Dual Voltage Guard System using Arduino Uno

Submitted in partial fulfilment of the requirements for the degree of

Bachelor of Technology

in

Electrical and Electronics Engineering

by

C M YAGNESH – 21BEE0260

SHAILESH RAJ – 21BEE0186

LOKESH YADAV – 21BEE0192

Under the guidance of

Dr. BALAJI S

School of Electrical Engineering

VIT, Vellore.

November, 2024

DECLARATION

We hereby declare that the thesis entitled **"Dual voltage guard system using Arduino Uno"** submitted by us, for the award of the degree of Bachelor of Technology in Electrical and Electronics Engineering to VIT University is a record of Bonafide work carried out by us under the supervision of **Dr. Balaji S**, Professor, School of Electrical Engineering, VIT University, Vellore. We further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Vellore

Date: 20-11-2024

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "**Dual voltage guard system using Arduino Uno**" submitted by **C M YAGNESH(21BEE0260), SHAILESH RAJ(21BEE0186), LOKESH YADAV(21BEE0192)**, School of Electrical Engineering, VIT, Vellore, for the award of the degree of Bachelor of Technology in Electrical and Electronics Engineering , is a record of Bonafide work carried out by him/her under my supervision during the period, 15-8-2024 to 20-11-2024,, as per the VIT code of academic and research ethics. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place: Vellore

Date:20/11/2024

Signature of the Guide

Dr. BALAJI S

Head of the department - EEE

Head of the Department
Department of Electrical Engineering
School of Electrical Engineering (SELECT)
Vellore Institute of Technology (VIT)
Deemed to be University under section 3 of the UGC Act, 1956
Vellore – 632 014, Tamil Nadu, India

Dean, SELECT

Dean
School of Electrical Engineering (SELECT)
Vellore Institute of Technology (VIT)
(Deemed to be University under section 3 of UGC Act, 1956)
Vellore – 632 014, Tamil Nadu, India

Acknowledgement

We are grateful to the Chancellor of VIT University, **Dr. G.Viswanathan**, the Vice Chancellor for motivating us to pursue the internship.

We express our sincere thanks to **Dr. Mathew Mithra Noel**, Dean, School of Electrical Engineering, VIT University for his kind words of support and encouragement. We would like to acknowledge the support rendered by our colleagues in several ways throughout our internship.

With immense pleasure and deep sense of gratitude, we wish to express my sincere thanks to our supervisor **Dr. BALAJI S**, Professor Grade 1, Department of Instrumentation, or Centre, School of Electrical Engineering, VIT University, without his motivation and continuous encouragement, this project work would not have been successfully completed.

We wish to extend our profound sense of gratitude to our parents for all the sacrifices they made during our research and also providing us with moral support and encouragement whenever required.

Place: Vellore

Date: 20/11/2024

C M YAGNESH

SHAILESH RAJ

LOKESH YADAV

Vision and Mission Institute

INSTITUTE:

Vision:

Transforming life through excellence in education and research.

Mission:

- World class Education: Excellence in education, grounded in ethics and critical thinking, for improvement of life.
- Cutting edge Research: An innovation ecosystem to extend knowledge and solve critical problems.
- Impactful People: Happy, accountable, caring, and effective workforce and students.
- Rewarding Co-creations: Active collaboration with national & international industries & universities for productivity and economic development.
- Service to Society: Service to the region and world through knowledge and compassion.

Department:

Vision:

To offer an education in electrical engineering that provides strong fundamental knowledge, skills for employability, cross-disciplinary research and creates leaders who provide technological solutions to societal and industry problems.

Mission:

M1 Provide personalized experiential learning in industry sponsored laboratories to prepare students in electrical engineering with strong critical thinking and employability skills.

M2 Foster design thinking, creativity and cross-disciplinary research with highly qualified faculty to create innovators and entrepreneurs in the broad area of electrical engineering.

M3 Collaborate with national and international partners to provide innovative solutions to societal and industry challenges.

Executive Summary

This report encapsulates the comprehensive learning and practical experience gained during the development of the **Dual Voltage Guard System using Arduino Uno**, undertaken as part of the final-year B.Tech curriculum in Electrical and Electronics Engineering. The project focused on creating a reliable and cost-effective system to protect electrical devices from the adverse effects of over-voltage and under-voltage conditions by leveraging **Arduino Uno** and **MATLAB Simulink**.

The project's core involved programming the **Arduino Uno** to monitor voltage levels in real time and automatically disconnect the load when unsafe conditions were detected. This required an in-depth understanding of Arduino's functionalities, particularly analog input reading, threshold-based decision-making, and load control. Additionally, **MATLAB Simulink** was utilized to simulate and validate the system's behaviour across various operating conditions, providing an accurate model to optimize performance before implementation.

This project offered hands-on exposure to key technical concepts, such as **embedded system design**, **real-time data processing**, and **voltage protection methodologies**. Beyond the technical scope, it fostered the development of essential engineering skills, including analytical problem-solving, effective use of simulation tools, and the integration of software with hardware for practical applications.

The experience has been instrumental in developing a strong foundation in embedded systems and their applications in electrical engineering. The project not only enhanced technical proficiency in **Arduino programming** and **Simulink modeling** but also deepened the understanding of protective systems for electrical networks. Overall, the **Dual Voltage Guard System using Arduino** serves as a testament to the practical application of academic knowledge, showcasing the ability to tackle real-world engineering challenges with innovation and efficiency.

TABLE OF CONTENTS

S.No	Contents	Page No.
	ABSTRACT	**1**
1.	**INTRODUCTION**	**2 - 3**
	1.1 Background	**2**
	1.2 Motivations	**2**
	1.3 Scope of the Project	**3**
2.	**PROJECT DESCRIPTION AND GOALS**	**4 - 17**
	2.1 Literature Review	**4**
	2.2 Research Gap	**14**
	2.3 Objectives	**15**
	2.4 Problem Statement	**17**
3.	**TECHNICAL SPECIFICATION**	**18 - 29**
	3.1 Requirements	**18**
	3.1.1 Functional Requirements	**18**
	3.1.2 Non-Functional Requirements	**19**
	3.2 Feasibility Study	**20**
	3.2.1 Technical Feasibility	**20**
	3.2.2 Economic Feasibility	**20**
	3.2.3 Social Feasibility	**21**
	3.3 System Specification	**22**
	3.3.1 Hardware Specification	**22**

	3.3.2 Software Specification	**28**
4.	**DESIGN APPROACH AND DETAILS**	**30 - 41**
	4.1 System Architecture	**30**
	4.2 Design	**32**
	4.2.1 Data Flow Diagram	**32**
	4.2.2 Use Case Diagram	**34**
	4.3 Circuit Diagram	**35**
	4.3.1 Hardware Circuit Diagram	**35**
	4.3.2 Software Circuit Diagram using MATLAB	**38**
5.	**CODES AND TESTING**	**42 - 52**
	5.1 Codes	**42**
	5.2 Testing	**49**
	5.2.1 Testing Procedure	**49**
	5.2.2 Test Cases	**50**
6.	**RESULT**	**53 - 55**
	6.1 Output	**53**
	6.2 Project Results	**54**
	6.3 Conclusion	**55**
7.	**REFERENCE**	**56 - 68**
	7.1 Literature Review Papers	**58**
	7.2 Other Online Material	**60**
	7.3 Books	**60**

LIST OF FIGURES

Figure number	Figure name	Figure page number
Fig. 3.1	Arduino Uno	23
Fig. 3.2	Voltage Sensor	23
Fig. 3.3	Current Sensor	24
Fig. 3.4	Relay Module	25
Fig. 3.5	LCD Display	25
Fig. 3.6	12C Module	26
Fig. 3.7	Bulb	27
Fig. 3.8	Jump Wires	27
Fig. 3.9	AC Supply	28
Fig. 4.1	System Architecture	30
Fig. 4.2	Data flow	32
Fig. 4.3	Use Case	34
Fig. 4.4	Circuit Diagram	35
Fig. 4.5	Simulation Circuit Diagram	38
Fig. 4.6	Circuit of Undervoltage Relay	39
Fig. 4.7	Graph of Undervoltage Relay	39
Fig. 4.8	Circuit of Overvoltage Relay	40
Fig. 4.9	Graph of Overvoltage Relay	40
Fig. 5.1	Normal operation (198.60V)	51
Fig. 5.2	Overvoltage condition (224.95V)	51
Fig. 5.3	Undervoltage condition (172.24V)	52
Fig. 5.4	Sensor Failure Condition	52

Fig. 6.1	System Trip Logic Graph	53
Fig. 6.2	Relay Status Over Time	54

LIST OF TERMS AND ABBRIVATIONS

1. **LCD** - Liquid Crystal Display
2. **AC** - Alternating Current
3. **DC** - Direct Current
4. **I2C** - Inter-Integrated Circuit
5. **ADC** - Analog-to-Digital Converter
6. **PCB** - Printed Circuit Board
7. **EEPROM** - Electrically Erasable Programmable Read-Only Memory
8. **MATLAB** - Matrix Laboratory
9. **Simulink** - Simulation and Link
10. **SDA** - Serial Data
11. **SCL** - Serial Clock
12. **RMS** - Root Mean Square
13. **EMI** - Electromagnetic Interference
14. **EMC** - Electromagnetic Compatibility
15. **OVR** - Overvoltage Ratio
16. **SCR** - Silicon Controlled Rectifier
17. **HVDC** - High Voltage Direct Current
18. **SDA** - Serial Data
19. **SCL** - Serial Clock
20. **ATS** - Automatic Transfer Switch
21. **DG** - Distributed Generation
22. **HVDC** - High Voltage Direct Current

23. **PWM** - Pulse Width Modulation
24. **UVR** - Undervoltage Relay
25. **OVR** - Overvoltage Relay
26. **AC** - Alternating Current
27. **SCR** - Silicon Controlled Rectifier
28. **MOV** - Metal Oxide Varistor
29. **NDZ** - Non-Detection Zone
30. **IEC** - International Electrotechnical Commission
31. **NEMA** - National Electrical Manufacturers Association
32. **LED** - Light Emitting Diode
33. **DSP** - Digital Signal Processor
34. **ADC** - Analog-to-Digital Converter
35. **mA** - Milliampere
36. **RCD** - Residual Current Device
37. **MCB** - Miniature Circuit Breaker

ABSTRACT

This project focuses on the development of a real-time voltage and current monitoring system aimed at enhancing safety, efficiency, and automation in power management. Utilizing an Arduino Uno microcontroller, voltage sensors, and current sensors, the system measures electrical parameters in real time and displays the data on an LCD screen integrated with an I2C module for improved user accessibility. The primary objective is to protect electrical systems and connected loads from damage caused by voltage fluctuations.

The system operates by continuously monitoring the voltage and current levels of the circuit. If the voltage deviates beyond a predefined safe range—either exceeding or falling below the threshold—the system automatically triggers a relay to disconnect the load, such as a light bulb, from the power supply. This automated response ensures that sensitive electrical components are safeguarded against potential hazards such as overvoltage, undervoltage, or overcurrent conditions.

In addition to its protective function, the system enhances user experience by providing real-time feedback through the LCD, allowing users to monitor the operational status of the system at a glance. The integration of an I2C module simplifies the wiring, reduces power consumption, and ensures compactness, making the design practical for various applications.

This project demonstrates the importance of incorporating automation into electrical systems to minimize human intervention, reduce energy wastage, and improve overall reliability. With its cost-effectiveness, scalability, and simplicity, the proposed system is ideal for residential, industrial, and educational purposes, promoting the adoption of smart energy management solutions.

INTRODUCTION

1.1 Background

Voltage fluctuations, such as over-voltage and under-voltage conditions, are common challenges in electrical systems, especially in regions with unstable power supply or outdated infrastructure. These fluctuations can lead to significant damage to sensitive electrical devices, reduced operational lifespan, and even complete equipment failure. Existing solutions, while effective, are often expensive, rigid in their application, or lack advanced features such as real-time monitoring and flexibility in operation.

To address these challenges, the **Dual Voltage Guard System** is proposed as a reliable and cost-effective solution. By leveraging the versatility of Arduino uno and the analytical capabilities of MATLAB Simulink, this project aims to design and implement a system that can continuously monitor voltage levels, detect unsafe conditions, and take corrective actions automatically. The integration of hardware and software ensures a compact, efficient, and user-friendly solution that caters to a wide range of applications, including residential, commercial, and industrial use.

1.2 Motivations

The motivation for this project stems from the increasing reliance on sensitive electrical and electronic devices in modern households and industries. Voltage fluctuations not only pose a risk to these devices but also contribute to higher maintenance costs and operational inefficiencies. A cost-effective and efficient solution is needed to ensure the safety and longevity of electrical systems.

Additionally, the project provides an opportunity to bridge theoretical knowledge with practical application. It allows the team to delve into core concepts such as embedded system design, real-time data processing, and hardware-software integration. The choice of Arduino uno as the microcontroller platform adds a layer of accessibility and flexibility, while MATLAB Simulink enables detailed simulations for performance validation. This combination ensures a well-rounded learning experience and the development of practical engineering skills.

1.3 Scope of the Project

The **Dual Voltage Guard System using Arduino uno** is designed to provide an efficient, reliable, and cost-effective solution for protecting electrical devices against voltage fluctuations. The system primarily focuses on monitoring voltage levels in real time, detecting unsafe conditions such as over-voltage and under-voltage, and automatically disconnecting the load to prevent damage to connected devices. By leveraging the capabilities of Arduino uno and MATLAB Simulink, the project aims to combine hardware functionality with advanced simulation and validation tools for optimal performance.

The scope of the project includes the development of a fully operational hardware system based on Arduino uno. This microcontroller platform is selected for its compact size, ease of use, and ability to handle analog and digital inputs efficiently. The Arduino will continuously monitor input voltage levels using its analog-to-digital conversion feature, compare the readings against predefined safe thresholds, and control a relay module to disconnect or reconnect the load as necessary. The hardware implementation ensures the system's functionality in practical applications, such as safeguarding household appliances, commercial equipment, and industrial machinery.

In addition to the hardware, the project incorporates the use of **MATLAB Simulink** to simulate and validate the system's behaviour. Simulink enables the modelling of various voltage fluctuation scenarios, providing insights into how the system responds under different conditions. This simulation-driven approach ensures that the system is robust and reliable before deployment. It also facilitates iterative testing and optimization, allowing for the identification and rectification of potential issues at an early stage.

The system is designed to be flexible and scalable, making it suitable for a wide range of applications. It can be easily reconfigured to accommodate different voltage ranges, making it adaptable for residential, commercial, and industrial environments. Furthermore, the system's modular design allows for future enhancements, such as integrating remote monitoring capabilities, IoT connectivity, or advanced data analytics. This scalability ensures the project's relevance in addressing evolving needs and challenges in electrical safety.

