

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS  
*Campus* DIVINÓPOLIS  
GRADUAÇÃO EM ENGENHARIA MECATRÔNICA

Rayner Ribeiro de Moraes

DESENVOLVIMENTO DE SISTEMAS DE DOMÓTICA CONTROLADOS  
POR DISPOSITIVOS MÓVEIS

Divinópolis  
2015

Rayner Ribeiro de Moraes

DESENVOLVIMENTO DE SISTEMAS DE DOMÓTICA CONTROLADOS  
POR DISPOSITIVOS MÓVEIS

Monografia de Trabalho de Conclusão de Curso apresentado ao Colegiado de Graduação em Engenharia Mecatrônica como parte dos requisitos exigidos para a obtenção do título de Engenheiro Mecatrônico.

Áreas de integração: Eletrônica e Programação.

Orientador: Prof. Mes. Alan Mendes Marotta  
Co-orientador: Eng. Apolo Malta

Divinópolis  
2015

Rayner Ribeiro de Moraes

DESENVOLVIMENTO DE SISTEMAS DE DOMÓTICA CONTROLADOS  
POR DISPOSITIVOS MÓVEIS

Monografia de Trabalho de Conclusão de Curso  
apresentado ao Colegiado de Graduação em Enge-  
nharia Mecatrônica como parte dos requisitos exi-  
gidos para a obtenção do título de Engenheiro Me-  
catrônico.

Áreas de integração: Eletrônica e Programação.

Comissão Avaliadora:

Prof. Mes. Alan Mendes Marotta  
CEFET/MG *Campus V*

Prof. Dr. Valter Júnior de Souza Leite  
CEFET/MG *Campus V*

Prof. Dr. João Carlos de Oliveira  
CEFET/MG *Campus V*

Divinópolis  
2015

---

Centro Federal de Educação Tecnológica de Minas Gerais  
CEFET-MG / *Campus V* - Divinópolis  
Curso de Engenharia Mecatrônica

Monografia intitulada “DESENVOLVIMENTO DE SISTEMAS DE DOMÓTICA CONTROLADOS POR DISPOSITIVOS MÓVEIS” de autoria do graduando Rayner Ribeiro de Moraes, aprovada pela banca constituída pelos seguintes professores:

---

Prof. Mes. Alan Mendes Marotta  
CEFET-MG / *Campus* Divinópolis - Orientador

---

Prof. Dr. Valter Júnior de Souza Leite  
CEFET-MG / *Campus* Divinópolis

---

Prof. Dr. João Carlos de Oliveira  
CEFET-MG / *Campus* Divinópolis

---

Prof. Dr. Lúcio Flávio Santos Patrício  
Coordenador do Curso de Engenharia Mecatrônica  
CEFET-MG / *Campus* Divinópolis

Divinópolis  
2015

“SOMOS O QUE ESCOLHEMOS SER.  
O DESTINO POUCO TEM A VER  
COM ISSO”.

Autor Desconhecido.

# Agradecimentos

Aos meus pais, Fernando e Patrícia, por todo apoio, investimento, amor e tentativas de me compreender no decorrer desse último ano. Aos meus irmãos Kevin e Yanne, pelo companheirismo e cumplicidade. Às minhas avós Aparecida, Lázara e Vitória, pela confiança e contínuas tentativas de me ensinar sobre felicidade e compaixão.

Ao meu orientador Alan Marotta, por sua orientação, não somente nesse trabalho, como também durante todo o meu percurso no CEFET-MG. Obrigado por todo apoio, ensinamento, confiança em minha capacidade e por ter se tornado um exemplo de pessoa e profissional para mim.

Aos meus amigos de graduação obrigado por, além do auxílio e compartilhamento de tantos conhecimentos técnicos, terem também compartilhado bons momentos dentro e fora do CEFET-MG. Agradeço em especial ao Apolo, sem sua participação, como amigo e co-orientador, esse trabalho não seria o mesmo. Agradeço também aos companheiros de laboratório, André Palhares, Edson Ribeiro, Hugo Machado, Alan Crístopher e Frederico Rabelo, já que ninguém faz um trabalho sozinho.

Aos professores e funcionários do CEFET-MG, obrigado por proporcionarem a infraestrutura e base necessárias ao trabalho e por estarem sempre dispostos a ajudar.

Aos meus amigos e família, que de perto ou de longe, me acompanharam e me ouviram nessa jornada, obrigado a todos vocês! Eu não teria chegado até aqui sozinho.

Enfim, à todos os que auxiliaram de alguma forma no desenvolvimento deste trabalho, meus sinceros agradecimentos.

# Resumo

O presente trabalho apresenta uma proposta de simplificação de sistemas domóticos para iluminação, segurança, refrigeração e entretenimento, e a criação de uma interface do usuário que pode ser controlada por redes sem fio com dispositivos móveis. Para o desenvolvimento deste projeto, a estrutura de comunicação e transmissão de dados foi realizada a partir de uma topologia em estrela, em que o microprocessador *Raspberry pi* está centralizado gerenciando os microcontroladores *Atmega 328* e interligando todos os equipamentos. A adoção do padrão *Ethernet* possibilitou um controle dinâmico com baixa sensibilidade a ruídos dos sinais recebidos pelo roteador emitido pela plataforma móvel. Dessa forma, com um simples comando pelo navegador de um equipamento remoto (*laptop*, *tablet* ou *smartphone*) é possível, por exemplo, acender uma lâmpada, regular a intensidade do ventilador e monitorar em tempo real os sensores de segurança da residência, independente da localização do usuário. A implementação desse projeto de automação garante alto grau de sustentabilidade quanto ao impacto ecológico consequente, proporcionando economia de energia e longa durabilidade dos equipamentos. No que se refere à viabilidade econômica foi possível estabelecer custo de implementação reduzido, com rápido retorno de investimento. Além disso, os módulos foram desenvolvidos de forma a serem compactos quanto à montagem física do sistema, garantindo conforto e segurança ao usuário.

Palavras-chave: Domótica. Controle remoto. Redes sem fio. Economia de energia.

