Smart Home Automation

A PROJECT REPORT

Submitted in partial fulfilment of requirements for the
Award of the degree

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

By

Ms.B.Siva Latha	**(20471A04I4)**
Ms.Aparna	**(20471A04I6)**
Mr.D.Madhu	**(20471A04M4)**
Ms.G.Naga Lakshmi	**(20471A04J1)**
Mr.P.Navya Sai	**(20471A04L7)**

Under the guidance of
Mr.P.S.S Chakravarthy
Professor

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NARASARAOPETA ENGINEERING COLLEGE (Autonomous)
(Approved by AICTE, New Delhi, Accredited NBA Tier-1, Accredited by NAAC with 'A+' Grade,
Permanently Affiliated to J.N.T.U.K, Kakinada
Kotappakonda Road, Yallamanda,
Narasaraopet - 522601, Palnadu (Dist.), Andhra Pradesh.

March, 2024

NARASARAOPETA ENGINEERING COLLEGE (Autonomous)
(Approved by AICTE, New Delhi, Accredited NBA Tier-1, Accredited by NAAC with ‘A+’ Grade,
Permanently Affiliated to J.N.T.U.K, Kakinada
Kotappakonda Road, Yallamanda,
Narasaraopet - 522601, Palnadu (Dist.), Andhra Pradesh.
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project entitled Smart Home Automation is the Bonafide work carried out by Ms.B.Siva Latha(20471A04I4), Ms.C.Aparna(20471A04I6), Mr.D.Madhu (20471A04M4), Ms.G.Naga Lakshmi (20471A04J1), Mr.P.Navya Sai (20471A04L7) in partial fulfilment of the requirements for the award of the degree Bachelor of Technology in Electronics and Communication Engineering from Jawaharlal Nehru Technological University Kakinada, Kakinada during the year 2022-23 under my supervision and guidance.

Guide	Head of the Department
(Mr.P.S.S.Chakravarthy)	(Dr. V. Venkata Rao)

External Examiner

ACKNOWLEDGEMENTS

It is our pleasure to express our deep and sincere gratitude to our Project Guide **Mr.P.S.S Chakravarthy**, Assistant Professor/ Associate Professor, for extending her sincere and heartfelt guidance throughout this project work.

It is our desire to express sincere thanks to our Project coordinator **Dr.Shaik .Bajidvali**, Associate Professor, for extending his cooperation and guidance throughout this project work.

We greatly indebted to **Dr. V. Venkata Rao**, Professor & Head of the Department, for providing motivation and valuable suggestions at every stage of our course of study.

We are thankful to our principal **Dr. M. Sreenivasa Kumar** for providing us sparkling environment during our course of study.

We also express sincere thanks to **Sri M.V. Koteswara Rao**, Chairman of Narasaraopeta Engineering College, Narasaraopet for providing excellent infrastructural facilities to complete our course.

We extend our sincere thanks to all other **Teaching** and **Non-Teaching Faculty members** of the department for their cooperation and encouragement throughout our course of study.

We affectionately acknowledge our **friends** for their motivation and suggestions which helps us in successfully completing our course.

We have no words to acknowledge the warm affection, constant inspiration and encouragement that we received from our **Parents and Family Members.**

Ms.B.Siva Latha	(20471A04I4)
Ms.C.Aparna	(20471A04I6)
Mr.D.Madhu	(20471A04M4)
Ms.G.Naga lakshmi	(20471A04J1)
Mr.P.Navya Sai	(20471A04L7)

CONTENTS

ABSTRACT

An IoT involves extending Internet connectivity beyond standard devices, such as desktops, laptops, smart phones which is Embedded with technology, these devices make communication over the Internet, and user can be remotely monitored and controlled. This proposed system focused on five different techniques and systems such as Low cost Wi-Fi based home automation system, it facilitates global connectivity over world-wide physical objects to serve people in a collaborative manner automatically and intelligently . Home automation using symmetric encryption scheme uses the chaos-based cryptography and message authentication code for data transmission to ensure the security . Home automation using set of sensors, which focuses on building a smart wireless home security system which sends alerts using internet . Ethernet based system works on real time monitoring and voice control, so that the devices and switches can be remotely controlled with or without android based app . Home automation system via WWW, which provides an interface to home appliances using telephone line or internet to supply management and observance through smart phone . This paper presents, Home automation by IoT in which a method is defined that uses Arduino controller to control and manage different electrical appliances of the home. This system performs its task with the help of the sensors which help in controlling the state of the devices.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

(Alphabetical order)

[.

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

An Embedded System is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function. A good example is the microwave oven. Almost every household has one, and tens of millions of them are used everyday, but very few people realize that a processor and software are involved in the preparation of their lunch or dinner.

This is in direct contrast to the personal computer in the family room. It too is comprised of computer hardware and software and mechanical components (disk drives, for example). However, a personal computer is not designed to perform a specific function rather; it is able to do many different things. Many people use the term general-purpose computer to make this distinction clear. As shipped, a general-purpose computer is a blank slate; the manufacturer does not know what the customer will do wish it. One customer may use it for a network file server another may use it exclusively for playing games, and a third may use it to write the next great American novel.

Frequently, an embedded system is a component within some larger system. For example, modern cars and trucks contain many embedded systems. One embedded system controls the anti-lock brakes, other monitors and controls the vehicle's emissions, and a third displays information on the dashboard. In some cases, these embedded systems are connected by some sort of a communication network, but that is certainly not a requirement.

At the possible risk of confusing you, it is important to point out that a general-purpose computer is itself made up of numerous embedded systems. For example, my computer consists of a keyboard, mouse, video card, modem, hard drive, floppy drive, and sound card-each of which is an embedded system. Each of these devices contains a processor and software and is designed to perform a specific function. For example, the modem is designed to send and receive digital data over analog telephone line. That's it and all of the other devices can be summarized in a single sentence as well.

If an embedded system is designed well, the existence of the processor and software could be completely unnoticed by the user of the device. Such is the case for a microwave oven, VCR, or alarm clock. In some cases, it would even be possible to build an equivalent device that does not contain the processor and software. This could be done by replacing the combination with a custom integrated circuit that performs the same functions in hardware. However, a lot of flexibility is lost when a design is hard-cooled in this way. It is mush easier, and cheaper, to change a few lines of software than to redesign a piece of custom hardware.

1.2 OBJECTIVES OF THE PROJECT

Given the definition of embedded systems earlier is this chapter; the first such systems could not possibly have appeared before 1971. That was the year Intel introduced the world's first microprocessor. This chip, the 4004, was designed for use in a line of business calculators produced by the Japanese Company Busicom. In 1969, Busicom asked Intel to design a set of custom integrated circuits-one for each of their new calculator models. The 4004 was Intel's response rather than design custom hardware for each calculator, Intel proposed a general-purpose circuit that could be used throughout the entire line of calculators. Intel's idea was that the software would give each calculator its unique set of features.

The microcontroller was an overnight success, and its use increased steadily over the next decade. Early embedded applications included unmanned space probes, computerized traffic lights, and aircraft flight control systems. In the 1980s, embedded systems quietly rode the waves of the microcomputer age and brought microprocessors into every part of our kitchens (bread machines, food processors, and microwave ovens), living rooms (televisions, stereos, and remote controls), and workplaces (fax machines, pagers, laser printers, cash registers, and credit card readers).

It seems inevitable hat the number of embedded systems will continue to increase rapidly. Already there are promising new embedded devices that have enormous market potential; light switches and thermostats that can be central computer, intelligent air-bag systems that don't inflate when children or small adults are present, pal-sized electronic organizers and personal digital assistants (PDAs), digital cameras, and dashboard navigation systems. Clearly, individuals who possess the skills and desire to design the next generation of embedded systems will be in demand for quite some time**.**

Real Time Systems:

One subclass of embedded is worthy of an introduction at this point. As commonly defined, a real-time system is a computer system that has timing constraints. In other words, a real-time system is partly specified in terms of its ability to make certain calculations or decisions in a timely manner. These important calculations are said to have deadlines for completion. And, for all practical purposes, a missed deadline is just as bad as a wrong answer.

The issue of what if a deadline is missed is a crucial one. For example, if the real-time system is part of an airplane's flight control system, it is possible for the lives of the passengers and crew to be endangered by a single missed deadline. However, if instead the system is involved in satellite communication, the damage could be limited to a single corrupt data packet. The more severe the consequences, the more likely it will be said that the deadline is "hard" and thus, the system is a hard real-time system. Real-time systems at the other end of this discussion are said to have "soft" deadlines.

All of the topics and examples presented in this book are applicable to the designers of real-time system who is more delight in his work. He must guarantee reliable operation of the software and hardware under all the possible conditions and to the degree that human lives depend upon three system's proper execution, engineering calculations and descriptive paperwork.

Application Areas

Nearly 99 per cent of the processors manufactured end up in embedded systems. The embedded system market is one of the highest growth areas as these systems are used in very market segment- consumer electronics, office automation, industrial automation, biomedical engineering, wireless communication, data communication, telecommunications, transportation, military and so on.

Consumer appliances:

At home we use a number of embedded systems which include digital camera, digital diary, DVD player, electronic toys, microwave oven, remote controls for TV and air-conditioner, VCO player, video game consoles, video recorders etc. Today's high-tech car has about 20 embedded systems for transmission control, engine spark control, air-conditioning, navigation etc. Even wristwatches are now

becoming embedded systems. The palmtops are powerful embedded systems using which we can carry out many general-purpose tasks such as playing games and word processing.

Office automation:

The office automation products using em embedded systems are copying machine, fax machine, key telephone, modem, printer, scanner etc.

