EP 315: Microprocessor Lab

Gesture Mouse

by

Kartik Patekar(160260018)

Saipriya Satyajit(160260023)

Report of Arduino Project

Under Guidance of

Professor Pradeep Sarin

Physics Department

IIT-Bombay

November 2018

Contents

1 Abstract **1**

2 Project Report **3**

2.1 Block Diagram 3

2.2 Details of Major Work 3

2.2.1 Interfacing Devices 3

2.2.2 Establishing Data Transfer 4

2.2.3 Control and Sensing 4

2.2.4 Single Mouse Click 5

2.2.5 Advanced Mouse Activities 5

2.2.6 Assembling the board 6

2.3 Output component 6

Chapter 1

Abstract

The device made in this project is a Gesture Mouse. The idea of this device is to control the mouse pointer of a computer using hand movements and gestures. Such a device can find its application as a substitute of the wireless mouse and will be much cooler than it. It will enable us to control the pointer using gestures, allowing us to read papers or watch movies comfortably from far away. It can also allow disabled people to use the computer using their feet. The major components that were required for the project were

- Magnetometer (HMC5883l)
- Arduino Nano
- Bluetooth module
- Voltage regulator (KA7805).

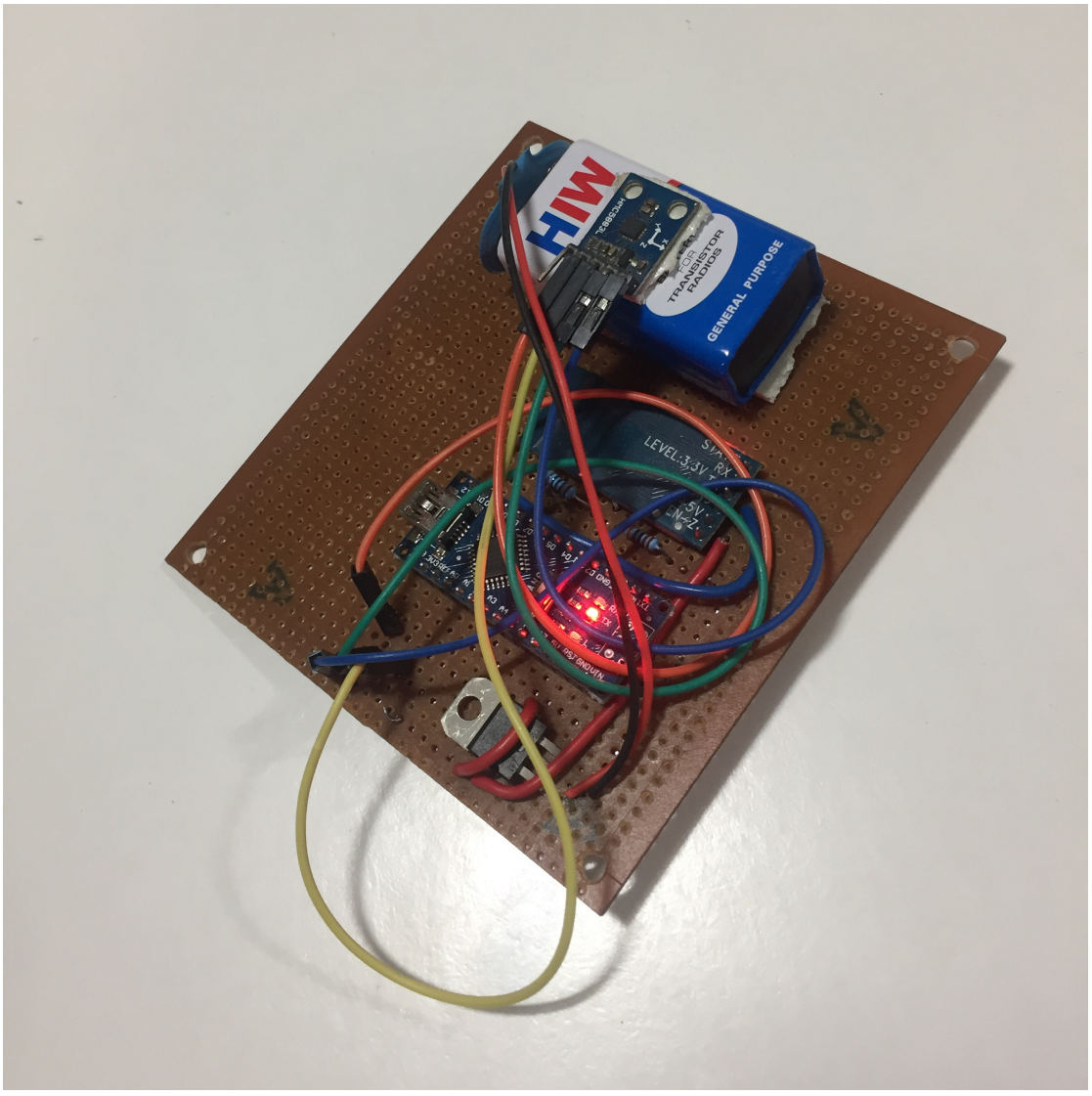
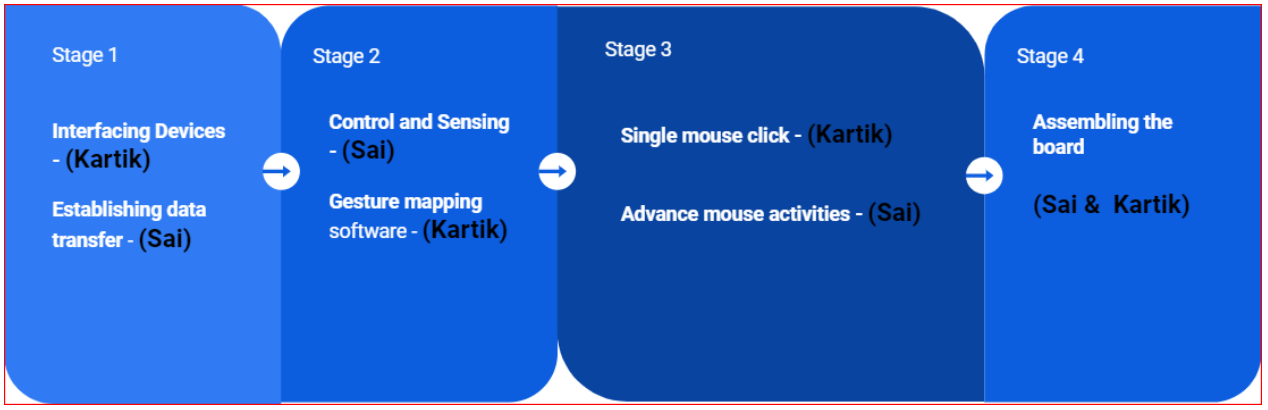


