TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

KATHMANDU ENGINEERING COLLEGE

KALIMATI, KATHMANDU

MAJOR PROJECT PROPOSAL REPORT ON

Your Project

BY

KRITIKA DAHAL - KAT078BEI007

MAHAN SIGDEL - KAT078BEI009

NISHANA THAPA - KAT078BEI011

SABINA GHIMIRE - KAT078BEI016

TO

DEPARTMENT OF ELECTRONICS, COMMUNICATION AND INFORMATION ENGINEERING

KATHMANDU, NEPAL

MAY , 2024

Acknowledgement

We would like to express our sincere gratitude to the Department of Electronics, Communication, and Information, Kathmandu Engineering College, for providing us with an opportunity to initiate our major project as a part of a syllabus. And a special thanks goes to our supervisor, **Er. Sagun Manandhar** and **Er. Anmol Bajracharya**, for assisting and aiding us in every possible way in this project. We are deeply thankful to the Project Coordinator, **Er. Sujan Sapkota**. We would also like to mention a special note of thanks to the esteemed Head of the Department of Electronics, Communication, and Information, **Er. Suramya Dahal**. We would also like to extend our gratitude to every teacher of the Department of Electronics, Communication and Information, for their guidance.

Table of Contents

List of Figures

List of Tables

List of Abbreviation

AI Artificial Intelligence

GPIO General Purpose Input Output

IoT Internet of Things

ML Machine Learning

RAM Random Access Memory

OS Operating System

BRNN Bidirectional Recurrent Neural Networks

GRU Gated Recurrent Units

MIMO Multiple Input Multiple Output

SoC System-on-Chip

USB Universal Serial Bus

Wi-Fi Wireless Fidelity

NLP Natural Language Processing

TTS Text-To-Speech

API Application Programming Interface

ARM Advanced RISC Machine

RISC Reduced Instruction Set Computer

IP Internet Protocol

MCU Micro-Controller Unit

Chapter 1: Introduction

1.1 Background Theory

The rapid advancement of human-computer interaction (HCI) technologies has paved the way beyond traditional input devices towards more intuitive interfaces. Among these, Gesture-based systems have emerged as a powerful mode of interaction where users engage with digital systems through physical movements, particularly of the hands. Human hands are capable of complex movements and precise control, making them an ideal medium for gesture-based input. This evolution is particularly significant in applications such as gaming, virtual and augmented reality, assistive technology, and robotics.

Leveraging this, modern wearable systems incorporate a variety of sensors to translate hand and finger gestures into digital commands. The MPU6050, a 6-DoF inertial measurement unit (IMU) combining a 3-axis gyroscope and a 3-axis accelerometer, is widely used for real-time tracking of wrist orientation and movement, crucial for recognizing hand gestures [1], [2]. For detecting finger movements, flex sensors are employed, which vary resistance based on bending, providing analog signals proportional to flexion [3]. Alternatively, mechanical switches, activated by levers attached to finger components, offer a digital input method with tactile feedback. Data from these sensors are processed by microcontrollers like the Arduino Mega, which provides ample I/O support. For seamless interaction with external applications, the ESP8266 Wi-Fi module facilitates wireless data transmission, ensuring low-latency communication between the wearable device and the host system [4].

Device housing commonly utilizes 3D-printed PLA filament, chosen for its lightweight properties, customizability, and suitability for ergonomic wearable enclosures that maintain sensor alignment [5]. In the software domain, gesture data is mapped to real-time responses within a digital environment. Game engines, such as Unreal Engine, support external hardware integration, enabling the visualization of hand movements from a first-person perspective [6]. This establishes a seamless connection between physical actions and virtual reactions, particularly effective in immersive puzzle-based or simulation games. Ultimately, gesture-based input systems enhance user experience by aligning digital control with natural human motion, proving ideal for applications ranging from gaming to assistive technology.