# Abstract

This work aims a proposal to simplify home automation systems for lighting, security, cooling and entertainment, and creation of a user interface that can be controlled by wireless networks by mobile devices. For the development of this project, the structure of communication and data transmission will be carried out from a star topology, in which the microprocessor Raspberry PI is centered managing microcontrollers Atmega 328 and connecting all equipment. Adoption of Ethernet protocol enables dynamic control with low sensitivity to noise signals received from the router issued by the mobile platform. Thus, with a simple command by a remote device browser (laptop, tablet or smartphone) it will be possible, for example, turn on a light, adjust fan intensity monitoring in real-time safety sensors of the home, regardless of user location. Implementation of this project aims a high degree of sustainability as the consequent ecological impact, providing energy savings and extended duration of the equipment. As regards to the economic viability it was possible to establish low implementation cost, with quick investment return. Moreover, the modules were designed to ensure acceptable compactness as to physical mounting system, ensuring comfort and safety to the user..

Key-words: Home automation. Remote control. Wireless networks. Energy saving.

# Sumário

<b>Lista de Figuras</b>	<b>xiii</b>
<b>Lista de Tabelas</b>	<b>xiv</b>
<b>Lista de Acrônimos e Notação</b>	<b>xv</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Definição do Problema . . . . .	1
1.2 Motivação . . . . .	4
1.3 Objetivos do Trabalho . . . . .	4
1.4 Organização do Documento . . . . .	6
<b>2 Fundamentação Teórica</b>	<b>7</b>
2.1 Domótica . . . . .	7
2.2 Mercado . . . . .	10
2.3 Projetos . . . . .	11
2.4 Estado da Arte . . . . .	12
2.4.1 Iluminação . . . . .	13
2.4.2 Refrigeração . . . . .	18
2.4.3 Segurança . . . . .	18
2.4.4 Entretenimento . . . . .	18
2.5 Redes . . . . .	19
2.5.1 Wi-Fi . . . . .	21
2.5.2 Bluetooth . . . . .	21
2.5.3 ZigBee . . . . .	21
2.5.4 Comparação das redes sem fio . . . . .	22
2.5.5 Ethernet . . . . .	22
2.5.6 A Internet . . . . .	23
2.6 Comunicação Serial . . . . .	25
2.7 Linguagens de Programação . . . . .	27
2.7.1 HTML . . . . .	28
2.7.2 C e C++ . . . . .	29
2.7.3 Java . . . . .	29
2.8 Microcomputadores . . . . .	30

---

<b>3</b>	<b>Metodologia</b>	<b>32</b>
3.1	Materiais e Métodos . . . . .	37
3.1.1	Configuração do Sistema . . . . .	37
3.1.2	Microprocessador Raspberry Pi . . . . .	40
3.1.3	Microcontrolador Atmega 328 . . . . .	42
3.1.4	Módulo Dimer . . . . .	43
3.1.5	Módulo Relé . . . . .	49
3.1.6	Módulo Controle CC . . . . .	50
3.1.7	Medidor de Corrente . . . . .	51
3.1.8	Fonte de Corrente Contínua . . . . .	52
3.1.9	Topologia de Instalação . . . . .	54
3.1.10	Planta do Sistema . . . . .	57
3.2	Ferramentas . . . . .	57
3.2.1	Interface Principal . . . . .	59
3.2.2	Páginas de Controle . . . . .	60
3.2.3	Controle pela Internet . . . . .	67
<b>4</b>	<b>Resultados e Discussões</b>	<b>69</b>
4.1	Economia de Energia . . . . .	69
4.2	Redução do Custo de Implementação . . . . .	73
4.3	Retorno de Investimento . . . . .	74
4.4	Durabilidade de Equipamentos . . . . .	76
4.5	Compactidade com Alta Taxa de Transferência de Dados . . . . .	77
<b>5</b>	<b>Considerações Finais</b>	<b>79</b>
5.1	Conclusão . . . . .	79
5.2	Propostas de Continuidade . . . . .	80
<b>A</b>	<b>Apêndice - Códigos</b>	<b>82</b>
A.1	Código HTML - Página Inicial . . . . .	82
A.2	Código Java Server Page - Controle dos Sistemas Individuais . . . . .	85
A.3	Código Java Script - Tratamento do Controle . . . . .	86
A.4	Código Cascading Style Sheet - Visual dos Elementos . . . . .	89
A.5	Código Java Server Page - Medição . . . . .	89
A.6	Código Java Script - Gráfico de Consumo . . . . .	90
A.7	Código Java Server Page - Comunicação com Microcontrolador . . . . .	90
A.8	Código Java - Comunicação da porta serial . . . . .	93
A.9	Código C++ - Microcontrolador 1 . . . . .	94
A.10	Código C++ - Microcontrolador 2 . . . . .	97
<b>B</b>	<b>Apêndice - Configuração Raspberry Pi</b>	<b>100</b>
B.1	Especificações . . . . .	100
B.2	Hardware . . . . .	101
B.3	Instalação do Sistema Operacional . . . . .	102
B.4	Primeiro BOOT . . . . .	103
B.5	Atualização do Sistema . . . . .	105
B.6	Instalação de Programas . . . . .	106
B.7	Transferência de Arquivos ( <i>DEPLOY</i> ) . . . . .	107
B.8	Controle pela Internet . . . . .	108

B.9 Execução do Projeto . . . . .	108
B.10 Comandos Úteis . . . . .	110
<b>Referências</b>	<b>111</b>

