A

PROJECT REPORT

ON TILT ANGLE MEASUREMENT

USING MPU6050

Submitted by:

Lohitha Bommisetty

TITLE : Tilt angle measurement using MPU6050

INTRODUCTION:

The ability to measure tilt angles is crucial in various applications, including robotics, gaming, virtual reality, and motion tracking systems. This project focuses on developing a tilt angle measurement system using the MPU6050 sensor, an Arduino microcontroller, and an OLED display.

PURPOSE:

The primary purpose of the Tilt Angle Measurement Project is to create a reliable and user-friendly system for measuring and displaying tilt angles using the MPU6050 sensor, Arduino microcontroller, and OLED display.

OBJECTIVES:

- To interface the MPU6050 sensor with an Arduino.
- To read and process accelerometer and gyroscope data to calculate tilt angles.
- To display the calculated tilt angles on an OLED display.
- To create a user-friendly interface for real-time tilt angle measurement.

COMPONENTS REQUIRED:

Hardware:

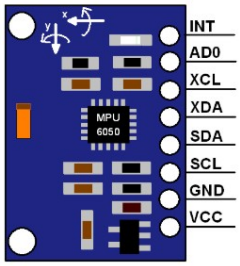
- Arduino Uno or any compatible board (e.g., Arduino Nano, Arduino Mega)
- MPU6050 sensor module
- OLED display (e.g., SSD1306, 128x64 pixels)
- Breadboard and jumper wires
- Resistors (if needed for pull-up on I2C lines)

Software:

- Arduino IDE
- Libraries:
  - Wire.h for I2C communication
  - Adafruit_GFX.h and Adafruit_SSD1306.h for OLED display
  - MPU6050.h for interfacing with the MPU6050

MPU6050:

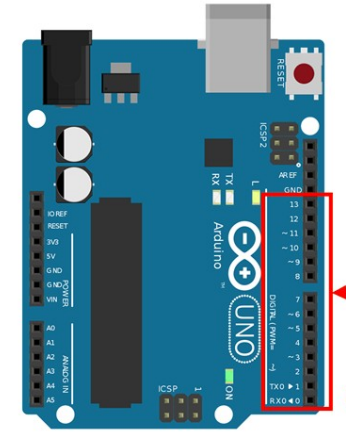
MPU6050 sensor module is complete 6-axis Motion Tracking Device. It combines 3-axis Gyroscope, 3-axis Accelerometer and Digital Motion Processor all in small package. Also, it has additional feature of on-chip Temperature sensor. It has I2C bus interface to communicate with the microcontrollers.



MPU6050 Pinout

ARDUINO:

Arduino boards typically use microcontrollers from the Atmel AVR family (like the ATmega328 on the Arduino Uno). These microcontrollers can read inputs from various sensors and control outputs like motors, LEDs, and displays.