1.2 Problem Statement

1.3 Objectives

1.4 Scope

The project encompasses several key areas for development and future expansion:

- Integration with machine learning models to improve gesture recognition accuracy
- Addition of haptic feedback for more immersive interaction
- Expansion to full-body motion capture using additional wearable sensors
- Development of a mobile or desktop interface for visualizing and mapping gestures
- Incorporating voice + gesture multimodal control systems

1.5 Applications

The data glove system finds applications in various fields including:

- Virtual and Augmented Reality (VAR)
- Assistive Technology
- Gaming
- Robotics Control
- Smart Environments / IoT Applications
- Educational Tools

Chapter 2: Literature Review

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Chapter 3: Related Theory

3.1 Hardware

MPU6050 Sensor: The MPU6050 is a widely used 6-axis MEMS-based Inertial Measurement Unit (IMU) that integrates a 3-axis accelerometer and a 3-axis gyroscope within a single chip.It is capable of detecting linear acceleration in the range of $\pm 2g$ to $\pm 16g$ and angular velocity from $\pm 250°/s$ to $\pm 2000°/s$, making it highly suitable for motion tracking and gesture recognition applications. A notable feature of the MPU6050 is its onboard Digital Motion Processor (DMP), which performs real-time sensor fusion using algorithms such as Kalman or complementary filtering. This significantly reduces noise and drift in gyroscopic data, enabling stable orientation tracking through the calculation of quaternions or Euler angles (roll, pitch, and yaw). The sensor communicates with microcontrollers through the I²C interface, supporting clock speeds between 100 kHz and 400 kHz for efficient data exchange. Its compact design, reliability, and real-time capabilities make it ideal for wearable systems. It enables precise and responsive motion capture for interactive systems in gesture-based applications.

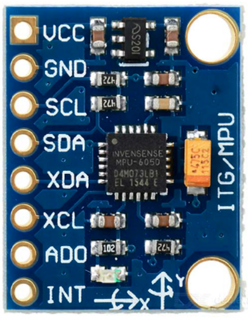


Figure 3.1: **MPU6050 Sensor**

Arduino Mega: The Arduino Mega is an open-source microcontroller board based on the ATmega2560, designed for projects requiring extensive input/output operations and greater memory capacity. It features 54 digital I/O pins, 16 analog inputs, and four UARTs for serial communication, making it suitable for complex hardware interfacing. With 256 KB of flash memory and a 16 MHz clock speed, it can handle multiple sensors and real-time data processing efficiently. In this project, the Arduino Mega serves as the central controller, managing input from multiple MPU6050 sensors and switches to ensure accurate and synchronized gesture-based interactions within the game environment.

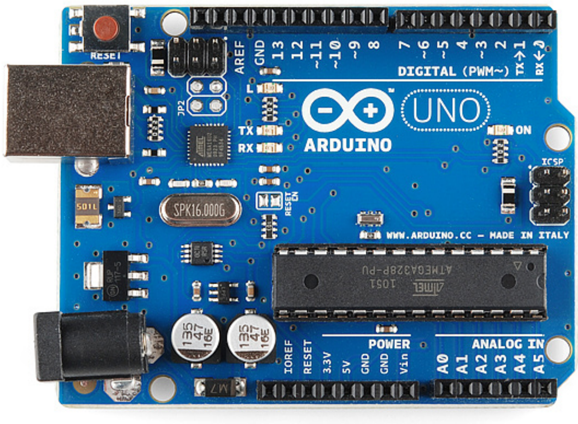


Figure 3.2: **Arduino Uno Board**

ESP8266: The ESP8266 is a low-cost, high-performance Wi-Fi microcontroller based on a 32-bit RISC CPU core (Tensilica L106), operating at 80–160 MHz. It integrates TCP/IP protocol stack and supports IEEE 802.11 b/g/n standards, enabling wireless connectivity with low power consumption (80 mA active mode). It enables devices to connect to wireless networks and communicate over the internet or within local networks. The ESP8266 supports multiple modes such as station, access point, and both simultaneously, making it highly versatile for wireless communication. It can be programmed using the Arduino IDE and is capable of handling HTTP requests, data transfer, and remote control functionalities. The ESP8266 is used to explore wireless communication possibilities between the hardware controller and the game system, potentially allowing untethered interaction.

