IOT Based Energy Management and Safety Control System

SIVASANKARI V, KALAIYARASU S , SAKTHIVEL B

Department of Artificial Intelligence and Data Science, Nehru Institute of Engineering and Technology, Coimbatore, Tamil Nadu 641005, India

MENTOR : **MS.SHANMUGAPRIYA**

Department of Artificial Intelligence and Data Science Nehru Institute of Engineering and Technology

ABSTRACT

Nowadays, energy management and monitoring are gaining significant attention due to increasing electricity demand and the need for efficient energy usage. Traditional energy meters often require manual readings and do not provide real-time information, which can lead to inefficiencies and wastage. The IoT-Based Smart Energy Meter allows direct measurement and wireless transmission of electrical parameters for improved monitoring and control. In addition, integrating sensors for current and voltage with the ESP8266 microcontroller enables real-time processing, display on an LCD, and remote monitoring via the ThingSpeak platform. The system provides alerts when energy consumption approaches predefined limits, and automatically disconnects the power supply using a relay module if usage exceeds the maximum threshold. This approach offers significant advantages in terms of energy efficiency, reduced wastage, electrical safety, and convenience for users.

This paper presents a system for real-time monitoring and control of electricity consumption using current and voltage sensors interfaced with ESP8266. The measured data is processed, displayed locally, and transmitted to a cloud platform for remote access. The proposed system demonstrates a practical implementation of smart energy management for households and small buildings. It can serve as a foundation for developing more advanced IoT-based energy monitoring and control solutions.

Furthermore, the system maintains historical energy consumption data on the cloud platform, allowing users to analyze usage patterns, identify high-consumption appliances, and make informed decisions to optimize energy usage and reduce electricity costs.

CHAPTER 1

INTRODUCTION

- Electricity consumption in households and small buildings is increasing rapidly due to the growing number of electrical appliances.
- Traditional energy meters do not provide real-time monitoring or automated control, which can lead to energy wastage and higher electricity bills.
- IoT technology enables devices to be connected and monitored remotely through mobile or web applications.
- The IoT-Based Smart Energy Meter measures electricity usage in real-time using current and voltage sensors.
- The system transmits energy consumption data to a cloud platform for remote monitoring, allowing users to track usage anytime and anywhere.
- Alerts are generated when energy consumption approaches predefined limits, helping users stay within safe usage thresholds.
- If energy usage exceeds the maximum limit, a relay module automatically disconnects the power supply to prevent wastage and ensure safety.

- This approach provides efficient energy management, reduces unnecessary electricity consumption, and enhances electrical safety in households and small buildings.

- The proposed system demonstrates the practical application of IoT technology in smart energy monitoring, automated control, and remote management.

- Therefore, a reliable system that measures, processes, and controls energy consumption in real-time is essential for modern energy efficiency.

OVERVIEW

Nowadays, IoT technology is gaining mainstream attention due to its potential in efficient energy monitoring and management. With the increasing number of electrical appliances in households and small buildings, traditional energy meters are no longer sufficient, as they do not provide real-time monitoring or automated control. The use of current and voltage sensors allows direct measurement of electrical parameters, which can be processed to calculate energy consumption and power usage. Moreover, integrating microcontrollers like ESP8266 with LCD displays and IoT platforms enables real-time processing, visualization, and remote monitoring of electricity usage.

The proposed system works as follows: the current and voltage sensors continuously measure electrical parameters, which are then sent to the ESP8266 microcontroller for processing. The processed data are displayed on an LCD, providing immediate feedback to the user, while the same data is transmitted to an IoT platform such as ThingSpeak for remote monitoring. The system can detect high energy usage and send alerts efficiently, allowing users to manage electricity consumption proactively. When consumption exceeds predefined thresholds, the relay module automatically disconnects the power supply to prevent wastage and ensure safety. The overall system not only measures and monitors energy usage but also provides real-time feedback, automated control, and remote access. The detailed design and operation of the system are described in the following sections.

MOTIVATION

Efficient energy management is one of the most important aspects of modern households and small buildings. However, many people face challenges in monitoring electricity consumption accurately due to the limitations of traditional energy meters, which require manual readings and do not provide real-time information. This often leads to energy wastage, higher electricity bills, and potential safety hazards.

The IoT technology, combined with sensors and microcontrollers, provides a promising solution by enabling real-time monitoring, automated control, and remote access to energy usage data. This can significantly improve energy efficiency, reduce unnecessary electricity consumption, and enhance electrical safety.

The project to develop a system that can continuously measure voltage and current, process the data in real-time using an ESP8266 microcontroller, and provide immediate feedback to the user through an LCD display and cloud-based IoT platform. By integrating alerts and automatic power cut-off via a relay module, the system can proactively prevent energy wastage and unsafe conditions.

In addition, implementing a practical and cost-effective solution using readily available components makes smart energy management more accessible to everyday households. This motivates the development of an efficient, accurate, and user-friendly IoT-based Smart Energy Meter for real-time energy monitoring and control.

LITERATURE SURVEY

Smart Metering System for Real-Time Energy Monitoring-2025 Pandey, A[1], Saket, P[2], Bansal, V[3], Ahmed, M[4]Department of Electrical Engineering, XYZ University, India

In their study, Pandey et al. (2025) investigate the design and implementation of an IoT-based smart metering system aimed at real-time monitoring of electricity consumption in residential and small commercial buildings. The research addresses a significant gap in conventional energy management, as traditional meters fail to provide instantaneous feedback, automated alerts, or remote monitoring capabilities. These limitations often result in inefficient energy usage, increased electricity costs, and potential safety hazards in households and small-scale buildings.

The authors propose a comprehensive framework for smart energy monitoring that integrates IoT-enabled sensors, a microcontroller, and a cloud platform. The system employs current and voltage sensors to capture real-time electrical parameters, including instantaneous current, voltage, and active power. These measurements are then processed using an ESP8266 microcontroller, which calculates energy consumption and transmits the data to a cloud-based server for visualization and analysis. By leveraging cloud computing, the system allows users to monitor their energy usage remotely through web or mobile applications, providing both convenience and actionable insights into consumption patterns.

The study emphasizes the importance of threshold-based alerts and automated control for enhancing energy efficiency. The system is programmed to notify users when energy usage approaches predefined limits, helping them manage consumption proactively. Furthermore, the integration of a relay module allows the system to automatically disconnect the power supply when consumption exceeds maximum thresholds. This feature not only prevents energy wastage but also mitigates risks associated with electrical overloads or unsafe conditions, highlighting the system's potential for improving both efficiency and safety.

Pandey et al. also explore the design considerations for accurate and reliable energy measurement. The study discusses sensor calibration techniques, signal processing methods, and real-time data handling to ensure precise readings under varying load conditions. The ESP8266 microcontroller is chosen for its low power consumption, high processing capability, and built-in Wi-Fi connectivity, which simplifies integration with the IoT platform and ensures timely data transmission. The cloud platform supports data logging, visualization, and analytics, enabling users to track historical consumption trends, identify peak usage periods, and make informed decisions to reduce energy costs.

Additionally, the study investigates the scalability and adaptability of the proposed system. The authors highlight that the framework can be extended to multiple households or small buildings, supporting multiple nodes that communicate with a central server. This scalability ensures that the system is suitable for both individual homes and community-level energy management applications. The modular design allows easy integration of additional features such as power factor monitoring, renewable energy source integration, or predictive analytics for energy optimization.

Performance evaluation in the study demonstrates significant improvements over traditional metering systems. The proposed IoT-based smart energy meter provides accurate real-time measurements, timely alerts, and automated power control, resulting in measurable reductions in energy wastage and improved electrical safety. The research underscores the practical applicability of IoT-enabled energy monitoring systems in promoting efficient energy management and cost savings, while also providing a foundation for future development of smart grids and intelligent energy management systems.

Overall, Pandey et al. (2025) provide a detailed and practical implementation of a smart metering system, showcasing how IoT integration, sensor technology, and cloud-based monitoring can transform energy management in households and small buildings. Their work highlights the potential of combining real-time measurement, automated alerts, and remote monitoring to enhance energy efficiency, safety, and user convenience, forming a valuable reference for the development of modern smart energy solutions.

Smart Energy Metering for Economic and Environmental Efficiency -2022 Ahmad, F[1], Khan, S[2], Ali, R.[3]. [1,2,3] Department of Electrical Engineering, University of Lahore, Pakistan.

In their study, Ahmad et al. (2022) explore the design, implementation, and practical benefits of IoT-enabled smart energy meters with a focus on economic and environmental efficiency. The research addresses a critical limitation in traditional energy metering systems, which provide only manual readings and lack real-time monitoring capabilities. Such conventional systems often lead to inefficient energy usage, increased electricity costs, and a higher environmental footprint due to unnecessary energy wastage. Recognizing these challenges, the authors propose a comprehensive smart energy metering framework that leverages IoT technology, microcontrollers, and cloud-based monitoring to achieve real-time energy management.

