ISSN PRINT 2319-1775 Online 2320-7876, www.ijfans.org
Vol.11, Iss.10, Oct- 2022

Research Paper

System to Design a Baseless and Wireless Mouse using Different Sensors

Dr. Manoj Verma[1], Dr. Anish Kumar Choudhary[2], Mr. Shreyas Pagare[3], Dr. Rupesh Shukla[4], Dr. Manish Shrivastava[5]

[1]CDGI Indore, [2]CDGI Indore, [3]CDGI Indore, [4]ILVA Commerce and Science College Indore, [5]CDGI Indore

Abstract

It is possible to utilize hand gestures for nonverbal communication in multiple fields, including but not limited to communication for deaf-mute individuals, robot control, human-computer interaction (HCI), home automation, and medical applications. The design and development of a baseless, wireless mouse employing several sensors is the subject of this essay. This paper also describes the implementation of establishing a connection between various sensors using ESP32 microcontroller. This microcontroller also used to interact between the system and device using inbuilt Bluetooth module in ESP32. This article also describes understanding of various technologies and modules which are used in the design and development of the mouse.

General Terms

Wire Less Mouse, Sensor.

Key words: ESP32 microcontroller, Bluetooth, Mouse.

Introduction

Systems that rely on sensor technology or the Internet of Things are now advancing quickly and are able to detect and convey external information in a number of scenarios. Companies from a variety of industries may now use data remotely, which has been especially useful when there have been lockdowns due to the pandemic[1]. IoT sensors are now offered in a variety of shapes and sizes, making it even simpler to adapt the technology to match the demands of individual businesses. The market, however, cannot afford to get stale [1]. Using development boards has enabled communication between numerous sensors and computer system interaction as a result of increased development. The development boards' built-in Bluetooth chips are making communication simpler [1][3]. The most recent mouse technology available today, however, is only available as an optical or virtual mouse. These technologies either use IR cameras or light sensors[2].

Making an efficient mouse technology that can interface various sensors with a development board, which is also used to interact it with the system through a wireless medium, can provide the functionalities of an optical mouse while providing stability to use the mouse for the system in a surface independent medium, which is more dependable to use for an interacting user.

Typically, a mouse uses two axis coordinates to move the pointer across a system window.[4]

By filtering the sensor's raw accelerometer signals, accelerometer sensors are used to determine the orientation of the object. The triaxial accelerometer sensor, which produces the sensors orientation by filtering the raw signals provided by its movement, is utilized for cursor movement in the system window. The buttons of a regular mouse are used to conduct various clicks; when pushed, the click associated with that button is executed. Two more types of accelerometers are employed in the system to provide the raw acceleration in the suggested direction in order to achieve similar functionality [5].

In order to determine the click for the system, the raw acceleration data from both sensors is analyzed. The ESP32 development board has Arduino code that is created to generate and filter raw data. Python code is used to get the data from the system and connect the device to the system wirelessly. [6]. Later, the received information was once again filtered and put to use in order to move the pointer and track system clicks.[2]These two algorithms operate concurrently and are linked through Bluetooth utilizing the BLE technology included into the ESP32 development board to communicate with the system's built-in Bluetooth connection.[2][6]

1. Literature Review

Earlier technology or mouse to control the cursor and all related operations are generally trackball, optical mouse, wireless mouse and 3D mouse. All have different techniques to perform the operations.

Trackball

This is an earlier model of computer mouse that is utilized in many laptop keyboards as a computer control device. The device, which resembles an upside-down mouse with a ball inside, is made comprised of a ball held by a socket and sensors that detect rotation about two axes. Although the trackball was created in 1952, optical mice have now taken its place [7]. To move a cursor without using their arms, the user rolls the ball with their thumb, fingers, or palm. To improve grip, some trackballs feature textured rubber or metal surfaces, while others could have removable coverings. The device's front, back, bottom, and any of its buttons provide the user with the ability to issue a variety of instructions, including, such as selecting an object or dragging and dropping it [7].

