

CONTROLE DE CONSUMO ELÉTRICO INTELIGENTE

Electrical consumption intelligent control

Everton Pereira da Cruz¹
Pedro Sidnei Zanchet¹
Alex Sandro Vermohlen¹
Allan Ricardo Nunes¹
José Diego Zunino¹

Resumo: O presente projeto consiste na mensuração do consumo de energia residencial, obtendo em tempo real, através de aplicativo, o valor referente à quantidade de consumo de energia e seu custo financeiro, possibilitando ao usuário compreender consumos e gastos obtidos, auxiliá-lo na tomada de decisão e poder controlar diretamente o uso de tais parâmetros. Também demonstrará qual o racional de cálculo utilizado para construir gráficos, bem como a administração das informações quanto a sua coleta e tratamento. Enfim, irá abordar a ideia de administrar valores de consumo elétrico através de conceitos matemáticos, físicos e de arquitetura orientada aos serviços.

Palavras-chave: Controle. Arduino. Corrente. Sensor. *Web Service*. Cidades Inteligentes.

Abstract: The present project consists in the measurement of the residential energy consumption, obtaining in real time through the application value referring to the amount of energy consumption and its financial cost, allowing the user to understand the consumptions and expenses obtained and to be able to assist in decision making and power Control the use of such parameters. It will also demonstrate the calculation rationale used to construct graphs as well as the administration of the information regarding its collection and treatment. Finally, it will address the idea of managing electric consumption values through mathematical, physical, and service-oriented architecture concepts.

Keywords: Control. Arduino. Chain. Sensor. Web Service. Smart Cities.

Introdução

Nos atuais métodos de cobrança de energia elétrica, o proprietário de uma residência não consegue acompanhar e controlar o que está realmente gastando e o que precisará pagar no próximo mês. Deste modo, o mesmo não consegue controlar efetivamente seus gastos, impossibilitando-o de saber se está dentro do seu orçamento ou não.

De um modo geral, esse projeto pretende auxiliar o usuário com o controle de consumo de energia elétrica residencial, visto que ele pode gerar relatórios diários, semanais ou mensais para saber quais os horários que está sendo desperdiçada mais energia. Outro ponto é utilizar como comparativo para a conta de luz do final do mês gerada pela companhia de energia de cada cidade.

O objetivo desta ferramenta é o controle do gasto em energia elétrica em residências, empresas e locais públicos, em que o proprietário do estabelecimento poderá acompanhar em tempo real o seu gasto com a energia e também tirar relatório de histórico de gastos.

Segundo França (2015), metrificar o desempenho de um processo é contribuir para o alcance do objetivo do mesmo. Desta forma, quantificar o consumo através do monitoramento proativo permite ao usuário administrar melhor a relação de consumo/gasto da energia elétrica, bem como otimizar qualquer interface que dependa de tal gasto.

¹ Centro Universitário Leonardo Da Vinci – UNIASSSELVI – Rodovia BR 470 – KM 71 – nº 1.040 – Bairro Benedito – Caixa Postal 191 – 89130-000 – Indaial/SC Fone (47) 3281-9000 – Fax (47) 3281-9090 – Site: www.uniasselvi.com.br

Arduino e sensores

A atual proposta consiste em mensurar o consumo de energia elétrica de uma determinada residência, ou partindo para uma visão maior, abordar o consumo até mesmo de um condomínio através de contabilização individual das residências presentes no mesmo, partindo da premissa de monitorar a interface que alimenta as residências por corrente elétrica.

Segundo Tojano (2015), corrente elétrica consiste no movimento ordenado de cargas elétricas, através de um condutor elétrico. Para isto, nosso projeto prevê a utilização de um sensor específico ligado ao cabo de alimentação (fase) que por sua vez estará ligado entre o poste de distribuição e a determinada residência.

De acordo com Beock (2012), sensores são dispositivos que trabalham com medidas de grandezas físicas. Estas são medidas pelos sensores, que por sua vez são combinadas a fim de obter informações sobre o meio físico, em que estão presentes.

Portanto, tal sensor estará atrelado ao Arduino, monitorando de forma contínua a intensidade de energia que está sendo transmitida em tal cabo.

Arduino é um pequeno computador que você pode programar para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele. O Arduino é o que chamamos de plataforma de computação física ou embarcada, ou seja, um sistema que pode interagir com seu ambiente por meio de *hardware* e *software* (MCROBERTS, 2011, p. 22).

Segundo Monk (2014), Arduino é uma pequena placa de microcontrolador que contém um conector USB que permite ligá-la a um computador, além de diversos pinos que permitem a conexão com circuitos eletrônicos externos, como motores, relés, sensores luminosos, diodos etc., portanto, podem ser programados para trabalharem de forma autônoma.

O objetivo era elaborar um dispositivo que fosse ao mesmo tempo barato, funcional e fácil de programar, sendo dessa forma acessível a estudantes e projetistas amadores. Além disso, foi adotado o conceito de *hardware* livre, o que significa que qualquer um pode montar, modificar, melhorar e personalizar o Arduino, partindo do mesmo *hardware* básico (BLOG FILIPEFLOP, 2014).

Desta forma, o atual trabalho consiste em ser implementado sobre tal interface.

Leitura e envio de dados

A cada determinado intervalo de tempo, o Arduino apurará e contabilizará este dado de entrada (proveniente do sensor de corrente) de forma numérica, a fim de resguardar tal valor. Deste modo, para controlar toda a taxa de consumo de energia em uma residência, é necessário ainda um sensor de corrente elétrica. *Hardware* este que permite a leitura dessa energia que está sendo movimentada no sensor que de fato irá realizar a coleta de dados e enviará ao Arduino.

Dados estes que, por sua vez, serão em seguida enviados à internet através de um dispositivo secundário chamado de “*Internet Shield*”.

O Arduino Ethernet Shield torna sua placa Arduino *on-line* de uma maneira fácil e rápida, basta encaixar este Shield em sua placa Arduino e plugá-lo na rede, e com este tutorial você poderá em minutos monitorar o estado de chaves e sensores pelo seu browser, computador e celular de qualquer parte do mundo! (BLOG FILIPEFLOP, 2014).

Este último é acoplado ao Arduino como se fosse uma extensão do mesmo, o que possibilita comunicar o controlador com a interface *web*.

Estipulada a leitura (sensor), contabilização (Arduino), e destino (Internet), tais dados serão enviados através de um “*Web Service*”, que finalmente irão ser abrigados em um banco de dados.

De acordo com Breda e Moraes (2007), *Web Services* disponibilizam uma maneira de diferentes tipos de aplicação, possivelmente rodando em diversas plataformas e sistemas operacionais a fim de interagirem.

Desta forma, até este ponto, tem-se a leitura da intensidade de distribuição de energia de uma residência, que serão lidas e enviadas à internet, que por fim poderão ser acompanhadas em uma interface *web*, podendo então serem visualizadas em um *smartphone* ou computador etc.

