double sensorValue1 = 0;

double sensorValue2 = 0;

int crosscount = 0;

int climb\_flag = 0;

int val[100];

float tem\_degree, temp\_forn;

int max\_v = 0;

double VmaxD = 0;

double VeffD = 0;

double Veff = 0;

const int analogchannel = 0; // Connect current sensor with A0 of Arduino

int sensitivity = 70; // use 100 for 20A Module and 66 for 30A Module

float adcvalue = 0;

int offsetvoltage = 2500;

double Voltage = 0; // voltage measuring

double ecurrent = 0; // Current measuring

float power = 0;

int in\_Voltage = 230;

unsigned long previousMillis = 0; // Store the previous time in milliseconds

unsigned long interval = 1000; // The time interval for updating energy consumption (in milliseconds)

float totalEnergy; // Total energy consumption in kWh

float unitPrice = 1.90;

int address = 0; // Starting address in EEPROM

bool powerOn = true; // Flag to track power state

void setup()

{

Serial.begin(9600);

}

void loop()

{

voltage\_read();

current\_measure();

if (ecurrent < 0.0)

{

ecurrent = 0.0;

Serial.print(ecurrent);

Serial.print(",");

Serial.print(Veff);

Serial.print(",");

Serial.println();

}

else

{

Serial.print(ecurrent);

Serial.print(",");

Serial.print(Veff);

Serial.print(",");

Serial.println();

}

delay(500);

}

void current\_measure()

{

unsigned int temp = 0;

float maxpoint = 0;

int i = 0;

for (i = 0; i < 500; i++)

{

if (temp = analogRead(analogchannel), temp > maxpoint)

{

maxpoint = temp;

}

}

adcvalue = maxpoint;

Voltage = (adcvalue / 1024.0) \* 5000; // Gets you mV

ecurrent = ((Voltage - offsetvoltage) / sensitivity);

ecurrent = (ecurrent) / (sqrt(2));

ecurrent = ecurrent + 0.050;

}

void voltage\_read()

{

for (int i = 0; i < 100; i++)

{

sensorValue1 = analogRead(A1);

if (analogRead(A1) > 511)

{

val[i] = sensorValue1;

}

else

{

val[i] = 0;

}

delay(1);

}

max\_v = 0;

for (int i = 0; i < 100; i++)

{

if (val[i] > max\_v)

{

max\_v = val[i];

}

val[i] = 0;

}

if (max\_v != 0)

{

VmaxD = max\_v;

VeffD = VmaxD / sqrt(2);

Veff = (((VeffD - 420.76) / -90.24) \* -210.2) + 210.2;

}

else

{

Veff = 0;

}

if (Veff < 100 && Veff > 50)

{

//Serial.println("POWER OFF");

Veff = 0;

delay(1000);

powerOn = false; // Set power state flag to off

}

else

{

powerOn = true; // Set power state flag to on

}

Veff = Veff - 40;

if(Veff <= -0)

Veff=0;

VmaxD = 0;

}