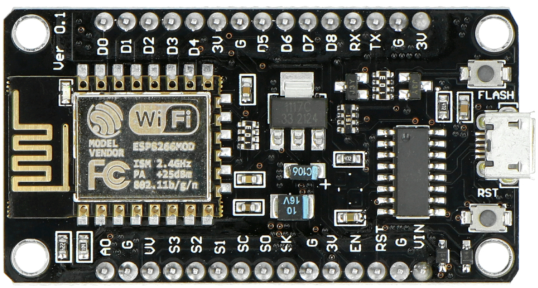


Figure 3.3: **ESP8266 Wi-Fi Module**

Flex Sensors: Flex sensors are passive resistive devices that change their resistance based on the amount of bend applied to them. Typically constructed using a flexible substrate coated with conductive ink, their resistance increases as the sensor is bent. This property allows them to detect the degree of bending or curvature, making them suitable for applications involving motion capture, wearable electronics, and gesture recognition. When integrated with microcontrollers, the analog resistance change can be converted into meaningful input data. In this project, flex sensors are considered for detecting finger movements by measuring the degree of bend in each finger.

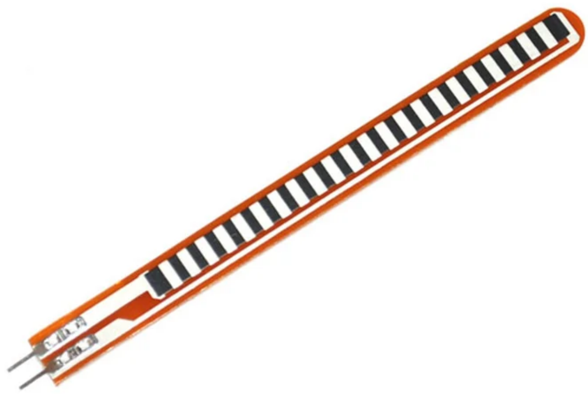


Figure 3.4: **Flex Sensor**

3.2 Software

Arduino IDE: The Arduino Integrated Development Environment (IDE) is the core programming tool for the Arduino Uno microcontroller in this project. Using C/C++, it configures the microcontroller to process digital signals from switches and analog data from MPU6050 sensors. Essential libraries like Wire.h and MPU650.h manage I²C communication, while serial protocols handle data transmission to the rendering engine. The IDE's debugging tools, including serial monitors, are crucial for verifying signal integrity and latency. Firmware algorithms integrate sensor fusion and debouncing logic for accurate gesture detection. This open-source platform significantly aids rapid prototyping and hardware-software integration for real-time interactive systems.

Rendering Engine: The Rendering Engine is a fundamental software component that generates real-time visual output based on physical hand gestures captured by the hardware. Initially considering Unreal Engine, this prototype utilizes Blender's integrated EEVEE and Cycles engines for rendering and simulation. The engine displays a first-person perspective with visible virtual hands, mirroring gestures like finger bends and wrist rotations instantly. This real-time visual feedback is vital for player immersion and interaction. The engine processes data from the microcontroller, supplied through middleware, to dynamically adjust hand poses and object interactions. Its capability to reflect hardware input with minimal latency ensures intuitive and responsive gameplay, serving as the core of the gesture-controlled gaming experience. Optimized for low latency with GPU acceleration, it manages environmental elements and physics simulations via Blender's Python API.

Blender: Blender is an open-source 3D creation suite which is utilized for designing and developing game assets. This includes 3D modeling, UV mapping, texturing, and rigging, crucial for creating realistic hand models and interactive objects. Blender's EEVEE and Cycles engines provide real-time previews and high-fidelity final renders, respectively. Its Python scripting capabilities facilitate customization and workflow automation. Blender functions as both a design tool and a visual integration layer, ensuring in-game visuals accurately reflect physical gestures captured by the hardware with minimal latency through automated workflows linking it to the Arduino's output.

Python: Python serves as the middleware layer, connecting hardware data with the rendering engine. Custom scripts, utilizing libraries like PySerial, parse serial data from the Arduino, converting raw sensor values into actionable game inputs. Python's integration with Blender via the bpy module maps gestures to in-game animations and automates tasks. It can also be used externally to interpret sensor data and relay it to the rendering engine via custom protocols. Python's versatility and extensive library support are crucial for efficient data translation and seamless interaction between the wearable hardware and the virtual environment, enabling real-time mapping of physical movements to game commands.

