

NEUROSYN: A HYBRID BRAINWAVE–GESTURE CONTROL SYSTEM

Dr. Meril Cyriac¹, Dr. Kanjana G², Dr. Reena M. Roy³, Soju Ravi K⁴,
Smitha K. S⁵, Hema S⁶

¹²³⁴⁵⁶Department of Electronics and Communication Engineering, LBS Institute of Technology for Women,
(Affiliated to APJ Abdul Kalam Technological University),
Poojappura, Thiruvananthapuram – 695012, Kerala, India

¹merilcyriac@lbsitw.ac.in, ²kanjana@lbsitw.ac.in, ³reenamroy@lbsitw.ac.in, ⁴sojuravi@lbsitw.ac.in,
⁵kssmitha@lbsitw.ac.in, ⁶hema@lbsitw.ac.in

Abstract—Today, automation technology is becoming increasingly accessible, user friendly, and efficient, enabling novel solutions for improving the quality of life. This project explores the integration of brain-computer interfaces (BCI) with the Internet of Things (IoT) to provide a smart assistive system for individuals with limited physical mobility. The system employs an EEG headset to capture real-time brain signals, which are transmitted via Bluetooth to an ESP32 microcontroller for processing. Based on the interpreted brain signals, the system can control electrical appliances such as lights and fans by switching a relay module ON or OFF. The system also supports hand gesture control as a backup input method, ensuring reliable operation even when brain signals are unstable. Additionally, the system is designed to detect abnormal brain activity, automatically triggering alerts through an application, and activating a buzzer to notify caregivers or family members. The application also provides a real-time display of appliance status, ensuring users are always aware of the current system state. The expected outcome of this project is a reliable and user-friendly brain-controlled automation system that enhances safety, and convenience for physically challenged individuals.

Keywords—Brain–Computer Interface (BCI), Internet of Things (IoT), EEG Signals, Assistive Technology.

How to cite this article: Cyriac M, Kanjana G, Roy RM, Ravi K S, Smitha KS, Hema S. NEUROSYN: A Hybrid Brainwave–Gesture Control System. *Int J Drug Deliv Technol.* 2026;16(72s): 1598-1607. DOI: 10.25258/ijddt.16.72s.179

1. Introduction

The evolution of automation technologies over the past decade has significantly transformed modern living by enabling intelligent and responsive environments. The integration of the Internet of Things (IoT) has played a crucial role in this transformation by connecting physical devices to the internet, allowing them to communicate, monitor, and operate autonomously. Smart home automation systems, in particular, have gained widespread attention due to their ability to enhance comfort, energy efficiency, and convenience. These systems typically rely on conventional control methods such as manual switches, mobile applications, and voice assistants. However, despite their advantages, such methods are not universally accessible, especially for individuals with severe physical disabilities or mobility impairments.

A large section of the population, including paralyzed patients, elderly individuals, and people suffering from neurological disorders, faces significant challenges in interacting with traditional automation systems. These users often depend on caregivers for performing even simple daily tasks such as switching ON/OFF appliances. This dependency not only reduces their independence but also introduces delays in response during critical situations. In addition, voice-based systems may not function effectively for users with speech impairments, while mobile-based interfaces require a certain level of motor control and familiarity with

technology, further limiting accessibility. Brain–Computer Interface (BCI) technology has emerged as a promising solution to address these limitations by enabling direct communication between the human brain and external electronic systems. BCI systems operate by capturing electroencephalogram (EEG) signals, which represent the electrical activity generated by neurons in the brain. These signals are processed and interpreted to identify patterns corresponding to different mental states such as attention, relaxation, or intentional focus. By mapping these patterns to control commands, users can interact with devices without requiring any physical movement, making BCI a powerful tool in assistive and rehabilitation technologies.

Despite its advantages, the implementation of EEG-based control systems presents several technical and practical challenges. EEG signals are extremely low in amplitude and are highly sensitive to noise and interference. External factors such as muscle movements, blinking, environmental electrical noise, and emotional fluctuations can introduce distortions in the signal. Moreover, different users exhibit variations in brain signal patterns, making it difficult to design a one-size-fits-all solution. These challenges necessitate the use of robust signal processing techniques and reliable decision-making mechanisms to ensure accurate interpretation of user intent.

Another major concern in brainwave-controlled systems is the issue of false triggering. Situations

such as increased attention due to unrelated thoughts or emotional responses can generate signal patterns similar to intentional control signals. Without proper filtering and threshold mechanisms, this can lead to unintended switching of appliances. Therefore, it becomes essential to implement threshold-based validation, continuous monitoring, and stability checks before executing control actions. These mechanisms help in improving the reliability and safety of the system during real-time operation.

The integration of IoT further enhances the system by enabling real-time data visualization, remote monitoring, and control. Users and caregivers can access system status, appliance states, and alert notifications through a connected interface, ensuring transparency and ease of use. The use of an LCD within the system provides immediate local feedback, displaying parameters such as control mode, signal values, and appliance status, thereby improving interaction and usability. From a system design perspective, the combination of wireless communication, embedded processing, and real-time decision-making creates a compact and efficient architecture suitable for practical deployment. The use of cost-effective components such as ESP32 and commercially available EEG headsets makes the system economically viable, which is an important factor for large-scale adoption. Additionally, the modular nature of the design allows for future enhancements, such as integration with machine learning algorithms for improved signal classification or expansion to control a wider range of devices. The inclusion of a gesture-based fallback system significantly enhances the robustness of the overall design. In cases where brain signals are unstable, inconsistent, or unavailable, the system can switch to gesture mode, allowing the user to control appliances using hand movements detected by sensors. This multi-input approach ensures uninterrupted functionality and reduces the dependency on a single mode of interaction. It also improves user confidence, as the system remains responsive even under varying conditions. In addition to control functionality, the system incorporates continuous monitoring of brain signal patterns to identify abnormal conditions. Sudden or irregular variations in EEG signals may indicate stress, discomfort, or other neurological anomalies. The system is designed to detect such patterns and trigger an alert mechanism, which includes activating a buzzer and sending notifications through an IoT platform. This feature introduces an additional dimension of safety, enabling timely intervention in critical situations.



Fig 1: FT&S Mind Link Head Band

To address these limitations, the proposed system adopts a hybrid control architecture that combines brainwave-based input with an alternative gesture-based control mechanism [14]. The system utilizes an EEG headband to acquire real-time brain signals, which are transmitted wirelessly via Bluetooth to an ESP32 microcontroller. The microcontroller processes the incoming data and applies threshold-based logic to determine whether a valid control condition has been satisfied. Once validated, the system generates control signals to operate electrical appliances through a relay module. Another major concern in brainwave-controlled systems is false triggering. Situations such as increased attention due to unrelated thoughts or emotional responses can generate signal patterns similar to intentional control signals. Without proper filtering and threshold mechanisms, this can lead to unintended switching of appliances. Therefore, it becomes essential to implement threshold-based validation, continuous monitoring, and stability checks before executing control actions.

2. LITERATURE REVIEW

Brain–Computer Interface (BCI) technology has been recognized as a potential solution for establishing communication between the human brain and external devices. Recent developments in electroencephalography (EEG), machine learning, and deep learning have improved the applications of EEG-based brain–computer interfaces in healthcare, rehabilitation, smart homes, and other fields.

Zhang et al. [1] proposed a unified deep learning framework for Brain–Computer Interface and Internet of Things (IoT). The study introduced the concept of human–thing cognitive interaction and demonstrated the feasibility of using EEG signals to control IoT devices. The framework showed improved reliability and scalability for cognitive IoT applications compared to conventional approaches. Bang et al. [2] introduced a novel multimodal human–computer interface that utilizes gaze and EEG signals for improved accuracy. The approach demonstrated better performance than conventional EEG-based interfaces by minimizing false triggering and increasing the accuracy of control commands.

Recent developments in computational intelligence have significantly improved the processing of EEG signals. Gu et al. [3] presented a comprehensive survey of EEG-based brain–computer interface technologies that have been widely applied in signal

acquisition, preprocessing, feature extraction, classification, and applications. The review highlighted the superior performance of deep learning algorithms in EEG decoding while presenting limitations in signal-to-noise ratio and inter-subject variability. The application of transfer learning in EEG-based brain-computer interface has gained significant research interest due to its ability to improve the performance of classification models while reducing training time.

