IoT BASED SMART METERING SYSTEM

A Project report submitted in partial fulfilment of the requirements for the degree of B. Tech in Electrical Engineering

By

Rupam Saha (11701618029)
Debarun Das (11701618049)
Abhradeep Roy (11701618066)
Tanuj Das (11701618011)

Under the supervision of

Mr. Subhasis Bandhopadhyay
Assistant Professor
Department of Electrical Engineering

श्रमम् बिना न किमपि साध्यम्

Department of Electrical Engineering
RCC INSTITUTE OF INFORMATION TECHNOLOGY
CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015, WEST BENGAL
Maulana Abul Kalam Azad University of Technology (MAKAUT)

Department of Electrical Engineering
RCC INSTITUTE OF INFORMATION TECHNOLOGY
CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015, WEST BENGAL
PHONE: 033-2323-2463-154, FAX: 033-2323-4668
Email: hodeercciit@gmail.com, Website: http://www.rcciit.org/academic/ee.aspx

CERTIFICATE

To whom it may concern

This is to certify that the project work entitled IoT Based Smart Metering System is the work carried out by RUPAM SAHA (11701618029), DEBARUN DAS (117016180), ABHRADEEP ROY (11701618066), TANUJ DAS (117016180), the students of B. Tech in the Department of Electrical Engineering, RCC Institute of Information Technology (RCCIIT), Canal South Road, Beliaghata, Kolkata-700015, affiliated to Maulana Abul Kalam Azad University of Technology (MAKAUT), West Bengal, India, during the academic year 2021 -22, in partial fulfilment of the requirements for the degree of Bachelor of Technology in Electrical Engineering and that this project has not submitted previously for the award of any other degree, diploma and fellowship.

…………………………………
External Examiner

subhasis bandopadhyay

(Mr. Subhasis Bandhopadhyay)
Assistant Professor
Department of Electrical Engineering
RCC Institute of Information Technology

Countersigned by

……………………………………………………
(Prof. Dr. Debasish Mondal)
Head of Department (HOD) of Electrical Engg.
RCC Institute of Information Technology

ACKNOWLEDGEMENT

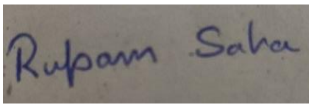
It is our great fortune that we have got opportunity to carry out this project work under the supervision of **Mr. Subhasis Bandhopadhyay** in the Department of Electrical Engineering, RCC Institute of Information Technology (RCCIIT), Canal SouthRoad, Beliaghata, Kolkata-700015, affiliated to Maulana Abul Kalam Azad University of Technology (MAKAUT), West Bengal, India. We express our sincere thanks and deepest sense of gratitude to our guide for his constant support, unparalleled guidance and limitless encouragement.

We would like to convey our gratitude to all the faculty members and staffs of the Department of Electrical Engineering, RCCIIT for their whole hearted cooperation to make this work turn into reality.

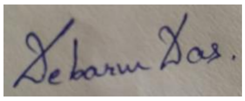
We are very thankful to our department and to the authority of RCCIIT for providing all kinds of infrastructural facility towards the research work.

Thanks to the fellow members of our group for working as a team.

Rupam Saha (11701618029)

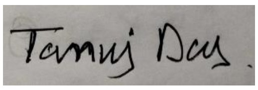


Debarun Das (117018049)



Abhradeep Roy (11701618066)

Tanuj Das (11701618011)



ABSTRACT

1

The emerging IoT concept recently in our lives, has offered the chance to establish energy efficient smart devices, systems and cities. Due to the urgent need for conserving energy, this paper proposes an IoT based energy efficient wireless smart metering system design. It competes with the existing meters as being a low cost completely integrated metering system. It offers an ease for the users in the form of a database for the electricity supplier company. The proposed system design has an accuracy level of 98% and it is about 28% lower cost than its peer in the global market. The proposed design reduced the power consumption by 12%.

Contents

Topic	Page no.
Acknowledgement	**1**
Abstract	**2**
Chapter I (Introduction)	
1.1 Introduction	**4**
1.2 Internet of Things (IoT)	**5**
1.3 How does IoT works?	**5**
1.4 Organisation of Thesis	**5**
Chapter II (Literature)	
2.1 Literature Review	**6**
Chapter III (Theory)	
3.1 Smart Meters	**7**
3.2 Microcontroller	**7**
3.2.1 How do microcontroller work?	**7**
3.2.2 What are the elements of microcontroller?	**8**
3.2.3 Microcontroller features	**9**
3.2.4 Microcontroller Applications	**9**
3.2.5 Comparison of Microcontroller Microprocessor	**10**
3.3 Atmega328p	**10**
3.3.1 Features and parametric	**10**
3.3.2 Advantages and disadvantages	**11**
3.2.3 Product application	**12**
3.2.4 Block Diagram	**12**
3.2.5 Pin configuration	**13**
3.2.6 Pin description	**14**
Chapter IV (Hardware Modelling and Design)	
4.1 Proposed Metering System	**16**
4.1.1 Sensing Unit	**17**
4.1.2 Control Unit	**17**
4.2 System Overview	**18**
Chapter V (Hardware implementation and Software programming)	
5.1 Circuit Diagram	**19**
5.2 Flowchart	**19**
5.3 Programming on Arduino	**20**
5.4 Photograph of the prototype	**27**
Chapter VI (Conclusion)	**28**

Chapter VII (Reference) **28**

Chapter I (Introduction)

1.1 Introduction:

The drastic increase in urban areas over the past few years requires sustainable, efficient, and smart solutions for transportation, governance, environment, quality of life, and so on. Since the evolution of the IOT concept in the early 2000s, the Internet of Things offers many sophisticated and ubiquitous applications for smart cities. It is the ability of devices, networks and sensors to communicate with each other, with and without any human interaction. Internet of things (IoT) basic simplified workflow can be described as follows: object sensing, identification and communication of object specific information, trigger an action and lastly, the smart device or system provides its service. The energy demand of IoT applications is increased, while IoT devices continue to grow in both numbers and requirements. Therefore, new smart city must have the ability to efficiently utilize energy and handle the associated challenges.

The conventional electromechanical energy meters have a great contribution in dissipation of energy. Mainly because they are post-paid meters, customers cannot keep track of their energy consumption, except on a monthly basis. However, Smart meters when programmed with home appliances can be used to control the electricity consumption. Moreover, Smart meters are less error prone. Smart Meters are electronic measurement devices used by utilities to communicate information for billing customers and operating their electric systems. It is simply consisting of a metering unit and a communication interface between the metering unit and the utility. The communication interface module is connected to the microcontroller to transmit and receive information between the utility and the metering unit installed in the house.