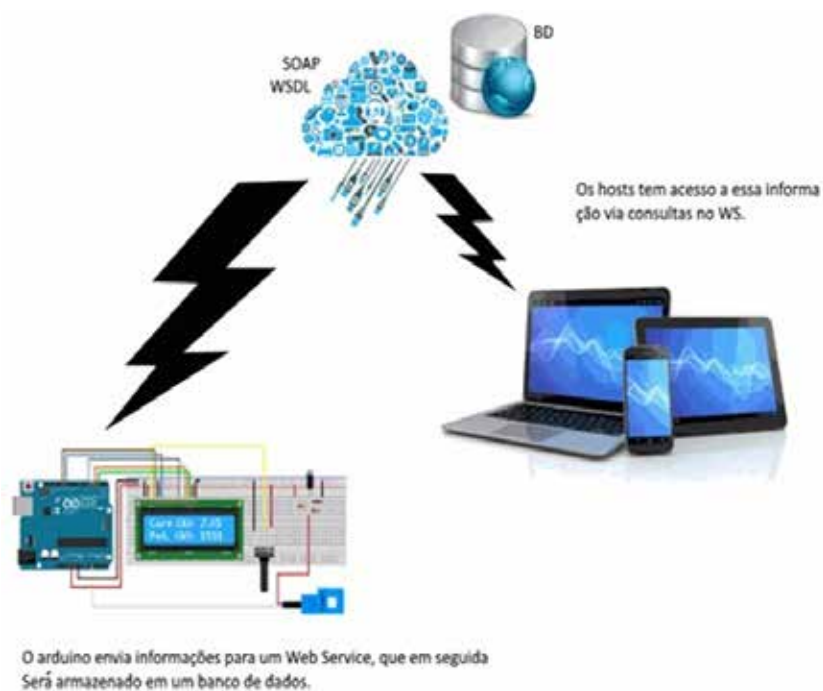
Com as informações de leitura resguardadas em um banco de dados, será possível não só visualizar o consumo da rede elétrica, mas também, visualizar demais indicadores que poderão ser gerados com base em tais dados.

Este cenário pode ser aplicado a demais residências, replicando em cada uma o cenário proposto acima. Portanto, este conceito poderá ser utilizado para monitorar demais áreas urbanas utilizando-se da faceta de conjunto. Vale lembrar que, uma vez que tais informações serão enviadas à internet (e alocadas em uma base de dados), será possível prover maiores integrações, como por exemplo, até mesmo integrar e utilizar recursos distintos da página do órgão responsável pela distribuição de energia, ou então diferentes aplicativos, visando aperfeiçoar ou prover qualquer necessidade do usuário ou até mesmo do prisma gerencial (gerenciamento de uma cadeia de residências).

Modelo proposto e caso de uso

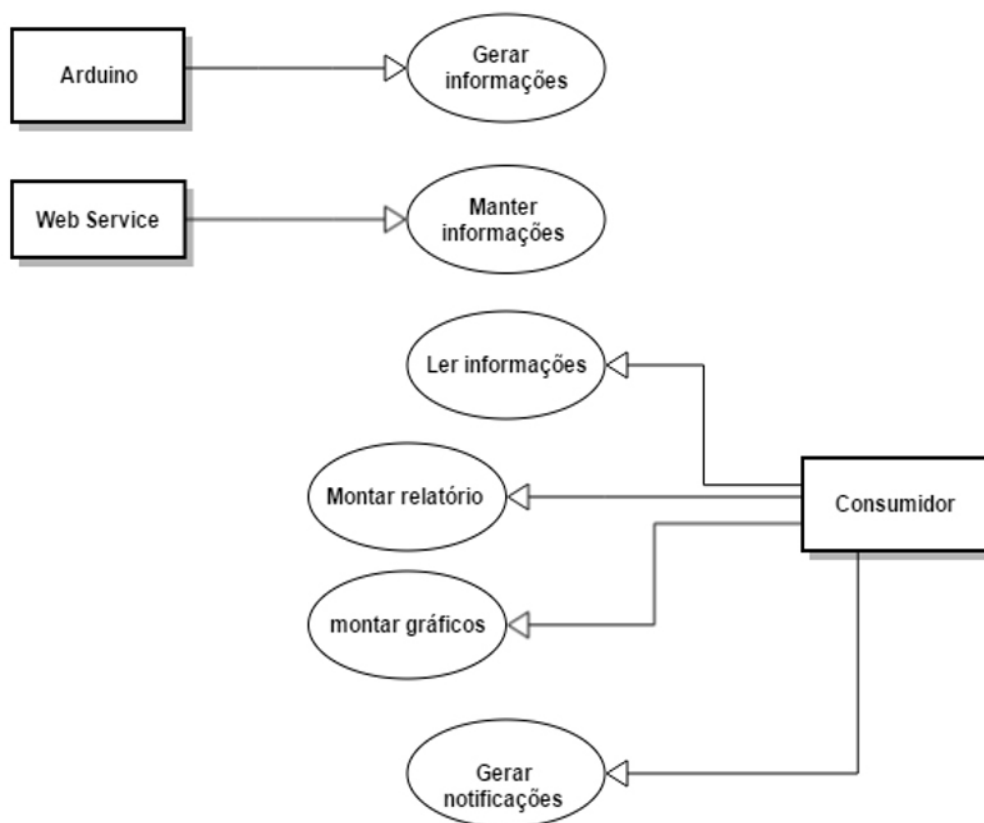
A seguir, podemos observar tanto o modelo proposto, como seu caso de uso:

Figura 1. Ilustração do modelo proposto



Fonte: Elaborado pelos autores (2016).

Figura 2. Caso de uso



Fonte: Elaborado pelos autores (2016).

Especificação de *Hardware*

Abaixo estão os componentes utilizados para o desenvolvimento deste projeto, bem como a especificação de cada um deles. Vale salientar que ao longo do desenvolvimento do projeto, tais componentes poderão ser substituídos para atender ao escopo do trabalho.

Demais componentes serão necessários para este projeto, diodos, resistores, *jumpers*/ fios etc., no entanto, discorreremos sobre os principais componentes.

- Componente: Arduino R3

Figura 3. Arduino R3



Fonte: Disponível em: <<http://blog.filipeflop.com/arduino/medidor-de-energiaeletrica-com-arduino.html>>. Acesso em: 3 jul. 2017.

Abaixo segue especificação do componente Arduino Uno R3:

Quadro 1. Especificações Arduino Uno R3

Microcontrolador	ATmega328
Voltagem Operacional	5V
Voltagem de entrada (recomendada)	7-12V
Voltagem de entrada (limites)	6-20V
Pinos E/S digitais	14 (dos quais 6 podem ser saídas PWM)
Pinos de entrada analógica	6
Corrente CC por pino E/S	40 mA
Corrente CC para o pino 3,3V	50 mA
Flash Memory	32 KB (ATmega328) dos quais 0,5KB são utilizados pelo bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidade de Clock	16 MHz

Fonte: Disponível em: <<https://www.arduino.cc/en/Main/ArduinoBoardUno>>. Acesso em: 3 jul. 2017.

