SMART ENERGY MONITORING APPLICATION

A PROJECT REPORT

Submitted by

S. PREETHI [REGISTER NO: 211417104200]

M. SANGAVI PRIYAA [REGISTER NO: 211417104237]

SHANMUGAM MITHRA [REGISTER NO: 211417104253]

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123

ANNA UNIVERSITY: CHENNAI 600 025

AUGUST 2021

BONAFIDE CERTIFICATE

Certified that this project report “SMART ENERGY MONITORING APPLICATION**”** is the bonafide work of **S.PREETHI [REGISTER NO: 211417104200], M. SANGAVI PRIYAA [REGISTER NO: 211417104237], SHANMUGAM MITHRA [REGISTER NO: 211417104253]** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr.S.MURUGAVALLI,M.E.,Ph.D.,	**Dr.L.Jaba Sheela**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	PROFESSOR
DEPARTMENT OF CSE,	DEPARTMENT OF CSE,
PANIMALAR ENGINEERING COLLEGE,	PANIMALAR ENGINEERING COLLEGE,
NASARATHPETTAI,	NASARATHPETTAI,
POONAMALLEE,	POONAMALLEE,
CHENNAI-600 123.	CHENNAI-600 123.

Certified that the above candidates were examined in the Anna University Project Viva-Voce Examination held on....05/08/2021.....

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.,** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like to extend our heartfelt and sincere thanks to our Directors **Tmt.C.VIJAYARAJESWARI**, **Dr.C.SAKTHIKUMAR,M.E.,Ph.D.,** and **Tmt. SARANYASREE SAKTHIKUMAR B.E.,M.B.A.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr.K.Mani, M.E., Ph.D.,** for his timely concern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr.S.MURUGAVALLI , M.E.,Ph.D.,**for the support extended throughout the project.

We would like to thank my project guide **Dr.L. JABA SHEELA M.E, Ph.D.,** and all the faculty members of the Department of CSE for their advice and suggestions for the successful completion of the project.

We would like to thank god for showering his blessing on us and we also thank our parents much for their support both emotional and financial over the years.

S. PREETHI
M. SANGAVI PRIYAA
SHANMUGAM MITHRA

ABSTRACT

Internet of Things (IOT) is revolutionizing Industries faster than ever. IOT is the network of physical objects or things embedded with electronics, software, sensors, and network connectivity. It enables these objects to collect and exchange data. In this project, we will be developing a system which will automatically monitor all the currently operating appliances in a building. This system can generate alerts or take intelligent decisions using concept of IOT along with real-time data updating with cloud server. IOT has given us a promising way to build powerful industrial systems and applications by using Wi-Fi devices, LTE modems, smart relay switches and sensors. The main contribution of the project is that it summarizes the uses of IOT in organizations like colleges or schools. It also shows how we can monitor and control the appliance by increasing productivity and reducing expenses. Our system uses a microcontroller for processing all user commands and an EEPROM chip to save data. Digital Voltmeter, Ammeter, and Watt meter can be linked to the same IOT device without any additional circuits so thereby reducing cost. A Wi-Fi router and modem is linked to the IOT module to update each machine status. Appliance can receive user commands over Internet Protocol. On sending commands from mobile through the internet, it will be received by the modem linked with IOT circuit. The modem then decodes information and passes it to the microcontroller for further processing. We can get real-time data of the power consumed by appliance. We can extract data stored in the chip by importing it into a MS excel file. The chip can store data up to three months. It is also ensured that the hardware and technology used in the proposed idea are cheap, easily available, and replicable. The experimental results highlight its significance and validate the proof of the concept.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

S.NO	ABBREVIATION	EXPANSION
1.	OPS	Overload Protection Service
2.	PCR	Power Consumption Recoder
3.	CSV	Comma Separated Values
4.	EEPROM	Electronically Erasable Programmable Read Only Memory
5.	MOSFET	Metal Oxide Semiconductor Field Effect Transistor
6.	IP	Internet Protocol
7.	IOT	Internet Of Things
8.	PNP	Positive Negative Positive
9.	OPC UA	Open Platform Communications United Architecture
10.	PLC	Programmable Logic Controller
11.	LTE	Long Term Evolution
12.	EMMC	Embedded Multimedia Card
13.	SFDP	Serial Flash Discoverable Parameter
14.	JEDEC	Joint Electron Device Engineering Council

CHAPTER 1

CHAPTER 1

INTRODUCTION

1.1. OVERVIEW

This system makes sure that we can view the current consumption patterns of any institution, building or office from anywhere. We can view in real-time which building in an organization is consuming the most power. The real time data is presented in pictorial representation for easy analysis. We can view the real-time data using this project. This will help us to save energy and money. We will be creating an application and will be using IOT to connect this application with our hardware device. The hardware device that we will be creating will contain the ammeter, voltmeter and wattmeter to measure the amount of current flowing through. This data will be sent to the application in our mobile phone.

1.2. PROBLEM DEFINITION

Everyone owns a phone these days and viewing data from mobile phone from anywhere through an application is very convenient for everyone. A person that owns a large institution or organization will find it difficult to monitor all the buildings. These people will be paying a lot of money every month just for their current bills. It is important to use technology for them to identify which building consumes more current so that they can make changes accordingly to save energy and money. It is possible to view real time data using mobile application to make changes to your organization to save energy using this project. We can see if any appliances are still in use in non-operating hours.

CHAPTER 2

CHAPTER 2
LITERATURE SURVEY

1. **TITLE:** Smart Energy Efficient Home Automation System Using IOT

   AUTHORS: Satyendra K. Vishwakarma, Prashant Upadhyaya, Babita Kumari, Arun Kumar Mishra

   YEAR: 2019

   DESCRIPTION:
   Satyendra K. Vishwakarma[15] have proposed a system that can access and control the home equipments from every corner of the world. For this system, Internet connectivity module is attached to the main supply unit of the home system which can be accessed through the Internet. For wireless connectivity, the static IP address is used. Home automation is based on multimodal application that can be operated using voice recognition command of the user using the Google Assistant or through a web based application. Thus, main objective of this work is to make our home automation system more secure and intelligent.

   MERITS:
   The advantage of Google assistance and web based application can be helpful in controlling the appliance of the system. Thus, the proposed model is designed to provide better flexibility and making the system more robust.

   DEMERITS:

   There is no EEPROM chip used to store critical machine data's during power failures. Instead of Wi-Fi, Bluetooth technology is used which leads to limited device controlling distance. It can just only Turn-On or Turn-Off devices. No option is given to view current machine status over internet.

2. **TITLE:** A Step Towards Home Automation Using IOT

   AUTHORS: Harsh Kumar Singh; Saurabh Verma; Shashank Pal; Kavita Pandey

YEAR: 2019

DESCRIPTION:

Harsh Kumar Singh [12] has proposed a system to develop home automation system based on IOT using Wi-Fi based microcontroller. As scope of technology is widening every day, we are making our tech advance in mobile, robotics, machine learning, then why an exception for our home. Today's houses are gradually transferring from ordinary/human's input-based appliances to smart/IOT enabled appliances to be controlled remotely. At Present, existing home automation systems use technology that is limited to only that device. So, in a nutshell, we are making our devices IOT enabled not our homes. As far as this paper is concerned, Node MCU (ESP8266) microcontroller along with relays is used to control electrical switches remotely from the server which is built on Node.js. User can control switches using a Web Application after authenticating.

MERITS:

Implementation of automation system with Z-wave is less complicated and has greater range than automation system implemented with ZigBee.

DEMERITS:

No protection circuit added to connected devices to prevent voltage spikes. No MOSFET used for fast switching and endurance. This device can control only home appliance only and incompatible for sensitive devices.

3. **TITLE:** Smart Home Automation Using IOT-Based Sensing and Monitoring Platform

AUTORS: Majid Al-Kuwari, Abdul rhman Ramadan,Yousef Ismael, Laith Al-Sughair

YEAR: 2019

DESCRIPTION:

Majid Al-Kuwari[1] have proposed a system that monitors some parameters to help maintain them within an acceptable range. This paper presents the complete design of an IOT based sensing and monitoring system for smart home automation. The proposed design uses the EmonCMS platform for collecting and visualizing monitored data and remote controlling of home appliances and devices. The selected platform is very flexible and user-friendly. The sensing of different variables inside the house is conducted using the Node MCU-ESP8266 microcontroller board, which allows real-time data sensing, processing and uploading/downloading to/from the CMS cloud server.

MERITS:

Automation can be accomplished by using the Internet of Things (IoT). This gives the inhabitant accesses to certain data in the house and the ability to control some parameters remotely.

DEMERITS:

Microcontroller functions cannot be reprogrammed for adding additional functions from Android application. No secured Authentication over controlling via internet. It can be used for monitoring small homes only. Not suitable for large buildings.

4. **TITLE:** Self-Commissioning Industrial IoT-Systems in Process Automation: Reference Architecture

AUTHORS: Heiko Koziolek; Andreas Burger; Jens Doppelhamer

YEAR: 2018

DESCRIPTION:

Heiko Koziolek[7] have proposed a system where commissioning shall be largely automated, but they have suffered from semantic ambiguities and usually rely on proprietary information models. We propose a novel reference architecture for PnP in IOT systems, which is based on OPC UA and PLC open standards and can reduce industrial device commissioning times across vendor products to a few seconds. Our proof-of-concept implementation can handle more than 500 signals per millisecond during runtime, sufficient for most application scenarios.

MERITS:

PnP in IOT systems, which is based on OPC UA and PLC open standards and can reduce industrial device commissioning times across vendor products to a few seconds.

DEMERITS:

Devices can be controlled using laptop only. No provision given for controlling devices via android mobiles. Since cable is used as interface between the IOT chip and Controller. Each and every time we need to plugin for making any changes in coding.

5. **TITLE:** Enhanced Home Automation System using Internet of Things

   AUTHORS: S.L.S. Sri Harsha; S. Chakrapani Reddy; S. Prince Mary

   YEAR: 2017

   DESCRIPTION:

   S.L.S. Sri Harsha[6] have proposed a system that suggest a highly intractable and environmentally sustainable form of Home Automation System using Internet of Things a means to control the appliances at home via a device with access to the Internet. The key components of this system are a pocket sized microprocessor-Raspberry Pi and a microcontroller-Arduino Uno and an Android application to visualize the data provided by the Raspberry Pi and also to send, receive and process the requests. The Raspberry Pi acts as the brain of this system, processing the requests, responding to the requests made by the Android application, communicating with the Arduino and also acts as a server to store the data given by the sensors. All the sensors and actuators are connected to the Arduino which is connected to the Pi using a USB cable. Our main objective of developing this model is to create a home automation system which interacts with the user through various push notifications based on concerned parameters which is also eco-friendly.

   MERITS:

   For this system, Internet connectivity module is attached to the main supply unit of the home system which can be accessed through the Internet. It can access and control the home equipments from every corner of the world.

DEMERITS:

Complex wiring method used which leads to more maintenance if any wire fault occurs. Unable to export data to other devices for Report generation.

CHAPTER 3

CHAPTER 3

SYSTEM ANALYSIS

System Analysis is an activity that encompasses most of the tasks that we have collectively called computer system engineering. System Analysis is conducted with the following objectives in mind:

- Identity the customers need
- Evaluate the system concept for feasibility
- Perform economic and technical analysis
- Allocate functions to networks, software, people, database and other system elements
- Establish cost and schedule constraints
- Create a system definition that forms the foundation for all subsequent engineering work.

3.1 Existing System

The existing system contains automation system for home. However, it does not contain EEPROM chip to store the data of the system. The existing system uses Bluetooth technology which does not allow user to control long distance transmissions. It is not possible to know the status of the machine through the internet. The existing system does not contain MOSFET for fast switching.

