http://dx.doi.org/10.21622/ACE.2026.06.1.1955

A REMOTE CONTROLLED IOT SMART METER MONITORING SYSTEM WITH THEFT DETECTION

Ayoola E. Akindele[1], Ivan Kholodilin[2] and Afolabi Awodeyi[3]

[1,2] South Ural State University, Department of Mechatronics and Robotics Engineering, Chelyabinsk, Russia

[3] Southern Delta University, Department of Computer Engineering, Ozoro, Delta State, Nigeria

Emails: {ayoola.akindele@ieee.org, kholodilinii@susu.ru, awodeyiai@dsust.edu.ng}

Received 12 February 2026 - Accepted 21 March 2026 - Published 10 June 2026

ABSTRACT

This research addresses the challenges of energy administration and security in developing nations. It proposes an Internet of Things (IoT)-based smart meter monitoring system that integrates security controls to prevent theft and allows for remote monitoring of energy usage. The system utilizes an Atmega microcontroller for connectivity, a GSM module for sending data, a non-invasive AC current sensor for accurate readings, and a user interface for data visualization and control. Upon detecting theft, the system sends an alert to the user and disconnects the power supply. Users can also remotely monitor energy consumption and turn off the meter through the GSM connection.

Keywords: *ESmart Meter; Energy Monitoring; Internet of things; Atmega Microcontroller; Sensing system; Current Sensor; GSM interaction*

1. INTRODUCTION

Technology is reshaping our world, and the Internet of Things (IoT) is at the heart of this change. IoT connects everyday devices from smart watches to traffic lights, allowing them to share data effortlessly. This isn't just convenience; it's a revolution. In healthcare, wearables monitor vitals in real time, helping doctors catch issues early. Smart cities use IoT to ease traffic, cut energy waste, and even manage trash collection efficiently. Meanwhile, military systems leverage IoT for better situational awareness, and retailers use it to personalize shopping and track inventory. Factories automate processes, and oil pipelines detect leaks, all thanks to IoT. One exciting IoT innovation is the smart energy meter. Beyond tracking usage, it enhances security by detecting tampering, lets users monitor consumption remotely, and even allows control of appliances via a simple app. This isn't just about saving money; it's about smarter, safer energy use. By merging IoT with energy solutions, we're stepping into a future where sustainability and efficiency go hand in hand. The potential is huge, and the best part? It's just the beginning.

2. REVIEW OF RECENT SMART METER AND ELECTRICITY THEFT DETECTION SYSTEMS

Recent research on IoT-based smart energy meters has focused on improving energy efficiency, billing accuracy, and theft detection. For example, the research by Yaghmaee and Hejazi in 2018[1] demonstrated an IoT-based metering system for real-time monitoring, but it lacked integrated theft prevention mechanisms. Also, Barman et al. in 2018 [8] proposed a smart meter for efficient energy utilization, but their system did not incorporate automated disconnection upon detecting anomalies. Tahir et al. in 2022 [3] introduced blockchain-based smart metering to enhance data security, although the approach increases system

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

complexity and cost. In contrast, simpler GSM-based systems, such as the research of Agnihotri in 2022 [9], focus primarily on communication without advanced detection mechanisms. More recent studies have explored electricity theft detection using IoT and machine learning techniques, but many of these systems rely on complex data-driven models that require extensive datasets and computational resources, which limit their applicability in resource-constrained environments.

When comparing, the proposed system adopts a hardware-driven detection approach using tamper sensors and current monitoring that offers lower computational requirements, faster response time, and ease of implementation in real-world environments.

3. SYSTEM ARCHITECTURE AND THEFT DETECTION ALGORITHM

3.1. SYSTEM ARCHITECTURE

The proposed system follows a modular architecture consisting of four primary layers:

a) Sensing Layer

Includes the AC current sensor and tamper switch for detecting energy consumption and unauthorized access.

b) Processing Layer

The ATmega328p microcontroller processes sensor inputs and executes decision-making algorithms.

c) Communication Layer

i) ESP8266 module for cloud-based monitoring
ii)GSM module for SMS alerts and remote control

d) Control Layer

This is the relay module responsible for disconnecting power upon theft detection.

3.2. THEFT DETECTION ALGORITHM

The theft detection mechanism is based on threshold monitoring and tamper event detection. The algorithm's operation is as follows:

a) Initialize system parameters and communication modules

b) Continuously read current sensor values

c) Monitor tamper switch status

d) Compare the measured current with the expected consumption pattern

e) If the tamper signal is triggered OR abnormal current deviation is detected:

i) Flag as theft event
ii)Send SMS alert via GSM module
iii) Upload event to the cloud via Wi-Fi
iv) Activate the relay to disconnect the supply

f) Else:

i) Continue normal monitoring

g) Repeat the process in a real-time loop

4. SYSTEM ANALYSIS AND DESIGN

This section delves into the intricacies of the research, encompassing the technological and systemic approaches employed. It details the justifications for the chosen methodologies and components, highlighting their advantages and suitability in achieving the research objectives. The section proceeds with an extensive description of:

a) Hardware Design and Algorithms: This includes detailed explanations of the hardware elements, associated algorithms, computations performed, and development environments used to manage the device.

b) Software Development: This explores the programming languages, frameworks, and functionalities employed in the developed software.

c) Sequencing Methods: This explains the essential sequencing methods implemented to ensure proper functioning and interaction of all system components.

