الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DR. YAHIA FARES DE MEDEA



Faculté de **T**echnologie

Mémoire de Fin d'Études

Présenté pour l'obtention du Diplôme de Master en Génie Électrique

Spécialité : Instrumentation

Etude, conception et réalisation d'un robot mobile auto-balance

---

Présenté par :

EULDJI Riadh

MAMI Mohamed

Proposé et Dirigé par :

Dr. SKENDER Mohamed Rèdha

Promotion 2019

Remerciements

Grace à Dieu le tout puissant et miséricordieux de nous doter de la santé, la volonté, la patience, durant toutes ces longues années.

Nous exprimons nos vifs remerciements également à notre promoteur Dr. SKENDER Mohamed Rédha qui a guidé ce travail; son enthousiasme, ses nombreux conseils, sa patience et son aide efficace pour la réalisation de ce mémoire.

Nos sincères remerciements s'adressent à tous enseignants et M.EULDJI Rafik qui nous ont aidé pour terminer ce travail.

Nous tenons aussi à adresser nos remerciements aux membres de jury Pour nous avoir fait l'honneur et le plaisir d'accepter et juger notre travail.

En fin, nous remercions toutes les personnes qui ont contribué de près ou de loin, au bon déroulement de notre travail et de nos études.

Dédicaces

Je remercie ALLAH le tout puissant de m'avoir donné la force et le courage pour accomplir ce modeste travail.

J'ai l'honneur de dédier ce modeste travail à :

A mes très chers parents qui m'ont soutenu et encouragé durant toute la période de mes études et à qui je leur souhaite une longue et heureuse vie, que dieu les protège et me donne la force pour que je puisse leur rendre un petit peu de leur bienfait et leur bonté.

A mon très cher frère Et ma très chère sœur.

A Mon cher grand père et Ma chère grande mère que je leur souhaite une longue vie.

A toute la famille

Je remercie mon ami et binôme Mohamed.

Et tous mes amis surtout M.A et O.A et T.M

A toute ma promotion d'Instrumentation

Et enfin, à tous ceux qui m'aiment

Riadh

Dédicaces

J'ai l'honneur de dédier ce modeste travail à :

A mes très chers parents qui m'ont soutenu et encouragé durant toute la période de mes études et à qui je leur souhaite une longue et heureuse vie, que dieu les protège et me donne la force pour que je puisse leur rendre un petit peu de leur bienfait et leur bonté.

A mes cher frères, ma sœur et sa petite famille.

A Mon cher grand père et Mes chères grandes mère que je leur souhaite une longue vie.

A mon cher binôme Riadh

A mes chers amis

A toute ma promotion d'Instrumentation

Et enfin, à tous ceux qui j'aime et qui m'aime

Mohamed

ملخص

يتضمن هذا المشروع دراسة و إنشاء الروبوت المتحرك ذاتي التوازن بهدف تحقيق الاستقرار حول نقطة التوازن غيرالمستقر, لذلك قمنا باستعمال اثنين من أجهزة الاستشعار لتقييم زاوية الميل : الجيروسكوب و جهاز قياس التسارع . وبسبب الأخطاء و الضوضاء التي تصدر من هذه الأجهزة و العوامل الخارجية قد تتأثر الحسابات التي يتم جمعها خلال التجربة لذلك قمنا باستعمال مرشح مكمل لمعالجة هذه الأخطاء و إعطاء أفضل تقدير لزاوية الميل, قمنا أيضا بتوظيف متحكم تناسبي, تكاملي, تفاضلي في لوحة الأردوينو لضمان استمرارية توازن الروبوت.

الكلمات المفتاحية : روبوت ذاتي التوازن, أجهزة الاستشعار , فلتر مكمل , متحكم بي اي دي , أردوينو .

Résumé

Ce projet consiste à étudier et réaliser un robot mobile auto-balance. Le but, est de stabiliser ce robot autour de son point d'équilibre instable. Pour cela, deux capteurs sont mis en œuvre pour estimer son angle d'inclinaison ; un gyroscope et un accéléromètre. Les mesures récupérées lors de l'expérimentation sont influencées par des erreurs et des bruits qui proviennent de ces deux capteurs et les effets externes. La technique du filtre complémentaire est adoptée pour traiter ces erreurs et donner une meilleure estimation a l'angle d'inclinaison. nous avons également utilisé un contrôleur PID dans la carte arduino pour assurer la continuité de l'auto-équilibrage.

mots clés : mobile auto-balance, capteurs, filtre complémentaire, contrôleur PID, arduino

Abstract

This project involves studiying and creating a self balancing mobile robot. The goal is to stabilize the robot around its unstable equilibrium point. For that, we used two sensors to estimate its inclination angle; a gyroscope and an accelerometer. The measurements collected during the experiments are influenced by errors and noise coming from these two sensors and external effects. A complementary filter technique is adopted to address these errors and to give a better estimation of the inclination angle, we also used a PID controller in the arduino board to ensure the continuity of the self balancing .

key words : self balancing robot, sensors, complementary filter, PID controller, arduino .

Sommaire

Introduction Generale 1

CHAPITRE I : Généralités sur les robots mobiles

I.1 Introduction.....3

I.2 Les Robots mobiles.....3

I.2.1 Définition d'un robot mobile 3

I.2.2 Les types des robots mobiles 4

I.2.2.1 Robots mobiles à pattes 4

I.2.2.2 Robots mobiles à chenille.....6

I.2.2.3 Robots volants.....7

I.2.2.4 Robots navigants.....8

I.2.2.5 Robots mobiles à roues.....8

I.3 Robot a pendule inversé.....11

I.3.1 définition 11

I.3.2 Intérêt de l'étude d'un pendule inversé 12

I.4 Critère de choix.....13

I.4.1 Les Avantages 13

I.4.2 Les inconvénients 14

I.5 Exemple d'applications d'un robot auto-balancé.....14

I.6 Conclusion.....16

CHAPITRE II : Modélisation et Simulation de la Commande

II.1 Introduction..... 17

II.2 Modélisation 17

II.2.1 Model mathématique du moteur a courent continue..... 17

II.2.2 Modèle dynamique pour un pendule inversé à deux roues 21
II.2.3 Modèle d'état de la dynamique linéaire 27
II.2.4 Fonction de transfert de la dynamique linéaire 30
II.2.5 Problématique de la commande 31
II.3 Commande du robot auto-balancé (pendule inversé) 32
II.3.1 Commandabilité et observabilité 30
II.3.2 Commande du système on boucle fermée 33
II.4 Simulation du la command par LQR 38
II.5 Conclusion 42

CHAPITRE III : Description mécanique et électronique

III.1 Introduction 43
III.2 Partie mécanique 43
III.2.1 Choix des matériaux et passage à l'usinage 44
III.2.2 La construction mécanique 44
III.2.2.1 Les roues 44
III.2.2.2 les moteurs Courant Continu 45
III.2.2.3 Les plaques d'étages 47
III.2.2.4 Les tiges , les écrous et les rondelles 47
III.2.2.5 les pièces du forme L 48
III.2.2.46 Servomoteur MG 995 49
III.3 description de la partie électrique 50
III.3.1 Alimentation 50
III.3.2 Le convertisseur boost 51
III.3.3 MPU6050 52

III.3.3.1 Gyroscope 53
III.3.3.2 Accéléromètre 55
III.3.3.3 Avantages des capteurs MEMS 56
III.3.4 Le contrôleur L298N 57
III.3.5 Le capteur HC-SR04 59
III.3.6 Circuit de command ARDUINO 61
III.4 Conclusion 64

CHAPITRE IV : Implémentation et tests de la commande

IV.1 Introduction 65
IV.2 Implémentation du program sur la carte arduino 66
IV.2.1 Calibration du module MPU6050 66
IV.2.2 Détermination de l'angle d'inclinaison 67
IV.2.3 Acquisition des donnés 70
IV.2.4 Fusionner les résultats avec le filtre complémentaire 71
IV.2.5 L'utilisation du contrôleur PID 72
IV.2.6 Génération des impulsions de commande 74
IV.2.7 Organigramme du programme implémenté 76
IV.3 Performance obtenue 77
IV.4 Conclusion 78
Conclusion Generale... 79

Liste des figures

CHAPITRE I : Généralités sur les robots mobiles

FIGURE I.1 : le premier robot mobile Shakey 3
FIGURE I.2 : robot a une pate 4
FIGURE I.3 : l'évolution du robot bipieds selon Sony 4
FIGUREI.4: le robot 4 pattes par Sony 5
FIGURE I.5 : le robot chien développe par Boston Dynamics 5
FIGURE I.6 : robot mobile hexapode 6
FIGURE I.7 : un mini robot mille pattes 6
FIGURE I.8 : robot mobile a chenille 7
FIGURE I.9 : Robots volants (drones) 7
FIGURE I.10: Robots navigants AUV 8
FIGURE I.11 robot auto-balance unicycle 9
FIGURE I.12 : robot tricycle. 9
FIGURE I.13 : robot mobile de type voiture 10
FIGURE I.14 : robot mobile omnidirectionnel 10
FIGURE I.15 : Les trois cas du pendule inversé 12
FIGURE I.16 : segway pour les handicapés 13
FIGURE I.17 : le segway réalise par Kevin Halsall 15

CHAPITRE II : Modélisation et Simulation de la Commande

FIGURE II.1 : schéma global de la commande 17
FIGURE II.2 : Schémas d'un moteur DC 18
FIGURE II.3 : Schémas corps libre pour les deux roues 21
FIGUREII.4: Schémas corps libre du châssis 24
FIGURE II.5 : Pôles-zéros du système en boucle ouverte 31
FIGURE II.6 : Réponse impulsionnelle du système en boucle ouverte 31
FIGURE II.7 : le filtre complémentaire 38
FIGURE II.8 : Résultat de la variation sur les paramètres de la matrice Q 39

FIGURE II.9 : la position ne converge pas au consigne 39
FIGURE II.10: la réponse impulsionnel par la command LQR 39
FIGURE II.11: la répartition des nouveaux pôles et zéros 40
FIGURE II.12 : le block simulink. 40
FIGURE II.13 : la variation de positon de l'angle selon la consigne 41

CHAPITRE III : Description mécanique et électronique

FIGURE III.1 : robot mobile réalisé 43
FIGURE III.2 : les dimensions des roues 45
FIGURE III.3 : le moteur utilisé et ses accessoires. 45
FIGURE III.4: l'assemblage du moteur avec la roue 46
FIGURE III.5 : les plaque du séparations 47
FIGURE III.6 : le châssis du robot 47
FIGURE III.7 : les tiges, rondelles et les écrous 48
FIGURE III.8 : support du forme L 48
FIGURE III.9 : l'emplacement du servomoteur 48
FIGURE III.10: Les constituants d'un servomoteur 49
FIGURE III.11 : le servomoteur MG 995 High speed avec ses accessoires 49
FIGURE III.12 : Schéma électrique de la réalisation 50
FIGURE III.13 : Alimentation 6V utilisé 51
FIGURE III.14 : l'hacheur boost 52
FIGURE III.15 : capteur MPU6050 52
FIGURE III.16: La microstructure du gyroscope 53
FIGURE III.17 : Gyroscope Piezo Electric 54
FIGURE III.18: La microstructure de l'accéléromètre. 55
FIGURE III.19 : Accéléromètre Piezo Electric 56
FIGURE III.20 : contrôleur L298N 57
FIGURE III.21: les composantes de l298n 57
FIGURE III.22: Schéma de principe de fonctionnement du pont H 58
FIGURE III.23: principe de fonctionnement de hc-rs04 60
FIGURE III.24 : brochage du HC-RS04 60

FIGURE III.25: les entrés et les sorties de ARDUINO UNO 61
FIGURE III.26: l'interface IDE 63

CHAPITRE IV : Implémentation et tests de la commande

FIGURE IV.1 : Schémas synoptique de la commande 65
FIGURE IV.2 : Résultat de la calibration.....67
FIGURE IV.3 : Valeurs retournées par le MPU6050.....67
FIGUREIV.4: le changement du l'Angel par rapport l'axe Z.....68
FIGURE IV.5 : Mode de Fonctionnement I2C.....70
FIGURE IV.6 : Le bus de transfert I2C 71
FIGURE IV.7 : la fusion du capteur par filtre complémentaire.....71
FIGURE IV.8 : Les valeurs finales du paramètre PID 73
FIGURE IV.9 : Correction de l'angle a partir la consigne.....74
FIGURE IV.10: Déclarations des pins L298N 75
FIGURE IV.11**:** Organigramme du pogrome implémente.....76
FIGURE IV.12 Test sur le robot.....77

Liste des tableaux

Tableau I.1 : Comparaison des différents types de robots à roues 11
Tableau II.1 : Les valeurs des paramètres.....28
Tableau II.2 : Tableau récapitulant l'influence d'un PID parallèle 34
Tableau II.3 : Paramètres PID selon Ziegler-Nichols 35
Tableau III.1 : Les pièces utilisées dans la structure mécanique de notre robot 44
Tableau III.2 : Les caractéristiques du moteur utilisé 46
Tableau III.3 : Les caractristiques du L298N 59
Tableau III.4 : caractéristiques du UNO 62

Nomenclature

x	vecteur d'état de processus.
u	fonction d'entrée de commande
I_R	inertie de rotor (kg.m^2).
τ_m	Couple de moteur (Nm) .
τ_e	Couple appliqué (Nm) .
M_w	la masse de la roue (kg) .
m_p	la masse du châssis du robot (kg).
I_w	moment d'inertie des roues (kg.m^2).
I_p	moment d'inertie du châssis du robot (kg.m^2)..
$P_L P_R, H_L H_R$	forces de réaction entre la roue et le châssis(Nm.rad/s).
l	distance entre les centres du roue et le centre de la gravité du robot (m) .
C_L, C_R	couple appliqué aux roues par les moteurs (Nm).
$H_{fR} H_{fL}$	forces de frottement entre la terre et les roues (Nms).
θ_w	angle de rotation des roues (rad).
θ_p	angle de rotation du châssis (rad).
k_m	Constante de couple (Nm/A).
k_e	Constante de force électromotrice (Nm/A).
R	Résistance nominale (ohms).

L	Inductance de rotor (H) .
V_a	Tension appliquée (V) .
$\dot{\theta}_w$	vitesse angulaire de roue (rad/s) .
K_f	Constante de frottement (Nms/rad) .
θ	Position angulaire de l'arbre (rad) .
w	Vitesse angulaire de l'arbre (rad/s) .
α	Accélération angulaire de l'arbre (rad/s^2) .
V_e	tension f.e.m (V) .
i	Courant à travers l'armature (A) .
k_p	Gain proportionnel
k_i	Gain intégrale
k_d	Gain dérivatif
PID	proportional integral drives
LQR	linear quadratic regulator
SP	set point
$I2C$	Inter-Integrated Circuit
PMW	pulse width modulation
$MEMS$	microelectronic microsystem

Introduction Générale

Introduction Générale

Depuis fort longtemps, l'homme rêve de créer des machines intelligentes capables d'effectuer des taches à leur place. Ainsi, les humains auraient plus de temps à consacrer pour leur loisir, ou prendraient moins de temps et de risques pour effectuer des taches dangereuses.

Les récents développements de la microélectronique, notamment les MEMS et les réseaux de capteurs, semblent mener le domaine de la robotique mobile vers des horizons fort prometteurs. En effet, de plus en plus de compagnies privées ou même des particuliers sont a la demande d'appareils mobiles autonomes, que ce soit en sécurité, entretien ou même en loisir.

Le développement de la robotique mobile est synonyme de développement des techniques de commande ; on cherche à intégrer des techniques de plus en plus performantes sur des structures de plus en plus compactes. Les méthodes de navigation sont variées, et les théories se multiplient et s'approfondissent.

Parmi ces défis, le pendule inversé a toujours été l'une des problématiques majeures de tout système dont le centre de gravité se trouve au-dessus de son axe de pivotement ou de rotation. On trouve des exemples d'applications dans de nombreux domaines mais plus particulièrement dans les transports [1].

Au début des années 2000, des véhicules électriques à auto-équilibrage à deux roues ont pris part à la littérature. Jusqu'à présent, ils étaient populaires en tant que transporteur humain dans le secteur automobile et en tant que système important dans les applications robotiques.

Leur problème de stabilité et leur conception en tant que véhicule électrique intelligent rendent le système intéressant dans les environnements universitaires. De nombreuses études sur ce système sont en cours dans les branches du génie électrique, électronique en particulier l'instrumentation, mécanique, mécatronique et informatique dans les universités. Ce système à deux roues s'explique par une interaction entre les technologies robotiques et l'automobile.

À l'avenir, les systèmes à deux roues participeront généralement à la vie quotidienne en tant que transporteur humain. Ces véhicules électriques seront utilisés dans les usines, les centres commerciaux, les aéroports, les transports urbains et des environnements similaires.

Leur grande maniabilité, leur rayon de braquage nul, leur réponse rapide, leurs caractéristiques dimensionnelles et leurs propriétés robotiques rendent ces systèmes utilisables dans les zones étroites.

Les contrôleurs intelligents et les systèmes de capteurs dans les technologies robotiques sont largement utilisés dans les véhicules terrestres. Le transfert de technologie de la robotique aux véhicules augmente. La popularité du véhicule électrique encourage de nouvelles conceptions inspirées par des robots. Il est donc important de se concentrer sur les conceptions et les recherches sur les véhicules électriques. [2]

L'objectif de ce travail est l'étude et la réalisation du robot mobile auto-balance à deux roues motrices. Nous présenterons une généralité sur les robots mobiles avec la modélisation du robot réalisé, par la suite nous porterons une description sur la partie mécanique et les composants utilisés dans la partie électronique. Enfin, nous réaliserons une commande réelle au robot réalisé.

Pour ce faire, dans le premier chapitre, nous aborderons l'état de l'art de la robotique mobile. Nous passerons par les différents type existants ainsi que le pendule inverse qui est le robot auto-balance.

Dans le second chapitre Nous mettrons les équations du mouvement pour un pendule inversé à deux roues et le modèle linéaire pour un moteur à courant continu seront développés ainsi que la représentation du modèle dans l'espace d'état. Nous expliquerons quelques types de commandes, et nous choisissons par la suite la commande par LQR, puis nous réaliserons une simulation par MATLAB/SIMULINK .

Dans le troisième chapitre nous présenterons la partie mécanique et électronique du robot et on terminera par le circuit de commande.

Enfin, dans le dernier chapitre, nous nous intéresserons sa la commande réel du robot en exprimant la connexion de tout les composants avec le protocole de communication et le transfert des données entre eux, et en fin mettre le robot en position horizontale est stable avec le résultât obtenue .

CHAPITRE I

Généralités sur les robots mobiles

I.1 Introduction:

La robotique est un ensemble des disciplines (mécanique, électronique, automatique, informatique), elle se subdivise en deux types : les robots industriels et les robots mobiles, avant d'entrer spécifiquement au travail que nous avons effectué dans la conception de notre système, nous ferons dans ce premier chapitre un tour d'horizon sur les robots mobiles en général et ses différents types, ensuite on s'intéressera particulièrement aux robots mobiles auto-balancés à deux roues (pendule inversé). On terminera par une présentation des avantages et inconvénients et un exemple d'application des robots à pendule inversé.

I.2 Les Robots mobiles :

I.2.1 Définition d'un robot mobile:

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

Cette machine est polyvalente et capable de s'adapter à certaines variations de ses conditions de fonctionnement. Elle est dotée de fonctions de percéption, de décision et d'action. Ainsi, le robot devrait être capable d'effectuer des tâches diverses, de plusieurs manières, et d'accomplir correctement sa tâche, même s'il rencontre de nouvelles situations inattendues. [3].



Figure I.1 : le premier robot mobile Shakey (SRI, 1966-1972)

I.2.2 Les types des robots mobiles :

Il est possible de classifier les robots mobiles de nombreuses manières et selon de nombreux critères tels : le domaine d'application, le degré d'autonomie, le degré d'intelligence, etc. Nous avons préféré les classifier, puisque leur mobilité est l'une de leurs caractéristiques essentielles, selon leur mode de locomotion [4].

On peut classer les robots mobiles en cinq catégories selon leur mode de locomotion [5] :

I.2.2.1 Robots mobiles à pattes : On distingue notamment et selon le nombre des pattes :

a) **Les robots à une patte** : Un robot doté d'une seule patte a été assemblé au Massacheutts Institute de Technologie (MIT). Le robot sautille sur une patte orientable à deux degrés de liberté, dotée d'un actionneur linéaire pour la propulsion verticale. Il n'est pas autonome , il est utilisé pour assister les gens qui n'avaient pas des pieds.



Figure I.2 : robot à une patte

b) **Les robots à deux pattes** : Les robots à deux pattes ou bipèdes sont surtout étudiés au Japon où les progrès dans ce segment sont assez signifiants. le progresse est commencé dans l'année 1986 , Honda et son humanoïde P2, font rêver le monde entier de robots domestiques.



Figure I.3 : l'évolution du robot bipèdes selon Sony

c) **Les robots à quatre pattes :** Encore une fois, c'est au Japon qu'il faut chercher les robots quadrupèdes les plus évolués : le robot de compagnie AIBO (robot chien) de la firme SONY, notamment dans sa seconde génération est assez « vivant ». Il reproduit en effet une large palette de comportements « canins ».



Figure I.4 : le robot 4 pattes par Sony



Figure I.5 : le robot chien développé par Boston Dynamics

d) **Les robots à six ou huit pattes :** Ce sont les robots à pattes les plus répandus à travers le monde. Avec leurs six pattes ils sont statiques mais aussi facile à programmer et mettre en œuvre, chaque patte n'ayant besoin que de deux degrés de liberté.



Figure I.6 : robot mobile hexapode

e) **Les robots « mille-pattes »** : Ces robots sont très utilisés dans l'entretien et la réparation des conduites et autres endroits inaccessibles. La demande professionnelle pour ce type de robots est très importante et de coté scientifiques ont mis au point un point un robot de l'université de Hong Kong miniature capable de se déplacer dans tous les milieux et d'administrer un médicament à un endroit précis de l'organisme.



Figure I.7 : un mini robot mille pattes

I.2.2.2 Robots mobiles à chenille : Les chenilles assurent à un mobile une meilleure adhérence au sol. Elles sont employées lorsque le sol est perturbé, essentiellement en environnement extérieur. La commande est de type différentiel (differential drive). On peut citer le robot ANT de la West Virginia University.

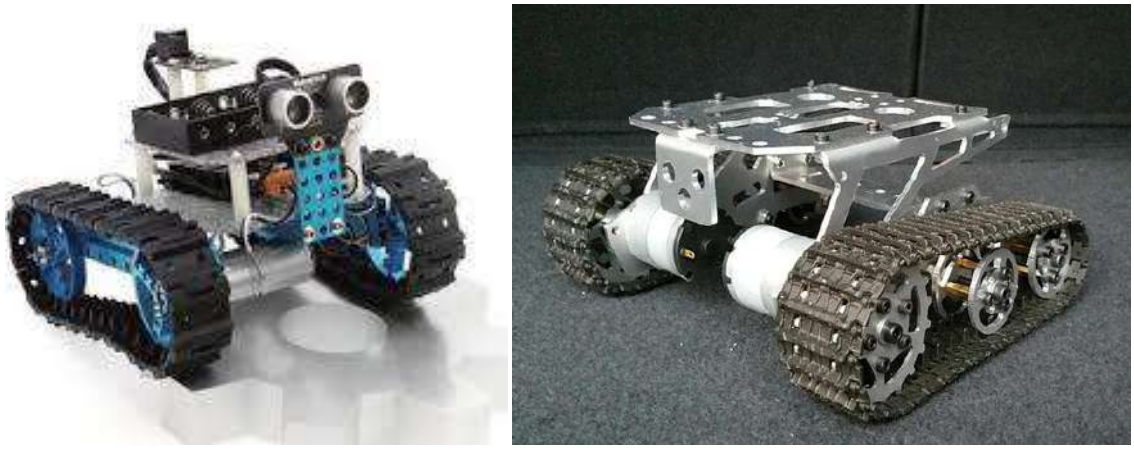


Figure I.8 : robot mobile à chenille

I.2.2.3 Robots volants (drones) : Les drones, petits avions autoguidés ou téléguidés ont de nombreuses applications militaires et éventuellement civiles (surveillance d'autoroutes). Des projets visent à réaliser un avion autonome de moins de 20cm d'envergure, transportant une caméra infrarouge et un transmetteur radio. Il est plus facile de faire voler un avion de 50cm d'envergure ne pesant que quelques grammes (show flyer) ; ces avions indoor sont délicats à construire et fragiles à manipuler mais ils résistent aux obstacles, à cause de leur faible inertie. Le vol stationnaire est encore plus facile à maitriser. Le concours des robots volants organisé chaque année à Atlanta en Georgie montre combien il est difficile de stabiliser en vol un hélicoptère et de commander ses déplacements avec assez de précision.



Figure I.9 : Robots volants (drones)

I.2.2.4 Robots navigants : La construction d'un robot bateau ou sous-marin est facilitée par le fait que le poids n'est pas un facteur limitatif à la mobilité. Les torpilles sont bien connues des militaires, mais il y a peu de recherches civiles qui ont été lancées sur les robots mobile sous-marins appelés AUV (Autonomous Underwater Vehicle). Ces derniers sont notamment utilisés dans le secteur pétrolier (off-shore) et dans l'exploration sous-marine.



Figure I.10 : Robots navigants AUV

I.2.2.5 Robots mobiles à roues :

La commande se fait par la motorisation des roues installées. on peut distinguer quatre modèles des robots mobiles à roues [6]:

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

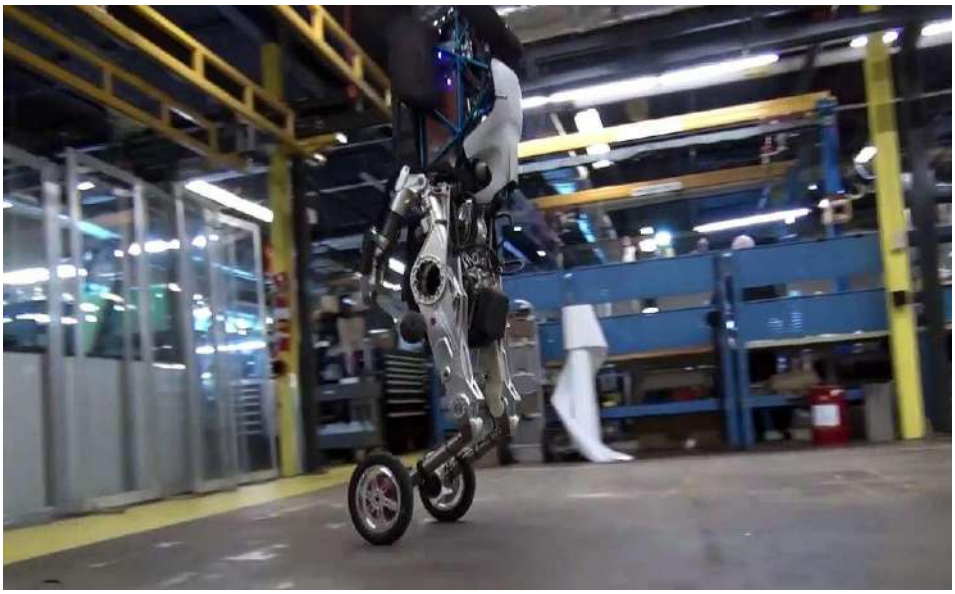


Figure I.11 : robot auto-balance unicycle

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



Figure I.12 : robot tricycle

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

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

Toutes les autres propriétés du robot voiture sont identiques au robot tricycle, le deuxièmes 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.



