CHAPTER 1
INTRODUCTION

The Internet of Things (IoT) is interring communication of embedded devices using networking technologies. The IoT will be one of the important trends in future, can affect the networking, business and communication. In this paper, proposing a IoT based prepaid energy meter. Most of the energy meters are designed to bill as per the units of energy consumed. These meters need to be manually read by people in order to provide monthly/quarterly bills. We here propose a IoT based smart electricity meter. The system is designed to allow amount of energy to be used as long as the account has balance pending. It also allows the operator to recharge the user account using IoT. The system first accepts account recharge and allows to use only limited units of energy as per recharge and then cuts off the supply. The prepaid electricity billing meter could be widely used to provide a new more customized electricity billing system, where users may recharge when they intend to use that facility. It also consists of a ESP8266 module that allows the operator to recharge the meter remotely using email message service. This puts forward an innovative electricity billing and “use as needed” electricity usage scheme. It also eliminates the need for manual electricity meter reading tasks. This meter is digital meter with short circuit protection, Over voltage protection, Anti-Theft detection.

CHAPTER 2
BLOCK DIAGRAM

Block diagram of IoT Based Energy Meter Reading Using Node MCU will show you the complete visualblock representation to understand the project idea.

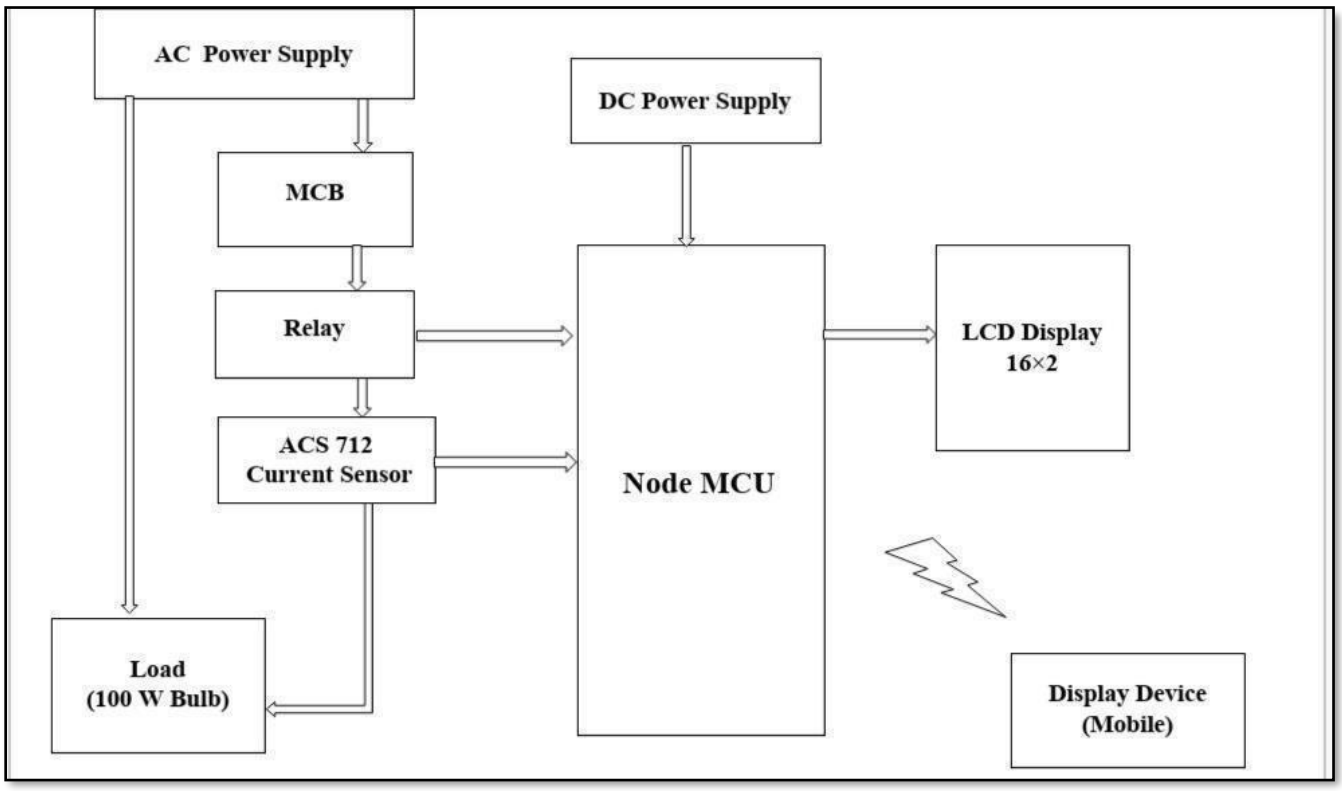


Fig 2.1: Block Diagram of Prepaid Energy Meter

Components Used in Project:

- Node MCU
- Relay
- 16x2 LCD Module
- MCB
- Diode 1N4007
- COPPER PCB
- ACS712 Current Sensor

CHAPTER 3
CIRCUIT DIAGRAM

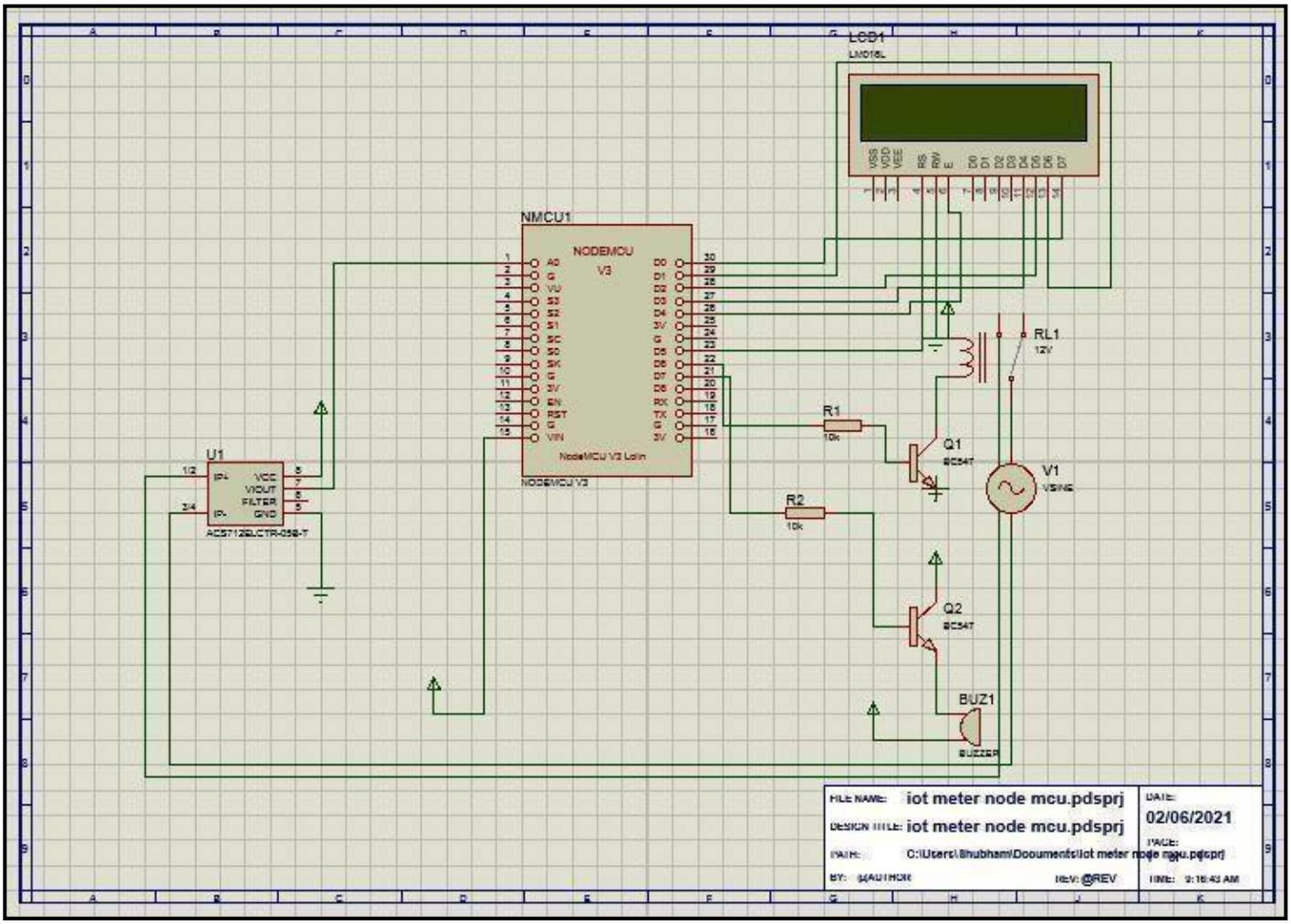


Fig 3.1: Simulation of Prepaid Energy Meter

The complete circuit diagram for IoT based Energy Meter is shown below. Although it is very simple, you should follow the graphical representation for a better understanding and make sure the connections are correct. Be advised that working with mains requires practice and hence do not build this circuit if you are notsure how to do it.

