

# AUTOMAÇÃO DE CASAS E ESTABELECIMENTOS COMERCIAIS ATRAVÉS DE MICROCONTROLADORES: UMA REVISÃO DA APLICABILIDADE DO ARDUINO.

Bruno Strippoli dos Santos<sup>1</sup>  
Lucas Medon Bianco Fonseca<sup>2</sup>  
Lucas Sernaglia<sup>3</sup>  
João Emmanuel das Neves Alkmin<sup>4</sup>

DOI: 10.47283/244670492020080270

## Resumo

Nos últimos anos, o aumento da demanda global por energia criou diversos desafios desde sua constituição original, o que temos hoje é uma busca por fontes cada vez mais sustentáveis e formas de otimizar o consumo e o abastecimento energético. Esse artigo de revisão tem como objetivo investigar e estabelecer de forma prática e sucinta uma forma de implementar um sistema de leitura e controle automatizados de baixo custo nos setores residenciais e comerciais. Através da revisão de estudos e artigos, será apresentado passo a passo como adaptar os componentes de uma residência para que eles possam ser controlados de forma remota utilizando microcontroladores como recurso. No trabalho em questão, serão explorados alguns recursos e configurações de uma placa Arduino UNO que possui uma IDE (Ambiente de Desenvolvimento Integrado) que permite a um usuário programar funções específicas para seus equipamentos, transformando-os em dispositivos inteligentes (*Smart Appliances*). O protótipo aplicado nesse estudo pode ser replicado e incorporado na rede elétrica de estabelecimentos e residências junto com sensores e reguladores para obtenção de controle mais preciso e complexo.

**Palavras-chaves:** *Smart Appliances, Arduino, Microcontroladores, energia sustentável.*

## Abstract

*With each passing year, the ever growing energy demand across the world never ceases to bring up new challenges, one of the most important of them being finding sustainable and optimized alternatives for our energy needs. The main objective of this article is to investigate and establish a practical way to set up a low cost automated system capable of managing the energy needs of a residence or business building. This was possible through the review and analysis of other studies and articles, ultimately describing how to remotely connect house appliances to microcontrollers, in this particular project the Arduino UNO board was chosen to demonstrate that it is possible, through its IDE, to develop methods that change how appliances act, effectively making them Smart Appliances. The prototype applied in this study can be replicated and incorporated in the energy frame of homes and commercial buildings along with sensors and regulators for more precise and complex control.*

**Keywords:** *Smart Appliances, Arduino, Microcontrollers, sustainable energy.*

## Introdução

---

<sup>1</sup> Discente da Fatec Americana. E-mail: brunosantos@gmail.com

<sup>2</sup> Discente da Fatec Americana. E-mail: lucasmbfonseca@gmail.com

<sup>3</sup> Discente da Fatec Americana. E-mail: lsernaglia@gmail.com

<sup>4</sup> Docente da Fatec Americana. E-mail: jeneves@gmail.com

Nos últimos anos, o aumento da demanda global por energia criou diversos desafios desde sua constituição original. O que se tem hoje é uma busca por fontes cada vez mais sustentáveis e formas de otimizar o consumo e o abastecimento energético. Atualmente, a maioria dos países e empresas privadas de fornecimento de energia adaptaram suas redes elétricas para o sistema *Smart Grid*, o qual incorpora os benefícios da computação distribuída e de comunicações para fornecer informações em tempo real e permitir o equilíbrio quase instantâneo da oferta e da procura dos bens energéticos (TORRI, 2012).

De acordo com Cangussu (2008 apud TORRI, 2012, p.49-50) o setor residencial nos Estados Unidos assume a segunda posição no país no consumo de energia elétrica, correspondendo junto com o setor comercial a um total de 36,25% consumido em 2007, isso significa que a maior parte da energia produzida no país é destinada a usuários domésticos e empresas comerciais e seus eletrônicos e eletrodomésticos. Enquanto isso, nos países da União Européia, o consumo residencial representa 42% de toda a energia consumida pelo bloco (TOMPROS et al, 2008, apud TORRI, 2012, p.50).

Torri (2012) comparou tecnologias de medidores de energia e mostra que o sistema de medição avançada incorporada em *Smart Grids (Advanced Metering Infrastructure - AMI)* possui superioridade em relação aos sistemas manual e de medição automática (*Automatic Metering Reading - AMR*). A coleta de dados da AMI é feita de forma remota, a gravação das informações é baseada na utilização do usuário e feita a cada hora, ou intervalos mais frequentes. É possível implementar um display de consumo de energia, gerenciar interrupções e emitir respostas de acordo com a demanda energética.