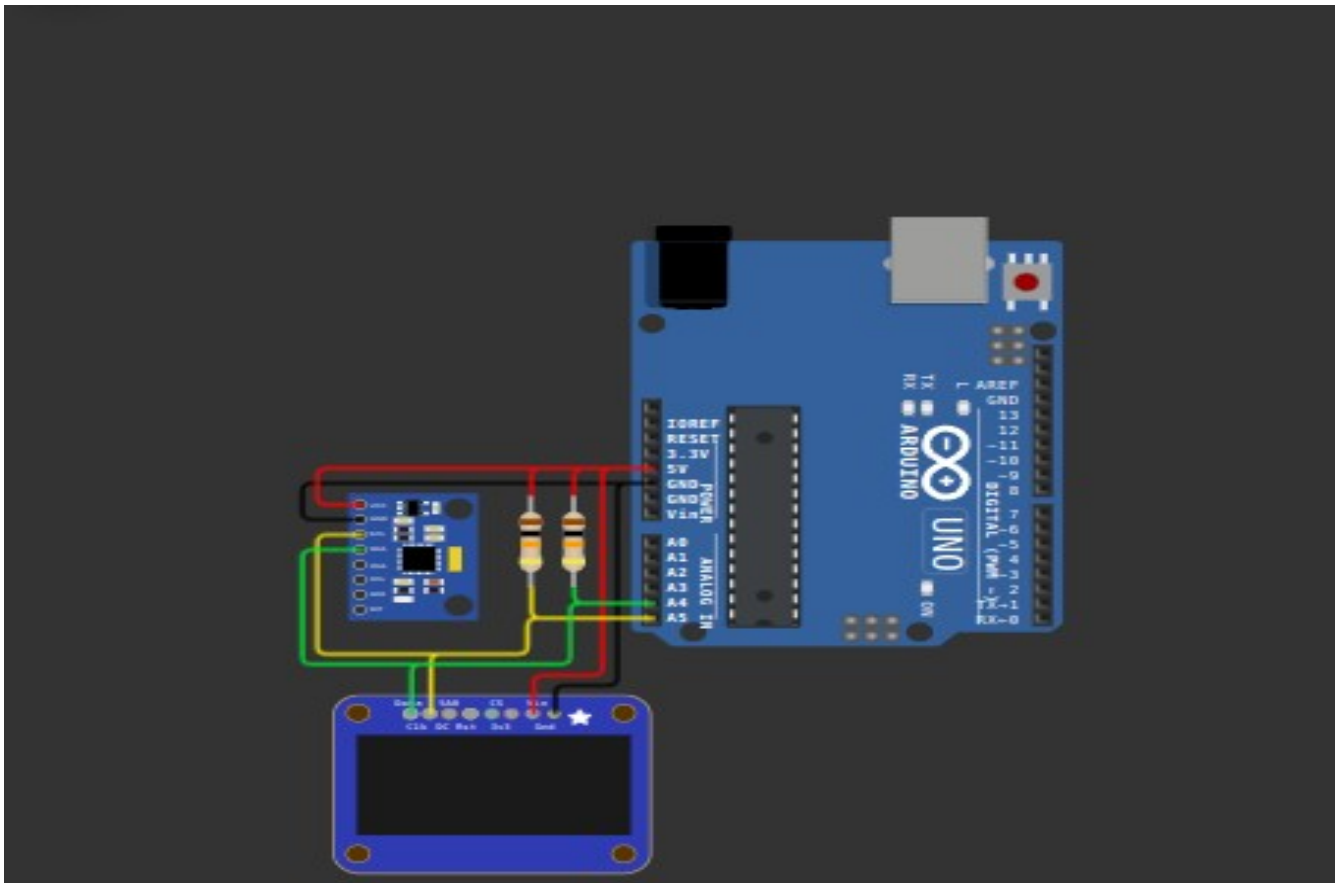
The Arduino Uno R3 has 14 digital I/O pins, of which 6 can be used for PWM (Pulse Width Modulation) output. These pins can be configured as either inputs or outputs, allowing users to connect various components like sensors, LEDs, and motors.

The board features 6 analog input pins (A0 to A5) that can read varying voltage levels (0 to 5V) and convert them into digital values (0 to 1023). This is useful for interfacing with analog sensors.

OLED:

OLED stands for Organic Light-Emitting Diode. It is a display technology that uses organic compounds to emit light when an electric current is applied. Unlike traditional LCDs (Liquid Crystal Displays), which require a backlight to illuminate the screen, OLED displays generate their own light, allowing for thinner, more flexible, and more energy-efficient screens.

