

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**PROTOTIPAÇÃO DE UM SISTEMA DE CONTROLE
DE ILUMINAÇÃO DE BAIXO CUSTO FOCADO EM
EFICIÊNCIA ENERGÉTICA**

TRABALHO DE CONCLUSÃO DE CURSO

Alexandre Cervi Soares

Santa Maria, RS, Brasil

2014

**PROTOTIPAÇÃO DE UM SISTEMA DE CONTROLE DE
ILUMINAÇÃO DE BAIXO CUSTO FOCADO EM EFICIÊNCIA
ENERGÉTICA**

Alexandre Cervi Soares

Trabalho de Conclusão de Curso apresentado ao Curso de Ciência da
Computação, da Universidade Federal de Santa Maria (UFSM, RS), como
requisito parcial para a obtenção do grau de
Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Benhur de Oliveira Stein

**378
Santa Maria, RS, Brasil**

2014

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**A Comissão Examinadora, abaixo assinada, aprova o
Trabalho de Conclusão de Curso**

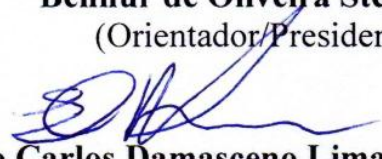
**PROTOTIPAÇÃO DE UM SISTEMA DE CONTROLE DE
ILUMINAÇÃO DE BAIXO CUSTO FOCADO EM EFICIÊNCIA
ENERGÉTICA**

elaborado por
Alexandre Cervi Soares

como requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

COMISSÃO EXAMINADORA:


Benhur de Oliveira Stein, Dr.
(Orientador/Presidente)


João Carlos Damasceno Lima, Dr. (UFSM)


Bruno Romeró de Azevedo (UFSM)

Santa Maria, 15 de julho de 2014.

RESUMO

Trabalho de Conclusão de Curso
Curso de Ciência da Computação
Universidade Federal de Santa Maria

PROTOTIPAÇÃO DE UM SISTEMA DE CONTROLE DE ILUMINAÇÃO DE BAIXO CUSTO FOCADO EM EFICIÊNCIA ENERGÉTICA

AUTOR: Alexandre Cervi Soares

ORIENTADOR: Benhur de Oliveira Stein

Data e Local da Defesa: Santa Maria, 15 de julho de 2014.

O conforto visual está diretamente relacionado à forma como percebemos o ambiente ao nosso redor, afetando a qualidade das atividades diárias. No entanto, a manutenção desse conforto está fortemente ligada ao consumo energético. É fundamental não comprometer a incidência luminosa necessária à realização de tarefas apenas com o intuito de reduzir os gastos com eletricidade. Porém, é preciso considerar que um ambiente com muitas fontes artificiais de luz consome mais energia. Nesse sentido, este projeto buscou desenvolver um meio de controle autônomo para a adequação da luminosidade de residências. O sistema proposto compreende tanto fontes luminosas naturais quanto artificiais, considerando-se que o conforto visual não é fácil de ser aferido. Para os propósitos do trabalho, foram utilizados conhecimentos e dados da Norma Brasileira de Iluminação (NBR) para Ambientes de Trabalho – NBR ISO/CIE 8995-1. A partir dela foi prototipado e implantado um hardware de caráter autônomo, para o controle de um sistema de iluminação baseado em uma política de cenários, o que implica na manipulação automática de vários conjuntos de fontes luminosas artificiais. Para isso, empregou-se a plataforma de desenvolvimento de hardware open source arduino, contemplando soluções de baixo custo. O experimento faz uso de um sensor de luminosidade para aferir a incidência luminosa no interior dos ambientes para, então, controlar a iluminação requerida por cada caso. Para a avaliação do sistema, os dados de incidência luminosa nos cenários foram comparados aos valores expressos pela NBR ISO/CIE 8995-1, sendo obtidos resultados satisfatórios. Ainda, verificou-se que a versatilidade da plataforma arduino propicia que novas iterações e melhoramentos sejam realizados, possibilitando a conexão da central de controle de iluminação com outros dispositivos.

Palavras-chave: Conforto Visual. Incidência luminosa. Controle Autônomo. Arduino. Cenários.

ABSTRACT

Conclusion Coursework
Course of Computer Science
Federal University of Santa Maria

PROTOTYPING OF A LOW COST LIGHTING CONTROL SYSTEM WITH FOCUS IN ENERGY EFFICIENCY

AUTHOR: Alexandre Cervi Soares

ADVISER: Benhur de Oliveira Stein

Date and Place of Presentation: Santa Maria, July 15, 2014.

The visual comfort is directly related with the way that we perceive the ambient around us, affecting the quality of daily activities. However, the maintenance of this comfort is tightly connected with energy consumption. It is fundamental not to compromise the needed luminosity to accomplish a given task just in order to reduce the electricity's outgoings. Nevertheless, it is necessary to consider that an ambient with many artificial light sources will consume more power. Therefore, this project searched to develop a way of autonomous control to adjust residencies lighting. The proposed system comprises both natural and artificial light sources, considering that visual comfort is not easy to be measured. For the purposes of this work, were used guidelines from Brazilian Standard Lighting (NBR) for Workplaces – NBR ISO/CIE 8995-1. As from this knowledge, an autonomous hardware to control a lighting system based on scenarios was prototyped and impacted, which implies the automatic handling of multiple sets of artificial light sources. On this project it was employed the arduino open source hardware platform, contemplating low cost solutions. The experiment uses a luminosity sensor to measure the light incidence inside the environments to, then, control the illumination on each case. For the system's evaluation, the data of light incidence on the scenarios were compared to those expressed by NBR ISO/CIE 8995-1, with the obtaining of satisfactory results. Still, it was found that the versatility of arduino platform propitiates the performing of new iterations and enhancements, allowing the connection the lighting control with other devices.

Keywords: Visual Comfort. Light Incidence. Autonomous Control. Arduino. Scenarios.

SUMÁRIO

INTRODUÇÃO	7
1 DELINEAMENTOS DO PROJETO	9
1.1 Objetivo Geral	9
1.2 Objetivos Específicos	9
1.3 Justificativa	9
1.4 Revisão de Literatura.....	10
1.4.1 Automação.....	11
1.4.2 Eficiência Energética e Conforto Visual	12
2 DESENVOLVIMENTO	14
2.1 Plataforma de hardware	14
2.1.1 Arduino.....	15
2.1.2 Hardware de controle.....	17
2.1.2.1 Comandos eletromecânicos	18
2.1.2.2 Controle de intensidade das fitas de led	20
2.2 Touchpad.....	22
2.3 Modos de Iluminação	30
2.3.1 Controle Manual	30
2.3.2 Controle Automático	31
2.3.2.1 Cenários de iluminação disponíveis	31
2.3.2.2 Sensores	32
2.4 Conectividade.....	36
2.5 Montagem das interfaces de comunicação	41
3 IMPLANTAÇÃO E METODOLOGIA	44
3.1 Implantação.....	44
3.2 Metodologia.....	46
3.2.1 Medição da incidência luminosa	48
3.2.2 Resultados das medições	50
3.2.3 Hardware para medições de iluminância locais	52
3.2.4 Pesquisa dos Materiais	54
3.2.5 Peças defeituosas ou substituídas	57
3.3 Implicações da pesquisa	57
CONCLUSÃO	59
REFERÊNCIAS	61
APÊNDICE	63

INTRODUÇÃO

A automação é utilizada em diversos setores, como fábricas, indústrias automobilísticas, hospitais, residências e sistemas de segurança. Caracteriza-se por proporcionar melhor controle e maior precisão sobre os processos executados em um dado momento, os quais podem ser de naturezas distintas e representam tarefas que devem ser realizadas. Um exemplo bastante prático de sistema automatizado é uma linha de montagem de automóveis.

No âmbito desta pesquisa, a automação se apresenta como objeto de estudo em Ciência da Computação, considerando-se que a excelência das soluções depende de softwares capazes de controlarem processos automatizados. Nesse sentido, percebe-se que há necessidade de agregar uma ampla gama de conhecimentos externos ao próprio software e que também são indispensáveis noções técnicas de outras áreas para que se possa automatizar um processo. Ainda, é preciso conhecer, com relativa precisão, o meio onde a automatização será introduzida para que seja possível a correta interpretação e coleta dos requisitos.

O foco do presente estudo está situado na automação de sistemas monitores para residências, mais especificamente na prototipação de um controle de iluminação autônomo baseado em uma política de cenários. Nos primórdios do desenvolvimento de tecnologias para residências – como, por exemplo, os primeiros aparelhos televisores –, especulava-se sobre a possibilidade de produtos que tornariam a vida do cidadão mais confortável. Entretanto, ao logo desse desenvolvimento, os sistemas de automação residencial tornaram-se conhecidos, genericamente, como algo de caráter ficcional, luxuoso ou supérfluo. Tal visão originou-se, em grande medida, devido aos exageros ficcionais concebidos em obras televisivas e cinematográficas. Ainda, considerando-se a realidade do mercado, percebe-se que muitas das soluções comerciais disponíveis têm elevado custo, fato que torna a utilização de tais sistemas proibitiva ou inviável. Um exemplo disso são os interruptores de luz com interface Bluetooth ou de toque: os modelos simples podem custar mais de R\$100,00 e soluções com opções superiores podem ultrapassar, facilmente, o valor de R\$2.000,00¹. Esse cenário revela a urgência da busca por soluções integradas, de modo a facilitar a aquisição e a instalação de soluções automatizadas em ambientes residenciais. Nesse sentido, é importante ressaltar que

¹ Dados com base em valores médios encontrados em diversas lojas físicas e de e-commerce voltadas a produtos eletrônicos.

os dispositivos autônomos possuíam natureza independente ou atuavam como sistemas menores individuais. No entanto, “[...] a ideia de conferir interoperabilidade a estes dispositivos através de uma ‘linguagem’ comum, continua crescendo” (DELGADO; GROUT; PICKING, 2006, p.1).

Com base nos conhecimentos apresentados, neste projeto serão apresentadas questões relativas ao desenvolvimento de soluções de baixo custo que viabilizem a automatização da luminosidade de uma residência. A plataforma de hardware selecionada para o experimento é o arduino, que se caracteriza por proporcionar um ambiente de desenvolvimento open source. Também serão utilizados materiais oriundos de reciclagem e outros equipamentos eletroeletrônicos.

O trabalho possui a finalidade de estudar a utilização e a aplicação de conceitos, bem como de compreender as necessidades básicas para a implantação de um protótipo de controle de iluminação. Tem-se, como meta, a partir do desenvolvimento do trabalho, também promover a divulgação e a discussão das tecnologias de hardware open source entre as comunidades acadêmica e empresarial.

1 DELINEAMENTOS DO PROJETO

Neste capítulo serão apresentados os objetivos, a justificativa e a fundamentação teórica para o trabalho. Cada ponto foi detalhado individualmente nos itens a seguir.

1.1 Objetivo Geral

Este projeto experimental tem o objetivo de desenvolver um sistema de controle autônomo de iluminação residencial, de baixo custo, que possibilite a melhor utilização de energia elétrica e interações com outros sistemas e interfaces.

1.2 Objetivos Específicos

Os estudos dos sistemas de automação são compostos por conhecimentos distintos. Assim, para alcançar o objetivo geral do trabalho e compreender as questões e tecnologias voltadas à problemática em questão, foram delimitados os seguintes objetivos específicos:

- Estudar conceitos de eficiência energética e normatização de iluminação;
- Pesquisar materiais reciclados, de baixo custo e fácil aquisição, para o desenvolvimento do sistema;
- Projetar e implantar um protótipo de uso a partir da tecnologia estudada.

1.3 Justificativa

O controle de um ambiente residencial pode exigir muito dos moradores. A disciplina individual afeta diretamente na manutenção da casa e de seus custos. Assim, a eficiência energética figura como um campo fundamental, que, muitas vezes, passa despercebido no dia

a dia. A proposta deste experimento, de implantação de um sistema monitor de controle da iluminação, que considere questões de conforto luminoso dos ambientes, por exemplo, pode ser uma estratégia de grande valia para a diminuição de gastos excessivos com energia elétrica. Isso porque a necessidade de iluminação varia conforme os diferentes tipos de trabalhos realizados. Essa concepção, de forma análoga, pode ser estendida a vários sistemas, como os de controle de aquecimento a gás e de abastecimento de água, dentre outros, priorizando o mínimo desperdício de energia. Outra questão importante é a segurança. É possível implantar um sistema monitor integrado aos dispositivos de segurança já existentes na casa para proporcionar maior controle das funcionalidades e novas políticas de segurança automatizadas.

Sistemas de computação e de comunicação estão se tornando interdependentes em muitas áreas da sociedade. As aplicações que integram os variados sistemas também se tornam cada vez mais evidentes. Essa tendência força os pesquisadores de computação a desenvolver processos inovadores e, também, a redefinir os já existentes. Em relação às residências, a integração dos sistemas autônomos é de grande importância. É necessário fornecer acesso às informações disponibilizadas pelo sistema e às suas funcionalidades, para que a usabilidade seja a mais natural possível, com procedimentos ergonomicamente satisfatórios à experiência dos consumidores. Ou seja, a logística e a praticidade das funções são essenciais para que o sistema de automação possa ser realmente eficiente. Nesse sentido, o desenvolvimento e a implantação do presente projeto visam à prototipagem de uma central de automação residencial de baixo custo, para controle de iluminação, conectada à internet, a qual possuirá modos de cena para atividades distintas e controles de iluminação automáticos.

1.4 Revisão de Literatura

Atualmente, existem inúmeras plataformas para desenvolvimento de hardware open source. Dentre as mais populares, encontram-se o arduino², o raspberry pi³ e o teensy⁴. A escolha do tipo que será utilizado e a sua aplicação, individual ou conjunta, é apoiada por uma grande comunidade de desenvolvedores que provêm suporte. Além disso, há a disponibilidade de acesso aos repositórios de software oficiais. A opção por uma plataforma, em detrimento

² Definição disponível em: <www.arduino.cc>. Acesso em: 10 mar. 2014.

³ Definição disponível em: <www.raspberrypi.org/help/what-is-a-raspberry-pi>. Acesso em: 10 mar. 2014.

de outras, pode alterar a perspectiva e adicionar ou remover barreiras no desenvolvimento do projeto. No entanto, é importante ressaltar que, por tratar-se de um experimento em open source, a portabilidade é sempre uma opção, o que possibilita a adaptação a diferentes ambientes e recursos.

A plataforma arduino é definida como “[...] uma prototipagem eletrônica open source baseada em hardware e software fáceis de utilizar. É destinado para artistas, designers, hobbistas e quaisquer pessoas interessadas em criar objetos ou ambientes interativos” (DEFINIÇÃO de Arduino, 201-, tradução minha).

1.4.1 Automação

Automatizar processos que auxiliem na execução de tarefas nem sempre é algo trivial. Devem-se considerar diversos níveis de abstração. A logística relativa à usabilidade das funções autônomas é fundamental para o aproveitamento dos recursos energéticos disponíveis. Há mais de uma possibilidade para implantar uma determinada automatização de um processo, dependendo, principalmente, das condições do local de instalação e do tipo de tecnologia de controle utilizada. Dentre elas, podem-se citar a CLPD⁵ e a FPGA⁶. A modelagem das características do ambiente, em todos os casos, é necessária para integrar as novas funcionalidades ao meio, conferindo naturalidade e fluidez aos processos.

Chama-se automação o conceito de

[...] tornar automáticas atividades repetitivas a partir do uso de sistemas e equipamentos que efetuem coleta de dados e atuam nos processos, minimizando a necessidade da interferência humana e resultando em maior velocidade nas operações, redução de erros, controle e, principalmente, em fidelidade de informações e elementos essenciais para um gerenciamento eficaz. (DEFINIÇÃO de Automação, 2013).

