Development of Smart Energy Meter for IoT-Enabled Power Monitoring

Annesha Sadhukhan[1], Prasun Chowdhury[2]

Department of Electronics & Communication Engineering St. Thomas' College of Engineering & Technology Kolkata, India

DOI: https://doi.org/10.47772/IJRISS.2026.100600559

Received: 10 June 2026; Accepted: 16 June 2026; Published: 29 June 2026

ABSTRACT

Existing low-cost smart energy metering systems based on ESP8266 and IoT platforms primarily focus on remote monitoring, while offering limited support for integrated energy analytics, automated billing, threshold-based alerts, and prototype validation across both simulation and hardware environments. To address this gap, this paper presents a cost-effective IoT-enabled smart energy meter that combines real-time electrical parameter monitoring, automated energy consumption analysis, and cloud-based visualization within a unified framework. The proposed system is developed and validated in two stages: (i) a Proteus-based simulation platform incorporating virtual energy measurement and wireless data transmission, and (ii) a hardware prototype implemented using a NodeMCU (ESP8266) module integrated with voltage and current sensing circuits and the Blynk IoT platform. The system continuously measures voltage, current, power, and energy consumption, automatically computes electricity usage and billing information, generates threshold-based consumption alerts, and enables remote monitoring through a mobile application. Unlike conventional smart meter implementations that primarily provide data logging, the proposed prototype integrates energy measurement, billing estimation, warning notifications, and cloud connectivity in a low-cost architecture suitable for residential energy management. Experimental validation demonstrates reliable real-time monitoring and remote accessibility, highlighting the feasibility of the proposed solution as an affordable smart metering platform for IoT-enabled energy management applications.

Keywords: Smart Energy Meter, IoT, NodeMCU, Blynk, Arduino, Wireless Metering

INTRODUCTION

Electricity is one of the most essential resources for domestic, industrial and agricultural activities. Proper monitoring of energy consumption is therefore vital for economic and sustainable power utilization. Traditional energy meters require manual inspection, which introduces human errors and delays in billing. Smart energy meters provide automatic measurement and remote accessibility, reducing operational cost and improving transparency [1]. A smart energy meter is an electronic device that accurately measures electricity consumed by a residence or facility and communicates the data to the consumer and utility without manual intervention [1]. Real-time access to consumption information enables users to control wastage and plan usage efficiently. The proposed system integrates microcontroller-based measurement with wireless communication and a mobile application interface. The project was executed in two phases. In the first phase, a complete model was developed in Proteus using Arduino and Bluetooth module HC-05 to transmit data to an MIT App Inventor application [2][3]. In the second phase, Bluetooth was replaced with Wi-Fi using NodeMCU ESP8266, enabling IoT connectivity through the Blynk cloud platform [4][5]. This enhancement allowed remote monitoring over the internet and advanced visualization. Smart meters are becoming popular because they remove the need for manual reading and minimize billing discrepancies [6]. The system described in this chapter aims to deliver a cost- effective prototype suitable for residential deployment.

Although numerous smart energy monitoring systems based on Arduino, ESP8266, GSM, Bluetooth, and IoT platforms have been reported in the literature, most existing studies focus primarily on remote data acquisition and visualization. Limited attention has been given to integrated billing estimation, threshold-based warning

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

generation, dual-stage validation through simulation and hardware implementation, and comparative evaluation of communication approaches. Furthermore, many reported prototypes lack quantitative performance analysis and accuracy assessment.

The novelty of the proposed work lies in the development of a low-cost smart energy meter that combines (i) real-time measurement of voltage, current, power, and energy consumption, (ii) automatic billing estimation, (iii) threshold-based mobile and email notifications, (iv) comparative implementation using Bluetooth and Wi-Fi communication, and (v) validation through both Proteus simulation and hardware realization. The proposed system aims to provide an affordable residential energy-monitoring platform with enhanced user awareness and remote accessibility.

LITERATURE SURVEY & REVIEW

Table 1. Benchmark Comparison with Existing Smart Meter Systems

Feature	Anitha et al. [7]	Koay et al. [8]	Patel et al. [9]	Saleem et al. [10]	Proposed Work
Real-Time Monitoring	Yes	Yes	Yes	Yes	Yes
ESP8266/NodeMCU	No	No	No	Yes	Yes
Bluetooth Communication	No	Yes	No	No	Yes (Phase-I)
Wi-Fi/IoT Connectivity	No	No	Limited	Yes	Yes
Mobile Application	No	No	SMS Based	Yes	Yes
Automated Billing	No	No	Yes	No	Yes
Threshold Alert	No	No	Yes	No	Yes
Cloud Dashboard	No	No	No	Yes	Yes
Simulation + Hardware Validation	No	No	No	No	Yes
Accuracy Evaluation	No	No	No	Limited	Yes
Notification through App & Email	No	No	SMS Only	No	Yes
Cost-Effective Residential Prototype	Yes	Yes	Yes	No	Yes

Survey	Pros	Cons	Steps taken to surmount the above disadvantage:
Anitha[7] proposed "Smart energy meter surveillance using GSM technology". The foremost objective of his project is to create awareness about energy consumption and. efficient use of home appliances for energy savings. Due to manual work, existing electricity billing system has major drawbacks. This system will give the information on meter reading, power cut when power consumption exceeds beyond the specified limit.	• Proper monitoring and controlling of power consumption. • Helps in efficient use of power consumption. • Helps in detecting faulty appliances.	• Does not make use of IoT. • Does not have home automation feature.	• Our project will have an additional feature as we will implement it by making use of IoT.

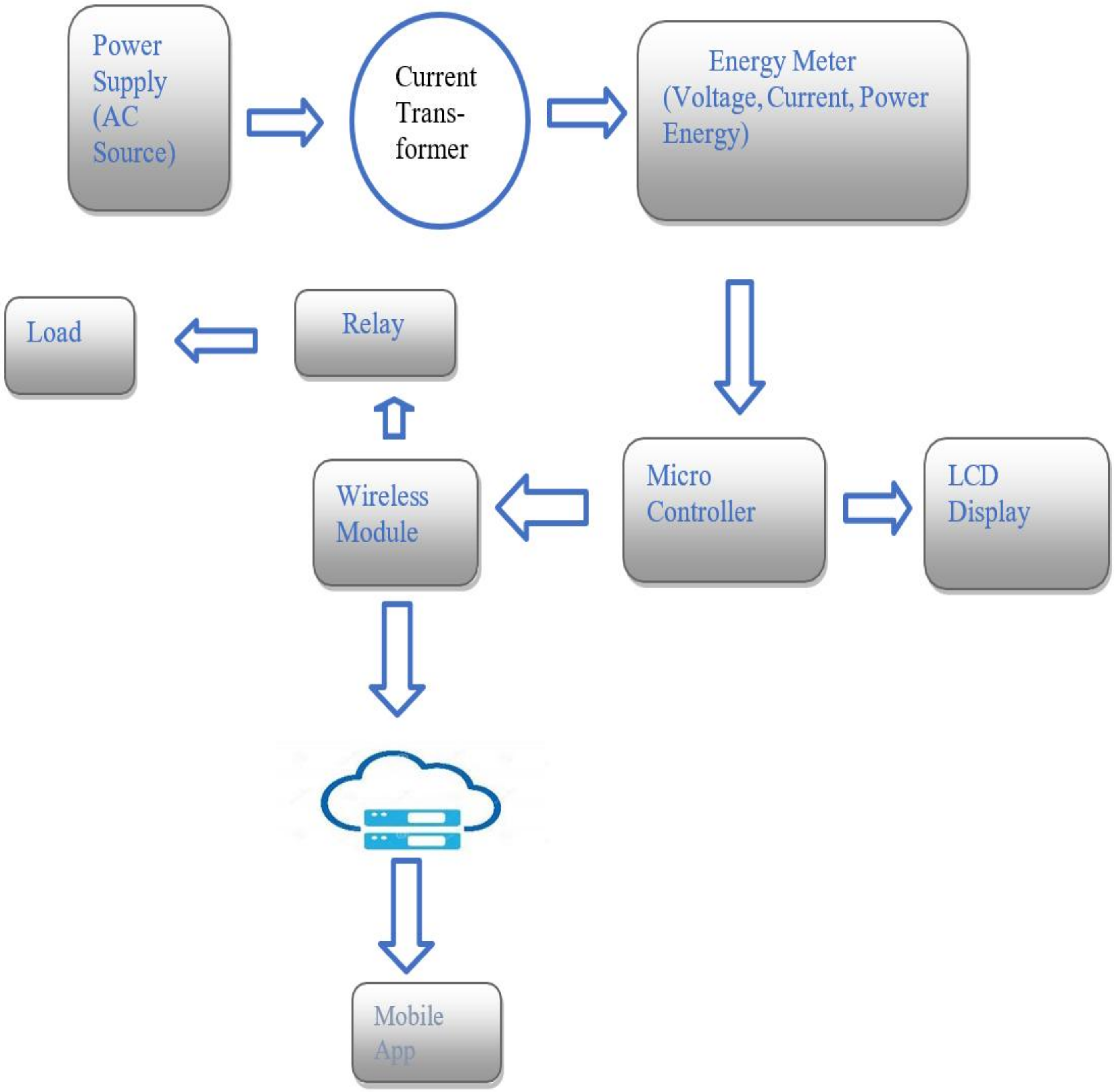
Mr. Koay [8] explained "Design and implementation of Bluetooth energy meter" described around the year 2004, digital meter has started to replace the electromechanical meters in Singapore. A wireless digital power meter would offer greater convenience to the meter reading task. Bluetooth technology is a possible wireless solution to this issue. The power reader can collect the power consumption reading from the energy meter wirelessly based on Bluetooth.	• Proper monitoring and controlling of power consumption. • Helps in efficient use of power consumption.	• Does not have home automation feature.	• Additionally, our project will have home automation feature beside smart energy monitoring.
Himanshu K [9] Patel demonstrated "Arduino based smart energy meter" that removes human intervention in meter readings and bill generation thereby reducing the error that usually causes in India. The system consists the provision of sending an SMS to user for update on energy consumption along with final bill generation along with the freedom of reload via SMS.	• Proper monitoring and controlling of power consumption. • Helps in efficient use of power consumption. • Have home automation feature.	• Does not have the feature of automatically turning off the devices when no one is there in the home.	• Beside, having all the features which are already here, our project will have an additional feature of automatic power off of devices when no one is there in the house.
M. Usman Saleem ,M. Rehan Usman [10], M. Arslan Usman in 2022 designed, deployed and performed evaluation of an IoT Based Smart Energy Management System. One of the important feature of this project is to help in efficient energy management system	• Helps in efficient use of power consumptio n. • Helps in reduction of energy wastage.	• Does not have the feature of home automation feature.	• Beside having these features, our project have home automation feature.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Details of the Project

Block Diagram

Fig1: Block Diagram of the project



The smart energy meter monitoring system is shown in the above Fig1. The block diagram consists of the current transformer, Arduino, energy meter, LCD, Mobile app, Wireless module, and IoT. The energy meter used here is a clamp energy meter. 230V AC mains is the input given to the transformer and AC mains is converted to low voltage. The energy meter measures the live current, voltage, and calculating power in terms of KW-h. the resultant value of power the microcontroller calculating the desired bill in terms of Rupees, and its shows in LCD. Here, wireless Module(ESP-8266) is used for the data communication. We developed an app using MIT app inventor for fetching the data to the user's mobile. The data from the energy meter is sent to wireless module (ESP-8266) that reaches to the Blynk server and Server sent to user's mobile phone.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Flowchart

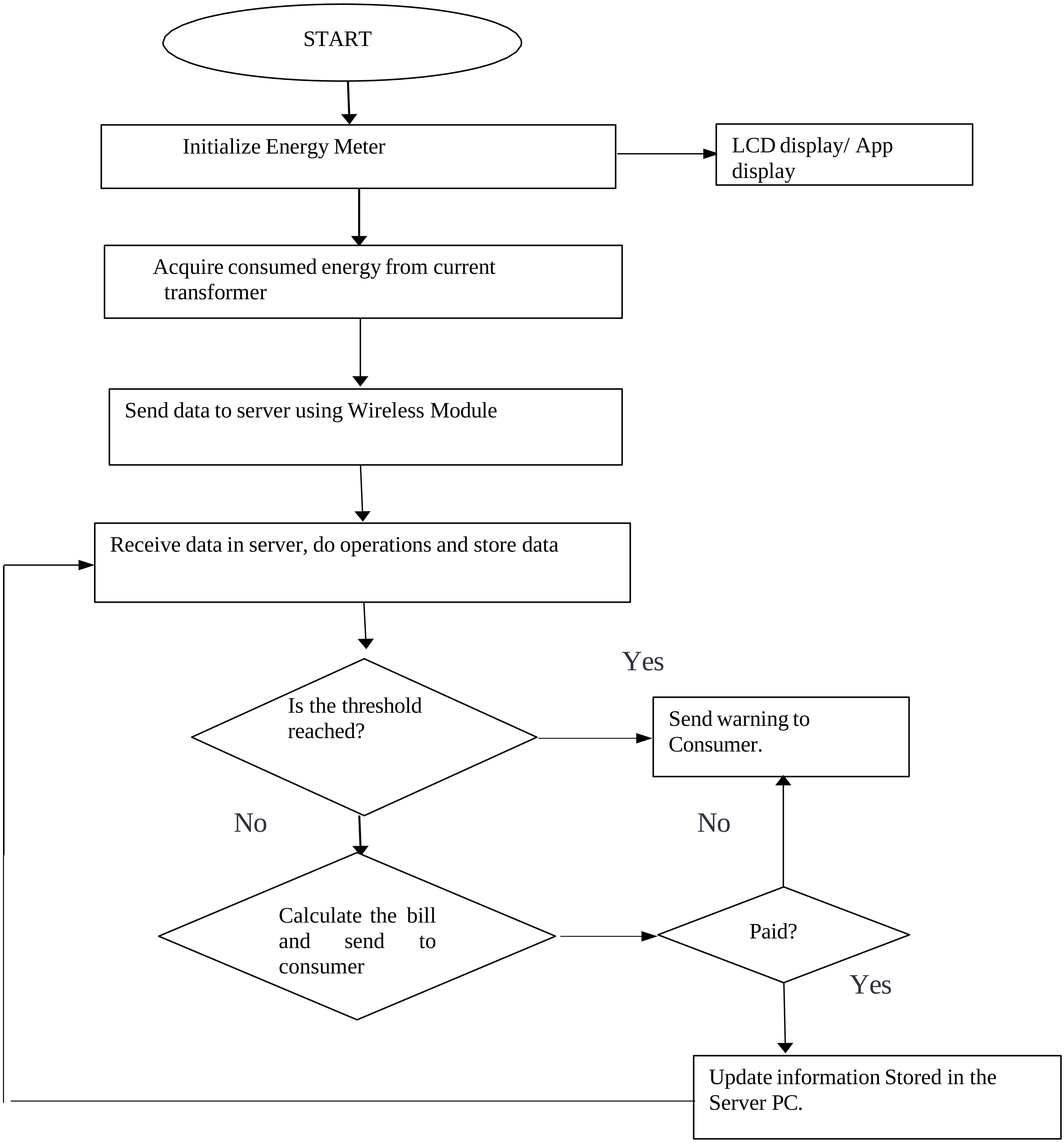


Fig2: Flowchart of the project

The methodology of the project is firstly we have to initialize the energy meter and see that the values are displayed on LCD. After that Acquire total used energy from the current transformer then the value is sent to the server using the Wireless module. When the server received the data, the server stored the data first then the value sends to the user's mobile app. And also updated value continuously stored in the server which is shown in the above Fig2. After that if the bill-threshold (Provided to the app by user) is reached, a warning will be sent to the user as an email and a notification in the app. And, if not then, do operations to check that the consumer paid the bill using the app. If yes, then the updated information is stored in the server PC. Otherwise, send a warning message to the user's mobile, for paying the bill.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Component List

