



UNIVERSIDADE FEDERAL DE OURO PRETO - UFOP
ESCOLA DE MINAS - EM
COLEGIADO DO CURSO DE ENGENHARIA DE CONTROLE
E AUTOMAÇÃO - CECAU



HENRIQUE AMARAL MISSON

MONITORAMENTO E CONTROLE ENERGÉTICO RESIDENCIAL VIA
SMARTPHONE

Ouro Preto, 2016

HENRIQUE AMARAL MISSON

**MONITORAMENTO E CONTROLE ENERGÉTICO RESIDENCIAL VIA
*SMARTPHONE***

Monografia apresentada ao Curso de Engenharia Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para obtenção do Grau de Engenheiro de Controle e Automação.

Orientador: Dr. Paulo Marcos de Barros Monteiro

Co-orientador: M.Sc. Adrielle de Carvalho Santana

Ouro Preto
Escola de Minas - UFOP
Março/2016

Monografia defendida e aprovada, em 10 de março de 2016, pela comissão avaliadora constituída pelos professores:



Prof. Dr. Paulo Marcos de Barros Monteiro - Orientador



Profa. M. Sc. Adrielle de Carvalho Santana – Professora Convidada



Prof. Dr. Henor Artur de Souza – Professor Convidado

M678m Misson, Henrique Amaral.

Monitoramento e controle energético residencial via
Smartphone

[manuscrito] / Henrique Amaral Misson. – 2016.

46f. : il., color., tab.

Orientador: Prof. Dr. Paulo Marcos de Barros Monteiro e
Adrielle de

Carvalho Santana.

AGRADECIMENTOS

Agradeço a Deus por mais uma conquista. Meu pai, meu maior exemplo, pelos ensinamentos, pela confiança. Agradeço a Vanusa que esteve presente em grande parte da minha vida. Agradeço às minhas irmãs Débora e Beatriz pela dedicação e incentivo. Aos meus avôs e avós, tios, tias, primos e primas pelo apoio. À Geórgia, pelo carinho, amor, dedicação e paciência.

À Universidade Federal de Ouro Preto e a École des Mines de Douai pelos conhecimentos passados, aos professores do DECAT, em especial, Paulo Monteiro e Adrielle de Carvalho que me apoiaram no desenvolvimento desse projeto.

Por fim, agradeço a República Sem Destino, minha casa, onde eu passei momentos inesquecíveis em todos os anos de Ouro Preto.

“A base de toda a sustentabilidade é o desenvolvimento humano que deve contemplar um melhor relacionamento do homem com os semelhantes e a Natureza.”

NAGIB ANDERÁOS NETO

RESUMO

O controle e monitoramento energético residencial visa auxiliar a população na redução do desperdício da energia elétrica em casa, contribuindo para o desenvolvimento sustentável, além de possibilitar uma grande economia na conta de energia. O projeto teve como objetivo o desenvolvimento de um sistema baseado na placa *Arduino Mega 2560* para realização de leituras da corrente de equipamentos residenciais que permitirem o cálculo do consumo em KWh e valor a ser pago em reais, além do controle liga/desliga desses equipamentos. Foi criado um aplicativo para *Smartphone* a partir do ambiente *App Inventor* que permitiu o acesso remoto desses dados em tempo real. A viabilidade de um projeto como este foi constatada nos resultados satisfatórios obtidos.

Palavras-chave: Energia Elétrica, Aplicativo, Dispositivos móveis, Controle, Monitoramento, Automação residencial.

ABSTRACT

The control and home energy monitoring aims to help people in reducing the energy waste at home and contributing to sustainable development, besides allowing big savings on your energy bill. This project aimed to develop a system based on Arduino Mega 2560 board for performing current readings of residential equipment that allowed the calculation of consumption in KWh and amount to be paid in “reais”, in addition to the on / off control of such equipment. An application for Smartphone was created from the App Inventor environment that allowed remote access such data in real time. The feasibility of a such project was found in satisfactory results.

Keywords: Eletricity, App, Mobile devices, Control, Monitoring, Home Automation.

LISTA DE FIGURAS

Figura 2.1 - Arduino Mega 2560	6
Figura 2.2 - Shield Ethernet.....	7
Figura 2.3 - Interface de programação IDE.....	8
Figura 2.4 – Sensor de Corrente ACS712	10
Figura 2.5 - Relé de 2 canais	10
Figura 3.1 – Configuração Darlington do sistema	14
Figura 3.2 – Ligação ACS712, Arduino e rede elétrica	15
Figura 3.3 – Circuito proposto.....	15
Figura 3.4 – Ambiente de programação do sistema.	17
Figura 3.5 – Ambiente App Inventor.	18
Figura 3.6 – Aplicativo “Monografia Misson”.....	18
Figura 3.7 – Programação em blocos App Inventor.....	19
Figura 3.8 – Código QR para instalação do aplicativo no dispositivo móvel.....	20
Figura 4.1 – Circuito Proposto	21
Figura 4.2 – Controle das lâmpadas	22
Figura 4.3 – Aplicativo em funcionamento	22
Figura B.1 – Programação da tela 1 do <i>App Inventor</i>	34
Figura B.2 – Programação da chamada do <i>IP</i> pelo aplicativo.....	34
Figura B.3 – Inicialização das variáveis globais no <i>App Inventor</i>	34
Figura B.4 – Programação do <i>status</i> das lâmpadas no aplicativo	34
Figura B.5 – Programação dos caracteres nos campos do consumo e preço.....	35
Figura B.6 – Programação dos botões liga/desligado do aplicativo.....	35
Figura B.7 – Programação do <i>clock</i> e do retorno à tela inicial do aplicativo.....	35

LISTA DE TABELAS

Tabela 2.1- Tabela de consumo médio de equipamentos domésticos	9
--	---

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Formulação do Problema.....	1
1.2	Objetivo	3
1.3	Justificativa.....	3
1.4	Estrutura do Trabalho	3
2	REVISÃO BIBLIOGRÁFICA	5
2.1	<i>Arduino</i>	5
2.2	<i>Ethernet Shield</i>	6
2.3	Interface de programação	7
2.4	Custo da energia elétrica e consumo elétrico de equipamentos residenciais	8
2.5	Sensores de corrente	9
2.6	Relés	10
2.7	Sistema operacional <i>Android</i>	11
2.8	Plataforma <i>App Inventor</i>	11
3	METODOLOGIA.....	13
3.1	Desenvolvimento do circuito.....	13
3.2	Desenvolvimento da Programação	15
3.3	Desenvolvimento do aplicativo	17
4	RESULTADOS E DISCUSSÕES	21
4.1	Testes e Resultados.....	21
4.2	Dificuldades encontradas.....	23
5	CONSIDERAÇÕES FINAIS.....	24
5.1	Conclusão	24
5.2	Sugestões para projetos futuros	24
	REFERÊNCIAS BIBLIOGRÁFICAS	25
	APÊNDICE A	28
	APÊNDICE B	35

1 INTRODUÇÃO

1.1 Formulação do Problema

O consumo energético no Brasil vem se destacando por ser um dos temas mais debatidos na atualidade. Isto se faz devido a vários fatores como o consumo irracional por parte da população, gerando uma alta demanda na produção, e o constante aumento do custo desta energia no país.

A disponibilidade de energia elétrica se enquadra dentro dos principais fatores de desenvolvimento para as nações, sendo assim, a energia elétrica passa a ser uma variável estratégica de crescimento pretendido. Com este crescimento, certamente, decorrências serão geradas no sistema de produção de energia elétrica, pois este se encontra presente em todos os aspectos do consumo final individual e coletivo, e também como importante fator de produção (BORENSTEIN; CAMARGO; CUNHA; SANTANA; PINTO; ARAÚJO, 1999).

O setor energético brasileiro é muito diversificado, sendo que a maior parte da produção vem de usinas hidrelétricas, devido a grande malha hídrica que o país possui. Apesar de ser uma fonte renovável, os impactos ambientais que estes causam são de tamanha dimensão. Além desta, o país possui outras fontes de energia principais como as termoelétricas e as eólicas.

A energia elétrica é fundamental para o ser humano, e devido a isto, deve-se utilizá-la de forma racional, contribuindo assim para a preservação do meio ambiente. O aumento do desperdício de energia elétrica implica obrigatoriamente no aumento da potência instalada de geração. Este aumento representa um custo elevado, tanto ambiental quanto em investimentos em equipamentos (MARTINS; ALVEAL; SANTOS, 1999).

O setor residencial é responsável por cerca de 26% do consumo total de energia elétrica no Brasil, com uma projeção de aumento deste consumo até 2022 (EPE, 2012).

Para que o usuário doméstico faça um controle efetivo do consumo de energia é necessário um monitoramento, de fácil acesso, às informações relativas ao consumo desta energia que está sendo gasta.

Uma das alternativas viáveis e que vem se tornando cada vez mais populares no mercado, é a utilização da automação residencial.