Ou seja, a automação define-se como a área que estuda elementos de controle de atividades, que podem ser independentes ou integradas, em uma rede de outros processos. Com relação ao conceito apresentado, o foco do presente trabalho não será o desenvolvimento

⁴ Definição disponível em: <www.pjrc.com/teensy>. Acesso em: 10 mar. 2014.

⁵ *Complex Programmable Logic Device*. Em português, “Dispositivo Lógico Complexo Programável”.

⁶ *Field Programmable Gate Array*. Em português, “Arranjo de Portas Programável em Campo”.

de interfaces de comunicação sofisticadas para os usuários. O objetivo estará na elaboração de um sistema básico, o qual proporcione meios de comunicação flexíveis.

1.4.2 Eficiência Energética e Conforto Visual

A eficiência energética diz respeito à utilização consciente e adequada dos recursos disponíveis. O objetivo é realizar tarefas consumindo o mínimo de recursos necessários, sem prejudicar a qualidade da execução. “O uso eficiente da energia apresenta-se como uma das principais dimensões de sustentabilidade a serem obtidas no espaço habitado” (LAMBERTS, 2010, p. 17).

As tarefas realizadas em uma sala multiuso requerem diferentes níveis de iluminação, a depender da tarefa realizada. De modo geral, a questão da luminosidade adequada ao ambiente não é suficientemente visível aos moradores, uma vez que os olhos se adaptam rapidamente às nuances de luz. Segundo a Norma Brasileira de Iluminação para Ambientes de Trabalho (NBR ISO/CIE 8995-1), uma boa iluminação

[...] propicia a visualização do ambiente que permite que as pessoas vejam, se movam com segurança e desempenhem tarefas visuais de maneira eficiente, precisa e segura, sem causar fadiga visual e desconforto. A iluminação pode ser natural, artificial ou uma combinação de ambas. (ABNT, 2013, p. 8)

Tendo base nessas informações, a adequação da luminosidade é o maior enfoque deste trabalho, através do desenvolvimento de um sistema monitor com funcionalidades de controle autônomas, existindo também a modalidade de controle manual. Objetiva-se, com o protótipo, a melhor adequação do conforto luminoso do ambiente. Também foram consideradas questões de eficiência a fim de diminuir o consumo de eletricidade para a iluminação dos ambientes, aproveitando a incidência luminosa natural.

O conforto visual é o principal determinante da necessidade de iluminação de um edifício. A boa iluminação deve ter direcionamento adequado e intensidade suficiente sobre o local de trabalho, bem como proporcionar boa definição de cores e ausência de ofuscamento. Os ambientes construídos (internos e externos) são iluminados para permitir o desenvolvimento de tarefas visuais (leitura, visão, manufatura, consertos, etc). (DUTRA; LAMBERTS; PEREIRA, 1997, p. 44).

Ainda, segundo a NBR ISO/CIE 8995-1,

Convém que a instalação do sistema de iluminação atenda aos requisitos de iluminação de um ambiente específico, de uma tarefa ou de uma atividade sem desperdício de energia. Entretanto, é importante não comprometer os aspectos visuais de uma instalação de iluminação simplesmente para reduzir o consumo de energia. (ABNT, 2013, p. 17).

A possibilidade de recolherem-se dados de sensores que monitoram o estado de uma residência é de grande valia para os moradores, uma vez que tais informações podem contribuir para a melhor gestão dos recursos disponíveis.

A partir dessas definições, o sistema será constituído por três abstrações principais: a) interfaces – meios de comunicação entre o usuário e a central de automação; b) central – responsável pelo processamento dos sinais recebidos pelos sensores e pelo posterior disparo de ações; c) execuções – efeitos físicos visíveis advindos das ações tomadas pela central em função de manifestações e/ou fenômenos externos capturados pelos sensores.

2 DESENVOLVIMENTO

A partir das questões relatadas anteriormente buscou-se pesquisar meios que propiciassem o desenvolvimento de um sistema, englobando software e hardware, que fosse capaz de suprir os requisitos inicialmente propostos. É importante reforçar que o experimento deste trabalho foca-se na prototipação de um sistema de controle de iluminação autônomo. O objeto do estudo é a descoberta de meios eficientes para unir recursos de baixo custo, oriundos de reciclagem, com conhecimentos das áreas de Ciência da Computação e Arquitetura. A partir disso, o intuito é o de confeccionar o protótipo com base nas funcionalidades descritas no Capítulo 1. É de suma importância ressaltar que este trabalho também possui a meta de auxiliar em futuras pesquisas da área.

2.1 Plataforma de hardware

Uma das principais questões para a criação de programas computacionais que necessitam de interações com o meio físico é o desenvolvimento do hardware específico. Isso porque ele é essencial para a execução de determinadas funcionalidades e/ou captura de dados advindos de sensores que interagem com o contexto de um ambiente de automação.

O desenvolvimento de hardware exige conhecimento técnico na área de eletrônica e programação, para que, então, seja possível uma prototipação bem sucedida. É interessante e oportuno ressaltar que, após a definição do escopo para o projeto de uma automação, deve-se considerar a necessidade de muitas iterações teórico-práticas do protótipo. Tal cenário se deve às versatilidades de tecnologias inerentes ao desenvolvimento da própria automação. O elo entre a finalidade do projeto e os recursos disponíveis varia de maneira irregular entre os diferentes contextos encontrados ou propostos em ambiente automatizado. É possível compreender o desenvolvimento de um hardware de controle para iluminação de modos distintos por meio de diferentes tecnologias. Neste trabalho, optou-se, então, pela abordagem teórico-prática.

A decisão sobre qual plataforma utilizar recai, principalmente, no contexto de prototipação inicial e definição do escopo do projeto. Cada tipo possui prós e contras que devem ser cuidadosamente medidos para que se tenha o máximo de desempenho. Ou seja,

deve-se optar por plataformas que possibilitem desenvolver aquilo que se almeja, utilizando uma quantidade de recursos, materiais ou intelectuais, de modo eficiente. Também é necessário considerar o ambiente onde a tecnologia será aplicada e questões de valores de aquisição do *software development kit* (*sdk* – kit de desenvolvimento de software), consumo energético e interfaces disponíveis. Por fim, deve-se avaliar o preço do próprio hardware de prototipação.

2.1.1 Arduino

Para este trabalho, foi escolhida a plataforma de desenvolvimento arduino, uma marca com hardware e *sdk* desenvolvidos em open source. Devido à sua flexibilidade, permite a existência de muitos modelos de hardwares similares que fazem uso da mesma *sdk* (conhecidos como compatíveis ou clones) e, portanto, apresentam funcionalidades baseadas no arduino original.

Além da tarefa de se optar por uma ou outra plataforma, dentro do universo arduino existem muitos modelos de placas de prototipação. Elas apresentam diferentes características, como: poder computacional, número de conexões, quantidade de memória RAM e interfaces. Cada modelo traz atributos diversos que podem ser úteis aos projetos.

As limitações ou recursos de hardware existentes devem ser tratados de acordo com a necessidade. Vale salientar que algumas porções do experimento são específicas para determinados recursos e tipos de componentes. Por isso, nesta seção, não serão aprofundadas questões de funcionamento do arduino, mas será realizada uma análise dos principais componentes da automação.

O arduino é uma peça de hardware que conta, basicamente, com um processador, memória, uma conexão serial para comunicação com computador, conexões de entrada e saída (analógicas e digitais) e um conversor do tipo ADC (para que as entradas analógicas possam ser processadas). Essas ligações possibilitam realizar a interface do processador com outros periféricos, sensores, hardwares customizados e diversos equipamentos e peças eletrônicas ou eletromecânicas. O real propósito do arduino, no presente experimento, é o de processar informações advindas do meio físico e, a partir delas, disparar atividades com os usuários.

O projeto do sistema de automação foi realizado sob as plataformas *arduino nano*, *arduino leonardo* e *arduino UNO* (Figura 1). Essas versões estão entre os modelos mais simples disponíveis. O padrão *nano*, como o próprio nome sugere, é uma placa de proporções modestas, idêntica ao popular *arduino UNO*. A escolha dessas placas deu-se, principalmente, pelo fato de o autor desta pesquisa já estar familiarizado com a tecnologia e também por ter o material disponível prontamente para a execução do projeto.



Figura 1 – Tipos de arduino utilizados na prototipação do sistema.

Na imagem, da esquerda para a direita, temos o *arduino UNO* (revisão 3), o *arduino nano* e o *arduino leonardo*. Os dois últimos são compatíveis, ou seja, são clones das placas fabricadas pela marca arduino, os quais funcionam com a mesma *sdk*.

A escolha pela plataforma em questão foi devida à diversa gama de peculiaridades e recursos que a torna atrativa aos objetivos propostos. Um ponto fundamental para a seleção dos modelos de arduino utilizados foi a boa relação entre o seu consumo energético e as capacidades de execução de tarefas. Como já discutido anteriormente, existem muitas abordagens para sistemas automatizados. No entanto, é necessário ponderar entre aquilo que se deseja executar e os recursos necessários para o projeto, de modo que não haja desperdícios. Outro fator que corrobora para a escolha está ligado à possibilidade de que, após a conclusão do protótipo, seja construída a versão de um hardware próprio (ou seja, um compatível/clone), baseada no arduino, com o mesmo funcionamento de *sdk* da plataforma.

A confecção de uma placa adaptada ao *sdk* do arduino é bastante simples. Em poder de tal recurso, fica claro que, após a conclusão do protótipo, pode-se desenvolver um modelo de hardware minimalista, que compreenda somente os recursos necessários para determinada tarefa, evitando gastos com energia e equipamento. O arduino oficial, propriamente, caracteriza-se por ser simples e muito versátil. No entanto, a possibilidade de desenvolver modelos para atender necessidades específicas a um contexto é muito atrativa e encaixa-se bem na proposta desenvolvida durante este projeto.

Nesse sentido, o arduino facilita imensamente a tarefa inicial de prototipação. Propicia que os entusiastas, os desenvolvedores e a comunidade em geral tenham acesso a um ambiente que interconecta software e hardware. Essa abstração é muito interessante aos olhos das diversas comunidades tecnológicas. Muitos códigos, técnicas e aplicações para diversos dispositivos são gerados em função de descomplicar os primórdios da projeção dos hardwares.

A programação utilizada pela plataforma arduino ocorre através de uma versão adaptada da linguagem C, motivo pelo qual se torna uma ferramenta de grande versatilidade. Esse fato proporciona um maior controle de software sobre o hardware.

2.1.2 Hardware de controle

Em conjunto com o arduino, utilizaram-se outros recursos de hardware para prover soluções aos requisitos identificados no escopo do protótipo. A maioria do material utilizado é oriunda de reciclagem ou de reutilização equipamentos. Portanto, os componentes empregados neste trabalho são de fácil aquisição, conforme estabelecido nos objetivos. É importante lembrar que foram pesquisados materiais e soluções de baixo custo, considerando-se que tal escolha não implicasse negativamente na qualidade e segurança do protótipo. Caso não existisse disponibilidade de materiais para o reaproveitamento de componentes, considerou-se também a compra de peças necessárias a preço acessível, com base em uma pesquisa de valores.

O acionamento dos conjuntos de lâmpadas é realizado por meio de relés, as quais substituem os tradicionais botões eletromecânicos utilizados na maioria das residências. O comando dos cenários de iluminação, a aquisição de dados dos sensores e o controle

independente de cada conjunto de luzes podem ser realizados via comandos seriais ou por touchpads.

O controle de iluminação do ambiente é, na verdade, a aplicação prática dos conhecimentos adquiridos neste projeto. Para promover o comando do circuito de lâmpadas do hardware, foram implementados dois métodos: a) Substituição dos botões eletromecânicos e b) Controle de intensidade das fitas led. Ambas as opções estão descritas a seguir.

2.1.2.1 Comandos eletromecânicos

No projeto, existem sete conjuntos de fontes luminosas, os quais estão divididos entre lâmpadas e fitas de led. Para a substituição dos botões eletromecânicos, geralmente utilizados no acionamento de cargas, no caso específico das luzes, foram utilizadas sete relés para cada um dos terminais de comando de iluminação, totalizando quatorze. Os relés também são eletromecânicos, mas o seu acionamento é realizado através da aplicação de tensão nos terminais de polarização. A Figura 2, abaixo, mostra a abordagem utilizada para realizar a interface do arduino com os relés.

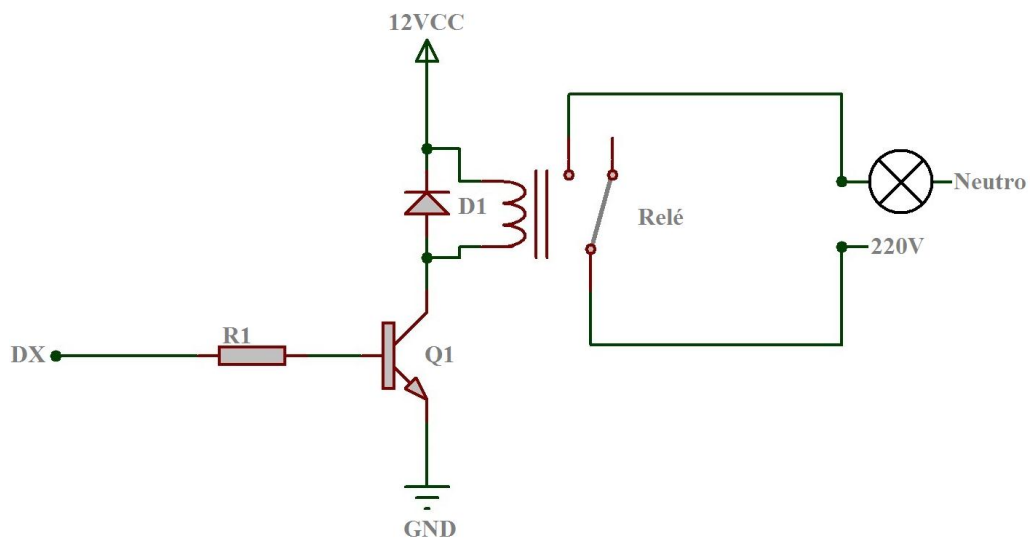


Figura 2 – Circuito para interface do relé com o arduino.

Na imagem, da esquerda para a direita, DX se refere ao pino digital onde será conectado o circuito; R1 é o resistor utilizado, sendo que, nesse caso, o seu valor é de 1k e a sua potência nominal é de 1/2watt; Q1 é um transistor BC547, de base NPN, que realiza o chaveamento do circuito dada uma tensão mínima em seu pino base; D1 é um diodo, do tipo 1N5407, que atua como polarizador para o relé.

A utilização desse circuito é necessária, pois os sete relés utilizados em cada arduino geram o consumo médio de 45ma. Multiplicando-se a corrente utilizada pelos relés pela quantidade total necessária deles, teremos um consumo de 315ma, o que gera muita corrente para o regulador de tensão do arduino. Com o circuito, a alimentação dos relés é realizada por uma fonte externa de 12vcc, independente do arduino. Cabe a ele apenas ativar os pinos digitais, como HIGH e LOW, ou seja, 0V e 5V, respectivamente, ativando ou desativando a base do transistor BC547, o qual irá permitir, ou não, a passagem de corrente, ativando ou desativando o relé. Também se deve lembrar de que o GND da fonte utilizada para energizar os relés deve ser conectado ao GND do arduino. A Figura 3 apresenta uma das ações do protótipo desenvolvido, de comando de lâmpadas do sofá, como forma de exemplificar a função de controle do circuito.

```
void mrl0sofa(){
    //mrl0 sofá
    if( flag1 == 0 ){ // verifica se o relé está desligado
        digitalWrite(3, HIGH); // liga o relé
        flag1 = 1; // seta o flag como 1
    }
    else if( flag1 == 1){ // verifica se o relé está ligado
        digitalWrite(3, LOW); // desliga o relé
        flag1 = 0; // seta o flag como 0
    }
}
```

Figura 3 – Exemplo de função que comanda o circuito do relé.