No Brasil, o consumo médio residencial subiu quase 50% em menos de 20 anos de acordo com pesquisas realizadas pelo EPE e IBGE (NETA, 2017). Dado esse fato, devemos procurar buscar soluções em pequena escala que possam gerar um impacto no consumo energético geral.

O conceito das *Smart Grids* pode ser trazido para dentro de nossas próprias casas através de diversas modificações que podemos realizar em nossos próprios aparelhos e suas fontes de alimentação.

Esse artigo de revisão tem como objetivo investigar e estabelecer de forma prática e sucinta uma forma de implementar um sistema de leitura e controle automatizados de baixo custo nos setores residenciais e comerciais, seguindo as tendências globais de implantação de conceitos de *Smart Grids*, gerando economia para os usuários e consequentemente diminuindo a demanda geral de energia elétrica.

Nos últimos anos o volume de estudos a respeito do tema cresceu de forma exponencial. No entanto, ainda se carece de um trabalho que especifique de forma clara e direta como esse recurso pode ser implementado e desfrutado. O objetivo desse trabalho é estabelecer essas diretrizes usando um software e hardware específicos de baixo custo já existente no mercado, interpretado por uma linguagem de programação específica e que se conectam ao elemento de Hardware (microcontroladores) inseridos na matriz energética e nos *Smart Appliances*.

Através da revisão de estudos e artigos, será demonstrado passo a passo como adaptar os componentes de uma residência para que eles possam ser controlados de forma remota utilizando microcontroladores como recurso. Nesse artigo em questão, será utilizada como referência uma placa Arduino que possui uma IDE (Ambiente de desenvolvimento integrado) e que permite a um usuário programar funções específicas para seus equipamentos, transformando-os em dispositivos inteligentes (*Smart Appliances*). Será utilizada como referência na elaboração de um protótipo uma placa Arduino UNO.

## 1 Smart Appliances

SCHMIDT (2001) define *Smart Appliances* como sendo dispositivos que são atentos aos seus respectivos ambientes e CALDAS (2015) os define de forma mais específica:

“*Smart Appliances*, dispositivos eletrônicos que dispõem de uma interface que permite a recessão de pedidos e ações e o seu processamento. Para possibilitar a sua gestão, é necessário ter uma rede que permita a todas as *Smart Appliances* estarem ligadas nesse ponto” (CALDAS, 2015, p.12).

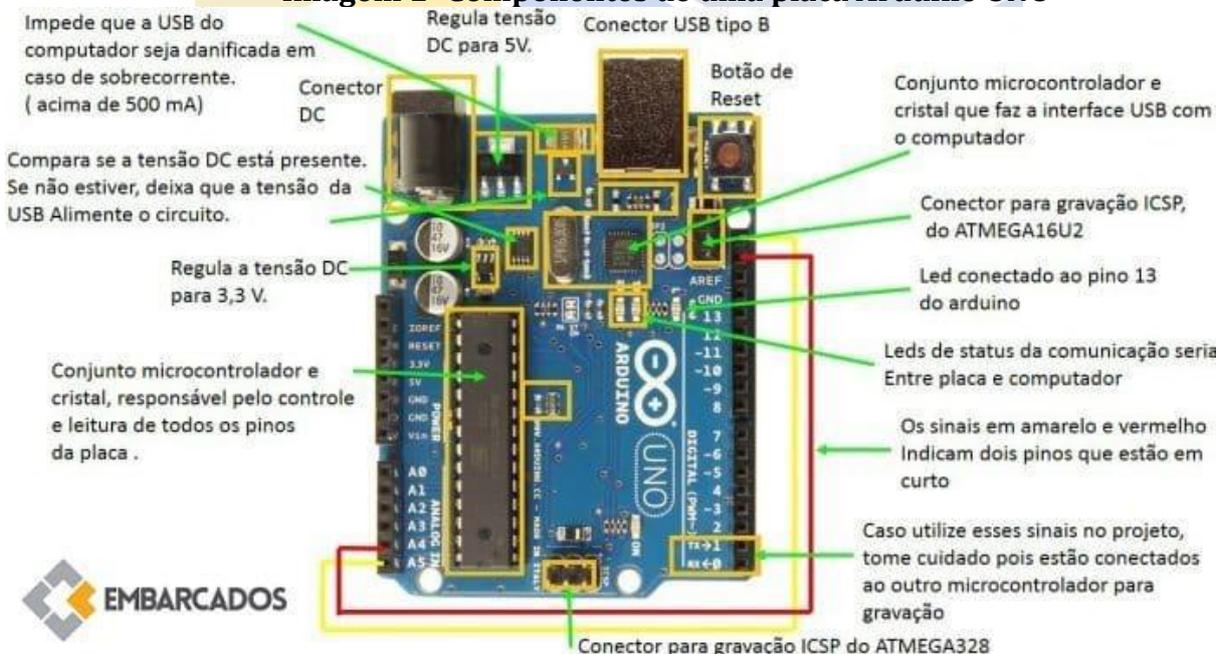
Dadas as definições supracitadas, pode-se dizer ainda que é possível adaptar dispositivos que não são originalmente caracterizados como *Smart Appliances* para que se encaixem na definição de um, sendo esse o escopo deste projeto. Serão destrinchados os requisitos básicos para tal e elaborar um protótipo reproduzível em diferentes escalas e naturezas.