The proposed system integrates current and voltage sensors with a microcontroller to accurately measure electrical parameters, including instantaneous current, voltage, power consumption, and cumulative energy usage. The microcontroller processes this data in real-time, calculating energy consumption metrics and transmitting them to a cloud-based IoT platform for visualization, analysis, and long-term storage. This cloud connectivity enables users to access consumption data remotely through web or mobile applications, allowing for continuous monitoring and better awareness of energy usage patterns.

A key feature of Ahmad et al.'s system is the implementation of automated alerts and power control mechanisms. The smart energy meter continuously monitors energy consumption against predefined thresholds and sends notifications to the user when consumption exceeds safe or cost-effective limits. In addition, the system integrates a relay module that automatically disconnects the power supply in cases of overconsumption or unsafe electrical loads. This feature not only helps reduce electricity wastage but also enhances safety by mitigating risks associated with electrical overloads or potential equipment damage.

The study emphasizes the economic benefits of deploying such a system. By providing real-time feedback and alert mechanisms, users can proactively manage energy consumption, leading to a reduction in electricity bills. Furthermore, the system's cloud-based platform allows users to analyze historical consumption trends, identify peak usage periods, and optimize energy usage schedules. This data-driven approach enables households and small businesses to make informed decisions regarding appliance usage, load management, and overall energy efficiency.

From an environmental perspective, the reduction in unnecessary energy consumption contributes directly to lower carbon emissions and a smaller ecological footprint. Ahmad et al. highlight that widespread adoption of IoT-enabled smart energy meters can significantly improve energy sustainability, particularly in residential areas and small-scale commercial setups. By integrating real-time monitoring with automated control, the system ensures that energy is consumed efficiently and only when needed, aligning with global goals of sustainable energy management.

The authors also address technical considerations, such as sensor calibration, microcontroller selection, and data communication reliability. The choice of a low-power microcontroller ensures continuous operation without excessive energy draw, while the cloud platform provides reliable storage, visualization, and remote accessibility. The modular design of the system allows scalability, making it possible to deploy multiple meters across households or buildings, all connected to a central monitoring server for community-level energy management.

Performance evaluation in the study demonstrates that the system accurately measures energy consumption, reliably triggers alerts, and effectively disconnects loads in response to threshold violations. Users reported enhanced awareness of energy usage, cost savings, and improved safety, confirming the practical utility of the approach. The research underscores the importance of combining IoT technology, sensor networks, and cloud computing to modernize energy management systems while achieving economic and environmental benefits.

Overall, Ahmad et al. (2022) provide a detailed and practical implementation of a smart energy metering system, highlighting its potential to improve economic efficiency, reduce energy wastage, enhance safety, and promote sustainable energy usage. The study contributes significantly to the field of smart energy management and provides a strong reference for future research and development of IoT-based energy monitoring solutions.

Sustainable Smart Meter Solutions for Residential Grids-2024 Altınok, A[1], Yilmaz, T[2], Demir, S[3], Kaya, M[4] Department of Electrical Engineering, Istanbul Technical University, Turkey

In their study, Altınok et al. (2024) explore the design and implementation of IoT-enabled smart energy meters for residential grids, with an emphasis on sustainability, energy efficiency, and user convenience. The research addresses a critical limitation of traditional energy meters, which are primarily designed for manual readings and billing purposes, and fail to provide real-time monitoring, automated alerts, or adaptive load control. Conventional meters often result in inefficient energy consumption, higher electricity bills, and increased environmental impact. Recognizing these challenges, the authors propose a comprehensive framework that combines sensor networks, microcontroller-based processing, and cloud-based monitoring to deliver sustainable energy management solutions for modern households.

The proposed system integrates current and voltage sensors with a microcontroller (ESP32/ESP8266) to measure key electrical parameters, including real-time voltage, current, active power, and cumulative energy consumption. These measurements are processed locally by the microcontroller and transmitted to a cloud-based platform, enabling users to monitor consumption remotely via web and mobile applications. The system provides a visual dashboard displaying energy usage patterns, historical data analysis, and predictive insights to inform household energy management decisions.

A major contribution of the study is the incorporation of threshold-based alerts and automated load control. The smart meter continuously compares energy consumption against pre-defined thresholds and sends notifications to users when consumption approaches or exceeds safe or cost-effective limits. Additionally, the system includes a relay module that can automatically disconnect power to appliances or circuits exceeding the set thresholds. This feature not only helps reduce energy wastage but also improves electrical safety by preventing overloads and potential hazards.

Altınok et al. emphasize the economic and environmental benefits of such IoT-based solutions. By providing real-time monitoring and proactive control, households can significantly reduce unnecessary energy usage, resulting in lower electricity bills and cost savings over time. From an environmental perspective, the reduction in energy wastage contributes to a smaller carbon footprint, aligning with sustainable energy management goals and national energy-saving initiatives. The study highlights that widespread adoption of smart meters can play a significant role in promoting energy sustainability at both individual and community levels.

The authors also discuss the technical design considerations of the system, including sensor calibration, microcontroller selection, and reliable wireless data communication. The microcontroller is optimized for low-power operation and high processing capability, ensuring continuous real-time monitoring without significant energy overhead. The modular design allows scalability, enabling the system to be deployed across multiple households or residential complexes with centralized monitoring. Furthermore, the cloud platform supports data logging, trend analysis, and remote accessibility, allowing users to analyze historical consumption patterns and implement strategies for load management and energy optimization.

Performance evaluation in the study demonstrates that the system provides accurate measurements, timely alerts, and reliable automated load disconnection. Users reported enhanced awareness of energy usage, improved control over appliance consumption, and increased safety. The authors also highlight the potential for integrating renewable energy sources and predictive analytics, further improving the efficiency and sustainability of residential energy management systems.

Overall, Altınok et al. (2024) provide a detailed and practical implementation of sustainable smart energy meters, combining IoT technology, sensor networks, microcontroller processing, and cloud-based monitoring. The study demonstrates how real-time monitoring, automated alerts, and load management can improve energy efficiency, reduce electricity costs, and enhance safety in modern households. This research contributes significantly to the field of smart energy solutions and provides a strong foundation for future development of scalable, sustainable, and intelligent energy management systems.

Cloud-Integrated Smart Energy Meter for Efficient Household Energy Management-2023 ***Authors:*** **Ramesh, T[1], Prasad, V[2], Sharma, P[3], Gupta, R[4], Department of Electrical Engineering, VIT University, India**

In their study, Ramesh et al. (2023) present the design and implementation of a cloud-integrated smart energy meter for household energy management, with the goal of improving energy efficiency, reducing wastage, and enabling real-time remote monitoring. The research addresses limitations of conventional energy meters, which rely on manual readings and lack real-time monitoring capabilities. Traditional systems often fail to provide actionable feedback to users, resulting in inefficient energy use, higher electricity bills, and potential safety hazards. Recognizing these challenges, the authors propose a comprehensive IoT-enabled framework that combines advanced sensing technology, microcontroller-based processing, and cloud computing to deliver practical, scalable, and efficient energy management solutions.

The system integrates high-precision voltage and current sensors with a microcontroller (ESP32/ESP8266) to continuously capture real-time electrical parameters, including instantaneous voltage, current, active and reactive power, and cumulative energy consumption. The microcontroller processes this data in real-time and sends it securely to a cloud-based IoT platform, where the information is visualized through interactive dashboards. This platform allows users to access historical and live consumption data remotely via web and mobile applications. Users can track daily, weekly, and monthly energy usage trends, identify peak consumption periods, and make informed decisions to reduce wastage and lower electricity costs.

A key feature of the proposed system is its threshold-based alert mechanism and automated load management. The smart energy meter continuously monitors energy consumption against predefined limits. When consumption approaches or exceeds these thresholds, notifications are sent to the user via the cloud platform, providing timely feedback for corrective action. In addition, the system incorporates a relay module that can automatically disconnect specific loads or appliances exceeding safe consumption levels. This functionality not only reduces energy wastage but also enhances household safety by preventing overloads, potential electrical fires, and damage to appliances.

Ramesh et al. also emphasize the economic and environmental benefits of the proposed system. By allowing households to monitor energy usage in real-time and respond proactively to high consumption, the system helps reduce electricity bills while minimizing unnecessary energy wastage. The reduction in energy consumption contributes to lower carbon emissions, supporting sustainable energy practices and aligning with global energy efficiency goals. The study demonstrates that wide-scale implementation of cloud-integrated smart energy meters can have a significant impact on energy conservation at both household and community levels.

The authors provide a detailed discussion of the technical design and system architecture. Sensor calibration, microcontroller programming, and reliable wireless data communication are addressed to ensure accurate measurement and efficient data transfer. The ESP32/ESP8266 microcontroller is selected for its high processing power, low energy consumption, and built-in Wi-Fi capabilities, allowing seamless integration with cloud services. The modular design of the system ensures scalability, enabling deployment across multiple households or residential complexes. Each smart meter operates independently while transmitting data to a centralized cloud server, facilitating community-level monitoring and analytics.