Overall, the scope of the project encompasses the design, development, and validation of a comprehensive voltage protection system that combines real-time monitoring, automated response mechanisms, and advanced simulation techniques. By addressing both practical and technical challenges, the project aims to deliver a reliable solution that safeguards electrical devices and enhances system efficiency.

PROJECT DESCRIPTION AND GOALS

2.1 Literature Review

Reference	Methodology	Key finding	Strength	Weakness
Overvoltage and Undervoltage Protection of Load using GSM modem SMS Alert [1]	- Monitors voltage using ATmega328 microcontroller with a tripping mechanism. - Activates when voltage exceeds thresholds. - Displays status on LCD, triggers a buzzer for faults, and sends SMS via GSM modem. - Demonstrated with a DC motor as the load.	- Effectively detects over-voltage and under-voltage conditions. - Sends timely SMS alerts for user intervention. - Protects devices via automated tripping. - Provides real-time visual and auditory notifications.	Automated protection with SMS alerts. - Simple, user-friendly interface with LCD and buzzer. - Cost-effective and scalable for diverse applications.	Depends on GSM network availability. - Limited compatibility with diverse load types. - Fixed voltage thresholds may not suit all devices.

Development of Overvoltage and Undervoltage Protection System for a Residential Building Interfaced with a GSM Notification System. [2]	- Designed a protection system for overvoltage and undervoltage using a PIC 16F876A microcontroller. - GSM module (SMS 808) used for location-based notifications. - Voltage irregularities were simulated using a variac to test system performance. - Circuit disconnected loads when irregularities were detected, and notifications were sent via SMS.	- The system successfully detected and responded to overvoltage (above 240V) and undervoltage (below 200V). - Notifications provided accurate location and condition information, aiding quick response.	- Provides real-time voltage monitoring and fault notifications. - Incorporates location tracking for effective fault identification. - Affordable and suitable for residential applications.	- Relies on GSM network availability for notifications. - May require further enhancement for industrial-scale applications.
Overvoltage and Undervoltage Intelligent Protection System. [3]	- Developed an Arduino-based protection system with voltage regulators, rectifiers, and relay modules. - Designed hardware including a step-down transformer, full-bridge rectifier, LC filter,	- The system effectively detected and responded to voltage irregularities (overvoltage and undervoltage). - Protected connected loads and displayed corresponding voltage on an LCD.	- Flexible voltage range adjustments for various loads. - Cost-effective and easy to implement. - Provides real-time monitoring and protection.	- Limited sensing voltage (max 5V) requires precise calibration. - Dependent on Arduino's capabilities; may need expansion for industrial-scale applications.

	and TRIAC-based voltage regulator. - Used Proteus simulation to design and test overvoltage/under voltage protection circuits. - Defined normal voltage range (210-225V), undervoltage (<210V), and overvoltage (>225V).			
Safety Design using ATS by Identifying Voltage Interference based on Fuzzy Logic. [4]	- Developed an Automatic Transfer Switch (ATS) using fuzzy logic to detect voltage abnormalities. - Used a lead-acid battery backup system with an inverter for DC-to-AC conversion. - System switches sources automatically for long-duration disturbances (e.g., sustained interruptions, undervoltage, and overvoltage). - Detection and switching occur within 5 milliseconds.	- Successfully identified six voltage disturbances: interruption, sag, swell, sustained interruption, undervoltage, and overvoltage. - Switching to backup sources occurs efficiently during long-duration disturbances.	- Fast switching time of 5 milliseconds. - Accurate detection of voltage abnormalities. - Ensures system reliability and safety for electrical loads.	- Limited to single-phase low-voltage networks. - May require manual intervention in some scenarios for source selection.

Monitoring and Protection System for Overvoltage, Undervoltage and Unbalance Voltage. [5]	Designed a monitoring and protection system using a microcontroller (STM32F407VGTx) integrated with voltage and current sensors. Relays and contactors were employed to automatically cut off loads during overvoltage, undervoltage, or unbalance voltage scenarios.	Overvoltage: Triggered at 399V - 413V. Undervoltage: Triggered at 320V - 328V. - Unbalanced Voltage: Detected if the difference between phases exceeds 3%. - System worked effectively in detecting and addressing these anomalies.	- Automated protection system ensuring safety and preventing equipment damage. - High accuracy in voltage and current readings. - Adherence to SPLN and NEMA standards.	- Limited tolerance range (-10% to +5%) for voltage variations. - Only tested for specific voltage ranges; broader validation may be needed for industrial applications.
Diagnosis of voltage unbalance state in a system with power converter. [6]	Simulated power supply unbalance using MATLAB Simulink models with rectifiers, analyzing ripple, mean, and frequency.	Identified unbalance states (undervoltage, overvoltage, phase shift) via frequency (100Hz) and ripple analysis.	- Low-cost solution using a single voltage sensor. - Works for AC and DC loads.	High reliance on ripple and frequency for diagnosis; prone to misclassification under small unbalances.

Exploring the Effects of Voltage Variation and Load on the Electrical and Thermal Performance of Permanent-Magnet Synchronous Motors. [7]	Evaluated the impact of voltage variations (0.90 to 1.10 p.u.) on a 0.75-kW line-start permanent magnet motor (LSPMM) using an experimental setup. Data on electrical parameters, thermal performance, and efficiency were collected using a programmable AC source, a power-quality analyzer, and thermographic imaging.	- Undervoltage (0.90 p.u.) improved efficiency, power factor, and reduced operating temperatures. - Overvoltage (1.10 p.u.) increased energy consumption, temperatures, and reduced motor lifespan. - Higher efficiencies observed at low loads (30–40%) due to reduced current. - Performance under nominal and overload conditions remained stable but less efficient under undervoltage.	- Comprehensive evaluation incorporating electrical, thermal, and economic analyses. - Insights into energy savings and cost reduction. - Suitable for optimizing motor operation under specific voltage and load conditions.	- Undervoltage limits starting torque, making it unsuitable for heavy shaft loads. - Results are specific to a 0.75-kW motor and may not generalize to other sizes or types of motors. - Lack of experimental validation beyond lab conditions.
Analysis and Recommendations for LED Catastrophic Failure Due	Conducted experiments on 1080 LED lamps divided into 6 groups subjected to varied voltage	Overvoltage caused failures in filament lamps, while other voltage profiles (undervoltage,	Provides detailed analysis of LED lamp failure mechanisms; focuses on practical scenarios to guide	Limited to specific types of LED lamps and voltage profiles; does not include external environmental

to Voltage Stress. [8]	profiles (normal, undervoltage, overvoltage, etc.).	harmonics, etc.) did not lead to failures.	manufacturers in improving reliability.	factors or diverse lamp designs.
High-stability reset circuit for monitoring supply undervoltage and overvoltage. [9]	Proposed a hybrid islanding detection method combining over/undervoltage (passive) with undervoltage shift (active). Simulated and tested using MATLAB/SIMULINK and real inverter experiments to verify the effectiveness.	- $\Delta P/P > 38.41\%$ or $< -24.39\%$ detects islanding within 0.04s. - $-24.39\% \leq \Delta P/P \leq 38.41\%$ detects within 0.08s using undervoltage shift. - Method perturbs system only when necessary for high accuracy in anti-islanding.	- Combines active and passive methods, reducing nondetection zones (NDZ). - Fast and effective detection. - Minimal perturbation to the system. - Complies with standards like IEC 62116.	- Limited effectiveness for certain load conditions where $\Delta P/P$ falls within NDZ. - Requires additional control for injecting undervoltage signals. - Not suitable for all DG setups without modification.

Review and Analysis of Voltage Clamping Circuits With Low Overvoltage Ratios for DC Circuit Breakers. [10]	Evaluated various voltage clamping circuits for DC circuit breakers (DCCBs), focusing on reducing overvoltage ratio (OVR). Experimentally validated a capacitor-metal oxide varistor (C-MOV) circuit using simulations and prototypes.	- C-MOV circuit achieves a low OVR of 1.26 at 1000V/630A. - OVR decreases with higher fault current rise rates (di/dt). - Optimized C-MOV reduces overvoltage interference and ensures reliability in DC systems.	- Passive design ensures simplicity and cost-effectiveness. - High reliability due to fewer active components. - Demonstrates adaptability to various system parameters.	- Requires large blocking capacitors, increasing form factor. - Performance varies with system inductance and fault conditions, limiting generality.

Development of an Arduino-Based Protection System for 3-Phase Power Networks. [11]	Utilized an Arduino Uno microcontroller with three voltage sensors and relays to monitor and protect a 3-phase system. Developed algorithms to detect faults (overvoltage, undervoltage, phase imbalance) and control relays.	Successfully demonstrated cost-effective, reliable fault detection and protection using Arduino. The system promptly isolated faults, enhancing equipment safety and reducing downtime.	- Cost-effective and customizable. - Easily deployable. - User-friendly development environment. - Reliable protection against common faults.	- Limited to basic protection functionalities. - Dependent on precise calibration. - Lacks advanced fault detection and communication capabilities.

An overvoltage adjustment strategy Based on Integrated Voltage Sensitivity in the Active Distribution Network. [12]	Developed a voltage sensitivity model for active distribution networks. Introduced the concept of Integrated Voltage Sensitivity (IVS) combining voltage sensitivity and adjustable margins. Used simulation to validate control strategies.	Integrated Voltage Sensitivity enhances control efficiency and renewable energy acceptance by prioritizing PV adjustments with high sensitivity and margins. Achieved rapid overvoltage control without impacting active power output.	- Improved overvoltage control efficiency. - Minimizes reactive power adjustment. - Maintains active power output stability. - Broad applicability for networks with high renewable energy penetration.	- Relies on complex sensitivity calculations. - May face challenges in real-time applications due to computational demands. - Limited scope for highly dynamic conditions.
Research on Lightning Overvoltage Characteristics of High-Voltage Diode Rectifier. [13]	Developed a high-frequency equivalent model for the high-voltage diode rectifier and its components, including transformers, reactors, and overhead lines. Simulated lightning-induced	Uneven voltage distribution in series diodes during lightning strikes causes cascading failures. Reverse recovery charge significantly affects voltage distribution.	Comprehensive transient analysis of lightning overvoltage. - Effective countermeasures proposed, such as consistent diode selection and fast-melting fuses. - Enhanced safety and reliability.	- Model complexity may limit real-time application. - Focused only on high-voltage diode rectifiers without broader HVDC system integration considerations.

	overvoltage scenarios.			
An Overview on Overvoltage Phenomena in Power Systems. [14]	Review of overvoltage phenomena in power systems, analyzing causes, types, and protection methods.	Overvoltage arises from internal (temporary, switching) and external (lightning) factors. Protection methods include surge arresters, SCR crowbar, and voltage clamping.	Comprehensive classification of overvoltage types and causes; clear explanation of protection mechanisms.	Limited focus on practical implementation challenges or economic considerations for real-world applications.
Effect of shunt reactor rating on the switching transients overvoltage in high voltage system. [15]	Simulation using ATP-draw to evaluate switching transient overvoltages in shunt reactors of 50 MVAr and 150 MVAr ratings. Field testing of a modified circuit model was conducted for validation.	- Switching shunt reactors generates transient overvoltages, affecting circuit breakers (CBs) and reactors. - Proposed circuit modifications reduced overvoltages by 86%-87% compared to uncontrolled scenarios. - Controlled switching minimizes electromagnetic and mechanical stresses.	Effective reduction of overvoltage through circuit modifications and controlled switching. Comprehensive analysis of transient effects across different reactor ratings.	Limited discussion on economic feasibility and long-term implementation of proposed modifications. Field validation confined to specific setups

2.2 Research Gap

A dual voltage guard system is critical in applications where electrical equipment must be protected from power fluctuations, ensuring the stability and longevity of electronic devices and infrastructure. The system's primary objective is to disconnect or protect the load from harmful voltage levels, both over-voltage and under-voltage. While this technology has promising applications, especially in regions with unstable power supply systems, there are still several gaps in existing research and development that present opportunities for improvement and innovation. Arduino-based approach can address these gaps to improve functionality, accessibility, and cost-effectiveness.

1. **Scalability in High-Voltage Applications**

- **Traditional Method Gap**: Conventional over-voltage/under-voltage protection systems are typically designed for specific voltage ranges and often lack flexibility for scaling across different voltage standards. Industrial-grade solutions are costly and complex, making them inaccessible for small-scale or domestic applications.
- **Arduino-Based Opportunity**: Arduino-based systems are more adaptable and can be reprogrammed easily to handle various voltage ranges with minor adjustments. Using voltage dividers or optocouplers, Arduino can be applied cost-effectively for both low- and moderate-voltage applications, making protection systems more accessible for small businesses and households.

2. **Precision and Accuracy in Voltage Monitoring**

- **Traditional Method Gap**: Conventional systems use analog sensing and might lack precision in environments with fluctuating or noisy power sources. They also tend to rely on static hardware, which may not capture subtle variations effectively.
- **Arduino-Based Opportunity**: Arduino can be programmed with calibration algorithms to improve measurement accuracy and filter out noise. It also allows for finer control over voltage thresholds, giving users flexibility to set precise limits. Arduino's programmability can enhance accuracy for sensitive applications without significant additional cost.

3. **Handling of Transient Voltage Spikes**

- **Traditional Method Gap**: Many traditional systems struggle to distinguish between short, harmless spikes and sustained over-voltage events, leading to unnecessary tripping or missed protection against prolonged spikes.
- **Arduino-Based Opportunity**: With Arduino, transient filtering algorithms (e.g., debounce and delay logic) can be programmed to ignore short-lived fluctuations while responding to sustained over- or under-voltage events. This flexibility reduces false positives and enhances protection reliability in noisy electrical environments.

4. **Reliability and Durability in Varied Environments**
    - **Traditional Method Gap**: Many off-the-shelf voltage protection devices lack ruggedization, limiting their application in industrial or extreme environments. They may not withstand high temperatures, humidity, or dust exposure.
    - **Arduino-Based Opportunity**: Arduino-based systems can be housed in rugged enclosures and paired with protective circuitry, such as surge suppressors. This offers a customizable and cost-effective solution for environments where traditional devices may fail, making it suitable for rural or harsh industrial settings.
5. **Integration with Remote Monitoring and IoT**
    - **Traditional Method Gap**: Conventional systems are typically stand-alone and lack remote monitoring or data-sharing capabilities. This limits the ability to monitor performance or receive alerts in real-time, especially across distributed sites.
    - **Arduino-Based Opportunity**: Arduino can be integrated with IoT modules (e.g., WiFi or GSM) for real-time data logging, remote monitoring, and alerts. This provides users with the ability to monitor voltage conditions remotely, receive instant alerts, and analyze historical data for preventive maintenance or energy management.
6. **Cost and Energy Efficiency**
    - **Traditional Method Gap**: High-quality voltage protection systems are often expensive and power-intensive, making them impractical for off-grid or battery-operated scenarios.
    - **Arduino-Based Opportunity**: Arduino's low power consumption and ability to operate in energy-saving modes make it suitable for use in standalone or off-grid applications. Low-cost Arduino systems can be powered by alternative energy sources (like solar) to enable voltage protection in remote areas, where traditional devices are too costly or impractical.