Figure-1: Trackball Mouse

Optical Mouse

With an optical computer mouse, movement on a surface is converted to pointer movement on a screen. Additionally, it has buttons for a variety of functions. Nowadays, practically all mice are categorized as optical mice, as opposed to the now-outdated mechanical mice, regardless of whether they are wired or wireless or use LEDs or lasers [8]. An optical mouse makes use of a light source, a photo detector (usually a CMOS sensor), and a digital signal processor (DSP) to detect movement. Light from the light source first shines onto the surface. The light bounces back and is captured by a CMOS sensor, which records thousands of photos per second and sends them to a DSP for processing. The computer then receives this information and adjusts the pointer location on the screen as necessary [8].

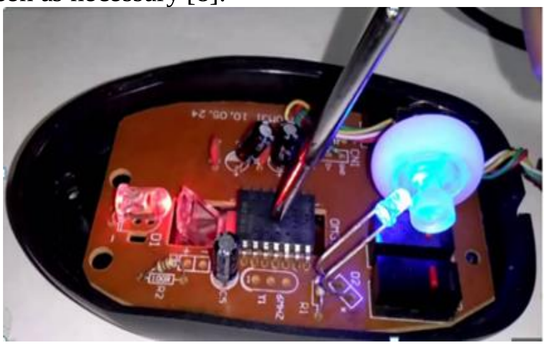


Figure 2: Optical Mouse

Wireless Mouse

Radio Frequency (RF) technology underlies the operation of wireless mice. The transmitter and the receiver are the two parts of a wireless mouse. A wireless mouse's internal transmitter emits electromagnetic impulses or radio data signals, which are nothing more than information about mouse movements that has been encoded [9].

The receiver, a USB 2.0 or USB 3.0 device, receives the encoded data from the transmitter and decodes it before sending it to the driver software set up in the operating system controlled by the CPU [10]. The receiver runs on computer or laptop power and does not require batteries to function. Although a transmitter is a standalone device, the receiver's CPU analyses data from the movement of cursors across windows, clicks, and scrolls, and then relays that information to the transmitter so it can respond appropriately to changes in movement [9].

Figure 3: Wireless Mouse

ESP32 Microcontroller

The ESP32 is really a line of inexpensive module-compatible microcontroller chips. The ESP32 is an improved version of the ESP8266, a chip that "surprise" experiment in the west in 2014. Its capabilities were largely unknown at the time since the original ESP8266 initially appeared on a module called the ESP-01, which had very little English documentation. The ESP8266 swiftly gained popularity as the certification was translate into English and a huge number of experimenters learned about its capabilities [11].

The ESP32 design was enhanced in a number of areas over the ESP32 design. While the ESP8266 just has WiFi (which, of course, the ESP32 also has), it also has BLE (Bluetooth Low Energy). It comes with a dual-core architecture and is speedier. Additionally, it has an ultra-low power mode that is perfect for battery-powered applications [12]. Programming will be made simpler by the inbuilt micro-USB port found on many of these boards. Some boards lack this capability and necessitate the use of an additional FTDI adaptor for programming. The ESP32-WROOM chip serves as the foundation for most of these boards.The ESP32 DEV KIT and ESP32 NODEMCU boards are readily available on Amazon and eBay and highly well-liked[12]. Additionally, the well-known ESP32-Cam board combines an ESP32 with a tiny video camera and a microSD card connection (this board requires an FTDI adapter for programming).