Industrial automation:

Today a lot of industries use embedded systems for process control. These include pharmaceutical, cement, sugar, oil exploration, nuclear energy, electricity generation and transmission. The embedded systems for industrial use are designed to carry out specific tasks such as monitoring the temperature, pressure, humidity, voltage, current etc., and then take appropriate action based on the monitored levels to control other devices or to send information to a centralized monitoring station. In hazardous industrial environment, where human presence has to be avoided, robots are used, which are programmed to do specific jobs. The robots are now becoming very powerful and carry out many interesting and complicated tasks such as hardware assembly.

Medical electronics:

Almost every medical equipment in the hospital is an embedded system. These equipments include diagnostic aids such as ECG, EEG, blood pressure measuring devices, X-ray scanners; equipment used in blood analysis, radiation, colonoscopy, endoscopy etcDevelopments in medical electronics have paved way for more accurate diagnosis of diseases.

Computer networking:

Computer networking products such as bridges, routers, Integrated Services Digital Networks (ISDN), Asynchronous Transfer Mode (ATM), X.25 and frame relay switches are embedded systems which implement the necessary data communication protocols. For example, a router interconnects two networks. The two networks may be running different protocol stacks. The router's function is to obtain the data packets from incoming pores, analyze the packets and send them towards the destination after doing necessary protocol conversion. Most networking equipments, other than the end systems (desktop computers) we use to access the networks, are embedded systems

. **Telecommunications**:

In the field of telecommunications, the embedded systems can be categorized as subscriber terminals and network equipment. The subscriber terminals such as key telephones, ISDN phones, terminal adapters, web cameras are embedded systems. The network equipment includes multiplexers, multiple access systems, Packet Assemblers Dissemblers (PADs), sate11ite modems etc. IP phone, IP gateway, IP gatekeeper etc. are the latest embedded systems that provide very low-cost voice communication over the Internet.

Wireless technologies:

Advances in mobile communications are paving way for many interesting applications using embedded systems. The mobile phone is one of the marvels of the last decade of the 20'h century. It is a very powerful embedded system that provides voice communication while we are on the move. The Personal Digital Assistants and the palmtops can now be used to access multimedia services over the Internet. Mobile communication infrastructure such as base station controllers, mobile switching centers are also powerful embedded systems.

Insemination:

Testing and measurement are the fundamental requirements in all scientific and engineering activities. The measuring equipment we use in laboratories to measure parameters such as weight, temperature, pressure, humidity, voltage, current etc. are all embedded systems. Test equipment such as oscilloscope, spectrum analyzer, logic analyzer, protocol analyzer, radio communication test set etc. are embedded systems built around powerful processors. Thank to miniaturization, the test and measuring equipment are now becoming portable facilitating easy testing and measurement in the field by field-personnel.

Security:

Security of persons and information has always been a major issue. We need to protect our homes and offices; and also the information we transmit and store. Developing embedded systems for security applications is one of the most lucrative businesses nowadays. Security devices at homes, offices, airports etc. for authentication and verification are embedded systems. Encryption devices are nearly 99 per cent of the processors that are manufactured end up in~ embedded systems. Embedded systems find applications in . every industrial segment- consumer electronics, transportation, avionics, biomedical engineering, manufacturing, process control and industrial automation, data communication, telecommunication, defense, security etc. Used to encrypt the data/voice being transmitted on communication links such as telephone lines. Biometric systems using fingerprint and face recognition are now being extensively used for user authentication in banking applications as well as for access control in high security buildings.

Finance:

Financial dealing through cash and cheques are now slowly paving way for transactions using smart cards and ATM (Automatic Teller Machine, also expanded as Any Time Money) machines. Smart card, of the size of a credit card, has a small micro-controller and memory; and it interacts with the smart card reader! ATM machine and acts as an electronic wallet. Smart card technology has the capability of ushering in a cashless society. Well, the list goes on. It is no exaggeration to say that eyes wherever you go, you can see, or at least feel, the work of an embedded system!

1. 3 LITERATURE SURVEY

This project discusses the integration of IPv6 and DomoNet in Smart Home systems, highlighting the need for self-organization capabilities to enhance performance. It identifies challenges such as managing increasing applications, security issues, connectivity problems, and real-time functioning. Key areas of concern include standards, identification, privacy, authentication, security, integration, coordination, data storage, and network self-organization. The project suggests future research directions focusing on security enhancement, data extraction improvement, and network self-organization in the IoT environment.

The project explores the implementation of IoT-based control mechanisms for lighting in older buildings to reduce power consumption. A test-bed was utilized to measure light variations in an office room, and a control algorithm was created to adjust lighting based on outdoor light changes. Simulations indicated a notable decrease in power usage with the IoT system compared to traditional systems, highlighting the energy-saving potential in older homes with IOT-enabled control systems.

This paper discusses the security aspects, penetration tests, and collaborative modeling of a smart home system using mathematical models and simulations. It highlights the use of Arduino and Raspberry Pi for sensor networking, data transmission, and IOT functionalities, emphasizing the importance of security in IOT systems.

The project describes the development of two remote monitoring systems for household appliances using cell phones. The systems can detect intrusion, send alerts, trigger alarms, control appliances remotely, and have security features to prevent unauthorized access. One system uses miscalls for communication, while the other utilizes SMS and MMS with Java programming. The systems were developed with input from students and faculty at Babasaheb Naik College of Engineering in India.

1.4 ORGANIZATION OF THE REPORT

Chapter 1 describes an introduction about the project. In chapter 2, the existing methods of the project are presented. In chapter 3, the proposed method of the project is presented. In chapter 4, the results of the project and Advantages, Limitations and Applications are presented. In chapter 5, the conclusion and future scope of the project are presented.

CHAPTER 2

EXISTING METHODS

CHAPTER 2

EXISTING METHODS

2.1 Circuit Connection of Switch Board

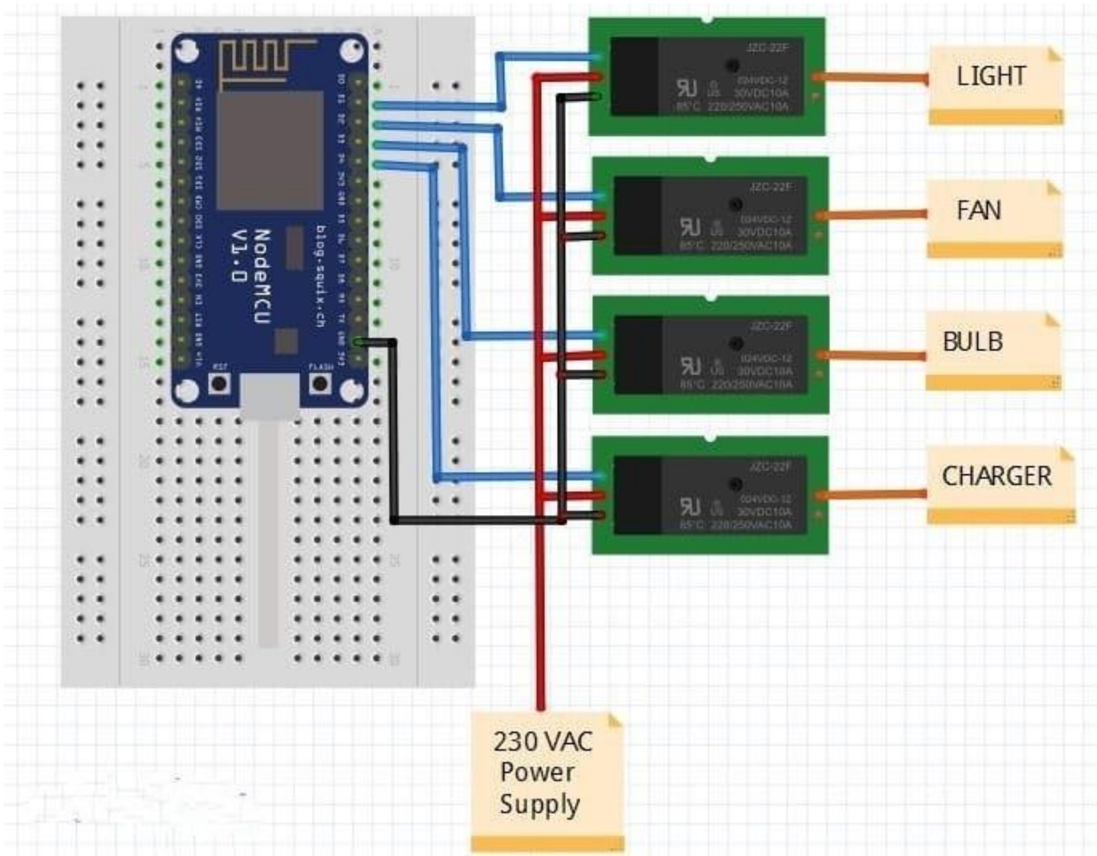

Fig.2.1 Circuit Connection of Switch Board

Working

A switchboard circuit diagram illustrates the electrical connections between various components within a switchboard. Switchboards are used in electrical power distribution systems to control and distribute electricity to various circuits and loads. Here's an overview of the theory behind a typical switchboard circuit diagram:

1. **Power Source**: The power source, typically the main electrical supply, is usually depicted at the top of the diagram. This could be a generator, transformer, or utility power supply.

2. **Main Switches and Circuit Breakers**: These are the primary switches that control the flow of electricity into the switchboard. They act as the main disconnect switches and protect the circuit against overloads and short circuits. Circuit breakers are designed to trip and interrupt the circuit in case of a fault.

3. **Busbars**: Busbars are conductive bars or strips within the switchboard that serve as electrical pathways for distributing power. They connect the main switches and circuit breakers to the various branch circuits within the switchboard.