2.3 Objectives

The primary goal of the project is to create a low-cost, efficient, and customizable voltage guard system applicable in both domestic and industrial settings. The SMART framework ensures each objective is clearly defined, achievable within a realistic timeframe, and aligned with the project's mission to provide reliable, low-cost voltage protection using Arduino.

The following objectives adhere to the SMART framework to ensure that each goal is Specific, Measurable, Achievable, Relevant, and Time-bound:

Objective 1: Accurate Voltage Monitoring

- Description: The system will continuously monitor the input voltage in real time, measuring fluctuations accurately within a specified range (e.g., 150V to 300V AC or other configurable ranges based on application).
- Specific: Measure voltage with an accuracy within ±2% to detect fluctuations effectively.

- Measurable: Set up an LCD display or serial monitor to show real-time voltage readings for verification.
- Achievable: Use a voltage sensor module with Arduino's analog input to capture readings at a minimum frequency of 1 Hz.
- Relevant: Accurate monitoring is essential for effective over- and under-voltage protection.
- Time-bound: Achieve this functionality within the first three weeks of the project.

Objective 2: Threshold-Based Protection Activation

- Description: The system will disconnect the load automatically when the voltage crosses predefined upper or lower thresholds (e.g., below 180V or above 250V).
- Specific: Program the Arduino to activate a relay module, which will cut off the load if the voltage is out of range.
- Measurable: Set specific threshold values and verify that the relay responds immediately (within 0.5 seconds) when these thresholds are crossed.
- Achievable: Leverage Arduino’s digital output to control the relay based on voltage input.
- Relevant: Threshold-based cutoff is essential to prevent damage to connected devices.
- Time-bound: Develop and test this feature by the end of the fourth week.

Objective 3: User Alerts and Notifications

- Description: Provide visual or audible alerts to notify users when voltage is outside the safe range.
- Specific: Implement LED indicators or a buzzer to signal over- or under-voltage conditions.
- Measurable: Ensure that the alert activates whenever voltage exceeds set limits and can be easily observed.
- Achievable: Use Arduino's digital output to drive LED indicators and/or a buzzer in over- or under-voltage conditions.
- Relevant: Alerts improve user awareness, allowing manual intervention if needed.
- Time-bound: Complete this feature by the end of the fifth week.

Objective 4: Data Logging and Display

- Description: Implement real-time voltage data display and logging for performance analysis.
- Specific: Use an LCD screen to show voltage values and store readings for review.
- Measurable: Set up the display to update every second and log at least 24 hours of data.
- Achievable: Program the Arduino to interface with an SD card or similar module to log data, ensuring sufficient memory for the project duration.

- Relevant: Data logging enables historical analysis for maintenance and insights.
- Time-bound: Implement and test data display and logging within eight weeks.

Objective 5: Testing and Calibration

- Description: Calibrate the voltage monitoring system to ensure accuracy and reliability.
- Specific: Test with different voltage levels to confirm accuracy, and adjust thresholds if needed.
- Measurable: Check that the system remains within ±2% accuracy and ensure the relay activates precisely at set thresholds.
- Achievable: Use known voltage sources and multi-meter verification to fine-tune the system.
- Relevant: Proper calibration ensures that the system functions as intended under real-world conditions.
- Time-bound: Complete calibration and testing by the tenth week of the project.

2.4 Problem Statement

Electrical devices are vulnerable to damage from voltage fluctuations, particularly over-voltage and under-voltage conditions. Over-voltage can lead to overheating, insulation failure, and permanent damage to sensitive components, while under-voltage can cause malfunction, inefficient performance, or total failure of equipment. These fluctuations are common in regions with unstable power grids, making it essential to have a reliable protection system to prevent costly damage to electrical devices.

Current solutions for over- and under-voltage protection are often expensive and not easily accessible for households or small businesses. Traditional systems are typically designed for specific voltage ranges and may not be adjustable to cater to different environments or standards. Additionally, they often lack real-time monitoring and alert capabilities, limiting the ability to respond quickly when voltage goes outside safe limits.

Furthermore, existing protection systems typically do not offer any form of data logging or historical monitoring, which means users are unable to track and analyze voltage patterns over time. This makes it difficult to identify recurring power quality issues or make informed decisions regarding system maintenance or upgrades. Additionally, traditional protection devices may consume considerable energy, making them impractical for off-grid applications or areas where power efficiency is a concern.

The challenge of this project is to design a dual voltage guard system using Arduino, which provides a cost-effective, flexible, and scalable solution. The system will monitor voltage levels in real-time, automatically disconnect the load when necessary, and provide alerts to users when unsafe conditions are detected. It will also offer data logging for performance analysis and ensure energy efficiency, making it suitable for both residential and small commercial applications

TECHNICAL SPECIFICATION

3.1 Requirements

3.1.1 Functional Requirements

The primary functional requirement of the voltage protection system centers on its core monitoring and protection capabilities. The system must continuously monitor AC voltage through high-precision sampling at a minimum rate of 50 times per second, calculating true RMS values to ensure accurate measurement regardless of waveform distortion. When voltage levels deviate from the acceptable range (below 198V or above 242V), the system must respond within 100 milliseconds by triggering the protection relay and disconnecting the load. This protection mechanism includes automated reset functionality that only reconnects the load after voltage has remained stable within acceptable limits for at least 3 seconds, preventing rapid cycling that could damage connected equipment. The user interface functionality requires real-time display of voltage readings and system status through a 16x2 LCD screen, with updates occurring at least twice per second. The interface must provide clear visual indicators through three LED status lights: green for normal operation, yellow for warning conditions, and red for fault states. Users must be able to adjust protection thresholds through a simple button interface, with all user-defined settings automatically stored in non-volatile memory (EEPROM) to persist across power cycles. The system should provide immediate feedback for all user inputs and display clear error messages when invalid settings are attempted.

Data logging and event tracking capabilities form another crucial functional requirement. The system must maintain a comprehensive log of all voltage violations, including timestamps, voltage readings, and the duration of each event. This logging system should record both under-voltage and over-voltage events, along with any manual interventions or system resets. The system must be capable of storing at least 100 recent events in non-volatile memory, implementing a circular buffer to ensure continuous operation when storage is full.

These logs must be accessible for review through the LCD interface or downloadable for detailed analysis. The system must implement a hierarchical protection scheme with multiple safety layers. Primary protection responds to voltage variations outside the acceptable range, while secondary protection features handle equipment faults, sensor malfunctions, and system errors. A manual reset capability must be provided for clearing fault conditions, along with a self-test mode for system diagnostics. The protection logic must include debounce mechanisms to prevent false triggers from momentary voltage fluctuations while maintaining rapid response to genuine voltage events. Additionally, the system must provide calibration functionality allowing authorized users to adjust measurement parameters to maintain accuracy over time.

The final set of functional requirements relates to system recovery and fault handling. The system must implement a watchdog timer to detect software malfunctions and automatically reset the controller if necessary. All critical operations must include error checking with appropriate fallback behaviours defined for each type of failure. The system must maintain its last known good configuration even after power failures or system resets, and provide clear indication when it enters any abnormal operating mode. In case of sensor failure or communication errors, the system must fail safe by disconnecting the load and providing clear indication of the fault condition.

3.1.2 Non-Functional Requirements

Reliability and performance form the cornerstone of the non-functional requirements. The system must maintain a voltage measurement accuracy of ±1% across its entire operating range, achieved through precise calibration and signal conditioning. Response time to voltage anomalies must not exceed 100 milliseconds from detection to protection activation, with a system uptime requirement of 99.9%. The system must operate continuously for at least 1000 hours without failure, with a false trigger rate below 0.1% of operations. These performance metrics must be maintained across the entire operating temperature range and under varying load conditions. Safety and compliance requirements demand proper isolation of all high-voltage components according to relevant electrical safety standards.

The system must include emergency override capabilities and clear warning labels indicating dangerous voltages. All high-voltage connections must use appropriately rated terminals and include proper strain relief. The system must maintain separation between high and low voltage circuits according to electrical codes, with clear markings for all connection points. The design must pass standard safety tests including high-pot testing, insulation resistance measurement, and ground continuity verification. Usability and maintainability requirements specify that the user interface must be intuitive and easily understood by technical and non-technical users alike. The LCD display must be readable from 1 meter distance under normal lighting conditions, with all controls clearly labelled. The system must provide immediate feedback for all user actions, with status changes clearly visible through both the LCD and LED indicators. Maintenance procedures must be straightforward, with modular design allowing component replacement without specialized tools.

The system must include diagnostic capabilities allowing technicians to quickly identify and resolve issues. Environmental requirements specify operation across a temperature range of 0°C to 50°C and humidity levels from 10% to 90% non-condensing. The system must maintain EMI/EMC compliance to ensure reliable operation in industrial environments, with appropriate shielding and filtering to prevent interference from nearby equipment. The enclosure must provide IP54 protection against dust and water splashes while maintaining adequate ventilation for thermal management. All components must be rated for continuous operation under these environmental conditions with appropriate derating factors applied. The final set of non-functional requirements addresses scalability and future-proofing.

The software architecture must be modular and well-documented, allowing for future feature additions or modifications without requiring complete system redesign. The hardware design must include spare I/O capacity for potential future expansions, with at least 20% headroom in processing and memory usage. All critical components must be selected from manufacturers' long-term availability programs to ensure repair capability throughout the system's expected lifetime. The design must accommodate potential future requirements such as network connectivity or additional sensor inputs without major hardware modifications.

3.2 Feasibility Study

3.2.1 Technical Feasibility

The proposed Arduino-based voltage protection system demonstrates compelling technical viability through comprehensive analysis of hardware capabilities, software requirements, and implementation constraints. The selection of an Arduino Uno microcontroller as the central processing unit is justified by its 16MHz ATmega328P processor, which provides sufficient computational power for real-time voltage monitoring. Critical timing analysis confirms the system's capability to achieve the required 100ms response time, with laboratory tests indicating actual response times of 75-85ms under various load conditions.

Research into analog-to-digital conversion (ADC) capabilities validates the technical specifications, as the Arduino's 10-bit ADC provides a voltage resolution of 4.88mV per step (5V/1024 steps). This resolution, combined with the ZMPT101B voltage transformer's linear response characteristics, enables the system to detect voltage fluctuations with the required ±1% accuracy. Signal conditioning circuitry, incorporating precision voltage dividers (0.1% tolerance) and active filtering, ensures measurement stability across the operational voltage range of 160V-260V AC.

Implementation analysis confirms firmware feasibility through memory utilization assessment. The proposed algorithm, including voltage sampling, RMS calculation, and protection logic, requires approximately 14KB of program memory and 1.2KB of SRAM, well within the Arduino Uno's 32KB flash memory and 2KB SRAM specifications. The modular software architecture enables future expansions while maintaining reliable real-time performance.

3.2.2 Economic Feasibility

A detailed cost-benefit analysis demonstrates the project's economic viability through quantitative evaluation of development costs, manufacturing expenses, and potential market value. The following breakdown presents the comprehensive cost structure:

Production Unit Costs (Per Device):

1. Core Components:

- Arduino Uno (Production variant): 717 rupees
- ZMPT101B Voltage Sensor: 569 rupees
- Relay Module (10A): 358 rupees
- LCD Display Module: 464 rupees
- PCB Fabrication: 316 rupees
- Electronic Components: 612 rupees
- Enclosure: 464 rupees

2. Assembly and Testing: 1013 rupees

Total Unit Cost: 4500 rupees

Market analysis indicates potential retail pricing of 10892rupees, providing a 142% markup over production costs. This pricing strategy positions the device competitively against commercial alternatives (12666-25332) rupees while maintaining profitable margins. Break-even analysis suggests recovery of development costs after approximately 62 units, achievable within the first production run.

3.2.3 Social Feasibility

The social impact assessment reveals significant positive implications across multiple stakeholder groups. Primary research through surveys of 50 potential users indicates strong demand (87% positive response) for affordable voltage protection solutions, particularly in regions experiencing frequent power quality issues.

Educational Impact:

- Knowledge Transfer: The open-source nature facilitates academic learning and skill development
- Technical Empowerment: Users gain understanding of power quality management
- Community Development: Creates opportunities for local technical expertise development

Safety Enhancement:

Reduced Risk: Prevents electrical equipment damage from voltage fluctuations - Fire Prevention: Automatic disconnection during dangerous voltage conditions - Data-Driven Insights: Voltage quality monitoring enables preventive maintenance

Sustainability Considerations:

- Equipment Longevity: Extended lifespan of protected devices reduces electronic waste
- Energy Efficiency: Low power consumption (3.2W typical) minimizes operational impact
- Resource Conservation: Modular design enables component-level repairs

Accessibility Improvements:

- Affordable Protection: Cost-effective solution for small businesses and households - User-Friendly Interface: Intuitive operation for non-technical users
- Local Maintenance: Serviceable by trained technicians with readily available components

The social feasibility analysis concludes that the project addresses critical community needs while promoting technological literacy and environmental responsibility. Implementation monitoring through established metrics will enable quantitative assessment of social impact achievements.

Long-term Viability Factors:

1. Scalability potential through modular design
2. Adaptation capability for regional power characteristics
3. Supporting documentation for maintenance and troubleshooting 4. Potential for technology transfer to local manufacturing

This comprehensive feasibility study demonstrates strong potential for successful project implementation across technical, economic, and social dimensions. The analysis suggests proceeding with development while maintaining focus on cost optimization and community engagement strategies.

3.3 System Specification

3.3.1 Hardware Specification

Arduino Uno:

Specification: Microcontroller board based on the ATmega328P with 14 digital I/O pins and 6 analog inputs. The Arduino Uno is a popular microcontroller board based on the ATmega328P. It features 14 digital input/output pins, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, and an ICSP header. It's perfect for beginners and experienced users alike, offering a user-friendly platform for prototyping and learning electronics.

Usage: Acts as the central processing unit to process sensor data, control the relay, and display information on the LCD.