SYSTEM ANALYSIS

The design and implementation of the IoT Smart Energy Meter Monitoring and Energy Theft Sensing System comprises two key sections:

a) Hardware Components: This section focuses on the selection, integration, and functionalities of the various hardware components utilized in the system. Prior to integration with the ATmega microcontroller, a thorough analysis of each individual hardware component is conducted, elucidating its role and functionalities within the system.

b) Software Development: This section explores the software specifically developed for the system. It delves into the programming languages, frameworks, and functionalities employed to achieve the desired outcomes.

Table 1, summarizing the primary elements involved in the research project, is included within this section. This table provides a concise overview of the essential hardware and software components utilized in the system design and implementation.

Table 1: Primary elements of this research

COMPONENT No.	SYSTEM COMPONENT UNIT	PARTS AND DEPICTIONS
1	Power Unit	i. 12V DC lithium-ion battery ii. DC-DC Multi-output Buck Converter (3.3V/5V/9V/12V) iii. A step-down transformer 240/12V 50Hz iv. Optocoupler PC817 IC
2	Sensing & Control Unit	i. Tamper sensor ii. AC Current Sensor iii. ESP8266 Transmitting Module iv. Input voltage of 12V v. ATMega328p microcontroller vi. GSM Module vii. 12V DC supply (Battery) viii. 5V relay
3	Communication Unit	i. ESP8266 Transmitting Module GSM Module
4	Software	i. Arduino IDE (C++)

The system comprises several interconnected components fulfilling specific functions:

i) Sensing:

Analog Sensors: The GSM module and tamper switch are connected to specific analog pins of

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

the microcontroller, enabling them to provide analog outputs between 0 and 1023, readable by the system.

ii) Communication:

Digital Pins: The ESP8266 module, offering Wi-Fi connectivity, is connected to designated digital pins of the microcontroller.

SPI Bus: The Arduino communicates with the Wi-Fi shield utilizing the SPI bus.

iii) Power Control:

Relay: Controlled by a HIGH signal through the tamper switch and switched off by a LOW signal, the relay connects loads to specific digital pins of the Arduino Uno. This enables monitoring and theft detection functionalities upon activation.

A visual representation of these connections is depicted in Figure 1

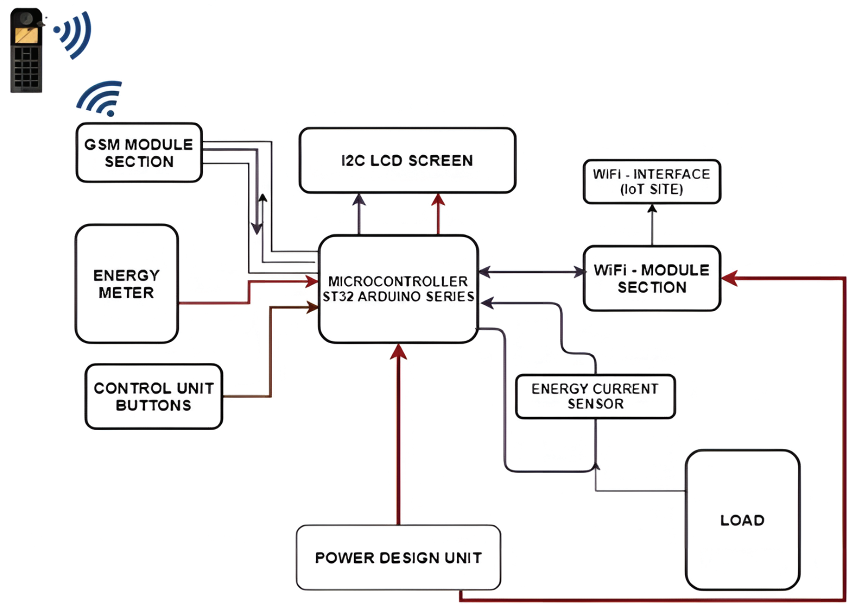

Figure 1: Hardware Block Diagram

HARDWARE DESIGN

This section looks into the essential hardware components of the IoT Smart Energy Meter Monitoring and Energy Theft Sensing System, detailing their design and implementation. It focuses on the circuit design and interconnections between components crucial for achieving the system's functionality. These components are described below:

Tamper Switch: An electromechanical switch triggered by motion or force, useful for tasks like counting, speed detection, and object sensing [4]. It has three terminals: Common, NO (Normally Open), and NC (Normally Closed), operating like a sensor to detect object presence/absence.

GSM Module: These connect devices to GSM-GPRS networks, enabling communication. GSM (Global System for Mobile Communications) is widely used, while GPRS (Global Packet Radio Service) boosts data speeds [10]. A GSM/GPRS module includes a modem, power supply, and interfaces (e.g., RS-232, USB) [6].