Figure I.13 : robot mobile de type voiture

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

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



Figure I.14 : robot mobile omnidirectionnel

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

Tableau 1.1 : Comparaison des différents types de robots à roues :

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

D'âpres le tableau **I.1** l'application de chaque type de robot est différents par rapport au autres car chaque type avis des inconvénients et des avantages, ca nous donne que il n'y a pas un type simple a appliqué.

I.3 Robot à pendule inversé:

1.3.1 définition : Un pendule inversé est un pendule dont le centre de gravité est au dessus de son point de pivotement. Il est souvent mis en œuvre avec le point de pivot monté sur un chariot qui peut se déplacer horizontalement. Alors qu'un pendule normal est stable lorsqu'il est accroché à un point, un pendule inversé est intrinsèquement instable, et doit être activement équilibré afin de rester debout, soit en appliquant un couple au point de pivot ou en déplaçant le point de pivot à l'horizontale dans le cadre d'un système de rétroaction.

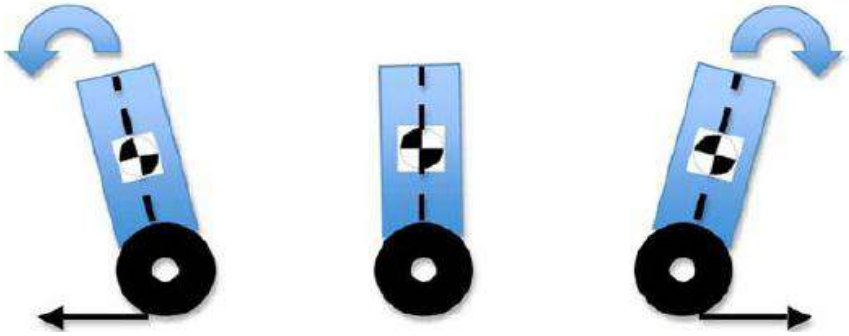


Figure 1.15 - Les trois cas du pendule inversé

Le Robot pendule inversé est un système mécatronique consistant en un pendule inversé mobile. Ce système présente plusieurs défis dont le fait que celui-ci est un système instable et requiert donc un correcteur pour le stabiliser en le maintenant en équilibre ,cela signifie que le robot est libre de tomber en avant ou en arrière sans aucunes forces appliquées. Le Robot est en équilibre lorsque son centre de gravité et les roues sont situées sur une ligne verticale imaginaire identique. Sinon, les roues devraient suivre la chute du robot jusqu'à ce qu'il se remet en équilibre. L'utilisation de deux roues seulement permet d'avoir un poids plus léger et une manœuvre plus lisse. La figure (I.16) illustre le fonctionnement de base d'un robot auto-équilibré [7].

I.3.2 Intérêt de l'étude d'un pendule inversé:

L'étude du pendule inversé a assez bien d'importance. Il y a plusieurs raisons à cela, l'homme est un pendule triple inversé où les chevilles, genoux et hanches représentent les axes de rotations, quand nous sommes en position debout, nos articulations travaillent sans arrêt pour nous y maintenir. Les spécialistes qui travaillent à la réalisation de prothèses pour les hanches sont amenés à utiliser le modèle du pendule double inversé pour calculer l'ensemble des contraintes auxquelles est soumise la prothèse. Dans le même ordre d'idée, la robotique utilise ce genre de concept. 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 ralenti en se penchant en arrière. Le système est le même que le pendule inversé. La stabilité est aussi assurée par des gyroscopes qu'on va la bien détailler dans ce projet. Les étudiants en instrumentation ,automatique, et en robotique font des laboratoires de recherches avec ce genre de dispositif. Parce qu'ils font intervenir beaucoup de notions intéressantes pour eux : Programmation, électronique, automatisation, mécanique ,…etc.

I.4 Critère de choix :

I.4.1 Les Avantages : À l'avenir, les systèmes à deux roues participeront généralement à la vie quotidienne en tant que transporteur humain. Ces véhicules électriques seront utilisés dans les usines, les centres commerciaux, les aéroports, les transports urbains et des environnements similaires. Leur grande maniabilité, leur rayon de braquage nul, leur réponse rapide, leurs caractéristiques dimensionnelles et leurs propriétés robotiques rendent ces systèmes utilisables dans les zones étroites.

Accroissement de la capacité de production par remplacement du l'homme dans l'exécution des taches pénibles ou dangereuses .Compte tenu de la simplicité du mécanisme de locomotion utilisé et aussi le travaille continuellement, sans ressentir une fatigue ou l'ennui, et ne nécessitent pas une assurance médicale ou de vacances



Figure 1.17 : segway pour les handicapés

I.4.2 Les inconvénients :

On distingue sans trop d'ambiguïté un certain nombre de problèmes en robotique mobile en général. Bien évidemment, l'aspect matériel, qui consiste à choisir et dimensionner aussi bien la structure mécanique du système que sa motorisation, son alimentation et l'architecture informatique de son système de contrôle-commande apparaît comme le premier point à traiter.

Le choix de la structure est souvent effectué parmi un panel de solutions connues et pour lesquelles on a déjà résolu les problèmes de modélisation, planification et commande. Le choix des actionneurs et de leur alimentation est généralement assez traditionnel. De la même façon, les architectures de commande des robots mobiles ne sont pas différentes de celles des systèmes automatiques ou robotiques plus classiques. On y distingue cependant, dans le cas général, deux niveaux de spécialisation, propres aux systèmes autonomes : une couche décisionnelle, qui a en charge de la planification et la gestion (séquentielle, temporelle) des évènements et une couche fonctionnelle, chargée de la génération en temps réel des commandes des actionneurs. Bien évidemment, l'architecture du robot dépend fortement de l'offre et des choix technologiques du moment. Les problèmes spécifiques à la robotique mobile n'apparaissent finalement que lorsque l'on dispose d'une structure mobile dont on sait actionner les roues. Tous les efforts du roboticien vont alors consister à mettre en place les outils permettant de faire évoluer le robot dans son environnement de manière satisfaisante, qu'il s'agisse de suivre un chemin connu ou au contraire d'aller d'un point à un autre en réagissant à une modification de l'environnement ou à la présence d'un obstacle [8].

I.5 Exemple d'applications d'un robot auto-balance :

Une des applications du robot mobile auto-balance dans le domaine médical est le segway pour les handicapés, Les personnes en situation de handicap pourraient connaître une petite révolution technologique, grâce à cet inventeur qui a eu l'idée de désosser un Segway pour en faire un fauteuil roulant. Ce dernier se dirige grâce aux muscles des abdominaux. Cette nouvelle génération de fauteuil permettrait à ses propriétaires de se déplacer en ayant les deux mains libres. Un pas en avant, notamment dans le monde du handisport.

Baptisé l'Ogo, ce prototype pour le moment unique a été construit pour la première fois par **Kevin Halsall** il y a 4 ans pour son ami paraplégique **Marcus Thompson** à partir d'un Segway démantelé. Cette 3ème version pouvant atteindre les 20 km/h, fabriquée à la

main et en fibre de verre pourrait être bientôt prête pour la vente. Dans sa publication, Halsall confie à propos de sa première expérience en Segway que « la première chose que j'ai pensé a été 'si je n'avais plus de jambes [le Segway] serait la chose parfaite que j'aimerais adapter ». Le fauteuil roulant électrique dispose de larges roues tout-terrain et permet au pilote de manipuler le véhicule par l'intermédiaire du siège (breveté) en utilisant ses muscles abdominaux. [9]



Figure 1.18 : le segway réalisé par Kevin Halsall .

I.6 Conclusion :

L'objectif de notre étude consiste principalement à l'étude et à la réalisation d'un robot mobile auto-balance, pour cela nous avons présenté dans ce chapitre des généralités importantes sur les robots mobiles et ses types, nous avons vu aussi les robots équilibristes qu'il est le cas d'application de notre projet et leurs intérêts d'étude. Dans le chapitre suivant on s'intéressera à la commande (modélisation et simulation) du robot mobile, ce qui fera l'objet en détail.

CHAPITRE II

Modélisation et Simulation de la Commande

II.1 Introduction:

Afin de pouvoir concevoir un correcteur capable de stabiliser le robot , il est nécessaire dans un premier temps de mettre au point un modèle mathématique qui représente fidèlement son comportement et contrôlé le système par LQR et la fusion des donnes par le filtre de kalman avec la simulation nécessaire.[10]

Dans ce chapitre, les équations du mouvement pour un pendule inversé à deux roues et le modèle linéaire pour un moteur à courant continu sont détaillées ainsi la commande par Matlab et les fusion des donnes avec les résultats trouves.

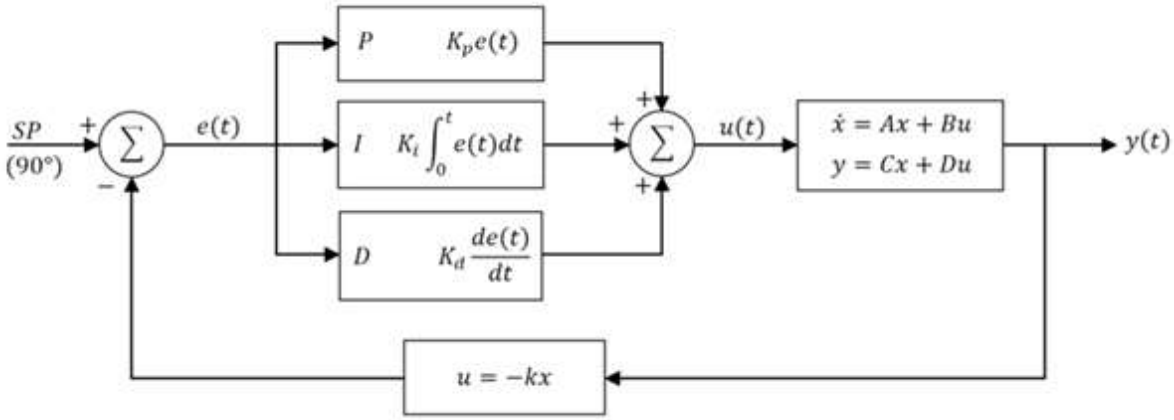


Figure II.1 schéma global de la commande

II.2. Modélisation:

II.2.1. Model mathématique du moteur a courent continue:

Le robot est mobile grâce à deux moteurs à courant continu (DC).

Dans cette section, le modèle d'espace d'état du moteur à courant continu est décrit. ce modèle est ensuite utilisé dans le modèle dynamique du robot équilibriste pour fournir une relation entre la tension d'entrée aux moteurs et le couple de contrôle nécessaire pour équilibrer le robot.

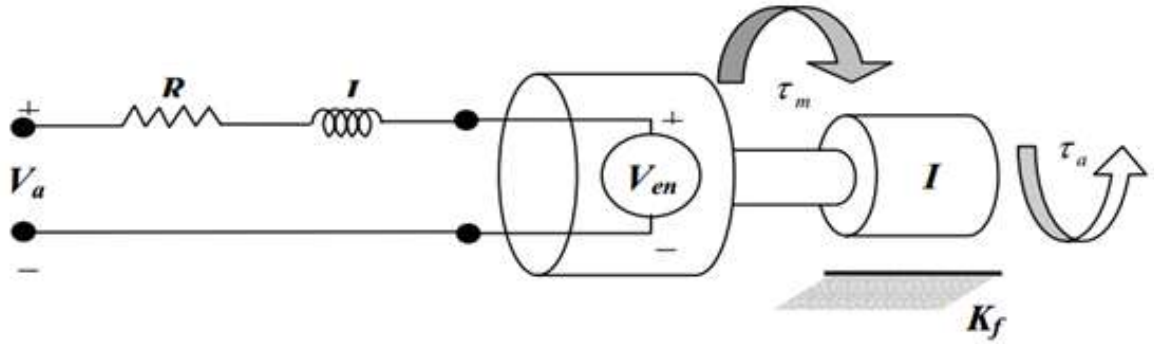


Figure II.2- Schémas d'un moteur DC

La Figure (II.2) illustre un modèle linéaire efficace pour un moteur à courant continu. Lorsqu'une tension est appliquée aux bornes du moteur, un courant « **i** » est généré dans l'armature du moteur. Le moteur produit un couple "$\boldsymbol{\tau_m}$" , qui est proportionnel au courant. Cette relation peut être exprimée par :

$$\boldsymbol{\tau_m = k_m i} \qquad \textbf{(II.1)}$$

Une paire de résistance- inductance en série avec une tension Va, peut être utilisé pour modéliser , le circuit électrique du moteur. Une force électromotrice est produite parce que les bobines du moteur sont traversées par un champ magnétique.[9] La tension produite peut être approximée par une fonction linéaire de la vitesse de l'arbre :

$$V_e = k_e \omega \qquad \textbf{(II.2)}$$

Avec :

V_e	tension f.e.m
K_e	Constante de force électromotrice
$\boldsymbol{w}$	Vitesse angulaire de l'arbre

À ce point, une équation différentielle linéaire pour le circuit électrique du moteur à courant continu est déduite en utilisant la loi des tensions de « Kirchhoff ». [11]

Cette loi indique que la somme de toutes les tensions dans le circuit doit être égale à zéro. Pour le moteur à courant continu, cela peut s'écrire comme :

$$\mathrm{V_a} - \mathrm{Ri} - \mathrm{L}\frac{\mathrm{di}}{\mathrm{dt}} - \mathrm{V_e} = 0 \qquad \textbf{(II.3)}$$

Pour décrire l'équation de mouvement pour le moteur, le frottement sur l'arbre du moteur est approximée par une fonction linéaire de la vitesse de l'arbre.

L'approximation est effectuée pour que le frottement sur l'arbre du moteur *kf* soit une fonction linéaire de la vitesse de l'arbre.

La loi de Newton sur le mouvement indique que la somme de tous les couples produit sur l'arbre est liée linéairement à l'accélération de l'arbre par la force d'inertie de l'induit I_R.[11]

Soit l'équation:

$$\sum \mathrm{M} = \tau_\mathrm{m} - \mathrm{k_f}\omega - \tau_\mathrm{a} = \mathrm{I_R}\dot{\omega} \qquad \textbf{(II.4)}$$

Avec:

τ_m	Couple de moteur
τ_a	Couple applique
$\mathrm{I_R}$	inertie de rotor
$\dot{\omega}$	Vitesse angulaire de l'arbre
$\mathrm{k_f}$	Constante de frottement
ω	angle de l'arbre
K_m	Constante de couple

En introduisant les équations (II.1) ,(II.2) dans les équations (II.3) , (II.4) puis en réarrangeant les termes des dérivés, on obtient les deux équations fondamentales suivantes et qui résument le mouvement du moteur.

$$\frac{di}{dt} = +\frac{V_a}{L} + \frac{\mathrm{R}}{L}i + \frac{\mathrm{k_e}}{L}\omega \qquad \textbf{(II.5)}$$

$$\frac{d\omega}{dt} = +\frac{k_m}{I_R}i + \frac{\mathrm{k_f}}{I_R}\omega + \frac{\tau_\mathrm{a}}{I_R} \qquad \textbf{(II.6)}$$

Les deux équations sont des fonctions linéaires du courant et de la vitesse et elles contiennent des dérivés du premier ordre, l'inductance du moteur est prise nulle et les frottements du moteur sont négligeables pour simplifié le model du moteur à courant continu qui est suffusion pour l'équilibre du robot.[11]

Par conséquent, (II.5) et (II.6) deviennent :

$$\mathrm{i} = \frac{\mathrm{V_a}}{\mathrm{R}} - \frac{\mathrm{k_e}}{\mathrm{R}}\omega \qquad \textbf{(II.7)}$$

$$\frac{d\omega}{\mathrm{dt}} = \frac{k_m}{I_R}\, i - \frac{\tau_\mathrm{a}}{I_R} \qquad \textbf{(II.8)}$$

En remplacent l'équation (II.7) dans l'équation (II.8),une relation approximative du moteur a courant continu est obtenu en fonction de la vitesse actuelle du moteur ,la tension appliquée est le couple comme :

$$\frac{\mathrm{d}\omega}{\mathrm{dt}} = \frac{\mathrm{k_m k_a}}{\mathrm{I_R R}} - \frac{\mathrm{k_m V_e}}{\mathrm{I_R R}}\omega - \frac{\tau_\mathrm{a}}{\mathrm{I_R}} \qquad \textbf{(II.9)}$$

Etant donné que l'inductance du moteur est négligée, le courant à travers les enroulements n'est pas pris en compte dans l'équation de mouvement du moteur. Le courant atteint son état constant par rapport à la vitesse de l'arbre, qui va prendre beaucoup plus de temps pour arriver à la vitesse finale lors d'un changement de la tension d'entrée.

La dynamique du moteur peut être représentée par un modèle d'espace d'état. C'est un système d'équations différentielles de premier ordre avec les paramètres position, θ, et vitesse, ω, qui représentent uniquement le fonctionnement. Les entrées du moteur sont alors la tension appliquée et le couple appliqué.[11]

$$\begin{bmatrix}\dot{\theta}\\ \dot{\omega}\end{bmatrix}\begin{bmatrix}0 & 0\\ 0 & -\frac{\mathrm{k_m k_e}}{\mathrm{I_R R}}\end{bmatrix}\begin{bmatrix}\theta\\ \omega\end{bmatrix}\begin{bmatrix}0 & 0\\ \frac{\mathrm{k_m}}{\mathrm{I_R R}} & \frac{-1}{\mathrm{I_R}}\end{bmatrix}\begin{bmatrix}\mathrm{V_a}\\ \tau_\mathrm{a}\end{bmatrix} \qquad \textbf{(II.10)}$$

$$\mathrm{y} = [0 \quad 1]\begin{bmatrix}\theta\\ \omega\end{bmatrix} + [0 \quad 0]\begin{bmatrix}\mathrm{V_a}\\ \tau_\mathrm{a}\end{bmatrix} \qquad \textbf{(II.11)}$$

II.2. 2 Modèle dynamique pour un pendule inversé à deux roues:

Le Pendule inversé à deux roues, a un comportement similaire avec un pendule sur un chariot. la dynamique des roues et le pendule sont analysés séparément au début, mais cela finira par deux équations de mouvement qui décrivent complètement le comportement du robot équilibriste.[12]

Comme le comportement du robot peut être influencé par des perturbations ainsi que le couple de rotation de moteur, le modèle mathématique devra tenir compte de ces forces. premièrement on obtient les équations de mouvement associé aux roues gauche et droite la figure (II.3) ci-dessous montre le diagramme de corps libre pour les deux roues. Etant donné que les équations des roues gauche et droite sont tout à fait analogues, seule l'équation de la roue droite est donnée.[12]

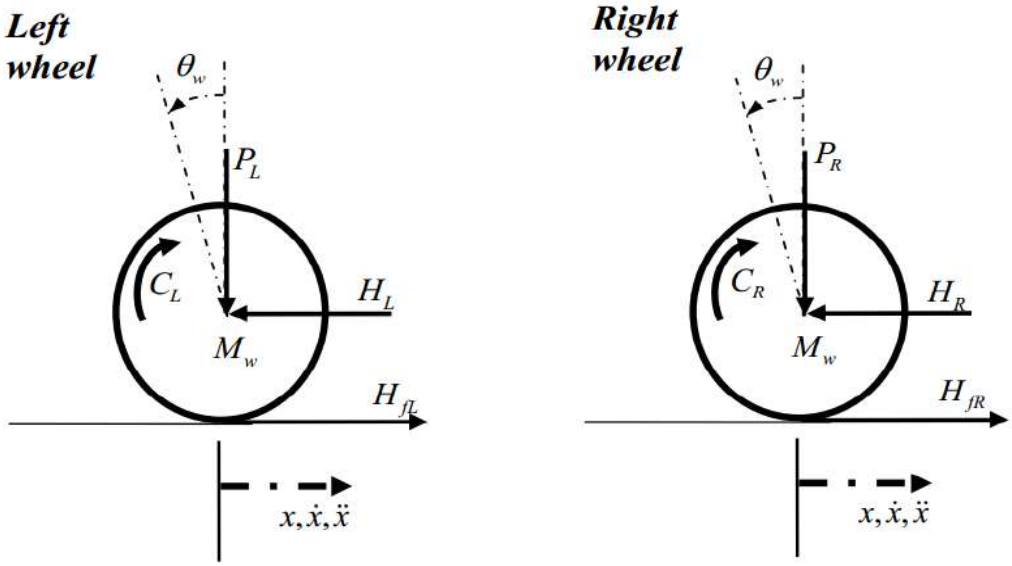


Figure II.3 : Schémas corps libre pour les deux roues

Utilisant la loi de mouvement de Newton, la somme des forces sur la direction x est :

$$\sum \mathrm{F_x} = \mathrm{Ma}$$

$$M_w \ddot{x} = H_{fR} - H_R \quad \textbf{(II.12)}$$

La Somme des forces autour du centre de la roue donne :

$$\sum M_0 = Ia$$

$$I_w \ddot{\theta}_w = C_R - H_{fR} r \quad \textbf{(II.13)}$$

À partir de la dynamique du moteur à courant continu, le couple du moteur peut être exprimé sous la forme :

$$\tau_m = I_R \frac{d\omega}{dt} + \tau_a \quad \textbf{(II.14)}$$

En réarrangeant l'équation et remplacent par les paramètres du moteur à courant continu, le couple de sortie aux roues est atteinte :

$$C = I_R \frac{d\omega}{dt} = \frac{-k_m k_e}{R} \dot{\theta}_w + \frac{k_m}{R} V_a \quad \textbf{(II.15)}$$

Par conséquent, l'équation (II.13) devient:

$$I_w \ddot{\theta}_w = \frac{-k_m k_e}{R} \dot{\theta}_w + \frac{k_m}{R} V_a - H_{fR} r \quad \textbf{(II.16)}$$

Ainsi,

$$H_{fR} = \frac{-k_m k_e}{Rr} \dot{\theta}_w + \frac{k_m}{Rr} V_a - \frac{I_w}{r} \ddot{\theta}_w \quad \textbf{(II.17)}$$

Avec :

H_{fR}	forces de frottement entre la terre et les roues.
C_R	couple appliqué aux roues par les moteurs
I	Les inerties
r	Rayon des roues
M_w	la masse de la roue

L'équation (II.15) est remplacée dans (II.12) pour obtenir l'équation des roues gauche et droite.

- **Pour la roue gauche:**

$$M_w \ddot{x} = \frac{-k_m k_e}{Rr} \dot{\theta}_w + \frac{k_m}{Rr} V_a - \frac{I_w}{r} \ddot{\theta_w} - H_L \quad \textbf{(II.18)}$$

- **Pour la roue droite:**

$$M_w \ddot{x} = \frac{-k_m k_e}{Rr} \dot{\theta}_w + \frac{k_m}{Rr} V_a - \frac{I_w}{r} \ddot{\theta_w} - H_R \quad \textbf{(II.19)}$$

Avec

H_R et H_L	forces de réaction entre la roue et le châssis
I_w	moment d'inertie des roues
$\dot{\theta}_w$ et $\ddot{\theta_w}$	vitesse et accélération angulaire de roue

Parce que le mouvement linéaire agit sur le centre de la roue, la rotation angulaire peut être transformée en mouvement linéaire par une simple transformation :

$$\ddot{\theta_w} r = \ddot{x} \rightarrow \ddot{\theta_w} = \frac{\ddot{x}}{r}$$

$$\dot{\theta}_w r = \dot{x} \rightarrow \dot{\theta}_w = \frac{\dot{x}}{r}$$

Par la transformation linéaire, l'équation (II.18) et (II.19) devient :

- **Pour la roue gauche:**

$$M_w \ddot{x} = \frac{-k_m k_e}{Rr^2} \dot{x} + \frac{k_m}{Rr} V_a - \frac{I_w}{r^2} \ddot{x} - H_L \quad \textbf{(II.20)}$$

- **pour la roue droite**

$$M_w \ddot{x} = \frac{-k_m k_e}{Rr^2} \dot{x} + \frac{k_m}{Rr} V_a - \frac{I_w}{r^2} \ddot{x} - H_R \quad \textbf{(II.21)}$$

Ajoutant le équation (II.20) (II.21)

$$(M_w + \frac{I_W}{r^2})\ddot{x} = \frac{-2k_m k_e}{Rr^2}\dot{x} + \frac{2k_m}{Rr}V_a - (H_L + H_R) \quad \textbf{(II.22)}$$

Le châssis du robot peut être modélisé comme un pendule inversé. La figure (II.4) montre le schéma du corps libre du châssis.[12]

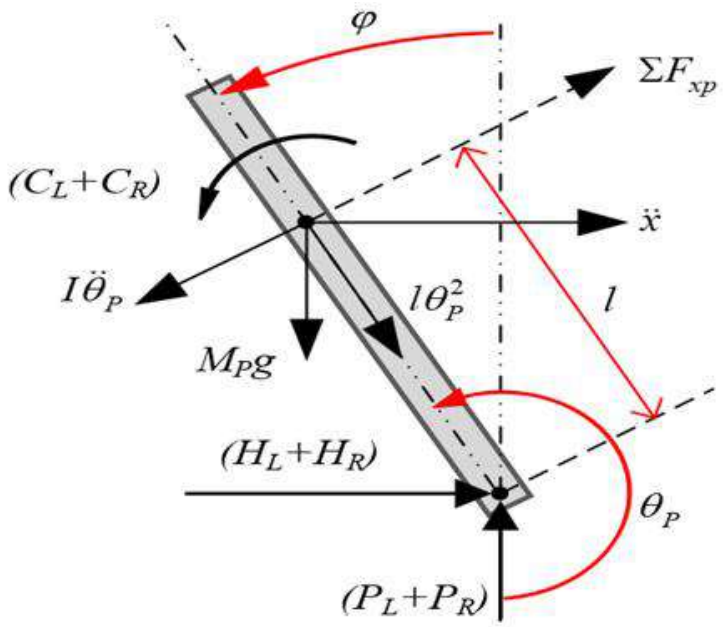


Figure II.4 : Schémas corps libre du châssis

Encore une fois, en utilisant la loi du mouvement de Newton, la somme des forces dans le sens horizontal est :

$$\sum F_x = M_p\ddot{x}$$

$$(H_L + H_R) - M_P l\ddot{\theta}_P \cos\theta_P + M_P l\dot{\theta}_P^2 \sin\theta_p = M_p\ddot{x} \quad \textbf{(II.23)}$$

Ainsi,

$$(H_L + H_R) = M_p\ddot{x} + M_P\, l\, \ddot{\theta}_P\, \cos\theta_P - M_P\, l\dot{\theta}_P^2\, \sin\theta_p \quad \textbf{(II.24)}$$

La somme des forces perpendiculaires au pendule:

$$\sum F_{xp} = M_p\ddot{x}\cos\theta_P$$

$$(H_L + H_R)\cos\theta_P + (P_L + P_R)\sin\theta - M_P g \sin\theta_P - M_P g \sin\theta_P - M_P l\ddot{\theta_P} = M_p \ddot{x}\cos\theta_p \quad \text{(II.25)}$$

Somme des moments autour du centre de masse du pendule

$$\sum M_0 = Ia$$

$$-(H_L + H_R)l\cos\theta_p - (P_L + P_R)\,l\sin\theta_P - (C_L + C_R) = I_p\ddot{\theta}_p \quad \textbf{(II.26)}$$

Avec :

$H_L et\ H_R$	forces de réaction entre la roue et le châssis
$P_L et\ P_R$	forces de réaction entre la roue et le châssis
$C_L et\ C_R$	couple appliqué aux roues par les moteurs
I_p	moment d'inertie du châssis du robot.