We have used NodeMCU with ACS712 Current Sensor, the current sensor will measure the current consumed by our AC load and the NodeMCU will measure this current, calculate the power (assuming the voltage is constant) and send the power value to a cloud platform like Adafruit IO. A visual info-graphic circuit diagram is also given below for your convenience.

CHAPTER 4
INTRODUCTION TO NODE MCU

- **INTRODUCTION TO NODE MCU: -**

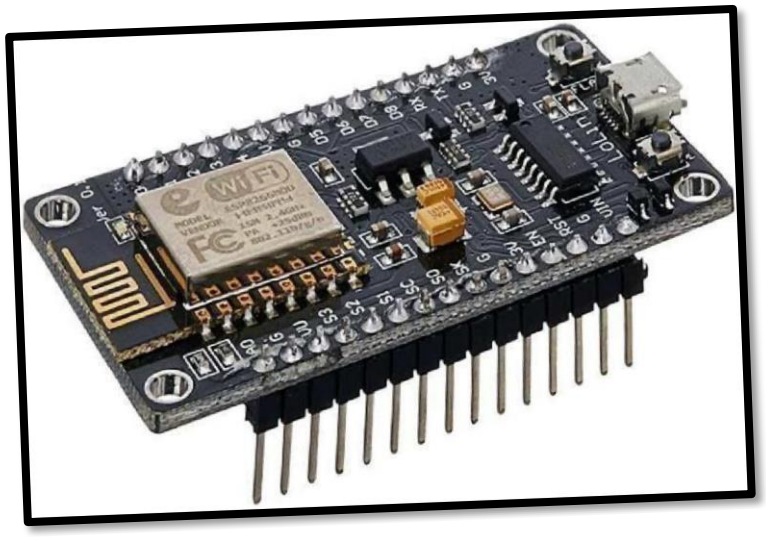


Fig 4.1: Node MCU

NodeMCU is an open-source Lua based firmware and development board specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module.

The NodeMCU ESP8266 development board comes with the ESP-12E module containing ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects.

NodeMCU can be powered using Micro USB jack and VIN pin (External Supply Pin). It supports UART, SPI, and I2C interface.

PIN DIAGRAM: -

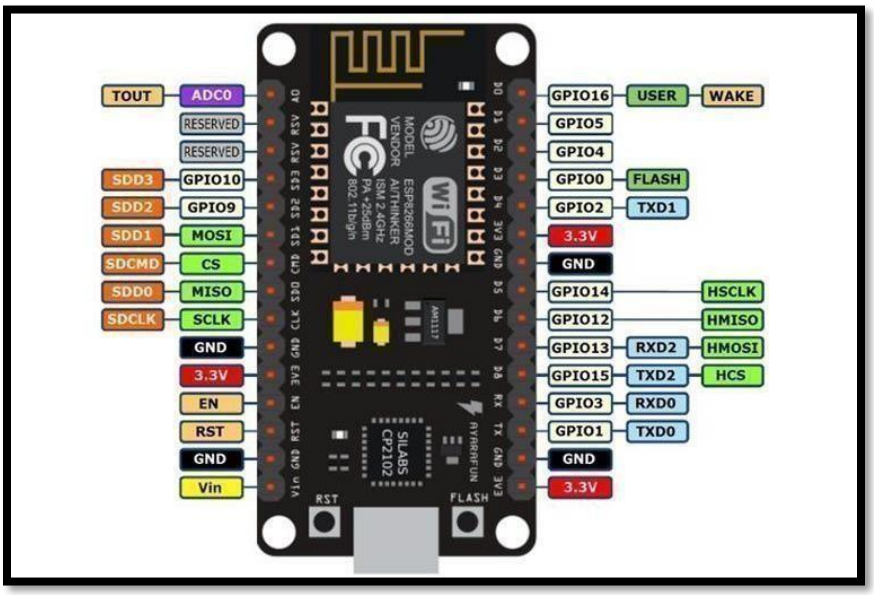


Fig 4.2: Pin diagram of NODE MCU (ATMEGA328P) Pin

Descriptions:

Table 4.1: Pin Description of Node MCU

Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	**Micro-USB:** NodeMCU can be powered through the USBport **3.3V:** Regulated 3.3V can be supplied to this pin to powerthe board **GND:** Ground pins **Vin:** External Power Supply
Control Pins	**EN, RST**	The pin and the button resetsthe microcontroller

Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 to GPIO16	NodeMCU has 16 general purpose input-output pins on itsboard
SPI Pins	SD1, CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.

FEATURES: -

1. Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
2. Operating Voltage: 3.3V
3. Input Voltage: 7-12V
4. Digital I/O Pins (DIO): 16
5. Analog Input Pins (ADC): 1
6. UARTs: 1
7. SPIs: 1
8. I2Cs: 1
9. Flash Memory: 4 MB
10. SRAM: 64 KB
11. Clock Speed: 80 MHz
12. USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
13. PCB Antenna
14. Small Sized module to fit smartly inside your IoT projects

CHAPTER 5
HARDWARE DESCRIPTION

- **ACS712 Current Sensor Module:-**

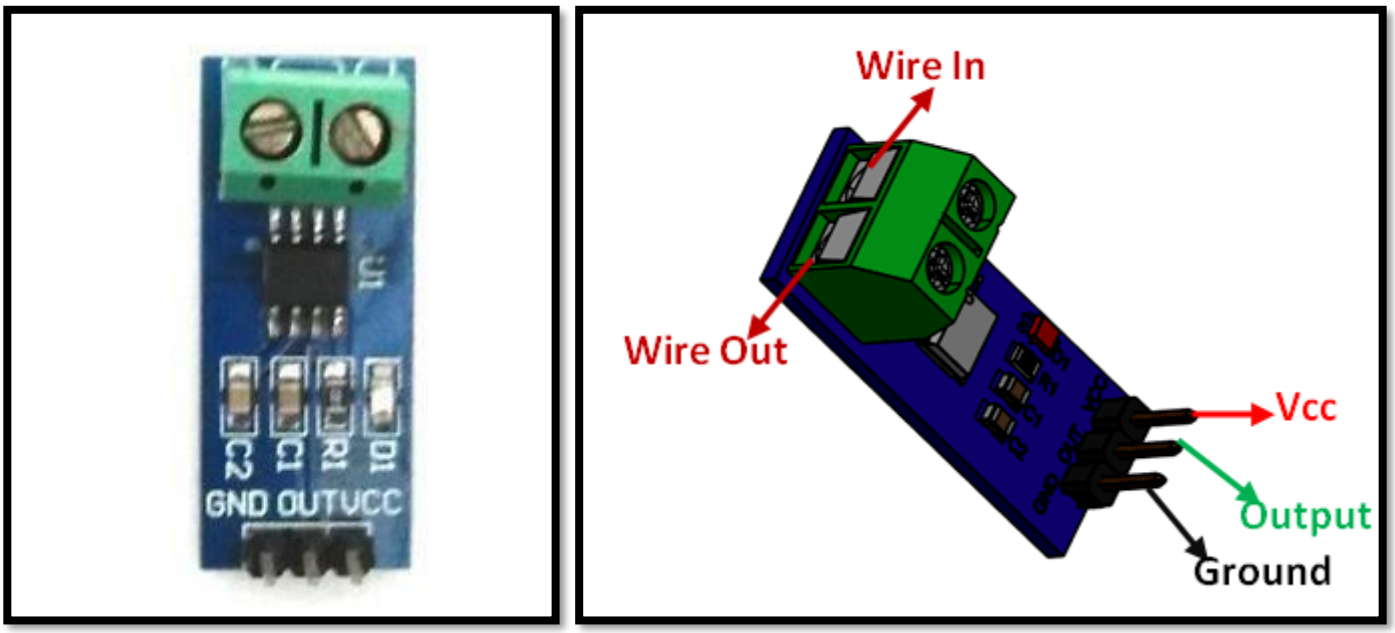


Fig 5.1: ACS712 Current Sensor Module

- **Pin Configuration:**

Table 5.1: Pin Configuration of ACS712 Current Sensor

Pin Number	Pin Name	Description
1	Vcc	Input voltage is +5V for typical applications
2	Output	Outputs Analog voltage proportional to current
3	Ground	Connected to ground of circuit
T1	Wire In	The wire through current has to be measured is connected here
T2	Wire Out	

- **Specifications**
  1. Measures both AC and DC current
  2. Available as 5A, 20A and 30A module
  3. Provides isolation fromthe load
  4. Easyto integrate with MCU, since it outputs analog voltage
  5. Scale Factor

- **Relay**

The advantage of relays is that it takes a relatively small amount of power to operate the relay coil, but the relay itself can be used to control motors, heaters, lamps or AC circuits which themselves can draw a lot more electrical power.

