IoT Based Battery Management System

Nitin M Mane[1], Dnyaneshwar V Kale[2], Sonal B Thombre[3], Rohit S Piske[4]

[1,2,3,4]Dept. Electronics and Telecommunication VPKBIET College of Engineering Baramati, India.
Professor[4], Dept. of Electronics and Telecommunication VPKBIET College of Engineering Baramati, Maharashtra, India.

---***---

Abstract - *An IoT-based battery management system (BMS) is a technology that uses the internet of things (IoT) to monitor and control batteries in various applications. The BMS consists of sensors, microcontrollers, communication modules, and cloud-based servers that work together to collect data, analyze it, and optimize battery usage. The system allows for real-time monitoring of the battery's status, including its voltage, current, temperature, and state of charge, and provides early warning of potential battery failures. By using the data collected from the BMS, the system can optimize battery charging and discharging processes, extend battery life, and reduce maintenance costs. This abstract provides an overview of the IoT-based BMS technology and its benefits, including improved battery performance, increased reliability, and reduced environmental impact.*

Key Words: Increased the Battery Life, Show the battery SOC and Temperature, monitor battery health, Cell balancing, notified to the users remotely anywhere in the world.

1. INTRODUCTION

BMS is used in many contemporary and commercial frameworks to improve battery activity and for evaluation in order to maintain the battery state, to the greatest extent possible, away from a disastrous condition and to lengthen battery lifespan. The battery's charge, temperature, and current are therefore checked using a variety of processes.

BMS framework assesses and shows the battery temperature, charging/releasing current, and SOC for the thought about model battery. For observing, computerized and simple sensors with microcontrollers are utilized. The battery data and the acquired outcomes making sense of the framework's fundamental attributes are introduced on versatile, and a few exploratory outcomes are given on the portable screen. In this demonstration, we'll build an Internet of Things weather station using a DHT11 temperature and humidity sensor, an ESP8266 development board kit, and the Blynk IoT Cloud for remote monitoring. The entire system is powered by a single 3.7V lithium-ion battery. This battery has a 10 hour operating time for NodeMCU sheets. We wish to use the charging module TP4056 to recharge the battery once again. But occasionally I forget to charge, which results in the collapse of the entire system. In order to overcome this problem, I thought about adding a battery observation framework to a similar project. In our previous battery status observing System, we can major the battery voltage and rate. In any case, presently, with the assistance of the Blynk IoT, we can straightforwardly tell the clients remotely when the battery rate is under a limit esteem. We can check the Humidity and Temperature sensor data and the battery voltage and battery percentage.

The remaining paper is divided into the following sections. Section II discusses a review of the relevant literature. System Description is covered in Section III. Section IV discusses prototype-related hardware and software design. Section V discusses analysis and outcomes from the implementation with a view to verifying our system. The Conclusion and Future Work are under Section VI.