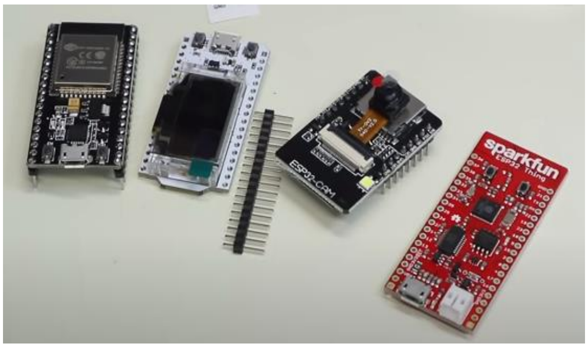


Figure 4. ESP32 development board

Programming of ESP32

The Arduino platform is the most straightforward method to begin developing code for the ESP32 platform. Based on Atmel microcontrollers, this open-source platform was designed for quick prototyping. The Java-based Arduino integrated development environment [13] is used to create software for microcontrollers. Although it has been somewhat changed and now contains features such as Wiring Language, this application was created with a learning environment in mind.A microcontroller with a boot program is required for wiring. These days, various types of microcontrollers may be loaded in the Arduino IDE in addition to Atmel microcontrollers. In order to install support for the development of ESP32 microcontrollers in the Arduino IDE environment, Espressif chose to use this option and construct a plugin known as the Arduino core for the ESP32 Wi-Fi chip [13]. This environment's straightforward configuration and strong community support are advantages.You may start prototyping the ESP32 microcontroller fairly rapidly with the help of this environment. Simpler initiatives should be developed in this setting since larger projects lose their focus and capacity to maintain themselves. Additionally, this platform is excellent for gaining expertise with the ESP32 microcontroller's development [14].

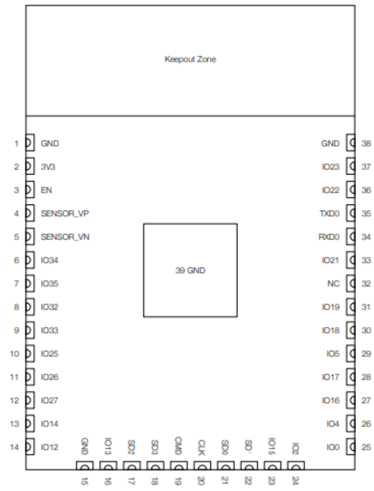


Figure 5: ESP32 Pin configuration

MPU6050 Sensor

Have you ever wondered how, when you tilt or rotate your smartphone, the screen automatically adapts in a frame? With the aid of the IMU (Inertial Measurement Unit) MPU6050, we can complete the above task. IMU, or inertial measurement unit, is the scientific name for MPU6050. Three-axis accelerometer and three-axis gyroscope data is determined by this six-axis motion tracking gadget. The main benefit of this board is the inclusion of a digital motion processor. The crucial function of the DMP (Digital Motion Processor) is that it does extremely complex calculation and operations on its own, sparing the microcontroller from such tasks. It offers motion information like roll, pitch, yaw, angles, feeling of landscape and portrait, etc.[15]

External IIC modules, like a magnetometer, can be connected to the module's two auxiliary pins.

Figure 6: MPU6050 Sensor

Working of MPU6050

The MPU6050 serves as the module's primary component, and its circuitry is quite straightforward. A voltage regulator is also necessary because the module requires 3.3V to operate. A 4.7k resistor is used to pull the interrupt pin low, and two 4.7k resistors are used to pull the IIC lines high [17].

The IIC bus may be used to read data from the MPU6050 module. Any change in motion will have an effect on the mechanical system, which will change the voltage [20]. The IC then reads these voltage changes exactly using a 16-bit ADC, records them in the FIFO buffer, and activates the INT pin [17][19]. As a consequence, a microcontroller may read data from this FIFO buffer via IIC communication since the data is ready for reading. This module, however, is used on a variety of platforms, including Arduino, by utilizing publicly available libraries [18].

Interfacing MPU6050 with ESP32 Microcontroller

It is very easy to interface the MPU6050 with ESP32. Four pins of MPU6050 are used to interface it with the ESP32 [11].

