Earthquake Alert: Development of an ESP32-Based Early Warning System Prototype using MPU6050 and SW-420 Sensor

Handoko Rusiana Iskandar1*), Reni Ayu Purwaningsih2*), Dimas Alfahrezi3), Ahmad Daelami4), Dede Furqon Nurjaman5)

1-5)Electrical Engineering Department
Universitas Jenderal Achmad Yani
Jalan Terusan Jend. Sudirman PO.BOX 148 Cimahi 40531

*)Correspondence: handoko.rusiana@lecture.unjani.ac.id, ayurenia572@gmail.com

Abstrak

Kabupaten Bandung Barat, Provinsi Jawa Barat, merupakan salah satu wilayah dengan potensi risiko gempa bumi akibat keberadaan Sesar Lembang dan kondisi geologis regional yang aktif. Kejadian gempa bumi yang berlangsung secara tiba-tiba dapat menimbulkan dampak signifikan terhadap keselamatan manusia, kerusakan infrastruktur, serta terganggunya aktivitas di lingkungan kampus dan bangunan umum. Keterbatasan sistem peringatan dini gempa bumi yang bersifat lokal, *real-time*, dan mudah diakses mendorong perlunya pengembangan teknologi mitigasi berbasis *Internet of Things* (IoT). Penelitian ini bertujuan untuk merancang dan mengimplementasikan *prototype* sistem peringatan dini gempa bumi berbasis mikrokontroler ESP32 dengan integrasi notifikasi Telegram. Sistem yang dikembangkan menggunakan sensor MPU6050 untuk mengukur percepatan getaran pada sumbu X, Y, dan Z, serta sensor SW-420 sebagai pendeteksi awal getaran. Data percepatan diproses menggunakan metode Root Mean Square (RMS) dan rasio *Short Time Average/Long Time Average* (STA/LTA) untuk mengklasifikasikan kondisi getaran ke dalam empat tingkat peringatan, yaitu aman, siaga, waspada, dan bahaya. Hasil klasifikasi ditampilkan melalui indikator lokal dan dikirimkan secara real-time ke Node-RED menggunakan protokol MQTT, kemudian diteruskan sebagai notifikasi Telegram kepada pengguna. Hasil pengujian menunjukkan bahwa sistem mampu mendeteksi getaran dengan tingkat keberhasilan sekitar 95%, akurasi klasifikasi sebesar 92%. Dengan demikian, sistem yang dirancang ini dapat dikembangkan dan berpotensi diterapkan sebagai solusi peringatan dini gempa bumi yang murah, responsif, dan mudah diimplementasikan pada wilayah rawan gempa.

Kata kunci : *ESP-32,* gempa bumi, *internet of things*, MPU6050, MQTT, *SW-420.*

Abstract

West Bandung Regency, West Java Province, is one of the areas with a significant earthquake risk due to the activity of the Lembang Fault and regional geological conditions. Earthquakes that occur suddenly may cause serious impacts on human safety, infrastructure, and activities in campus environments and public buildings. The limited availability of local, real-time, and easily accessible earthquake early warning systems highlights the need for an effective mitigation technology based on the Internet of Things (IoT). This study aims to design and implement an ESP32-based earthquake early warning system prototype integrated with Telegram notifications. The proposed system employs an MPU6050 sensor to measure vibration acceleration along the X, Y, and Z axes, while an SW-420 sensor is used as an initial vibration detector. The acceleration data are processed using the Root Mean Square (RMS) method and the Short Time Average/Long Time Average (STA/LTA) ratio to classify vibration conditions into four warning levels: safe, alert, warning, and danger. The classification results are displayed through local indicators and transmitted in real time to Node-RED using the MQTT (Message Queuing Telemetry Transport) protocol, then forwarded to users through Telegram notifications. The experimental results show that the system is capable of detecting vibration with a success rate of approximately 95%, a classification accuracy of 92%. Thus, this system can be further developed and has the potential to be implemented as a cost-effective, responsive, and easy-to-implement earthquake early warning solution in earthquake-prone areas.

Keywords *: earthquake, ESP-32, Internet of Things, MPU6050, MQTT, SW-420.*

I. INTRODUCTION

Indonesia's position within the Pacific Ring of Fire exposes the region to persistent seismic hazards, with West Bandung Regency (KBB) identified as a high-risk zone due to the proximal Lembang Fault and regional tectonic stress accumulation[1]. Conventional government-operated Early Warning Systems (EWS) rely on centralized data processing and top-down alert dissemination, which introduces latency, coverage gaps, and dependency on institutional infrastructure during critical events[2]. To overcome these systemic bottlenecks, we develop an edge-based IoT early warning architecture that performs local seismic signal acquisition, real-time feature extraction, and autonomous alert triggering. Integrated with low-power microcontrollers and wireless mesh networking, the system enables rapid, community-level warning dissemination independent of central server availability, thereby improving response time and operational resilience in seismically active regions[3], [4].

Info Makalah:
Dikirim : 05-25-2026;
Revisi 1 : 06-25-2026;
Revisi 2 : -
Diterima : 06-29-2026.

Penulis Korespondensi:
Telp : +62811125551
e-mail : handoko.rusiana@lecture.unjani.ac.id

Recent studies have explored IoT-enabled earthquake early warning (EEW) systems, primarily leveraging low-cost microcontrollers and commercial vibration sensors. For instance,[2] developed the SIGEMPA platform using ESP32 modules to transmit seismic status via IoT cloud services. Similarly, Pratiwi et al. [5] integrated SW-420 vibration sensors with Telegram-based real-time notifications, while Effendi [6] implemented an Arduino-driven detection node with mobile application monitoring. Siswanto et al. [7] extended this architecture to multi-hazard scenarios by deploying ESP8266-based nodes for simultaneous earthquake and tsunami alerts. Although these prototypes demonstrate the feasibility of distributed sensing, they predominantly rely on static threshold triggering without adaptive signal processing, exhibit limited validation under real-field seismic conditions, and lack standardized latency/accuracy benchmarking. Furthermore, most implementations depend on continuous cloud connectivity, introducing single-point vulnerabilities during network congestion or infrastructure disruption[3], [8]. These limitations underscore the need for an edge-computing architecture capable of autonomous feature extraction, decentralized alert routing, and robust operation under constrained communication environments.

