Contents

Introduction:

Measure Tilt Angle using MPU6050 & STM32F103C8 Microcontroller. This can be done by simply interfacing MPU6050 6 axis Gyro/Accelerometer Sensor with STM32 .
Accelerometer sends X, Y, and Z acceleration forces. We need to convert the forces into X, Y, Z 3D angle to determine the 3D Orientation of the sensor.

DEGREES OF FREEDOM

In physics, the degrees of freedom (DOF) of a mechanical system is the number of independent parameters that define its configuration or state. It is important in the analysis of systems of bodies in mechanical engineering, structural engineering, aerospace engineering, robotics, and other fields.
The position of a single railcar (engine) moving along a track has one degree of freedom because the position of the car is defined by the distance along the track. A train of rigid cars connected by hinges to an engine still has only one degree of freedom because the positions of the cars behind the engine are constrained by the shape of the track.
An automobile with highly stiff suspension can be considered to be a rigid body traveling on a plane (a flat, two-dimensional space). This body has three independent degrees of freedom consisting of two components of translation and one angle of rotation. Skidding or dri ing is a good example of an automobile's three independent degrees of freedom.
The position and orientation of a rigid body in space is defined by three components of translation and three components of rotation, which means that it has six degrees of freedom.
The exact constraint mechanical design method manages the degrees of freedom to neither underconstrain nor overconstrain a device.
Six degrees of freedom (6DOF) refers to the six mechanical degrees of freedom of movement of a rigid body in three-dimensional space. Specifically, the body is free to change position as forward/backward (surge), up/down (heave), le/right (sway) translation in three perpendicular axes, combined with changes in orientation through rotation about three perpendicular axes, o en termed yaw (normal axis), pitch (transverse axis), and roll (longitudinal axis).
Three degrees of freedom (3DOF), a term o en used in the context of virtual reality, typically refers to tracking of rotational motion only: pitch, yaw, and roll.
MEMS
MEMS, or Micro-Electro-Mechanical Systems, refer to devices that combine electronic and mechanical components on a microscopic scale. These systems are made possible through the use of microfabrication techniques, which allow for the creation of tiny structures with high precision and accuracy.
MEMS devices have a wide range of applications, including sensors, actuators, and microfluidics.

For example, MEMS accelerometers are used in smartphones to detect motion and orientation, while MEMS pressure sensors are used in medical devices to monitor blood pressure.

One of the key advantages of MEMS devices is their small size and low power consumption, which makes them ideal for use in portable and wearable devices. They are also highly reliable and can be produced in large quantities at low cost.

MEMS technology has been around for several decades and has evolved rapidly over the years. MEMS devices are typically made using silicon, which is a highly versatile material that can be used to create a wide range of structures and components. The manufacturing process involves etching tiny structures onto a silicon wafer using photolithography, which is similar to the process used to manufacture computer chips.

MEMS devices are typically powered by electricity and use various sensing mechanisms, such as capacitive sensing, piezoelectric sensing, and magnetic sensing, to detect changes in the environment. The devices can also be designed to respond to changes in the environment by actuating or controlling various mechanical components, such as microvalves, microactuators, and microfluidic pumps.

MEMS devices have a wide range of applications across various industries, including healthcare, consumer electronics, automotive, and aerospace. In healthcare, MEMS devices are used for drug delivery, blood glucose monitoring, and implantable medical devices. In consumer electronics, MEMS devices are used in smartphones, smartwatches, and fitness trackers. In automotive, MEMS devices are used for airbag deployment, tire pressure monitoring, and engine control. In aerospace, MEMS devices are used for navigation, satellite control, and environmental monitoring.

The MPU-6050 incorporates InvenSense's MotionFusion™ and run-time calibration firmware that enables manufacturers to eliminate the costly and complex selection, qualification, and system level integration of discrete devices in motion-enabled products, guaranteeing that sensor fusion algorithms and calibration procedures deliver optimal performance for consumers.

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. This module also has a (DMP) Digital Motion Processor inside it which is powerful enough to perform complex calculation and thus free up the work for Microcontroller.