# Lista de Figuras

1.1	Consumo de energia mundial . . . . .	2
1.2	Distribuição de consumo elétrico em uma residência . . . . .	3
1.3	Usuários de telefones celulares em todo mundo . . . . .	3
1.4	Penetração global de dispositivos . . . . .	5
2.1	Residência Inteligente . . . . .	9
2.2	Fluxo de projeto integrado em uma residência . . . . .	11
2.3	Eficiência, IRC e vida útil de lâmpadas. . . . .	15
2.4	Diagrama com os diversos tipos de controle de iluminação . . . . .	16
2.5	Topologias de cabeamento de comunicação em residências . . . . .	19
2.6	Integração de sistemas residenciais em topologia tipo estrela . . . . .	20
2.7	Tecnologias de redes domésticas . . . . .	20
2.8	Cabo Cat5 usado no padrão Ethernet . . . . .	23
2.9	Pilha de protocolos da Internet . . . . .	24
2.10	Partes básicas de um microcomputador. . . . .	30
2.11	Diagrama de blocos simples de um microcontrolador . . . . .	31
3.1	Fluxograma da metodologia do projeto. . . . .	33
3.2	Esquema de conexão do projeto. . . . .	37
3.3	Processo de configuração do Raspberry PI. . . . .	40
3.4	Microprocessador Raspberry pi Modelo B . . . . .	41
3.5	Microcontrolador Atmega 328 . . . . .	42
3.6	Equivalência de pinagem entre Atmega 328 e Plataforma Arduino. . . . .	43
3.7	Configuração do dimer . . . . .	44
3.8	Medição dos processos da etapa de passagem por zero do dimer. . . . .	45
3.9	Saídas para carga de acordo com o ângulo de disparo do dimer. . . . .	47
3.10	Projeto esquemático do dimer em Proteus. . . . .	48
3.11	Módulo dimer desenvolvido. . . . .	48
3.12	Configuração do relé. . . . .	49
3.13	Projeto esquemático do relé em Proteus. . . . .	50
3.14	Módulo relé desenvolvido. . . . .	50
3.15	Configuração do controle CC. . . . .	50
3.16	Projeto esquemático do circuito de controle CC em Proteus. . . . .	51
3.17	Módulo controle de CC desenvolvido. . . . .	51
3.18	ACS714. . . . .	52
3.19	Circuito medidor de corrente. . . . .	52

---

3.20	Fonte de alimentação CC. . . . .	53
3.21	Conector principal de alimentação da placa mãe da fonte CC. . . . .	54
3.22	Conector de alimentação da fonte CC. . . . .	54
3.23	Instalação Convencional de lâmpadas . . . . .	55
3.24	Instalação automatizada de lâmpadas . . . . .	56
3.25	Instalação Convencional de tomadas . . . . .	56
3.26	Instalação automatizada de tomadas . . . . .	56
3.27	Montagem da planta desenvolvida. . . . .	57
3.28	Planta residencial utilizada na planta. . . . .	58
3.29	Fluxo de informação dos algoritmos do sistema. . . . .	59
3.30	Página HTML - Controle de Sistemas Residenciais. . . . .	61
3.31	Página <i>JavaServer</i> - Controle de Iluminação. . . . .	62
3.32	Controle de Iluminação de lâmpada incandescente. . . . .	63
3.33	Controle de Iluminação de lâmpada de corrente contínua. . . . .	64
3.34	Controle de refrigeração. . . . .	65
3.35	Controle de entretenimento. . . . .	66
3.36	Página de medição de corrente. . . . .	67
4.1	Lâmpadas de corrente contínua 12V de 1 pólo. . . . .	70
4.2	Esquema da montagem experimental. . . . .	71
4.3	Montagem experimental. . . . .	72
4.4	Dimensões de um Raspberry PI módulo B e Laptop de 17". . . . .	78
B.1	Raspberry Pi. . . . .	100
B.2	Hardware do Raspberry Pi. . . . .	101
B.3	Raspbian. . . . .	103
B.4	Programa Win 32 Disk Manager. . . . .	103
B.5	Modo de configuração do Raspberry Pi. . . . .	104
B.6	Ambiente de área de trabalho do Raspberry Pi. . . . .	105

# Lista de Tabelas

1.1	Consumo de energia elétrica por setor no Brasil . . . . .	2
2.1	Comparação entre alguns protocolos de comunicação . . . . .	9
2.2	Análise de custos comparativos entre três tipos de lâmpadas (1). . . . .	16
2.3	Análise de custos comparativos entre três tipos de lâmpadas (2). . . . .	16
2.4	Comparação entre padrões de tecnologias de comunicação de redes sem fio. . . . .	22
2.5	Frame unitário do protocolo Ethernet . . . . .	23
3.1	Cronograma de trabalho para projeto disciplina de TCC. . . . .	35
3.2	Orçamento: Lista de equipamentos e recursos utilizados. . . . .	36
3.3	Saídas CC da fonte de alimentação . . . . .	53
4.1	Amostragem experimental da lâmpada de LED 12V CC. . . . .	69
4.2	Amostragem experimental da lâmpada incandescente 12V CC. . . . .	70
4.3	Características de lâmpadas incandescentes de acordo com a tensão . . . . .	76

# Lista de Acrônimos e Notação

AURESIDE	Associação Brasileira de Automação Residencial
PDE	Plano Decenal de Expansão de Energia
EPE	Empresa de Pesquisa Energética
MME	Ministério de Minas e Energia
ABNT	Associação Brasileira de Normas Técnicas
ANSI	American National Standards Institute (Instituto Nacional Americano de Padrões)
TIA	Telecommunications Industries Association (Associação das Indústrias de Telecomunicações)
EIA	Electronic Industries Alliance (Aliança das Indústrias Eletrônicas)
ISO	International Organization for Standardization (Organização Internacional para Padronização)
IEC	International Electrotechnical Commission (Comissão Eletrotécnica Internacional)
NBR	Norma Brasileira
CFTV	Circuito Fechado de Televisão
DVD	Digital Versatile Disc (Disco Digital Versátil)
CD	Compact Disc (Disco Compacto)
USB	Universal Serial Bus (Barramento Serial Universal)
WIFI	Wireless Fidelity (Fidelidade sem Fio)
IEEE	Institute of Electrical and Electronic Engineers (Instituto de Engenheiros Eletricistas e Eletrônicos)
PAN	Personal Area Network (Rede de Área Pessoal)
WPAN	Wireless Personal Area Network (Rede de Área Pessoal sem Fio)
WLAN	Wireless Local Area Network (Rede de Área Local sem Fio)
IRC	Índice de Reprodução de Cores