4. **Branch Circuits**: Branch circuits are the individual circuits that supply power to specific loads or areas within a facility. Each branch circuit is typically protected by its own circuit breaker.

5. **Load Devices**: Load devices represent the electrical loads connected to the switchboard, such as lights, appliances, motors, and other equipment.

6. **Control Devices**: Control devices such as switches, relays, and contactors are used to control the operation of various loads within the system. They may be depicted in the circuit diagram to illustrate how they are connected to the power source and loads.

7. **Monitoring and Protection Equipment**: This includes devices such as meters, gauges, and protective relays that monitor the electrical parameters of the system and provide protection against faults and abnormal conditions.

When interpreting a switchboard circuit diagram, it's essential to understand the symbols used to represent various components. These symbols are standardized and widely recognized within the electrical engineering field. Understanding the connections and functions of each component in the diagram is crucial for designing, troubleshooting, and maintaining electrical power distribution systems. Additionally, safety considerations are paramount when working with electrical systems, so proper training and adherence to safety protocols are essential for anyone involved in the installation, maintenance, or repair of switch board circuits.

2.2 Switch Board

Fig.2.2 Switch Board

Normal switches are fundamental components in electrical circuits, designed to control the flow of electricity by making or breaking the circuit connection. These switches come in various forms, including toggle switches, rocker switches, push-button switches, and rotary switches, each serving specific purposes and applications. The theory behind normal switches revolves around their basic operation, construction, and the principles of electrical conductivity.

At its core, a normal switch consists of a mechanism that can physically open or close an electrical circuit. When the switch is in the "on" or closed position, it provides a conductive path for electricity to flow through, allowing the circuit to be energized and power to reach connected devices or loads. Conversely, when the switch is in the "off" or open position, it interrupts the flow of electricity, effectively cutting off power to the circuit and isolating the connected devices from the power source.

The mechanism within a normal switch typically involves movable contacts that can be manipulated to establish or break the electrical connection. For instance, in a toggle switch, flipping the lever or toggle causes internal contacts to move, either closing or opening the circuit.

Similarly, in a push-button switch, pressing the button actuates the mechanism to make or break the circuit connection. The design of the switch mechanism determines its specific mode of operation and how it responds to user input.

Normal switches are crucial for controlling lighting, appliances, machinery, and various electrical devices in homes, businesses, and industrial settings. They offer a simple yet effective means of managing electrical power, providing convenience and safety by allowing users to turn devices on or off as needed. Moreover, switches play a vital role in circuit protection, enabling quick de-energization of circuits in emergency situations or during maintenance and repairs.

Understanding the theory behind normal switches involves grasping concepts related to electrical conductivity, circuit continuity, and the principles of electromechanical operation. Engineers and technicians must consider factors such as switch ratings, contact materials, and environmental conditions when selecting switches for specific applications to ensure reliable performance and compliance with safety standards. Additionally, proper installation, maintenance, and inspection of switches are essential to prevent malfunctions, electrical hazards, and potential damage to equipment or property. Overall, normal switches are indispensable components in electrical systems, providing versatile control capabilities and contributing to the efficient and safe operation of various electrical devices and installations.

Drawbacks

- While normal switchboards serve as fundamental components in electrical systems, they do have some drawbacks that limit their functionality and efficiency in certain contexts. One significant drawback is the lack of advanced features and capabilities compared to more modern switchboard technologies. Normal switchboards typically offer basic on/off control without additional functionalities such as dimming, remote operation, or programmable scheduling. This limitation can restrict the flexibility and convenience of controlling electrical devices, especially in environments where advanced automation and customization are desired.
- Another drawback of normal switchboards is their manual operation, which relies on physical interaction with the switches to control electrical circuits. This manual operation can be inconvenient, particularly in large-scale installations where numerous switches need to be managed individually. It can also lead to user error or oversight, as there is no built-in intelligence or feedback mechanism to indicate the status of connected devices or circuits.

- Additionally, normal switchboards may lack integration with modern smart home or building automation systems, making it challenging to incorporate them into broader interconnected networks for centralized control and monitoring. As a result, achieving seamless integration with other smart devices and technologies may require additional retrofitting or upgrading, which can be costly and complex.

- Furthermore, normal switchboards typically do not offer comprehensive energy monitoring or optimization features. Without the ability to track energy usage or implement energy-saving strategies automatically, users may miss opportunities to reduce electricity consumption and improve efficiency.

- Lastly, normal switchboards may pose safety risks if not properly maintained or if they become damaged over time. Wear and tear on switch mechanisms, loose connections, or outdated components can increase the likelihood of electrical faults, arcs, or fires. Moreover, manual switching operations in hazardous environments or under adverse conditions may present safety hazards to users.

CHAPTER 3

PROPOSED METHOD

CHAPTER 3

PROPOSED METHOD

3.1 BLOCK DIAGRAM SMART HOME

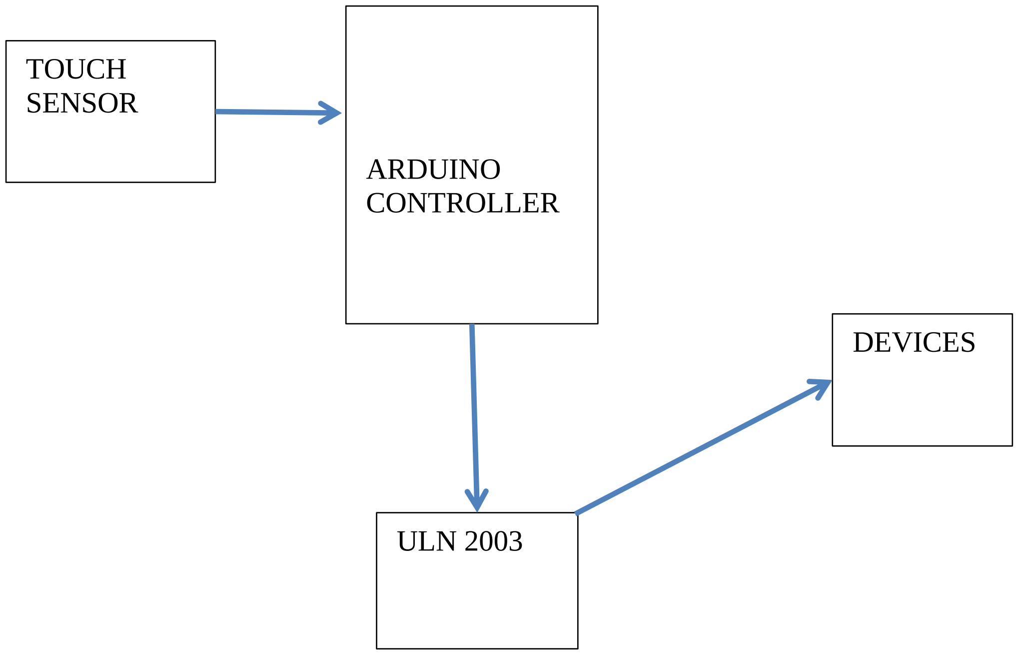

Fig 3.1 Block Diagram of Smart Home

Smart home automation circuits integrate advanced technologies such as microcontrollers, sensors, actuators, and communication protocols to enable intelligent control and automation of various household devices and systems. At the core of smart home automation circuit theory lies the principle of interconnectedness and intelligence, allowing users to remotely monitor, manage, and optimize their home environment for convenience, comfort, energy efficiency, and security.

These circuits typically consist of a central controller, often a microcontroller or a specialized home automation hub, which serves as the brain of the system. The controller communicates with sensors deployed throughout the home, such as motion sensors, temperature sensors, light sensors, and door/window sensors, to gather data about the environment and occupants' activities. This data is then processed to make informed decisions about controlling connected devices.

Actuators, such as smart switches, dimmers, motorized blinds, and smart plugs, are used to execute commands issued by the controller, enabling remote or automated control over lighting, HVAC systems, window treatments, appliances, and other electrical devices. Communication protocols like Wi-Fi, Zigbee, Z-Wave, or Bluetooth facilitate data exchange between the controller and connected devices, allowing for seamless integration and interoperability within the smart home ecosystem.

Machine learning algorithms and artificial intelligence techniques may also be employed to analyze data trends, predict user preferences, and optimize system behavior over time. This enables smart home automation circuits to adapt to changing conditions, learn from user interactions, and anticipate their needs, enhancing the overall user experience.

Moreover, smart home automation circuits often feature connectivity with mobile devices and cloud-based platforms, enabling remote access and control via smartphone apps or web interfaces. This remote access empowers users to monitor and manage their homes from anywhere, offering convenience, flexibility, and peace of mind.

Security is another crucial aspect of smart home automation circuit theory. Encryption protocols, authentication mechanisms, and secure communication channels are implemented to safeguard sensitive data and prevent unauthorized access or tampering with the system.

3.2 Touch sensor

Fig: 3.2 Touch Sensor

Infrared (IR) sensors operate on the principle of detecting infrared radiation emitted or reflected by objects in their vicinity. These sensors typically consist of an IR emitter and an IR detector, often placed in close proximity to each other. The theory behind IR sensors revolves around their ability to detect changes in the intensity or frequency of infrared radiation, which can indicate the presence or absence of objects, motion, or temperature variations.

When an IR emitter emits infrared radiation, it interacts with objects in its field of view. These objects absorb, reflect, or transmit varying amounts of infrared energy depending on their material composition, surface characteristics, and temperature. The IR detector, positioned to receive the reflected or emitted infrared radiation, captures these changes in energy levels.