A montagem do circuito para os relés, mostrada na Figura 4 (p. 20), foi feita através da técnica de soldagem por pontos, na parte inferior do relé, conectando os componentes diretamente uns aos outros. Após a confecção de cada relé, os circuitos foram encobertos com uma resina epóxi, para fins de isolamento dos componentes, evitando a ocorrência de curto-circuito.

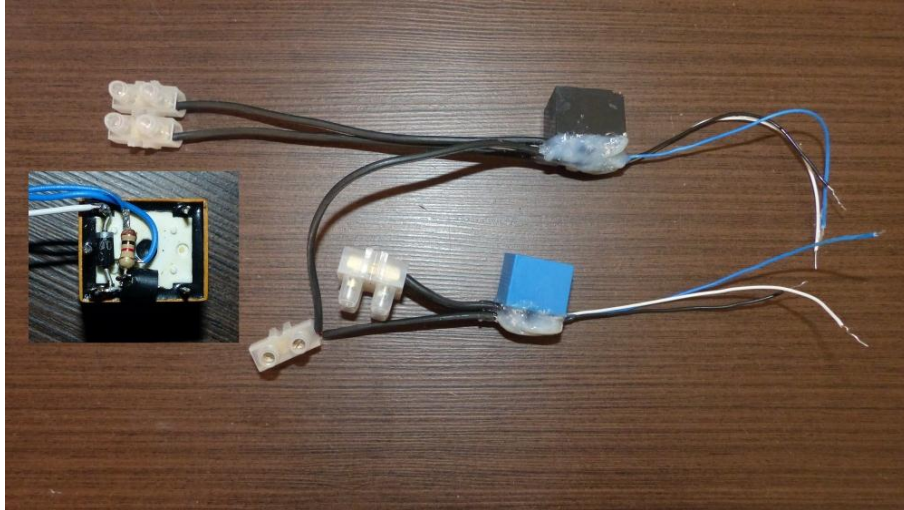


Figura 4 – Montagem do circuito de controle para os relés.

Na imagem acima, da esquerda para a direita, tem-se, primeiramente, um relé de cabeça para baixo, com circuito já montado. Ao lado, à direita, há dois relés prontos, já encobertos pela resina de isolamento.

2.1.2.2 Controle de intensidade das fitas de led

O controle de intensidade das fitas de led, utilizadas para a sanca de iluminação, foi realizado com o auxílio de dois transistores de potência TIP120, de base NPN. A interface do transistor com o arduino ocorre através do circuito apresentado na Figura 5 (p. 21). Para a montagem, foram utilizados quatro componentes: dois transistores TIP120 e dois resistores de 220ohms. Semelhante ao que ocorre com o relé, nesse circuito, o TIP120 atua regulando a quantidade de corrente que irá percorrer as fitas de led. Quanto maior a corrente, mais intensa a iluminação, e, portanto, quanto menor a corrente, menos intensa a luminosidade. Os terminais dos transistores são conectados ao arduino através de um resistor limitador de tensão de 220ohms. O circuito deve ser conectado em pinos que possuam função de modulação por largura de pulso (*pwm*), para que seja possível controlar a tensão que alimentará a base dos transistores, e, conseqüentemente, a intensidade das fitas de led.

Quando forem utilizadas correntes maiores que um ampère, é recomendável que sejam adicionados dissipadores de calor nos transistores.

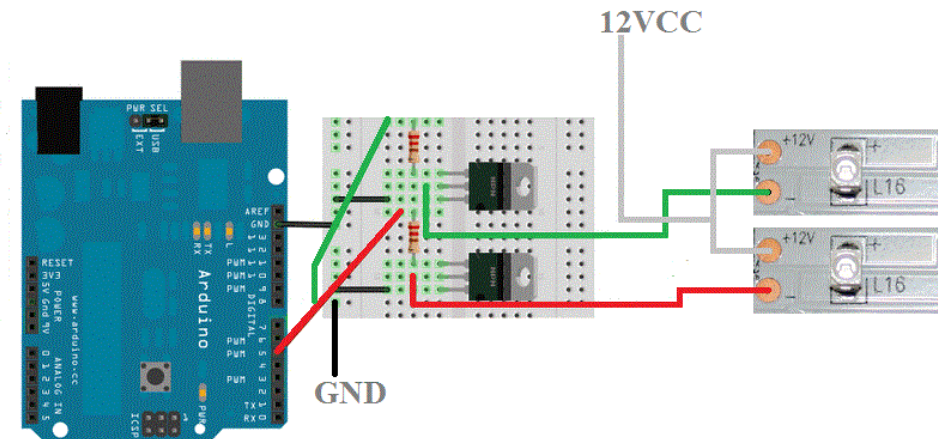


Figura 5 – Circuito para controle das fitas led.

```
void sancadown(){
  led = led - 17; // modifica o pwm de saída
  if( led < 0 ) led = 0; // se o valor já for 0, continua 0
  analogWrite(5, led); // base do transistor conectada no pino pwm 5
}

void sancaup(){
  led = led + 17; // modifica o pwm de saída
  if( led > 255 ) led = 255; // se o valor já for 255, continua 255
  analogWrite(5, led); // base do transistor conectada no pino pwm 5
}
```

Figura 6 – Função de controle das fitas led.

No código da Figura 6, a constante dezessete foi utilizada para modificar o valor da variável *led* que, então, será escrita na saída *pwm*. É possível enviar valores, que variam entre 0 e 255, através das saídas *pwm* do arduino. Partindo-se desse conhecimento, a constante dezessete é obtida através da divisão de 255 por 15, que resulta em 17. Dessa forma, existem 15 possíveis intensidades para as fitas de led. A Figura 7 (p. 22) mostra a confecção do circuito de controle para essa finalidade.

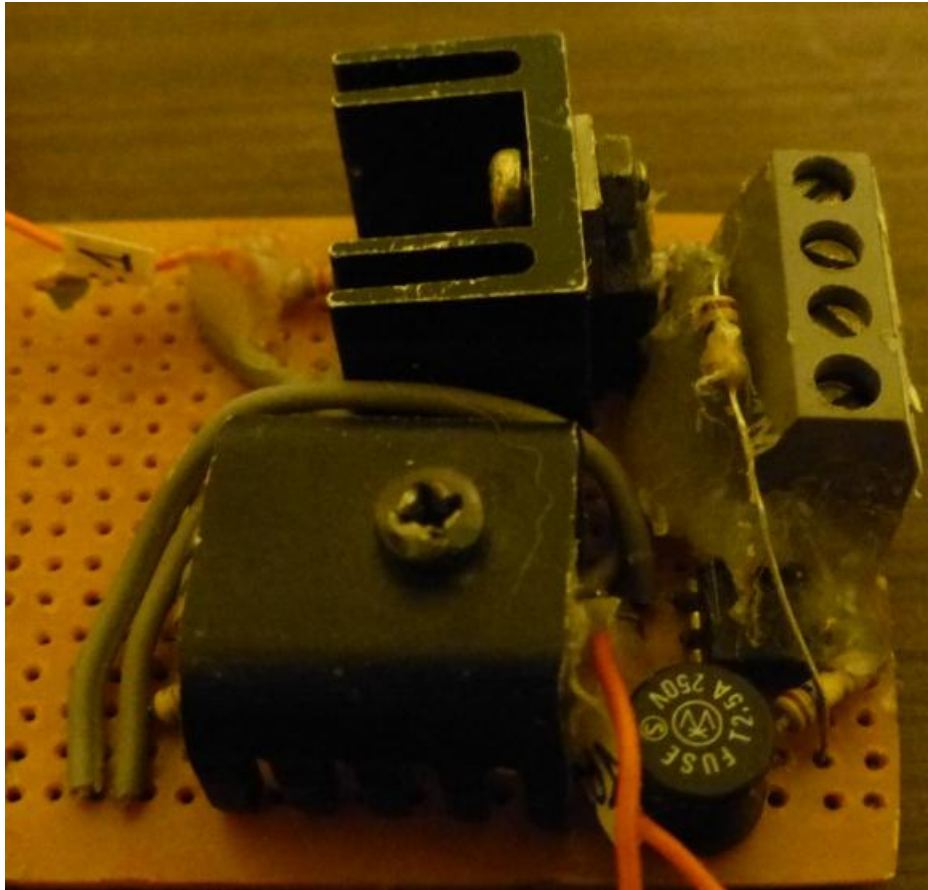


Figura 7 – Montagem do circuito de controle das fitas de led.

2.2 Touchpad

Além da possibilidade de controle do sistema por outros dispositivos, é também necessária uma interface física para comandar a iluminação do ambiente, de maneira a não depender de equipamentos intermediários. Para tal tarefa, foram utilizados dois touchpads retirados de notebooks que não estavam em uso, seja por mau funcionamento ou por não servirem mais a seus propósitos originais. Muitos dos touchpads encontrados em notebooks, desde os mais antigos aos mais novos, possuem chips controladores da marca *Synaptics*⁷.

A reutilização desses mouses deu-se através da pesquisa dos datasheets dos chips controladores. O intuito foi o de encontrar os pinos do chip correspondentes à configuração do padrão PS/2, detalhado, na página a seguir (p. 23), através da Tabela 1 e da Figura 8.

⁷ Disponível em: <<http://www.synaptics.com/>>. Acesso em: 28 mar. 2014.

Tabela 1 – Pinos do conector PS/2.

Pinos	Função
1	DATA
2	N/C
3	GND
4	VCC
5	CLOCK
6	N/C

Fonte: Adaptado do site PINOUTS.RU⁸.

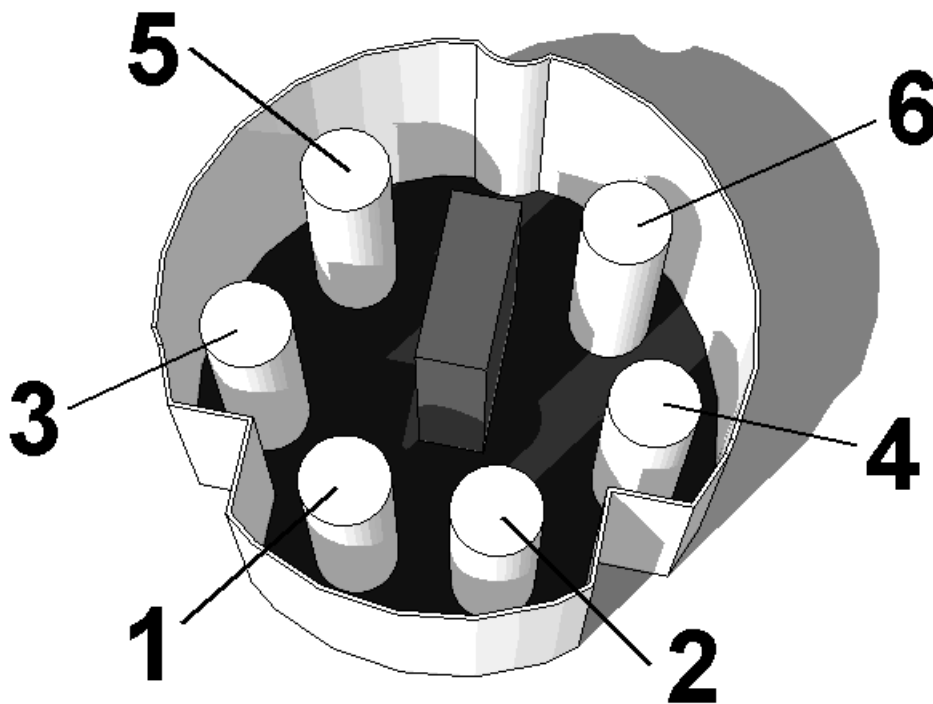


Figura 8 – Conector PS/2 macho.

Fonte: Imagem do site Commodore DTV Hacking⁹.

⁸ Disponível em: <http://pinouts.ru/Inputs/PS2Mouse_pinout.shtml>. Acesso em: 5 abr. 2014.

⁹ Disponível em: <http://dtvhacking.cbm8bit.com/common/connectors/ps2-male_640.gif>. Acesso em: 5 abr. 2014.

Os touchpads removidos dos notebooks, como dito anteriormente, são mouses. Em vista desse fato, foi utilizada a biblioteca PS2 do arduino¹⁰ para estabelecer comunicação e fazer a captura de dados. Assim, foi possível a leitura das coordenadas x, y e z. O par ordenado [x,y] corresponde ao ponto onde o touchpad foi pressionado, e a coordenada z indica a pressão atribuída a esse toque. Os valores de x e y foram utilizados para averiguar em que área houve o contato e se o local tocado compreende alguma funcionalidade implementada. Já os valores de z foram utilizados para diferenciar os toques com intenção de operação da central dos levianos.

Abaixo, na Figura 9, tem-se um fragmento de código que exemplifica a utilização de z para detectar toques com intenção de uso do sistema. Logo abaixo, na mesma imagem, está a separação em áreas do lado esquerdo do touchpad, onde o toque corresponde às coordenadas [cx,cy].

```

if (cx > 0 and mz > 10) repeatCnt++; // verifica se a pressão no touch é superior a 10
else repeatCnt = 0;
if (repeatCnt > 2){// impede a ocorrência de toques falsos
  if(dynamic == 0){
    else if ((cx > 1150 && cx < 1900)&&(cy > 1500 && cy < 2500)){
      Serial.println("Esquerda Cima");
      ar70jantar();
      delay(touch_time);
    }
    else if ((cx > 1900 && cx < 2650)&&(cy > 1500 && cy < 2500)){
      Serial.println("Esquerda Meio ");
      mrl0sofa();
      delay(touch_time);
    }
    else if ((cx > 2650 && cx < 3400)&&(cy > 1500 && cy < 2500)){
      Serial.println("Esquerda Meio l");
      sanca();
      delay(touch_time);
    }
    else if ((cx > 3400 && cx < 4150)&&(cy > 1500 && cy < 2500)){
      Serial.println("Esquerda baixo l");
      ledsmovel();
      delay(touch_time);
    }
    else if ((cx > 4150 && cx < 4900)&&(cy > 1500 && cy < 2500)){
      Serial.println("Esquerda baixo ");
      pendente();
      delay(touch_time);
    }
  }
}

```

Figura 9 – Fragmento da função que detecta toques no touchpad.

¹⁰ Disponível em: <<http://playground.arduino.cc/ComponentLib/Ps2mouse>>. Acesso em: 15 abr. 2014.

Os touchpads utilizados possuem o chip controlador *Synaptics*, modelo t1006. Além desse, existem outros chips que são comumente encontrados. O t1004 é um bom exemplo que também pode ser utilizado, sem que haja a necessidade de muitas modificações na codificação. Os pinos do chip referentes à comunicação serial PS/2 podem ser visualizados na Figura 10.



Figura 10 – Chip controlador do touchpad.

Depois da identificação dos pinos correspondentes e de ser realizada a soldagem de um cabo PS/2 em cada touchpad, eles foram conectados a um computador de mesa tradicional para verificação do funcionamento (Figura 11, p. 26).



Figura 11 – Teste de funcionamento do touchpad.

Para conectar o touchpad ao arduino, deve-se acionar a alimentação, ligar os pinos de cinco volts e o de GND do touchpad nos pinos correspondentes e, por último, conectar os fios de clock e data, separadamente, a qualquer uma das entradas digitais do arduino. Essa montagem pode ser visualizada no esquema da Figura 12.

