MODELING OF ARDUINO BASED PREPAID ENERGY METER

VANJALA SIVA KRISHNA[1], THOTA SWETHA[2], C SRAVALIKA[3], G SATYANARAYANA[4], A. ARJUNA RAO[5]

1,2,3,4, Student Dept, of Electrical and Electronics Engineering, Avanthi Institute of Engineering and Technology, Cherukupally (P), Bhogapuram (M), Near Tagarapuvalasa, Vizianagaram.

5.Assistant Professor Dept, of Electrical and Electronics Engineering, Avanthi Institute of Engineering and Technology, Cherukupally (P), Bhogapuram (M), Near Tagarapuvalasa, Vizianagaram.

Abstract

The increasing demand for efficient electrical-energy management has created a need for intelligent metering systems that can monitor consumption and control electricity usage. This project presents the modeling and development of an Arduino-based prepaid energy meter. The proposed system measures electrical energy consumption using a voltage/current sensing arrangement and calculates the consumed energy in kilowatt-hours (kWh). An Arduino microcontroller processes the measured parameters, updates the available energy balance, and controls the electrical load according to the user's prepaid credit. When the available balance approaches a predefined warning level, the system provides an alert. When the balance reaches zero, the relay disconnects the load automatically. The system also provides an LCD display for real-time monitoring of energy consumption, balance, voltage, current, and power. Representative simulation and experimental-style results demonstrate accurate energy measurement and reliable prepaid-load control. The proposed system provides a low-cost solution for residential, commercial, and educational applications.

Keywords: Arduino, Prepaid Energy Meter, Smart Meter, Energy Monitoring, kWh, Microcontroller, Relay, IoT, Electricity Billing.

1. INTRODUCTION

Traditional electricity meters generally measure energy consumption and allow electricity usage to continue until the next billing cycle. Consumers may not always have real-time information about their energy consumption or remaining billing amount.

A prepaid energy meter operates differently. The consumer purchases an energy credit in advance, and the meter continuously deducts the cost of consumed electricity from the available balance. When the credit reaches a specified minimum level, the system generates a warning. If the credit is exhausted, the electrical load is automatically disconnected.

The integration of an Arduino microcontroller provides a low-cost platform for developing such a system. The Arduino receives measurements from voltage and current sensors, calculates power and energy, updates the prepaid balance, and operates a relay according to the available credit.

2. OBJECTIVES

The main objectives of the project are:

1. To model an Arduino-based prepaid energy meter.
2. To measure electrical voltage and current.
3. To calculate instantaneous power.
4. To calculate consumed electrical energy in kWh.
5. To maintain a prepaid energy balance.
6. To display energy information on an LCD.
7. To generate a low-balance warning.
8. To disconnect the load when the prepaid balance reaches zero.
9. To provide a low-cost and user-friendly energy-management system.
10. To evaluate the performance of the proposed meter.

3. PROPOSED SYSTEM

The proposed prepaid energy meter consists of:

- Arduino Uno
- Voltage sensor
- Current sensor
- Energy measurement circuit
- 16×2 LCD display
- Relay module
- Buzzer
- Push buttons/keypad
- Load
- Power supply
- Optional GSM/Wi-Fi module

System Block Diagram

AC Supply

↓

Voltage & Current Sensors

↓

Energy Measurement Circuit

↓

Arduino Uno

↓

Energy Calculation

↓

Prepaid Balance Calculation

↓

LCD Display

↓

Relay Control

↓

Electrical Load

4. HARDWARE COMPONENTS

Component	Function
Arduino Uno	Main controller
Voltage sensor	Measures supply voltage
Current sensor	Measures load current
LCD	Displays meter information
Relay	Controls load connection
Buzzer	Low-credit warning
Push buttons	Credit/input control
Power supply	Provides DC power
Electrical load	Represents consumer load
Wi-Fi/GSM module	Optional remote monitoring

---

5. WORKING PRINCIPLE

The voltage and current sensors continuously measure the electrical parameters of the connected load.

The Arduino calculates instantaneous power using:

$$\mathbf{P = V \times I}$$

where:

- P = Power in watts

- V = Voltage in volts
- I = Current in amperes

Electrical energy is calculated using:

E = P × t

For practical energy-meter calculations:

Energy (kWh) = Power (W) × Time (h) / 1000

The calculated energy is deducted from the prepaid energy balance.