Chapter 4: Feasibility Study

4.1 Technical Feasibility

The project demonstrates strong technical feasibility by combining electronics, communica tion, and information processing in a practical and achievable manner. The Arduino Mega microcontroller acts as the brain of the system, providing sufficient processing power and I/O pins to accommodate multiple sensor inputs, including MPU6050 sensors, tactile push buttons, and optional flex sensors. These electronic components assist in accurately detecting hand gestures and finger movements. For communication, the ESP8266 module is considered for wireless data transfer between the hardware system and the game engine, facilitating real-time interaction with low delay. Serial communication protocols (either wired or wireless) will be utilized to ensure smooth and continuous data exchange from the Arduino to the computer running the game.

The information processing aspect involves translating raw sensor inputs into meaningful com mands for the game. This includes gesture detection using both switch activations and motion readings from the MPU. All of this will be managed using standard Arduino programming, with open-source libraries that expedite development. The modular hardware setup allows for f lexible testing, updates, and part replacement if necessary. Our design supports iterative devel opment, meaning we can test and improve as we build. Key strengths of the project include its adaptability, cost-effective components, and future scalability, making it a reliable and feasible system to implement with the tools and skills available to our team.

4.2 Economic Feasibility

The proposed system demonstrates strong economic feasibility with a cost-effective approach to hardware development and implementation. Most of the required electronic components and sensors are readily available in the local market and within our college resources. The primary components, like Arduino Mega, ESP8266, MPU sensors, and connecting wires, are economically accessible, with relatively low procurement costs compared to specialized gam ing interface systems. The use of PLA filament provides a budget-friendly prototyping solu tion, allowing multiple design iterations without significant financial investment. Open-source software platforms like Arduino IDE and Unreal rendering Engine further reduce development expenses by eliminating expensive proprietary software licensing costs.

The project's modular design enables incremental development, meaning team members can progressively invest in components as needed, spreading out potential expenses. Potential cost savings are achieved through utilizing existing college laboratory equipment and leveraging team members' existing technical skills, which minimizes additional training or external con sultation expenses. The overall economic viability is enhanced by the project's scalable na ture, potential for future refinement, and the use of widely available, low-cost technological components. The minimal financial requirements make this project an economically attractive research and development initiative within the current institutional infrastructure.