Segundo Muratori e Dal Bó (2013), a automação residencial consiste na automatização e no controle, aplicado à residência. Isto se faz mediante o uso de equipamentos interligados entre si e com os diversos ambientes da casa, na qual são capazes de seguir as instruções de um programa previamente estabelecido e com possibilidades de alterações conforme os interesses do usuário. Em consequência, essa automação permite uma maior qualidade de vida e conforto, reduzindo o trabalho doméstico, aumentando o bem-estar e a segurança, além de várias outras aplicações que podem ser continuamente atualizadas.

Ultimamente, novas tecnologias vêm surgindo para aprimorar o ensino e a obtenção de informações. Aliado a automação residencial e visando a facilidade e conforto do usuário, segundo Ciocari (2013) é possível fazer o monitoramento e controle da energia gasta nas residências utilizando como comunicação um recurso cada vez mais presente na vida das pessoas, os dispositivos móveis como *Smartphones* e *tablets*.

Estes dispositivos utilizam como sistemas operacionais predominantes o sistema *Android*, uma plataforma prática, dinâmica e robusta.

Segundo Rovadosky; Pavan; Dalbosco e Cervi (2012), a evolução da mobilidade permitiu a esses dispositivos móveis proporcionarem aos usuários a comunicação a qualquer momento e em qualquer lugar, promovendo uma maior acessibilidade o que confirma o diferencial dos *Smartphones* e *tablets*.

1.2 Objetivo

Desenvolvimento de um sistema de automação residencial que permita ao usuário monitorar o consumo de energia elétrica da casa, assim como ter informações sobre o custo desta. Além do monitoramento, o usuário poderá fazer o controle dos equipamentos elétricos residenciais por meio de um aplicativo que será criado para *Smartphones*.

1.3 Justificativa

O consumo de energia elétrica é um dos assuntos mais debatidos na atualidade dentro do contexto de sustentabilidade.

No Brasil, este consumo vem em alto crescimento ao longo dos anos. No setor residencial, um dos principais motivos deste crescimento é devido ao aumento do número de habitantes e conseqüentemente de domicílios. Segundo o Ministério de Minas e Energia (2001) a previsão do número de domicílios particulares fixos com energia elétrica que era 56,5 milhões (Censo Demográfico IBGE, 2010) passará a ser cerca de 75 milhões de unidades em 2020.

Por se tratar de um tema tão importante e devido à necessidade de sistemas acessíveis que possam auxiliar na redução deste consumo, este trabalho justifica-se por permitir ao usuário monitorar os equipamentos energéticos residenciais, obtendo informações sobre a potência consumida e o custo que esta sendo gerado por estes, em tempo real, através de um aplicativo que será criado. Além do monitoramento, o aplicativo permitirá que o usuário controle (Liga/Desliga) dos equipamentos. Toda a comunicação será feita através de dispositivos móveis, como *Smartphones* e *tablets*.

1.4 Estrutura do Trabalho

O presente trabalho é dividido em 6 capítulos principais com seus respectivos sub capítulos. A divisão ficou da seguinte forma:

- O capítulo 1 apresenta a introdução do tema abordado, assim como os problemas que deverão ser resolvidos ao final do projeto. Uma justificativa a respeito assunto foi colocada. Além disto, foi descrito também o objetivo geral.

- O capítulo 2 apresenta a revisão bibliográfica que tem como objetivo descrever as ferramentas utilizadas no projeto prático para que possa entender as características e funcionalidades do projeto prático.
- O capítulo 3 apresenta o desenvolvimento da prática, onde foram divididos em três sub tópicos de acordo com a sequência de montagem.
- O capítulo 4 apresenta os resultados obtidos, assim como as discussões sobre este e as sugestões de trabalhos futuros que podem complementar esse projeto.
- O capítulo 5 é a conclusão sobre o sistema proposto final, suas considerações e viabilidade com relação ao tema proposto.
- O capítulo 6 apresenta as referências bibliográficas, ou seja, os trabalhos já realizados que deram um suporte para o desenvolvimento deste.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo apresenta-se toda a fundamentação teórica necessária para o desenvolvimento deste projeto. Alguns conceitos e definições relevantes para o entendimento do sistema proposto são abordados. Assim, torna-se necessária uma análise dos principais tópicos envolvidos:

- *Arduino*;
- Módulo *Ethernet Shield* para *Arduino*;
- Interface de Programação *IDE*;
- Custo da energia elétrica e consumo elétrico de equipamentos residenciais;
- Sensores de corrente;
- Relés;
- Sistema Operacional *Android*;
- Plataforma *App Inventor*.

2.1 *Arduino*

O *Arduino* surgiu em 2005, na Itália, criado pelo professor Massimo Banzi, professor associado da *Interaction Design Institute Ivrea*, que desejava ensinar seus alunos um pouco de eletrônica e programação de dispositivos.

Esta plataforma pode ser usada para desenvolver objetos iterativos permitindo o controle de vários dispositivos. Projetos do *Arduino* podem ser independentes, ou podem se comunicar com o software ligado ao computador.

O *Arduino* também possui uma plataforma de programação própria chamada *IDE* (*Integrated Development Environment*) com linguagem bem simples e de fácil implementação.

Existem vários tipos de *Arduino* no mercado. Estes são diferenciados devido a algumas características como o número de entradas determinando assim o tamanho da ferramenta e a robustez de processamento.

Neste projeto é utilizado o *Arduino Mega 2560*, como mostra na figura 2.1 a seguir.

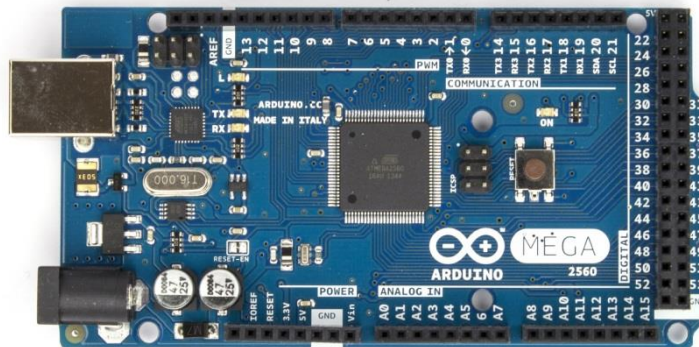


Figura 2.1 - *Arduino Mega 2560*

Fonte: *ARDUINO*, 2015.

As características da ferramenta utilizada, segundo o site oficial do *Arduino Mega 2560* (*ARDUINO*, 2015):

- ✓ Microcontrolador: ATmega2560
- ✓ Tensão de Operação: 5V
- ✓ Tensão de entrada (recomendada): 7-12V
- ✓ Tensão de entrada (limites): 6-20V
- ✓ Pinos de entrada/saída digitais: 54 (15 podem fornecer saída PWM)
- ✓ Pinos de entrada analógica: 16
- ✓ Corrente DC por pino de E/S: 40mA
- ✓ Corrente DC por pino 3.3V: 50mA
- ✓ Memória Flash: 256Kb
- ✓ SRAM: 8Kb
- ✓ EEPROM: 4Kb
- ✓ Frequência de *clock*: 16MHz

2.2 *Ethernet Shield*

Uma das vantagens do *Arduino* é a existência de vários módulos compatíveis que permitem ao usuário ampliar a capacidade do sistema ou especificar uma aplicação desejada. Os módulos são placas de circuito impresso adaptáveis que se encaixam a placa principal, onde desempenham uma função específica no sistema.

O *Ethernet Shield* (Figura 2.2) é o módulo responsável por estabelecer uma conexão entre a placa principal do *Arduino* com a Internet, além de possibilitar o acesso às informações na sua rede local através de um cabo de rede. Segundo o site oficial (*ARDUINO*,

2015), a plataforma de comunicação com a placa principal é feita utilizando o barramento SPI (Serial Peripheral Interface), através dos pinos 10, 11, 12 e 13. No pino 10 é feita a seleção do W5100, chip da WIZnet. Este fornece o protocolo TCP/IP para o *Arduino* na rede, possibilitando toda a comunicação com outro dispositivo via internet.

O TPC/IP é um conjunto de protocolos de rede que permite dois computadores, por exemplo, comuniquem entre si, ou seja, enviar e receber informações.

Para a montagem do sistema, basta encaixar o *Ethernet Shield* no *Arduino Mega 2560*, nos terminais correspondentes e ligar o cabo de rede proveniente do roteador na entrada RJ45 do módulo.



Figura 2.2 - *Shield Ethernet*

Fonte: USINAINFO, 2015.

2.3 Interface de programação

A interface de programação do *Arduino* utiliza um software *IDE* (*Integrated Development Environment*) bem simples e uma linguagem própria e simplificada.

A comunicação entre a programação e o microcontrolador é feita por meio de um cabo USB, permitindo o upload dos programas para a placa.

O software (Figura 2.3) contém um texto editor para escrita do código, uma área de mensagens, uma barra de ferramentas com botões para funções comuns e uma série de menus.