FIGURE 1.1: Final Project

Chapter 2

Project Report

2.1 Block Diagram



2.2 Details of Major Work

2.2.1 Interfacing Devices

In the first week, we worked on interfacing arduino with MPU-6050 (accelerometer+gyroscope) and HC-05 (bluetooth). MPU-6050 had its dedicated library while Bluetooth can be interfaced using SoftwareSerial library.

Our intention was to use accelerometer data (acceleration) to detect motion of mouse and calculate the position of pointer. However, After during stage two we realized that the accuracy provided by Accelerometer was too low for precise movement of the mouse pointer. We tried to use the gyroscope data (angular acceleration) but faced the same issue.

We finally used the QMC-5883l (Magnetometer) which provides the magnetic field in 3 directions. It can be used to find the orientation of device with respect to a reference direction. For interfacing, we used the library Mecha_QMC5883L.

2.2.2 Establishing Data Transfer

We then used Python code on the PC to establish continuous data transfer with the bluetooth module. We used the pybluez library of python to connect with the bluetooth device and receive data from it. On arduino, we wrote a simple code to send numbers to the PC through the bluetooth.

2.2.3 Control and Sensing

We tried to use accelerometer/gyroscope for calculating the position of the mouse. However, there were a few issues. The data of MPU-6050 is not accurate enough for our purpose as we had to integrate it twice to obtain the position. Aso, the data provided contains the acceleration due to gravity, and hence to take care of it we had to keen an axis fixed. This was an issue as we wanted the mouse to move in all three direction (third direction to implement click).

Unlike accelerometer, magnetometer gives orientation of the device instead of its acceleration. Since we wanted the velocity to be higher at larger angle, we used a linear relationship between velocity and angle. Also, for stabilisation, the velocity should be very low when the angle is small, and therefore flattened it out near origin.