3.2 Proposed system

The proposed system can be used to control large industries and contains MOSFET for fast switching options. It allows the device to be connected using Wi-Fi or 4G network which allows data transmissions over long distances. This system is accessible through Android phones. It is possible to generate reports using this system. The wiring method of this system is simple. This device is authenticated and makes sure that data is accessible only by authorized person.

3.3 Technology Stack

3.3.1. Software Requirements

Application development:

- Android studio
- Php

Android Studio:

Android Studio provides a unified environment to build apps for Android applications. The coding here is done using JAVA language .The role of android studio here is to develop the application which the client will be using on future. In our project all the basic layout of application is been designed by Android studio. Application components are the essential building blocks of an Android application. These components are loosely coupled by the application manifest file AndroidManifest.xml that describes each component of the application and how they interact. A login application is the screen asking for credentials to login into the application. You might have seen it when logging into facebook, twitter e.t.c

Navigation between pages is next priority.Moving from the login page to the next main display page and further travelling pages requires the navigation option and it is accomplished by Android studio .Alert dialog box Some times in application, if you wanted to ask the user about taking a decision between yes or no in response of any particular action taken by the user, by remaining in the same activity and without changing the screen, you can use Alert Dialog . Here alert dialog box is used while setting the values for OPS module ,Timer module, schedule timer and also to show that the device is in offline state.

PHP:

The **PHP Hypertext Pre-processor (PHP)** is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web-based software applications. In this project php is used to manage dynamic content, databases, session tracking.The main use of PHP in this application is to create more interactive application that looks catchy to user. Android studio only provides the basic layout where as PHP creates User interface even more interactive, it gives lot of options for creating web applications according to their choice of programming language. PHP is a server-side scripting language which here it is used for the better outlook.

3.3.2. Hardware Requirements

This project contains three release which includes monitoring, maintenance alerts and emergency shutdown. The hardware chips used for monitoring include LM324N, CEP50N06, microcontroller and EMMC. The software coding of this project uses embedded C. This project uses LTE, FTP and LAN networks for communications. This project gives alerts when any fault occurs in the monitoring system. This is possible using predictive analysis, PCR and OPS module. This hardware device displays alerts and we can generate report using it.

MICROCONTROLLER-MSP8266

Fig: 3.1 Microcontroller (MSP8266)

A microprocessor development board is a printed circuit board containing amicroprocessor and the minimal support logic needed for an engineer to become acquainted with the microprocessor on the board and to learn to program it. It also served users of the microprocessor as a method to prototype applications in products. Unlike a general-purpose system such as a home computer, usually a development board contains little or no hardware dedicated to a user interface. It will have some provision to accept and run a user-supplied program, such as downloading a program through a serial port to flash memory, or some form of programmable memory in a socket in earlier systems. A software development kit (SDK or "devkit") is typically a set of software development tools that allows the creation of applications for a certain software package, software framework, hardware platform, computer system, video game console, operating system, or similar development platform.

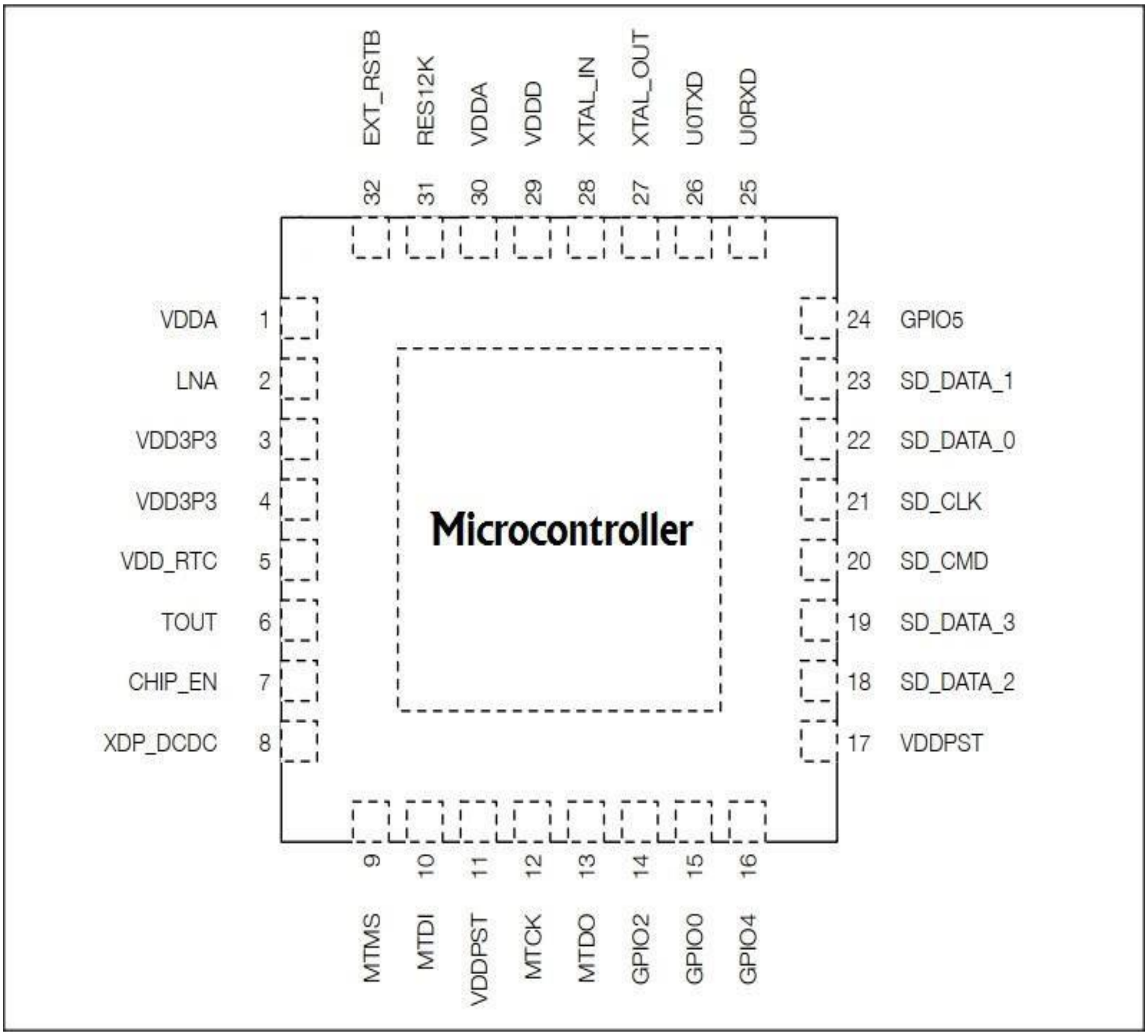


Fig: 3.2 Microcontroller pin diagram (MSP8266)

Table: 3.1 Microcontroller pin description (MSP8266)

Pin	Name	Type	Function
1	VDDA	P	Analog power 2.5V~3.6V
2	LNA	I/O	RF antenna interface chip output impedance=39+j6Ω It is suggested to retain π-type matching network to match the antenna.
3	VDD3P3	P	Amplifier power 2.5V~3.6V
4	VDD3P3	P	Amplifier power 2.5V~3.6V
5	VDD_RTC	P	NC (1.1V)
6	TOUT	I	ADC pin. It can be used to test the power supply voltage of VDD3P3 (pin 3&4) and the input power voltage of TOUT (pin 6). However, these two functions cannot be used simultaneously.
7	CHIP_EN	I	Chip Enable High: On, chip works properly. Low: off, small current consumed.
8	XPD_DCDC	I/O	Deep-Sleep wakeup (need to be connected to EXT_RSTB); GPIO16
9	MTMS	I/O	GPIO 14; HSp1_CLK
10	MTDI	I/O	GPIO 12; HSP1_MISO
11	VDDPST	P	Digital/IO power supply (1.8V~3.6v)
12	MTCK	I/O	GPIO 13; HSP1 _MOSI; UART_CTS
13	MTDO	I/O	GPIO 15; HSP1_MOSI; UART_CTS
14	GPIO2	I/O	UART TX during flash programming; GPIO2

15	GPIO0	I/O	GPIO); SPI_CS2
16	GPIO4	I/O	Digital/IO power supply (1.8V~3.6V)
17	VDDPST	P	Connect to SD_D2 (Series R:200Ω); SPIHD; HSPIHD; GPIO8
18	SDIO_DATA_2	I/O	Connect to SD_D2 (Series R:200Ω); SPIHD; HSPIHD; GPIO9
19	SDIO_DATA_3	I/O	Connect to SD_D2 (Series R:200Ω); SPIHD; HSPIHD; GPIO10
20	SDIO_CMD	I/O	Connect to SD_D2 (Series R:200Ω); SPIHD; HSPIHD; GPIO11
21	SDIO_CLK	I/O	Connect to SD_D2 (Series R:200Ω); SPIHD; HSPIHD; GPIO6
22	SDIO_DATA_0	I/O	Connect to SD_D2 (Series R:200Ω); SPIHD; HSPIHD; GPIO7
23	SDIO_DATA_1	I/O	Connect to SD_D1(Series R:200Ω)
24	GPIO5	I/O	GPIO5
25	U0RXD	I/O	UART Rx during flash programming; GPIO3
26	U0TXD	I/O	UART Rx during flash programming; GPIO1
27	XTAL_OUT	I/O	Connect to crystal oscillator output, can be used to provide BT clock input
28	XTAL_IN	I/O	Connect to crystal oscillator input
29	VDDD	P	Analog power 2.5V~3.6V
30	VDDA	P	Analog power 2.5V~3.6V
31	RES12K	I	Serial connection with a12kΩ resistor and connect to the ground
32	EXT_RSTB	I	External reset signal (Low voltage level: active)

EMMC FLASH MEMORY W25Q32JV

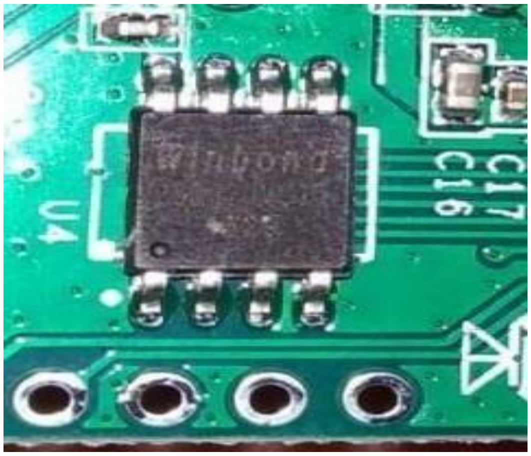


Fig: 3.3 EMMC Flash Memory (W25Q32JV)

The W25Q32JV (32M-bit) Serial Flash memory provides a storage solutionfor systems with limited space, pins and power. The 25Q series offers flexibility and performance well beyond ordinary Serial Flash devices. They are ideal for code shadowing to RAM, executing code directly from Dual/Quad SPI (XIP) and storing voice, text and data. The device operates on 2.7V to 3.6V power supply with current consumption as low as 1µA for power-down. The W25Q32JV array is organized into 16,384 programmable pages of 256-bytes each. Up to 256 bytes can be programmed at a time. Pages can be erased in groups of 16 (4KB sector erase), groups of 128(32KB block erase), groups of 256 (64KB block erase) or the entire chip (chip erase). The W25Q32JV has1,024 erasable sectors and 64 erasable blocks respectively. The small 4KB sectors allow for greater flexibility in applications that require data and parameter storage. The W25Q32JV supports the standard Serial Peripheral Interface (SPI), and a high performance Dual/Quad output as well as Dual/Quad I/O SPI: Serial Clock, Chip Select, Serial Data I/O0 (DI), I/O1 (DO), I/O2, and I/O3. SPI clock frequencies of up to 133MHz are supported allowing equivalent clock rates of 266MHz (133MHz x 2) for Dual I/O and 532MHz (133MHz x 4) for Quad I/O when using the Fast Read Dual/Quad I/O instructions. These transfer rates can outperform standard Asynchronous 8 and 16-bit Parallel Flash memories. Additionally, the device supports JEDEC standard manufacturer and device ID and SFDP Register, a 64-bit Unique Serial Number and three 256-bytes Security Registers.