Wu et al. [4] presented a review of recent advances in transfer learning for EEG-based brain-computer interface applications. The study demonstrated the effectiveness of transfer learning in minimizing the calibration burden and improving classification accuracy across different subjects and tasks. The review highlighted the importance of domain adaptation and deep transfer learning techniques in improving the performance of BCI systems.

The application of BCI in smart home automation has been widely investigated as a potential solution for controlling home appliances using EEG signals. Maleki et al. [5] presented a review of BCI-based smart home technologies that have been widely explored in the literature. The study highlighted different signal acquisition and processing techniques used in the development of smart home systems and control strategies for managing home appliances. The review demonstrated that BCI-based smart home systems could be used to improve the quality of life for the elderly and physically challenged individuals.

Ajmeria et al. [6] presented a comprehensive survey of EEG-based brain-computer interface technologies for Industrial Internet of Things (IIoT) applications. The study highlighted the significance of artificial intelligence and cloud computing technologies in processing large volumes of EEG data while ensuring minimal latency and reliable communication. The review also identified several research challenges in the application of BCI in IIoT, including cybersecurity concerns and the need for scalable solutions.

The increasing popularity of smart home technologies has prompted researchers to design intelligent systems that can support individuals with minimal manual effort. Zielonka et al. [7] presented a review of recent advances in smart home technologies that have been widely explored in the literature. The study highlighted the significance of artificial intelligence, IoT, wireless communication, and human-computer interaction technologies in the development of smart home systems. The review demonstrated that BCI technologies could be incorporated into smart home systems to support individuals with severe motor disabilities.

The application of visual stimuli in BCI systems has been widely investigated due to their high information transfer rate and improved control capabilities. Park et al. [8] presented an augmented

reality-based home appliance control system that utilizes steady-state visual evoked potentials for improved performance. The study demonstrated the feasibility of using visual stimuli to operate home appliances while incorporating augmented reality for enhanced user experience.

Similarly, the integration of eye gaze and steady-state visual evoked potential (SSVEP) has been demonstrated as an effective method for operating smart home appliances. Putze et al. [9] presented a novel approach for smart home control using eye gaze and SSVEP-based brain-computer interface. The multimodal interface demonstrated improved accuracy and reduced cognitive workload compared to conventional approaches.

Deep learning algorithms have gained significant research interest in the field of EEG-based brain-computer interface due to their ability to automatically extract discriminative features from EEG signals. Gu et al. [10] demonstrated that convolutional neural networks, recurrent neural networks, and hybrid deep learning approaches have shown superior performance in EEG classification tasks. The review highlighted the importance of large-scale EEG datasets and optimized neural network architectures for improved performance. Al-Saegh et al. [11] presented a review of deep learning algorithms for motor imagery EEG classification. The study compared the performance of convolutional neural networks, long short-term memory networks, and other deep learning approaches in decoding motor imagery EEG signals. The review concluded that hybrid deep learning approaches could offer better performance in motor imagery classification tasks.

Aggarwal and Chugh [12] presented a review of machine learning algorithms for EEG-based brain-computer interface applications. The study highlighted the significance of feature extraction and selection methods in improving the performance of machine learning classifiers such as support vector machines, random forests, k-nearest neighbors, artificial neural networks, and deep learning approaches. Al-Qaysi et al. [13] conducted a systematic review of non-invasive EEG acquisition equipment for assistive and rehabilitation applications. The study compared the performance of different EEG acquisition systems, including electrodes and signal processing techniques, to identify the most effective and cost-effective solutions for BCI applications.

Flores Vega et al. [14] presented a novel fuzzy temporal convolutional neural network approach for P300-based smart home control applications. The study demonstrated the effectiveness of incorporating fuzzy logic and temporal convolutional neural networks in improving the accuracy of P300-based brain-computer interface systems. Hsu [15] introduced a new approach for motor imagery classification based on wavelet

fractal features and neuro-fuzzy prediction. The study demonstrated the effectiveness of the proposed method in extracting discriminative features from EEG signals while improving classification accuracy.

Abd Rani et al. [16] presented an EEG-based eye blink detection system for controlling home lighting systems. The study demonstrated that simple EEG-based commands could be used to operate home appliances, providing a low-cost solution for individuals with limited mobility. Abdulwahab et al. [17] presented a comprehensive survey of EEG-based brain-computer interface applications and implementations in different fields, including healthcare, gaming, robotics, rehabilitation, education, and assistive technologies. The study highlighted the current challenges and future research directions for EEG-based BCI technologies.

Abhang et al. [18] discussed EEG signal analysis methods and their application in emotion recognition. The study demonstrated the significance of EEG-based brain-computer interface in understanding human emotions and cognitive states. Al-Fahoum and Al-Fraihat [19] presented a review of different feature extraction methods for EEG signals in time and frequency domains. The study highlighted the importance of using hybrid feature extraction methods for improved classification performance.

Gouda et al. [20] presented a comprehensive review of EEG-based brain-computer interface applications in smart home environments. The study evaluated different human-computer interaction methods, communication protocols, and machine learning algorithms used in smart home automation systems. The review concluded that hybrid brain-computer interface approaches that incorporate deep learning, IoT, and multimodal interaction offer the best performance in smart home applications.

In general, the reviewed studies demonstrate that EEG-based brain-computer interface technologies have been significantly improved by recent advances in deep learning, transfer learning, and IoT technologies. However, several challenges, including signal noise, inter-subject variability, calibration time, and cybersecurity concerns, still hinder the widespread adoption of BCI technologies. Future research should focus on developing efficient deep learning algorithms, hybrid multimodal approaches, and secure communication protocols to improve the performance and reliability of EEG-based brain-computer interface systems.

3. PROBLEM STATEMENT

The rapid advancement of home automation technologies has significantly improved convenience and efficiency in modern households. However, most existing systems are designed with conventional interaction methods such as manual switches, smartphone applications, or voice-

controlled assistants. While these approaches are effective for the general population, they present serious limitations for individuals with physical disabilities, elderly users, and patients with restricted mobility. Such users often face difficulties in performing even basic tasks like controlling household appliances, leading to increased dependency on caregivers and reduced independence in daily life.

In addition to accessibility challenges, traditional automation systems lack alternative interaction mechanisms that can function reliably across different conditions. Voice-controlled systems may fail in noisy environments or for users with speech impairments, while mobile-based applications require precise motor control and continuous engagement. These constraints highlight the absence of inclusive design in many existing solutions. There is a clear need for systems that can interpret user intent without relying on physical movement or verbal communication, thereby enabling seamless and hands-free interaction. Brain-Computer Interface (BCI) technology provides a promising approach by enabling direct control through brain signal.

However, the practical implementation of EEG-based systems introduces several challenges related to signal reliability and accuracy. EEG signals are inherently weak, highly variable, and susceptible to noise caused by environmental interference, muscle activity, and emotional fluctuations. This makes it difficult to consistently distinguish intentional commands from unintentional variations, often resulting in false triggering or failure in executing desired actions, thereby reducing system reliability in real-time applications. Another critical limitation is the lack of robustness in systems that rely on a single mode of input. Purely brainwave-based systems do not provide a fallback mechanism when signal acquisition becomes unstable or inconsistent, leading to system inoperability under certain conditions. Furthermore, many existing solutions do not incorporate mechanisms to monitor abnormal brain activity, which is particularly important in assistive systems where user safety is a priority. The absence of such features limits the applicability of these systems in real-world scenarios.

Additionally, current automation solutions often lack effective real-time feedback and monitoring capabilities. Users are not always provided with clear information regarding system status, signal interpretation, or appliance conditions, making it difficult to verify whether their input has been correctly recognized. The absence of integrated alert mechanisms further reduces system effectiveness in critical situations. These limitations collectively indicate the need for a reliable, multi-modal automation system that ensures accurate control,

continuous monitoring, and enhanced safety through integrated feedback and alert features.

4. METHODOLOGY

The proposed Brainwave-Based IoT Home Automation System is designed as a multi-layered, real-time control framework that integrates brain signal acquisition, data processing, intelligent decision-making, and device actuation. The system follows a modular and scalable architecture in which each layer performs a dedicated function while maintaining continuous interaction with other layers. This structured design ensures efficient handling of noisy EEG signals, accurate interpretation of user intent, and reliable execution of control actions. The workflow operates in a closed-loop manner, allowing continuous monitoring, validation, and feedback for improved system performance.

