

**MCTA 3203**

**MECHATRONIC SYSTEM INTEGRATION**

**SECTION 1**

**SEMESTER 2 2024/2025**

**LECTURER:**

**SIR WAHJU SEDIONO**

**SIR ZULKIFLI BIN ZAINAL**

**LAB REPORT 4A:**

**SERIAL COMMUNICATION (INERTIAL MEASUREMENT UNIT)**

**GROUP 6**

**DATE OF SUBMISSION: 7 APRIL 2025**

|  |  |
| --- | --- |
| **GROUP MEMBERS** | **MATRIC NUMBERS** |
|  Muhammad Nabil Iman bin Abd Rahman | (2313551) |
|  Muhammad Nazhan bin Mohamed Nadzri | (2313703) |
|  Muhammad Haziq Ajmal bin Md Kamal | (2312821) |

# ABSTRACT

This experiment focuses on integrating an MPU6050 IMU sensor with an Arduino board to develop a hand gesture recognition system. By capturing accelerometer and gyroscope data, the system identifies and categorizes predefined hand movements through a threshold-based algorithm. Sensor data is transmitted to a personal computer via serial communication, where Python is used for processing and visualizing the gestures. This setup highlights the practical application of IMU sensors in motion detection and serves as a foundation for future development in gesture-controlled systems.

**TABLE OF CONTENTS**

[ABSTRACT 2](#_TOC_250009)

[INTRODUCTION 2](#_TOC_250008)

[MATERIAL AND EQUIPMENT 3](#_TOC_250007)

[EXPERIMENTAL SETUP 3](#_TOC_250006)

[RESULT 4](#_TOC_250005)

[DISCUSSION 4](#_TOC_250004)

[CONCLUSION 5](#_TOC_250003)

[RECOMMENDATION 5](#_TOC_250002)

[ACKNOWLEDGEMENT 6](#_TOC_250001)

[STUDENT’S DECLARATION 6](#_TOC_250000)

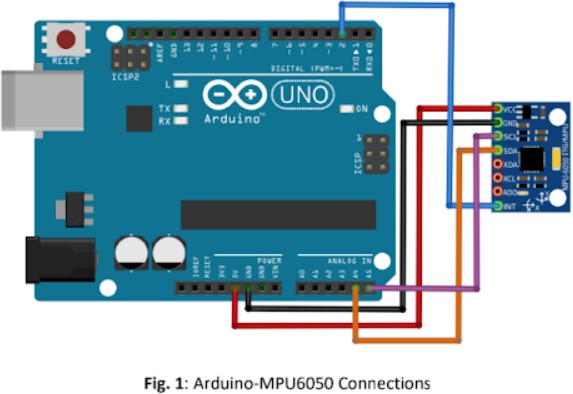
# INTRODUCTION

In the field of mechatronics, sensors and microcontrollers are fundamental for creating systems that interact with their environment. The MPU6050 sensor, which is the subject of this lab, is well-suited for various applications requiring motion and orientation data due to its compact size, affordability, and simple interface. Connecting an Arduino board to a personal computer with an MPU6050 involves initial steps in both hardware setup and coding. This integration allows for efficient reading and processing of data from the MPU6050, enabling innovative projects such as gesture recognition systems using both Arduino and Python.

# MATERIAL AND EQUIPMENT

* Arduino board
* MPU6050 sensor
* Computer with Arduino IDE and Python installed
* Connecting wires: Jumper wires or breadboard wires to establish the connections between the Arduino, MPU6050, and the power source.
* USB cable: A USB cable to connect the Arduino board to your personal computer. This will be used for uploading the Arduino code and serial communication.
* Power supply: If your Arduino board and MPU6050 require an external power source, make sure to have the appropriate power supply.
* LEDs of different colours

# EXPERIMENTAL SETUP

****

1. Connect the MPU6050 sensor to the Arduino board using the appropriate pins. The MPU6050 typically utilizes I2C communication, so ensure the SDA and SCL pins of the MPU6050 are connected to the corresponding pins on the Arduino (usually A4 and A5 on most Arduino boards).
2. Connect the power supply and ground pins of the MPU6050 to the Arduino's 5V and GND pins, respectively.
3. Confirm that the Arduino board is connected to your PC via USB.

# RESULT

The experiment's results, including a video of the experiment and images of the wiring, are available via the provided GitHub link.

# DISCUSSION

This experiment successfully established a serial communication link between the MPU6050 sensor, Arduino, and a PC, facilitating real-time data acquisition and gesture recognition. The threshold-based algorithm proved effective for identifying simple gestures, such as directional movements, by analyzing accelerometer and gyroscope values. However, the system's accuracy might be affected by noise in sensor readings or rapid hand motions, suggesting that advanced filtering techniques or machine learning models could enhance reliability.

Visualizing hand movement paths in an x-y coordinate system could further improve the system's usability by providing intuitive feedback to users. The incorporation of LEDs to indicate detected gestures adds a tangible output mechanism, making the system more interactive. Future work could involve integrating additional sensors or refining the gesture library to support more complex movements. Overall, this project demonstrates the MPU6050 sensor's versatility in motion-based applications and highlights the potential for scalable gesture recognition systems in fields such as virtual reality, robotics, and human-computer interaction. The combination of software processing and hardware integration exemplifies a practical approach to mechatronics system design.

# CONCLUSION

# In conclusion, this experiment successfully demonstrated the application of the MPU6050 IMU sensor in hand gesture recognition. Interfacing the MPU6050 with the Arduino allowed for real-time data capture of accelerometer and gyroscope readings, which were then used to classify specific hand gestures. The MPU6050, being small and easy to integrate, is well-suited for such applications, enabling effective motion and orientation tracking. By processing and analyzing the sensor data, the algorithm accurately recognized gestures based on predefined criteria. Furthermore, the visualization of hand paths in an x-y coordinate system clearly depicted each movement, demonstrating the sensor's capability in motion tracking applications. This experiment unequivocally proves the MPU6050's potential for developing gesture-based systems and showcases its viability in real-world applications extending into human-computer interaction and wearable technology. RECOMMENDATION

Firstly, students should familiarize themselves with the sensor calibration methods, the data collection process, and the interpretation of accelerometer and gyroscope data. For enhanced visualization of hand movements, it is recommended to overlay axis labeling, grids, or dynamic updates to display real-time motion traces. This can aid in better understanding movement patterns and gestures. To facilitate smooth and efficient experimentation, students must comprehensively describe the MPU6050 IMU interfacing experiment, including all necessary details for serial communication between the Arduino and PC.

# ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Sir Wahju Sediono for their guidance and support throughout this project. Our thanks also extend to the teaching assistants, for their

constructive feedback and assistance,which greatly contributed to the completion of this work.

# STUDENT’S DECLARATION

# Certificate of Originality and Authenticity

This is to certify that we are **responsible** for the work submitted in this report, that the original work is our own except as specified in the references and acknowledgement, and that the original work contained herein have not been untaken or done by unspecified sources or persons.

We hereby certify that this report has **not been done by only one individual and all of us have contributed to the report.** The length of contribution to the reports by each individual is noted within this certificate.

We also hereby certify that we have **read and understand** the content of the total report and no further improvement on the reports is needed from any of the individual's contributors to the report.

We therefore, agreed unanimously that this report shall be submitted for **marking** and this **final printed report** has been **verified by us.**