Performance evaluation in the study highlights the system's accuracy, reliability, and practical usability. Real-time measurements of voltage, current, and power were found to be precise, with minimal error, and the alert and automated disconnection features functioned reliably under various load conditions. Users reported improved awareness of energy consumption, better control over appliances, and enhanced safety, confirming the effectiveness of the proposed approach. Additionally, the cloud platform provides historical consumption analysis, predictive insights, and energy optimization suggestions, further supporting cost-effective and sustainable energy management.

The study also explores future extensions, including integration with renewable energy sources, smart home automation systems, and predictive algorithms for energy optimization. By combining IoT-enabled sensing, cloud analytics, and automated control, the system demonstrates a comprehensive approach to modern household energy management.

Overall, Ramesh et al. (2023) provide a thorough and practical implementation of a cloud-integrated smart energy meter, emphasizing real-time monitoring, automated alerts, load management, and remote accessibility. The research highlights significant economic and environmental benefits while providing a scalable and user-friendly solution for modern households. This study contributes substantially to the development of IoT-based smart energy management systems and serves as a benchmark for future research in intelligent energy monitoring, sustainable consumption, and smart grid applications.

CHAPTER 2

SYSTEM DESIGN

- The system captures brain signals using EEG sensors and converts them into text and speech.

- The system measures real-time voltage and current using sensors.

- Current and voltage sensors transmit data to the ESP8266 microcontroller.

- The microcontroller calculates power consumption and energy usage.

- Processed data is sent to an IoT platform (ThingSpeak) for remote monitoring.

- An LCD display shows real-time energy consumption and usage alerts.

- The system triggers notifications when energy exceeds predefined thresholds.

- A relay module automatically disconnects power in overconsumption conditions.

- Data logging and visualization allow analysis of historical energy patterns.

- Safeguards handle sensor errors and ensure reliable operation.

- Designed to improve energy efficiency, safety, and household monitoring.

EXISTING SYSTEM

- Traditional energy meters only record the total electricity consumption over a billing period and do not provide real-time monitoring.

- Users have no visibility into the energy usage of individual appliances, making it difficult to identify high-consumption devices.

- There are no alerts or notifications when electricity consumption approaches predefined limits, leaving users unaware of potential overuse.

- Automatic control or disconnection of power in cases of overconsumption is not available, which can result in energy wastage.

- Historical data analysis and consumption trends are not supported, preventing users from making informed decisions for efficient energy use.

- Energy management depends entirely on manual monitoring and user awareness, which can be inconsistent and prone to errors.

- Lack of real-time feedback and alerts can lead to higher electricity bills and increased safety risks, such as appliance overloads traditional meters do not support integration with smart devices or IoT platforms, limiting modern energy management capabilities

PROBLEM STATEMENT

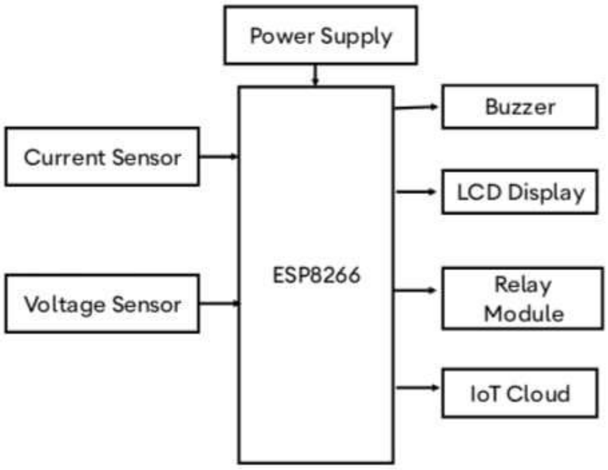
Households and small buildings currently lack intelligent systems capable of **real-time electricity monitoring**. Users are often unaware of **high energy consumption** by individual appliances or overall usage trends. Traditional energy meters only record total consumption and do not provide **instant feedback**, making it difficult to identify energy wastage or inefficient usage patterns. This lack of **visibility and control** leads to higher electricity bills, increased energy wastage, and limits the ability of users to implement **energy-saving strategies** effectively.

Existing meters also do not include **alerts or automatic control mechanisms** to disconnect power when usage approaches maximum thresholds. The absence of these safety and efficiency features can result in **unnecessary electricity consumption**, operational inefficiency, and potential **electrical hazards**, such as overloads or overheating, reducing overall **energy management effectiveness** in residential and small commercial buildings.

PROPOSED SYSTEM

The proposed system integrates **IoT technology** and embedded systems to monitor electricity consumption in real time **Current and voltage sensors** continuously measure electrical parameters, which are processed by a **microcontroller** (ESP8266) to calculate power usage the processed data is transmitted to an **IoT platform** via Wi-Fi, enabling **remote monitoring** users can access real-time electricity consumption through **mobile or web applications** the system sends **alerts** when usage approaches predefined limits **relay module** automatically disconnects power if the limit is exceeded, preventing energy wastage.

BLOCK DIAGRAM



HARDWARE REQUIREMENTS

- ESP8266 Microcontroller
- Current Sensor Module
- Voltage Sensor Module
- LCD Display
- Relay Module
- Buzzer
- Power Source

SOFTWARE REQUIREMENTS

- Arduino IDE
- Embedded C programming

CHAPTER 3

HARDWARE DESCRIPTION

NODE MCU

NodeMCU is an open-source Lua based firmware and **development board** specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module

Programming NodeMCU ESP8266 with Arduino IDE

The NodeMCU Development Board can be easily programmed with Arduino IDE since it is easy to use.

Programming NodeMCU with the Arduino IDE will hardly take 5-10 minutes. All you need is the Arduino IDE, a USB cable and the NodeMCU board itself. You can check this Getting Started Tutorial for NodeMCU to prepare your Arduino IDE for NodeMCU.

NodeMCU is an open source firmware for which open source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit).[8] The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits.

Both the firmware and prototyping board designs are open source.

The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.

The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications (see related projects).

HISTORY

NodeMCU was created shortly after the ESP8266 came out. On December 30, 2013, Espressif Systems began production of the ESP8266. NodeMCU started on 13 Oct 2015, when Hong committed the first file of nodemcu-firmware to GitHub. Two months later, the project expanded to include an open-hardware platform when developer Huang R committed the gerber file of an ESP8266 board, named devkit v0.9 Later that month, Tuan PM ported MQTT client library from Contiki to the ESP8266 SoC platform, and committed to NodeMCU project, then NodeMCU was able to support the MQTT IoT protocol, using Lua to access the MQTT broker. Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glib to the NodeMCU project, enabling NodeMCU to easily drive LCD, Screen, OLED, even VGA displays.

In the summer of 2015 the original creators abandoned the firmware project and a group of independent contributors took over. By the summer of 2016 the NodeMCU included more than 40 different modules.

ESP8266 ARDUINO CORE

As Arduino.cc began developing new MCU boards based on non-AVR processors like the ARM/SAM MCU and used in the Arduino Due, they needed to modify the Arduino IDE so that it would be relatively easy to change the IDE to support alternate toolchains to allow Arduino C/C++ to be compiled for these new processors. They did this with the introduction of the Board Manager and the SAM Core. A "core" is the collection of software components required by the Board Manager and the Arduino IDE to compile an Arduino C/C++ source file for the target MCU's machine language. Some ESP8266 enthusiasts developed an Arduino core for the ESP8266 WiFi SoC, popularly called the "ESP8266 Core for the Arduino IDE".[17] This has become a leading software development platform for the various ESP8266-based modules and development boards, including NodeMCUs. NodeMCU is an open-source LUA based firmware developed for the ESP8266 wifi chip. By exploring functionality with the ESP8266 chip, NodeMCU firmware comes with the ESP8266 Development board/kit i.e. NodeMCU Development board.

NodeMCU Development Board/kit v0.9 (Version1)

Since NodeMCU is an open-source platform, its hardware design is open for edit/modify/build.

NodeMCU Dev Kit/board consist of ESP8266 wifi enabled chip. The **ESP8266** is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer to the ESP8266 WiFi Module.

There is Version2 (V2) available for NodeMCU Dev Kit i.e. **NodeMCU Development Board v1.0 (Version2)**, which usually comes in black colored PCB.

NodeMCU Development Board/kit v1.0 (Version2)

For more information about NodeMCU Boards available in the market refer to NodeMCU Development Boards

NodeMCU Dev Kit has **Arduino like** Analog (i.e. A0) and Digital (D0-D8) pins on its board.

It supports serial communication protocols i.e. UART, SPI, I2C, etc.

Using such serial protocols we can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards, etc.

How to start with NodeMCU?

NodeMCU Development board is featured with wifi capability, analog pin, digital pins, and serial communication protocols.

To get started with using NodeMCU for IoT applications first we need to know about how to write/download NodeMCU firmware in NodeMCU Development Boards. And before that where this NodeMCU firmware will get as per our requirement.

There are online NodeMCU custom builds available using which we can easily get our custom NodeMCU firmware as per our requirement.