TCP	Transmission Control Protocol (Protocolo de Controle de Transmissão)
IP	Internet Protocol (Protocolo da Internet)
HTTP	Hyper Text Transfer Protocol (Protocolo de Transferência de Hipertexto)
HTML	Hypertext Markup Language (Linguagem de Marcação de Hipertexto)
WWW	World Wide Web (Rede de Alcance Mundial)
ASCII	American Standard Code for Information Interchange (Código Padrão Americano para o Intercâmbio de Informação)
OOP	Object Oriented Programming (Programação Orientada à Objeto)
CPU	Central Processing Unit (Unidade Central de Processamento)
RAM	Random Access Memory (Memória de Acesso Aleatório)
SRAM	Static Random Access Memory (Memória Estática de Acesso Aleatório)
DRAM	Dynamic Random Access Memory (Memória de Acesso Aleatório Dinâmico)
ROM	Read Only Memory (Memória Apenas de Leitura)
PROM	Programmable Read Only Memory (Memória Programável Apenas de Leitura)
EPROM	Erasable Programmable Read Only Memory (Memória Programável Deletável Apenas de Leitura)
EEPROM	Electrical Erasable Programmable Read Only Memory (Memória Programável Deletável Eletricamente Apenas de Leitura)
OS	Operational System (Sistema Operacional)
GPU	Graphics Processing Unit (Unidade de Processamento Gráfico)
HD	High Definition (Alta Definição)
HDMI	High Definition Multimedia Interface (Interface de Multimídia de Alta Definição)
RCA	Radio Corporation of America (Corporação de Rádio da América)
PWM	Pulse Width Modulation (Modulação por Largura de Pulso)
PHP	Hypertext Preprocessor (Pré-processador de hypertexto)
IEA	International Energy Agency Agência Internacional de Energia
ROI	Return of Investment (Retorno de Investimento)
PRI	Prazo de Retorno de Investimento

# Introdução

A parte introdutória desta monografia apresenta um panorama geral do trabalho desenvolvido, fazendo um convite ao leitor para a leitura dos demais capítulos deste documento. Apresenta-se aqui, os recursos aplicados à Domótica com ênfase na Automação Residencial, justificando o desenvolvimento desse projeto, seus objetivos gerais e específicos, a forma como foi delimitada esta questão e a estrutura geral dos capítulos do relatório.

## 1.1 Definição do Problema

A Domótica vem crescendo de forma significativa no Brasil nos últimos anos influenciada pelo crescimento da economia e pelo fácil acesso à tecnologias. No entanto, segundo dados apontados pela Aureside (Associação Brasileira de Automação Residencial), o país carece de profissionais especializados capazes de suprir as atuais necessidades do mercado e do cliente (MEDEIROS, 2009).

Em um ambiente doméstico é preciso que toda forma de tecnologia possua interfaces e sistemas amigáveis que sejam fáceis de manusear e flexíveis o suficiente para se ajustar às necessidades específicas de cada cliente. Essa premissa sugere que usuários domésticos são adversos a treinamentos, sendo preciso sempre desenvolver projetos que possam ser facilmente administrados (MURATORI, 2008).

No que se refere à sustentabilidade, o consumo de energia elétrica vem apresentando um aumento constante através dos anos. De acordo com Filho (2013), o consumo de eletricidade na última década cresceu a uma taxa média de 5.8% ao ano, enquanto o consumo residencial em si evidenciou 6.3%. Os estudos do Plano Decenal de Expansão de Energia (PDE) afirmam a previsão de um crescimento à 4.9% ao ano pelos próximos dez anos. A Figura 1.1 apresenta o consumo total de energia mundial. É importante ressaltar que

o Brasil ocupa o sétimo lugar no ranking dos principais consumidores com 293 MTOE (Megatonelada de Óleo Equivalente).

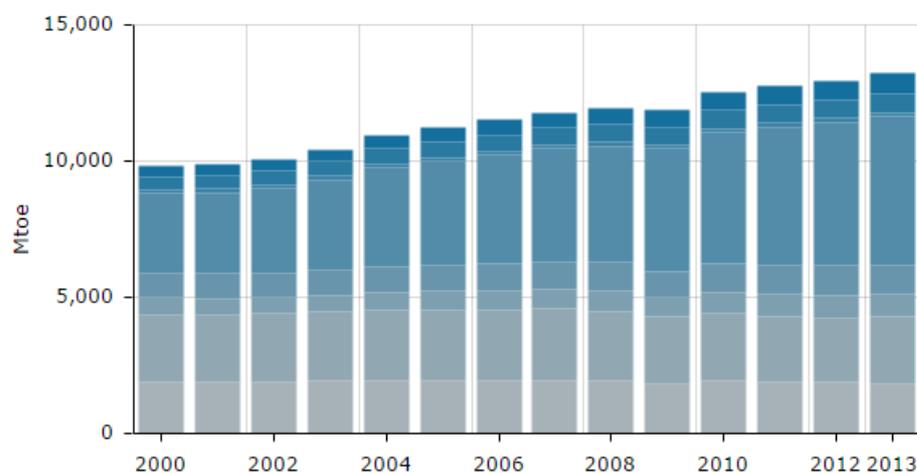


Figura 1.1: Consumo de energia mundial (Adaptado de ENERDATA, 2014).

Segundo o Ministério de Minas e Energia através da Empresa de Pesquisa Energética (2014) o setor residencial obteve 6.2% de aumento de 2012 para 2013, conforme a Tabela 1.1. Atualmente a classe representa 27% de toda energia elétrica consumida no país, superior à média mundial que gira em torno de 20% (MORGAN, 2013).

Tabela 1.1: Consumo de energia elétrica por setor no Brasil (Adaptado de EPE, 2014).