Connections of MPU6050 with ESP32

1. VCC to VIN

ISSN PRINT 2319-1775 Online 2320-7876, www.ijfans.org
Vol.11, Iss.10, Oct- 2022

2. GND to GND
3. SDA to SDA (GPIO 21)
4. SCL to SCL (GPIO 22)

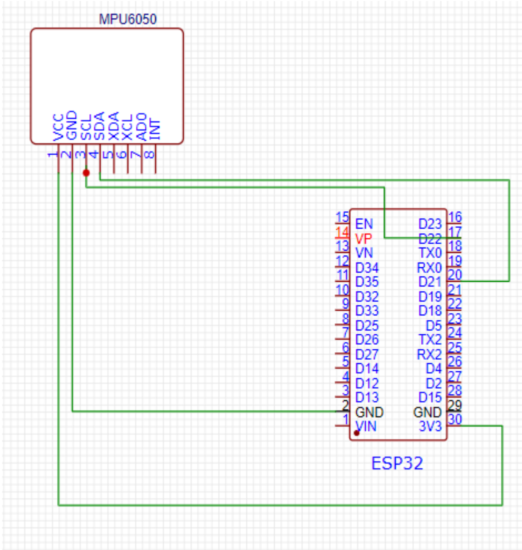


Figure 7: ESP32 interfacing with MPU6050

ADXL335 Sensor

The ADXL335 is a 3-axis accelerometer that is compact, thin, and low-power, offering voltage outputs that are signal-conditioned [21]. Its capabilities include detecting acceleration with a minimum full-scale range of 3g and monitoring both static and dynamic acceleration caused by motion, shock, or vibration in tilt-sensing applications [21]. The user can choose the accelerometer's bandwidth by adjusting the Cx, Cy, and Cz capacitors at the Xout, Yout, and Zout pins. Bandwidths can be selected from a range of 0.5 Hz to 1600 Hz for the X and Y axes and 0.5 Hz to 550 Hz for the Z axis to suit the application [22]. Additionally, the ADXL335 is available in a small, low-profile 16-lead plastic lead frame chip scale package, measuring only 4mm x 4mm x 1.45mm [23].

Working of ADXL335

An Analog Devices ADXL335 compact, low-power, low-noise triple axis MEMS accelerometer serves as the module's central component. It is capable of measuring both dynamic acceleration and static acceleration brought on by motion, shock, or vibration [22] [23].

When a force is applied to a mass, it generates displacement, which causes it to accelerate according to Newton's Second Law of Motion, F=ma, as observed by the mass's sensor. Analog accelerometers, like digital accelerometers, are based on two principles: piezoelectric sensing and capacitive sensing. Both have distinct advantages and disadvantages [21]. Similarly, the ADXL335 accelerometer operates on the capacitive sensing principle because it is an analog accelerometer. The capacitance of an accelerometer with capacitive sensors varies as it is moved in any direction. The interface controller detects changes in capacitance when its analog voltages change [22].

ISSN PRINT 2319-1775 Online 2320-7876, www.ijfans.org
Vol.11, Iss.10, Oct- 2022

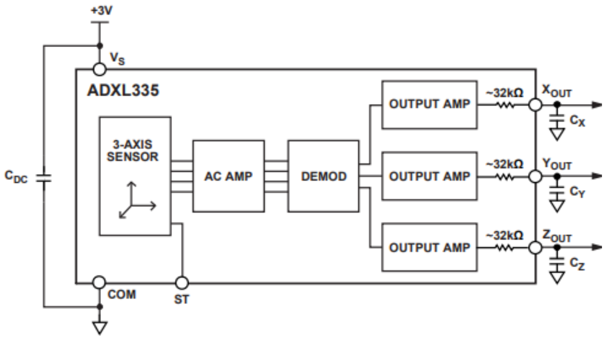


Figure 8: Integrated Circuit of ADXL335

Interfacing ADXL335 with ESP32

Connection of ADXL335 with ESP32 are very simple. For supplying power to the accelerometer VCC pin of ADXL335 is connected to VIN of ESP32. Ground is provided by connecting GND to GND in ESP32. ADC pins of ESP32 are used to get the analog output from the accelerometer. ADXL335 consist three output pins each responsible to give voltage along the respective axis [11] [24].

