**Introduction**

To be able to measure and compensate for imbalance is essential for stabilizing

mechanisms. The technique is applied in everything from self-stabilizing cameras to

helicopters and noise reducing equipment.

This report describes the development of a self-stabilizing platform, and includes theory

about sensors, filters and motor modelling, and also practical tests. The purpose is to

answer how the system will behave when a load is placed asymmetrically on the

platform and if it is possible to compensate for the imbalance that occurs.

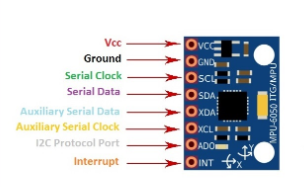
The tilt of the platform is measured by an IMU, a sensor combining accelerometers and

gyroscopes. A Kalman filter is used to combine the data. From this a signal, with noise

levels within the requirements, was obtained.

**MPU6050:**

The **MPU6050** is a Micro Electro-Mechanical Systems (MEMS) which consists of a 3-axis Accelerometer and 3-axis Gyroscope inside it. This helps us to measure acceleration, velocity, orientation, displacement and many other motion related parameter of a system or object.



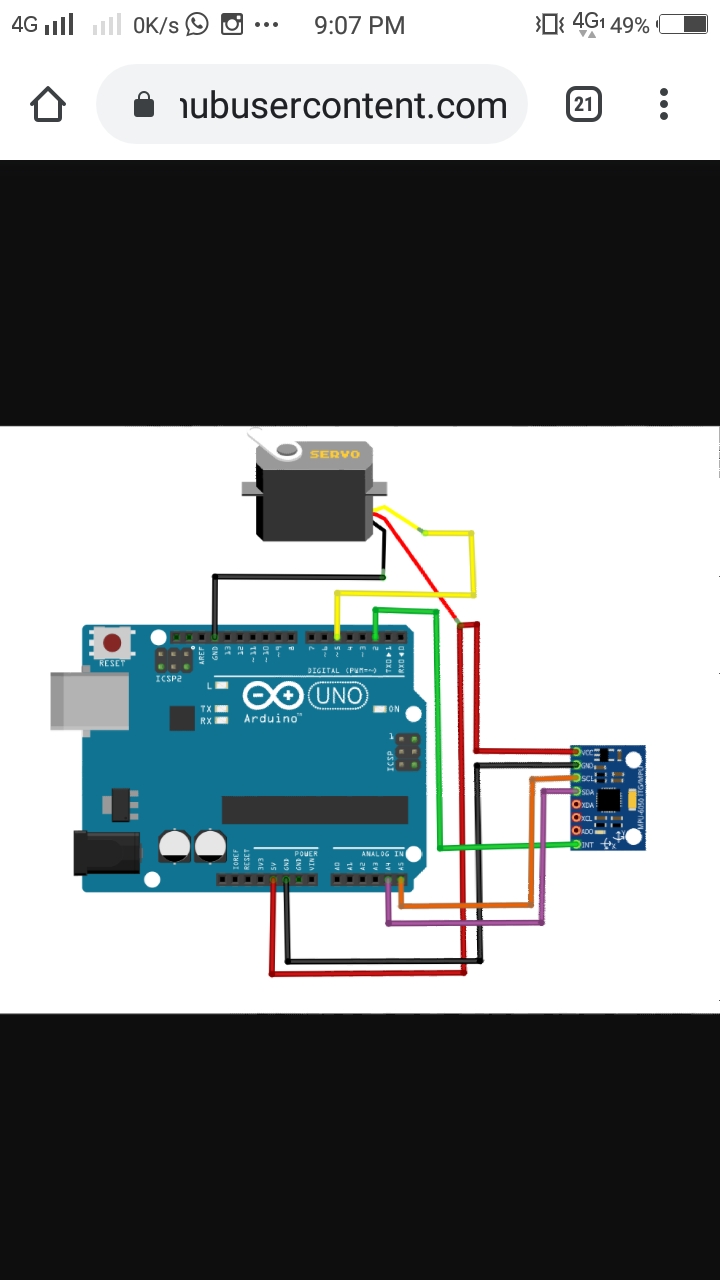
**Problem Statement**

#### Self-stabilizing platform finds its use in various applications from as simple as camera stabilization to surgery platforms. It can also be used in ships or vehicles to develop self-leveling anti motion sickness seats, or making Earthquake proof buildings this same mechanism is also used inself balancing scooter also known as hoverboard . The applications can be unlimited and depends upon your imagination. Over here we will show you how to make your own one axis Self-stabilizing platform. You can modify the project and you two motors to make a two axis Self-stabilizing platform