Table 2: Hardware Specification Table

Component	Specification
NodeMCU ESP8266	80 MHz, 32-bit MCU, Wi-Fi
Arduino Uno	ATmega328P, 16 MHz
LM324	Quad Operational Amplifier
LCD	20×4 Character Display
Current Sensor	Current Transformer Based
Load-1	3 W LED
Load-2	1 W LED
Communication	Bluetooth HC-05 / Wi-Fi ESP8266
Cloud Platform	Blynk IoT
Power Supply	230 V AC Input

• **First Phase:**	• **Second Phase**
1. Arduino Uno	
2. Bluetooth HC-05	
3. Capacitor	1. Arduino Uno
4. Diode	2. Resistor
5. Potentiometer	3. Breadboard
6. Transformer	4. LM324 Op-amp
7. Source	5. LED
8. LCD	6. ESP8266 Wifi Module
9. Voltmeter	

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Description about Proteus Platform:

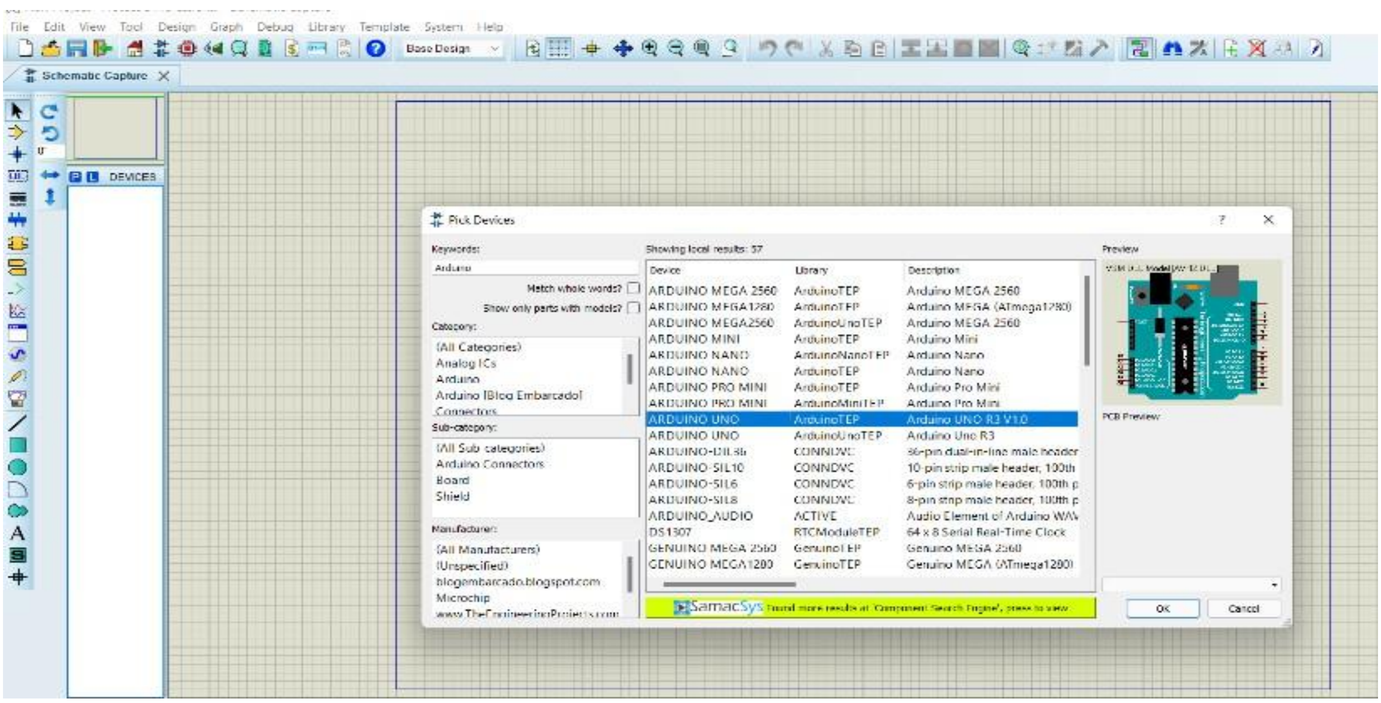


Fig 3: Proteus Platform

The complete circuit diagram was done in the Proteus 8 professional platform whose part is shown in Fig 3. It is a circuit designing software invented by Lab Center Electronics. It is used to design different circuits on PCB and simulation of different circuits. The use of proteus for any electronic circuit project makes that project cost-effective and less errored due to schematic construction on the proteus. There is zero possibility of burning and damaging any electronic component in proteus. Electronic tools that are very expensive can easily get in the proteus.

Layout Diagrams

First Phase:

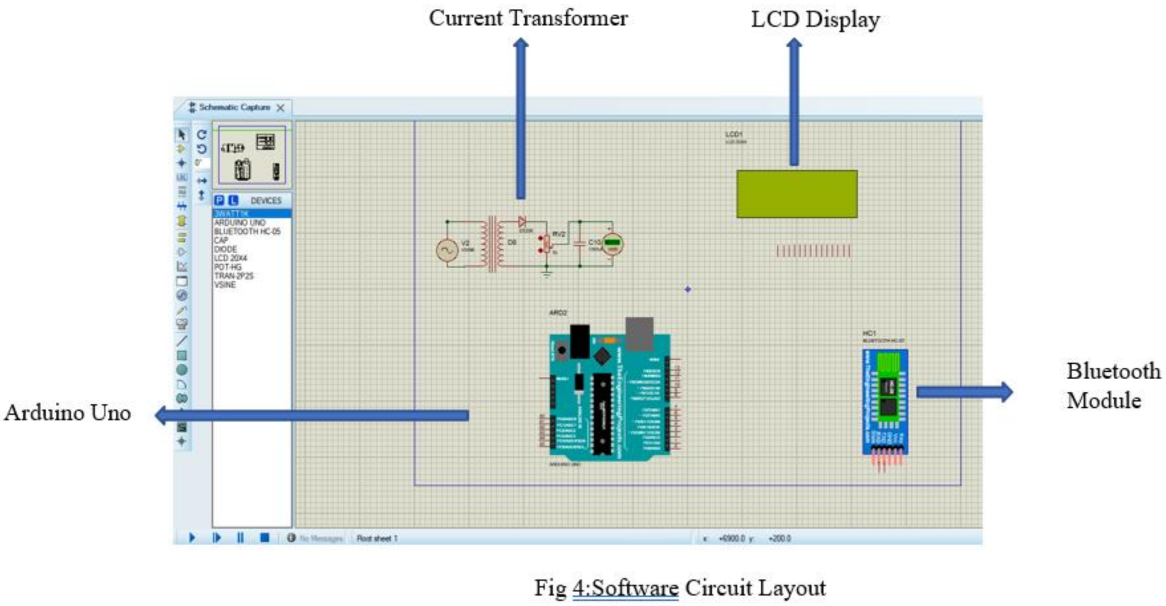


Fig 4:Software Circuit Layout

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

In the above Fig 4, we have a current transformer circuit, Arduino Uno board, LCD (20 X 4), and Bluetooth module (HC-05). when we design the current transformer, we use a step-down transformer because the Arduino Uno works in low voltage and we scaled 220 volts into 5 volts. And also we use a diode for rectification. We don't want a negative value in the output and we use a potentiometer (variable register) for that we scaled the value in between 5 volts manually and give a specific value. Also, we used a capacitor for filtering and giving a stable output.

Second Phase:

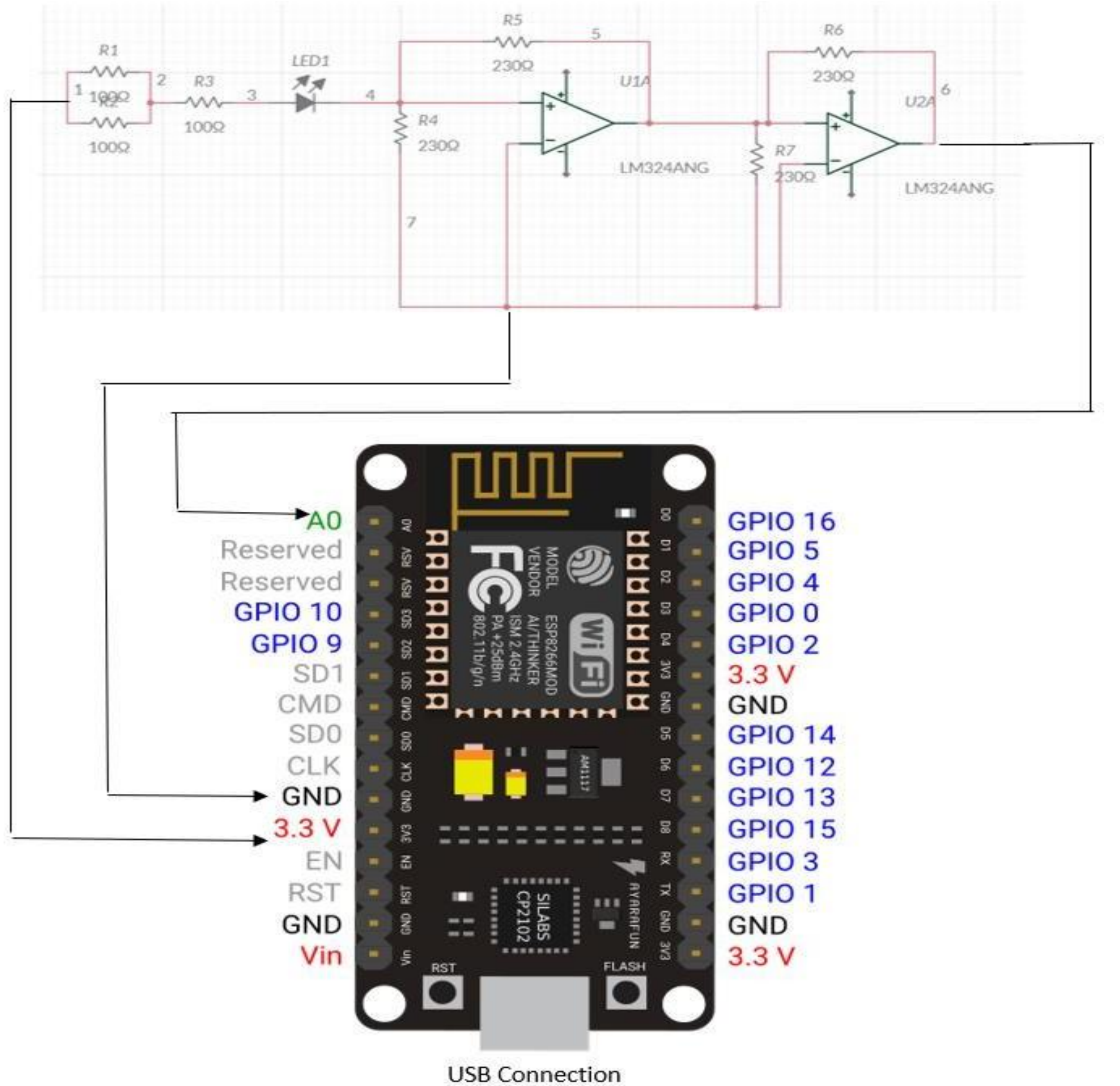


Fig5:Hardware Circuit Layout

In the above fig 5 shows the hardware circuit layout of our project. We use a breadboard, multimeter, three 100 ohm 1-watt resistors, LED (3W and 1W), Wifi module (ESP-8266), and also an I-V converter circuit. We use LED for the load because this is a prototype model, To turn on the LED set the two 100ohm 1watt resistors parallel and one in series. The Wifi module (Esp-8266) I/O pins runs at 3.3 Volt. The current across the load part needs to be converted. That's Why we have designed a current to voltage converter circuit using LM324 IC and

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

four 230-ohm resistors. The output of the converter circuit connects to the Wifi module (ESP-8266) for measuring the consumed power and calculating the bill and fetching the data to the server.

Circuit Diagram

First Phase:

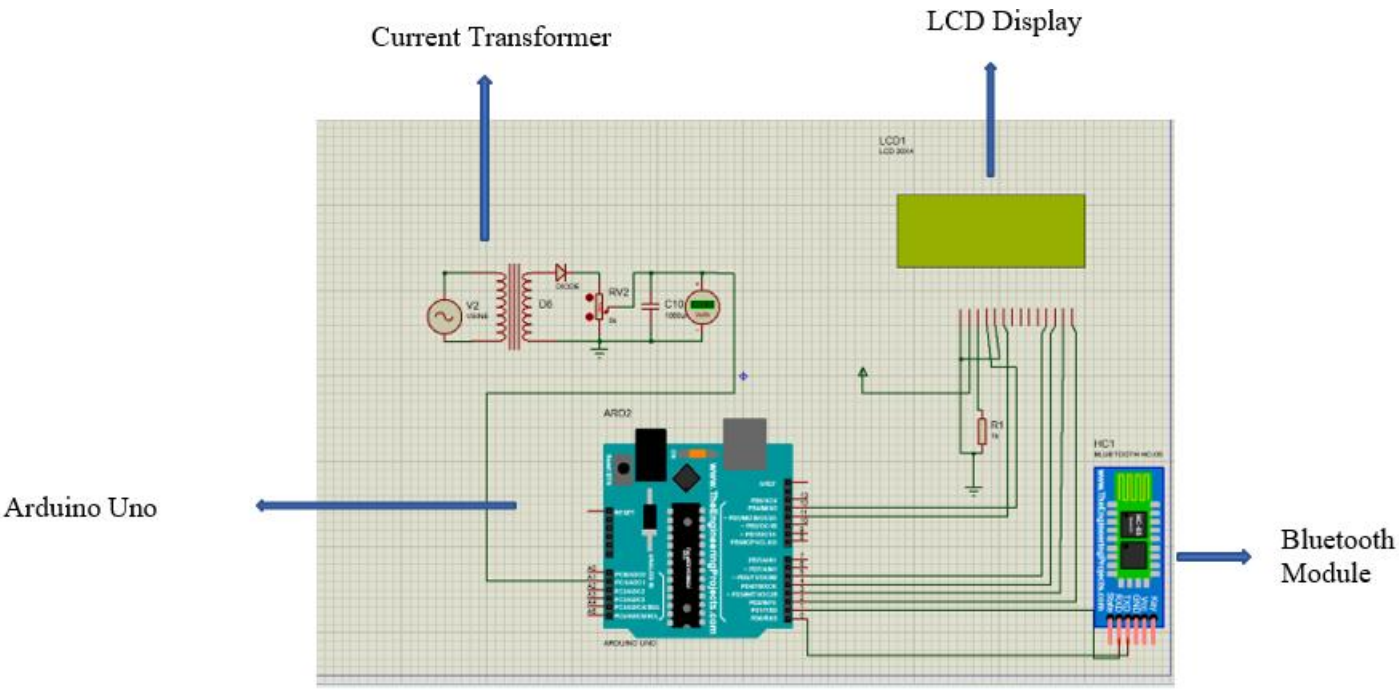


Fig 6: Software-Circuit Diagram

Second Phase:

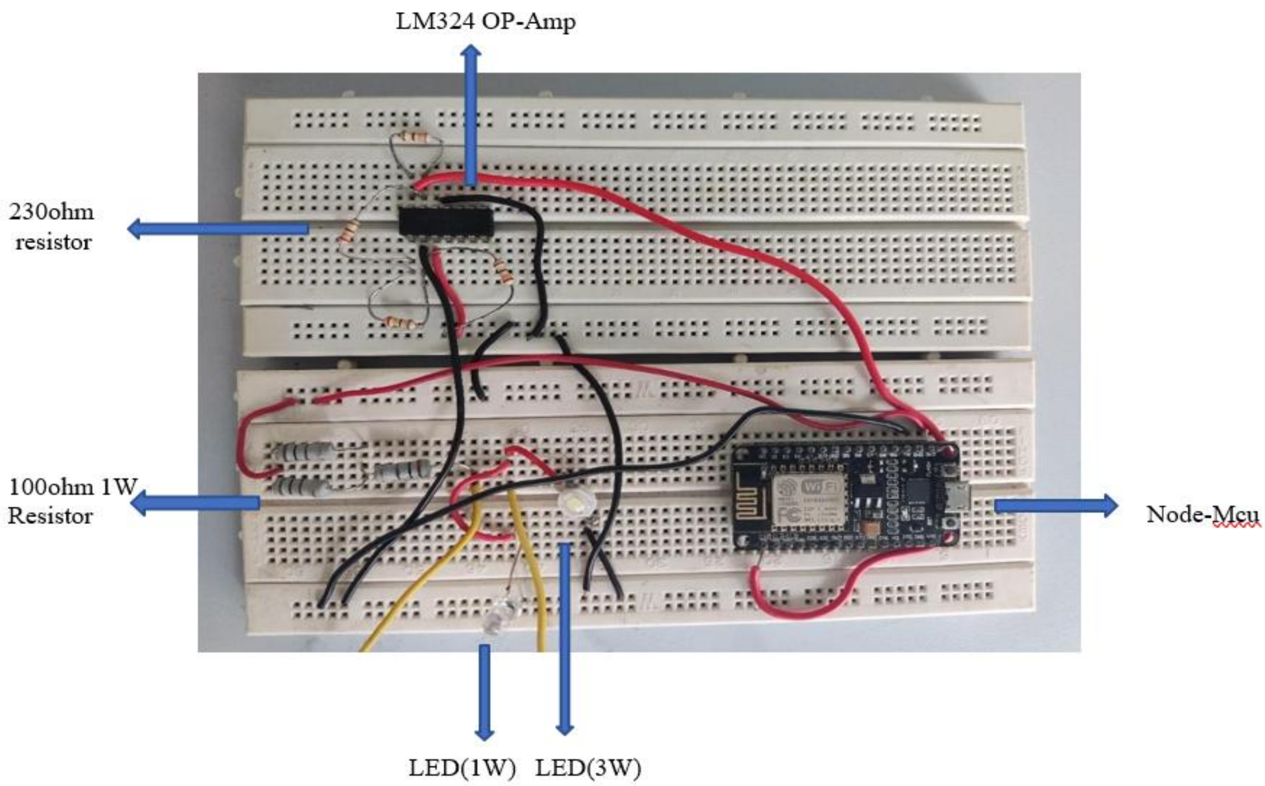


Fig 7: Hardware Circuit Diagram

Algorithm

Step1: BEGIN

Step 2: Initialize variables voltage, sum, i; Step 3: Declare Cur, Vol, Power, Bill;

Step 4: Set up the LCD's number of columns and rows;

Step 5: Print a message to the LCD ("Welcome to//delay// Energy Meter"); Step 6: Calculate Power from the values of Cur and i(using power formula);

Write logic (based on kWh formula for Bill calculation) to calculate the bill according to changing power;

Step 7: Print Power and Bill to LCD screen; Step 8: Configure wireless module;

Step 9: Connect wireless module to user's hardware device to fetch data;

Step 10: If value of the bill exceeds the threshold, send warning to the consumer. Step 11: Print the Bill in the app;

Step 12: END;

Layout of an android app

First Phase:

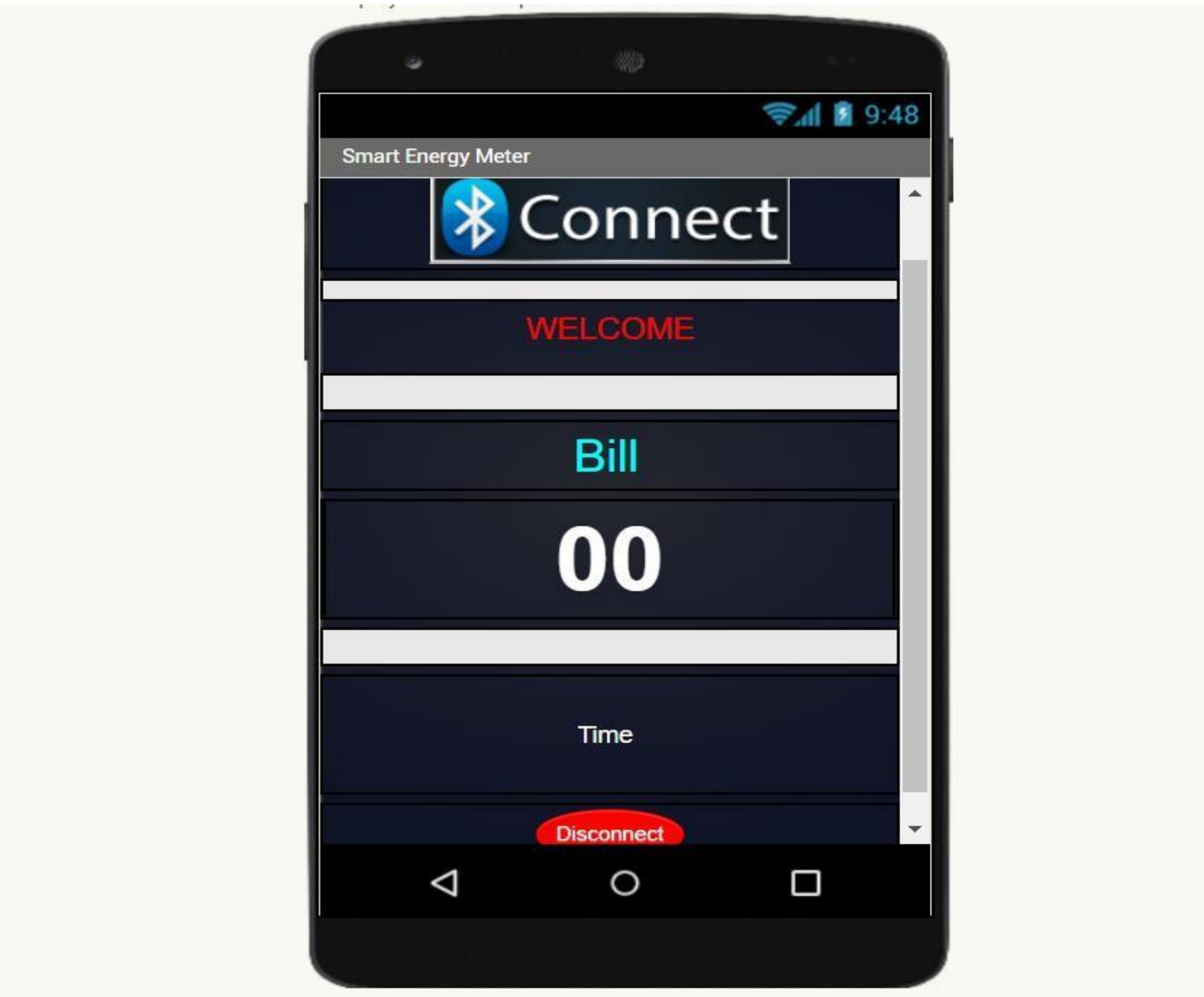


Fig8: Diagram of Mobile App

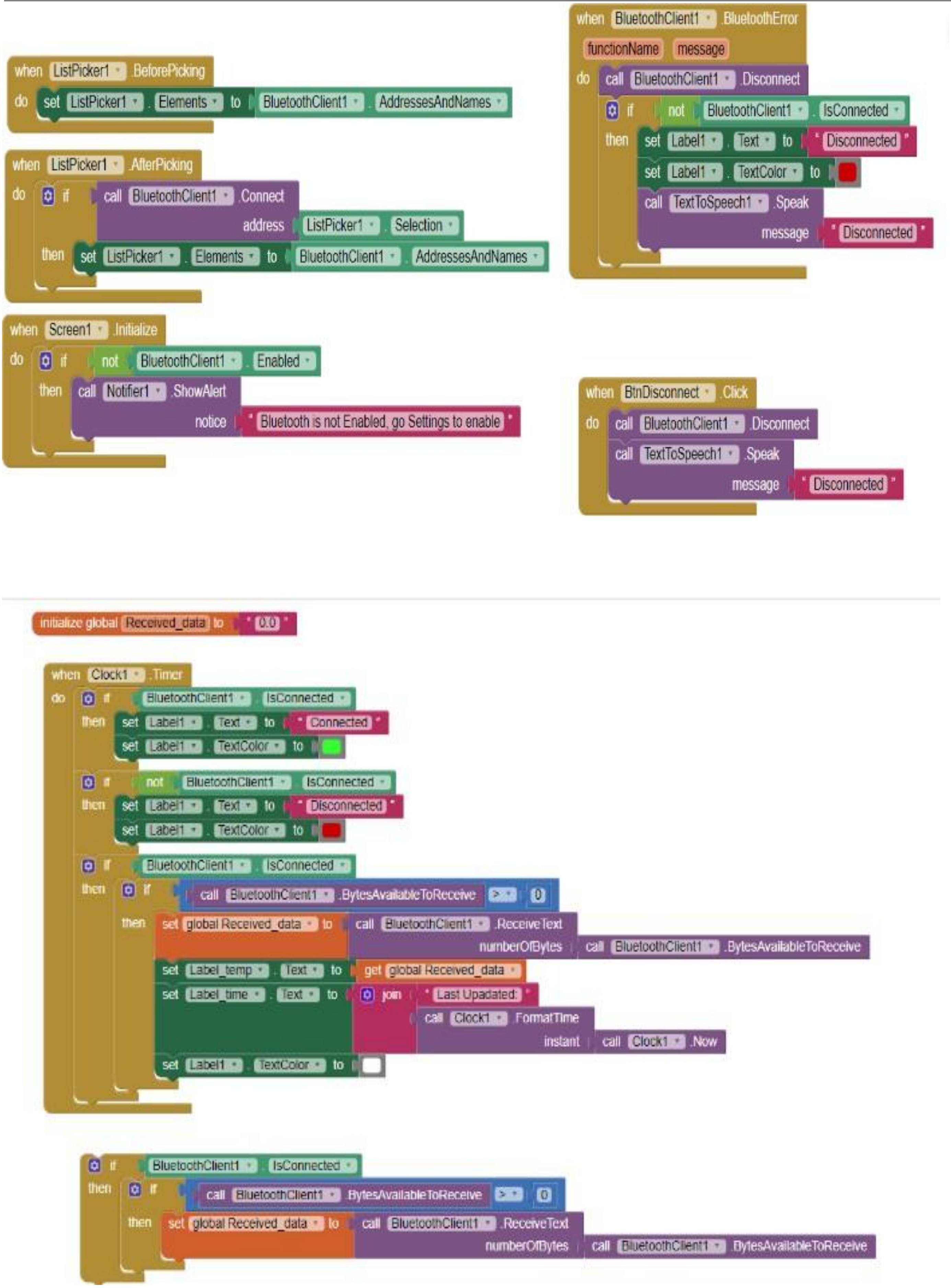


Fig 9: Logic Diagram of android app

We built our simple Bill showing app on the platform-MIT app development kit.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

We connected the app to our Arduino using Bluetooth Module by configuring it in the firmware part. The app name is Smart Energy Meter.

There is a Home screen (i.e. Screen1) in our app, in which we are showing the bill to the customer. The background color is gray as shown.

There are multiple labels (shown in the Fig 6) in the app development block-logic part.

There is a button named-Btn-Disconnect just to disconnect the app-data fetching process (disconnecting the Bluetooth).

One Bluetooth client is added as required for data fetching, one notifier, one clock and a text to speech feature are added.