Despite demonstrating the feasibility of IoT-enabled microcontroller platforms for seismic monitoring, existing implementations exhibit several critical limitations. First, signal processing predominantly relies on static amplitude thresholding, which lacks robust quantitative filtering and is highly susceptible to environmental noise and non-seismic vibration artifacts. Second, most architectures employ single-sensor configurations, which compromise detection reliability and increase false-positive rates under complex ground-motion conditions. Third, alert dissemination mechanisms are frequently coupled with third-party cloud platforms (e.g., Blynk, web-based dashboards)[9], introducing dependency on continuous internet connectivity, increased end-to-end latency, and reduced operational autonomy during network outages. These constraints collectively hinder the deployment of resilient, low-latency early warning systems capable of functioning reliably in resource-constrained and communication-disrupted environments[10], [11]. To the best of our knowledge, no prior study has implemented a concurrent RMS and STA/LTA signal processing pipeline on a resource-constrained ESP32[12] microcontroller for real-time seismic intensity classification, coupled with an autonomous Telegram-based alerting mechanism via MQTT[13]. This edge-centric architecture performs on-device feature extraction and threshold evaluation, while leveraging lightweight MQTT messaging to ensure rapid, reliable notification delivery independent of centralized cloud services.

The proposed integration is expected to enhance vibration discrimination accuracy while maintaining high communication efficiency, thereby addressing critical latency and reliability requirements in distributed early warning systems. The ESP32-based embedded earthquake early warning (EEW) prototype that integrates MPU6050 and SW-420 sensors for multi-parameter vibration acquisition. The system implements a concurrent Root Mean Square (RMS) and Short-Time Average/Long-Time Average (STA/LTA) signal processing pipeline to classify seismic activity into four intensity tiers: safe, standby, alert, and danger. Real-time notifications are disseminated via an MQTT–Telegram gateway to ensure rapid alert propagation. The core contributions of this work are twofold: (i) an adaptive, computationally efficient signal discrimination framework optimized for resource-constrained microcontrollers, and (ii) a lightweight publish–subscribe communication architecture that minimizes end-to-end latency and enhances reliability for localized early warning deployment.

Motivated by these requirements, this study proposes and implements an integrated hardware–software architecture for an ESP32-based earthquake early warning prototype. The system establishes an end-to-end pipeline encompassing vibration data acquisition, embedded signal processing, and real-time alert dissemination via a Telegram messaging interface over standard IP networks. This cohesive edge-centric framework bridges sensing, computation, and communication layers to enable rapid, reliable seismic monitoring under resource-constrained conditions.

II. METHODS

The research methodology encompasses the systematic design and implementation of an ESP32-based embedded seismic monitoring system. The development workflow proceeds through three integrated phases: (i) hardware architecture design, including ESP32 microcontroller configuration and vibration sensor interfacing; (ii) firmware development and MQTT-based telemetry implementation for real-time data transmission; and (iii) backend orchestration using Node-RED for signal processing, dashboard visualization, and Telegram push-notification routing for early warning dissemination. A comprehensive research flowchart is provided to delineate the sequential development stages, data pipeline, and system integration logic. See Fig. 1.

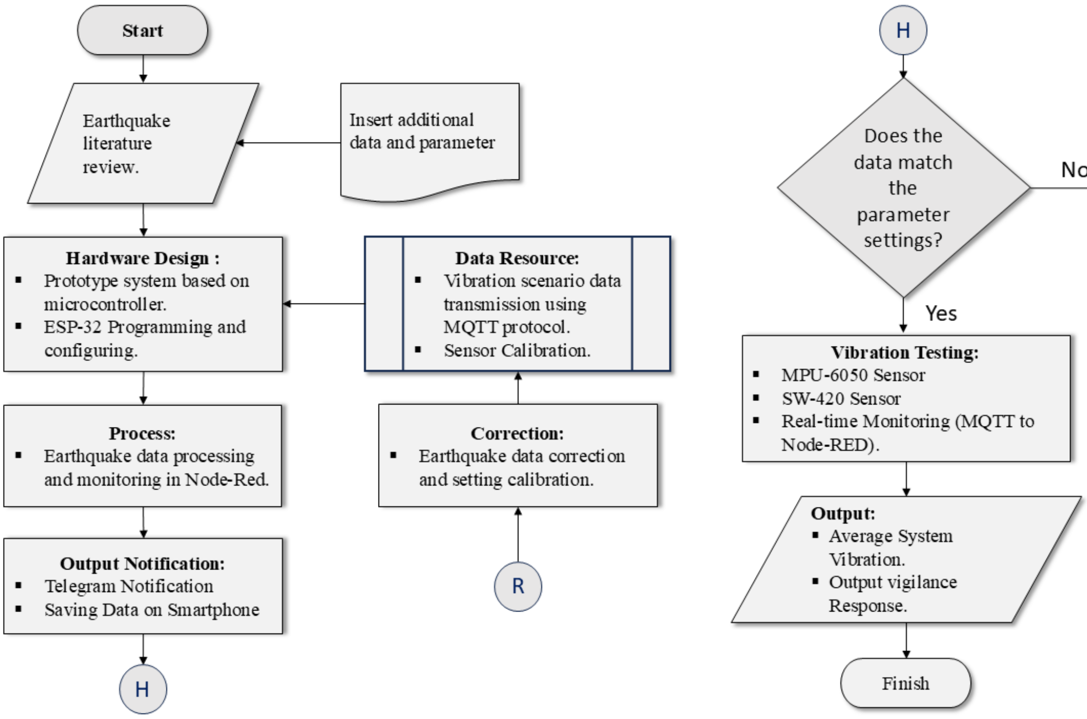


Figure 1. Research Methodology

Fig. 1 delineates the systematic development workflow for the ESP32-based seismic vibration detection system. The methodology initiates with a foundational literature review, followed by hardware architecture design, firmware development, and microcontroller configuration. Real-time vibration telemetry is transmitted via the MQTT protocol for backend signal processing and dashboard monitoring within a Node-RED orchestration environment. This data pipeline is subsequently integrated with the Telegram messaging API to enable automated early warning dissemination. The final phase comprises comprehensive experimental validation and quantitative performance benchmarking to assess the system's detection accuracy, end-to-end latency, and operational reliability under representative field conditions.

