Universidade Federal de Uberlândia
FEELT – Faculdade de Engenharia Elétrica

Projeto Interdisplinar I

Controle e Monitoramento de Energia

Professor: Luiz Cláudio Theodoro

Aluno: Pedro Henrique Dias Faquim

Matrícula: 11121EEL036

UBERLÂNDIA
2013

Sumário

Introdução

No início do semestre, incertezas sobre do que se tratava a disciplina e que projeto eu iria desenvolver rodeavam a minha cabeça. Não sabia exatamente o que e como fazer dessa experiência algo produtivo não só para este semestre letivo, mas para a minha vida. Depois de reuniões e apresentações de temas que poderiam ser trabalhados e desenvolvidos, encontrei uma maneira de desenvolver algo que tivesse relação com o interessante trabalho proposto por meu grupo em ESOF (Engenharia de Software). Após pesquisas relacionadas a projetos já desenvolvidos ou em desenvolvimento na área de controle e monitoramento de energia, encontrei uma maneira de transformar a ideia de se monitorar o consumo de energia em grandes fábricas e indústrias em algo semelhante para se aplicar em ambiente menores, como por exemplo, em residências. A partir de então, passei a focar meus esforços na construção de um medidor de consumo de energia.

O medidor é um aparelho que tem como função captar dados relacionados à corrente e à tensão para realizar cálculos e fornecer ao consumidor do produto informações a respeito do consumo em Watts, a taxa de consumo em KW/h e o gasto acumulado, em reais, levando em consideração a taxa cobrada pela concessionária de energia da região em que o consumidor reside para a taxa de consumo por hora.

Entre as vantagens deste produto estão a possibilidade oferecida ao consumidor de instalação do produto com a ajuda de um manual de instruções sem a necessidade de contar com a ajuda de um técnico ou especialista em instalações elétricas por se tratar de um produto que contém em sua estrutura um sensor de corrente não-invasivo, que oferece maior segurança e praticidade ao comprador, e a possibilidade de se acompanhar o levantamento realizado sem a necessidade de se utilizar um software para processamento e divulgação de informações a serem disponibilizadas pelo medidor, já que essas informações são prontamente disponibilizadas por meio do LCD a partir do momento que que o medidor começa a funcionar. Além disso, o medidor pode ser alimentado por meio do Arduino com o uso de um cabo USB que o conecta a um computador ou por meio de uma fonte ligada em tomada e conectada a uma das entradas presente no Arduino, o que garante ao consumidor a possibilidade de escolha de como colocar o produto para funcionar de acordo com suas possiblidades.

Pensando no projeto a ser desenvolvido não só como mais um trabalho a ser entregue, mas sim como algo maior, me interessei pela possibilidade de desenvolver este medidor por saber que se trata de um aparelho que, pelo fato de tornar possível o monitoramento do consumo, gera interesse do mercado consumidor. Além disso, eu teria uma grande oportunidade de aprender um pouco mais sobre circuitos elétricos e eletrônica, além de poder trabalhar com o Arduino, um produto incrível que fornece diferentes possibilidades graças à sua capacidade de permitir criações que tem como princípio a interação entre objetos eletrônicos e de poder contar, após finalizadas as tarefas relacionadas ao desenvolvimento deste trabalho, com o medidor para realizar o monitoramento do consumo de energia do apartamento em que resido com minha família.

Componentes

Para a construção do medidor, foi necessário dividir o processo de acordo com as funções a serem cumpridas pelo medidor (captação de corrente e tensão para a realização de cálculos a fim de fornecer informações a respeito do consumo monitorado).

