Inspiring Excellence

CSE360: Computer Interfacing

1. **Project Title : Sensor-Controlled 4-DOF Robotic Arm Using IMU and I2C Protocol**

Group Number : 07

Section: 05

ID	Name
21201489	MOHAMMAD AZMAIN HOSSAIN ALFI

Date : 05/05/2025

Table of Contents

Abstract

Gesture-based control of robots is becoming an effective way to simplify how we interact with machines, offering a more intuitive and hands-free approach compared to traditional controls. This project presents the development of a 4-degree-of-freedom (4-DOF) robotic arm that is controlled using real-time hand gestures. It utilizes the MPU6050 inertial measurement unit (IMU), which integrates a 3-axis gyroscope and 3-axis accelerometer. The arm is actuated using four continuous-rotation SG90 servo motors and an Arduino Nano microcontroller. Communication between the MPU6050 and the microcontroller is facilitated through the I2C protocol, enabling efficient and synchronized data transfer. The yaw, pitch, and roll values from the IMU sensor are used to control the movement of the robotic arm's joints, while two momentary switches are implemented to operate the pincher (gripper) mechanism. The system demonstrates an application of sensor-based control in robotics, showcasing how hand motion can be translated into robotic movement. The project aims to explore interfacing techniques, real-time sensor integration, and communication protocols for gesture-controlled devices, with expected outcomes including smooth gesture-to-motion translation and reliable control of the robotic arm.

2. Introduction

2.1 Problem Statement

Traditional robotic arm control systems often rely on joysticks, keyboards, or pre-programmed instructions, which can present steep learning curves and limited real-time adaptability. These conventional methods may also restrict accessibility for individuals with physical limitations (Riley, 2022). Moreover, in dynamic or hazardous environments, such control mechanisms can be inefficient, as they do not offer the intuitive and responsive interaction that hands-free or gesture-based methods provide (Rudd, Daly, & Cuckov, 2020). There is an increasing need for more natural and responsive control interfaces that closely mimic human gestures to enhance precision, usability, and safety in robotic applications.

2.2 Objectives

The main objective of this project is to design and implement a 4-DOF robotic arm that responds to real-time hand gestures captured using an MPU6050 inertial measurement unit (IMU) and controlled via I2C communication with an Arduino Nano microcontroller. Specifically, the project aims to:

- Develop a functional gesture-based interface that maps pitch, roll, and yaw data from the IMU to the robotic arm's movement.
- Achieve reliable motion control over the waist, shoulder, and elbow joints using continuous-rotation SG90 servos.
- Integrate two momentary switches for manual operation of the clamp (gripper) to ensure intuitive grasping and releasing.
- Demonstrate how sensor-based input through I2C can be used to simplify robotic control in real-world scenarios.
- Provide a cost-effective and replicable model for educational, assistive, and industrial purposes.

2.3 Significance

This project demonstrates the potential of gesture-controlled robotic systems using inertial measurement and interfacing technologies. By integrating an MPU6050 IMU with an Arduino Nano via I2C communication, the system offers intuitive, real-time control of a 4-DOF robotic arm, eliminating the need for complex programming or manual input devices. This approach not only simplifies human-machine interaction but also increases accessibility for users with physical limitations.

In real-world scenarios, such gesture-based systems have applications in healthcare (e.g., robotic surgery or assistive prosthetics), industrial automation (e.g., remote operation in hazardous environments), and education (e.g., teaching embedded systems and robotics). The use of low-cost, open-source hardware makes the project highly replicable and adaptable, serving as a practical example of modern interfacing principles in embedded and mechatronic systems.

3. Interfacing Design

3.1 Interfacing Components:

Arduino Nano Microcontroller : Acts as the central processing unit for reading sensor data and controlling the servos. It processes the IMU inputs and communicates with the servos through PWM signals.

MPU6050 IMU Sensor: Provides real-time pitch, roll, and yaw data using an onboard accelerometer and gyroscope. It communicates with the Arduino via the I2C protocol and translates hand orientation into movement instructions.

SG90 Continuous Rotation Servos (x4): Used to move each joint of the robotic arm (waist, shoulder, elbow, and clamp). These servos rotate continuously in either direction, making them suitable for smooth rotational motion control.