The module also have two auxiliary pins which can be used to interface external IIC modules like an magnetometer, however it is optional. Since the IIC address of the module is configurable more than one MPU6050 sensor can be interfaced to a Microcontroller using the AD0 pin. This module also has well documented and revised libraries available hence it's very easy to use with famous platforms like Arduino. So if you are looking for a sensor to control motion for your RC Car, Drone, Self balancing Robot, Humanoid, Biped or something like that then this sensor might be the right choice for you.

Finally the introduction provides an overview of the project titled "Measuring Tilt Angle using MPU6050 & STM32F103C8 Microcontroller." It explains that the project aims to use the MPU6050 sensor and STM32F103C8 microcontroller to accurately measure the tilt angle of an object. The MPU6050 is a 6-DOF IMU sensor with a three-axis accelerometer and gyroscope, while the STM32F103C8 is a powerful microcontroller. The project's objectives include interfacing the sensor and microcontroller, calibrating the sensor, implementing algorithms for tilt angle calculation, developing a user-friendly interface, and adding additional features. The implementation steps involve hardware setup, sensor initialization, data acquisition, calibration, tilt angle calculation, data display or transmission, and testing/refinement. The project's conclusion highlights its cost-effectiveness and applicability in robotics, navigation systems, gaming, and other fields.

Literature survey-

This paper discusses the features of MEMS-gyroscope, which is part of the measuring system MPU6050. It also discusses the receiving of the data from its sensors. This system has a digital interface I2C. Work with MPU6050 demands the use of STM32VLDISCOVERY evaluating board with microcontroller STM32F100RBT6B stated on it. System MPU6050 is connected to the I2C-module of the microcontroller. The sensor is a slave and microcontroller is master [1].

The MPU-60X0 is the world's first integrated 6-axis MotionTracking device that combines a 3axis gyroscope, 3-axis accelerometer, and a Digital Motion Processor™ (DMP) all in a small 4x4x0.9m package. With its dedicated I2C sensor bus, it directly accepts inputs from an external 3-axis compass to provide a complete 9-axis MotionFusion™ output. The MPU-60X0 MotionTracking device, with its 6-axis integration, on-board MotionFusion™, and run-time calibration firmware, enables manufacturers to eliminate the costly and complex selection, qualification, and system level integration of discrete devices, guaranteeing optimal motion performance for consumers.The MPU-60X0 is also designed to interface with multiple noninertial digital sensors, such as pressure sensors, on its auxiliary I2C port [2].

A documentation published by STMicroelectronics .N.V a Dutch multinational corporation, which describes all the required technical specifications of the STM32 family of microcontrollers, specifically STM32 F1. The high cost of data collection, and the necessity to adapt to very specific environments (e.g., personalized health care) is prompting to introduce mechanisms to learn new tasks during run-time, using collected data as training input .These applications are enabled by a train-then-deploy paradigm: edge devices are exclusively responsible for inference, whereas training is delegated to high-end, power-hungry GPUs in data centers [3].

This paper describes highly-optimized AES-{128,192,256}-CTR assembly implementations for the popular ARM Cortex-M3 and M4 embedded microprocessors. These implementations are about twice as fast as existing implementations. Additionally, we provide the fastest bitsliced constant-time and masked implementations of AES-128-CTR to protect against timing attacks, power analysis and other (first-order) side-channel attacks.[4]

The key component used in the tutorials is the STM32 VL discovery board produced by STMicroelectronics (ST) and available from many electron- ics distributors for approximately $10. 4 This board,includes a user configurable STM32 F100 micro-controller with 128 KB flash and 8 KB ram as well as an integrated hardware debugger interface based upon a dedicated USB connectedSTM32F103[5].

