****

Department of CSE

Course Code : CSE 436

Course Title : Computer Interface Lab

Report On : Self Balancing Robot with Encoder Motor

Submission Date : 23/01/19

Submitted by, Sumitted to,

Mohammad Tipu Sultan (15101036) Md. Abdullah Al Omar

Md.Eyaseen Arafat (15101038) Lecturer,Dept. of CSE

Md. Asif Khan (15101040) University of Asia Pacific

**Self Balancing Robot**

**Overview :**

The objective of this project was to design and implement a self- balancing algorithm using the Encoder Motor. The implementation utilized both an accelerometer and a rate-gyroscope built into the micro-controller in order to achieve a vertical balance. The fusion of both sensor data into a single usable value was achieved through a complementary filter. Consequently, the output of the complementary filter was designed to be primarily dependent on the gyroscope data, to which a fraction of the accelerometer data was added to compensate for the gyroscopic drift.

The control loop, which included both the software implementation of the complementary filter and the PID controller. Additionally, a pulse-width-modulation signal generator was implemented in software using the interrupt service routines of the micro-controller.



**Contents:**

Introduction------------------------------------------------------------------------01

Discussion-------------------------------------------------------------------------02-03

Diagrams and Figures----------------------------------------------------------04

Required Components---------------------------------------------------------05

Uses of Components-----------------------------------------------------------06-08

Use of this robot-----------------------------------------------------------------09

Future plans----------------------------------------------------------------------09

References-----------------------------------------------------------------------09

Introduction :

There are many projects on building self-balancing robots - ranging from simple DIY analog balancing bots to sophisticated self-balancing scooters. Self-balancing electro-mechanical systems have various uses, including the ability to showcase the computational performance of new and emerging embedded processors. This project is intended to be a showcase of the capabilities of the encoder motor.

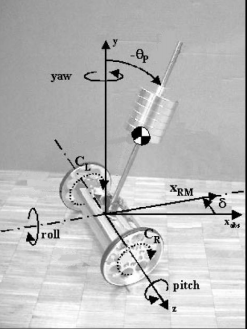
The prime objective of the project is to design code and implement an efficient self-balancing algorithm using encoder motor. The project would primarily focus on the software implementation of the algorithm by incorporating a physical model of the robot and the available sensors onboard the Encoder motor controller. The software itself is intended to be portable and can be used on any self-balancing platform based on encoder motor controller by tuning a few parameters.

Many relatively simple implementations of self-balancing robots by hobbyists use two IR reflectance sensors to determine the tilt angle of the robot. Almost all high-end self-balancing applications including the self-balancing scooter use either an accelerometer or a gyroscope or even a combination of both to maintain the robot vertical.

The performance characteristic of each filter is analyzed and weighed according to their individual pros and cons in the Discussion section of this report. This report discusses the theoretical considerations made at the start of the project, the steps that were taken to implement the self-balancing code, the analysis of the final results, the interpretation of the final results and finally the recommendations for future continuation of the project.

**Discussion**

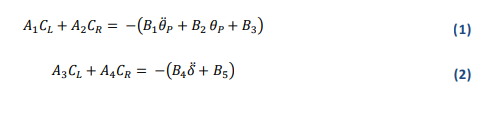
The theory, including the methods and considerations behind the physical model, the filtering algorithm, and the software design are all discussed in this section.



**Physical Model**

The robot has the ability to rotate around the z-axis (pitch) by an angle with a corresponding angular velocity . The linear movement of the robot is characterized by the position xRM and the velocity vRM. Furthermore, the robot can rotate about the vertical axis (yaw) by an angle δ with a corresponding angular velocity ̇ . The control system of the robot would be based on two state controls implemented through software: one controlling the stability around the lateral axis (pitch) and a second one about the vertical axis. Both state controls’ outputs would then be combined and the robot’s motion would be controlled by applying torques CL and CR to the corresponding wheels.

Equations R1 and R2 can be used to calculate the individual torques required for the robot’s motion.

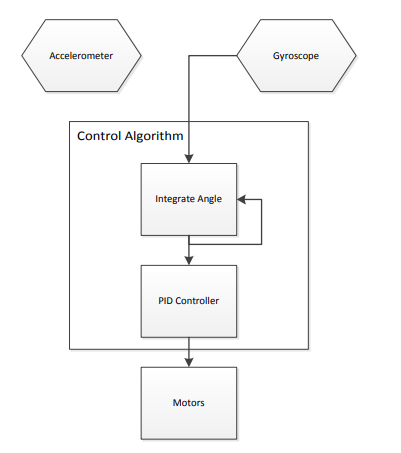


Where R1 models the robot when it is balanced at a stationary position and R2 models the robot when it is turning about its y-axis while maintaining a balanced posture (see Appendix A. for the derivation and the coefficient values).