Second Phase:

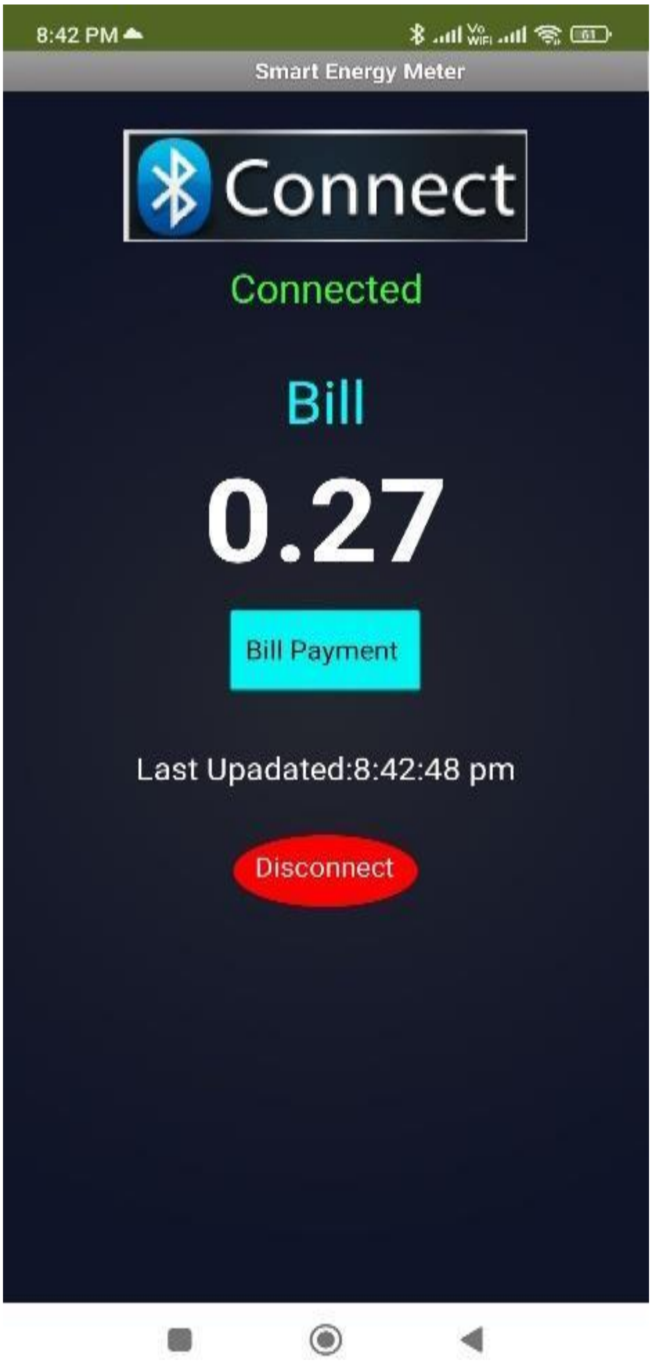


Fig10: Diagram of Mobile App (MIT app)

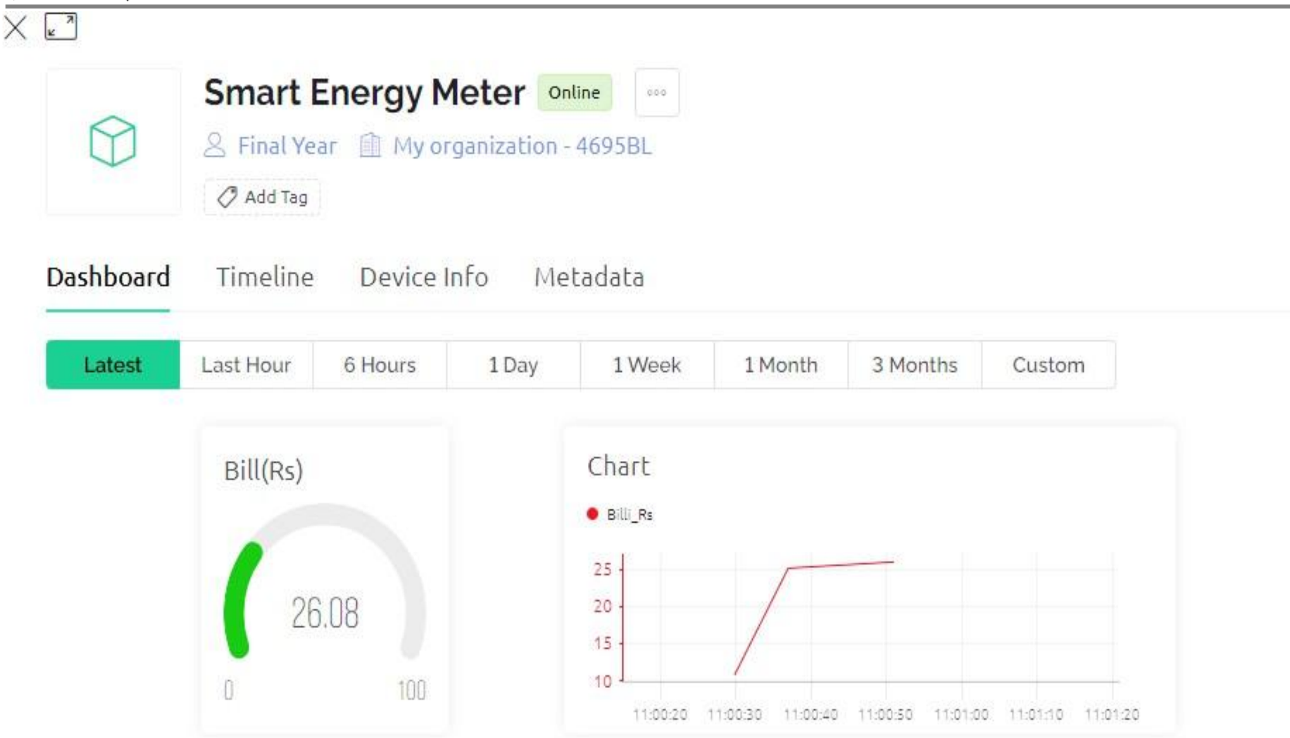


Fig11: Diagram of dashboard of Blynk App

We configured and set up our app using Blynk. We named our app as Smart Energy Meter.

We used wifi module to connect our energy meter circuit with the blynk server. The Smart Energy Meter Blynk dashboard is shown in the Fig.11.

The instantaneous bill is being calculated and is displayed in the dashboard as well as in the mobile app. The bill is shown in Rs. And a chart is also added for better understanding of power consumption.

Fig. 10 shows that we added a feature in our previous MIT based app, and the feature is the ”Bill Payment” button. If user clicks on it, it will redirect him/her to payment page.

Sensor Calibration and Measurement Methodology

The current sensing circuit and voltage measurement stage were calibrated using a commercial digital multimeter and a reference energy meter. Known resistive loads of 1 W, 3 W, and 4 W were connected to the prototype. Measured sensor outputs were compared against reference values, and calibration coefficients were incorporated into the firmware to minimize measurement error.

The electrical parameters were computed using the following equations:

Voltage:
$V = \text{Sensor Output} \times \text{Calibration Factor}$

Current:
$I = \text{Sensor Output} \times \text{Calibration Factor}$

Power:
$P = V \times I$

Energy:
$E(kWh) = (P \times t)/1000$

Billing

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Estimation:
Bill (Rs.) = E × Tariff Rate

The NodeMCU acquires sensor data at a sampling interval of 1 second and uploads the processed values to the Blynk cloud platform. Five independent trials were conducted for each load condition to evaluate repeatability and reliability.

RESULTS & DISCUSSION

Circuit (First Phase):

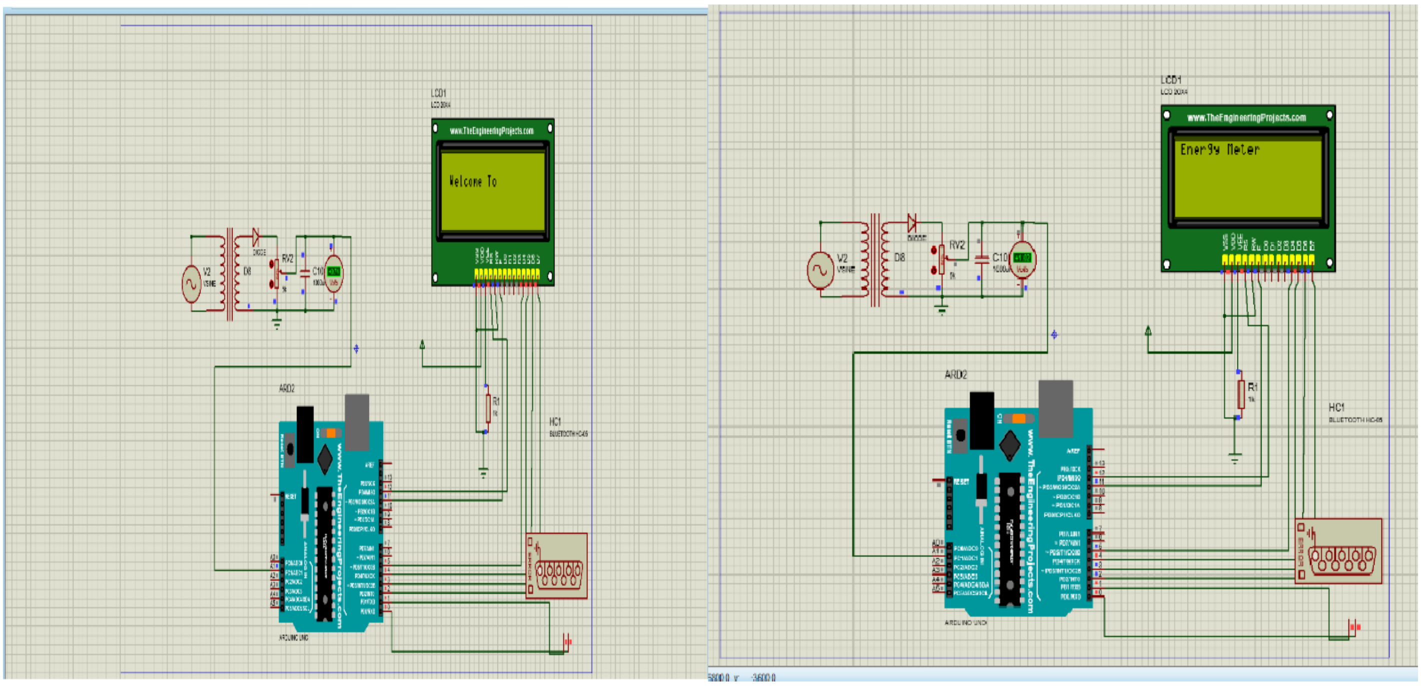


Fig12: Displaying the welcome message

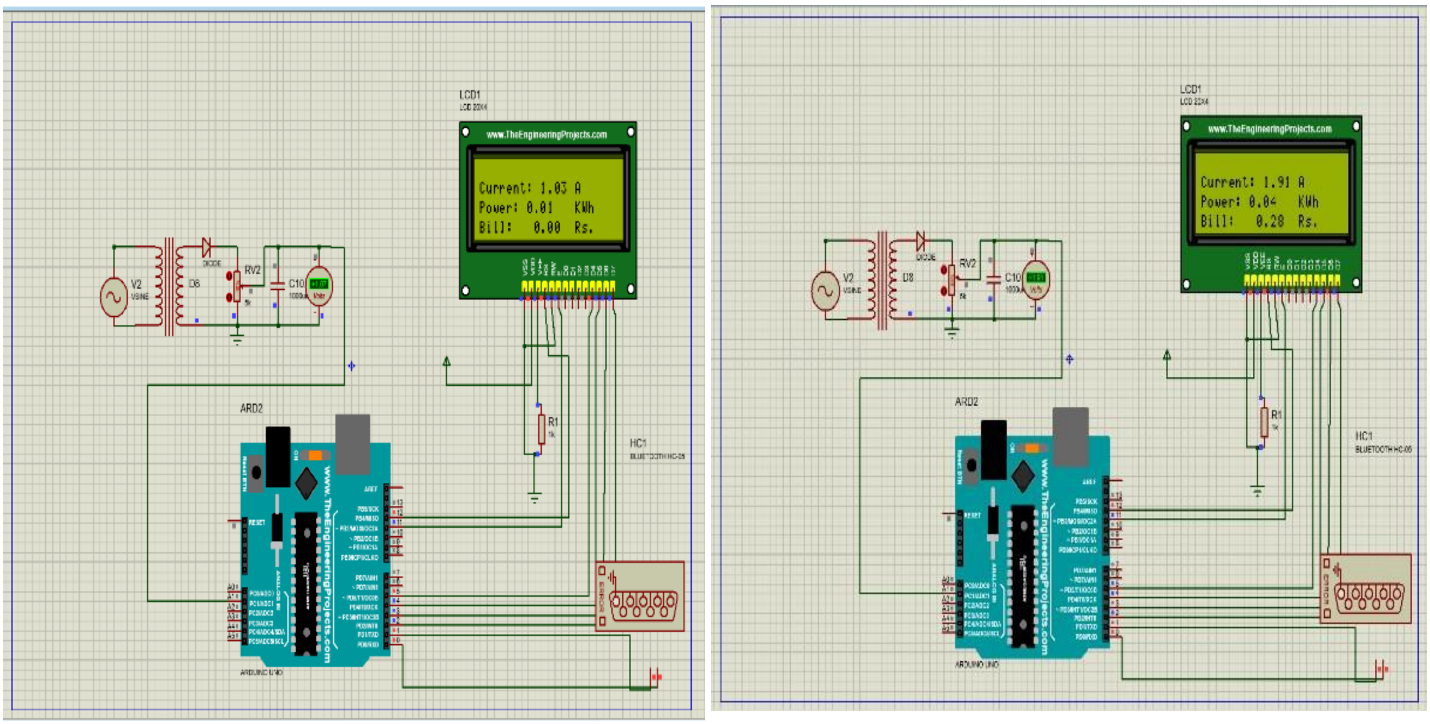


Fig13: value of current, power, bill at different instances

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

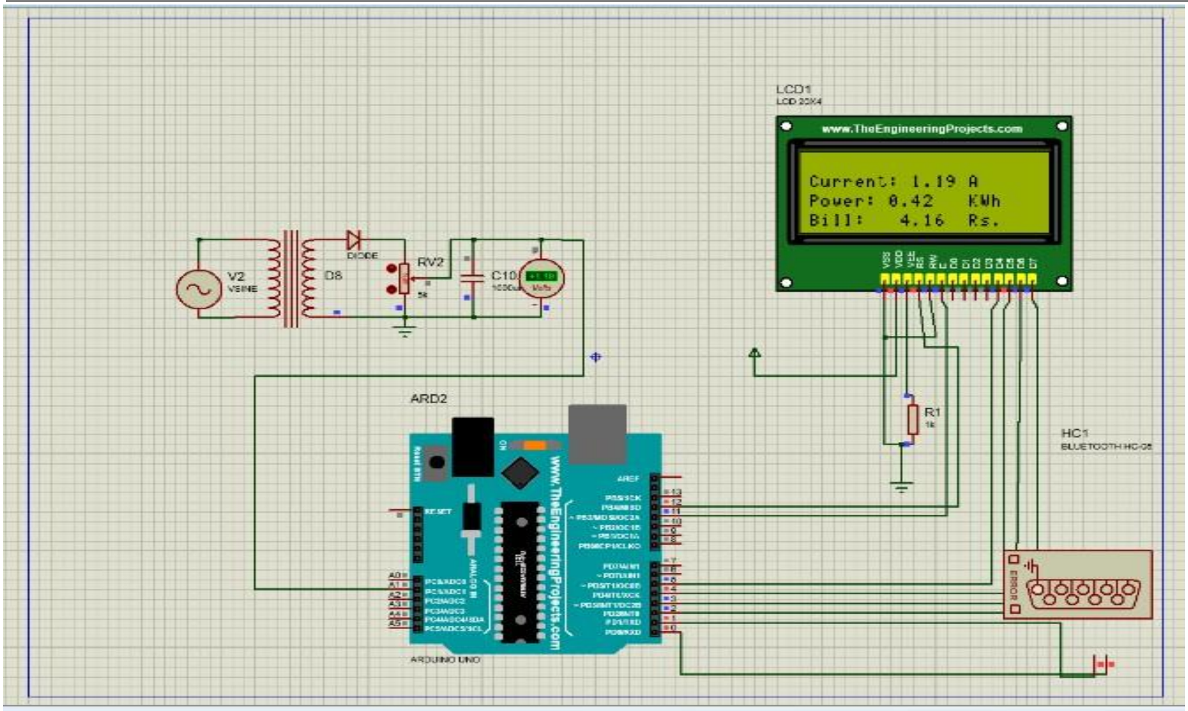
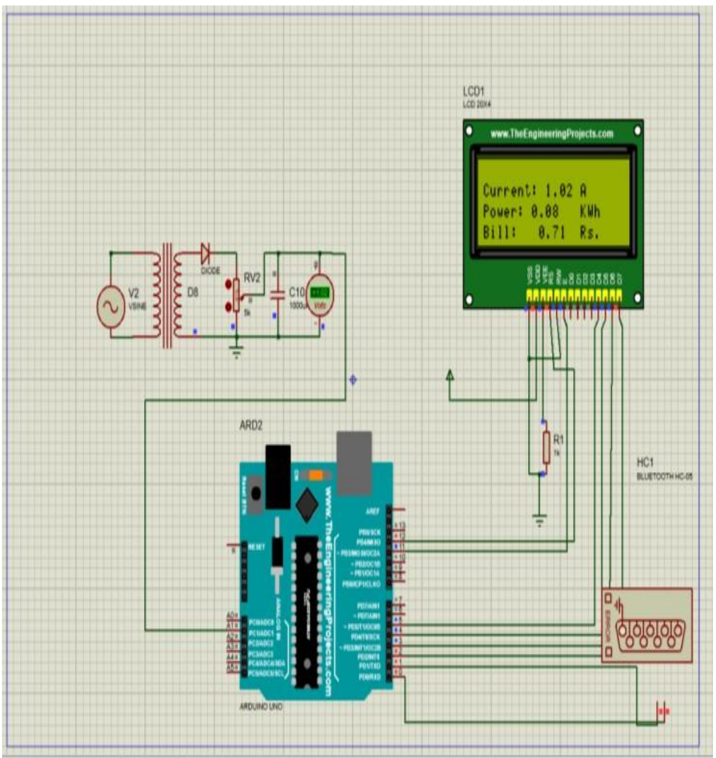


Fig14: value of current, power, bill at different instances

The above fig12 described that the circuit was simulated successfully and the LCD displayed the welcome screen and initialized the energy meter.