Le couple, appliqué sur le pendule à partir du moteur tel qu'il est définie dans l'équation (II.15),après une transformation linéaire, est :

$$\mathrm{C_L} + \mathrm{C_R} = \frac{-2\mathrm{k_m}\,\mathrm{k_e}}{\mathrm{R}}\frac{\dot{\mathrm{x}}}{\mathrm{r}} + \frac{2\mathrm{K_m}}{\mathrm{R}}\mathrm{V_a}$$

Remplacer ceci dans l'équation (II.26) donne:

$$-(H_L + H_R)l\cos\theta_p - (R_L + P_R)l\sin\theta_P - \left(\frac{-2K_m K_e}{R}\frac{\dot{x}}{r} + \frac{2K_m}{R}V_a\right) = I_P\ddot{\theta}_p$$

Ainsi,

$$-(H_L + H_R)l\cos\theta_p - (P_L + P_R)l\sin\theta_P = I_P\ddot{\theta}_p - \frac{-2K_m K_e\dot{x}}{Rr} + \frac{2K_m}{R}V_a \quad \textbf{(II.27)}$$

Multiplier l'équation (II.25) par -*l*,

$$-[(H_L + H_R)l\cos\theta_p + (P_L + P_R)l\sin\theta_P] + M_P g l\sin\theta_P + M_P l^2\ddot{\theta}_P =$$

$$-M_P l\ddot{x}\cos\theta_P \quad \textbf{(II.28)}$$

L'équation (II.27) est remplacée dans l'équation (II.28),

$$I_P\ddot{\theta}_P - \frac{2k_e k_m}{Rr}\dot{x} + \frac{2k_m}{R}V_a + M_p gl\sin\theta_p + M_p l^2\ddot{\theta}_p = -M_p l\ddot{x}\cos\theta_p \quad \textbf{(II.29)}$$

Pour éliminer (H L + H R) de la dynamique du moteur, l'équation (II.24) est remplacée dans l'équation (II.22)

$$2\left(M_w + \frac{I_w}{r^2}\right)\ddot{x} = \frac{-2k_m k_e}{Rr^2}x + \frac{2K_m}{Rr}\dot{V}_a - M_p\ddot{x} - M_p l\ddot{\theta}_p\cos\theta_p + M_p l\dot{\theta}^2\sin\theta_p \quad \textbf{(II.30)}$$

Réorganisant les équations (II.29) et (II.30) donne les équations non - linéaires de mouvement du système :

$$\left(I_p + M_p l^2\right)\ddot{\theta}_p - \frac{2k_e k_m}{Rr}\dot{x} + \frac{2k_m}{R}V_a + M_p gl\sin\theta_p = -M_p l\ddot{x}\cos\theta_p \quad \textbf{(II.31)}$$

$$\frac{2\mathrm{k_m}}{\mathrm{R}}\mathrm{V_a} = 2\left(\mathrm{M_w} + \frac{2\mathrm{I_w}}{\mathrm{r^2}} + \mathrm{M_p}\right)\ddot{\mathrm{x}} - \frac{2\mathrm{k_e k_m}}{\mathrm{Rr^2}}\dot{\mathrm{x}} + \mathrm{M_p l}\ddot{\mathrm{x}}\cos\theta_\mathrm{p} - \mathrm{M_p l}\dot{\theta}_\mathrm{p}^2\sin\theta_\mathrm{p} \quad \textbf{(II.32)}$$

Les deux équations ci-dessus peuvent être linéarités en supposant $\theta_p = \pi + \phi$, où ϕ représente petit angle de la direction verticale vers le haut ,car le robot est en état d'équilibre lorsque l'angle égale a π . Cette simplification a été utilisée pour permettre d'obtenir de façon linéaire les contrôleurs de l'espace d'état pouvant être mis en œuvre.[12] Donc*:*

$$\cos\theta_p = -1, \qquad \sin\theta_p = -\varphi, \qquad \text{et}\left(\frac{\mathrm{d}\theta_\mathrm{p}}{\mathrm{dt}}\right)^2 = 0$$

L'équation linéaire du mouvement est :

$$(\mathrm{I_p} + \mathrm{M_p l^2})\ddot{\varphi} - \frac{2\mathrm{k_m k_e}}{\mathrm{Rr}}\dot{\mathrm{x}} + \frac{2\mathrm{k_m}}{\mathrm{R}}\ddot{\mathrm{V}}_\mathrm{a} - \mathrm{M_P gl}\varphi = -\mathrm{M_p l}\ddot{\mathrm{x}} \quad \textbf{(II.33)}$$

$$\frac{2\mathrm{k_m}}{\mathrm{R}}\mathrm{V_a} = \left(2\mathrm{M_w} + \frac{2\mathrm{I_W}}{\mathrm{r^2}} + \mathrm{M_p}\right)\ddot{\mathrm{x}} + \frac{2\mathrm{k_m k_e}}{\mathrm{Rr^2}}\dot{\mathrm{x}} + \mathrm{M_p l}\ddot{\varphi} \quad \textbf{(II.34)}$$

Afin d'obtenir la représentation dans l'espace d'état du système, les équations (II.33) et (II.34) sont réarrangées de telle façon que :

$$\ddot{\varphi} = \frac{M_p l}{(I_P + M_P L^2)}\ddot{x} + \frac{2k_m k_e}{Rr(I_p + M_p l^2)}\dot{x} - \frac{2k_m}{R(I_p + M_p l^2)}V_a + \frac{M_p g l}{(I_p + M_p l^2)}\varphi \qquad \textbf{(II.35)}$$

$$\ddot{x} = \frac{2k_m}{Rr\left(2M_w + \frac{2I_w}{r^2} + M_p\right)}V_a - \frac{2k_e k_m}{Rr^2\left(2M_w + \frac{2I_w}{r^2} + M_p\right)}\dot{x} + \frac{M_p l}{\left(2M_w + \frac{2I_w}{r^2} + M_P\right)}\ddot{\varphi} \qquad \textbf{(II.36)}$$

En injectant l'équation (II.35) dans l'équation (II.34) et l'équation (II.36) dans l'équation (II.33), et après quelques opérations algébriques, l'équation dans l'espace d'état du système est obtenue.[12]

II.2.3 Modèle d'état de la dynamique linéaire:

En définissant :

le vecteur d'état $x(t) = [x(t)\dot{x}(t)\varphi(t)\dot{\varphi}(t)]$

le vecteur de sortie $y(t) = [x(t)\varphi(t)]^T$

$x(t)$ le déplacement horizontale des roues selon l'axe des abscices

$\varphi(t)$ l'angle de rotation de la tige du pendule selon l'axe des cordonnes

on obtient le modèle d'état suivant :

$$\begin{cases} \mathbf{x(t)} = \mathbf{A}\dot{\mathbf{x}}(\mathbf{t}) + \mathbf{Bu(t)} \\ \mathbf{y(t)} = \mathbf{Cx(t)} \end{cases}$$

Ou encore :

$$\begin{bmatrix} \dot{x} \\ \ddot{\varphi} \\ \dot{x} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{2k_e k_m}{Rr^2\alpha} & \frac{M_p^2 g l \beta}{\alpha} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{2k_m k_e (r\beta - M_p l)}{Rr^2\alpha} & \frac{M_p g l \beta}{\alpha} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2k_m (I_p + M_p l^2 - M_p l r)}{Rr\alpha} \\ 0 \\ \frac{2k_m (r\beta - M_p l)}{Rr\alpha} \end{bmatrix} V_a \qquad \textbf{(II.37)}$$

Où:

$\beta = (2M_w + \frac{2I_w}{r^2} + M_p)$ V_a : la tension d'alimente du moteur

$$\alpha = [I_p\beta + 2M_p l^2(M_w + \frac{I_w}{R^2})]$$

Dans le modèle ci-dessus et d'âpres la cas du faible vitesse , on suppose que les roues du véhicule resteront toujours en contact avec le sol, et qu'il n'y a pas de glissement au niveau des roues. Les forces de virage sont également considérée comme négligeables.[12]

Tableau II.1 Les valeurs des paramètres

Variables	Nomenclatures	Valeurs	Unités
k_m	Constante de couple moteur	0.57	N.m/A
k_e	Constante force électromotrice	0.48	V.s/rad
M_p	Masse du corps	1.6	Kg
M_w	Masse des roues	0.056	Kg
l	Longueur au centre de masse du corps	0.11	M
r	Rayon des roues	0.035	M
I_P	Inertie du corps	0.0001	Kg.m2
I_w	Inertie des roues	0.000045	Kg.m2
R	Résistance nominale	5	Ohm
G	La gravité	10	m/s2

```
clear all;
clc ;
%%%% le valeurs du parametres de tableux II.1
g=10; %la gravité  (m/s^2)
r=0.035; %(m)
Mw=0.056; %la masse de la roue(kg)
Mp=1.6; %la masse du robot(corps) (kg)
Iw=0.000045; %Inertie du roue(kg*m^2)
Ip=0.0001; %Inertie du corps(kg*m^2)
l=0.11; %Longueur au centre de masse du corps(m)
%les varibales lies au moteurs
Km = 0.57; %le couple du moteur  (Nm/A)
Ke = 0.48;%Constante force électromotrice (Vs/rad)
R = 5; %Résistance nominale(Ohm)
```

Les valeurs des paramètres du moteur sont obtenues à partir de leurs fiches techniques tandis que les valeurs des moments d'inertie du robot et du roue ont été calculer est indique sur l'annexe. Les poids des différentes composantes ont été obtenus en les pesants avec une balance électronique. Le tableau (II.1) regroupe toutes les valeurs des paramètres précédents: on utilise Matlab pour calculer les matrice A et B:

```
%%% calcul de A et B
beta = (2*Mw+(2*Iw/r^2)+Mp);
alpha = (Ip*beta + 2*Mp*l^2*(Mw + Iw/r^2));
A = [0 1 0 0;
0 (2*Km*Ke*(Mp*l*r-Ip-Mp*l^2))/(R*r^2*alpha) (Mp^2*g*l^2)/alpha 0;
0 0 0 1;
0 (2*Km*Ke*(r*beta - Mp*l))/(R*r^2*alpha) (Mp*g*l*beta)/alpha 0];
B = [ 0;
(2*Km*(Ip + Mp*l^2 - Mp*l*r))/(R*r*alpha);
0;
(2*Km*(Mp*l-r*beta)/(R*r*alpha))];
C = [1 0 0 0 ; 0 0 1 0];
D = [0;0];
```

A= [0 1 0 0 ; 0 - 315.2 82.18 0 ; 0 0 0 1; 0 -2690 833.7 0] ,

B =[0 ; 22.99 ; 0 ; 196.2],

C= [1 0 0 0 ; 0 0 1 0] ,

D= [0 ; 0].

La matrice C a deux lignes car on a deux sortie qui sont la position du chassie et l'angle (pendule).

II.2.4. Fonction de transfert de la dynamique linéaire :

Nous système possède une entrée et deux sortie ce pour ça on va trouver deux fonction de transfert. L'application de la transformée de La place a la représentation d'état donne pour fonction de transfert G(s) :

$$\mathrm{G(s)} = \mathrm{C(sI - A)^{-1}B}$$

D'où l'expression de la fonction de transfert :

$$\mathrm{G(s)} = \frac{\det(sI - A + BC)}{\det(sI - A)}$$

```
27          %création du system
28 -        sys=ss(A,B,C,D);
29          %calcul de function de transfer
30 -        tf(sys);
```

On trouve:

$$\begin{cases} G1(s) = \dfrac{x(t)}{u(t)} = \dfrac{22.99S^2 + 5.716 * 10^{-13} - 3042}{S^4 + 315.2 * S^3 - 833.7 * S^2 - 4.172 * 10^4 * S} \\ G2(s) = \dfrac{\emptyset(t)}{u(t)} = \dfrac{192.2 * S - 1.638 * 10^{-11}}{S^3 + +315.2 * S^2 - 833.7 * s - 4.172 * 10^4} \end{cases}$$

Avec

$G1(s)$ c'est la fonction de transfert de la sortie X sur l'entre u et $G(s)$c'est la fonction de transfert de la sortie $\emptyset$ sur l'entre u.

En réalité, les valeurs des paramètres ne peuvent être connues avec exactitude. Aussi, celles-ci peuvent varier dans le temps à cause de certains phénomènes tels que l'usure des composantes et la décharge des batteries. De plus, le modèle a été établi sous certaines hypothèses, en négligeant certains phénomènes et a été approximé linéairement à un point de fonctionnement. Pour toutes ces raisons, le modèle contient des incertitudes et le contrôleur qui sera abordé a la section du command devra être en mesure de stabiliser le système malgré leur présence.

II.2.5. Problématique de la commande:

La Figure (II.5) montre les Pôles et Zéros du système. Le système est instable puisqu'il y a un pôle sur le plan droit du graphe. Dans le meilleur des cas, tous les pôles doivent être sur le plan gauche du graphe pour que le système soit stable.

Les pôles du système sont situés à **-317,-10.4, 0, 12.6**.

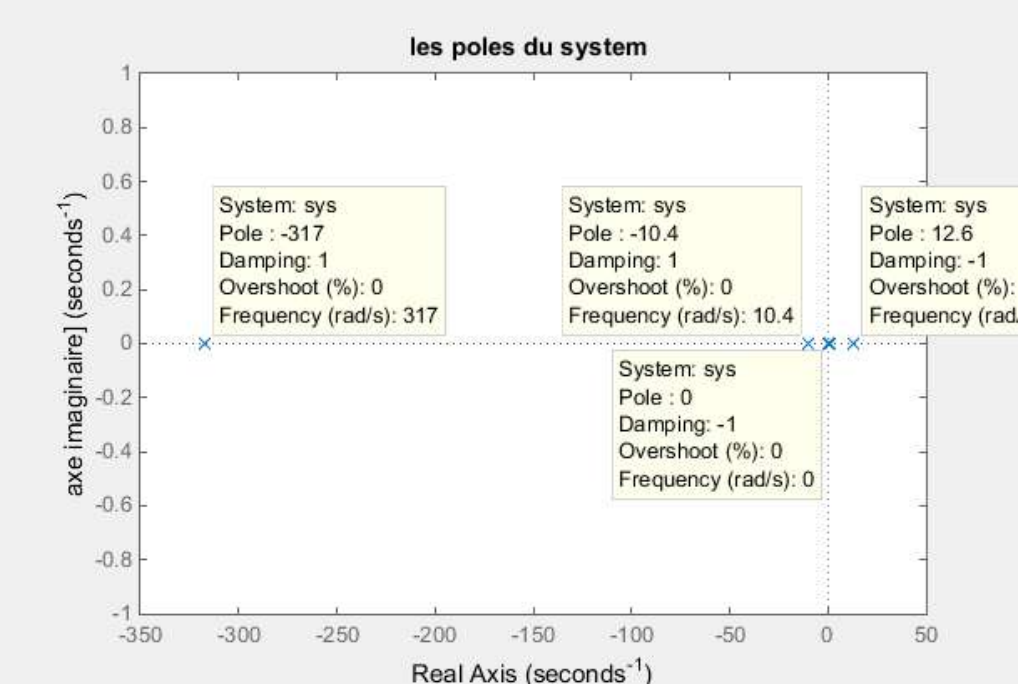


Figure II.5 : Pôles-zéros du système en boucle ouverte

Comme le système est instable, une entrée impulsionnelle appliquée au système en boucle ouverte augmentera l'angle d'inclinaison et la position du robot d'une façon exponentielle.

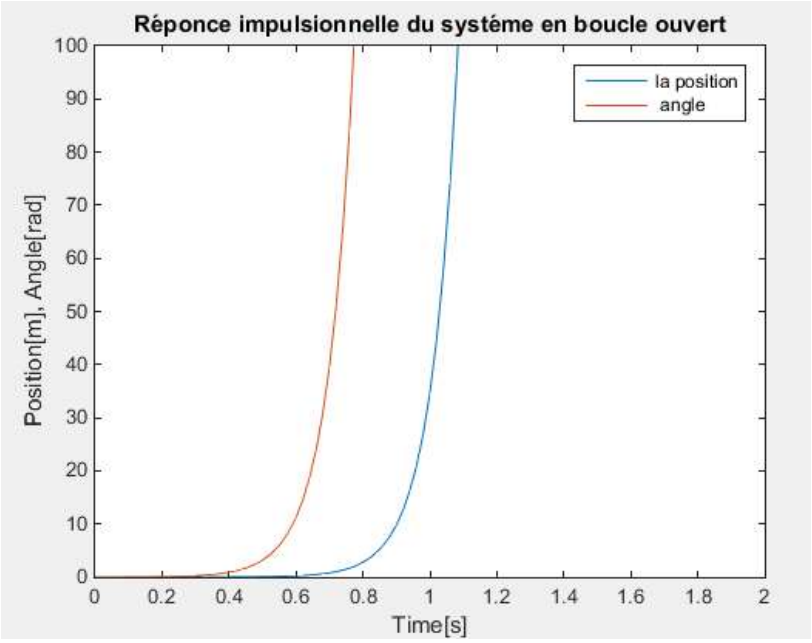


Figure II.6 : Réponse impulsionnelle du système en boucle ouverte

La figure (II.6) montre la réponse impulsionnelle du système indiquant l'instabilité du système.[12]

II.3 Commande du robot auto-balance (pendule inverse):

De nombreux systèmes non linéaires se comportent comme de véritables systèmes linéaires. Les techniques développées dans ce chapitre seront alors utilisées pour la commande des systèmes non linéaires.

Considérons donc le système linéaire suivant :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad \textbf{(II.38)}$$

Notons **m, n, p** les dimensions respectives des vecteurs **u, x et y.** Rappelons que **A** est appelée matrice d'évolution, **B** est la matrice de commande et **C** est la matrice d'observation.

L'objectif d'une régulation ou d'un asservissement est d'assurer le fonctionnement d'un procédé selon des critères prédéfinis par un cahier des charges malgré les perturbations agissant sur ce système asservi. Le cahier des charges qui est traduit le plus souvent en consigne définit des critères qualitatifs à imposer comme la stabilité, la précision, la rapidité, la bande passante….etc.[13]

II.3.1: Commandabilité et observabilité:

Avant de présenter les lois de commandes du système à régler, nous devons montrer que le système à piloter est commandable et observable. Un système est complètement contrôlable si la matrice $\mathbf{C_m}$ est de rang n où n est le nombre de variables d'état. Par conséquent, il faut que le déterminant de la matrice de commandabilité soit différent de zéro. La même condition est exigée par rapport à la matrice observable $\mathbf{O_m}$ **[13] .**

Donc, il est possible de contrôler le système avec une commande par retour d'état :

$$C_m = [B\ AB \ldots\ldots A^{n-1}B]$$
$$O_m = [C^T A^T C^T \ldots\ldots (A^T)^{n-1} C^T] \quad \textbf{(II.39)}$$

a. Commandabilité:

Application numérique :

C_m= [B AB A^2BA^3B]

$$C_m = 10^9 * [\ 0 \quad 0 \quad 0 \quad 0.0023 \ ; \ 0 \ 0 \ 0.0023 \quad -0.7302$$

$$; 0 \quad 0 \quad -0.0001 \quad 0.0197 \ ; \ 0 \quad -0.0001 \quad 0.019 * -6.2404\ ]$$

$$\det(\mathrm{C_m}) = 73.5707 * 10^{11}$$

Le déterminant de $Cm \neq 0$, le rang de Cm est n = 4. Donc notre système estcomplètement contrôlable.[8]

b. Observabilité:

Application numérique :

O_m= [$C^TA^TC^T$ …… $(A^T)^{n-1}C^T$] **(II.40)**

rang = 4

Om = 10^5 [0 0 0 0 ; 0 0 0 0 ; 0 0 0 0 ; 0 0 0 0 ; 0 - 0.0032 0.0008 0 ; 0 0.269 0.00083 0;
0 0.9937 - 0.2591 0.0008 ; 0 8.4811 -2.2110 0.0083]

Le rang Om de est n= 4. Plus de détails sur la génération de la Om a été observé et il est égal au nombre de variables d'état. Par conséquent, le système d'équilibrage dynamique du robot est complètement observable.

II.3.2 Commande du système on boucle fermée :

A) La commande PID:

Il existe trois actions correctives élémentaires (proportionnelle , intégrale ,dérivée) qui permettent, individuellement, de corriger telle ou telle performance. Elles sont relativement simples à réaliser mais, en général, dégradent d'autres performances .

Elles sont utilisables lorsque le cahier des charges est peu exigeant. Dans le cas contraire, il faut envisager de combiner ces différentes actions au sein d'un correcteur plus complexe.

Le système robotique ayant une dynamique extrêmement instable, l'utilisation d'un système de contrôle était extrêmement nécessaire pour le stabiliser autour du point de consigne (position verticale). Ainsi, le contrôleur PID a été mis en œuvre afin de maintenir la verticalité du robot [14][15].

L'objectif principal du contrôleur PID était de maintenir la valeur de l'angle de pitch du robot à une valeur proche de la valeur de consigne , qui dans notre cas était de 178 ° car le capteur qui mesuré l'angle est monté parallèlement au sol. La valeur de erreur a été transmise au contrôleur d'âpres la soustraction de la valeur de consignes avec la valeur de l'angle mesure.

L'erreur a ensuite été utilisée pour calculer le signal de correction ou le signal d'entrée comme suit:

$$u(t) = K_p . e(t) + K_t . \int_0^t e(t)dt + K_d . \frac{de(t)}{dt} \quad \textbf{(II.41)}$$

Voici:

K_p , K_t et K_d . les gains pour le contrôleur PID. Ces effets sont résumés dans le tableau II.2:

Tableau II.2 Tableau récapitulant l'influence d'un PID parallèle sur le système qu'il corrige si l'on augmente séparément l'action proportionnelle (P), intégrale (I) ou dérivée (D)[15].

Coefficient	**Temps de montée**	**Temps de stabilisation**	**Dépassement**	**Erreur statique**
Kp	Diminue	Augmente	Augmente	Diminue
Ki	Diminue	Augmente	Augmente	Annule
Kd	Aucune effet	Diminue	Diminue	Aucune effet

Il a été noté que le terme intégral dans l'équation ci-dessus était l'accumulation d'erreur au fil du temps et le terme dérivé était le taux de changement d'erreur. Par conséquent, les termes individuels ont été remplacés comme suit :

$$e(t) = e \qquad \int_0^t e(t)dt = E * dt ; \qquad \frac{de(t)}{dt} = \frac{e - e_0}{dt} \quad \textbf{(II.42)}$$

Ici, e est l'erreur actuelle, E est la somme de toutes les erreurs sur la durée, $\mathbf{e_0}$est l'erreur précédente et dt est le temps d'échantillonnage. Par conséquent, l'équation (II.41) a été réécrite comme suit:

$$u(t) = k_p * e + k_i * E * dt + k_d \frac{e-e_0}{dt} \text{ [16]} \quad \textbf{(II.43)}$$

Physiquement, le terme proportionnel corrige le système linéairement par rapport à l'erreur et donc souvent construit un décalage d'erreur. Le terme intégral était particulièrement utile lorsqu'un décalage d'erreur persistant avait été développé. le terme dérivé a résisté à tout changement soudain de l'erreur et a donc apaisé la réponse du système. Ainsi, le proportionnel terme a corrigé l'erreur présente, le terme intégral a compensé les erreurs passées et le terme dérivé prédit et évite les erreurs futures. Pour ajuster rapidement les paramètres des régulateurs P, PI et PID, on a utiliser la Méthodes de Ziegler et Nichols. La méthode du pompage permet d'amener le système bouclé à sa limite de stabilité [17].

Tableau II.3 : Paramètres PID selon Ziegler-Nichols

Type	K_P	T_i	T_d
P	$0.5.K_{cr}$	-	-
PI	$0.45.K_{cr}$	$0.83.T_{cr}$	-
PID	$0.6.K_{cr}$	$0.5.T_{cr}$	$0.125.T_{cr}$

B) le filtre de kalman :

Le filtre de Kalman est un filtre récursif, qui utilise une approche temporelle . Il permet d'estimer l'état (cet état correspond à l'information utile) d'un système dynamique à partir d'une information, à priori, sur l'évolution de cet état (modèle) et de mesure réelle.

L'utilisation de ce filtre est particulièrement recommandée lorsque l'imprécision sur les données traitées peut être interprétée comme un bruit aléatoire, dont la distribution est connue à priori **.**

Cela signifie que pour estimer l'état courant d'un système, l'estimation de l'état précédent et des mesures actuelles sont nécessaires. Le principe d'un tel filtre repose sur un modèle à variables d'état basé sur deux équations :

- L'équation récursive de modélisation du processus aléatoire x_k((vecteur d'état) discret, dite équation d'état ou équation de transition.
- L'équation d'observation (ou équation de mesure) du processus y_k(vecteur de mesure).[18][19][20]

L'objectif est de déterminer une estimation récursive des x_kà partir des mesures $y_k$et du modèle de génération du processus.