GSM Modem: Functions like a mobile phone with a SIM card, allowing computers to use cellular networks. Common uses include SMS/MMS messaging and mobile internet access.

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

It can be a standalone device or integrated into phones, with connectivity via serial, USB, or Bluetooth [5].

SIM7600CE-T 4G Shield: An Arduino-compatible module supporting 4G/3G/2G and GNSS positioning. It offers LTE CAT4 speeds (150 Mbps downlink, 50 Mbps uplink) and low-power SMS/data transmission, ideal for IoT applications [6].

Arduino Microcontroller (Atmega): They receive the input from the sensors, perform arithmetical operations on it, make decisions, trigger actuators, and display outputs on the LCD (Liquid Crystal Display). The Arduino Uno board uses an ATmega328p chip. The microcontroller comprises 14 digital pins and 6 analog pins, a 16MHz crystal oscillator, and connects to a computer via a USB connector. The digital pins can be used for both input and output, but they have only two states, HIGH or LOW. The Arduino IDE, an open-source platform where instructions are compiled and uploaded onto the Arduino Microcontroller through the Universal Serial Bus interface, is used to run the hardware. In-System Programming was employed on the Arduino board to code the ATmega328p microcontroller on the sensor unit board and the ATmega32 microcontroller on the main controller (ISP). Also, in using an Arduino, it is easy to connect with other micro-controllers [7].

I2C LCD Display Module: The character display on the HD44780 platform and the I2C LCD adapter make up this robust, efficient display module. A total of 32 ASCII characters, grouped into two rows of 16 characters each, can be seen on its LED-backlit display. I2C is the interface employed for communication by this 16x2 LCD screen. It signifies that the display absolutely uses the four pins VCC, GND, SDA, and SCL. It will spare at least 4 Arduino digital or analog pins. Also, the Gravity interface makes it simpler to utilize for carrying out the project design.

Wi-Fi Module (ESP8266): An open-source Internet of Things framework called Esspressif runs on the ESP8266 Wi-Fi SoC. (System-on-Chip for Wi-Fi). Using the ESP8266, it is conceivable to host a program as well as assign every Wi-Fi networking function to another application processor. The ESP8266 module is pre-programmed with an AT command set firmware, allowing it to connect to the Arduino Microcontroller with only a few clicks and get about the same level of Wi-Fi functionality as a Wi-Fi Shield [8]. Its primary function is to link these systems to the Internet. This Wi-Fi has the following features: Compactness, High durability, Low power consumption, 100m Amps, +19.5dBm output power in 802.11b mode, and integrates an Integrated TCP/IP protocol stack. The ESP8266 needs a 3.3V to function accurately; any higher voltage will burn it or damage it. A voltage divider was introduced into the system for the required voltage by using the values of a selected resistor and engaging in the calculation of the Voltage divider:

$$Vout = Vin \,.\frac{R2}{R1+R2}$$

$$Vout = 5v \,.\frac{2kohms}{1kohms+2kohms}$$

$$Vout = 5v \,.\frac{2kohms}{3kohms}$$

$$Vout = 5v\ (0.6667)$$

(approx.) $Vout = 3.3V$

Energy Current Sensor: As a result, cutting cables or reconnecting circuits is no longer necessary. To read the current AC value, just clamp the AC transformer probe to the AC line and connect the 3.5mm headphone connector to the signal conversion module. A 3V3 or 5V microcontroller is intended to be compatible with the analog output. It is practical for measuring and monitoring AC current. The following characteristics apply to this research project: high safety, several ranges for diverse monitoring settings, and interoperability with 3V3 or 5V microcontrollers like the Arduino microcontroller. Non-contact monitoring.

Relay Module: Relays are switches that permit and prohibit devices in response to Arduino commands. There are three contractors in a fundamental relay: a normally open (NO), a normally closed (NC), and a common (COM). The COM is coupled with NC in the ON input state. A relay module's operational voltage ranges from 0V to 5.0V DC.

Power Source: The ATmega328 microcontroller operates at a voltage of 0V to 5.0VDC, and a voltage higher than 5.5V would burn the chip out. To establish this system, a rechargeable 12V DC battery is connected to multiple-output DC–DC converters, which are essential in this research for a multitude of applications where different DC output voltages are required. This voltage regulator is used to establish the connection. A 12V rechargeable battery is used to actualize this circuit. A 12V charger is used to charge the 12V battery [10].

Energy Meter Reading

For every kilowatt per hour, the watt-hour LED within the smart energy meter glows at different periods. Using an LED indicator, we can assess how worn out an office is. The electrical inspiration that develops when force is expended is used by these LED squints. The optocoupler PC817 IC is bundled with this drive. Due to the microcontroller's automated PIN commitment, the IC's yield is given. The number of major efforts can be established with the use of a counter, and the percentage of energy expended can then be calculated. The meter's beat is used to connect it to the microcontroller. Additionally, the fee per unit is calculated using the latency between beats. The essentiality meter illuminates the LED only once after each cycle. As a result, if a 100-watt bulb is used to light the fire, the flame will flash several times in a minute.