To know more about how to build custom NodeMCU firmware online and download it refer to Getting started with NodeMCU

How to write codes for NodeMCU?

After setting up ESP8266 with Node-MCU firmware, let's see the IDE (Integrated Development Environment) required for the development of NodeMCU.

NodeMCU with ESPlorer IDE

Lua scripts are generally used to code the NodeMCU. Lua is an open-source, lightweight, embeddable scripting language built on top of C programming language.

For more information about how to write Lua script for NodeMCU refer to Getting started with NodeMCU using ESPlorerIDE

NodeMCU with Arduino IDE

Here is another way of developing NodeMCU with a well-known IDE i.e. Arduino IDE. We can also develop applications on NodeMCU using the Arduino development environment. This makes it easy for Arduino developers than learning a new language and IDE for NodeMCU.

For more information about how to write Arduino sketch for NodeMCU refer to Getting started with NodeMCU using Arduino IDE

The difference in using ESPlorer and Arduino IDE

Well, there is a programming language difference we can say while developing an application for NodeMCU using ESPlorer IDE and Arduino IDE.

We need to code in C\C++ programming language if we are using Arduino IDE for developing NodeMCU applications and Lua language if we are using ESPlorer IDE.

Basically, NodeMCU is Lua Interpreter, so it can understand Lua script easily. When we write Lua scripts for NodeMCU and send/upload it to NodeMCU, then they will get executes sequentially. It will not build a binary firmware file of code for NodeMCU to write. It will send Lua script as it is to NodeMCU to get executed.

In Arduino IDE when we write and compile code, the ESP8266 tool chain in the background creates a binary firmware file of code we wrote. And when we upload it to NodeMCU then it will flash all NodeMCU firmware with newly generated binary firmware code. In fact, it writes the complete firmware.

That's the reason why NodeMCU not accept further Lua scripts/code after it is getting flashed by Arduino IDE. After getting flashed by Arduino sketch/code it will be no more Lua interpreter and we got an error if we try to upload Lua scripts. To again start with Lua script, we need to flash it with NodeMCU firmware.

Since Arduino IDE compiles and upload/writes complete firmware, it takes more time than ESPlorer IDE.

NodeMCU ESP8266 Specifications & Features

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included on board, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

2.3 PIN CONFIGURATION

The ESP8266 microcontroller, such as the **NodeMCU ESP-12E module**, has several pins that allow connection to sensors, actuators, and communication devices. Below is a summary of the main pin functions:

- **3V3 (3.3V)** – Power supply pin for the module. Provides 3.3V regulated output.

- **GND** – Ground pin, used to complete the circuit.

- **Vin** – Input voltage pin (can accept 5V from external power source).

- **GPIO0 – GPIO16** – General-purpose input/output pins used for digital signals, PWM, I2C, SPI, or interrupts.
- **GPIO0, GPIO2, GPIO15** – Used for boot mode selection at start up; must be correctly configured for normal operation.
- **ADC (A0)** – Analog-to-digital converter input; measures analog voltages from 0 to 1V (requires voltage divider for higher voltages).
- **TX / RX (UART0)** – Serial communication pins for

programming, debugging, and data transmission.

- **RST** – Reset pin; connecting to LOW resets the module.
- **EN / CH_PD** – Enable pin; must be HIGH to enable the chip.
- **SCL / SDA** – I2C clock and data pins, used for interfacing I2C sensors.
- **SPI Pins (SD3, SD2, CMD, CLK, SD0)** – Used for SPI communication with external devices.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM.

Memory

The **ESP8266** has a **32-bit Tensilica LX106 microcontroller** with **160 KB of SRAM** for program execution and up to **4 MB of flash memory** (varies by module) for storing programs and data. Unlike the Arduino Uno, the ESP8266 does not have a dedicated EEPROM, but emulated EEPROM libraries allow data storage in flash memory.

Input and Output

The ESP8266 has **11 digital GPIO pins** (GPIO0–GPIO16, some reserved for boot), which can be configured as input or output using pinMode(), digitalWrite(), and digitalRead() functions. They operate at **3.3 volts**. Each pin can provide or receive a maximum of **12 mA**.

Specialized functions of the pins include:

PWM – Most GPIO pins support 10-bit PWM output using the analogWrite () function, allowing dimming of LEDs or motor control.

- **UART / Serial** – TX and RX pins (typically GPIO1 = TX, GPIO3 = RX) are used for serial communication with computers or other devices.

- **ADC** – A0 pin provides a 10-bit analog-to-digital conversion input for sensors (0–1 V range; higher voltages require a voltage divider).

- **I2C (TWI)** – GPIO4 (SDA) and GPIO5 (SCL) pins support I2C communication using the Wire library.

- **SPI** – GPIO12 (MISO), GPIO13 (MOSI), GPIO14 (SCK), and GPIO15 (CS/SS) support SPI communication.

- **External Interrupts** – Most GPIO pins can be configured to trigger interrupts on rising, falling, or change events using the attach Interrupt () function.

- **Built-in LED** – GPIO2 is connected to the on-board LED on some NodeMCU boards; setting HIGH turns it off and LOW turns it on (active-low).

2.4 COMMUNICATION

The **ESP8266 microcontroller** depends on a host computer for developing, compiling, and uploading programs. The software used on the host computer is called an **Integrated Development Environment (IDE)**. In this project, the **Arduino IDE** is used to write and upload the program to the ESP8266 module. The Arduino IDE is an open-source platform that supports programming in **C/C++ based language**, making it easy for beginners as well as advanced users to develop embedded system applications.

The ESP8266 provides multiple communication interfaces that allow it to communicate with computers, sensors, displays, and other microcontrollers. One of the main advantages of the ESP8266 is its **built-in Wi-Fi capability**, which enables wireless communication and integration with **IoT platforms** for remote monitoring and control.

The ESP8266 supports **UART serial communication**, which is used for programming the module and for data exchange with a computer through the USB interface. The serial communication pins **TX (GPIO1)** and **RX (GPIO3)** are used to transmit and receive data. Through this interface, sensor readings and debugging information can be displayed on the **Serial Monitor** of the Arduino IDE.

In addition to UART communication, the ESP8266 also supports other communication protocols such as **I2C (Inter-Integrated Circuit)** and **SPI (Serial Peripheral Interface)**. The I2C protocol allows the ESP8266 to communicate with devices such as **LCD displays, sensors, and other peripheral modules** using only two communication lines: **SDA (data line)** and **SCL (clock line)**. The SPI protocol is used for high-speed communication with devices such as memory modules and other microcontrollers.

The Arduino IDE provides a large collection of **libraries** that simplify the process of programming hardware components. Libraries allow developers to reuse existing code and integrate different hardware modules easily. For example, libraries are available for **LCD displays, sensors, Wi-Fi communication, and relay control**, which are used in this project to

simplify programming and improve system functionality.
The structure Arduino program consists mainly of two functions: **setup()** and **loop()**. The **setup() function** runs once when the ESP8266 starts and used to initialize pins, sensors, communication protocols, and other hardware components. The **loop() function** runs continuously, allowing microcontroller to read sensor data, process information, and update the display or send data to IoT platform thus, the communication features of the ESP8266 enable efficient interaction between the **current sensor, voltage sensor, LCD display, relay module, and IoT platform**, allowing real-time monitoring and control of electricity consumption in the smart energy meter system.

VOLTAGE SENSOR

A voltage sensor is a device that measures voltage. Voltage sensors can measure the voltage in various ways, from measuring high voltages to detecting low current levels. These devices are essential for many applications, including industrial controls and power systems.

How does a voltage sensor work?

A voltage sensor is a device that measures the voltage of an electrical circuit. Voltage sensors are used in many applications, including monitoring and controlling equipment and machinery. Different types of voltage sensors work in various ways; here is an example:

- **Electromagnetic.** This type uses an electromagnetic field to detect changes in voltage. The sensor's exposure to an electric current generates a magnetic field. It induces currents in nearby conductors, such as wires or circuit boards, sensitive enough to detect these changes. This type of sensor is often used with microcontrollers since they can easily measure changes in electromagnetic fields around them with the help of built-in analog-to-digital converters (ADCs)

What are the different types of voltage sensors?
There are two types of voltage sensors: those that measure voltage and those that measure current (or amperage). The first type is a voltage-sensitive resistor (VSR), while the second is an operational amplifier (op-amp).

VSRs are used in electronic circuits where you need to sense small changes in voltage, such as sensors for measuring battery life or solar panel output. Op-amps can be used for tasks like driving motors or controlling LEDs, and they're also often found inside microcontrollers like Arduino boards.

What are the advantages of using a voltage sensor?
Voltage sensors are used to monitor and control many things. They can be used in industrial equipment and consumer products. Their advantages include:

- 24/7 monitoring of equipment
- Alerts if voltage data is critical (too high or too low)
- Their degree of accuracy for voltage monitoring is high
- Generally are eco-friendly
- Help you save on costs for electrical circuits and equipment since it monitors and prevents overheating or underpowering
- Help you prevent unexpected downtime on your equipment

Who uses voltage sensors?
Voltage sensors are used in various industries, including the automotive, manufacturing, maintenance, and medical fields.