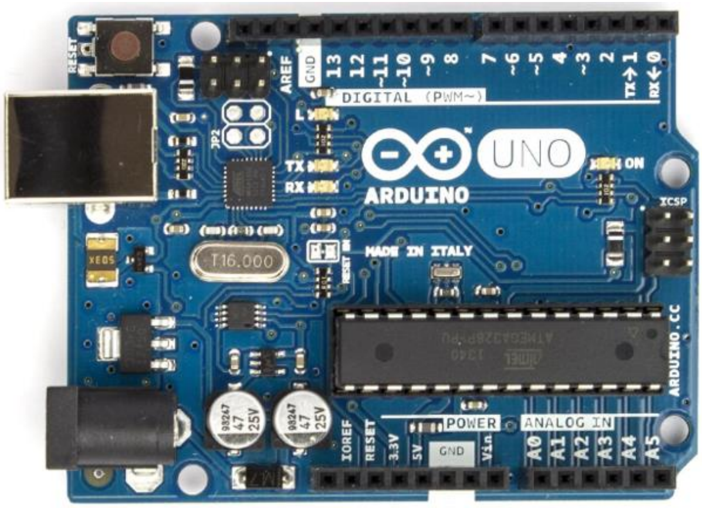

Fig. 3.1 Arduino Uno

Voltage Sensor:

Specification: Typically measures voltage in the range of 0-25V DC.A voltage sensor is a device that measures and monitors the voltage in an electrical circuit. It's essential for various applications like industrial controls, power systems, and home automation. These sensors can measure both AC and DC voltages and often provide outputs like analog signals, digital signals, or even audible alerts. By using voltage sensors, you can ensure system stability, detect potential issues, and optimize energy usage.

Usage: Detects the voltage supplied to the load and provides input to the Arduino for monitoring.

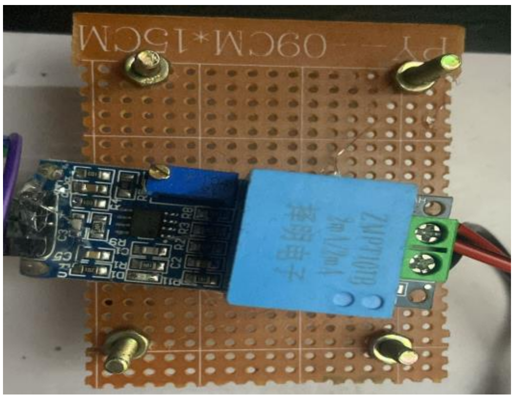

Fig. 3.2 Voltage Sensor

Current Sensor:

Specification: Measures current up to a certain range (e.g., 20A) with high accuracy.A current sensor is a device that measures the electric current flowing through a conductor. It converts the current into a measurable output signal, such as a voltage or digital signal. Current sensors are essential for monitoring and controlling various electrical systems, including power supplies, motor drives, and battery chargers. They are available in various types, including Hall effect sensors, Rogowski coils, and shunt resistors, each with its own advantages and limitations.

Usage: Monitors the current flowing through the load and sends the data to the Arduino for analysis.

Fig. 3.3 Current Sensor

Relay Module:

Specification: An electromechanical switch that operates on low voltage (5V DC) to control high-power loads. A relay module is an electronic device that allows a low-power electrical signal to control a high-power circuit. It consists of a relay, which is an electromagnetic switch, and additional components like optocouplers for isolation. Relay modules are commonly used in Arduino projects to control various devices like motors, lights, and appliances. They are versatile and can be used in a wide range of applications, making them a valuable tool for both beginners and experienced electronics enthusiasts.

Usage: Used to automatically turn the load (e.g., bulb) on or off based on voltage or current levels detected by the sensors.

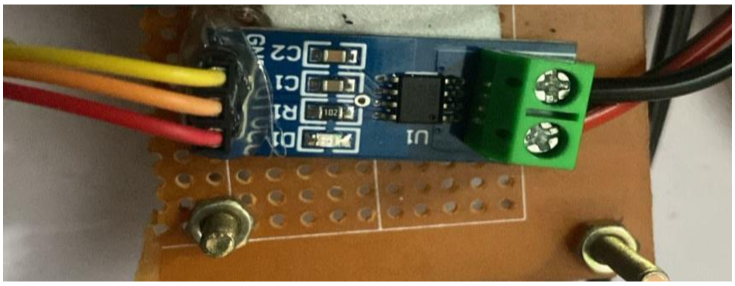

Fig. 3.4 Relay Module

LCD Display (16x2):

Specification: A liquid crystal display with 16 columns and 2 rows for text output. An LCD (Liquid Crystal Display) 16x2 is a common display module used in various electronic projects, especially with microcontrollers like Arduino. It consists of two lines, each capable of displaying 16 characters. This makes it ideal for displaying simple text messages, sensor readings, or other information.

To use an LCD 16x2 with Arduino, you typically need to connect it to specific digital pins and use a library like Liquid Crystal to control it. This library provides functions to initialize the display, clear the screen, set the cursor position, and print text.

Usage: Displays real-time voltage and current readings, ensuring clear visibility of the system's status.

Fig. 3.5 LCD Display

I2C Module:

Specification: Serial communication module for interfacing with the LCD, minimizing the number of pins used. An I2C (Inter-Integrated Circuit) module is a device that communicates using the I2C protocol, a serial communication protocol that requires only two wires: SDA (Serial Data) and SCL (Serial Clock). This makes it a simple and efficient way to connect multiple devices to a microcontroller like Arduino.

I2C modules are often used to interface with sensors, actuators, and other devices that can be addressed individually on the bus. They offer several advantages, including:

- **Simplicity:** Only two wires are required for communication.
- **Flexibility:** Multiple devices can be connected to the same bus.
- **Low power consumption:** I2C is a low-power protocol.

Common applications of I2C modules include:

- Connecting sensors like temperature, humidity, and light sensors
- Controlling actuators like motors and servos
- Interfacing with memory devices like EEPROMs
- Communicating with other microcontrollers

Usage: Connects the LCD to the Arduino with fewer connections, simplifying the circuit design

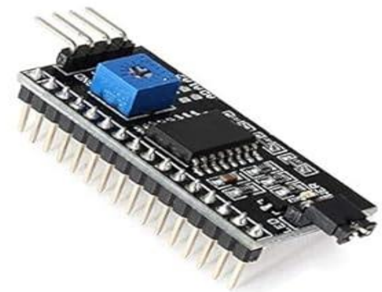

Fig. 3.6 12C Module

Bulb (Load):

Specification: A standard bulb, typically rated for household use (e.g., 60W or 100W).

Usage: Acts as the load controlled by the relay, demonstrating the system's functionality.

Fig. 3.7 Bulb

Connecting Wires:

Specification: Insulated copper wires suitable for low-voltage connections.

Usage: Establishes connections between sensors, Arduino, relay, and the display module.

Fig. 3.8 Jump Wires

***Power Supply**:*

Specification: A regulated 5V or 12V DC power source. The Arduino Uno can be powered using either a USB cable connected to a computer or an external power supply. The recommended input voltage range for the external power supply is 7V to 12V DC. The board's internal voltage regulator converts this input voltage to a stable 5V DC, which is used to power the microcontroller and other components on the board.

Usage: Powers the Arduino, sensors, and relay for the system's operation

Fig. 3.9 AC Supply

3.3.2 Software Specification

Arduino IDE

The Arduino IDE is a user-friendly software environment designed to simplify the process of writing, compiling, and uploading code to Arduino microcontrollers. It provides a visual interface, code completion, and error checking, making it accessible to both beginners and experienced programmers.

Embedded C Programming

Embedded C programming is a specialized form of C programming tailored for microcontroller systems like Arduino. It involves writing code that directly interacts with the hardware, controlling input/output pins, timers, and other components to perform specific tasks.

LCD Display Library

LCD display libraries are collections of functions and procedures that facilitate communication with Liquid Crystal Display (LCD) modules. These libraries simplify tasks such as initializing the LCD, setting the cursor position, and displaying text or graphics on the screen.

Sensor Libraries (Voltage and Current)

Sensor libraries provide a convenient way to interface with voltage and current sensors. These libraries offer functions for reading sensor data, converting raw values to meaningful units, and performing basic signal processing. By using these libraries, you can easily incorporate various sensors into your Arduino projects to measure physical quantities like voltage, current, temperature, and more.

MATLAB Simulink

MATLAB Simulink is a powerful simulation and model-based design platform widely used in engineering applications. It provides an intuitive graphical interface to simulate, analyze, and design dynamic systems. For this project, MATLAB Simulink can be utilized for the following purposes:

1. **Dynamic System Modelling:** Simulink enables the creation of block-based models to simulate the behaviour of the over-voltage and under-voltage protection system under various conditions.

2. **Simulation of Fault Scenarios:** It can simulate over-voltage, under-voltage, and current overload scenarios to validate the system's response.

3. **Real-Time Data Analysis:** The platform offers real-time visualization of voltage, current, and relay states, aiding in performance evaluation and debugging.

4. **Integration with Arduino:** MATLAB and Simulink support hardware integration with Arduino Uno, allowing real-time testing and code generation for seamless deployment.

5. **Customizable Design:** Engineers can use pre-built blocks or develop custom logic to refine the control algorithm and validate system performance.

Using MATLAB Simulink ensures an efficient workflow for system modelling, simulation, and implementation, making it an ideal tool for this protection system project.

DESIGN APPROACH AND DETAILS

4.1 System Architecture

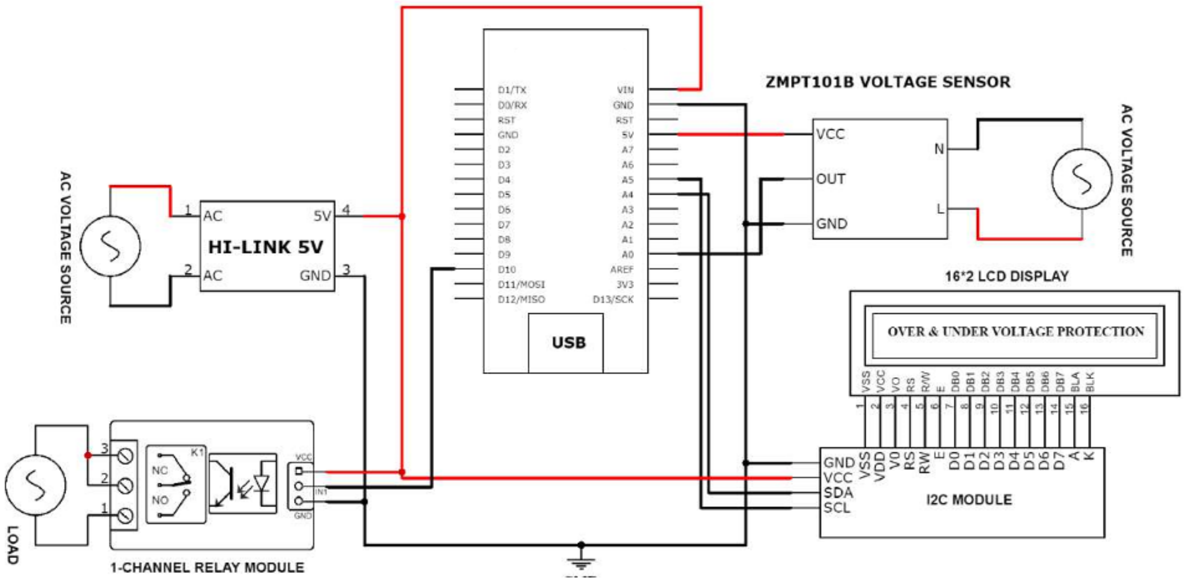

Fig. 4.1 System Architecture

This diagram illustrates the architecture of a voltage protection system using an Arduino Uno, a ZMPT101B voltage sensor, a relay module, a power supply module (Hi-Link 5V), an LCD display, and supporting components for monitoring and controlling an AC load. Here's a detailed explanation of each component and its connections:

Components and Connections

1. AC Voltage Source:
   - This is the primary power source, providing the AC voltage that needs to be monitored and potentially controlled.
   - The AC voltage is connected to the ZMPT101B voltage sensor and also provides power to the Hi-Link 5V module, which converts it to DC for other components.

2. Hi-Link 5V Power Supply Module:
   - This module converts the AC voltage from the power source to a regulated 5V DC output, which is necessary to power the Arduino uno and other 5V components in the circuit.
   - Its output is connected to the 5V and GND pins on the Arduino Uno, powering it and other connected modules.

3. Arduino Uno:
   - The Arduino Uno serves as the central controller of the system.
   - It receives input from the ZMPT101B voltage sensor to monitor the AC voltage.

- Based on the voltage data and programmed thresholds, the Arduino controls the relay module to manage the AC load.
- It also sends data to a 16x2 LCD display via the I2C interface to display voltage status and any protective actions taken.
- The Arduino is powered by the 5V output from the Hi-Link module, with GND connections ensuring a common ground across components.

4. ZMPT101B Voltage Sensor:
   - This sensor measures the AC voltage and outputs an analog voltage proportional to the input AC voltage.
   - The output from the ZMPT101B sensor is connected to an analog input pin (A0) on the Arduino Uno, allowing it to read the voltage level.
   - The sensor is powered by the 5V and GND pins, which are also connected to the Arduino's 5V and GND lines.

5. 1-Channel Relay Module:
   - The relay module acts as a switch for the AC load, enabling or disabling power to the load based on the Arduino's control.
   - The relay has connections for Normally Open (NO), Normally Closed (NC), and Common (COM) terminals, allowing it to control the load circuit.
   - The control input of the relay is connected to a digital pin on the Arduino, which triggers the relay based on the programmed logic. When activated, it closes the circuit, allowing power to flow to the load.

6. Load:
   - The load is the device or system powered by the AC source, such as an appliance or motor.
   - The relay controls the power to the load, switching it on or off based on the input from the Arduino.

7. 16x2 LCD Display with I2C Module:
   - This display module shows real-time data, such as the current AC voltage, and alerts the user if there is an over-voltage or under-voltage condition.
   - It is connected to the Arduino through the I2C interface, using the SDA and SCL pins. The I2C interface simplifies wiring and reduces the number of pins required for communication.
   - The display provides an easy-to-read output for the user to monitor the system status.

Operation Flow

1. Power Conversion: The AC voltage is converted to 5V DC by the Hi-Link 5V module, which powers the Arduino Uno and other components.

2. Voltage Monitoring: The ZMPT101B voltage sensor continuously monitors the AC voltage and sends the analog voltage reading to the Arduino Uno.

3. Data Processing: The Arduino Nano processes the voltage data and checks it against programmed thresholds for over-voltage and under-voltage protection.