2. LITERAURE SURVAY

This section will provide a quick overview of the available literature on Energy Management and Smart Home Systems. IoT-based automated temperature and humidity monitoring and control system was constructed utilizing a raspberry pi in one of the research projects mentioned. Pi gets the detected data for temperature and humidity, which are then transmitted to the internet. [1]. Here created an IoT-based energy management system where temperature and light intensity sensors are used, and readings are relayed to an Arduino microcontroller. The Arduino microcontroller is set up to regulate the appliance's usage as necessary. Along with managing appliance usage, the Raspberry Pi3 calculates each appliance's overall power consumption on a regular basis and plots the results as a graph using data from Hall Sensors that are wirelessly transmitted through a Wi-Fi module. All appliances with different climatic conditions have their graphical data on power usage vs time sent to a cloud server [2]. This IoT Based Battery Management System detection of hydrogen gas released by batteries has been covered in this study. The fundamental battery parameters also assist in keeping track of the battery's health. The Battery Management System will benefit from having cloud and IoT integration since it will make data analysis easier. This BMS also has a GPS tracker, [3] which makes it possible to track cars and hence give fast assistance. [4] demonstrates a full battery management system that continuously checks vital indicators and balances a battery pack's active cells as needed. The BMS system has a microprocessor (MSP430) embedded within it for monitoring and controlling units. The BMS is an electrical gadget that may be utilised in daily life, as we have discovered. This can raise the system's effectiveness, power quality, and power factor. [5] It is described how well the sensors, processor, and communications work and how accurate they are. After that, a MATLAB-created SOC estimate technique called Co-estimate was put into use on the battery hardware testbed. It is demonstrated that the developed battery hardware testbed can execute CoEstimation effectively and successfully by comparing the results from the testbed with the results from the MATLAB simulation. [6] Additionally, it illustrates that the proposed hardware architecture may be used to implement the Co-Estimation method as a workable solution in EV applications. This research offers a PLC and IoT-based energy management solution for VPP in the unified electricity market. For the DA market, it is predicted that university loads and solar generation would increase. The system makes use of automation that is PLC and IoT-based by managing the supply and loads of VPP for the best possible energy management. The PLC and Raspberry Pi are used to manage the used source and loads by sending control signals to the relay switches. The NodRed programming tool is used to programmed PLCs.

3. SYSTEM DESCRIPTION

We can check the Temperature & Humidity sensor data along with the battery voltage and battery percentage in both the charging and discharging on both cases. They show smartphones or Computer dashboards from anywhere in the world

3.1 Voltage Detection

The cut-off voltage of the battery is 2.8V, while the battery's maximum voltage is 4.2V. The ESP8266 Analogue Pin will successfully support any voltage below 3.3V. We must first descend from the higher voltage level. Two 100K resistors are present, and the supply voltage is 4.2V. A result of 2.1V will occur from this. The base value is 2.8V in essence, and the cut-off voltage drops down to 1.4V using the same voltage divider organisation. As a result, the ESP8266 Analogue Pin maintains both the higher and lower voltage. If the voltage rises beyond 4.2V, the automated supply of power will be cut off.

3.1.1 Voltage Divider Network Calculations

The cut-off voltage of the battery is 2.8V, while the battery's maximum voltage is 4.2V. The ESP8266 Analogue Pin can readily accommodate any voltage lower than 3.3V. We must first descend to the higher voltage level.

Two 100K resistors are present, and the supply voltage is 4.2V. A result of 2.1V will occur from this. The base voltage is also 2.8V, and the cut-off voltage drops down to 1.4V using the same voltage divider structure. Consequently, the ESP8266 Analogue Pin maintains both the upper and lower voltage.

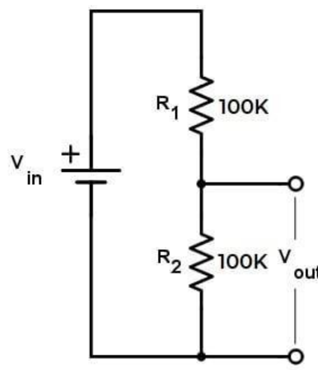

Fig 1. Voltage Driver Network Calculation circuit diagram

3.2 Temperature and Humidity Detection

We will plan a framework to screen DHT11 temperature and dampness, battery voltage, alongside charging and releasing status. For the microcontroller, we use NodeMCU, which has an ESP8266 Wi-Fi-empowered chip. You can likewise utilize a more modest board like the Wemos D1 Mini. This Wi-Fi chip can associate with the Wi-Fi organization and routinely transfer the information to the server. The Out pin of the DHT11 sensor is associated with the D4 pin of the NodeMCU. SCL and SDA pins are associated with D1 and D2 pins. Though VCC and GND of DHT11

3.2 CALCULATE BATTERY PERCENTAGE

We show the battery percentage on the Blynk template dashboard and show it on the mobile screen. Battery percentage showing in between 1 to 100.special feature is Blynk IoT user can access the data directly anywhere in the world. The mechanized Battery Monitoring System vehicle with different control components can be worked by making associations. For the microcontroller, we utilize the NodeMCU, which has an ESP8266 WIFI-empowered chip. You can likewise utilize more modest sheets like the Wemos D1 Mini. This Wi-Fi chip can interface with a Wi-Fi organization and transfer information to the server routinely.