- Componente: *Ethernet Shield W5100*

Para o envio de dados para a Internet/*Web Service*, será utilizado o *Ethernet Shield*. Segue abaixo informações do mesmo:

Figura 4. *Ethernet Shield W5100*



Fonte: Disponível em: <ftp://imall.iteadstudio.com/Shield/IM120911007/DS_IM120911007_W5100_Ethernet_Shield.pdf>. Acesso em: 3 jul. 2017.

Bem como especificação do mesmo:

Quadro 2. Especificações *Ethernet Shield*

Tamanho do PCB	55.88mm X 68.58mm X 1.6mm
Indicadores	TX,RX,COL,FEX,SPD,LNK
Fonte	5V
Protocolo de Com.	SPI
RoHS	SIM

Especificação	Min.	Max	Unidade
Voltagem	3V	5,5	VDC
Voltagem Entrada VH	3	5,5	V
Voltagem de Entrada VL	-0,3	0,5	V
Corrente	-	100	mA

Fonte: Disponível em: <ftp://imall.iteadstudio.com/Shield/IM120911007/DS_IM120911007_W5100_Ethernet_Shield.pdf>. Acesso em: 3 jul. 2017.

- Componente: Sensor de Corrente Não Invasivo 100A SCT-013

Para a captura dos dados, deverá utilizado o sensor abaixo:

Figura 5. Sensor de Corrente Não Invasivo 100A SCT-013

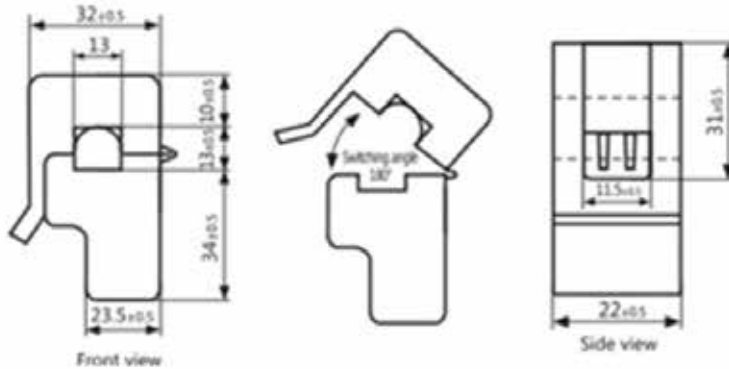


Fonte: Disponível em: <http://garden.seeedstudio.com/images/1/10/SCT013000_datasheet.pdf>. Acesso em: 3 jul. 2017.

Abaixo segue especificação do componente do sensor de corrente:

Figura 6. Especificações Sensor de Corrente

Corrente	0-100A
Saída	Corrente/33mA
Resistente a Fogo	Sim
Material	Ferrite
Temperaturas	-25 °C à 70 °C



Fonte: Disponível em: <http://garden.seeedstudio.com/images/1/10/SCT013000_datasheet.pdf>. Acesso em: 3 jul. 2017.

Software

O *software* disponibilizado em cada Arduino é desenvolvido no próprio *framework* do *hardware*, disponibilizado pela própria fabricante. Seu método de desenvolvimento é bem simples e necessita de conhecimentos em programação e um pouco de eletrônica (visto que se trata de componentes elétricos e controle de saídas digitais e analógicas).

O *software* para programação do Arduino é uma IDE que permite a criação de *sketches* para a placa Arduino. A linguagem de programação é modelada a partir da linguagem *Wiring*. Quando pressionado o botão *upload* da IDE, o código escrito é traduzido para a linguagem C e é transmitido para o compilador *avr-gcc*, que realiza a tradução dos comandos para uma linguagem que pode ser compreendida pelo microcontrolador (EVANS; NOBLE; HOCHENBAUM, 2013).

Para os autores, nesse aplicativo será lida a quantidade de energia utilizada pelo sensor, que por sua vez será repassada ao Arduino, realizando o tratamento dos dados, e em seguida enviando estes ao *Web Service*, onde serão armazenados na base de dados.

Web Service

Conforme o tópico anterior, este projeto tem o conceito de disponibilizar todos os dados obtidos em *web* métodos disponibilizados em um *Web Service*.

De acordo com Lecheta (2015), *Web Services* são utilizados como forma de integração e comunicação de sistemas, de modo que um sistema possa realizar uma chamada para um serviço de outro sistema, a fim de obter informações. Estas chamadas podem enviar e receber informações em diversos formatos, sendo que atualmente os mais populares são XML e JSON. Uma das grandes vantagens na construção de *Web Services* é que eles permitem acessar os serviços de uma forma padronizada e independente da linguagem de programação.

É possível escrever um *Web Service* com a linguagem Java, mas consumi-lo em qualquer outra linguagem, como *C#, PHP, Python* etc. Este conceito trouxe um grande avanço na maneira como os sistemas são construídos, pois permite separar e quebrar o sistema em partes, onde então, após a leitura dos dados por parte do Arduino, será chamado esse método que tem o objetivo de armazenar as informações no bando de dados. Todos os dados recuperados ficarão disponibilizados *on-line*, em que cada residência poderá ter seu registro com as informações recuperadas ao longo do tempo.

Um outro método disponibilizará esses dados, retornando os dados de determinada residência quando solicitado. Essa função pode ser integrada em várias plataformas, desde sistemas *on-line* até *smartphones*.

Funcionamento

Inicialmente será ligado o “Sensor de Corrente Não Invasivo 100A SCT-013” à rede elétrica da residência (no canal “fase”), a qual deverá possuir a tensão de 220 Volts. Em seguida, conectar o pino de energia, neutro e sinal à Placa Arduino UNO R3, a qual já deverá ter o programa carregado em sua memória.

A placa “*Ethernet Shield W5100*”, sendo um módulo serial sem fio, deve ser conectada ao Arduino Uno R3. Em seguida, a placa *ethernet* deve ser conectada via cabo RJ45 a um *modem* com acesso à internet.

Os demais componentes, como diodos e resistores, devem ser aplicados de acordo com as especificações técnicas do sensor e Arduino.

O hardware e o software do Arduino são ambos de fonte aberta, o que significa que o código, os esquemas, o projeto etc. podem ser utilizados livremente por qualquer pessoa e com qualquer propósito. Dessa forma, há muitas placas-clone e outras placas com base no Arduino disponíveis para compra, ou que podem ser criadas a partir de um diagrama (MCROBERTS, 2011, p. 22).

Em seguida, o próprio *software* carregado no Arduino irá realizar a comunicação com o *Web Service*, que já estará preparado para receber os dados gerados pelo mesmo. *Web Service* este que será desenvolvido em PHP.

Todos os dados coletados pelo *Web Service* serão armazenados em um banco de dados MYSQL, que por sua vez também serão disponibilizados no mesmo servidor. Para a consumação destas informações, bastará ser feita integrações com o *Web Service*, chamando o método de leitura de informações. Este método pode ser invocado por aplicações em *forms*, aplicações *web*, Android, IOS e a todas as plataformas de desenvolvimentos que disponham de integrações SOA.