Un filtre de Kalman est un algorithme qui peut être utilisé pour estimer des variables inconnues réduire le bruit généré par les mesures. Pour un modèle espace-temps discret sur le formulaire

$$x_{k+1} = A_{xk} + B_{uk} + v_k \quad \textbf{(II.44)}$$

$$y_k = c_{xk} + e_k \quad \textbf{(II.45)}$$

ou :

$E\{V_K V_J^T\} = Q_V \delta_{KJ}$, $E\{e_K e_J^T\} = Q_e \delta_{KJ}$, $E\{V_K e_J^T\} = Q_{ve} \delta_{KJ}$.

le filtre de Kalman est construit selon les équations II.46-II.50:

$$\hat{y}_k = C\hat{x}^k \quad \textbf{(II.46)}$$

$$R_k = Q_e + CP_x C^T \quad \textbf{(II.47)}$$

$$P_{k+1} = AP_x A^T + Q_v - K_k R_K K_K^T \quad \textbf{(II.48)}$$

$$K_k = (AP_K C^T + Q_{ve})(Q_e + CP_K C^T) \quad \textbf{(II.49)}$$

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + K_k(y_k - \hat{y}_k) \quad \textbf{(II.50)}$$

Dans le cas de l'estimation de l'angle avec les mesures d'un accéléromètre et d'un gyroscope, un filtre de Kalman plus simple peut être implémenté selon les équations suivantes [21]:

$$\hat{x}_{k|k-1} = F_{xk-1\hat{|}k-1} + B\,\dot{\theta}_K$$

$$P_{K\hat{|}k-1} = Fp_{k\hat{|}k-1} F^T + Q_K$$

$$\hat{Y}_K = Z_k - H\hat{x}_{k\hat{|}k-1}$$

$$S_K = HP_{K|K-1}H^T + R$$

$$K_t = P_{K|K-1}H^T S_K^{-1}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + k_k\hat{y}_k$$

$$P_{k|k} = (I - k_k H)P_{k|k-1} \quad \textbf{(II.51)}$$

$$\hat{X}_K = \begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}, \text{F=} \begin{bmatrix} 1 & -h \\ 0 & 1 \end{bmatrix}, \text{B=} \begin{bmatrix} h \\ 0 \end{bmatrix}, Q_k = \begin{bmatrix} Q_\theta & 0 \\ 0 & Q_{\dot{\theta}_b} \end{bmatrix}, \text{R= var}(e_k) \quad \textbf{(II.52)}$$

où θ est l'angle, $\dot{\theta}_b$ le biais des mesures de le gyroscope, h le temps d'échantillonnage, Q est le bruit du processus et R bruit de mesure. Les bruits sont les paramètres qui peuvent êtres à l'écoute. La matrice P initiale peut être définie sur une matrice 2x2 avec des zéros.[21]

C. Régulateur linéaire-quadratique:

Le LQR a été utilisé comme régulateur principal. Il est concerné avec le fonctionnement d'un système dynamique à un coût minimum et peut être décrit par un ensemble d'équations:

$$\mathrm{u(t)} = -\mathrm{L\hat{x}(t)}$$

$$\dot{\mathrm{x}}(\mathrm{t}) = \mathrm{A\hat{x}(t)} + \mathrm{Bu(t)} + \mathrm{K(y(t) - C\hat{x}(t))}$$

$$0 = \mathrm{Ap} + \mathrm{PA^T} + \mathrm{NR_1N^T} - \left(\mathrm{pC^T} + \mathrm{NR_{12}}\right)\mathrm{R_2^{-1}}(\mathrm{PC^T} + \mathrm{NR_{12}})^{\mathrm{T}} \quad \textbf{(II.53)}$$

$$\mathrm{K} = (\mathrm{PC^T} + \mathrm{NR_{12}})\mathrm{R_2^{-1}} \quad \textbf{(II.54)}$$

$$\mathrm{L} = \mathrm{Q_2^{-1}B^TS} \quad \textbf{(II.55)}$$

$$0 = \mathrm{A^TS} + \mathrm{SA} + \mathrm{M^TQ_1M} - \mathrm{SBQ_2^{-1}B^TS} \quad \textbf{(II.56)}$$

l'équation algébrique de Riccati est résolue pour calcules le gain de contrôle K qui est évaluées pour différentes valeurs de matrices de pondération Q et R, les équations donnent finalement un contrôleur de retour d'état.[19]

D) Filtre complémentaire :

Lorsqu'il s'agit de capteurs multiples présentant divers défauts, un filtre complémentaire peut être utilisé pour minimiser ces défauts. L'idée d'un filtre complémentaire est de combiner et filtrer des signaux provenant de différents capteurs de telle sorte qu

$$\text{angle} = \text{a} \cdot (\text{angle} + \text{gyro} \cdot \text{dt}) + (1 - \text{a}) \cdot \text{accel}$$

$$\text{où} \quad a = \frac{\propto}{\alpha + dt}$$

où α est la constante de temps souhaitée et $dt = \frac{1}{fs}$ avec fs est la fréquence d'échantillonnage.[21]

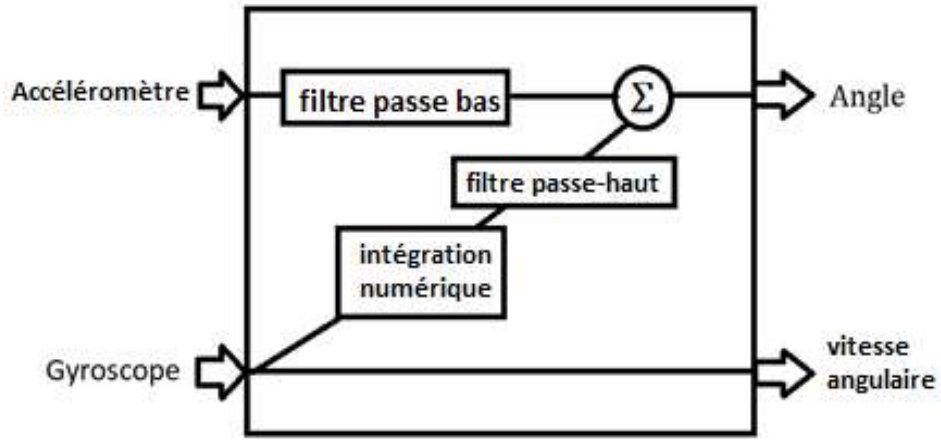


Figure II.7 : le filtre complémentaire

II.4 Simulation du la commande par LQR :

En utilisant MATLAB, l'équation algébrique de Riccati est résolue et le gain de contrôle K est évaluées pour différentes valeurs de matrices de pondération Q et R qui sont choies arbitrairement . La réponse du le système est également simulée.

la matrice Q est de forme suivante :

$$Q = [a\ 0\ 0\ 0; 0\ b\ 0\ 0\ ; 0\ 0\ c\ 0\ ; 0\ 0\ 0\ d\ ]$$

où les valeurs de a, b, c, d sont les pondérations pour les états respectifs la position ,

la vitesse , l'angle et la vitesse angulaire tanque la matrice de pondération R est une valeur scalaire car il n'y a qu'une seule entrée de contrôle dans le système.

Les valeurs de la matrice Q sont ajustées en fonction de la réponse requise par le système,

une valeur plus élevée des pondérations indique l'importance des états comparés aux autres.

Comme les éléments de la matrice Q et R sont déterminés de manière arbitraire, plusieurs simulations et des essais de mise en œuvre sont effectués pour obtenir la réponse souhaitée pour l'équilibrage au système de robot.[21]

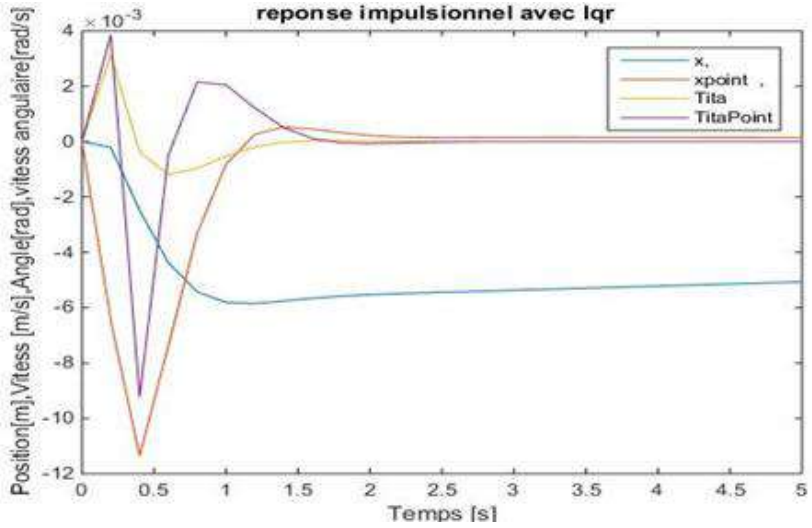
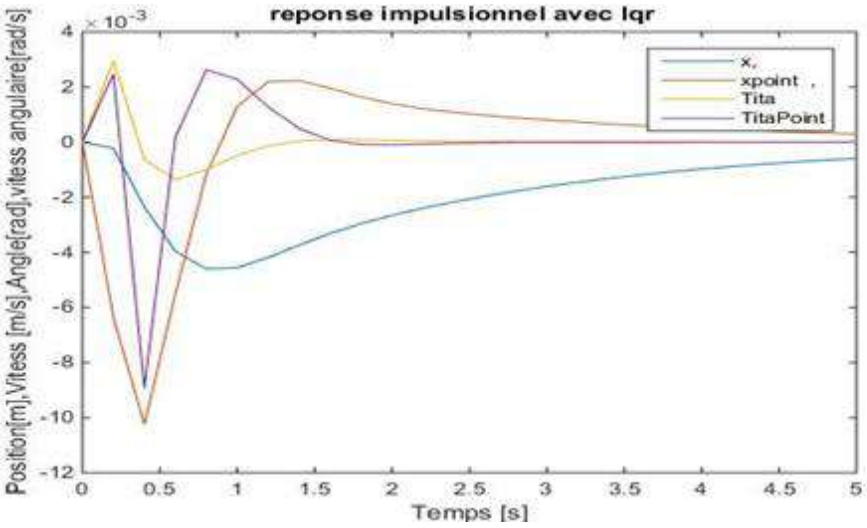


Figure II.8 : Résultat de la variation sur les paramètres de la matrice Q



FigureII.9 : la position ne converge pas au consigne

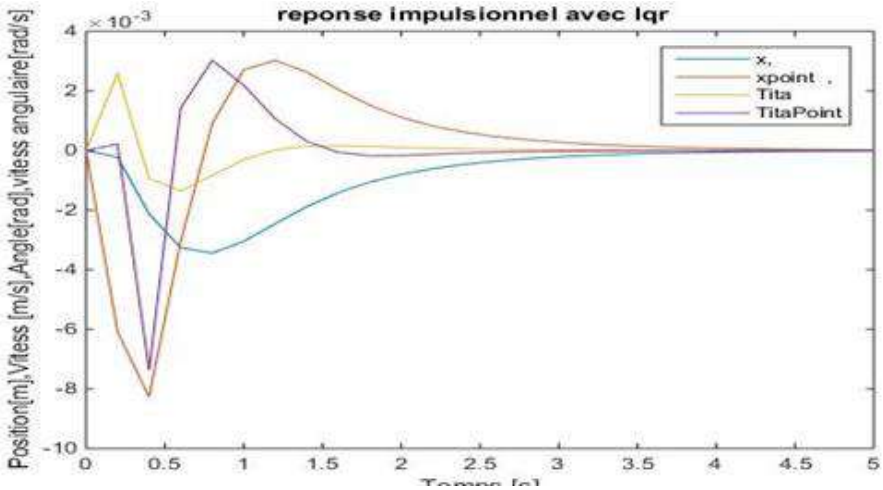


Figure II.10 : la réponse impulsionnel par la command LQR

La réponse la plus acceptable pour le système est illustrée à la figure II.10.

la pondération de la vitesse angulaire et de la vitesse linéaire reste inchangée car la réponse des états dépend des deux autres ,

les nouvelles pôles sont situés à : **-6250, -2.1,-3.15+ i2.55, -3.15-i2.55**.

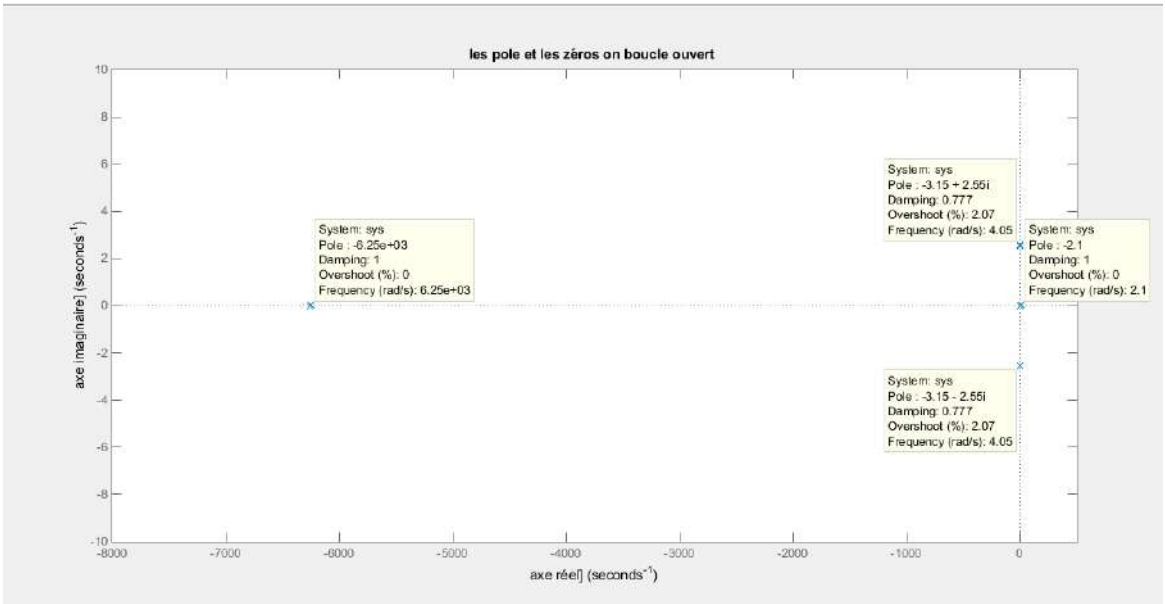


Figure II.11: la répartition des nouveaux pôles et zéros

- ✓ **le block de simulink criée :** ce block représente le block crée sous Matlab,il se compose d'un régulateur LQR ,du system et de la référence, les sous-block se trouvent sur l'annexe

- ✓ **simulation du block**

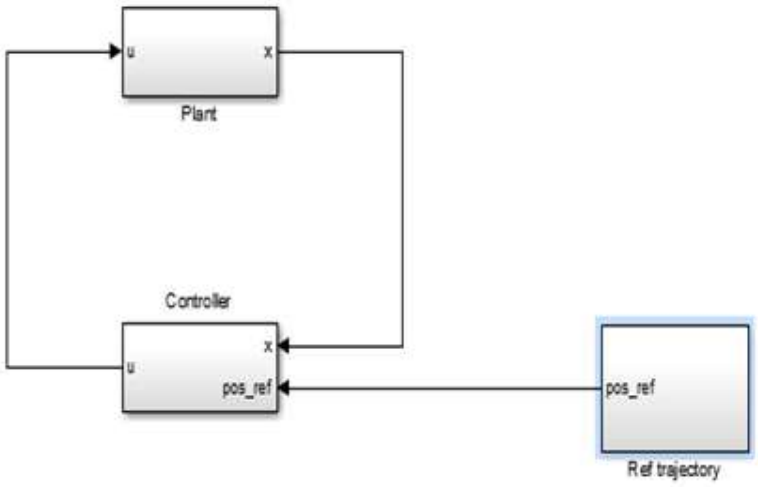


Figure II.12 : le block simulink

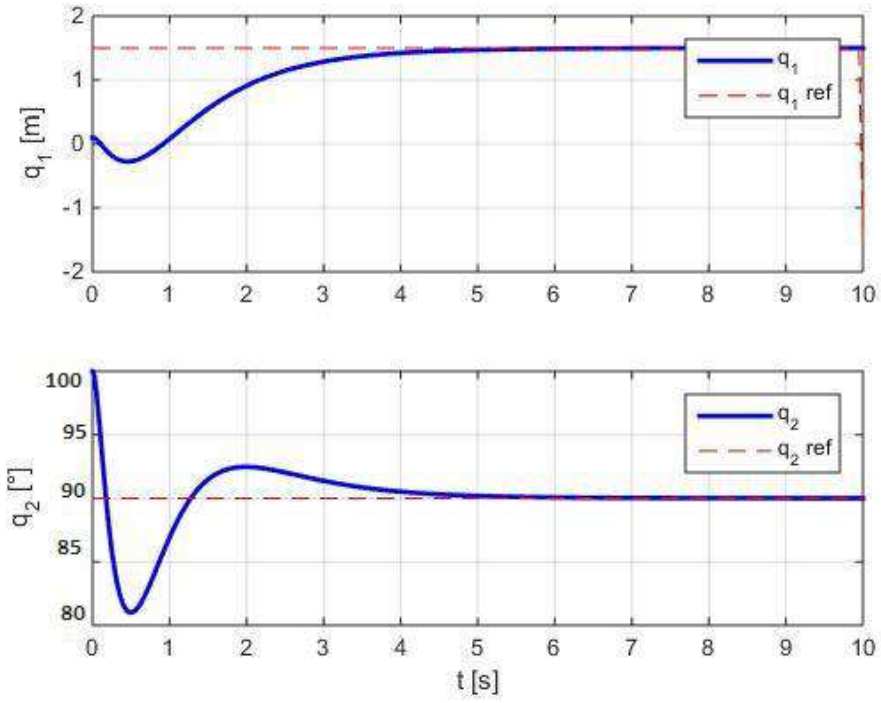


Figure II.13 : la variation de positon de l'angle selon la consigne

NB: q1 est la position et q2 est l'angle du robot, le régulateur LQR utiles les quatres paramètres du robot (positon , vitesse , angle , vitesse angulaire) pour commander le robot afin d'assurer sa position et son angle selon la consignes ,d'âpres l'utilisations de ce régulateur , les pôles et les zéros du system seront déplacés au demi-plan négatif, le système deviens par conséquent stable .

II.5. Conclusion :

Dans ce chapitre, on a détaillé le modèle mathématique du robot auto-balance afin d'analyser son comportement en boucle ouverte. ou on a prouvé à partir de sa réponse impulsionnel que le système est instable. Pour assurer la stabilité de notre robot on va introduire des régulateurs et choisir la command par LQR grâce à sa précision, ce dernier utilisera la postions ,la vitesse ,l'angle et la vitesse angulaire, ces paramètres permettront de réguler le robot d'une manière plus précise .

CHAPITRE III

Description mécanique et électronique

III.1 Introduction :

Le robot auto-balance est une structure mécanique spéciale contrôlée à l'aide des actionneurs à travers un circuit électrique de commande. Ce chapitre est divisé en deux grandes parties, la description de la partie mécanique ainsi la description de la partie électrique , avec le circuit de commande à la fin du chapitre .

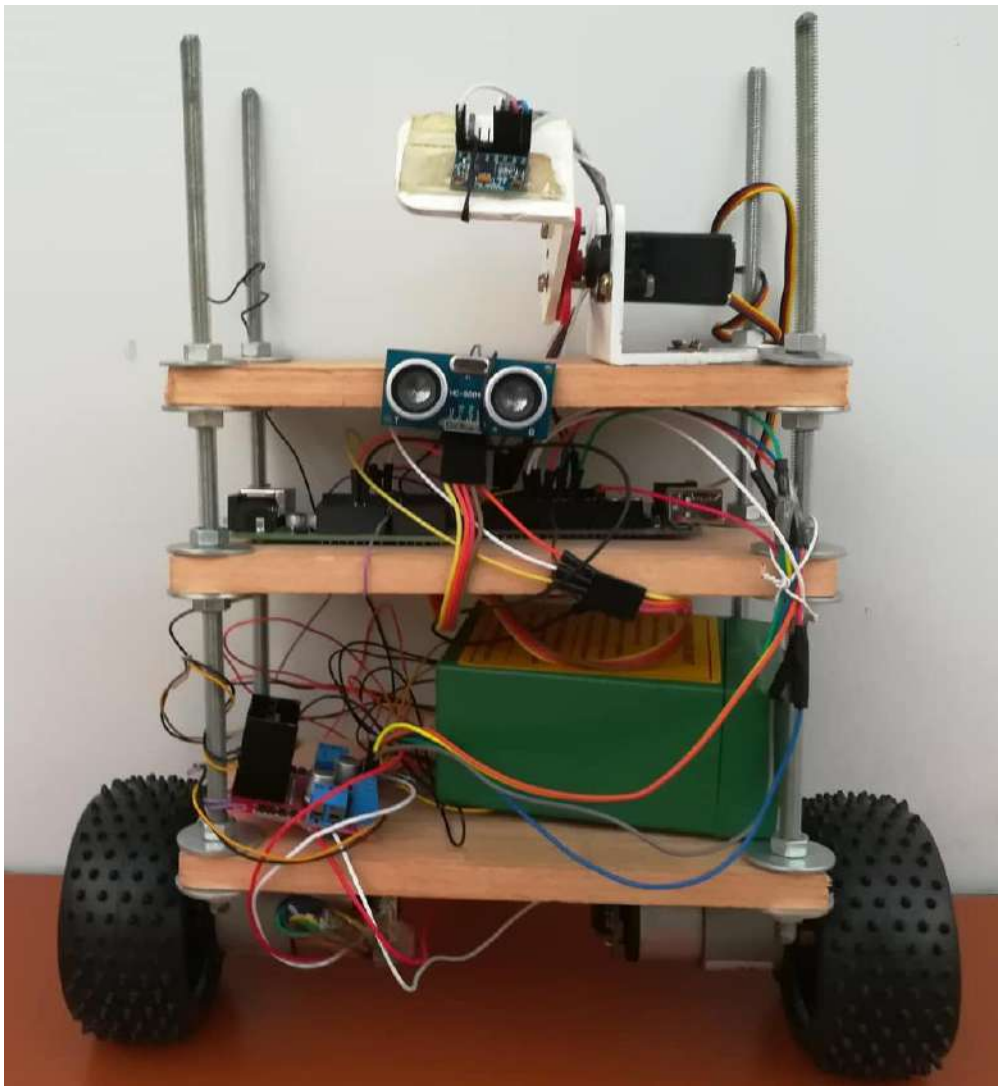


Figure III.1 : robot mobile réalisé

III.2 Partie mécanique :

Pour procéder notre constitution du projet, il faut choisir le matériel de la conception, puisqu'il nous faut un matériel compatible avec la puissance, l'alimentation, la taille, le fonctionnement et le poids du robot, nous avons choisi une conception mécanique relativement simple mais efficace permettant d'atteindre le but recherché, notre structure mécanique est composée de différentes pièces métalliques et en bois pour construire la forme générale avec les composants nécessaires pour le balancement du robot.

III.2.1 Choix des matériaux et passage à l'usinage :

Le matériel choisi pour la réalisation de ce modèle est le bois grâce à ses caractéristiques physiques en l'occurrence sa solidité et sa légèreté qui assurera un bon équilibre du rebot.

L'assemblage est fait à l'aide d'un système (tiges, écrous, rondelles).avec deux supports ajoutés au robot en aluminium.

III.2.2 La construction mécanique :

La structure mécanique est composée de différentes pièces mentionnées et expliquées par le tableau **III.1**.

Tableau III.1 : les pièces utilisées dans la structure mécanique de notre robot

Nom de la pièce	Nombre des pièces	Dimensions (cm)
Les roues	2	D = 8 l = 3.5
Moteur courent continue	2	5*2
Les plaques	3	1 * 10 *18
Les tiges	4	L = 25
les rondelles	24	R = 0.875
Les écrous	24	R = 0.5
Pièce du forme L	2	4*5 5*7
Servomoteur	1	4.7x1.97x4.29
Alimentation	1	10*4*6

III.2.2.1 Les roues :

Notre robot se meut à l'aide de deux roues parallèles et coaxiales, elles sont munies de pneus en caoutchouc qui permettent de rouler sans glisser, chaque roue est actionnée séparément par un moteur à courant continue.



Figure III.2 : les dimensions des roues

III.2.2.2 les moteurs Courant Continu :

Chaque roue du robot est motorisée indépendamment de l'autre, offrant le contrôle de lacet par différentiation, voir inversion du régime de rotation. Le moteur utilisé est de type moteur à courant continu alimenté à 12V.



Figure III.3 : le moteur utilisé et ses accessoires .



Figure III.4 : l'assemblage du moteur avec la roue

Les caractéristiques du notre moteur sont déterminées dans le tableau suivant [22] :

Tableau III.2 : les caractéristiques du moteur utilisé

Type	12V 107tr/min
Tension nominal	12V
Vitesse à vide	107 TR/MIN 0.15A
Efficacité maximale	4.3kg.cm/85 tr/min/3.8 W/0.60A
Puissance Max	10.0kg.cm/52 rpm/6.0 W/1.12A
Couple de décrochage	20.0kg.cm 3.4A
Ralentisseur Rapport De Réduction	1: 90
Hall Résolution	Hall x Ratio 11x0 = 990PPR
Force électromotrice	0.48 red.V /S

III.2.2.3 Les plaques d'étages:

ces plaques représentent les étages du robot pour regrouper les composants nécessaires, on a utilisé des plaques en bois pour diminuer le poids global du robot, le bois est parmi les meilleurs isolants pour éviter les courts circuits entres les composants électriques tel que le gyroscope, la figure III.6 présente le fixage du moteur au plaque du bois avec les roues associées aux moteurs.



Figure III.5 : les plaque du séparations

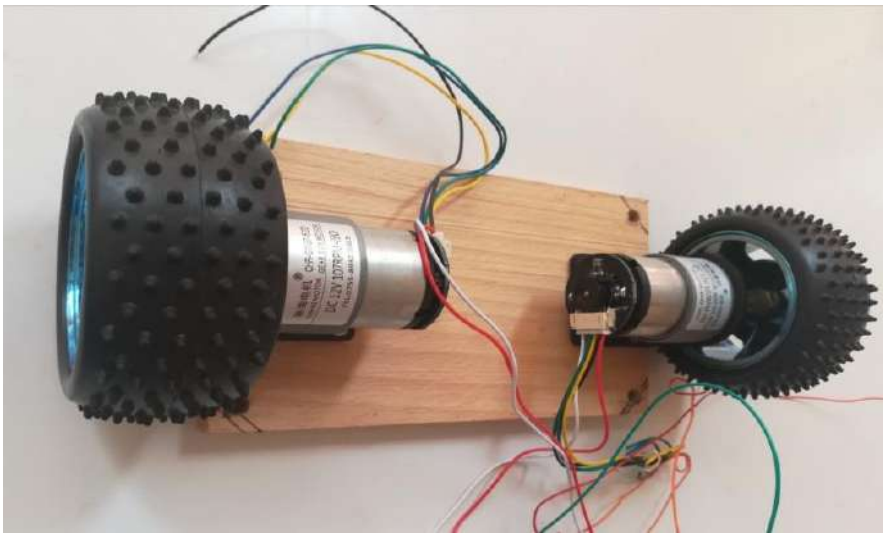


Figure III.6 : le châssis du robot

III.2.2.4 Les tiges , les écrous et les rondelles :

On a utilisé des tiges métalliques et des rondelles pour l'assemblage des pièces entre eux , l'association des tiges avec les plaques est faites par des rondelles et des écrous ce qui permet de regrouper le robot d'une forme solide est stable.