## 2 Arduino

De acordo com o fabricante, Arduino é um dispositivo de Hardware *open-source* que pode ser reproduzido de forma genérica. As placas Arduino tem a capacidade de leitura de dispositivos de entrada (Como sensores de uma lâmpada, um dedo em um botão e até mesmo uma mensagem no Twitter) e transformá-los em uma resposta de saída, como ativar um motor, ligar uma lâmpada LED e até mesmo publicar algum conteúdo programado online.

Ao programar uma série de instruções em sua IDE (*Integrated Development Environment*) – Ambiente de Desenvolvimento Integrado você pode dizer à sua placa Arduino o que fazer, graças ao microcontrolador embutido na placa.

**Imagem 1- Componentes de uma placa Arduino UNO**



Fonte: EMBARCADOS, 2020

## 3 Definição de microcontroladores

Santos (2019) traz em sua tese a definição de microcontroladores como equipamentos programáveis de pequeno porte e de baixo custo que possui uma unidade de processamento, memória, entrada e saída e que tem como função realizar ações de controle de forma remota.

Existem diversos modelos de microcontroladores que apresentam essas características como padrão, podendo variar entre si na pinagem (quantidade de entradas e saídas) e na capacidade de seus processadores, podendo haver microcontroladores de ambientes multiprogramados, ou seja, que aceitam mais de uma série de instruções ao mesmo tempo. O Arduino possui tais características e se encaixa no escopo deste projeto pela facilidade de implementação, baixo custo e forma simples de gerenciamento, no entanto ele não tem a capacidade de se conectar com uma rede por si só, para isso, utilizam-se os *Shields* que expandem as funcionalidades do Arduino.

Existem diversos tipos de *Shields* que permitem ao Arduino se conectar a uma rede *wifi* ou através de *Bluetooth* ou até mesmo um *Ethernet Shield* para conectar-se diretamente à rede via cabo.

Um microcontrolador contém um microprocessador e periféricos como: dispositivos de entrada e saída (E/S), memória de dados e programa, conversores analógicos - digitais (A/D) e digitais - analógicos (D/A), temporizadores, geração PWM, entre outros, integrando em um único circuito. Com o auxílio de um software de ambiente para desenvolvimento integrado (IDE) é possível desenvolver um programa (algoritmo) para o controle dos periféricos do microcontrolador, e, em consequência atuar sobre os dispositivos externos (IBRAHIM, 2002).

As aplicações geralmente são dedicadas para uma determinada tarefa, sendo alguma delas:

- Automação industrial e residencial;
- Eletrodomésticos;
- Instrumentos de medição;
- Dispositivos móveis

#### **4 Como se comunicar com a placa arduino**

Existe um serviço consolidado e de baixo custo que oferece uma ferramenta completa para entusiastas e profissionais. O *Blynk* é um aplicativo personalizável que te permite controlar e reportar placas de Hardware programáveis baseadas em Arduino. O serviço é composto por 3 elementos: O *Blynk App*, *Blynk Server* e *Blynk Libraries*.