Momentary Push Buttons (x2): These are used to control the clamp mechanism. One button opens the clamp, and the other closes it, providing simple manual input for grasping operations.

Jumper Wires: Used for connecting all components on the breadboard and enabling communication and power flow between the microcontroller, IMU sensor, servos, and switches.

680-ohm Resistors: Used for debouncing the push buttons or as pull-down resistors to prevent floating states in digital input pins.

Breadboard: Serves as a prototyping platform to assemble and test the components without soldering. It enables easy wiring and reconfiguration during testing.

3.2 System Block Diagram

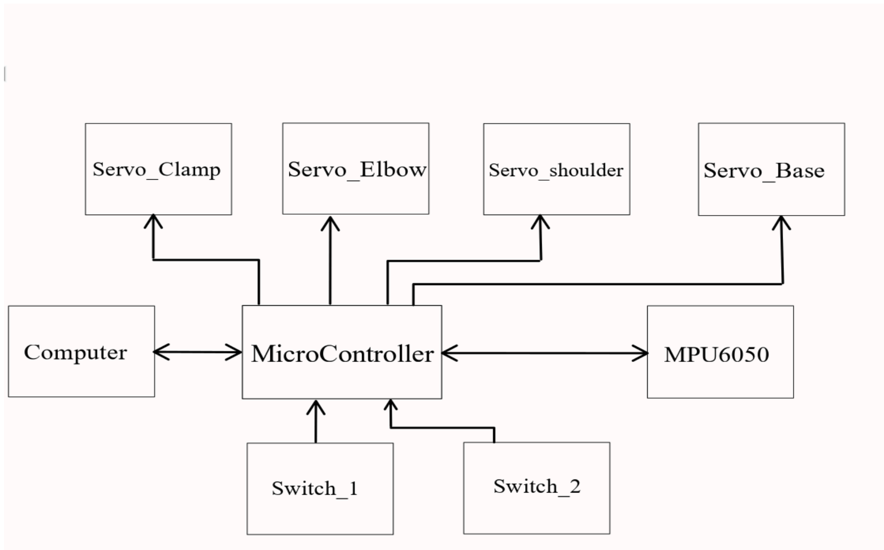


Figure:1 - System Block Diagram

3.3 Interfacing Challenges

Signal Noise from IMU Sensor: The IMU can produce noisy data due to vibrations or inconsistent movement. Using the Digital Motion Processing (DMP) library and sensor fusion algorithms helps to stabilize and filter raw data.

Voltage Mismatch: The servos and IMU sensor must be supplied with stable and correct voltages. A regulated power supply and proper grounding are essential to ensure reliable performance.

I2C Communication Conflicts: The I2C bus can be sensitive to improper wiring or insufficient pull-up resistance. Using correct pull-up resistors and keeping wire lengths short can minimize interference.

Servo Interference : When multiple servos operate simultaneously, they may cause brownouts or power drops. Powering servos externally with appropriate current ratings and isolating ground connections between logic and motor power can prevent this issue.

Physical Wear and Accuracy : SG90 servos are not precision servos and may lose calibration over time. Replacing them with higher-torque, metal gear servos would improve durability and movement precision.

4. Communication Protocols

4.1 Protocol Selection

The primary communication protocol used in this project is Inter-Integrated Circuit (I2C), which facilitates data transmission between the Arduino Nano and the MPU6050 IMU sensor. I2C is a widely-used serial communication protocol that allows multiple slave devices to be connected to a single master using just two communication lines: SDA (Serial Data) and SCL (Serial Clock).

4.2 Protocol Justification

The I²C (Inter-Integrated Circuit) protocol was selected over alternatives such as SPI and UART due to its minimal pin requirements, native support in the Arduino IDE, and direct compatibility with the MPU6050 sensor. I²C utilizes only two lines—Serial Data (SDA) and Serial Clock (SCL)—which simplifies wiring and conserves microcontroller I/O pins, making it ideal for embedded systems with limited resources (Texas Instruments, 2022).