Mobile Alert

The microcontroller and GSM module are interconnected, and the GSM module serves as a channel for SMS messages with usage estimates for each day, and also to know if any tampering is occurring. With the flexible number that has been registered on the smart meter, the message may be sent daily near the end.

SOFTWARE DESIGN

According to where the microcontroller will be utilized, a suitable programming language ought to be adopted for the software's design and implementation [2]. Software development allows for the implementation of various programming languages. The analysis of the software requirements, which encompass the ability to interoperate with the system control unit and peripherals, the potential to translate an analog signal to machine code, the potential for managing the system, and the capabilities to choose a preference, ultimately determines the selection of programming language based on the design methodology[7].

Assembly language integration alongside C programming is the most effective for this design, grounded on the aforementioned specifications for the overall software design. Hence, we will discuss how to program the Arduino board, emphasizing the role of software in system operation. With CodeBlocks, the Arduino UNO is created. An original open-source Arduino program is termed Arduino IDE. A USB cable is used to link the microcontroller to the computer. The Arduino's ADC tracks shifts in the analog input voltage and translates them into a binary value between 0 and 1023 bits. It returned a number between 0 and 1023 via an analog-to-read output and evaluated it using the output voltage.

Volume 6, Issue 1, June 2026- ISSN 2735-5985
http://dx.doi.org/10.21622/ACE.2026.06.1.1955

CIRCUIT DIAGRAM

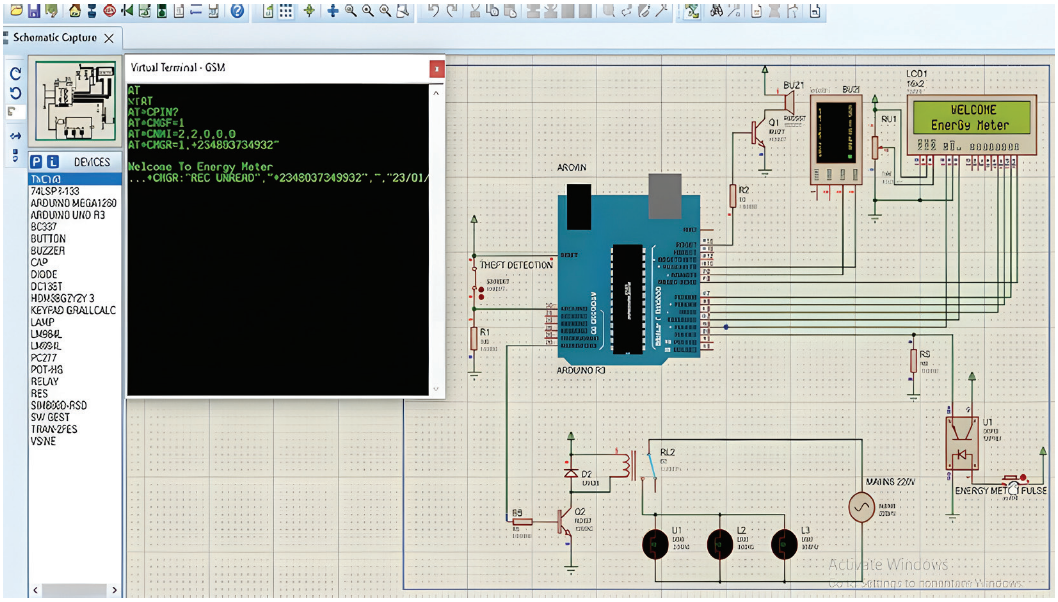

Figure 2: Circuit Diagram

The circuit design shown in Figure 2 is made using the hardware schematic shown in Figure 1. The software employed to build the circuit schematic and run the circuit simulation is Proteus Design Suite version 8.8. Printed circuit board schematics and designs are developed using the Proteus software tool, which is used for the electrical and electronic design (PCB). In the creation of electrical circuit testing, it is used. Automating the process of integrating components helps to conserve time and resources that might otherwise be lost. For example, unlike a physical component that could be damaged, if a component is connected incorrectly, it will only produce errors. The hardware system's various components were chosen from the Proteus development package's component menu. Each component is chosen and dragged to its proper location in the development environment. The GSM module, Relay module, and I2C LCD display are linked to the Arduino microcontroller's primary component through the digital pins depicted in Figure 2. The Arduino analyses the signal obtained from the energy meter via the tamper switch and transmits it to the GSM module and the ESP8266 Wi-Fi transmitter. From there, the data is transmitted to the cloud for the user to view and monitor, and then to the mobile devices of the designated users to monitor and also prevent energy theft. Finally, the receiver is to receive information from the transmitter and then process the information that it receives on a serial monitor. To provide an interface for reading values during simulation, the serial monitor is linked to pins 0 and 1. The components that will be used in the design, which have been correctly evaluated and researched, are all illustrated by the circuit diagram above.