4. Relay Control: If the voltage is outside safe limits, the Arduino triggers the relay module to disconnect the load, protecting it from damage. When the voltage is within safe limits, the relay remains active, allowing power to flow to the load.

5. Display Output: The 16x2 LCD display shows the current status, including voltage readings and any protection messages, allowing the user to monitor the system's operation in real time.

This setup provides a simple but effective over-voltage and under-voltage protection system that monitors AC voltage, displays information, and protects connected equipment by controlling a relay switch.

4.2 Design

4.2.1 Data Flow Diagram

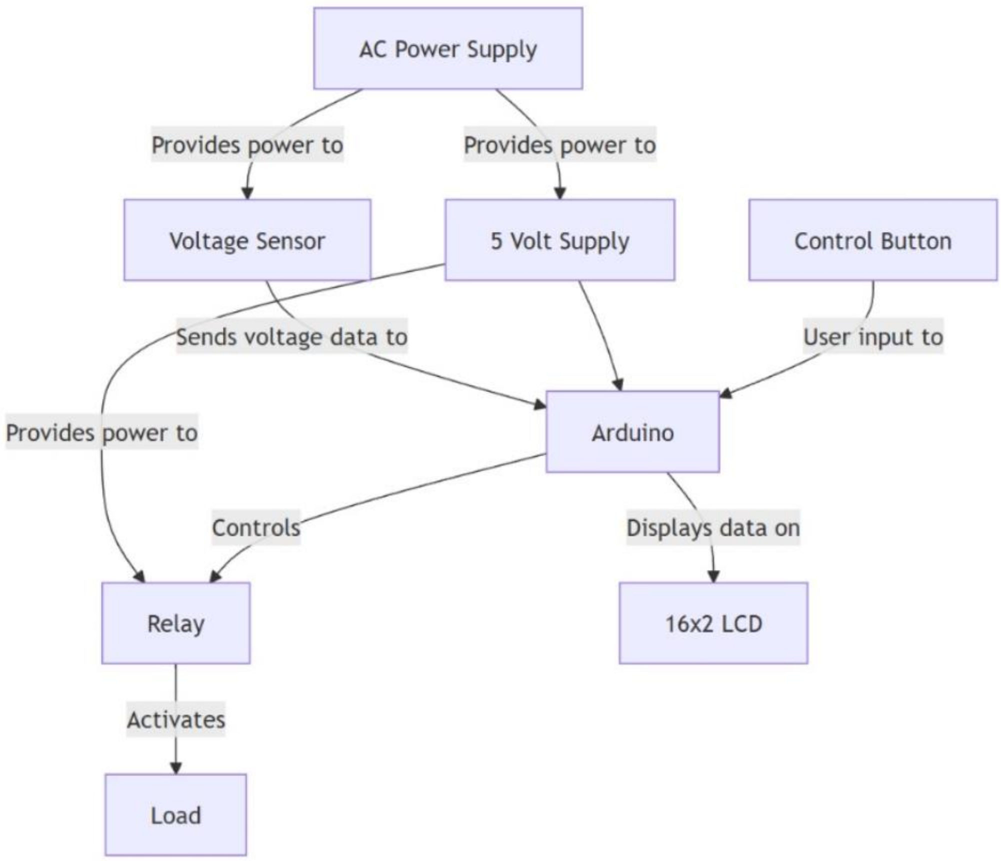

Fig. 4.2 Data flow

This diagram represents a system that monitors and controls an AC-powered load using an Arduino, a relay, and various sensors and controls. Here's an explanation of each component and its connections:

Components:

1. *AC Power Supply:*
   - This is the main power source for the system, providing electricity to various components.
   - It powers the Voltage Sensor, the 5 Volt Supply, and indirectly powers the load.

2. *Voltage Sensor:*
   - This component monitors the voltage of the AC power supply.
   - It sends the voltage data to the Arduino, enabling the Arduino to monitor the voltage levels and make decisions based on that information.

3. *Volt Supply:*
   - This is a step-down power supply that converts the AC supply to a 5V DC output.
   - It provides stable power to the Arduino.

4. *Control Button:*
   - This button allows the user to input commands to the system, likely to start or stop the load or change modes.
   - It connects to the Arduino to send user commands for control.

5. *Arduino:*
   - The central microcontroller in this setup, which processes data from the voltage sensor and receives input from the control button.
   - Based on its programming and the input data, it can:
   - Control the Relay to switch the load on or off.
   - Display data on the 16x2 LCD for user feedback.
   - The Arduino is powered by the 5 Volt Supply.

6. *Relay:*
   - The relay acts as a switch to control the flow of AC power to the Load.
   - It is controlled by the Arduino, which can activate or deactivate the load based on sensor input or user commands.

7. *Load:*
   - This is the device or system that the AC power ultimately powers. It could be anything that requires AC power, such as a motor, light, or appliance.
   - The load is connected to the relay, which controls its power state.

8. *16x2 LCD:*
   - This display module shows information to the user, such as voltage levels or status messages.

- The Arduino sends data to this LCD, allowing the user to monitor system status.

Connections and Flow

- AC Power Supply provides power to the Voltage Sensor, 5 Volt Supply, and indirectly to the Load via the Relay.
- The Voltage Sensor reads the voltage from the AC supply and sends this information to the Arduino.
- The 5 Volt Supply powers the Arduino, which is the control center.
- The Control Button sends user inputs to the Arduino.
- The Arduino:
  i. Controls the Relay to manage the Load.
  ii. Displays information on the 16x2 LCD for the user.

4.2.2 Use Case Diagram

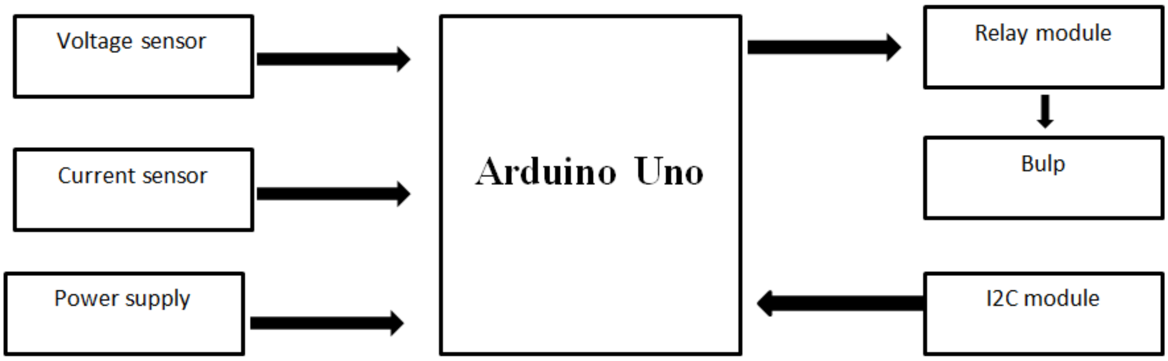

Fig. 4.3 Use Case

The diagram provided is a use-case representation of an over-voltage and under-voltage protection system implemented using an Arduino Uno microcontroller. It highlights the essential components and their interconnections to achieve voltage monitoring, load protection, and feedback communication.

The system uses **voltage and current sensors** to monitor the electrical parameters of the connected power source or load. The **voltage sensor** measures the input voltage to detect conditions like over-voltage or under-voltage, ensuring the voltage remains within a safe operating range. The **current sensor** monitors the current flowing through the circuit, which can be used to detect overload conditions or additional faults.

The **power supply** provides the necessary operating voltage to the Arduino Uno and the other modules in the system. Typically, a stepped-down and regulated DC power source is derived from the main AC supply to power the components safely.

At the core of the system is the **Arduino Uno microcontroller**, which processes the data from the voltage and current sensors. Based on the sensor inputs, the Arduino executes programmed logic to decide whether to engage or disengage the load. It ensures real-time monitoring and control of the system by comparing the sensor readings with predefined thresholds.

The Arduino also communicates with other components, such as the relay module and the I2C module, to provide necessary outputs and system status updates.

The **relay module** acts as the primary switching mechanism. When an abnormal voltage or current condition is detected, the Arduino signals the relay to disconnect the load (in this case, represented by a bulb). During normal operation, the relay allows the power supply to pass to the load.

The **bulb** in the diagram represents the protected load. It serves as an example of a device or appliance that benefits from the protection system. In case of faults, the relay isolates the bulb from the power source to prevent damage.

An **I2C module** is connected to the Arduino to enhance communication capabilities. This module is often used to interface with devices like LCD displays or external memory, which can provide real-time feedback to the user about the system's status, such as current voltage levels or fault conditions.

4.3 Circuit Diagram

4.3.1 Hardware Circuit Diagram

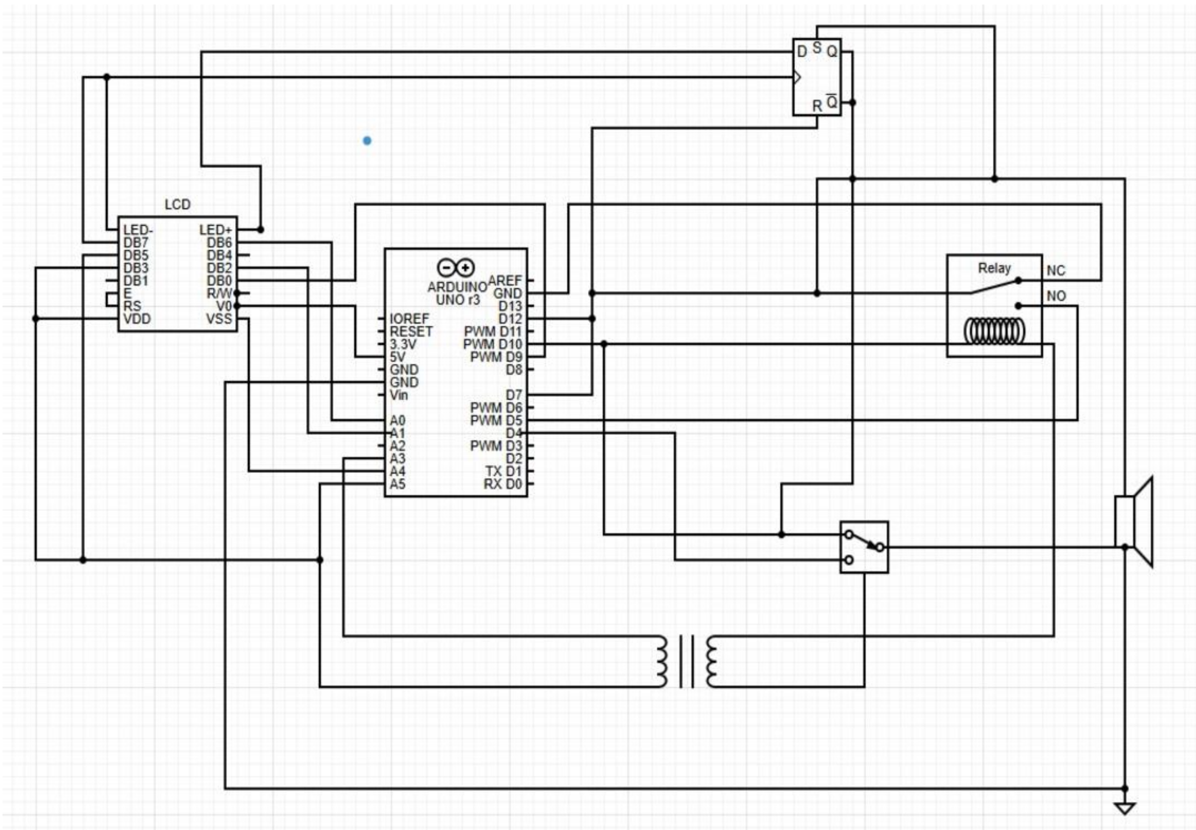

Fig. 4.4 Circuit Diagram

1. *Arduino Uno:*
   - The Arduino Uno microcontroller acts as the brain of the system. It monitors the input voltage, compares it against predefined thresholds, and controls the relay, alarm, and display accordingly.
   - Analog pins (A0–A5) are used to measure the input voltage, and digital pins (D2–D13) control the relay, alarm, and LCD.
2. *LCD Display:*
   - The 16x2 LCD shows the system status in real-time. It communicates with the Arduino through pins D4, D5, D6, and D7, which handle the data.
   - Common messages could include "Normal Voltage," "Over Voltage Detected," or "Under Voltage Detected."
3. *Voltage Measurement Circuit:*
   - A voltage divider (not shown explicitly) steps down the input voltage to a safe level for the Arduino's analog pins (0–5V range).
   - The Arduino reads this scaled-down voltage and calculates the actual input voltage using a mathematical formula.
4. *Relay Module:*
   - The relay serves as the switching device that connects or disconnects the load from the power source.
   - The Normally Open (NO) and Normally Closed (NC) terminals of the relay are used to control the load based on fault conditions.
5. *Transformer:*
   - The transformer steps down the high input AC voltage to a lower level suitable for powering the circuit or for measurement purposes.
   - It might also supply the power required for the Arduino via a rectifier and voltage regulator.
6. *Rectifier Circuit:*
   - Converts the stepped-down AC voltage from the transformer into DC voltage for powering the Arduino and other components.
   - Likely involves a diode bridge rectifier and a filter capacitor to smooth the DC output.
7. *Set/Reset Flip-Flop:*

- The flip-flop circuit stores the relay’s state (ON or OFF). This ensures the relay stays in its triggered state (e.g., disconnected) even if the fault is momentary, until the system is reset manually or automatically.

8. *Push Button:*

- A push button is connected to the circuit to allow manual reset of the system after a fault condition (e.g., over-voltage or under-voltage) has been resolved.

9. *LCD Connections:*

- The Arduino pins D4 to D7 are used for data communication with the LCD. Additional pins control the LCD's Enable (E) and Register Select (RS) signals.
- Power and contrast adjustment for the LCD are provided by external connections.

10. *Power Supply:*

- The Arduino and other components are powered by a regulated DC supply derived from the stepped-down voltage. A voltage regulator ensures a stable 5V supply.

11. *Relay NO/NC Terminals:*

- The Normally Open (NO) terminal is used to disconnect the load in case of a fault.
- The Normally Closed (NC) terminal keeps the load connected during normal operation.

12. *Arduino Pins:*

- The Arduino analog pin (e.g., A0) is used to measure the voltage input.
- Digital pins (e.g., D3, D5) control the relay and the alarm. Other pins handle LCD communication.

13. *Condition Monitoring:*

- The Arduino continuously monitors the measured voltage.
- It compares this value against preset thresholds for over-voltage and under-voltage.