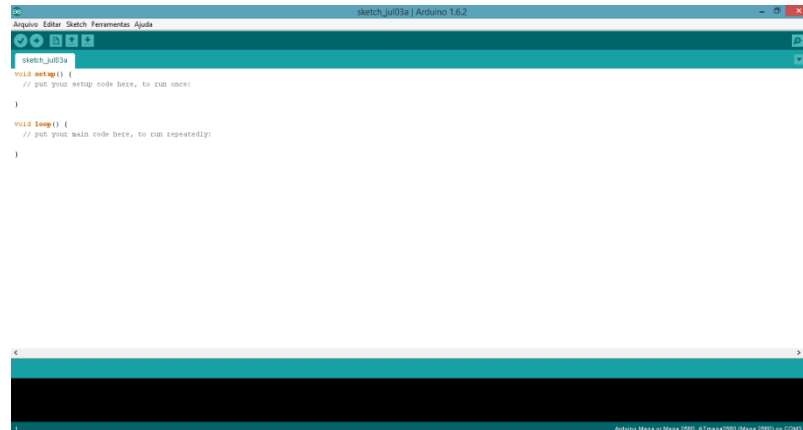


Figura 2.3 - Interface de programação *IDE*

Além de ser uma estrutura de fácil utilização, esta interface ainda possui outras características úteis como a compilação de programas, inclusão de bibliotecas dos mais diversos dispositivos compatíveis e exemplos de aplicações.

Outra importante característica é a presença de um monitoramento da comunicação serial (Serial Monitor) que permite ao usuário identificar erro nos códigos, assim como visualizar o comportamento do dispositivo programado.

2.4 Custo da energia elétrica e consumo elétrico de equipamentos residenciais

A Agência Nacional de Energia Elétrica (ANEEL) é a responsável pelo cálculo das tarifas de energia elétrica de todas as distribuidoras do país. O cálculo leva em conta os custos com a aquisição de energia elétrica, os custos relativos ao uso dos sistemas de distribuição e transmissão, as perdas técnicas e não técnicas e os encargos diversos e impostos.

A tarifa de energia é cobrada por unidade de energia (R\$/KWh). Os custos da energia elétrica são separados em grupos e subgrupos, conforme o perfil da unidade consumidora. No caso do setor residencial que utiliza baixa tensão (110 V/220 V), este pertence ao grupo B (CEMIG, 2015).

O consumo médio de energia (KWh) dos equipamentos residenciais é calculado de acordo com a equação 2.1:

$$\frac{\text{Potência do equipamento (W)} \times \text{N}^{\circ} \text{ de horas utilizadas} \times \text{N}^{\circ} \text{ de dias de uso ao mês}}{1000} \quad (2.1)$$

Para o custo mensal de energia, basta multiplicar o consumo médio em KWh pelo valor da tarifa cobrada em cada estado pela concessionária.

Na tabela 2.1 mostram-se alguns exemplos de consumo médio residencial:

Tabela 2.1- Tabela de consumo médio de equipamentos domésticos

Aparelhos Elétricos	Dias Estimados Uso/Mês	Média Utilização/Dia	Consumo Médio Mensal (KWh)
Aspirador de pó	30	20min	7,17
Chuveiro Elétrico – 5500W	30	32min	88,00
Computador	30	8h	15,12
Fogão Elétrico	30	1h	68,55
Microondas	30	20min	13,98
Geladeira 2 portas	30	24h	48,24
Lâmpada incandescente – 40W	30	5h	6,00
Lavadora de roupas	12	1h	1,76

2.5 Sensores de corrente

Os sensores de corrente são responsáveis por fornecer um nível adequado de tensão as entradas analógicas do microcontrolador que no caso do *Arduino* são 5 volts.

Atualmente existem várias técnicas para medir corrente elétrica. As mais utilizadas são resistivas, transformadores de corrente e efeito Hall.

O sensor utilizado neste projeto será o sensor de corrente linear ACS712 (Figura 2.4) fabricado pela Allegro Micro Systems, baseado em efeito Hall. Este fornece medições precisas de corrente tanto para sinais CA quanto CC. A escolha por este dispositivo se deu pela sua linearidade, um isolamento satisfatório entre a parte de potência e a parte de tratamentos de dados e um tamanho reduzido em um encapsulamento SOIC8 (ALLEGRO MICROSYSTEMS, INC., 2010).

O princípio básico do efeito Hall é a força Lorentz. Quando um elétron se move em uma direção 'v', perpendicular ao campo magnético aplicado 'B', apresenta uma força 'F' (força Lorentz) perpendicular ao sentido da corrente 'v' e ao campo magnético 'B'. Em resposta a esta força 'F', os elétrons se movem em um caminho curvo ao longo do condutor e então uma tensão surge através da placa, esta é a tensão de Hall 'V_H'. Sua polaridade é

determinada pela direção do campo magnético aplicado e é proporcional à força do campo magnético (ALLEGRO, 2015).

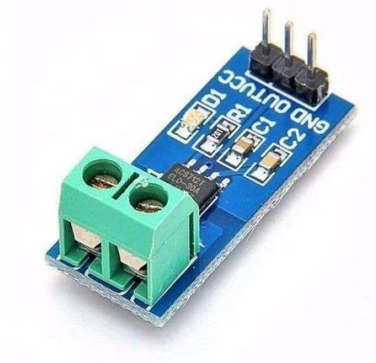


Figura 2.4 – Sensor de Corrente ACS712

Fonte: INOBOT, 2016.

2.6 Relés

Os relés são dispositivos comutadores eletromecânicos compostos por um eletroímã (bobina), uma armadura de ferro móvel, conjuntos de contatos, mola de rearme e terminais.

O funcionamento dos relés é bem simples, na qual uma corrente ao passar pela bobina, cria-se um campo magnético que atrai um ou uma série de contatos fechando ou abrindo circuitos.

O módulo relé (Figura 2.5) utilizado é o de 5V, que permite o controle de cargas como lâmpadas, motores, fechaduras e eletrodomésticos, desde que a corrente de operação não ultrapasse 10A. A placa é equipada com transistor, conector, diodo e relé. Com isto se torna mais fácil a ligação com o *Arduino*.



Figura 2.5 - Relé de 2 canais

Fonte: FILIPEFLOP, 2015.

As especificações do módulo relé podem ser encontradas consultando o seu Datasheet:

- ✓ Tensão de operação : 5 VDC
- ✓ Modelo Relé : SRD-05VDC-SL-C
- ✓ Permite controlar cargas de 220V AC
- ✓ Nível de sinal dos pinos IN1 e IN2 : 5 VDC
- ✓ Corrente de operação : 15 ~ 20 mA
- ✓ Tempo de resposta : 5 ~ 10 ms
- ✓ 4 furos de 3mm para fixação, nas extremidades da placa
- ✓ Dimensões reduzidas : 51 x 38 x 20 mm

Um dos motivos de utilização deste dispositivo é a fácil comunicação e implementação de um programa com o *Arduino*.

2.7 Sistema operacional *Android*

Atualmente, um dos sistemas operacionais mais utilizados em dispositivos móveis é o sistema operacional *Android*.

O *Android* é um sistema operacional aberto, baseado no sistema Linux. Seu funcionamento é idêntico a outros sistemas operacionais (como Windows, Mac OS, Ubuntu), cuja função é gerenciar todos os processos dos aplicativos e do hardware de um dispositivo móvel para que funcione perfeitamente.

O *Android* surgiu em 2003, na cidade de Palo Alto na Califórnia (EUA) e foi desenvolvido por Andy Rubin, Rich Miner, Nick Sears e Chris White. Uma das principais vantagens do sistema *Android* é a integração dos serviços Google, na qual esta oferece uma infinidade de aplicativos.

A escolha deste sistema operacional no projeto foi a facilidade de se criar um aplicativo e o sistema mais utilizado nos tempos de hoje.

2.8 Plataforma *App Inventor*

O *App Inventor* é uma ferramenta desenvolvida pela equipe da Google e repassada ao MIT que é o responsável para a manutenção desta. Esta é uma ferramenta para o desenvolvimento de aplicativos que possui uma interface gráfica de programação,

possibilitando ao usuário, com certa simplicidade, criar aplicações para dispositivos móveis que utilizam o sistema operacional *Android*.

A interface utiliza a biblioteca Java de código aberto Open Blocks para a criação de um ambiente visual de programação semelhante a um diagrama de blocos. A implantação de um aplicativo é feita via navegador web.

A utilização desta ferramenta requer conhecimentos básicos de lógica de programação, principalmente para aplicações com elevado grau de complexidade, embora seja intuitivo sua didática.

Durante o desenvolvimento do aplicativo, o usuário poderá testá-lo através de uma ferramenta própria da plataforma, onde simula um aparelho celular virtual com o sistema operacional *Android* ou testá-lo no celular do usuário através do aplicativo do *App Inventor* para *Android*. Quando o aplicativo (*apk*) estiver concluído, o desenvolvedor empacota-o, resultando numa aplicação pronta para ser instalada no *Smartphone*.