FLOW-CHART DETAIL

This research work makes use of the prime cloud-based platform, which is an open-source program. A GSM module and an ESP8266 module with the tamper sensor can be used to send the sense of the theft through the tamper switch on the meter and send it to the user's designated device for assessment and control. Updates of all information are discovered throughout the day through the user's device to check the energy metering status. The user-designated numbers are first included in the system program to track the system's multiple indicators before making use of these features. The system's operational flow chart is shown in Figure 3.

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

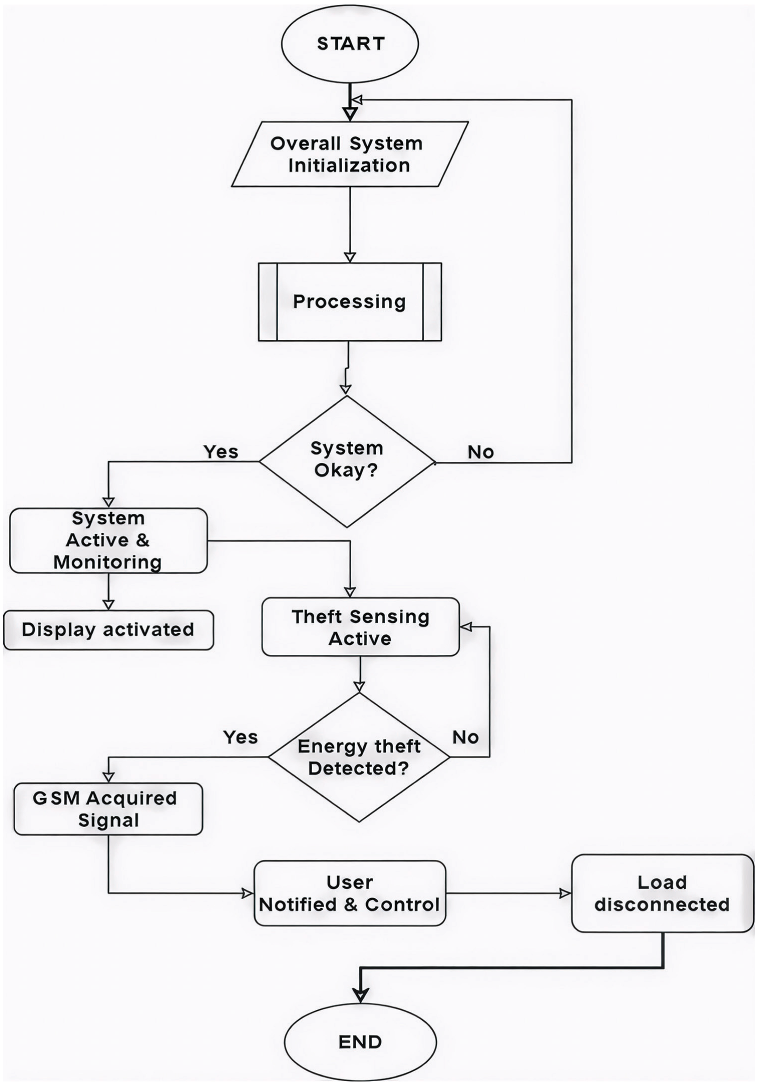

Figure 3: Flow-Chart Diagram

With the system design in place, the next step was to implement the system and evaluate its performance. This section details the implementation process and presents the results obtained.

4. SYSTEM IMPLEMENTATION AND RESULT

a) Hardware Implementation

This involves the building and synchronization of the physical components, sensors, power supplies, and controllers that make up the system's hardware. The breadboarding of the components, the mounting and connecting of components, and the packaging are all included in the physical execution of the project:

1.) Breadboarding of components

The different parts were breadboarded before the project was built, as shown in Figure 4, to ensure their functionality and that the circuit diagram is working accurately. In this stage, different components were tested, and the best ones suited to the use of the project were used. The testing of the hardware part of the project was done with the use of a breadboard before the components were further soldered to a continuous type Vero board, as revealed in Figure 4.

Volume 6, Issue 1, June 2026- ISSN 2735-5985
http://dx.doi.org/10.21622/ACE.2026.06.1.1955

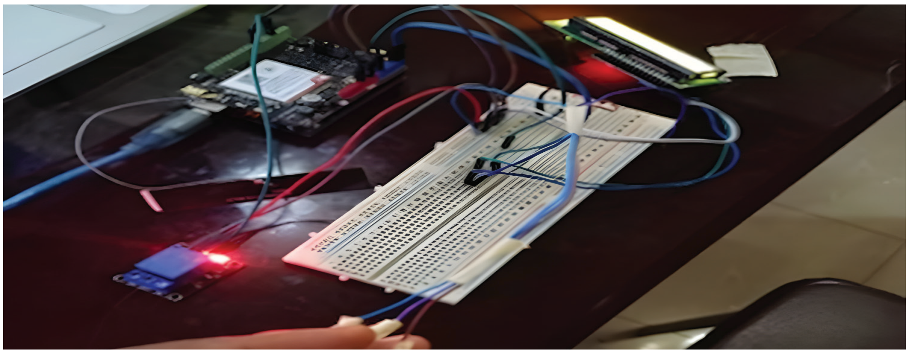

Figure 4: Components' Functionality testing on the Breadboard platform

2.) Implementation and Integrations of Components

Each component was put on the board and cemented on the surface, as shown in Figure 5, after they were tested to operate well with each other and produce desired outcomes with a high level of accuracy. After that, tests were performed to check that all of the components were in working order and had not been damaged during the installation. It should be noted that the connection and bonding operation was carried out in stages to address errors early on.

