#include "EmonLib.h" // Include Emon Library

EnergyMonitor emon1,emon2,emon3; // Create an instance

#include <SPI.h>

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SSD1306.h>

Adafruit\_SSD1306 display(-1);

void setup()

{

Serial.begin(9600);

// initialize with the I2C addr 0x3C

display.begin(SSD1306\_SWITCHCAPVCC, 0x3C);

// Clear the buffer.

display.clearDisplay();

// Display Inverted Text

display.setTextColor(WHITE);

display.setCursor(25,0);

display.setTextSize(2.5);

display.println(" VIEW");

Serial.println("VIEW");

display.setCursor(0,15);

display.setTextSize(1);

display.println(" (NBA accredited)");

Serial.println (" (NMA accredited)");

display.setCursor(75,25);

display.setTextSize(1.5);

display.println("-SRP12");

Serial.println("-SRP12");

display.display();

delay(4000);

display.clearDisplay();

display.setTextSize(1.5);

display.setCursor(0,0);

display.println("HACKTHON BATCH:20NM5A0209,");

Serial.println("HACKTHON BATCH:20NM5A0209,");

display.setCursor(0,10);

display.println("20NM5A0208,20NM5A0220,");

Serial.println("20NM5A0208,20NM5A0220,");

display.setCursor(0,20);

display.println("20NM5A0236,19NM1A0208.");

Serial.println("20NM5A0236,19NM1A0208.");

delay(4000);

display.display();

display.clearDisplay();

Serial.flush();

emon1.voltage(0, 135,1.5); // Voltage: input pin, calibration, phase\_shift

emon1.current(2, 0.2); // Current: input pin, calibration.

emon2.voltage(0, 135,1.5); // Voltage: input pin, calibration, phase\_shift

emon2.current(3, 0.2); // Current: input pin, calibration.

emon3.voltage(0, 132,1.5); // Voltage: input pin, calibration, phase\_shift

emon3.current(4, 0.2); // Current: input pin, calibration.

}

void loop()

{

emon1.calcVI(20,2000); // Calculate all. No.of wavelengths, time-out

emon1.serialprint(); // Print out all variables

emon2.calcVI(20,2000); // Calculate all. No.of wavelengths, time-out

emon2.serialprint(); // Print out all variables

emon3.calcVI(20,2000); // Calculate all. No.of wavelengths, time-out

emon3.serialprint(); // Print out all variables

unsigned int long timemillis=millis(); //keeping the track of the time since the device is switched ON

unsigned int long time=timemillis/1000;

float realPower= emon1.realPower; //extract Real Power into variable

float apparentPower = emon1.apparentPower; //extract Apparent Power into variable

float powerFActor = emon1.powerFactor; //extract Power Factor into Variable

int supplyVoltage = emon1.Vrms; //extract Vrms into Variable

float Irms = emon1.Irms; //extract Irms into Variable

float realPower2= emon2.realPower; //extract Real Power into variable

float apparentPower2 = emon2.apparentPower; //extract Apparent Power into variable

float powerFActor2 = emon2.powerFactor; //extract Power Factor into Variable

int supplyVoltage2 = emon2.Vrms; //extract Vrms into Variable

float Irms2 = emon2.Irms; //extract Irms into Variable

float realPower3= emon3.realPower; //extract Real Power into variable

float apparentPower3 = emon3.apparentPower; //extract Apparent Power into variable

float powerFActor3 = emon3.powerFactor; //extract Power Factor into Variable

int supplyVoltage3 = emon3.Vrms; //extract Vrms into Variable

float Irms3 = emon3.Irms; //extract Irms into Variable

float units=abs(realPower\*time)/abs(1000\*3600);

float units2=abs(realPower2\*time)/abs(1000\*3600);

float units3=abs(realPower3\*time)/abs(1000\*3600);

// Display Inverted Text FOR METER-1,2,3

display.setTextColor(WHITE);

display.setTextSize(1.5);

display.setCursor(0,0);

display.println("SupplyVolt(V)=");

Serial.println("SupplyVolt(V)=");

display.setCursor(100,0);

display.println("supplyVoltage");

Serial.println("supplyVoltage");

display.setCursor(0,10);

display.println("E(W)");

Serial.println("E(W)");

display.setCursor(0,20);

display.println("I(A)");

Serial.println("I(A)");

display.setCursor(30,10);

display.println(units);

Serial.println(units);

display.setCursor(30,20);

display.println(Irms);

Serial.println(Irms);

display.setCursor(60,10);

display.println(units2);

Serial.println(units2);

display.setCursor(60,20);

display.println(Irms2);

Serial.println(Irms2);

display.setCursor(90,10);

display.println(units3);

Serial.println(units3);

display.setCursor(90,20);

display.println(Irms3);

Serial.println(Irms3);

display.display();

//delay(2000);

display.clearDisplay();

Serial.flush();

}