

Determination of Crosstalks using Mems

Mr. E Satheesh Kumar

Department of Electronics
and Communication
Engineering Annamacharya
Institute of Technology and
Sciences

Tirupati, India

esatheesh79@gmail.com

Shaik Khaja Nayab

Rasool

Department of Electronics
and Communication
Engineering
Annamacharya Institute of
Technology

and Sciences Tirupati, India

[khajanayabrasoolshaik750](mailto:khajanayabrasoolshaik750@gmail.com)

[@gmail.com](mailto:khajanayabrasoolshaik750@gmail.com)

Sanappireddy Karthik

Kumar Reddy

Department of Electronics and
Communication Engineering
Annamacharya Institute of
Technology and Sciences

Tirupati, India

sanappureddykarthik192@gmail.com

[il.com](mailto:sanappureddykarthik192@gmail.com)

Chinthagumpala Lohitha

Department of Electronics and
Communication Engineering
Annamacharya Institute of
Technology and Sciences

Tirupati, India

chinthagumpalalohitha@gmail.com

[com](mailto:chinthagumpalalohitha@gmail.com)

Syed Muhammad Ilyas

Department of Electronics and Communication Engineering
Annamacharya Institute of Technology and Sciences

Tirupati, India

syedilyas4403@gmail.com

Abstract: Micro-Electro-Mechanical Systems (MEMS) accelerometers are extensively employed in motion sensing and embedded monitoring applications. However, the performance of these sensors can be influenced by cross-axis interference, commonly referred to as cross-talk, which affects the accuracy of acceleration measurements. This paper presents the design and implementation of a real-time cross-talk detection and analysis system using a three-axis MEMS accelerometer interfaced with an Arduino Uno microcontroller. The system utilizes the MPU-6050 sensor to capture acceleration data along the X, Y, and Z axes. The collected signals are analyzed using cross-correlation techniques to evaluate the level of interference between orthogonal sensing axes and to identify unintended signal coupling. Such cross-axis effects can introduce significant errors in applications such as robotics, wearable devices, structural health monitoring, and inertial navigation systems. To enable remote data access and real-time

monitoring, a NodeMCU ESP8266 module is integrated to transmit sensor data to a cloud-based visualization platform. The acceleration data is continuously processed and displayed through graphical representations, allowing users to observe axis interactions and signal dependencies in real time. Experimental results demonstrate that the proposed system effectively identifies cross-axis interference while maintaining a low-cost hardware configuration with scalable IoT connectivity. The developed approach provides a practical framework for enhancing the reliability and accuracy of MEMS-based motion sensing systems.

Keywords— MEMS accelerometer, Cross-axis interference, Cross-talk detection, Motion sensing, Arduino Uno, MPU-6050 sensor, Internet of Things (IoT), NodeMCU ESP8266, Cross-correlation analysis, Real-time monitoring, Embedded systems, Wireless sensor networks.

I. INTRODUCTION

Motion sensing and multi-axis acceleration measurement have become fundamental requirements in many engineering applications such as robotics, wearable devices, structural health monitoring, and inertial navigation systems. The development of Micro-Electro-Mechanical Systems (MEMS) accelerometers has enabled compact, energy-efficient, and highly sensitive sensing devices capable of measuring acceleration along three orthogonal axes. These sensors allow accurate detection of motion in three-dimensional space and are widely integrated into embedded and IoT-based systems for real-time monitoring and control.

Despite their advantages, multi-axis MEMS accelerometers are often affected by cross-axis interference, commonly referred to as cross-talk. This phenomenon occurs when motion along one sensing axis generates unintended responses in the other axes. Such interference may arise due to mechanical coupling within the sensor structure, electronic noise in signal conditioning circuits, or imperfections in fabrication processes. As a result, cross-talk can lead to inaccurate motion estimation and reduced reliability in applications such as robotic navigation, human activity monitoring, and vibration analysis.

Previous studies have explored the design principles and performance characteristics of MEMS accelerometers. For example, Priya *et al.* presented an overview of MEMS accelerometer design, operating mechanisms, and application areas in sensing systems [1]. Yan *et al.* investigated cross-talk effects in multi-axis accelerometers and analyzed the impact of mechanical coupling on sensor output signals [2]. In addition, Kim and Park evaluated cross-axis sensitivity in three-axis MEMS sensors and highlighted the importance of accurate detection and compensation techniques for improving measurement reliability [6]. These works demonstrate the need for effective methods to detect and quantify cross-axis interference in MEMS-based sensing systems.