CHAPTER 4

CHAPTER 4

SYSTEM DESIGN

4.1 ER Diagram

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. They mirror grammatical structure, with entities as nouns and relationships as verbs. The ER model becomes an abstract data model that defines a data or information structure which can be implemented in a database. This ER models shows a one to many relationship between the entities user and login since many authorized users can login to the application. However, only one app screen would be visible. Every user has attributes such as unique id, password and email. Each block acts as an entity with the attributes current, power, voltage and amount. Each user need to register if they are using this application for the first time.

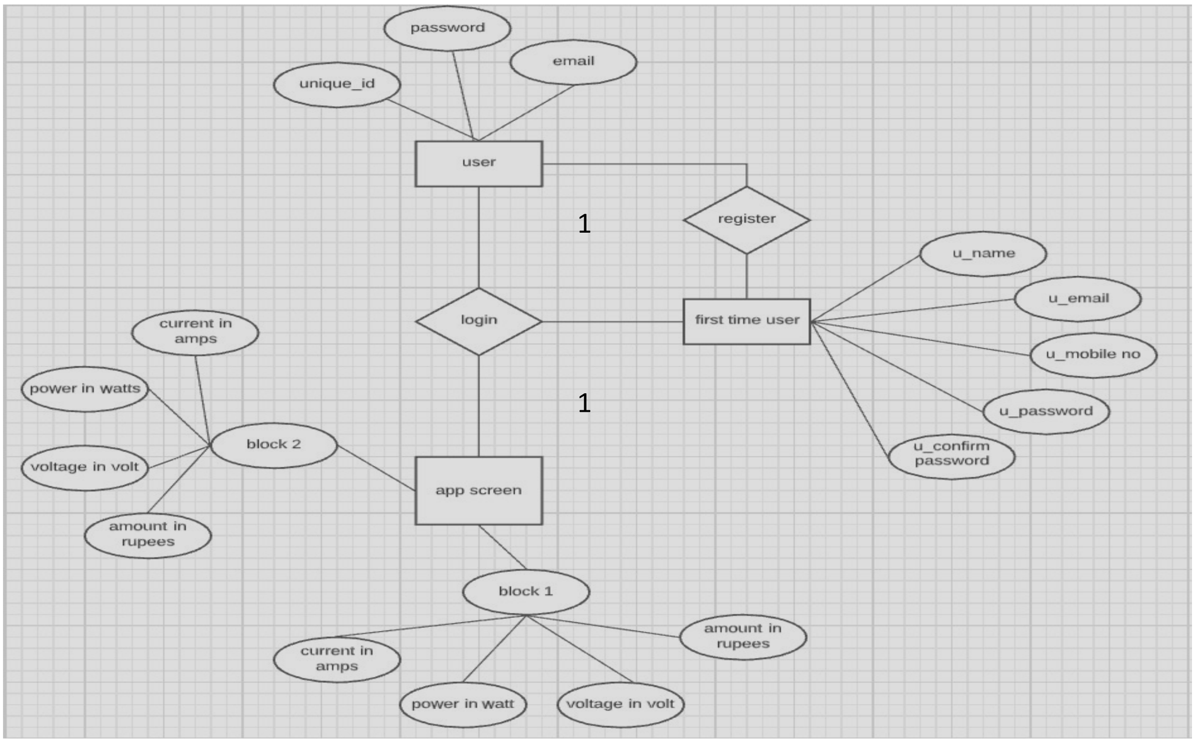


Fig 4.1 ER Diagram

4.2 Data Dictionary

A data dictionary is a centralized repository of metadata. Metadata is data about data. Some examples of what might be contained in an organization's data dictionary include: The names of fields contained in all of the organization's databases. In this case we are using date as the primary key and the other keys are Kw/h and cost.

Table 4.1 Data Dictionary

Name	Field Type	Size	Constraint
Date	INT	10	PRIMARY KEY
Kw/h	INT	10	NOT NULL
Cost	INT	10	NOT NULL

Table 4.2 Sample Table

Date	Kw/h	Cost
2020.12.24	0.50	100
2021.03.02	1.00	200

4.3 UML Diagrams

(i) Use Case Diagram

It only summarizes some of the relationships between use cases, actors, and systems. A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. Use case diagrams are typically developed in the early stage of development and people often apply use case modelling to specify the context of a system. This diagram is used to generate test cases. This diagram contains two actors that are the user and server. All the test cases show what are the actors involved in them. The test cases include login and authentication. It shows how this application works.

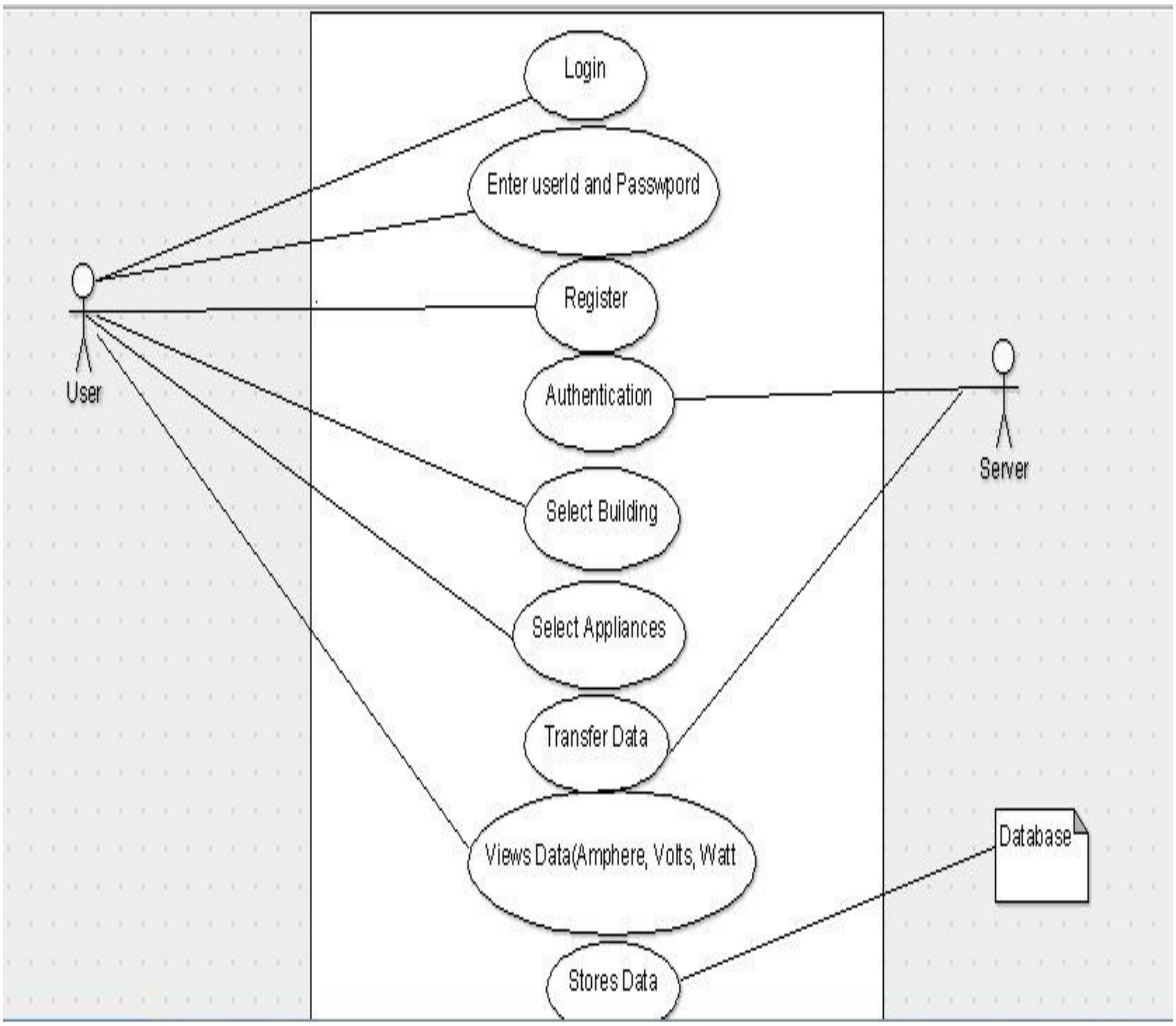


Fig 4.2 Use Case Diagram

(ii) Class Diagram

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code. In software engineering, a class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects. This class diagram contains the attributes in each class and the operations involved in them. For example this application contains a class called control circuit that contains attributes like on and off. It has operations like checkstatus(), powertracking() and monitor().

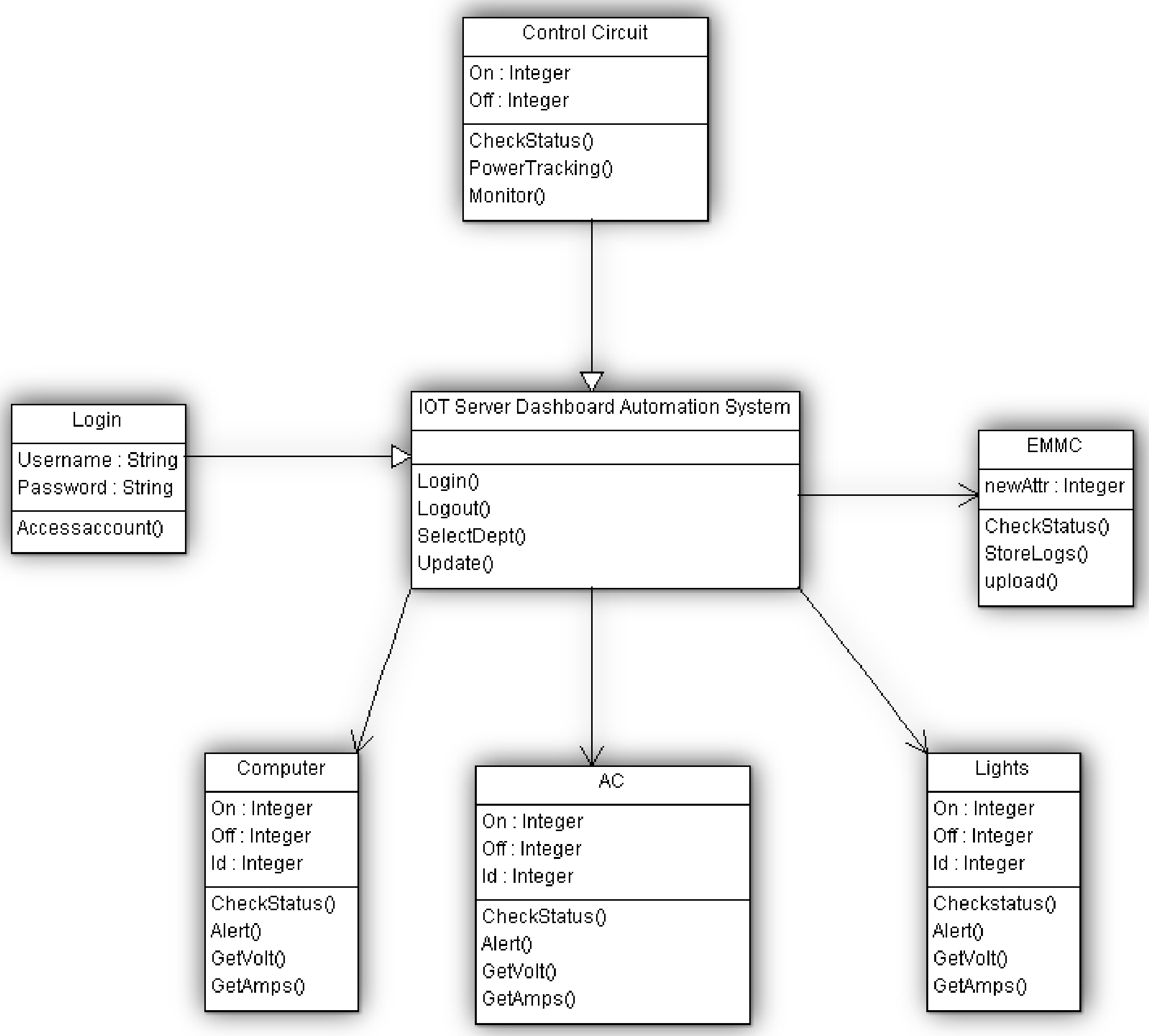


Fig 4.3 Class Diagram

(iii) Activity Diagram

An activity diagram is a behavioural diagram that depicts the behaviour of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organizational processes. This activity diagram shows the start process from login to how the application connects to the IOT server. It shows how the microcontroller then receives the data to view it.