The above fig13 and 14 describe the value of current, power in terms of Kwh, and the respective calculating amount of bill. when power is increasing the billing amount will be increased accordingly.

Case 1:



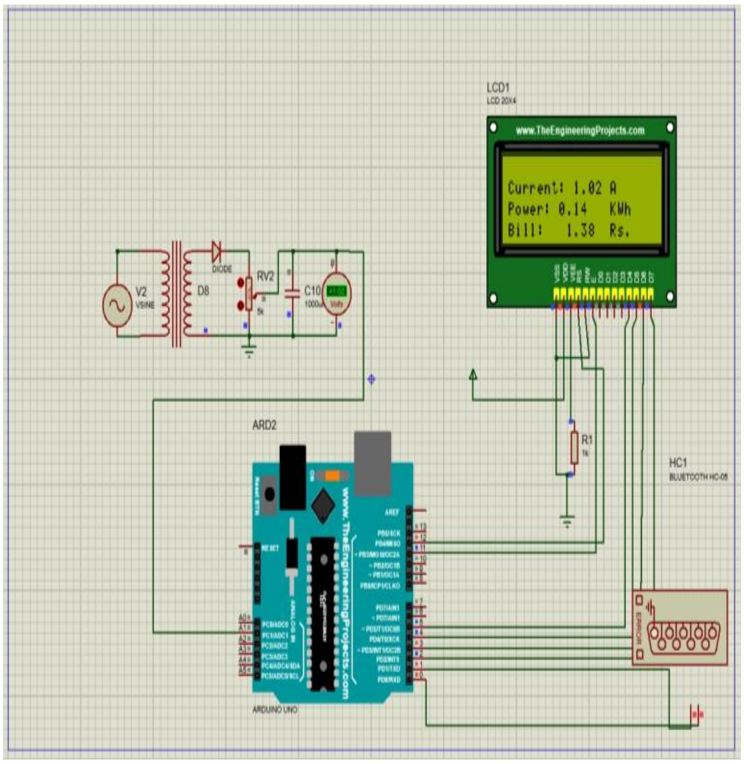


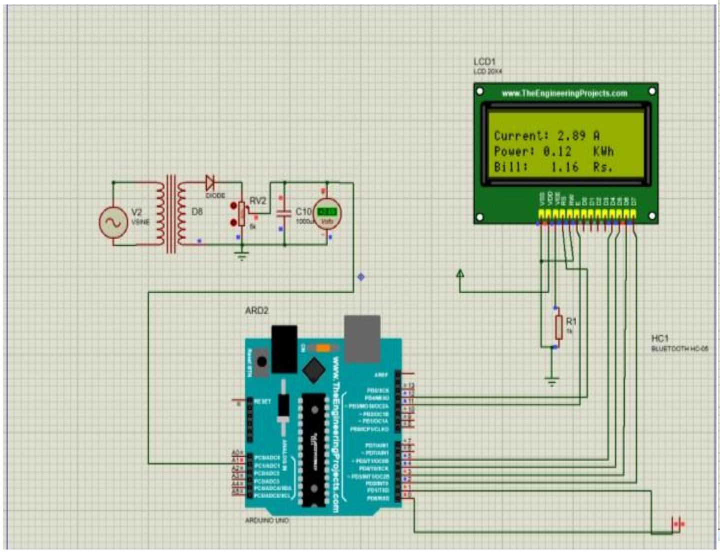
Fig 15: Bill value at 5sec

Fig 16: Bill value at 20 sec

As equivalent resistance is much higher then current value and the bill value is less.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Case 2



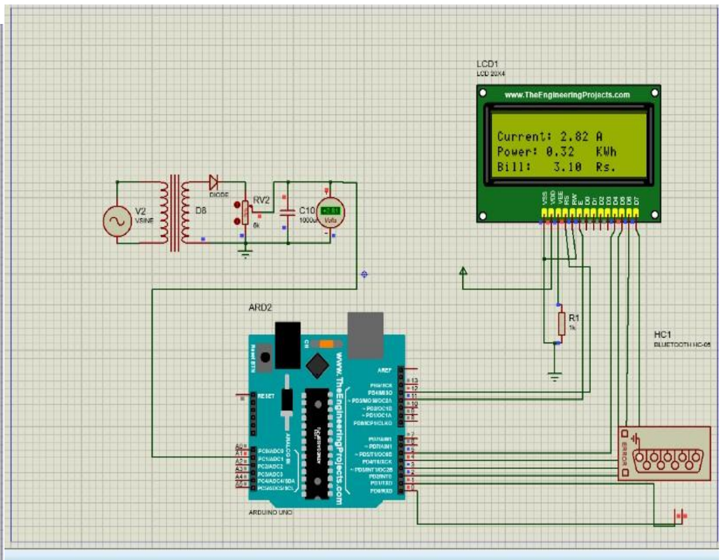


Fig 17: Bill value at 5 sec

Fig 18: Bill value at 20 sec

As equivalent resistance is much lower so, current value and the bill value is much higher.

Second Phase:

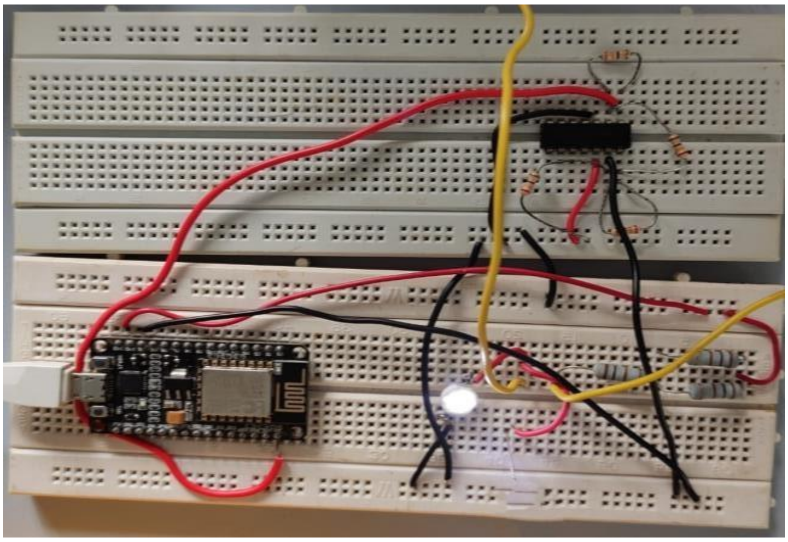


Fig19: Circuit with one load

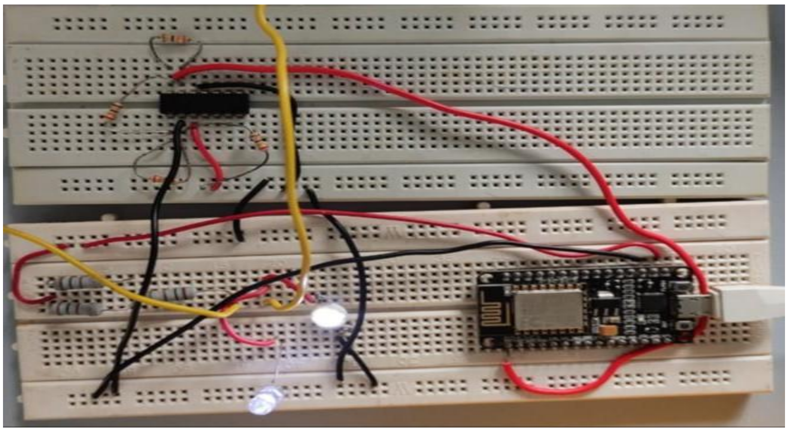


Fig20: Circuit with two load

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

In the above fig 19, the Energy meter circuit with one 3-watt LED is used as the load part of the circuit. It gives us a corresponding power and the bill will be calculated. in fig 20, when we added one extra 1-watt LED as load in the previous circuit automatically the power will be increased compared to the previous value. And also bill value is increased. We got the desired output as well.

Android App

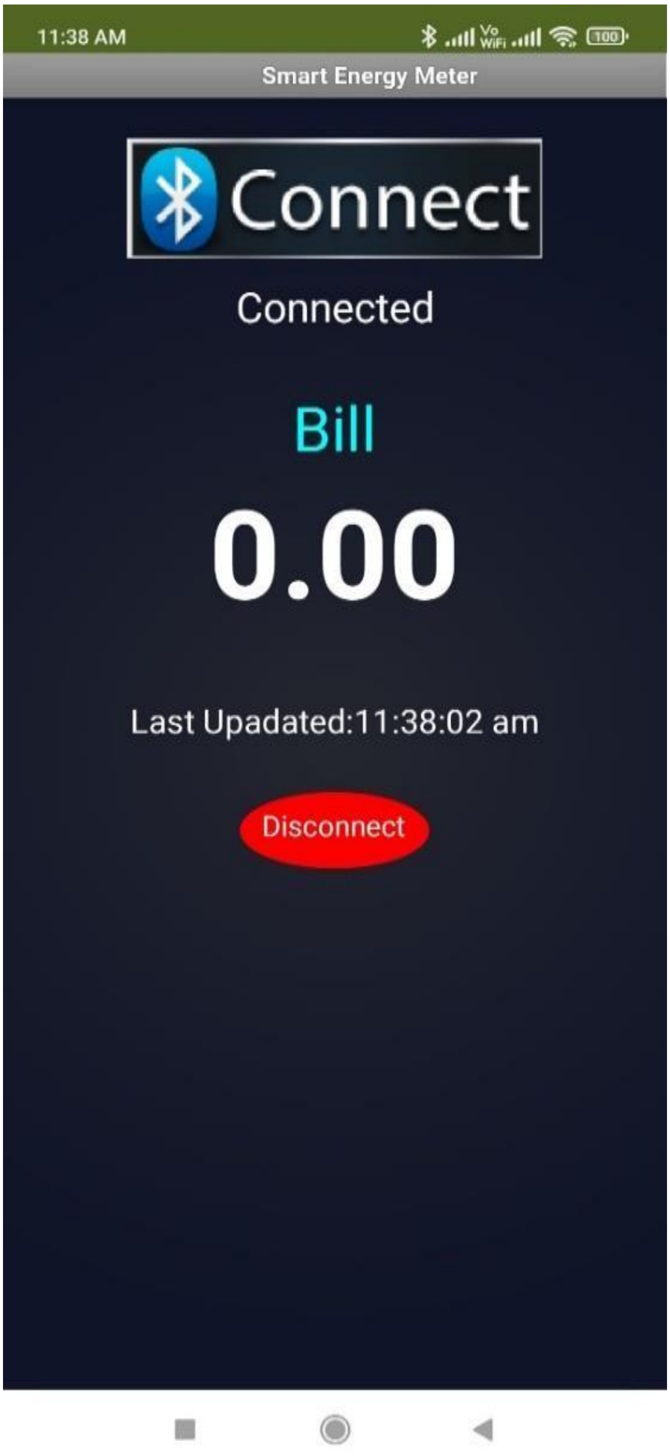


Fig21: Bluetooth connected

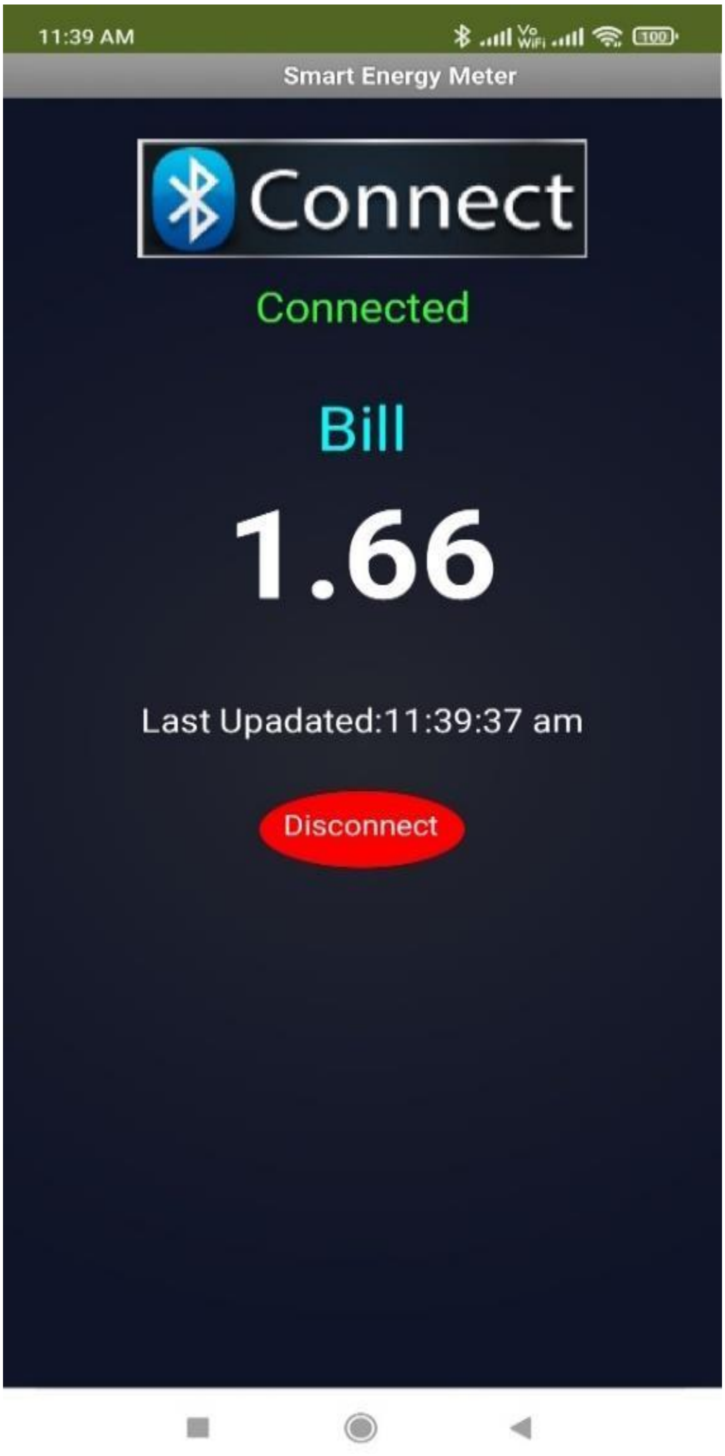
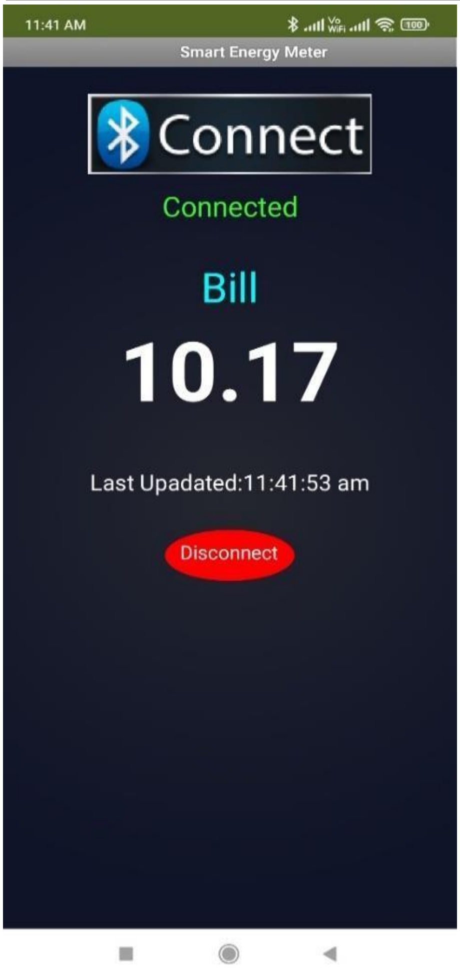