In passive infrared (PIR) sensors commonly used for motion detection, changes in infrared radiation levels are detected as objects move within the sensor's detection range. This triggers a response, such as activating lights, alarms, or other devices, based on programmed logic. PIR sensors are sensitive to changes in heat patterns caused by movement, making them effective for detecting human or animal motion in security systems, occupancy sensing applications, and automatic lighting systems.

Infrared temperature sensors, on the other hand, measure the infrared radiation emitted by objects to determine their surface temperature. These sensors are based on the principle of blackbody radiation, where the intensity and wavelength distribution of emitted infrared radiation are directly related to the object's temperature. By detecting and analyzing the infrared spectrum, these sensors can provide accurate temperature measurements without physical contact, making them suitable for non-contact temperature monitoring in industrial processes, medical applications, and environmental monitoring.

Infrared proximity sensors utilize the reflection or absorption of infrared radiation to detect the presence of objects within a certain distance from the sensor. By emitting infrared light and measuring the intensity of the reflected signal, these sensors can determine the distance to the target object. This proximity sensing capability is used in various applications such as robotics, automotive systems, and consumer electronics for touchless interfaces, object detection, and obstacle avoidance.

3.3 ULN 2003

Fig.3.3 ULN 2003

The ULN2003 is an integrated circuit (IC) commonly used as a high-voltage, high-current Darling ton transistor array. The theory behind the ULN2003 revolves around its ability to provide robust and efficient control of inductive loads such as relays, solenoids, motors, and lamps in various electronic applications. The IC consists of seven Darlington pairs, each composed of two bipolar transistors configured in a cascaded arrangement, which offers high current gain and low saturation voltage characteristics.

The ULN2003 is designed to interface with microcontrollers, logic circuits, or other digital control signals, providing a convenient means of driving loads that require higher current or voltage than what the control signal can directly handle. Each Darlington pair within the ULN2003 can switch loads up to 500 mA and withstand voltages of up to 50V, making it suitable for driving a wide range of loads in industrial, automotive, and consumer electronics applications.

The theory of operation behind the ULN2003 involves the input control signals activating the Darling ton pairs, causing them to conduct and provide a path for current flow from the high-voltage supply to the load. The IC incorporates protection diodes to suppress voltage spikes generated by inductive loads during switching, ensuring reliable and safe operation. Additionally, the ULN2003 features integral fly back diodes that provide a path for the inductive load's back EMF (electromotive force) when the load is turned off, preventing damage to the transistors and the IC.

The ULN2003's versatility, robustness, and ease of use make it a popular choice for driving various loads in electronic circuits, particularly in applications where space, cost, and complexity are considerations. Its compact package, low power dissipation, and high noise immunity further enhance its suitability for demanding environments. Whether used in industrial automation, automotive systems, robotics, or hobbyist projects, the ULN2003 provides reliable and efficient control of inductive loads, demonstrating the effectiveness of Darlington transistor arrays in simplifying circuit design and improving performance in diverse electronic applications.

3.4 ARDUINO CONTROLLER

Fig.3.4 Arduino Controller

The theory behind an Arduino controller revolves around its role as a versatile and user-friendly microcontroller platform designed for a wide range of electronic projects and applications. At its core, an Arduino controller is built around an Atmel AVR or ARM microcontroller, featuring various input/output (I/O) pins, analog-to-digital converters (ADCs), digital-to-analog converters

(DACs), communication interfaces, and a bootloader for easy programming via a USB connection.

Arduino controllers provide a simplified and accessible way for hobbyists, students, makers, and professionals to prototype, develop, and deploy embedded systems without requiring in-depth knowledge of electronics or programming. The Arduino development environment, based on the Wiring programming language and the Arduino Integrated Development Environment (IDE), offers a straightforward platform for writing, compiling, and uploading code to the controller.

The theory of operation behind an Arduino controller involves writing code (sketches) that define the behavior and functionality of the microcontroller. Users can interact with sensors, actuators, displays, and other electronic components by programming the Arduino to read inputs, process data, and control outputs based on predefined logic and algorithms. The controller's I/O pins can be configured as digital inputs or outputs, analog inputs, or specialized communication interfaces such as I2C, SPI, UART, and USB, providing flexibility for interfacing with a wide range of peripherals and devices.

Arduino controllers support a vast ecosystem of shields, modules, and libraries that extend their capabilities and simplify integration with external hardware components and sensors. This modularity and expandability make Arduino controllers suitable for a diverse array of projects, including robotics, home automation, wearable devices, Internet of Things (IoT) applications, interactive art installations, and educational tools.

Moreover, Arduino controllers foster a vibrant community of enthusiasts and developers who share knowledge, collaborate on projects, and contribute to the open-source ecosystem by creating and sharing code, libraries, tutorials, and documentation. This collaborative environment encourages innovation and creativity while empowering individuals to explore and experiment with electronics, programming, and digital technology.

3.5 DEVICES

Fig.3.5 Devices

Smart home automation devices encompass a diverse range of interconnected electronic devices and systems designed to enhance convenience, efficiency, comfort, and security within residential environments. The theory behind smart home automation devices revolves around integrating advanced technologies such as sensors, actuators, microcontrollers, communication protocols, and software applications to enable intelligent control and automation of various home appliances, utilities, and systems.

These devices often include smart thermostats, lighting controls, smart plugs, smart locks, door and window sensors, motion detectors, security cameras, smoke detectors, and voice-controlled assistants. Each device serves specific functions and interacts with other devices within the smart home ecosystem to create seamless and personalized experiences for users.

The theory of operation behind smart home automation devices involves collecting data from sensors deployed throughout the home, such as motion sensors, temperature sensors, and door/window sensors, to monitor the home environment and occupants' activities. This data is then processed by centralized controllers or distributed computing systems to make informed decisions and trigger actions based on predefined rules, schedules, or user preferences.

Actuators, such as motorized valves, relays, servo motors, and smart switches, receive commands from the controller and execute actions such as adjusting thermostat settings, turning lights on or off, unlocking doors, or activating alarms. Communication protocols like Wi-Fi, Zigbee, Z-Wave, Bluetooth, or Thread facilitate data exchange between devices and controllers, enabling seamless integration and interoperability within the smart home ecosystem.

Smart home automation devices often feature connectivity with mobile devices and cloud-based platforms, allowing users to remotely monitor, control, and manage their homes from anywhere using smartphone apps or web interfaces. This remote access provides users with convenience, flexibility, and peace of mind, allowing them to stay connected and in control of their homes even when away.

Furthermore, advancements in artificial intelligence, machine learning, and natural language processing enable smart home automation devices to learn from user behavior, adapt to changing preferences, and anticipate their needs over time. This adaptive intelligence enhances the overall user experience and efficiency of smart home systems.

3.6 TOUCH PANEL

Fig.3.6 Touch Panel

The theory behind a smart home automation touch panel revolves around creating an intuitive and user-friendly interface for controlling various aspects of a home's automation system. Touch panels serve as interactive control hubs that enable users to monitor, manage, and customize connected devices and systems with simple touch-based gestures.

At its core, a smart home automation touch panel integrates advanced technologies such as touchscreen displays, microcontrollers, communication protocols, and user interface (UI) software to provide seamless interaction with the home automation ecosystem. The touchscreen display acts as a visual interface, presenting information, menus, and controls in an easily navigable format. Users can interact with the touch panel by tapping, swiping, or dragging their fingers across the screen to access different features and functions.

The touch panel serves as a centralized control point for a wide range of smart home devices, including lighting, HVAC systems, security cameras, door locks, motorized shades, entertainment systems, and more. Through the touch panel interface, users can adjust settings, create schedules, set preferences, and automate routines to customize their living environment according to their needs and preferences.

The theory of operation behind a smart home automation touch panel involves integrating seamless communication with connected devices and systems using wireless or wired protocols such as Wi-Fi, Zigbee, Z-Wave, Bluetooth, or Ethernet. This enables real-time monitoring and control of devices from the touch panel, regardless of their location within the home or the type of technology they use.

Moreover, smart home automation touch panels often incorporate advanced features such as voice control integration, gesture recognition, multi-room synchronization, and compatibility with virtual assistants like Amazon Alexa or Google Assistant. These features enhance the usability and versatility of the touch panel, allowing users to interact with their home automation system in ways that are natural, convenient, and efficient.

Additionally, the touch panel interface can be customized to suit individual preferences and aesthetics, with options for personalized layouts, themes, and user profiles. This flexibility enables users to tailor the touch panel experience to their unique requirements, whether they prioritize simplicity, functionality, or aesthetics.

CHAPTER 4

RESULTS AND DISCUSSION

CHAPTER 4

RESULTS AND DISCUSSION

4.1 RESULTS AND DISCUSSION

4.4.1 LIGHTS ON STATE:

Fig 4.1 Lights on state

The concept of the "lights on" state refers to the condition in which a lighting system is actively providing illumination within a space. The theory behind the lights on state involves understanding the various factors that influence when and why lights are turned on, as well as the implications of maintaining this state.

The decision to turn lights on is typically influenced by environmental factors such as ambient light levels, time of day, occupancy, and user preferences. In spaces where natural light is insufficient or unavailable, artificial lighting sources are activated to provide adequate illumination for visibility, safety, and task performance. Similarly, in indoor environments during nighttime or low-light conditions, lights are commonly switched on to ensure visibility and enhance comfort.

The lights on state is crucial for creating functional and inviting spaces, supporting activities such as reading, working, cooking, and socializing. It contributes to a sense of safety and security by reducing the risk of accidents, deterring intruders, and providing reassurance to occupants.

In addition to the practical considerations, the lights on state has implications for energy consumption and sustainability. Keeping lights on unnecessarily or for extended periods can lead to wasted energy and increased utility costs. Therefore, strategies such as using energy-efficient lighting technologies, implementing occupancy sensors, daylight harvesting systems, and smart lighting controls are employed to optimize energy usage and minimize environmental impact while maintaining adequate illumination levels.