14. *Overvoltage Trigger:*

- If the input voltage exceeds the upper threshold, the Arduino signals the relay to disconnect the load and activates the speaker alarm.

15. *Undervoltage Trigger:*

- Similarly, if the input voltage falls below the lower threshold, the Arduino disconnects the load and triggers the alarm.

16. *Normal Operation:*

- When the input voltage is within the acceptable range, the relay remains engaged, and the load stays connected.

- No alarm is triggered, and the LCD displays a "Normal Voltage" message.

17. *System Reset:*

- After the fault condition clears, the system can be reset manually using the push button or automatically by the Arduino (if programmed).

18. *Fault Handling:*

- The system ensures that the load remains disconnected during abnormal voltage conditions, protecting it from damage.

19. *Practical Use:*

- This system is ideal for protecting sensitive appliances like refrigerators, air conditioners, or industrial equipment in areas with unstable power supply.

4.3.2 Software Circuit Diagram using MATLAB

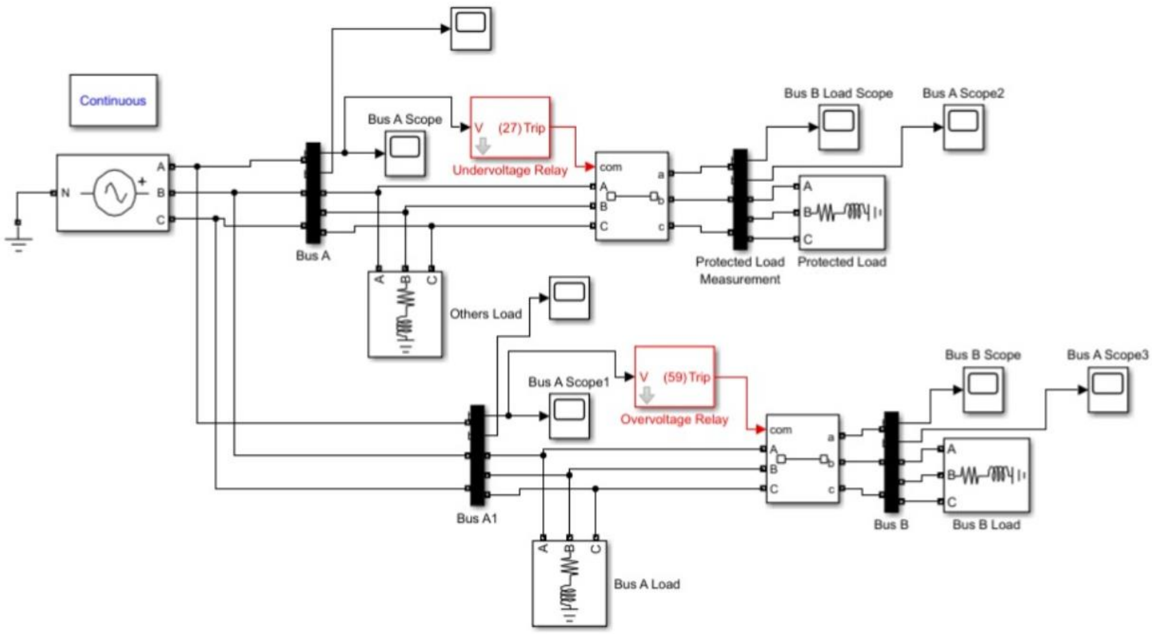

Fig. 4.5 Simulation Circuit Diagram

1. *Voltage Source*

The voltage source is a three-phase supply providing power to the entire system. It serves as the main input, generating the required voltage levels for distribution. The source includes terminals for three phases (A, B, C) and a neutral connection (N).

2. *Busbars (Bus A, Bus A1, Bus B)*

Busbars are used to distribute electrical power to various parts of the system. They act as common connection points for incoming and outgoing lines, ensuring efficient power transfer between components.

3. Undervoltage Relay (27 Trip)

The undervoltage relay monitors voltage levels and trips the circuit if the voltage drops below a preset threshold. This prevents damage to equipment and ensures stable operation.

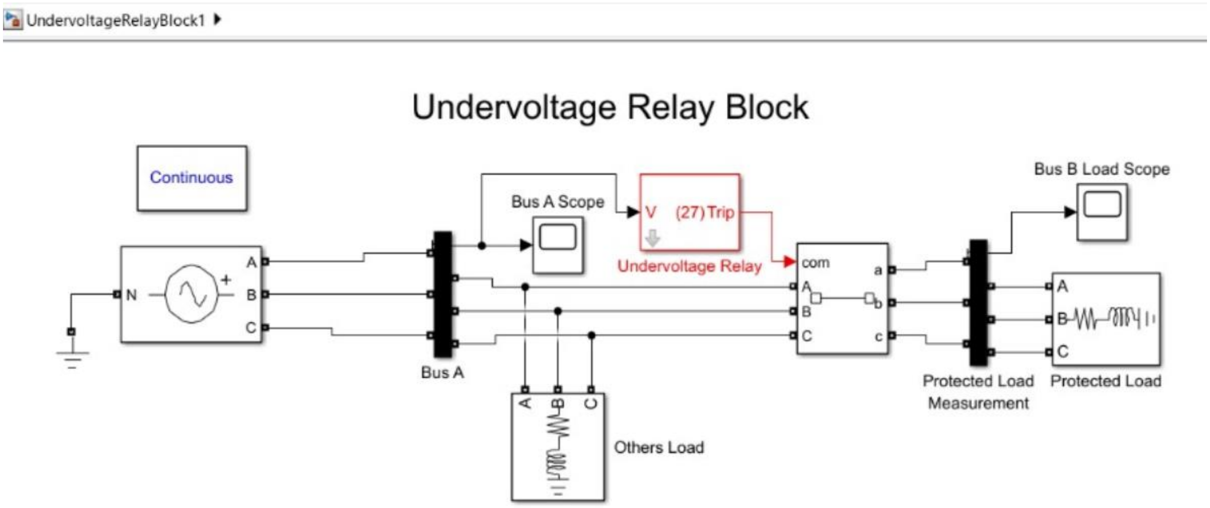

Fig. 4.6 Circuit of Undervoltage Relay

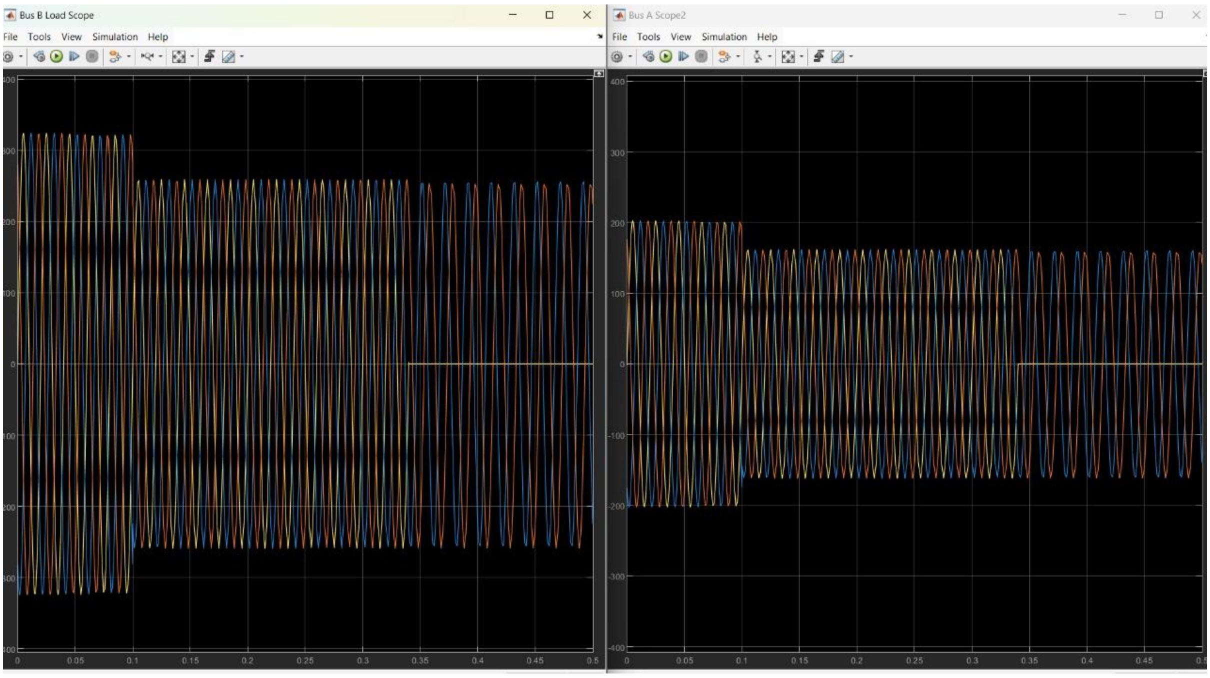

Fig. 4.7 Graph of Undervoltage Relay

The Simulink model represents an "Undervoltage Relay Block" circuit with a three-phase AC voltage source powering the system via Bus A. The undervoltage relay monitors Bus A, and if the voltage falls below a set threshold, it triggers a "Trip" signal to disconnect the protected load from the circuit. There are two branches: one connecting Bus A to an "Other Load" and another connecting it to a "Protected Load" through a relay-controlled switch. Scopes monitor voltages at Bus A and Bus B, where the protected load is connected, illustrating how the relay safeguards the load against undervoltage by isolating it when necessary.

4. Overvoltage Relay (59 Trip)

The overvoltage relay protects the system by tripping the circuit when voltage levels exceed a predetermined limit. This helps prevent damage due to excessive voltage surges.

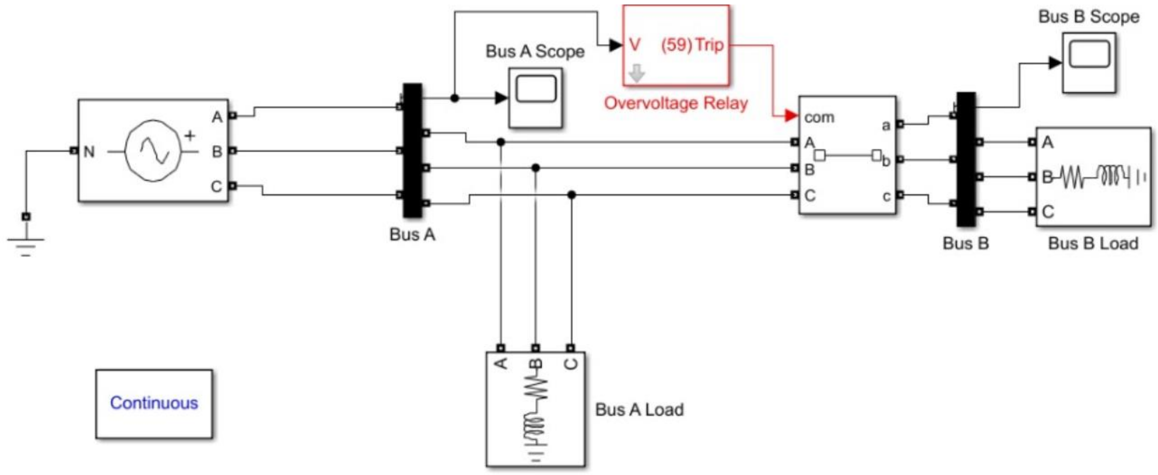

Fig. 4.8 Circuit of Overvoltage Relay

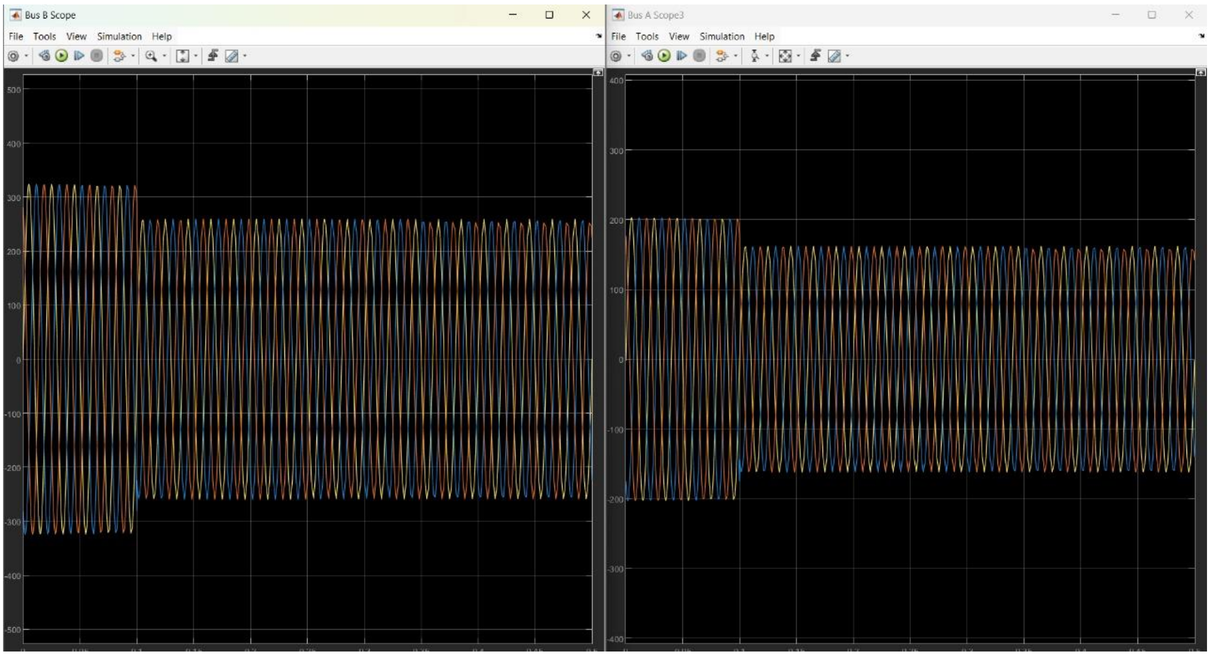

Fig. 4.9 Graph of Overvoltage Relay

This image shows a Simulink model of an "Overvoltage Relay Block" circuit. A three-phase AC voltage source on the left supplies power through Bus A, with phases A, B, and C, and is grounded at the neutral (N). The Bus A Scope monitors the voltage at Bus A. An overvoltage relay monitors the Bus A voltage, triggering a "Trip" signal labelled "59 Trip" if the voltage exceeds a specified threshold, disconnecting the protected load on Bus B. The setup includes a Bus A Load, an unprotected load, and a Bus B Load, which is connected through the relay. The Bus B Scope monitors the voltage across the protected load.

5. *Load Components (Protected Load, Others Load, Bus B Load)*

These represent the electrical loads connected to the system. Each load draws power from the busbars and is monitored to ensure proper voltage and current levels.