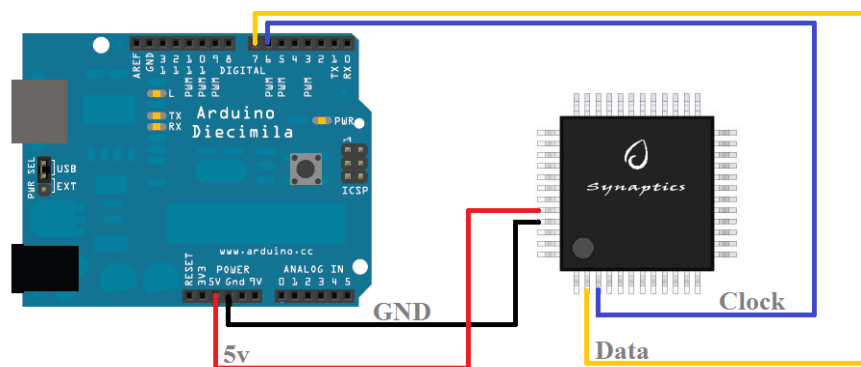


Figura 12 – Conexão do chip controlador do touchpad com o arduino.

Partindo desse ponto, é possível abrir o arquivo exemplo da biblioteca PS2, configurar, no código, as entradas digitais correspondentes a clock e data, fazer o upload do programa para o arduino e, então, iniciar qualquer console serial e pressionar o touchpad. Se a

montagem e a configuração estiverem corretas, deverão aparecer as coordenadas x e y da posição em que o touchpad foi tocado.

Os touchpads foram divididos em áreas, equivalentes a botões, de modo que cada espaço, quando pressionado, ative ou desative determinado padrão de iluminação. A seguir, na Figura 13, está um trecho do código que evidencia a divisão de áreas de toque.

```

else if ((cx > 1150 && cx < 1900)&&(cy > 1500 && cy < 2500)){
  Serial.println("Esquerda Cima");
  ar70jantar();
  delay(touch_time);
}
else if ((cx > 1900 && cx < 2650)&&(cy > 1500 && cy < 2500)){
  Serial.println("Esquerda Meio ");
  mrl0sofa();
  delay(touch_time);
}
else if ((cx > 2650 && cx < 3400)&&(cy > 1500 && cy < 2500)){
  Serial.println("Esquerda Meio l");
  sanca();
  delay(touch_time);
}
else if ((cx > 3400 && cx < 4150)&&(cy > 1500 && cy < 2500)){
  Serial.println("Esquerda baixo l");
  ledsmovel();
  delay(touch_time);
}
else if ((cx > 4150 && cx < 4900)&&(cy > 1500 && cy < 2500)){
  Serial.println("Esquerda baixo ");
  pendente();
  delay(touch_time);
}
else if ((cx > 1300 && cx < 2500) && (cy > 2500 && cy < 3700)){
  Serial.println("Meio-Cima");
  mrl0corredor();
  delay(touch_time);
}
else if ((cx > 2500 && cx < 3250) && (cy > 2500 && cy < 3700)){
  Serial.println("Meio");
  apagatudo();
  delay(touch_time);
}
}

```

Figura 13 – Fragmento de código que exemplifica a separação das áreas do touchpad pelo par ordenado [cx,cy].

Na prática, as divisões podem ser observadas através da Figura 14 (p. 28), que traz o exemplo do touchpad 1, com 12 funcionalidades, as quais estão elencadas na Tabela 2 (p. 28).



Figura 14 – Divisão de áreas do touchpad 1.

Tabela 2 – Descrição das funcionalidades do touchpad 1.

ÁREA	ACIONAMENTO
1	Lâmpadas da porta
2	Lâmpadas da entrada do escritório
3	Lâmpadas da mesa da sala de jantar
4	Lâmpadas do sofá
5	Sanca
6	Iluminação Central
7	Iluminação Externa
8	Apaga todas as lâmpadas acesas
9	Liga todas as lâmpadas
10	Salva o padrão de iluminação atual
11	Não utilizada
12	Não utilizada

Já outro exemplo, o do touchpad 2, conforme a Figura 15, foi dividido em quinze áreas.



Figura 15 – Divisão de áreas do touchpad 2.

Assim como no caso anterior, as funcionalidades do touchpad 2 foram agrupadas por área e acionamento. A descrição de cada tarefa encontra-se na Tabela 3.

Tabela 3 – Descrição das funcionalidades do touchpad 2.

(continua)

ÁREA	ACIONAMENTO
1	Iluminação da sala de jantar
2	Pendente da sala de jantar – sofá
3	Iluminação dos móveis – sanca
4	Iluminação do sofá – móveis
5	Sanca – pendente

(conclusão)	
6	Iluminação central – plafon, fita de led e sofá
7	Iluminação do corredor
8	Desliga todas as luzes
9	Liga todas as luzes
10	Aumenta a intensidade da iluminação central
11	Diminui a intensidade da iluminação central
12	Altera os cenários de iluminação
13	Salva o cenário de iluminação atual
14	Não utilizado
15	Não utilizado

2.3 Modos de Iluminação

O controle das luzes baseia-se em duas políticas principais: a) Controle Automático e b) Controle Manual. Ambos os modos estão delineados nos itens a seguir.

2.3.1 Controle Manual

Como o nome sugere, esta modalidade permite o controle dos conjuntos de lâmpadas de modo individual. É um controle semelhante ao utilizado em botões eletromecânicos. O usuário pode definir quais conjuntos de iluminação deseja acender ou apagar. Desse modo, é possível criar um cenário próprio, com combinações de luminosidade, no intuito de propiciar um ambiente suficientemente confortável e eficiente em um dado instante para uma determinada tarefa.

As opções de comando manual estão disponíveis para qualquer um dos touchpads, exemplificados na seção anterior, e compreendem os acionamentos listados pelas Tabelas 2 e 3 (p. 28 e 29).

2.3.2 Controle Automático

Quando selecionadas as opções pertencentes a esse modo, o controle do conjunto de lâmpadas aceso dependerá de dois fatores principais. São eles: a) Cenário de iluminação selecionado e b) Quantidade de luminosidade capturada pelo sensor instalado na sala. Esses aspectos estão explicados nas subseções seguintes.

2.3.2.1 Cenários de iluminação disponíveis

Salas e ambientes semelhantes se caracterizam como locais direcionados a diferentes tipos de atividades (áreas multiuso). Tendo em vista a irregularidade das possíveis tarefas realizadas, deve-se considerar também a necessidade de uma iluminação adequada para cada tarefa em questão. Os modos de iluminação pré-programados estão definidos pela Tabela 4.

Tabela 4 – Cenários de Iluminação.

Opção	Modo
1	Leitura
2	Cinema
3	Jantar
4	Suave
5	Música
6	Automático

2.3.2.2 Sensores

Para que o sistema de automação seja autossuficiente, é necessário que informações advindas do meio externo sejam coletadas e interpretadas. A requisição dos dados é realizada pelo arduino. O sensor de luminosidade retorna valores que, após o devido tratamento e formatação, serão utilizados a fim de que o sistema possa tomar decisões baseadas em uma política de regras de acionamento para os conjuntos de luzes. Ou seja, as flutuações físicas detectadas pelo sensor provocam disparos de tarefas programadas para determinada situação.

Como método para aferir o quanto de luminosidade, artificial ou natural, tem-se disponível em um determinado local, é possível utilizar ferramentas, como o luxímetro, ou por meio de cálculo luminotécnico. A conta engloba informações a respeito das quantidades e características das fontes luminosas de um ambiente para verificar a iluminância. Por outro lado, o luxímetro atua como um medidor, bastando posicioná-lo no local em que se deseja realizar a medição. Juntamente com essas ferramentas, também são utilizadas técnicas e modelos advindos de normatizações (as quais podem variar de acordo com os padrões adotados por cada país).



Figura 16 – Exemplo de instrumento utilizado para medição de incidência luminosa. Fonte: Repositório de imagens – Equipamentos Técnicos de Precisão e Científicos Ltda¹¹.

¹¹ Disponível em: <<http://www.itise.pt/images/luximetro/lux1335.jpg>>. Acesso em: 15 abr. 2014.

Como apresentado anteriormente, foi utilizada, como referência para este trabalho, a normatização da NBR ISO/CIE 8995-1 (ABNT, 2013). Ela trata, exclusivamente, de padrões de iluminação para ambientes de trabalho e técnicas para medição da iluminância nesses locais. A sua utilização no projeto será discutida mais adiante, na seção de metodologia (Capítulo 3).

Existe uma vasta gama de sensores para capturar dados de iluminância. Alguns dos exemplos mais comuns são: *ldr* (resistor variável com luz)¹², *fotodiodo* e *fototransistor*. É possível utilizar também sensores de smartphones. Ainda, pode-se optar por tipos como o bh1750fvi, que envia diretamente para o arduino a informação (já processada) sobre quantos lux (medida de iluminância) possui um determinado ambiente, sem necessidade de cálculos ou conversões.

Seguindo a proposta deste trabalho, optou-se por utilizar um *ldr*. Ele se caracteriza por ser um resistor que varia de resistência de acordo com a quantidade de luz incidente em sua superfície. É importante ressaltar que esse tipo de sensor não é muito preciso. Sua utilização é mais propícia para averiguar a existência ou não de luz, ou seja, para reportar se um ambiente está escuro ou claro. No entanto, é possível, com alguma calibração e compensação matemática, obter-se resultados suficientemente satisfatórios para os fins do projeto. A escolha desse componente deu-se pela simplicidade de interface com o arduino e a possibilidade de fácil aquisição. A peça utilizada foi reciclada de uma fotocélula defeituosa. Apesar da aparente facilidade de utilização do sensor, ele apenas fornece uma determinada tensão como saída, a qual depende diretamente da tensão de entrada e da iluminância do ambiente. Toda a interpretação das leituras obtidas e a posterior conversão dos números para a unidade lux foram realizadas via software.

A iluminância, como apontado acima, é medida em uma unidade conhecida como lux (*lx*). Existe uma fórmula de compensação matemática que relaciona a quantidade de *lx* com a resistência padrão de um *ldr* comum. O cálculo é dado pela equação 1 (Figura 17).

$$R_L = R_p / \text{Lux Kohm}$$

Figura 17 – Equação 1: Relação entre resistência nominal do *ldr* e a unidade lux.
Fonte: EMANT Pte Ltd¹³.

¹² Termo traduzido do original *Light Dependant Resistor (ldr)*.

¹³ Disponível em: <<http://www.emant.com/316002.page>>. Acesso em: 20 mar. 2014.

A variável R_p , explicitada na fórmula, refere-se ao valor de resistência padrão para um *ldr* comum, sendo medida em ohms. Para o sensor utilizado neste trabalho, constatou-se, com o auxílio de um multímetro, que o valor de resistência é de 600ohms. Portanto, a relação matemática fica estabelecida conforme a Figura 18.

$$R_L = 600 / \text{Lux Kohm}$$

Figura 18 – Equação 2: Relação entre resistência nominal do *ldr* e a unidade lux.
Fonte: EMANT Pte Ltd¹⁴.

Ainda, o arduino possui um ADC (*Analogue to Digital Converter*), ou seja, um conversor analógico-digital. Os *ldr* são sensores analógicos e, por isso, outra informação que deve ser considerada para formular uma equação que converta os dados lidos pela porta analógica do arduino, em lux, é a própria resolução disponível para as portas analógicas da plataforma. Valores de tensão analógicos não podem ser recebidos pelos pinos digitais do processador do arduino, portanto, faz-se necessário converter as tensões em números utilizáveis.

O conversor ADC do arduino possui uma resolução de 10bits, recebendo, como entrada, tensões que podem variar entre 0 e 5 volts, as quais podem ser mapeadas para valores correspondentes, que variam entre 0 e 1023. Têm-se, dessa maneira, 2¹⁰ valores, ou seja, 1024 números para avaliar a variação de entrada. Conclui-se, então, que a resolução das entradas analógicas é dada pela regra apresentada na equação da Figura 19.

$$\text{Resolução} = 5v/1024 = 0,0048828125volts$$

Figura 19 – Equação 3: Resolução das portas analógicas do arduino.

Um fator que pode prejudicar as leituras das tensões nas portas analógicas é a flutuação da própria tensão. Nesse caso, a solução adotada para incrementar a precisão da captura da leitura, evitando oscilações bruscas de tensão, foi adicionar um resistor de 10kohms entre os

¹⁴ Disponível em: <<http://www.emant.com/316002.page>>. Acesso em: 20 mar. 2014.

pinos de entrada de tensão e de leitura. Partindo dessas informações, tem-se um circuito como o exemplificado na Figura 20.

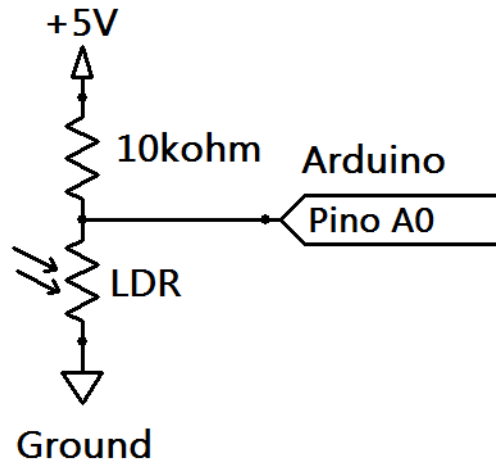


Figura 20 – Circuito do sensor de luminosidade.

Para ser possível medir os valores de incidência luminosa, substitui-se a equação 3 (Figura 19) na equação 2 (Figura 18). Na sequência, adiciona-se o valor do resistor de pullUp¹⁵. Nesse momento, pode ser gerada a fórmula da equação 4 (Figura 21).

$$\text{Lux} = \text{ResistênciaDoLdr} / (\text{pullUp} * ((\text{tensãoAplicadaNoLdr} - \text{tensãoLidaNoPinoA0}) / \text{tensãoAplicadaNoLdr}))$$

Figura 21 – Equação 4: Fórmula para o cálculo da incidência luminosa com um sensor ldr.

Através da codificação da fórmula acima, podemos criar uma função simples, porém eficiente, para medir a iluminância. Na Figura 22 (p. 35), é apresentado um fragmento de código da função de leitura para o sensor *ldr* (o circuito pode ser observado na Figura 23, p. 35).

¹⁵ Resistor que garante os níveis lógicos esperados se dispositivos externos forem desconectados.

```
int lux_meter(){  
  ldr = analogRead(pinLdr); // lê a voltagem no pino analógico  
  voltsLdr=ldr*0.0048828125; // mapeia a entrada 0->5v para 0->1024  
  return lux=100/(Resistência*((5-voltsLdr)/voltsLdr)); // calcula o Lux  
}
```

Figura 22 – Função que calcula a iluminância aferida pelo *ldr*.

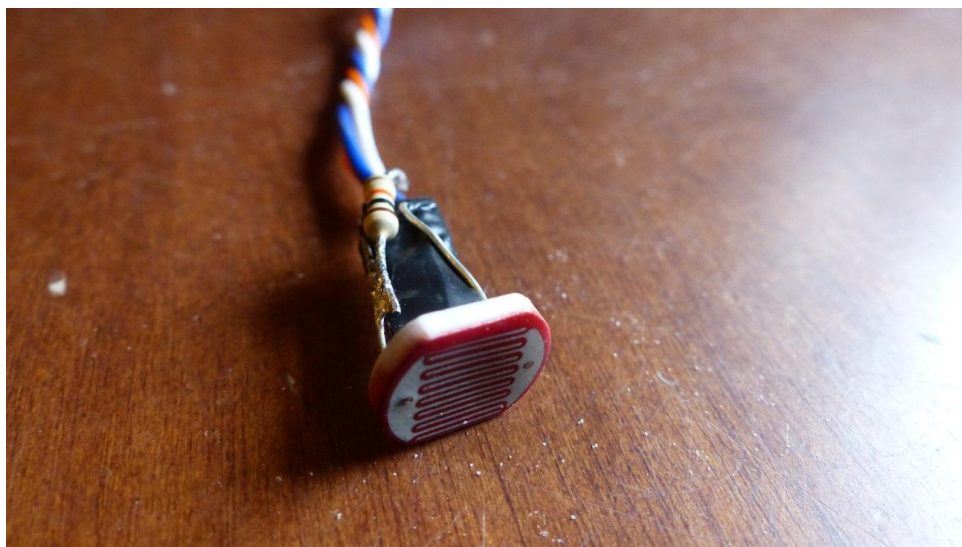


Figura 23 – Circuito do sensor *ldr*.