3.4 DETECT HARMFUL GASES

We Continuously monitor the presence gases and inform to Raspberry pi. In case gas sensor detect the unwanted of flammable dangerous gas it informs the user and shut down the System. Gas sensor detect the leakage gases of li-Ion battery. For detect the gases we use the MQ17 Gas sensor.

This gas sensor measured the carbon monoxide, carbon dioxide, Hydrogen and flammable Gases.

3.5 CURRENT CALCULATION

The Hall sensor is used to measure the current flow in the circuit. The Allegro™ ACS712 is a Hall sensor imparting in your price range and unique answer in industries, business and verbal exchange structures. The hall sensor is utilised to gauge current flow in the cable. The hall sensor can gauge the current by applying a rigid resistance to the wire. In order to pass past the corridor sensor, a portion of the wire from the transistor that is heading to the appliance is cut down. Three pins make up the corridor sensor: a voltage pin, a ground pin, and an output pin. The output pin is connected to the analogue pin within the Arduino, and the voltage pin is connected to the 5-volt supply from the Arduino. The floor pin is connected to the ground. The Arduino pin A1 is connected to the analogue pin A2 of the hall sensor, which measures the current in the fan, and the analogue pin A2 of the light sensor, which measures the current in the light.

4. HARDWARE AND SOFTWARE DESIGN

In this section we shortly discussed about how will be work our System and what is the contribution of all components.

4.1 HARDWARE

In starting when our system will initialize charging is going on. ESP8266 Wi-Fi kit continuously send the system data to Blynk cloud and also Raspberry pi 3. Pi continuously monitors the battery condition and SOC. In case battery charging percentage go above 100 percent or fixed charging percentage. It will cut off the charging and send the notification to the user and in case battery percentage going below 20 percentage it will be inform the user through notification. Wi-Fi module continuously measure the battery surrounding Temperature and Humidity for battery better life. In case battery surrounding temperature going high or increased certain fix value. Pi will be starting the cooling fan for maintain the temperature. We used DHT11 Temperature sensor for measuring the temperature and Humidity. Gas sensor MQ7 used for sense the gases of surrounding battery area or also monitors the battery leakage gases, sometimes causes of internal Li-Ion battery chemical reaction battery leaked the harmful and flammable gases. For avoiding this thereat we use the gas sensor.

The circuit's current flow may be measured using a hall sensor. The ESP8266 will get information from the Hall sensor about how much current is being supplied to the appliance. After that, the raspberry pi will receive the current from the ESP8266. The pi will receive the current consumed, figure out the power spent, post it to a webpage, and also draw a graph depending on the current consumption.

All system is connected with help of jumping wires. All component and each sensor are connected through the jumping wire. For charging purpose, we used the TP 4056 Charging module. This module taking care from voltage fluctuations.