O *Blynk App* é a interface propriamente dita, é o ambiente que lhe permite criar aplicações que interagem com o hardware através de um espaço específico, através dessa interface um usuário consegue criar e manipular diversas funções numa única placa. Já *Blynk Server* é o ambiente programado em nuvem onde ficam armazenadas todas as informações e estados do aplicativo e também da placa, ele é o intermediário que viabiliza a comunicação dos sensores do hardware, dos comandos do aplicativo e armazena as respostas geradas. O *Blynk Libraries* é responsável por gerir toda a conexão do *hardware* com o servidor *Blynk* e gerir as requisições de entrada e saída de dados e comandos. A forma mais fácil e rápida é utilizá-la como bibliotecas Arduino. Esses três componentes fazem parte de uma única ferramenta que interagem para viabilizar a aplicabilidade do Arduino (mas não somente) na automação de *Smart Appliances* ou até mesmo em processos de produção de baixa escala (CALDAS, 2015)

Como o *Blynk* possui um servidor próprio, é possível conectar sua placa Arduino à rede sem o uso de shields e baseado nas definições de Schmidt (2001) e Caldas (2015), o *Blynk App* pode ser usado como referência na construção de um protótipo, pois possui uma interface que permite executar ações e pedidos a placa Arduino, gerenciando-o dessa forma, bem como um

servidor próprio para eventualmente conectar outros *Smart Appliances* a uma única rede se necessário.

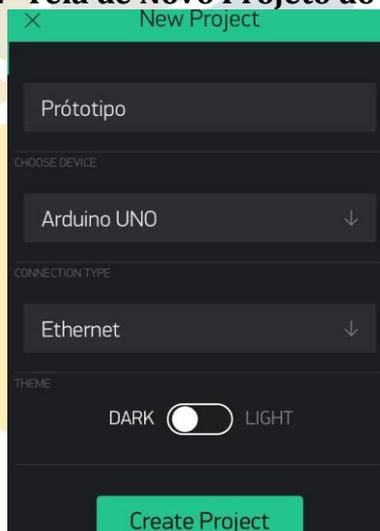
## 5 Criando um protótipo de controle de led usando arduino e o *Blynk App*.

Para criar um protótipo de controle de LED's usando Arduino e o Blynk App primeiro deve ser criada uma conta Arduino em <<https://auth.arduino.cc/register>> e em seguida feito o download em <<https://www.arduino.cc/en/Main/Software>>. Crie também uma conta *Blynk* e em seguida, faça o download do *Blynk App* disponível para IOS e Android em suas respectivas lojas.

### 5.1 Configurando o protótipo no *Blynk App*.

Logo na interface principal do aplicativo é possível iniciar um projeto ao clicar em “*New Project*”. Na tela seguinte, insira o nome do projeto e selecione o tipo de placa que será utilizada (o aplicativo é compatível com uma série de placas, a placa selecionada será a do Arduino UNO para construção do protótipo). Deve ser selecionada também a forma como a placa Arduino irá se conectar com o servidor *Blynk*(no propósito do protótipo, via USB). Por fim, o usuário deve clicar em “*Create Project*”. O aplicativo irá gerar uma chave de autenticação que será enviada no e-mail cadastrado que deverá ser usada no código-fonte na interface do Arduino que será apresentado posteriormente.

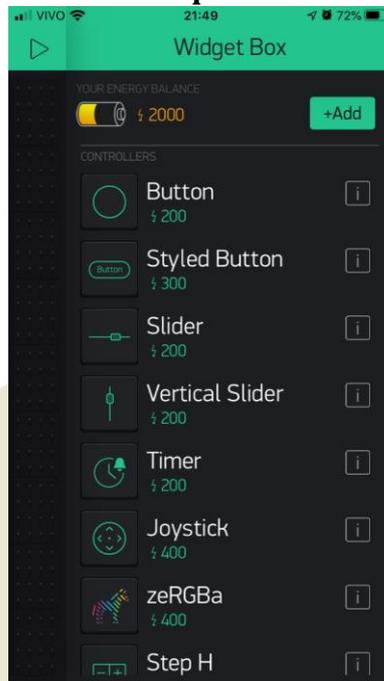
Imagem 2- Tela de Novo Projeto do Blynk App



Fonte: Documentação Blynk, 2020

Com o projeto aberto, o usuário pode clicar em qualquer ponto da tela para adicionar um botão. O aplicativo irá disponibilizar algumas alternativas de botões, selecionaremos um botão comum (primeira opção, “*Button*”).