Moreover, the lights on state can influence human circadian rhythms and well-being, particularly in indoor environments where exposure to natural light is limited. Research suggests that appropriate lighting conditions, including the duration, intensity, and color temperature of artificial lighting, can impact mood, alertness, and overall health. Therefore, designing lighting systems that mimic natural daylight patterns and allow for dynamic adjustment based on user needs and preferences can help promote productivity, comfort, and well-being.

4.1.2 LIGHTS OFF STATE

Fig 4.2 Lights Off State

The "lights off" state represents the condition in which lighting systems are deactivated, resulting in darkness or reduced illumination within a space. The theory behind the lights off state involves understanding the factors that influence when lights are turned off and the implications of maintaining this state.

Turning lights off is typically determined by factors such as occupancy patterns, time of day, natural light levels, and energy conservation goals. In spaces where occupants have left or where natural daylight provides sufficient illumination, lights are often switched off to conserve energy and reduce electricity consumption. This practice aligns with sustainability initiatives and efforts to minimize environmental impact by reducing unnecessary energy usage.

The lights off state is essential for optimizing energy efficiency and reducing operational costs associated with lighting systems. By implementing strategies such as occupancy sensors, timers, and automated controls, lighting can be dynamically adjusted based on occupancy patterns and natural light availability, ensuring that lights are only active when needed. These measures help organizations and individuals achieve energy savings and contribute to their sustainability objectives.

Furthermore, the lights off state can influence the comfort and well-being of occupants, particularly during nighttime hours when reduced light levels promote relaxation and restful sleep. Creating environments with appropriate lighting conditions supports healthy circadian rhythms and contributes to overall occupant satisfaction and productivity.

However, it's essential to balance energy-saving efforts with considerations for safety and security. While turning lights off can reduce energy consumption, maintaining adequate lighting levels in certain areas, such as entryways, corridors, and outdoor spaces, is necessary to ensure visibility, deter crime, and mitigate safety hazards.

4.1.3 PHOTO FRAME ON STATE

Fig.4.3 Photo Frame On State

The "lights on" state in a photo frame refers to the condition in which the internal illumination of the frame is activated, providing a backlight or accent lighting at top off the roof. The theory behind the lights on state involves enhancing the visual presentation of the displayed content, drawing attention to the framed image, and creating a visually appealing focal point within the space.

Photo frames with built-in lighting elements are designed to highlight and complement the aesthetics of the displayed photographs or artwork. By illuminating the image from behind or along its edges, the lights on state adds depth, dimensionality, and visual interest to the displayed content, enhancing its visibility and attractiveness. This lighting effect can help to create a captivating ambiance and draw viewers' attention to the framed piece, making it a focal point within the room or environment.

The lights on state in a photo frame is often activated in settings where additional accent lighting is desired to enhance the display. For example, in dimly lit rooms or areas with limited natural light, the internal illumination of the photo frame can provide supplementary lighting to improve visibility and showcase the displayed content more effectively. Similarly, in environments where ambient lighting is adjustable, activating the lights on state in the photo frame can contribute to creating a desired atmosphere or mood.

Moreover, the lights on state in a photo frame can serve practical purposes such as emphasizing specific details or colors within the displayed image, creating visual contrast, or adding a decorative element to the surrounding space. By strategically positioning the lighting elements and adjusting their intensity, designers can customize the illumination to suit the aesthetics and objectives of the display.

4.1.4 PHOTO FRAME OFF STATE

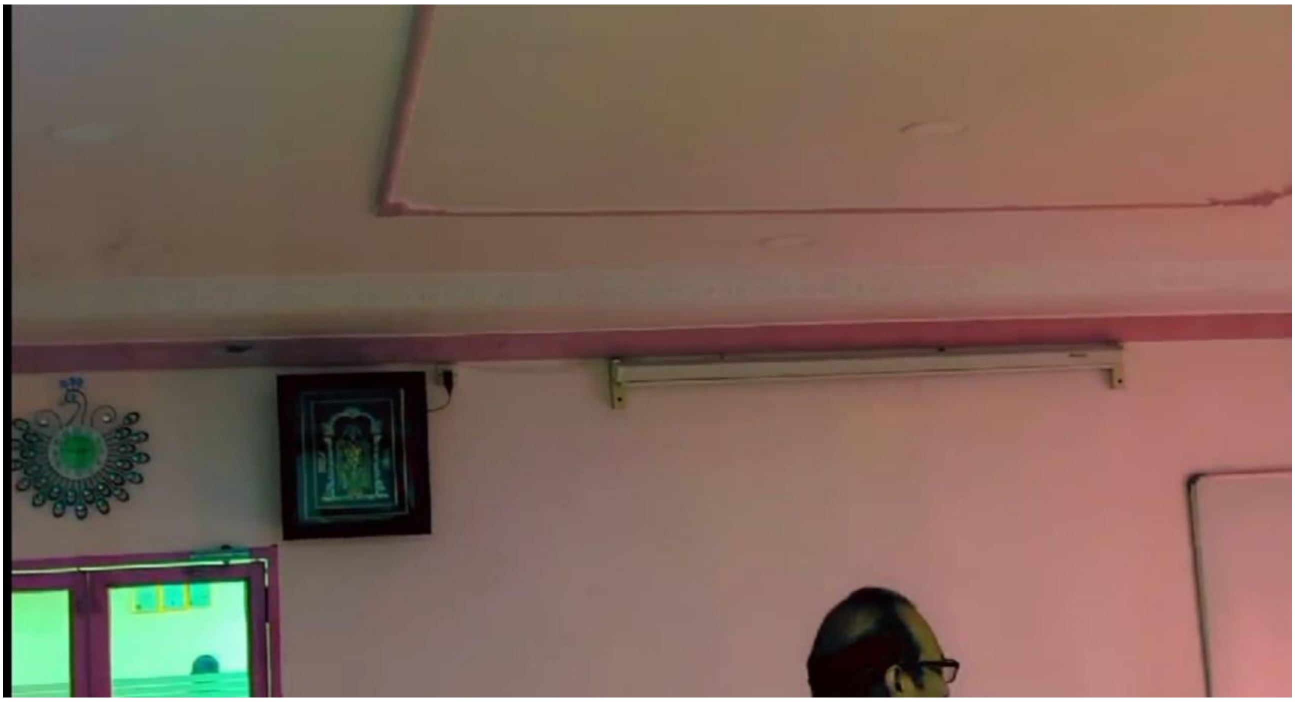

Fig.4.4 Photo Frame Off State

The "lights off" state in a photo frame refers to the condition in which the internal illumination of the frame is deactivated, resulting in the absence of backlight or accent lighting around the displayed photograph or artwork. The theory behind the lights off state involves creating a subdued or natural presentation of the displayed content, allowing it to stand on its own without additional visual embellishments.

When the lights are off in a photo frame, the focus shifts to the inherent qualities of the displayed photograph or artwork, such as its composition, colors, and subject matter. By eliminating the artificial illumination, the frame achieves a more neutral appearance that enables viewers to engage with the content in its purest form. This minimalist approach emphasizes the authenticity and integrity of the displayed images, fostering a deeper connection and appreciation for the artwork or memories captured within.

The lights off state in a photo frame is often preferred in environments where a more understated and tranquil ambiance is desired. By removing the artificial lighting, the frame integrates seamlessly into its surroundings, blending into the décor without drawing unnecessary attention to itself. This minimalist aesthetic contributes to a sense of calmness and simplicity within the space, allowing viewers to focus on the content of the frame without distraction.

Furthermore, the lights off state in a photo frame can serve practical purposes such as conserving energy and minimizing visual clutter. By deactivating the internal illumination when not in use, the frame reduces electricity consumption and environmental impact, aligning with sustainability objectives. Additionally, the absence of backlighting eliminates potential glare or reflections on the displayed content, ensuring optimal viewing conditions and reducing eye strain for viewers.

4.1.5 FAN OFF STATE

Fig.4.5 Fan Off State

The "fan off" state refers to the condition in which a ceiling or room fan is deactivated, resulting in the cessation of airflow and movement within the space. The theory behind the fan off state involves creating a quieter and more still environment, where the fan ceases to provide cooling or air circulation.

When a fan is turned off, the room returns to a state of tranquility, with ambient noise levels decreasing and the absence of the gentle breeze created by the fan blades. This quieter atmosphere can be conducive to relaxation, concentration, or conversation, depending on the context and preferences of the occupants.

The fan off state is typically employed when the cooling or air circulation provided by the fan is no longer needed or desired. For example, during cooler weather or when the room temperature is comfortable, turning off the fan can help conserve energy and reduce electricity consumption. Similarly, in spaces where the fan's operation may be disruptive, such as during meetings, presentations, or sleep, deactivating the fan can promote a more conducive environment for focused activities or rest.

Moreover, the fan off state allows for the maintenance of the fan and its associated components, as periods of inactivity provide opportunities for cleaning, inspection, and repair. By periodically turning off the fan when not in use, users can extend its lifespan and ensure optimal performance over time.

4.1.6 FAN ON STATE

Fig.4.6 Fan On State

The "fan on" state signifies the condition in which a ceiling or room fan is activated, facilitating airflow and movement within a space. The theory behind the fan on state involves harnessing the fan's rotational motion to circulate air, dissipate heat, and maintain thermal comfort.

When a fan is turned on, its blades begin to rotate, creating a downward airflow that helps to distribute air throughout the room. This movement promotes air circulation, which can aid in cooling occupants by enhancing evaporative cooling effects on the skin and reducing the perceived temperature in warmer environments. Additionally, the gentle breeze generated by the fan can create a sensation of comfort and freshness, particularly during hot or humid weather conditions.