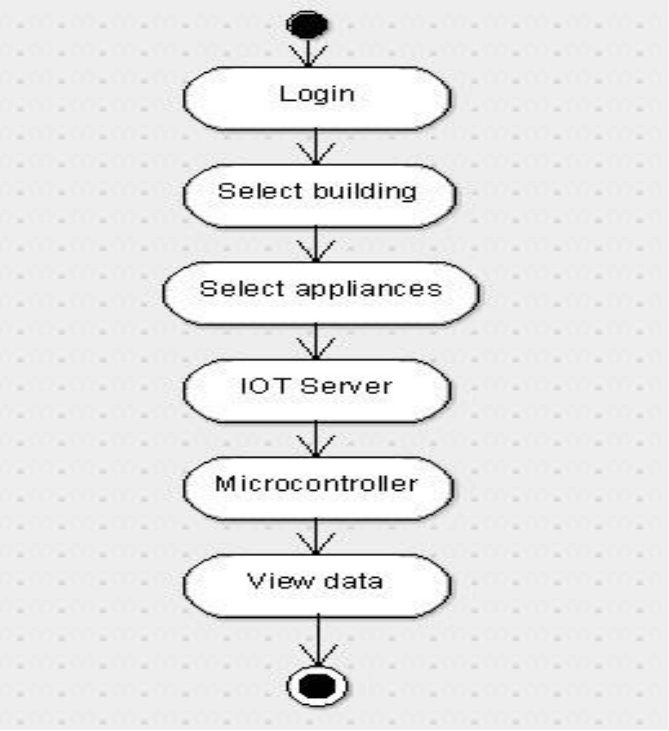


Fig 4.3 Activity Diagram

(iv)Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lines, different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur.

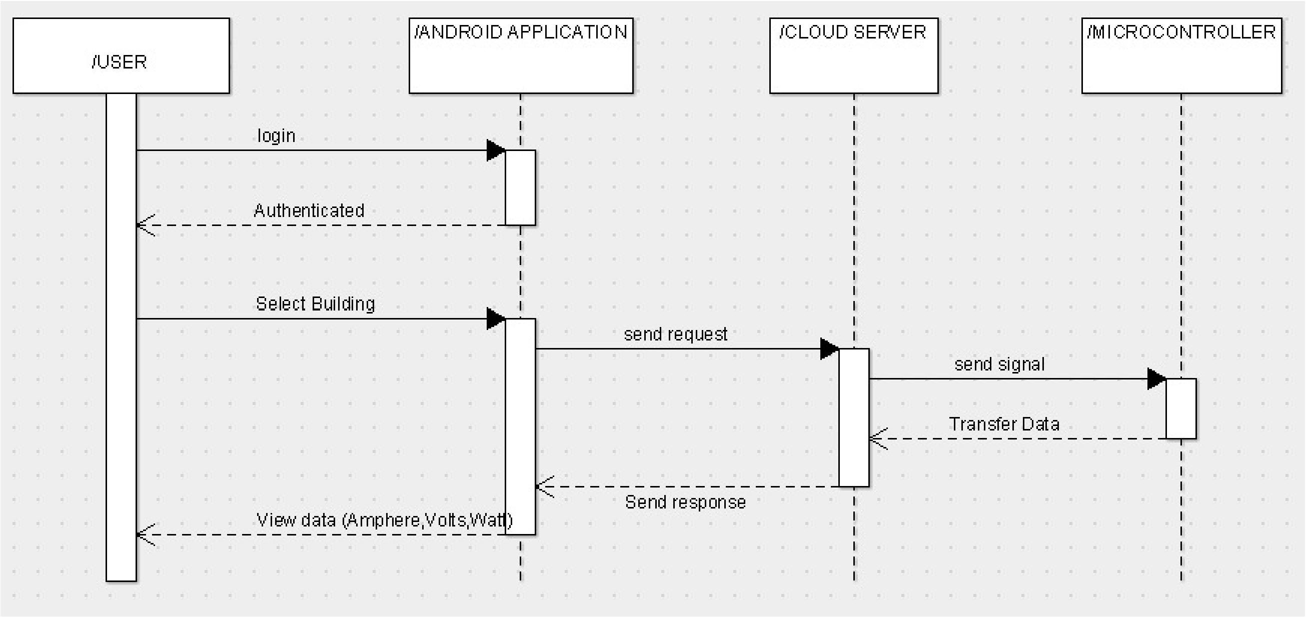


Fig 4.5 Sequence Diagram

CHAPTER 5

CHAPTER 5

5. SYSTEM ARCHITECTURE

5.1 Architecture Overview

The overall architecture of the energy monitoring application is mentioned in the figure 5.1 Architecture Diagram.

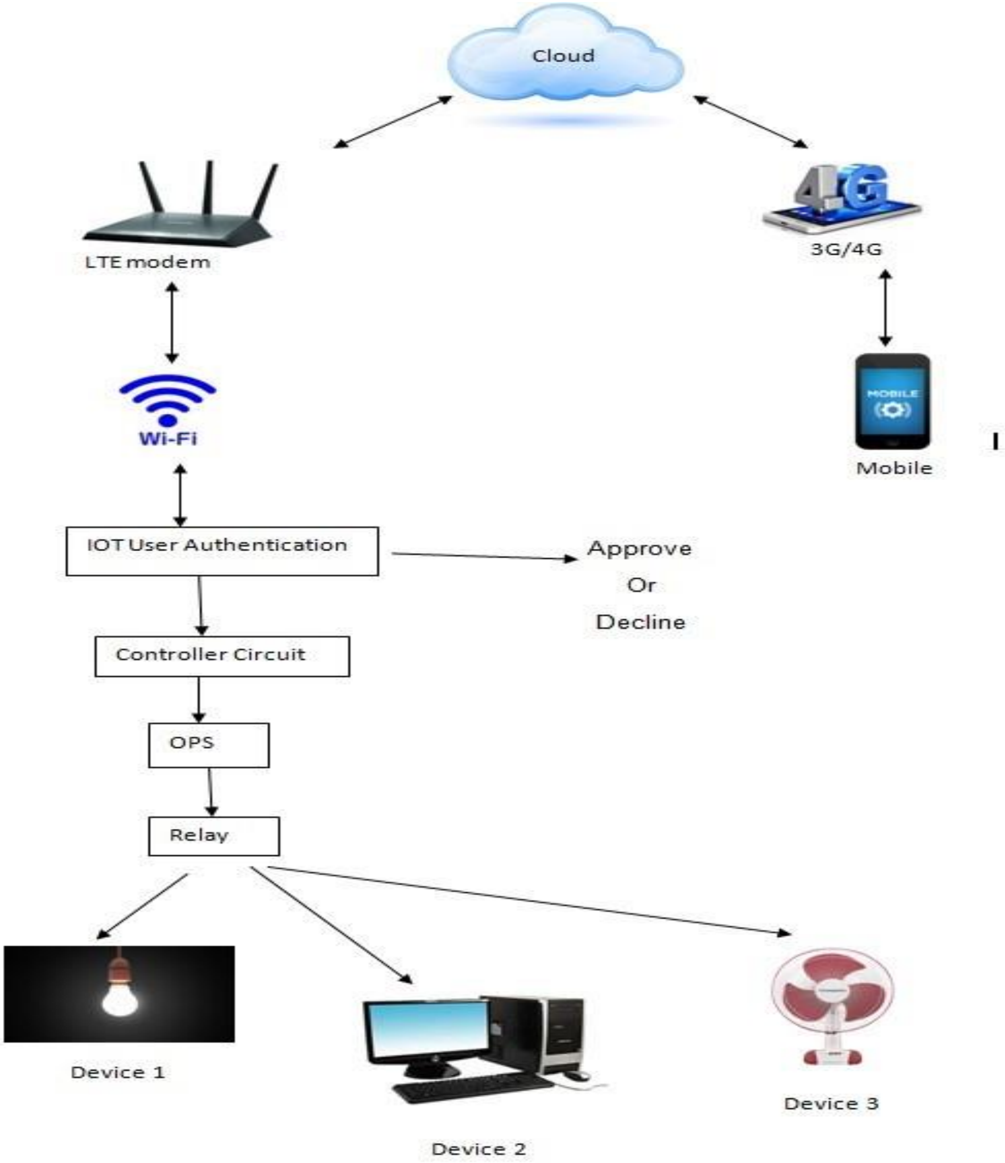


Fig: 5.1 Architecture Diagram

The concept behind this project is to monitor the total amount of current in ampere, voltage in volts and power in watts. The flow of execution starts with the mobile that is connected to a 4G network. Mobile consists of the application that has been developed for controlling of the other side part. Once the on operation is pressed in the application the cloud server receives the signal and passes it to the LTE modem that is connected to the hardware part. The LTE modem is further connected with Wi-Fi router which has the capability to connect all other devices like micro-controller, etc. Following to this there is user authentication to verify for valid and authorized users in order to avoid misusing the application. If once approved the flow will direct to controller circuit, OPS and relay .Finally the flow of power is given to the devices connected to it. For example fan, light, AC etc. System architecture finely describes the overall process that happens between the software application and the hardware devices. The overall system send and receive information less than a second that takes the performance of application one step higher than the other application status. An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. It is an important tool as it provides an overall view of the physical deployment of the software system and its evolution roadmap.

5.1 Circuit Design:

Fig 5.2 projects the real picture of the hardware part designed for the energy monitoring application. The devices can be connected using plug-points to view the amount of energy consumed. The image below meant to depict what the physical arrangement of the wires and the components we connect is called physical design or the layout. It shows the relative positions of all the elements and their connections to one another. From fig 5.2 separate devices like LTE modem , plug points ,microcontroller, capacitor, heat sink, self-recovery fuse, relay switch ,buck booster which is used to boost the power can be seen. A circuit diagram is a simplified representation of the components of an electrical circuit using either the images of the distinct parts or standard symbols. All the connection seen is drilled on a single wooden board for mobility purpose. The three separate plug points are the main points in which the devices should be connected and the respected output will be displayed on the side of application. The below proposed system can monitor to limited devices since it contain only three plug ins. This can even be developed for larger industries and any current consuming blocks.

