International Islamic University Islamabad

Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

Instrumentation & Measurements
Complex engineering problem

Students Name : Bushra Nazir

Registration no : 1071-F22

Date of Submission : 31-5-2025

Submitted To: Dr. Abdulbasit

Table of Contents

Introduction

Modern video games, such as Tekken, are typically controlled using handheld devices where each button corresponds to an action like a punch, kick, or jump. While effective, this method limits player interaction to finger movements only.

In this project, I propose transforming the traditional button-based control into a motion-based system, where physical body movements—such as actual punches or kicks—trigger in-game actions.

I was inspired by a personal promise I made to my younger brothers to modify their PlayStation controller during the summer vacation so they could play using their own body movements. By using this assignment as a foundation, I aim to better understand the circuit design and sensor integration required for the system. This not only fulfills my academic goals but also helps me prepare for building the real version of the project later.

Design Methodology

This project aims to convert body movements into game controls. The idea is to allow players to use actions like a kick, punch, or jump to play, instead of pressing buttons on the controller.

To do this,

1. Motion sensors are placed on the player's hands and legs. These sensors detect fast or sudden movement, such as a punch or a kick. Each sensor is connected to a small wireless module that sends the motion data using Bluetooth. This allows the player to move freely without being limited by wires.

2. Since I want the player to move freely without being limited by wires, the system should be wireless. A Bluetooth module is used to send the signal from the sensors to the main processing unit.

3. A microcontroller receives the signal, analyzes the data, and identifies which action was performed. Based on that, it sends a command to a small physical device attached to the controller,

4. which presses the correct button.

This method helps connect physical body movements to the game in real time, making the experience more interactive and engaging.

Analysis

The sensor used in this system can measure movement in three directions: X, Y, and Z. It gives data about how fast something is moving and how it is rotating . This helps us understand what kind of movement is happening for example, if the hand moves forward fast, it could be a punch; if the leg moves upward quickly, it might be a kick. To tell the difference between real actions and normal body motion, we can set a certain limit . Small or slow movements can be ignored, while strong and fast movements will be counted as valid actions. Using both acceleration and rotation data together helps to make the system more accurate.

Even though the system is not fully built yet, this analysis shows that by reading the sensor data and checking if it passes certain limits, we can detect body movements and use them to control the game.

Sensor Selection

I considered several sensors for this project.

1. MPU6050   Price: $1.5– $3

Pros:
- Has 3 axes of acceleration (X, Y, Z) and 3 axes of rotation.
- Communicates with Arduino via I2C (easy to connect).
- Very popular in motion tracking projects.
- Very cheap and widely available.
- Has lots of Arduino libraries, making it easy to program.

Cons:
- No magnetometer, so it doesn't measure Earth's magnetic field (no compass).
- Some small drift over time in gyro data.

2. MPU9250   Price: $4 – $7

Pros:
- Higher accuracy than MPU6050.
- Measures all directions precisely (good for full body tracking).
- Suitable for AR and VR projects.

Cons:
- More expensive.
- Programming is a bit more complex.
- Often overkill for simple projects like this one.

3. LSM9DS1   Price: $8 – $12

Pros:

- Very high accuracy in motion and orientation tracking.
- Reduces drift found in some other sensors.
- Great for full 3D body tracking.

Cons:

- Expensive.
- Too complex for a simple game controller project.
- Harder to find locally compared to MPU6050.