In the maintenance industry, voltage sensors are used to monitor the voltage of assets and equipment. For example, if the sensor is wireless, it can be placed anywhere on an asset. The data can be relayed back to a CMMS (for example), where a maintenance manager can make adjustments based on their preventive maintenance plan. Below are some examples of voltage sensors in maintenance:

1. **Power failure detection:** the process of detecting a power failure so that the system can safely switch to an alternate power source.
2. **Load sensing:** a method of measuring the load on a motor and adjusting its speed accordingly.
3. **Safety switching:** refers to a device that shuts off power in case of an overload or fault condition to prevent equipment damage.
4. **Motor overload control:** a technique for preventing motor damage due to overloading by using thermal sensors, pressure sensors, current sensors, or other methods to detect the condition of the motor and avoid damage.
5. **Temperature control:** refers to controlling temperature by regulating airflow or adding insulation around machinery components.
6. **Fault detection:** refers to identifying faults within machinery components using sensors, alarms, or other devices to

perform maintenance before severe damage occurs.

VOLTAGE SENSOR:

Senses the AC mains voltage using a transformer or resistive voltage divider.

1. Outputs a scaled analog voltage proportional to input voltage.
2. Microcontroller reads the analog voltage and calculates **actual AC voltage** using a calibration factor.

POWER CALCULATIONS:

Real-time **power** can be calculated as:

P=V×IP = V \times IP=V×I

CURRENT SENSORS:

Current sensors are critical components in a wide range of electrical and electronic systems, ensuring that equipment and gadgets operate safely and efficiently. We will examine current sensors in this article, including their kinds, variables to consider when choosing a current sensor, performance comparison, and applications.

What is a Current Sensor?

A current sensor detects and measures the electric current passing through a conductor. It turns the current into a quantifiable output, such as a voltage, current, or digital signal, which may be utilised in a variety of applications for monitoring, control, or protection.

Types of Current Sensors and Their Advantages and Disadvantages

We may categorize current sensors based on the fundamental physical concept that underpins their construction. These current sensor concepts include Ohm's Law, Faraday's Law of Induction, Magnetic Field, and Optical Sensing. With this classification, we will now cover the many types of common current sensors.

Ohm's Law-based Current Sensing

Ohm's law states that voltage appears proportionate to the current supplied across a resistive element. Various sensors, including the frequently used shunt resistors and copper lines, have been devised using this fundamental approach.

SHUNT RESISTOR:

These are precision resistors with low resistance that are connected in series with the current carrying wire. The voltage drop across the shunt resistor is proportional to the current flowing through it, and this voltage may be measured to find out how much current is passing through it. Shunt resistors are capable of measuring both AC and DC currents. Because the shunt resistor is in the current conducting path, it can cause considerable power loss that grows with the square of the current. In high current applications, this power loss may limit the usage of shunt resistors.

Coaxial shunts have been widely used in a variety of applications requiring quick rise-time transient currents and large amplitudes. However, due to their tiny size and cheap cost, low-cost surface mounted devices (SMDs) are preferred by highly integrated electronic gadgets.

COPPER TRACE:

It is also feasible to use the intrinsic resistance of a conducting element in the circuit instead of a specific shunt resistor. However, because a copper line has a low resistance and hence a low voltage drop, a high gain amplifier is required to obtain a useful signal.

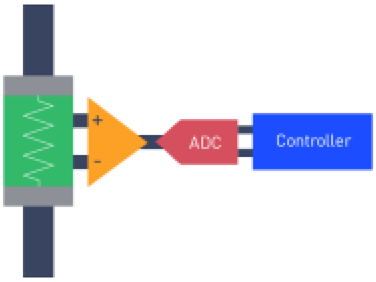


Figure 1: Use of Shunt Resistor For Current Sensing

Figure 1 depicts a typical application of a shunt resistor for current sensing. The produced voltage is then amplified to take use of the ADC's entire measuring range before being supplied into a digital controller for processing.

A significant drawback of this sort of current sensor is the intrinsic electrical connection that exists between the current to be measured and the sensing circuit. Electrical isolation can be obtained by utilizing an isolation amplifier, which raises the sensor's price. It can also reduce the bandwidth, accuracy, and thermal drift of the original current sensing technology. As a result, in situations where isolation is critical, current sensing devices based on physical principles with built-in electrical isolation are preferable.

Faraday's Law Of Induction-Based Current Sensing

Faraday's Law of Induction, which says that the total electromotive force (emf) generated in a closed circuit is proportionate to the time rate of change of the total magnetic flux linking the circuit, is widely used in current sensing devices. Two common Faraday's law-based sensing devices are current transformers (CTs) and Rogowski coils. When electrical isolation is required for safety reasons, these sensors automatically provide the necessary isolation between the current to be measured and the output signal. This makes present detecting equipment valuable.

CURRENT TRANSFORMERS(CTs):

The CT has a main winding (usually single-looped), a core, and a secondary winding. It is an effective high AC current measuring sensor. As a result, a large main current can be transformed into a smaller secondary current. This gadget doesn't require any additional driving circuitry because it is passive. Another important feature is that it can monitor very high current while using very little electricity. The ferrite material used in the core, however, can get saturated by a very high primary current or a substantial DC component in the current, causing the signal to be distorted. Another problem is that the core will have hysteresis once it has been magnetic, which will lower accuracy unless it is demagnetized once again. Furthermore, because the fundamental idea relies on the detection of a flux change, which is proportional to a current change, they cannot detect DC currents in standard form.

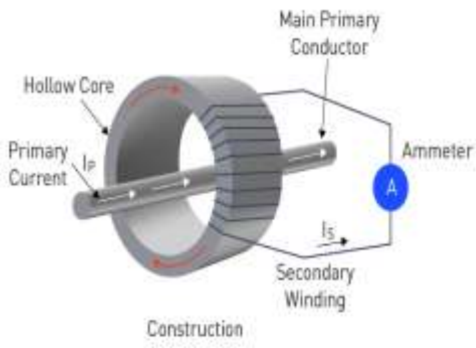


Figure 2: Basic Construction Of CT

Figure 2 illustrates how CT operates. Depending on the turns ratio, the change in the main current Ip is reflected on the secondary side as Is, which may be utilized for sensing. A shunt resistor that generates output voltage proportionate to the primary current can be used to monitor the output current. This provides isolation, minimal losses, a straightforward operating principle, and a voltage output that doesn't require extra amplification for current sensors. An analog-to-digital converter (ADC) may be able to sample the output voltage directly.

The primary current drop ratio is shown by the CT ratio. A current transformer's accuracy is gauged by its current transformer accuracy class, sometimes referred to as current transformer class or CT class. Based on their accuracy class, CTs are divided into two categories: metering accuracy CTs and relaying accuracy CTs. Metering Accuracy CTs are designed to be exceedingly accurate at all current ratings, even at very low current. They are evaluated for particular common loads. Due to its high level of precision, utility companies routinely utilize these CTs to assess usage for billing purposes. Relaying accuracy CTs are less accurate than metering accuracy CTs. They are designed to work with the minimal level of precision needed for equipment protection.

Due to their low cost and ability to create an output signal that is right away compatible with an analog-to-digital converter, current transformers are frequently utilized in power conversion applications. They also play a significant role in power distribution networks operating at 50/60 Hz line frequencies.

ROGOWSKI COILS:

These air-core coils are flexible and are wound around a conductor. A voltage in the conductor that is proportional to the rate of change of the current is induced by the changing magnetic field caused by the current in the conductor Ip. The main use of Rogowski coils is to measure AC currents, especially in high-frequency applications.

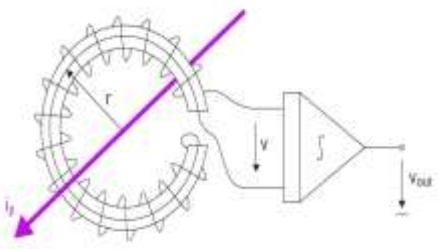


Figure 3: Schematics Of A Typical Rogowski Coil

Figure 3 shows the schematic of a typical Rogowski coil. The primary current's derivative determines how much voltage is generated. In order to get the requisite current sensing, an integrator at the output is required.

The Rogowski coil has a reduced sensitivity since the current transformer is unable to employ a magnetic core with high permeability. The Rogowski coil stands out since it is fundamentally linear and doesn't show saturation. Rogowski coils can be used to detect currents in power distribution systems, short-circuit testing systems, electromagnetic launchers, slip-ring induction motors, and lightning test facilities. Comparable to modern transformers in price.

Magnetic-Field Sensors for Current Sensing

It is difficult to detect currents that generate static magnetic fields using Faraday's law of induction. On the other hand, magnetic field sensors can identify both static and moving magnetic fields. They present themselves as a desirable alternate choice for current sensing.