1. Signal Acquisition Layer

The signal acquisition layer is responsible for capturing real-time brainwave data from the user using an EEG headband. The headset measures electrical activity generated by neurons and internally processes raw EEG signals to extract parameters such as attention and meditation levels. These values are transmitted wirelessly via Bluetooth to the ESP32 microcontroller at regular intervals. Since EEG signals are highly sensitive and prone to fluctuations, maintaining stable sensor contact and minimizing motion artefacts is essential. The continuous and real-time nature of this layer ensures that the system always has updated input data for further processing and control.

2. Data Processing and Filtering Layer

After acquisition, the EEG data is received by the ESP32 and subjected to preprocessing and filtering. This layer focuses on improving signal quality by reducing noise and eliminating sudden fluctuations that may arise due to external disturbances or unintended cognitive activity. Threshold-based filtering is applied to identify stable signal regions, while temporal consistency checks are used to ensure that the signal remains above a threshold for a specific duration before being considered valid. This prevents rapid switching and enhances the stability of the system. The processing is performed in real time, ensuring minimal latency between input and response. Decision-Making and Control Logic Layer.

The decision-making layer interprets the filtered EEG data and maps it to corresponding control actions. The system uses predefined threshold values and logical conditions to determine whether a signal represents an intentional command. For instance, a sustained attention level above a certain threshold is treated as a trigger to toggle an appliance. Additional logic is implemented to avoid repeated triggering from continuous high signals by introducing state-based control and delay mechanisms. This layer also manages mode

selection, allowing the system to dynamically prioritize brainwave input or gesture input depending on signal reliability.

3. Gesture-Based Fallback Mechanism

To address the limitations of EEG signal instability, a gesture-based fallback mechanism is integrated into the system. This subsystem utilizes motion sensing components to detect hand gestures, which are then mapped to predefined control actions such as turning appliances ON or OFF. The system continuously evaluates the quality of EEG signals, and when instability or inconsistency is detected, it shifts control to gesture mode. This transition is handled seamlessly without interrupting system operation. The presence of this fallback mechanism significantly improves robustness and ensures that the system remains functional under all conditions.

4. Actuation and Device Control Layer

The actuation layer is responsible for executing control decisions through hardware components. The ESP32 microcontroller sends digital control signals to a relay module, which acts as a switching interface between low-voltage control circuitry and high-voltage electrical appliances. Based on the received signals, the relay activates or deactivates connected devices such as lights and fans. Proper electrical isolation is maintained to ensure user safety and protect the microcontroller from high-voltage exposure. The switching process is designed to be fast and reliable, ensuring immediate response to user commands.

5. Monitoring System Layer

This layer continuously monitors EEG signal patterns to detect abnormal or irregular activity. The system analyzes variations in signal values and identifies patterns that deviate significantly from normal ranges. This feature is particularly useful in assistive scenarios, where abnormal brain activity may indicate discomfort, stress, or potential health issues. The monitoring process runs in parallel with the control system, ensuring uninterrupted operation.

6. Feedback and IoT Integration Layer

The final layer provides real-time feedback and enhances user interaction through both local and remote interfaces. An LCD is used to show parameters such as control mode, EEG values, and appliance status, enabling users to understand system behaviour instantly. In addition, IoT integration is achieved using a web server hosted on the ESP32, where data is transmitted and accessed through the HTTP protocol. This allows users and caregivers to monitor system activity remotely through a web interface. System parameters such as signal values, control status, and device states are updated in real time, improving transparency and usability. The implementation of the IoT web server also enables future expansion, such as remote control, data logging, and system configuration through the web interface.

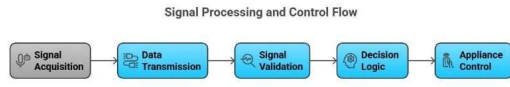


Fig 2: Proposed System Methodology Block Diagram

5. RESULT

System Overview

The developed system integrates EEG-based brainwave sensing with IoT-enabled automation to achieve real-time control of household appliances. The EEG headset captures brain signal parameters such as attention levels and transmits them via Bluetooth to the ESP32 microcontroller, where the signals are processed using threshold-based logic to generate control actions. A gesture-based module is incorporated as a fallback mechanism to ensure continuous operation when brain signals are unstable. The relay module executes switching of electrical appliances based on processed commands, while an LCD and IoT platform provide real-time feedback and remote monitoring. Additionally, the system includes an alert mechanism that detects abnormal signal variations and triggers notifications and buzzer alerts, enabling both automation and safety features within a unified framework.

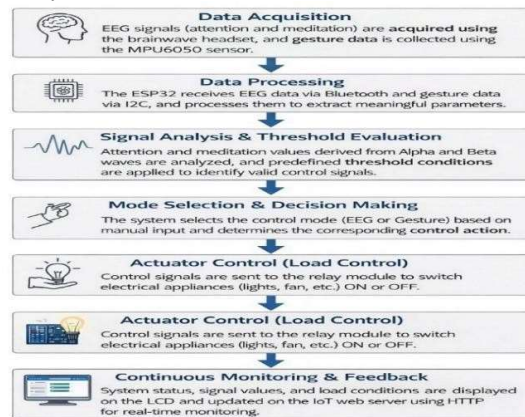


Fig 3: Overall System Workflow of Brainwave-Based Automation

System Architecture

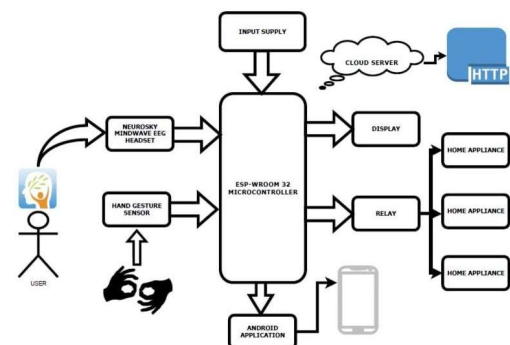


Fig 4: System Architecture of the Proposed BCI-IoT System

a. Signal Acquisition Unit

The unit uses the FT&S Mind Link headband to capture real-time brain signals such as attention levels, which are processed internally and transmitted via Bluetooth communication to the ESP32 controller.

b. Processing Unit

The ESP32 microcontroller receives EEG data through Bluetooth Serial and processes it using Arduino IDE-based embedded code, applying threshold logic and filtering techniques to interpret user intent and avoid false triggering.

c. Gesture Module

This utilizes the MPU6050 sensor to detect hand movements and orientation, where sensor data is processed through programmed logic to map gestures into appliance control commands when brain signals are unstable.

d. Actuation Unit

A relay module connected to the ESP32 GPIO pins is used to control electrical appliances, where the programmed switching logic turns devices ON or OFF based on processed inputs.

e. Feedback Unit

A 16×2 LCD with an I2C interface displays real-time parameters such as attention level, system mode, and appliance status, while the ESP32 hosts an IoT web server and uses the HTTP protocol to enable remote monitoring of system data through a web interface.

Hardware Implementation

The hardware implementation of the proposed system is centered around the ESP32 microcontroller, which acts as the main processing and control unit. An EEG headband is used to acquire brainwave signals such as attention levels, and these signals are transmitted to the ESP32 via Bluetooth communication. A 16×2 LCD module with an I2C interface is connected to the ESP32 to display real-time system parameters including signal values, control mode, and appliance status. For gesture-based fallback control, an MPU6050 sensor is interfaced with the ESP32 through I2C communication, enabling detection of hand movements and orientation. The control of electrical appliances is achieved using a relay module connected to the GPIO pins of the ESP32, allowing safe switching of loads such as lights or fans [17]. The ESP32 also establishes communication with a cloud server using the HTTP protocol through its built-in Wi-Fi capability, enabling transmission of system data for remote monitoring. All components are interconnected to ensure synchronized operation, forming a compact and efficient embedded system suitable for real-time home automation applications.

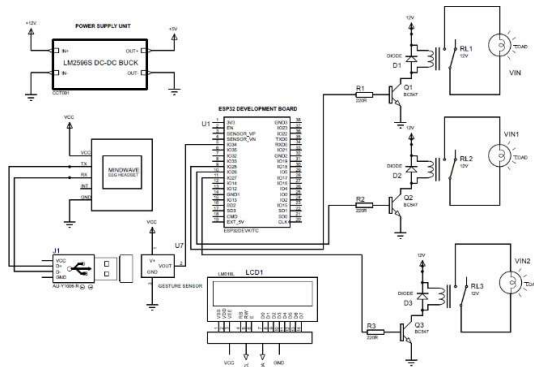


Fig 5: Circuit Diagram of Hardware Implementation

Software Implementation

The software implementation of the system is carried out using embedded C/C++ programming in the Arduino IDE, where the ESP32 microcontroller executes all control logic and communication tasks. The EEG data received via Bluetooth Serial is continuously read and processed to extract parameters such as attention level, which are evaluated using predefined threshold conditions to determine control actions. Sensor data from the MPU6050 is acquired through I2C communication and processed to detect gesture-based inputs, which are used as a fallback when EEG signals are unstable. The control logic incorporates filtering, delay mechanisms, and state-based conditions to ensure stable operation and prevent false triggering of appliances. GPIO pins are programmed to control the relay module for switching devices, while the LCD is updated in real time to display system status, signal values, and operating mode.