Implementação – Corrente alternada

A energia elétrica que é distribuída aos usuários domésticos e indústrias nas cidades é alternada. Isto significa que num instante de tempo ela é positiva e em outro instante ela é negativa, e seus valores máximos (pico) atingem valores maiores que o valor nominal. Por exemplo: uma rede de energia elétrica com 220 Volts tem o seu valor de pico em aproximadamente 311 Volts, ambas em frequência de 60 Hz.

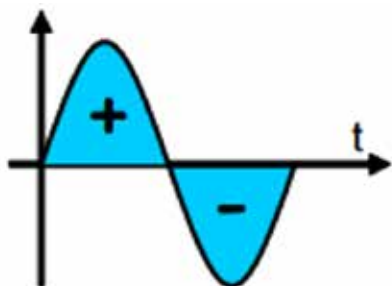
Na prática, o que se tem na rede elétrica CA é um sinal senoidal de 60 ciclos por segundo (60Hz), cuja tensão varia a todo instante desde +311,1V a -311,1V, passando por zero a cada meio ciclo. A tensão eficaz de 220V é o valor correspondente a uma tensão contínua constante que produziria o mesmo efeito da rede CA numa dada resistência (MUSSOI, 2006, p. 32).

A corrente elétrica alternada (CA) possui como característica marcante a sua variância no tempo, alternando o sentido quando passa no condutor. Devido a ser alternada, possui características como frequência, amplitude e fase.

Onda senoidal

É a representação gráfica da forma com que uma onda evolui ao longo do tempo. Normalmente os fenômenos ondulatórios obedecem a funções matemáticas periódicas. A onda senoidal ou sinusoidal obedece a uma função seno ou cosseno e é a forma de onda mais simples e adotada neste caso de medição da corrente elétrica.

Figura 7. Onda Senoidal



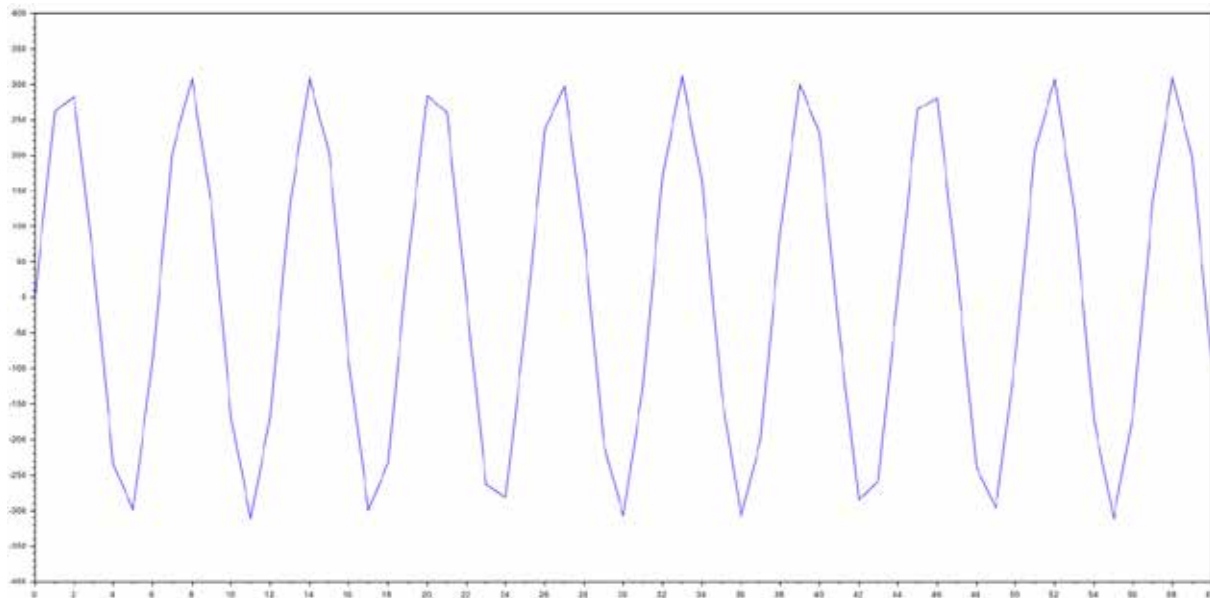
Fonte: Adaptado de Mussoi (2006, p. 6).

Coleta simulada de dados

Para melhor exemplificar, realizamos a simulação quanto à coleta de dados pertinentes à corrente elétrica através do programa *SCILAB* (saiba mais em: <<http://www.scilab.org/>>). Portanto, utiliza-se desta interface para demonstrar qual o racional que deverá ser utilizado, bem como suas propriedades e particularidades.

Desta forma, pode-se verificar na imagem abaixo a onda senoidal obtida em tal cenário:

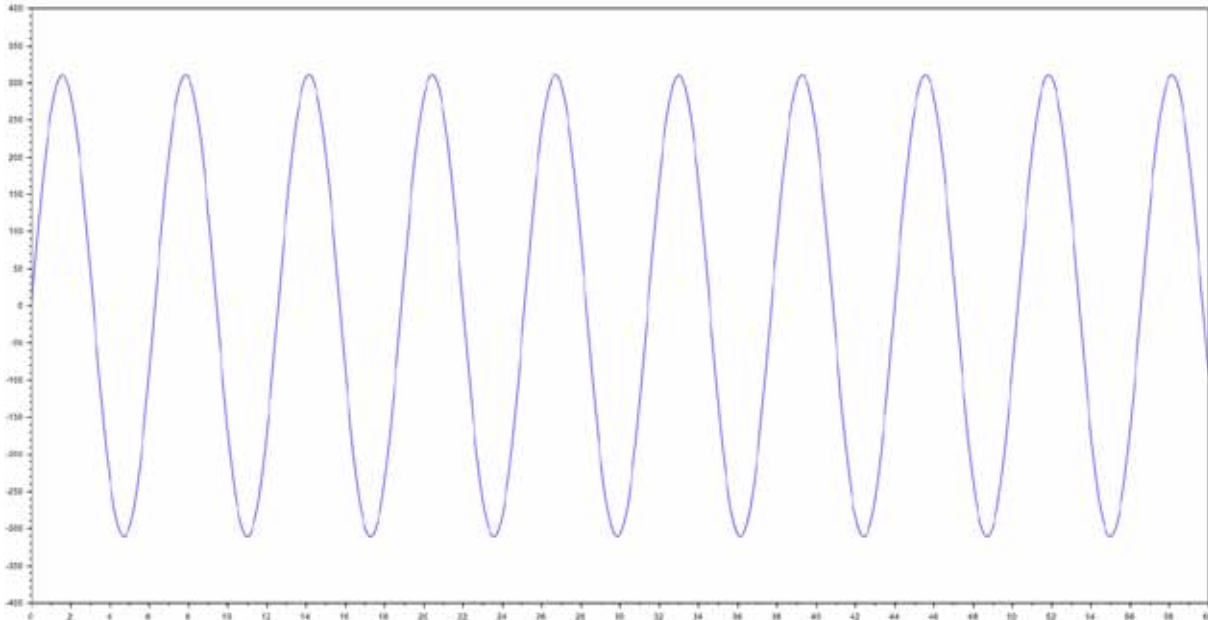
Figura 8. Onda Corrente Senoidal



Fonte: Elaborado pelos autores (2016).