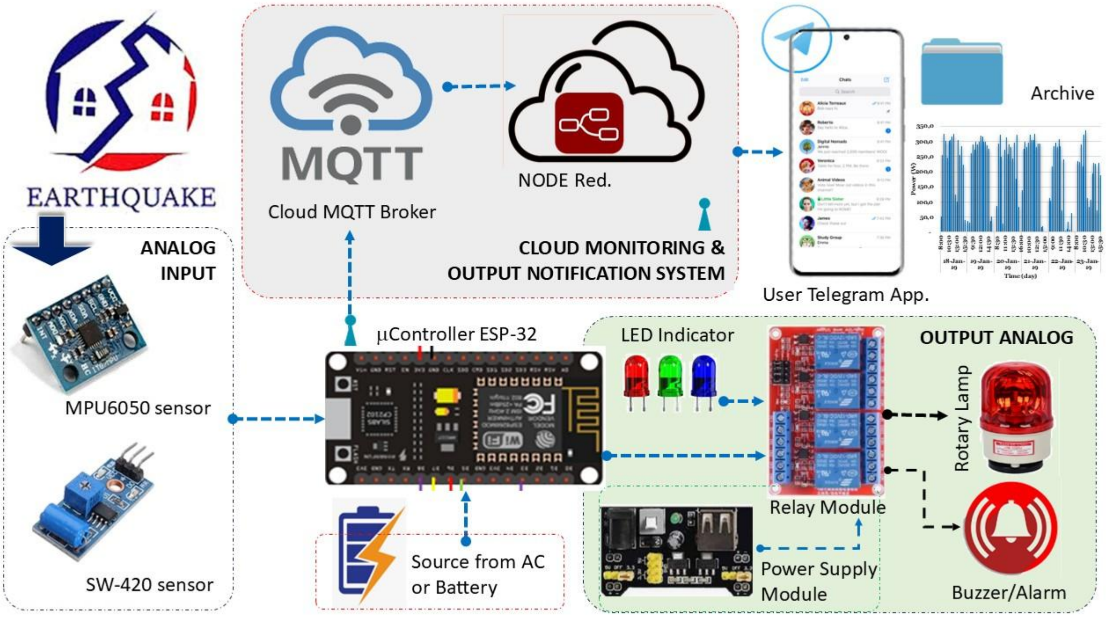


Figure 2. Block Diagram of IoT Prototype System

A. System Design

Fig. 2 illustrates the architectural framework of the ESP32-based earthquake early warning (EEW) prototype, integrated with Telegram-based alert dissemination via the MQTT protocol[14]. The system acquires ground-motion data through triaxial acceleration and vibration sensors, followed by real-time embedded signal processing to enable autonomous early-warning decision-making. By leveraging the ESP32's low-power computational capabilities for edge analytics [15], and MQTT's lightweight publish–subscribe messaging for reliable telemetry [16], the architecture ensures rapid detection-to-alert latency while maintaining operational efficiency under resource-constrained conditions.

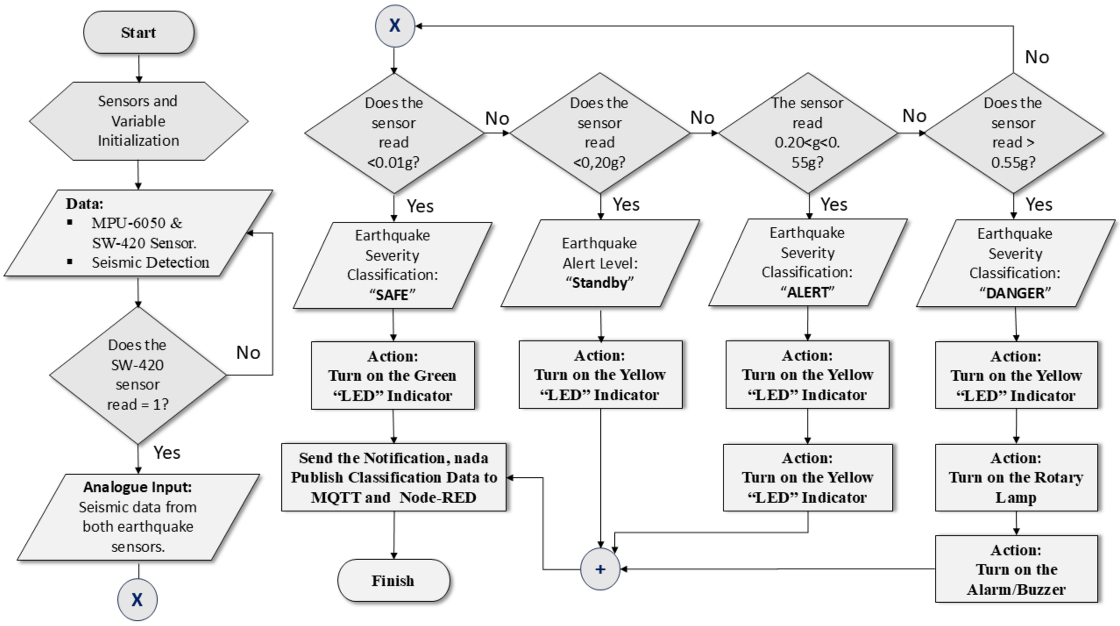


Figure 3. Flowchart System

Fig. 3 illustrates the operational workflow of the proposed earthquake early warning system. The sequence encompasses sensor initialization, continuous vibration acquisition, real-time acceleration data processing, intensity classification, and automated alert dissemination. This structured pipeline ensures deterministic state transitions and facilitates systematic validation, implementation tracing, and performance benchmarking of the embedded monitoring architecture [17],[18].

As illustrated in Fig. 3, the operational workflow of the proposed earthquake early warning system initiates with microcontroller initialization and calibration of the SW-420 vibration sensor and MPU6050 triaxial accelerometer. The system continuously acquires sensor data to monitor ground motion. When the SW-420 output exceeds a predefined threshold, the MPU6050 acceleration magnitude is processed to classify seismic intensity into four tiers: safe (<0.10 g), indicated by a green LED; standby (0.10–0.20 g), indicated by a yellow LED; warning (0.20–0.65 g), activated with a yellow LED and rotary beacon; and danger (>0.65 g), triggering a red LED, rotary beacon, and acoustic siren. Following classification, telemetry data and alert status are transmitted via MQTT to a Node-RED dashboard for real-time visualization, after which the system resumes continuous monitoring. This deterministic control loop ensures rapid detection-to-alert latency while maintaining operational stability under field conditions.

B. *Hardware Specification*

This study develops an ESP32-based seismic monitoring system by introducing a detection architecture that integrates the SW-420 vibration detector with the MPU6050 triaxial accelerometer. The proposed prototype uses a two-stage detection mechanism for experimental validation of ground motion, with real-time telemetry transmitted via MQTT to a Node-RED backend for signal processing, dashboard visualization, and automatic alert dissemination. The board exposes most of the microcontroller's I/O pins via two-row connectors on both sides, facilitating direct interfacing with external sensors, actuators, and communication modules[19]. Its compact form factor and fully accessible pinout make it well-suited for rapid prototyping and embedded system deployment in research-grade applications. For further component specifications, see Tables 1–3.