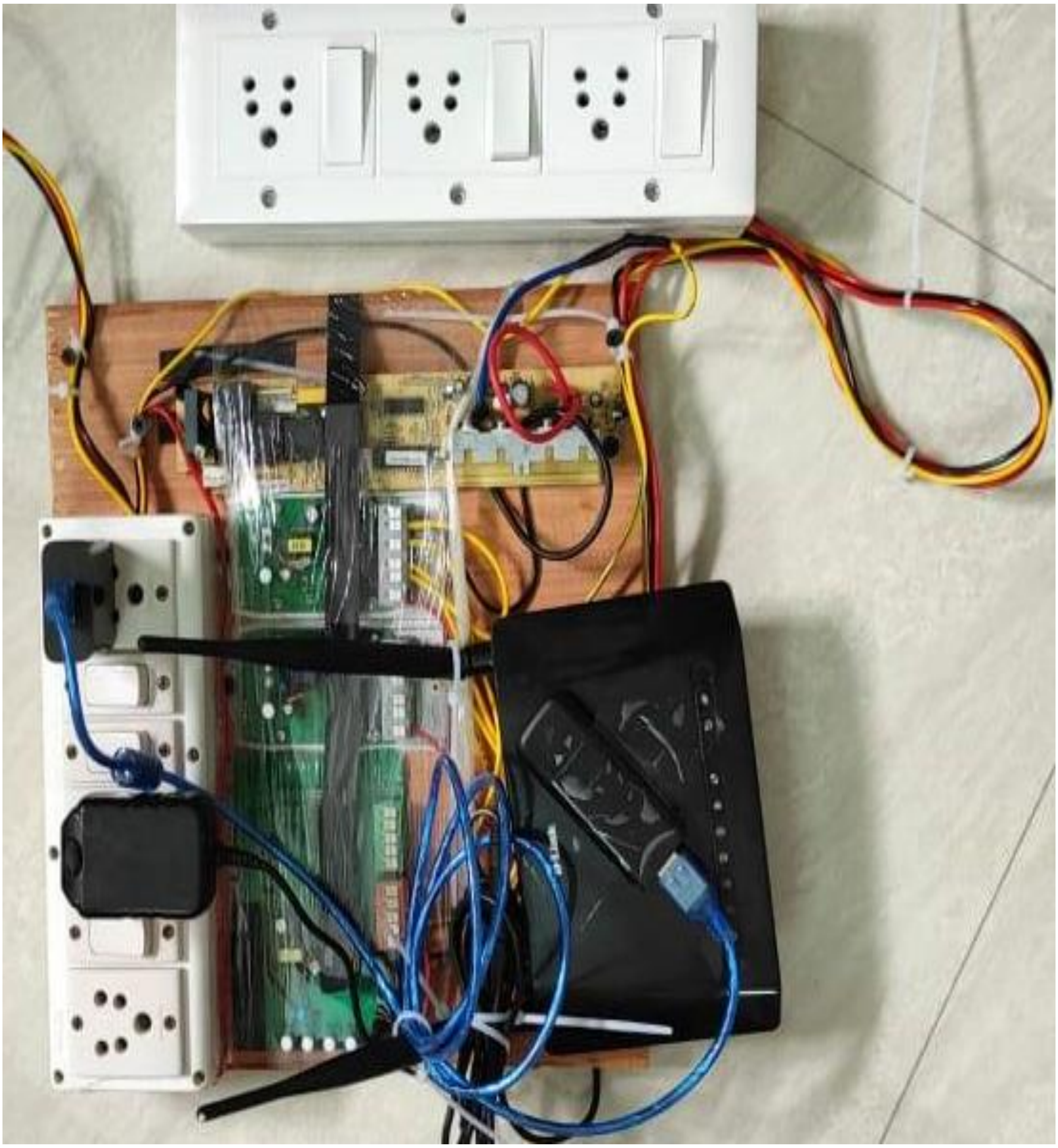


Fig 5.2 Hardware Circuit design

5.2. Module Design Specification

Module – 1 Power Consumption Recorder

PCR is Power Consumption Recorder. In PCR module entire power details are displayed in Kwh. This module is used to avoid data loss during power cut these entire records are synchronized to an EEPROM chip/SD card mounted to the microcontroller board. In PCR module Power consumed by each and every device and Overall industry power, Peak hour power consumption is monitored and Stored as data logs. The recorded data can be imported to android mobiles over Wi-Fi network/4G. The data will be saved to download folder in .csv file which can be opened using MS excel for detailed analysis and can be exported for Report generation.

The power consumption meter measures the apparent power in a 3-phase electrical installation and calculates its true power consumption remotely. The meter is designed to be installed in the general electrical distribution panel of the building and the measured data are transmitted wirelessly using an IEEE 802.15.4 network to an IOT gateway. Regarding the electrical setup, 3-phase electric power installations are a common practice for most public and private non-housing buildings, such as schools. Three separate single-phase supplies, with a fourth neutral connection, provide a constant voltage to power most common single-phase appliances. In order to measure the total power consumption of such an installation, it is necessary to independently measure the power consumption of each phase and add up the total consumption, as if the installation consisted of three separate lines.

Module – 2 Overload Production Service

This module is to protect the loads. Maximum current, maximum power, minimum power values can be set here to protect the loads. OPS module executes its function before operating the relays. So, the relays switch according to the set value. In OPS module, the safety limit for the running machineries can be set. E.g.: If volt limit is 250v Ac. If any sudden over voltage comes in power line this module sends immediate command to IoT main board relays to shutdown the plant likewise safety amps and safety power threshold can be set. The main function of overload Production service is to protect the device from damaging due to the supply of high power which may cause a short circuit and pushes the connected devices to damage state. There are four options available inside the OPS module, they are minimum power, maximum power, minimum voltage, maximum voltage . The user can set the values of above mentioned according to their device capability. The minimum power value which the user can set is 10amps, because below of that flow of power will not cause any damage to the devices. All these are set based on testing the project with various devices been connected to it.

Suppose if the power is set to 250 amps, and if the connected devices consumes more amount than the value that is already set it causes the device to shut-down immediately avoiding the upcoming damages that may cause a major accident to devices. The insertion of Overload Production Service can also consume cost which will be spending on the addition of new devices.

Module - 3 Real Time Statistics

Real-time data (RTD) is information that is delivered immediately after collection. There is no delay in the timeliness of the information provided. Real-time data is often used for navigation or tracking. Real-time energy data is a key metric for an energy management program to be successful. With live data being collected it is displayed on the screen of the application without any delay in time. Being able to monitor energy data in real time provides an understanding of energy consumption and performance across separate operations units. Real time statistics here provides the live data captured on the other side using the hardware.

The real value of the captured energy in various forms like current in ampere, voltage in volts and power in watts are displayed live with a clear screen in the application. If the value increases it also increases in the screen of application. Furthermore, by using real-time energy data and monitoring, energy managers can identify and manage energy spikes throughout the day. By reviewing energy trends and spikes over the course of the day, energy managers can determine best practice methods to reduce equipment energy consumption, as well as determine the most efficient scheduling to reduce energy consumption when spikes occur. Having real-time energy data improves awareness of energy usage and consumption across your entire organization. It will also improve employee engagement in corporate sustainability goals, encourages employees to reduce energy consumption and energy costs, and helps communicate large amounts of data to a variety of audiences.

Module – 4 Schedule

A schedule or a timetable, as a basic time-management tool, consists of a list of times at which possible tasks, events, or actions are intended to take place, or of a sequence of events in the chronological order in which such things are intended to take place. Here schedule module is used for the same purpose as stated above. We are able to schedule the start and stop timing of the entire building using schedule module. Long time scheduling is available using this methodology. Once the scheduling is done everything becomes automatic which can only be controlled by the app located in the mobile phone. We can set for a whole month and even for a year using this schedule module. The proposed algorithm works in three subsequent phases namely real-time monitoring (RTM), scheduling and real-time control (RTC).Proposed system is two way communication system hence communication takes place on both the sides and we can send and receive information.

Module – 5 Timer

A timer is a device that measures time, especially one that is part of a machine and causes it to start or stop working at specific times. Electronic timers automatically switch on the lights when it gets dark. Similarly the timer module here is used to monitor the amount of energy consumption for a specific time. Clicking on the Start button will turn the Timer On and by leaving it as the same one can turn off when the job is completely done. A good example of this can be a printer for which the energy consumption should be monitor so that we could predict the amount of energy it consumes individually. They are used to keep a record of time for different events occurring in the system. Secondly, it can be used to repeat or initiate an action after/at a known period of time. This feature is very commonly used in this application.

Timer is more or less equal to countdown timer where specific time can be fixed once, to get the amount of power consumed by a single device. This feature could be more useful in cases where a specific device should be monitored and find the amount in rupees .Timer is used to monitor energy in short period and that’s makes different between schedule and timer.

This is the main use of timer module in this system. It’s an additional module which is helpful to monitor the energy consumption.

Module – 6 Loop Timer

This Loop timer module is used to loop the time at specific time intervals.If something runs in a loop, or is on a loop, it runs continuously, so that the same things are repeated again and again: The tape ran in a continuous loop, repeating the same songs over and over. Similarly this module helps in setting a loop when something wants to run in some specific order again and again. Sometimes, we hope our device can work repeatedly in a shorter cycle (in minutes or hours), such as turning on every 30 minutes. In this case, a loop timer does what we desire. The main function here is to operate the device at already allocated time limits.

Module – 7 History Record and Download

This module is used to display the content that is stored during the runtime of energy monitoring .In history record, user can view the amount of energy consumed in graph manner, so that there could be high vision of the stored data. The data stored is displayed in month wise which is created for user readability purpose. Additional advantage here is user can view the amount in rupees that is generated based on the energy that is consumed by the devices. By storing your data online you are reducing the burden of your hard disk, which means you are eventually saving disk space. World Wide accessibility: This is the main advantage of online data storage. You can access your data anywhere in the world.

There is another option called download where user can download the stored data of the energy monitoring device. The file will be downloaded in local directory once by clicking the download option that is readily available. The file is downloaded in **Excel** format with an extension of **.CSV,** which is helpful for future purpose or any further references related to consumption of power.