This paper presents the design and implementation of a full integrated smart energy efficient metering system. The design provides a low power consumption smart metering system. The proposed design is implemented at two ends, one on the consumer end for IoT operation and other on the service provider end for managing customers' data through an implemented and customized database, which is the overall system block units, from the main supply in the house to the back end, electricity supplier, managing unit.

1.2 Internet of Things (IoT)

The Internet of Things has arrived and it's going to introduce incredible opportunity over the next five years. And while smart things are exactly that, the IoT industry has a long way to go in terms of overall security. Many of today's IoT devices are rushed to market with little consideration for basic security and privacy protections: "Insecurity by design."

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. The connectivity, networking and communication protocols used with these web-enabled devices largely depend on the specific IoT applications deployed.

IoT can also make use of artificial intelligence (AI) and machine learning to aid in making data collecting processes easier and more dynamic.

1.3 How does IoT works?

An IoT ecosystem consists of web-enabled smart devices that use embedded systems, such as processors, sensors and communication hardware, to collect, send and act on data they acquire from their environments. IoT devices share the sensor data they collect by connecting to an IoT gateway or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices -- for instance, to set them up, give them instructions or access the data.

1.4 Organization of Thesis

The thesis is organised into five chapters including the chapter of introduction. Each chapter is different from the other and is described along with the necessary theory required to comprehend it.

Chapter II (Literature)

2.1 Literature Review

The history and evolution of energy meter dates back to the 1880s, a little before the widespread use of electricity. The era of gas lamps also contained the energy measurement system where the amount of energy consumed per household was calculated. With the discovery and use of electricity, the electric lamps rapidly replaced the gas lamps, proving to be brighter and more cost efficient. A new system for consumption measurement was required. DC meters measured charge in the unit ampere-hours. With time the DC meters proved to be insufficient. Then came Edison’s meters classified as electrolytic and electrochemical meters. The electrochemical meters, though sufficed the purpose, were labour-intensive to read. Thus, not welcomed by users. It was in 1889, that a Hungarian named Otto Blathy created and patented the first AC meter using kilowatt-hour as the standard unit, whose operating principle is used till date.

In Korea, Son et. al. (2010) proposed a system based on IoT which comprises modem of mains communication, a unit of Wi-Fi and a unit of theft detection. The whole system was divided into two individual subsystems. The first unit was required to be set up at the client location whereas the second one was needed to be set up at the control station of the electricity provider. Here, they proposed use of three MCUs. Two of the microcontrollers required to be set up at client’s end for the purpose of theft detection and IoT while the other MCU was needed to be set up at the control station. Nevertheless, this system required utilization of two individual systems to function which is not cost-efficient.

In India, Jain et. al. (2011) proposed digital prepaid meters to replace electromechanical meters. Here it was a rgued that inapt meter monitoring and reading causes a significant loss of electricity revenue which can be reduced by use of their proposed prepaid system. Mobile would be used as communication medium for this system of prepaid card to stay connected with electricity supplier. This proposed system was appropriate for collection of revenue from clients. However, this research did not mention the required infrastructure and module of communication clearly. Authors discovered that most systems in the reviewed literature entail high implementation costs. Thus, it is necessary to build a cost-efficient system that will establish remote energy meter controlling, monitoring and thwart tampering with the meter. We put forward a cost-effective smart energy metering system based on IoT which will be constantly in connection with the server administered by government authorities who will supervise any problem that arises in the energy meters set up. It will help decrease corruption in the power sector of Bangladesh and enhance meter security as well.

Chapter III (Theory)

3.1 Smart Meters

The conventional electromechanical energy meters have a great contribution in dissipation of energy. Mainly because they are post-paid meters, customers cannot keep track of their energy consumption, except on a monthly basis. However, Smart meters when programmed with home appliances can be used to control the electricity consumption. Moreover, Smart meters are less error prone. Smart Meters are electronic measurement devices used by utilities to communicate information for billing customers and operating their electric systems. It is simply consisting of a metering unit and a communication interface between the metering unit and the utility. The communication interface module is connected to the microcontroller to transmit and receive information between the utility and the metering unit installed in the house.

3.2 Microcontroller

A microcontroller is a compact integrated circuit designed to govern a specific operation in an embedded system. A typical microcontroller includes a processor, memory and input/output (I/O) peripherals on a single chip.

3.2.1 How do microcontrollers work?

A microcontroller is embedded inside of a system to control a singular function in a device. It does this by interpreting data it receives from its I/O peripherals using its central processor. The temporary information that the microcontroller receives is stored in its data memory, where the processor accesses it and uses instructions stored in its program memory to decipher and apply the incoming data. It then uses its I/O peripherals to communicate and enact the appropriate action. Microcontrollers are used in a wide array of systems and devices.

Devices often utilize multiple microcontrollers that work together within the device to handle their respective tasks.

3.2.2 What are the elements of microcontroller?

The core elements of a microcontroller are:

1. The processor (CPU) -- A processor can be thought of as the brain of the device. It processes and responds to various instructions that direct the microcontroller's function. This involves performing basic arithmetic, logic and I/O operations. It also performs data transfer operations, which communicate commands to other components in the larger embedded system.
2. Memory -- A microcontroller's memory is used to store the data that the processor receives and uses to respond to instructions that it's been programmed to carry out. A microcontroller has two main memory types:
   - Program memory, which stores long-term information about the instructions that the CPU carries out. Program memory is non-volatile memory, meaning it holds information over time without needing a power source.
   - Data memory, which is required for temporary data storage while the instructions are being executed. Data memory is volatile, meaning the data it holds is temporary and is only maintained if the device is connected to a power source.
3. I/O peripherals -- The input and output devices are the interface for the processor to the outside world. The input ports receive information and send it to the processor in the form of binary data. The processor receives that data and sends the necessary instructions to output devices that execute tasks external to the microcontroller.Other supporting elements of a microcontroller include:

- Analog to Digital Converter (ADC) -- An ADC is a circuit that converts analog signals to digital signals. It allows the processor at the center of the microcontroller to interface with external analog devices, such as sensors.
- Digital to Analog Converter (DAC) -- A DAC performs the inverse function of an ADC and allows the processor at the center of the microcontroller to communicate its outgoing signals to external analog components.
- System bus -- The system bus is the connective wire that links all components of the microcontroller together.

- Serial port -- The serial port is one example of an I/O port that allows the microcontroller to connect to external components. It has a similar function to a USB or a parallel port but differs in the way it exchanges bits.

3.2.3Microcontroller features

- A microcontroller's processor will vary by application. Options range from the simple 4-bit, 8-bit or 16-bit processors to more complex 32-bit or 64-bit processors. Microcontrollers can use volatile memory types such as random-access memory (RAM) and non-volatile memory types - this includes flash memory, erasable programmable read-only memory (EPROM) and electrically erasable programmable read-only memory (EEPROM).