	2009	2010	2011	2012	2013	$\Delta\%$	Part. %
<b>Brasil</b>	384.30	415.68	433.03	448.17	463.33	3.4	100
<b>Residencial</b>	100.77	107.21	111.97	117.64	124.89	6.2	27.0
<b>Industrial</b>	161.79	179.47	183.57	183.47	184.60	0.6	39.8
<b>Comercial</b>	65.25	69.17	73.48	79.22	83.69	5.6	18.1
<b>Rural</b>	17.30	18.90	21.02	22.95	23.79	3.7	5.1
<b>Poder público</b>	12.17	12.81	13.22	14.07	14.60	3.8	3.2
<b>Iluminação pública</b>	11.78	12.05	12.47	12.91	13.51	4.6	2.9
<b>Serviço público</b>	12.89	13.58	13.98	14.52	14.84	2.2	3.2
<b>Próprio</b>	2.31	2.45	3.29	3.35	3.37	0.5	0.7

Quanto ao consumo interno de uma residência, a Figura 1.2 apresenta a distribuição da energia elétrica gasta. Nota-se que mais da metade de toda essa energia concentra-se nos gastos da geladeira (30%) e chuveiro (25%), seguidos pela iluminação (20%). A tarifa média referente ao consumo do setor possui valor aproximado de R\$285,00 por MWh. Esse valor, apesar de ter apresentado um decaimento no último ano, manteve-se crescente

por quase uma década e mantem-se sempre como a tarifa mais cara entre os setores (EPE, 2014).

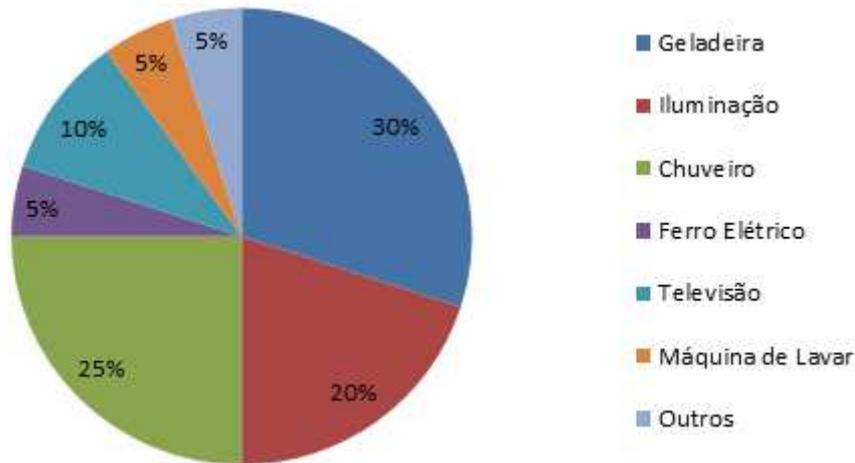


Figura 1.2: Distribuição de consumo elétrico em uma residência (Adaptado de COPEL, 2014).

Paralelamente ao aumento do consumo de energia, o número de usuários de telefones celulares tem crescido em escala mundial. Conforme a projeção apresentada na Figura 1.3, a expectativa é de que 4.77 bilhões de pessoas possuam telefones móveis em 2015. Entre 2012 e 2017, a adoção à essa tecnologia aumentará de 58.2% para 69.4% de toda população (EMARKETER, 2014). Os números demonstram a necessidade de desenvolvimento de projetos que utilizem plataformas disponíveis em celulares, já que estes estão mais acessíveis ao usuário.

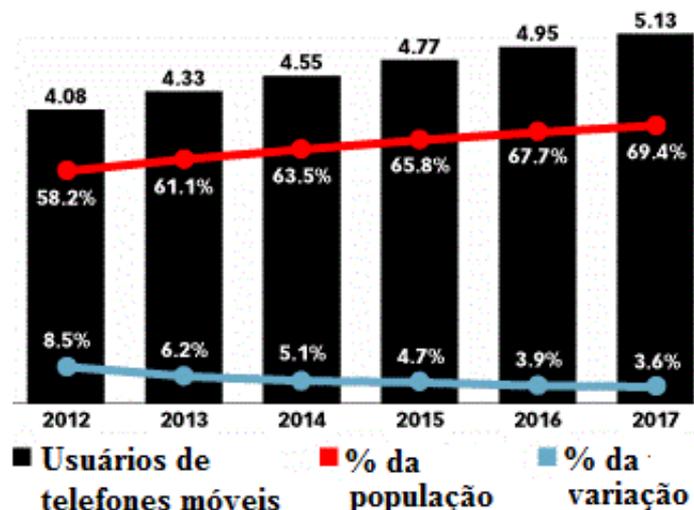


Figura 1.3: Usuários de telefones celulares em todo mundo em bilhões (Adaptado de EMARKETER, 2014).

## 1.2 Motivação

Com o avanço da tecnologia e o fácil acesso a ela em países emergentes como o Brasil, as classes sociais de quase todos os níveis podem ter algum tipo de acesso ao conforto dentro de casa. Segundo a Aureside (2014), a procura de serviços de automação residencial cresceu 300%. Dados apontam que o faturamento anual do segmento cresce em média 30% ao ano, sendo que no país, dois dos cinquenta milhões de residências, contam com algum sistema automatizado (MIRANDA, 2008).

Conciliadas aos números, pesquisas da Aureside indicam ainda que o brasileiro se mostra mais receptivo quanto à aceitação da tecnologia dentro de casa, impulsionado pelos princípios sustentáveis e o desejo de aumentar o conforto (MURATORI, BÓ, 2013). Estima-se que 1.8 milhão de residências estejam aptas a receber a implementação de sistemas de automação (PEGN, 2013.)

A iniciativa deste projeto se baseia no intuito de promover benefícios como conforto, comodidade, segurança, economia e a própria valorização da residência ao usuário. E de certa forma, além da beneficiação pretendida aos moradores, o meio ambiente também é favorecido. A presença da automação está diretamente relacionada à sustentabilidade. É possível reduzir consideravelmente o consumo de energia e água, simplesmente controlando o desligamento de lâmpadas, televisões e refrigeradores (DELGADO, 2013). Segundo Filho (2011), essa redução da energia é estimada em aproximadamente 30%. Isso pode ser garantido através do controle automático de atividades, regulando seu período de funcionamento e intensidade afim de evitar desperdícios.

Além disso, a viabilização do acesso remoto através de redes sem fio torna-se a solução viável à crescente utilização de equipamentos eletrônicos como *smartphones*, *tablets* e *laptops* (REIS, 2012). Os usuários de telefones móveis têm migrado para dispositivos que ofereçam suporte mais acessível com 3G e 4G, sendo que dos 4.77 bilhões de usuários em 2015, 2.03 bilhões (42.6%) já utilizam *smartphones* (EMARKETER, 2014). Atualmente, 6% da população possuem *tablet*, 20% computadores e 22% *smartphones*, conforme a Figura 1.4. Os números apontam a perspectiva da criação de sistemas residenciais que possam ser controlados remotamente, como o proposto.