Componentes utilizados:

- Arduino
- Protoboard
- Fios para ligação (jumpers)
- LCD
- Potenciômetro para ajuste de contraste do LCD

Para captar dados relacionados à tensão:

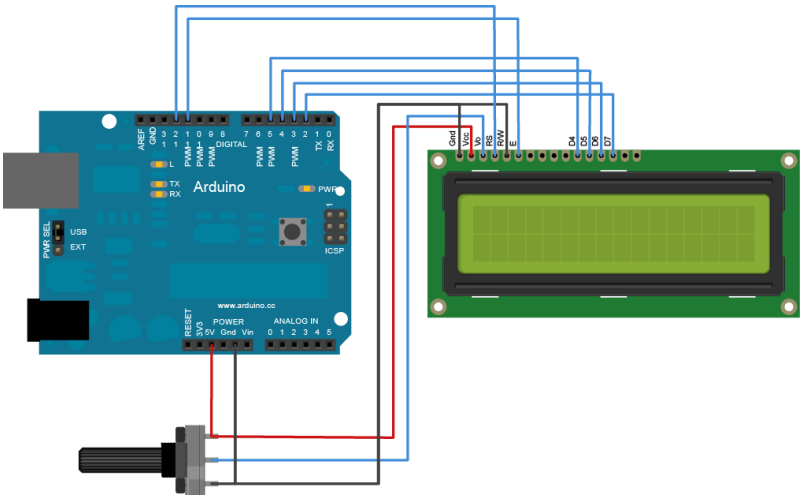
- Fonte 9V CA-CA
- Resistor de 100kOhm para diminuir divisor de tensão
- Resistor de 10kOhm para diminuir divisor de tensão
- 2 Resistores de 10kOhm para polarização de divisor de tensão
- Capacitor de 10uF

Para captar dados relacionados à corrente:

- Sensor de corrente SCT-013-000
- Resistor de carga de 100Ohm para 3.3V (Power - Arduino) como tensão fornecida
- 2 Resistores de 10kOhm
- Capacitor de 10uF

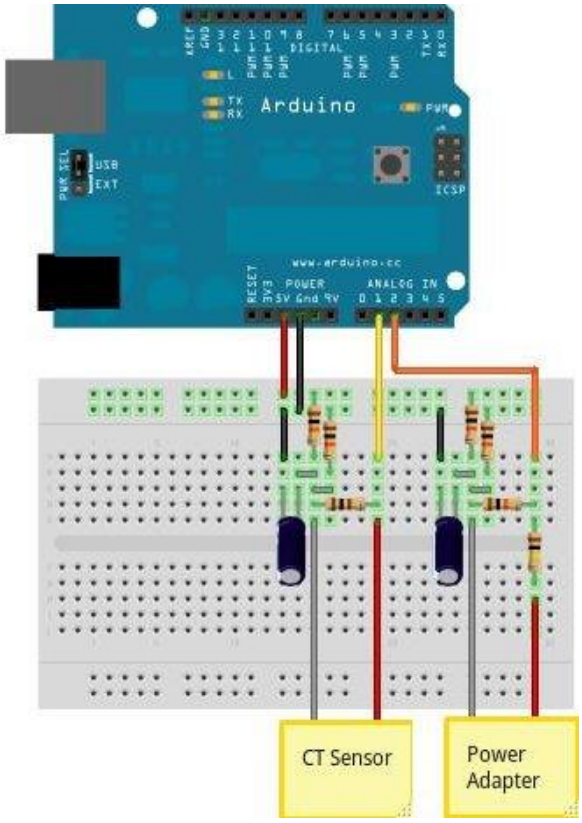
Ligação: Arduino – LCD – Potenciômetro

A ligação entre estes componentes acontece com o auxílio de um protoboard.



Ligação: Arduino – Sensor de Corrente SCT-013-000 – Fonte

Para ligarmos os componentes e formarmos, assim, o circuito que será a base do medidor, precisamos usar o protoboard.



Construção do Medidor

Todas as informações para o desenvolvimento do projeto foram retiradas dos endereços abaixo:

http://openenergymonitor.org/emon/buildingblocks/how-to-build-an-arduino-energy-monitor

http://www.youtube.com/watch?v=XkzRabGOAMo

http://www.projetoarduino.com.br/index.php

Para o desenvolvimento, a base foi a configuração das ligações demonstrada nas imagens anteriormente colocadas.