The electro-mechanical relay is an output device (actuator) which come in a whole host of shapes, sizes and designs, and have many uses and applications in electronic circuits. But while electrical relays can be used to allow low power electronic or computer type circuits to switch relatively high currents or voltages both "ON" or "OFF", some form of relay switch circuit is required to control it.

The design and types of relay switching circuits is huge, but many small electronic projects use transistors and MOSFETs as their main switching device as the transistor can provide fast DC switching (ON-OFF) control of the relay coil from a variety of input sources so here is a small collection of some of the more common ways of switching relays.

- **NPN Relay Switch Circuit**

A typical relay switch circuit has the coil driven by a NPN transistor switch, TR1 as shown depending on theinput voltage level. When the Base voltage of the transistor is zero (or negative), the transistor is cut-off and acts as an open switch. In this condition no Collector current flows and the relay coil is de-energized because being current devices, if no current flows into the Base, then no current will flow through the relay coil.

If a large enough positive current is now driven into the Base to saturate the NPN transistor, the current flowing from Base to Emitter (B to E) controls the larger relay coil current flowing through the transistor from the Collector to Emitter.

For most bipolar switching transistors, the amount of relay coil current flowing into the Collector would be somewhere between 50 to 800 times that of the required Base current to drive the transistor into saturation.The current gain, or beta value (β) of the general purposessBC109 shown is typically about 290 at 2ma (Datasheet).

The design and types of relay switching circuits is huge, but many small electronic projects use transistors and MOSFETs as their main switching device as the transistor can provide fast DC switching (ON-OFF) control of the relay coil from a variety of input sources so here is a small collection of some of the more common ways ofswitching relays.

A typical relay switch circuit has the coil driven by a NPN transistor switch, TR1 as shown depending on theinput voltage level. When the Base voltage of the transistor is zero (or negative), the transistor is cut-off and acts as an open switch. In this condition no Collector current flows and the relay coil is de-energized because being current devices, if no current flows into the Base, then no current will flow through the relay coil.

The advantage of relays is that it takes a relatively small amount of power to operate the relay coil, but the relay itself can be used to control motors, heaters, lamps or AC circuits which themselves can draw a lot more electrical power.

But while electrical relays can be used toallow low power electronic or computer type circuits to switch relatively high currents or voltages both “ON” or “OFF”, some form of relay switch circuit is required to control it.

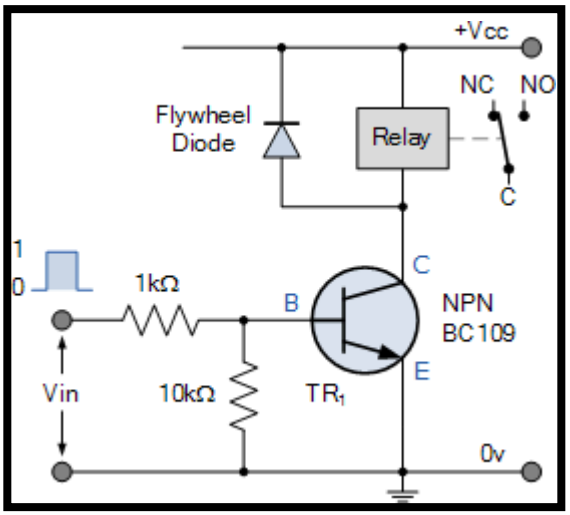
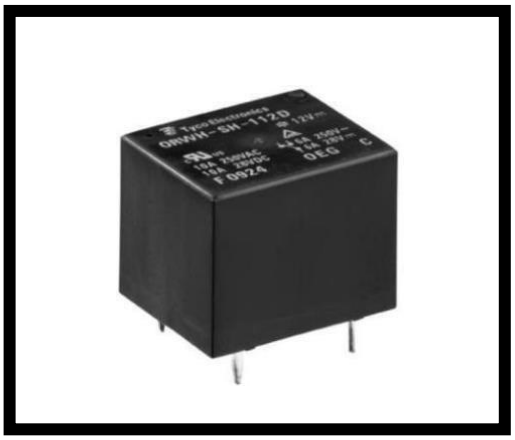


Fig 5.2: Circuit Diagram of Relay Driver Circuit



Relay

- **16x2 LCD Module**

We come across LCD displays everywhere around us. Computers, calculators, television sets, mobile phones, digital watches use some kind of display to display the time. An LCD is an electronic display module which uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used in diys and circuits. The 16×2 translates o a display 16 characters per line in 2 such lines. In this LCD each character is displayed in a 5×7 pixel matrix.

RS (Register select)

A 16X2 LCD has two registers, namely, command and data. The register select is used to switch from one register to other. RS=0 for command register, whereas RS=1 for data register.

Command Register: The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. Processing for commands happen in the command register.

Data Register: The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD. When we send data to LCD it goes to the data register and is processed there. When RS=1, data register is selected.

Displaying Custom Characters on 16X2 LCD

Generating custom characters on LCD is not very hard. It requires the knowledge about custom generated random access memory (CG-RAM) of LCD and the LCD chip controller. Most LCDs contain Hitachi HD4478 controller. CG-RAM is the main component in making custom characters. It stores the custom characters once declared in the code. CG-RAM size is 64 bytes providing the option of creating eight characters at a time. Each character is eight bytes in size.

CG-RAM address starts from 0x40(Hexadecimal) or 64 in decimal. We can generate custom charactersat these addresses. Once we generate our characters at these addresses, now we can print them on the LCD.

Table 5.2: LED Interface character address with command

CG-RAM Characters	CG-RAM Address (Hexadecimal)	Commands to display Generated Characters
1st Character	0x40	0
2nd Character	0x48	1
3rd Character	0x56	2
4th Character	0x64	3
5th Character	0x72	4
6th Character	0x80	5
7th Character	0x88	6
8th Character	0x96	7

- **MCB:**

A miniature circuit breaker automatically switches off electrical circuit during abnormal condition of the network means in over load condition as well as faulty condition. Nowadays we use an MCB in low voltage electrical network instead of fuse. The fuse may not sense it but the miniature circuit breaker does it in a more reliable way. MCB is much more sensitive to over current than fuse. Handling a MCB is electrically safer than a fuse. Quick restoration of supply is possible in case of fuse as because fuses must be re-wireable or replaced for restoring the supply. Restoration is easily possible by just switching it ON. Let's look at the working of the miniature circuit breaker.

MCB

Working principle of the miniature circuit breaker

Whenever continuous over current flows through MCB, the bimetallic strip is heated and deflects by bending. This deflection of bimetallic strip releases mechanical latch. As this mechanical latch is attached with operating mechanism, it causes to open the miniature circuit breaker contacts, and the MCB turns off thereby stopping the current to flow in the circuit. To restart the flow of current the MCB must be manually turned ON. This mechanism protects from the faults arising due to over current or over load. But during short circuit condition, current rises suddenly, causing electromechanical displacement of plunger associated with a tripping coil or solenoid. The plunger strikes the trip lever causing immediate release of latch mechanism consequently open the circuit breaker contacts. This was a simple explanation of miniature circuit breaker working principle.

An MCB is very simple, easy to use and is not generally repaired. It is just easier to replace. The trip unit is the main part, responsible for its proper working. There are two main types of trip mechanism. A bi-metal provides protection against over load current and an electromagnet provides protection against short-circuit current.