3 METODOLOGIA

Apresenta-se as etapas realizadas ao longo do projeto. O princípio do projeto é criar um sistema utilizando a plataforma *Arduino* e seus módulos que a partir de um aplicativo desenvolvido na plataforma online *App Inventor* tem como objetivo controlar e monitorar o consumo energético do ambiente proposto.

Para a realização deste sistema foi estudado os equipamentos e ferramentas necessárias para a construção do circuito físico e suas funcionalidades. A programação da placa responsável pela integração do sistema físico com a internet foi feita inteiramente no ambiente de programação do *Arduino* por possuir uma interface simples e prática. A ligação do entre o aplicativo e o circuito foi feita na programação do ambiente de desenvolvimento do aplicativo, que utiliza códigos em blocos. A rede entre ambos é feita através do *IP* definido para a comunicação da placa Ethernet com a web.

Os testes foram realizados no laboratório da Escola de Minas, onde o circuito foi ligado a duas lâmpadas em um protótipo de uma casa comprovando a viabilidade deste projeto.

3.1 Desenvolvimento do circuito

O objetivo deste projeto é desenvolver um sistema mais simples e eficiente possível. Para a construção do circuito, primeiramente foi definido o microcontrolador *Arduino* como a ferramenta principal, sendo ela a responsável pela ligação entre o equipamento da casa que se deseja fazer a leitura de consumo energético e a comunicação em rede com o usuário final que deseja monitorar e controlar esses equipamentos. A definição dessa peça principal foi motivada pela facilidade de interação com as demais ferramentas utilizadas e por apresentar uma linguagem simples de programação, *open source* que possibilita a integração de várias bibliotecas e os respectivos exemplos.

A partir da estrutura principal, foram pesquisados os componentes mais adequados para cada funcionalidade. Para o controle (liga/desliga) foi utilizado um módulo relé, que é basicamente um “interruptor” acionado por uma determinada tensão. Este módulo tem um dos contatos ligado diretamente na rede elétrica da casa e outro no bocal da lâmpada. Estes são ligados no terminal neutro e normalmente aberto. A comunicação com o *Arduino* é feita a partir de jumpers que interligam os pinos GND, VCC e o pino IN do módulo que se conectam ao GND, VCC (5V) e a uma porta digital da placa principal. Como a corrente fornecida pelo

Arduino não suporta o relé, um circuito com transistor, diodo e resistor foi construído, acrescentando ao circuito interno do próprio módulo que já possui transistor, o que gera um valor amplificado da corrente e protege a placa principal de qualquer dano. Esta configuração é chamada de Configuração *Darlington* (Figura 3.1), que tem como objetivo principal conseguir alta impedância de entrada e alto ganho de corrente, pois são conectados dois transistores (um que existe no módulo relé e outro fora) de maneira que o ganho de corrente do arranjo seja multiplicado.

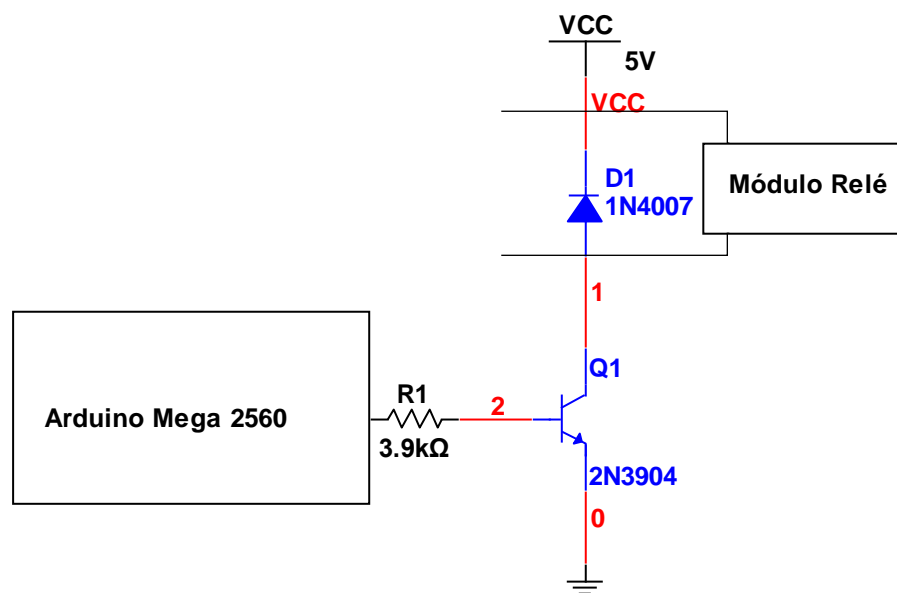


Figura 3.1 - Configuração *Darlington* do sistema

A leitura da corrente que passa nos equipamentos elétricos é realizada pelo módulo ACS 712, um sensor de corrente de efeito *Hall* linear. Este módulo tem uma ligação direta com os equipamentos e a outra com a rede elétrica da casa. A comunicação com o *Arduino* (Figura 3.2) é feita a partir de jumpers que interligam os pinos GND, VCC e o pino OUT do módulo que se conectam ao GND, VCC (5V) e analógico A0 da placa principal. Algumas características desse módulo devem ser esclarecidas antes de usá-lo. Cada sensor, dependendo da sua faixa de leitura (no caso deste projeto as leituras vão de -30A a +30A), apresenta uma sensibilidade de tensão por Ampère (no caso, 66mV/A) que deve ser levada em conta nos cálculos da corrente, de acordo com o *DataSheet* do produto.

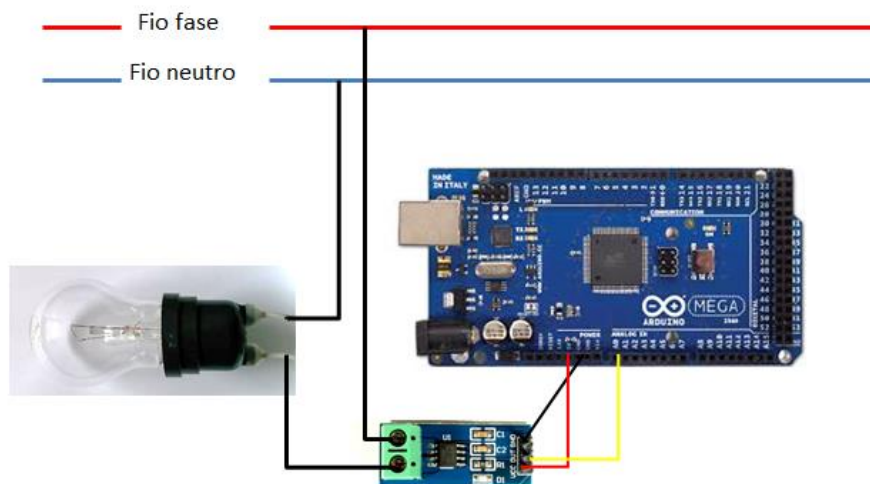


Figura 3.2 – Ligação ACS712, *Arduino* e rede elétrica

Os componentes propostos atendem perfeitamente ao objetivo do projeto. O sistema físico tem ligação em uma rede de 110V. Além desta, tem-se dois soquetes de lâmpadas ligadas a dois módulos relés e um sensor de corrente que se comunicam com a placa *Arduino* (Figura 3.3).

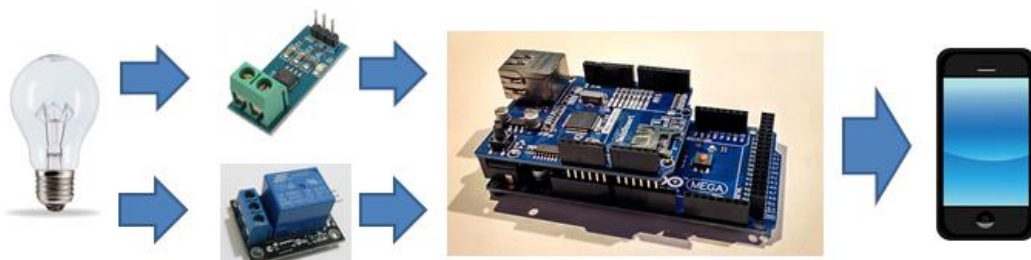


Figura 3.3 – Circuito proposto.

3.2 Desenvolvimento da Programação

A programação do circuito foi realizada no ambiente *IDE* do *Arduino* (Figura 3.4). Para integração dos relés, *shield Ethernet* e sensor de corrente foram instaladas as bibliotecas referentes a cada componente disponíveis no site oficial do *Arduino*. Estas bibliotecas contêm funções específicas de cada componente para o bom funcionamento, assim como exemplos de várias aplicações o que auxilia nesta etapa do projeto.

A comunicação com a rede web é feita por meio do módulo *Ethernet Shield* que possui o ethernet chip *Wiznet W5100* e fornece um endereço de *IP* compatível com os protocolos *TCP* e *UDP*. Estes são protocolos importantes para a base de comunicação e transmissão de dados de toda internet.