Figure III.7 : les tiges, rondelles et les écrous

III.2.2.5 les pièces du forme L :

Ce sont des supports d'aluminium avec un angle de 90° , son rôle a pour but d'ajuster le gyroscope au point d'équilibre ainsi le mouvement du servomoteur, ce dernier fait changer l'angle qui peut varier la position du robot.

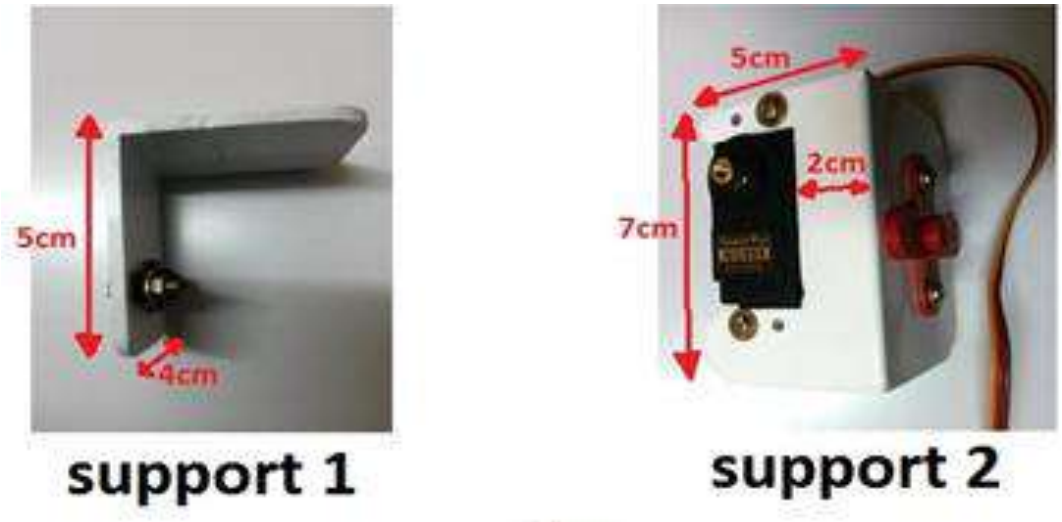


Figure III.8 : support du forme L



Figure III.9 : l'emplacement du servomoteur

III.2.2.6 Servomoteur MG 995 :

Un servomoteur est un moteur à courant continu spécial avec un réducteur, il est équipé d'un circuit électronique pour commander le sens de rotation, ainsi que la position de l'arbre du moteur [23] [24].ils sont faciles pour la commande avec une précision élevée et délivrent un couple considérable. Pour ces raisons, on a choisi de travailler avec le servomoteur de référence MG 995 High speed (Figure III.11)achetés au niveau d'un magasin de vente de pièces électroniques. Les détails sur le principe et la commande du servomoteur sont expliqués dans la partie Actionneur.

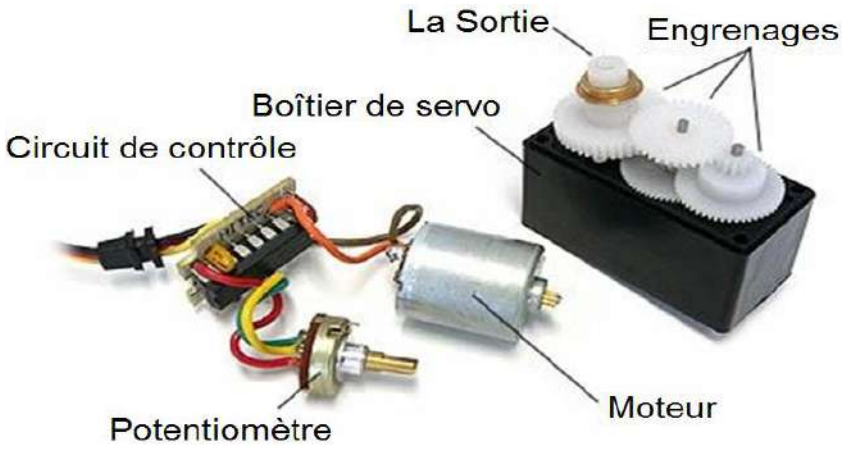


Figure III.10 : Les constituants d'un servomoteur

Caractéristique du servomoteur utilisé :

- Poids :55g
- Dimension: 40.7x19.7x42.9mmapprox.
- Vitesse de fonctionnement: 0.2s/60º(4.8V),0.16s/60º(6V)
- Tension de fonctionnement: 4.8Va7.2V

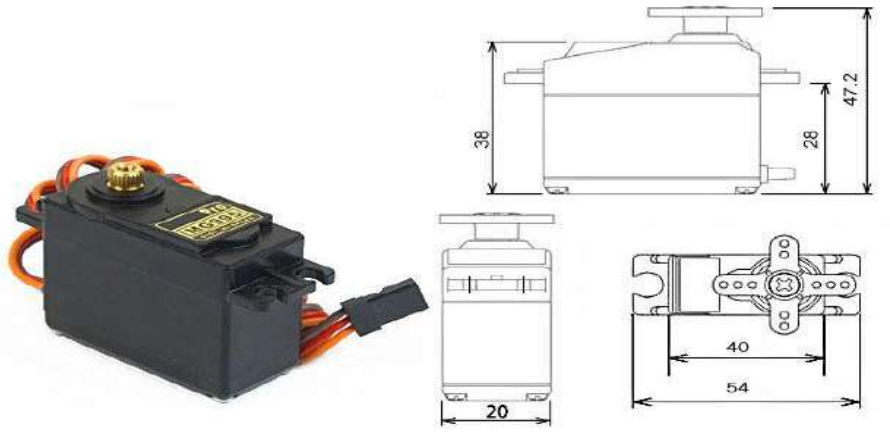


Figure III.11 : le servomoteur MG 995 High speed avec ses accessoires

III.3 description de la partie électrique :

Dans un robot mobile la partie électronique joue un rôle très important pour faire fonctionner le système , la mise en marche de tous les systèmes doit disposer des certains éléments électroniques présentent la partie de la commande qui donne des ordres et fait le traitement d'information reçue par les différents capteurs.

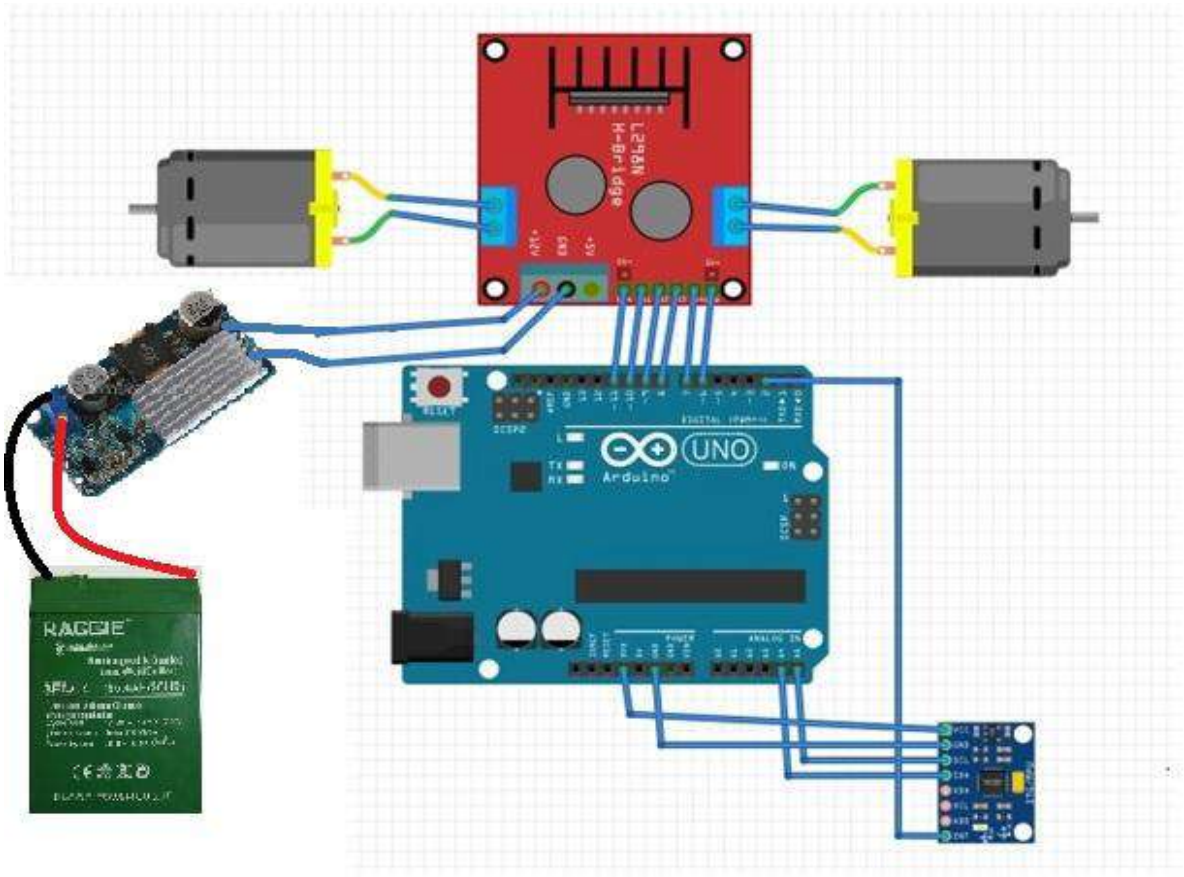


Figure III.12 : Schéma électrique de la réalisation

III.3.1 Alimentation :

L'alimentation du robot réalisé est faite grâce a une sources de tension continue de 6 V, on a utilisé une batterie de série 3FM-4 aviez de haut performance capable a recharge 1000 fois , elle sest très utilisable dans les onduleur, les télémètre de durée de vie de 5 ans dans les conditions normales [25] (température 25 dégrée) du caractéristique suivants :

- curent initiale : 1.5 A ,

- capacité : 1Ah ,

-tension continue 6v.



Figure III.13 : l'alimentation 6V utilisé.

III.3.2 le convertisseur boost :

C'est un hacheur parallèle capable a converti une tension continu-continu a une tension continue variable à partir d'une tension continue fixe. La tension continue de départ peut être un réseau alternatif redressé et filtré, une batterie d'accumulateurs, une alimentation stabilisée…

On distingue deux types de convertisseurs continu-continu. Ceux qui sont non isolés, que l'on appellera hacheurs, et ceux qui comportent un transformateur assurant l'isolation galvanique, que l'on appelle alimentations à découpage (cas des alimentations de PC…). Par la suite, nous n'étudierons que les premiers[26].

On utiliser ce model qui est dans la figure III.14

Caractéristique [27]:

- ✓ le module a une puissance de sorite et une efficacité élèves (jusqu'a 96.7%)
- ✓ la sortie peut atteindre 75W
- ✓ Sortie grande puissance
- ✓ haute efficacité (96.7%)
- ✓ tension de travail large
- ✓ la puissance de sortie atteint 75 W

Lorsque la tension d'entrée et de sortie est supérieure à 20 V, elle peut être obtenue lorsque la puissance de sortie de 128 W

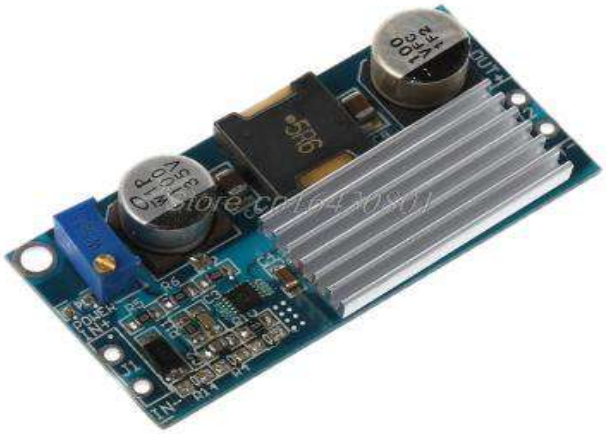


Figure III.14 l'hacheur boost

En +: entrée Positive OUT +: sortie Positive

IN-: entrée négative OUT-: sortie négative

III.3.3 MPU-6050:

Le dispositif MPU-6050 (Motion Processing Unit) combine un gyroscope 3 axes et un accéléromètre à 3 axes sur le même silicium avec un bord Digital Motion Processor ™ (DMP ™) capable de traiter des algorithmes complexes de Motion Fusion 9 axes L'unité pour accéder également aux capteurs et magnétomètres externes via le bus I2C maître auxiliaire.

Pour le suivi de la précision des mouvements à la fois rapides et lents, le module disposent d'un gyroscope et un accéléromètre programmable par l'utilisateur avec une plage à pleine échelle [28]

- Alimentation: 2,3 à 3,4 Vcc
- Consommation: 3,9 mA maxi

Le MPU6050 utilise le protocole I2C pour communiquer avec la carte Aduino.

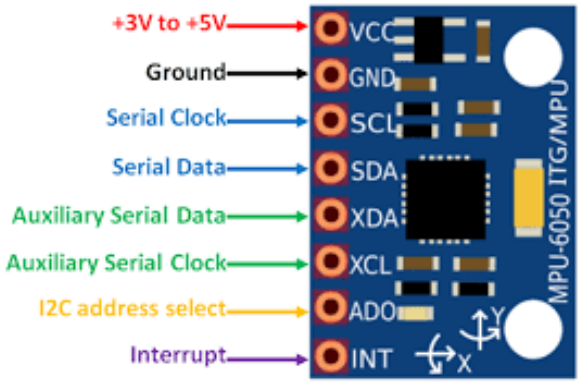


Figure III.15: capteur MPU6050

Ce composant permet de calculer la vitesse angulaire et l'accélération linéaire d'un objet. Le composant est le GY-521, fonctionnant en I2C ou en SPI, qui est un composant MEMS, ce qui signifie qu'il est fortement intégré.

L'utilité de chaque pin et come suite :

VCC: pour l'alimentation du module par 3.3v ou 5v

GND: pour branche la masse au module .

SCL: Utilisé pour fournir une impulsion d'horloge pour la communication I2C

SDA: Utilisé pour transférer des données via une communication I2C

XDA, XCL: Peut être utilisé pour interfacer d'autres modules I2C avec MPU6050. C'est optionnel

ADO: Si plusieurs MPU6050 sont utilisées dans une même MCU, cette broche peut être utilisée pour modifier l'adresse.

INT : Broche d'interruption pour indiquer que les données sont disponibles pour que la MCU puisse les lire..

III.3.3.1 Gyroscope :

Le gyroscope est un capteur inertiel qui permet d'évaluer la rotation angulaire autour d'un axe. Un gyroscope à 3 axes mesure la rotation autour chacun des axes X, Y, Z. Il calcule le nombre de degrés par seconde qui a été faite en rotation.[28]

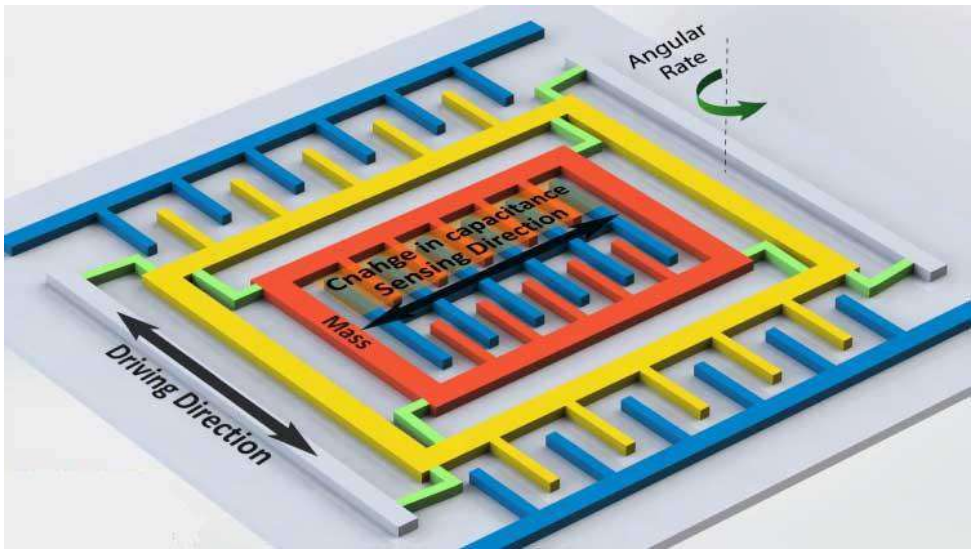


Figure III.16 : La microstructure du gyroscope.

Le gyroscope est un appareil qui exploite le principe de la conservation du moment angulaire en physique (ou encore stabilité gyroscopique ou effet gyroscopique). Cette loi fondamentale de la mécanique veut qu'en l'absence de couple appliqué à un solide en rotation, celui-ci conserve son axe de rotation invariable. Lorsqu'un couple est appliqué à l'appareil, il provoque une précession ou une nutation du solide en rotation. Dans les capteurs : un gyroscope est un capteur de position angulaire et un gyromètre un capteur de vitesse angulaire. Le gyroscope donne la position angulaire (selon un, deux ou les trois axes) de son référentiel par rapport à un référentiel inertiel (ou galiléen) [28]

le gyroscope de module mpu6050 fonctionnent sur le principe de l'accélération de Coriolis. Imaginez qu'il y a une fourchette, comme la structure, qui est en mouvement constant de va-et-vient. Maintenu en place à l'aide de cristaux piézo électriques. Chaque fois, vous essayez de basculer cet arrangement, les cristaux subissent une force dans le sens de l'inclinaison. Ceci est causé en raison de l'inertie de la fourche mobile. Les cristaux produisent ainsi un courant dans un consensus avec l'effet piézoélectrique, et ce courant est amplifié. Les valeurs sont ensuite affinées par le microcontrôleur.[29]

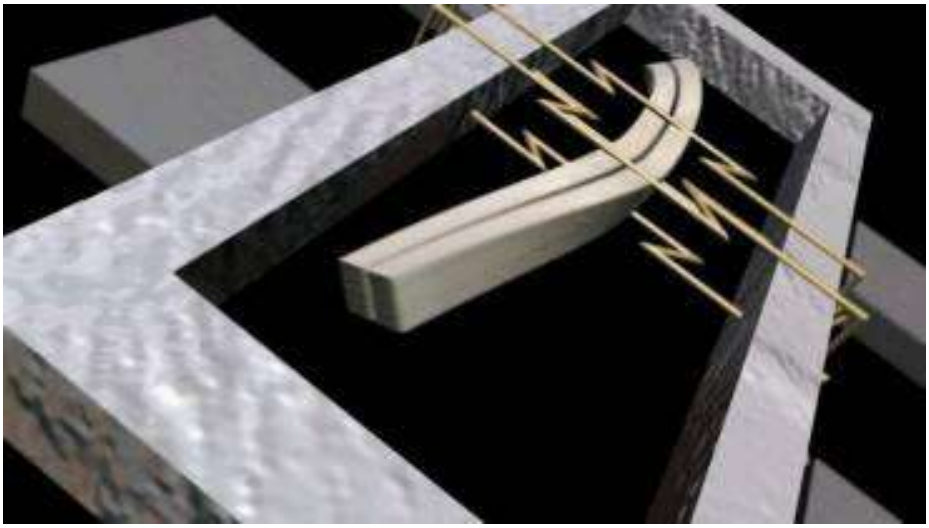


Figure III.17 : Gyroscope Piezo Electric

Caractéristiques Gyroscope : [30]

- Digital-output X-, Y-, et Z-Axis angular rate sensors (gyroscopes) programmable avec une plage à pleine échelle ±250, ±500, ±1000, and ±2000°/sec

- tolérance de calibration: ±3%

- Interface de communication : I2C

III.3.3.2 Accéléromètre:

L'accéléromètre est un capteur qui permet de connaître l'accélération linéaire selon trois axes X, Y et Z. Bien que l'accélération linéaire soit définie en m/s2 (la gravitation terrestre, soit environ g = 9,81 m/s2).

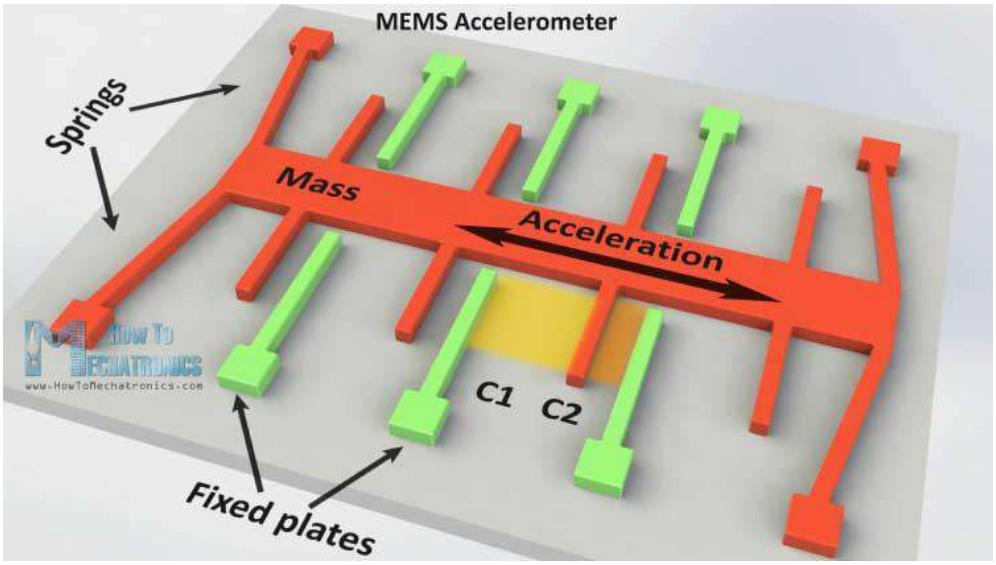


Figure III.18: La microstructure de l'accéléromètre.

Le principe de la plupart des accéléromètres est basé sur la loi fondamentale de la dynamique F = m·a (F : force (N), m : masse (kg), a : accélération (m/s2) aussi notée γ). Plus précisément, il consiste en l'égalité entre la force d'inertie de la masse sismique du capteur et une force de rappel appliquée à cette masse. [28].

l' accéléromètre du mpu6050 basé sur l'effet piézo-électrique. Ici, imaginez une boîte parallélépipédique, ayant une petite boule à l'intérieur, comme dans l'image ci-dessus. Les murs de cette boîte sont fabriqués avec des cristaux piézo électriques. Chaque fois que vous inclinez la boîte, la balle est forcée de se déplacer dans le sens de l'inclinaison, en raison de la gravité. Le mur avec lequel la balle entre en collision, crée minuscules piézo courants électriques. Il y a au total trois paires de parois opposées dans un parallélépipède. Chaque paire correspond à un axe dans l'espace 3D: X, Y et Z axes. En fonction du courant produit à partir des parois piézo électriques, on peut déterminer la direction d'inclinaison et de sa grandeur.[31]

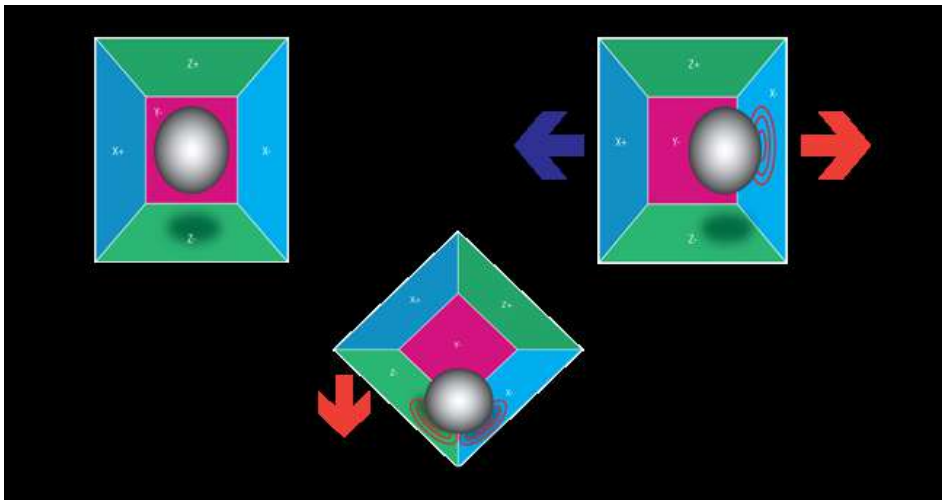


Figure II1.19 : Accéléromètre Piezo Electric

Caractéristiques Accéléromètre :

- Digital-output triple-axis accéléromètre avec une plage programmable à pleine échelle ±2g, ±4g, ±8g et ±16g

- tolérance de calibration: ±3%

- interface de communication : I2C

III.3.3.3 Avantages des capteurs MEMS:

Actuellement les capteurs MEMS sont dépassés en termes de précisions par les capteurs optiques. Toutefois ils sont les plus utilisés dans les applications courantes (car bas coût). Les principaux avantages des capteurs MEMS sont [32] :

- Dimensions réduites

- Faible poids

- Faible consommation de puissance

- Court temps de démarrage

- Production peu coûteuse (à grande échelle)

- Fiabilité élevée

- Entretien réduit

III.3.4 Le contrôleur L298N:

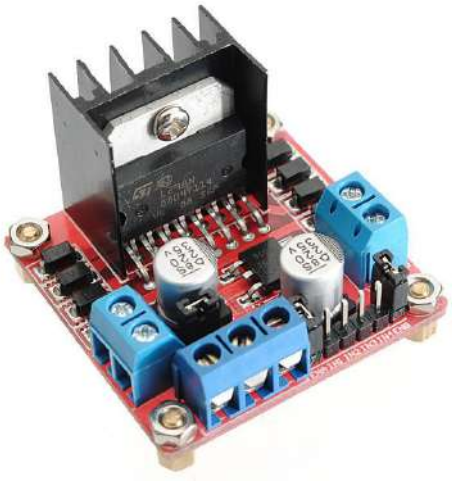


Figure III.20 : contrôleur L298N

Ce " breakout board " est un **Double Pont-H destiné au contrôle de moteur continu** (H-Bridge Motor Driver). Il est basé sur le composant L298N qui est un double Pont-H conçu spécifiquement pour ce cas d'utilisation. C'est un module extrêmement utile pour le contrôler de robots et ensembles mécanisés. Il peut contrôler deux moteur courant continu ou un moteur pas-à-pas 4 fils 2 phases. il est conçu pour supporter des tensions plus élevées, des courants importants tout en proposant une commande logique TTL (basse tension, courant faibles, idéal donc pour un microcontrôleur), il peut piloter des charges inductives comme des relais, solénoïdes, moteurs continus et moteurs pas-à-pas.