Arduino

Arduino é uma plataforma de prototipação open-source baseado em uma simples placa com entradas e saídas tanto digitais como analógicas. Possui um próprio ambiente de desenvolvimento que implementa a Linguagem C. O Arduino pode ser usado para desenvolver objetos interativos autônomos ou pode ser conectado a um software em seu computador (ex. Flash, Processing, MaxMSP). O Ambiente de desenvolvimento (IDE) open-source pode ser obtido gratuitamente (atualmente disponível para Mac OS X, Windows e Linux).

O Arduino utilizado foi o Arduino Uno R3. É uma placa com micro controlador Atmega328. Possui 14 entradas/saídas digitais (das quais 6 podem ser usadas como saídas PWM), 6 entradas analógicas, um cristal oscilador de 16MHz, conexão USB, uma entrada para fonte, soquetes para ICSP, e um botão de reset. A placa contém todo o necessário para usar o micro controlador. Simplesmente conecte-a a um computador com o cabo USB - AB ou ligue a placa com uma fonte AC-DC (ou bateria). O Uno seleciona automaticamente a fonte de alimentação (USB ou fonte externa).

A placa Uno se diferencia das outras por não utilizar o chip da FTDI USB-to-Serial. Ao invés deste chip, um Atmega8U2 já programado faz a função de converter os dados da USB para Serial.

Características:

Tamanho: 5,3cm x 6,8cm x 1,0cm

Micro controlador: ATmega328

Tensão de operação: 5V

Tensão de entrada (recomendada): 7-12V

Tensão de entrada (limites): 6-20V

Pinos de entrada/saída (I/O) digitais: 14 (dos quais 6 podem ser saídas PWM)

Pinos de entrada analógicas: 6

Corrente DC por pino I/O: 40mA

Corrente DC para pino de 3,3V: 50mA

Memória Flash: 32KB (dos quais, 0,5KB são usados pelo *bootloader*)

SRAM: 2KB

EEPROM: 1KB

Velocidade de Clock: 16MHz

Temperatura de operação: de 10º a 60º

Arduino Software:

A interface de programação open-source do Arduino facilita a escrever e gravar o código para a placa I/O. Ele roda em Windows, Mac OS X e Linux.

Datasheet: http://www.atmel.com/Images/doc8161.pdf

Sensor de Corrente SCT-013-000

Transformador de corrente não-invasivo. Normalmente utilizado para construção de monitores de energia pessoais. Adequado para o monitoramento de medição de corrente, proteção de motor CA, equipamentos de iluminação, compressor de ar e outros.

Características:

Modelo: SCT-013-000
Variação da temperatura de operação entre -25°C e 70°C

Especificações:

Elétricas:

Tensão de Alimentação:

Corrente Nominal de Entrada: 60A/100A

Força Dielétrica: 6000V AC/minuto

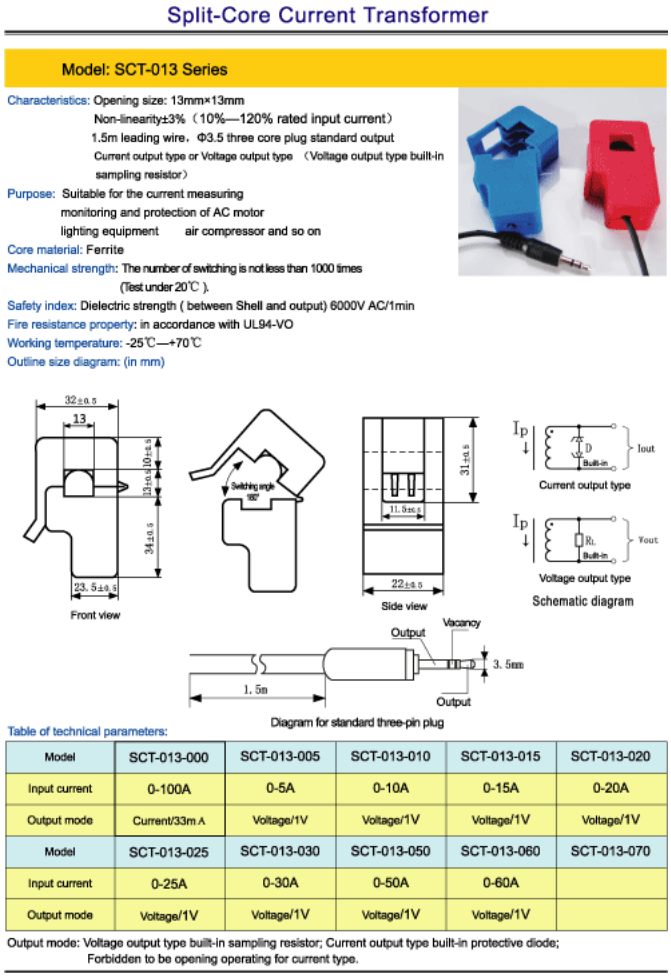
Mecânicas:

Material Externo: Plástico

Núcleo: Ferrite

Abertura: 13x13mm

Ligação de Saída: 3,5mm
Comprimento do cabo: 1m/39,4"
Tamanho Total: 5,7x3,2x2,1cm/2,2x1,3x0,8 (Comp.xLarg.xAlt.)
Peso: 76g



Split-Core Current Transformer

Model: SCT-013 Series

Characteristics: Opening size: 13mm×13mm
Non-linearity±3%（10%—120% rated input current）
1.5m leading wire，Φ3.5 three core plug standard output
Current output type or Voltage output type （Voltage output type built-in sampling resistor）
Purpose: Suitable for the current measuring monitoring and protection of AC motor lighting equipment air compressor and so on
Core material: Ferrite
Mechanical strength: The number of switching is not less than 1000 times (Test under 20℃).
Safety index: Dielectric strength (between Shell and output) 6000V AC/1min
Fire resistance property: in accordance with UL94-VO
Working temperature: -25℃—+70℃
Outline size diagram: (in mm)

Table of technical parameters:

Model	SCT-013-000	SCT-013-005	SCT-013-010	SCT-013-015	SCT-013-020
Input current	0-100A	0-5A	0-10A	0-15A	0-20A
Output mode	Current/33m A	Voltage/1V	Voltage/1V	Voltage/1V	Voltage/1V
Model	SCT-013-025	SCT-013-030	SCT-013-050	SCT-013-060	SCT-013-070
Input current	0-25A	0-30A	0-50A	0-60A	
Output mode	Voltage/1V	Voltage/1V	Voltage/1V	Voltage/1V	

Output mode: Voltage output type built-in sampling resistor; Current output type built-in protective diode;
Forbidden to be opening operating for current type.

Datasheet: http://vis.openenergymonitor.org/ShopPhotos/SCT013-000_datasheet.pdf

Fonte

A Fonte CA-CA de 9V, neste caso, é utilizada para fazer a medição da tensão para que cálculos relacionados a tensão real, tensão aparente e fator de tensão possam ser realizados. A Fonte permite que a medição seja realizada de um modo seguro, já que o

transformador presente em sua estrutura trabalha como um “isolante” (separa a alta da baixa tensão), o que evita o trabalho com altas tensões que são extremamente perigosas.

LCD

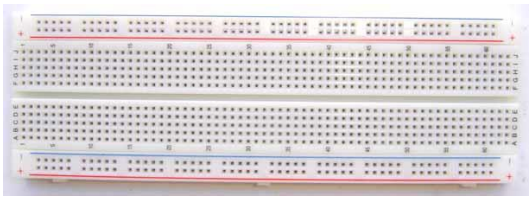
Este é um display básico de 16 caracteres por 2 linhas. Texto preto sobre fundo verde. Utiliza o extremamente comum chipset HD44780 de interface paralela. O código da interface está disponível gratuitamente. Você vai precisar de aproximadamente 11 pinos de entrada/saída (I/O) para fazer interface com esta tela LCD. Inclui LED backlight.