A cada dez segundos, uma nova verificação de incidência luminosa é realizada. Se as condições de iluminação da sala forem alteradas, deve ser realizada a manipulação dos conjuntos de fontes luminosas artificiais para que se obtenha o nível de iluminância correto ao cenário selecionado.

2.4 Conectividade

Outro fator de suma importância que não havia sido previsto, inicialmente, na prototipação, foi a necessidade de prover meios distintos de comunicação com a central de controle. Isso ocorre em vista dos diferentes padrões e tecnologias de comunicação existentes em outros dispositivos. Na primeira iteração do protótipo, estava previsto apenas o uso da

porta USB. É válido lembrar que a conexão USB do arduino com o computador ocorre através de uma interface serial que utiliza um adaptador implantado no hardware.

A comunicação do processador avr do arduino com outros dispositivos – não inclusos, os barramentos de conexões analógicas e digitais, exceto os pinos rx e tx – acontece através do padrão serial ttl, que é, basicamente, de uma conexão serial rs232. A diferenciação entre os dois padrões está no nível de hardware, mais especificamente na tensão das vias rx e tx. A maioria dos microcontroladores produzidos, atualmente, possui uma interface UART¹⁶, que é a parte do hardware responsável pela comunicação do processador com outros dispositivos via serial.

Os comandos recebidos pela central de automação são formatados como strings. Portanto, a partir do momento em que é provida a conexão com o arduino, através de um software capaz de enviar ordens por serial, em uma interface de hardware, torna-se possível operar os recursos disponibilizados pelo controle de iluminação. Na proposta do protótipo, notou-se a necessidade de expandir as vias de conexão. Para tanto, além da USB, foram estudadas e implantadas as interfaces de Bluetooth e radiofrequência. A Figura 24 mostra o compartilhamento de informações entre os terminais de comunicação e a plataforma.

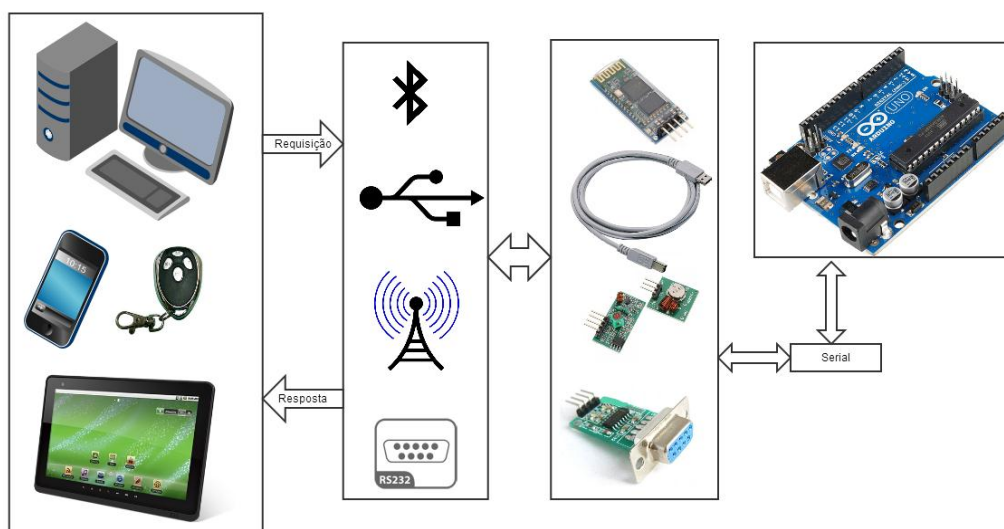


Figura 24 – Fluxo de informações entre os possíveis terminais de comando e o arduino.

¹⁶ Disponível no link: < <http://playground.arduino.cc/Referencia/Serial>>. Acesso em: 10 mar. 2014.

A Figura 24 (p. 37) apresenta, de modo simplificado, o fluxo de requisições entre os dispositivos e o controle de iluminação. Ela mostra, de forma geral, a interconexão entre os componentes que compreendem a central. Esse esboço caracteriza e contextualiza o projeto experimental desenvolvido neste trabalho. Da esquerda para a direita, têm-se exemplos de dispositivos que podem se comunicar com o controle de luzes. No centro, estão as tecnologias de comunicação disponibilizadas e, à direita, o arduino irá, então, processar as requisições. O conjunto proposto na imagem representa didaticamente a central, lembrando-se de que ao arduino serão também conectados os hardwares expostos nas Figuras 4 (p. 20), 7 (p. 22), 12 (p. 26) e 23 (p. 36). É importante salientar que todas as requisições geram uma resposta, mas que nem todas as vias de comunicação viabilizam o envio e recebimento de dados. Como exemplo dessa situação, estão os módulos Bluetooth e de radiofrequência. As suas utilizações serão discutidas mais adiante.

Como já mencionado, a comunicação com o arduino é realizada via serial. As opções de conexão apresentadas atuam como uma ampliação da própria interface. Ou seja, existe um hardware responsável pela tecnologia de comunicação em questão, seja ela sem fio ou por cabos, e, conjuntamente, há uma interface serial que faz a troca de dados com o microcontrolador. A partir desse panorama propício, nota-se que não se tem a necessidade de desenvolver um código específico para cada meio de conexão. São precisas algumas modificações no software e, então, basta apenas selecionar a interface de entrada e verificar se existem dados a serem lidos. Nessa situação, os dados são armazenados em uma variável e, posteriormente, passam por um parser, que irá interpretar se as informações lidas são válidas ou não. No arduino, os códigos desenvolvidos têm caráter autônomo, sendo que as tarefas são executadas independentemente das requisições externas (Figura 25, p. 39).

No desenvolvimento do protótipo, averiguou-se que seria interessante, e igualmente necessário, imbuir flexibilidade comunicativa, possibilitando a integração com outras tecnologias. Para tanto, foram desenvolvidos comandos que podem ser recebidos pela central (Quadro 1, p. 39), visando à execução de tarefas e à requisição de dados dos sensores. Dispositivos que possuam interface serial rs232, USB ou Bluetooth e programas que possam enviar comandos em série, no formato de strings, através de qualquer uma das interfaces, poderão se comunicar com a central. É possível, por exemplo, conectar o sistema a um computador ou a uma automação já existente, bastando desenvolver um programa, em qualquer linguagem, que possibilite a chegada de informações até as interfaces. Esse cenário também permite o uso de aparelhos como tablets e smartphones.

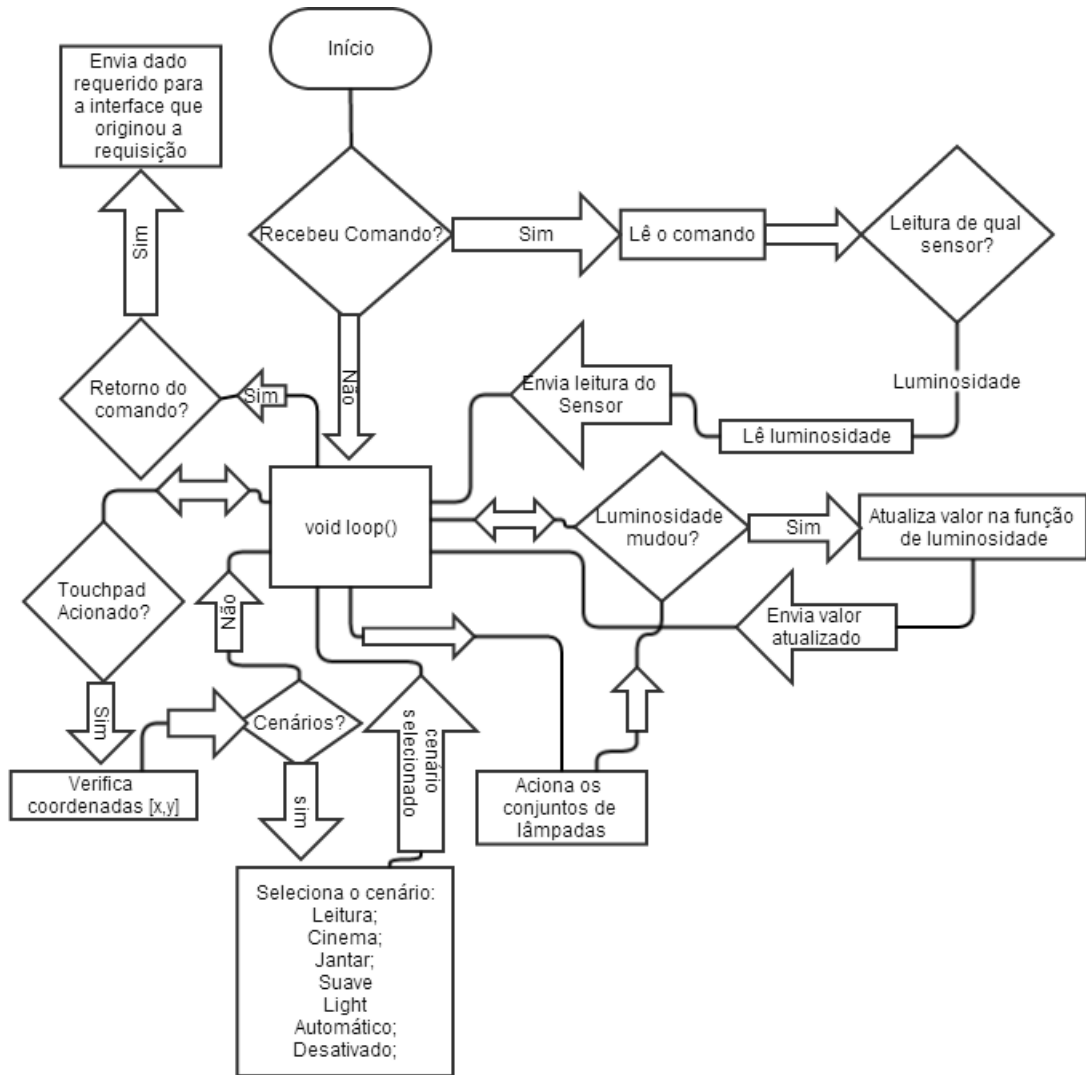


Figura 25 – Fluxograma de operação do software programado no arduino.

(continua)

Controles		Comando	Retorno
Cenários	Leitura	r	Ok
	Cinema	m	Ok
	Jantar	j	Ok
	Suave	s	Ok
	Música	v	Ok
	Automático	a	Ok
Sensores	Luminosidade	l	Valor em lux da incidência luminosa na sala

(conclusão)

Comandos individuais	Lâmpadas do corredor	b	Ok
	Lâmpadas da sala de jantar	c	Ok
	Pendente da sala de jantar	d	Ok
	Iluminação dos móveis	e	Ok
	Iluminação do sofá	f	Ok
	Iluminação central	g	Ok
	Sanca	h	Ok
	Iluminação da entrada	i	Ok
	Iluminação externa	o	Ok
	Apagar todas as luzes	x	Ok
	Acender todas as luzes	Y	Ok

Quadro 1 – Lista de comandos disponíveis para a central.

Muitos dispositivos, tais como roteadores, televisores e eletrodomésticos modernos, possuem interfaces seriais. Suas funcionalidades principais estão voltadas para outras atividades. Entretanto, é interessante pensar que podemos reprogramar tais equipamentos para adquirirem outras finalidades, além das já previstas. É possível, por exemplo, modificar o firmware de um equipamento de interface serial, para que ele se comunique com a central de iluminação, ou conectar um smartphone via Bluetooth, utilizando seus sensores para disparar eventos. De maneira semelhante, ocorre a comunicação via radiofrequência.

Após definida a interface de recebimento dos dados, o tratamento da informação ocorre como se ela fosse advinda de uma conexão serial. Essa padronização comunicativa é interessante, uma vez que podemos desenvolver o hardware e testar os comandos através de um terminal. Depois de alcançados os objetivos, torna-se possível a agregação de outros meios de comunicação, de modo que a atualização não implique na necessidade de refazer o trabalho anterior.

Para esclarecimentos, o protótipo compreende os meios de comunicação descritos acima. No entanto, não serão detalhadas suas utilizações e também não serão desenvolvidos programas para comunicação com a central por meio de tais conexões. Elas estão disponíveis

para serem utilizadas por outros dispositivos que se encaixem nas descrições detalhadas nesta seção.

2.5 Montagem das interfaces de comunicação

Com exceção da conexão USB, os demais meios necessitam de hardware externo e de pequenas modificações no software. Para a comunicação do arduino com o Bluetooth (Figura 26), foi utilizado um módulo hc-06. Nesse modelo, a ligação só é possível em uma via, ou seja, só há recebimento dados, sem a possibilidade de envio de informações. É possível, por exemplo, desenvolver um programa para smartphones que envie comandos para acender lâmpadas ou mudar os cenários de iluminação. O código que mostra o funcionamento do circuito encontra-se na Figura 27 (p. 42).

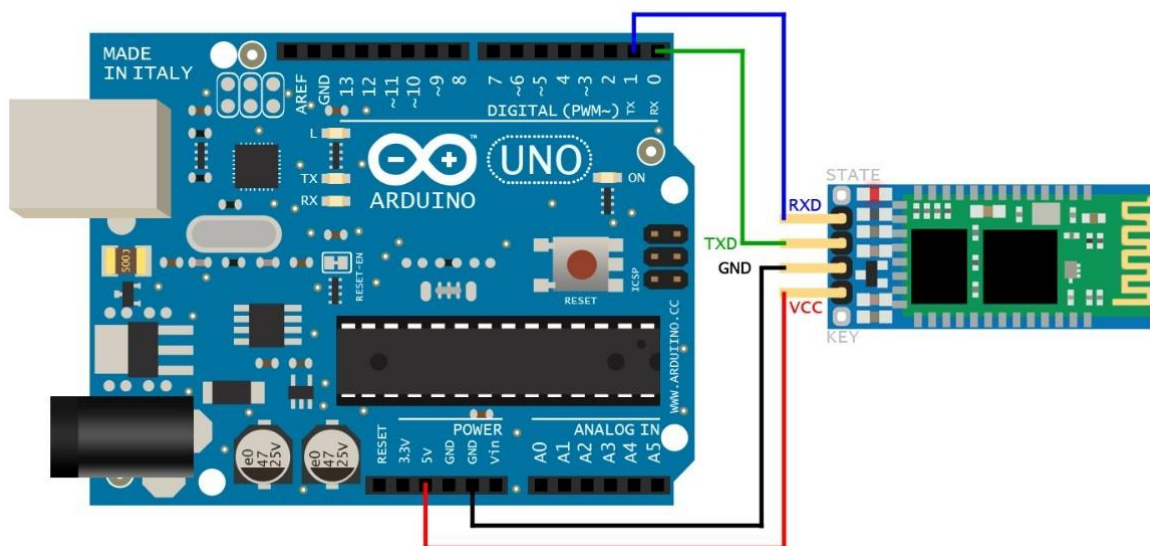


Figura 26 – Interface do módulo Bluetooth com o arduino.
Fonte: Hacktronics India¹⁷.