A conexão é feita através de um cabo de rede que liga o módulo ao roteador. No código de programação fornecemos os endereços da máscara e IP da rede local, e é criado um servidor *Web* que utiliza uma página de internet para fornecer dados para usuários que são acessados através de um cliente *Web*.

Para o relé, primeiro é necessário definir a porta digital que será ligada, o que definirá o estado do componente, no caso da porta digital, este só poderá assumir dois estados, nível alto e baixo, ou seja, se passa corrente ou não para ativar o equipamento. O estado do equipamento é definido como *LOW* (Desligado) ou *HIGH* (Ligado).

A programação do sensor de corrente exige a inicialização de algumas variáveis importantes que serão usados para o cálculo da corrente, potência, consumo e valor a pagar. Esse tipo de sensor trabalha com uma escala de 0 a 1023 de acordo com as variações de corrente vinda do *Arduino*, e como este trabalha em Volts, precisamos converter e ajustar o valor lido para a mesma unidade. Com isto, devemos criar uma variável que receba o valor do seguinte cálculo: $5/1023 = 0.004887586$. Outra variável lê o pino analógico A0 e ajusta o valor lido já que a saída do sensor é metade do Vcc fornecido pelo *Arduino* para corrente igual à zero. O valor lido pelo sensor é convertido para Volts com base na variável que foi feita o cálculo anteriormente. A partir desta variável, calculamos a corrente considerando a sensibilidade do sensor que foi mencionado anteriormente (66mV/A). A potência é o valor da corrente encontrado com o cálculo da sensibilidade multiplicado pela tensão na rede, que no caso é 110V. O consumo é resultado da potência calculada somado ao consumo calculado anteriormente. O preço a pagar é o consumo multiplicado pelo coeficiente definido pela concessionária de energia de cada estado, no caso de Minas Gerais, o KWh é R\$ 0,88 com adicional de impostos.



```

teste_monografia_acs712_rele_ethernet
#include <SPI.h>
// Biblioteca utilizada para comunicação com o Arduino
#include <Ethernet.h>

// A linha abaixo permite definir o endereço físico (MAC ADDRESS) da...
//placa de rede.
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };

byte ip[] = { 192, 168, 1, 178 }; //Define o endereço IP.
//byte ip[] = { 200, 239, 164, 97 }; //Define o endereço IP.

// Porta onde estará aberta para comunicação Internet e Arduino.
EthernetServer server(80);

String resdString;
const int sensorPin = A0;
float sensorValue_max = 0;
float sensorValue = 0;
float currentValue = 0;
float voltapozUnidade = 0.0048828125;
double potencia=0, potenciaseg = 0;
double consumo;
double preco;
unsigned long tempo, tempol, tempoT = millis();
int Pin = 9; // Pino digital onde será ligado e desligado o LED.
int Pin2 = 8;

boolean estado1 = false;
boolean estado2 = false;

```

Figura 3.4 – Ambiente de programação do sistema.

O código da programação pode ser visto no Apêndice A.

3.3 Desenvolvimento do aplicativo

Um dos pontos principais deste projeto é justamente fazer um sistema que seja de fácil acesso aos usuários no âmbito de controlar e monitorar o consumo energético da casa. A partir desse lema e com a rápida difusão dos dispositivos móveis na sociedade, um aplicativo com uma interface simples e funcional foi proposta.

O ambiente do *App Inventor* fornece todos os suportes necessários para se criar aplicativo, além de sua programação simples e um emulador que permite testá-lo seja com um aparelho virtual com sistema *Android* ou com o próprio aparelho físico.

Para montagem do layout (Figura 3.6) são fornecidas várias ferramentas interativas no lado esquerdo da página como botões, caixas de texto, utilização da câmera do aparelho, suportes de voz, sensores, chamadas, dentre outras que permitem fazer várias possibilidades de aplicações.

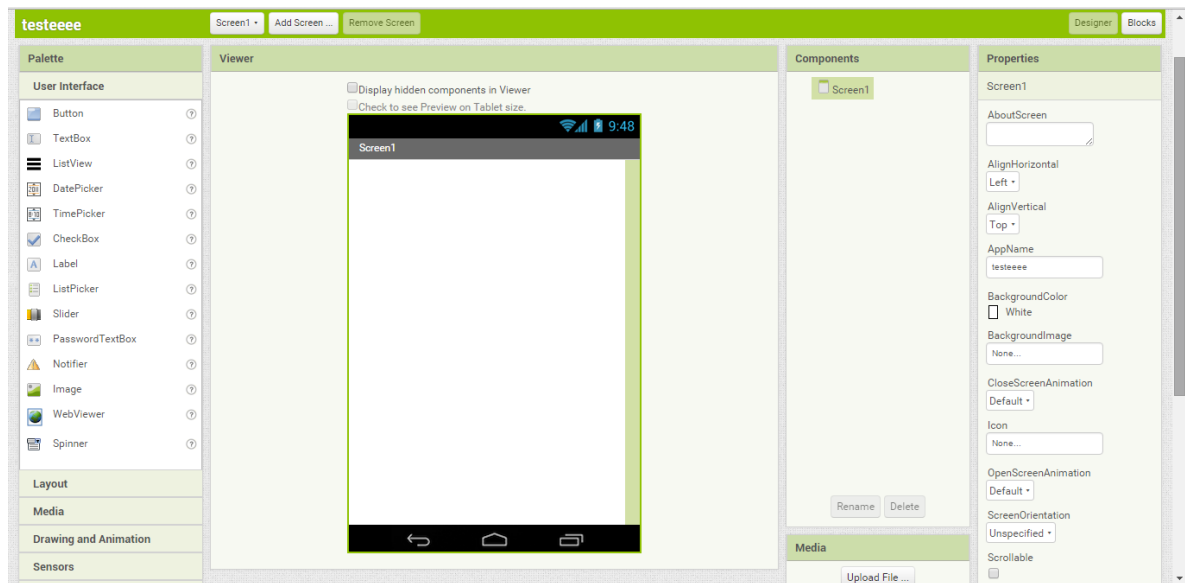


Figura 3.5 – Ambiente App Inventor.

O aplicativo “Monografia Misson” (Figura 3.6) possui duas telas, uma principal de recepção em que o usuário precisa apenas se conectar e outra tela que apresenta as funcionalidades. Essa segunda tela tem os botões de ligar e desligar para duas lâmpadas, os status de cada lâmpada (Ligada ou desligada) e os campos que fornecem o consumo e valor a pagar.



Figura 3.6 – Aplicativo “Monografia Misson”

A programação é do tipo blocos (Figura 3.7), na qual o usuário agrupa vários blocos contendo funções para montar um programa final.

A conexão entre esse aplicativo e o circuito físico é feita através do endereço *Ip* que foi inserido na programação do *Arduino* e que corresponde a rede local.

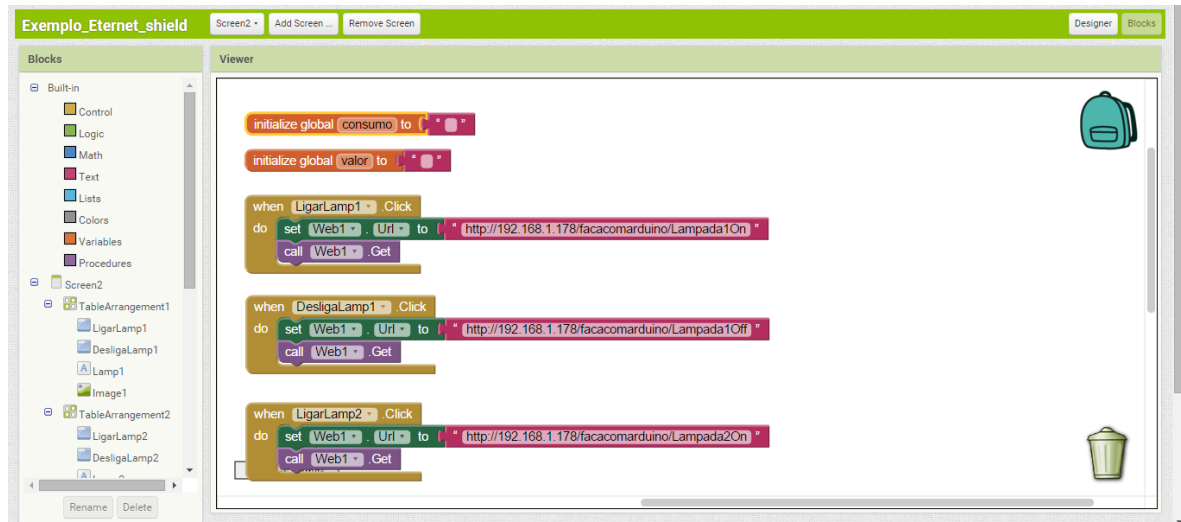


Figura 3.7 – Programação em blocos *App Inventor*.