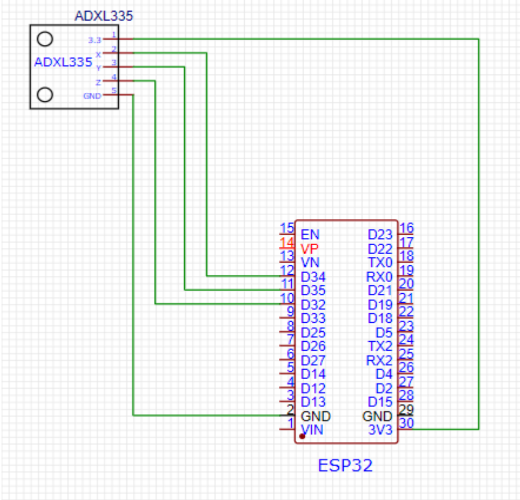


Figure: ESP32 interfacing ADXL Sensor

Proposed Device Architecture

Rather than other mouse, this wearable mouse is made up in form of rings which user wear in finger and then operates over the system from anywhere within an enclosed Bluetooth range. Device comprises of one microcontroller, three accelerometer sensor one of which output the raw gyroscope and accelerometer values is used for cursor position and other two is used to perform click operations on the sensors.

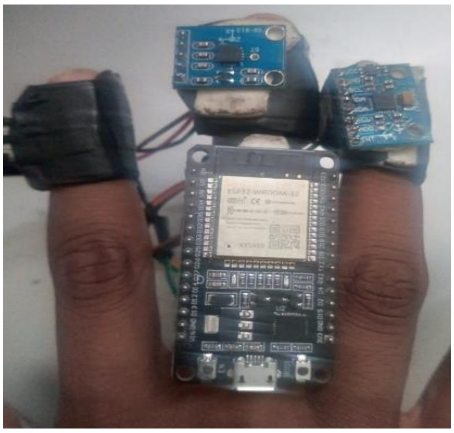


Figure: Prototype Mouse model

ISSN PRINT 2319-1775 Online 2320-7876, www.ijfans.org
Vol.11, Iss.10, Oct- 2022

A. Position of microcontroller

As the microcontroller is a central system controller which is controlling all the functionalities therefore its position should define the stability of the system and also comfort for its user. Therefore, it is placed on the right-side finger.

B. Position of cursor controlling sensor

Cursor movement of the system is controlled by MPU6050 sensor. Finger movement of left side is more flexible than right side and to balance the weight of the microcontroller this sensor is fixed at left side finger.

C. Position of another two sensor

Other two accelerometer are used to perform clicks on the sensor. It is fixed to place one sensor on one finger.

D. Position of Power Supply Battery

Small Lithium-ion batteries providing power supply to the microcontroller which connects to all the sensors for power supply. Power batteries are placed on right finger along with the charging port for the device.

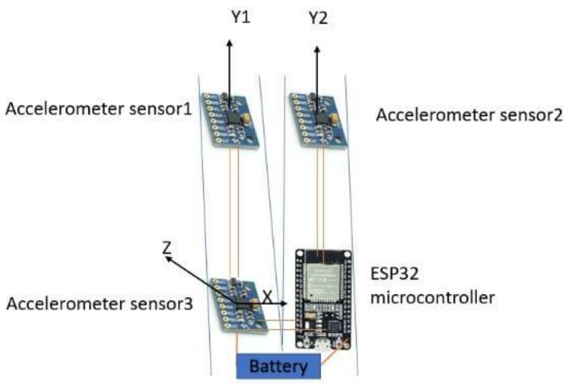


Figure: Position of All Sensors

Circuit Design

For making the device its circuit design is more important than anything to complete the interaction between sensor and microcontroller and all the connections should be correct to avoid any harm. In the article we are using ESP32 microcontroller development board of 30 pin configuration to interface it with multiple sensors.