4. MPU6886   Price: $4 – $6

Pros:

- Good accuracy.
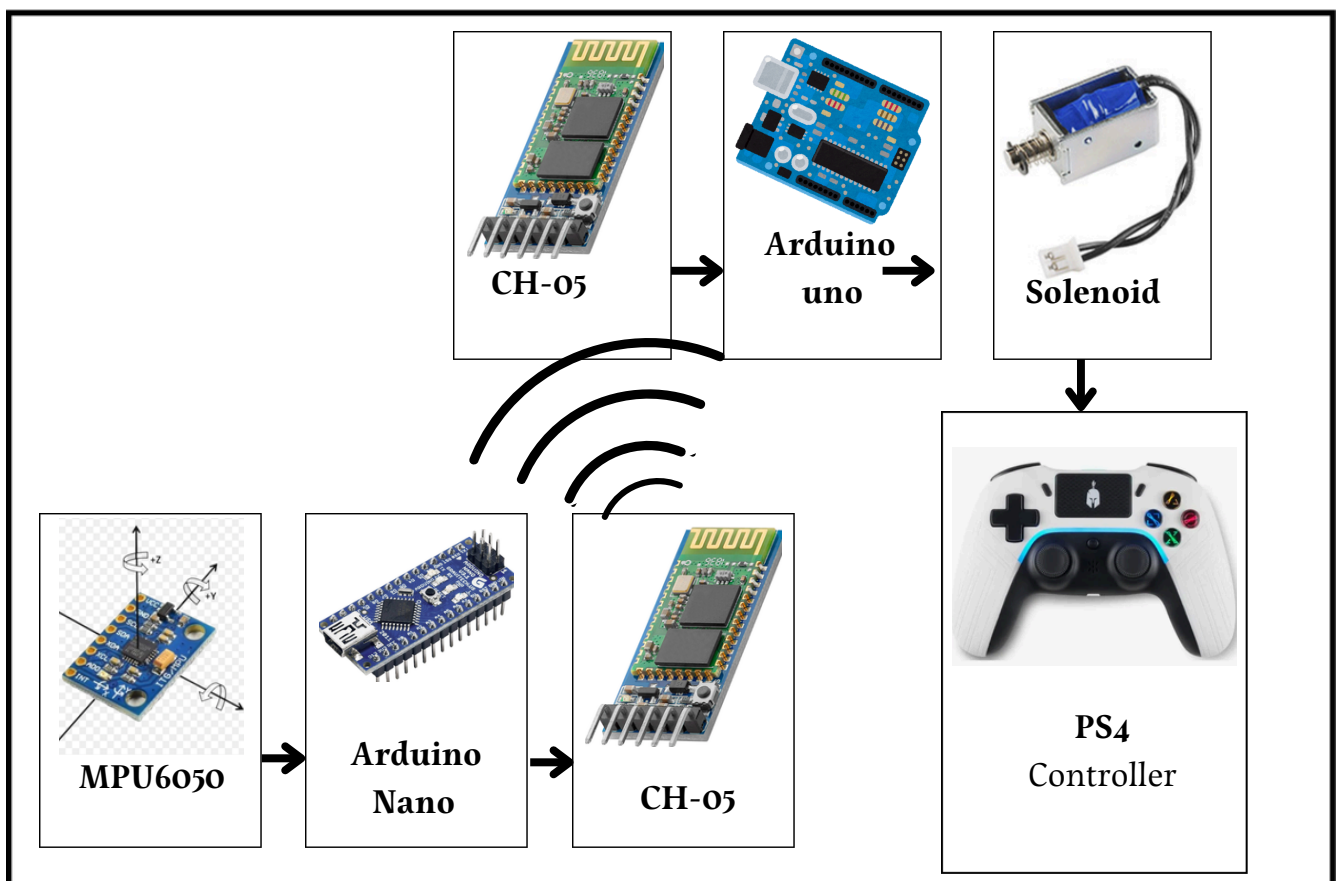- Software compatible with MPU6050 libraries.

Cons:

- Not as widely available as MPU6050.
- No big advantage over MPU6050 for this project.

Why I chose MPU6050?
Because it's super cheap and perfect for my budget, plus I know my brothers will probably break everything the next day anyway :) !

Block Diagram

Conclusion

This project shows how motion sensors can be used to make gaming more interactive and fun. By using real body movements like kicks and punches instead of buttons, players can enjoy a more physical and immersive experience. The system is simple, affordable, and can be improved in the future with more advanced sensors or machine learning for better motion detection.