We used the following function to obtain velocity in x and y direction:

$$v = 6 * (theta - 10) \quad if\ \theta > 15 \tag{2.1}$$

$$v = -6 * (theta + 10) \quad if\ \theta < -15 \tag{2.2}$$

$$v = 30 * (theta/15)^4 \quad if\ |\theta| <= 15 \tag{2.3}$$

The graph is shown below.

Using the data of magnetometer, the arduino calculates the velocity and integrate to find the change in coordintes of the mouse. Then it sends the data to python code on the PC, where we used pyautogui library to control the mouse position.

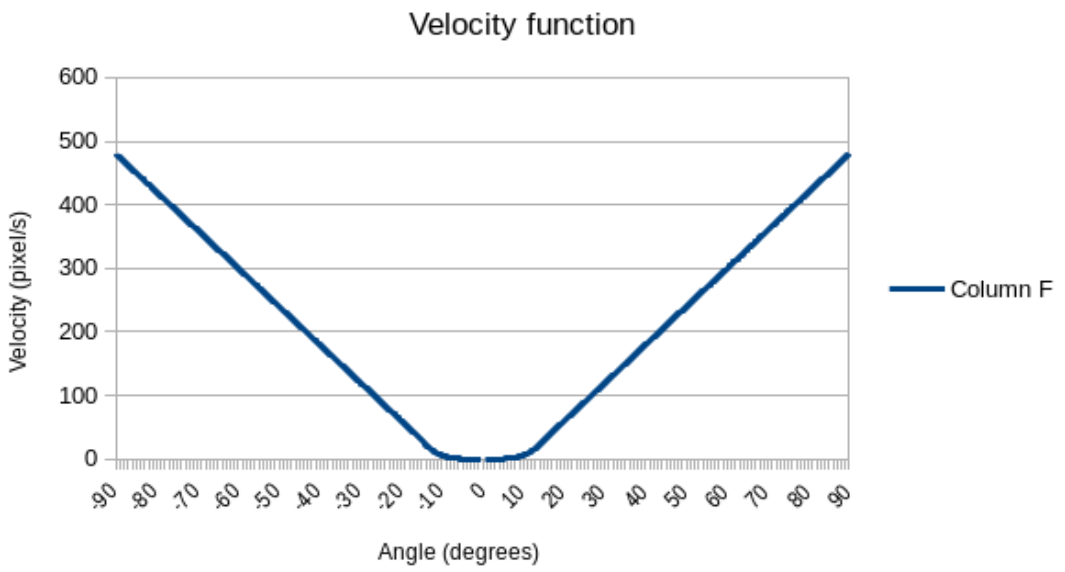


FIGURE 2.1: Velocity function

2.2.4 Single Mouse Click

Since we used the motion in 3-d to control the mouse pointer, we had to come up with a complicated motion to implement multiple mouse functionalities. Since the motion should be comfortable enough for human hand, we chose to implement left click using sudden twist of hand in the clockwise direction. The left mouse button clicks when the angular velocity of the roll direction becomes larger than 6 rad/s.

We faced the issue that such sudden movement affected the velocity in x and y direction as well. To solve this issue, we paused the mouse position for 0.4 seconds after a sudden motion in roll direction.

2.2.5 Advanced Mouse Activities

We next implemented right click and and left mouse button hold functionality in the mouse.

For right click, we used the sudden twist in the anti-clockwise direction. The method was similar to the left click.

However, for holding down the left mouse button, we had to use the same motion as left click. Since this was impossible, we scrapped off the left click, and estabilited hold functionality by holding the button on sudden movement in clockwise direction and releasing on sudden movement in anti-clockwise direction. The left click could

now be done by performing hold and release in quick succession. All other mouse functionalities(right click, mouse motion) were paused during holding and for next 0.4 seconds.

2.2.6 Assembling the board

In the last stage, we assembled everything on a PCB (instead of a glove) which can be used to control the mouse. We did this on the last day but ended up burning the magnetometer, and then later replaced it with a new magnetometer.

The final working board is shown in figure 2.3. It is a wireless device with an onboard power source. We used a KA-7805 to convert the voltage from 9V to 5V

2.3 Output component

In the final demonstration, we were able to control the mouse smoothly and were able to implement single clicks (both left and right), and also long press of a click which enabled us to implement scrolling using the scrollbar. The magnetometer reads the magnetic field components in three directions and then Arduino calculates the angles of orientation using which as input it then calculates the mouse coordinates which are sent to the computer using a bluetooth module where they are used as input to the python code which controls the mouse.