Additionally, the ESP32 hosts an IoT web server and uses HTTP communication over Wi-Fi to provide real-time system data through a web interface for remote monitoring. The overall software structure is designed to handle multiple inputs, prioritize control modes dynamically, and ensure synchronized operation between sensing, processing, and actuation components in real time.

Fig 6: Software of the System Operation Brainwave-Based Control System

EEG Signal Acquisition

The EEG headset was used to acquire real-time brain signals, including attention and meditation parameters derived from underlying brainwave activity. These parameters are primarily associated with Beta waves (linked to concentration and active thinking) and Alpha waves (associated with relaxation and calm mental states). The acquired data was transmitted to the ESP32 microcontroller via Bluetooth communication for further processing. Continuous monitoring of these parameters revealed that attention and meditation values varied

```

1: BluetoothSerial bl;
2: LiquidCrystal_IK lcd(16,2);
3: // EEG Mode
4: struct {
5:   float * mindwave;
6:   float * attention;
7:   float * meditation;
8:   float * focus;
9:   float * stress;
10: } data;
11: // Variables
12: const int MPU_addr=0x68;
13: int16_t AcX,AcY,AcZ;
14: // Relay pins
15: #define RELAY1 25
16: #define RELAY2 26
17: #define RELAY3 27
18: #define RELAY4 33
19: // Mode button
20: #define MODE_BTN 16
21: // Pin definitions
22: #define VIN1 13
23: #define VIN2 14
24: #define VIN3 15
25: // Pin definitions
26: #define VIN1 13
27: #define VIN2 14
28: #define VIN3 15
29: // Pin definitions
30: #define VIN1 13
31: #define VIN2 14
32: #define VIN3 15
33: // Pin definitions
34: #define VIN1 13
35: #define VIN2 14
36: #define VIN3 15
37: // Pin definitions
38: #define VIN1 13
39: #define VIN2 14
40: #define VIN3 15
41: // Pin definitions
42: #define VIN1 13
43: #define VIN2 14
44: #define VIN3 15
45: // Pin definitions
46: #define VIN1 13
47: #define VIN2 14
48: #define VIN3 15
49: // Pin definitions
50: #define VIN1 13
51: #define VIN2 14
52: #define VIN3 15
53: // Pin definitions
54: #define VIN1 13
55: #define VIN2 14
56: #define VIN3 15
57: // Pin definitions
58: #define VIN1 13
59: #define VIN2 14
60: #define VIN3 15
61: // Pin definitions
62: #define VIN1 13
63: #define VIN2 14
64: #define VIN3 15
65: // Pin definitions
66: #define VIN1 13
67: #define VIN2 14
68: #define VIN3 15
69: // Pin definitions
70: #define VIN1 13
71: #define VIN2 14
72: #define VIN3 15
73: // Pin definitions
74: #define VIN1 13
75: #define VIN2 14
76: #define VIN3 15
77: // Pin definitions
78: #define VIN1 13
79: #define VIN2 14
80: #define VIN3 15
81: // Pin definitions
82: #define VIN1 13
83: #define VIN2 14
84: #define VIN3 15
85: // Pin definitions
86: #define VIN1 13
87: #define VIN2 14
88: #define VIN3 15
89: // Pin definitions
90: #define VIN1 13
91: #define VIN2 14
92: #define VIN3 15
93: // Pin definitions
94: #define VIN1 13
95: #define VIN2 14
96: #define VIN3 15
97: // Pin definitions
98: #define VIN1 13
99: #define VIN2 14
100: #define VIN3 15
101: // Pin definitions
102: #define VIN1 13
103: #define VIN2 14
104: #define VIN3 15
105: // Pin definitions
106: #define VIN1 13
107: #define VIN2 14
108: #define VIN3 15
109: // Pin definitions
110: #define VIN1 13
111: #define VIN2 14
112: #define VIN3 15
113: // Pin definitions
114: #define VIN1 13
115: #define VIN2 14
116: #define VIN3 15
117: // Pin definitions
118: #define VIN1 13
119: #define VIN2 14
120: #define VIN3 15
121: // Pin definitions
122: #define VIN1 13
123: #define VIN2 14
124: #define VIN3 15
125: // Pin definitions
126: #define VIN1 13
127: #define VIN2 14
128: #define VIN3 15
129: // Pin definitions
130: #define VIN1 13
131: #define VIN2 14
132: #define VIN3 15
133: // Pin definitions
134: #define VIN1 13
135: #define VIN2 14
136: #define VIN3 15
137: // Pin definitions
138: #define VIN1 13
139: #define VIN2 14
140: #define VIN3 15
141: // Pin definitions
142: #define VIN1 13
143: #define VIN2 14
144: #define VIN3 15
145: // Pin definitions
146: #define VIN1 13
147: #define VIN2 14
148: #define VIN3 15
149: // Pin definitions
150: #define VIN1 13
151: #define VIN2 14
152: #define VIN3 15
153: // Pin definitions
154: #define VIN1 13
155: #define VIN2 14
156: #define VIN3 15
157: // Pin definitions
158: #define VIN1 13
159: #define VIN2 14
160: #define VIN3 15
161: // Pin definitions
162: #define VIN1 13
163: #define VIN2 14
164: #define VIN3 15
165: // Pin definitions
166: #define VIN1 13
167: #define VIN2 14
168: #define VIN3 15
169: // Pin definitions
170: #define VIN1 13
171: #define VIN2 14
172: #define VIN3 15
173: // Pin definitions
174: #define VIN1 13
175: #define VIN2 14
176: #define VIN3 15
177: // Pin definitions
178: #define VIN1 13
179: #define VIN2 14
180: #define VIN3 15
181: // Pin definitions
182: #define VIN1 13
183: #define VIN2 14
184: #define VIN3 15
185: // Pin definitions
186: #define VIN1 13
187: #define VIN2 14
188: #define VIN3 15
189: // Pin definitions
190: #define VIN1 13
191: #define VIN2 14
192: #define VIN3 15
193: // Pin definitions
194: #define VIN1 13
195: #define VIN2 14
196: #define VIN3 15
197: // Pin definitions
198: #define VIN1 13
199: #define VIN2 14
200: #define VIN3 15
201: // Pin definitions
202: #define VIN1 13
203: #define VIN2 14
204: #define VIN3 15
205: // Pin definitions
206: #define VIN1 13
207: #define VIN2 14
208: #define VIN3 15
209: // Pin definitions
210: #define VIN1 13
211: #define VIN2 14
212: #define VIN3 15
213: // Pin definitions
214: #define VIN1 13
215: #define VIN2 14
216: #define VIN3 15
217: // Pin definitions
218: #define VIN1 13
219: #define VIN2 14
220: #define VIN3 15
221: // Pin definitions
222: #define VIN1 13
223: #define VIN2 14
224: #define VIN3 15
225: // Pin definitions
226: #define VIN1 13
227: #define VIN2 14
228: #define VIN3 15
229: // Pin definitions
230: #define VIN1 13
231: #define VIN2 14
232: #define VIN3 15
233: // Pin definitions
234: #define VIN1 13
235: #define VIN2 14
236: #define VIN3 15
237: // Pin definitions
238: #define VIN1 13
239: #define VIN2 14
240: #define VIN3 15
241: // Pin definitions
242: #define VIN1 13
243: #define VIN2 14
244: #define VIN3 15
245: // Pin definitions
246: #define VIN1 13
247: #define VIN2 14
248: #define VIN3 15
249: // Pin definitions
250: #define VIN1 13
251: #define VIN2 14
252: #define VIN3 15
253: // Pin definitions
254: #define VIN1 13
255: #define VIN2 14
256: #define VIN3 15
257: // Pin definitions
258: #define VIN1 13
259: #define VIN2 14
260: #define VIN3 15
261: // Pin definitions
262: #define VIN1 13
263: #define VIN2 14
264: #define VIN3 15
265: // Pin definitions
266: #define VIN1 13
267: #define VIN2 14
268: #define VIN3 15
269: // Pin definitions
270: #define VIN1 13
271: #define VIN2 14
272: #define VIN3 15
273: // Pin definitions
274: #define VIN1 13
275: #define VIN2 14
276: #define VIN3 15
277: // Pin definitions
278: #define VIN1 13
279: #define VIN2 14
280: #define VIN3 15
281: // Pin definitions
282: #define VIN1 13
283: #define VIN2 14
284: #define VIN3 15
285: // Pin definitions
286: #define VIN1 13
287: #define VIN2 14
288: #define VIN3 15
289: // Pin definitions
290: #define VIN1 13
291: #define VIN2 14
292: #define VIN3 15
293: // Pin definitions
294: #define VIN1 13
295: #define VIN2 14
296: #define VIN3 15
297: // Pin definitions
298: #define VIN1 13
299: #define VIN2 14
300: #define VIN3 15
301: // Pin definitions
302: #define VIN1 13
303: #define VIN2 14
304: #define VIN3 15
305: // Pin definitions
