#include<stdlib.h>

#include <LiquidCrystal.h>

//variable IO assignments

const int DCsense1 = A0; //DC current ADC, 5A, buck-boost

const int DCsense2 = A1; //DC current ADC, 30A, battery

const int ACsense = A3; //AC current ADC, 5A, inverter transformer

const int Vsense1 = A4; //battery output, range 0-13

const int Vsense2 = A5; //buck boost output, range 0-45

//constants relevant to the ACS712 specs

const int scalefactor\_5A = 185; // 185mV/A

const int scalefactor\_30A = 66; // 66mV/A

const int offset = 2500; //2500mV offset

//variables relevant to measuring DC current

double DCraw = 0;

double milivolts = 0;

double current1 = 0; //DC out

double current2 = 0; //battery out

//variables relevant to measuring AC current

double vpp = 0; //peak to peak voltage

double vrms = 0; //rms voltage

double irms = 0; //rms current

//variables relevant to measuring voltage

double v1 = 0;

double v2 = 0;

double vbatt = 0;

double vconverter = 0;

//lcd pin assingments

LiquidCrystal lcd(7,8,9,10,11,12);

static char out[10]; //storage array for double->chars

void setup() {

Serial.begin(9600);

lcd.begin(16,2);

}

void loop() {

//measure voltages

v1 = analogRead(Vsense1);

v2 = analogRead(Vsense2);

//scale v1, v2, v3 depending on ADC source

vbatt = (v1/1024.0)\*13;

vconverter = (v2/1024.0)\*45;

/\*DC current algorithm: measure ADC from ACS712 chip, covert ADC to millivolts, subtract the offset (2.5v), divide by scale factor\*/

DCraw = analogRead(DCsense1); //ADC 0-5V analog to 0-1024 digital

milivolts = (DCraw/1024.0)\*5000; //convert ADC to milivolts

current1 = ((milivolts - offset)/scalefactor\_5A);

DCraw = analogRead(DCsense2); //ADC 0-5V analog to 0-1024 digital

milivolts = (DCraw/1024.0)\*5000; //convert ADC to milivolts

current2 = ((milivolts - offset)/scalefactor\_30A);

/\*AC current algorithm: measure Vpp, divide to 2 to get peak voltage, multiply by 0.77 to get rms voltage, multiply by scale factor to convert to rms current\*/

vpp = getVPP();

vrms = (vpp/2.0)\*0.707;

irms = (vrms\*1000)/scalefactor\_5A;

//print variables to LCD

//print battery voltage

dtostrf(vbatt, 3, 1, out); //convert double variable to a character array of 3 characters, with 1 character after the decimal

lcd.setCursor(0,0);

lcd.print("Vb");

lcd.setCursor(2,0);

lcd.print(out);

//print output of buck-boost

dtostrf(vconverter, 3, 1, out);

lcd.setCursor(6,0);

lcd.print("Vo");

lcd.setCursor(8,0);

lcd.print(out);

//print battery current

dtostrf(current1, 2, 0, out);

lcd.setCursor(0,1);

lcd.print("Ib");

lcd.setCursor(2,1);

lcd.print(out);

//print dc current

dtostrf(current2, 1, 0, out);

lcd.setCursor(9,1);

lcd.print("Idc");

lcd.setCursor(12,1);

lcd.print(out);

//print ac current

dtostrf(irms, 1, 0, out);

lcd.setCursor(4,1);

lcd.print("Iac");

lcd.setCursor(7,1);

lcd.print(out);

}

float getVPP()

{

float result;

int value;

int maxvalue = 0;

int minvalue = 1024;

uint32\_t start = millis();

while ((millis()-start)<1000) //sample for 1 second

{

value = analogRead(ACsense);

if(value > maxvalue)

{

maxvalue = value;

}

if(value < minvalue)

{

minvalue = value;

}

}

result = ((maxvalue - minvalue)\*5.0)/1024.0;

return result;

}