- Generally, microcontrollers are designed to be readily usable without additional computing components because they are designed with sufficient onboard memory as well as offering pins for general I/O operations, so they can directly interface with sensors and other component.

3.2.4 Microcontroller applications

Microcontrollers are used in multiple industries and applications, including in the home and enterprise, building automation, manufacturing, robotics, automotive, lighting, smart energy, industrial automation, communications and internet of things (IoT) deployments.

One very specific application of a microcontroller is its use as a digital signal processor. Frequently, incoming analog signals come with a certain level of noise. Noise in this context means ambiguous values that cannot be readily translated into standard digital values. A microcontroller can use its ADC and DAC to convert the incoming noisy analog signal into an even outgoing digital signal.

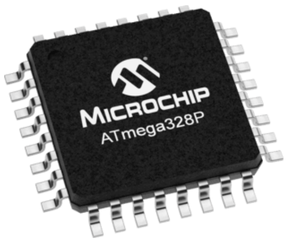


Fig 1**: ATmega328p**

3.2.5 Comparison of Microcontroller and Microprocessor

The distinction between microcontrollers and microprocessors has gotten less clear as chip density and complexity has become relatively cheap to manufacture and microcontrollers have thus integrated more "general computer" types of functionalities. On the whole, though, microcontrollers can be said to function usefully on their own, with a direct connection to sensors and actuators, where microprocessors are designed to maximize compute power on the chip, with internal bus connections (rather than direct I/O) to supporting hardware such as RAM and serial ports. Simply put, coffee makers use microcontrollers; desktop computers use microprocessors.

3.3ATmega328p

ATmega328P is a high performance yet low power consumption 8-bit AVR microcontroller that's able to achieve the most single clock cycle execution of 131 powerful instructions thanks to its advanced RISC architecture. It can commonly be found as a processor in Arduino boards such as Arduino Fio and Arduino Uno.

3.3.1 Features and Parametric

Features:

High endurance non-volatile memory segments

- In system self-programmable flash program memory
- Programming Lock for software security

Peripheral features

- Two 8-bit Timer/Counter with separate prescaler, compare mode.
- One 16-bit Timer/Counter with separate prescaler, compare mode, and capture mode
- Temperature measurement

- Programmable serial USART and watchdog timer with separate on-chip oscillator

Unique features compared to other microcontrollers (ARM, 8051, PIC):

- Power-on reset and programmable brown-out detection
- Internal calibrated oscillator
- External and Internal interrupt sources
- Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby.

Parametric:

Program Memory Type	Flash
Program Memory Size	32
CPU Speed (MIPS/DMIPS)	20
SRAM (KB)	2,048
Data EEPROM/HEF (bytes)	1,024
Digital Communication Peripheral	1-UART, 2-SPI, 1-I2C
Capture/Compare/PWM Peripheral	1 Input Capture, 1CCP, 6PWM
Timers/Counters	2 * 8-bit, 1* 16-bit
Number of Comparators	1
Temperature Range	-40 to 85deg
Operating Voltage Range (V)	1.8 to 5.5V
Pin Count	32
Low Power	Yes

3.3.2 Advantages and Disadvantages:

Advantages:

- Processors are simpler to use, with the usage of 8bit and 16bit instead of 32/64bit which are more complex

- Readily usable without additional computing components with 32k bytes of onboard self-programmable flash program memory as well as 23 programmable I/O lines
- Code Efficient, all 31 registers are directly connected to the arithmetic logic unit (ALU), making it 10 times faster than conventional CISC microcontrollers
- Optimized for AVR enhanced RISC instruction set

Disadvantages:

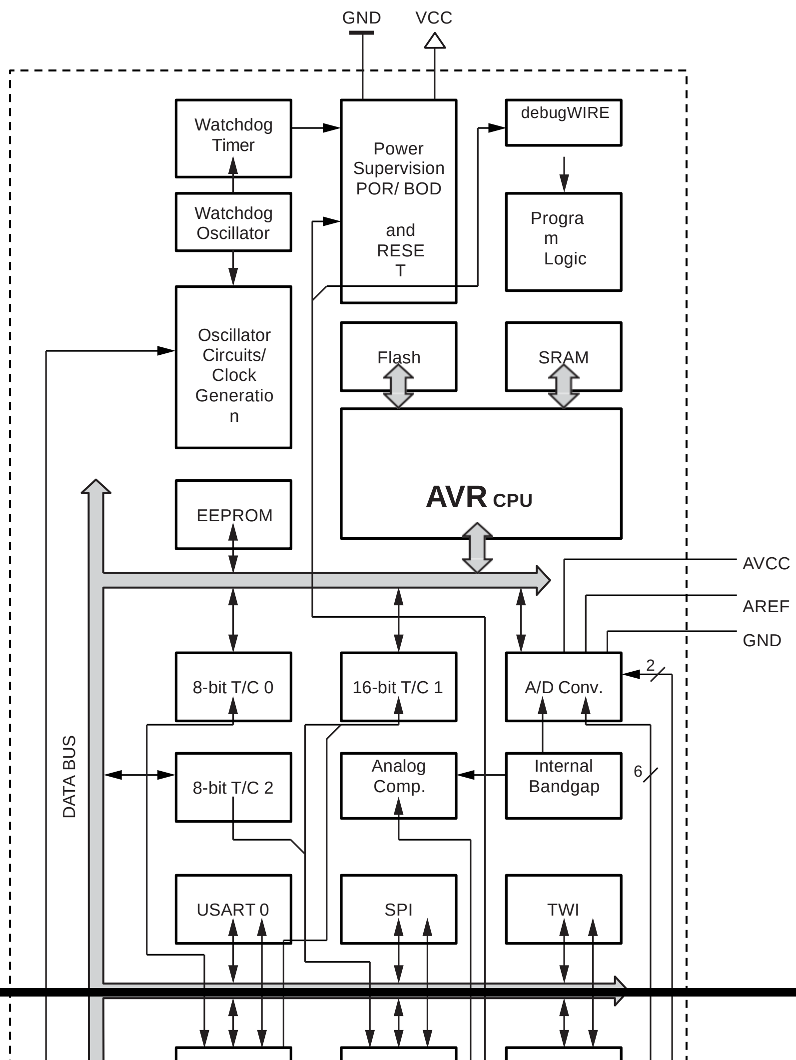
Lacks performance compared to higher bit microcontrollers.