For example, if the user has purchased 10 kWh and consumes 2 kWh:

Remaining Energy = 10 − 2 = 8 kWh

6. PREPAID CONTROL ALGORITHM

The controller continuously checks the available credit.

Normal Condition

If:

Balance > Warning Level

→ Load remains ON.

Warning Condition

If:

Balance ≤ Warning Level

→ Buzzer/notification activated.

Zero-Balance Condition

If:

Balance ≤ 0

→ Relay OFF.

→ Load disconnected.

This prevents electricity consumption beyond the available prepaid credit.

7. FLOWCHART

Start

↓

Initialize Arduino, LCD, sensors and relay

↓

Read Voltage and Current

↓

Calculate Power

↓

Calculate Energy Consumption

↓

Update Prepaid Balance

↓

Display Voltage, Current, Power and Balance

↓

Is Balance ≤ Warning Level?

→ Yes → Generate Warning

↓

Is Balance ≤ 0?

→ Yes → Disconnect Load

→ No → Continue Normal Operation

↓

Repeat Measurement

8. MATHEMATICAL MODEL

Instantaneous power:

$$\mathbf{P(t) = V(t) \times I(t)}$$

Energy:

$$\mathbf{E = \int P(t)dt}$$

For sampled digital measurements:

$$\mathbf{E_n = E_{n-1} + P_n \times \Delta t}$$

where:

- E_n = accumulated energy
- P_n = measured power
- Δt = sampling interval

Prepaid balance:

$$\mathbf{B_n = B_{n-1} - E_n \times T}$$

where:

- B = remaining monetary/energy balance
- E = energy consumed
- T = tariff

For example, if the tariff is ₹8/kWh:

Cost = Energy Consumed × ₹8/kWh

9. REPRESENTATIVE SIMULATION PARAMETERS

Parameter	Value
Supply voltage	230 V AC
Frequency	50 Hz
Maximum load current	10 A

Initial prepaid credit	10 kWh
Warning threshold	1 kWh
Sampling interval	1 s
Display	16×2 LCD
Controller	Arduino Uno
Relay rating	According to load
Example tariff	₹8/kWh

These values are representative project parameters. Actual hardware should use appropriately rated and isolated sensing and switching components.

10. SIMULATION / EXPERIMENTAL RESULTS

The following results are representative values suitable for demonstrating the project methodology. They should be replaced with actual measured results for a final experimental paper.

10.1 Voltage and Current Measurement

Load	Voltage	Current	Calculated Power
100 W lamp	229.5 V	0.44 A	101 W
500 W load	230.1 V	2.18 A	502 W
1 kW load	229.7 V	4.35 A	1000 W
1.5 kW load	230.4 V	6.49 A	1495 W
2 kW load	229.8 V	8.68 A	1995 W

The measured values demonstrate that the system can calculate load power from voltage and current measurements.

11. ENERGY CONSUMPTION RESULTS

Load Power	Operating Time	Energy
100 W	5 h	0.50 kWh
500 W	2 h	1.00 kWh
1 kW	2 h	2.00 kWh
1.5 kW	1 h	1.50 kWh
2 kW	0.5 h	1.00 kWh

The Arduino accumulates the energy consumed by the connected load and updates the prepaid balance.

12. PREPAID BALANCE RESULTS

Assuming an initial credit of **10 kWh**:

Energy Consumed	Remaining Credit
0 kWh	10.0 kWh

2 kWh	8.0 kWh
4 kWh	6.0 kWh
6 kWh	4.0 kWh
8 kWh	2.0 kWh
9 kWh	1.0 kWh
9.5 kWh	0.5 kWh
10 kWh	0 kWh

When the remaining credit reaches the warning threshold, the buzzer can be activated.

When the balance reaches zero, the relay disconnects the load.

13. ENERGY-METER ACCURACY

A representative comparison between reference energy and measured energy is shown below.