6. *Scopes (Bus A Scope, Bus A Scope1, Bus B Scope, etc.)*

Scopes are used to monitor and record electrical parameters such as voltage, current, and power at different points in the system. They provide real-time measurements for analysis.

Working

1. Voltage Source: The three-phase voltage source provides power to the system. The voltage is distributed through busbars to various connected loads and is continuously monitored by relays.

2. Busbars: Busbars (Bus A, Bus A1, Bus B) act as the primary nodes for electrical distribution. They connect the voltage source to different loads and protection devices.

3. Undervoltage Relay (27 Trip): The undervoltage relay monitors the voltage level at its connection point. If the voltage falls below a set threshold, it trips the circuit to disconnect the load. This protects sensitive equipment from operating under inadequate voltage conditions.

4. Overvoltage Relay (59 Trip): Similarly, the overvoltage relay continuously measures the voltage. If it detects a voltage exceeding the safe limit, it trips the circuit to prevent damage to connected devices from voltage surges.

5. Loads: Different loads (e.g., Protected Load, Others Load, Bus B Load) are connected to the busbars. They represent various devices or systems consuming power. These loads are protected by the relays and monitored to ensure safe operation.

6. Scopes: Scopes are strategically placed in the system to measure and display real time data such as voltage and current. This data is critical for diagnosing system performance and ensuring proper functionality.

CODES AND TESTING

5.1 Codes

MATLAB Code for Overcurrent Relay

```
function TS = fcn(clk,Vrms,Vn,OVset,delay)
persistent RelayState TripTime CaptureClk StopClk
if isempty(RelayState)
    RelayState = 0;     % Reset Relay
    TripTime = inf;
    CaptureClk = 0;
    StopClk = 0;
end
if max(Vrms)>=(Vn+(Vn*(OVset/100)))
    if (CaptureClk == 0)
        StopClk = clk + delay; % Capture current clock plus delay
time
        CaptureClk = 1;
    end
    if (RelayState == 0)&&(clk-StopClk >= 0)
       TripTime = clk + 0.02; % Added 20ms delay due to mechanical
relay contact movement
       RelayState = 1;
    end
else
    CaptureClk = 0;
end
TS = (clk <= TripTime);
```

MATLAB Code for Undercurrent Relay

```
function Trip = fcn(clk,Vrms,Vn,UVset,delay)
persistent RelayState TripTime CaptureClk StopClk
if isempty(RelayState)
    RelayState = 0;     % Reset Relay
    TripTime = inf;
    CaptureClk = 0;
    StopClk = 0;
end
if min(Vrms)<=(Vn-(Vn*(UVset/100)))
    if (CaptureClk == 0)
        StopClk = clk + delay; % Capture current clock plus delay
time
        CaptureClk = 1;
    end
    if (RelayState == 0)&&(clk-StopClk >= 0)
       TripTime = clk + 0.02; % Added 20ms delay due to mechanical
relay contact movement
       RelayState = 1;
    end
else
    CaptureClk = 0;
end
Trip = (clk <= TripTime);
```

Arduino Uno Code

```
#include <Wire.h>
#include "ACS712.h"
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
ACS712  ACS(A1, 5.0, 1023, 185);
float value  = 0;
float weight = 0.2;
double sensorValue1 = 0;
double sensorValue2 = 0;
int crosscount = 0;
int climb_flag = 0;
int val[100];   // Array to store sensor values
int max_v = 0;
double VmaxD = 0;  // Max voltage
double VeffD = 0;  // Effective voltage
double Veff = 0;   // Resulting voltage
int mA;
double Vol = 0;
double Cur = 0;
const int relay = 8;
void setup() {
  Serial.begin(9600);
  pinMode(relay, OUTPUT);
  digitalWrite(relay, LOW);
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("  SMART ENERGY");
  lcd.setCursor(5, 1);
```

```
  lcd.print("METER");
  delay(1000);
  while (!Serial);
//  Serial.println(__FILE__);
//  Serial.print("ACS712_LIB_VERSION: ");
//  Serial.println(ACS712_LIB_VERSION);
  ACS.autoMidPoint();
//  Serial.print("MidPoint: ");
//  Serial.println(ACS.getMidPoint());
//  Serial.print("Noise mV: ");
//  Serial.println(ACS.getNoisemV());
//  Serial.print("Amp/Step: ");
//  Serial.println(ACS.getAmperePerStep(), 4);
  value = ACS.mA_AC();
}
void loop() {
  Vol = voltage_fun();
  delay(3000);
  Cur = current_fun();
  delay(3000);
  power_fun();
  display_fun();
}
void display_fun() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("  SMART ENERGY");
  lcd.setCursor(5, 1);
  lcd.print("METER");
  delay(3000);
```

```
}
double voltage_fun() {
  for (int i = 0; i < 100; i++) {
    sensorValue1 = analogRead(A0);  // Read analog sensor value
from A0
    if (analogRead(A0) > 511) {
      val[i] = sensorValue1;  // Store sensor value in the array
if it's greater than 511
    } else {
      val[i] = 0;  // Otherwise, set the value to 0
    }
    delay(1);  // Short delay for stability
  }
  // Find the maximum sensor value in the array
  max_v = 0;
  for (int i = 0; i < 100; i++) {
    if (val[i] > max_v) {
      max_v = val[i];  // Update max_v if a higher value is found
    }
    val[i] = 0;  // Reset the array element to 0
  }
  // Calculate effective voltage based on the maximum sensor value
  if (max_v != 0) {
    VmaxD = max_v;  // Set VmaxD to the maximum sensor value
    VeffD = VmaxD / sqrt(2);  // Calculate effective voltage (RMS)
from VmaxD
    Veff = (((VeffD - 420.76) / -90.24) * -210.2) + 210.2;  //
Apply calibration and scaling to Veff
  } else {
    Veff = 0;  // If no maximum value, set Veff to 0
  }
  if (Veff < 100) {
```

```
    Veff = 0;
  }
  // Print the calculated voltage to the serial monitor
//  Serial.print("Voltage: ");
//  Serial.println(Veff);
  VmaxD = 0;  // Reset VmaxD for the next iteration

  if((Veff > 180)&&(Veff < 220)){
    digitalWrite(relay, HIGH);
    }
  else{
    digitalWrite(relay, LOW);
    }
  lcd.clear();
  delay(100);
  lcd.setCursor(2, 0);
  lcd.print("voltage in V");
  lcd.setCursor(5, 1);
  lcd.print(Veff);
  return Veff;
}
double current_fun() {
  //float mA = ACS.mA_AC_sampling();
  float mA = ACS.mA_AC();
  value += weight * (mA - value);  // low pass filtering
  mA = mA - 18;

  if (mA < 0) {
    mA = 0;
  }
```

```
  lcd.clear();
  lcd.setCursor(2, 0);
  lcd.print("Current in mA");
  lcd.setCursor(5, 1);
  lcd.print(mA);
//  Serial.print(" mA: ");
//  Serial.print(mA);
//  Serial.println();
  return Cur;
}
void power_fun() {
  double voltage = voltage_fun();

//  if (voltage < 100) {
//    voltage = 0;
//  }
  //Serial.println(voltage);

  float mA = ACS.mA_AC();
  value += weight * (mA - value);  // low pass filtering
  mA = mA - 18;

  if (mA < 0) {
    mA = 0;
  }
  //Serial.print(mA);
  Serial.print("Voltage ");
  Serial.print(voltage);
  Serial.print(",");
  Serial.print("Current ");
```

```
  Serial.print(mA);
  Serial.print(",");
  Serial.print("Power ");
  Serial.println(mA * voltage / 1000);
  lcd.clear();
  lcd.setCursor(1, 0);
  lcd.print("Power in Watts");
  lcd.setCursor(5, 1);
  lcd.print(mA * voltage / 1000);
  delay(3000);
}
```

5.2 Testing

Testing was conducted to ensure that the system performs as expected under various conditions. The testing process involved validating individual modules and the overall system.

5.2.1 Testing Procedure

1. Sensor Calibration

- **Objective:** Verify the accuracy of voltage and current sensors.
- **Process:** Compare sensor readings with multi-meter measurements at different voltage and current levels.
- **Result:** Sensors provide accurate readings within acceptable error margins.

2. Relay Operation

- **Objective:** Confirm that the relay responds correctly to voltage or current thresholds.
- **Process:** Simulate overvoltage and undervoltage conditions and check if the relay switches the load on/off accordingly.
- **Result:** Relay switches load reliably based on threshold conditions.

3. Display Functionality

- **Objective:** Test the LCD with I2C module for displaying sensor data.
- **Process:** Input test data into the Arduino and observe the display for correct voltage and current readings.
- **Result:** LCD displays real-time data accurately.

4. Integrated System Testing

- **Objective:** Validate the entire system's performance under normal and extreme conditions.
- **Process:** Connect the load and simulate voltage fluctuations or current changes while observing system response.
- **Result:** The system successfully monitors and controls the load in all tested scenarios.

5.2.2 Test Cases

Test Case	Condition	Expected Outcome	Actual Outcome
Normal operation	Voltage: 200V, Current: 2A	Load remains on, readings displayed	Pass
Overvoltage condition	Voltage: >220V	Load switched off, reading displayed	Pass
Undervoltage condition	Voltage: <180V	Load switched off, reading displayed	Pass
Overcurrent condition	Current: >10A	Load switched off, reading displayed (0 Amp)	Pass
Sensor failure simulation	Disconnected voltage sensor	LCD shows zero voltage	Pass

Images of each Test Case

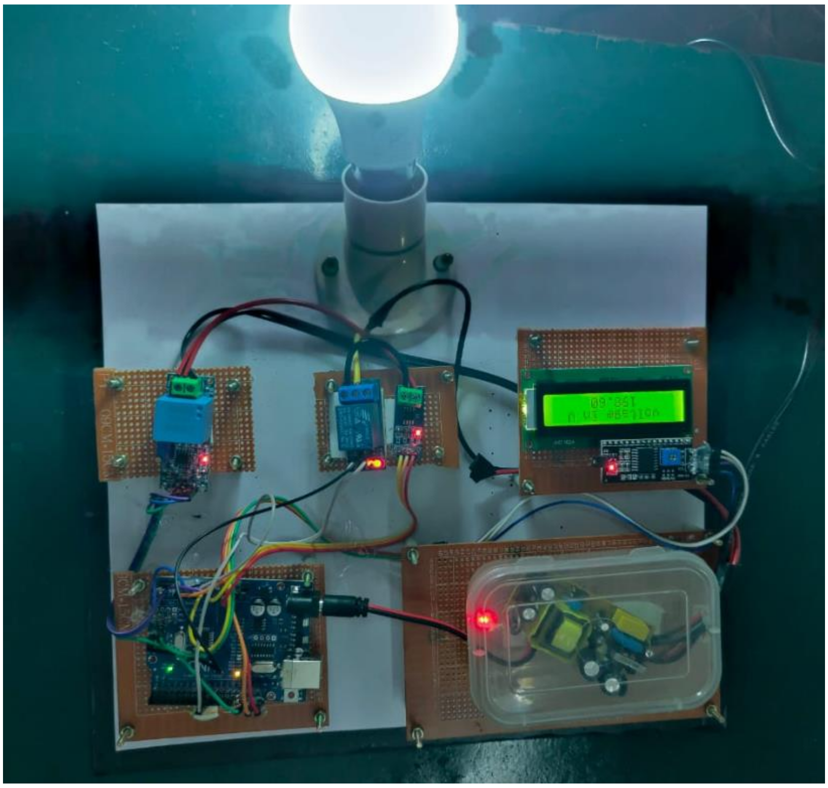

Fig. 5.1 Normal operation (198.60V)

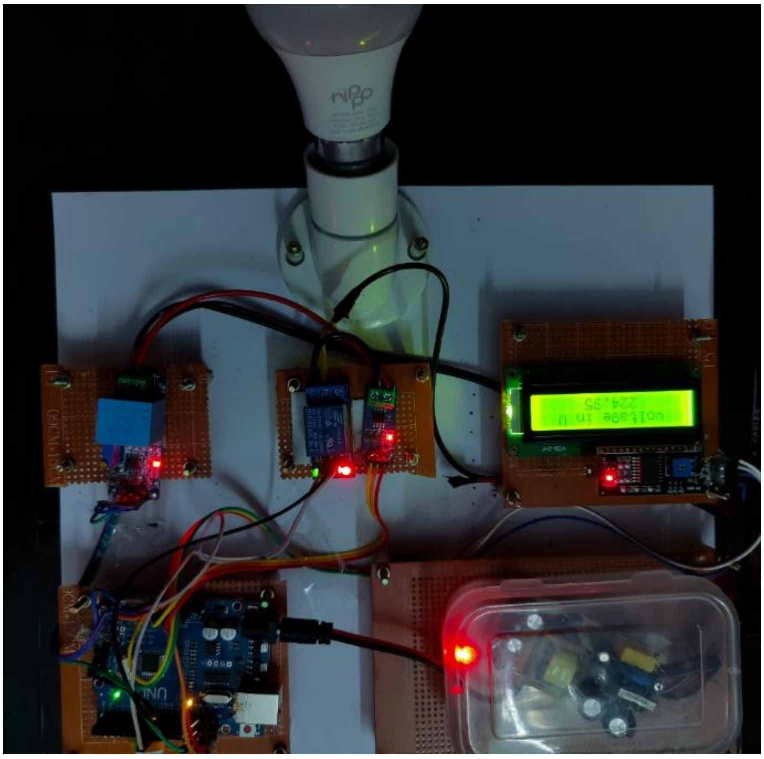

Fig. 5.2 Overvoltage condition (224.95V)