3.2.3Product Applications

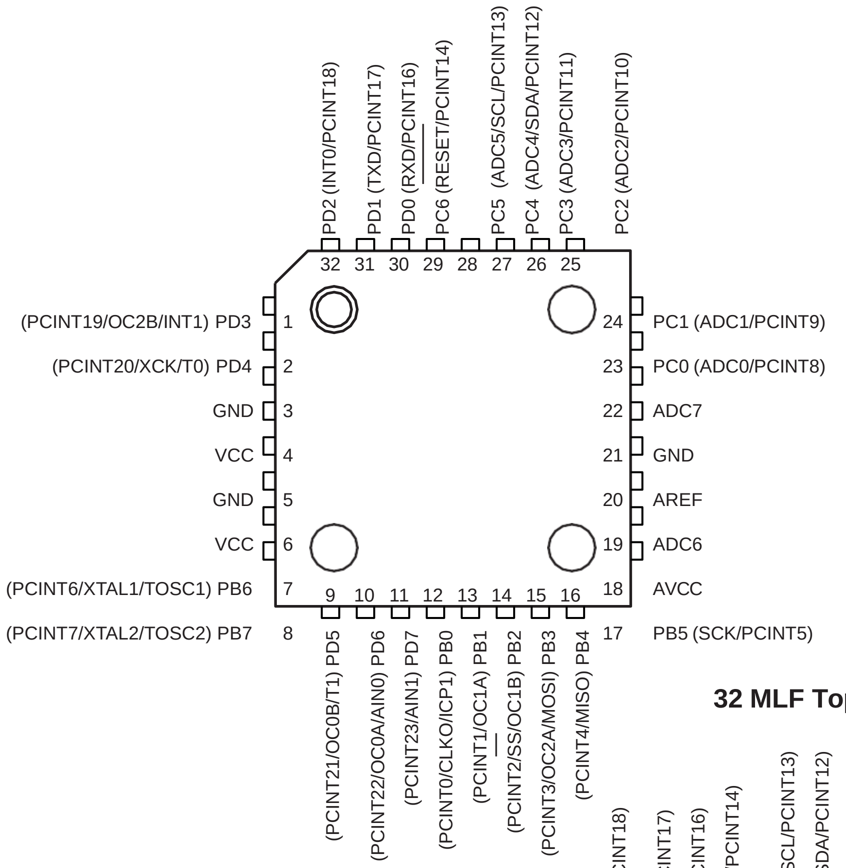
The ATmega328P is supported with a full suite of program and system development tools which includes: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

The fast PWM mode that provides a high-frequency PWM waveform generation allows for it to be suited for power regulation, rectification, and DAC applications.

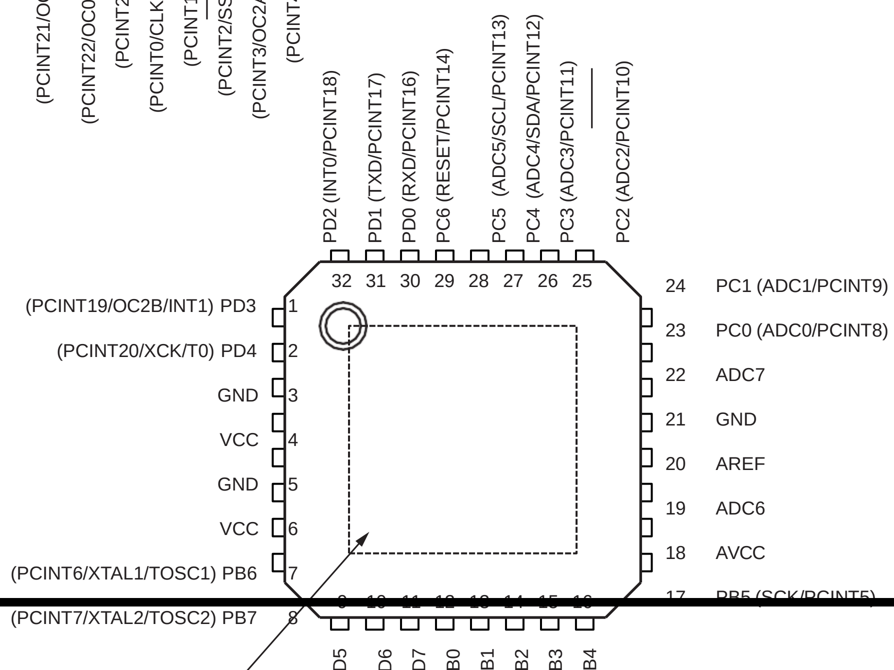
3.2.3 Block Diagram



3.2.4 Pin Configurations



32 MLF Top View



3.2.5Pin Description

- **VCC:**

Digital supply voltage

- **GND:**

Ground

- **Port B (PB7:0) XTAL 1/XTAL2/TOSC1/TOSC2:**

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting oscillator amplifier.

If the internal calibrated RC oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of port B are elaborated in Section 13.3.1 “Alternate Functions of Port B” on page 65 and Section 8. “System Clock and Clock Options” on page 24.

- **Port C (PC5:0):**

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5..0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

- **PC6/RESET:**

If the RSTDISBL fuse is programmed, PC6 is used as an input pin. If the RSTDISBL fuse is unprogrammed, PC6 is used as a reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 28-4 on page 261. Shorter pulses are not guaranteed to generate a reset.

The various special features of port C are elaborated in Section 13.3.2 “Alternate Functions of Port C” on page 68.

- **AREF:**

AREF is the analog reference pin for the A/D converter.

- **ADC7:6 (TQFP AND QFN/MLF Package Only):**

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

- **Disclaimer:**

Typical values contained in this datasheet are based on simulations and characterization of actual ATmega328P AVR® microcontrollers manufactured on the typical process technology. automotive min and max values are based on characterization of actual ATmega328P AVR microcontrollers manufactured on the whole process excursion (corner run).

- **Automotive Quality Grade:**

The ATmega328P have been developed and manufactured according to the most stringent requirements of the international standard ISO-TS-16949. This data sheet contains limit values extracted from the results of extensive characterization (temperature and voltage). The quality and reliability of the ATmega328P have been verified during regular product qualification as per AEC-Q100 grade 1. As indicated in the ordering information paragraph, the products are available in only one temperature.

Chapter IV(Hardware Modeling and Design)

4.1 Proposed Metering System:

The proposed IoT based metering system is designed to contribute in establishing energy efficient smart cities. Its main aim is to minimize the rapid increase in average electricity consumption issue and even cut off the theft . The conventional electromechanically post-paid metering system, where the consumer is charged on a monthly basis, provides no sense of the energy used until the end of the month. However, it guarantees that customers will always have electricity supplies, except during blackouts.