With the rapid expansion of the Internet of Things (IoT), sensor-based monitoring systems can now be integrated with wireless communication and cloud platforms to enable remote data access and real-time visualization. Bhattacharya and Ghosh proposed an IoT-based data acquisition system that supports continuous monitoring using MEMS sensors [3]. Similarly, Choi *et al.* developed a wireless sensing framework that allows sensor data to be transmitted to cloud platforms for graphical visualization and analysis [4]. Furthermore, Khan *et al.* demonstrated the use of NodeMCU ESP8266 for cloud-integrated monitoring systems, highlighting its effectiveness in wireless embedded applications [7].

Motivated by these advancements, this project proposes an IoT-enabled cross-talk detection and monitoring system for a three-axis MEMS accelerometer. The system integrates an MPU-6050 accelerometer sensor, an Arduino Uno microcontroller, and a NodeMCU ESP8266 module to acquire acceleration data along the X, Y, and Z axes. The collected data is analyzed to identify correlations between axis signals, enabling the detection of cross-axis interference in real time. Additionally, wireless communication through the ESP8266 module allows the measured data to be transmitted to a cloud-based dashboard for remote monitoring and graphical visualization.

The proposed framework provides a cost-effective and scalable platform for studying cross-talk effects in MEMS accelerometers while enabling real-time monitoring through IoT connectivity. Such a system can contribute to improved accuracy and reliability in motion sensing applications across various engineering domains.

II LITERATURE SURVEY

Micro-Electro-Mechanical Systems (MEMS) accelerometers have become an essential component in modern sensing technologies due to their small size, low power consumption, and high sensitivity. These sensors are widely used in applications such as robotics, structural monitoring, automotive systems, and wearable electronics. However, multi-axis accelerometers often suffer from cross-axis

sensitivity, which can affect the accuracy of motion detection.

Priya *et al.* [1] provided a detailed overview of MEMS accelerometer design principles and their application in various sensing systems. Their work highlighted the advantages of MEMS technology in enabling compact and efficient sensing devices capable of measuring acceleration along multiple axes. The study also emphasized the importance of improving measurement accuracy in multi-axis accelerometer systems.

Yan *et al.* [2] conducted an analysis of cross-talk effects in multi-axis MEMS accelerometers. Their research demonstrated that mechanical coupling between sensing elements can cause unwanted responses in adjacent axes, which may lead to errors in motion detection. The authors suggested that effective analysis methods are necessary to identify and minimize cross-axis interference.

Bhattacharya and Ghosh [3] introduced an IoT-based real-time data acquisition system using MEMS sensors. Their work showed how integrating sensor systems with embedded controllers and network connectivity can enable continuous monitoring and remote access to sensor data. Such systems improve data accessibility and support real-time analysis in monitoring applications.

Choi *et al.* [4] proposed a wireless data acquisition system combined with cloud-based visualization for MEMS accelerometers. Their system enabled real-time transmission and graphical display of sensor measurements through cloud platforms, demonstrating the advantages of wireless monitoring in sensor-based applications.

Bogue [5] discussed the design considerations and industrial applications of MEMS sensors for motion measurement. The study highlighted challenges associated with sensor reliability and performance, particularly when dealing with multi-axis sensing environments.

Kim and Park [6] experimentally evaluated cross-axis sensitivity in three-axis MEMS accelerometers. Their findings showed that cross-

axis interference can significantly affect sensor output accuracy, and therefore reliable detection and compensation techniques are required for improved measurement performance.

Khan *et al.* [7] presented a cloud-integrated monitoring system using the NodeMCU ESP8266 platform. Their research demonstrated that low-cost wireless modules can effectively transmit sensor data to cloud servers, enabling scalable and remote monitoring solutions for IoT-based sensing applications.

III.EXISTING SYSTEM

In typical motion sensing applications, MEMS-based accelerometers are employed to capture acceleration components along three orthogonal axes (X, Y, and Z). Devices such as the ADXL335 and MPU-6050 are commonly preferred due to their affordability, compact design, and compatibility with embedded systems. Conventional implementations generally treat each axis as an independent sensing channel, assuming that the interaction between axes is negligible. To improve signal quality, standard calibration procedures, including bias correction, are applied, and noise reduction is achieved using basic filtering techniques such as low-pass and moving average filters.