306: #define VIN1 13
307: #define VIN2 14
308: #define VIN3 15
309: // Pin definitions
310: #define VIN1 13
311: #define VIN2 14
312: #define VIN3 15
313: // Pin definitions
314: #define VIN1 13
315: #define VIN2 14
316: #define VIN3 15
317: // Pin definitions
318: #define VIN1 13
319: #define VIN2 14
320: #define VIN3 15
321: // Pin definitions
322: #define VIN1 13
323: #define VIN2 14
324: #define VIN3 15
325: // Pin definitions
326: #define VIN1 13
327: #define VIN2 14
328: #define VIN3 15
329: // Pin definitions
330: #define VIN1 13
331: #define VIN2 14
332: #define VIN3 15
333: // Pin definitions
334: #define VIN1 13
335: #define VIN2 14
336: #define VIN3 15
337: // Pin definitions
338: #define VIN1 13
339: #define VIN2 14
340: #define VIN3 15
341: // Pin definitions
342: #define VIN1 13
343: #define VIN2 14
344: #define VIN3 15
345: // Pin definitions
346: #define VIN1 13
347: #define VIN2 14
348: #define VIN3 15
349: // Pin definitions
350: #define VIN1 13
351: #define VIN2 14
352: #define VIN3 15
353: // Pin definitions
354: #define VIN1 13
355: #define VIN2 14
356: #define VIN3 15
357: // Pin definitions
358: #define VIN1 13
359: #define VIN2 14
360: #define VIN3 15
361: // Pin definitions
362: #define VIN1 13
363: #define VIN2 14
364: #define VIN3 15
365: // Pin definitions
366: #define VIN1 13
367: #define VIN2 14
368: #define VIN3 15
369: // Pin definitions
370: #define VIN1 13
371: #define VIN2 14
372: #define VIN3 15
373: // Pin definitions
374: #define VIN1 13
375: #define VIN2 14
376: #define VIN3 15
377: // Pin definitions
378: #define VIN1 13
379: #define VIN2 14
380: #define VIN3 15
381: // Pin definitions
382: #define VIN1 13
383: #define VIN2 14
384: #define VIN3 15
385: // Pin definitions
386: #define VIN1 13
387: #define VIN2 14
388: #define VIN3 15
389: // Pin definitions
390: #define VIN1 13
391: #define VIN2 14
392: #define VIN3 15
393: // Pin definitions
394: #define VIN1 13
395: #define VIN2 14
396: #define VIN3 15
397: // Pin definitions
398: #define VIN1 13
399: #define VIN2 14
400: #define VIN3 15
401: // Pin definitions
402: #define VIN1 13
403: #define VIN2 14
404: #define VIN3 15
405: // Pin definitions
406: #define VIN1 13
407: #define VIN2 14
408: #define VIN3 15
409: // Pin definitions
410: #define VIN1 13
411: #define VIN2 14
412: #define VIN3 15
413: // Pin definitions
414: #define VIN1 13
415: #define VIN2 14
416: #define VIN3 15
417: // Pin definitions
418: #define VIN1 13
419: #define VIN2 14
420: #define VIN3 15
421: // Pin definitions
422: #define VIN1 13
423: #define VIN2 14
424: #define VIN3 15
425: // Pin definitions
426: #define VIN1 13
427: #define VIN2 14
428: #define VIN3 15
429: // Pin definitions
430: #define VIN1 13
431: #define VIN2 14
432: #define VIN3 15
433: // Pin definitions
434: #define VIN1 13
435: #define VIN2 14
436: #define VIN3 15
437: // Pin definitions
438: #define VIN1 13
439: #define VIN2 14
440: #define VIN3 15
441: // Pin definitions
442: #define VIN1 13
443: #define VIN2 14
444: #define VIN3 15
445: // Pin definitions
446: #define VIN1 13
447: #define VIN2 14
448: #define VIN3 15
449: // Pin definitions
450: #define VIN1 13
451: #define VIN2 14
452: #define VIN3 15
453: // Pin definitions
454: #define VIN1 13
455: #define VIN2 14
456: #define VIN3 15
457: // Pin definitions
458: #define VIN1 13
459: #define VIN2 14
460: #define VIN3 15
461: // Pin definitions
462: #define VIN1 13
463: #define VIN2 14
464: #define VIN3 15
465: // Pin definitions
466: #define VIN1 13
467: #define VIN2 14
468: #define VIN3 15
469: // Pin definitions
470: #define VIN1 13
471: #define VIN2 14
472: #define VIN3 15
473: // Pin definitions
474: #define VIN1 13
475: #define VIN2 14
476: #define VIN3 15
477: // Pin definitions
478: #define VIN1 13
479: #define VIN2 14
480: #define VIN3 15
481: // Pin definitions
482: #define VIN1 13
483: #define VIN2 14
484: #define VIN3 15
485: // Pin definitions
486: #define VIN1 13
487: #define VIN2 14
488: #define VIN3 15
489: // Pin definitions
490: #define VIN1 13
491: #define VIN2 14
492: #define VIN3 15
493: // Pin definitions
494: #define VIN1 13
495: #define VIN2 14
496: #define VIN3 15
497: // Pin definitions
498: #define VIN1 13
499: #define VIN2 14
500: #define VIN3 15
501: // Pin definitions
502: #define VIN1 13
503: #define VIN2 14
504: #define VIN3 15
505: // Pin definitions
506: #define VIN1 13
507: #define VIN2 14
508: #define VIN3 15
509: // Pin definitions
510: #define VIN1 13
511: #define VIN2 14
512: #define VIN3 15
513: // Pin definitions
514: #define VIN1 13
515: #define VIN2 14
516: #define VIN3 15
517: // Pin definitions
518: #define VIN1 13
519: #define VIN2 14
520: #define VIN3 15
521: // Pin definitions
522: #define VIN1 13
523: #define VIN2 14
524: #define VIN3 15
525: // Pin definitions
526: #define VIN1 13
527: #define VIN2 14
528: #define VIN3 15
529: // Pin definitions
530: #define VIN1 13
531: #define VIN2 14
532: #define VIN3 15
533: // Pin definitions
534: #define VIN1 13
535: #define VIN2 14
536: #define VIN3 15
537: // Pin definitions
538: #define VIN1 13
539: #define VIN2 14
540: #define VIN3 15
541: // Pin definitions
542: #define VIN1 13
543: #define VIN2 14
544: #define VIN3 15
545: // Pin definitions
546: #define VIN1 13
547: #define VIN2 14
548: #define VIN3 15
549: // Pin definitions
550: #define VIN1 13
551: #define VIN2 14
552: #define VIN3 15
553: // Pin definitions
554: #define VIN1 13
555: #define VIN2 14
556: #define VIN3 15
557: // Pin definitions
558: #define VIN1 13
559: #define VIN2 14
560: #define VIN3 15
561: // Pin definitions
562: #define VIN1 13
563: #define VIN2 14
564: #define VIN3 15
565: // Pin definitions
566: #define VIN1 13
567: #define VIN2 14
568: #define VIN3 15
569: // Pin definitions
570: #define VIN1 13
571: #define VIN2 14
572: #define VIN3 15
573: // Pin definitions
574: #define VIN1 13
575: #define VIN2 14
576: #define VIN3 15
577: // Pin definitions
578: #define VIN1 13
579: #define VIN2 14
580: #define VIN3 15
581: // Pin definitions
582: #define VIN1 13
583: #define VIN2 14
584: #define VIN3 15
585: // Pin definitions
586: #define VIN1 13
587: #define VIN2 14
588: #define VIN3 15
589: // Pin definitions
590: #define VIN1 13
591: #define VIN2 14
592: #define VIN3 15
593: // Pin definitions
594: #define VIN1 13
595: #define VIN2 14
596: #define VIN3 15
597: // Pin definitions
598: #define VIN1 13
599: #define VIN2 14
600: #define VIN3 15
601: // Pin definitions
602: #define VIN1 13
603: #define VIN2 14
604: #define VIN3 15
605: // Pin definitions
606: #define VIN1 13
607: #define VIN2 14
608: #define VIN3 15
609: // Pin definitions
610: #define VIN1 13
611: #define VIN2 14
612: #define VIN3 15
613: // Pin definitions
614: #define VIN1 13
615: #define VIN2 14
616: #define VIN3 15
617: // Pin definitions
618: #define VIN1 13
619: #define VIN2 14
620: #define VIN3 15
621: // Pin definitions
622: #define VIN1 13
623: #define VIN2 14
624: #define VIN3 15
625: // Pin definitions
626: #define VIN1 13
627: #define VIN2 14
628: #define VIN3 15
629: // Pin definitions
630: #define VIN1 13
631: #define VIN2 14
632: #define VIN3 15
633: // Pin definitions
634: #define VIN1 13
635: #define VIN2 14
636: #define VIN3 15
637: // Pin definitions
638: #define VIN1 13
639: #define VIN2 14
640: #define VIN3 15
641: // Pin definitions
642: #define VIN1 13
643: #define VIN2 14
644: #define VIN3 15
645: // Pin definitions
646: #define VIN1 13
647: #define VIN2 14
648: #define VIN3 15
649: // Pin definitions
650: #define VIN1 13
651: #define VIN2 14
652: #define VIN3 15