On the other hand, Prepaid meters offer many advantages both to the utility provider and the consumers. To the utility provider, this reduces many issues tremendously arising from meter readers such as delays, wrong and infrequent meter reading resulting in bulk amount of billing that consumers would need to pay and further consequent in not paying, disputes and so forth. Additionally, the prepayment metering system encourages users to control their energy consumption, in order to avoid cut off, due to zero credit. Thus, the proposed metering system uses the prepayment metering as an attempt for building energy efficient new cities. Figure 2 shows the complete proposed system design with the components and building components for each unit block, which will be further discussed in the this and the following sections.

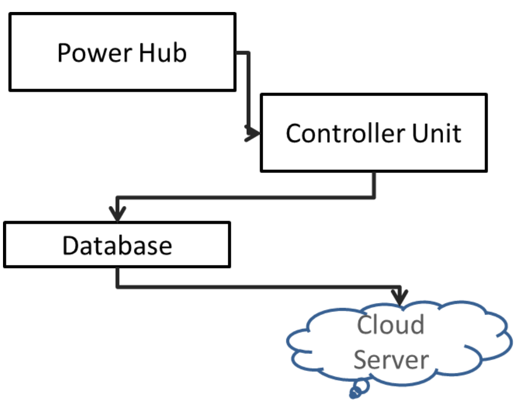


Fig 2: **Proposed Metering System Block Diagram**

4.1.1 Sensing Unit

The sensing unit comprises the voltage transformer and the current sensor that are connected to the main supply. The voltage transformer used was a 240v to 6v step down transformer. Whereas, the current sensor was a SCT-013-000 non-intrusive sensor, clipped over a single wire either live or neutral, to sense the passing current. This sensor's output is voltage, so, to get current reading, a burden resistor is connected to the two terminals of the sensor. The reading of both components are transferred to the next unit for processing.

4.1.2 Control Unit

The control unit consists mainly of the microcontroller, which is ATmega328 based Arduino Nano. Data received from the sensing unit are passed as an input to the built-in analog to digital converter (ADC) in the microcontroller. Afterwards, the data is processed then used in units in kilowatt hour and remaining balance and units.

These calculations were performed as follows: first calibration values for the current and voltage are calculated as shown in equations (1) and (2), where 0.707 value is used to convert the measured value to RMS value, 1024 is the maximum reading of Arduino input pins, 36 is the value of the burden resistance used with the current sensor and 240 is the maximum voltage that can be read by the meter. Current value read from the supply is calculated using a loop which takes 200 samples from the micro controller analog pins connected to the sensor and finds the maximum and minimum values. Then the current is calculated as shown in equation (3). On the other hand, voltage is calculated by taking the reading of the analog pin of the microcontroller connected to the transformer as shown in equation (4), using software function analogRead(). Figure 3 represents the voltage and current waveforms passed to the meter.

CalibrationI = 0.707/(1024*36)................(1)

CalibrationV = (0.707*240)/1024............(2)

Current=(maxI-minI) * CalibrationI..........(3)

Voltage=analogRead(A 1) * CalibrationV (4)

Afterwards, the control unit issues warnings for the user when either 12% or 8% is left in the balance. At zero balance, 24-hours allowance time is given to user, as a chance to recharge before electricity is cutoff. The calculated values are then passed to the database.

4.2 System Overview

System Overview IOT Based Smart Metering System based on Arduino ATmega 2560 which controls the various activities of the system. The system communicates over internet by using wireless communication by using Arduino Red Board. Two bulbs are used in this project to demonstrate as a valid consumer and a bulb to demonstrate an invalid consumer as theft. The foremost thing that this project facilitates is re-connection of transmission line to active grid. If an Energy Grid becomes faulty and there is another Energy Grid, the system switches the Transmission Lines towards this Grid thus facilitating uninterrupted electricity supply to that particular region whose Energy Grid went OFF. And if the theft is detected it will show up in LCD display then with the help of relay the theft is been cut off and the supply is again distributed to the consumers, this information of which Grid is active is updated over a webpage where the authorities can login and can view the updates. Apart from monitoring the Grid this project has advances capabilities of monitoring energy consumption and even detect theft of electricity. Theft conditions are simulated in the system using two switches. Switching one each time will simulate a theft condition and also will notify the authorities over the IOT interface. In this way the Smart Energy Grid project makes sure that the electricity supply is continuous and helps in maintaining a updated record of consumption and theft information which is quite a valuable information for the energy producing companies.

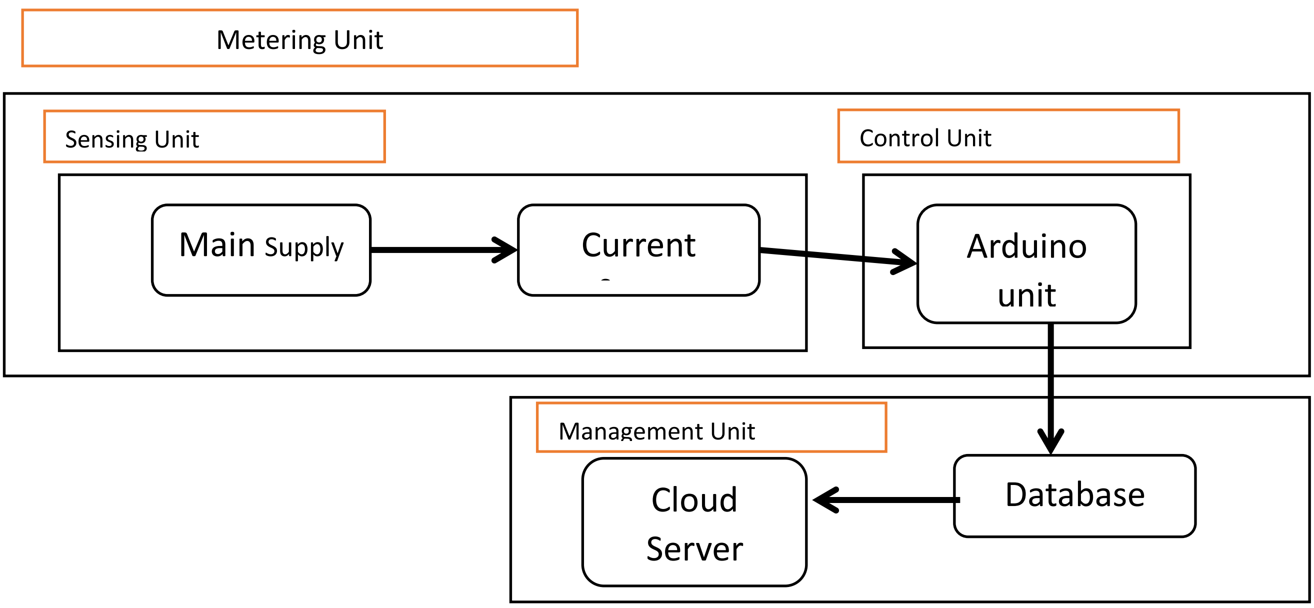


Fig 3: **Overall System Block Diagram**

Chapter V (Hardware Implementation and Software Programming)