The the video of the working of the mouse can be seen here

Other photos and videoes related to the project can be found here

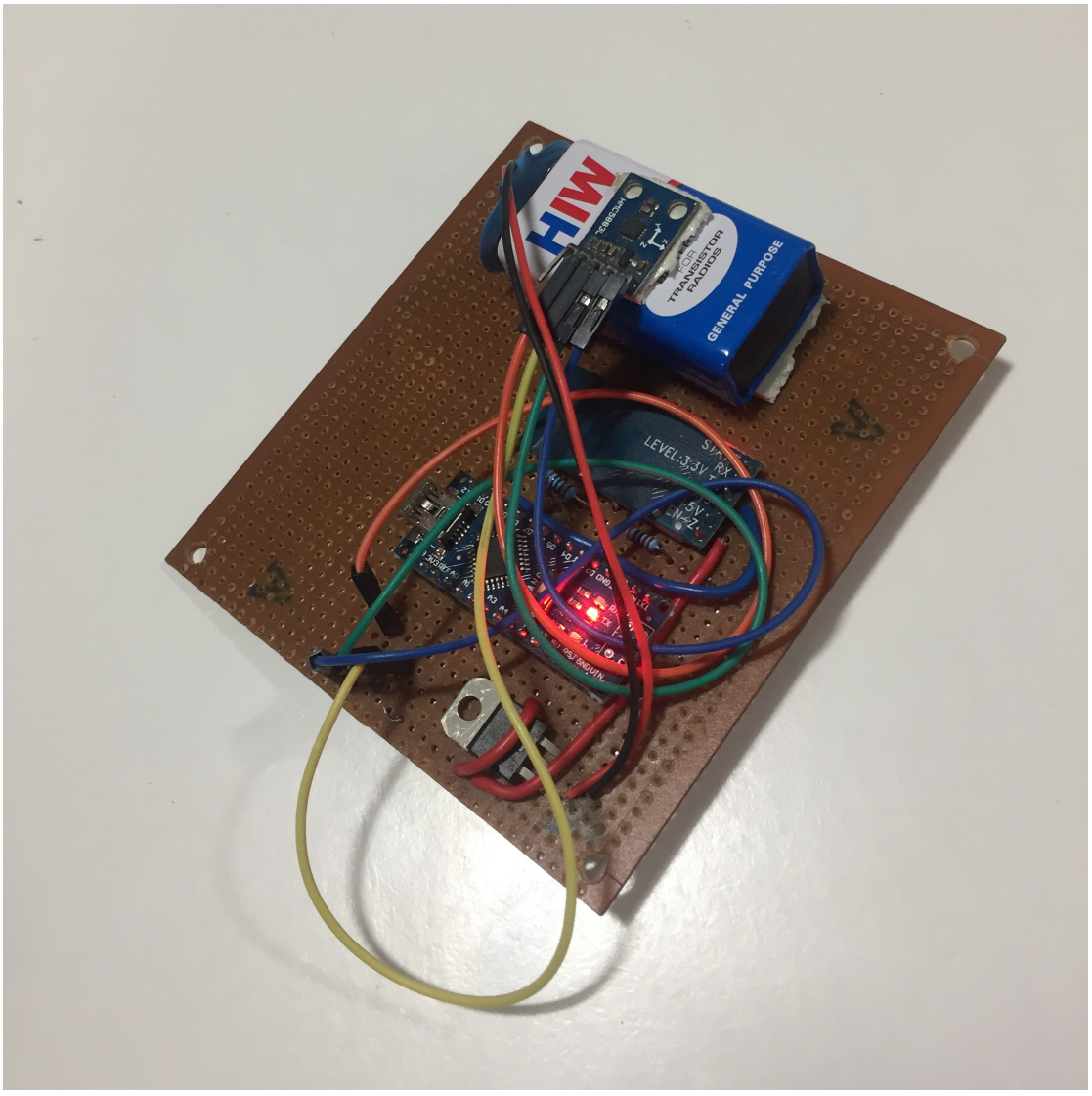


FIGURE 2.2: Final Project