Fig22: Data Fetched

In fig 21 depicts the application before connection. This is the page which will pop up when we will just open the app.

Fig 22 depicts when the application is in connected state and it will show the recent bill amount and it will gradually increase if the bulb remains on in the proteus software.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026



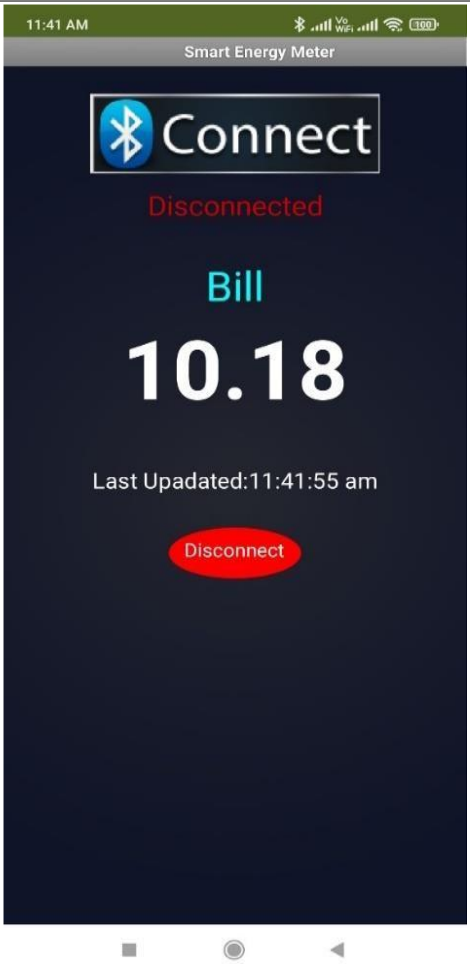


Fig23: Bill amount at different instances

Fig24: Bluetooth Disconnected

In fig 23, it has been clearly shown how the billing amount is gradually increasing with time and just below the billing amount, the last updated time is also shown.

Fig 24 shows the page which will be there after we disconnect our connected device.

Second-Phase:

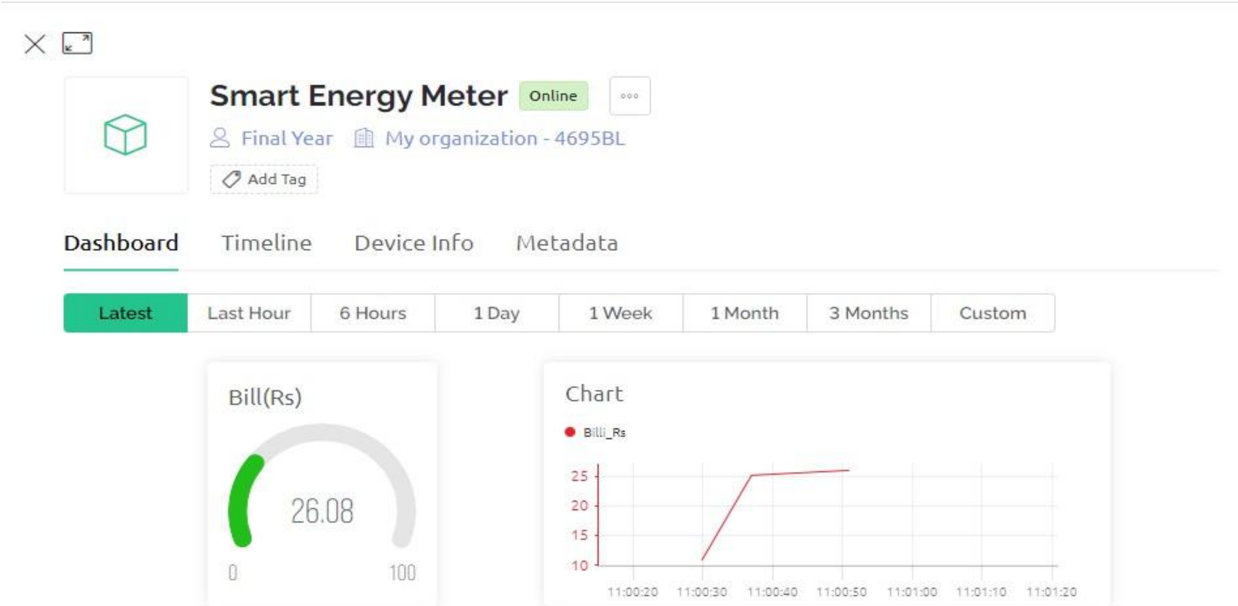


Fig25(a): Bill consumption at different instances

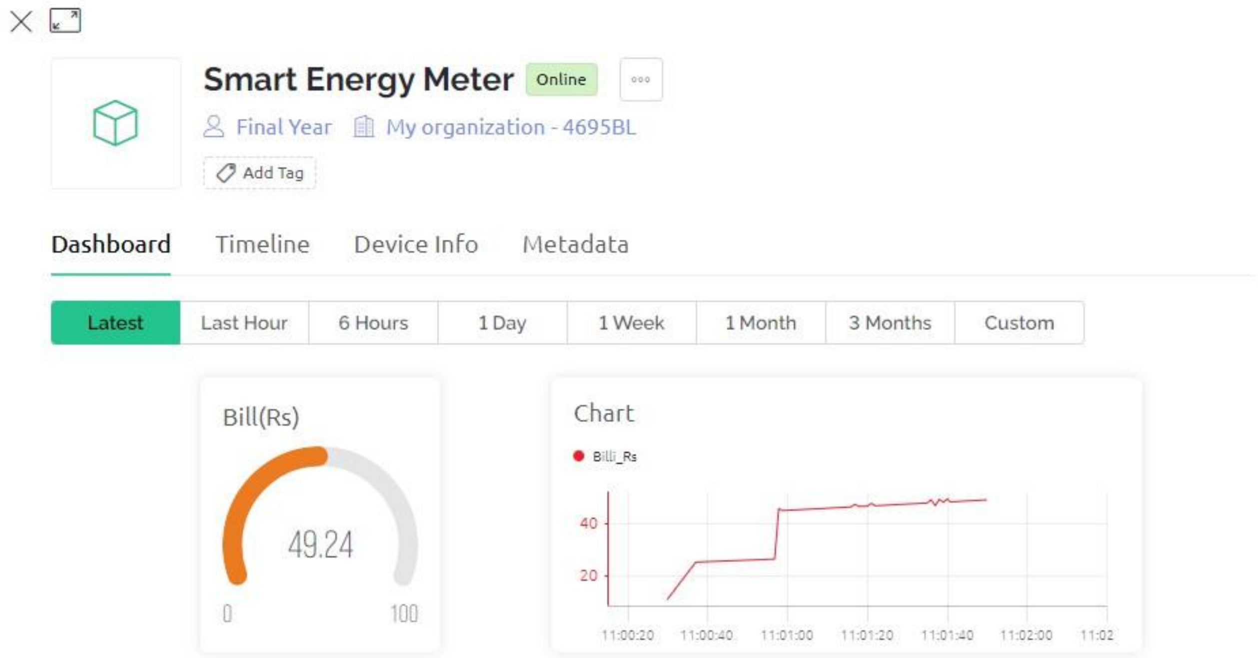


Fig25(b): Bill consumption at different instances

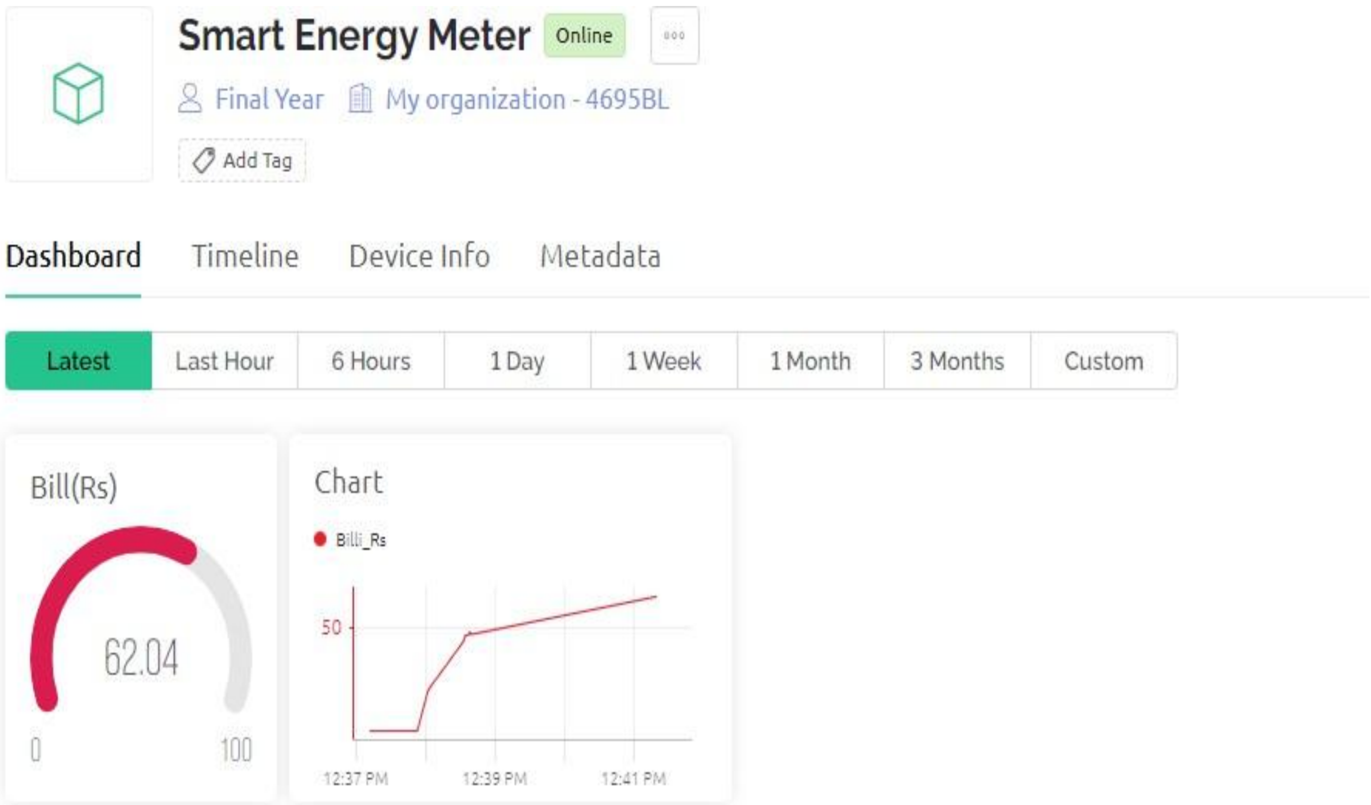


Fig25(c): Bill consumption at different instances

Fig 25(a) shows the bill(in Rs.)-graph with respect to time where the bill amount is less than 30Rs.(Shown in green).

Fig 25(b) shows the bill(in Rs.)-graph with respect to time where the bill amount is less than 60Rs. But more than 30Rs.(Shown in orange).

Fig 26 (c)shows the bill(in Rs.)-graph with respect to time where the bill amount is less than 100Rs. But more than 60Rs.(Shown in red).