However, this simplified assumption often ignores the presence of inter-axis interference, commonly referred to as cross-talk. In practice, acceleration along one axis can unintentionally influence the outputs of the other axes, introducing errors in the measured data. This phenomenon becomes particularly critical in applications involving dynamic motion, such as robotic navigation, vibration analysis, and structural monitoring, where even small inaccuracies can significantly affect system performance.

Current approaches to addressing cross-talk are largely limited to offline data analysis or mathematical modeling, which are not well-suited for real-time implementation. Additionally, many existing systems rely on wired communication for data acquisition and lack interactive visualization tools, thereby restricting their usage to laboratory settings. These limitations reduce their effectiveness

in modern applications that demand remote monitoring, scalability, and real-time insights.

To overcome these challenges, there is a growing need for a unified system that not only detects and quantifies cross-talk in real time but also integrates wireless communication and cloud-based visualization. Such an approach would enhance measurement accuracy and enable more reliable deployment of multi-axis sensing systems in practical, real-world environments.

IV. PROPOSED SYSTEM

The proposed system overcomes the limitations of conventional approaches by integrating a 3-axis MEMS accelerometer with an Arduino Uno for real-time acquisition of multi-axis acceleration data, along with a NodeMCU ESP8266 module for wireless communication. The accelerometer continuously captures motion data along the X, Y, and Z axes, which is then processed to evaluate cross-talk effects between the axes. In particular, cross-correlation analysis is performed to quantify the degree of interdependence between signals, especially between the X and Y axes.

The Arduino Uno is responsible for sensor interfacing, initial signal conditioning, and real-time computation of correlation metrics. The processed data is then transmitted via the NodeMCU ESP8266 to a cloud-based platform, where it is presented through dynamic graphs and tabular formats. This enables users to visualize inter-axis relationships and monitor system behavior in real time from remote locations.

By identifying and quantifying cross-talk, the system significantly improves the accuracy of motion sensing. In addition, the integration of wireless communication and cloud visualization reduces reliance on bulky and expensive laboratory setups, making the solution more accessible and scalable. Overall, the proposed approach combines embedded processing, signal analysis, and IoT-based monitoring to deliver a cost-effective and practical solution for precise multi-axis motion measurement. This makes it well-suited for applications such as robotics, wearable systems, and structural health

monitoring, where real-time accuracy and reliability are essential.

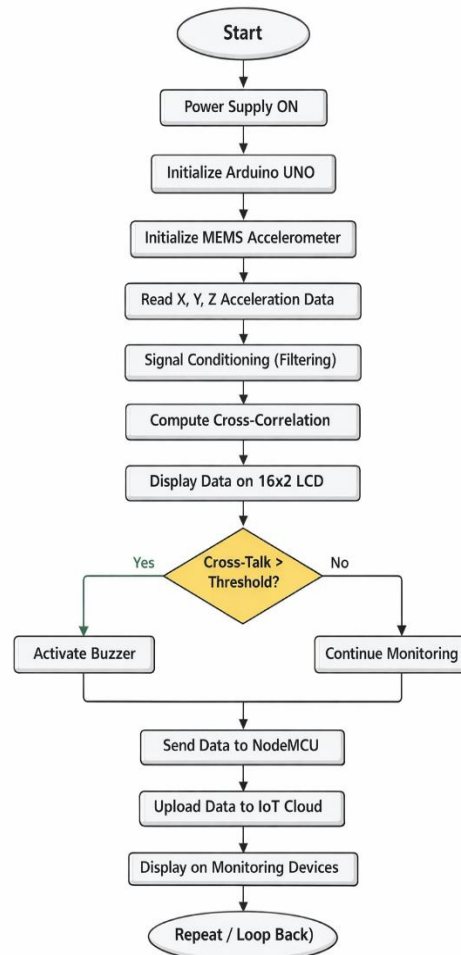


Fig 1: Flow Diagram of Proposed System

The developed motion monitoring system is intended to provide continuous and intelligent detection of movement with high efficiency and reliability. The system becomes operational once power is supplied to the Arduino Uno along with all interfaced components. During the initialization phase, the Arduino configures the MEMS accelerometer, LCD display, buzzer, and NodeMCU (ESP8266) module, ensuring that each unit is ready for synchronized operation.

The MEMS accelerometer functions as the primary sensing element, constantly capturing motion and orientation data across the three axes (X, Y, and Z). This data is transmitted to the Arduino Uno, which serves as the central controller. The controller processes the incoming signals in real time and evaluates them against predefined threshold values to determine whether the detected motion is within

acceptable limits.