This protocol supports sufficient data rates for real-time sensor reading, with standard modes operating up to 100 kbit/s and fast modes reaching up to 400 kbit/s, which is adequate for applications involving motion detection and control (Advanced Energy, 2020). Additionally, I²C's built-in addressing allows multiple devices to share the same bus without conflict, and its clock synchronization features help prevent data collisions, ensuring reliable communication between the sensor and the microcontroller (GeeksforGeeks, 2023).

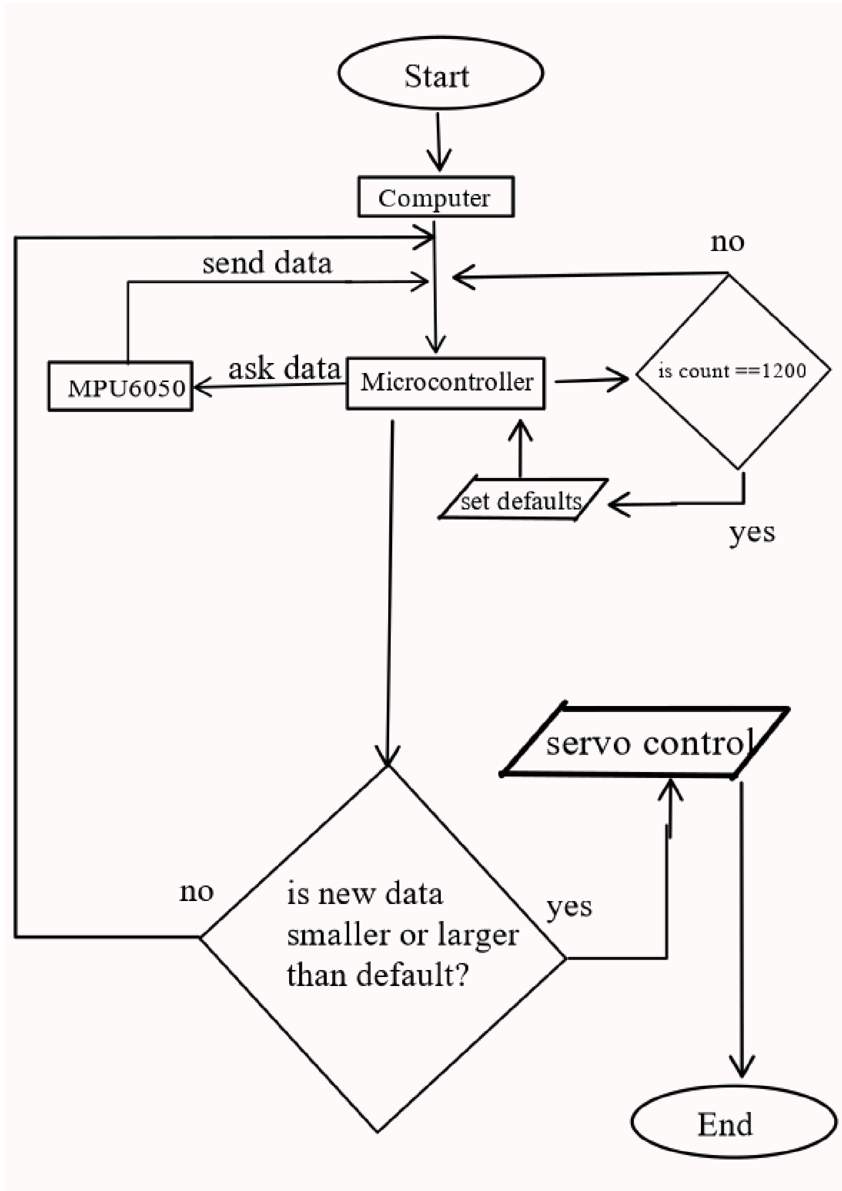
The MPU6050 sensor is specifically designed to communicate via the I²C protocol, facilitating straightforward integration with the Arduino Nano. The Arduino's Wire library provides comprehensive support for I²C communication, further simplifying the development process (Arduino Documentation, n.d.).