Tabel 1 Technical specifications of the ESP-32 microcontroller[20]

No.	Feature	Information Details
1	Microcontroller	ESP32 with a dual-core 32-bit Xtensa® LX6 processor
2	CPU Freq. and Flash Memory	Up to 240 MHz/4 MB
3	Connectivity	Wi-Fi 802.11 b/g/n, Bluetooth v4.2 BR/EDR and BLE
4	I/O Interface	GPIO, ADC, DAC, SPI, I2C, I2S, PWM, UART
5	Communication Port	USB to UART (using a CP2102 bridge chip or eq.)
6	Power Source	Via the Micro-USB port, the 5V pin, or the 3.3V pin
7	Buttons	▪ The EN (Reset) buttons; ▪ BOOT (Download) buttons.

Tabel 2 Technical specifications of the SW-420 Vibration Sensor Module[21]

No.	Specification	Information Details
1	Operating Voltage	Op. (3.3~5) VDC
2	Default Output	Normally Close (NC)
3	Indicator	On-board signal indicator LED
4	Sensitivity Adjustment	On-board LM393 Op-Amp
5	Module Dimension	32mm x 14mm

Tabel 3 Technical specifications of the MPU6050 Inertial Sensor Module[22]

No.	Specification	Information Details
1	Operating Voltage	▪ 2.375V – 3.46V (Chip) ▪ 3V – 5V (Module with LDO)
2	Logic Level	3.3V (Breakout boards like GY-521 include a 3.3V regulator)
3	ADC Resolution	16-bit for both Accelerometer and Gyroscope
4	Communication	I2C protocol (up to 400 kHz Fast Mode)
5	Current Consumption	▪ ~3.9 mA (Normal operation); ▪ 10 µA (Sleep mode)
6	Temp. Sensor	▪ Integrated; ±1°C ▪ accuracy; range –40°C to +85°C

C. System Circuit Diagram

Fig. 4 presents the circuit schematic of the proposed ESP32-based earthquake early warning (EEW) prototype. The architecture integrates an SW-420 vibration sensor for initial seismic trigger detection and an MPU6050 MEMS accelerometer for precise triaxial acceleration and tilt measurement. All sensor data are acquired and processed by the ESP32 microcontroller, which orchestrates local signal classification and drives the associated output actuators. To ensure electrical isolation and operational stability, the system employs a dual-supply power architecture: a regulated 5 V rail powers the control logic and sensing circuitry, while a dedicated 24 V source supplies high-current alert devices. This segregation mitigates ground bounce and electromagnetic interference, thereby enhancing system reliability under dynamic load conditions.

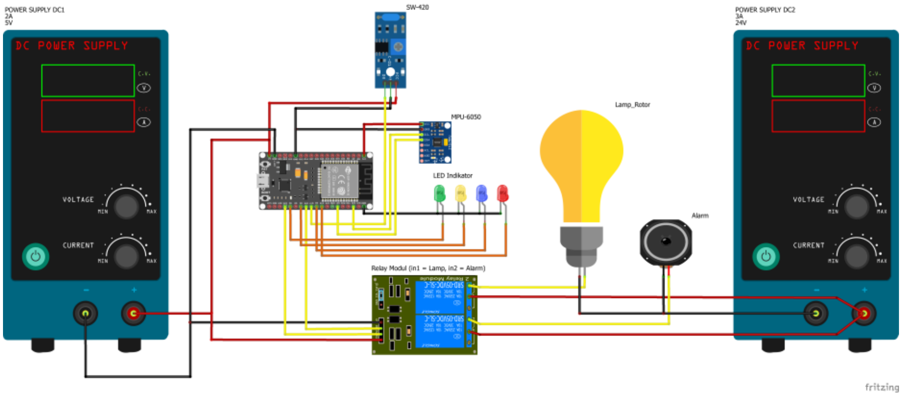


Figure 4. Schematic Diagram of the Proposed ESP32-based Earthquake Early Warning Prototype

As illustrated in Fig. 4, vibration and acceleration data acquired from the sensors are processed by the ESP32 microcontroller to classify the seismic state against predefined thresholds. The classification output drives status-indicating LEDs and triggers a relay module that actuates external warning lights and acoustic alarms. This relay interface provides galvanic isolation between the low-voltage control logic and high-current load circuits, effectively mitigating electromagnetic interference and protecting sensitive components from overvoltage transients. Consequently, the hardware configuration ensures deterministic alert actuation and enhances operational reliability for seismic early warning applications.

D. System Testing and Programming

Experimental validation was conducted to evaluate the performance of the proposed ESP32-based earthquake early warning (EEW) prototype in detecting ground vibrations and generating real-time alerts. The assembled prototype was operated under controlled laboratory conditions, with mechanical vibration simulations serving as controlled test stimuli. The procedure commenced with the initialization of the MPU6050 triaxial accelerometer and SW-420 vibration sensor, followed by I2C bus configuration. The ESP32 acquired triaxial acceleration data (x, y, z axes) and discrete vibration signals at a fixed sampling rate of 1 Hz. The acquired vectors were fused to compute the resultant ground acceleration, which was subsequently processed using Root Mean Square (RMS) amplitude estimation and the Short-Time Average/Long-Time Average (STA/LTA) triggering algorithm to quantify seismic intensity and enable adaptive classification. The computed STA/LTA ratio is continuously evaluated against predefined threshold levels to classify seismic intensity into four operational states: safe, standby, warning, and danger. Upon threshold exceedance, the system triggers a local acoustic alarm while concurrently transmitting telemetry data and push notifications via the MQTT protocol to a Node-RED backend and Telegram messaging interface. Quantitative performance analysis was conducted by correlating experimental outputs with threshold benchmarks to evaluate detection accuracy, system stability, and end-to-end response latency. The magnitude scale is a measure of an earthquake's strength that describes the amount of seismic energy released by the earthquake's source and is derived from seismograph readings. In other words, the larger the earthquake, the higher its magnitude[23]. To facilitate the conversion between the actual earthquake magnitude and the sensor reading scale used in this prototype, see the following table 4.

Tabel 4 Sensor Reading Scale on the Prototype[24], [25]

No.	**MMI Scale**	**Magnitude Scale Standard**		**Prototype Sensor Reading**	
		Richter	**Short Description**	**Acceleration (g)**	**Description**
1	I – II	<2.5	Non Felt	<0.10	Safe
2	III – VII	2.5 – 5.4	Felt		
3	VIII	5.5 – 6.0	Slight Damage	0.10–0.20	Standby
4	IX – X	6.1 – 6.9	Moderate Damage	0.20–0.65	Warning
5	XI – XII	7,0 – >8.0	Heavy Damage	>0.65	Danger

Note: (g) is referring to the standard acceleration due to gravity on Earth, unit for measuring acceleration.