## 1.3 Objetivos do Trabalho

O presente Trabalho de Conclusão de Curso teve como objetivo principal simplificar

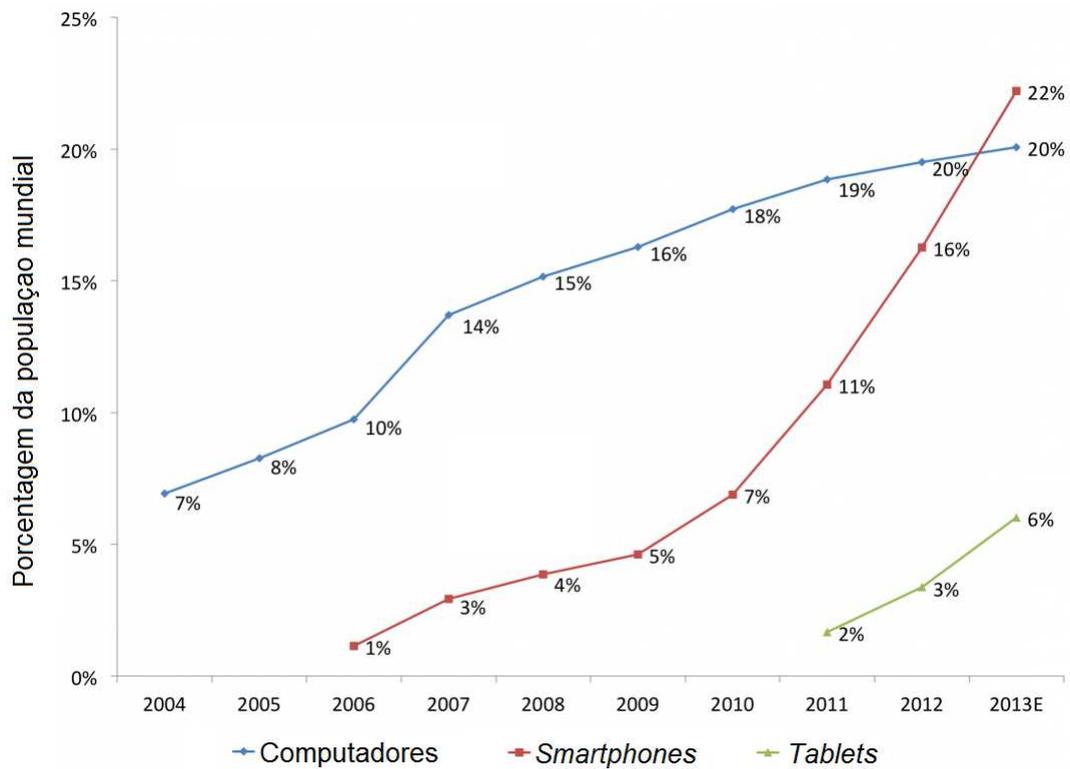


Figura 1.4: Penetração global de dispositivos (Adaptado de HEGGEESTUEN, 2013).

sistemas Domóticos e criar uma interface do usuário que pode ser controlada via rede sem fio por dispositivos móveis.

Como objetivos específicos podem-se destacar:

- Criação de um *website*:

O desenvolvimento de uma interface virtual de controle possibilitará o acesso remoto por plataformas móveis de controle como *smartphones*, *tablets* e *laptops*.

- No que se refere à iluminação:

Ligar e desligar lâmpadas;

Aumentar e diminuir intensidade de lâmpadas;

- Em relação à refrigeração:

Ligar e desligar ventiladores;

Aumentar e diminuir intensidade do ventilador;

## 1.4 Organização do Documento

Esta monografia apresenta o desenvolvimento de técnicas em automação aplicadas ao controle a partir de um dispositivo móvel em sistemas residenciais. Este documento é composto por cinco capítulos:

**Capítulo 1:** É realizada uma introdução do projeto e aos conceitos de domótica. É apresentada a descrição do tema, definição do problema, fatores de motivação, além dos objetivos desejados durante o desenvolvimento da proposta.

**Capítulo 2:** Apresenta a informação teórica necessária. Mostra-se a Domótica sob um ponto de vista tecnológico e o estado da arte, contendo também os protocolos de comunicação de redes, as linguagens de programação envolvidas e os sistemas específicos em questão.

**Capítulo 3:** Contém a metodologia a desenvolvida no projeto. Em materiais e métodos, é fornecido todo o planejamento descrevendo detalhadamente as individualidades de cada sistema, bem como parâmetros de escolha de componentes, circuitos e simulações. Em ferramentas encontra-se o princípio de funcionamento do projeto, o percurso da informação e as explicações necessárias.

**Capítulo 4:** São apresentados os resultados obtidos e as discussões dos fatos analisados em comparação com a fundamentação teórica. Listam-se os principais pontos de destaque verificados no desenvolvimento do projeto e a consequente viabilização.

**Capítulo 5:** No quinto parágrafo encontram-se uma análise conclusiva e uma lista de sugestões de implementações a serem desenvolvidas futuramente no projeto a partir da análise da metodologia atual.

## Fundamentação Teórica

Nesse capítulo são apresentados os conceitos relacionados à ciência da Domótica, baseada na integração de dispositivos automatizados. No decorrer desse capítulo, serão descritos os principais agentes de um sistema Domótico, bem como suas funções e classificações. Além disso, é feita uma revisão da literatura, apresentando o que há de mais moderno no setor, em uma análise dos sistemas, equipamentos, topologias, padrões de comunicação e linguagens de programação envolvidas.

