IMU; PID regulation; A4988 Driver; stepper motor.