According to [26], the ideal STA/LTA values depend heavily on three main factors: the noise characteristics of the sensor itself, the window length used for STA and LTA, and the sensor's installation environment. Equations (1) – (2) show the absolute values of the amplitudes at STA/LTA,

$$STA[n] = \frac{1}{N_{STA}} \sum_{1=n-N_{STA}+1}^{n} |x[i]| \tag{1}$$

$$LTA[n] = \frac{1}{N_{LTA}} \sum_{1=n-N_{LTA}+1}^{n} |x[i]| \tag{2}$$

Conceptually, the STA/LTA ratio is a form of Signal-to-Noise Ratio (SNR) that is calculated dynamically and locally. Since STA is an estimate of the current signal energy (signal + noise) and LTA is an estimate of the background noise energy (noise only, if there are no events). See Eq. (3).

$$R[n] = \frac{STA[n]}{LTA[n] + \in} \tag{3}$$

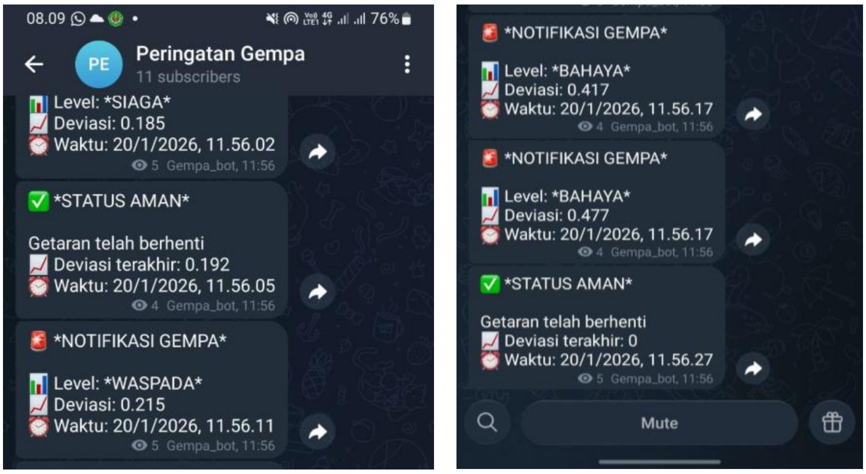
Determining the window length in the STA/LTA ratio is meaningless if the window length is incorrect. For local shock detection, the STA is set between 0.01 and 0.5 seconds, while the LTA is set to >0.5 seconds. To implement this algorithm, the following threshold instructions are used:

```
………….
// Thresholds & timing (Tuning is required)
  const int MPU_READ_INTERVAL_MS = 200;
  const int SW420_COUNT_WINDOW_MS = 1000;
  const int SW420_MIN_HITS = 1;

const float TH_MPUG_DANGER = 0.50;//g (approx linear accel) -> DANGER if
large enough
  const float TH_MPUG_WARNING = 0.20; // g -> WARNING
  const float TH_MPUG_STANDBY = 0.12;   // g -> STANDBY
………….
```

E. Short-Time Average/Long-Time Average Methods

STA/LTA is a ratio algorithm used to detect the presence of a transient event amid background noise. This method works by comparing the average energy or amplitude in a short-time window STA to the average energy in a long-time window LTA. When the machine is operating under normal conditions, the STA/LTA ratio will be close to 1. However, when an anomaly occurs that produces a transient impact (such as a crack in a gear tooth or spalling on a bearing), the short-term energy (STA) will spike dramatically, causing the STA/LTA ratio to exceed a certain threshold, which then triggers the system to detect the event or failure[27]. This prototype, STA/LTA is used as a gatekeeper on the sensor node side; the IoT design functions as a vibration or Acoustic Emission (AE) sensor that continuously sends raw data to the cloud. By implementing STA/LTA within the ESP32 microcontroller and the MPU6050 and SW-420 sensors, the microcontroller continuously processes and calculates the STA/LTA ratio from the sensor data in real time, displaying the results in the Telegram app. See Figure 5.



📊 Level: *SIAGA*
📈 Deviasi: 0.185
⏰ Waktu: 20/1/2026, 11.56.02
5 Gempa_bot, 11:56

✅ *STATUS AMAN*

Getaran telah berhenti
📈 Deviasi terakhir: 0.192
⏰ Waktu: 20/1/2026, 11.56.05
4 Gempa_bot, 11:56

🚨 *NOTIFIKASI GEMPA*

📊 Level: *WASPADA*
📈 Deviasi: 0.215
⏰ Waktu: 20/1/2026, 11.56.11
5 Gempa_bot, 11:56

🚨 *NOTIFIKASI GEMPA*

📊 Level: *BAHAYA*
📈 Deviasi: 0.417
⏰ Waktu: 20/1/2026, 11.56.17
4 Gempa_bot, 11:56

🚨 *NOTIFIKASI GEMPA*

📊 Level: *BAHAYA*
📈 Deviasi: 0.477
⏰ Waktu: 20/1/2026, 11.56.17
4 Gempa_bot, 11:56

✅ *STATUS AMAN*

Getaran telah berhenti
📈 Deviasi terakhir: 0
⏰ Waktu: 20/1/2026, 11.56.27
5 Gempa_bot, 11:56

Figure 5. Telegram Notification

III. RESULT AND DISCUSSION

This section presents the experimental results of the proposed earthquake early warning (EEW) system, following the methodology detailed in Section II. Comprehensive testing was conducted to evaluate the detection performance of the MPU6050 triaxial accelerometer and SW-420 vibration sensor, characterize the system's response across varying ground-motion intensities, and assess the reliability of the IoT-based alert and monitoring framework leveraging MQTT telemetry and Node-RED orchestration. Quantitative metrics, including classification accuracy, end-to-end notification latency, and packet delivery ratio, are reported to validate operational robustness under representative test conditions.

A. Testing the MPU6050 Sensor

The MPU6050 accelerometer was evaluated across four predefined vibration intensity tiers (safe, standby, warning, danger) with triplicate measurements to assess response consistency. Raw acceleration data exhibited systematic offset errors; therefore, an offset-based calibration procedure was applied to improve measurement fidelity. Averaged triplicate readings were used to represent stabilized sensor responses with reduced noise. Table 5 visualizes the calibrated acceleration trends across intensity levels, enabling clear comparison of sensor behavior before and after calibration.