Proposed solution: ARM CORTEX

_

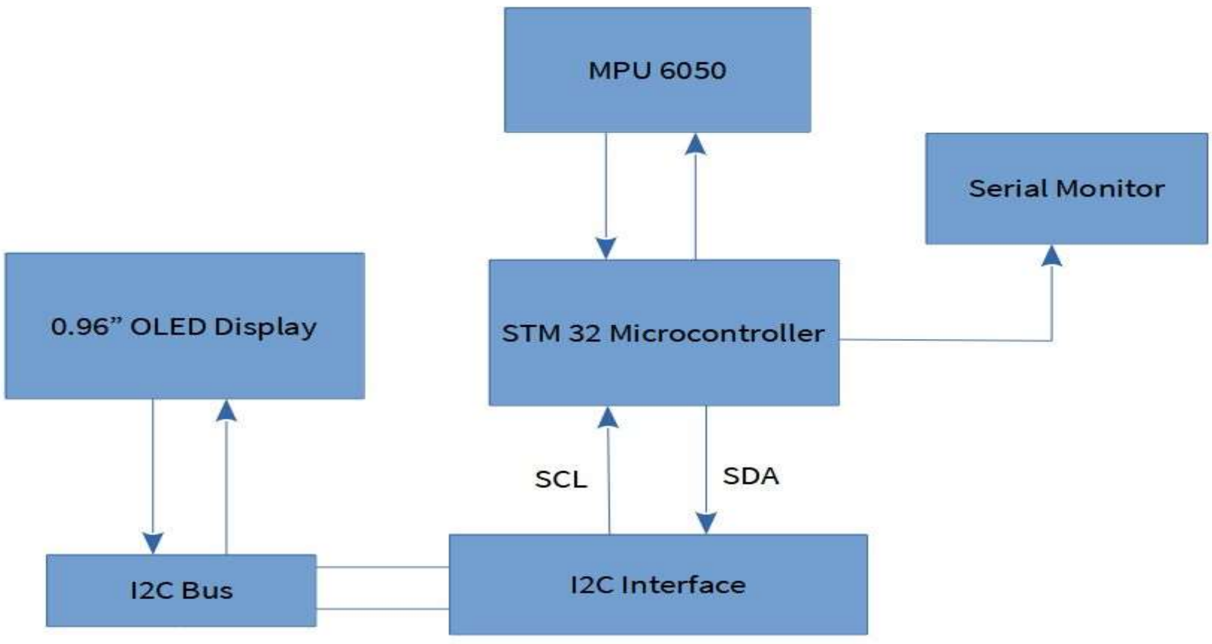

Fig 1 Block Diagram

The ARM Cortex is a family of 32-bit RISC (Reduced Instruction Set Computing) microprocessors designed and licensed by ARM Holdings. The Cortex processors are widely used in embedded systems and have become a popular choice for a wide range of applications due to their high performance, low power consumption, and flexible architecture.

The Cortex processors are divided into three main categories: the Cortex-A series, Cortex-R series, and Cortex-M series. The Cortex-A series is designed for high-performance applications, such as smartphones and tablets, while the Cortex-R series is designed for realtime applications, such as automotive and industrial control systems. The Cortex-M series is designed for low-power applications, such as microcontrollers for IoT devices and wearables.

The Cortex processors are highly configurable, allowing developers to customize the processor to meet the specific needs of their application. The processors feature a range of advanced features, including advanced SIMD (Single Instruction, Multiple Data) instructions, hardware virtualization support, and advanced debug and trace capabilities.

In addition to the Cortex processors, ARM Holdings also provides a range of development tools and so ware, including the Keil MDK-ARM development environment and the ARM CMSIS (Cortex Microcontroller So ware Interface Standard) software library. These tools and so ware make it easier for developers to design and develop so ware for ARM-based systems.

Overall, the ARM Cortex is a powerful and flexible family of microprocessors that has become a popular choice for a wide range of applications. Its advanced features and configurability make it a great platform for building complex embedded systems that require high performance, low power consumption, and advanced functionality.

STM32 MCU

The STM32 is a family of 32-bit microcontrollers designed and manufactured by STMicroelectronics. The STM32 microcontrollers are widely used in a wide range of applications, including automotive, industrial control, robotics, and consumer electronics, due to their high performance, low power consumption, and rich set of features. (Refer Fig)

The STM32 microcontrollers are based on the ARM Cortex-M processor architecture, which provides a powerful and flexible platform for embedded systems development. The CortexM processor architecture is designed for low power and high performance, and it features a rich set of peripherals, including USB, CAN, Ethernet, and LCD interfaces.