**Control Algorithm Design**

Since calculating the pitch angle accurately is crucial for the self-balancing apparatus, a few methods were considered in order to measure the pitch angle and its rate of change. 2.2.1 Using an IR Tilt Sensor A method that is very popular with hobbyists is to use two infra-red sensors placed on either side of the robot. The control algorithm uses the sensors to measure the distance to the ground on either side of the robot. By comparing the values from the two sides, it is able to determine the pitch angle accurately. This method uses very little hardware and requires a very simple control algorithm.

**Illustration Figure:**

****

**Required Components:**

* Arduino Uno R3
* MPU6050
* 6V 210RPM Gear Motor Set
* L298N Motor Drive
* Switch Button
* Screw Nut Set
* Acrylic Sheet
* Li-ion Battery and Charger
* 9V Battery
* Jumper Wires
* Hot Glue Gun

**Uses of Components:**

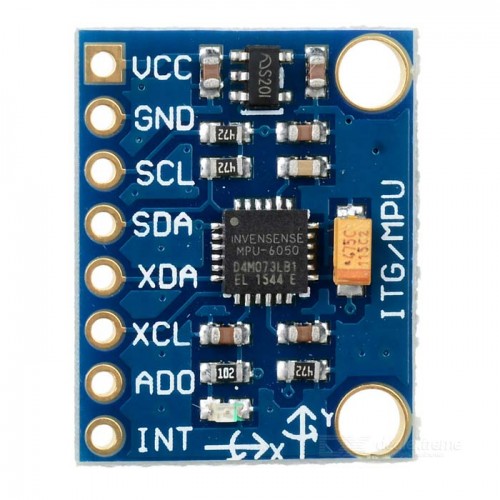
* **Arduino Uno:**

The  **Arduino Uno R3** is a microcontroller board based on a removable, dual-inline-package (DIP) ATmega328 AVR microcontroller. .

* It has 20 digital input/output pins.

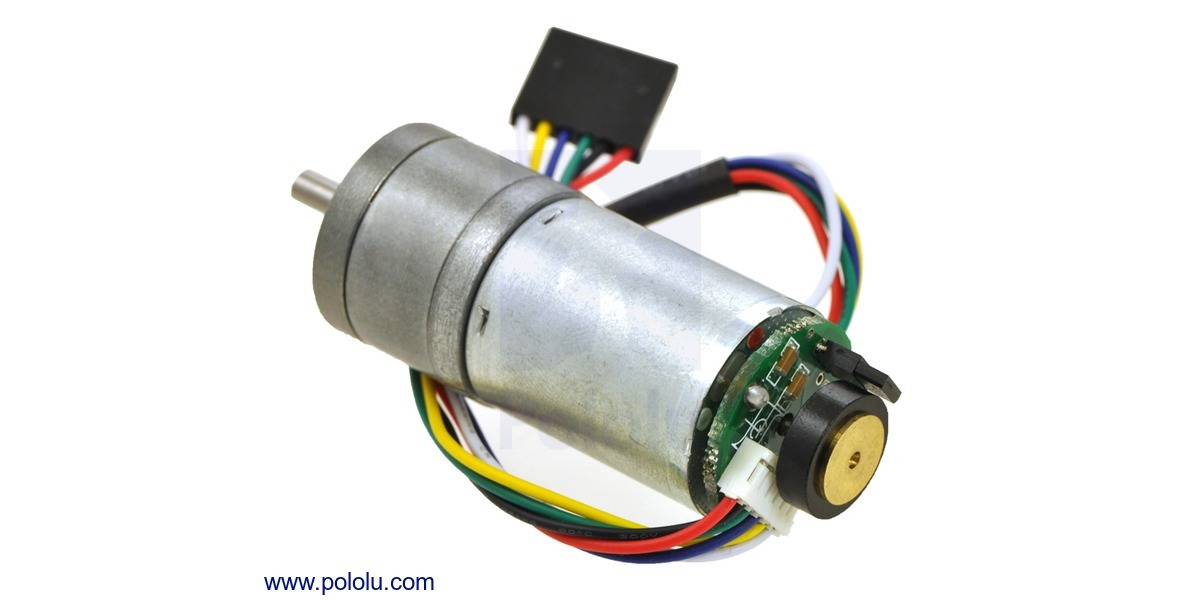
**MPU 6050:**

* The **MPU 6050** is a 6 DOF (degrees of freedom) or a six-axis IMU sensor, which means that it gives six values as output: three values from the accelerometer and three from the gyroscope.
* It is a sensor based on MEMS (micro electro mechanical systems) technology.