The fan on state is commonly utilized to supplement or complement other cooling systems, such as air conditioners or natural ventilation, by improving air distribution and thermal comfort within the space. By circulating air more efficiently, fans can help to reduce the reliance on mechanical cooling systems, leading to energy savings and lower electricity costs. Moreover, in environments where air conditioning is not feasible or cost-effective, fans serve as an economical and environmentally friendly alternative for maintaining comfort.

Furthermore, the fan on state can contribute to indoor air quality by promoting ventilation and reducing the buildup of stagnant air, odors, and pollutants. By continuously circulating air, fans help to disperse airborne particles and maintain a healthier indoor environment for occupants.

In addition to its functional benefits, the fan on state can also serve aesthetic and decorative purposes, as ceiling fans often feature stylish designs and integrated lighting options that enhance the visual appeal of the space. Moreover, the gentle sound of the fan's operation can create a soothing ambiance that masks background noise and promotes relaxation or concentration, depending on the preferences of the occupants.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

Integration of IOT technologies, the smart home automation system enhances convenience, energy efficiency, security, and overall comfort for the residents.

Smart home automation offers numerous benefits, revolutionizing the way we interact with our living spaces. Through seamless integration of technology, smart homes provide convenience, efficiency, and enhanced security. However, while the potential for interconnectedness is vast, it's essential to prioritize privacy, security, and interoperability to ensure a smooth user experience.

In conclusion, the future of smart home automation holds great promise. As technology continues to advance, we can expect even more sophisticated systems that adapt to our needs, enhance our comfort, and contribute to sustainable living. However, it's crucial to approach these advancements thoughtfully, considering the implications on privacy, security, and accessibility for all users. With careful planning and innovation, smart home automation will undoubtedly continue to enrich our lives in meaningful ways.

5.2 FUTURE SCOPE

The future scope of smart home automation is poised for exponential growth and innovation. As technology evolves the Internet of Things (IOT) to create homes that are not just smart but truly intelligent. These homes will seamlessly anticipate and adapt to our needs, learning from our habits and preferences to personalize experiences. With advancements in sensor technology and connectivity, we can expect homes to become more proactive in managing energy consumption, optimizing resources, and promoting sustainability. Moreover, the integration of augmented reality (AR) and virtual reality (VR) could revolutionize how we interact with our homes, enabling immersive experiences for design, maintenance, and entertainment. As the smart home ecosystem expands, interoperability will be key, allowing different devices and platforms to communicate seamlessly. However, amidst these exciting prospects, it's crucial to address concerns regarding privacy, security, and accessibility to ensure that the benefits of smart home automation are inclusive and responsibly implemented. Overall, the future of smart home automation holds tremendous promise, offering unparalleled convenience, efficiency, and quality of life enhancements for residents around the globe.

REFERENCES

1.Kalafat, E.; Leslie, K.; Bhide, A.; Thilaganathan, B.; Khalil, A. Smart Home following home Appliances monitoring in home. 2019, 18, 14–20. [Google Scholar] [CrossRef]

2. K. Ashton, That ‘Internet of Things’ Thing.

3.BCC Research,“Sensors: Technologiesandglobalmarkets,” BCCResearch, 2k011.

4.I. F.Akyildiz, Y.Weilian, Y.Sankarasubramaniam and E. Cayirci, “Asurvey on sensor networks,” *Communications Magazine, IEEE*, vol. 40, no. 8, pp. 102-114, August 2002.

5.Pranay P. Gaikwad, Jyotsna P. Gabhane, Snehal S. Golait , “A Survey based on Smart Homes System Using Internet-of-Things,” IEEE 2015

APPENDIX

APPENDIX-A

INTRODUCTION TO SOFTWARE

The Arduino Integrated Development Environment (IDE) is a software platform used to program Arduino microcontrollers. It provides a user-friendly interface for writing, compiling, and uploading code to Arduino boards. With a simple and intuitive design, the Arduino IDE is accessible to beginners while offering advanced features for experienced users, making it a popular choice for hobbyists and professionals alike. Its open-source nature encourages collaboration and the sharing of code libraries, fostering a vibrant community of developers worldwide.

Fig.Arduino IDE

The Arduino Integrated Development Environment (IDE) is a software platform designed to facilitate the programming of Arduino microcontroller boards. It provides a comprehensive set of tools for writing, compiling, and uploading code to Arduino hardware, simplifying the development process for both beginners and experienced users. With a user-friendly interface and a simple programming language based on Wiring, the Arduino IDE enables users to quickly prototype and deploy projects without the need for extensive programming knowledge. Additionally, the IDE offers a rich library of pre-written code examples and community-contributed libraries, fostering a collaborative environment where users can easily share and build upon each other's work. Overall, the Arduino IDE serves as a central hub for the Arduino ecosystem, empowering creators to bring their ideas to life through the seamless integration of hardware and software development.

ARDUINO SOFTWARE INTERFACE

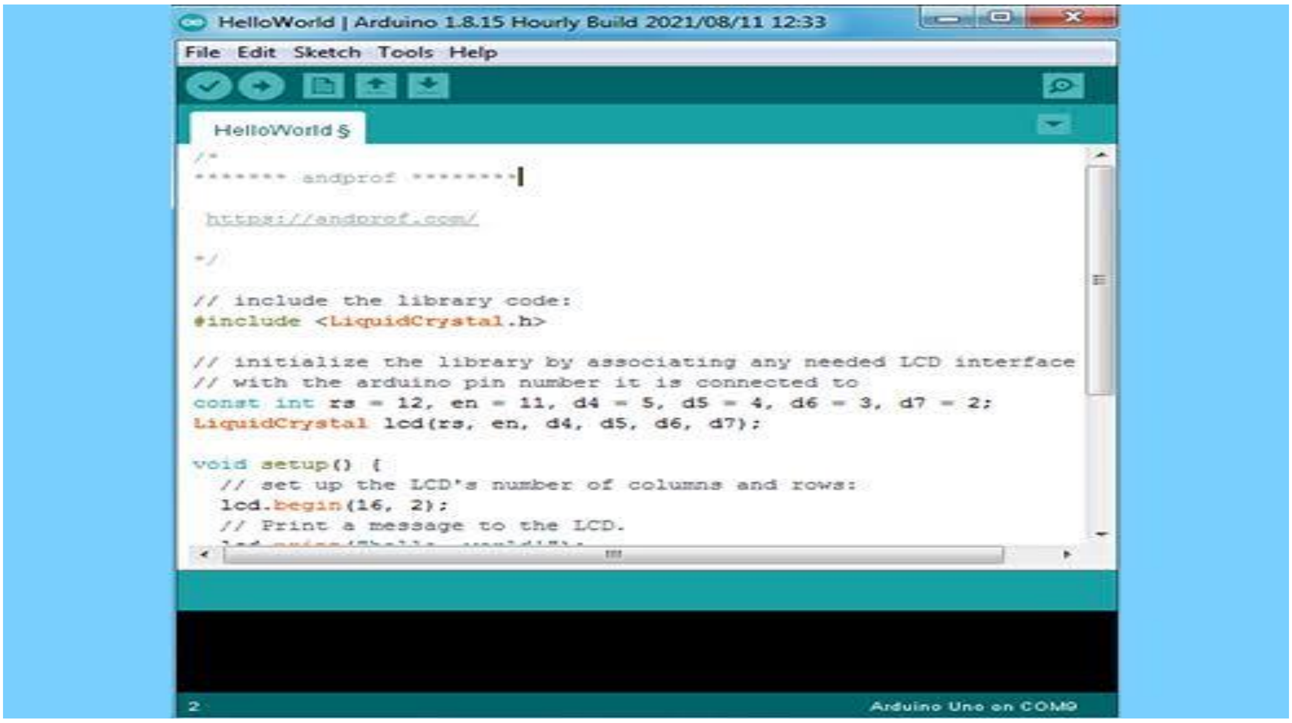

Fig. Arduino Software Interface

The Arduino software interface serves as a pivotal tool in the development of embedded systems and IOT projects. Its user-friendly design and intuitive layout make it accessible to both novice and experienced programmers. The interface consists of several key components, including the code editor, serial monitor, and library manager, which collectively streamline the programming workflow. The code editor provides a convenient platform for writing and editing code in the Arduino programming language, while the serial monitor allows users to communicate with their Arduino boards and debug their projects in real-time. Additionally, the library manager grants access to a vast repository of pre-written code libraries, enabling developers to easily integrate complex functionalities into their projects without reinventing the wheel. Overall, the Arduino software interface plays a crucial role in empowering creators to unleash their creativity and bring their innovative ideas to life in the realm of embedded systems and IOT.