### 2.1 Domótica

A Domótica se fundamenta na integração de sistemas constituintes de uma residência com o objetivo de melhor satisfazer as necessidades básicas de segurança, gestão energética, comunicação e conforto. A etimologia da palavra resulta da junção do latim *Domus*, que significa casa e da palavra robótica, com origem no tcheco *Robota*, que significa controle automatizado de algo. Dessa forma, a domótica, também comumente associada à automação residencial ou automação doméstica, pode ser entendida como conjunto de técnicas e processos que podem ser aplicados simplificando a vida diária das pessoas em qualquer processo domiciliar (RODRIGUES, 2012).

A automação teve seu surgimento ainda no início das civilizações, ausente de uma data que se caracterize como marco. Isso porque qualquer instrumento ou técnica desde então desenvolvida pode ser considerada como facilitadora das atividades do dia a dia. Com a revolução industrial no século XVIII, propiciou-se ainda mais o surgimento da domótica, a partir da mecanização recorrente nos processos envolvidos (TEZA, 2002). Cronologicamente, uma das primeiras grandes contribuições para a domótica foi a invenção das máquinas elétricas de lavar roupas no início do século XX. Algumas tecnologias foram posteriormente cruciais para a otimização da área, como por exemplo a televisão, compu-

tadores pessoais e a própria internet.

A década de 70 é considerado como um marco no histórico da domótica, quando são lançados nos Estados Unidos os primeiros módulos inteligentes de automação. Dessa forma as primeiras redes domiciliares surgiram utilizando a própria rede elétrica como meio de transmissão de dados, como o X10. Considerado pioneiro, o X10 é o protocolo de comunicação internacional de dispositivos eletrônicos. Suas principais características se baseiam no fato de ser modulado diretamente na rede elétrica e ser considerado *Plug and Play*, em referência de sua capacidade de apenas precisar ser conectado para ser utilizado. É simples e barato, fato que justifica sua utilização até em alguns projetos atuais. No entanto, esse padrão além de ser muito lento é muito vulnerável a ruídos e possui um grande decaimento de sinais ao longo das fases de energia. Dessa forma a confiabilidade de transmissão de frames não ultrapassa 70% (MACEDO, 2009).

Desde então, vários outros protocolos foram criados no intuito de estabelecer a comunicação dentro de sistemas residenciais. O EIB, *European Installation Bus*, veio em sequência. Com o intuito de solucionar problemas de complexidade de instalação, os dados são transmitidos através de cabos telefônicos. Apesar da vantagem de poder comunicar os dispositivos utilizando redes já instaladas, sua estrutura descentralizada necessita de várias ligações e que hoje se tornaram inviáveis. Na mesma vertente, há o padrão *LonWorks*. Nele é utilizado um chip nomeado *Neuron-Chip*, que a partir de microcontroladores possibilitava uma comunicação eficiente. No entanto, apesar de apresentar confiabilidade e robustez, o seu alto custo fez com que sua implementação não fosse amplamente difundida na domótica, pois existem recursos semelhantes e mais baratos. O *Smart House* também foi desenvolvido utilizando um sistema microcontrolado, que executa a lógica e gerencia o banco de dados transmitidos. E assim como o *LonWorks*, teve sua disseminação comprometida devido ao preço (LINS, MOURA, 2009). A Tabela 2.1 apresenta uma comparação entre os primeiros protocolos de comunicação presentes na indústria.

Com essa premissa surge então nos dias atuais o conceito de residência inteligente. Ela se difere ao de uma residência comum basicamente por dois motivos: o primeiro porque utiliza dispositivos que desempenham funções especiais que auxiliam na gestão do edifício e o segundo se baseia na presença de conceitos modernos de arquitetura e de construção, o que possibilita o uso mais eficaz da energia. A Figura 2.1 sintetiza esse conceito acerca de uma residência inteligente.

Tabela 2.1: Comparação entre alguns protocolos de comunicação (Adaptado de SENA, 2005).

	<b>X10</b>	<b>EIB</b>	<b>LonWorks</b>	<b>Smart House</b>
Facilidade e rapidez de instalação	Boa	Moderada	Moderada	Difícil
Espansibilidade (número de dispositivos)	256	65536	19	900
Comunicação dos dois sentidos	Sim	Sim	Sim	Sim
Modularidade	Sim	Sim	Sim	Sim
Meios de comunicação usados	Rede elétrica, rádiofrequência, infravermelhos.	Rede elétrica, par trançado, rádiofrequência, infravermelhos.	Rede elétrica, par trançado, rádiofrequência, infravermelhos, cabo coaxial, fibra óptica.	Cabeamento específico (coaxial).
Custo	Baixo	Moderado	Moderado	Elevado

Figura 2.1: Residência Inteligente (RODRIGUES *apud* BARROS, 2012).

Os sistemas residenciais automatizados podem ser classificados em quatro principais grupos (AMORY e JÚNIOR, 2001):

- **Segurança:** trata em proteger o usuário e seus pertences em eventualidades como de invasão, vazamentos e explosões;
- **Conforto:** promoção de lazer e facilidade em tarefas que precisariam de certo grau de esforço tais como abertura de portões a distância e acionamento da iluminação por presença;
- **Energia:** recursos que evitam o desperdício energético, como o desligamento automático de luzes quando não houver pessoas no ambiente e acionamento programado eletrodomésticos fora dos horários de pico.
- **Comunicação:** sistemas que facilitem o trânsito de informações e conectividade como a chamada automática à polícia e bombeiros e interligação entre casas e escritórios.

## 2.2 Mercado

Segundo a Aureside (2014), no Brasil a domótica é uma ciência muito nova, ainda em processo de emergência, mas que demonstra um alto desenvolvimento em um futuro próximo. Em pesquisas mundiais, os brasileiros demonstraram um interesse à tecnologia de automação para residências acima da média mundial (78% dos entrevistados contra 66% na média global) segundo a revista alemã *Germany Trade and Invest* (2013).

No mercado da automação residencial, segundo a Aureside, o Brasil conta com aproximadamente 250 empresas dedicadas a projetos de automação de casas e apartamentos. Isso indica que o cenário tem evoluído de maneira positiva nos últimos anos, impulsionado pela divulgação de novas tecnologias, aceitação do consumidor e acessibilidade dos preços. Segundo Muratori e Bó (2013), o panorama da última década a respeito da evolução da adoção de algumas tecnologias indica o crescimento notável do uso de sistemas de automação nos próximos anos.