**Components Required**

1. [Arduino UNO](http://www.dnatechindia.com/arduino_uno_r3.html)
2. [MPU6050 MEMS accelerometer and a MEMS gyro](http://www.dnatechindia.com/gy521-three-axis-gyro-accelerometer-mpu-6050-module.html)
3. [MG-90 Servo Motor](http://www.dnatechindia.com/mg-90-s-metal-gear-micro-servo-motor.html)
4. [Various Jumper wirezs](http://www.dnatechindia.com/index.php?route=product/search&search=jumper)

**Schematic Diagram**



**Code**

#include <Wire.h> //header file

#include <MPU6050.h> //header file for MPU6050

#include <Servo.h> //header file for servo motor

Servo sg90;

int servo\_pin = 2; //initializing the values

MPU6050 sensor ;

int16\_t ax, ay, az ;

int16\_t gx, gy, gz ;

void setup ( )

{

sg90.attach ( servo\_pin );

Wire.begin ( );

Serial.begin (9600);

Serial.println ( "Initializing the sensor" );

sensor.initialize ( ); //initialize the sensor

Serial.println (sensor.testConnection ( ) ? "Successfully Connected" : "Connection failed");

delay (1000);

Serial.println ( "Taking Values from the sensor" );

delay (1000);

}

void loop ( )

{

sensor.getMotion6 (&ax, &ay, &az, &gx, &gy, &gz); //MPU is 6 axis IMU sensor get

accelerometer and gyroscope

ax = map (ax, -17000, 17000, 0, 180) ; // map is used to value from Low would get mapped

to low and High value Mapped to High and in between to be mapped in between

Serial.println (ax); //printing the values

sg90.write (ax); //the given mapped values are assigned to the serve motor and it rotates

delay (20);

}



**PLAGIARISM SCAN REPORT**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | **Date** | 2020-02-07 |  |
|  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  | **Words** | 94 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| **0%** | **100%** |  | **Characters** | 631 |  |
|  |  |  |  |
| **Plagiarised** | **Unique** |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |



**Content Checked For Plagiarism**

#include <Wire.h> #include <MPU6050.h> #include <Servo.h> Servo servo1; int servopin = 2; MPU6050 sense; int a\_x, a\_y, a\_z ; int g\_x,

g\_y, g\_z ; void setup ( ) { servo1.attach ( servopin ); Wire.begin ( ); Serial.begin (9600); Serial.println ( "Initialize the sensor" ); sensor.initialize ( ); Serial.println (sensor.testConnection ( ) ? "Successfully\_Connected" : "Connection\_failed"); delay (1000); Serial.println ( "Taking Values from sensor" ); delay (1000); } void loop ( ) { sensor.getMotion(&a\_x, &a\_y, &a\_z, &g\_x, &g\_y, &g\_z); a\_x = map (a\_x, -17000, 17000, 0, 180) ; Serial.println (a\_x); servo1.write (a\_x); delay (20); }

**Matched Source**

No plagiarism found



**Challenges Faced**

My project title is design a self-stabilizing using Arduino, so I had made a self-stabilizing platform using one servo motor and accelerometer MPU6050.

Here we use only one servo motor it handles less weight

**References**

1. Arduino, 2015. *Arduino – Rotary Encoders.* Available at:

http://playground.arduino.cc/Main/RotaryEncoders, [Accessed: 2015-05-03]

1. Arduino, 2015. *Datasheet – Keyes L298 Motor driver*. Available at:

https://arduino-info.wikispaces.com/file/view/L298-2.pdf/281104616/L298-2.pdf.

[Accessed: 2015-04-12]

1. Arduino Uno, 2015. *Arduino – ArduinoBoardUno.* Available at:

http://arduino.cc/en/main/arduinoBoardUno, [Accessed: 2015-02-19]

1. Atmel, 2015. *Data sheet – Atmgea32*. Available at:

http://www.atmel.com/devices/atmega32.aspx. [Accessed: 2015-03-03]

Because We Can! labs, 2014. *CoffeeMate - Arduino controlled self-stabilizing platform,*

1. *gyro / accellerometer as sensors.* Available at:

https://www.youtube.com/watch?v=EbMFuhaoxpo. [Accessed: 2015-02-17]

Danapalasingam K. A., Leth J., la Cour-Harbo A., and Bisgaard M., 2010. *Robust Helicopter*

1. *Stabilization in the Face of Wind Disturbance.* Available at:

http://ieeexplore.ieee.org/xpls/abs\_all.jsp?arnumber=5717418. [Accessed: 2015-05-

05]

1. Dimension Engineering, 2015. *A beginner’s guide to accelerometers*. Available at:

http://www.dimensionengineering.com/info/accelerometers. [Accessed: 2015-03-20]