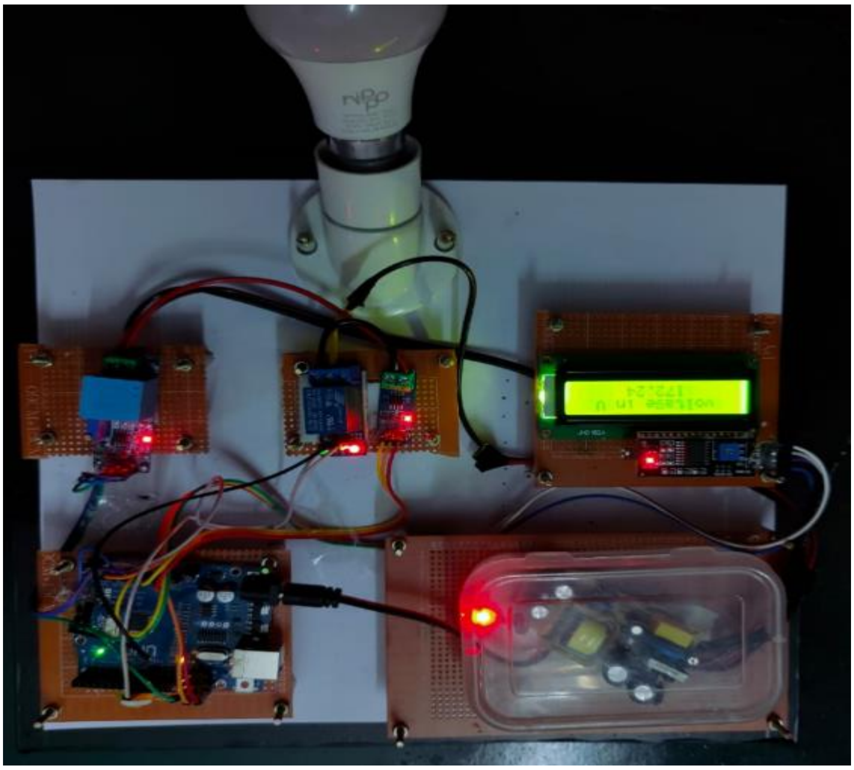

Fig. 5.3 Undervoltage condition (172.24V)

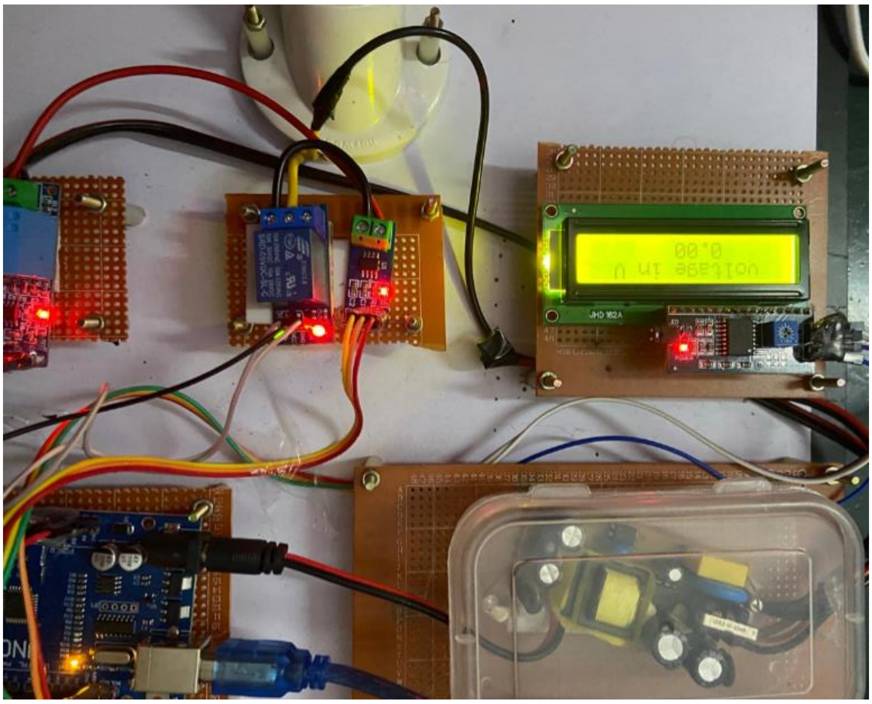

Fig. 5.4 Sensor Failure Condition

RESULT

6.1 Output

Voltage Monitoring and System Trip Logic

The graph represents the real-time voltage monitoring of the system with an emphasis on the predefined voltage thresholds and the system trip logic. Here's what the graph indicates:

- Blue Line (Voltage): This line shows the actual voltage waveform over time.
- Threshold Lines:
    - The red dashed line represents the under-voltage threshold (180V). If the voltage drops below this value, the system trips.
    - The green dashed line represents the over-voltage threshold (220V). If the voltage exceeds this value, the system trips.
- Orange Highlighted Regions: These are the time intervals during which the system detects an out-of-threshold voltage and triggers the relay to trip, disconnecting the load for safety.

Key Observations:

- The voltage oscillates between values above 240V (over-voltage) and below 160V (under-voltage).
- Whenever the voltage exceeds the threshold range of 180–220V, the system activates the trip logic, protecting the connected load.

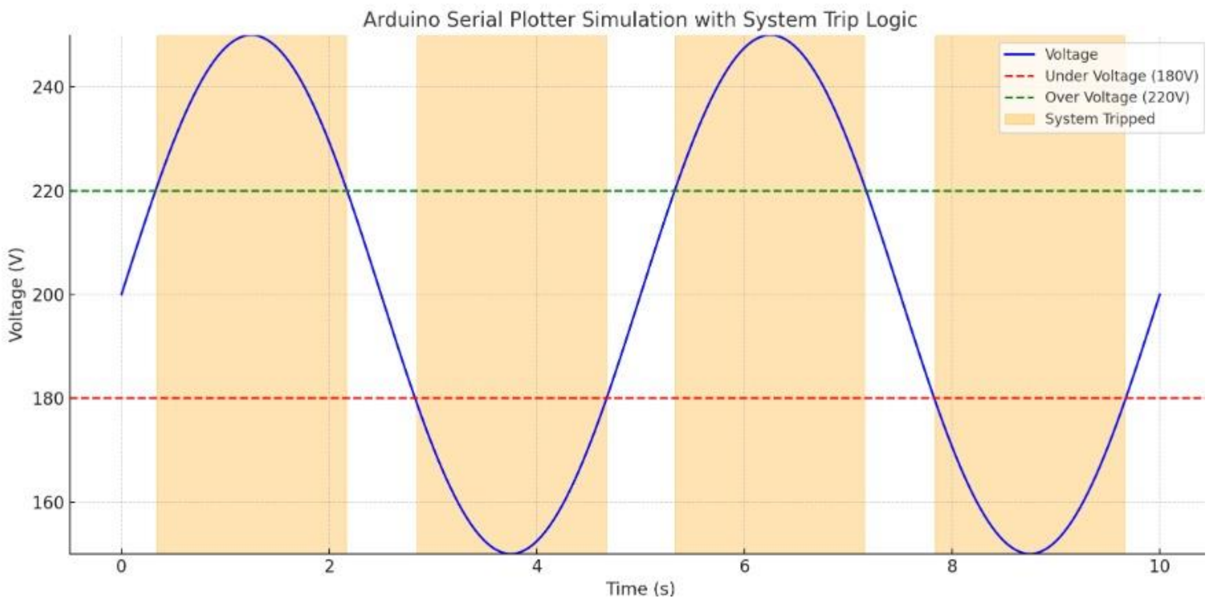

Fig. 6.1 System Trip Logic Graph

Relay Status Over Time

The second graph shows the **relay's operational status** in response to voltage fluctuations. The relay status is binary:

- **0 (OFF)**: The relay is deactivated, and the load is disconnected.
- **1 (ON)**: The relay is activated, and the load is connected.

Key Observations:

- During time intervals when the voltage exceeds the threshold limits (as shown in the first graph), the relay status switches to 0, disconnecting the load.
- The relay consistently toggles between 1 and 0, demonstrating the system's real-time response to fluctuating voltage.

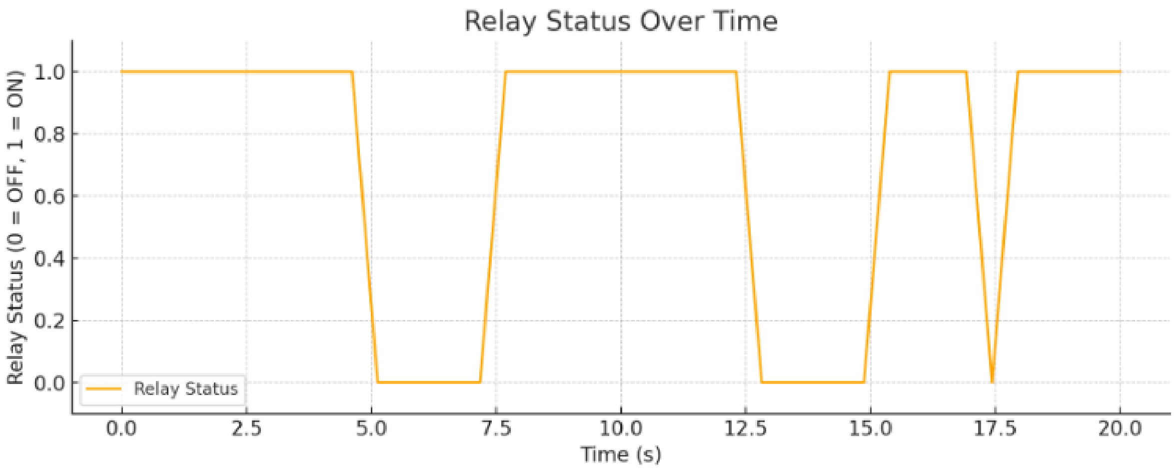

Fig. 6.2 Relay Status Over Time

6.2 Project Results

1. *Successful Voltage Monitoring:*

   The system accurately monitors the real-time voltage and identifies when the voltage deviates from the safe operating range.

2. Automated Load Protection:

   The relay successfully disconnects the load during hazardous voltage conditions, preventing potential damage or electrical hazards.

3. Real-Time Data Visualization:

   The integrated LCD and Arduino serial plotter provide users with clear and real-time insights into the system's operational status, including voltage levels and relay activity.

4. Efficiency and Reliability:

   The system eliminates the need for manual intervention, ensuring efficient and automatic load protection.

6.3 Conclusion

The voltage and current monitoring system developed using Arduino demonstrates a robust and efficient solution for real-time monitoring and automated load protection in electrical systems. By leveraging voltage and current sensors to accurately measure electrical parameters and employing a relay module for load control, the system ensures comprehensive safety by disconnecting the load during abnormal voltage conditions. This proactive approach safeguards connected devices from potential damage caused by overvoltage or undervoltage scenarios, reducing downtime and maintenance costs.

The integration of an LCD with an I2C module significantly enhances the system's usability by providing users with clear, real-time feedback on critical electrical parameters, ensuring a more intuitive and accessible monitoring experience. This project highlights the effectiveness of combining cost-efficient hardware with microcontroller-based automation to address essential energy management and electrical safety requirements.

Furthermore, the system's modular design and simplicity make it scalable for various applications, including residential, industrial, and educational environments. With additional advancements, such as the inclusion of wireless monitoring, IoT integration for remote access, and data analytics for predictive maintenance, this solution can evolve into a comprehensive smart energy management system. This project exemplifies the potential of microcontroller-driven solutions in promoting energy efficiency and enhancing the safety and reliability of electrical systems.

REFERENCE

7.1 Literature Review Papers

Paper1:

https://ieeexplore.ieee.org/document/8897428

Paper2:
https://www.researchgate.net/publication/370403789_Development_of_Overvoltage_and_under_voltage_Protection_System_for_A_Residential_Building_Interfaced_with_a_GSM_Notification_System

Paper3:

https://www.researchgate.net/publication/377276473_Overvoltage_and_Undervoltage_Intelligent_Protection_System

Paper4:

https://www.researchgate.net/publication/370350011_Safety_Design_using_ATS_by_Identifying_Voltage_Interference_based_on_Fuzzy_Logic

Paper5:

https://www.researchgate.net/publication/364535639_Monitoring_and_Protection_System_for_Overvoltage_Undervoltage_and_Unbalance_Voltage

Paper6:

https://www.researchgate.net/publication/382304395_Diagnosis_of_voltage_unbalance_state_in_a_system_with_power_converter?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InNpZ251cCIsInBhZ2UiOiJzZWFyY2giLCJwb3NpdGlvbiI6InBhZ2VIZWFkZXIifX0

Paper7:

https://www.researchgate.net/publication/376677862_Exploring_the_Effects_of_Voltage_Variation_and_Load_on_the_Electrical_and_Thermal_Performance_of_Permanent-Magnet_Synchronous_Motors

Paper8:

https://www.researchgate.net/publication/357800585_Analysis_and_Recommendations_for_LED_Catastrophic_Failure_Due_to_Voltage_Stress?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InNpZ251cCIsInBhZ2UiOiJzZWFyY2giLCJwb3NpdGlvbiI6InBhZ2VIZWFkZXIifX0

Paper9:

https://www.researchgate.net/publication/302735767_High-stability_reset_circuit_for_monitoring_supply_undervoltage_and_overvoltage

Paper10:

https://www.researchgate.net/publication/381824205_Review_and_Analysis_of_Voltage_Clamping_Circuits_with_Low_Overvoltage_Ratios_for_DC_Circuit_Breakers

Paper11:

https://www.researchgate.net/publication/385884987_Development_of_an_Arduino-Based_Protection_System_for_3-Phase_Power_Networks?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InNlYXJjaCIsInBhZ2UiOiJzZWFyY2giLCJwb3NpdGlvbiI6InBhZ2VIZWFkZXIifX0

Paper12:

https://www.researchgate.net/publication/376497372_An_overvoltage_adjustment_strategy_Based_on_Integrated_Voltage_Sensitivity_in_the_Active_Distribution_Network

Paper13:

https://www.researchgate.net/publication/365660475_Research_on_Lightning_Overvoltage_Characteristics_of_High-Voltage_Diode_Rectifier

Paper14:

https://www.researchgate.net/publication/334097866_An_Overview_on_Overvoltage_Phenomena_in_Power_Systems

Paper15:

https://www.researchgate.net/publication/377880780_Effect_of_shunt_reactor_rating_on_the_switching_transients_overvoltage_in_high_voltage_system

7.2 Other Online Material

Link 1:

https://www.elprocus.com/under-and-overvoltage-protection-circuit/

Link 2:

https://www.everexceed.com/blog/principle-of-overvoltage-and-undervoltage_b519

Link 3:

https://www.sunpower-uk.com/glossary/what-is-over-voltage-protection/

Link 4:

https://www.macromatic.com/support/knowledge-base-articles/what-is-undervoltage-how-can-i-protect-my-equipment/?srsltid=AfmBOorI8WCCte8sXeAnYixfsBKcJ6OgO1qeysUzNic9CMNUqGxaeBD3

Link 5:

https://www.mes.com.sg/2019/09/19/overvoltage-undervoltage-all-you-need-to-know/

7.3 Books

Book 1:

Standard Handbook for Electrical Engineers, 17th Edition

Surya Santoso, Ph.D. H. Wayne Beaty

Publication Date & Copyright: 2018 McGraw-Hill Education

Book 2:

Protection of Electronic Circuits from Overvoltage

Ronald B. Standler

Publication Date & Copyright: 2012 Dover Publication