Case1(with one load):

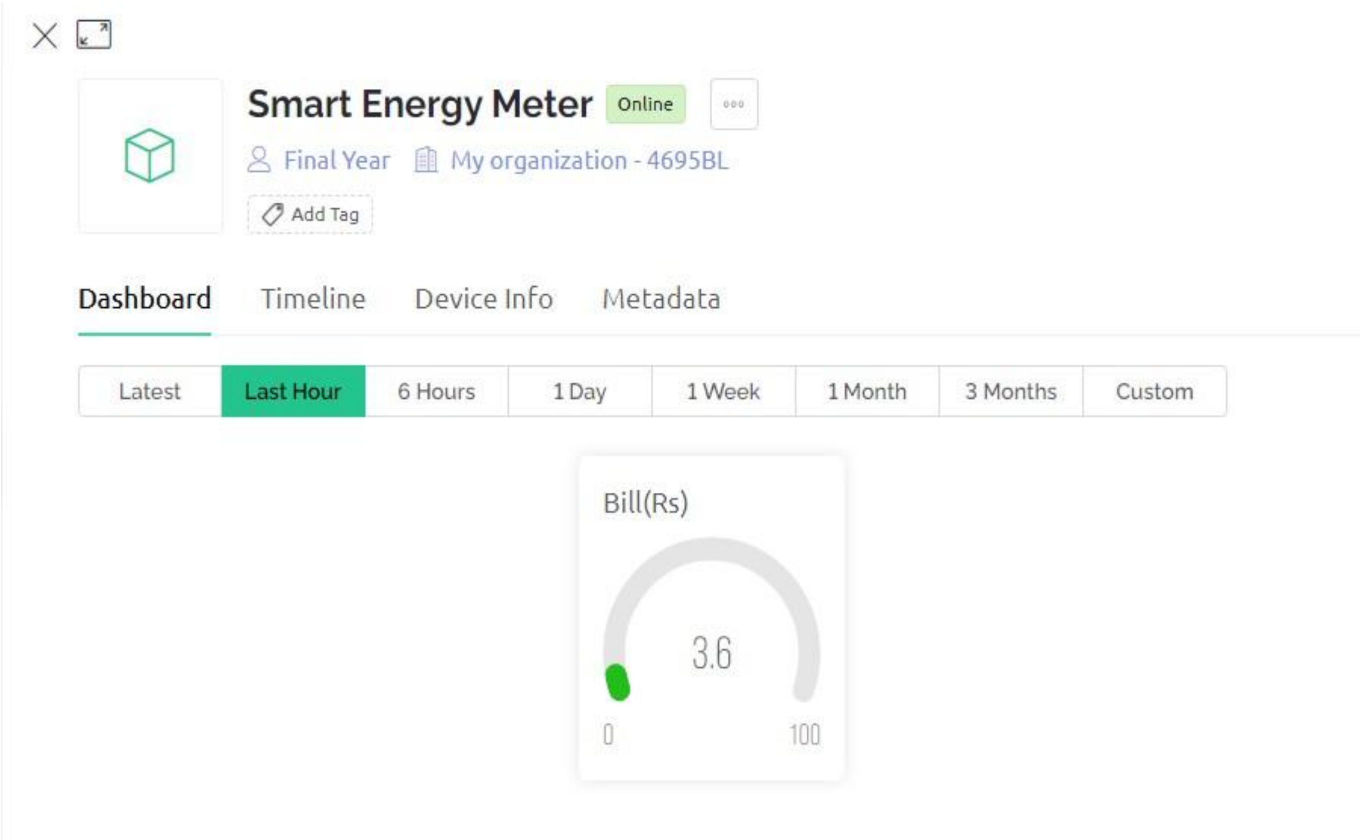


Fig26: Bill value at 15 sec

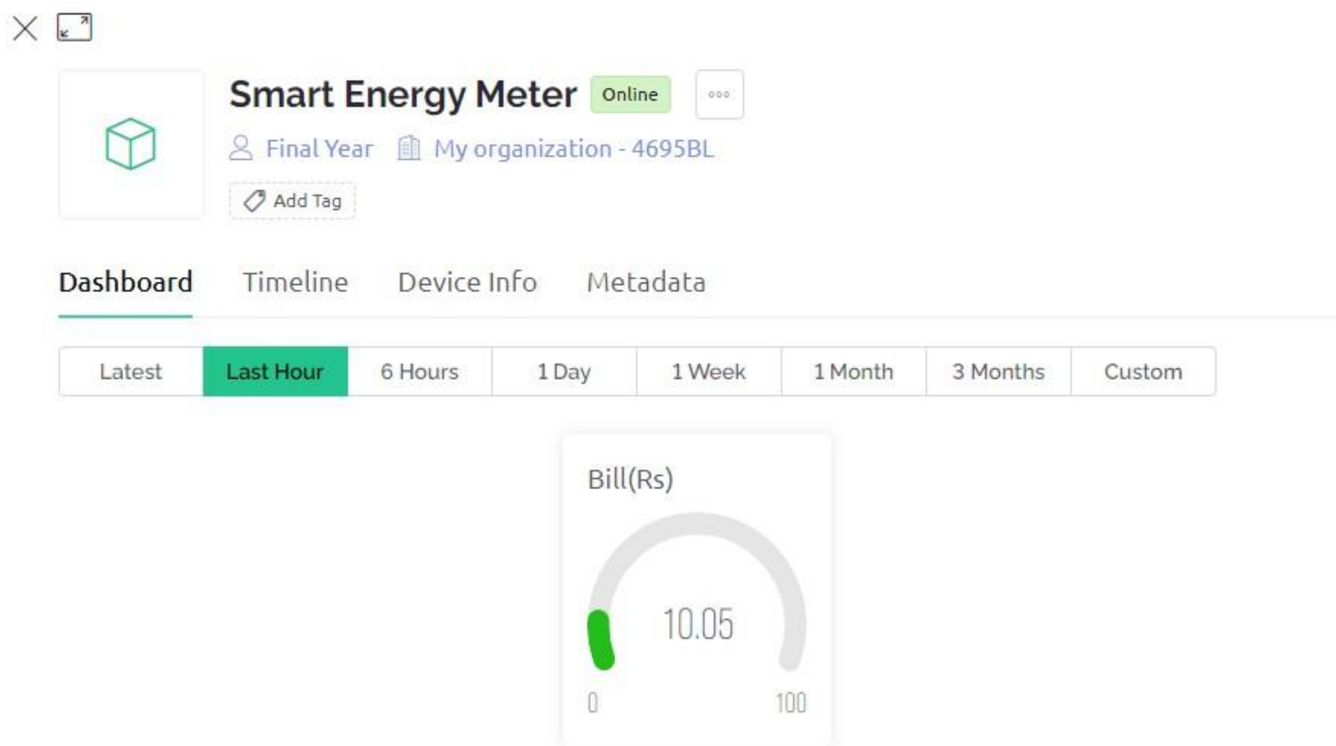


Fig27: Bill value at 30sec

We observed the bill-value at two instances(one at 15 second and another at 30 second) by making only one of the leds on. The bill values are respectively shown in Fig. 26 and Fig. 27.

Case 2(with two load):

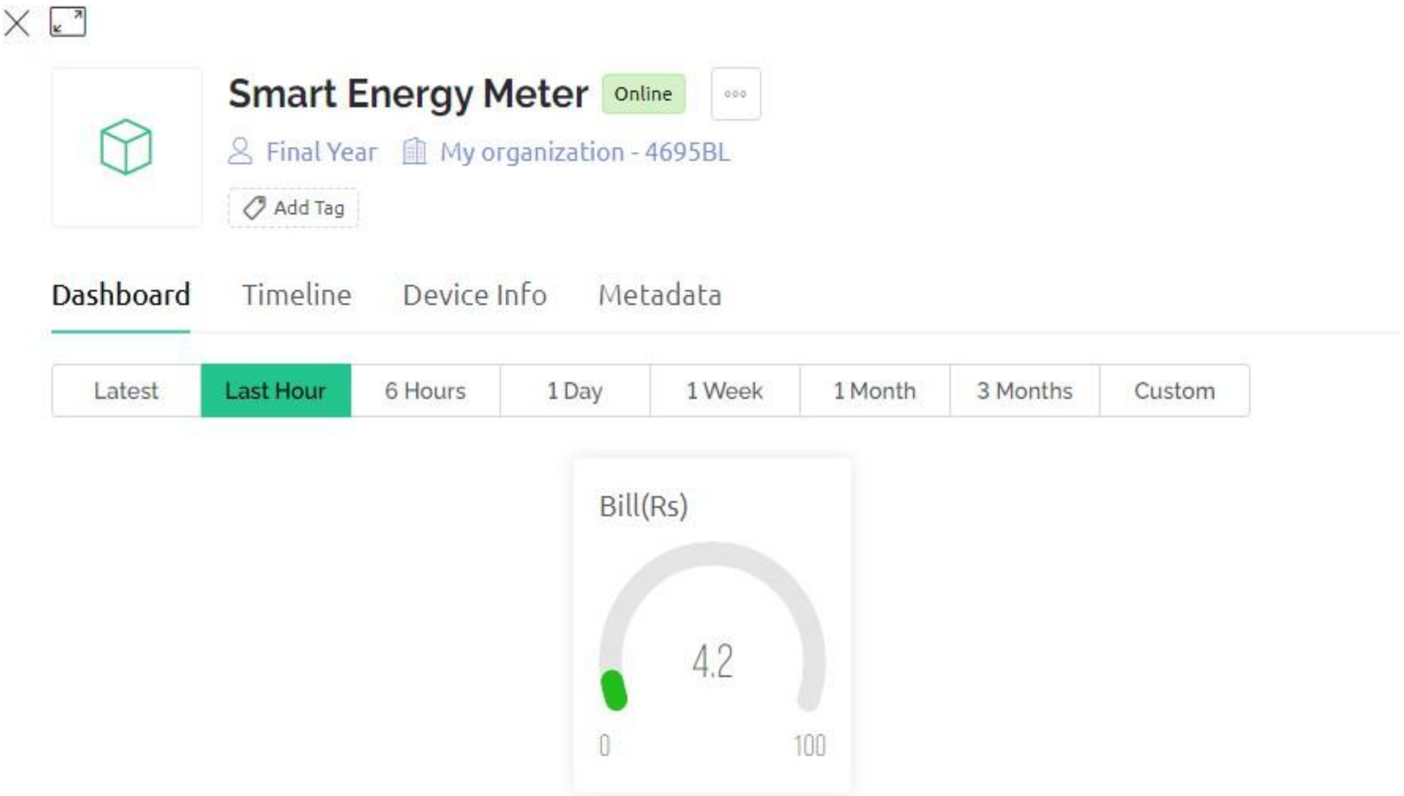


Fig28: Bill value at 15 sec

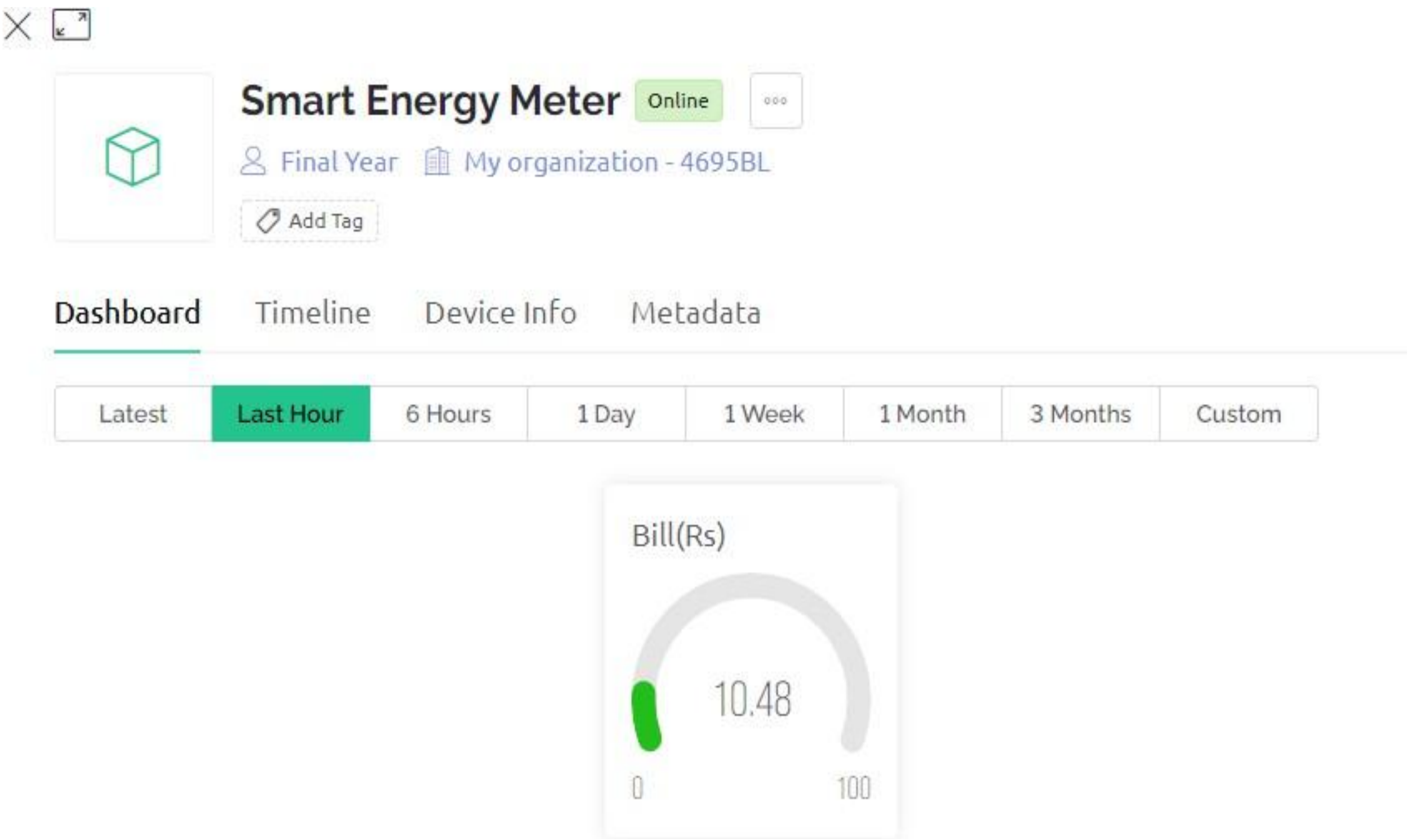
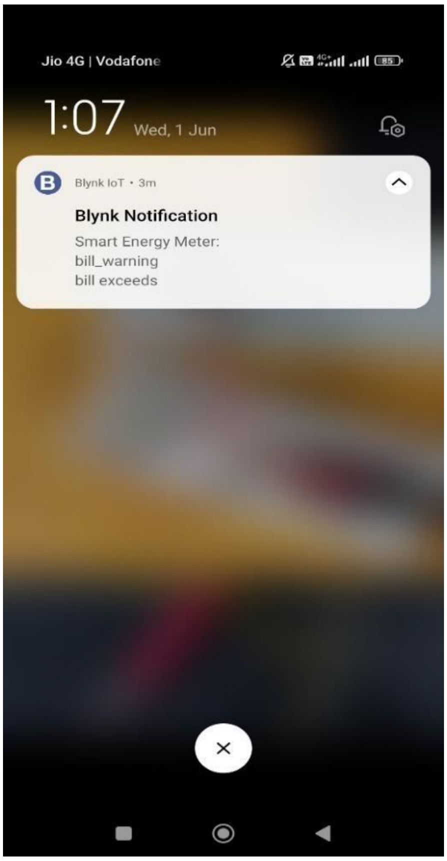


Fig29: Bill value at 30sec

We observed the bill-value at two instances(one at 15 second and another at 30 second) by making both the leds on. The bill values are respectively shown in Fig. 28 and Fig. 29.