Les deux types de moteurs peuvent être contrôlés aussi bien en vitesse (PWM) qu'en direction. Toutes les sorties en puissance sont déjà protégées par des diodes anti-retour.
Il s'agit d'un module prêt à l'emploi. [32]

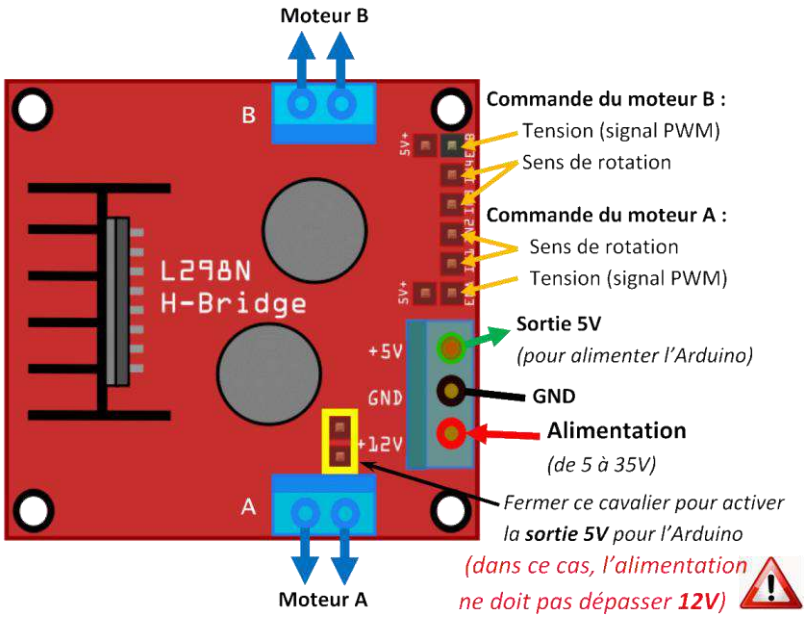


Figure III.21. les composantes de L298N

Pour active ou modifier le sens de rotation d'un moteur á courant continu, il suffitde r l'alimentation un ses bornes (induit). La structure permettant de réaliser cette inversion est appelée« pont en H ».

a) **Le Principe de fonctionnement du pont en H**

Le pont en H peut être symbolisé par une structure à quatre interrupteurs ouverts au repos (Figure *III.25*). Dans ce cas, la tension aux bornes du moteur est nulle. Pour mettre en rotation le moteur il suffit de fermer un couple d'interrupteurs (voir Figure *IV. 14*) et de laisser les deux autres au repos . Le moteur est donc alimenté avec la tension UM=VCC.

Pour modifier le sens de rotation, il suffit de permuter le couple d'interrupteurs fermés et ouverts. On aura donc UM= - VCC. [30]

Le L298 est un module qui dispose de 2 hacheurs 4-quadrants dans un circuit intégré de type Multi watt, Figure (4.5). Le hacheur est un convertisseur statique continu-continu permettant de fabriquer une source de tension continue variable à partir d'une source de tension continue fixe.

Il reçoit les signaux de commande PWM (*Pulse Width Modulation*) ou MLI (*Modulation de Largeur d'Impulsion*) issus de l'Arduino. Selon le rapport cyclique de ces signaux, le hacheur délivre la tension adéquate.

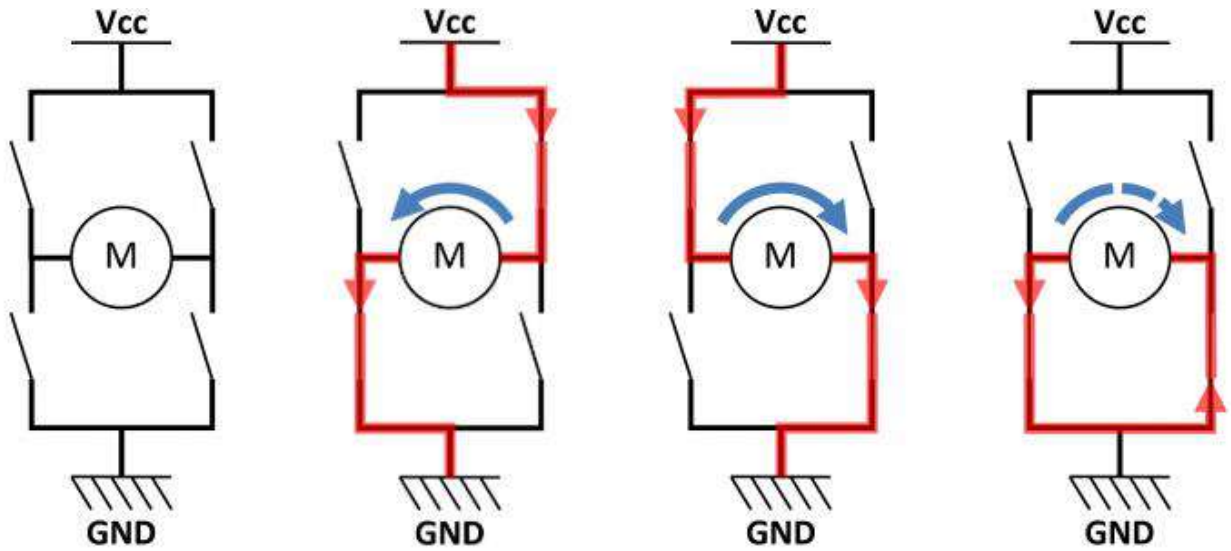


Figure III.22: Schéma de principe de fonctionnement du pont H.

b) caractéristiques du L298N : Ce tableau résume les caractéristiques et spécifications du de L298N:

Tableau III.3 : Les caractristique du L298N

les caractéristiques	Les spécifications
Léger , petit	Alimentation de la charge: de +6V à+35V
Des capacités hors-pair pour contrôle moteur	Courant Max (en pointe):2A
4 trous de fixation standard	Tension de commande logique Vss: de +5 à +7V (alimentation interne de+5V)
4 trous de fixation standard	Courant de commande logique: de 0 à36mA
Indicateur LED pour alimentation5V et de rotation moteur	Tensions pour contrôle du sens:Low - 0.3V~1.5V, high:2.3V~Vss
4 Sélecteurs pour les résistances pull up	Tensions pour contrôle "Enable":Low - 0.3V~1.5V, high:2.3V~Vss
Un sélecteur pour sélectionner la source D'alimentation	Puissance Max: 25W (Température 75 ℃)
Un dissipateur (pour dissiper la chaleur en cas de forte charge)	Température de fonctionnement: de -25℃ à+130℃
Un dissipateur (pour dissiper la chaleur en cas de forte charge)	Dimensions: 60mm x54mm
Diodes de protections	Poids: ~48g

III.3.5 Le capteur HC-SR04 :

Le capteur HC-SR04 utilise les ultrasons pour déterminer la distance d'un objet. Il offre une excellente plage de détection sans contact, avec des mesures de haute précision et stables. Son fonctionnement n'est pas influencé par la lumière du soleil ou des matériaux sombres, bien que des matériaux comme les vêtements puissent être difficiles à détecter.[34]

Le principe de fonctionnement du capteur est entièrement basé sur la vitesse de son Voilà comment se déroule une prise de mesure :

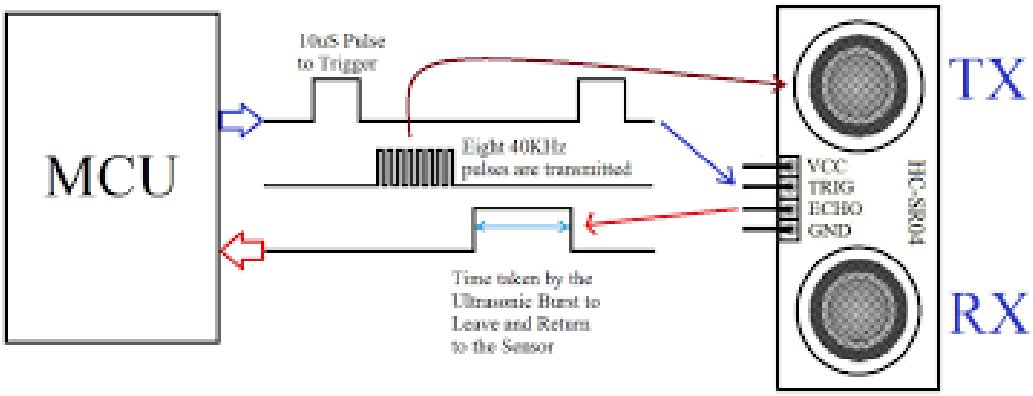


Figure III.23 : principe de fonctionnement de HC-RS04

1. On envoie une impulsion HIGH de 10µs sur la broche TRIGGER du capteur.
2. Le capteur envoie alors une série de 8 impulsions ultrasoniques à 40KHz
3. Les ultrasons se propagent dans l'air jusqu'à toucher un obstacle et retourne dans l'autre sens vers le capteur(vitesse du son dans l'air (340m/s). .
4. Le capteur détecte l'écho et clôture la prise de mesure. Le signal sur la broche ECHO du capteur reste à HIGH durant les étapes 3 et 4, ce qui permet de mesurer la durée de l'aller-retour des ultrasons et donc de déterminer la distance.

Le montage est d'une simplicité déconcertante :

-L'alimentation 5V de la carte Arduino va sur la broche VCC du capteur.

-La broche GND de la carte Arduino va sur la broche GND du capteur.

- La broche D2 de la carte Arduino va sur la broche TRIGGER du capteur.

- La broche D3 de la carte Arduino va sur la broche ECHO du capteur.

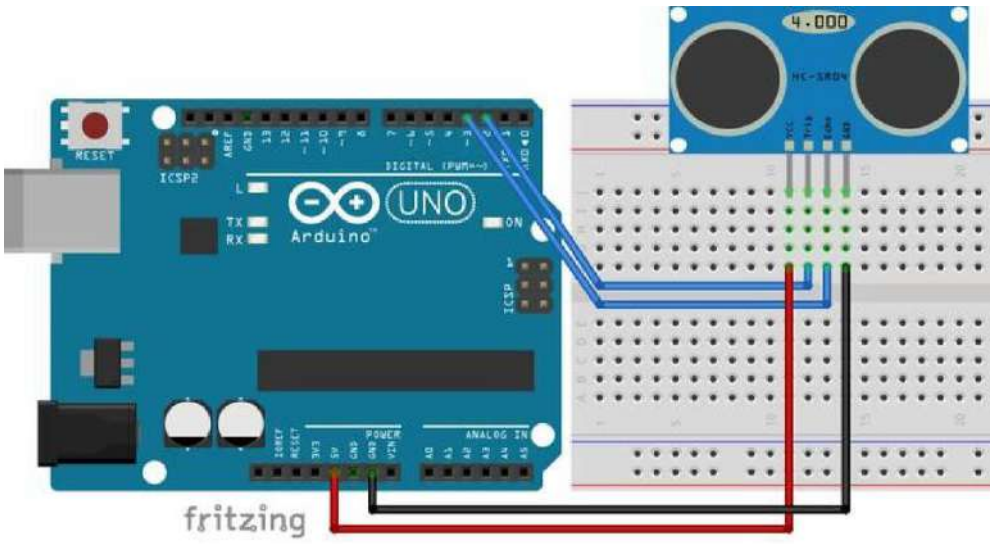


Figure III.24 : brochage du HC-RS04

Caractéristique du HC-RS04

- Dimensions : 45 mm x 20 mm x 15 mm
- Plage de mesure : 2 cm à 400 cm
- Résolution de la mesure : 0.3 cm
- Angle de mesure efficace : 15 °
- Largeur d'impulsion sur l'entrée de déclenchement : 10µs (Trigger Input Pulse width)

III.3.6 Circuit de commande ARDUINO :

ARDUINO est une plateforme matérielle et logicielle de développement d'applications embarquées. Côté matériel, elle se compose d'une carte électronique basée autour d'un microcontrôleur (ATMEL AVR) comportant un certain nombre d'entrées et de sorties (les ports) permettant la connexion de capteurs, ou d'actionneurs. Le logiciel de programmation des modules ARDUINO est une application Java, libre et multi- plateformes, servant d'éditeur de code et de compilateur, et qui peut transférer le programme au travers de la liaison USB. Le langage de programmation utilisé est un mélange de C et de C++, restreint et adapté aux possibilités de la carte. [36]

a) ARDUINO UNO

Le modèle UNO de la société ARDUINO est une carte électronique dont le cœur est un microcontrôleur ATMEL de référence ATMega328. Le microcontrôleur ATMega328 est un microcontrôleur 8bits de la famille AVR dont la programmation peut être réalisée en langage C. Les différents éléments.[36]

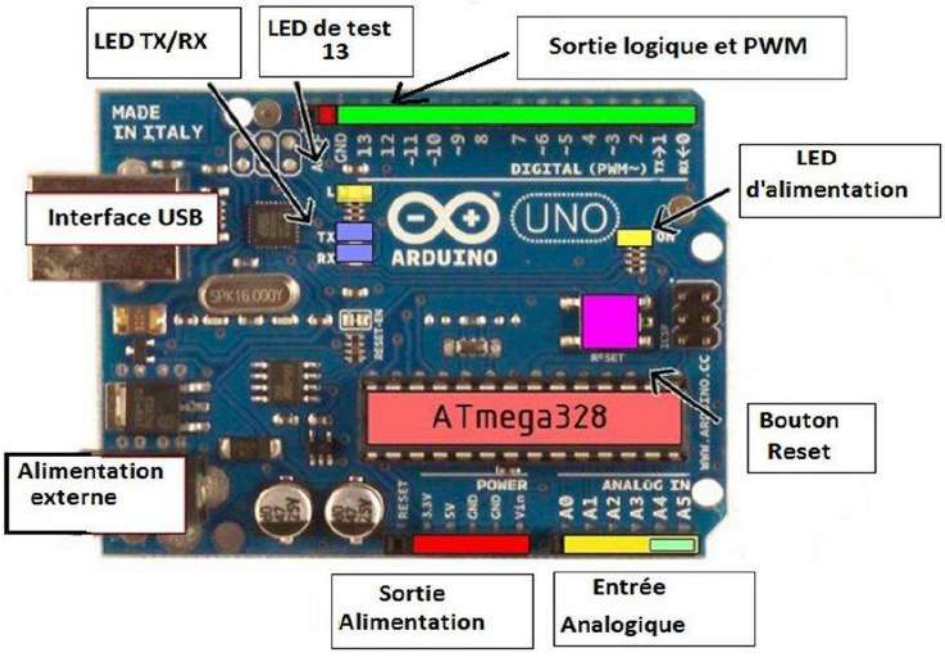


Figure III.25 : les entres et les sortie d' ARDUINO UNO

b) Les principaux caractéristique :

Tableau III.4 : caractéristique du UNO

Microcontrôleur	ATmega328P
Tension de fonctionnement	5V
Tension d'entrée (recommandé)	7-12V
Tension d'entrée (limite)	6-20V
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Mémoire flash	32 KB (ATmega328P)
Courant max par broches E/S	S 40 mA (200mA cumulé pour l'ensemble des broches)

c**) Programmation de la carte ARDUINO**

Le logiciel ARDUINO a pour fonctions principales :

- De pouvoir écrire et compiler des programmes pour la carte ARDUINO

- De se connecter avec la carte ARDUINO pour y transférer les programmes

- De communiquer avec la carte ARDUINO

L'interface de programmation L'interface de programmation du logiciel ARDUINO est la suivante :

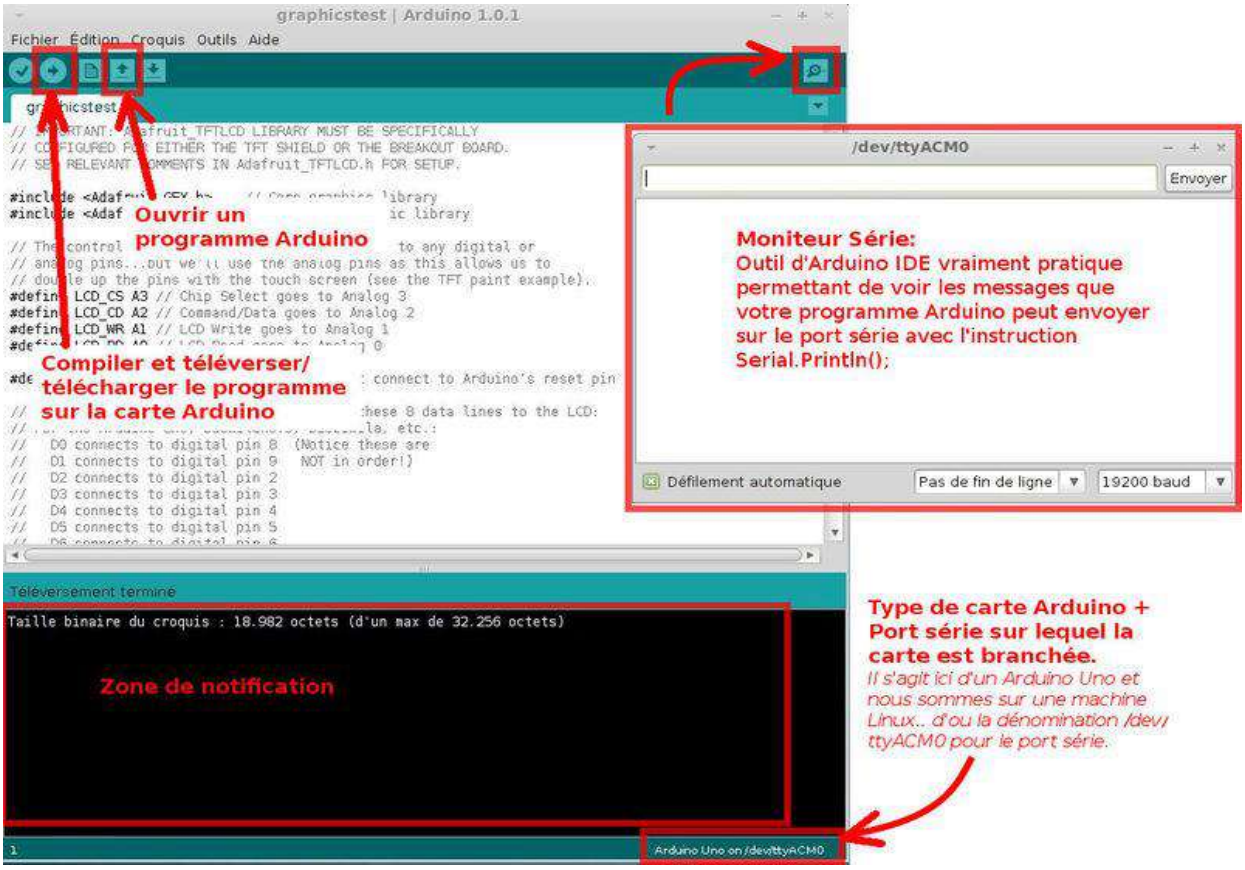


Figure III.26 : l'interface IDE

III.4 Conclusion

On a exposé les deux descriptions fondamentales de notre robot mobile avec le circuit de commande .

premièrement, la description mécanique qui fait appel à une description de tous partie , cela nous a conduit à citer les matériaux utilisés et leurs critères qui permettent non seulement la réalisation de notre projet mais encore son bon fonctionnement avec les actionneurs utilisés. un détail sur chaque compensant est ajouté à la fin de cette première description.

La deuxième e partie de ce chapitre est la description électronique, elle parle de circuit d'alimentation du robot et le circuit de commande qui détaille les différents composants de la carte électrique et le fonctionnement de chaqu'une avec les capteurs utilisés, nous terminera ce chapitre par l'arduino qui est le cœur du commande.

CHAPITRE VI

Implémentation et tests de la commande

IV.1 Introduction:

Pour procéder notre réalisation de robot mobile auto-balancé il fallait de commander le robot, pour cela nous avons utilisé la commande classique par PID pour stabiliser l'angle d'inclination qui est fournis par un capteur MPU6050 et le corriger par un filtre complémentaire.

Dans ce chapitre nous nous intéresserons à la commande hardware du robot avec les résultats obtenues, le schéma de la figure IV.1 montre la chaine d'acquisition des données qui explique comment les informations sont transférées du MPU6050 vers les deux moteurs (le transfert des angles) à travers la carte arduino et le circuit puissance.

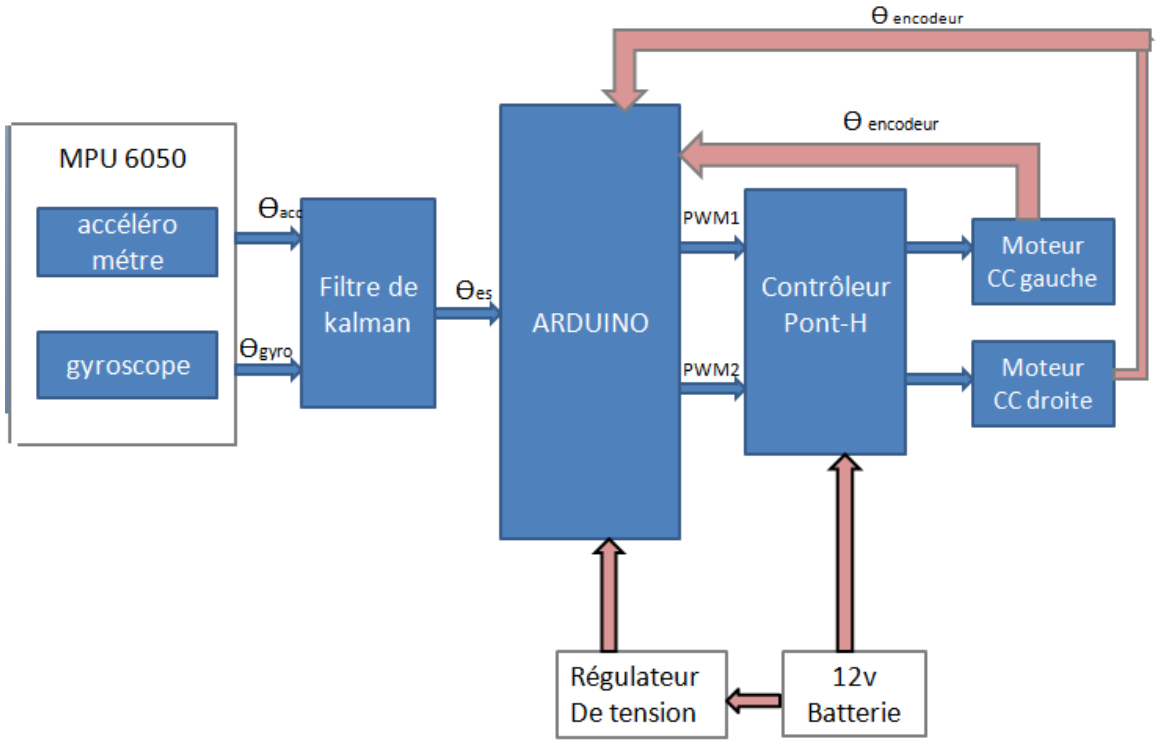


Figure IV.1: Schéma synoptique de la commande

Les entrées : deux angles fournis par le capteur mpu6050 .

les sorties :

a) translation du robot.

b) changement de l'angle pour ajuster le robot au point d'équilibre.

Le transfert des données entre l'MPU6050 et l'arduino est fait à l'aide de la liaison I2C, Le circuit de commande va convertir le code générant le signal de commande qui correspond à l'information obtenue par le l'MPU6050.

Finalement, les actionneurs transforment l'information électrique vers une rotation mécanique . L'ensemble de la structure mécanique du robot se met en mouvement pour assurer l'auto-équilibre.

IV.2 Implémentation du programme sur la carte arduino :

IV.2.1 Calibration du module MPU6050 :

Premièrement avant l'utilisation du capteur MPU6050 il faut le calibrer, La calibration est une étape indispensable avant le démarrage du calcul, il sert à fixer les références (0,0,0) du module. Pour cela, nous avons appliqué l'algorithme suivant :

- Mettre le module sur une plate horizontale;
- Mettre tous les offsets à 0 par l'instruction :

("SetXGyroOffset/setYGyro.ffset/setZGyroOffset/ setZAccelOffset"=0);

- Passer le programme à Arduino et ouvrir le moniteur en série pour voir les valeurs retournées;
- Laisser le module opérant pour quelques minutes (5-10) donc la température est stabilisée;
- Vérifier les valeurs dans le moniteur en série et on les note;
- Mettre à jour nos offsets et faire passer le programme à nouveau à la carte;
- Répéter ce processus jusqu'à ce que notre programme retourne a 0 pour chaque gyro, 0 pour X et Y accel et +16384 pour Z accel.[37].

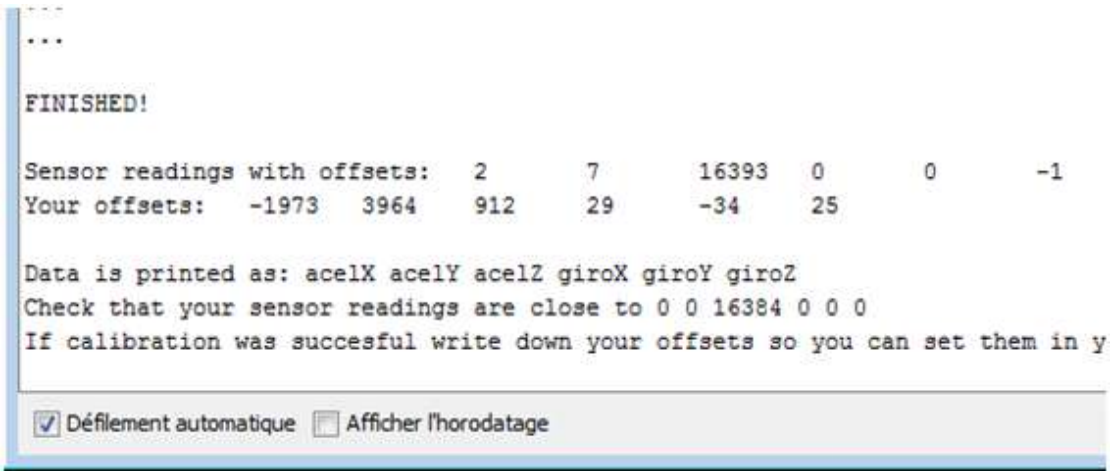


Figure IV.2 : Résultat de la calibration

Une fois le composant est branché correctement et programmé on peut visualiser les valeurs retournées sur la console Arduino.