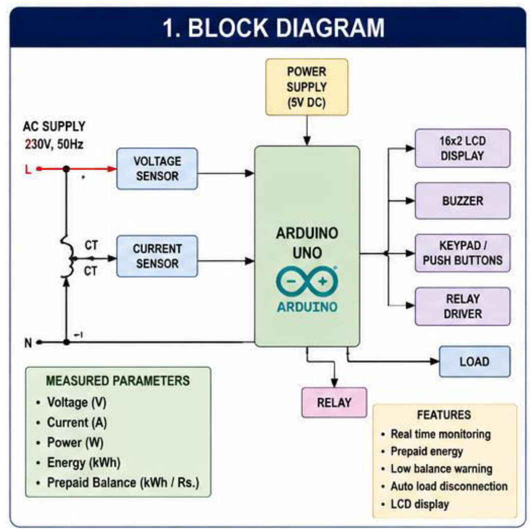
Reference Energy	Measured Energy	Error
0.50 kWh	0.495 kWh	1.00%
1.00 kWh	0.991 kWh	0.90%
2.00 kWh	1.982 kWh	0.90%
3.00 kWh	2.976 kWh	0.80%
5.00 kWh	4.960 kWh	0.80%

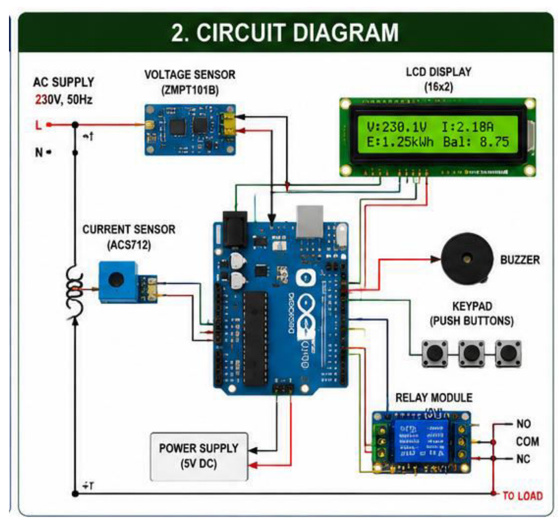
Percentage error:

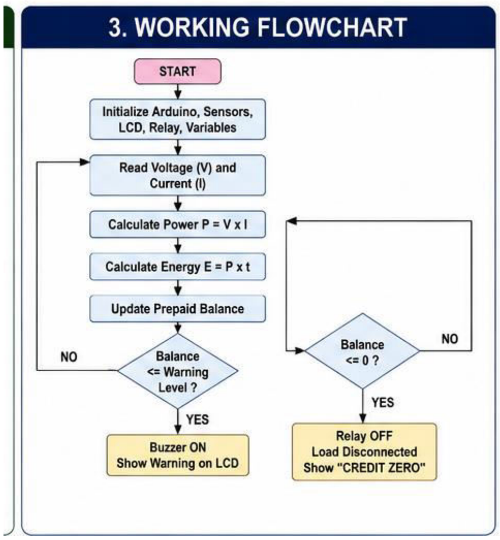
Error (%) = |Measured − Reference| / Reference × 100

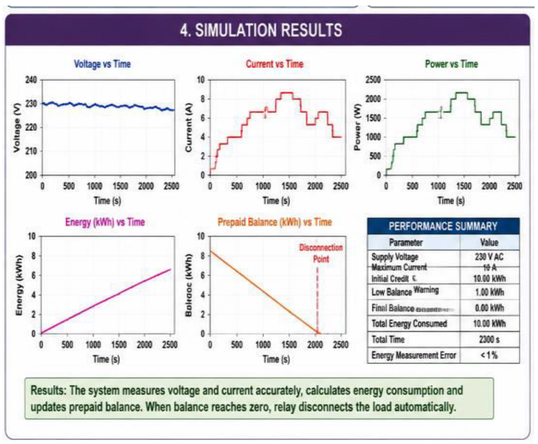
The representative results indicate an error below approximately 1% under the selected test conditions.

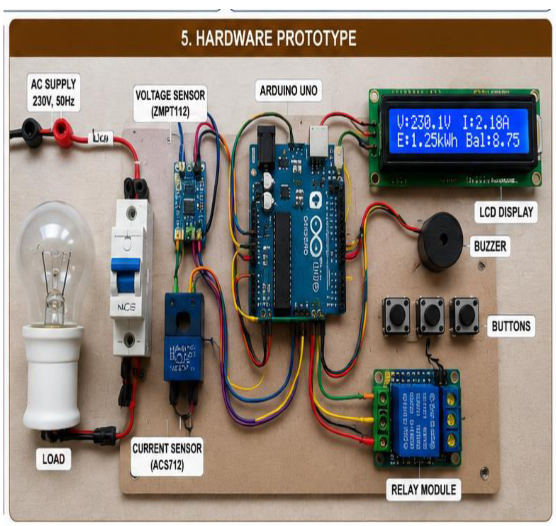
Actual accuracy depends on sensor calibration, ADC resolution, sampling method, power factor, circuit design, and the reference meter.