Características:

Dimensões: 3.15" x 1.425" x 0.300"

Datasheet: http://www.robocore.net/upload/lojavirtual/GDM1602K-Extended.pdf

Protoboard



Uma placa de ensaio ou matriz de contato, (ou protoboard, ou breadboard em inglês) é uma placa com furos e conexões condutoras para montagem de circuitos elétricos experimentais. A grande vantagem da placa de ensaio na montagem de circuitos eletrônicos é a facilidade de inserção de componentes, uma vez que não necessita soldagem. As placas variam de 800 furos até 6000 furos, tendo conexões verticais e horizontais.

Na superfície de uma matriz de contato há uma base de plástico em que existem centenas de orifícios onde são encaixados os componentes. Em sua parte inferior são instalados contatos metálicos que interligam eletricamente os componentes inseridos na placa. Geralmente suportam correntes entre 1 A e 3 A. Os contatos metálicos estão em diferentes sentidos na matriz.

Conclusão

A disciplina, por meio das instruções dadas pelo professor Luiz Cláudio Teodoro, instruções essas que nos indicavam a necessidade de construirmos ou desenvolvermos um projeto, modificou a minha maneira de pensar a respeito de trabalhos acadêmicos relacionados à parte prática do curso. As dificuldades encontradas e as possibilidades disponíveis para a construção do medidor me fizeram levar em consideração não só minhas dificuldades ou interesses, mas dificuldades e interesses de outros, passei a pensar num todo, como se estivesse realmente trabalhando em um projeto a ser desenvolvido para ser colocado à venda, percebi que é necessário se analisar todas as possibilidades a fim de se conseguir, ao final do trabalho, um produto capaz de atender não só as minhas necessidades, mas também as do mercado consumidor de maneira eficaz, prática e fácil.

Concluo que as atividades desenvolvidas, além de terem me ajudado a entender um pouco mais sobre elétrica e eletrônica, e até sobre automação, me ajudaram a mudar a minha maneira de pensar e perceber o que acontece ao meu redor, o que está envolvido no desenvolvimento de um trabalho, um projeto.