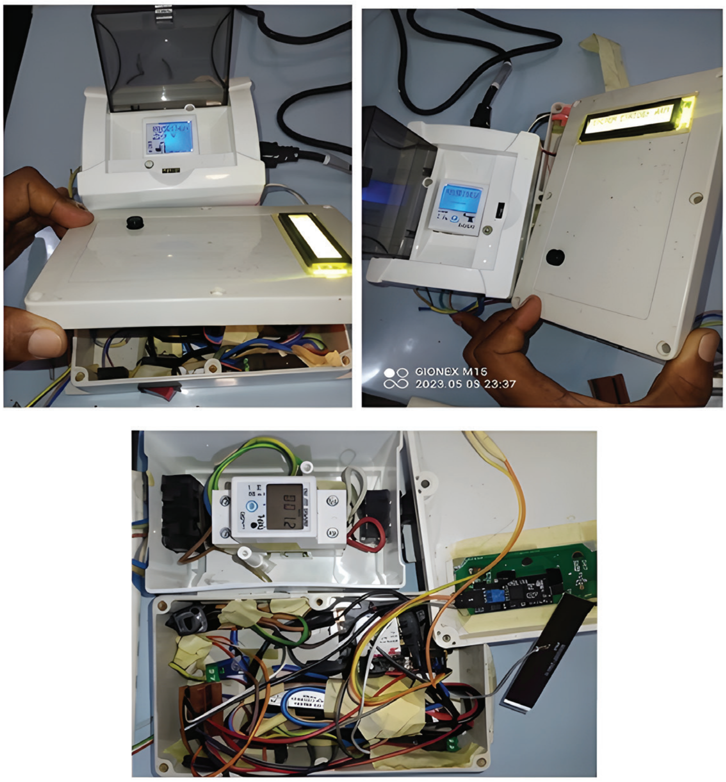

Figure 5: installation of components in a project package for meter monitoring and detection

3.) Packaging

The project was packaged in a particle board, as shown in Figure 4, after the mounting and soldering of various components. An arrangement was made to give the project a compact size. In an attempt to provide an internal power source for the controller, the GSM communication module, and other project-related sensors, a 12-volt DC lithium-ion battery was attached to the package box. Also, an internal pathway was made for a connector to

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

be linked to a DC power source. The use of the intelligent power system kept the total build and construction fairly compact.

b) Software Implementation

It is comprised of the system's functionality, which also comprises the Proteus 8 Professional, the Arduino compiled code utilized to code it, and the embedding of the project into an application.

For this research work, the Arduino IDE was employed. The system's pre-installed Arduino IDE was upgraded to the most recent release, Arduino 1.8.15. The essential libraries, as well as the Adafruit GSM module and ESP8266 libraries, were successfully updated. The initial execution of the code was performed using the setup as well as the program development workspace, and subsequent runs were performed using the loop. It was run and compiled, and the bugs were fixed in the Arduino microcontroller. Sometimes, when applicable, the # include function was used to use the libraries.

During this research, the hardware block diagram layout shown in Figure 1 was employed in order to establish the appropriate location of components and their connections, as well as the viability of the previously established design. After designing an IoT-based energy meter monitoring and theft sensing system circuit, simulations were performed in Proteus to enhance the circuit parameters that preceded the hands-on construction of the system design. The complete diagram of the circuit is revealed in Figure 2, displaying all critical components before the circuit's wiring. The Proteus schematic capture and component mode from the library's toolbar were used to select each component, although some libraries were nonexistent.