O primeiro bloco a ser colocado é quando o usuário entra na segunda tela, onde é chamada a página web do sistema a partir do IP. Uma vez a conexão estabelecida, são feitos os arranjos dos blocos correspondentes aos botões liga/desliga que chamam os endereços web adequados a cada função.

O status de cada lâmpada é feito agrupando os blocos que contém a resposta do status fornecida pelo *Arduino* utilizando uma condição *if* para diferenciar se o equipamento está ligado ou desligado.

Os valores que aparecem nas caixas de texto correspondentes ao consumo e valor a pagar são resultado dos arranjos dos blocos que contém a resposta dessas variáveis pelo *Arduino* na página internet. Para resgatar apenas esses valores, é necessário fazer a contagem exata de onde esses caracteres estão na programação por trás da página web.

O aplicativo finalizado e testado tem-se dois modos de fazer o *download* e instalá-lo no dispositivo móvel. Uma das opções é fazer o *download* direto para o computador e passar para o aparelho ou simplesmente fornecer o código QR (*Quick Response*), um código de barras bidimensional, que resultará no aplicativo pronto para ser instalado (Figura 3.8).

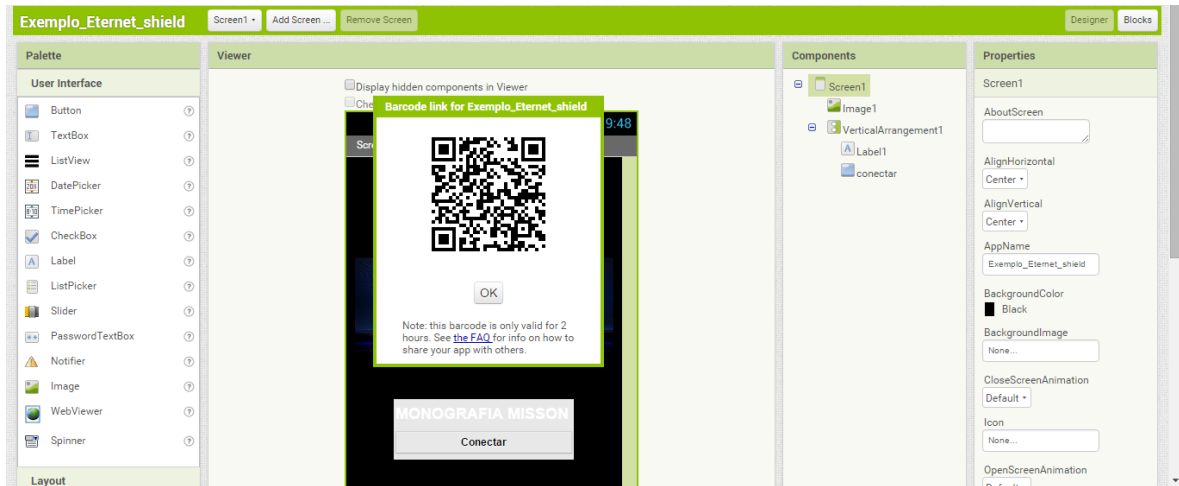


Figura 3.8 – Código QR para instalação do aplicativo no dispositivo móvel

O código da programação pode ser visto no Apêndice B.

4 RESULTADOS E DISCUSSÕES

Este capítulo tem como objetivo apresentar os resultados obtidos a partir dos testes realizados, assim como discutir as dificuldades encontradas durante o desenvolvimento do projeto.

4.1 Testes e Resultados

O sistema de controle e monitoramento energético via *Smartphone* proposto foi testado em um protótipo de uma casa para apresentar os resultados finais.

Após o circuito desenvolvido como descrito no capítulo anterior e as programações da placa principal e do aplicativo feitas e compiladas, foram realizadas as montagens descritas como se pode ver na figura 4.1.

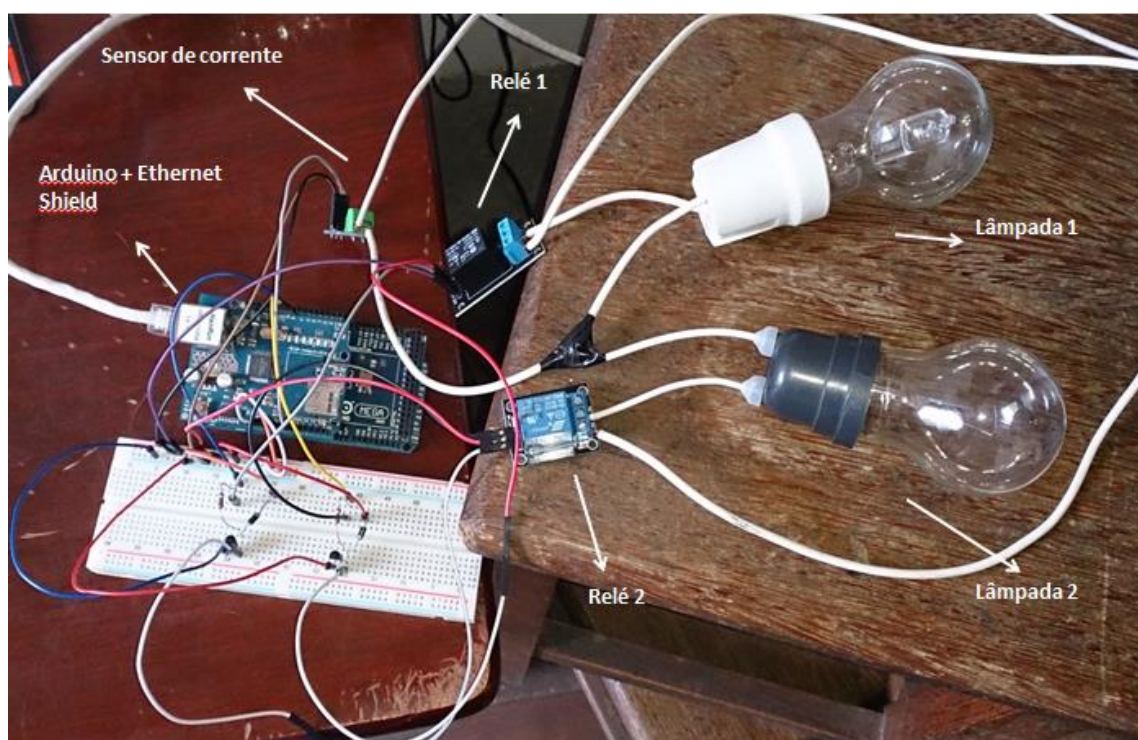


Figura 4.1 – Circuito Proposto

O controle Liga/Desliga das lâmpadas foi realizado utilizando os relés que fecham ou abrem os contatos permitindo a passagem de corrente elétrica no caminho. A configuração *Darlington* foi desenvolvida no *protoboard* utilizando um resistor de 3.9K, um transistor 2N3904 e um diodo 1N4007 para cada par de lâmpada + relé. Esta configuração permitiu amplificar a corrente fornecida pelo *Arduino* e manter o sistema funcionando por um tempo maior. As luzes acesas podem ser vistas na figura 4.2.

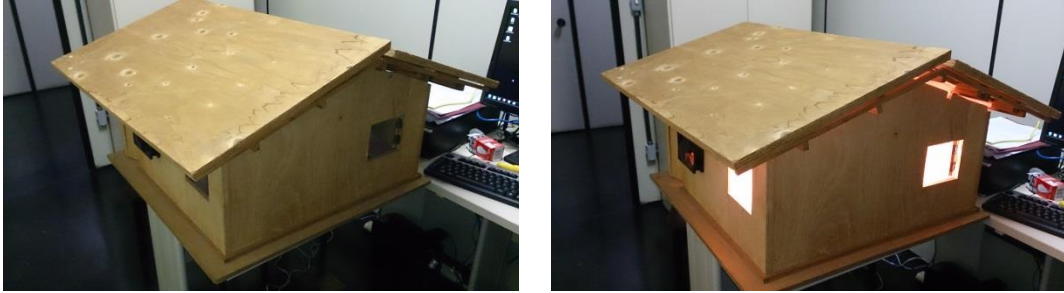


Figura 4.2 – Controle das lâmpadas

O monitoramento energético realizado quando se tem as luzes acesas no ambiente pode ser visto no aplicativo para *Smartphone* (Figura 4.3), onde há medições do consumo em KWh e do valor total a pagar de acordo com a tarifa da concessionária do estado. Estas medições são feitas em tempo real e são acumulativas, ou seja, se o aplicativo for reinicializado no dispositivo móvel, a contagem continua no momento em que ela havia parado. Os dados coletados podem ser vistos também em uma página web, onde contém além do consumo e do valor, a corrente e a potência em tempo real.



Figura 4.3 - Aplicativo em funcionamento

A partir dos testes realizados foi constatado que o projeto de controle e monitoramento energético residencial atingiu com êxito seu objetivo geral.

4.2 Dificuldades encontradas

As principais dificuldades encontradas foram em relação às programações do *Arduino* para se fazer a ligação e a transmissão dos dados com a internet e do aplicativo no *App Inventor*.