Relacionando a Figura 6 com a Figura 7 pode-se observar que para esta última as curvas são apresentadas de modo irregular. Neste caso, será necessário o ajuste de curva para o fator de precisão. Deste modo, utilizamos um segundo programa para ajustar a curva deste cenário. Trata-se do aplicativo *Curve Expert*. Este, por sua vez, atenua as curvas entre os pontos.

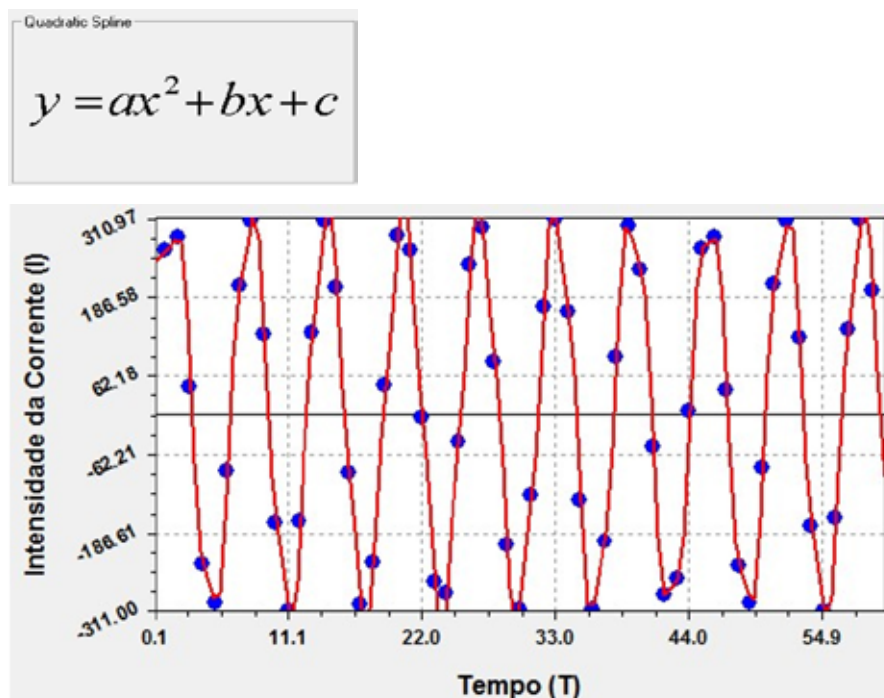
Figura 9. Ajuste de Curva Realizado



Fonte: Elaborado pelos autores (2016).

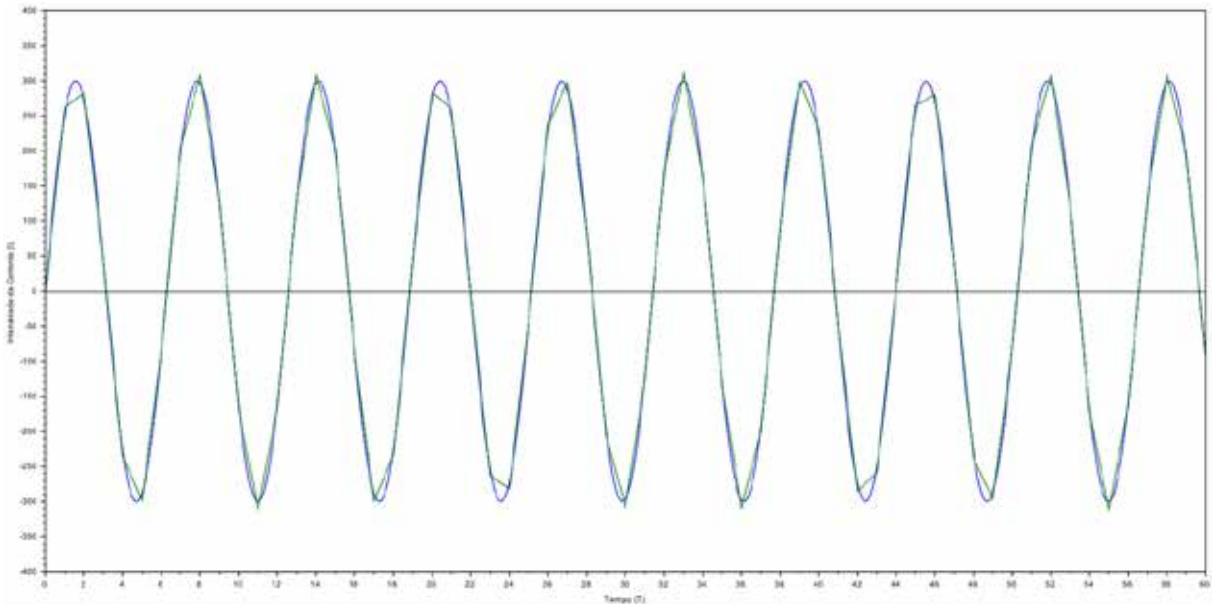
Portanto, tem-se finalmente o gráfico contendo as curvas ajustadas. Na imagem abaixo pode-se comparar o ajuste realizado.

Figura 10. Ajuste de curva pelo aplicativo *Curve Expert*



Fonte: Elaborado pelos autores (2016).

Figura 11. Relação Ajuste de Curva



Fonte: Elaborado pelos autores (2016).

Como afirmou-se anteriormente, a corrente alternada justamente alterna sua composição em vários sentidos, como frequência, amplitude e fase. Desta forma, é necessário correlacionar a frequência com o tempo de leitura do sensor. Este último coleta dados a cada 1 segundo. Portanto, deve-se particionar a onda em 60 vezes (devido aos 60 Hz). Obtemos então o seguinte cálculo:

$$T = 1/F$$

Onde:

T = Tempo.

1 = 1 Segundo.

F = Frequência.