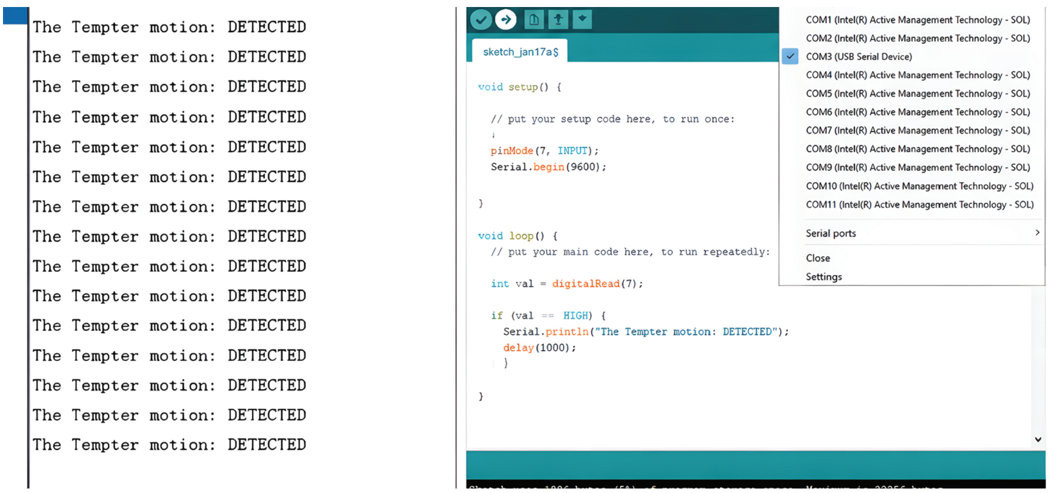

Figure 6: Software performance testing

c. System and Result

Electronics testing is a crucial stage in the design and implementation of systems. Unit testing, program evaluation, subsystem evaluation, system testing, and validation are just a few examples of the numerous test kinds performed in this work. Before performing a comprehensive overall system test, the sensor parts, subsystems, and hardware were evaluated.

i) Unit testing: The IoT Smart Energy Meter Monitoring and Energy Theft Sensing

System components were first tested for continuity and power rating with the aid of a multimeter to ensure their functionality before integration into the system. Tests were carried out on the power supply and the microcontroller.

ii) Subsystem testing: The subsystem test was carried out to ensure each unit was

Volume 6, Issue 1, June 2026- ISSN 2735-5985
http://dx.doi.org/10.21622/ACE.2026.06.1.1955

functional while gradually integrating the units. This test was conducted while gradually developing the system; the sensors were tested using Arduino and other components to ensure optimal functionality. It also involves the correct placement of the components to determine their voltage and current consumption.

iii) System testing: During system testing, the IoT Smart Energy Meter Monitoring and

The Energy Theft Sensing System is tested as an entire unit and as a unified system. It was at this stage that it was discovered the transceivers were not supplied with the required voltage.

Figure 7: Final Package of the Meter and the Sensing Display box.

iv) Acceptance testing: The IoT Smart Energy Meter Monitoring and Energy Theft Sensing

Systems were connected to a voltage source and operated by an individual after the construction had been completed.

v) Program Testing: After developing, compiling, and debugging the program code to

ensure there are no errors, the code is then uploaded onto the microcontroller and tested. Some libraries were missing and had to be downloaded and added before the code could function appropriately.

vi) System's Accuracy in Detecting Energy Theft: In a series of tests conducted over a

For a period of 30 days, the IoT Smart Energy Meter Monitoring and Energy Theft Sensing System was evaluated for its accuracy in detecting energy theft. A total of 100 simulated theft scenarios were introduced, with different methods of unauthorized energy consumption, such as bypassing the meter and tampering with the wiring. The system successfully detected 95 out of the 100 theft scenarios, resulting in an accuracy rate of 95%. The false - positive rate was 3%, with 3 instances where the system incorrectly flagged normal energy usage as theft. These results are summarized in Table 2:

Table 2: System's Accuracy in Detecting Energy Theft

S/N	Test parameter	VALUES
1	Total number of theft scenarios	100
2	Number of correctly detected thefts	95

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

3	Accuracy rate	95%
4	False - positive rate	3%

The response time for remote control actions was measured using a mobile device to send commands to the smart meter. A total of 50 remote control commands, including turning off the meter and adjusting energy consumption settings, were sent. The average response time was found to be 5 seconds, with a minimum response time of 3 seconds and a maximum response time of 8 seconds. The response time distribution is shown in Figure 8:

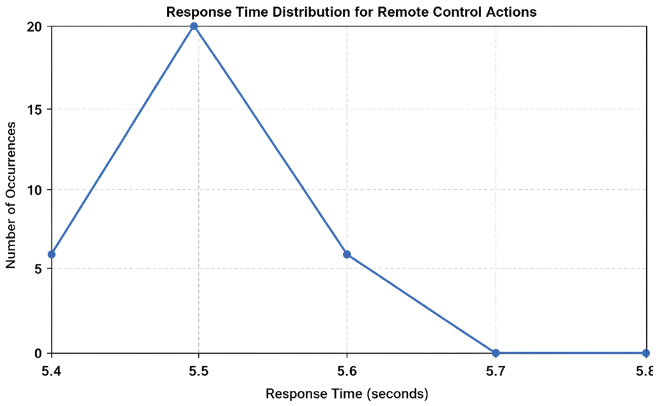

Figure 8: Quantitative data on system performance

vii) Power consumption: The power consumption of the system was measured over a 24-hour period

hour period. The components of the system, including the Atmega microcontroller, GSM module, and sensors, were monitored. The total power consumption was 12 Wh (watt - hours) per day. The breakdown of power consumption by component is as follows:

Table 3: Power consumption details

COMPONENT	POWER CONSUMPTION PER DAY (Wh)
Atmega microcontroller	4
GSM module	5
Sensors (tamper sensor, AC current sensor, etc.)	3

5. NOVEL CONTRIBUTION AND DISTINCTION FROM EXISTING SYSTEMS

The contribution of this paper to IoT-based smart energy metering is by integrating real-time theft detection, dual communication architecture, and automated control mechanisms into a unified system. While previous studies have implemented IoT-based smart meters using GSM or Wi-Fi independently, the proposed system combines both communication channels to enhance reliability and redundancy.