Com relação ao programa da placa principal com o *Ethernet Shield*, foram feitas pesquisas para se chegar ao código final, onde foi feito uma página web em *http* para registrar os dados em tempo real.

A maior dificuldade para se fazer o aplicativo foi na recuperação dos caracteres do consumo e valor a pagar que estavam armazenados na página internet. Para recuperá-los, foi preciso contar todas as letras e dígitos contidos na programação até achar o valor requerido.

Na realização do circuito, um problema encontrado foi para amplificação do sinal dos relés, já que no início, como a corrente que o *Arduino* fornece é pequena em relação ao que o relé exige, este desarmava rapidamente desligando o sistema. Esse problema foi solucionado com a proposta da configuração de *Darlington*.

5 CONSIDERAÇÕES FINAIS

5.1 Conclusão

Com base nos resultados obtidos, podemos concluir que é viável a construção de medidores de energia elétrica eficientes e de baixo custo.

O sistema de controle e monitoramento energético residencial via Smartphone é uma ferramenta de auxílio para que o usuário minimize o desperdício de energia elétrica no ambiente residencial.

A partir desse projeto pode-se perceber uma das aplicações que pode-se fazer no âmbito da automação residencial, uma área que vem se destacando no cenário mundial e que permite a população usufruir de seu espaço pensando no desenvolvimento sustentável. Uma das opções mais viáveis hoje em dia para se construir aplicações nesta área é o microcontrolador *Arduino* e seus módulos como foi visto nesse projeto.

5.2 Sugestões para projetos futuros

Algumas sugestões para a melhoria deste projeto são:

- Desenvolvimento do aplicativo para redes diferentes da local, como o 3G, que permita o usuário controlar a residência em qualquer local do mundo, uma vez que o projeto foi feito utilizando o celular e o sistema conectados em uma mesma rede.
- Utilização de um sensor de tensão para obter uma precisão maior no cálculo da potência.
- Utilização de um banco de dados para salvar o histórico de um determinado tempo de leitura, o que solucionaria o problema de quando o *Arduino* é desconectado, perder as informações acumuladas e reinicializar toda a contagem do consumo e preço.
- Comunicação *wireless* entre uma rede de *Arduinos* conectados a um *Arduino* central para tratamento dos dados.

REFERÊNCIAS BIBLIOGRÁFICAS

ABRADEE. Acesso em 29/05/2015. Disponível em: <http://www.abradee.com.br/setor-de-distribuicao/tarifas-de-energia/tarifas-de-energia>

ALLEGRO MICROSYSTEMS INC. Disponível em: www.allegromicro.com/~media/Files/Datasheets/ACS712-Datasheet.ashx Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor. Acesso em: 04/06/2015.

ANEEL. Acesso em: 29/05/2015. Disponível em: <http://www.aneel.gov.br/area.cfm?idArea=93&idPerfil=4&idiomaAtual=0>>.

ARDUINO. Acesso em: 25/05/2015. Disponível em: <http://www.arduino.cc/>

BEGHINI, L.B. **Automação Residencial de Baixo Custo por Meio de Dispositivos Móveis com Sistema Operacional *Android***. 2013. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) – Universidade Federal de São Carlos, São Carlos.

BORENSTEIN, C. R., CAMARGO, C. C. de B., CUNHA, C. J. C.A., SANTANA, E. A., PINTO, H. Q., ARAÚJO, J. L. R. **Regulação e gestão competitiva no setor elétrico brasileiro**. Porto Alegre. Editora Sagra-Luzzatto, 1999. V 500. 280p.

CIOCARI, L. **Controle e Monitoramento do Consumo de Energia Elétrica de Equipamentos Residenciais via *Android***. 2013. Trabalho de Conclusão de Curso (Pós-Graduação em Desenvolvimento de Produtos Eletrônicos) – Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina, Florianópolis.

CEMIG. Acesso em: 03/07/2015. Disponível em: http://www.cemig.com.br/pt-br/atendimento/Paginas/valores_de_tarifa_e_servicos.aspx

DE PAULA, G.J. **Medidor de Demanda de Energia Elétrica Residencial com Acesso Remoto**. 2013. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Centro Universitário de Brasília, Brasília.

EPE. Acesso em 29/05/2015. Disponível em <http://www.epe.gov.br/Paginas/default.aspx>

FILIFELOP. Acesso em 29/05/2015. Disponível em: <http://blog.filipeflop.com/modulos/controle-modulo-rele-Arduino.html>

FERREIRA, J.B. **Análise de Formas de Medição de Consumo de Energia Elétrica no Setor Residencial**. 2012. Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife.

INOBOT. Acesso em 20/01/2016. Disponível em: <http://www.inobot.com.br/pd-21d18a-sensor-de-corrente-ac712-30a.html>

MARCHESAN, M. **Sistema de Monitoramento Residencial Utilizando a Plataforma Arduino**. 2012. Trabalho de Conclusão de Curso (Graduação de Tecnologia em Redes de Computação) – Universidade Federal de Santa Maria, Santa Maria.

MARTINS, A.R.S., ALVEAL, C., SANTOS, E.M. et al. **Eficiência energética: integrando usos e reduzindo desperdícios**. Agência Nacional de Energia Elétrica, 1999.

MURATORI, J.R, DAL BÓ, P.H. **Automação residencial: conceitos e aplicações**. 1ª edição, Editora Educere. 70-77p. 2013.

PEREIRA, E.H.H. **Soluções inteligentes e de baixo custo para automação residencial utilizando *smatphones***. 2014. Trabalho de Conclusão de Curso (Curso de Engenharia Elétrica com ênfase em Sistemas de Energia e Automação) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.

PROCELINFO. Acesso em 29/05/2015. Disponível em:
<http://www.procelinfo.com.br/main.asp?View=%7BE6BC2A5F-E787-48AF-B485-439862B17000%7D>

ROVADOSKY, D. S. ; PAVAN, W. ; DALBOSCO, J. ; CERVI, C. R. . **Uma Ferramenta de Realidade Aumentada Usando Dispositivo Móvel com Sistema Operacional *Android***. Revista Brasileira de Computação Aplicada , v. 4, p. 25-37, 2012.

USINAINFO. Acesso em 05/05/2015. Disponível em:
<https://www.usinainfo.com.br/module/csblog/detailpost/113-81-sensor-de-gas-inflamavel-e-fumaca-mq-2-para-arduino.html?viewall>

APÊNDICE A

```

#include <SPI.h>

// Biblioteca utilizada para comunicação com o Arduino

#include <Ethernet.h>

// A linha abaixo permite definir o endereço físico (MAC ADDRESS) da placa de rede.
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };

byte ip[] = { 192, 168, 1, 178 }; //Define o endereço IP.
//byte ip[] = { 200, 239, 164, 97 }; //Define o endereço IP.

// Porta onde estará aberta para comunicação Internet e Arduino.
EthernetServer server(80);

String readString;

const int sensorPin = A0; //Sensor de corrente - Pino analógico A0

float sensorValue_aux = 0;

float sensorValue = 0;

float currentValue = 0;

float voltsporUnidade = 0.0048828125; //variável que converte 1023 em Volts

double potencia=0, potenciaseg = 0;

double consumo;

double preco;

unsigned long tempo, tempo1, tempoT = millis();

int Pin = 9; // Pino digital onde será ligado e desligado a lâmpada 1.
int Pin2 = 8; // Pino digital onde será ligado e desligado a lâmpada 2.

boolean estado1 = false; //estado da lâmpada 1 - desligado
boolean estado2 = false; //estado da lâmpada 1 - desligado