653: // Pin definitions
654: #define VIN1 13
655: #define VIN2 14
656: #define VIN3 15
657: // Pin definitions
658: #define VIN1 13
659: #define VIN2 14
660: #define VIN3 15
661: // Pin definitions
662: #define VIN1 13
663: #define VIN2 14
664: #define VIN3 15
665: // Pin definitions
666: #define VIN1 13
667: #define VIN2 14
668: #define VIN3 15
669: // Pin definitions
670: #define VIN1 13
671: #define VIN2 14
672: #define VIN3 15
673: // Pin definitions
674: #define VIN1 13
675: #define VIN2 14
676: #define VIN3 15
677: // Pin definitions
678: #define VIN1 13
679: #define VIN2 14
680: #define VIN3 15
681: // Pin definitions
682: #define VIN1 13
683: #define VIN2 14
684: #define VIN3 15
685: // Pin definitions
686: #define VIN1 13
687: #define VIN2 14
688: #define VIN3 15
689: // Pin definitions
690: #define VIN1 13
691: #define VIN2 14
692: #define VIN3 15
693: // Pin definitions
694: #define VIN1 13
695: #define VIN2 14
696: #define VIN3 15
697: // Pin definitions
698: #define VIN1 13
699: #define VIN2 14
700: #define VIN3 15
701: // Pin definitions
702: #define VIN1 13
703: #define VIN2 14
704: #define VIN3 15
705: // Pin definitions
706: #define VIN1 13
707: #define VIN2 14
708: #define VIN3 15
709: // Pin definitions
710: #define VIN1 13
711: #define VIN2 14
712: #define VIN3 15
713: // Pin definitions
714: #define VIN1 13
715: #define VIN2 14
716: #define VIN3 15
717: // Pin definitions
718: #define VIN1 13
719: #define VIN2 14
720: #define VIN3 15
721: // Pin definitions
722: #define VIN1 13
723: #define VIN2 14
724: #define VIN3 15
725: // Pin definitions
726: #define VIN1 13
727: #define VIN2 14
728: #define VIN3 15
729: // Pin definitions
730: #define VIN1 13
731: #define VIN2 14
732: #define VIN3 15
733: // Pin definitions
734: #define VIN1 13
735: #define VIN2 14
736: #define VIN3 15
737: // Pin definitions
738: #define VIN1 13
739: #define VIN2 14
740: #define VIN3 15
741: // Pin definitions
742: #define VIN1 13
743: #define VIN2 14
744: #define VIN3 15
745: // Pin definitions
746: #define VIN1 13
747: #define VIN2 14
748: #define VIN3 15
749: // Pin definitions
750: #define VIN1 13
751: #define VIN2 14
752: #define VIN3 15
753: // Pin definitions
754: #define VIN1 13
755: #define VIN2 14
756: #define VIN3 15
757: // Pin definitions
758: #define VIN1 13
759: #define VIN2 14
760: #define VIN3 15
761: // Pin definitions
762: #define VIN1 13
763: #define VIN2 14
764: #define VIN3 15
765: // Pin definitions
766: #define VIN1 13
767: #define VIN2 14
768: #define VIN3 15
769: // Pin definitions
770: #define VIN1 13
771: #define VIN2 14
772: #define VIN3 15
773: // Pin definitions
774: #define VIN1 13
775: #define VIN2 14
776: #define VIN3 15
777: // Pin definitions
778: #define VIN1 13
779: #define VIN2 14
780: #define VIN3 15
781: // Pin definitions
782: #define VIN1 13
783: #define VIN2 14
784: #define VIN3 15
785: // Pin definitions
786: #define VIN1 13
787: #define VIN2 14
788: #define VIN3 15
789: // Pin definitions
790: #define VIN1 13
791: #define VIN2 14
792: #define VIN3 15
793: // Pin definitions
794: #define VIN1 13
795: #define VIN2 14
796: #define VIN3 15
797: // Pin definitions
798: #define VIN1 13
799: #define VIN2 14
800: #define VIN3 15
801: // Pin definitions
802: #define VIN1 13
803: #define VIN2 14
804: #define VIN3 15
805: // Pin definitions
806: #define VIN1 13
807: #define VIN2 14
808: #define VIN3 15
809: // Pin definitions
810: #define VIN1 13
811: #define VIN2 14
812: #define VIN3 15
813: // Pin definitions
814: #define VIN1 13
815: #define VIN2 14
816: #define VIN3 15
817: // Pin definitions
818: #define VIN1 13
819: #define VIN2 14
820: #define VIN3 15
821: // Pin definitions
822: #define VIN1 13
823: #define VIN2 14
824: #define VIN3 15
825: // Pin definitions
826: #define VIN1 13
827: #define VIN2 14
828: #define VIN3 15
829: // Pin definitions
830: #define VIN1 13
831: #define VIN2 14
832: #define VIN3 15
833: // Pin definitions
834: #define VIN1 13
835: #define VIN2 14
836: #define VIN3 15
837: // Pin definitions
838: #define VIN1 13
839: #define VIN2 14
840: #define VIN3 15
841: // Pin definitions
842: #define VIN1 13
843: #define VIN2 14
844: #define VIN3 15
845: // Pin definitions
846: #define VIN1 13
847: #define VIN2 14
848: #define VIN3 15
849: // Pin definitions
850: #define VIN1 13
851: #define VIN2 14
852: #define VIN3 15
853: // Pin definitions
854: #define VIN1 13
855: #define VIN2 14
856: #define VIN3 15
857: // Pin definitions
858: #define VIN1 13
859: #define VIN2 14
860: #define VIN3 15
861: // Pin definitions
862: #define VIN1 13
863: #define VIN2 14
864: #define VIN3 15
865: // Pin definitions
866: #define VIN1 13
867: #define VIN2 14
868: #define VIN3 15
869: // Pin definitions
870: #define VIN1 13
871: #define VIN2 14
872: #define VIN3 15
873: // Pin definitions
874: #define VIN1 13
875: #define VIN2 14
876: #define VIN3 15
877: // Pin definitions
878: #define VIN1 13
879: #define VIN2 14
880: #define VIN3 15
881: // Pin definitions
882: #define VIN1 13
883: #define VIN2 14
884: #define VIN3 15
885: // Pin definitions
886: #define VIN1 13
887: #define VIN2 14
888: #define VIN3 15
889: // Pin definitions
890: #define VIN1 13
891: #define VIN2 14
892: #define VIN3 15
893: // Pin definitions
894: #define VIN1 13
895: #define VIN2 14
896: #define VIN3 15
897: // Pin definitions
898: #define VIN1 13
899: #define VIN2 14
900: #define VIN3 15
901: // Pin definitions
902: #define VIN1 13
903: #define VIN2 14
904: #define VIN3 15
905: // Pin definitions
906: #define VIN1 13
907: #define VIN2 14
908: #define VIN3 15
909: // Pin definitions
910: #define VIN1 13
911: #define VIN2 14
912: #define VIN3 15
913: // Pin definitions
914: #define VIN1 13
915: #define VIN2 14
916: #define VIN3 15
917: // Pin definitions
918: #define VIN1 13
919: #define VIN2 14
920: #define VIN3 15
921: // Pin definitions
922: #define VIN1 13
923: #define VIN2 14
924: #define VIN3 15
925: // Pin definitions
926: #define VIN1 13
927: #define VIN2 14
928: #define VIN3 15
929: // Pin definitions
930: #define VIN1 13
931: #define VIN2 14
932: #define VIN3 15
933: // Pin definitions
934: #define VIN1 13
935: #define VIN2 14
936: #define VIN3 15
937: // Pin definitions
938: #define VIN1 13
939: #define VIN2 14
940: #define VIN3 15
941: // Pin definitions
942: #define VIN1 13
943: #define VIN2 14
944: #define VIN3 15
945: // Pin definitions
946: #define VIN1 13
947: #define VIN2 14
948: #define VIN3 15
949: // Pin definitions
950: #define VIN1 13
951: #define VIN2 14
952: #define VIN3 15
953: // Pin definitions
954: #define VIN1 13
955: #define VIN2 14
956: #define VIN3 15
957: // Pin definitions
958: #define VIN1 13
959: #define VIN2 14
960: #define VIN3 15
961: // Pin definitions
962: #define VIN1 13
963: #define VIN2 14
964: #define VIN3 15
965: // Pin definitions
966: #define VIN1 13
967: #define VIN2 14
968: #define VIN3 15
969: // Pin definitions
970: #define VIN1 13
971: #define VIN2 14
972: #define VIN3 15
973: // Pin definitions
974: #define VIN1 13
975: #define VIN2 14
976: #define VIN3 15
977: // Pin definitions
978: #define VIN1 13
979: #define VIN2 14
980: #define VIN3 15
981: // Pin definitions
982: #define VIN1 13
983: #define VIN2 14
984: #define VIN3 15
985: // Pin definitions
986: #define VIN1 13
987: #define VIN2 14
988: #define VIN3 15
989: // Pin definitions
990: #define VIN1 13
991: #define VIN2 14
992: #define VIN3 15
993: // Pin definitions
994: #define VIN1 13
995: #define VIN2 14
996: #define VIN3 15
997: // Pin definitions
998: #define VIN1 13
999: #define VIN2 14
1000: #define VIN3 15