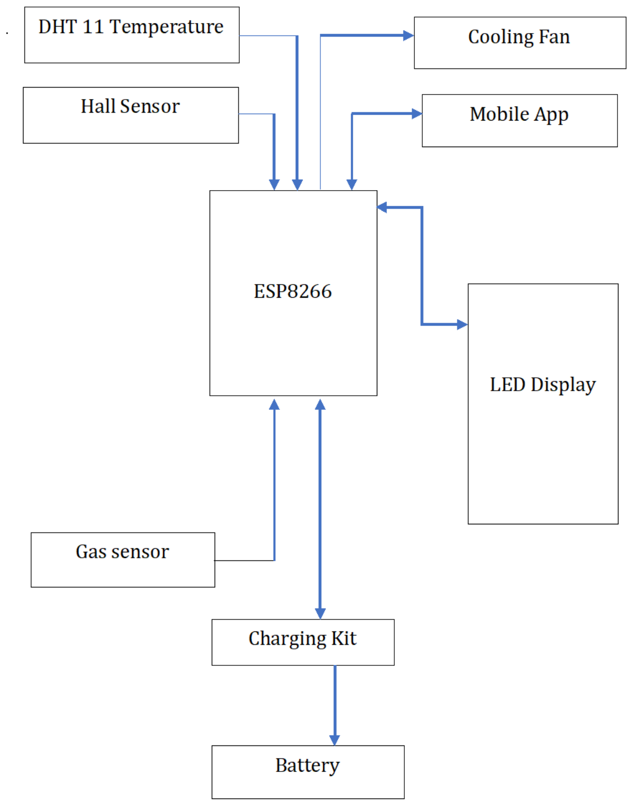

Fig 2. Block Diagram

Mobile app Blynk 2.0 is used for remotely monitor and operate to our system. we can operate our system anywhere this is the biggest advantage of Blynk 2.0. It continuously monitors our system and show the user accurate data. Using Blynk IoT App we start the charging our battery or we can stop the charging our battery. LED (light emitted diode) display we use for show the date of our system, just like Temperature, Humidity, Voltage, Current, etc. Blynk application is user friendly and easy to operate. It continuously plots the graph and show the data according to system changes. It plotted the graph in every minute.

Whole BMS system is operated on 3.7 Volte Li-Ion battery and this battery charging with TP4056 charging module. We use the two resistors for divide the voltage and provide the safe voltage towards the battery better life. In the whole system ESP8266 is play the very essential and important role for our BMS. It continually sends the data to the cloud database and Blynk cloud platform. The data are just like battery voltage, presence of gases surrounding battery, battery Temperature, battery current, battery charging percentage. Using Pi, we maintain the battery SOC and battery balancing and cell charging. It informed the whole battery data or information remotely to the user.

Name and specification of component	Application of the component
Node MCU esp8266	An ESP8266 Wi-Fi module is a SOC computer chip chiefly utilized for the improvement of end-point IoT (Internet of things) applications. It is alluded to as an independent remote handset, accessible at a small cost. It is utilized to empower the web association with different uses of installed frameworks.
TP4056 ChargingModule	The TP4056 chip is a lithium-Ion battery charger for a solitary cell battery, safeguarding the cell from over and undercharging
DHT11 Temperature Sensor	This sensor is used in various applications such as measuring humidity and temperature. values in heating, ventilation and air conditioning systems.
Li-Ion Battery	Utilizations of battery incorporate giving reinforcement power during a blackout. At home, the batteries are regularly wired to electrical machines.
Jumper Wires	Jumper wires are simply cables with connector pins on both ends that may be used to link two focuses wirelessly.
Breadboard	The primary utilization of a breadboard is to shape straightforward electrical associations among various parts with the goal that you can really look at your circuit prior to fastening it to the board
100k Resistor	100K ohm(1/4W) Resistor Fixed Resistor (100000 ohms 5) The carbon film resistor is a kind of fixed resistor that utilizes carbon film to restrict the electric current to a certain level.
Cooling Fan (Motor)	DC 5V 2510 Cooling Fan. This mini fan has the ability to run at a speed of 6800~1300 rpm
Gas Sensor	The MQ-7 can detect CO-Gas concentration anywhere from 10 to 500 ppm.
LED Display	LED Display is used to show the measures of the project
74HCT4051 analog multiplexer/demultiplexer	8-channel analogue multiplexer/demultiplexer (74HC4051; 74HCT4051) has three digital select inputs (S0 to S2), active-LOW enable input (E), eight independent inputs/outputs (Y0 to Y7), and a common input/output (Z). S0 to S2 choose (low impedance ON-state) one of the eight switches when E is LOW.
Hall Sensor	The hall sensor is employed to gauge the amount of wire current. By attaching a set resistance to the wire, the hall sensor is able to measure the current.