HALL-EFFECT SENSORS

Figure 4: Hall Principle

These sensors are based on the Hall effect, which states that when a magnetic field is applied across the cross-section of a current carrying conductor, a voltage difference is formed across the conductor (see Figure 4). We can determine the direction of the electromotive force (emf) that is formed orthogonal to the current and magnetic field using the Fleming left-hand rule. The amplitude and relative angle of the current and magnetic field determine the resultant vector voltage, which is proportional to the Hall constant. The magnetic field is created by the current to be detected, which creates a magnetic field that provides voltage that may be analysed.

To make the output usable in most applications, signal conditioning is required. Signal conditioning electronics often require an amplifier stage as well as temperature correction. A differential amplifier with these properties may be simply combined with the Hall element using standard bipolar transistor technology. Temperature compensation is also simple to implement. Figure 5 shows a typical design of a Hall-effect current sensor.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

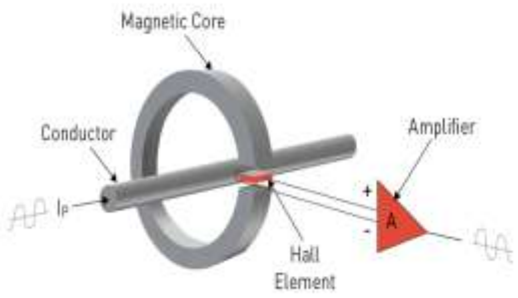


Figure 5: Use Of Hall Effect In Current Sensing

Hall-effect sensors are used in a variety of applications, including power conversion systems, welding equipment, motor drives, radar devices, and the electrowinning industry.

FLUXGATE SENSORS

The basic fluxgate sensors make use of the nonlinear connection between the magnetic field, H, and magnetic flux density, B, within a magnetic material, resulting in a change in the material's permeability.

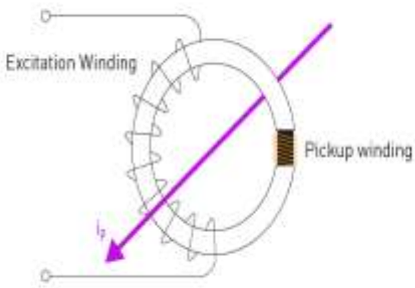


Figure 6: Basic Fluxgate Configuration

Figure 6 depicts a simple design of a fluxgate sensor for sensing current I_p. On the magnetic core, two windings are installed: the excitation winding and the pickup winding. The excitation winding is coupled to a sinusoidal current source, which creates a magnetic field of excitation. As a result, voltage is generated in the pickup winding, which may then be used for sensing. Given that the external field is minimal in comparison to the excitation field, the peak in output voltage is proportional to the external field and may be used to quantify it.

Due to their high cost and limited area, isolated fluxgate sensors have proved commercially useful primarily in high precision applications. Because of their high accuracy, fluxgates are employed in calibration systems, diagnosis systems, laboratory equipment, and medical systems.

MAGNETO–RESISTIVE(MR)SENSORS

Figure 7 shows a magneto-resistor which is a two-terminal device whose resistance varies in response to an applied magnetic field. The magnetic field's influence on material resistance is referred to as the MR effect. Using the MR effect, changes in the current to be detected cause variations in the magnetic field, which are reflected on the voltage that may then be analyzed.

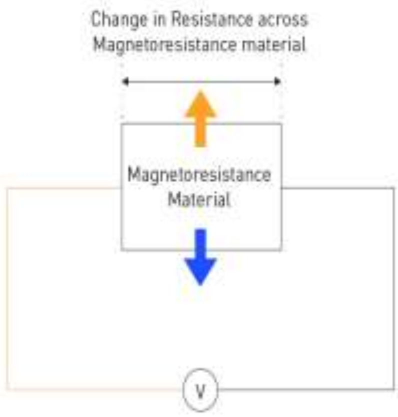


Figure 7: Principle Of Magneto-Resistive Current Sensing

These resistors are frequently used in a bridge design to compensate for thermal drift. Current sensors may be made using the MR effect. Anisotropic MagnetoResistance (AMR), Giant Magneto Resistance (GMR), Giant Magneto Impedance (GMI), and Tunnel MagnetoResistance (TMR) are examples of common forms.

Configurations of Magnetic Field-Based Current Sensors

These magnetic field-based current sensors are widely utilized in both closed-loop and open-loop applications.

The magnetic field sensor's output voltage is employed as an error signal in a closed-loop setup to push a current via a second transformer winding. This current generates a magnetic field, which compensates for the flux inside the magnetic core and forces it to zero. Under ideal conditions, the secondary current is proportional to the primary current ip. Figure 8 shows the closed loop sensor arrangement.

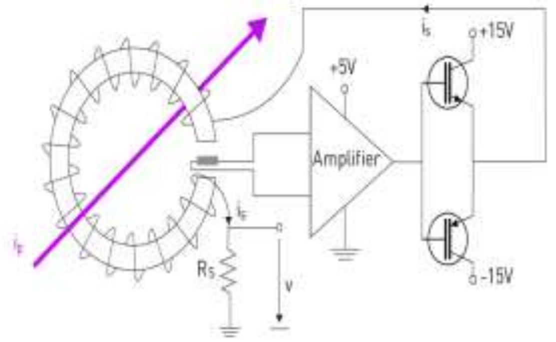


Figure 8: Closed-Loop Sensor Configuration

Figure 9 shows an open loop design with no feedback signal for adjustment.

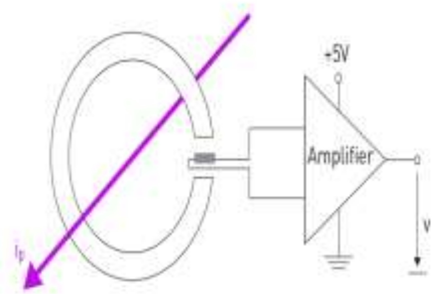


Figure 9: Open-Loop Sensor Configuration

With open-loop sensors, temperature drift or other non-linearities in the sensor will result in inaccuracies. On the contrary, closed-loop sensors enhance accuracy and saturation performance. The sensor configuration used is determined by the specific application and needs. Due to their cheap cost, compactness, and power needs, open-loop current sensors are quite widespread. Closed-loop current sensors are recommended in applications that require great accuracy and resistance to saturation, particularly in environments with broad temperature changes or electrical noise.

Fiber-Optic Current Sensors

To create current sensors, the Faraday magneto-optic effect can be used. The Faraday effect causes polarisation rotation that is proportionate to the magnetic field projection along the light transmission route. Most optically transparent dielectric materials exhibit this phenomenon when subjected to magnetic fields. Figure 10 depicts how a magnetic field, maybe created by a detected current, might cause polarization owing to Faraday's effect.

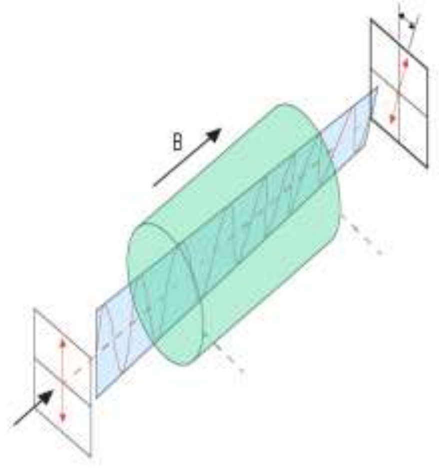


Figure 10: Faraday's Effect Resulting In Polarization of Light

Rotation can be detected by polarisers and analyzers at the input and output. By measuring the rotation of incident polarisation, the magnetic field and consequently current may be estimated.

Circuit Diagram of ESP8266 with Current & Voltage Sensors

The circuit diagram consists of connecting the **Current Sensor** and **Voltage Sensor** outputs to the **analog input pins of ESP8266** while providing proper power supply connections.

CONNECTIONS:

- **Current Sensor OUT → ESP8266 Analog Pin A0**
- **Voltage Sensor OUT → ESP8266 Analog Pin A1**
- **Current Sensor VCC → ESP8266 3.3V**
- **Current Sensor GND → ESP8266 GND**
- **Voltage Sensor VCC → ESP8266 3.3V**
- **Voltage Sensor GND → ESP8266 GND**
- **16×2 LCD Display** connected to ESP8266 digital pins (e.g., D1-D4) for real-time display of voltage, current, and power readings.

- **Relay Module** connected to a digital pin of ESP8266 for automatic load control when energy consumption exceeds thresholds.

COMPONENTS REQUIRED

ESP8266 Module (NodeMCU / Wemos D1 Mini)

- ACS712 Current Sensor Module
- ZMPT101B Voltage Sensor Module
- 16×2 LCD Display (I2C preferred for easy wiring)
- Relay Module
- Connecting Wires
- Mini Breadboard
- Resistors (as required for voltage divider)

CIRCUIT DESIGN

The circuit design is straightforward:

1. **Power Supply:**

o ESP8266 powered via USB or 3.3V regulated supply.

o Current & voltage sensors powered from 3.3V of ESP8266.