When the measured motion remains within the specified range, the system indicates normal operation by updating the status on the LCD display. In contrast, if the sensed values exceed the defined thresholds, the system classifies the condition as abnormal. In such situations, the Arduino promptly activates the buzzer to generate an alert, thereby notifying nearby users of unusual activity. At the same time, a warning message is displayed on the LCD to provide a clear visual indication of the event.

To extend its functionality beyond local monitoring, the system incorporates wireless communication for remote access. The processed motion data is forwarded from the Arduino to the NodeMCU (ESP8266), which connects to a Wi-Fi network and transmits the information to a cloud-based platform. This enables real-time monitoring and visualization of system data from remote devices such as smartphones and computers.

In summary, the system integrates sensing, processing, alerting, and IoT-based communication into a unified framework, offering a practical and scalable solution for accurate motion detection and remote monitoring in real-world applications.

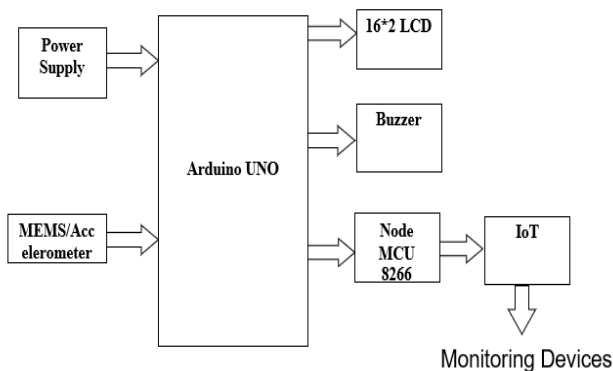


Fig 2: Block Diagram of Proposed System

V. WORKING METHODOLOGY

Arduino:

The Arduino Uno, built around the ATmega328P microcontroller, functions as the

central processing unit of the system. It establishes communication with the MPU-6050 sensor using the I2C protocol to obtain real-time acceleration data. The acquired signals are further processed within the microcontroller, where necessary signal conditioning and cross-correlation analysis are performed to evaluate inter-axis relationships. In addition to data processing, the Arduino also coordinates overall system operations, ensuring synchronized functioning of all connected modules. Its capability to handle continuous data acquisition and perform real-time computations makes it well-suited for embedded motion monitoring applications.

Specification:

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Clock Speed: 16 MHz
- Digital I/O Pins: 14 (6 PWM)
- Analog Input Pins: 6
- Flash Memory: 32 KB
- Communication: UART, I2C, SPI

MPU-6050:

The MPU-6050 functions as the primary sensing component responsible for motion detection within the system. It captures acceleration data along the three orthogonal axes-X, Y, and Z, enabling accurate monitoring of movement and orientation. The collected measurements are utilized to identify motion patterns and evaluate cross-talk effects between the axes through correlation-based analysis. Owing to its high sensitivity and stable performance, the sensor provides reliable data, making it suitable for precise motion analysis and real-time signal processing applications.

Specifications:

- Accelerometer Range: $\pm 2g, \pm 4g, \pm 8g, \pm 16g$
- Gyroscope Range: ± 250 to ± 2000 $^{\circ}/\text{sec}$
- Communication Protocol: I2C
- Operating Voltage: 3.3V – 5V
- Built-in 16-bit ADC
- Digital Motion Processor (DMP)

NodeMCU ESP8266 (Wi-Fi Module):

The NodeMCU ESP8266 module provides wireless communication capability for the system. It is responsible for transmitting the processed acceleration data and computed cross-correlation values to a cloud-based platform for remote access. Through its built-in

Wi-Fi functionality, the module enables seamless real-time data transfer, allowing users to monitor system performance and visualize results from any location. Its low-cost and efficient design makes it well-suited for IoT-based applications requiring continuous connectivity and remote supervision.

Specifications:

- Microcontroller: ESP8266
- Operating Voltage: 3.3V
- Wi-Fi Standard: 802.11 b/g/n
- Clock Speed: 80 MHz (up to 160 MHz)
- Flash Memory: 4 MB (typical)
- GPIO Pins: 11

16×2 LCD Display (Local Monitoring):

The 16×2 LCD module is utilized to present real-time acceleration readings along with the corresponding cross-talk status. It offers instant visual feedback, allowing users to quickly assess system conditions during operation. Due to its simplicity, clarity, and ease of integration, the display serves as an effective interface for local, on-site monitoring.