Anexos

Programação Arduino

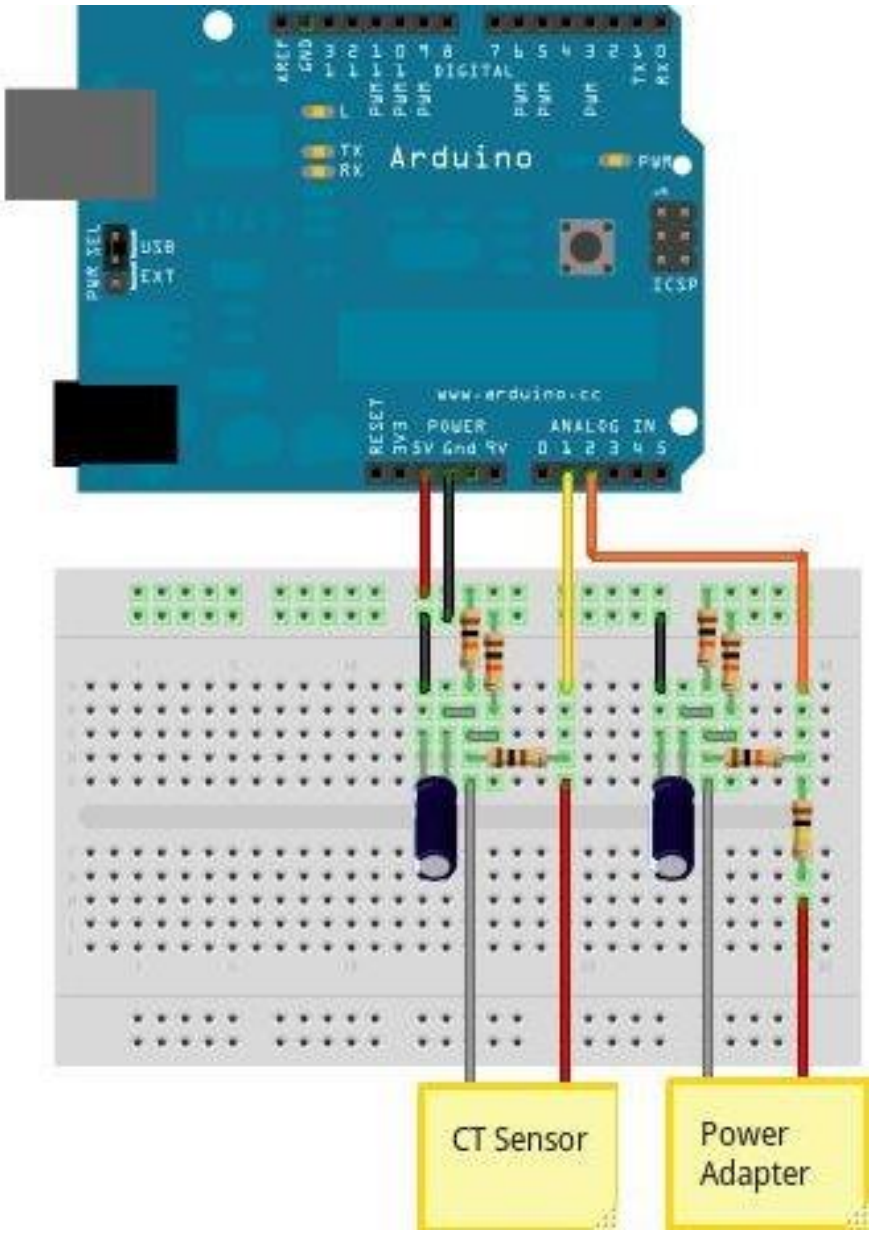
```
// EmonLibrary examples openenergymonitor.org, Licence GNU GPL V3

#include "EmonLib.h"                   // Include Emon Library
//#include "RTClib.h"
#include <avr/eeprom.h>
#include <Adafruit_GFX.h>
#include <Adafruit_PCD8544.h>

#define eeprom_read_to(dst_p, eeprom_field, dst_size) eeprom_read_block(dst_p,
(void *)offsetof(__eeprom_data, eeprom_field), MIN(dst_size,
sizeof((__eeprom_data*)0)->eeprom_field))
#define eeprom_read(dst, eeprom_field) eeprom_read_to(&dst, eeprom_field,
sizeof(dst))
#define eeprom_write_from(src_p, eeprom_field, src_size)
eeprom_write_block(src_p, (void *)offsetof(__eeprom_data, eeprom_field),
MIN(src_size, sizeof((__eeprom_data*)0)->eeprom_field))
#define eeprom_write(src, eeprom_field) { typeof(src) x = src;
eeprom_write_from(&x, eeprom_field, sizeof(x)); }
#define MIN(x,y) ( x > y ? y : x )
/*
 * __eeprom_data is the magic name that maps all of the data we are
 * storing in our EEPROM
 */
struct __eeprom_data {
  double flash_kwhtotal;
};

// pin 7 - Serial clock out (SCLK)
// pin 6 - Serial data out (DIN)
// pin 5 - Data/Command select (D/C)
// pin 4 - LCD chip select (CS)
// pin 3 - LCD reset (RST)
//Adafruit_PCD8544 display = Adafruit_PCD8544(7, 6, 5, 4, 3);
Adafruit_PCD8544 display = Adafruit_PCD8544(3, 4, 5, 6, 7);

#define NUMFLAKES 10
#define XPOS 0
#define YPOS 1
#define DELTAY 2


#define LOGO16_GLCD_HEIGHT 16
#define LOGO16_GLCD_WIDTH  16

//RTC_DS1307 RTC;            // Create an instance
EnergyMonitor emon1;      // Create an instance

//Cria variaveis globais
double kwhTotal;
double vlreais;
unsigned long ltmillis, tmillis, timems, previousMillis;
char charBuf[30];
void setup()
{
```

```
  Serial.begin(9600);
  display.begin();
  display.setContrast(37);
  /*if (! RTC.isrunning()) {
    Serial.println("RTC is NOT running!");
  }*/
  //Theoretical CT sensor calibration
  //CT Ratio / Burden resistance = (100A / 0.05A) / 64 Ohms = 50
  emon1.current(1, 29.41); // Current: input pin, calibration for 24 Ohms
  eeprom_read(kwhTotal, flash_kwhtotal);
  previousMillis = millis();

    // text display tests
  /*display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(BLACK);
  display.setCursor(0,0);
  display.println("Hello, world!");
  display.setTextColor(WHITE, BLACK); // 'inverted' text
  display.println(3.141592);
  display.setTextSize(2);
  display.setTextColor(BLACK);
  display.print("0x"); display.println(0xDEADBEEF, HEX);
  display.display();
  delay(2000);*/

}

void loop()
{
  //Calculate amount of time since last realpower measurment.
  ltmillis = tmillis;
  tmillis = millis();
  timems = tmillis - ltmillis;
  double Irms = emon1.calcIrms(1480);  // Calculate Irms only

  //Calculate todays number of kwh consumed.
  //kwhTotal = kwhTotal + ((realPower/1000.0) * 1.0/3600.0 * (timems/1000.0));

  //Calculate todays number of kwh consumed.
  kwhTotal = kwhTotal + (((Irms*127.0)/1000.0) * 1.0/3600.0 *