The STM32 microcontrollers are available in a wide range of packages, from small 20-pin packages to large 144-pin packages, allowing developers to choose the right microcontroller for their specific

application. The microcontrollers also feature a range of advanced features, including advanced power management, hardware encryption, and advanced analog and digital signal processing. STMicroelectronics also provides a range of development tools and so ware to support the STM32 microcontrollers, including the STM32CubeMX code generator and the STM32CubeIDE development environment. These tools make it easier for developers to design, develop, and debug so ware for STM32-based systems.
Overall, the STM32 microcontrollers are a powerful and flexible platform for embedded systems development, offering high performance, low power consumption, and a rich set of features. The microcontrollers are widely used in a range of industries and applications, and their popularity continues to grow as new features and capabilities are added to the platform

MPU6050 :

The MPU-6050 incorporates InvenSense’s MotionFusion™ and run-time calibration firmware that enables manufacturers to eliminate the costly and complex selection, qualification, and system level integration of discrete devices in motion-enabled products, guaranteeing that sensor fusion algorithms and calibration procedures deliver optimal performance for consumers. (Refer Fig 2)
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. This module also has a (DMP) Digital Motion Processor inside it which is powerful enough to perform complex calculation and thus free up the work for Microcontroller.

Fig 2 MPU 6050

OLED display:

The 0.96" OLED display is a compact and versatile display module based on OLED technology. With a screen size of 0.96 inches, it offers high contrast, vibrant colors, and excellent visibility even in low-light conditions.(Refer Fig 3) The display typically has resolutions of 128x64 or 128x32 pixels and utilizes interfaces such as I2C or SPI for communication. It features low power consumption,

Fig 3 OLED

fast response times, and is lightweight and thin. The 0.96" LED display finds applications in wearable devices, IoT, embedded systems, and prototyping projects.

Working :

The "Measure Tilt Angle using MPU6050 & STM32F103C8 Microcontroller" project works by connecting the MPU6050 sensor to the STM32F103C8 microcontroller. The MPU6050 sensor is initialized by configuring its registers and settings to establish communication with the microcontroller. The raw acceleration and gyroscope values are continuously acquired from the sensor by reading the appropriate registers through the microcontroller. (Refer Fig 1 for block representation)

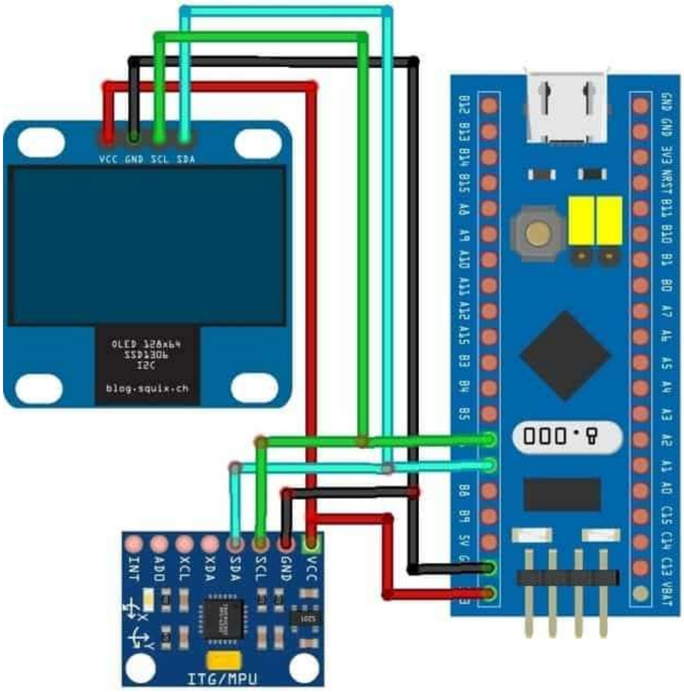

Fig 4 Circuit Schematic

The tilt angle calculation is performed using the accelerometer data, which provides information about the object's orientation relative to the gravitational force. Mathematical calculations, such as trigonometry or inverse tangent functions, are used to convert the sensor readings into tilt angles. To improve accuracy, complementary filtering or Kalman filtering algorithms may be implemented to combine the accelerometer and gyroscope data.