Also MPU6050 sensor module is used in it for detecting the orientation of the sensor to control the cursor position with respect to its orientation. Meanwhile, MPU6050 needs an I2C communication bus to connect it with ESP32. One of the three I2C bus is used in ESP32 to connect it with MPU6050. GPIO pins 22 and 21 are used as SCL and SDA respectively and interfaced with SCL and SDA of MPU6050.For providing current supply to the sensor VCC pin of the sensor is connected to 3V3 (3.3V) pin of ESP32 and GND pin GND pin respectively [13].

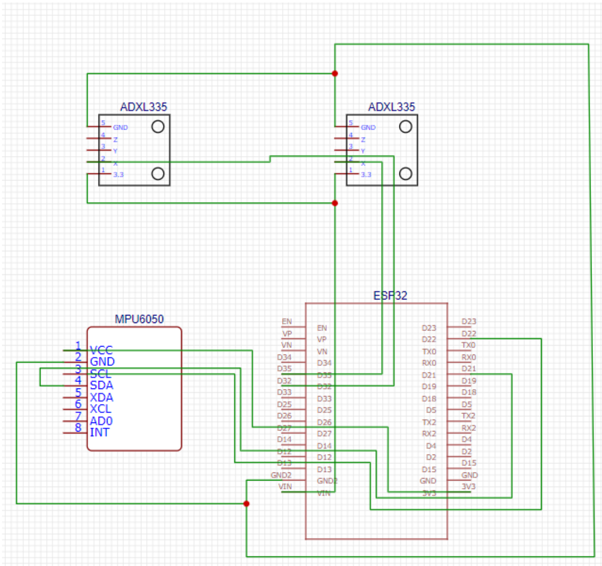
Another accelerometer sensor that is connected with ESP32 is ADXL335 which give analog reading of voltage proportional to the acceleration in the given direction.

For analog reading of each direction a separate ADC pin is required in ESP32. Because of the necessity of the ADC pins in ESP32 and to reduce the complexity of the circuit only single axis pins are connected with ESP32. GPIO pin 32 and 34 are connected with one axis out pin in both the sensors. To providing the power supply to both the sensors VCC pin of the sensors is connected with VIN (5V) and GND pin GND pin in ESP32 microcontroller [11].

Batteries are only connected with the microcontroller other sensors get power supply from the microcontroller.

A. Charging of the device

Device get the power supply from the Lithium-ion batteries connected to the microcontroller. Lithium-ion batteries are chargeable and can be charged using a USB portable charger.



Proposed Procedure

Sensors and microcontroller is major requirement for the device to run which can be easily available but the major component in working is the programmable code and availability of the modules within the system to successful run the code and manipulate all the changes in the system.

Arduino Language

Arduino Language code generally written in C language to fetch the data of the sensor. This code is written and then the microcontroller is booted by the code.

Python and its inbuilt-modules

Python is used as an interface language and major functionality changing language in the device working. Generally C language is used in connecting with kernel but due to high variety of inbuilt modules present in the python language this is used in the project for functionality management.

Modules of python used in the project

1. Pyautogui – For controlling the mouse
2. Pyserial - For connecting with the device
3. Tkinter - To build a Graphical Interface for the system

Working of Device

Getting raw values and transfer to the system

Device working is simple and easy to understand. As the device turns on programmable ESP32 microcontroller start fetching the raw values from the three accelerometers with the help of algorithm booted inside it. Algorithm is written in two different phases and in two different languages. One written in Arduino is booted within the microcontroller and another one written in python. Arduino language comprises the code that fetch the value from MPU6050 sensor using an inbuilt function motion6 of MPU6050 library of the Arduino and fetch the analog values from other two sensors. Arduino language also consists of the algorithm which filter the displacement of the sensor from the MPU6050. This data is transferred to the system using inbuilt Bluetooth functionality present within the microcontroller.