5. Implementation Plan

5.1 Methodology

```
//servo config
#include <Servo.h>
Servo waist;
Servo elbow;
Servo shoulder;
Servo clamp;
int stop = 93 //servo stop
//I2C config
#include "I2Cdev.h"
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
```

```
	#include "Wire.h"
#endif
#include "MPU6050_6Axis_MotionApps20.h"
void setup()
{
	#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
		Wire.begin();
		TWBR = 24;
	#elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
		Fastwire::setup(400, true);
	#endif
	mpu.initialize();
	waist.attach(6);
	shoulder.attach(9);
	elbow.attach(10);
	waist.write(93);
	shoulder.write(93);
	elbow.write(93);
	clamp.write(93)
```



}

5.2 Expected Outcomes

The gesture-controlled robotic arm is expected to interpret hand movements accurately using the MPU6050 IMU sensor and replicate those movements through coordinated actuation of the servos. The momentary push buttons should effectively control the opening and closing of the gripper. Overall, the system is expected to demonstrate reliable interfacing between hardware and software components, validating the chosen communication protocols and design methodology.

In terms of functionality, the arm should be able to perform basic pick-and-place tasks, track gesture orientation across four degrees of freedom, and respond in near-real-time. Reliability may vary based on sensor calibration, mechanical precision, and latency, but the system should offer an overall functional demonstration of gesture-based robotic control using I2C interfacing.

Challenges such as sensor drift, limited degrees of freedom, and mechanical inaccuracies have been acknowledged and will inform potential future improvements, including enhanced calibration techniques, upgraded actuators, and more sophisticated control algorithms.

6. Future Work and Potential Applications

6.1 Future Improvements

Incorporating additional sensors such as flex sensors and magnetometers can significantly improve motion detection accuracy. Flex sensors have been effectively utilized in robotic arms to capture nuanced finger movements, enhancing the precision of gesture-based controls (Kumar et al., 2022). Transitioning to more rigid materials like aluminum or utilizing 3D-printed components can reduce structural flex and improve the precision of the robotic arm. Studies have demonstrated that 3D-printed robotic arms can achieve high repeatability and are suitable for educational and light industrial applications (Ananias et al., 2023).
Implementing filtering techniques such as Kalman filters can mitigate sensor noise and drift, leading to more stable and reliable control of the robotic arm (Sepulveda Estay, 2023). Developing inverse kinematics algorithms enables the robotic arm to perform more precise and complex movements, allowing it to reach desired positions with greater accuracy (MathWorks, n.d.). Designing a gripper with proportional control can enable the robotic arm to handle a wider variety of objects and perform more delicate tasks. Proportional control allows for the adjustment of grip strength, reducing the risk of damaging objects during manipulation (Performance Motion Devices, n.d.). Additionally, integrating visual or haptic feedback can provide users with real-time information about the arm's position and status. Haptic feedback, in particular, has been shown to enhance user interaction by providing tactile sensations that correspond to the robotic arm's actions, thereby improving control and precision (Nature, 2023).

6.2 Potential Applications

Gesture-controlled robotic arms represent a growing area of innovation within human-machine interaction and control systems. By using an inertial measurement unit (IMU) to interpret hand movements, this project enables intuitive control of robotic mechanisms, eliminating the need for complex programming or manual interfaces. This method can significantly enhance usability in various sectors.

In the healthcare field, gesture-controlled robotic arms can support surgeons in minimally invasive procedures, allowing for finer control with reduced physical strain. Research has demonstrated the feasibility of using IMU-based gesture controls to enhance surgical precision and ergonomics (Praveen et al., 2017). Similarly, individuals with upper limb disabilities can benefit from prosthetic arms that mimic natural motion using IMU sensors, potentially improving mobility and quality of life (Ahmed et al., 2023).

In industrial automation, such systems can be applied to safely control robotic manipulators in hazardous environments—such as chemical plants or nuclear facilities—where direct human involvement is risky. Case studies have shown the effectiveness of gesture-controlled arms in such settings (Kumar & Sethi, 2022). Real-time control also benefits manufacturing lines where adaptive manipulation is needed, especially for tasks involving irregularly shaped objects or precision assembly.