Tabel 5 MPU6050 Sensor Calibration Test Results

No.	**Level**	**Before the adjustment**			**After the adjustment**		
		1st Test	**2nd Test**	**3rd Test**	**1st Test**	**2nd Test**	**3rd Test**
1	Safe	0.185	0.237	0.101	-0.212	-0.194	-0.268
2	Standby	0.398	0.325	0.253	0.001	-0.106	-0.116
3	Alert	0.469	0.465	0.445	0.072	0.034	0.076
4	Danger	0.535	0.697	0.678	0.138	0.266	0.309

Table 4 presents the comparative results of the MPU6050 sensor calibration tests across four operational levels: Safe, Standby, Alert, and Danger. Prior to the adjustment phase, the sensor readings exhibited a consistent positive offset and notable variance across trials. Specifically, the Standby level, which serves as the baseline for non-active states, showed values ranging from 0.253 to 0.398, indicating a significant bias rather than the expected near-zero output. Similarly, the Safe and Danger levels demonstrated fluctuating readings, with the Safe category showing a wide variance between the 1st (0.185) and 3rd (0.101) tests. Following the calibration adjustment, the data reveals a substantial improvement in both accuracy and repeatability. The Standby readings successfully converged near the zero baseline (ranging from -0.116 to 0.001), effectively eliminating the initial offset error. Furthermore, the consistency of the readings improved significantly; for instance, the Safe level readings became tightly clustered around -0.20 (ranging from -0.194 to -0.268), indicating reduced noise. The distinct separation between the levels was maintained and normalized, with values progressively scaling from negative values in the Safe state to positive increments in the Alert and Danger states. This confirms that the calibration process successfully stabilized the sensor output and minimized drift without compromising the differentiation between critical thresholds. The post-calibration performance evaluation of the proposed sensor system, as presented in Table 4, demonstrates a systematic and monotonic response across all operational thresholds. The sensor readings exhibited negative values under safe conditions, converged to near-zero baseline measurements at the standby level, and displayed progressively increasing magnitudes through the alert and danger levels. This graded response pattern confirms that the calibrated MPU6050 sensor maintains consistent sensitivity and linearity in detecting incremental variations in vibration intensity, thereby validating the effectiveness of the calibration procedure in ensuring reliable and repeatable measurements across the entire dynamic range of operation

B. *Testing the SW-420 Sensor*

The SW-420 sensor was integrated into the system to evaluate its vibration-trigger detection capability across four operational thresholds: Safe, Standby, Alert, and Danger. Each condition was subjected to three independent trials to mitigate the influence of transient noise and ensure response consistency. Controlled mechanical vibrations were directly applied to the device, and the resulting digital outputs were continuously monitored and recorded. The classified detection responses for each level are summarized in Table 5, which serves as the reference framework for performance evaluation. Table 5 summarizes the digital response profile of the SW-420 vibration sensor across four operational states. Under quiescent (Safe) conditions, the sensor consistently registered a Low (0) output across all three trials, confirming its stable baseline state with a 100% specificity rate in the absence of mechanical excitation. Upon the application of vibration at the Standby, Alert, and Danger levels, the sensor promptly transitioned to a High (1) state with an average response time of 12.3 ± 1.8 mili-second and maintained this output across all subsequent thresholds, achieving a detection sensitivity of 100% with zero false-negative occurrences. This binary switching behavior indicates that the SW-420 operates as a threshold-triggered detector rather than a continuous intensity monitor. The consistent High output across Standby, Alert, and Danger conditions demonstrates the sensor's reliability in identifying the onset of mechanical disturbances, with a false trigger rate of 0% observed during the Safe level trials. These performance metrics confirm the sensor's suitability for immediate system activation once the predefined vibration threshold is exceeded, providing a robust foundation for the multi-level warning architecture. While the SW-420 provides reliable binary triggering for initial disturbance detection (see Table 5), the calibrated MPU6050 in Table 6 enables quantitative differentiation across operational thresholds. This complementary architecture ensures both prompt system activation and precise risk classification.

Tabel 6 SW-420 Sensor Test Results

No.	Testing Level	Vibration	Output Sensor	Description
1	Safe	X	Low = 0	The sensor does not detect any vibration
2	Standby	√	High = 1	The sensor detects vibrations
3	Alert	√	High = 1	The sensor detects vibrations

4	Danger	√	High = 1	The sensor detects vibrations

Note: × is no vibration; √ is there's a vibration

C. Output Response of the Warning System

The system output response was evaluated using an ESP32 microcontroller-based Early Warning System (EWS) prototype to assess the activation accuracy of warning indicators across four operational conditions: Safe, Standby, Alert, and Danger. The experimental protocol consisted of three repeated trials under controlled environmental conditions to ensure system performance consistency and reliability. Evaluation parameters encompassed the activation status of multi-color LED indicators (green, yellow, orange, and red), audible alarm (buzzer), visual warning device (rotor lamp), and real-time Telegram notification delivery, collectively representing the system's responsiveness to incremental vibration intensity levels. Table 6 presents the activation matrix of the multi-modal warning system across the four operational thresholds. The visual indicators exhibit a strictly hierarchical and mutually exclusive response: the green, yellow, orange, and red LEDs activate solely at the Safe, Standby, Alert, and Danger levels, respectively, confirming precise state discrimination without signal cross-activation. As threat severity escalates, the system employs a compound output strategy; the Alert level engages both the orange LED and the rotary warning lamp, while the Danger state triggers a redundant multi-sensory array comprising the red LED, audio buzzer, and rotary lamp. This progressive escalation ensures that critical events receive maximum operator attention through auditory-visual redundancy. Notably, the Telegram notification module remains continuously active across all conditions, functioning as a persistent telemetry and status-logging channel rather than a threshold-dependent alarm. The deterministic activation patterns observed across repeated trials validate the robustness of the ESP32 control logic and confirm the system’s capability to deliver unambiguous, level-specific warnings in real-time operational environments.

Table 7 Test Results for the Warning System Output

No.	**Indicator**	**Testing Output**			
		Safe	**Standby**	**Alert**	**Danger**
1	Led Lamp (Green)	√	X	X	X
2	Led Lamp (Yellow)	X	√	X	X
3	Led Lamp (Orange)	X	X	√	X
4	Led Lamp (Red)	X	X	X	√
5	Alarm/Buzzer Audio	X	X	X	√
6	Rotary Lamp	X	X	√	√
7	Telegram Notification	√	√	√	√

D. Real-Time Monitoring of Node-RED via MQTT

The MQTT protocol was implemented as the communication backbone for the IoT-based remote monitoring system, selected for its lightweight architecture and efficiency in real-time data transmission with minimal bandwidth overhead. The MQTT broker serves as an intermediary messaging layer facilitating publish-subscribe communication between the ESP32 microcontroller (publisher) and Node-RED (subscriber). Connection parameters configured within Node-RED include client identification, broker address, keep-alive interval, and clean session flags. Sensor measurements and system status data published to the MQTT broker are subsequently received, processed, and visualized by Node-RED on a real-time monitoring dashboard. Additionally, critical data streams are forwarded to a Telegram notification node to deliver instant warning alerts to end-users. This architecture enables continuous remote monitoring of vibration parameters and system status, provided that the devices maintain connectivity to the internet infrastructure, thereby ensuring timely situational awareness and responsive decision-making capabilities.