- **MCB operation**

If circuit is overloaded for a long time, the bi-metallic strip becomes over heated and deformed. This deformation of bi metallic strip causes, displacement of latch point. The moving contact of the MCB is arranged by means of spring pressure, with this latch point, that a little displacement of latch causes, release of spring and makes the moving contact to move for opening the MCB.

The current coil or trip coil is placed so that during short circuit fault the magneto-motive force (mmf) of the coil causes its plunger to hit the same latch point and make the latch to be displaced. Again, when operating lever of the miniature circuit breaker is operated by hand, that means when MCB goes off position manually, the same latch point is displaced as a result moving contact separated from fixed contact in same manner.

It may be due to deformation of bi-metallic strip, or increased mmf of trip coil or maybe manual operation, the same latch point is displaced and same deformed spring is released, which ultimately responsible for movement of the moving contact. When the moving contact separated from fixed contact, there may be a high chance of arc. This arc then goes up through the arc runner and enters arc splitters and is finally quenched. When we switch it on, we reset the displaced operating latch to its previous on position and the MCB is ready for another switch off or trip operation.

- **Software used:**

Proteus:

The Proteus Design Suite is a proprietary software tool suite used primarily for electronicdesign automation.The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

It was developed in Yorkshire, England by Labcenter Electronics Ltd and is available inEnglish, French, Spanish and Chinese languages.

The Proteus Design Suite is a Windows application for schematic capture, simulation, and PCB (Printed Circuit Board) layout design. It can be purchased in many configurations, depending on the size of designs being produced and the requirements for microcontroller simulation.

Schematic capture in the Proteus Design Suite is used for both the simulation of designs and as the design phase of a PCB layout project. It is therefore a core component and is included with all product configurations.

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it.

- **PCB Layout:**

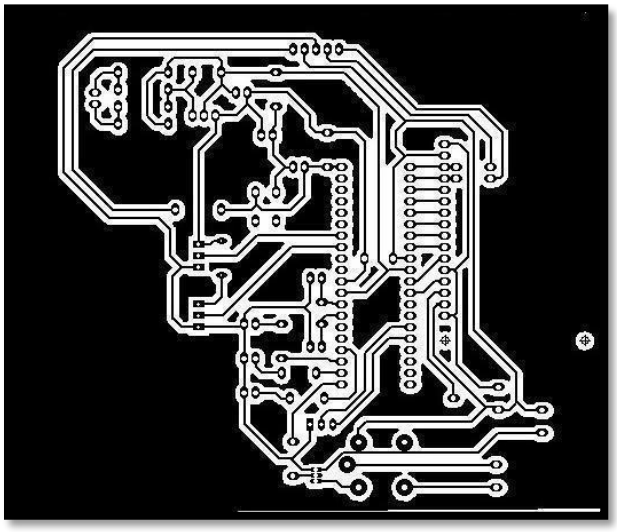


Fig 5.3: PCB Layout

- **Dip Trace:**

DipTrace is an EDA/CAD software for creating schematic diagrams and printed circuit boards. The developers provide a multi-lingual interface and tutorials (currently available in English and 21 other languages). DipTrace has 4 modules: schematic capture editor, PCB layout editor with built-in shape-based autorouter and 3D-preview & export, component editor, and pattern editor.

- **Basic Features:**

Simple user interface

Multi-sheet and hierarchical schematics

High-speed and differential signal routing

Smart manual routing modes

Wide import/export capabilities

CHAPTER 6
PROGRAMMING

ALGORITHM:

Algorithm is representation of working process of a particular task in terms if theoretical as shown in figure. Sequence of operation for smart electricity meter in electricity system: The following sequence of operation has been followed for controlling the electricity theft of energy meter.

Process:

Step1: Calculate the basic parameters for energy usage:

i. V and I terms using CT/PT

ii. Apparent power

ii. Energy

Step 2: Display energy reading on LCD;

Step 3: Enable IOT application to:

i. Obtain data from microcontroller.

ii. Generate billing information based on amount of energyconsumed

ii. Send billing information to consumer

Step 4: consumer gets the billing notifications through messages

Step 5: when balance is low IoT will send the notification to the consumer.

Step 6: if there islow balance in the sim then power supply will turn off.

FLOW CHART:

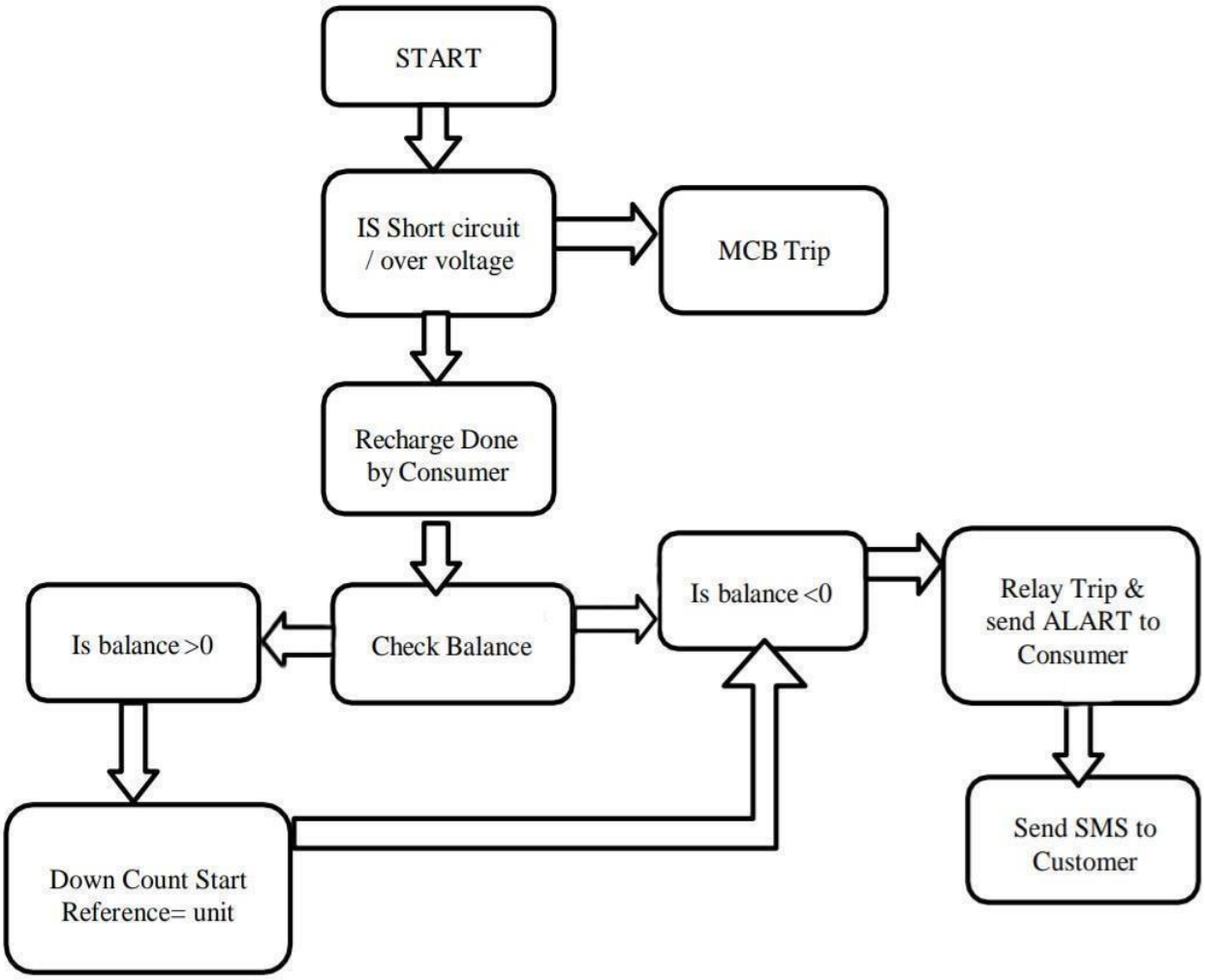


Fig 6.1: Flow chart of Prepaid Energy Meter

PROGRAM:

```
#include
<ESP8266WiFi.h>
#include
<ThingSpeak.h>
#include
<LiquidCrystal.h>
LiquidCrystal lcd(D5,D4,D3,D2,D1,D0);



const int Sensor_Pin = A0;
unsigned int Sensitivity = 185;           // 185mV/A for 5A, 100 mV/A for 20A and 66mV/A
for30AModule
float Vpp = 0; // peak-peak
voltagefloatVrms = 0; // rms
voltage
float Irms = 0;// rms current
float Supply_Voltage = 233.0;                 // reading from DMM
float Vcc = 5.0;            // ADC reference voltage // voltage at 5V
pinfloatpower = 0;          // power in watt
float Wh =0 ;               // Energy in
kWhunsigned long last_time =0;
unsigned long current_time
=0;unsigned long interval = 100;
unsigned int calibration = 100; // V2 slider calibrates
thisunsignedint pF = 85;            // Power Factor default 95
unsigned int bill_amount = 0;          // 30 daycost as present energyusage incl approx
PFunsignedint energyTariff = 8.0; // Energycost in INR per unit (kWh)
unsigned int A=0;

int
connection=D6;
int
buzzer=D7;//relay

unsigned long counterChannelNumber = 1389862;                  // Channel ID
const char * myCounterReadAPIKey = "5KIO37U4CH3HEXO9";                  // Read API Key
const int FieldNumber1 = 6;                                 // The field you wish to read //TEST
WASDONEON FIELD 3CHANGE TO FIELD 4
//const int FieldNumber2 = 2;                                 // The field you wish to read

String apiKey= "U73H1900HM0GZGML";               // Enter your Write API key
fromThingSpeak
```

```
const char *ssid =  "C15";            // replace with your wifi ssid and wpa2 key
//const char *pass =
"idle@3736";constchar *pass =
"12345678";
const char* server = "api.thingspeak.com";


WiFiClient client;

void getACS712() { // for ACVpp
  = getVPP();
  Vrms = (Vpp/2.0) *0.707;
  Vrms = Vrms - (calibration / 10000.0);                // calibtrate to zero with sliderIrms
  = (Vrms * 1000)/Sensitivity;
  if((Irms > -0.015) && (Irms < 0.008)){  // remove  low end chatterIrms = 0.0;
  }
  power= (Supply_Voltage * Irms) * (pF /
  100.0);last_time = current_time;
  current_time = millis();
  Wh = Wh+ power *(( current_time -last_time) /3600000.0) ; // calculating energy in Watt-
Hour
  bill_amount = Wh * energyTariff;
  Serial.print("voltage: ");
  Serial.print(String(Supply_Voltage,
  3));Serial.println(" V");
  lcd.clear();
  lcd.setCursor(0
  , 0);
  lcd.print("V:");
  lcd.print(String(Supply_Voltage, 3));lcd.println(" V");

  Serial.print("Irms: ");
  Serial.print(String(Irms,
  3));Serial.println(" A");
  lcd.setCursor(7, 0);
  lcd.print("I:");
  lcd.print(String(Irms,
  3)); lcd.println(" A");

  Serial.print("P: ");
  Serial.print(String(power,
  3));
```

```
  Serial.println(" W");
  lcd.setCursor(0, 1);
  lcd.print("P:");
  lcd.print(String(power, 3));
  lcd.println(" W");
 /*
 if(power>330) {
 theft=1;
 lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Theft Detected ");
  digitalWrite(buzzer,HIGH);
  delay(3000);
  digitalWrite(buzzer,LOW);
 }
 else
 {
 theft=0;
 }

  Serial.print(" Bill Amount: INR");
  Serial.println(String(bill_amount, 2));
  lcd.setCursor(7, 1);
  lcd.print(" Bal:");
  lcd.print(bill_amount);
  lcd.print("        ");
  */
}

float getVPP()
{
  float result; int
  readValue;
  int maxValue = 0;
  int minValue = 1024;
  uint32_tstart_time = millis();
  while((millis()-start_time) < 950) //read every 0.95 Sec
  {
     readValue = analogRead(Sensor_Pin);if
     (readValue > maxValue)
     {
         maxValue = readValue;
     }
     if (readValue < minValue)
```

```
        {
            minValue = readValue;
        }
      }
       result = ((maxValue - minValue) * Vcc) / 1024.0;return
       result;
     }



   void setup()
{

        Serial.begin(115200);
        lcd.begin(16, 2);
        pinMode(Sensor_Pin,INPUT);
        pinMode(connection,OUTPUT);
        pinMode(buzzer,OUTPUT);
        lcd.setCursor(0, 0);
        lcd.print("prepaid meter");
        lcd.setCursor(0, 1);
        lcd.print("System");
        delay(10);


        Serial.println("Connecting to ");
        Serial.println(ssid);


        WiFi.begin(ssid, pass);

       while (WiFi.status() != WL_CONNECTED)
       {
            delay(500);
            Serial.print(".");
       }
        Serial.println(""); Serial.println("WiFi
        connected");ThingSpeak.begin(client);
        delay(1000);
      A = ThingSpeak.readLongField(counterChannelNumber, FieldNumber1,myCounterReadAPIKey);
       Serial.println("A");
      //Serial.println("OK");
      getACS712();
unsigned int amount= A -
```

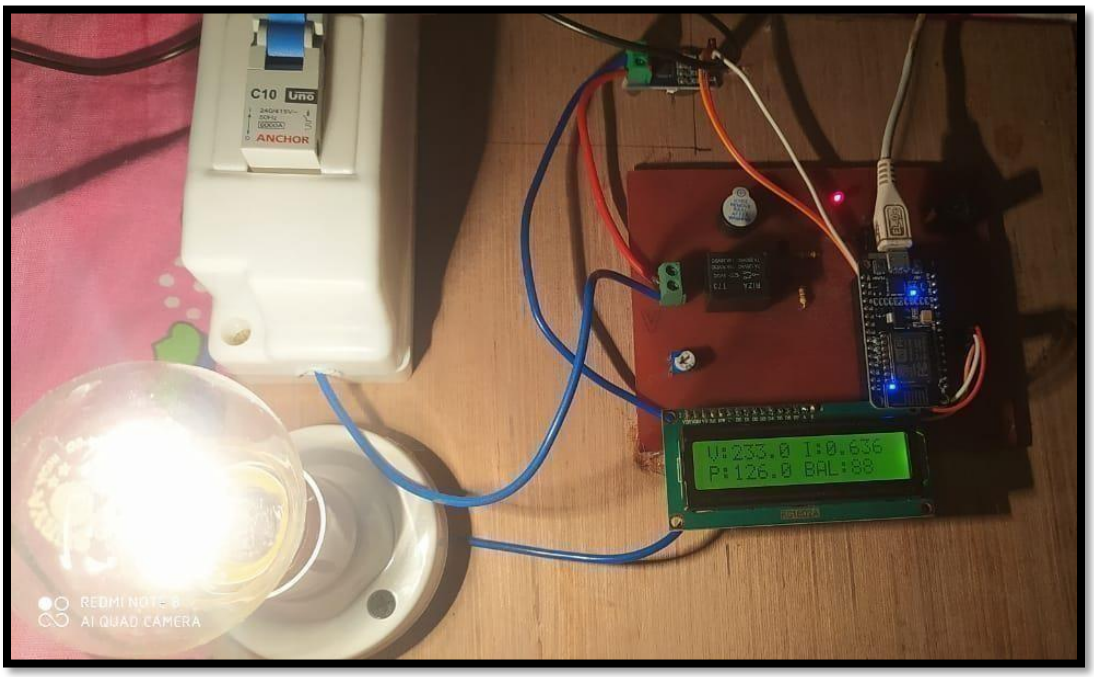
```
bill_amount; lcd.setCursor(7, 1);
lcd.print("BAL:");
lcd.print(amount);
      lcd.print("         ");
        delay(16000);

                    if (client.connect(server,80))          //    "184.106.153.149" or api.thingspeak.com
                  {

                      String postStr = apiKey;
                      postStr +="&field1=";
                      postStr += String(Supply_Voltage);
                      postStr +="&field2=";
                      postStr += String(Irms);
                      postStr +="&field3=";
                      postStr += String(power);
                      postStr +="&field4=";
                      postStr += String(bill_amount);
                      postStr +="&field5=";
                      postStr += String(Wh);
                      postStr +="&field6=";
                      postStr +=
                      String(amount);




                      postStr += "\r\n\r\n";


                      client.print("POST     /update     HTTP/1.1\n");
                      client.print("Host:
                                                  api.thingspeak.com\n
                      ");client.print("Connection: close\n");
                      client.print("X-THINGSPEAKAPIKEY:                          "+apiKey+"\n");
                      client.print("Content-       Type:       application/x-www-form-urlencoded\n");
                      client.print("Content-Length: ");client.print(postStr.length());
                      client.print("\n\n
                      ");
                    }  client.print(postS
          client.stop(  tr);
          );            Serial.println("%. Send to Thingspeak.");
           if(amount<=10)
```

```
{
  digitalWrite(connection,LOW);
 digitalWrite(buzzer,HIG
 H); delay(1000);
 digitalWrite(buzzer,LO
 W);
}
 else
{
 digitalWrite(connection,HIGH);
}

while(amount<=10)
 {

   A = ThingSpeak.readLongField(counterChannelNumber, FieldNumber1,myCounterReadAPIKey);
   if(A>19)
   {
    break;
   }

 }
      Serial.println("Waiting...");

// thingspeak needs minimum 1 sec delay between updatesdelay(15000);

}
```