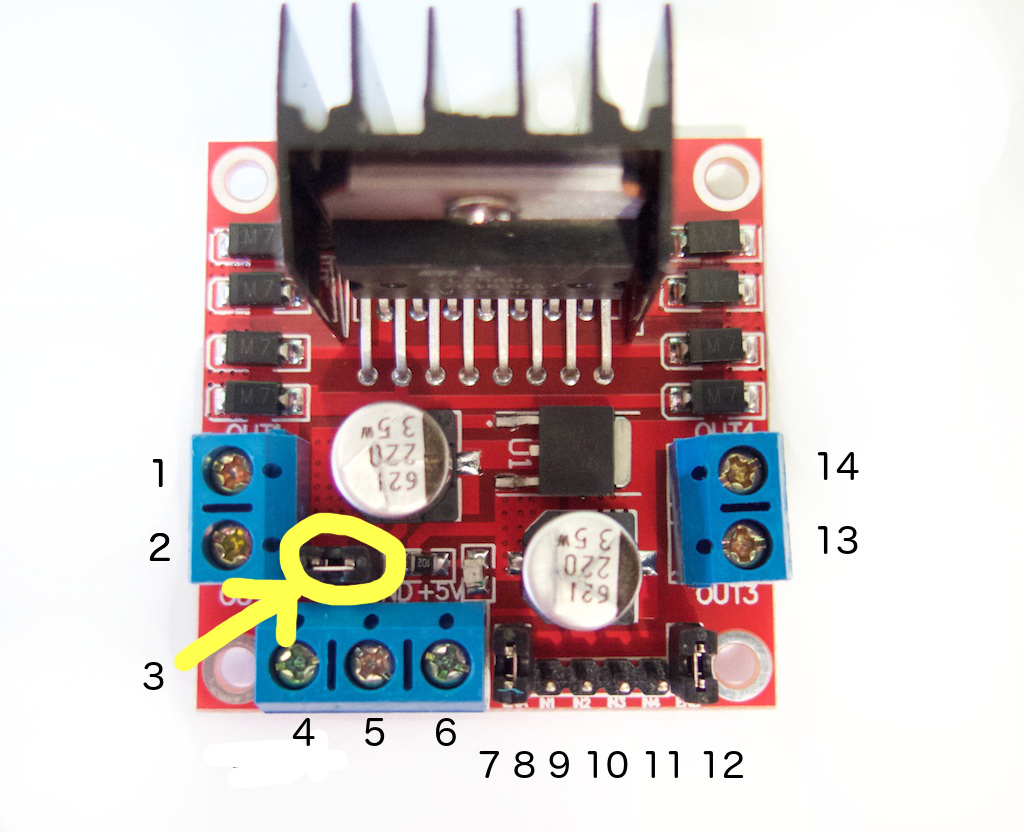
**RPM Encoder Gear Motor:**

* The versions of these gear motors with encoders use a A two-channel Hall effect sensor to detect the rotation of a magnetic disk on a rear protrusion of the motor shaft.
* The quadrature encoder provides a resolution of 48 counts per revolution of the motor shaft when counting both edges of both channels.



**L298N Motor Driver:**

* The L298N Motor Driver Module is a high voltage Dual H-Bridge
* H-bridge drivers are used to drive inductive loads that requires forward and reverse function with speed control such as DC Motors, and Stepper Motors.

****

**Use of this Robot:**

* Can be an assistant at any super shop.
* Can be a helping hand at any official environment.
* It can serve foods and goods.
* A modified version of it can be used as unicycle or self balancing vehicle.

**Future Plan:**

* Our plan is to make this robot more alert by adding sonar sensor and bluetooth.
* It will be able to response and come to the user if he/she calls it by it’s name.

**References:**

* [www.google.com](http://www.google.com)
* <https://www.youtube.com/watch?v=9W5S5nqRegU>
* <https://drive.google.com/file/d/1ydK8d2b6wmJMom_8qVyv7DWwxi9v6-Is/view>
* [www.techshopbd.com](http://www.techshopbd.com)
* <https://store.roboticsbd.com>