¹⁷ Disponível em: <http://hacktronics.co.in/images/arduino_Bluetooth_serial.jpg>. Acesso em 18 maio 2014.

```

#include <SoftwareSerial.h>
SoftwareSerial bluetooth(0, 1); //rx, tx
void setup(){
  Serial.begin(9600);
  bluetooth.begin(9600); // inicializa a porta serial setada pela biblioteca SoftwareSerial
  Serial.println("Serial pronto");
  bluetooth.println("Bluetooth pronto");
}
void loop(){
  if (bluetooth.available()){ // verifica se recebeu algum dado via bluetooth
    char a = (char)bluetooth.read(); // armazena o dado na variável c
    Serial.println(a); // envia a informação da variável c para a saída serial
  }
  if (Serial.available()) { // verifica se existem dados na serial
    char b = (char)Serial.read(); // armazena os dados lidos na variável b
    bluetooth.println(b); // exibe no terminal o conteúdo recebido
  }
}
}

```

Figura 27 – Código para exemplificar o circuito apresentado na Figura 26 (p. 41).

Para a comunicação via radiofrequência – Figuras 28 e 29 (p. 43) –, foi utilizado um par receptor e transmissor, com frequência de operação de 433.92mhz. Sua função é a de troca de informações entre os arduinos que controlam os touchpads. A aplicação desse recurso gerou menos gastos na metragem de fiação necessária para o acionamento das lâmpadas. No entanto, não foi realizado um estudo para delimitar a quantia (em R\$) exata sobre essa economia. A utilização de menos fios, tanto para a comunicação quanto para a energização das lâmpadas, também solucionou o problema de se passarem muitas vias e retornos de conexões elétricas pelos eletrodutos, uma vez que uma menor quantidade de material foi despendida.

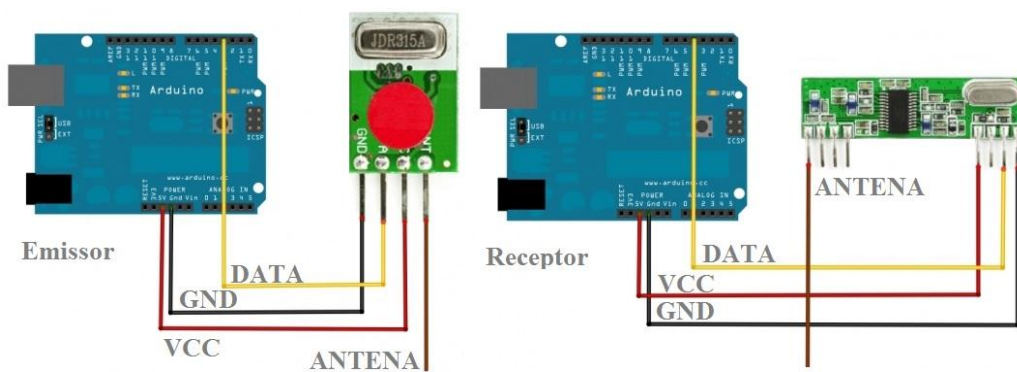


Figura 28 – Interface dos módulos rx e tx com o arduino por radiofrequência.

```

//Emissor

#include <VirtualWire.h>
void setup(){
  Serial.begin(9600);
  vw_set_ptt_inverted(true);
  vw_setup(2000); // Bits por segundo
}
void loop(){
  int aux=0;
  if(Serial.available()){
    aux = Serial.read();
    aux = aux - 48;
    if( aux == 1){
      const char *msg = "100"; // 100 = liga plafon
      vw_send((uint8_t *)msg, strlen(msg)); // envia a mensagem
      vw_wait_tx(); // espera toda a mensagem ser enviada
    }
  }
}

//Receptor

#include <VirtualWire.h>
uint8_t buf[VW_MAX_MESSAGE_LEN];
uint8_t buflen = VW_MAX_MESSAGE_LEN;
int c = 0;
void setup(){
  Serial.begin(9600);
  pinMode(4,INPUT);
  vw_set_ptt_inverted(true);
  vw_setup(2000); // Bits por segundo
  vw_rx_start(); // inicia o receptor
}
void loop(){
  if (vw_get_message(buf, &buflen)){ // testa se existem dados recebidos no buffer
    c = analogRead(4);
    if(buflen>=3){
      c = (buf[0]-48)*100 + (buf[1]-48)*10 + buf[2]-48;
      if ( c == 100 ) plafon(); // liga ou desliga a iluminação central
    }
  }
}

```

Figura 29 – Exemplo de código para envio e recebimento de uma string entre dois arduinos via radiofrequência.

É visível, tanto no código exemplificado na Figura 27 quanto no da Figura 29, que poucas linhas de software são suficientes para prover comunicação através dos hardwares expostos nas figuras 26 e 28. Em ambos são realizadas verificações de conteúdo nos canais de comunicação. Se houver informação a ser lida, ela é armazenada em uma variável e, então, estará pronta para ser utilizada pelo parser da central de automação. Esses códigos são exemplos de como ocorre a comunicação, pois suas utilizações na central de controle de iluminação são idênticas ao exposto nos exemplos.

Para o Bluetooth, o nome padrão do dispositivo é *linvor*, com senha 0000. Depois de conectado, basta abrir qualquer software que possibilite o envio de strings via Bluetooth para o comando de recursos da central. Para isso, podem-se utilizar emuladores de terminal, smartphones, dentre outros aparelhos.

Para a comunicação via radiofrequência, é apenas necessário que os dois arduinos estejam ligados e sejam enviados comandos da unidade emissora para a receptora. As duas peças de hardware utilizadas neste projeto foram adquiridas através de leilão no site *eBay*¹⁸. O valor do módulo Bluetooth hc-06 foi de US\$ 4,47 por unidade, e o do par receptor e transmissor foi de US\$0,99.

¹⁸ Disponível em: < <http://www.ebay.com/>>. Acesso em: 27 fev. 2014.

3 IMPLANTAÇÃO E METODOLOGIA

Esta seção apresenta a forma como o protótipo foi colocado em atividade e a metodologia aplicada para o desenvolvimento do trabalho.

3.1 Implantação

O processo de implantação do hardware de controle ocorreu após o término da prototipação. Essa tarefa durou um período que compreende oito dias, com a realização de diversas atividades. A listagem de ações do trabalho está apresentada abaixo, no Quadro 2.

Tarefa
Preparação física do ambiente
Passagem de cabeamento e fiações em geral
Posicionamento dos sensores
Montagem final do hardware
Testes preliminares

Quadro 2 – Cronograma das atividades de implantação.

Os três primeiros dias são referentes à preparação do local. Eles compreenderam atividades como quebra de paredes, instalação de caixas de luz e passagem de fiação (Figura 30, p. 45). No quarto dia, foram realizadas as colocações finais de cabeamentos que ainda não haviam sido concluídas e a identificação individual da fiação através de rótulos. A categorização foi realizada tanto para facilitar o trabalho de instalação do hardware quanto para referências futuras e manutenções. No quinto dia, foi instalado o sensor de luminosidade em sua posição final. Essa tarefa demorou mais do que o previsto, uma vez que foram realizadas algumas iterações experimentais a respeito da melhor localidade para o sensor. Nos dois dias seguintes, foi realizada a montagem final do hardware, que compreendeu as

instalações das seguintes peças: fontes de alimentação para os arduinos, conjuntos de relés (Figura 31, p. 46), touchpads (Figura 32, p. 46), terminais e conectores para os cabos. No último dia, foram realizados testes de funcionalidade preliminares. Salienta-se que, para essa fase de aplicação, todo o sistema havia sido testado previamente, durante a prototipação, e os eventuais problemas com componentes defeituosos já haviam sido solucionados. No entanto, identificaram-se algumas falhas do hardware, um acontecimento que já era esperado. O obstáculo identificado nos testes básicos de funcionalidade foi causado por falhas no cabeamento. Então, depois de serem reparados os conectores e os cabos deficientes, o sistema respondeu corretamente aos comandos.

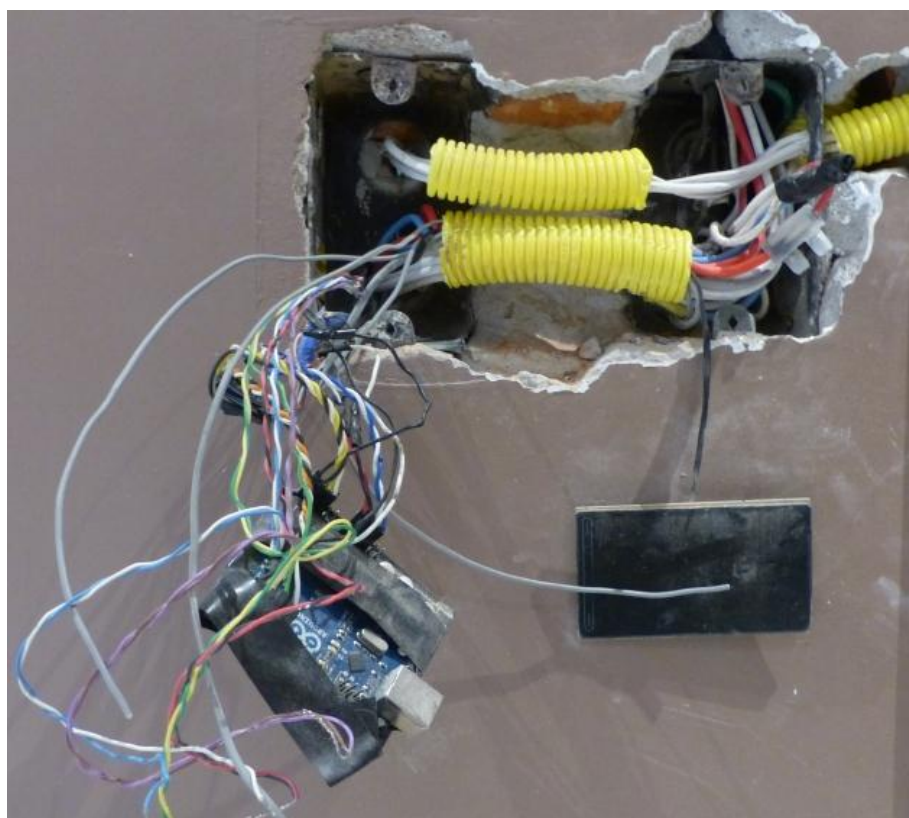


Figura 30 – Modificação de parede e passagem de cabeamento

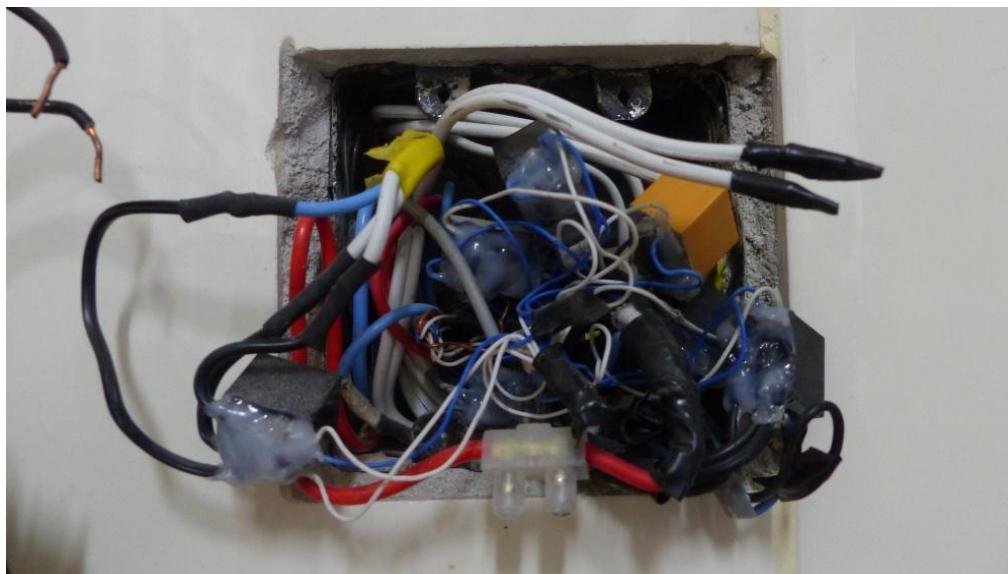


Figura 31 – Instalação e acondicionamento dos relés.



Figura 32 – Instalação dos touchpads.

3.2 Metodologia

O foco principal deste trabalho é a prototipação de um controle inteligente de luminosidade para um ambiente residencial. No caso, trabalhou-se com base em uma sala de

estar de, aproximadamente, 41m². Para dar base ao projeto, estudaram-se conteúdos da área de Arquitetura referentes à luminosidade, conforto visual e eficiência energética. Quanto ao protótipo de automação, esse foi desenvolvido de forma flexível, visando a posteriores complementos em iterações futuras. Durante a prototipação, foram contemplados recursos de software e hardware com o intuito de prover controle dinâmico para a iluminação de um ambiente. Ressalta-se que, dentro do cronograma inicialmente proposto, todas as atividades foram concluídas satisfatoriamente.

É importante frisar que este trabalho não trata propriamente de questões de projeto de iluminação. O objetivo foi avaliar a incidência luminosa na área de realização das tarefas, não considerando a potência das lâmpadas, sua localização, características e propriedades físicas. O processo de análise ocorreu por meio de frequentes medições de incidência luminosa, nos locais das tarefas, na busca por consolidar resultados com maior proximidade possível às normativas expressas pela NBR ISO/CIE 8995-1. Um fato esperado, porém não totalmente compreendido, foi a necessidade de sucessivas iterações, teóricas e práticas, do protótipo, até que fosse possível englobar todos os objetivos propostos em um sistema funcional usável.

O sensor de luminosidade referente à central foi posicionado estaticamente em um local no ambiente de automação. A escolha do melhor ponto deu-se em função das iterações, averiguando-se as incidências luminosas artificiais e naturais, de modo a propiciar uma melhor relação entre o algoritmo de verificação de intensidade luminosa e a quantidade de iluminância. O sensor utilizado para capturar os dados de incidência luminosa em cada localidade da sala não é estático. Ele foi posicionado no ambiente da tarefa selecionada, sendo realizadas diferentes medições, as quais dependeram da iluminação natural propiciada por diferentes horários e condições de luz.

Durante a aplicação, quando um cenário de iluminação era utilizado, as outras opções de conjuntos de luzes não referentes a ele ficavam inativas. Essa decisão se deveu ao fato de que, ao se acionar outra fonte luminosa no ambiente, adiciona-se imprevisibilidade ao sistema. Nesse sentido, haveria necessidade de uma quantidade maior de sensores de iluminância, sendo ideal a instalação de um para cada caso. Por exemplo, ao se ativar uma lâmpada muito próxima ao sensor, o valor lido não será preciso, ou seja, a medição não será válida. Isto é, a quantidade de luz capturada estará numericamente correta, entretanto, devido à posição do sensor e da fonte de luz, poderá haver disparidades na iluminação perceptível total do ambiente, o que atrapalhará o algoritmo de captura de intensidade luminosa.

3.2.1 Medição da incidência luminosa

Para a tarefa aferir a incidência luminosa presente em um determinado ambiente, há diferentes abordagens. Com base nas leituras realizadas para o trabalho, foram delimitadas algumas perspectivas. São elas: a) Calcular a relação entre as características e a área total do ambiente de trabalho com a potência luminosa total disponível no local; b) Utilizar algum equipamento que forneça informações de incidência luminosa, como fotômetros e luxímetros; c) Avaliar, através de experimentação, qual o melhor cenário para determinado tipo de trabalho.