MENU SECTION

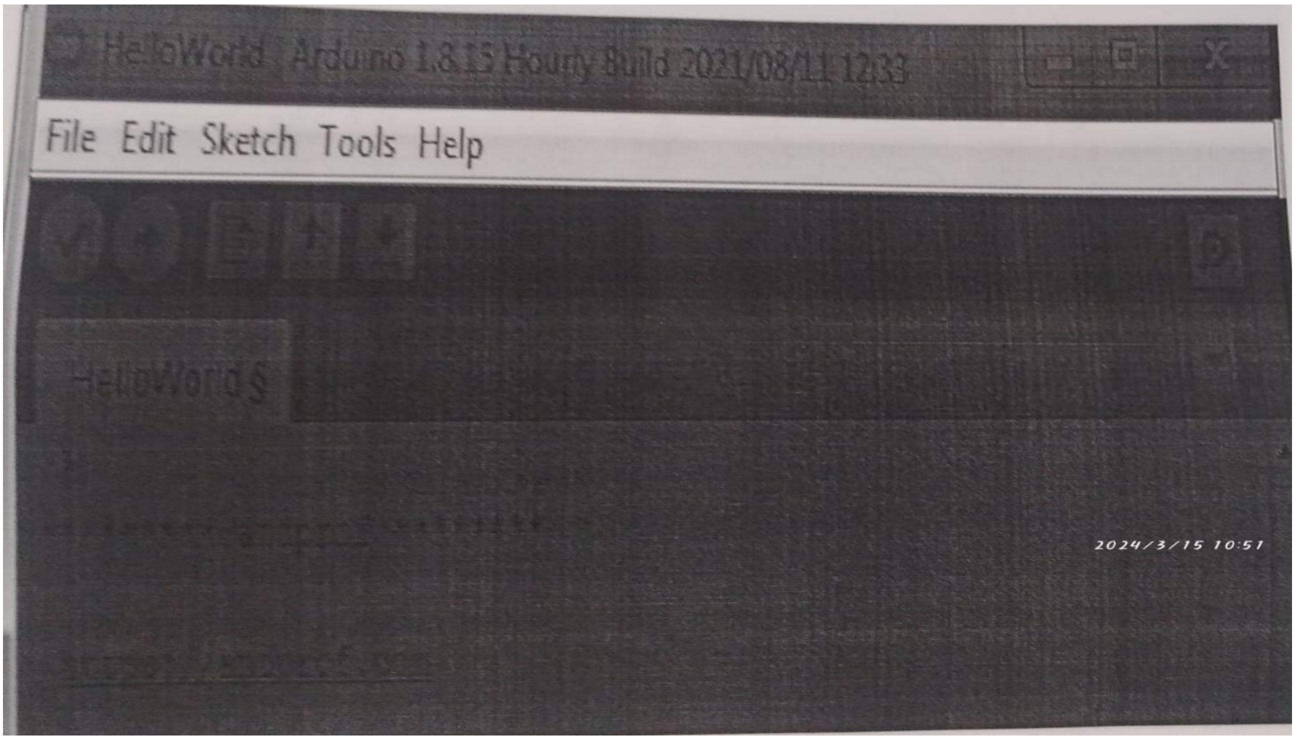

Fig.Menu Section

The menu bar in the Arduino IDE serves as a comprehensive toolbox for developers, offering a range of functionalities to streamline the coding process for Arduino projects. It typically includes options for creating, opening, and saving sketches, managing libraries, selecting the Arduino board and port, and uploading sketches to the board. The "Sketch" menu provides access to essential functions such as verifying and uploading code, while the "Tools" menu offers configuration settings for the board, serial port, and programmer. Additionally, features like the "Examples" menu provide beginners with a wealth of sample code to learn from and experiment with. Overall, the menu bar in the Arduino IDE is a vital component, providing users with a user-friendly interface to navigate various aspects of Arduino development.

TOOL BAR SECTION

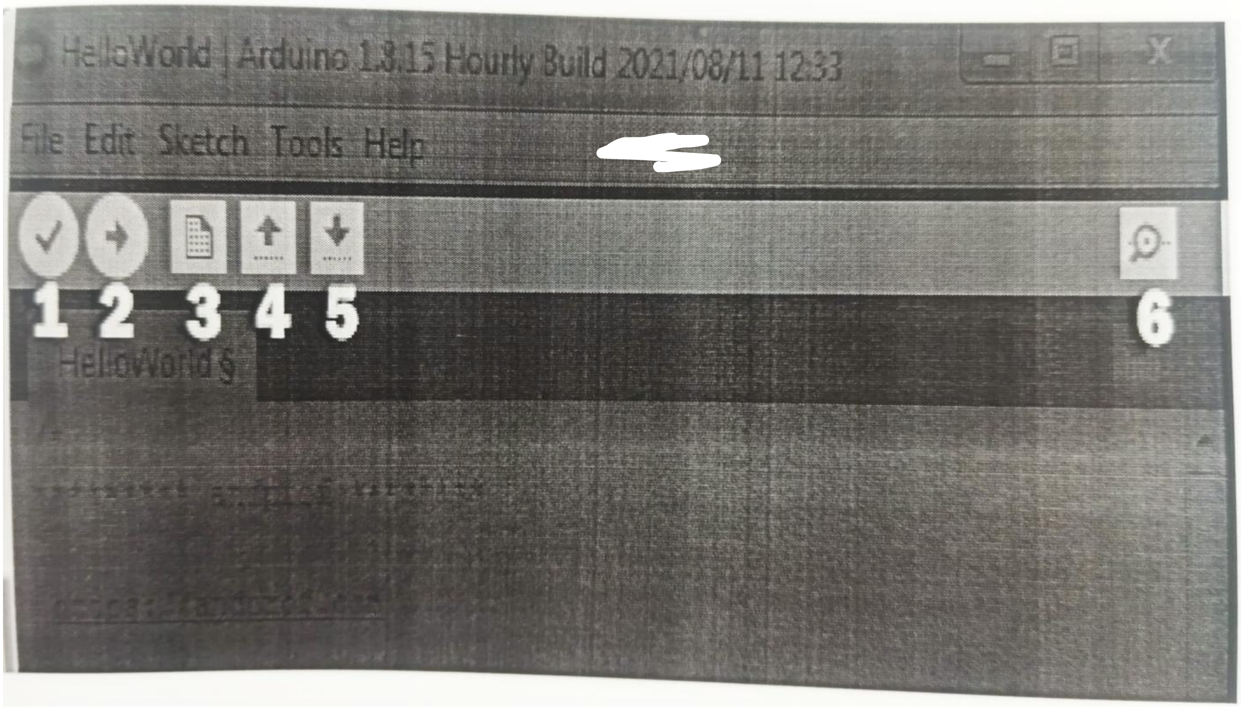

Fig.Tool Bar Section

The toolbar section in the Arduino Integrated Development Environment (IDE) constitutes a vital component that enhances the user experience and accelerates the development process. Positioned prominently at the top of the IDE window, the toolbar provides quick access to essential functions and commands, streamlining the programming workflow. It includes commonly used actions such as opening, saving, and verifying sketches, as well as uploading code to Arduino boards. Additionally, the toolbar typically features shortcuts for commonly used editing tools like copy, paste, and find, further improving efficiency in code writing and manipulation. Furthermore, the toolbar may include buttons for managing libraries, configuring board settings, and selecting communication ports, offering users convenient access to key functionalities without navigating through menus. Overall, the toolbar section serves as a central hub for accessing essential tools and commands, empowering users to focus on their projects and iterate rapidly in the development of Arduino-based applications.

STATUS BAR SECTION

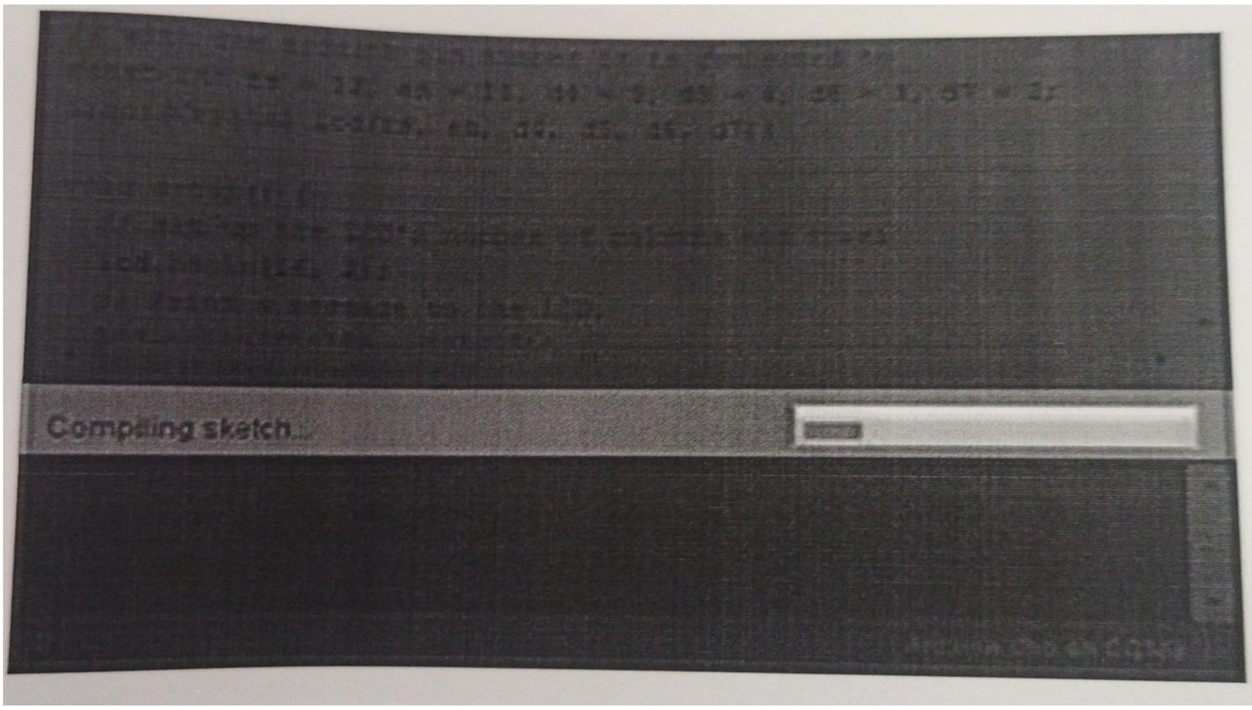

Fig. Status Bar Section

The status bar section in the Arduino Integrated Development Environment (IDE) provides users with valuable information about the current state of their projects and the IDE itself, enhancing the overall programming experience. Typically located at the bottom of the IDE window, the status bar displays important details such as the status of the compilation process, indicating whether the code is error-free or if there are any compilation errors that need to be addressed. It also shows the current line and column numbers, aiding in code navigation and debugging. Additionally, the status bar may include indicators for the selected programming language, the current encoding format, and the status of the serial monitor, providing users with relevant contextual information at a glance. Overall, the status bar section serves as a convenient reference point for monitoring the progress of programming tasks and ensuring a smooth development process within the Arduino IDE.