5.1 Circuit Diagram

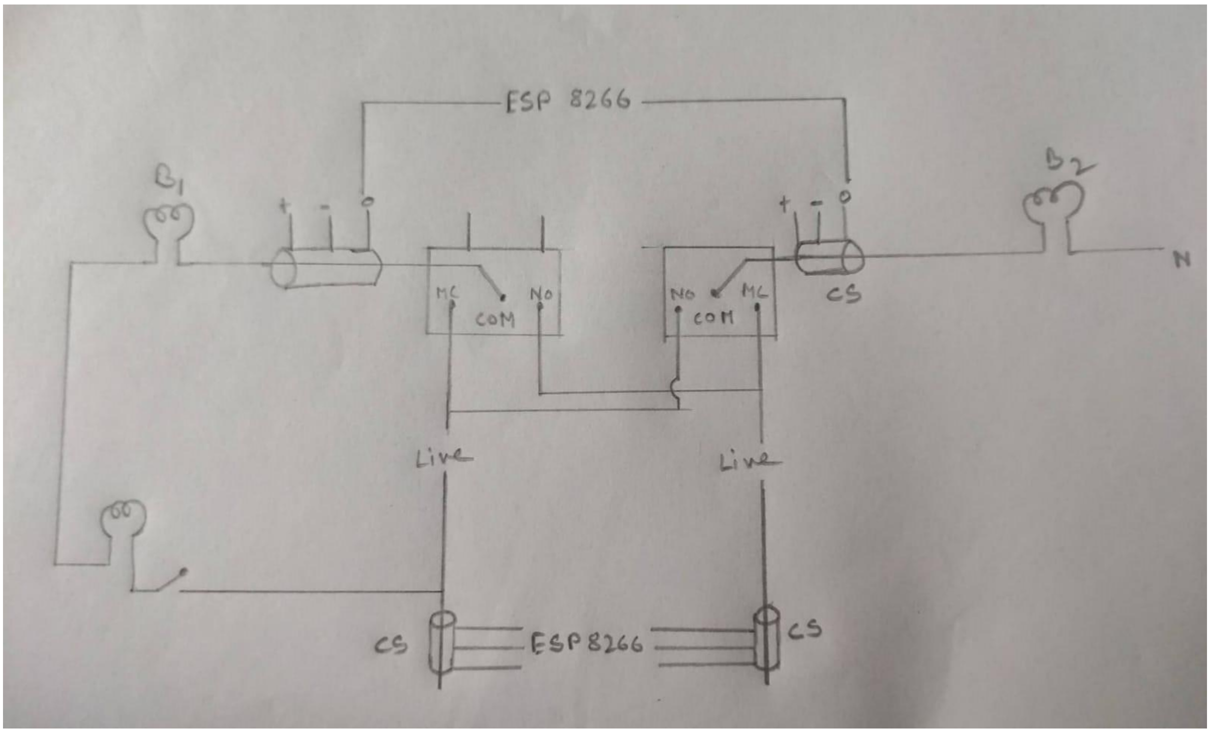


Fig 4: **Circuit Diagram of the Prototype**

5.2 Flowchart

Illustrates the final system working mechanism. The system operation starts by taking inputs from the current and voltage transformer. Then these inputs are passed to the analog to digital converter of the microcontroller unit where the billing data are calculated. Then a check is done on the billing data. the first check is whether the balance reached 8% or 12%. if the balance reached these low percentages warnings are issued. The second check is whether the balance is finished which if true the electricity is cut off and then the data is transmitted to the database which is managed through the website by the utility.