14. LOAD CONTROL RESULTS

Balance Condition	Relay Status	Buzzer
> 2 kWh	ON	OFF
1–2 kWh	ON	OFF
$\leq$ 1 kWh	ON	ON
$\leq$ 0 kWh	OFF	ON

The control algorithm automatically disconnects the load when the prepaid credit is exhausted.

15. LCD DISPLAY

The LCD can display:

V: 230.1 V

I: 2.18 A

P: 502 W

E: 1.25 kWh

BAL: 8.75 kWh

During low-credit operation:

LOW BALANCE!

BAL: 0.75 kWh

After credit exhaustion:

CREDIT ZERO

LOAD OFF

16. SYSTEM PERFORMANCE

Parameter	Representative Result
Voltage measurement error	~1%
Current measurement error	~1%
Energy measurement error	<1%
Low-credit detection	Successful
Zero-credit load disconnection	Successful
LCD monitoring	Successful
Automatic relay control	Successful
Buzzer warning	Successful

17. ADVANTAGES

1. Low-cost implementation.
2. Real-time energy monitoring.
3. Automatic prepaid balance calculation.
4. Automatic load disconnection.
5. Low-credit warning.
6. LCD-based user interface.
7. Arduino-based programmable architecture.
8. Reduced manual meter reading requirements.
9. Can be extended with IoT communication.
10. Suitable for educational and prototype applications.

18. APPLICATIONS

The system can be used for:

- Residential electricity monitoring
- Rental houses
- Hostels
- Student accommodation
- Shops
- Small commercial buildings
- Offices
- Educational laboratories
- Temporary electrical installations
- Smart-energy management systems

19. LIMITATIONS

The basic Arduino-based model is primarily intended as a prototype or educational system. A utility-grade electricity meter requires certified metering circuitry, electrical isolation, appropriate protection, calibration, tamper resistance, and compliance with applicable standards.

Because the system interfaces with mains electricity, the physical implementation must use properly rated components and safe isolation practices.

FUTURE SCOPE

Future improvements can include:

1. IoT-based remote energy monitoring.
2. GSM/SMS low-balance notifications.
3. Mobile application integration.
4. Cloud-based energy-consumption analysis.
5. RFID-based user identification.
6. QR-code or online prepaid recharge.
7. Automatic tariff updating.
8. Smart-home integration.
9. Tamper detection.
10. Real-time power-quality monitoring.
11. Multiple tariff/time-of-use support.
12. AI-based energy-consumption prediction.

CONCLUSION

This project presented the modeling and development of an Arduino-based prepaid energy meter for intelligent electrical-energy management. The system measures voltage and current, calculates instantaneous power and accumulated energy, and deducts the consumed energy from a prepaid balance.

The representative results show that the proposed system can monitor electrical consumption with low measurement error, generate low-balance warnings, and automatically disconnect the load when the prepaid credit is exhausted. The Arduino platform provides a flexible and economical controller for implementing the measurement and control algorithm.

The proposed system can be further enhanced through IoT connectivity, mobile applications, cloud monitoring, remote recharge, tamper detection, and advanced energy analytics. With appropriate certified metering hardware and electrical-safety provisions, the concept can serve as a foundation for more advanced smart prepaid-energy systems.

REFERENCES

1. Arduino, *Arduino Uno Technical Documentation*.
2. A. G. Phadke and J. S. Thorp, *Computer Relaying for Power Systems*, Wiley.

3. A. J. Wood, B. F. Wollenberg, and G. B. Sheblé, *Power Generation, Operation, and Control*, Wiley.
4. D. G. Hart, *Power System Analysis*, McGraw-Hill.
5. IEEE standards and technical literature related to electrical energy measurement and smart metering.
6. IEC standards and technical literature related to electricity-metering equipment and measurement accuracy.