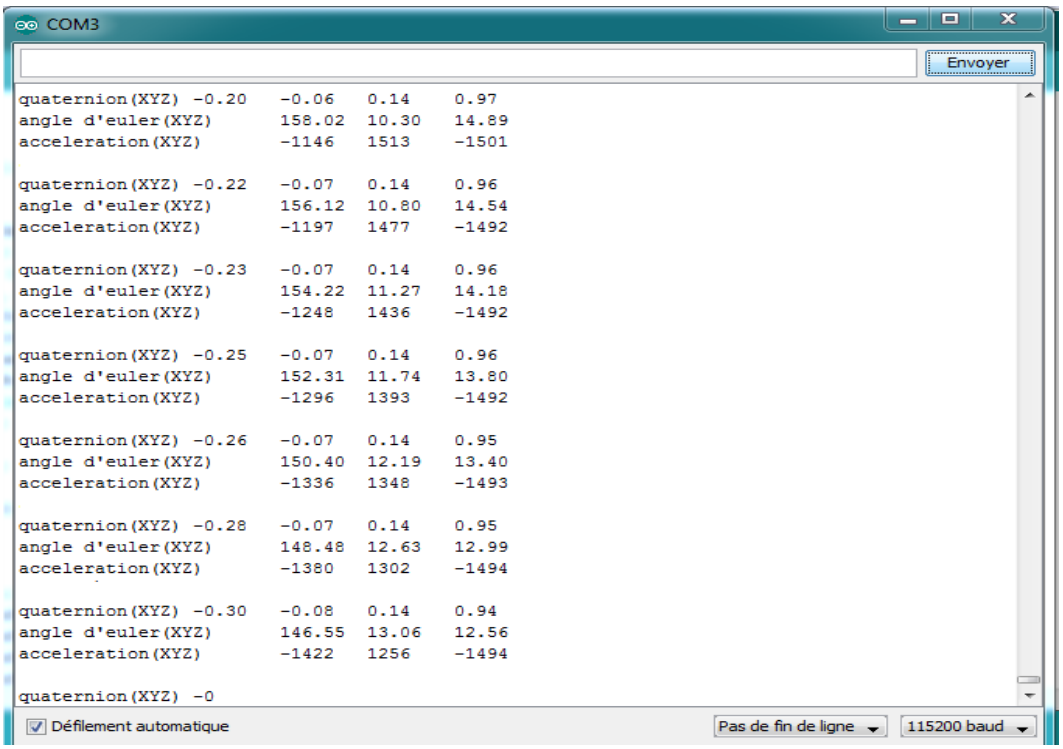


Figure IV.3: Valeurs retournées par le MPU6050

IV.2.2 Détermination de l'angle d'inclinaison :

Le MPU6050 possède un accéléromètre à 3 axes et un gyroscope à 3 axes. L'accéléromètre mesure l'accélération sur les trois axes et le gyroscope mesure la vitesse angulaire autour des trois axes.

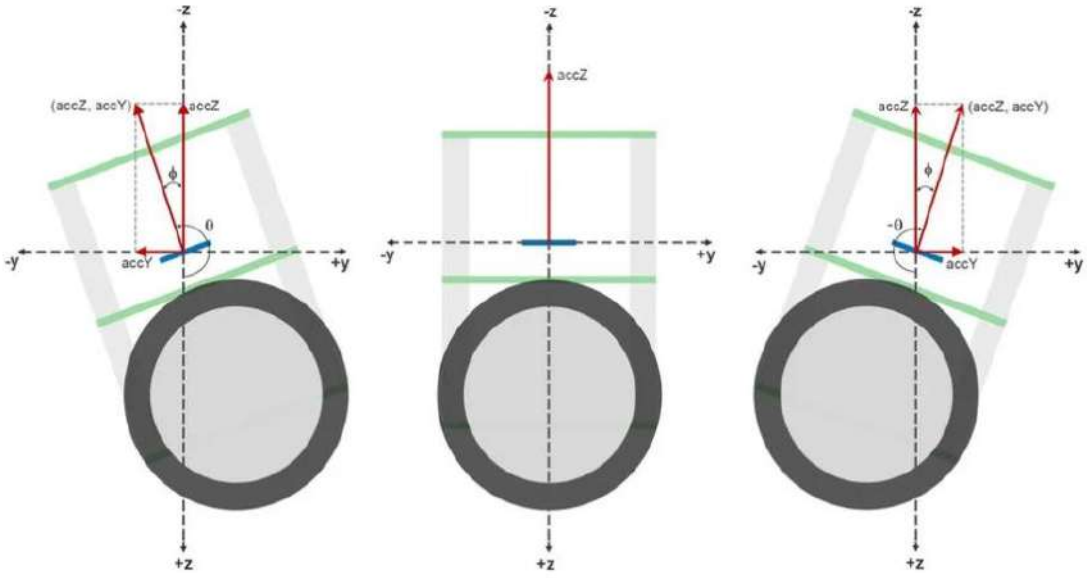


Figure IV.4: Le changement du l'Angel par rapport à l'axe Z .

a) **mesure de l'angle avec l'accéléromètre :**

Pour mesurer l'angle d'inclinaison du robot, nous avons besoin des valeurs d'accélération sur les axes y et z. La fonction **atan2 (y, z)** donne l'angle en radians entre l'axe z positif d'un plan et le point donné par les coordonnées (y, z) sur ce plan, avec le signe positif pour les angles dans le sens antihoraire (demi-droite). plan, y> 0) et signe négatif pour les angles dans le sens des aiguilles d'une montre (demi-plan gauche, y<0).

On peut faire ça avec le programme suivant :

```
void loop() {
  accZ = mpu.getAccelerationZ();
  accY = mpu.getAccelerationY();

  accAngle = atan2(accY, accZ)*RAD_TO_DEG;

  if(isnan(accAngle));
  else
    Serial.println(accAngle);
}
```

b) mesure de l 'angle avec le gyroscope :

Le gyroscope à 3 axes du MPU6050 mesure la vitesse angulaire (vitesse de rotation) le long des trois axes. Pour notre robot à auto-équilibrage, la vitesse angulaire sur l'axe des abscisses suffit à elle seule pour mesurer le taux de chute du robot.

Dans le code ci-dessous, nous lisons la valeur du gyroscope autour de l'axe des x, la convertissons en degrés par seconde, puis la multiplions par le temps de boucle pour obtenir le changement d'angle. Nous ajoutons ceci à l'angle précédent pour obtenir l'angle actuel.

```
void loop() {
  currTime = millis();
  loopTime = currTime - prevTime;
  prevTime = currTime;

  gyroX = mpu.getRotationX();
  gyroRate = map(gyroX, -32768, 32767, -250, 250);
  gyroAngle = gyroAngle + (float)gyroRate*loopTime/1000;

  Serial.println(gyroAngle);
}
```

La position du MPU6050 au démarrage du programme est le point d'inclinaison , dans notre cas c'est 180 dégrée . L'angle d'inclinaison sera mesuré par rapport à ce point.
l'angle mesuré augmente ou diminue progressivement. Ça ne va pas rester stable. Cela est dû à la dérive inhérente au gyroscope.

Dans le code ci-dessus, le temps de boucle est calculé à l'aide de la fonction **millis()** intégrée à l'IDE Arduino. Dans les étapes suivantes, nous utiliserons des interruptions de minuterie pour créer des intervalles d'échantillonnage précis. Cette période d'échantillonnage sera également utilisée pour générer la sortie à l'aide d'un contrôleur PID.
le module MPU-6050 utilise le protocole I2C pour communiquer avec la carte Arduino.

c) élimination de l'offset du MPU6050 :

Après que nous avons calibrer le MPU6050 ,nous avons implémenté les valeur des offset pour l'étalonnage du mpu6050 comme suite :

```
mpu.initialize();

devStatus = mpu.dmpInitialize();
mpu.setXGyroOffset(25);
mpu.setYGyroOffset(-34);
mpu.setZGyroOffset(29);
mpu.setZAccelOffset(912);
```

IV.2.3 Acquisition des donnés :

De nombreuses applications complexes qui relèvent du domaine de l'électronique ont pu voir le jour, dans leur majorité, ces applications sont développées soit à base de microcontrôleur, de microprocesseur ou encore de circuits logiques complexes. Cependant l'utilisation de ces composants dans un système, exige un protocole de communication. Dans cette section nous allons présenter brièvement le protocole I2C qu'on a utilisé dans notre projet.

Le bus **I2C** (**Inter Integrated Circuit**) a été développé par l'entreprise Philips. Ce bus est utilisé pour connecter les composants périphériques microcontrôleur. Le bus I2C est composé de deux fils, ce qui réduit la partie hardware de façon drastique. Ce bus comprend une ligne d'horloge **SCL (Serial Clock)** et une ligne de donnée **SDA (Serial Data)**. La communication est par conséquent synchrone. Son mode d'utilisation est souvent du type maître et esclave. Cependant, il soutient également le mode multi-maître.

La Figure suivante illustre une application typique avec le bus I2C:

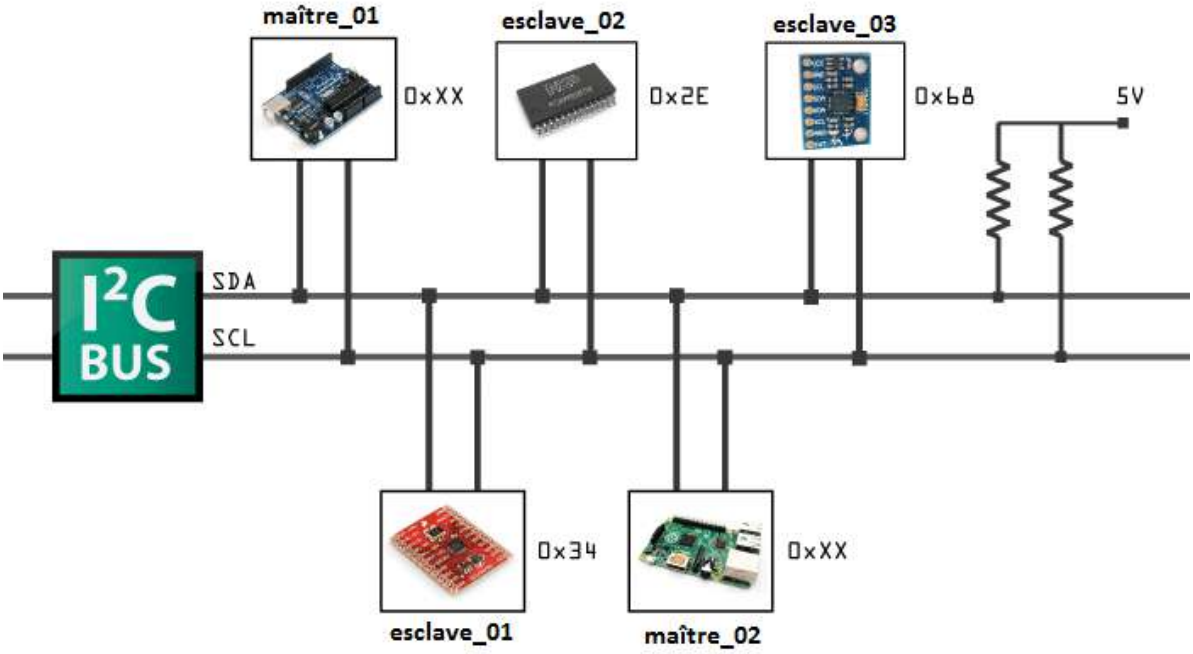


Figure IV.5 : Mode de Fonctionnement I2C

Principe de fonctionnement :

Le transfert des données entre le maître et l'esclave s'effectue de la manière suivante : Le maître démarre le transfert des données en envoyant le bit de démarrage. Tous les esclaves entrent ainsi dans un état passif d'écoute. Le maître envoie ensuite l'adresse de l'esclave, avec lequel il aimerait communiquer. Cette adresse est composée de 7 bits. Le maître envoi le bit R/W, qui fixe le sens de la communication : Ecriture depuis le maître à l'esclave ou lecture depuis l'esclave au maître. Le protocole I2C exige des confirmations (acknowledge) après chaque transmission d'un octet. L'esclave accrédite avec le premier bit de confirmation, qu'il est prêt pour la communication. Après chaque transmission d'un octet. Ce bit est fourni par le maître dans le mode lecture et par l'esclave dans le mode écriture. Le transfert se termine avec un bit d'arrêt (Stop).

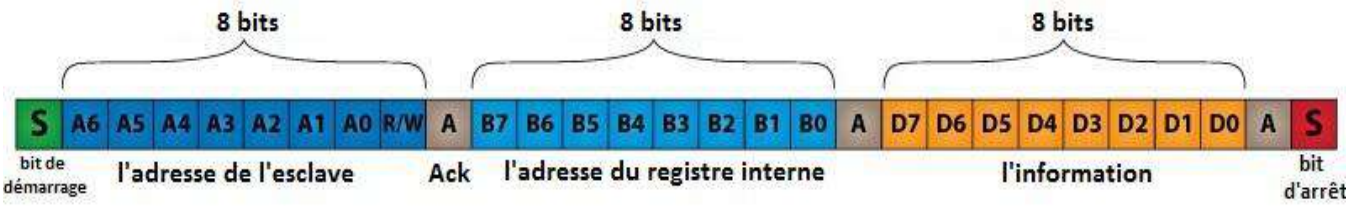


Figure IV.6 : Le bus de transfert I2C

IV.2.4 Fusionner les résultats avec le filtre complémentaire :

Un filtre complémentaire est utilisé quand il y a deux sources de mesure pour l'estimation de la même variable et les propriétés de bruit des deux mesures sont telles qu'une source ne peut fournir une mesure précise que dans la gamme de basses fréquences tandis que l'autre est exacte dans la région des hautes fréquences. [38]

Le schéma de principe est présenté sur la figure VI.7 et sera appliqué sur notre robot

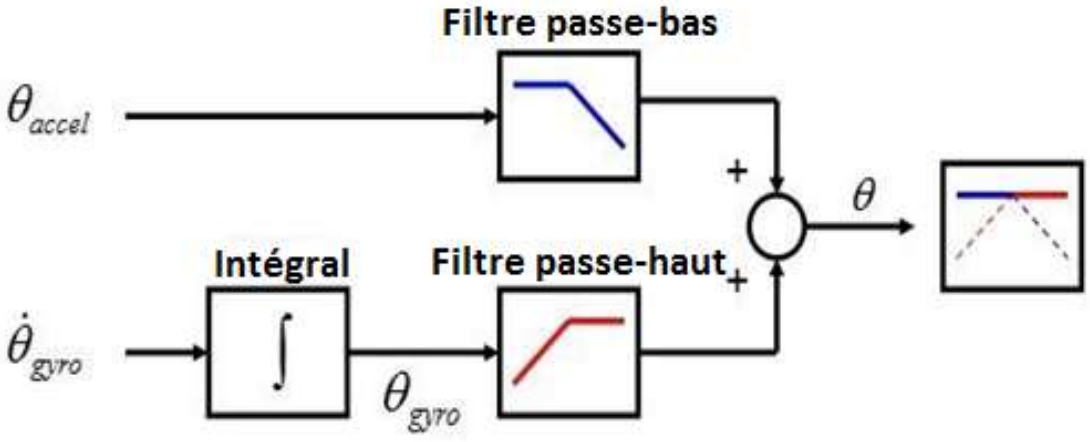


Figure VI.7 : La fusion du capteur par filtre complémentaire

La figure VI.7 montre que la vitesse angulaire mesurée par le gyroscope a été intégrée pour produire une estimation d'angle et a été passée par un filtre passe-haut. Ce dernier agit pour annuler l'effet du gyroscope à la gamme des basses fréquences et ne laisse passer que les signaux dans la gamme des hautes fréquences où ils sont plus précis. L'autre angle fourni par l'accéléromètre passe à travers un filtre passe-bas. Les mesures de l'accéléromètre ne sont exactes que dans le long terme ou dans la gamme des basses fréquences. Le filtre passe-bas agit pour annuler les composantes haute fréquences. On ne recherche en effet que la partie continue qui est un terme en g*sin (θ)

- **implémentation du filtre :**

```
}
double a=0.98;
double x_angleC = 180;
double Complementary(double newAngle, double newRate,double looptime) {
double dtC = float(looptime)/1000000.0;
x_angleC= a* (x_angleC + newRate * dtC) + (1-a) * (newAngle);
return x_angleC;
}
```

IV.2.5 L'utilisation du contrôleur PID:

En fait, les paramètres mécaniques du modèle lors de la simulation étaient approximatifs ; le prototype utilisé ne correspond pas à ces paramètres c.à.d. que les gains calculés par placement de pôles en simulation, ne conviendraient pas pour le prototype. De plus, comme les états ne sont pas tous mesurables et que nous avons montré qu'on obtenait des résultats identiques avec un régulateur PID, nous avons donc implanté en finalité sur ARDUINO le régulateur PID car il est moins gourmant en temps de calcul.

La simulation a montré un résultat très important, c'est que le système est commandable et que la stabilité peut être réalisée.

L'algorithme de contrôle PID est utilisé pour le contrôle de presque toutes les boucles dans les industries de transformation, et il est également la base pour de nombreux algorithmes de contrôle et des stratégies avancées.

L'équation d'un régulateur PID est donnée comme suit :

$$u(t) = K_p . e(t) + K_t . \int_0^t e(t)dt + K_d . \frac{de(t)}{dt}$$

Par l'utilisation de la méthode de Ziegler et Nichols on peut trouver les paramètres Kp, Ki et Kd du PID, premièrement on a augmenté la valeur du Kp, plus le gain proportionnel est grand, plus la réponse est rapide et l'erreur statique est petite, mais il y a le problème des oscillations qui sont importantes durant la phase transitoire, pour cela, on a changé la valeur du Kd qu' elle permet de réduire le dépassement et les oscillations, mais elle amplifie les bruits de mesure, finalement on a changé la valeur du Ki pour annuler l'erreur statique.

```
//PID
double originalSetpoint = 178;
double setpoint = originalSetpoint;
double movingAngleOffset = 0.1;
double input, output;
double Kp =250;
double Kd = 30;
double Ki = 40;
```

Figure IV.8 : Les valeurs finales du paramètre PID

Dans la fonction de boucle principale, nous vérifions si les données du MPU6050 sont prêtes à être lues. Si c'est le cas, nous l'utilisons pour calculer la valeur du PID, puis nous affichons la valeur d'entrée et de sortie du PID sur le moniteur série afin de vérifier la réponse du PID. Ensuite, en fonction de la valeur de la sortie, nous décidons si le bot doit avancer ou reculer ou rester immobile.

Puisque nous supposons que le MPU6050 retournera 178 lorsque le robot sera debout. Nous obtiendrons des valeurs de correction positives lorsque le bot tombera vers l'avant et des valeurs négatives si le bot reculera vers l'arrière. Nous vérifions donc cette condition et appelons les fonctions appropriées pour faire avancer ou reculer le robot.

❖ **implémentation PID:**

```
while (!mpuInterrupt && fifoCount < packetSize)
  {
      //no mpu data - performing PID calculations and output to motors
      pid.Compute();

      //Print the value of Input and Output on serial monitor to check how it is working.
      Serial.print(input); Serial.print(" =>"); Serial.println(output);

      if (input>150 && input<200){//If the Bot is falling

      if (output>0) //Falling towards front
      Forward(); //Rotate the wheels forward
      else if (output<0) //Falling towards back
      Reverse(); //Rotate the wheels backward
      }
      else //If Bot not falling
      Stop(); //Hold the wheels still
  }
```

la figure IV.9 exprime l'angle désiré qui est le setpoint déclaré et l'angle actuel du robot. l'erreur est la différence entre les deux angles , les effets du régulateur PID permet de balancer le robot par les moteurs DC , Pour équilibrer le robot, le point de consigne (setpoint) doit égal dans ce cas à 178 dégrée.

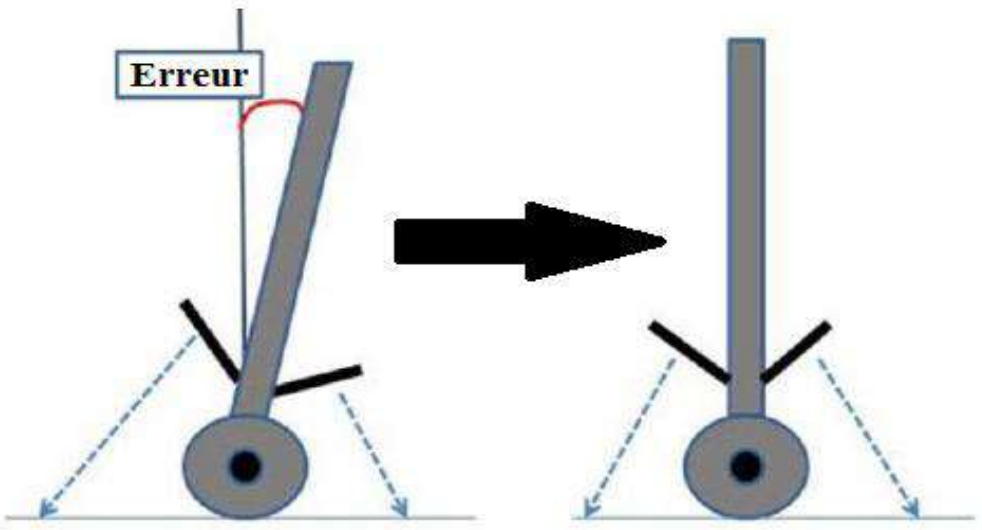


Figure IV.9 : Correction de l'angle a partir la consigne .

VI.2.6 Génération des impulsions de commande:

Pour varier la vitesse d'un moteur on peut varier la tension d'alimentation à ses bornes mais dans ce cas une partie importante de l'énergie est consommée par le dispositif d'alimentation, on préfère l'alimenter de façon discontinue avec un hacheur et ainsi varier la

tension moyenne à ses bornes. On parle alors de Modulation par Largeur d'Impulsions **(MLI)**, ou Pulse Wide Modulation **(PWM)**. La sortie PWM du microcontrôleur est utilisée pour commander le L298N. [39]

```
double vitessMoteurGauche =0.5;
double vitessMoteurDroite = 0.5;
//Déclaration des pins
int ENA = 5;
int IN1 = 6;
int IN2 = 7;
int IN3 = 8;
int IN4 = 9;
int ENB = 10;
```

Figure IV.10 : Déclarations des pins L298N

a) Programme qui commande les moteurs :

```
void MotorL298()
{
  output = (output / 2);
  if (output > 255)
  {
    output = 255;
  }
  if (output < -255)
  {
    output = -255;
  }
  if (output < 0)
  {
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
    analogWrite(ENA, (output*-1));
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
    analogWrite(ENB, (output*-1));
  }
  else
  {
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
    analogWrite(ENA, output);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);
    analogWrite(ENB, output);

  }
}
```

b) Interprétation du programme :

Pour la rotation du moteur A:

- IN1 raccordé à 5V et IN2 à GND moteur A tournera dans le sens horaire.

- IN1 raccordé à GND et IN2 à 5Vmoteur A tournera dans le sens Anti-horaire.

Pour la rotation du moteur B:

- IN3 raccordé à 5V et IN4 a GND moteur B tournera dans le sens horaire.
- IN4 raccordé à GND et IN4 à 5V moteur B tournera dans le sens Anti-horaire.

Si vous voulez contrôler la vitesse, vous pouvez connecter (ENA/ENB) sur une sortie PWM.

IV.2.7 Organigramme du programme implémenté :

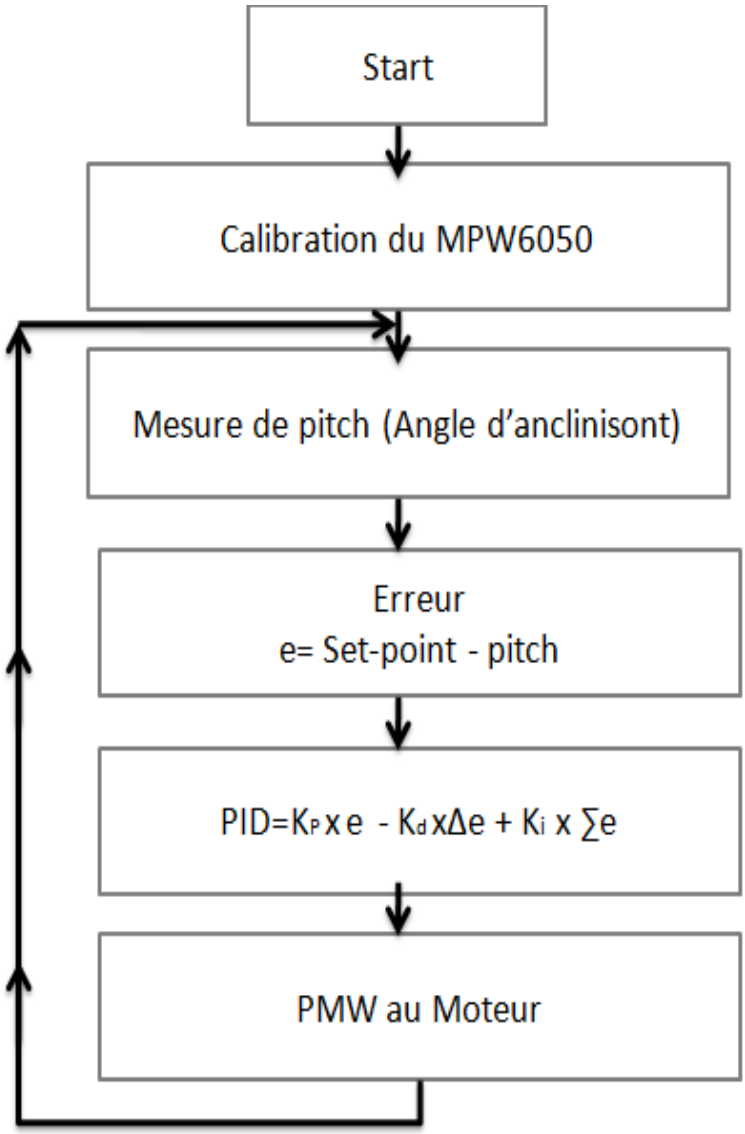


Figure IV.11: Organigramme du programme implémenté.

VI.3 Performance obtenue :

La figure IV.11 présente un test d'auto-équilibrage de notre robot, on peut remarquer que le robot oscille de droite et gauche pour maintenir l'équilibre.