SCHEMATIC:



PROGRAM:

```
#include <Arduino.h>

#include<Wire.h>

#include <Adafruit_GFX.h>

#include <Adafruit_SSD1306.h>


#define SCREEN_WIDTH 128

#define SCREEN_HEIGHT 64


// Declaration for an SSD1306 display

#define OLED_RESET    -1

#define SCREEN_ADDRESS 0x3C

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);


const int MPU_addr=0x68;

int16_t AcX,AcY,AcZ,Tmp,GyX,GyY,GyZ;


int minVal=265;

int maxVal=402;


double x;

double y;
```

```
double z;


void setup() {
Wire.begin();
Wire.beginTransmission(MPU_addr);
Wire.write(0x6B);
Wire.write(0);
Wire.endTransmission(true);
Serial.begin(9600);


if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C))
{
   Serial.println(F("SSD1306 allocation failed"));
   for(;;); // Don't proceed, loop forever
}
 display.display();
 delay(2);
 display.clearDisplay();


 display.clearDisplay();
 display.setTextColor(WHITE);
 display.setTextSize(2);
 display.setCursor(0,5);
 display.print("MPU Angles");
```

```
 display.display();
 delay(3000);
}

void loop() {
  Wire.beginTransmission(MPU_addr);
Wire.write(0x3B);
Wire.endTransmission(false);
Wire.requestFrom(MPU_addr,14,true);
AcX=Wire.read()<<8|Wire.read();
AcY=Wire.read()<<8|Wire.read();
AcZ=Wire.read()<<8|Wire.read();
int xAng = map(AcX,minVal,maxVal,-90,90);
int yAng = map(AcY,minVal,maxVal,-90,90);
int zAng = map(AcZ,minVal,maxVal,-90,90);

x= RAD_TO_DEG * (atan2(-yAng, -zAng)+PI);
y= RAD_TO_DEG * (atan2(-xAng, -zAng)+PI);
z= RAD_TO_DEG * (atan2(-yAng, -xAng)+PI);

Serial.print("AngleX= ");
Serial.println(x);

Serial.print("AngleY= ");
```

```
Serial.println(y);



Serial.print("AngleZ= ");
Serial.println(z);
Serial.println("-----------------------------------------");

display.clearDisplay();
display.setTextSize(2);
display.setCursor(0,0);
display.print("X: ");
display.println(x);

display.setTextSize(2);
display.setCursor(0,20);
display.print("Y: ");
display.println(y);

display.setTextSize(2);
display.setCursor(0,40);
display.print("Z: ");
display.println(z);
display.display();
```

```
delay(1000);
}
```

APPLICATIONS:

- Robotics:

  Stabilization , Navigation

- Gaming and Virtual Reality:

  Motion Tracking , Gesture Recognition

- Home Automation:

  Smart Devices , Security Systems

- Wearable Technology:

  Fitness Trackers , Health Monitoring

- Automotive Applications:

  Vehicle Stability Control , Navigation Systems

- Aerospace and Aviation

  Attitude Control , Flight Simulators

- Industrial Automation

  Machinery Monitoring: Robotic Arms

- Augmented Reality (AR):

  Interactive Experiences

FUTURE SCOPE:

Improved Accuracy and Precision:

- Development of advanced algorithms for sensor fusion, such as Kalman filters, to enhance the accuracy of tilt measurements.
- Integration of additional sensors (e.g., magnetometers) to provide more comprehensive orientation data.

Integration with IoT:

- Connecting the tilt measurement system to the Internet of Things (IoT) for real-time data monitoring and analysis.
- Enabling remote access and control through mobile applications or web interfaces, allowing users to monitor tilt angles from anywhere.

Machine Learning Applications:

- Utilizing machine learning techniques to analyze tilt data for predictive maintenance in industrial applications.
- Implementing adaptive algorithms that learn from user behavior to improve the responsiveness of the system in various applications.

Wearable Technology Enhancements:

- Development of compact, wearable devices that utilize tilt measurements for health monitoring, fitness tracking, and fall detection.
- Integration with other health sensors to provide a comprehensive health monitoring solution.

CONCLUSION:

In conclusion, the Tilt Angle Measurement Project not only serves as an educational tool but also lays the groundwork for future developments in motion tracking and orientation detection. As technology continues to evolve, the insights gained from this project can inspire further research .