Chapter 5: Methodology

5.1 System Block Diagram

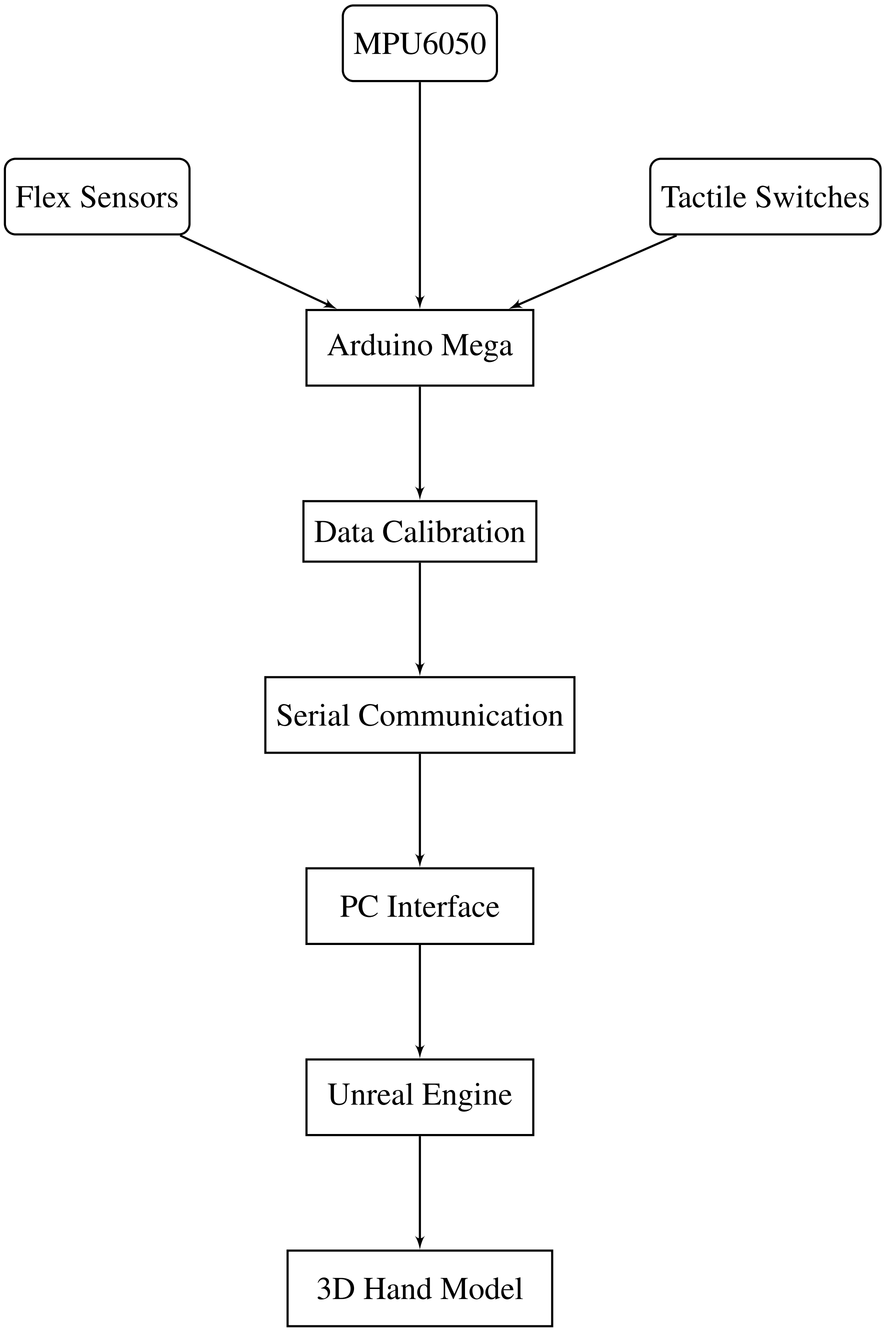


Figure 5.1: Detailed Block Diagram of Hand Motion Replicator System

5.2 Algorithm

1. System Initialization
   - Configure MPU6050 for 6-DoF motion tracking
   - Initialize analog inputs for flex sensors
   - Set up digital inputs for tactile switches
   - Configure serial communication parameters
2. Sensor Data Acquisition
   - Read accelerometer and gyroscope data from MPU6050
   - Measure resistance values from flex sensors
   - Monitor state of tactile switches
   - Apply calibration offsets
3. Data Processing
   - Filter sensor noise using moving average
   - Calculate hand orientation from IMU data
   - Convert flex sensor values to finger angles
   - Detect finger press events
4. Data Transmission
   - Package processed data into structured format
   - Implement error checking
   - Transmit data packets via serial communication
5. Virtual Model Update
   - Parse received data packets
   - Update hand skeleton parameters
   - Apply inverse kinematics for finger movements
   - Render updated hand model