The calculated tilt angle is then displayed on a 0.96" LED display module. The microcontroller interfaces with the display using appropriate libraries and functions to present the tilt angle in a user-friendly manner.

Overall, this project utilizes the MPU6050 sensor and STM32F103C8 microcontroller to acquire sensor data, perform calculations, and display the tilt angle on a compact LED display. It provides an efficient and cost-effective solution for measuring tilt angles in various applications.

Code:

```
#include <MPU6050_tockn.h>

#include <Wire.h>

#include<LiquidCrystal_I2C.h>

#include <SoftwareSerial.h>


int led_left2 = 11;

int led_left1 = 10;

int led_mid0 = 9;

int led_right1 = 8;

int led_right2 = 7;

SoftwareSerial espSerial(5, 6);

String str;


MPU6050 mpu6050(Wire);

long timer = 0;

void setup() {

  Serial.begin(115200);

  espSerial.begin(115200);

  Wire.begin();

  mpu6050.begin();

  mpu6050.calcGyroOffsets(true);

  pinMode(led_left2, OUTPUT);

  pinMode(led_left1, OUTPUT);

  pinMode(led_mid0, OUTPUT);

  pinMode(led_right1, OUTPUT);

  pinMode(led_right2, OUTPUT);
```

```
}

void loop() {
  mpu6050.update();

  if(millis() - timer > 1000){
    Serial.print("Temperature in Celsius ");Serial.println(mpu6050.getTemp());
    Serial.println("Acceleration in X-axis : ");Serial.print(mpu6050.getAccX());
    Serial.print("m/s^2 \n");
    Serial.println("Acceleration in Y-axis : ");Serial.print(mpu6050.getAccY());
    Serial.print("m/s^2\n");
    Serial.println("Acceleration in Z-axis : ");Serial.println(mpu6050.getAccZ());
    Serial.print("m/s^2\n");

    Serial.println("Rotational velocity about X-axis : ");Serial.print(mpu6050.getGyroX());
    Serial.print("rad/sec \n");
    Serial.println("Rotational velocity about Y-axis : ");Serial.print(mpu6050.getGyroY());
    Serial.print("rad/sec \n");
    Serial.println("Rotational velocity about Z-axis : ");Serial.print(mpu6050.getGyroZ());
    Serial.print("rad/sec \n");

    Serial.print("Angular acceleration about X axis: ");Serial.print(mpu6050.getAccAngleX());
    Serial.print("\n");
    Serial.print("Angular acceleration about Y axis: ");Serial.print(mpu6050.getAccAngleY());
    Serial.print("\n");

    Serial.print("Angular Rotation about X axis : ");Serial.print(mpu6050.getGyroAngleX());
```

```
  Serial.print("\n");
  Serial.print("Angular Rotation about Y axis : ");Serial.print(mpu6050.getGyroAngleY());
  Serial.print("\n");
  Serial.print("Angular Rotation about Z axis : ");Serial.print(mpu6050.getGyroAngleZ());
  Serial.print("\n");

  Serial.println("Pitch in degrees : ");Serial.print(mpu6050.getAngleX());
  Serial.print("\n");
  Serial.println("Roll in degrees : ");Serial.print(mpu6050.getAngleY());
  Serial.print("\n");
  Serial.println("Yaw in degrees : ");Serial.print(mpu6050.getAngleZ());
  Serial.print("\n");
  delay(1000);
   }
if(mpu6050.getAngleY()<-45)
    {
     // turn on level led
     digitalWrite(led_left2,HIGH);
     delay(250);
     digitalWrite(led_left2,LOW);
     delay(250);
     digitalWrite(led_left1,LOW);
     digitalWrite(led_mid0,LOW);
     digitalWrite(led_right1,LOW);
     digitalWrite(led_right2,LOW);
    }
    else if((mpu6050.getAngleY()>-45.00)&&(mpu6050.getAngleY()<-30.00))
```

```
{
 // turn on level led
 digitalWrite(led_left2,LOW);
 digitalWrite(led_left1,HIGH);
 delay(500);
 digitalWrite(led_left1,LOW);
 delay(500);
 digitalWrite(led_mid0,LOW);
 digitalWrite(led_right1,LOW);
 digitalWrite(led_right2,LOW);
 }
else if ((mpu6050.getAngleY()>-29.99)&&(mpu6050.getAngleY()<30.00))
{
 // turn on level led
 digitalWrite(led_left2,LOW);
 digitalWrite(led_left1,LOW );
 digitalWrite(led_mid0,HIGH);
 digitalWrite(led_right1,LOW);
 digitalWrite(led_right2,LOW);
}
else if((mpu6050.getAngleY()>30.01)&&(mpu6050.getAngleY()<45.00))
{
 // turn on level led
 digitalWrite(led_left2,LOW );
 digitalWrite(led_left1,LOW );
 digitalWrite(led_mid0,LOW );
 digitalWrite(led_right1,HIGH );
```

```
     delay(250);
     digitalWrite(led_right1,LOW );
     delay(250);
     digitalWrite(led_right2,LOW );
    }
    else if(mpu6050.getAngleY()>45.00)
    // turn on level led
    {
     digitalWrite(led_left2,LOW );
    digitalWrite(led_left1,LOW );
    digitalWrite(led_mid0,LOW );
    digitalWrite(led_right1,LOW );
    digitalWrite(led_right2,HIGH );
    delay(500);
    digitalWrite(led_right2,LOW );
    delay(500);
    }
if (espSerial.available()>0)
  {
   //int data = mpu6050.getAngleX();
   //Serial.println(data);
   espSerial.write(mpu6050.getAngleX());
  delay(70);
   espSerial.write(int(mpu6050.getAngleY()));
   delay(70);
  espSerial.write(int(mpu6050.getAngleZ()));
   delay(70);
```

```
  }
}
```