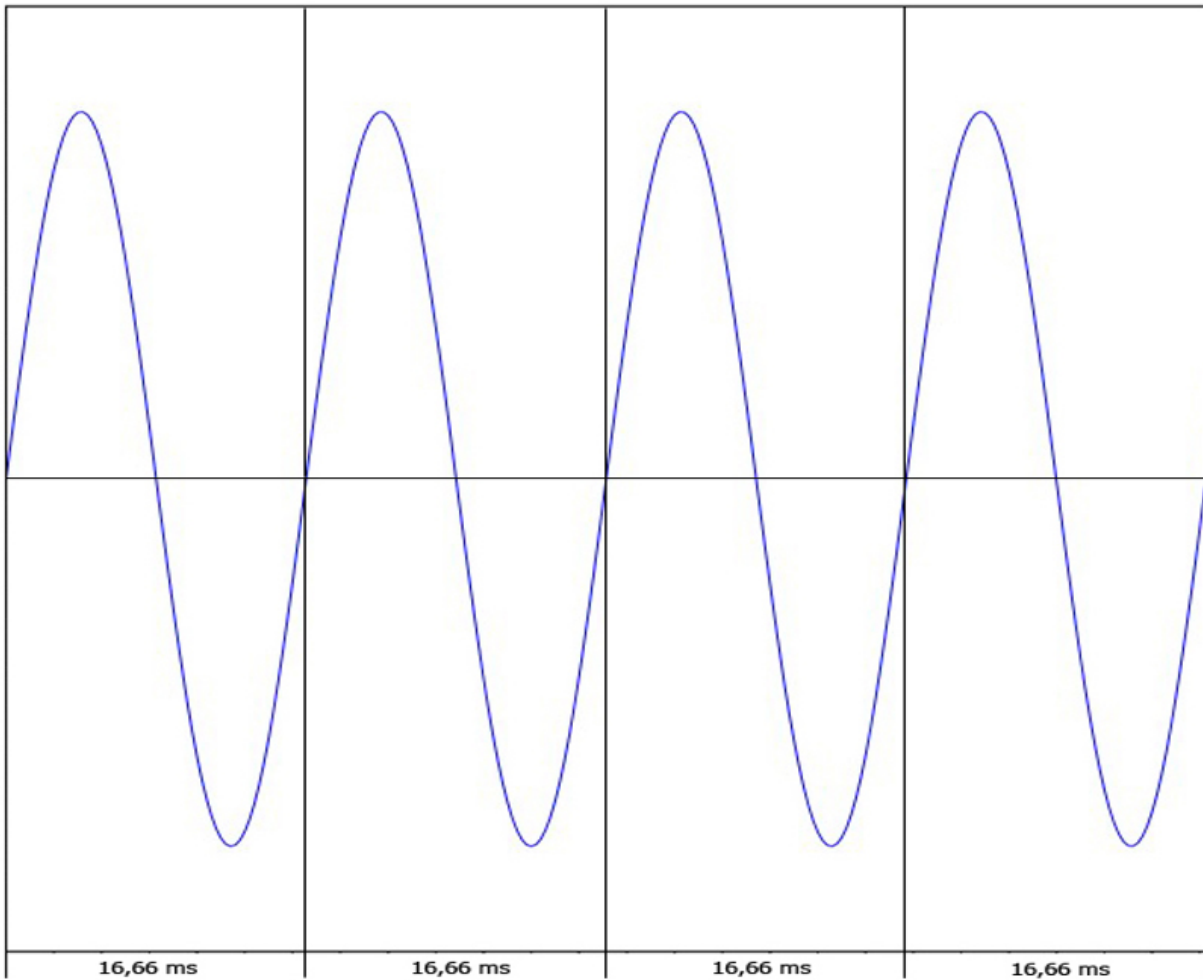
$$T = 1/60$$

T = 0,0166 Segundos

T = 16,66 milisegundos

Tendo como base essas informações, coleta-se os dados através do sensor de corrente a cada 1 segundo, e neste intervalo teremos 60 Hz, o que significa 60 repetições de ondas senoidais por segundo. Portanto, em cada parte dessa repetição (60 vezes) levará aproximadamente 16,66 milissegundos para ser realizada.

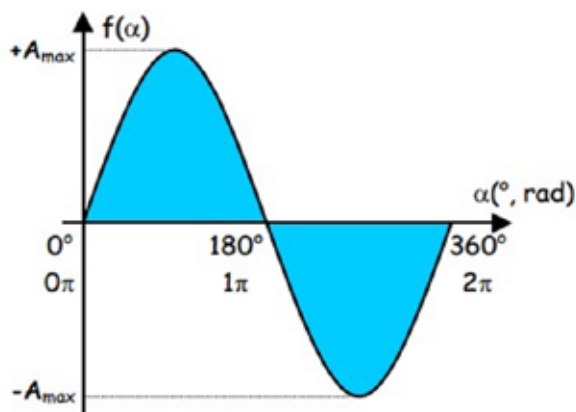
Figura 12. Intervalo de Repetições



Fonte: Elaborado pelos autores (2016).

Para encontrar a posição da onda dentro da repetição (posição inicial) utiliza-se o valor de π (π – calculado através da circunferência dividido pelo diâmetro) como referencial. Obtém-se, então, 360 graus – 180° ora positivo, tendendo a pico de 311 V, ora 180° negativo, tendendo a pico de -311 V.

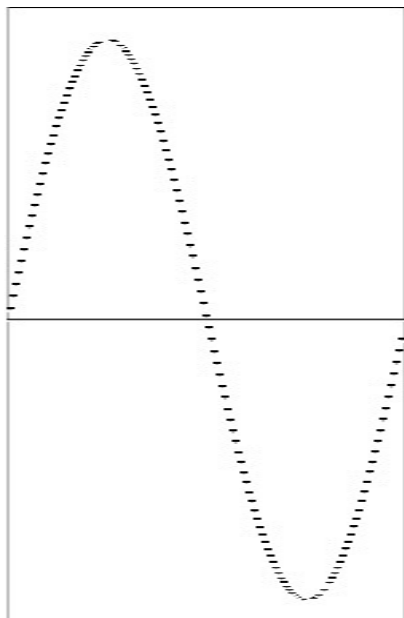
Figura 13. Representação 360°



Fonte: Adaptado de Mussoi (2006, p. 25).

Sabendo que uma volta completa de 1 Hz possui 360°, deve-se então particionar esta grandeza de 16,66 ms por 360°: teremos então, finalmente, o valor de **44 microssegundos**, valor este que corresponde ao tempo que o sensor deverá coletar os dados.

Figura 14. Representação 44 µs – Cada ponto equivale a 44 µs



Fonte: Elaborado pelos autores (2016).

Desta maneira, o sensor captará valores presentes da onda senoidal a cada 44 microssegundos, sendo que em 1 segundo teremos 360 valores lidos da corrente. Obtendo a média destes, teremos finalmente a corrente efetiva equivalente de tal período. Portanto, com a corrente elétrica obtida e sabendo a tensão da rede elétrica, pode-se finalmente calcular a potência. Esta, por sua vez, é dada segundo a equação abaixo:

$$P = V * I$$

Onde:

P = Potência (Watt/s ou J/s).

V = Tensão (Volts).

I = Corrente (A).

Desta forma, comparando com a equação acima, a grandeza que corresponde ao valor de “I” será informada pelo sensor, que será multiplicada por 220 (Tensão). Portanto, tais valores poderão ser lidos e manuseados de acordo com o *loop* a ser estipulado no Arduino.

Aplicação

Abaixo segue a codificação representando o modelo ideal de implementação:

- Definir variável para receber resultados.
- Realizar sequência de leitura em que o valor inicial será de 1 até 360.
 - o Recuperar o valor do sensor de corrente ligado ao Arduino.
 - o Com o valor obtido no passo acima, realizar a equação para chegar à potência.
 - o Aguardar 44 microssegundos para repetir esta sequência.

- Calcular a média dos 360 valores recuperados na sequência acima.
- Salvar resultado.
- Disponibilizar os dados para acesso via *Web Service*.
- Exibir resultado.