2. **Sensor Connections:**

o Connect the **OUT pin of ACS712** to **A0** of ESP8266.

o Connect the **OUT pin of ZMPT101B** to **A0** via voltage divider (if required for safe 0–1V input range).

3. **LCD Display:**

o Connect **SDA → D2** and **SCL → D1** (for I2C LCD) or use 4 digital pins for 4-bit parallel LCD.

o Displays **real-time current, voltage, and power** readings.

4. **Relay Module:**

o Connect **IN → digital pin D5** of ESP8266.

o Relay cuts off load automatically if energy exceeds predefined thresholds.

LCD DISPLAY

The LCD (Liquid Crystal Display) is an output device used to display text and numerical information. In this project, a 16×2 LCD display is used to show the converted text generated from EMG signals.

A 16×2 LCD means it has 16 characters per line and 2 lines. It is widely used in embedded systems because it is simple, low power consuming, and easy to interface with microcontrollers like Arduino.

The LCD helps the user to visually see the interpreted message generated from muscle activity signals.

Schematic of LCD Display

Internally, the 16×2 LCD consists of:

• Liquid crystal cells arranged in rows and columns

• HD44780 controller (built-in display controller)

• Character generator ROM (for ASCII characters)

• Control and data registers

The LCD operates in either 4-bit mode or 8-bit mode. In this project, 4-bit mode is used to reduce the number of Arduino pins required.

The display works by controlling liquid crystals that block or allow light to form characters on the screen.

Pins of the LCD Display

A standard 16×2 LCD has 16 pins:

VSS – Ground

VDD – +5V supply

V0 – Contrast adjustment (connected to potentiometer)

RS – Register Select

Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

RW – Read/Write

E – Enable

7–14. D0–D7 – Data pins

A – LED Backlight Positive

K – LED Backlight Ground

In 4-bit mode, only D4–D7 are used.

Important connections in this project:

• VSS → GND

• VDD → 5V

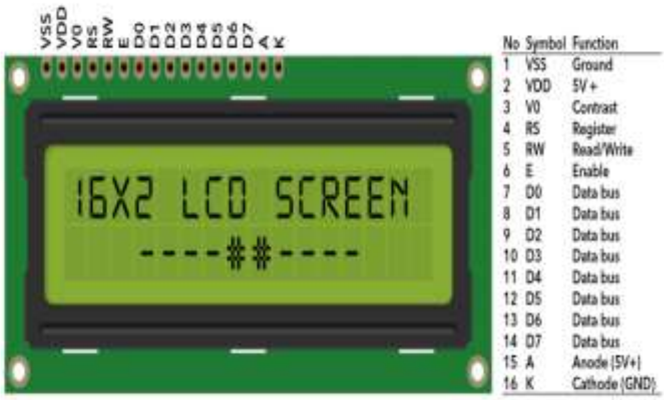
• V0 → Middle pin of 10K potentiometer

• RS → Digital Pin (e.g., 7)

• E → Digital Pin (e.g., 6)

• D4–D7 → Digital Pins (5–2)

RW→GND



CIRCUIT DESIGN

The LCD is connected to Arduino in 4-bit mode to save pins

• RS and E pins connected to digital output pins of Arduino

• Data pins D4–D7 connected to digital pins

• Contrast controlled using 10K potentiometer

• Backlight powered using 5V supply

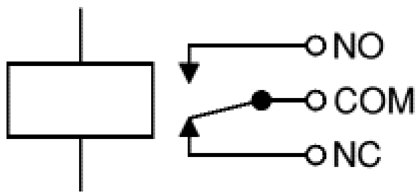
This configuration allows the Arduino to send character data and commands to the LCD.

RELAY:

A relay is an electrically operated switch. Current flowing through the coil of the relay creates a magnetic field which attracts a lever and changes the switch contacts. The coil current can be on or off so relays have two switch positions and they are double throw (changeover) switches. Relays allow one circuit to switch a second circuit which can be completely separate from the first. For example a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits; the link is magnetic and mechanical.

The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be as much as 100mA for relays designed to operate from lower voltages. Most ICs (chips) cannot provide this current and a transistor is usually used to amplify the small IC current to the larger value required for the relay coil. The maximum output current for the popular 555 timer IC is 200mA so these devices can supply relay coils directly without amplification.

Relays are usually SPDT or DPDT but they can have many more sets of switch contacts, for example relays with 4 sets of changeover contacts are readily available. Most relays are designed for PCB mounting but you can solder wires directly to the pins providing you take care to avoid melting the plastic case of the relay. The animated picture shows a working relay with its coil and switch contacts. You can see a lever on the left being attracted by magnetism when the coil is switched on. This lever moves the switch contacts. There is one set of contacts (SPDT) in the foreground and another behind them, making the relay DPDT.



The relay's switch connections are usually labeled COM, NC and NO:

- **COM** = Common, always connect to this, it is the moving part of the switch.
- **NC** = Normally Closed, COM is connected to this when the relay coil is **off**.
- **NO** = Normally Open, COM is connected to this when the relay coil is **on**.

SOFTWARE DESCRIPTION

ARDUINO SOFTWARE:

ARDUINO

Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.

Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can be communicate with software running on your computer (e.g. Flash, Processing, MaxMSP.) The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free.
The Arduino programming language is an implementation of Wiring, a similar physical computing platform, which is based on the Processing multimedia programming environment.

WHY ARDUINO ?

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

Inexpensive - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50

Cross-platform - The Arduino software runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.

Simple, clear programming environment - The Arduino programming environment is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with the look and feel of Arduino

Open source and extensible software- The Arduino software and is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based.

PROTEUS SOFTWARE:

Proteus 8 is a best simulation software for various designs with microcontroller. It is mainly popular because of availability of almost all microcontrollers in it. So it is a handy tool to test programs and embedded designs for electronics hobbyist. You can simulate your programming of microcontroller in Proteus 8 Simulation Software. After simulating your circuit in Proteus 8 Software you can directly make PCB design with it so it could be a all in one package for students and hobbyists. So I think now you have a little bit idea about what is proteus software.

Proteus is a Virtual System Modelling (VSM) that combines circuit simulation, animated components and microprocessor models to co-simulate the complete microcontroller based designs. This is the perfect tool for engineers to test their microcontroller designs before constructing a physical prototype in real time. This program allows users to interact with the design using on-screen indicators and LED and LCD displays and, if attached to the PC, switches and buttons.

Proteus comes with extensive debugging features, single stepping and variable display for a neat design prior to hardware prototyping. Proteus is the program to use when you want to simulate the interaction between software running on a microcontroller and any analog or digital electronic device connected to it.

Proteus was initially created as a multiplatform (DOS, Windows, Unix) system utility, to manipulate text and binary files and to create CGI scripts. The language was later focused on Windows, by adding hundreds of specialized functions for: network and serial communication, database interrogation, system service creation, console applications, keyboard emulation, ISAPI scripting (for IIS). Most of these additional functions are only available in the Windows flavour of the interpreter, even though a Linux version is still available.

Proteus was designed to be practical (easy to use, efficient, complete), readable and consistent.

Its strongest points are:

- powerful string manipulation;
- comprehensibility of Proteus scripts;
- Availability of advanced data structures: arrays, queues (single or double), stacks, bit maps, sets, AVL trees.
- The language can be extended by adding user functions written in Proteus or DLLs created in C/C++.

GET PROTEUS 8 SOFTWARE?

After getting introduction about what is proteus software if you wish to download and give a try to proteus 8 software then labcenter is giving latest version of proteus, proteus 8 demo version for free of cost. The latest Proteus Software version is Proteus 8. You can download it from here.

EXPLANATION:

If you have clear idea about what is proteus 8 simulation software then you can start proteus 8 simulations at this point of time. I am sure that you have your proteus 8. If you don't have then download from above link. To learn Proteus 8 You can start with a series of video tutorials or you can download a Proteus 8 Software Tutorial PDF. But main and most important thing with this software is hands on experiments. the more you work the better you become There is no shortcut.

After making your Schematic you have to do PCB designing so here is the video tutorial which describes Single layer auto routing.

At first sight, Proteus may appear similar to Basic because of its straight syntax, but similarities are limited to the surface:

LANGUAGE FEATURES:

Proteus has a fully functional, procedural approach; variables are untyped, do not need to be declared, can be local or public and can be passed by value or by reference; all the typical control structures are available (if-then-else; for-next; while-loop; repeat-until; switch-case);new functions can be defined and used as native functions.

Data types supported by Proteus are only three: integer numbers, floating point numbers and strings. Access to advanced data structures (files, arrays, queues, stacks, AVL trees, sets and so on) takes place by using handles, i.e. integer numbers returned by item creation functions.

Type declaration is unnecessary: variable type is determined by the function applied – Proteus converts on the fly every variable when needed and holds previous data renderings, to avoid performance degradation caused by repeated conversions.

There is no need to add parenthesis in expressions to determine the evaluation order, because the language is fully functional (there are no operators).

Proteus includes hundreds of functions for:

- accessing file system;
- sorting data;
- manipulating dates and strings;
- interacting with the user (console functions)
- Calculating logical and mathematical expressions.