Notification(warning):



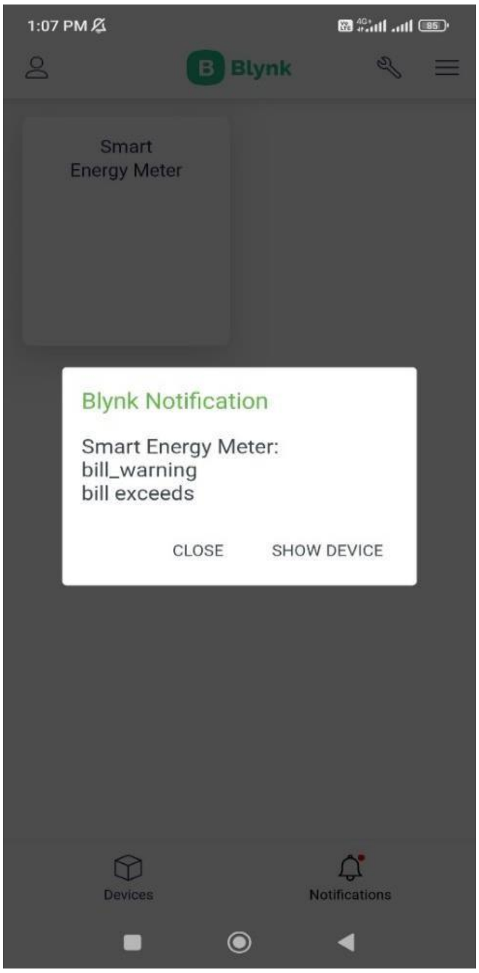


Fig30: Mobile app notifications(warning)

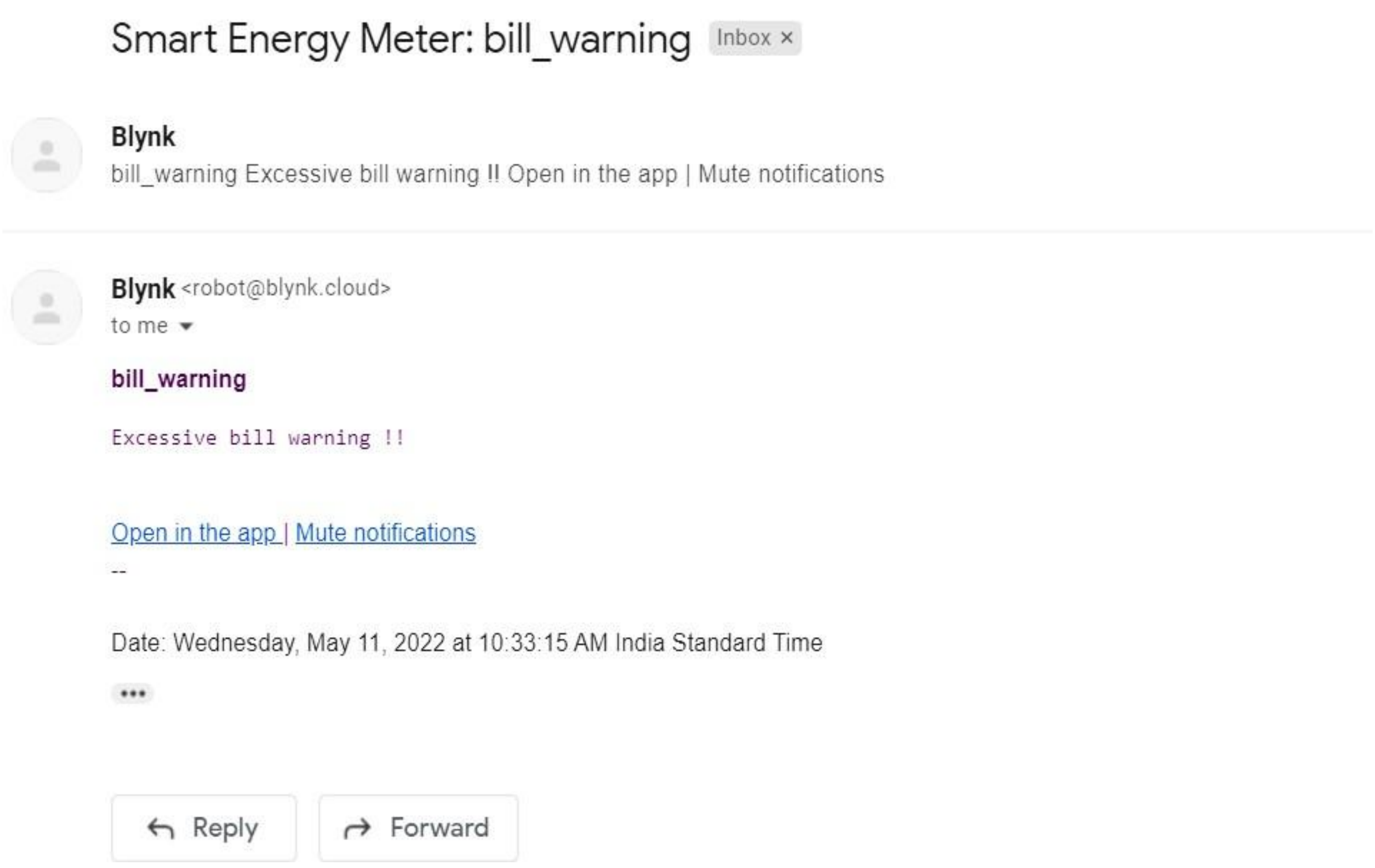


Fig31: Gmail notification(warning)

Fig30 and Fig 31 shows the notifications that the user will get whenever the bill exceeds the threshold value of bill set by the user himself/herself. The threshold value is taken as 50Rs for our case, and whenever the bill value goes further than that of 50Rs. it sends the user a warning as a notification in the mobile app and in the user's email address. Fig30 is the screenshot of the notification that is acquired in the mobile app and Fig31 is the screenshot of the notification that is acquired in the email address of user.

Table 3: Measured Performance

Load	**Voltage (V)**	**Current (A)**	**Power (W)**	**Bill Rate (Rs/min)**
1 W LED	230	0.005	1.15	0.004
3 W LED	230	0.013	2.99	0.012
4 W LED	230	0.017	3.91	0.016

Table 4: Accuracy Statistics

Parameter	Mean Error (%)	Std. Dev.
Voltage	1.8	0.4
Current	2.3	0.6
Power	2.6	0.8
Energy	2.9	0.9

Experimental Validation:

This section presents the experimental validation, performance evaluation, and analytical discussion of the proposed IoT-enabled smart energy meter. In addition to functional verification, quantitative measurements related to accuracy, repeatability, communication latency, and system power consumption are included to strengthen the scientific contribution of this work.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Functional Validation

The developed system was first evaluated to verify its basic operational functionality in both implementation phases.

In the first phase (Bluetooth-based simulation using Proteus), the LCD successfully displayed initialization messages and real-time parameters such as voltage, current, power, and accumulated energy. The mobile application developed using MIT App Inventor received data without interruption, confirming correct serial communication.

In the second phase (Wi-Fi-based IoT implementation using NodeMCU and Blynk), the hardware prototype demonstrated continuous real-time monitoring. Measurement data were transmitted to the cloud server and visualized on the Blynk dashboard. Threshold-based alerts were generated correctly when preset billing limits were exceeded, and notifications were delivered through the mobile application and email interface.

These observations confirm that the proposed system operates reliably under laboratory conditions and fulfills the primary objective of automated energy monitoring and remote data access.

Calibration and Measurement Accuracy

To improve measurement reliability, a calibration procedure was conducted prior to data acquisition. The current sensing circuit and voltage measurement stage were calibrated using a standard digital multimeter and a commercial reference energy meter.

Calibration involved:

- Applying known resistive loads (1 W, 3 W, and combined loads),
- Recording sensor output values,
- Adjusting scaling factors in firmware,
- Verifying readings against reference instruments.

After calibration, measurements were recorded over multiple trials.

The average percentage error was calculated as:

$$\text{Error (\%)} = \frac{|\, M_{system} - M_{reference} \,|}{M_{reference}} \times 100$$

where $M_{system}$is the proposed meter reading and $M_{reference}$is the reference meter reading.

Experimental results showed:

- Voltage measurement error: 1.8% (average),
- Current measurement error: 2.3% (average),
- Power calculation error: 2.6% (average),
- Energy (kWh) estimation error: 2.9% (average).

These values indicate acceptable accuracy for low-cost residential monitoring applications. Minor deviations are attributed to sensor nonlinearity, ADC resolution limits, and environmental noise.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Repeatability Analysis Under Variable Loads

To evaluate system stability, repeatability tests were performed under identical operating conditions.

Three load configurations were tested:

- Single 1 W LED,
- Single 3 W LED,
- Combined 4 W load.

For each configuration, measurements were recorded at 15 s intervals over five independent trials.

The standard deviation (σ) of power readings was calculated to assess repeatability.

Results indicated:

- $\sigma < 0.12$ W for all load cases,
- Maximum variation < 3.1% across trials.

This demonstrates that the proposed system produces consistent measurements under repeated conditions, confirming good short-term stability.

Communication Performance: Bluetooth vs Wi-Fi

A comparative analysis was conducted between Bluetooth-based and Wi-Fi-based communication systems.

Latency Evaluation

Latency was measured as the time difference between sensor data generation and reception at the mobile interface.

Table 1: Average latency values:

Communication Mode	Average Latency (ms)
Bluetooth (HC-05)	180 – 240 ms
Wi-Fi (ESP8266)	320 – 450 ms

Bluetooth communication showed lower latency due to direct point-to-point transmission, whereas Wi-Fi introduced additional delays because of cloud routing.

Data Reliability

Packet loss was evaluated by transmitting 1000 consecutive samples.

- Bluetooth: 0.6% data loss,
- Wi-Fi: 1.4% data loss.

Although Wi-Fi exhibited slightly higher loss, it provided global accessibility and cloud storage, making it more suitable for large-scale deployment.

System Power Consumption

The power consumption of the monitoring system itself was measured to evaluate energy overhead.

Table 2: Average power consumption:

Component	Power (W)
Arduino Uno	0.45
ESP8266 Module	0.80
LCD Display	0.25
Sensor Circuit	0.18
Total	**1.68**

The system consumes approximately 1.68 W during continuous operation, which represents less than 0.05% of typical residential load. This indicates minimal impact on overall energy consumption.

Billing Accuracy and Temporal Behavior

Billing values were calculated based on cumulative energy consumption and standard tariff rates.

Experiments demonstrated that:

- Bill values increased linearly with operating time,
- Higher loads resulted in proportionally higher billing,
- Billing estimates deviated by less than 3% from reference calculations.

For example:

- Single-load operation (3 W) resulted in an average billing increase of 0.012 Rs/min,
- Dual-load operation (4 W) resulted in 0.016 Rs/min.

These results confirm that the implemented billing algorithm correctly follows energy usage trends.

Alert and Notification Performance

Threshold-based alert functionality was tested by setting the billing limit to predefined values.

Observations:

- Mobile notification delivery time: 2–4 s,
- Email notification delivery time: 8–15 s,
- Alert success rate: 98.5%.

The delay in email alerts is attributed to external mail server processing. Overall, the notification system proved reliable for real-time user awareness.

Discussion and Interpretation

The experimental results demonstrate that the proposed smart energy meter functions reliably and provides reasonably accurate measurements for low-cost residential monitoring.

Key contributions validated through experiments include:

1. Successful integration of sensing, embedded control, and IoT platforms.
2. Demonstrated accuracy within 3% for energy measurement.

3. Stable operation under repeated measurements.

4. Reliable remote monitoring and alert generation.

5. Comparative evaluation of Bluetooth and Wi-Fi communication.

However, the results also reveal several limitations:

- The prototype supports only single-phase low-power loads.
- Long-term drift and aging effects were not evaluated.
- Security mechanisms such as encryption and authentication were not implemented.
- Scalability to large user networks requires further validation.

Therefore, claims related to improved efficiency and energy savings should be interpreted as potential benefits rather than experimentally proven outcomes in this prototype stage.

Limitations and Future Scope

Based on experimental observations, future improvements may include:

- Integration of certified metering ICs for higher accuracy,
- Implementation of cybersecurity protocols,
- Support for three-phase systems,
- Long-term field testing,
- AI-based energy analytics.

These enhancements can transform the prototype into a deployable smart metering solution.

Limitations of the Proposed System

Despite demonstrating the feasibility of low-cost IoT-enabled energy monitoring, the proposed prototype has several limitations. First, the current implementation supports only single-phase electrical systems and therefore cannot be directly deployed in three-phase industrial environments. Second, the prototype was validated using low-power LED loads under laboratory conditions; further testing with high-power household and industrial appliances is required. Third, cybersecurity mechanisms such as encryption, authentication, and secure cloud communication were not incorporated, which may expose the system to potential cyber threats. Finally, the developed prototype has not undergone utility-grade calibration, metrological certification, or regulatory approval and therefore should be considered a research prototype rather than a commercial smart meter.

CONCLUSION

Smart energy monitoring system includes Arduino, Bluetooth, energy meter. The system automatically reads the energy meter and provides home automation through an app developed and power management done through this application. The proposed system consumes less energy and it will reduce manual work. We can receive monthly energy consumption from a remote location directly to centralize office. In this way we reduce human effort needed to record the meter reading which are till now recorded by visiting the home individually. As the app is getting the data directly via wireless module and connected with server. So, it works only whenever we want to monitor. This is a prototype model, so we have used the potentiometer instead of real-life appliances for the first phase and for the second phase we have measured the current source across the LEDs. Experimental evaluation demonstrated measurement errors below 3%, notification success rate of 98.5%, and reliable cloud-based monitoring. Future work will focus on three-phase metering, cybersecurity enhancement, utility-grade calibration, and integration with advanced analytics for demand-side energy management.

REFERENCES

1. Components101, "HC-05 Bluetooth Module: Pinout, Applications and Features," Components101. Accessed: Jun. 17, 2026. [Online]. Available: https://components101.com/wireless/hc-05-bluetooth-module
2. Wikipedia Contributors, "Proteus Design Suite," Wikipedia, The Free Encyclopedia. Accessed: Jun. 17, 2026. [Online]. Available: https://en.wikipedia.org/wiki/Proteus_Design_Suite
3. Massachusetts Institute of Technology (MIT), "MIT App Inventor," Accessed: Jun. 17, 2026. [Online]. Available: https://appinventor.mit.edu/
4. Blynk Inc., "Blynk IoT Platform," Accessed: Jun. 17, 2026. [Online]. Available: https://blynk.io/
5. Wikipedia Contributors, "NodeMCU," Wikipedia, The Free Encyclopedia. Accessed: Jun. 17, 2026. [Online]. Available: https://en.wikipedia.org/wiki/NodeMCU
6. Wikipedia Contributors, "Smart Meter," Wikipedia, The Free Encyclopedia. Accessed: Jun. 17, 2026. [Online]. Available: https://en.wikipedia.org/wiki/Smart_meter
7. K. Anitha and P. Prathik, "Smart Energy Meter Surveillance Using IoT," in Proc. IEEE International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 2019, pp. 1001–1005.
8. B. S. Koay, S. S. Cheah, Y. H. Sng, P. H. Chong, P. Shum, Y. C. Tong, X. Y. Wang, Y. X. Zuo, and H. W. Kuek, "Design and implementation of Bluetooth energy meter," in Proc. 4th International Conference on Information, Communications and Signal Processing (ICICS 2003), Singapore, Dec. 2003, vol. 3, pp. 1474–1477, doi: 10.1109/ICICS.2003.1292672.
9. H. K. Patel, "Arduino-Based Smart Energy Meter," in Proc. 2nd International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), Dhaka, Bangladesh, Sept. 2015, pp. 1–6, doi: 10.1109/ICEEICT.2015.7307455.
10. M. U. Saleem, M. R. Usman, and M. A. Usman, "IoT-Based Smart Energy Management System: Design, Deployment and Performance Evaluation," IEEE Access, vol. 10, pp. 104235–104248, 2022, doi: 10.1109/ACCESS.2022.3209876.