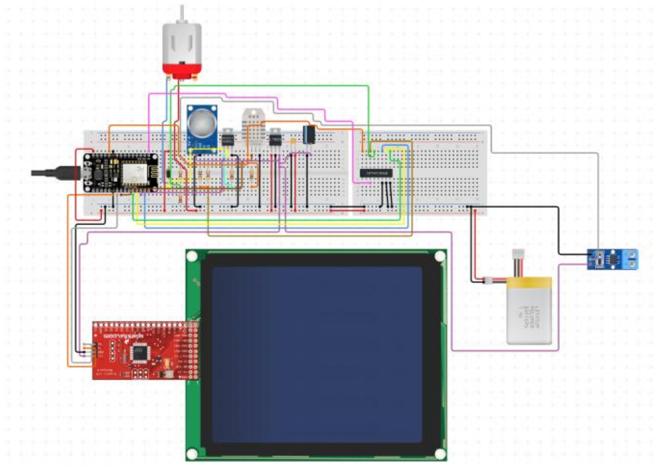

Fig 3. Circuit Diagram

4.2 SOFTWARE

For the software we use the Blynk IoT Platform. In this platform first you need create account, after creating account define the parameters what you want. In this module we designed for measure the temperature, voltage, current and presence of gases.

4.2.1 CONFIGURING THE BLYNK IoT DASHBOARD

Setting up the Blynk IoT Cloud dashboard is required before you can monitor sensor and battery data on the Blynk IoT Server. Please go to https://blynk.cloud/ to configure the Blynk Server. Create a new account or sign in if one already exists.

4.2.2 Making the Blynk Template

A template is a task that allows us to build a web and mobile dashboard for certain hardware. It is an IoT smart device in our scenario. You need to select the New Template in order to start a project.

- Enter a template name. I am Giving it “Smart IoT”.
- Select the hardware board (ESP8266).
- The connection type will be Wi-Fi.
- You can add a description of your project. If required
- Select "Done."
- The template has now been made.

You may then select a category based on your project after that. Don't worry if your category isn't on the list; you may select another. A temperature sensor is what I'm choosing.

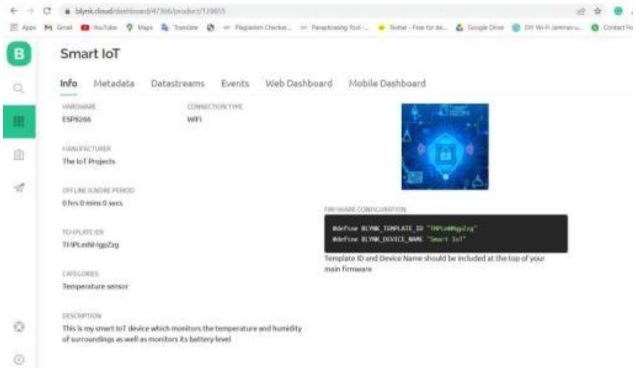

Fig 4. Controls of the vehicle in the android application

4.2.3 Making Events for Notification Alerts on Blynk

Systems for notifying and alerting users use events. I'm making events now to track Battery Percentage. A notice is delivered to your mobile phone and an event is triggered if the battery percentage falls below the threshold level.

- Press the add event button.
- Enter the event Name. For me, it's "Battery Low "
- Choose your event color. I am selecting Red.
- Select the type of event. I am choosing Critical.
- Enter the description of your event.
- Now go to the notification tab then enable notification.
- Then select push notification to device owner every 1 minute.
- Click on save

4.2.4 Making a New Blynk DataStream

An information stream resembles a pipeline or channel. The information will be sent or gotten through these information channels.