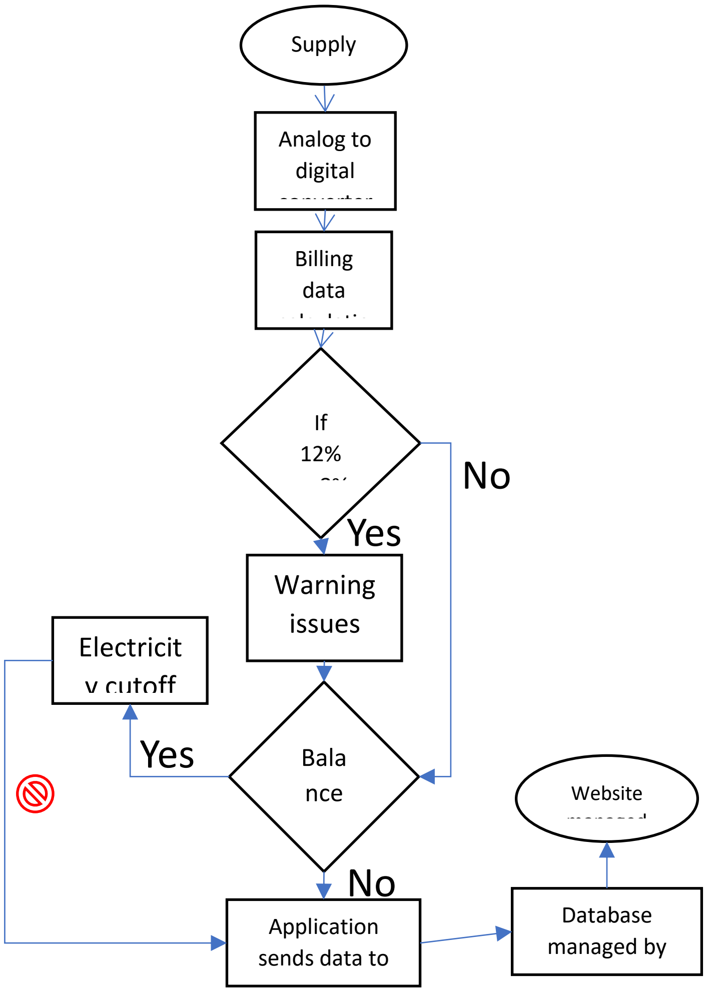


Fig 5: **System Flowchart**

5.3 Programming on Arduino

#define SAMPLES 300   //Number of samples you want to take everytime you loop

//#define ACS_Pin A7   //ACS712 data pin analong input

#define gcs1 A7

```
#define bcs1 A6
#define gcs2 A5
#define bcs2 A4

double gcs1_v=1.0;
double gcs2_v=1.0;
double bcs1_v=1;
double bcs2_v=1;


int g1=4;
int g2=3;

float High_peak,Low_peak;         //Variables to measure or calculate
float Amps_Peak_Peak, Amps_RMS;
int grid_switch=0;

void setup()
{
  Serial.begin(9600);
  pinMode(gcs1,INPUT);
  pinMode(gcs2,INPUT);
  pinMode(bcs1,INPUT);
  pinMode(bcs2,INPUT); //Define pin mode
  pinMode(g1,OUTPUT);
  pinMode(g2,OUTPUT);
  digitalWrite(g2,HIGH);
```

```
  digitalWrite(g1,HIGH);
  delay(1000);
  digitalWrite(g2,LOW);
  digitalWrite(g1,LOW);
  delay(500);
  digitalWrite(g2,HIGH);
  digitalWrite(g1,HIGH);



}


void loop()
{
  int i=1;
  double sum=0.0;
  double avg=0.0;


  while(i<=50){
  read_Amps(gcs1);                                //Launch the read_Amps function
  Amps_RMS = (Amps_Peak_Peak*0.3536*0.04)-0.1;     //Now we have the peak to peak value normally the formula requires only multiplying times 0.3536
    i++;                                   //but since the values will be very big you should multiply by 0.06, you can first not use it,
   sum=sum+Amps_RMS;                                  //do your calculations and compare them to real values measured by an Ammeter. eg: 0.06=Real value/Measured value
  }
  avg=sum/50;
```

```
  Serial.print("GCS1= ");
  Serial.println(avg);                    //Here I show the RMS value and the peak to peak value, you can print what you want and add the "A" symbol...
  gcs1_v=avg;
 i=0;
 sum=0.0;
 avg=0.0;




 while(i<=50){
  read_Amps(gcs2);                              //Launch the read_Amps function
  Amps_RMS = (Amps_Peak_Peak*0.3536*0.04)-0.1;     //Now we have the peak to peak value normally the formula requires only multiplying times 0.3536
      i++;                                      //but since the values will be very big you should multiply by 0.06, you can first not use it,
   sum=sum+Amps_RMS;                                    //do your calculations and compare them to real values measured by an Ammeter. eg: 0.06=Real value/Measured value
  }
  avg=sum/50;


  Serial.print("GCS2= ");
  Serial.println(avg);                    //Here I show the RMS value and the peak to peak value, you can print what you want and add the "A" symbol...
  gcs2_v=avg;
 i=0;
```

```
 sum=0.0;

 avg=0.0;




 while(i<=50){

  read_Amps(bcs1);                              //Launch the read_Amps function

  Amps_RMS = (Amps_Peak_Peak*0.3536*0.04)-0.1;     //Now we have the
peak to peak value normally the formula requires only multiplying times 0.3536

     i++;                                         //but since the values will be very big you
should multiply by 0.06, you can first not use it,

   sum=sum+Amps_RMS;                                    //do your calculations and
compare them to real values measured by an Ammeter. eg: 0.06=Real
value/Measured value

  }

  avg=sum/50;


  Serial.print("BCS1= ");

  Serial.println(avg);                   //Here I show the RMS value and the peak to
peak value, you can print what you want and add the "A" symbol...

 bcs1_v=avg;

 i=0;

 sum=0.0;

 avg=0.0;




while(i<=50){
```

```
  read_Amps(bcs2);                              //Launch the read_Amps function
  Amps_RMS = (Amps_Peak_Peak*0.3536*0.04)-0.1;     //Now we have the peak to peak value normally the formula requires only multiplying times 0.3536
     i++;                                       //but since the values will be very big you should multiply by 0.06, you can first not use it,
   sum=sum+Amps_RMS;                                     //do your calculations and compare them to real values measured by an Ammeter. eg: 0.06=Real value/Measured value
  }
  avg=sum/50;


  Serial.print("BCS2= ");
  Serial.println(avg);                    //Here I show the RMS value and the peak to peak value, you can print what you want and add the "A" symbol...
 bcs2_v=avg;
 i=0;
 sum=0.0;
 avg=0.0;


  Serial.println(gcs1_v);
  Serial.println(gcs2_v);
  if(gcs1_v<0)
  gcs1_v=1;
  if(gcs2_v<0)
  gcs2_v=1;
  if(gcs1_v<=0.42){


  Serial.println("Switching grid to 2");
  grid_switch=2;
```

```
  digitalWrite(g1,LOW);}
  else
  digitalWrite(g1,HIGH);

  if(gcs2_v<=0.3){
  Serial.println("Switching grid to 1");
  grid_switch=1;
  digitalWrite(g2,LOW);}
  else
  digitalWrite(g2,HIGH);
int p1=random(92,115);
int p2=random(92,115);

if((gcs1_v>0.6&&grid_switch==0)||(gcs2_v>0.6&&grid_switch==0)){
Serial.println("THEFT DETECTED GRID 1");
}
else if((gcs1_v>0.6&&grid_switch!=0)||(gcs2_v>0.6&&grid_switch!=0))
Serial.println("CANNOT DETERMINE POWER THEFT AS GRID IS AT
OVER DRAW");
else if((gcs1_v<0.6&&grid_switch==0)||(gcs2_v<0.6&&grid_switch==0)){
Serial.println(" TOTAL POWER CONSUMED FROM");
Serial.print("GRID 1   ");
Serial.println(p1);
Serial.print("GRID 2   ");
Serial.print(p2);}
```

```
  Serial.println();
}


void read_Amps(int ACS_Pin)            //read_Amps function calculate the
difference between the high peak and low peak
{                            //get peak to peak value
  int cnt;            //Counter
  High_peak = 0;      //We first assume that our high peak is equal to 0 and low
peak is 1024, yes inverted
  Low_peak = 1024;
      for(cnt=0 ; cnt<SAMPLES ; cnt++)          //everytime a sample (module
value) is taken it will go through test
      {
        float ACS_Value = analogRead(ACS_Pin); //We read a single value from
the module
               if(ACS_Value > High_peak)                //If that value is higher than the
high peak (at first is 0)
            {
              High_peak = ACS_Value;            //The high peak will change from 0 to
that value found
            }
               if(ACS_Value < Low_peak)                //If that value is lower than the
low peak (at first is 1024)
            {
              Low_peak = ACS_Value;             //The low peak will change from
1024 to that value found
            }
      }                                         //We keep looping until we take all samples and at
the end we will have the high/low peaks values
```

```
  Amps_Peak_Peak = High_peak - Low_peak;      //Calculate the difference
}
```