```

dynamically based on the user's cognitive state and level of focus.

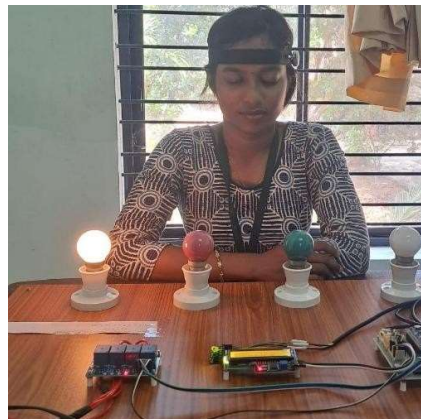


Fig 7: EEG Signal Acquisition and Brainwave Monitoring
EEG Signal Analysis and Threshold

Determination

The acquired EEG signals were analyzed to understand the variation of attention and meditation values corresponding to different brainwave patterns. Attention levels were observed to increase with higher Beta wave activity, while meditation levels showed higher values during dominant Alpha wave states. Based on these observations, threshold values were experimentally determined to distinguish intentional control signals from normal fluctuations. To enhance reliability, the system considered sustained signal levels above the threshold rather than instantaneous peaks, thereby reducing the impact of noise and unintended variations.

Brainwave-Based Load Control

The system utilized the processed EEG parameters to control electrical loads through a relay module. When the attention or meditation values satisfied the predefined threshold conditions for a specific duration, corresponding control actions were triggered. The mapping between brainwave-derived parameters and load control enabled hands-free operation of appliances. The system demonstrated stable performance under controlled conditions, with successful switching actions corresponding to consistent brain activity patterns.

Real-Time Feedback Using LCD

A 16×2 LCD was integrated to provide real-time visualization of EEG parameters, including attention and meditation values, along with system status. This enabled users to observe how variations in their mental state influenced system behaviour. The display served as an important interface for monitoring signal consistency and validating control actions during system operation.



Fig 8: EEG Signal Analysis and

Threshold Determination
Hybrid Multi-Model Control System

Gesture Signal Acquisition and Calibration

The MPU6050 sensor was used to acquire motion and orientation data corresponding to hand gestures. The sensor outputs, including acceleration and angular values, were continuously monitored to study variations for different hand movements. Based on experimental observation, distinct patterns were identified for specific gestures, and suitable threshold ranges were determined to differentiate

between control actions. This calibration enabled reliable detection of gestures under varying conditions.

Integration of EEG and Gesture Control

The EEG-based control system developed in Phase 1 was integrated with the gesture-based control module to form a unified system. Both input mechanisms were incorporated into a single control framework, allowing the system to process signals from either source. The integration ensured that the system could handle multiple input modalities while maintaining synchronization between sensing, processing, and actuation components.



Fig 9: Gesture-Based Control

Using MPU6050 Sensor

Multi-Load Control Implementation

The system was extended to control multiple electrical loads through the relay module. A total of four loads were implemented, where two loads were associated with brainwave-based control and two loads were assigned for gesture-based control. The control logic was designed to map specific input conditions to corresponding loads, enabling independent and flexible operation of appliances within the system.

Manual Mode Switching Mechanism

A manual mode switching mechanism was implemented to select between EEG-based control and gesture-based control. The system operates in EEG mode by default, and switching to gesture mode is performed through a manual input. This approach ensures controlled operation and prevents unintended switching between modes due to signal fluctuations. The selected mode is displayed on the LCD, allowing users to clearly identify the current control mechanism.



Fig 10: Real-Time Feedback
Display Using LCD

Real-Time System Operation and Feedback

The integrated system operates in real time, processing input signals and executing control actions with minimal delay. The 16×2 LCD provides continuous feedback, including control mode, signal values, and load status. This enables users to monitor system behaviour and interact effectively with the system during operation. The coordinated functioning of multiple modules ensures smooth and responsive performance.

IoT Web Server Communication Using HTTP Protocol

The ESP32 hosts an IoT-based web server and communicates using the HTTP protocol through its built-in Wi-Fi capability. The system transmits real-time data, including EEG parameters, control mode, and load status, to the web interface, allowing users to monitor system operation remotely through a browser. HTTP requests are used for data exchange between the ESP32 and the web server, enabling efficient and lightweight communication. This implementation enhances system accessibility and provides a simple platform for remote monitoring and future expansion of control features.

Overall Result

The overall system demonstrated successful implementation of a brainwave-based and gesture-assisted IoT home automation framework with reliable real-time performance. The EEG-based control system was able to interpret user intent through attention and meditation parameters derived from Beta and Alpha wave activity, enabling hands-free operation of electrical loads. The use of threshold-based logic with temporal validation ensured stable performance while minimizing false triggering due to signal fluctuations.

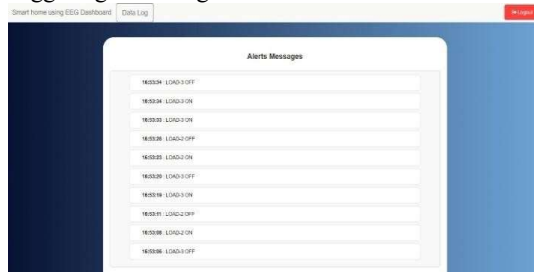


Fig 11: Multi-Load Control

Implementation and System Output

The integration of gesture control using the MPU6050 sensor enhanced system flexibility by providing an alternative control mechanism, allowing the system to operate effectively under varying conditions. The implementation of multi-load control, with separate loads assigned for EEG and gesture inputs, enabled independent and efficient appliance management. The manual mode switching mechanism ensured controlled operation and prevented unintended transitions between input modes.