In a solitary task or layout, there can be various DataStream. In our task, we get four information: Temperature, Humidity, Battery Voltage, and Battery Percentage. So, in this task, we made Four DataStream. Click on the new DataStream. Presently pick a virtual pin. Presently you need to give a name for each datum stream. From that point onward, you need to pick virtual pins for DataStream. For Temperature, Humidity, Battery Voltage, Battery Current and Battery Percentage, presences of gases I pick virtual pins V1, V2, V3, V4, V5 and V6 separately. I pick the datatype as Double. You likewise need to set units for DataStream according to the variable. I pick a degree centigrade for temperature. The rate for Humidity and Battery Percent then Volt for battery voltage. Likewise, set the base and greatest qualities for the information streams. We can screen the sensor information as well as battery charging/releasing status alongside Battery Voltage and Percentage.

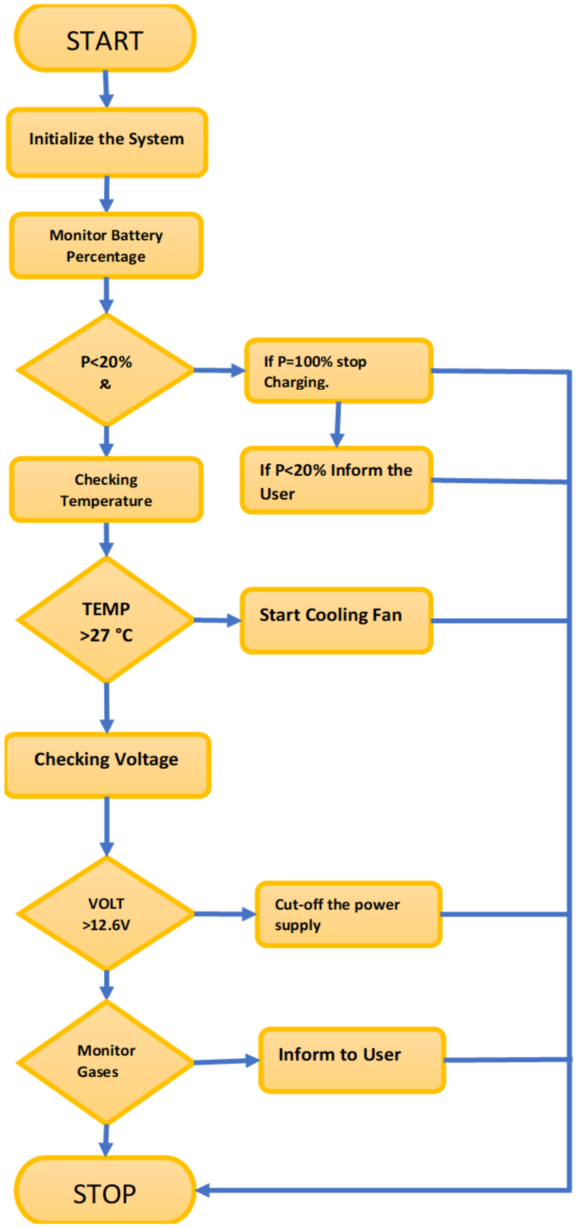

Fig 5. Flowchart

So, we can re-energize it on time. This can be immediately done utilizing a voltage divider circuit and the simple contribution on the NodeMCU ESP8266 board. Hall effect sensor use for measure the current of battery. Using Hall sensor show the information about battery current for current we use the above DataStream variable and presence of harmful gases or in case of battery internal chemical reaction battery leaked some flammable gases for that purpose we use the MQ8 sensor it senses the hydrogen gas in the environment.