5.4 Photograph of The Prototype

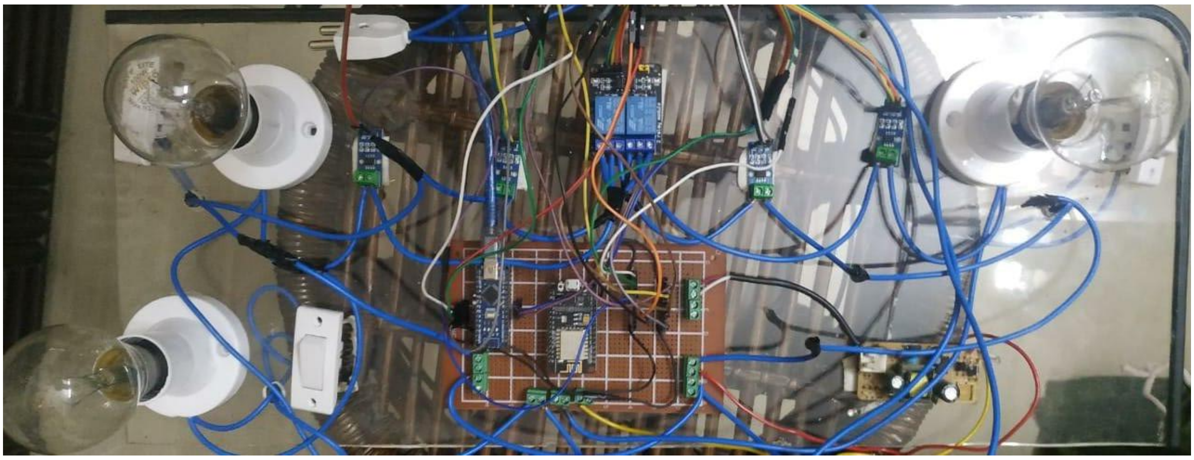


Fig 6: **System is Powered Off**

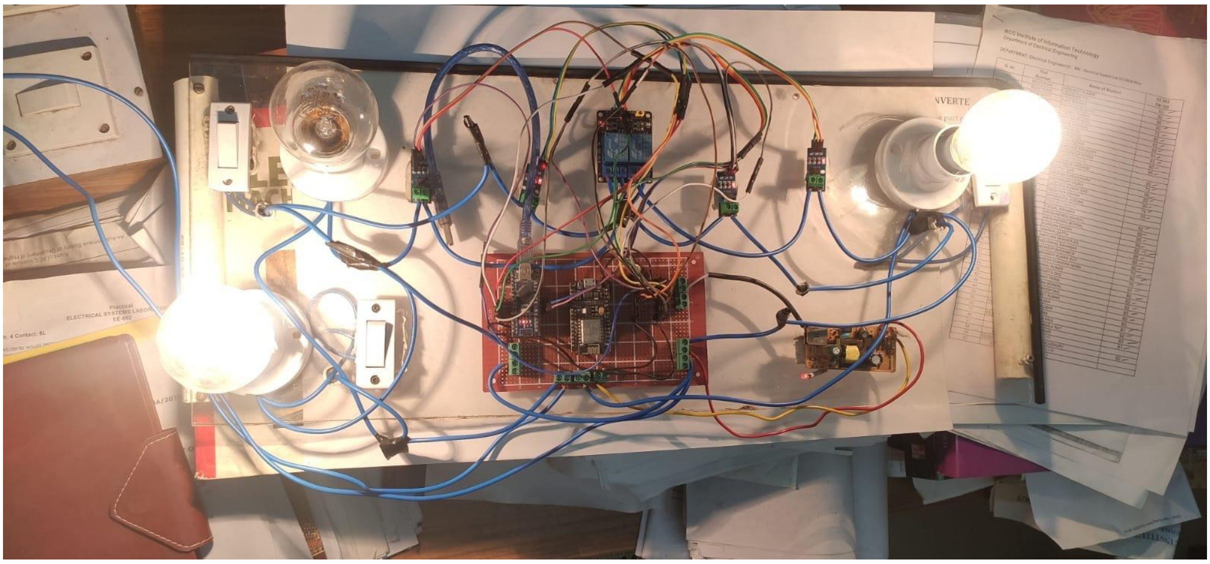


Fig 7: System is Powered On

Chapter VI: Conclusion

The advancement on technology and the need of electricity in every aspect of life has made the electrical power irreplaceable. Taking advantage of IOT smart meters has developed and took various shapes and types. Consequently, the design of a low cost, low power and user-friendly smart meter which helps in the conservation of energy can be of great importance.

The proposed smart meter design helps the users to track their energy consumption from LCD display and Android application. The users' data are saved and sent to the users' database through the mobile application, where the database is always updated by each meter's data then it can be edited by the utility website. The system has a 98% accuracy level compared with the existing conventional electromechanical meter. It is also 30% lower cost than its competing smart meter in the global market. The implemented Android application offers a consumption plans as well as online payment. The proposed design meter minimized the power consumption by 12% compared with the global market design.

Chapter VII: References

[1] Shalli Rani, etal. "A Novel Scheme for an Energy Efficient Internet of Things Based on Wireless Sensor Networks". Sensors ISSN 1424-8220, vol. 15, pp. 8603-28626. November 2015.

[2] Waleed Ejaz et al, "Efficient Energy Management for the Internet of Things in Smart Cities". Published in Efficient Energy Management for the Internet of Things in Smart Cities, vol. 55, Issue 1, January 2017.

[3] Jacques Benoit. “The Internet of Things and the energy sector: myth or opportunity”. Published in Power and Energy Automation Conference, Spokane, WA March 2016.

[4] Shikha Rastogi, Manisha Sharma and Pratibha Varshney. “Internet of Things based Smart Electricity Meters”, International Journal of Computer Applications, vol. 133. Issue 8, January 2016.

[5] Ellery E. Queen. “Smart Meters and Smart Meter Systems: A Metering Industry Perspective”. Washington, D.C., United States: Edison Electric Institute (EEI). March 2011, pp. 3-5.

[6] Patel Umang and Modi Mitul. “A Review on Smart Meter System”. International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, vol. 3. Issue 12, pp. 70-73. December 2015.

[7] Al-Naima, Jalil,” Building a prototype Prepaid Electricity Metering System Based on RFID”, International Journal of Electronics and Electrical Engineering, vol. 1, Issue 1, Iraq. 12 March 2012.

[8] Au Thien Wan, Suresh Sankaranarayanan and Siti Nurafifah Binti Sait. “Smart Agent Based Prepaid Wireless Energy Meter”. International Conference on Cloud Computing and Internet of Things, pp. 77-81. December 2014.

[9] Au Thien Wan, Suresh Sankaranarayanan and Siti Nurafifah Binti Sait. “Smart Agent Based Prepaid Wireless Energy Meter”. International Conference on Cloud Computing and Internet of Things, pp. 77-81. December 2014.

[10] Mohammed Reyasudin Basir Khan and Razali Jidin. “USB Data Acquisition Device for Energy Monitoring,” presented at Student Conference on Research and Development, Selangor, Malaysia, 2011.

[11] V.Preethi, M.Tech and G. Harish. “Design and Implementation of Smart Energy Meter”. International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India. August 2016.