(timems/1000.0));

  Serial.print("Watts: ");
  Serial.println(Irms*127.0);         // Apparent power
  Serial.print("Current: ");
  Serial.println(Irms);            // Irms
  Serial.print("kwhTotal: ");
  printFloat(kwhTotal, 10);
  Serial.println("");

  //grava na memoria a cada 1 minuto
  if ((millis() - previousMillis)>4000)
      {
        Serial.println("Gravando na EEprom");
        eeprom_write(kwhTotal, flash_kwhtotal);
        previousMillis=millis();
      }
  //convert double em string
```

```
  dtostrf(kwhTotal, 8, 7, charBuf);
  //Multiplica pelo valor kilowatt hora R$ 0.33 Reais
  vlreais = kwhTotal * 0.33;
  // text display tests
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(BLACK);
  display.setCursor(0,0);
  //display.println("Hello, world!");
  display.print("KW/h:");
  display.println(charBuf);
  display.setTextSize(1);
  display.setTextColor(BLACK);
  display.print("Consumo Watts:");
  //dtostrf(Irms, 10, 8, charBuf);
  display.print(Irms*126);
  display.println("Watts:");
  display.println("");
  display.print("R$: ");
  dtostrf(vlreais, 8, 7, charBuf);
  display.print(charBuf);
  display.display();
  //delay(2500);
}


void printFloat(float value, int places) {
  // this is used to cast digits
  int digit;
  float tens = 0.1;
  int tenscount = 0;
  int i;
  float tempfloat = value;

    // make sure we round properly. this could use pow from <math.h>, but
doesn't seem worth the import
  // if this rounding step isn't here, the value  54.321 prints as 54.3209

  // calculate rounding term d:  0.5/pow(10,places)
  float d = 0.5;
  if (value < 0)
    d *= -1.0;
  // divide by ten for each decimal place
  for (i = 0; i < places; i++)
    d/= 10.0;
  // this small addition, combined with truncation will round our values
properly
  tempfloat +=  d;

  // first get value tens to be the large power of ten less than value
  // tenscount isn't necessary but it would be useful if you wanted to know
after this how many chars the number will take

  if (value < 0)
    tempfloat *= -1.0;
  while ((tens * 10.0) <= tempfloat) {
    tens *= 10.0;
    tenscount += 1;
  }
```

```
  // write out the negative if needed
  if (value < 0)
    Serial.print('-');

  if (tenscount == 0)
    Serial.print(0, DEC);

  for (i=0; i< tenscount; i++) {
    digit = (int) (tempfloat/tens);
    Serial.print(digit, DEC);
    tempfloat = tempfloat - ((float)digit * tens);
    tens /= 10.0;
  }

  // if no places after decimal, stop now and return
  if (places <= 0)
    return;

  // otherwise, write the point and continue on
  Serial.print('.');

  // now write out each decimal place by shifting digits one by one into the
ones place and writing the truncated value
  for (i = 0; i < places; i++) {
    tempfloat *= 10.0;
    digit = (int) tempfloat;
    Serial.print(digit,DEC);
    // once written, subtract off that digit
    tempfloat = tempfloat - (float) digit;
  }
}
```