Receiving the raw values from the system

To receive the ESP32 data pyserial library of python is used which creates an object of connection which is setup over the COM port of windows10. After connecting the python program with serial portit start receiving the data from the system over an enclosed loop.Filtered value of one accelerometer is used for cursor control by manipulating it through moveRel() function of pyautogui library. This function changes the cursor position related to its current position. Unfiltered value from another two sensor is compared by each other and less value is used for respective click.

Conclusion

The development of an accelerometer-based mouse connected via ESP32 has significant implications for the future of human-computer interaction. This research paper has demonstrated the feasibility and effectiveness of using an accelerometer to control a computer mouse, as well as the benefits of using the ESP32 microcontroller to facilitate wireless communication.

One of the key advantages of this system is the potential to improve accessibility for individuals with physical disabilities. Traditional computer mice can be challenging for those with limited mobility or dexterity, whereas an accelerometer-based mouse provides a more intuitive and flexible input method. The wireless nature of the system also eliminates the need for a physical connection between the computer and the mouse, providing additional convenience and ease of use.

Another significant advantage of this system is the potential to reduce repetitive strain injuries. Traditional computer mice require frequent movement and clicking, which can lead to discomfort or pain in the hands and wrists over time. By using an accelerometer to control the mouse, users can reduce the frequency and intensity of physical movements, potentially reducing the risk of injury.

Furthermore, the ESP32 microcontroller provides a powerful and versatile platform for wireless communication. In addition to facilitating communication between the accelerometer and the computer, the ESP32 could be used for a wide range of other applications, such as data collection, IoT connectivity, or remote control. This versatility opens up a range of potential use cases for this system beyond just controlling a computer mouse.

One potential limitation of this system is the need for some degree of initial calibration to ensure accurate movement detection. However, this calibration process can be simple and straightforward, and once completed, the system operates smoothly and reliably.

Overall, the development of an accelerometer-based mouse connected via ESP32 represents an exciting advancement in the field of human-computer interaction. The system has the potential to improve accessibility, reduce the risk of repetitive strain injuries, and open up new possibilities for wireless communication and control. Further research could explore potential applications in other contexts, as well as refinements to the system design to further enhance its functionality and ease of use.

References

1. Meng Chen; Yifan Liu; John Chung Tam; Hoyin Chan; Xinyue Li; Chishing Chan.” Wireless AI-Powered IoT Sensors for Laboratory Mice Behavior Recognition” IEEE- June 2021.
2. Tomàs Pallejà,[1]and Jordi Palacín.”Optical Mouse Sensor for Eye Blink Detection and Pupil Tracking: Application in a Low-Cost Eye-Controlled Pointing Device”HINDWAI-2019
3. Thelma Androutsou 1,* , Spyridon Angelopoulos 2 , Evangelos Hristoforou 2 , George K. Matsopoulos 1 and Dimitrios D. Koutsouris 1- “A Multisensor System Embedded in a Computer Mouse for Occupational Stress Detection” MDPI 2023.
4. "Wireless Mouse Control System Based on Hand Gestures Recognition Using Machine Learning Techniques" by X. Liu, Y. Fang, and F. Wang. 2021.
5. “A Novel Wireless Mouse Design using Inertial Measurement Unit (IMU) Sensors and ZigBee Communication" by M. K. Alam, S. Nandi, and K. M. Iftekharuddin. 2021.
6. "A Smart Wearable Wireless Mouse Design with a Low-Power Microcontroller and Inertial Sensors" by T. Xie, J. Yang, H. Zhou, and W. Li. in 2020."A Wireless Mouse Design using Capacitive Sensing and Low-Power Microcontroller" by K. R. Remya and M. S. Anand. 2020.
7. "Design and Implementation of a Wireless Mouse using Accelerometer and Gyroscope Sensors" by R. K. Choudhary and P. D. Godse. 2019.
8. "Development of a Wireless Mouse using MEMS Accelerometer and Gyroscope Sensors" by P. Dutta and M. Dutta. 2019.
9. "Design and Development of a Wireless Mouse using Inertial Measurement Unit (IMU) Sensors and Bluetooth Communication" by M. S. Anand and P. R. Patil. 2018.
10. "A Wireless Mouse Design using Optical Sensors and Bluetooth Communication" by M. T. Islam, M. S. Islam, and M. N. H. Bhuiyan.