Specifications:

- Display Type: 16 characters × 2 lines
- Operating Voltage: 5V
- Interface: 4-bit / 8-bit parallel
- Backlight: LED

Buzzer (Alert System):

The buzzer serves as an audible alert mechanism within the system. It is activated whenever the cross-talk level exceeds a predefined threshold, indicating abnormal interaction between sensing axes. This feature enables quick identification of irregular conditions and ensures that users are immediately notified of potential issues during operation.
awareness.

Specifications:

- Operating Voltage: 3V–5V
- Type: Active or Passive
- Low power consumption

Power Supply Unit:

The power supply unit ensures a consistent and regulated voltage is delivered to all system modules, supporting stable and reliable operation throughout the system.

Specifications:

- Input: 7–12V (external adapter)

- Output: 5V regulated (Arduino)
- 3.3V regulated (ESP8266)

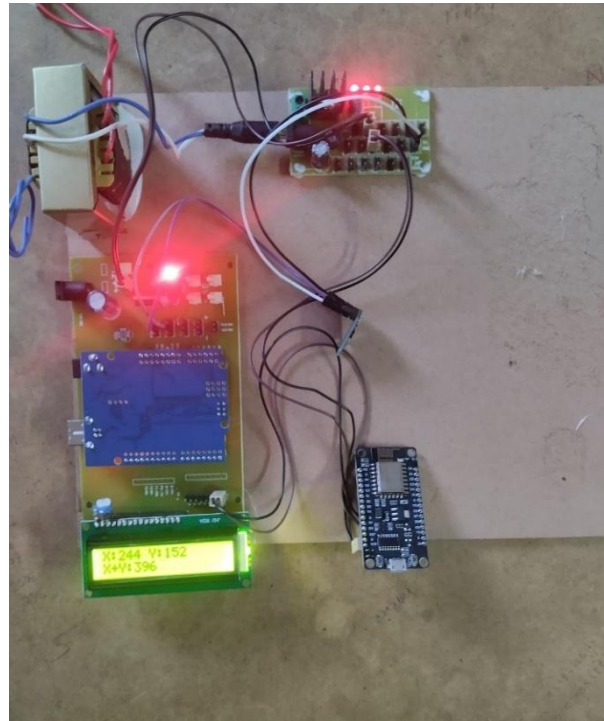


Fig 3 : Representation of Proposed System

The system operates using a regulated power supply that drives the Arduino Uno, which serves as the main control unit. A motion sensor (accelerometer) is used to capture vibrations or movement and forward the corresponding data to the controller for real-time analysis. Based on the sensed inputs, the Arduino evaluates the system condition and determines the required response. When abnormal activity is detected, the system immediately displays relevant information on an LCD module and triggers a buzzer to provide an audible warning to nearby users.

At the same time, the processed data is transmitted to a NodeMCU module equipped with Wi-Fi functionality. The NodeMCU forwards this information to a cloud-based platform, enabling remote access and data storage. This arrangement allows users to monitor system status in real time through smartphones, laptops, or other internet-connected devices, ensuring continuous supervision without the need for physical presence.

VI. RESULT

The proposed cross-talk detection and alert system

Microelectromechanical Systems, vol. 27, no. 6, pp. 1105–1115, Dec. 2018.

[3] S. Bhattacharya and R. Ghosh, “IoT-Based Real-Time Data Acquisition and Monitoring Using MEMS Sensors,” *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6432–6440, Aug. 2019.

[4] H. Choi, J. Lee, and S. Kim, “Wireless Data Acquisition and Cloud-Based Visualization for MEMS Accelerometers,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 9, pp. 7043–7052, Sep. 2019.

[5] R. G. Bogue, *The Design and Application of MEMS Sensors for Motion Measurement*, Measurement Science and Technology, vol. 30, no. 5, 2019.

[6] H. S. Kim and J. S. Park, “Evaluation of Cross-Axis Sensitivity in 3-Axis MEMS Accelerometers,” *Sensors and Actuators A: Physical*, vol. 295, pp. 200–208, Feb. 2019.

[7] M. S. Khan, A. Rehman, and M. Usman, “Cloud-Integrated Wireless Monitoring System Using NodeMCU ESP8266,” *IEEE Access*, vol. 8, pp. 14215–14223, Jan. 2020.