Vistas as opções, podemos observar que as abordagens de medição seguem caminhos distintos, mas visam a uma mesma solução: definir a adequação luminosa do ambiente dada uma determinada tarefa. Assim, um dos pontos fundamentais do protótipo desenvolvido se caracteriza pela necessidade de quantificar a iluminância dos ambientes. A partir da captura da incidência luminosa e da adequação dos dados para um formato válido, foi possível prover autossuficiência ao controle de iluminação.

Com base nos dados capturados pelo sensor, uma política de iluminação autônoma foi desenvolvida. As regras de acionamento não supervisionado dos diferentes conjuntos luminosos são baseadas na já citada NBR ISO/CIE 8995-1, que entrou em vigor em março de 2013. Para a validação da funcionalidade do protótipo e a elaboração dos cenários, usou-se a normativa como referência. Ela disponibiliza informações quantificadas numericamente a respeito da quantidade de iluminância (medida em lux) necessária em um ambiente para a realização das diferentes tarefas.

Como mencionado anteriormente no texto, uma sala em uma residência doméstica é concebida com propósitos variados. É um ambiente multiuso e suas finalidades irão depender, principalmente, da vontade dos inquilinos. Em espaços dessa natureza é comum imaginarmos muitos cenários. Ou seja, cada sala é diferente, seja em relação ao formato físico ou ao propósito. A união desses fatores caracteriza um cômodo singular, que necessita de iluminação adequada para que as atividades comumente realizadas no ambiente sejam efetuadas com sucesso, independente da incidência luminosa natural advinda do meio externo. Por fim, a utilização da sala reflete diretamente a rotina e o estilo de vida dos moradores.

Nessa etapa do estudo, foram definidos seis cenários diferentes: leitura, jantar, filmes, música e descanso. Após a delimitação, identificou-se a necessidade de criar padrões de iluminação para as tarefas selecionadas. Para tanto, tomaram-se como referenciais os valores

de iluminância elencados pela NBR ISO/CIE 8995-1 (Quadro 3). Na Figura 33, os cenários elencados estão dispostos na planta do local onde foi implantado o protótipo.

Cenário	Ambientes disponibilizados na NBR ISO/CIE 8995-1	Valor de iluminância requerido
Leitura	Escritório	500lx
Jantar	Sala de jantar	200lx
Cinema	Teatros	200lx
Suave	Salas de descanso	100lx
Música	Salas de concerto	300lx
Automático	Salas com multiuso	300lx

Quadro 3 – Valores de referência da NBR ISO/CIE 8995-1 para o estabelecimento de cenários.

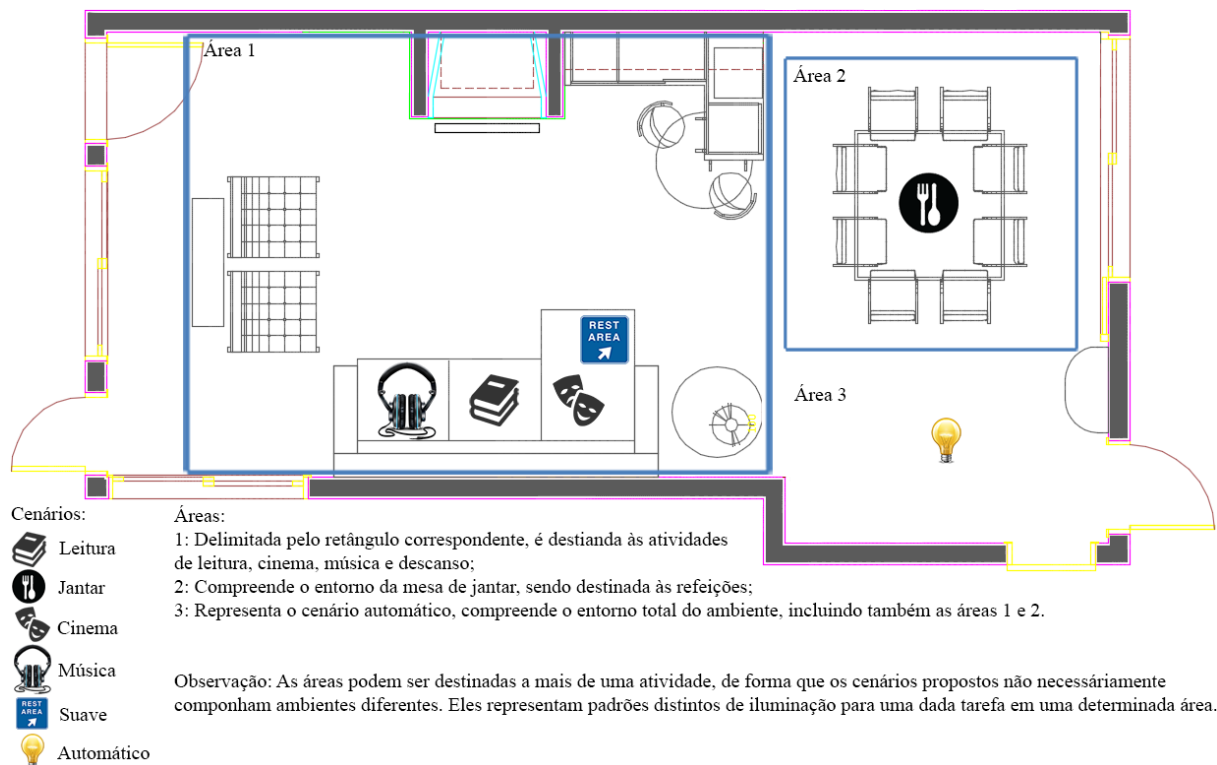


Figura 33 – Divisão dos cenários do ambiente em que o protótipo foi instalado com base na planta do local.

3.2.2 Resultados das medições

Na confecção do protótipo, realizou-se uma mediação da incidência luminosa nos locais destinados a cada tarefa específica. Ou seja, ao se selecionar um cenário, a iluminação foi adequada para a atividade requerida em um determinado ponto da sala. No entanto, isso não implica, diretamente, que as condições de iluminação foram válidas para qualquer localidade no ambiente. Essa situação ocorreu porque que o recinto onde a tecnologia foi implantada é muito amplo. Outra questão que influencia na situação apresentada é a própria natureza do local, pois uma sala, geralmente, serve a múltiplos usos. Por mais que a tratemos como um ambiente único, dentro dela podem existir subdivisões – explicitadas pelos tipos móveis e suas respectivas posições, evidenciando-se a divisão do cômodo em ambientes de trabalho distintos. Por exemplo, o cenário de leitura ajusta a iluminação no entorno do sofá e na área próxima a televisão, onde há lugar para sentar e apoiar os braços e, assim, ser realizada a leitura de um livro, uma carta ou qualquer outro tipo de documentação semelhante. Entretanto, não é garantido que a iluminância para leitura seja suficiente em outra localidade sala. Se o modo de leitura for selecionado no momento em que se estiver à mesa de jantar, existe a possibilidade de que a iluminação não seja adequada, uma vez que estamos fora da área de trabalho especificada para tal atividade. Abaixo, no Quadro 4, estão discriminados os valores obtidos nas medições para cada uma das tarefas, setorizadas em suas respectivas localidades, considerando-se a referência da NBR ISO/CIE 8995-1.

Cenário	Período			Referência NBR ISO/CIE 8995-1
	Dia ou ensolarado	Tarde ou nublado	Noite ou escuro	
Leitura	530lx	530lx	550lx	500lx
Jantar	230lx	206lx	210lx	200lx
Cinema	213lx	210lx	206lx	200lx
Suave	90lx	87lx	99lx	100lx
Música	323lx	315lx	317lx	300lx
Automático	321lx	320lx	321lx	300lx

Quadro 4 – Medições de iluminância nos locais das tarefas.

Durante a aplicação, realizaram-se três sequências de medição por cenário, as quais compreenderam também três avaliações diárias, em horários diferentes, para que fosse possível analisar os efeitos das diferentes incidências de luz natural sobre o controle iluminação. Para cada teste, realizou-se uma medição em diferentes turnos (manhã, tarde e noite). As condições de luz natural foram irregulares, mas, de forma geral, foi possível chegar a uma subdivisão. As medições, expostas na tabela acima, em comparação com as normativas, foram separadas em três colunas, denotadas como: dia ou ensolarado, tarde ou nublado e noite ou escuro.

Essa diferenciação justifica-se em função das características autônomas do sistema. A influência da iluminação natural que incide sobre o ambiente interno influencia de modo direto a iluminância. Dada uma determinada quantidade de luz, o sistema averigua, através de uma medição, que ocorre a cada dez segundos, se há iluminação suficiente para o cenário selecionado. Diferentes conjuntos de luzes e intensidades foram manipulados com os objetivos de evitar excessos de incidência luminosa e de consumo de energia elétrica e de adequar o ambiente às atividades dos ocupantes.

No total, foram realizadas trinta e seis medições, divididas em três verificações de luminosidade diárias para cada cenário, durante um período de três dias. Os resultados obtidos foram muito coerentes e próximos aos valores de referência utilizados. Os números apontados no Quadro 4 (p. 50) foram constantes a cada medição, variando entre dois e três lux, não sendo necessário, portanto, calcular as médias. Os valores agrupados no quadro são referentes à última medição, realizada no terceiro dia de testes. A situação analisada já era uma probabilidade, uma vez que a iluminação do local para uma tarefa deve ser constante e não apresentar grandes variações ao decorrer do tempo. Isso não quer dizer que foi realizada apenas uma medição, pois ocorreram várias aferições de incidência luminosa no projeto, tanto para comprovação das funcionalidades implantadas quanto para adequações necessárias.

Uma informação importante a se ressaltar é que, para a validação de dados, as medições devem ser efetuadas precisamente no local onde a tarefa do cenário foi realizada. Nos projetos luminotécnicos, pode ocorrer de não se ter conhecimento preciso dos tipos de móveis que serão utilizados no ambiente. Portanto, vide dados da NBR ISO/CIE 8995-1, é definida uma altura padrão de 0,75 metros, acima do piso, para o posicionamento do instrumento de medição.

3.2.3 Hardware para medições de iluminância locais

Para esta fase do trabalho, foi utilizado um telefone *Samsung S3* com o programa *Lux Meter*¹⁹ (Figura 34). A seleção do programa citado ocorreu pela necessidade de validar a teoria explorada na seção de desenvolvimento (Capítulo 2) referente à relação entre lux e resistência nominal de um *ldr* comum. O smartphone em questão possui um sensor de luminosidade embutido.

O hardware desenvolvido para a medição é composto por um *arduino UNO*, um *ldr* e um resistor de 10kohm de 1/2watt. O diagrama de conexão foi esquematizado no desenvolvimento do projeto e está explicitado nas Figuras 20 e 23 (p. 35 e 36). É o mesmo circuito utilizado na central de controle de iluminação.

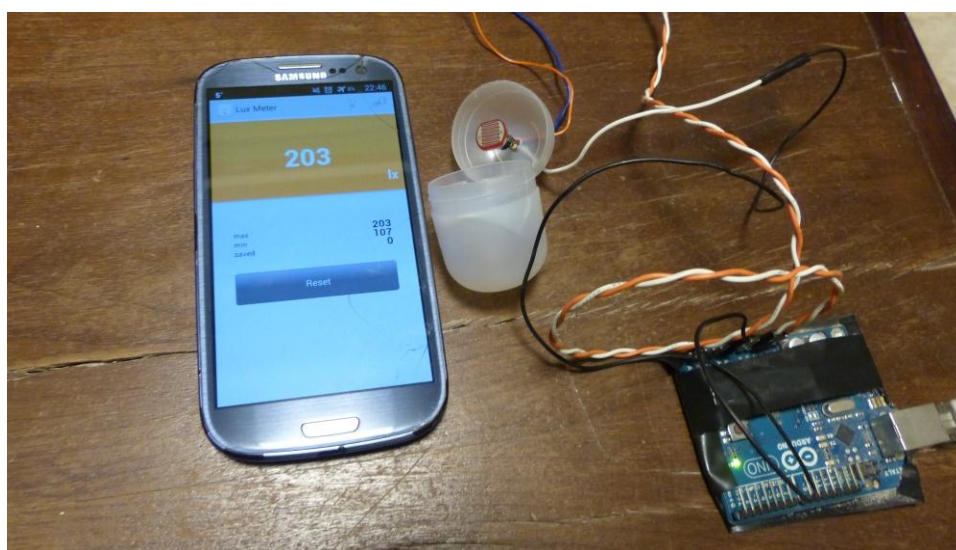


Figura 34 – Programa *Lux Meter* rodando no smartphone *S3* ao lado do hardware medição de incidência luminosa.

A diferença entre o sensor da central e o utilizado para medir a iluminância dos cenários está no código. A seguir, na Figura 35 (p.53), está apresentada a função responsável pela conversão dos dados do sensor em um valor de unidade lux.

¹⁹

Disponível

em:

<

https://play.google.com/store/apps/details?id=com.notquitethem.android.luxmeter&hl=pt_BR>. Acesso em:

```

int lux_meter(){
  valorLido = analogRead(ldrPin); // lê o pino analógico
  voltsLdr=valorLido*0.0048828125; // calcula a voltagem 5/1024
  return lux=600/(10*((5-voltsLdr)/voltsLdr)); // calcula o Lux
}

```

Figura 35 – Função que realiza a medição da incidência luminosa.

As duas funções, tanto a da Figura 22 (p. 36) quanto a da Figura 35, são idênticas. A diferença encontra-se na modificação do valor grifado – 100, na primeira imagem, e 600, na segunda. Esse número é referente à resistência nominal do *ldr* para a função que executa na central. O valor foi dividido por seis pelo fato de que o sensor não está próximo o suficiente de todos os espaços da sala onde os cenários de iluminação são aplicados. Assim, a utilidade do sensor é verificar nuances de luminosidade, não a incidência luminosa presente. A seleção do valor modificado foi fomentada também pelo auxílio do aplicativo Lux Meter. Os resultados obtidos pelo hardware desenvolvido foram muito próximos aos apresentados pelo aplicativo do smartphone. A seguir, na Figura 36, uma captura de tela mostra alguns valores processados pela função da Figura 35.

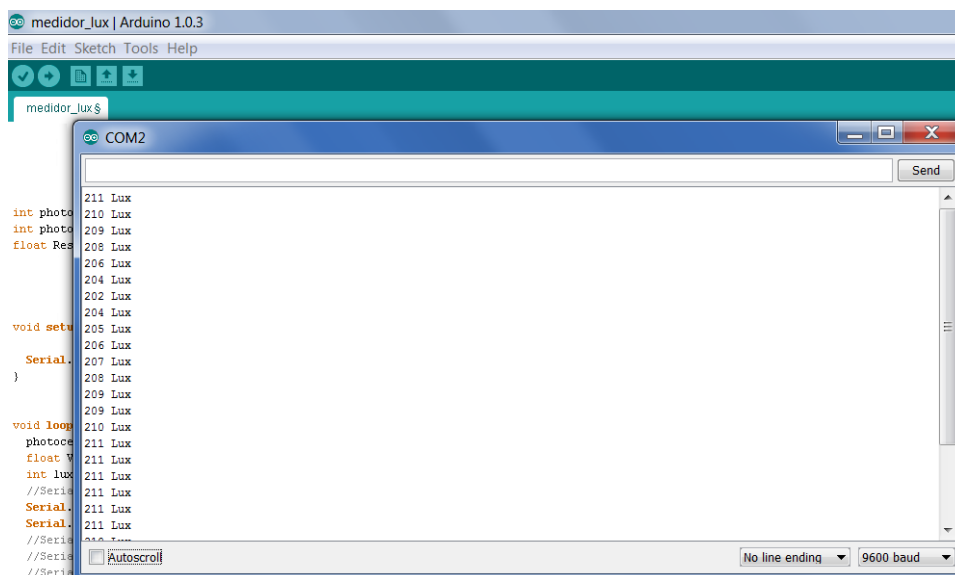


Figura 36 – Captura de tela dos resultados retornados pelo hardware de medição.