CHAPTER 6

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 Client Side Coding

```
package com.myapplication;

import android.app.Activity;
import android.graphics.Color;
import android.os.Bundle;
import android.view.View;

import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;


public class MainActivity extends Activity  {
   Button b1,b2;
   EditText ed1,ed2;

   TextView tx1;
   int counter = 3;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      b1 = (Button)findViewById(R.id.button);
      ed1 = (EditText)findViewById(R.id.editText);
      ed2 = (EditText)findViewById(R.id.editText2);

      b2 = (Button)findViewById(R.id.button2);
      tx1 = (TextView)findViewById(R.id.textView3);
      tx1.setVisibility(View.GONE);

      b1.setOnClickListener(new View.OnClickListener() {
         @Override         public void onClick(View v) {
            if(ed1.getText().toString().equals("admin") &&
               ed2.getText().toString().equals("admin")) {
                  Toast.makeText(getApplicationContext(),
                     "Redirecting...",Toast.LENGTH_SHORT).show();
               }else{
                  Toast.makeText(getApplicationContext(), "Wrong
                     Credentials",Toast.LENGTH_SHORT).show();

                  tx1.setVisibility(View.VISIBLE);
                  tx1.setBackgroundColor(Color.RED);
                  counter--;
            tx1.setText(Integer.toString(counter));

            if (counter == 0) {
              b1.setEnabled(false);
            }
          }
      }
    });

    b2.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        finish();
      }     });
}
```

```
// IOT dashboard settings

#definePARAM8_SIZE18// Number of param bytes (SetOption)
typedefunion{ // Restricted by MISRA-C Rule 18.4 but so useful...

uint32_tdata;   // Allow bit manipulation using

SetOption struct{ // SetOption0 .. SetOption31

uint32_tsave_state :1; // bit 0  - SetOption0 - Save power state and
use after restart


uint32_tbutton_restrict :1;      // bit 1  - SetOption1 - Control
button multipress


uint32_tvalue_units :1;          // bit 2  - SetOption2 - Add units to
JSON status messages


uint32_tmqtt_enabled :1;         // bit 3  - SetOption3 - Control

MQTT uint32_tmqtt_response :1;         // bit 4  - SetOption4 -

Switch between
MQTT RESULT or COMMAND


uint32_tmqtt_power_retain :1;          // bit 5  - CMND_POWERRETAIN

uint32_tmqtt_button_retain :1;         // bit 6  -

CMND_BUTTONRETAIN uint32_tmqtt_switch_retain :1;     // bit 7  -

CMND_SWITCHRETAIN uint32_ttemperature_conversion :1;   // bit 8

- SetOption8 - Switch
between Celsius or Fahrenheit
```

uint32_tmqtt_sensor_retain :1; // bit 9 - CMND_SENSORRETAIN

uint32_tmqtt_offline :1; // bit 10 - SetOption10 - Control

MQTT
LWT message format

uint32_tbutton_swap :1; // bit 11 (v5.1.6) - SetOption11 - Swap button single and double press functionality

uint32_tstop_flash_rotate :1; // bit 12 (v5.2.0) - SetOption12 - Switch between dynamic or fixed slot flash save location

uint32_tbutton_single :1; // bit 13 (v5.4.0) - SetOption13 -

Support only single press to speed up button press recognition

uint32_tinterlock :1; // bit 14 (v5.6.0) - CMND_INTERLOCK

uint32_tpwm_control :1; // bit 15 (v5.8.1) - SetOption15 -

Switch between commands PWM or

COLOR/DIMMER/CT/CHANNEL

uint32_tws_clock_reverse :1; // bit 16 (v5.8.1) - SetOption16 -

Switch between clockwise or counter-clockwise

uint32_tdecimal_text :1; // bit 17 (v5.8.1) - SetOption17 -

Switch between decimal or hexadecimal output

uint32_tlight_signal :1; // bit 18 (v5.10.0c) - SetOption18 -

Pair light signal with CO2 sensor

uint32_thass_discovery :1; // bit 19 (v5.11.1a) -

SetOption19 - Control Home Assistantautomatic discovery (See

SetOption59)

uint32_tnot_power_linked :1;// bit 20 (v5.11.1f) - SetOption20 -

Control power in relation to Dimmer/Color/Ct changes

uint32_tno_power_on_check :1; // bit 21 (v5.11.1i) -

SetOption21 - Show voltage even if powered off

uint32_tmqtt_serial :1; // bit 22 (v5.12.0f) -

CMND_SERIALSEND and CMND_SERIALLOG

uint32_tmqtt_serial_raw :1; // bit 23 (v6.1.1c) - CMND_SERIALSEND3

uint32_tpressure_conversion :1; // bit 24 (v6.3.0.2) - SetOption24 - Switch

```
between hPa or mmHg pressure unit
uint32_tknx_enabled :1;         // bit 25 (v5.12.0l)  - CMND_KNX_ENABLED
uint32_tdevice_index_enable :1;        // bit 26 (v5.13.1a)  - SetOption26 - Switch
between POWER or POWER1
uint32_tknx_enable_enhancement :1;           // bit 27 (v5.14.0a)  -
CMND_KNX_ENHANCED
uint32_thass_light :1; // bit 30 (v6.0.0b)        - SetOption30 - Enforce
HAssautodiscovery as light
uint32_tglobal_state :1;         // bit 31 (v6.1.0)         - SetOption31 - Control link led
blinking
};
} SysBitfield;
typedefunion{  // Restricted by MISRA-C Rule 18.4 but so useful...
uint32_tdata;   // Allow bit manipulation using SetOption
struct{  // SetOption50 .. SetOption81
uint32_ttimers_enable :1;       // bit 0 (v6.1.1b)          - CMND_TIMERS
uint32_t user_esp8285_enable :1;       // bit 1 (v6.1.1.14)  - SetOption51 - Enable
ESP8285 user GPIO's
uint32_ttime_append_timezone :1;    // bit 2 (v6.2.1.2)        - SetOption52 – Append
timezone to JSON time
uint32_tgui_hostname_ip :1;  // bit 3 (v6.2.1.20)  - SetOption53 – Show
hostanme and IP address in GUI main menu
uint32_t tuya_apply_o20 :1;   // bit 4 (v6.3.0.4)        - SetOption54 – Apply
SetOption20 settings to Tuya device
uint32_tmdns_enabled :1;       // bit 5 (v6.4.1.4)        - SetOption55 – Control
mDNS service
uint32_tuse_wifi_scan :1;       // bit 6 (v6.3.0.10)  - SetOption56 - Scan wifi
network at restart for configured AP's
uint32_tuse_wifi_rescan :1;    // bit 7 (v6.3.0.10)  - SetOption57 - Scan wifi
network every 44 minutes for configured AP's
```

```
uint32_treceive_raw :1;         // bit 8 (v6.3.0.11)  - SetOption58 - Add IR Raw
data to JSON message

uint32_thass_tele_on_power :1;         // bit 9 (v6.3.0.13)  - SetOption59 – Send
tele/%topic%/STATE in addition to stat/%topic%/RESULT
uint32_tdata;   // Allow bit manipulation using SetOption

struct{  // SetOption50 .. SetOption81

uint32_ttimers_enable :1;       // bit 0 (v6.1.1b)         - CMND_TIMERS

uint32_t user_esp8285_enable :1;      // bit 1 (v6.1.1.14)  - SetOption51 - Enable
ESP8285 user GPIO's

uint32_ttime_append_timezone :1;    // bit 2 (v6.2.1.2)        - SetOption52 – Append
timezone to JSON time

uint32_tgui_hostname_ip :1;  // bit 3 (v6.2.1.20)  - SetOption53 – Show
hostanme and IP address in GUI main menu

uint32_t tuya_apply_o20 :1;   // bit 4 (v6.3.0.4)        - SetOption54 – Apply
SetOption20 settings to Tuya device

uint32_tmdns_enabled :1;       // bit 5 (v6.4.1.4)        - SetOption55 – Control
mDNS service

uint32_tuse_wifi_scan :1;       // bit 6 (v6.3.0.10)  - SetOption56 - Scan wifi
network at restart for configured AP's
uint32_t user_esp8285_enable :1;      // bit 1 (v6.1.1.14)  - SetOption51 - Enable
ESP8285 user GPIO's

uint32_ttime_append_timezone :1;    // bit 2 (v6.2.1.2)        - SetOption52 – Append
timezone to JSON time

uint32_tgui_hostname_ip :1;  // bit 3 (v6.2.1.20)  - SetOption53 – Show
hostanme and IP address in GUI main menu
uint32_ttime_append_timezone :1;    // bit 2 (v6.2.1.2)        - SetOption52 – Append
timezone to JSON time
uint32_ttime_append_timezone :1;    // bit 2 (v6.2.1.2)        - SetOption52 – Append
timezone to JSON time
```

```
uint16_tweek :3;        // bits 1 - 3      = 0=Last week of the month, 1=First,
2=Second, 3=Third, 4=Fourth
uint16_tmonth :4;       // bits 4 - 7      = 1=Jan, 2=Feb, ... 12=Dec
uint16_tdow :3;         // bits 8 - 10  = day of week, 1=Sun, 2=Mon, ... 7=Sat
uint16_thour :5;        // bits 11 - 15 = 0-23
};
} TimeRule;
typedefunion {
uint32_t data;
struct {
uint32_ttime :11;       // bits 0 - 10 = minutes in a day
uint32_twindow :4;      // bits 11 - 14 = minutes random window
uint32_trepeat :1;      // bit 15
uint32_tdays :7;        // bits 16 - 22 = week day mask
uint32_tdevice :4;      // bits 23 - 26 = 16 devices
uint32_tpower :2;       // bits 27 - 28 = 4 power states - Off, On, Toggle,
Blink or Rule
uint32_tmode :2;        // bits 29 - 30 = timer modes - 0 = Scheduler, 1 =
Sunrise, 2 = Sunset
uint32_tarm :1;         // bit 31
};
} Timer;
typedefunion {
uint16_t data;
struct {
uint16_tpinmode :3;     // Pin mode (1 through 6)
uint16_tpullup :1;      // Enable internal weak pull-up resistor
uint16_tsaved_state :1;         // Save output state, if used.
```

```
uint16_tint_report_mode :2;   // Interrupt reporting mode 0 = immediate
telemetry & event, 1 = immediate event only, 2 = immediate telemetry only
uint16_tint_report_defer :4;   // Number of interrupts to ignore until reporting
(default 0, max 15)
uint16_tint_count_en :1;        // Enable interrupt counter for this pin
uint16_tint_retain_flag :1;     // Report if interrupt occured for pin in next
teleperiod
uint16_t spare13 :1;
uint16_t spare14 :1;
uint16_t spare15 :1;
};
} Mcp230xxCfg;
typedefunion {
uint8_t data;
struct {
uint8_t spare0 :1;
uint8_t spare1 :1;
uint8_t spare2 :1;
uint8_t spare3 :1;
uint8_t spare4 :1;
uint8_t spare5 :1;
uint8_t spare6 :1;
uint8_t mhz19b_abc_disable :1;        // Disable ABC (Automatic Baseline Correction for
MHZ19(B) (0 = Enabled (default), 1 = Disabled with Sensor15
command)
};
} SensorCfg1;
/*
```

```
struct SYSCFG {
unsigned long cfg_holder;      // 000 Pre v6 header
unsigned long save_flag;       // 004
unsigned long version;         // 008
unsigned long bootcount;       // 00C
*/
struct SYSCFG {
uint16_tcfg_holder;    // 000 v6 header
 // Over Load Protection Service
 Settings.flag2.current_resolution = 3;
 //  Settings.flag2.voltage_resolution = 0;
 //  Settings.flag2.wattage_resolution = 0;
 Settings.flag2.energy_resolution = ENERGY_RESOLUTION;
 Settings.param[P_MAX_POWER_RETRY] = MAX_POWER_RETRY;
 Settings.energy_power_delta = DEFAULT_POWER_DELTA;
 Settings.energy_power_calibration = HLW_PREF_PULSE;
 Settings.energy_voltage_calibration = HLW_UREF_PULSE;
 Settings.energy_current_calibration = HLW_IREF_PULSE;
 //  Settings.energy_kWhtoday = 0;
 //  Settings.energy_kWhyesterday = 0;
 //  Settings.energy_kWhtdy = 0;
 //  Settings.energy_min_power = 0;
 //  Settings.energy_max_power = 0;
 //  Settings.energy_min_voltage = 0;
 //  Settings.energy_max_voltage = 0;
 //  Settings.energy_min_current = 0;
 //  Settings.energy_max_current = 0;
 //  Settings.energy_max_power_limit = 0;     // MaxPowerLimit
 Settings.energy_max_power_limit_hold = MAX_POWER_HOLD;
 Settings.energy_max_power_limit_window = MAX_POWER_WINDOW;
```

```
//  Settings.energy_max_power_safe_limit = 0;          //
MaxSafePowerLimit
Settings.energy_max_power_safe_limit_hold = SAFE_POWER_HOLD;
Settings.energy_max_power_safe_limit_window =
SAFE_POWER_WINDOW;
//  Settings.energy_max_energy = 0;  // MaxEnergy
```