11. Mehendale, Ninad and Mehendale, Ninad, Object Detection using ESP 32 CAM (July 2, 2022).
12. Alexander Maier, Andrew Sharp, YuriyVagapov Comparative analysis and practical implementation of the ESP32 microcontroller module for the Internet of Things" Internet of things applications, IEEE 2017.
13. Punsara, K.K.T &Premachandra, H.H.R.C &Chanaka, A.W.A.D &Wijayawickrama, Raveen&Abhayasinghe, Nimsiri& de Silva, Rajitha. (2020). IoT Based Sign Language Recognition System. 162-167. 10.1109/ICAC51239.2020.9357267.
14. Al-dahan, Prof. Ziad &Bachache, Nasseer&Bachache, Nasseer. (2016). Design and Implementation of Fall Detection System Using MPU6050 Arduino. 180-187. 10.1007/978-3-319-39601-9_16.
15. Yan Li, Shen Mingxia, Yao Wen, Lu Mingzhou,, et al. Method of gesture recognition for lactating sows based on MPU6050 sensor [J], Chinese Journal of agricultural machinery,.2015,46 (5): 279-285.
16. Fu Zhongyun, Zhu Haixia,, and MPU6050. Research on the filtering algorithm based on inertial sensor [J], piezoelectric and acousto optic.2015,37 (5): 821-829.
17. ZhuoCongbin, Yang Longpin,, and. Design of fall detection and alarm system based on MPU6050 acceleration sensor [J],.2015,38 (4): 821-825.
18. V. Tripathi, A. Bansal, R. Gupta, "Development of Self-Stabilizing Platform Using MPU-6050 as IMU", in Advances in Signal Processing and Communication, Lecture Notes in Electrical Engineering, vol 526. Springer, Singapore.
19. Zhang Y, Li H, Shen S, Zhang G, Yang Y, Liu Z, Xie Q, Gao C, Zhang P, Zhao W. Investigation of Acoustic Injection on the MPU6050 Accelerometer. Sensors (Basel). 2019 Jul 12;19(14):3083. doi: 10.3390/s19143083. PMID: 31336934; PMCID: PMC6679069.
20. Yoga Priyana ,FolkesLaumal , Emir Husni "Development of Earthquake Early Warning System Using ADXL335 Accelerometer" InstitutTeknologi Bandung, Bandung 40132, Indonesia, 2018
21. Small, Low Power, 3-Axis ± 3g Accelerometer, ADXL335, Norwood, MA 02062-9106, USA : Analog Devices, Inc. All rights reserved, 2009.
22. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd Ed., India : Pearson Education, 2011.
23. Blaauw, D. & Sylvester, Dennis & Dutta, P. & Lee, Yoonmyung& Lee, I. & Bang, Suyoung& Kim, Y. & Kim, Gyouho&Pannuto, Pat &Kuo, Y.-S & Yoon, D. & Jung, W. & Foo, Z. & Chen, Y.-P & Oh, Sechang& Jeong, Seokhyeon& Choi, M.. (2014). IoT design space challenges: Circuits and systems. 1-2. 10.1109/VLSIT.2014.6894411.
24. S. K. Yoon, S. J. Kim, and U. K. Kwon, "A new circuit structure for near field wireless power transmission," in Circuits and Systems (ISCAS), 2012 IEEE International Symposium on. IEEE, 2012, pp. 982–985.