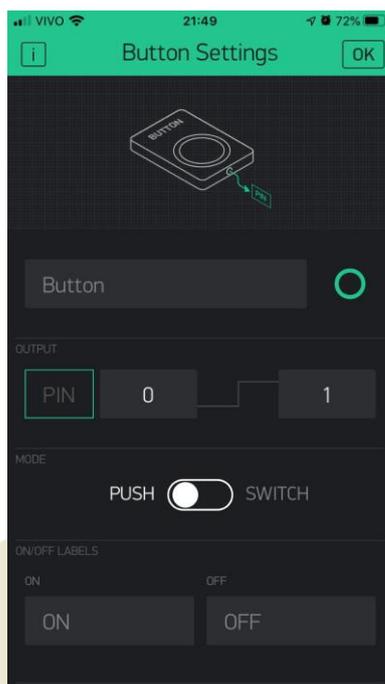
### Imagem 3- Estilos de botões personalizáveis do Blynk App



Fonte: Documentação Blynk, 2020

Em seguida deve-se clicar no botão para acessar suas propriedades. Nessa tela é possível atribuir a cor ao botão compatível ao LED utilizado na placa. É possível selecionar também qual entrada o aplicativo irá manipular na placa através da sessão “PIN” e por fim é possível selecionar o tipo de botão (“PUSH” ou “SWITCH”) de acordo com a preferência do usuário. Ao clicar em OK as configurações do botão ficarão salvas em seus projetos.

### Imagem 4- Configurações e pinagem dos botões selecionados em um projeto



Fonte: Documentação Blynk, 2020

## 5.2 Instalando a biblioteca *Blynk* na IDE do Arduino

Para o Arduino conseguir interpretar os comandos gerados pelo aplicativo *Blynk*, deve-se instalar sua biblioteca (“*Blynk Library*”) na IDE do Arduino. *Blynk Library*, de acordo com seu próprio site é uma extensão que manipula todas as rotinas de conexão e troca de dados entre a peça de hardware, a *Blynk Cloud* e o seu projeto no *Blynk App*. (BLYNK, 2020) .

O download da biblioteca pode ser feito em <<https://github.com/blynkkk/blynk-library/releases/tag/v0.6.1>> . Na IDE do Arduino o usuário deve selecionar a opção *Sketch -> Incluir Biblioteca -> Incluir Biblioteca.ZIP* e selecionar o diretório da biblioteca baixada.

## 5.3 Configurando as funções da placa Arduino em sua IDE

Nessa etapa deve-se inserir o código que irá executar a ação que desejada, no caso deste protótipo simples, o dispositivo móvel irá apenas controlar uma lâmpada de LED ligada a placa Arduino UNO utilizando o *Blynk App* em um dispositivo móvel. Já existe uma ampla gama de códigos e projetos disponibilizados pela própria comunidade do Arduino e pelo *Blynk App* e que podem ser acessados em <<https://examples.blynk.cc/>>. Para uma placa alimentada via USB, o seguinte caminho deve ser seguido: clique em *Arquivo-> Exemplos -> Blynk -> Boards\_USB\_Serial -> Arduino\_Serial\_USB*. Esse processo define a porta serial USB na qual o *Blynk* irá trabalhar. O usuário deve verificar no menu *ferramentas* se a sua placa está selecionada corretamente.

## 5.4 Chave de autenticação

Ao realizar os passos anteriores uma janela de código será aberta com uma linha descrita como: `char auth[] = "YourAuthToken";` no campo “*YourAuthToken*” o usuário deve

inserir a chave de autenticação disponibilizada pelo *Blynk App* e que foi enviada por e-mail no momento da criação de um novo projeto.

Isso permite que o acesso à manipulação da placa seja restrito a um único usuário. Após inserida a chave de autenticação, deve ser feito o upload do código pelo menu do *Sketch*.

## 6 Distribuição e organização física da placa

Os materiais necessários para montagem de um protótipo simples são:

- Placa arduino UNO
- LED difuso de qualquer cor (o arduino possui seu próprio LED, mas como desejamos aplicar um experimento reproduzível numa rede elétrica, iremos conectá-lo a um LED externo)
- Resistor de 150 Ohm até 1k (no caso da placa ser alimentada por uma fonte que não seja a USB do desktop) - O resistor serve para limitar a corrente e evitar a queima do LED, a tensão de um LED é de 2v enquanto a tensão da alimentação do arduino é de 5v, caso a porta serial definida seja a USB, não existe a necessidade de limitação de corrente (BAUERMEISTER, 2018)
- *Protoboard* - Placa de distribuição utilizada para construir circuitos elétricos. Ideal para protótipos pois pode ser reutilizada, sem a necessidade de confeccionar uma placa de circuito impresso.
- 2 Jumpers macho/macho - Responsáveis por conduzir o fluxo elétrico
- Computador - Será utilizado para alimentar a placa via USB e configurar a IDE do arduino.