CHAPTER 7
RESULT,
OBSERVATION PROCESS

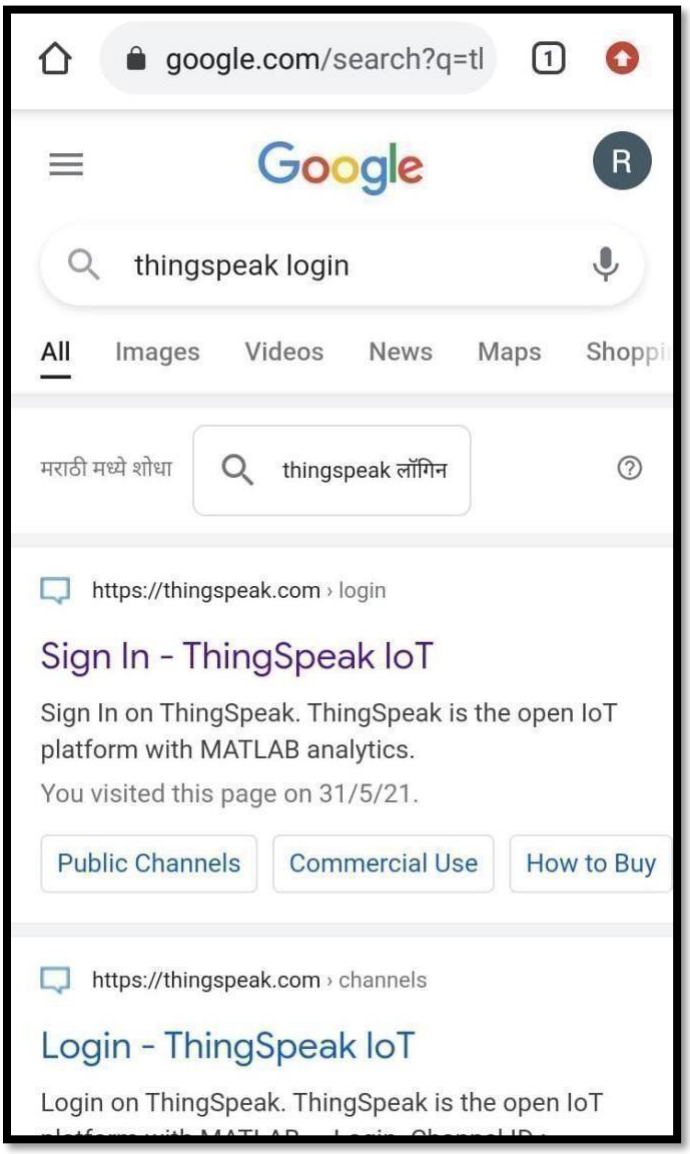
RESULT:

When power supply is given to the controller and controller take the readings from meter and send SMS to the user after every month. The accuracy of Smart Energy Meter is checked by comparing the readings that are displayed on the LCD and that are received by SMS. Smart Energy Meter is also checked by connecting and disconnecting the customer's connection. We connected different loads 100W, 200W, and 1000W and checked its performance

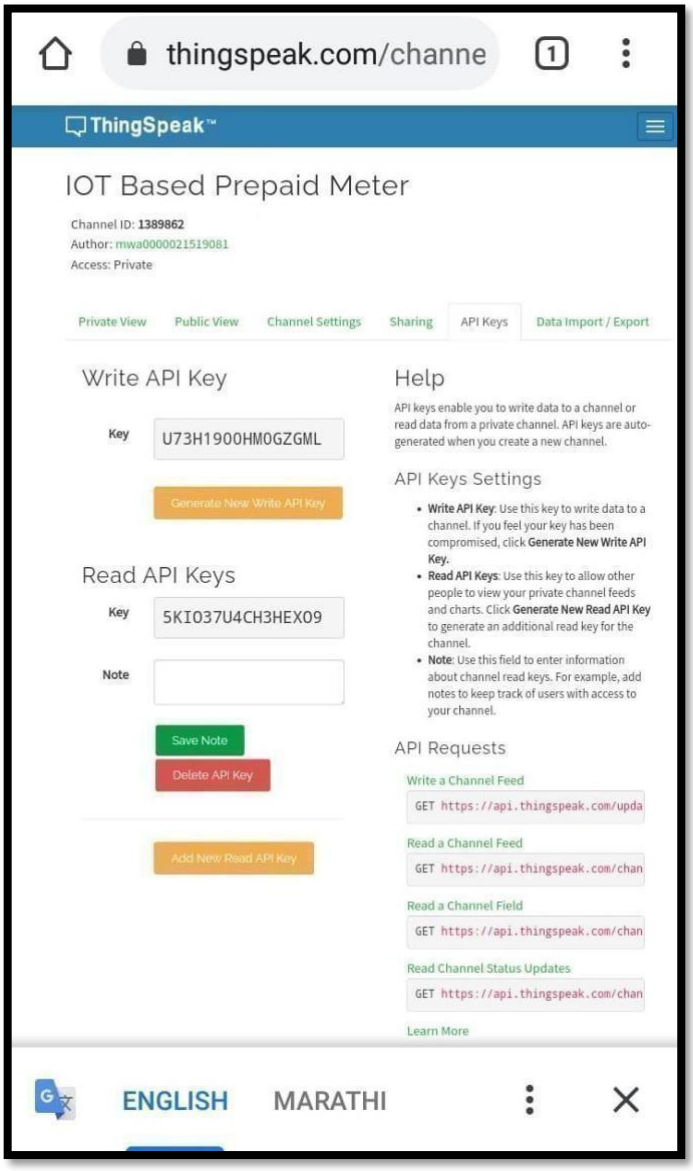


PROJECT IMPLEMENTATION

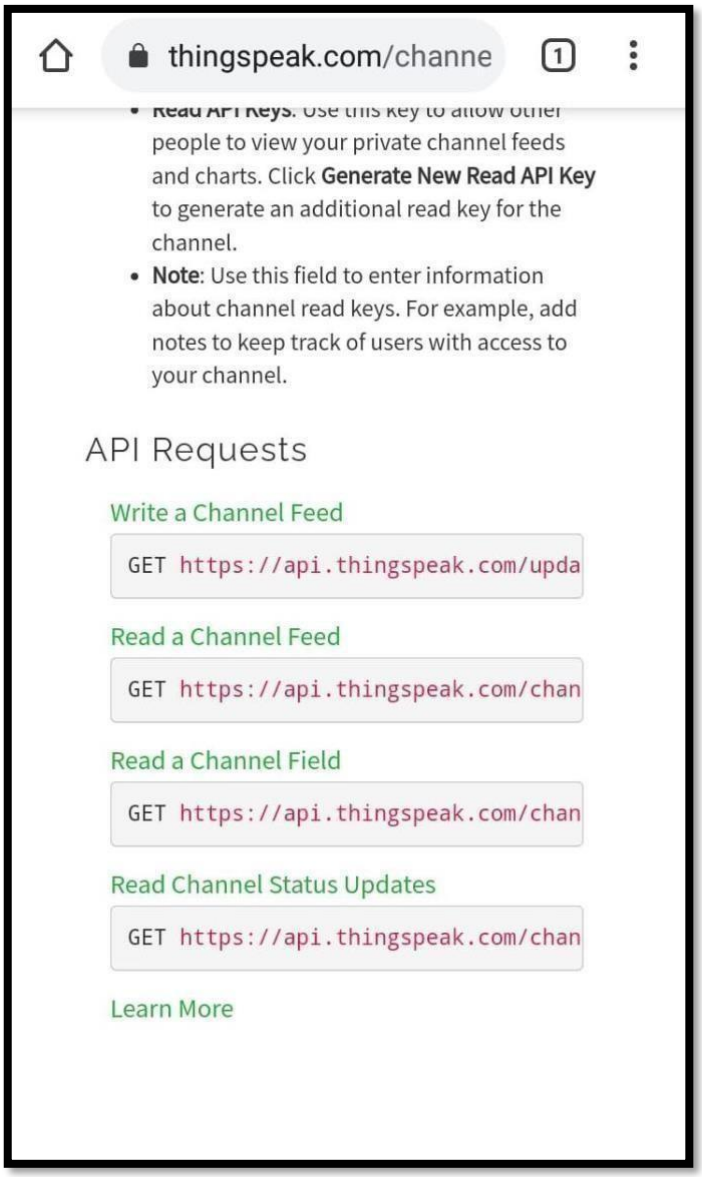
OBSERVATION PROCESS:



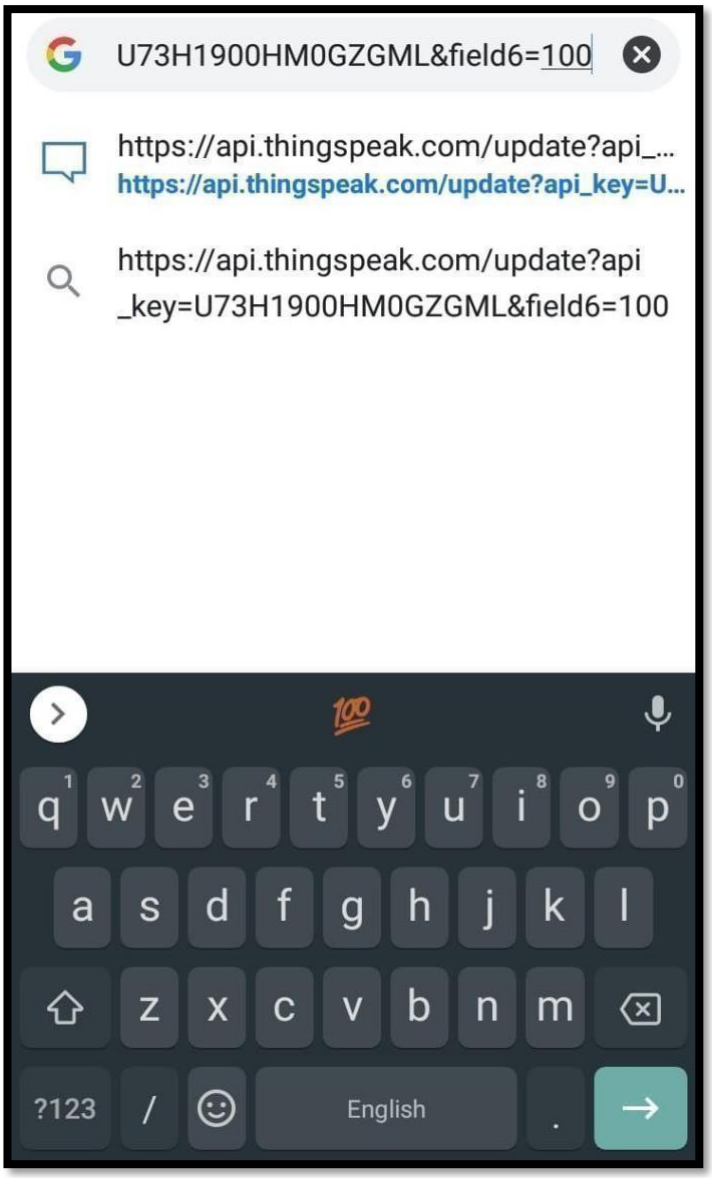
STEP 1



STEP 2

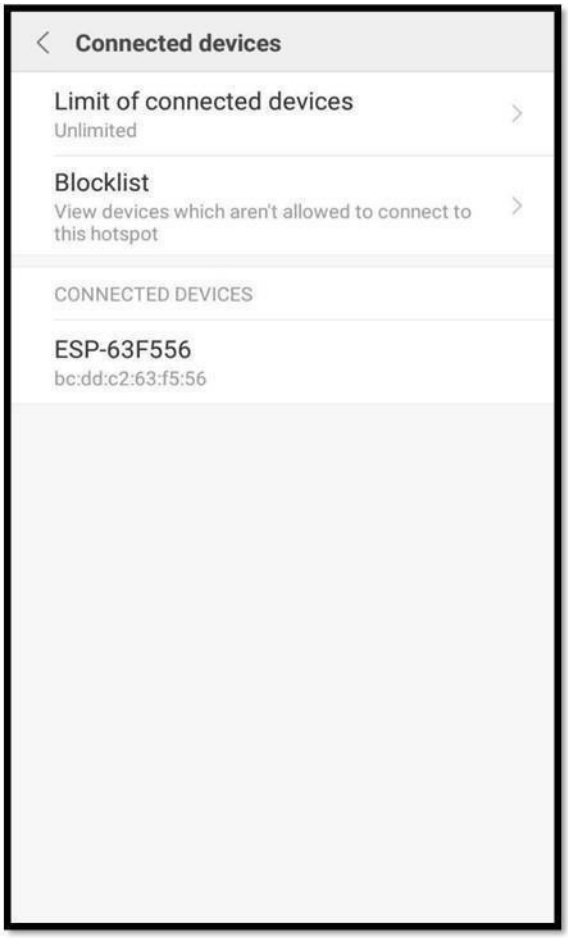

thingspeak.com/channe
Read API Keys: Use this key to allow other people to view your private channel feeds and charts. Click Generate New Read API Key to generate an additional read key for the channel.
Note: Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.
API Requests
Write a Channel Feed
GET https://api.thingspeak.com/upda
Read a Channel Feed
GET https://api.thingspeak.com/chan
Read a Channel Field
GET https://api.thingspeak.com/chan
Read Channel Status Updates
GET https://api.thingspeak.com/chan
Learn More

STEP 3


U73H1900HM0GZGML&field6=100
https://api.thingspeak.com/update?api_...
https://api.thingspeak.com/update?api_key=U...
https://api.thingspeak.com/update?api_key=U73H1900HM0GZGML&field6=100
English

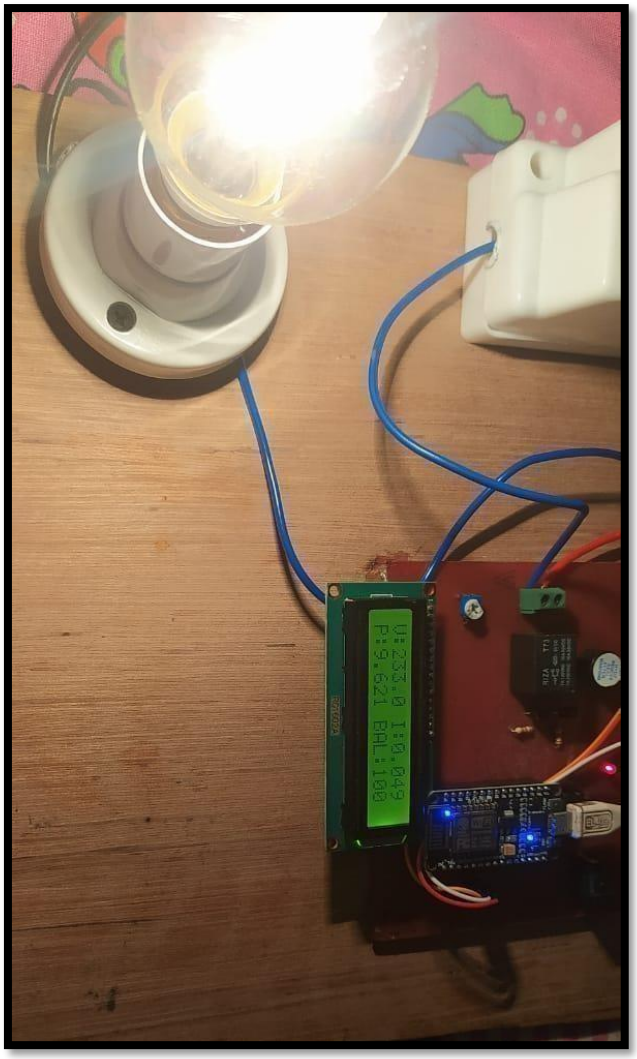
STEP 4


Portable hotspot
Portable Wi-Fi hotspot
Portable hotspot Idel active
Set up Wi-Fi hotspot
Idel None portable Wi-Fi hotspot
One-time data limit
Data limit will be applied next time your hotspot is on
MANAGE DEVICES
Connected devices
View connected devices
1 device