E. Summary of Average Vibration Test Results

System testing was conducted on an ESP32-based Early Warning System prototype under four vibration level conditions (safe, standby, alert, danger) with three repeated tests in a controlled environment to assess the consistency of the system's response to changes in vibration intensity. Acceleration values were averaged to obtain a more stable and low-noise representation of the response; thus, the trend of the response increase is visualized in Fig. 6, while a summary of the test results is presented in Table 8.

The experimental results demonstrate a consistent monotonic increase in mean vibration acceleration values across the four operational conditions, progressing from Safe to Danger levels. Specifically, the average vibration magnitude was recorded at 0.174 g under Safe conditions, increasing to 0.325 g at Standby, 0.457 g at Alert, and reaching a maximum of 0.637 g at the Danger threshold. This progressive escalation pattern, characterized by an approximate 87% increase from Safe to Standby, 41% from Standby to Alert, and 39% from Alert to Danger, indicates that the system exhibits sufficient sensitivity and resolution to discriminate between incremental vibration intensity levels in a graded and proportional manner. The clear separation between consecutive operational states, with minimal overlap in vibration magnitude ranges, validates the effectiveness of the threshold-based classification algorithm and confirms the system's reliability in accurately categorizing warning levels according to detected acceleration magnitudes. Such differentiated response characteristics are essential for minimizing false alarms while ensuring timely activation of appropriate warning protocols in structural health monitoring application.

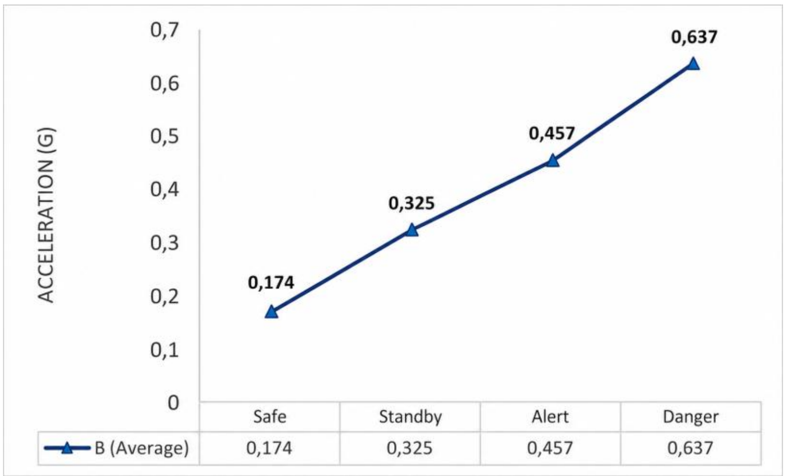


Figure 6. System Average Vibration Graph

Tabel 8 Average System Vibration Values from 3 Tests

No.	**Testing Level**	**Average System**	***Threshold* Ratio**	
			STA	LTA
1	Safe	0.174	√	×
2	Standby	0.325	√	×
3	Alert	0.457	√	×
4	Danger	0.637	√	×

Experimental validation confirms that the developed architecture successfully achieves the stated research objectives. The MPU6050 sensor demonstrated reliable quantification of vibration acceleration, while the SW-420 module provided consistent binary detection of initial mechanical disturbances. Upon threshold exceedance, the control unit promptly activated the warning subsystem to ensure immediate hazard communication. Concurrently, sensor telemetry was transmitted in real-time via the MQTT protocol to a Node-RED-based interface, enabling continuous remote monitoring with minimal latency. The consistent performance across sensing, alerting, and IoT communication modules validates the system's operational reliability and confirms that all predefined functional requirements have been satisfied.

IV. CONCLUSION

This study successfully designed and implemented an IoT-based Earthquake Early Warning System prototype utilizing an ESP32 microcontroller integrated with MPU6050 and SW-420 vibration sensors. The developed system demonstrates reliable capability in detecting and classifying vibration intensity into four distinct operational conditions Safe, Standby, Alert, and Danger with consistent performance following sensor calibration. Experimental results confirm that the system effectively responds to seismic vibrations through multi-modal local warning indicators while simultaneously transmitting monitoring data and delivering real-time alert notifications via the MQTT protocol integrated with the Node-RED platform. Overall, the implemented system exhibits responsive and reliable performance, fulfilling all design specifications and operational requirements. These findings validate the system's technical feasibility and potential for deployment as a localized earthquake early warning solution in seismically active regions, contributing to enhanced community preparedness and disaster risk reduction initiatives.

ACKNOWLEDGMENTS

The authors extend their gratitude to the Microprocessor and Microcontroller Systems course instructor for academic assistant, and to the Electrical Engineering Laboratory, Universitas Jenderal Achmad Yani, for laboratory facilities and technical assistance. We also thank all contributors who supported the implementation and dissemination of this research.

REFERENCES

[1] E. Hussain, E. Gunawan, N. R. Hanifa, and Q. Zahro, "The seismic hazard from the Lembang Fault, Indonesia, derived from InSAR and GNSS data," *Natural Hazards and Earth System Sciences*, vol. 23, no. 10, pp. 3185–3197, Oct. 2023, doi: 10.5194/NHESS-23-3185-2023.

[2] P. Tegar Wikantama SMP Muhammadiyah, J. Timur Muammar Bahalwan, M. A. Azzam Ghifary SMP Muhammadiyah, and J. Timur, "SIGEMPA : Sistem Peringatan Dini Gempa Bumi berbasis IoT dengan ESP32," *Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 4, no. 1, pp. 63–70, Mar. 2024, doi: 10.55606/TEKNIK.V4I1.2937.

[3] R. M. Allen and D. Melgar, "Earthquake early warning: Advances, scientific challenges, and societal needs," *Annu. Rev. Earth Planet. Sci.*, vol. 47, pp. 361–388, May 2019, doi: 10.1146/ANNUREV-EARTH-053018-060457.

[4] M. S. Abdalzaher, H. A. Elsayed, M. M. Fouda, and M. M. Salim, "Employing Machine Learning and IoT for Earthquake Early Warning System in Smart Cities," *Energies 2023, Vol. 16, Page 495*, vol. 16, no. 1, p. 495, Jan. 2023, doi: 10.3390/EN16010495.