```



```
void setup(){

pinMode(Pin, OUTPUT); // Define o Pino 9 como saída.
pinMode(Pin2, OUTPUT); // Define o Pino 8 como saída.
Ethernet.begin(mac, ip); // Chama o MAC e o endereço IP da placa Ethernet.
server.begin(); // Inicia o servidor que esta inserido junto a placa Ethernet.
}

void loop(){

EthernetClient client = server.available();

if (client) {

while (client.connected()) {

if (client.available()) {

char c = client.read();

if (readString.length() < 100) {

readString += c;

}

if (c == '\n') {

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println("Refresh: 1");

client.println();

// A partir daqui começa os códigos html.

client.println(F("<HTML>"));

client.println(F("<BODY>"));
```

```
client.println(F("<H1><font color=blue>Monografia Henrique Amaral  
Misson</font></H1>"));  
  
client.println(F("<hr />"));  
  
client.println(F("<br />"));  
  
client.println(F("</BODY>"));  
  
client.println(F("</HTML>"));  
  
if(readString.indexOf("facacomArduino/Lampada1On") > 0)  
{  
    digitalWrite(Pin, HIGH); // Liga Lampada.  
    estado1 = true;  
}  
else {  
    if(readString.indexOf("facacomArduino/Lampada1Off") > 0)  
    {  
        digitalWrite(Pin, LOW); // Desliga Lampada.  
        estado1 = false;  
    }  
}  
  
if(readString.indexOf("facacomArduino/Lampada2On") > 0)  
{  
    digitalWrite(Pin2, HIGH); // Liga Lampada.  
    estado2 = true;  
}  
else {  
    if(readString.indexOf("facacomArduino/Lampada2Off") > 0)  
    {  
        digitalWrite(Pin2, LOW); // Desliga Lampada.
```

```

        estado2 = false;
    }
}

for(int i=500; i>0; i--)
{
    sensorValue_aux = (analogRead(sensorPin) -511); // le o sensor na pino analogico A0 e ajusta
o valor lido ja que a saída do sensor é vcc/2 para corrente =0

    sensorValue += (sensorValue_aux*sensorValue_aux); // soma os quadardos das leituras no
laco
}

    sensorValue = (sqrt(sensorValue/ 500)) * voltsporUnidade; // finaliza o calculo da médiaa
quadratica e ajusta o valor lido para volts

    currentValue = (sensorValue/66)*1000; // calcula a corrente considerando a sensibilidade do
sensor (66 mV por amper)

    tempo1 = tempo;
    tempo = millis();
    tempoT = tempo - tempo1;

    potencia = (currentValue * 110); //cálculo da potência para rede 110V
    potenciaseg = potencia/3600; //cálculo da potência por segundos
    consumo = consumo + (potenciaseg/1000); //cálculo do consumo
    preco = consumo * 0.88; //cálculo do preço com coef. R$ 0,88

    if(digitalRead(Pin) == LOW && digitalRead(Pin2) == LOW){
        client.println(F("<br />"));
        client.println(F("corrente : "));
        client.println(0);
        client.println(F(" A"));
        client.println(F("<b></html>"));
        client.println(F("<br />"));
        client.println(F("<br />"));
    }
}

```

```
}  
else{  
    client.println(F("<br />"));  
    client.println(F("corrente : "));  
    client.println(currentValue);  
    client.println(F(" A"));  
    client.println(F("</b></html>"));  
    client.println(F("<br />"));  
    client.println(F("<br />"));  
}  
if(digitalRead(Pin) == LOW && digitalRead(Pin2) == LOW){  
    client.println(F("potencia : "));  
    client.println(0);  
    client.println(F(" W"));  
    client.println(F("</b></html>"));  
    client.println(F("<br />"));  
    client.println(F("<br />"));  
}  
else{  
    client.println(F("potencia : "));  
    client.println(potencia);  
    client.println(F(" W"));  
    client.println(F("</b></html>"));  
    client.println(F("<br />"));  
    client.println(F("<br />"));  
}  
    client.println(F("Consumo : "));  
    client.println(consumo,4);
```

```
client.println(F(" KWh"));
client.println(F("</b></html>"));
client.println(F("<br />"));
client.println(F("<br />"));
client.println(F("Valor a pagar : "));
client.println(preco,4);
client.println(F(" (R$)"));
client.println(F("<input type='button' value='Refresh'
onClick='window.location.href=window.location.href'>"));
client.println(F("</b></html>"));
client.println(F("<br />"));
client.println(F("<br />"));
if(estado1){
    client.println(F("Lampada 1 Ligada"));
}
else{
    client.println(F("Lampada 1 Desligada"));
}
client.println(F("</b></html>"));
client.println(F("<br />"));
client.println(F("<br />"));
if(estado2){
    client.println(F("Lampada 2 Ligada"));
}
else{
    client.println(F("Lampada 2 Desligada"));
}
```

```
readString="";  
delay(1);  
  
client.stop();  
}  
}  
}  
}  
}
```

APÊDICE B

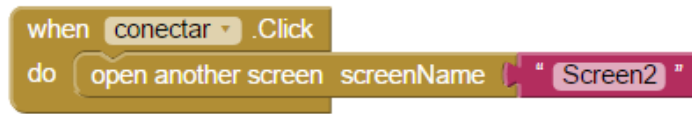


Figura B.1 – Programação da tela 1 do *App Inventor*

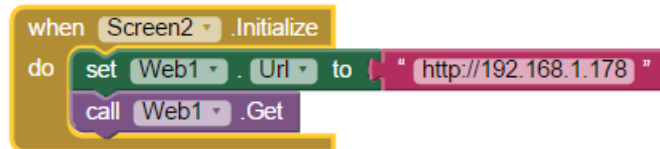


Figura B.2 – Programação da chamada do *IP* pelo aplicativo

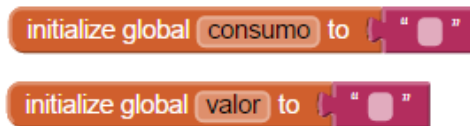


Figura B.3 – Inicialização das variáveis globais no *App Inventor*

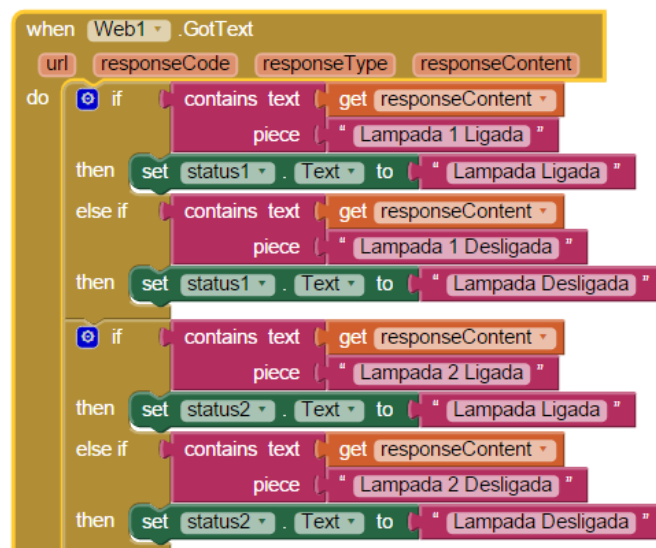


Figura B.4 – Programação do *status* das lâmpadas no aplicativo

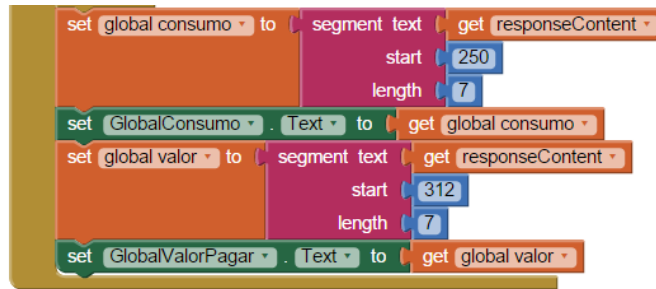


Figura B.5 – Programação dos caracteres nos campos do consumo e preço

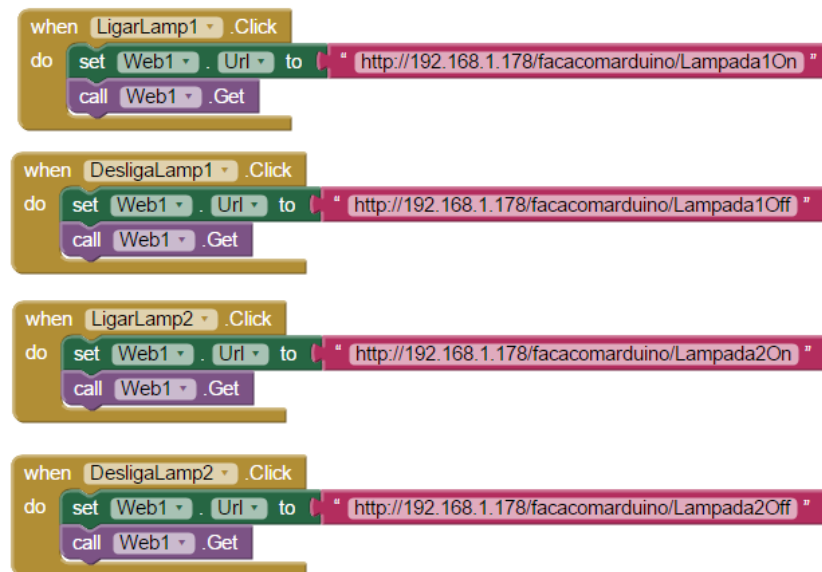


Figura B.6 – Programação dos botões liga/desligado do aplicativo

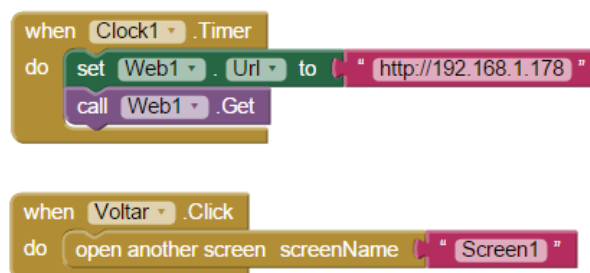


Figura B.7 – Programação do *clock* e do retorno à tela inicial do aplicativo