6.2 Server Side Coding

// Performing API Authentication

```
// Route change 0.0.0.0 mask 0.0.0.0 10.10.7.1
{
    "deviceid":"Factory xxxxx",
    "apikey":"xxxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "accept":"post"
}
http://10.10.7.1/ap
{
    "version": 2019,
    "ssid": [YOUR NETWORK SSID],
    "password": [YOUR NETWORK PASSWORD],
    "serverName": [IP OF YOUR SERVER],
    "port": [PORT OF YOUR SERVER]
}

{
     "error":          0,
     "reason": "ok",
     "IP": [YOUR WEBSOCKET SERVER IP], "port":
     [YOUR WEBSOCKET SERVER PORT
{
    "error" : 0,
    "date" : [DATE IN ISO FORMAT],
    "deviceid" : [ACTUAL DEVICE ID],
    "apikey" : "111111111-1111-1111-1111-111111111111"
}
```

```
{action: 'update', value: {switch :state?'on':'off'}
```

```
{action: 'update', value: {timers :d.timers}
```

```
{

enabled : true,

    type : 'once' OR 'repeat',at :
    time,

    do :

   {

     switch : 'on' OR 'off'

    }

  }
```

```
#include < ESP8266WiFi.h >
#include <WiFiClient.h>
#include <RelayCKT.h>
#include <ESP8266WebServer.h>
#include <ESP8266mDNS.h>
MDNSRespondermdns; // Microcontroller coding via opensource https://xod.io/ide/
// Replace with your network credentials
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";
ESP8266WebServer server(80);
String web_on_html = "<h1>IOTSERVER switch is ON</h1><p><a
href=\"on\"><button>ON</button></a>&nbsp;<a
href=\"off\"><button>OFF</button></a></p>";
```

```
    String web_off_html = "<h1>IOTSERVER switch is OFF</h1><p><a
    href=\"on\"><button>ON</button></a>&nbsp;<a
    href=\"off\"><button>OFF</button></a></p>";

    int gpio_13_led = 13;

    int gpio_12_relay = 12;

    void setup(void)

    {

    //  First Time Initialisation

    pinMode(gpio_13_led, OUTPUT);

    digitalWrite(gpio_13_led, HIGH);

    pinMode(gpio_12_relay, OUTPUT);

    digitalWrite(gpio_12_relay, HIGH);

    Serial.begin(115200);


WiFi.begin(ssid, password);Serial.println("Connecting to wifi.."
);

// Wait for connection

while (WiFi.status() != WL_CONNECTED)
{

digitalWrite(gpio_13_led, LOW);

delay(500);

Serial.print(".");

Serial.println(WiFi.localIP());

Serial.println(WiFi.status());

digitalWrite(gpio_13_led, HIGH);

delay(500);

}

Serial.println("");

Serial.print("Connected to ");

Serial.println(ssid);
Serial.print("IP address: ");

Serial.println(WiFi.localIP());

if (mdns.begin("esp8266", WiFi.localIP()))

{

Serial.println("MDNS responder started");

}

server.on("/", []()

{

if(digitalRead(gpio_12_relay)==HIGH)

{

server.send(200, "text/html", web_on_html);

}
else
{
server.send(200, "text/html", web_off_html);

}

};

server.on("/on", []()
{
server.send(200, "text/html", web_on_html);

digitalWrite(gpio_13_led, LOW);

digitalWrite(gpio_12_relay, HIGH);

delay(1000);

};

server.on("/off", []()
{
server.send(200, "text/html", web_off_html);

digitalWrite(gpio_13_led, HIGH);

digitalWrite(gpio_12_relay, LOW);

delay(1000);

};

server.begin();

Serial.println("Server ready..");

}

void loop(void)
{
server.handleClient();

}
```

CHAPTER 7

CHAPTER 7

TESTING

It is possible to include many devices in our project but we have used four devices as sample. As we can see from table 7.1, we can use devices like desktop PC, fan, AC and jet motor. There is a threshold range for every device and we can set this value. If the device voltage exceeds this limit then the device will shut down immediately. Our project acts in such a way that even if the power exceeds by 1W, the device will shutdown automatically.

Table 7.1 Testing for project

S.NO	DEVICE/ FACTORS	THRESHOLD RANGE	EXISTING	PROPOSED OPS CIRCUIT
1.	Dektop PC	Upto 230 V	230+/-20	230+/-1V
2.	Device 1/Fan	Upto 150 W	150+/-30	150+/-2W
3.	Device 2/ AC	Min 1500W	1500+/-100	1500+/-3W
4.	Jet Motor	Upto 15A	15+/-2A	15+/-0.2A

Module 1: Power Consumption Recorder

Table 7.2 Testcase 1

Test Case	Expected Result	Actual Result	Pass/Fail	Remarks
Energy consumption displayed in the screen	Power in watts, current in ampere, voltage in volts	Power in watts, current in ampere, voltage in volts	Pass	OK

Module 2: Overload Production Service

Table: 7.3 Testcase 2

Test Case	Expected Result	Actual Result	Pass/Fail	Remarks
Automatic shutdown of system when the maximum set value exceeds	System shutdown automatically	System shutdown automatically	Pass	OK

Module 3: Real Time Statistics

Table: 7.4 Testcase 3

Test Case	Expected Result	Actual Result	Pass/Fail	Remarks
Display output with no delay in time	Immediate output respected to input value	Immediate output respected to input value	Pass	OK

Module 4: Schedule

Table: 7.5 Testcase 4

Test Case	Expected Result	Actual Result	Pass/Fail	Remarks
Automatic On/Off at the scheduled time	On/Off of devices according to timing scheduled	On/Off of devices according to timing scheduled	Pass	OK

Module 5: Loop Timer

Table: 7.6 Testcase 5

Test Case	Expected Result	Actual Result	Pass/Fail	Remarks
Device Loops on (on/off) the specified time	Device runs for an hour and rests for 10 mins and repeat the same	Device runs for an hour and rests for 10 mins and repeat the same	Pass	OK

CHAPTER 8

CHAPTER 8

CONCLUSION

8. CONCLUSION

This proposed system thus helps us to easily monitor the energy by a single application which becomes more convenient in future world of digital India and all over the world. It reduces Energy Consumption and also detects abnormality in consumption. It is a better way to Optimize Cost and act as a Get Real-time Energy Insights. A way ahead it reduces manpower which also does Reduce Operational Costs.Helps in Boosting the Facility of Performance Most importantly it predicts Future Energy Needs.The main objectives of our proposed system are monitoring of current consumption easy, monitoring voltage and power consumptions,analysis of real time data, storing and access data from anywhere at any time, to view which building consumes most power so that we can save energy and money that you spend on paying current bills.

FUTURE ENHANCEMENTS:

Implementing of automated timing system which helps in turning the whole system on and off at the allocated time. Adding camera into the system, as the hardware part already has the capacity to support the camera in it. This future enhancement allows us to view the capturing video live from the same mobile application. Capability to view energy monitoring of various buildings located in various cities through the same mobile application.

APPENDICES

A.1. SAMPLE SCREENS

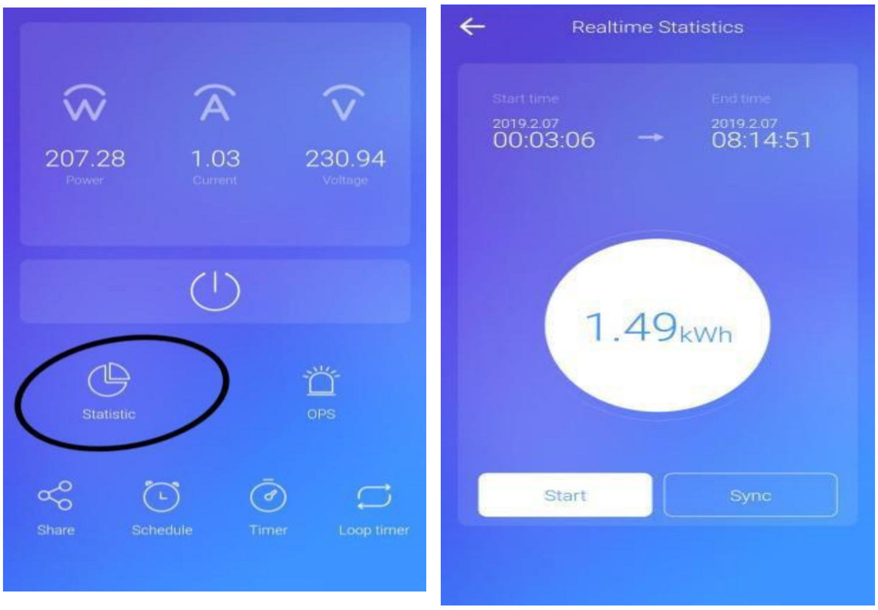


Fig A1.1 Statistics

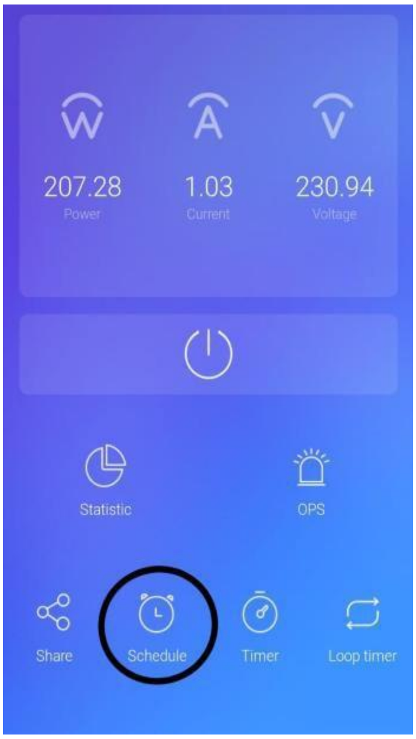
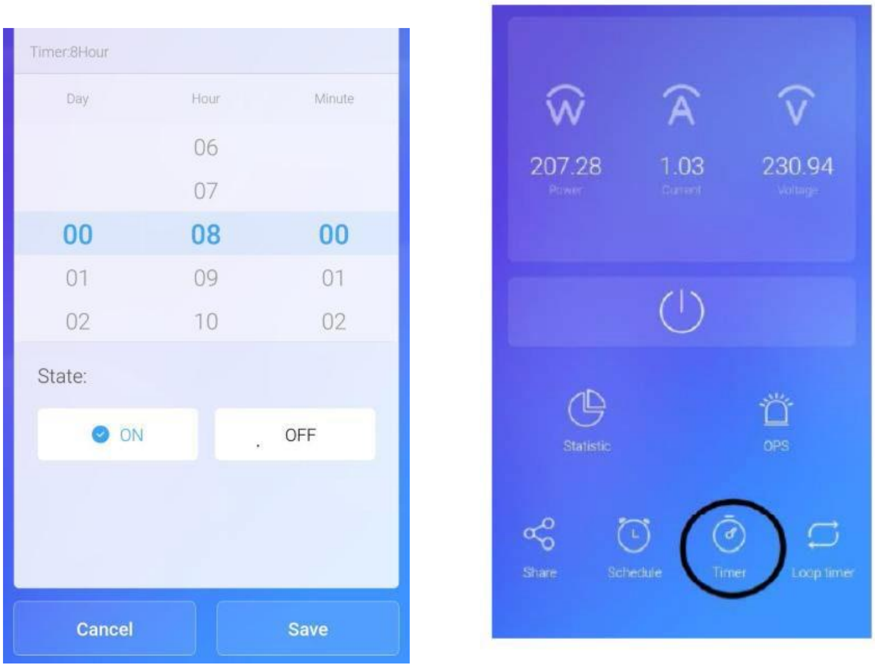