The system operated in real time with consistent response, supported by continuous feedback through

the LCD and remote monitoring via the IoT web server using HTTP communication. The combined functionality of signal processing, multi-modal control, and IoT-based monitoring demonstrates the effectiveness of the system in providing a practical, reliable, and scalable solution for smart home automation.

6. FUTURE SCOPE

The capacity to interpret and adapt to brain signals can be improved to upgrade the proposed system. Currently, the system's decision-making is threshold-based. More advanced techniques could involve machine learning and adaptive algorithms to recognize EEG patterns. Such techniques could allow the system to adjust to the brain patterns of the user, distinguishing between voluntary and involuntary actions. Furthermore, the use of advanced signal processing techniques such as filtering, artifact removal, and feature extraction from EEG data may improve the accuracy and precision of the proposed system. Advanced EEG equipment and devices that offer better electrodes' precision and stability can also be used to achieve better performance in real-life settings. In the future, the system could be expanded to control a larger number of devices. The system can be expanded by adding more IoT-enabled devices to manage and monitor a smart home. The web server could also be more interactive, with more options for viewing and interacting with collected data from the system. It would be interesting to also add data storage capabilities so that users could see long-term trends in how the system behaves. Moreover, additional functionalities could be provided to the system which could make it more intuitive and user-friendly. The gesture control module could also benefit from more sophisticated sensors or visual input to read more types of gestures. It could also automatically switch between gesture and brain control based on whether a reliable brain signal is detected, without the need to manually switch control modes. In the future, the system could also provide health monitoring functionalities, such as reading the brain activity to see the stress level or detect abnormalities. This would make the system a dual purpose tool which can manage as well as monitor a person with disabilities. Another idea is to make the system more compact and energy efficient, portable, which would allow it to support more practical applications in the future.

❖ CONCLUSION

The proposed Brainwave-Based IoT Home Automation System is a feasible solution for hands-free control of electrical appliances using EEG signals. The attention and meditation parameters extracted from the brainwave signals were successfully utilized to develop a control system that effectively decodes and translates the intent from brain signals into control commands. The thresholding of signals along with time domain

filtering was effective in suppressing erratic signals and ensuring reliable control of the system. The findings make it possible to consider further development of this system and consider EEG signals as a suitable input method for controlling home appliances.

The addition of gesture recognition was effective in adding another control methodology for the given application. The integration of an MPU6050 sensor is a good option for this particular application. The multi-load control function in the project is a suitable demonstration of control of multiple appliances using EEG and gesture inputs. Similarly, the manual switching between different control inputs ensures reliable switching between control methods.

The feedback provided by the LCD and IoT web server demonstrates the functionality of having multiple displays to provide updated information about the status of the application. Both the local and remote monitoring options can offer a more advanced level of interaction with the brain-computer interface. Overall, the proposed system is a great application of current technology that provides a solution for a specific need.

ACKNOWLEDGEMENT

We acknowledge the support given by Adithya Pallavi S P, Anudhyana A S, and Devana A S, students of the Electronics and Computer Engineering Program, LBS Institute of Technology for Women, Thiruvananthapuram.

References

- [1] X. Zhang, C. Wang, L. Zhang, X. Qin, and X. Liu, "Internet of Things Meets Brain–Computer Interface: A Unified Deep Learning Framework for Enabling Human-Thing Cognitive Interactivity," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2084–2092, 2019.
- [2] J. W. Bang, E. C. Lee, and K. R. Park, "New Computer Interface Combining Gaze Tracking and Brainwave Measurements," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 4, pp. 1646–1651, 2011.
- [3] X. Gu, Z. Cao, A. Jolfaei, P. Xu, D. Wu, T.-P. Jung, and C.-T. Lin, "EEG-Based Brain-Computer Interfaces (BCIs): A Survey of Recent Studies on Signal Sensing Technologies and Computational Intelligence Approaches and Their Applications," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 5, pp. 1645–1666, 2021.
- [4] D. Wu, Y. Xu, and B.-L. Lu, "Transfer Learning for EEG-Based Brain–Computer Interfaces: A Review of Progress Made Since 2016," *IEEE Transactions on Cognitive and Developmental Systems*, 2022.
- [5] M. Maleki, N. Manshour, and T. Kayikcioglu, "Brain–Computer Interface Systems for Smart Homes—A Review Study," *Recent Advances in Electrical & Electronic Engineering*, vol. 14, no. 2, pp. 144–155, 2021.
- [6] R. Ajmeria *et al.*, "A Critical Survey of EEG-Based BCI Systems for Applications in Industrial Internet of Things," *IEEE Communications Surveys & Tutorials*, 2022.
- [7] A. Zielonka, M. Woźniak, S. Garg, G. Kaddoum, M. J. Piran, and G. Muhammad, "Smart Homes: How Much Will They Support Us? A Research on Recent Trends and Advances," *IEEE Access*, vol. 9, pp. 26388–26419, 2021.
- [8] S. Park, H.-S. Cha, and C.-H. Im, "Development of an Online Home Appliance Control System Using Augmented Reality and an SSVEP-Based Brain–Computer Interface," *IEEE Access*, vol. 7, pp. 163604–163614, 2019.
- [9] F. Putze, D. Weiss, L.-M. Vortmann, and T. Schultz, "Augmented Reality Interface for Smart Home Control Using SSVEP-BCI and Eye Gaze," in *Proc. IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2019.
- [10] X. Gu *et al.*, "EEG-Based Brain–Computer Interfaces: Computational Intelligence and Deep Learning Techniques," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 5, pp. 1645–1666, 2021.
- [11] A. Al-Saegh *et al.*, "Deep Learning for Motor Imagery EEG-Based Classification: A Review," *Biomedical Signal Processing and Control*, vol. 63, 2021.
- [12] S. Aggarwal and N. Chugh, "Review of Machine Learning Techniques for EEG-Based Brain–Computer Interface," *Archives of Computational Methods in Engineering*, vol. 29, pp. 3001–3020, 2022.
- [13] M. A. Al-Qaysi, M. M. Zaidan, B. B. Zaidan, and A. T. Albahri, "Noninvasive Electroencephalography Equipment for Assistive, Adaptive, and Rehabilitative Brain–Computer Interfaces: A Systematic Literature Review," *Sensors*, vol. 21, no. 14, 4754, 2021.
- [14] C. Flores Vega *et al.*, "Fuzzy Temporal Convolutional Neural Networks in P300-Based Brain–Computer Interface for Smart Home Interaction," 2022.
- [15] W.-Y. Hsu, "EEG-Based Motor Imagery Classification Using Neuro-Fuzzy Prediction and Wavelet Fractal Features," *Biomedical Signal Processing and Control*, vol. 49, pp. 246–252, 2019.
- [16] M. S. B. Abd Rani *et al.*, "Detection of Eye Blinks from EEG Signals for Home Lighting System Activation," in *Proc. IEEE International Symposium on Mechatronics and Its Applications*, 2009.
- [17] S. Abdulwahab *et al.*, "A Survey in Implementation and Applications of Electroencephalograph (EEG)-Based Brain–Computer Interface," *Engineering and Technology Journal*, vol. 39, 2021.
- [18] P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, *Introduction to EEG- and Speech-Based Emotion Recognition*, Academic Press, 2016.
- [19] A. Al-Fahoum and A. Al-Fraihat, "Methods

of EEG Signal Features Extraction Using Linear Analysis in Frequency and Time-Frequency Domains," *ISRN Neuroscience*, 2014.

[18] S. K. Gouda, A. Choudhry, S. P. Satpathy, K. M. Shukla, A. K. Dash, and A. K. Pasayat, "Integration of EEG-Based BCI Technology in IoT Enabled Smart Home Environment: An In-Depth Comparative Analysis on Human–Computer Interaction Techniques," *Expert Systems with Applications*, vol. 294, 2025.

[19] A. Al-Fahoum and A. Al-Fraihat, "Methods of EEG Signal Features Extraction Using Linear Analysis in Frequency and Time-Frequency Domains," *ISRN Neuroscience*, 2014.

[20] S. K. Gouda, A. Choudhry, S. P. Satpathy, K. M. Shukla, A. K. Dash, and A. K. Pasayat, "Integration of EEG-Based BCI Technology in IoT Enabled Smart Home Environment: An In-Depth Comparative Analysis on Human–Computer Interaction Techniques," *Expert Systems with Applications*, vol. 294, 2025.