The novelty of this research lies in the following:

1. Hybrid Communication Framework
2. Unlike conventional systems that rely solely on GSM or Wi-Fi, this system integrates both ESP8266 (Wi-Fi) and GSM modules that ensure continuous data transmission even in cases of network failure.

3. Integrated Theft Detection with Automatic Disconnection
4. Existing systems often focus on monitoring and alerting. This system incorporates real-time tamper detection coupled with automatic relay-based power disconnection and actively prevents energy theft.
5. Remote User Control via GSM Interface
6. The system allows users to not only monitor consumption but also remotely control energy supply, which extends beyond basic monitoring functions reported in earlier works.
7. Low-Cost and Scalable Design
8. The architecture is designed using widely available components such as ATmega328p and ESP8266, which makes it suitable for deployment in developing regions with cost constraints.

6. CONCLUSION

This research developed an IoT - based smart meter monitoring system to tackle energy management and security issues in developing countries. The system, incorporating an Atmega microcontroller, GSM module, and non - invasive sensor, offers key features like theft detection and remote control. Upon detecting theft, it alerts users and cuts power, with high accuracy in testing. Remote monitoring enables real - time energy usage access, empowering users to manage consumption. Rigorous testing during implementation verified its functionality. However, scalability and power consumption optimization remain areas for future research. Overall, the system provides an effective energy management solution, with potential for broader impact on sustainable energy use as it evolves.

7. DECLARATIONS

AUTHOR CONTRIBUTION

The authors carried out the whole research.

FUNDING

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

COMPETING INTEREST

There is no competing interest.

CONSENT TO PUBLISH DECLARATION:

The authors consent to the publication of this paper. The authors have read and approved the final version of the manuscript and agreed to its submission. No part of this paper has been published or is under consideration elsewhere.

REFERENCES

[1] M. H. Yaghmaee and H. Hejazi, "Design and Implementation of an Internet of Things Based Smart Energy Metering," in *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, IEEE, Aug. 2018, pp. 191–194. doi: 10.1109/SEGE.2018.8499458.

[2] E. Adetiba *et al.*, "Development of an IoT Based Data Acquisition and Automatic Irrigation System for Precision Agriculture," in *2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON)*, IEEE, Apr. 2022, pp. 1–5. doi: 10.1109/NIGERCON54645.2022.9803132.

[3] M. Tahir, N. Ismat, H. H. Rizvi, A. Zaffar, S. M. Nabeel Mustafa, and A. A. Khan, "Implementation of a smart energy meter using blockchain and Internet of Things: A step toward energy conservation," *Front. Energy Res.*, vol. 10, Dec. 2022, doi: 10.3389/fenrg.2022.1029113.

http://dx.doi.org/10.21622/ACE.2026.06.1.1955

[4] S. G. Rajeshwari, N. K. Kalkeri, N. Pati, S. Jalihal, and S. Barigali, "IOT based smart energy meter for efficient energy utilization and billing," I*nt. J. Adv. Res. Ideas Innov. Educ. (IJARIIE)*, vol. 8, no. 4.

[5] A. Rachedi, M. H. Rehmani, S. Cherkaoui, and J. J. P. C. Rodrigues, "IEEE Access Special Section Editorial: The Plethora of Research in Internet of Things (IoT)," *IEEE Access*, vol. 4, pp. 9575–9579, 2016, doi: 10.1109/ACCESS.2016.2647499.

[6] "SIM7600G-H CAT4 4G Shield for Arduino | Ultimate Connectivity," 2025. [Online]. Available: https://wiki.dfrobot.com/SIM7600CE-T_4G(LTE)_Shield_V1.0_SKU_TEL0124

[7] A. E. Akindele, A. I. Afolabi, V. W. Oguntosin, O. A. Alashiri, V. O. Matthews, and O. O. Akande, "Development of an intelligent smart shopping Cart system," in *Lecture Notes in Engineering and Computer Science*, 2019.

[8] B. K. Barman, S. N. Yadav, S. Kumar, and S. Gope, "IOT Based Smart Energy Meter for Efficient Energy Utilization in Smart Grid," in *2018 2nd International Conference on Power, Energy and Environment: Towards Smart Technology (ICEPE)*, IEEE, Jun. 2018, pp. 1–5. doi: 10.1109/EPETSG.2018.8658501.

[9] N. Agnihotri, "GSM/GPRS Module : All You Need To Know," 2022. [Online]. Available: https://www.engineersgarage.com/gsm-gprs-module-all-you-need-to-know/

[10] A. R. and K. V., "IoT Based Smart Energy Theft Detection and Monitoring System for Smart Home," in *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, IEEE, Jul. 2020, pp. 1–6. doi: 10.1109/ICSCAN49426.2020.9262411.