CODE EDITOR SECTION

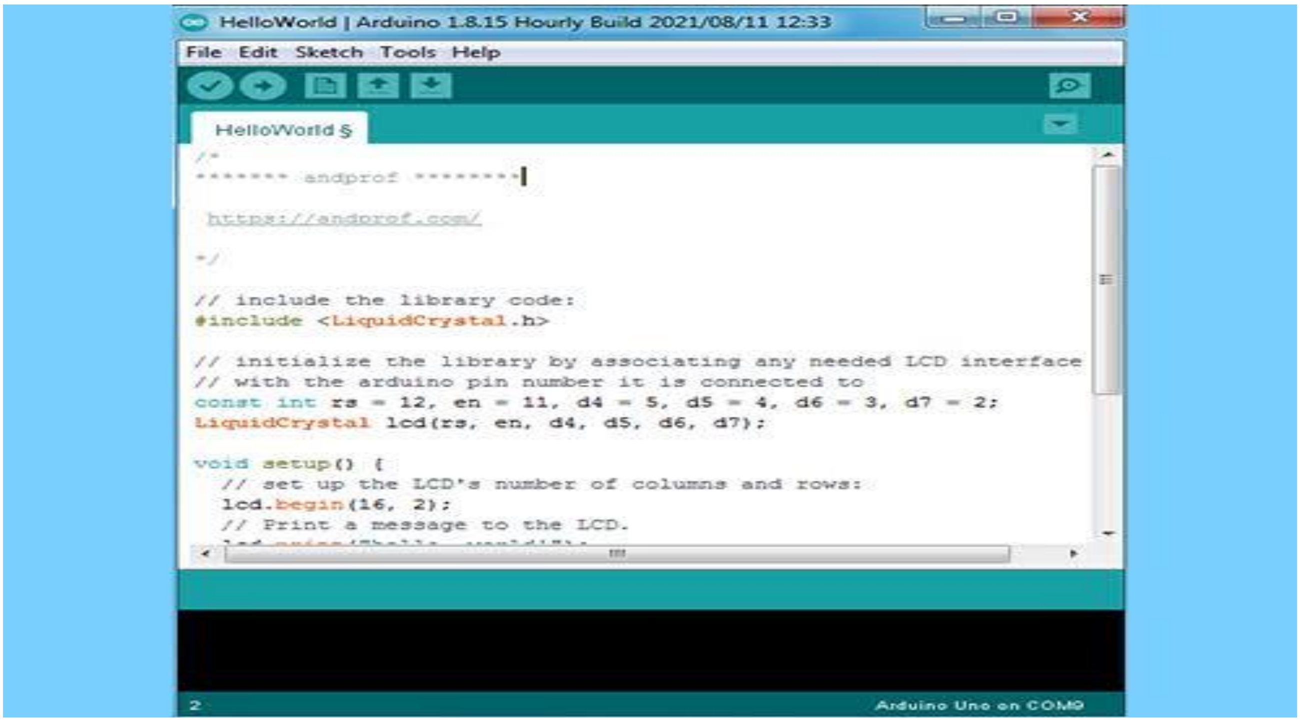

Fig.Code Editor

The "code bar sector" in Arduino theory typically refers to a specific area of memory on an Arduino microcontroller where the program code is stored. In Arduino, this is often referred to as the "Flash memory" or "Program memory." When you upload a sketch (program) to an Arduino board, it gets stored in this code bar sector. During execution, the Arduino's processor reads instructions from this memory to carry out the programmed tasks. This sector is crucial for storing and executing the instructions that define the behavior of your Arduino project. Efficient management of this memory is essential for optimizing code size and performance in embedded systems development.

APPENDIX-B: Project Code

```
#include <ESP8266WiFi.h>
// Add wifi access point credentiaals
const char* ssid     = "IOT_HOME";
const char* password = "12345678";
WiFiServer se
rver(80);// Set port to 80
String header; // This storees the HTTP request
#define Light 16
#define Socket 5
#define Fan   4

IPAddress local_IP(192, 168, 1, 184);
IPAddress gateway(192, 168, 1, 1);

IPAddress subnet(255, 255, 0, 0);
IPAddress primaryDNS(8, 8, 8, 8);


String BULB = "off";// state of appliance1
String SOCKET = "off";// state of appliance2
String FAN = "off";
 ////WiFi.mode(WIFI_STA);
void setup() {
  Serial.begin(9600);

them low to prevent flunctuations
  pinMode(Light, OUTPUT);
  pinMode(Socket, OUTPUT);
  pinMode(Fan, OUTPUT);
  digitalWrite(Light, LOW);
  digitalWrite(Socket, LOW);
  digitalWrite(Fan, LOW);
```

```
  //connect to access point
  WiFi.begin(ssid, password);
  Serial.print("Connecting to ");
  Serial.println(ssid);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  // Print local IP address and start web server
  Serial.println("");
  Serial.println("WiFi connected.");
  Serial.println("IP address");
  Serial.println(WiFi.localIP());// this will display the Ip address of the Pi which
should be entered into your browser
  server.begin();
}
void loop(){
  WiFiClient client = server.available();   // Listen for incoming clients
  if (client) {                             // If a new client connects,
    String currentLine = "";                // make a String to hold incoming data
from the client
    while (client.connected()) {            // loop while the client's connected
      if (client.available()) {             // if there's bytes to read from the client,
        char c = client.read();             // read a byte, then
        Serial.write(c);                    // print it out the serial monitor
        header += c;
        if (c == '\n') {                    // if the byte is a newline character
          // if the current line is blank, you got two newline characters in a row.
          // that's the end of the client HTTP request, so send a response:
          if (currentLine.length() == 0) {
            // HTTP headers always start with a response code (e.g. HTTP/1.1 200
OK)
            // and a content-type so the client knows what's coming, then a blank
line:
            client.println("HTTP/1.1 200 OK");
```

```
client.println("Content-type:text/html");
client.println("Connection: close");
client.println();

// turns the GPIOs on and off
if (header.indexOf("GET /Light/on") >= 0) {
  Serial.println("App 1 on");
  BULB = "on";
  digitalWrite(Light, HIGH);
} else if (header.indexOf("GET /Light/off") >= 0) {
  Serial.println("App 1 off");
  BULB = "off";
  digitalWrite(Light, LOW);
} else if (header.indexOf("GET /Socket/on") >= 0) {
  Serial.println("App 2 on");
  SOCKET = "on";
  digitalWrite(Socket, HIGH);
} else if (header.indexOf("GET /Socket/off") >= 0) {
  Serial.println("App 2 off");
  SOCKET = "off";
  digitalWrite(Socket, LOW);
}
else if (header.indexOf("GET /Fan/on") >= 0) {
  Serial.println("App 3 on");
  FAN = "on";
  digitalWrite(Fan, HIGH);
} else if (header.indexOf("GET /Fan/off") >= 0) {
  Serial.println("App 3 off");
  FAN = "off";
  digitalWrite(Fan, LOW);
}

// Display the HTML web page
client.println("<!DOCTYPE html><html>");
client.println("<head><meta name=\"viewport\"
```

```
content=\"width=device-width, initial-scale=1\">");
            client.println("<link rel=\"icon\" href=\"data:,\">");
            // CSS to style the on/off buttons
            // Feel free to change the background-color and font-size attributes to
fit your preferences
            // Web Page Heading
            client.println("<body><h1>IOT HOME AUTOMATION</h1>");

            // Display current state, and ON/OFF buttons for GPIO 5
            client.println("<p>Light - State " + BULB + "</p>");
            // If Appliance 1 is off, it displays the ON button
            if (BULB == "off") {
              client.println("<p><a href=\"/Light/on\"><button
class=\"button\">ON</button></a></p>");
            } else {
              client.println("<p><a href=\"/Light/off\"><button class=\"button
button2\">OFF</button></a></p>");
            }

            // Display current state, and ON/OFF buttons for GPIO 4
            client.println("<p>Socket - State " + SOCKET + "</p>");
            // If Appliance 2 is off, it displays the ON button
            if (SOCKET == "off") {
              client.println("<p><a href=\"/Socket/on\"><button
class=\"button\">ON</button></a></p>");
            } else {
              client.println("<p><a href=\"/Socket/off\"><button class=\"button
button2\">OFF</button></a></p>");
            }

             client.println("<p>Fan - State " + FAN + "</p>");
            // If Appliance 2 is off, it displays the ON button
            if (FAN == "off") {
              client.println("<p><a href=\"/Fan/on\"><button
class=\"button\">ON</button></a></p>");
```

```
            } else {
              client.println("<p><a href=\"/Fan/off\"><button class=\"button
button2\">OFF</button></a></p>");
            }
            client.println("</body></html>");

            // The HTTP response ends with another blank line
            client.println();
            // Break out of the while loop
            break;
          } else { // if you got a newline, then clear currentLine
            currentLine = "";
          }
        } else if (c != '\r') {  // if you got anything else but a carriage return
character,
          currentLine += c;      // add it to the end of the currentLine
        }
      }
    }
    // Clear the header variable
    header = "";
    // Close the connection
    client.stop();
    Serial.println("Client disconnected.");
    Serial.println("");
  }
}
```