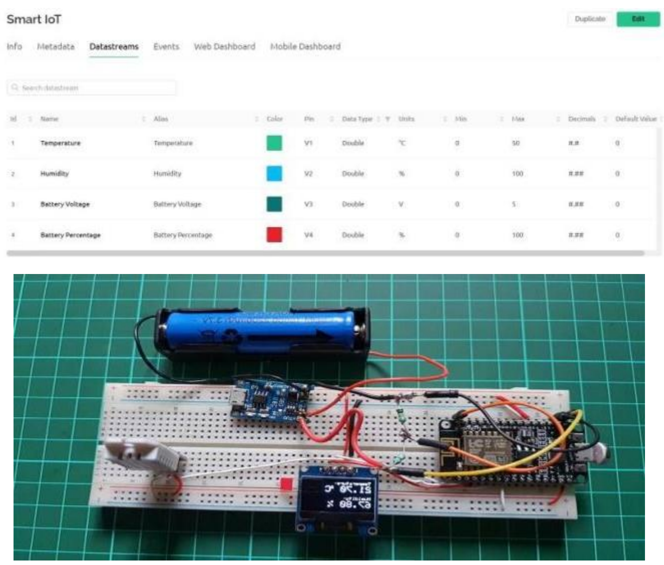

Fig 6. Blynk IoT data stream

5. IMPLEMENTATION RESULTS AND ANALYSIS

The complete hardware prototype of IoT based Energy Management system developed employing A ESP8266 as microcontroller unit. In addition, Temperature and Gas Sensors, Hall sensor these sensors are deployed on input side and output side we deploy the DC motor for cooling purpose and LED Display for data showing purpose. NodeMCU Wi-Fi Module we use for communication purpose. It transmits data of all sensors and send to the Blynk cloud platform on Blynk dashboard display the current drawn from each appliance for computing total power consumed and same plotted as graph. The results been updated as HTML Webpage in Cloud server. Fig. 5 Circuit Diagram and Fig. 7 shows the complete IoT based Energy Management System Prototype with all sensors and connection

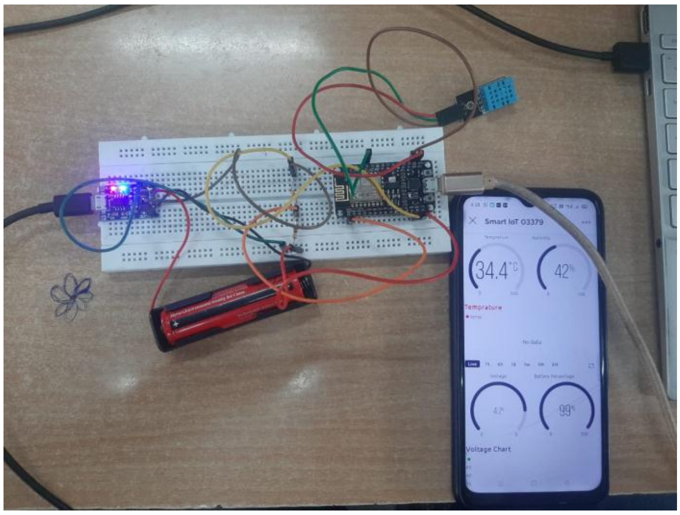

fig 7. Result Diagram

You can also check the when battery temperature is going high cooling fan is started automatically. Also, you notice the when battery charging percentage going below 20 it continuously sends the notification to the user mobile phone. One another Bonas point is it operated remotely anywhere in the world that point necessary in today era so we need to work on it.

You can see the in fig8 it the update the gauge and graph according to sensor data. It continuously plots the graph and showing every minute of battery status.