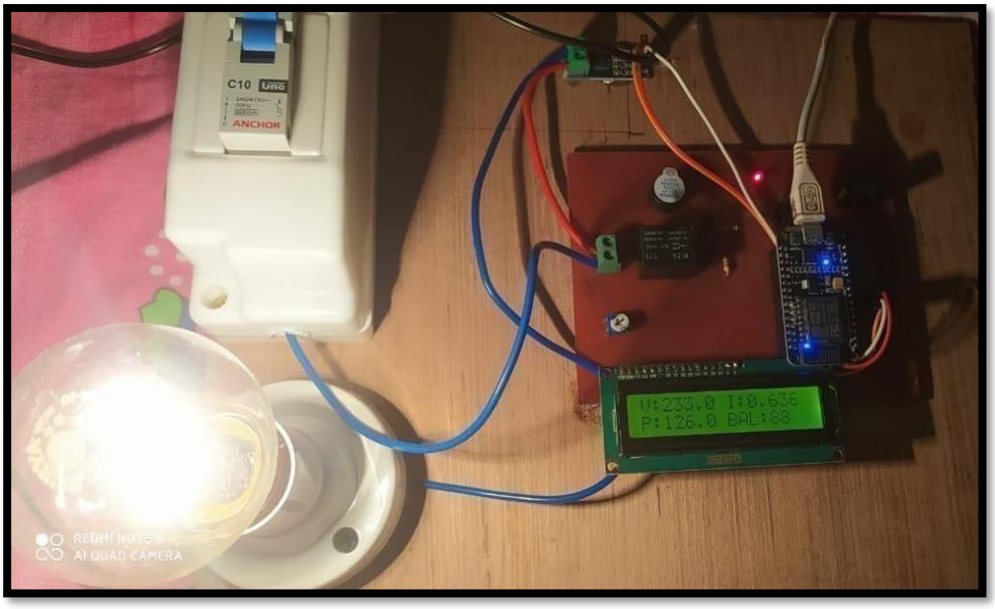

Connected devices
Limit of connected devices
Unlimited
Blocklist
View devices which aren't allowed to connect to this hotspot
CONNECTED DEVICES
ESP-63F556
bc:dd:c2:63:f5:56

STEP 5 **STEP 6**

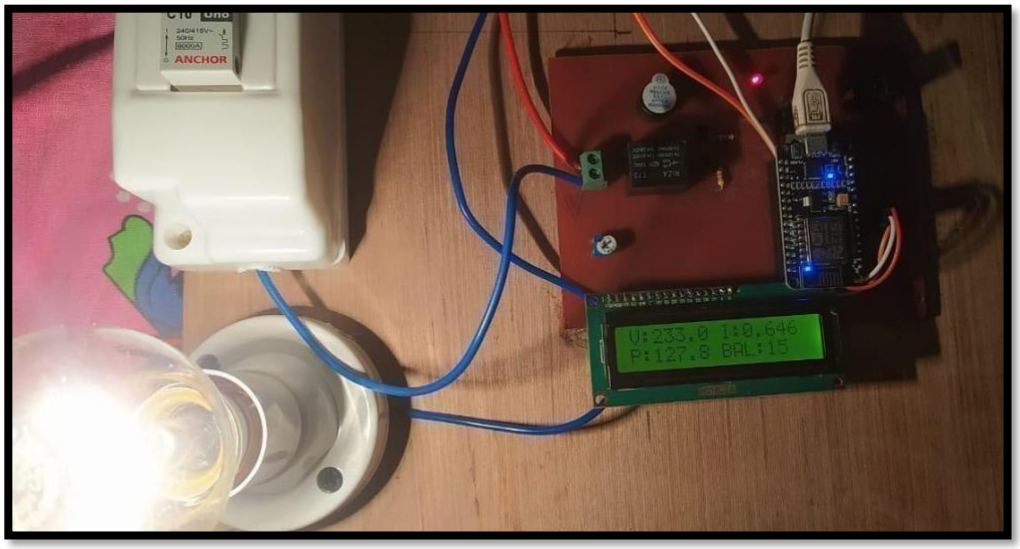
STEP 7



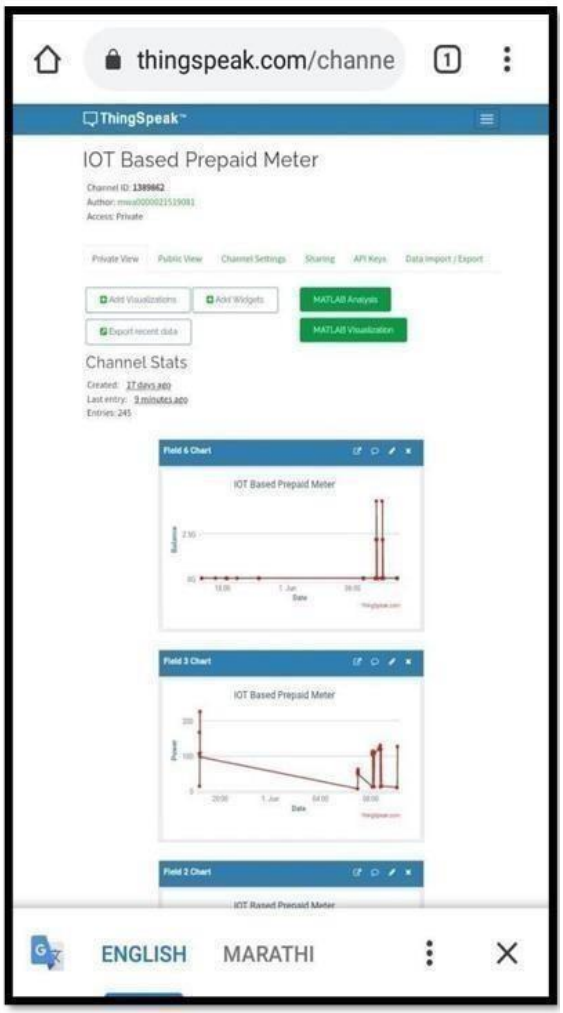
STEP 8



STEP 9



STEP 10



STEP 11 **STEP 12**

CHAPTER 8

FEATURES, ADVANTAGES, LIMITATIONS, APPLICATIONS

FEATURES:

- Over voltage protection
- Over current protection
- Short circuit protection
- Power theft protection
- Low balance alert

ADVANTAGES:

- It is highly accurate as the whole idea of reading the units and then billing automatically.
- Consumer cannot escape from paying the electricity bill and the State Electricity Board gets free form debts.
- On the consumer front, the tedious task of paying the bill and waiting anxiously for the bill is eliminated.
- Wastage of energy is diminished as now only the required energy will be consumed as allotted.
- The power grid can monitor the overall energy consumption and prevents from any damage to the circuit.
- Time saving device.
- Man power avoided.

LIMITATION:

- You must maintain your account balance at or above the disconnection balance,otherwise, your service may get disconnected.
- Billing System Fails if no IoT Network
- Coverage Requires Fixed IoT Number.
- Charges may be applicable for network use.

APPLICATION:

- ➢ Portable energy and power meters
- ➢ Grid monitoring
- ➢ Smart energy meter
- ➢ Smart plug
- ➢ Its applications include shopping malls
- ➢ Residential townships,
- ➢ Commercial buildings,
- ➢ Employe quarters etc.

CHAPTER 9
CONCLUSION, FUTURE SCOPE

CONCLUSION:

Smart energy meter with reading indication has been developed by using IoT. Which is more useful to consumer for billing and maintaining less bill payment and it decreases the human needs for paying and other issues related to billing. We can extend it for industrial purposes also by interfacing three phase meters but the circuit have to modify for getting proper voltage to the controller.

FUTURE SCOPE:

I thought there are few possibilities which can also be done on this project in future as I have provided flexibility in the project especially in controller section. The future research should include the proper methodology for measuring the power factor of the load.

Recommendations for future are as follows:

- Instead of IoT networks, some other means of communication should be used.
- In case of IoT, there must be security of IoT so that it can't be hacked.
- Power factor must be measured by different techniques.
- Linking of the data received by IoT to computer and developing a program which incorporates the tariff related to specific consumer and calculating the bill directly on the computer. In this way the computer will calculate the bill directly using the data received through IoT network.
- Smart energy Meter can be modified for the detection of illegal use of electricity.