In the field of education and research, low-cost platforms like Arduino and SG90 servos offer students and hobbyists an accessible entry point to learn sensor integration, real-time data acquisition, and servo actuation. Such platforms are widely adopted in engineering education to foster innovation in robotics and mechatronics (Arduino Education, 2019).

Furthermore, the use of the I2C protocol for communication between the IMU and microcontroller highlights efficient sensor interfacing practices. I2C allows multiple components to communicate over two wires, making it ideal for embedded systems with space and wiring constraints (JustDoElectronics, 2021).

Overall, this project demonstrates essential interfacing techniques while aligning with emerging trends in wearable technology, smart robotics, and human-machine synergy.

References

Advanced Energy. (2020). *Inter-Integrated Circuit (I²C) Basics* [Application Note]. https://www.advancedenergy.com/getmedia/1abf959e-3070-447f-9bad-b27d0e4c51ff/en-lv-i2c-basics-application-note.pdf

Ahmed, M. U., Rahman, M. M., Al-Amin, M., & Azam, M. S. (2023). IMU-based upper limb prosthesis for transhumeral amputees. *PubMed Central*. https://pubmed.ncbi.nlm.nih.gov/37941218/

Ananias, J., et al. (2023). Design and control of a 3D printed robotic arm. *ResearchGate*. https://www.researchgate.net/publication/374561152_Design_and_Control_of_a_3D_Printed_Robotic_ArmResearchGate+1DIVA Portal+1

Arduino Documentation. (n.d.). *Inter-Integrated Circuit (I²C) Protocol*. https://docs.arduino.cc/learn/communication/wire

Arduino Education. (2019). Building the ultimate robotics curriculum with Arduino. *Arduino Forum*. https://forum.arduino.cc/t/building-the-ultimate-robotics-curriculum-with-arduino/580964

GeeksforGeeks. (2023, May 15). *I²C Communication Protocol*. https://www.geeksforgeeks.org/i2c-communication-protocol/

JustDoElectronics. (2021). *Understand I2C communication protocol*. https://justdoelectronics.com/understand-i2c-communication-protocol/

Kumar, A., & Sethi, D. (2022). Gesture controlled robotic arm for hazardous environment handling. *IJIRST – International Journal for Innovative Research in Science & Technology, 6*(3). https://dif7uuh3zqcps.cloudfront.net/wp-content/uploads/sites/11/2022/07/03120901/Volume6_Issue3_Paper8_2022.pdf

Kumar, S., et al. (2022). Wireless hand motion controlled robotic arm using flex sensors. *ResearchGate*. https://www.researchgate.net/publication/365012063_Wireless_hand_motion_controlled_robotic_arm_using_flex_sensorsResearchGate+1Science Publishing Group+1

Maker Pro. (n.d.). How to interface Arduino and the MPU-6050 sensor. https://maker.pro/arduino/tutorial/how-to-interface-arduino-and-the-mpu-6050-sensor

MathWorks. (n.d.). What is inverse kinematics? *MathWorks*. https://www.mathworks.com/discovery/inverse-kinematics.htmlIntelligent Motion Laboratory+2MathWorks - Maker of MATLAB and Simulink+2Automatic Addison+2

Nature. (2023). The benefits of haptic feedback in robot-assisted surgery. *Nature*. https://www.nature.com/articles/s41598-023-46641-8

PNI Sensor. (n.d.). *Understanding magnetometers and their uses*. https://www.pnisensor.com/understanding-magnetometers-and-their-uses/pnisensor.com

Praveen, G., Vinayak, H., Bharath, N., & Basavaraj, N. (2017). Gesture controlled robotic surgical arm (GCRSA). *ResearchGate*. https://www.researchgate.net/publication/322230991_Gesture_controlled_robotic_surgical_arm GCRSA

Sepulveda Estay, D. (2023). The Kalman filter: A tool to help robots see through the noise. *Medium*. https://danielsepulvedaestay.medium.com/the-kalman-filter-a-tool-to-help-robots-see-through-the-noise-2687bf538cfb

Texas Instruments. (2022). *A basic guide to I²C* [Application Report SBAA565]. https://www.ti.com/lit/pdf/sbaa565