3.2.4 Pesquisa dos Materiais

Grande parte das peças que compõem o hardware do protótipo é de origem reciclada. A seguir, o Quadro 5 aponta os principais materiais que foram reutilizados de outros equipamentos (na lista, não estão compreendidos cabos, conectores, ferramentas e fiações em geral).

Componente		Tipo	Quantidade	Origem	Equipamento Reciclado
Relés		12v – 220v @ 10a	14 unidades	Reciclagem	Estabilizadores de tensão
Diodos		1N5407	14 unidades	Reciclagem	Receptores de televisão
Resistores	10k	1k @ 1 watt	2 unidades	Reciclagem	Receptores de televisão
	1k	10k @ 1 watt	14 unidades		Fontes de computador
Sensores	Luminosidade	LDR	2 unidades	Reciclagem	Fotocélula
Transistores	TIP120	NPN	2 unidades	Reciclagem	Amplificador de áudio
	BC547	NPN	14 unidades		Receptores de televisão
Touchpads		<i>Synaptics</i>	2 unidades	Reciclagem	Notebooks

Quadro 5 – Quantidades de peças do protótipo.

É importante considerar que, em alguns casos, a obtenção de tais componentes, advindos de reciclagem, pode implicar diretamente em três questões principais. São elas: a) O funcionamento dos utensílios; b) Obtenção das peças; c) Tempo e energia gastos para conseguir os materiais.

Se um equipamento é tirado de utilização, está em uma sucata, ou fora de uso por qualquer motivo, a possibilidade de que esteja com algum defeito é muito grande. Por isso, é

de suma importância ponderar a forma de aquisição dos equipamentos provenientes de reciclagem. A aquisição pode ser através de compra, doação ou reaproveitamento de recursos já existentes no local do trabalho. Em qualquer caso, deve-se assumir o risco de que as peças estejam com sua funcionalidade comprometida ou que seu tempo de vida útil esteja chegando ao fim. Dependendo do tipo de material procurado, algumas vezes, é recomendável que se compre uma peça nova. Se o preço for acessível, essa, certamente, é uma escolha sensata. Consideradas as razões citadas acima, deve-se perceber a necessidade de despender tempo para encontrar equipamentos que possam ser reutilizados para, posteriormente, realizar-se a ação de reciclagem. É essencial considerar que, além do tempo de projeto para remover os componentes e testá-los, existirão também gastos com energia elétrica, locomoção, dentre outros fatores.

Analisadas as questões apontadas e a partir de um contexto de trabalho específico, a reciclagem de componentes é muito bem-vinda e propícia a protótipos como o deste estudo. Não será feita uma discussão sobre de reciclagem de lixo eletrônico, mas é interessante pensar que podemos desenvolver projetos de grande notoriedade com materiais que estão ao nosso redor e que, outrora, foram considerados como resíduos inúteis. Não obstante, podemos obter bons componentes, inclusive, em alguns casos, com qualidade superior a de produtos encontrados comumente no comércio.

O Quadro 6, abaixo, evidencia os custos de obtenção (em R\$) das principais peças que integram o protótipo.

(continua)

Componente (em unidades)	Reciclagem	Internet (Brasil)	Internet (Exterior)	Loja Física 1	Loja Física 2
Resistor 1k (14 u)	Na	R\$4,00	U\$4,47	R\$1,40	R\$5,00
Resistor 10k (1 u)	Na	R\$0,25	U\$1,49	R\$0,10	R\$0,50
LDR (2 u)	Na	R\$2,21	U\$0,21	R\$7,00	R\$5,00
TIP120 (2 u)	Na	R\$2,70	U\$1,99	R\$4,00	R\$5,00
BC547 (14 u)	Na	R\$5,40	U\$7,00	R\$4,20	R\$7,00
Diodo 1N5407 (14 u)	Na	R\$2,00	U\$2,97	R\$7,00	R\$14,00
Relé (14 u)	Na	R\$27,44	U\$9,98	R\$49,00	R\$54,00
TX/RX RF (1 u)	Ne	R\$23,32	U\$0,99	Ne	Ne
HC-06 (1 u)	Ne	R\$30,00	U\$4,47	Ne	Ne

(conclusão)

Touchpad (2 u)	Na	R\$5,50	U\$5,82	Ne	Ne
Fonte 12vdc @600ma (2 u)	R\$20,00	R\$39,80	U\$11,90	R\$60,00	R\$58,00
<i>Arduino UNO</i> (1 u)	Ne	R\$75,00	US\$24,95	Ne	Ne
<i>Arduino Nano</i> (Compatível - 1 u)	Ne	R\$44,00	US\$4,69	Ne	Ne
1 x <i>Arduino Leonardo</i> (Compatível – 1u)	Ne	R\$114,00	US\$7,68	Ne	Ne
Somatório	R\$20,00	R\$370,12	U\$88,61 ou R\$196,71	R\$132,70	R\$148,50

Quadro 6 – Valores dos componentes (onde → Na: peças adquiridas sem custos; Ne: componentes não encontrados; valores em negrito: produtos adquiridos para o projeto).

Para a validação, os preços dos componentes utilizados no protótipo foram pesquisados em diferentes fontes. Não estão compreendidas especificidades, como cabos, conectores e semelhantes. Conforme os dados explicitados no quadro, verificou-se que existe muita variação de preço entre os componentes, dependendo, principalmente, do tipo de peça necessária. Algumas vezes, os valores são semelhantes e, em outros casos, ocorrem grandes diferenças. Além da diversidade de preços, em função da localidade das lojas, ainda podem ocorrer discrepâncias entre vendedores concorrentes. Geralmente, recomenda-se que peças de hardware mais comuns sejam compradas nas lojas locais, visto que há pouca variabilidade de valores e devido ao imediatismo da entrega. Para hardware específico, é interessante que se pesquisem opções de aquisição via internet. Um bom exemplo desse tipo de situação, no presente projeto, foi a compra da placa *arduino nano*, adquirida por U\$4,69, em uma loja estrangeira, com o valor do frete incluso. A mesma placa foi encontrada em sites brasileiros por valores que variaram entre R\$144,00 e R\$272,00, sem considerar os custos de entrega.

As peças utilizadas no protótipo, em sua maioria, são oriundas de reaproveitamento. Ou seja, não houve custos de compra, apenas gastos de tempo e de energia elétrica para que os componentes fossem retirados dos equipamentos e, posteriormente, testados. Para o protótipo, foram comprados três arduinos e duas fontes de doze volts usadas. Ao todo, o projeto custou **R\$102,12**, não inclusos valores de cabeamento, conectores e serviços de alvenaria. A

economia de custo, dado um cenário de reciclagem ideal, é muito válida. Isso se confirma com o protótipo desenvolvido, que contou com o uso de algumas peças novas, e, ainda assim, obteve uma economia no entorno de 50% (em comparação ao que seria gasto se o projeto contivesse apenas peças de primeiro uso). Dessa forma, a importância do reaproveitamento se destacou. Nesse sentido, dentre muitas questões, além da economia, uma que é muito pertinente relaciona-se à percepção de quanto material e lixo eletrônico é gerado pela sociedade. Tal fato evidencia, diretamente, a necessidade de se encontrarem melhores destinos para os resíduos tecnológicos.

3.2.5 Peças defeituosas ou substituídas

Dos componentes utilizados oriundos de reciclagem, apenas três apresentaram defeito e foram substituídos por outros (também reciclados). A Tabela 5 apresenta uma listagem desses materiais.

Tabela 5 – Componentes defeituosos.

Componente	Total	Defeituosos ou modificados
Relés	14	2
Transistores BC547	14	1

3.3 Implicações da pesquisa

Dentro do contexto de flexibilidade, foram imaginadas e estudadas algumas tecnologias e ideias que não estão contempladas no presente estudo. Entretanto, algumas delas foram testadas, sendo consideradas como de grande valia para próximos estudos e futuras iterações do sistema desenvolvido. Pode-se utilizar, por exemplo, outras placas da *arduino* ou tecnologias semelhantes. Na introdução, já foram citadas algumas delas, para implantar outros

módulos de automação, tais como: controle de nível na caixa de água e consumo, controle de temperatura de aquecedores a gás e interfaces via sistemas web ou smartphones.

Qualquer linguagem de programação que suporte comunicação, através de algum dos hardwares disponíveis neste protótipo (detalhados no Capítulo 2), possibilita projetar sistemas maiores. Existindo um hardware que ofereça conexão com o protótipo e um software que forneça o suporte, podemos interconectar diferentes tecnologias, de fins variados, no intuito de englobar todos os possíveis módulos de controle e sensores utilizados, formando um sistema unificado.

Uma aplicação considerada muito promissora é a utilização da api²⁰ Google Voice para comandar as funcionalidades de uma automação. Pode-se pensar, em um primeiro momento, que esse recurso é uma futilidade, uma vez que, por conta das próprias limitações de software e hardware, a tecnologia não é prática, ou mesmo usável, em nosso dia a dia. No entanto, ela tem o potencial de ser extremamente útil às pessoas que possuem algum tipo de desabilidade física (motora, visual, de movimentação). Um exemplo interessante para se começar a trabalhar com o Google Voice é o pacote PiAUISuite, desenvolvido por Steven Hickson²¹.

²⁰ *Application Programming Interface* ou, traduzido, Interface de Programação de Aplicações.

²¹ Disponível em: <<https://github.com/StevenHickson/PiAUISuite>>. Acesso em: 2 jun. 2014.

CONCLUSÃO

Os fins desta pesquisa foram relatados ao longo do desenvolvimento, e os resultados estão discriminados, numericamente, na seção de metodologia. Nem todos os valores aferidos foram totalmente satisfatórios, sendo que o principal ponto a ser reavaliado é quanto aos dados colhidos sobre a iluminação de alguns ambientes, como, por exemplo, a do cenário suave. Nas medições, ele não obteve os valores mínimos requeridos pela norma. Foi possível atingir valores superiores aos indicados, através do acionamento de mais conjuntos de luzes em dias com iluminação natural ideal. No entanto, quando utilizadas mais fontes luminosas, os resultados das medições chegaram a mais de duzentos lux, o que também foi considerado um problema. A proximidade dos valores encontrados para esse cenário torna a solução boa, mas se percebe que há necessidade de melhorias em iterações futuras.

Também se notou que, em outros cenários, a incidência luminosa chegou muito perto à referenciada pela norma. Porém, ocorreram alguns casos em que os valores capturados se distanciaram um pouco do valor ideal. É possível que esse fato tenha ocorrido em função de que a disposição dos conjuntos de fontes luminosas não fosse adequado para os tipos de cenários propostos, considerando-se a disposição da própria sala utilizada para o estudo. No entanto, o projeto de iluminação não é parte desta pesquisa e, portanto, mesmo com essa ressalva, os resultados aferidos foram considerados válidos.

Algumas outras questões tiveram grande importância durante o projeto, estando diretamente ligadas aos resultados. Com relação aos objetivos, em sua maioria, eles foram atingidos com sucesso. Entretanto, é necessário relatar que se encontraram muitas dificuldades e barreiras que forçaram mudanças ao decorrer do projeto. Essas transformações foram fundamentais e puderam ser realmente percebidas, uma vez que o protótipo não é apenas teórico. Os conhecimentos pesquisados foram implantados em um ambiente e suas implicações foram analisadas. A partir da instalação do sistema, das iterações e dos testes, foi possível identificar questionamentos que não teriam sido evidenciados apenas com um planejamento teórico. Alguns deles são: posicionamento dos sensores; posição dos botões no touchpad; dimensões do hardware de controle; limitações de hardware; necessidade de modificação física no local de implantação; demora na pesquisa e obtenção de material; necessidade de testes e calibração dos sensores; estudo de tecnologias dos sensores; usabilidade da central; testes de conformidade.

Apesar de algumas questões, de modo geral, o controle de iluminação do protótipo está devidamente implantado e funcional, sendo que todos os requisitos propostos foram atingidos. A partir do trabalho realizado, percebe-se a importância de estudos desse tipo no meio acadêmico e na sociedade. Pesquisas do campo focado neste projeto são de caráter multidisciplinar, envolvendo uma grande gama de conhecimentos e diversos profissionais para a elaboração de projetos que visem atender ao bem coletivo. Ao longo do desenvolvimento da prototipação, implicações surgiram e foram desenvolvidas e documentadas no trabalho, como forma de auxiliar nas discussões e em novos estudos da área. Por fim, conclui-se que o futuro de sistemas de automação depende da criatividade e da persistência dos acadêmicos e dos profissionais, corroborando para o desenvolvimento de tecnologias padronizadas que possam auxiliar no dia a dia do cidadão comum.

REFERÊNCIAS

ABNT. Associação Brasileira de Normas Técnicas. **NBR ISO/CIE 8995-1**: iluminação de ambientes de trabalho. Rio de Janeiro, 2013.

ASTOLFO, D. HC-06 Bluetooth Module. Disponível em: <<http://www.plastibots.com/index.php/2013/12/31/hc-06-bluetooth-module-getting-this-sucker-into-at-command-mode/>>. Acesso em: 27 maio 2014.

DEFINIÇÃO de Arduino. In: ARDUINO. [S.l.], [201-]. Disponível em: <www.arduino.cc/>. Acesso em: 10 mar. 2014.

DEFINIÇÃO de Automação. In: EWG Consultoria. Esteio, 2013. Disponível em: <www.ewgconsultoria.com.br/ewg/2013/bematech.php/>. Acesso em: 10 mar. 2014.

DELGADO, A. R.; GROUT, V.; PICKING, R. Remote-Controlled Home Automation Systems with Different Network Technologies. In: GLYNDWR University Cards. Reino Unido: Glyndwr University. Disponível em: <<http://www.glyndwr.ac.uk/CARDS/Papers/>>. Acesso em: 11 abr. 2014.

DUTRA, L.; LAMBERTS, R.; PEREIRA, F. O. R. **Eficiência Energética na Arquitetura**. São Paulo: PW Editores, 1997. Disponível em: <www.labee.ufsc.br/publicacoes/livros/>. Acesso em: 4 abr. 2014.

FILIFELOP. Módulo RF Transmissor + Receptor 433mhz AM. Disponível em: <<http://blog.filipeflop.com/wireless/modulo-rf-transmissor-receptor-433mhz-am.html>>. Acesso em: 25 maio 2014.

KAWASAKI, J. I. [Entrevista disponibilizada a Luciana Freitas]. 2013. Disponível: <http://media.wix.com/ugd/b0e126_7193b9ef9e7a394d3a71a2c1b8bb54cb.pdf>. Acesso em: 15 abr. 2014.

LAMBERTS, R; et al. **Casa Eficiente**: Bioclimatologia e Desempenho Térmico. Florianópolis: UFSC, 2010. Disponível em: <www.labee.ufsc.br/sites/default/files/publicacoes/livros/CasaEficiente_vol_I_WEB.pdf>. Acesso em: 17 mar. 2014.

PS2 mouse interface for Arduino. Disponível em: <<http://playground.arduino.cc/ComponentLib/Ps2mouse>>. Acesso em: 5 abr. 2014.

SERIAL. Disponível em: <<http://playground.arduino.cc/Referencia/Serial>>. Acesso em: 5 abr. 2014.

APÊNDICE

Apêndice A – Cronograma de Trabalho

Etapas / Meses (ano 2014)	Março	Abril	Maio	Junho	Julho
1. Definição do projeto	X				
2. Definição das características do protótipo	X	X			
3. Escolha das tecnologias		X	X		
4. Projeto do protótipo		X	X	X	
5. Implantação do projeto				X	X
6. Revisão bibliográfica	X	X	X	X	X
7. Apresentação					X

*Aplicação do Cronograma: X – etapas concluídas.