### 6.1 - Organização da Protoboard

Uma protoboard, como abordado acima é um componente fundamental na construção de um circuito elétrico experimental.

**Imagem 5- Organização de uma Protoboard**



Fonte: Website FILIPEFLOP, 2020

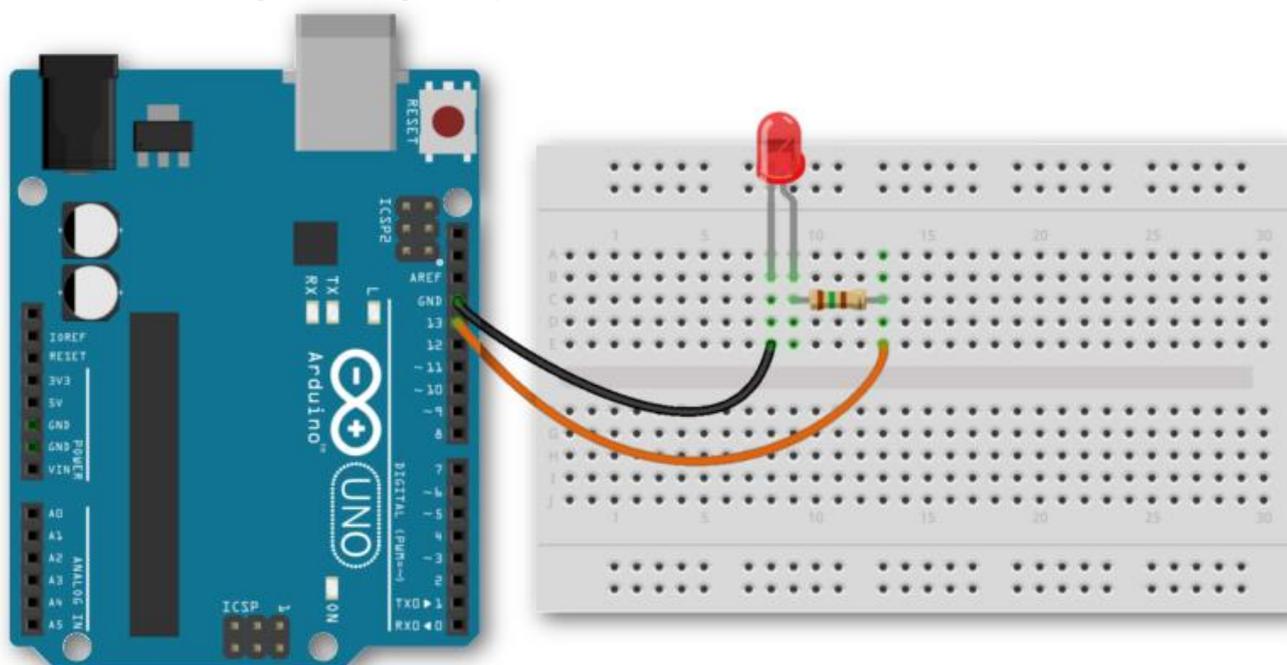
As conexões da sessão A e D na *protoboard* são interligadas de forma horizontal, enquanto que as conexões das sessões B e C são interligadas verticalmente. A área de trabalho para este protótipo são as sessões que se conectam verticalmente, ou seja, B e C.

Um LED possui polaridade, ou seja, um terminal positivo (terminal maior) e um negativo (terminal menor) e ao ser conectado os terminais não devem ser interligados para evitar a queima do mesmo, isso quer dizer que os terminais vão ser conectados em colunas

diferentes da sessão B ou C. Um dos Jumpers será conectado na mesma coluna do terminal negativo (terminal menor) do LED e a outra extremidade conectada ao componente *GND* (*Ground*) do Arduino. Por via das dúvidas deve ser utilizado um resistor, que deve ter uma extremidade (A) conectada na mesma coluna do terminal positivo (terminal maior) do LED, enquanto sua extremidade (B) deve ser conectada em uma coluna diferente e sem conectividade com as demais conexões. Por fim, o segundo Jumper é conectado na coluna da extremidade (B) do resistor enquanto sua extremidade é conectada a entrada de pinos digitais do Arduino (É necessário se certificar de conectar o mesmo pino que foi configurado no *Blynk App*, como visto na sessão 7.1).