Proteus supports associative arrays (called sets) and AVL trees, which are very useful and powerful to quickly sort and lookup values.

The functional approach and the extensive library of built-in functions allow to write very short but powerful scripts; to keep them comprehensible, medium-length keywords were adopted.The user, besides writing new high-level functions in Proteus, can add new functions in C/C++ by following the guidelines and using the templates available in the software development kit; the new functions can be invoked exactly the same way as the predefined ones, passing expressions by value or variables by reference. Proteus is an interpreted language: programs are loaded into memory, pre-compiled and run; since the number of built-in functions is large, execution speed is usually very good and often comparable to that of compiled programs. One of the most interesting features of Proteus is the possibility of running scripts as services or ISAPI scripts. Running a Proteus script as a service, started as soon as the operating system has finished loading, gives many advantages: no user needs to login to start the script; a service can be run with different privileges so that it cannot be stopped by a user. This is very useful to protect critical processes in industrial environments (data collection, device monitoring), or to avoid that the operator inadvertently closes a utility (keyboard emulation).The ISAPI version of Proteus can be used to create scripts run through Internet Information Services and is equipped with specific functions to cooperate with the web server.

For intellectual property protection Proteus provides:

digital signature of the scripts, by using the development key (which is unique);the option to enable or disable the execution of a script (or part of it) by using the key of the customer. Proteus is appreciated because it is relatively easy to write short, powerful and comprehensible scripts; the large number of built-in functions, together with the examples in the manual, keep low the learning curve.

The development environment includes a source code editor with syntax highlighting and a context-sensitive guide. Proteus does not need to be installed: the interpreter is a single executable (below 400 Kb) that does not require additional DLLs to be run on recent Windows systems.

HOW TO SETUP PROTEUS?

1. Start-up the Microsoft Windows.

2. Place the ECOM 4315 CD into CD ROM drive.

3. Double click on the software tools folder and then choose Proteus Simulator after

that double click Proteus 7.1 folder and run the setup.

4. Press next until you reach the window which ask for the key.

5. From browse for key; browse until you reach the same folder where the setup

exists, and then open the second folder and chose the MAXIM_LICENCE folder.

6. Then click on the top icon which is MAXIM and then press install.

7. Then choose yes and then close the window, after that browse and install the

program.

8. After that run the patch which exist in the same folder where the

MAXIM_LICENCE exists.

9. Then browse for c:\Program files\Labcenter electronics \Proteus.

10. Finally chose next, and then finish.

GET STARTED TO PROTEUS

1. From start menu chose the PROTEUS, and then chose ISIS(blue).

2. To get a part click on Devices (P).

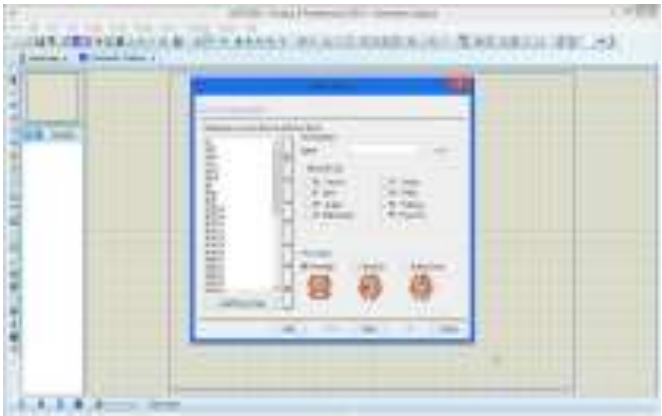
3. Write the name of the PIC16F84A.

4. To get LED write led then chose green led.

5. To get a resistor write 1k and chose the first part. Device Libraries

Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Proteus capture comes with pre-supplied libraries for TTL, CMOS, ECL, Microprocessor, Memory and Analogue ICs plus libraries with hundreds of named Bipolar, FET and Diode discrete semiconductors. Also included are manufacturer specific libraries from National Semiconductor, Philips, Motorols, Teccor, Texas Instruments, Dallas and Zetex amongst others - an expanding total, currently over 10,000 parts.

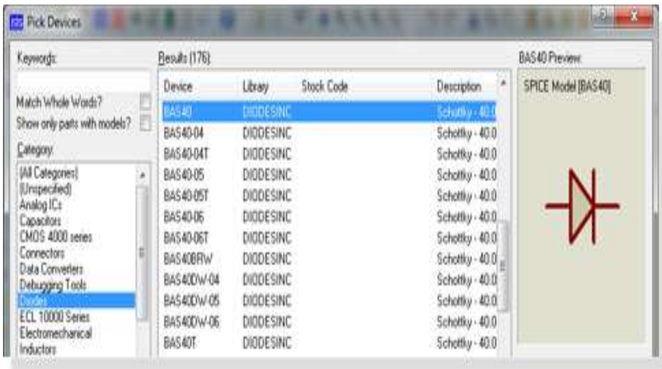


Importing Components from BSDL

Direct import of BSDL (Boundary Scan Description Library) files provides a semi-automatic way to quickly create and package larger components.

Libraries provided with Proteus also contain default properties for PCB packaging and/or simulator models. Both homogenous and heterogeneous multi-element devices are supported so you can place a relay's coil and contacts on separate parts of the drawing. Connector pins can also be distributed. Device editing is performed directly on the drawing using the general editing tools. Devices may be constructed from lines, boxes, circles, arcs, text and special pin objects. Non-electrical symbols may also be created for use in small mechanical or block diagrams.

A summary listing of included embedded peripheral libraries can be found here.



RESULT

The proposed system **"Smart Energy Meter "** was successfully implemented and tested. The **current sensor and voltage sensor** were able to measure electrical parameters accurately when properly connected to the circuit.

The captured **voltage and current signals** were processed by the **ESP8266 microcontroller**. Based on the measured electrical parameters, the system calculated **power consumption and energy usage** in real time. The measured values were displayed on the **16×2 LCD display**, allowing users to monitor electricity consumption continuously.

The system also enabled **IoT-based monitoring**, where the energy consumption data could be transmitted to an online platform through Wi-Fi. This allowed users to view electricity usage remotely using a **mobile or web application**. The system demonstrated real-time monitoring with minimal delay between data measurement and display.

Additionally, the **relay module** was integrated into the system to provide automatic power control. When the electricity consumption exceeded the predefined threshold limit, the relay automatically disconnected the power supply to prevent excessive energy usage and improve safety measure real-time **voltage and current values** accurately display electricity consumption information on the **LCD screen** provide **alerts when energy usage approached the set limit** automatically **disconnect the power supply using the relay module** when the threshold was exceeded operate continuously with **stable readings and reliable performance.**

The overall performance of the system was satisfactory. The project demonstrates that an **IoT-based smart energy meter can effectively monitor electricity consumption, improve energy management, and reduce unnecessary power wastage**, making it suitable for residential and small-scale energy monitoring applications.

REFERENCES

[1] A. Pandey, R. Sharma, and P. Gupta, "Smart Metering System for Real-Time Energy Monitoring," Department of Electrical Engineering, Indian Institute of Technology Delhi, India, 2025.

[2] S. Babu, R. Kumar, and P. Singh, "IoT-Based Smart Energy Meter for Automated Billing and Energy Monitoring," Department of Electrical and Electronics Engineering, Anna University, Chennai, India, 2022.

[3] F. Ahmad, S. Khan, and R. Ali, "Smart Energy Metering for Economic and Environmental Efficiency," Department of Electrical Engineering, University of Lahore, Pakistan, 2022.

[4] A. Altınok, T. Yilmaz, H. Demir, and M. Kaya, "Sustainable Smart Meter Solutions for Residential Grids," Department of Electrical Engineering, Istanbul Technical University, Turkey, 2024.

[5] T. Ramesh, V. Prasad, A. Kumar, and S. Rao, "Cloud-Integrated Smart Energy Meter for Efficient Household Energy Management," Department of Electrical Engineering, VIT University, India, 2023.

[6] V. C. Gungor and D. Sahin, "Smart Grid Technologies and Applications for Intelligent Energy Monitoring Systems," Department of Electrical Engineering, Koc University, Istanbul, Turkey, 2013.

[7] N. Kumar, R. Gupta, and P. Sharma, "Design and Implementation of IoT-Based Energy Meter for Real-Time Monitoring," Department of Electronics Engineering, Delhi Technological University, India, 2021.

[8] M. Hassan, M. Rahman, and S. Islam, "Wireless Smart Energy Meter Using ESP8266 for Remote Energy Monitoring," Department of Electrical Engineering, University of Dhaka, Bangladesh, 2020.

[9] V. Sharma, D. Patel, and S. Mehta, "IoT-Based Smart Power Monitoring and Control System," Department of Electrical Engineering, Gujarat Technological University, India, 2021.

[10] R. Karthik, S. Balaji, and P. Mohan, "Real-Time Energy Monitoring and Load Control Using Internet of Things Technology," Department of Electrical Engineering, SRM Institute .