Considerações finais

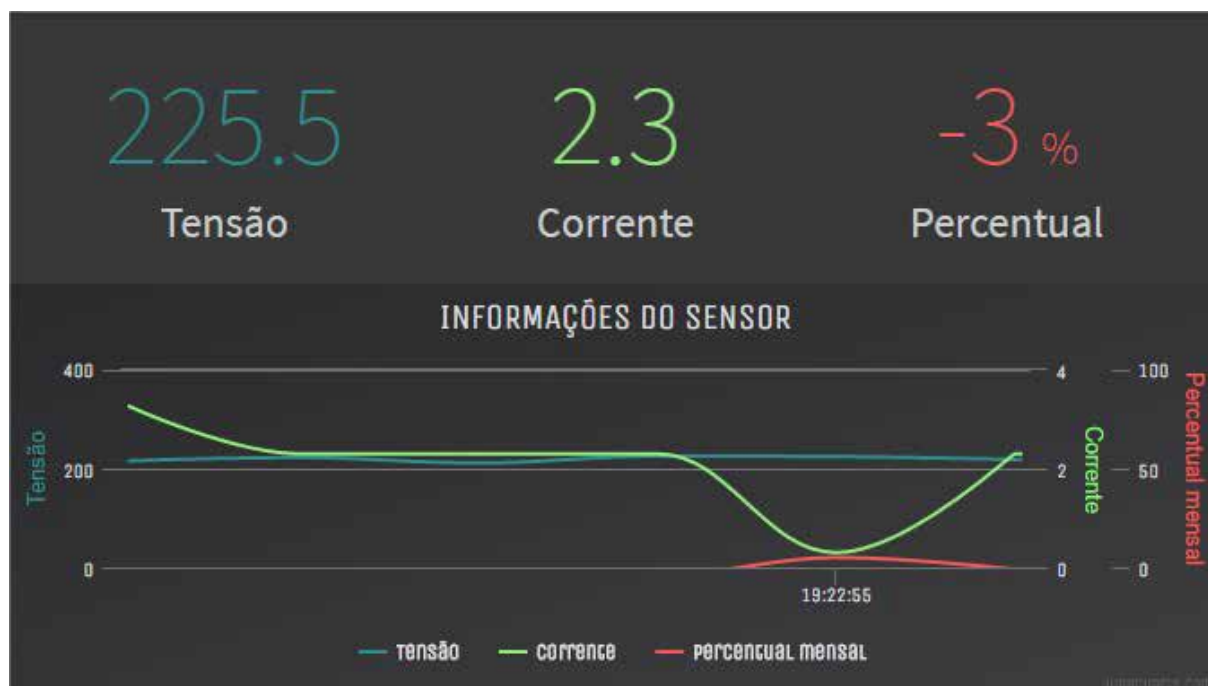
Através de uma solução com um custo de aquisição dos componentes eletrônicos não muito elevado, conseguimos trazer a ideia de um projeto em que se tem o objetivo de controlar gastos e precaver o dono da residência no gasto de energia.

O fato é que através de componentes acessíveis e de lógicas de matemática/física, é possível mensurar e avaliar dados que até então parecem ser invisíveis ao usuário final ou até mesmo não acessíveis a este.

A solução no futuro também pode ser utilizada em locais do governo público, e de certa forma abrir as informações para o público em relação ao gasto de energia, para assim conseguir dimensionar algumas despesas de cada estabelecimento, tentando reduzir os desvios de gastos públicos no pagamento da conta de energia.

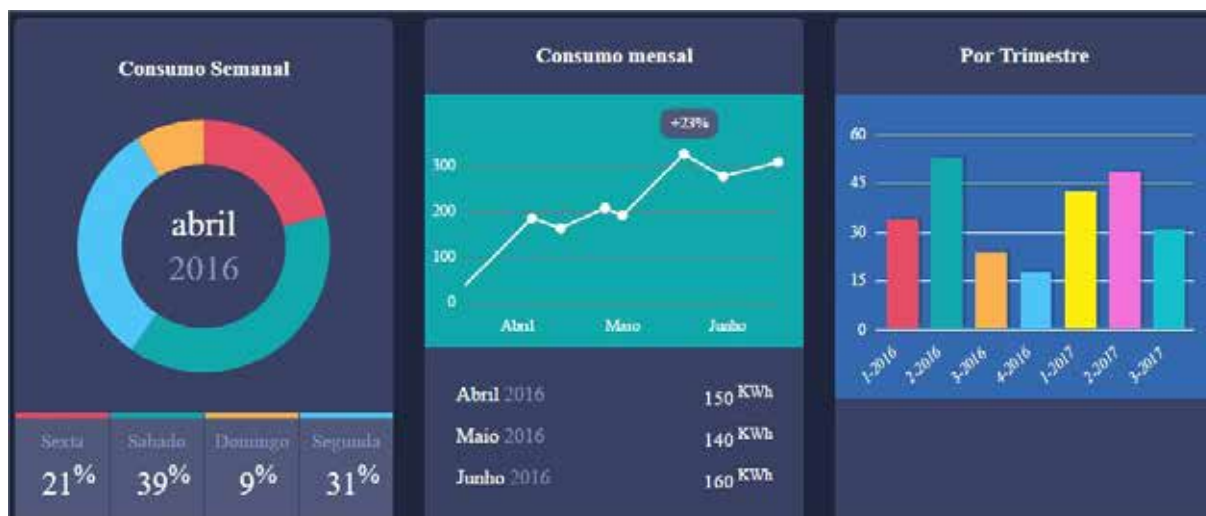
Utilizando uma interface amigável, os resultados disponibilizados via *Web Service* podem ser visualizados em qualquer dispositivo, desde que os mesmos disponham de conectividade com a internet, facilitando tanto o acompanhamento quanto a integração com demais serviços.

Figura 15. Representação de um protótipo de exibição



Fonte: Elaborado pelos autores (2016).

Figura 16. Representação de um protótipo de exibição



Fonte: Elaborado pelos autores (2016).

Os protótipos acima foram criados de modo a elucidar a ideia de integração com o projeto. Para salvar as informações apresentadas nos protótipos acima, utilizamos métodos específicos disponibilizados no *Web Service*, que salvam os valores recuperados do Arduino e geram indicadores apresentados no tópico acima.

Figura 17. Representação dos métodos no *Web Service*

RegistraMovimentacao

```
string RegistraMovimentacao(string nro_casa, string hora, string valor)
```

RegistraResidencia

```
string RegistraResidencia(string desc_casa)
```

ConsultarConsumoMensal

```
string ConsultarConsumoMensal(string cod_casa, string mes_referencia)
```

ConsultarConsumoDiario

```
string ConsultarConsumoDiario(string cod_casa, string dat_referencia)
```

Fonte: Elaborado pelos autores (2016).

Os métodos acima podem ser acessados no endereço: <<http://bluautopecas.com.br/webservice/ws01.php>>.

Vale lembrar que o presente projeto consolidou a ideia do grupo, demonstrando que é possível realizar efetivamente o projeto apresentando ideias e conceitos ao longo do documento.