Ligação: Arduino – Sensor de Corrente SCT-013-000 – Fonte

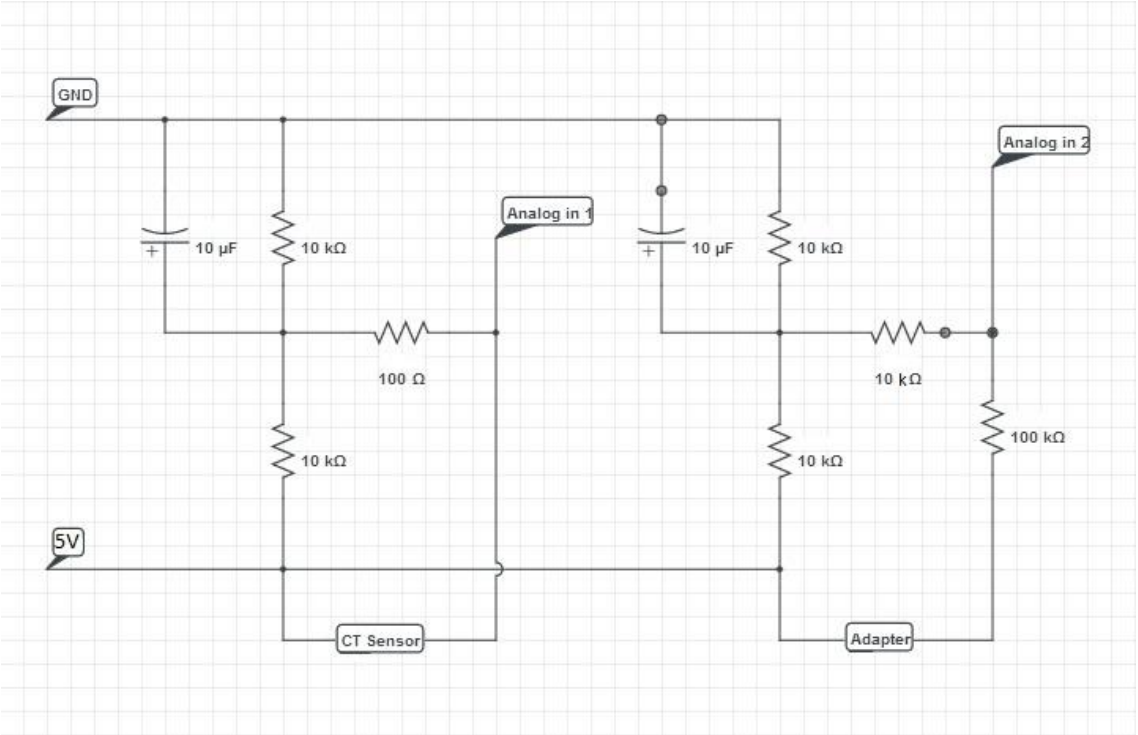


5 Resistores iguais (marrom-preto-laranja): 10kOhm

Resistor (marrom-preto-amarelo): 100kOhm

Resistor (marrom-preto-marrom): 100Ohm

Diagrama do Circuito



Referências Bibliográficas

http://openenergymonitor.org/emon/buildingblocks/how-to-build-an-arduino-energy-monitor

http://openenergymonitor.org/emon/buildingblocks/ct-sensors-introduction

http://openenergymonitor.org/emon/buildingblocks/ct-sensors-interface

http://openenergymonitor.org/emon/buildingblocks/measuring-voltage-with-an-acac-power-adapter

http://www.youtube.com/watch?v=XkzRabGOAMo

http://www.projetoarduino.com.br/index.php

http://www.arduino.cc/

http://store.radioit.com.br/

http://www.robocore.net/

http://pt.wikipedia.org/wiki/Placa_de_Ensaio

https://www.circuitlab.com/editor/#