Result:

- ✓ The project enables accurate and real-time measurement of the tilt angle of an object..
- ✓ The calculated tilt angle can be displayed on a 0.96" LED display module. Users can easily visualize the tilt angle readings in a user-friendly format.
- ✓ The project incorporates calibration techniques to eliminate bias and offset errors in the sensor readings.
- ✓ With the use of the 0.96" LED display and the compact STM32F103C8 microcontroller, the project offers a small and portable solution for tilt angle measurement.
- ✓ The project serves as a foundation for further customization and development.

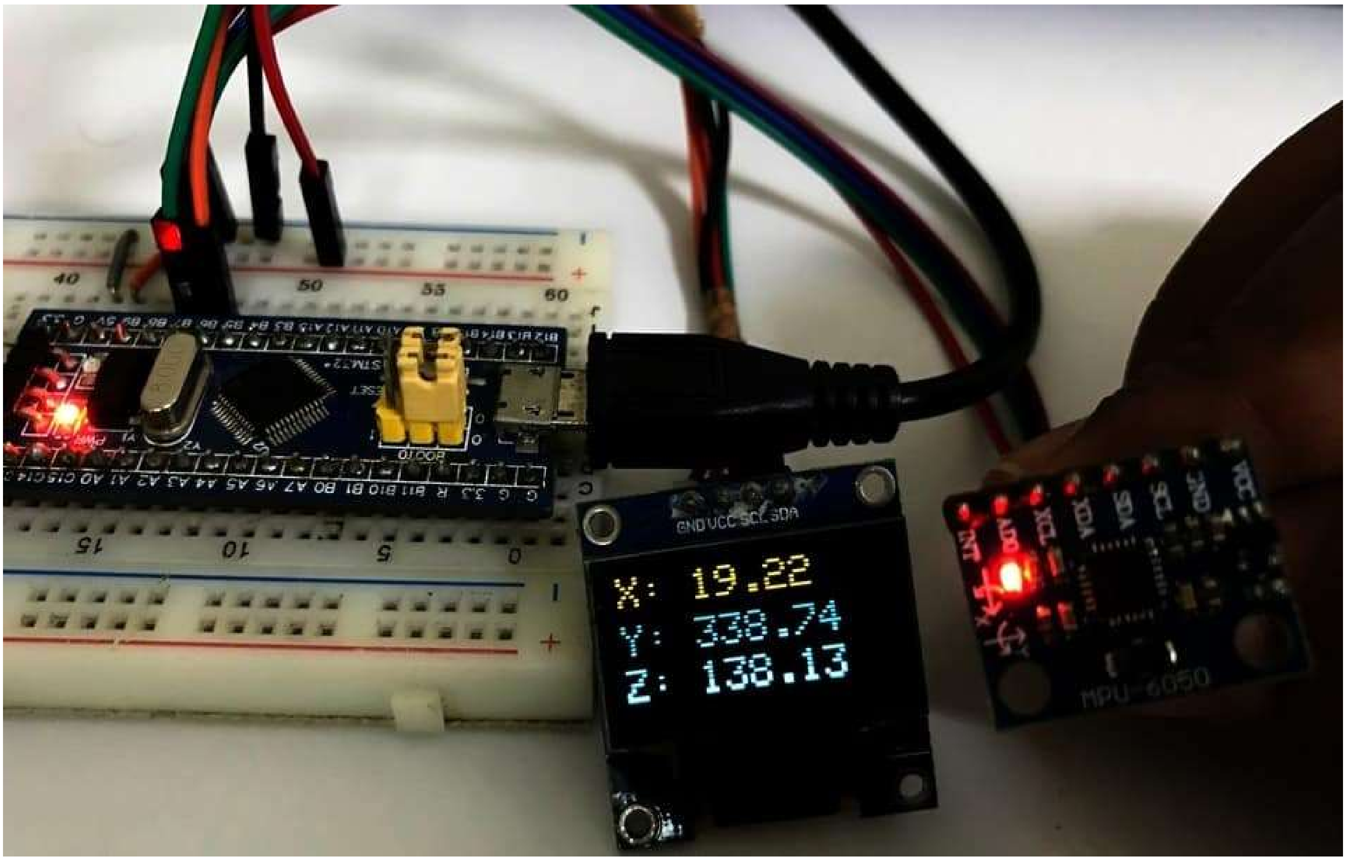

Fig 6 Results

Conclusion:

In conclusion, the MPU6050 is a versatile and popular MEMS sensor that combines a 3-axis gyroscope and a 3-axis accelerometer in a single package. When used with an STM32 microcontroller, the MPU6050 can provide accurate and reliable measurements of orientation and motion, making it an ideal choice for a wide range of applications, including robotics, gaming, and navigation.

By leveraging the power and flexibility of the STM32 microcontroller, developers can easily interface with the MPU6050 and integrate its data into their applications. The STM32's rich set of features, including its high processing power, low power consumption, and advanced peripherals, make it a great platform for building complex systems that require precise sensing and control.

Overall, the combination of the MPU6050 and STM32 offers a powerful and flexible solution for a wide range of applications, and their use together has become increasingly popular in the field of embedded systems. As technology continues to advance, we can expect to see even more innovative applications of this powerful combination.

References :

1) A paper published by D.S. Fedorov, A.Y. Ivoylov, V.A. Zhmud, V.G. Trubinz and from Novosibirsk State Technical University, Novosibirsk, Russia called “Using of Measuring System MPU6050 for the Determination of the Angular Velocities and Linear Accelerations”[1].

2) A Documentation published by TDK Inc. About the MPU 60X0 microelectromechanical system which describes all the technical details of the system [2].

3) A documentation published by STMicroelectronics .N.V a Dutch multinational corporation, which describes all the required technical specifications of the STM32 family of microcontrollers, specifically STM32 F1 [3]

4) The documentation published by Arm Holdings about the technical details of Arm® Cortex®-M3. This documentation also describes the differences between differences Arm Cortex-M series architectures [4].

5) Discovering the STM32 Microcontroller ,Geoffrey Brown ,©2012, December 10, 2012 [5].