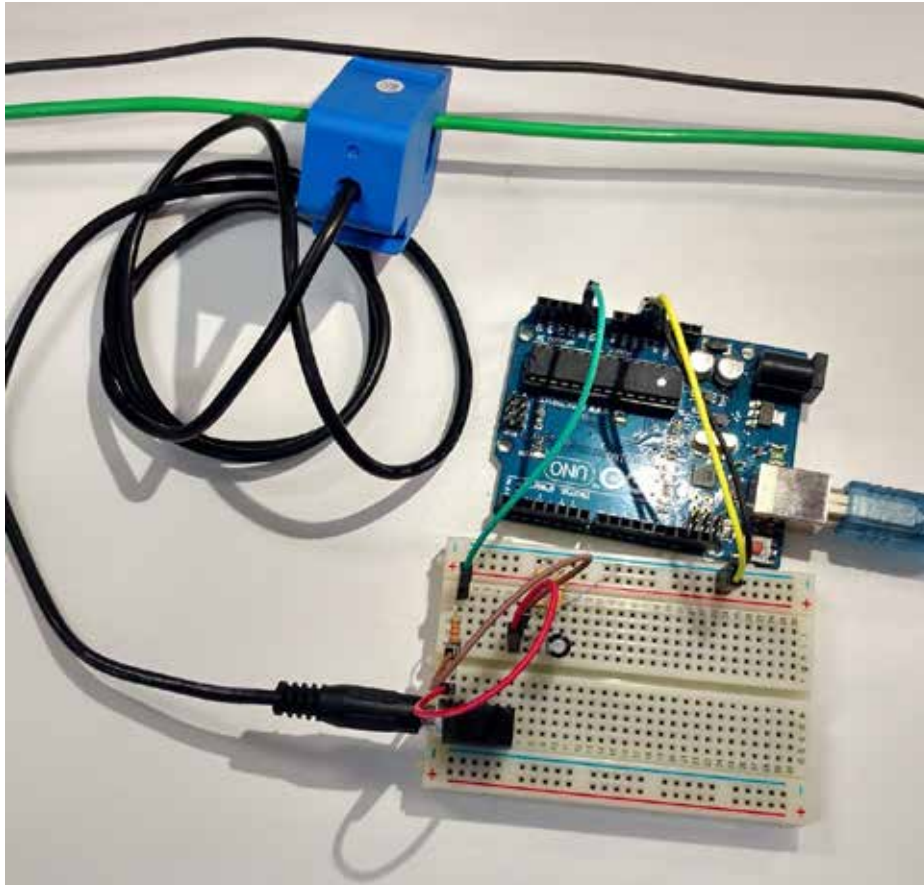
Referências

- ARDUINO E CIA. **Ethernet shield Wiznet W5100 - Parte 1**. 2013. Disponível em: <<http://www.arduinoecia.com.br/2013/06/ethernet-shield-wiznet-w5100-parte-1.html>>. Acesso em: 20 set. 2016.
- ARDUINO. **Arduino Ethernet Shield 2**. 2017. Disponível em: <<https://www.arduino.cc/en/Main/ArduinoEthernetShield>>. Acesso em: 20 set. 2016.
- BEOCK, Leandro Rodrigo Preis. **Robótica Educacional**. 2012. Disponível em: <<http://leandro-robotica.blogspot.com.br/2011/05/conceito-e-sensores-e-atuadores.html>>. Acesso em: 20 set. 2016.
- BLOG FILIPEFLOP. **Componentes eletrônicos**. 2014. Disponível em: <<http://blog.filipeflop.com/arduino/medidor-de-energia-eletrica-com-arduino.html>>. Acesso em: 20 set. 2016.
- _____. **Qual Arduino comprar? Conheça os tipos de Arduino**. 2014. Disponível em: <<http://blog.filipeflop.com/arduino/tipos-de-arduino-qual-comprar.html>>. Acesso em: 20 set. 2016.
- BREDA, Marcus; MORAES, Juliano. **Web Services**. 2007. Disponível em: <http://www.inf.pucrs.br/~gustavo/disciplinas/sd/material/Artigo_WebServices_Conceitual.pdf>. Acesso em: 20 set. 2016.
- EVANS, M.; NOBLE, J; HOCHENBAUM, J. **Arduino em Ação**. São Paulo: Novatec, 2013.
- FRANÇA, André Luiz. **KPI's – O que são e como utilizar de forma correta**. 2015. Disponível em: <<http://www.digai.com.br/2015/02/kpis-o-que-sao-e-como-utilizar-de-forma-correta/>>. Acesso em: 4 nov. 2016.
- LECHETA, Ricardo R. **Web Services RestFul**. São Paulo: Editora Novatec, 2015.
- MCROBERTS, Michael. **Arduino Básico**. São Paulo: Editora Novatec, 2011.
- MONK, Simon. **30 Projetos com Arduino**. São Paulo: Editora Bookman, 2014.
- MUSSOI, Fernando Luiz Rosa. **Sinais senoidais: tensão e corrente alternadas**. CEFET/SC, 3. ed. Florianópolis, 2006. Disponível em: <http://www.electronica-pt.com/images/fbfiles/files/Sinais_Senoidais_Tensao_e_Corrente_Alternadas_CEFET_SC.pdf>. Acesso em: 4 jul. 2017.
- NOVA ELETRONICA. **Conheça todas as placas Arduino**. [s.d.]. Disponível em: <<http://blog.novaeletronica.com.br/conheca-todas-placas->
- TOJANO, Lídia. **Corrente elétrica**. 2015. Disponível em: <<http://www.mundovestibular.com.br/articles/757/1/CORRENTE-ELETRICA/Paacutegina1.html>>. Acesso em: 20 set. 2016.

USINAINFO. **Sensor de Corrente Não Invasivo 100A SCT013**: Saída 50mA. [s.d.]. Disponível em: <<http://www.usinainfo.com.br/sensores-para-arduino/sensor-de-corrente-nao-invasivo-100a-sct013-saida-50ma-3417.html>>. Acesso em: 20 set. 2016.

VENTRON. **Montando Web Service para Arduino**. [s.d.]. Disponível em: <<http://www.ventron.com.br/como-montar-um-webserver-com-arduino-e-ethernet-shield/>>. Acesso em: 20 set. 2016.

APÊNDICE A – Protótipo físico implementado conforme projeto



APÊNDICE B – Captura de dados obtidos através da implementação em *hardware*

Irms: 0.27 A	Potencia: 58.98 W
Irms: 0.37 A	Potencia: 81.96 W
Irms: 0.44 A	Potencia: 95.86 W
Irms: 0.41 A	Potencia: 89.19 W
Irms: 0.37 A	Potencia: 82.36 W
Irms: 0.42 A	Potencia: 91.54 W
Irms: 0.36 A	Potencia: 80.01 W
Irms: 0.42 A	Potencia: 91.73 W
Irms: 0.47 A	Potencia: 103.14 W
Irms: 0.42 A	Potencia: 91.95 W
Irms: 0.37 A	Potencia: 81.19 W
Irms: 0.37 A	Potencia: 82.44 W

Artigo recebido em 30/05/17. Aceito em 10/07/17.