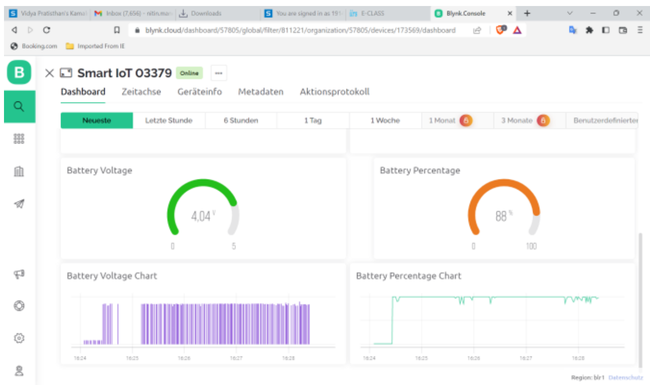

Fig 8. gauge and graph plotting on desktop screen

In the below diagram we show you codding of nodemcu. For codding we use the various libraries related to the sensors and output devices. In this code write about instructions how to perform work step by step.

First, we install the libraries in Arduino software then install the esp8266 board after completed this process. We can start the writing code for BMS. In a first line we define the sensor libraries and packages after define pakeges and libaries . we move the next step of code writing. In the code we mention the one loop for continuously measure the current and temperature humidity.

Fig 9. Codding of NodeMcu

6. CONCLUSON AND FURTHER WORK

It is very important for BMS to well-maintained the battery reliability and safety, the state monitoring and evaluation, cell balancing and charge control are well functional. Thus, this present paper is review on BMS, focussing study for optimization of BMS that will lead to reliability of BMS and optimize power performance. Monitor the battery anywhere in the world using IoT. It's taking the data from IoT embedded sensor and transmit the cloud. Provide better performance to user enhance the battery life. This IoT based battery management system help us to monitor the battery conditions and also help us to monitor the battery health.

Further we added the more sensor for humidity management and finding the efficient option for battery temperature and battery colling. we think about battery cooling and for that we will used solid colling material. This material automatically melts when battery getting hot and after battery cold. It will again get again it's first stage mean it again got it first stage which was solid.

We think about fully system is make as AI. We want to made an AI Based Battery management system. In this AI fully system is controlled by AI. Few things only handle by human hand.

7. ACKNOWLEDGEMENT

I want to publicly thank Mr. R. S. Piske, the project manager, and Dr. B. H. Patil, the department head of electronics and telecommunications. I appreciate all they did to assist me finish my project on an IoT-based battery management system.

I want to specifically thank our mentor, Mr. R. S. Piske, for all of his help and support this past year. Your helpful tips and recommendations were really beneficial to me while I worked on the project, and they continue to be so as I finish it. I will always be thankful to you for this.

I want to say that my batchmate and I finished this assignment fully on our own, without assistance from anybody else.

8. REFERENCES

[1] M. Lavanya, P. Muthukannan, Y.S.S. Bhargav, V. Suresh, "IoT Based Automated Temperature and Humidity Monitoring and Control", Journal of Chemical and Pharmaceutical Sciences ISSN: 0974- 2115.

[2] Vignesh Mani, Abhilasha, Gunasekhar," IoT Based Smart Energy Management System", International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 16 (2017) pp. 5455-5462

[3] Harish N[1], Prashal V[2] and Dr. D. Sivakumar[3]," IOT Based Battery Management System", International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 8 (2018) pp.5711-5714

[4] Abhay Shatrudhan Jha[1], Anuj Kalkar[2], Mandar Ghanekar[3], Neeraj Pal[4], Sangeeta kotecha[5], "Battery Management System", International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 02 | Feb 2020. e-ISSN: 2395-0056

[5] Seyed Mahmoud Salamati*, Cong Sheng Huang*, Bharat Balagopal*, and Mo-Yuen Chow*, "Experimental Battery Monitoring System Design for Electric Vehicle Applications", Electrical and Computer Engineering, North Carolina State University, Raleigh, U.S.A, IEEE 2018.

[6] Poushali Pal, A. K. Parvathy, K. R. Devabalaji, S. Joseph Antony, Simon Ocheme, Thanikanti Sudhakar Babu, Hassan Haes Alhelou, T. Yuvaraj," IoT-Based Real Time Energy Management of Virtual Power Plant Using PLC for Transactive Energy Framewor", IEEE| July-2021.