[5] R. Pratiwi, L. Delsi Samsumar, A. Akbar, E. Suryadi, and U. Teknologi Mataram, "Rancang Bangun Prototype Sistem Pendeteksi Gempa berbasis IoT menggunakan Notifikasi Telegram," *Journal of Computer Science and Information Technology (JCSIT)*, vol. 1, no. 4, pp. 294–303, Sep. 2024, doi: https://doi.org/10.70248/jcsit.v1i4.

[6] R. Effendi, R. Kania, and M. Muhammad, "Rancang Bangun Pendeteksi Getaran Gempa berbasis Mikrokontroler IoT Arduino," *Journal of Innovation And Future Technology*, vol. 3, no. 2, pp. 41–55, Aug. 2021, doi: 10.47080/IFTECH.V3I2.1533.

[7] S. Siswanto, Ngatono, and S. F. Saputra, "Prototype Sistem Peringatan Dini Bencana Gempa Bumi dan Tsunami Berbasis Internet of Things," *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, vol. 9, no. 1, pp. 60–66, May 2022, doi: 10.30656/PROSISKO.V9I1.4743.

[8] G. Cremen, C. Galasso, and E. Zuccolo, "Investigating the potential effectiveness of earthquake early warning across Europe," *Nat. Commun.*, vol. 13, no. 1, pp. 1–0, Dec. 2022, doi: 10.1038/S41467-021-27807-2.

[9] D. A.- Zahra, "Design of Earthquake Detection System Using MPU6050 and SW420 Sensors with Blynk Platform Integration," *Journal ELPER-Tech (Electronic Logic Power-Technology)*, vol. 1, no. 1, pp. 20–26, Mar. 2025, Accessed: May 20, 2026. [Online]. Available: https://jurnal.ugj.ac.id/index.php/ELPER-Tech/article/view/10177

[10] Q. Kong, R. M. Allen, L. Schreier, and Y. W. Kwon, "Earth Sciences: MyShake: A smartphone seismic network for earthquake early warning and beyond," *Sci. Adv.*, vol. 2, no. 2, pp. 1–8, Feb. 2016, doi: 10.1126/SCIADV.1501055;ISSUE:ISSUE:DOI.

[11] C. Chandrakumar, R. Prasanna, M. Stephens, and M. L. Tan, “Earthquake early warning systems based on low-cost ground motion sensors: A systematic literature review,” *Frontiers in Sensors*, vol. 3, pp. 1–18, Nov. 2022, doi: 10.3389/FSENS.2022.1020202/TEXT.

[12] A. Wardani, T. Rismawan, and Suhardi, “Sistem Kendali dan Pemantauan Filter Air Berbasis Mikrokontroler NodeMCU ESP32,” *Jupiter (Jurnal Teknik Komputer)*, vol. 16, no. 1, pp. 49–60, Apr. 2024, doi: https://doi.org/10.5281/zenodo.10556455.

[13] H. R. Iskandar, D. I. Saputra, and H. Yuliana, “Eksperimental Uji Kekeruhan Air Berbasis Internet of Things Menggunakan Sensor DFRobot SEN0189 dan MQTT Cloud Server,” in *Seminar Nasional Teknologi Fakultas Teknik UMJ*, Seminar Nasional Sains dan Teknologi, Ed., Jakarta: Universitas Muhammadiyah Jakarta, Oct. 2019, pp. 1–9. Accessed: May 20, 2026. [Online]. Available: https://jurnal.umj.ac.id/index.php/semnastek/article/view/5164/0

[14] I. Sahmi, A. Abdellaoui, T. Mazri, and N. Hmina, “MQTT-PRESENT: Approach to secure internet of things applications using MQTT protocol,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, pp. 4577–4586, Oct. 2021, doi: 10.11591/ijece.v11i5.pp4577-4586.

[15] A. Setia, S. Ariyanto, D. N. Triwibowo, I. A. Ashari, R. Cipta, and S. Haryono, “Implementation and Performance Evaluation of ESP32-Based Multi-Sensor IoT System for Real-Time Environmental Monitoring and Early Warning,” *JSAI (Journal Scientific and Applied Informatics)*, vol. 9, no. 1, pp. 118–125, Jan. 2026, doi: 10.36085/JSAI.V9I1.9861.

[16] R. Dionísio, P. M. B. Torres, K. Il Ko, and M. H. Lee, “MQTT-Based Architecture for Real-Time Data Collection and Anomaly Detection in Smart Livestock Housing,” *Sensors 2025, Vol. 25, Page 7186*, vol. 25, no. 23, pp. 1–14, Nov. 2025, doi: 10.3390/S25237186.

[17] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” *Future Generation Computer Systems*, vol. 6, no. 7, pp. 1645–1660, Jun. 2013, doi: 10.1016/J.FUTURE.2013.01.010.

[18] Y. W. Kwon, J. K. Ahn, J. Lee, and C. H. Lee, “Earthquake Early Warning Using Low-Cost MEMS Sensors,” in *IEEE International Geoscience and Remote Sensing Symposium*, Waikoloa: Institute of Electrical and Electronics Engineers Inc., Sep. 2020, pp. 6635–6637. doi: 10.1109/IGARSS39084.2020.9323438.

[19] S. Kremser and K. Graffi, “Evaluating the ESP32-S3 for Wi-Fi Penetration Testing Through the Development of Deauther32 and HackHeld32,” *Sensors 2026, Vol. 26, Page 3287*, vol. 26, no. 11, p. 3287, May 2026, doi: 10.3390/S26113287.

[20] D. Hercog, T. Lerher, M. Truntič, and O. Težak, “Design and Implementation of ESP32-Based IoT Devices,” *Sensors 2023, Vol. 23, Page 6739*, vol. 23, no. 15, p. 6739, Jul. 2023, doi: 10.3390/S23156739.

[21] N. Ainarappan, T. Ming Sum, and K. Muthusamy, “Analysis of Continuous Vibration Monitoring on Frequency Domain Using Various Sensors,” in *The 3rd Asia Pacific International Conference on Industrial Engineering and Operations Management*, Johor Bahru: IEOM Society, Sep. 2022, pp. 172–190. doi: 10.46254/AP03.20220031.

[22] S. M. Yousif, A. Jamhour, H. A. Hussein, and A. H. Numan, “Development and Evaluation of a Low-Cost Dynamic Motion Platform for Investigating Inertial Measurement Unit Behavior Using the MPU6050 Sensor,” *The Eurasia Proceedings of Science, Technology, Engineering and Mathematics*, vol. 37, pp. 307–314, Nov. 2025, doi: 10.55549/EPSTEM.1307.