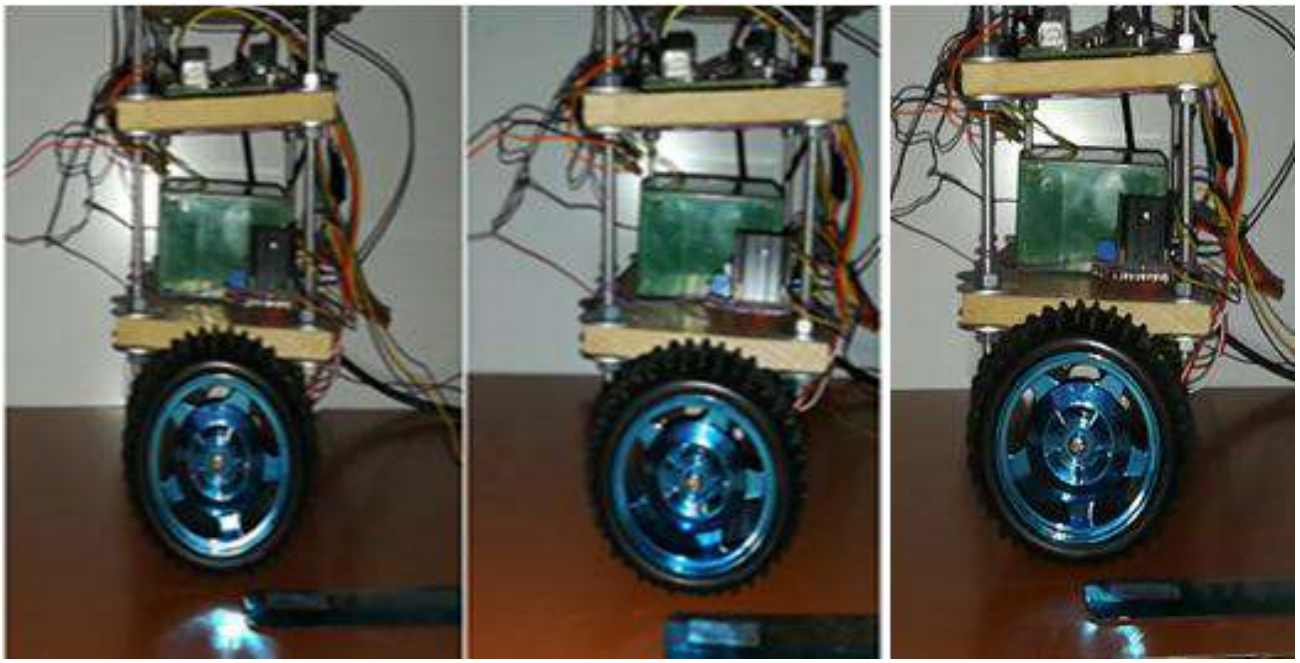


Figure IV.12 : Test sur le robot .

le bon fonctionnement du robot prouve que l'implémentation de filtre complémentaire et la commande par PID a ressuie, le robot est stable, il oscille parfaitement sur $\pm$ un degré.

VI.4 Conclusion :

L'expérimentation a montrée plusieurs difficultés et inexactitudes que l'étude théorique et de simulation n'ont pas pu mettre en exergue. Après de nombreux essais à chercher comment contourner le problème de la mesure de l'angle à l'aide du gyroscope, un filtrage complémentaire a permis de réduire l'erreur sur l'estimation de l'angle d'inclinaison en fusionnant deux mesures d'une seule variable, issues de deux capteurs différents.

L'implémentation du régulateur PID était nécessaire pour le contrôle de la position angulaire. La variation de la vitesse est assurée par la variation de la tension d'alimentation, à l'aide de la technique de la commande par PWM.

Conclusion Générale

Conclusion générale

A travers le travail accompli et les efforts consentis tout au long de la réalisation de ce projet qui a finalement réussi à atteindre ses objectifs d'équilibrer un robot à deux roues basé sur le modèle du pendule inversé. Une stratégie de contrôle a été mis en œuvre pour résoudre le problème de contrôle de l'équilibre de système. O na choisi la commande par PID avec un filtre complémentaire implémenter dans la carte de contrôle de robot, depuis le test le robot est capable de maintenir sa position verticale en ajustant légèrement ses roues. Ceci est réalisé au prix d'un effort énorme, commencent par la réalisation de la structure mécanique du robot vers la réalisation électrique en passant par les aspects suivants :

Aspect mécanique

- ✓ Etude du robot auto-balance du point de vue réalisation mécanique : la structure, le raccordement des pièces entre eux, du point de vue calcule du force : l'inertie de corps et de rue ,centre de masse gravites du robot .
- ✓ Etude pratique sur le robot du point de vue test des performances obtenu :
  - L' équilibre du robot au consignes désires

Aspect électronique

- ✓ Etude théorique et pratique sur le MPU6050.
- ✓ Etude théorique et pratique sur la carte du puissance.
- ✓ Etude théorique et pratique sur la carte Arduino .
- ✓ Etude théorique et pratique sur le HC-SR04
- ✓ Etude théorique sur le principe de la commande des servomoteurs et les moteurs a courent contenu .

Aspect informatique

- ✓ Programmation de la carte Arduino pour la commande des moteurs a courent continue.
- ✓ Etude théorique et pratique sur le protocole de communication I2C.
- ✓ évolution du notre compétence en C et métrise de l'utilisation du carte arduino.
- ✓ utilisation de matlab pour simulé la command LQR.

Aspect mathématique

- ✓ développement de model mathématique du robot
- ✓ étude théorique sur la command LQR
- ✓ Etude théorique sure le de filtre Kalman
- ✓ Etude théorique et pratique sure le filtre complémentaire.

On peut dire que les résultats sur les tests des caractéristiques du robot sont acceptables, le robot balance au consignes désire, l'implémentation du PID avec le filtre complémentaire donné un bon résultat.

En effet ce projet de réalisation a été une expérience très fructueuse, bénéfique et enrichissante. Il nous a permis d'exploiter les différentes notions théoriques pour le contrôle du robot. Il nous a permis également de mettre un premier pas dans le monde pratique de la robotique.

Ce projet nous avons mis directement en contact avec des langages de programmation puissants tel que le langage C, ainsi que différents logiciels spécialisés tel que Matlab.

En fin nous espérons qu'on va améliorer notre travail en future, comme perspectives nous proposons :

- ✓ Amélioration de la précision et la rapidité du robot.
- ✓ Commander le robot par des technique de l'intelligence artificielle telle que logique floue, SVM, l'optimisation ,réseau de neurone, la command adaptative …
- ✓ Commande du robot àdistance.
- ✓ Intègres une caméra kinect pour la visualisation.
- ✓ la planification automatique du trajectoire

Bibliographie

[1] A.KOUIDER AISSA et al. M.BELLILA, « étude et réalisation expérimental d'une commende destinée à control pendule inversé »,Université Djilali Bounaama Khemis Miliana, Khamis Milana, Mémoire de Master, Automatique des systèmes de Production ,2016-2017 .

[2]: Ayça Göçmen, « Design Of Two Wheeled Electric Vehicle » , Master, Département de génie mécatronique, University Atilim, turke,2011.

[3]: Yahia Chabane , « Elaboration d'un contrôleur adaptatif pour un robot mobile basé sur les réseaux immunitaires artificiels».Mémoire d'ingénieur d'état en informatique, informatique , Université Saad Dahlab, Blida, 2006.

[4] D. Filliat, « Introduction», in *Robotique Mobile*, D. Filliat, Fondation. Paris Tech, ENSTA Eds. UNIT : 2004,.

[5] KHEMISSAT Abdelmouneim ,«Plateforme de prototypage rapide pour la robotique mobile : Application à un robot Auto-balancé, » Genie électrique ,mémoire de master en automatique , UMBB, Boumerdes:2015.

[6] Belkhadria Khemisti, «commande d'un robot mobile par réseaux de neurones artificiels », Mémoire en vue de l'obtention du diplôme de magister en électronique. Option: Robotique. Université El Hadj Lakhdar Batna,2011

[7] H. R. MEMARBASHI, « *Design and parametric control of co-axes driven two-wheeled balancing robot* ». Genie mecanique ,Mémoire de master en mécatronique, Université de Massey, 2010.

[8]: Bayle, B. «, Robotique mobile »: note de cour, école Nationale Supérieure de Physique de Strasbourg, 2009.

[9] Elodie Legale . https://dailygeekshow.com/fauteuil-roulant-abdominaux-segway/ . 31 août 2015

[10] R. C. Ooi, « Balancing a two-wheeled autonomous robot » University of Western Australia, vol. 3, 2003.

[11] A.KOUIDER AISSA et al. M.BELLILA, « étude et réalisation expérimental d'une commende destinée à control pendule inversé »,Université Djilali Bounaama Khemis Miliana, Khamis Milana, Mémoire de Master, Automatique des systèmes de Production ,2016-2017 .

[12] R. C. Ooi, « Balancing a two-wheeled autonomous robot », University of Western Australia, vol. 3, 2003.

[13] O. Jamil et al. M. Jamil et al. Y. Ayaz et. Ahmad, « Modeling, control of a two-wheeled self-balancing robot, » 2014 International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE),Islamabad, 2014, pp. 191-199.

[14] K. Kankhunthodl et al. V. Kongratana et al. A. Numsomran et al. V. Tipsuwanporn, « Self-balancing Robot Control Using Fractional-Order PID Controller »,IMECS 2019, 13-Mar-2019, 2019, Hong Kong.

[15] Y. Granjon, « Automatique - Systèmes linéaires, non linéaires, temps continu, temps discret ,représentation d'état: Cours et exercices corrigés. »Dunod, 2010.

[16] F. Mudry, « Ajustage des Paramètres d'un Régulateur PID».Ecole d'ingénieurs de Canton de Vaud, 2006.

[17] K. Kankhunthodl « Instrumentation et régulation en 30 fiches - Dunod ».Ecole d'ingénieurs de paris .2009

[18].Kalman R. E. (1960). A New Approach to linear Filtering and Prediction roblems.

[19]. R. E. KALMAN « A New Approach to Linear Filtering and Prediction Problems1» Transaction of the ASME- Journal of Basic Engineering, 1960, 82 (series D): 35-45.

[20]. S.Harkat, « Application du Filtre de Kalman sur la variabilité pluviométrique dans le bassin versant de Chellif » Algerie,Université de Chlef,2016,188pp.

[21] Soltesz, K et al. Cervin, A, «Projects in Automatic Control 2017»,Département de contrôle automatique, Institut technologique de Lund, Université de Lund,2017.

[22] site Ali Express, site www.aliepress.com, date dernière consultation : 01/12/2018.

[23]R. Clavel, « Robots parallèles» . Techniques de l'Ingénieur, Art. 7710, Vol. S, Juillet 1994.

[24] R. Clavel, « Conception d'un robot parallèle rapide à 4 degrés de liberté», Thèse Doctorat N°925, Ecole Polytechnique fédérale de Lausanne (EPFL), 1991.

[25] Hitachi Group,«3FM -4», Hitachi Chemical Energy Technology , 1987 .

[26] Hazem Zine eddine,« Réalisation d'une alimentation variable SEPIC avec régulation de tension et protection contre surcharge », Université Larbi Ben M'Hidi , Oum El Bouaghi , Génie Électrique,2017.

[27] site Ali Express, site www.aliepress.com, date dernière consultation : 01/12/2018

[28] CHABBAR Houria, « Innovation et réalisation d'une solution de géolocalisation combinée à la télécommunication », Université Sidi Mohamed Ben Abdellah, Maroc, Ingénieur d'Etat en systemes electroniques & telecommunications,2014.

[29] O. Woodman, « An introduction to inertial navigation, » Computer Laboratory, University of Cambridge, United Kingdom 2007.

[30] Invenses technologies, MPU-6000 and MPU-6050 Product Specification, 2013, [en ligne], date de dernière mise à jour : 19 août 2011 .13 juin 2014.

[31]M. Lozeau, « Commande par supervision de systems mécatroniques via internet », Thèse de magister, ÉCOLE POLYTECHNIQUE DE MONTRÉAL, 2009.

[32] D.H.Titterton and J. L. Weston, Strapdown inertial navigation technology. USA : The Institution of Electrical Engineers, 2004.

[33] Lucien Bachelard «Introduction sur le L298N»,MicroNET 91,2017,disponible sur : https: //docplayer.fr/22075221-Introduction-sur-le-l298n.html [consulté le: 18-Mar-2019]

[34] skywodd, « Mesurer une distance avec un capteur à ultrason HC-SR04 et une carte Arduino / Genuino»,Carnet du maker, 2016 .

[35] Lucien Bachelard, « HC-SR04 - Module de détection aux ultrasons - Utilisation avec Picaxe, lu.bachelard@bluewin.ch, Le 28 novembre 2015» ,

[36] -H.TAKHI et al. C.ATTACHI Redouane, «Conception et réalisation d'un robot mobile à base d'Arduino», Université Amar Telidji, Laghouat, Mémoire de Master, Instrumentation, 2015

[37] [I2CdevLib, 2013] How to decide Gyro and accelerometer offset, 2013, date de dernière mise à jour : 14decembre 2013,

[38] Baker, N. P et al. Brown, C. P et al . Dowling, D. R. S.et al Modra, J. L et Tootell, D. J., „„„« SON of EDGAR» , Dept.Mecha.Eng.,Adelaide Univ.,Australia, Oct, 2006.

[39] D. KOUMARI, «REGULATION EN CASCADE D'UN MOTEUR A COURANT CONTINU, » Mémoire de fin d'étude, FHC, 2008.

ANNEXE

A) Spécification du moteur cc 12 V 107 tr/min :

Tension nominale	DC 12 V
Vitesse à vide	107 RPM 0.15A
Max Efficacité	4.3kg.cm/85 rpm/3.8 W/0.60A
Max Puissance	10.0kg.cm/52 rpm/6.0 W/1.12A
Couple de décrochage	20.0kg.cm 3.4A
ralentisseur Rapport De Réduction	1: 90
Hall Résolution	salle x Ratio 11x0 = 990PPR

B) Force électromotrice :

$k = \frac{E}{\omega}$

$\omega = 2\pi N$

avec N = 107 tr/min

=> N =107/60 = 1.78 tr/min

$\omega = 2\pi N$ => 2 * 3.14 * 1.78 = 11.18 rad/s

$E_N = U_N - RIN = 12 - 5 * 1 * 2$

=> $k_e = 0.48\, rad\ v/s$

couple de moteur : $k_m = 0.57\ N.m$

Inertie des roues :

$I_W = \frac{1}{2}MR^2$ avec M = 0.056Kg et R = 0.04 m

Ensuit $I_r = \frac{1}{2} * (0.056Kg) * (0.04m)^2$

$I_r = 4.5 * 10^{-4}$Kgm²

C) Inertie du corps :

$I_C = \frac{1}{12} m(h^2 + d^2)$ avec h = 0.24m et d = 0.1m

Ensuit $I_C = \frac{1}{12} * (1.6Kg) * ((0.24m)^2 + (0.1m)^2)$

$$I_C = 0.0001\text{Kgm}^2$$

D) Longueur au centre de masse du corps

$$l = 0.11\text{m}$$

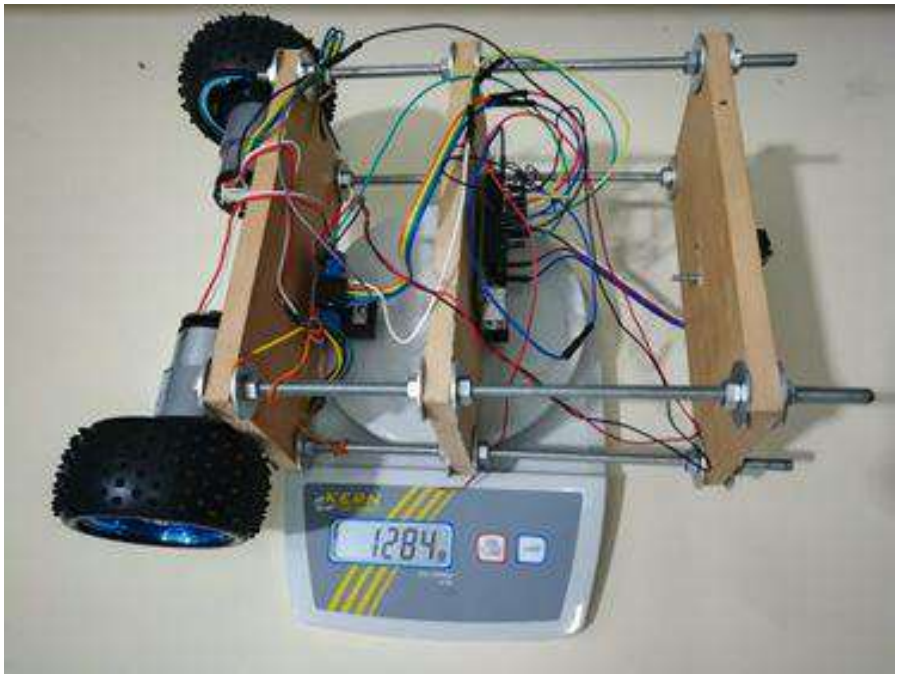


E) masse de roue = 56 g



F) masse du corps sans batterie

M_p = 1284g



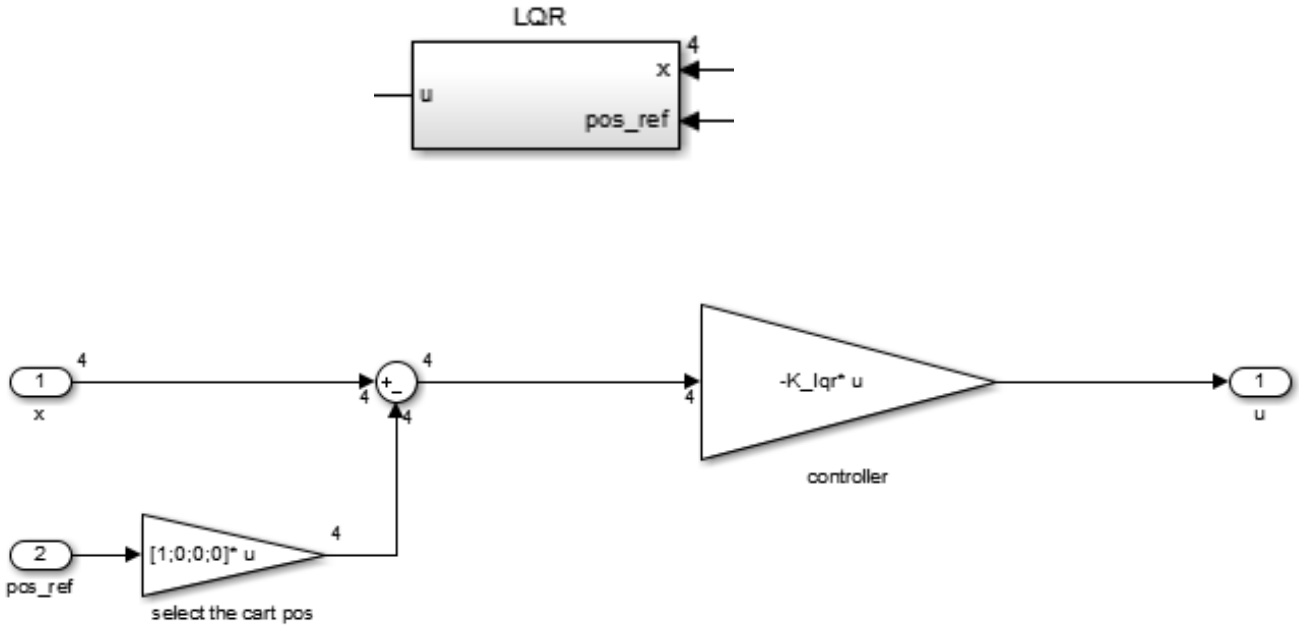
G) Résistance nominal

R = 5 Ohm

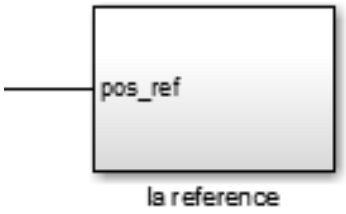


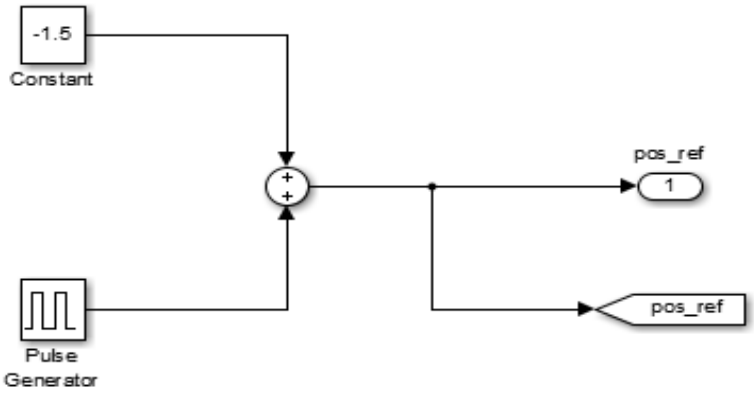
H) Les sous block de simulink :

1) le régulateur LQR :

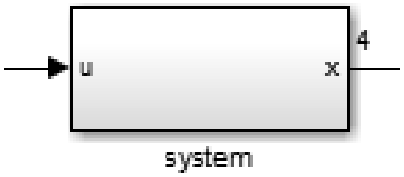


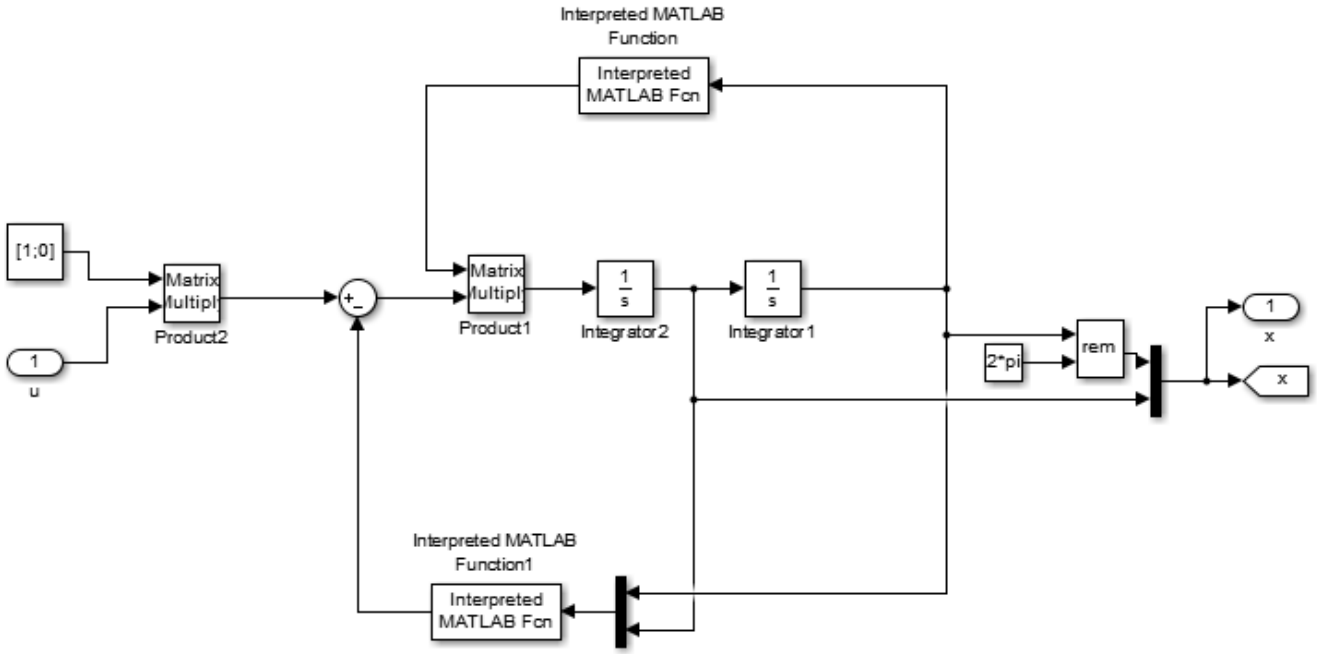
2) la référence



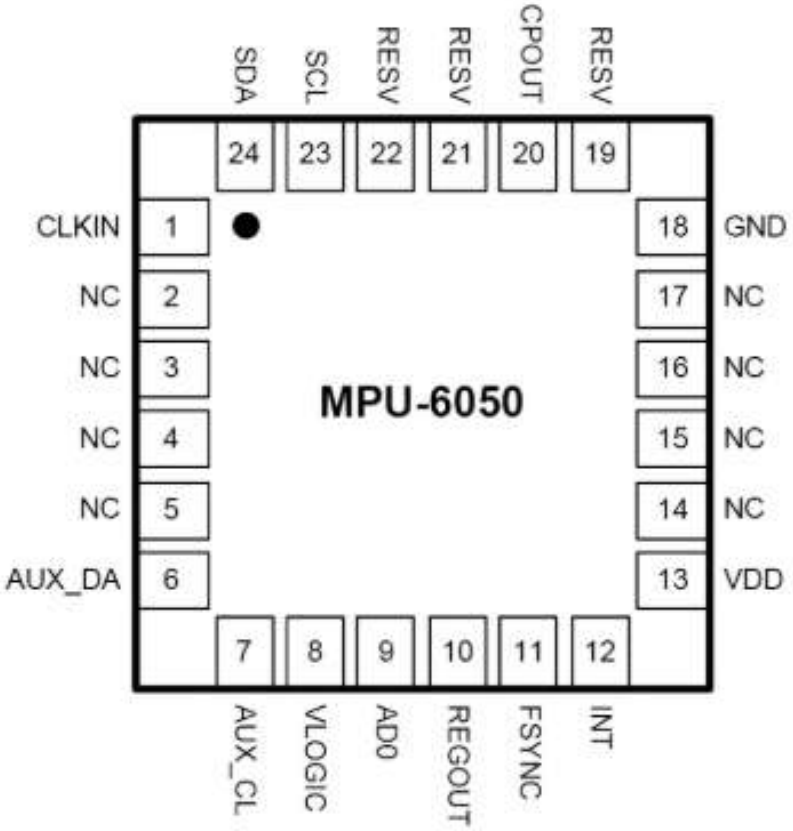


3) le model du robot





I) MPU6050 pins :



Pin Number	MPU-6050	Pin Name	Pin Description
1	Y	CLKIN	Optional external reference clock input. Connect to GND if unused.
6	Y	AUX_DA	I^2C master serial data, for connecting to external sensors
7	Y	AUX_CL	I^2C Master serial clock, for connecting to external sensors
8		/CS	SPI chip select (0=SPI mode)
8	Y	VLOGIC	Digital I/O supply voltage
9		AD0 / SDO	I^2C Slave Address LSB (AD0); SPI serial data output (SDO)
9	Y	AD0	I^2C Slave Address LSB (AD0)
10	Y	REGOUT	Regulator filter capacitor connection
11	Y	FSYNC	Frame synchronization digital input. Connect to GND if unused.
12	Y	INT	Interrupt digital output (totem pole or open-drain)
13	Y	VDD	Power supply voltage and Digital I/O supply voltage
18	Y	GND	Power supply ground
19, 21	Y	RESV	Reserved. Do not connect.
20	Y	CPOUT	Charge pump capacitor connection
22	Y	RESV	Reserved. Do not connect.
23		SCL / SCLK	I^2C serial clock (SCL); SPI serial clock (SCLK)
23	Y	SCL	I^2C serial clock (SCL)
24		SDA / SDI	I^2C serial data (SDA); SPI serial data input (SDI)
24	Y	SDA	I^2C serial data (SDA)
2, 3, 4, 5, 14, 15, 16, 17	Y	NC	Not internally connected. May be used for PCB trace routing.