A imagem a seguir ilustra como deve ficar a organização física da Protoboard e da placa Arduino UNO.

**Imagem 6- Organização das conexões entre o Arduino e a Protoboard**



Fonte: Website FILIPEFLOP, 2020

Feitos todos os procedimentos, a placa Arduino é ligada em seu computador através do cabo USB que o alimenta. Se todas as etapas foram seguidas, a placa já está conectada ao *Blynk Server* e pode ser controlada através do aplicativo em um dispositivo móvel.

## 7 Previsões de custo

Os componentes para a construção do protótipo podem ser adquiridos individualmente ou em kits. A seguir, é apresentada uma projeção do custo médio de um kit para iniciantes, suficiente para montar um protótipo inicial.

**Tabela 1- Média de custo de aquisição de um kit Arduino para iniciantes.**

Kit Arduino para iniciantes		
Loja 1	Loja 2	Loja 3

Custo: R\$111,51	Custo: R\$110,00	Custo: R\$95,00
Custo médio: R\$105,50		

Fonte: Os autores

### Considerações finais

Existem diversas formas de reduzir e controlar o consumo de energia elétrica nos setores residenciais e comerciais. Ao apresentar ao usuário uma forma de leitura em tempo real do consumo de energia é possível levar informações precisas para que o mesmo tome decisões baseadas em informações. O protótipo aplicado nesse estudo pode ser replicado e incorporado na rede elétrica de estabelecimentos e residências junto com sensores e reguladores para obtenção de controle mais preciso e complexo. O propósito deste trabalho foi apresentar a simplicidade de converter dispositivos eletrônicos comuns em *Smart Appliances* de baixo custo. Por fim, pode-se também identificar que a tendência global é prosseguir com avanços concernentes a *Smart Grids*, tendo em vista o mercado e comunidade que tem se formado ao redor dessa prática. Hoje podemos encontrar uma série de funções desenvolvidas por usuários com aplicabilidade em microcontroladores, sejam elas voltadas para a economia no consumo de energia, segurança através do controle de sistemas de monitoramento e até mesmo entretenimento.

### Referências:

- ARDUINO. **What is Arduino?** Disponível em <<https://www.arduino.cc/en/Guide/Introduction>>. Último acesso em 15/Jun 2020.
- BAUERMEISTER, G. **Guia do Usuário do ESP8266**. 2018. Disponível em: <https://www.filipeflop.com/blog/guia-do-usuario-do-esp8266/>. Acesso em: 14 de março de 2020.
- BLYNK. Disponível em <<https://blynk.io/en/getting-started>>. Último acesso em 21/Jun 2020.
- CALDAS, Hugo Eduardo Esteves. **Smart Home-Smart Appliances**. 2015.
- EMBARCADOS. Disponível em <<https://www.embarcados.com.br/introducao-ao-blynk-app/>>. Último acesso em 15/Jun 2020.
- FILIFELOP. **Primeiros passos com Arduino**. Disponível em <<https://www.filipeflop.com/blog/primeiros-passos-com-arduino/>>. Último acesso em: 22/Jun 2020.
- IBRAHIM, D. **Microcontroller Based Temperature Monitoring and Control**. 1ª Ed. Newnes, 2002.
- NETA, R. M. L *et al*. Consumo eficiente de energia elétrica. **Entre Aberta Revista de Extensão**, v. 2, n. 1, 2017.
- SANTOS, J. W e LARA JR, R. C. **Sistema de automatização residencial de baixo custo controlado pelo microcontrolador esp32 e monitorado via smartphone**. Trabalho de Conclusão de Curso. Universidade Tecnológica Federal do Paraná, 2019.
- SCHMIDT, A. e VAN LAERHOVEN, K. How to build smart appliances? **IEEE Personal Communications**, v. 8, n. 4, p. 66-71, 2001.
- TORRI, L. B. **Uma proposta de arquitetura extensível para micro medição em Smart Appliances**, Dissertação de Mestrado, Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