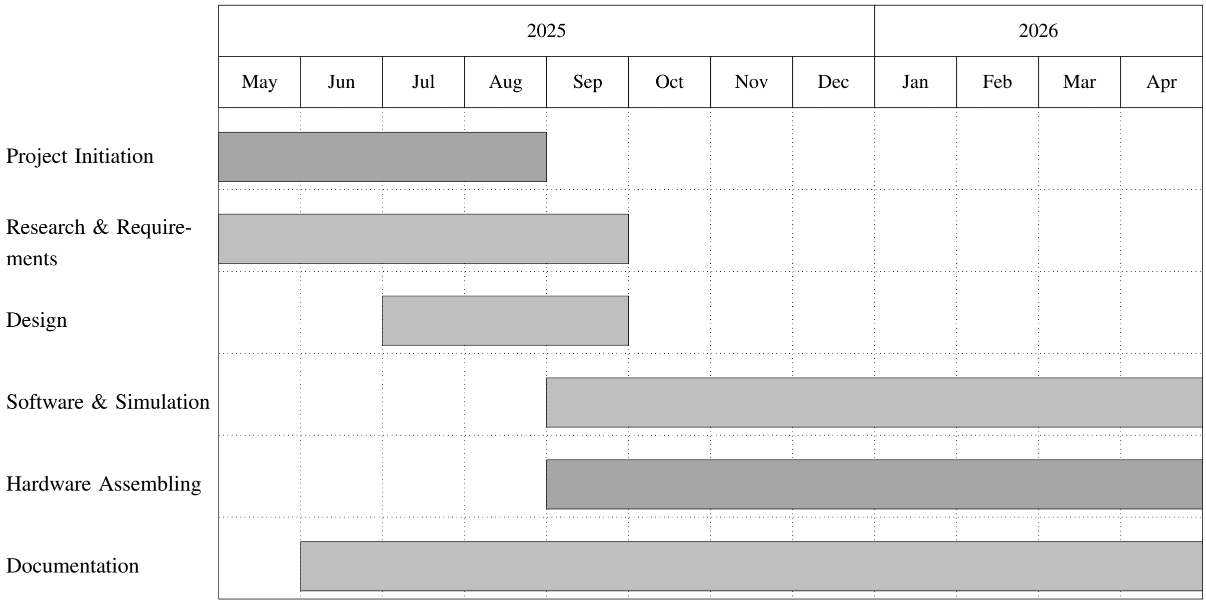
Chapter 6: Expected Output

- **Real-time Hand Motion Tracking:**
  The system will accurately track and measure hand movements using the MPU6050 sensor for orientation ($\pm 2^\circ$ accuracy), flex sensors for finger bending (0° to 90° range), and tactile switches for touch detection. The combined sensor data will provide comprehensive hand position and gesture information with minimal latency (¡100ms).

- **Virtual Hand Replication:**
  The tracked hand movements will be replicated in real-time on a 3D hand model in Unreal Engine. The virtual hand will accurately mirror all finger movements, hand rotations, and touch interactions with smooth articulation and natural movement visualization. The system will maintain a consistent frame rate above 30 FPS to ensure fluid motion reproduction.

- **Interactive Response System:**
  The system will provide immediate feedback through the virtual hand model, responding to user inputs with less than 50ms latency. This includes accurate finger bend representation ($\pm 5^\circ$ accuracy), precise hand orientation tracking, and immediate response to touch inputs with over 95% reliability. The communication system will maintain a stable 60 Hz update rate to ensure seamless interaction between the physical and virtual hands.

Gantt Chart


2025
2026
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Jan
Feb
Mar
Apr
Project Initiation
Research & Requirements
Design
Software & Simulation
Hardware Assembling
Documentation

Cost Estimation

Components and Cost Distribution

S.N	Components	Quantity	Price	Availability
1	Momentary tactile push button	25	1000	Daraz Nepal
2	MPU 6050	5	2500	Daraz Nepal
3	ESP 8266	1	900	Daraz Nepal
4	Arduino with usb cable	1	1700	Daraz Nepal
5	Wire diameter 0.4mm ballpoint spring	10	700	Daraz Nepal
6	Flex sensor	10	7000	Daraz Nepal
7	PLA(1kg)	1	3000	Daraz Nepal
8	Gloves	2	800	Daraz Nepal
9	Superglue(vega)	3	330	Daraz Nepal
10	Enamel wire(50m)	1	1300	Daraz Nepal
11	Connecting wire	40	900	Daraz Nepal
12	Joystick controller with cable	1	1700	Daraz Nepal
13	Li-ion battery(3.7v) with charger	4	4000	Daraz Nepal
		Total Cost	**NRs. 25,830**	

Table 6.1: Components and Cost Distribution

References

1. X. Zhang, M. Liu, and Y. Zhao, “Design of angle detection system based on MPU6050,” in *Proc. Int. Conf. Intelligent Transportation, Big Data & Smart City*, Changsha, China, 2017, pp. 159–162.

2. S. J. Park and S. Y. Lee, “Real-Time Motion Recognition Using an Accelerometer,” *Sensors*, vol. 15, no. 1, pp. 135–148, 2015.

3. P. Kumar, J. Verma, and S. Prasad, “Hand Data Glove: A Wearable Real-Time Device for Human-Computer Interaction,” *Int. J. Adv. Sci. Technol.*, vol. 43, pp. 15–23, Jun. 2012.

4. A. Khandual and A. Biswal, “IoT Based Smart Glove for Hand Gesture Recognition,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 5, no. 2, pp. 384–389, Mar. 2019.

5. M. Nasri et al., “Design and Implementation of a 3D Printed Hand Gesture Recognition System Using Arduino and Flex Sensors,” *IEEE Access*, vol. 7, pp. 55371–55378, 2019.

6. T. Baudel and M. Beaudouin-Lafon, “Charade: Remote Control of Objects Using Free-Hand Gestures,” *Commun. ACM*, vol. 36, no. 7, pp. 28–35, 1993.