Fig A1.2 Schedule



Timer:8Hour

Day	Hour	Minute
	06	
	07	
00	08	00
01	09	01
02	10	02

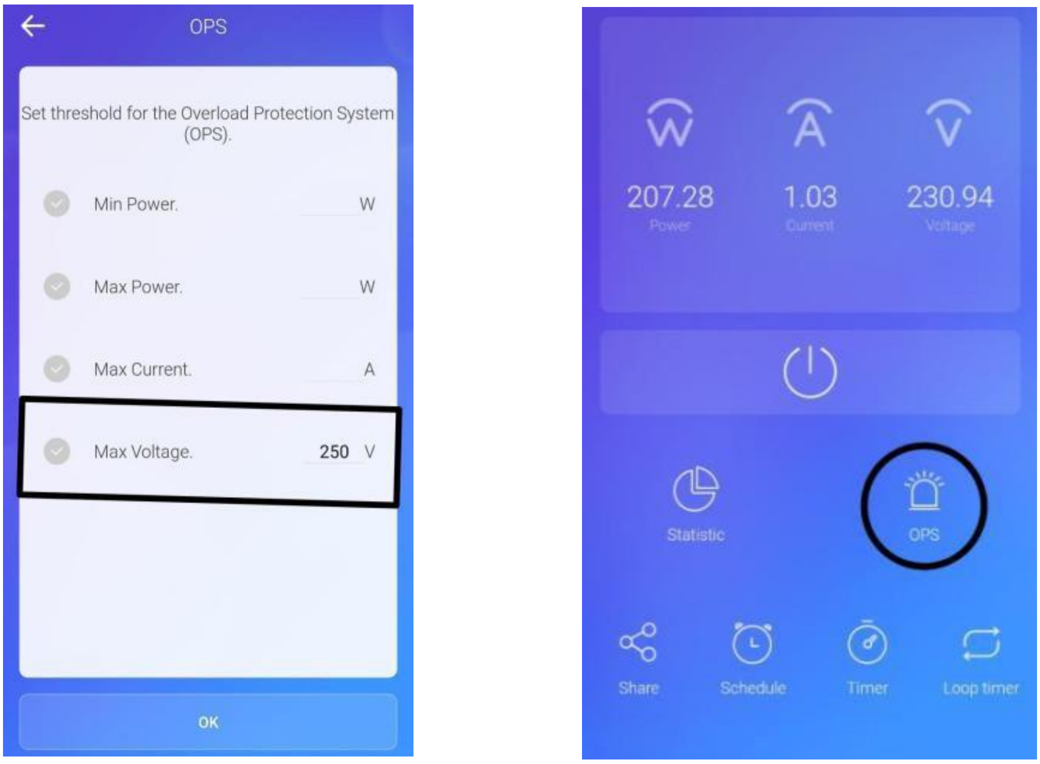
State: ● ON ○ OFF

207.28 Power

1.03 Current

230.94 Voltage

Fig A1.3 Timer



OPS

Set threshold for the Overload Protection System (OPS).

Min Power. W

Max Power. W

Max Current. A

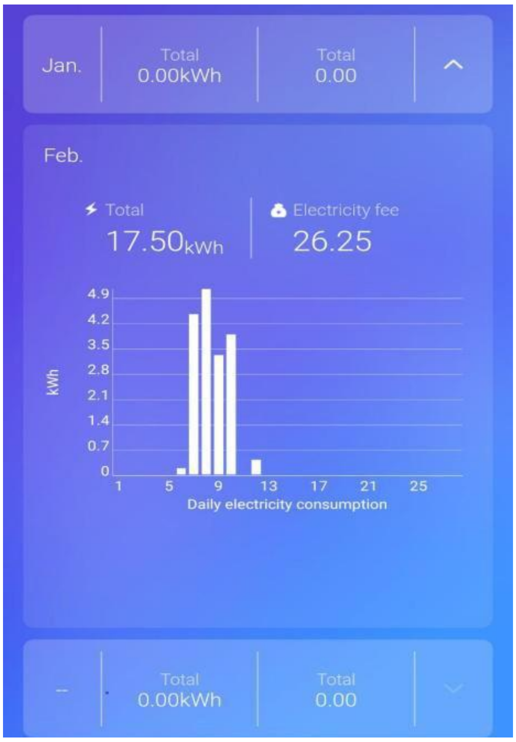
Max Voltage. 250 V

207.28 Power

1.03 Current

230.94 Voltage

Fig A1.4 OPS-Overload Protection Service



Jan. Total 0.00kWh Total 0.00

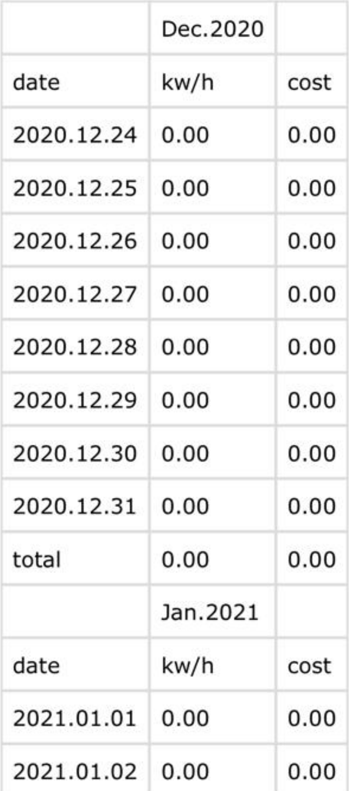
Feb.

Total 17.50kWh

Electricity fee 26.25

-- Total 0.00kWh Total 0.00

Fig A1.5 PCR-Power Consumption Recorder



	Dec.2020	
date	kw/h	cost
2020.12.24	0.00	0.00
2020.12.25	0.00	0.00
2020.12.26	0.00	0.00
2020.12.27	0.00	0.00
2020.12.28	0.00	0.00
2020.12.29	0.00	0.00
2020.12.30	0.00	0.00
2020.12.31	0.00	0.00
total	0.00	0.00
	Jan.2021	
date	kw/h	cost
2021.01.01	0.00	0.00
2021.01.02	0.00	0.00

Fig A1.6 Report

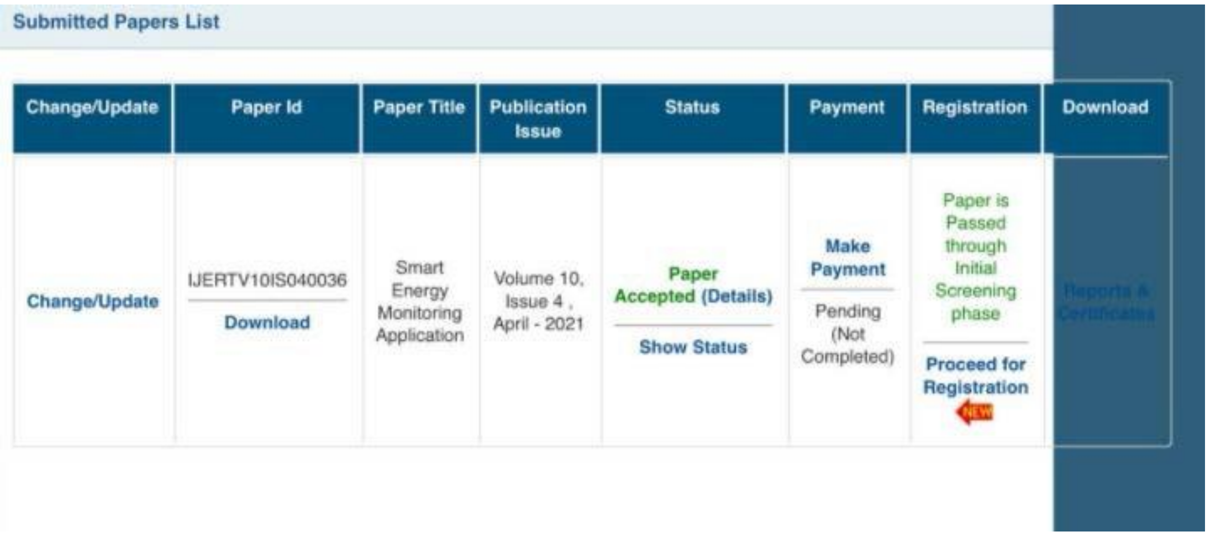
A.2 PUBLICATIONS

Journal name- Information Technology in Industry

Paper title- Smart Energy Monitoring Application

Publication issue - Volume 10, Issue 4, April – 2021

Paper Status - Paper accepted and it has passed through initial screening phase.



Submitted Papers List

Change/Update	Paper Id	Paper Title	Publication Issue	Status	Payment	Registration	Download
Change/Update	IJERTV10IS040036 Download	Smart Energy Monitoring Application	Volume 10, Issue 4 , April - 2021	Paper Accepted (Details) Show Status	Make Payment Pending (Not Completed)	Paper is Passed through Initial Screening phase Proceed for Registration NEW	Reports & Certificates

A2.1 Publication

REFERENCES

[1] Al-Kuwari, Majid, et al. "Smart-home automation using IoT-based sensing and monitoring platform." 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018). IEEE, 2018.

[2] Casini, Marco. "Internet of things for Energy efficiency of buildings." International Scientific Journal Architecture and Engineering 2.1 (2014): 24-28.

[3] DeSanctis, Martina, Romina Spalazzese, and CatiaTrubiani. "Qos-based formation of software architectures in the internet of things." European Conference on Software Architecture. Springer, Cham, 2019.

[4] Domb, Menachem. "Smart home systems based on internet of things." Internet of Things (IoT) for Automated and Smart Applications (2019).

[5] Han, Jinsoo, et al. "Smart home energy management system including renewable energy based on ZigBee and PLC." IEEE Transactions on Consumer Electronics 60.2 (2014): 198-202.

[6] Harsha, SLS Sri, S. Chakrapani Reddy, and S. Prince Mary. "Enhanced home automation system using internet of things." 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC). IEEE, 2017.

[7] Koziolek, Heiko, Andreas Burger, and Jens Doppelhamer. "Self-commissioning industrial IoT-systems in process automation: a reference architecture." 2018 IEEE International Conference on Software Architecture (ICSA). IEEE, 2018

[8] Kumar, Shiu, and Seong Ro Lee. "Android based smart home system with control via Bluetooth and internet connectivity." The 18th IEEE International Symposium on Consumer Electronics (ISCE 2014). IEEE, 2014.

[9] mLe, Shu-ping, et al. "Design and implementation of wireless power monitoring system for public buildings." Proceedings of 2011 International Conference on Electronic &

Mechanical Engineering and Information Technology. Vol. 1. IEEE, 2011.

[10] Sheela, L. J., R. Hemamalini, and M. Rejendran. "Cell phone the primary personal mobile computing devices." International Journal of Computer Trends and Technology 1.2 (2011): 87-90

[11] Pushparaj, S., et al. "IoT BASED SMART POWER MANAGING SYSTEM." Journal of Critical Reviews 7.8 (2020): 1216-1220.

[12] Singh, Harsh Kumar, et al. "A step towards Home Automation using IOT." 2019 Twelfth International Conference on Contemporary Computing (IC3). IEEE, 2019

[13] Sundarramurthi, M., AM AnjanaSundari, and AnandiGiridharan. "A Method of Designing Home Automation Control System (HACS) Using Virtual Assistant and Mobile Application." 2019 International Conference on contemporary Computing and Informatics (IC3I). IEEE, 2019.

[14] Vadivukarasi, K., and S. Krithiga. "Home security system using IoT." International Journal of Pure and Applied Mathematics 119.15 (2018): 1863-1868.

[15] Vishwakarma, Satyendra K., et al. "Smart energy efficient home automation system using iot." 2019 4th international conference on internet of things: Smart innovation and usages (IoT-SIU). IEEE, 201