O mundo, assim como o Brasil, tem investido consideravelmente no setor de automação residencial e isso tem gerado um superávit positivo. De acordo com a *Strategy Analytics* (2014), o mercado deverá atingir um faturamento de aproximadamente 50 bilhões de dólares em 2014. Perspectivas mostram ainda que esse valor deverá dobrar até 2018 (AURESIDE, 2014).

## 2.3 Projetos

Atualmente, um projeto de automação residencial oscila entre  $R\$8$ mil e  $R\$20$  mil em um imóvel avaliado que custa entre  $R\$400$  mil e  $R\$700$  mil, indicando uma média de 2 a 3% do valor da residência. Com essa faixa de investimento, é possível instalar uma central de gerenciamento controlando sistemas de iluminação, áudio e vídeo por meio de um controle remoto ou celular. É importante considerar também que a central de comando ou sistema supervisorio fixo corresponde ao item mais caro, sendo que uma vez instalado, o cliente pode incorporar outros recursos que variam normalmente entre  $R\$1$  mil e  $R\$2$  mil (FILHO, 2011).

De certa forma, o custo da automatização de uma residência varia de acordo com o grau de complexidade do projeto. Segundo Pradella (2013), com aproximadamente  $R\$7.000,00$  é possível adquirir uma solução básica com uma central de gerenciamento com capacidade de ajustar iluminação, áudio e vídeo. Com mais  $R\$10.000,00$  ou  $R\$12.000,00$  essa central terá condições também de monitorar a residência por meio de câmeras, controlar irrigação do jardim e abrir e fechar cortinas.

Segundo Muratori (2008), na implementação de sistemas domóticos é preciso primeiramente definir projetos básicos e gerais que posteriormente serão associados a projetos específicos, estando todos integrados num ambiente de automação. A Figura 2.2 apresenta essa organização a partir de um fluxo de projeto integrado para uma residência qualquer.

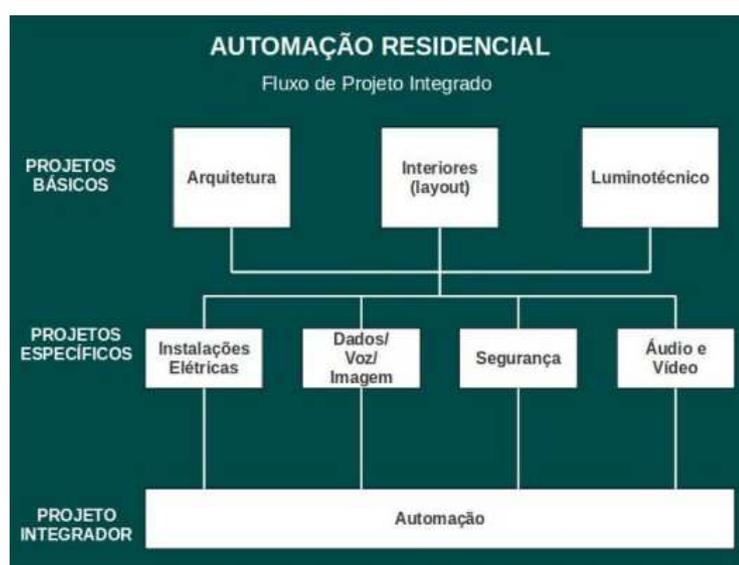


Figura 2.2: Fluxo de projeto integrado em uma residência (MURATORI, 2008).

Na prática, ainda segundo Muratori (2008), as empresas seguem uma sequência de

estruturação bem semelhante à apresentada no fluxograma. Em primeiro lugar se realiza o desenho da estrutura, onde é feito o dimensionamento e distribuição das conexões. Logo após, é analisada uma compatibilização a respeito dos diversos sistemas envolvidos, seguido de um detalhamento da fiação e acabamento, onde muitas vezes é preciso cabos especiais não comumente usados em instalações convencionais. Em seguida, são feitas planilhas de cargas automatizadas, constituída de uma lista completa de todos os equipamentos a serem incluídos paralelamente a uma especificação dos equipamentos identificando quais elementos já foram instalados e os que ainda precisam ser. Por último, é realizado um detalhamento das funcionalidades de todas as funções a serem executadas, em uma espécie de manual do usuário.

## 2.4 Estado da Arte

De um modo geral, as soluções tecnológicas na área da domótica se concentram no desenvolvimento de soluções para segurança, controle de temperatura, iluminação e distribuição de áudio e vídeo. Apenas uma pequena parte da indústria se dedica à busca de ideias inovadoras e abrangentes, como equipamentos e tarefas diretamente ligados à robótica. Esse grupo, frequentemente formado por grandes empresas e fabricantes, no entanto, são os responsáveis pela iniciativa, que vão desde a construção de casas conceito até conjuntos habitacionais completos.

Os protocolos de comunicação correspondem à grande parte dos estudos no desenvolvimento da automação em residências. Grande parte das indústrias ainda utilizam os tradicionais padrões baseados em cabos coaxiais, como o HomePNA (Home Phone Networking Alliance), e em comunicação pela rede de alimentação, como o X10. O HomePNA utiliza os cabos de telefonia e coaxiais já existentes para transmitir os dados de forma compartilhada de computadores, impressoras e outros periféricos. Já o X10, transmite os dados através da modulação direta na rede de energia elétrica. Sua disseminação baseia-se no fato em apresentar um baixo custo de implementação além da facilidade de uso.

Por ser uma área diretamente ligada ao cotidiano das pessoas, a Domótica é uma área que tem sido alvo de várias pesquisas nos últimos anos. Dessa forma, foram selecionados alguns trabalhos com o intuito de mostrar os aspectos que têm sido abordados atualmente.

O mercado tem investido fortemente em tecnologias sem fio e as pesquisas seguem

a mesma vertente. Barbosa (2014), apresenta o desenvolvimento de dispositivos de domótica de baixo custo visando a economia elétrica e segurança residencial. O uso da tecnologia ZigBee no controle de iluminação e tomadas, auxiliam na economia de energia e na segurança.

Atualmente, os projetos de infraestrutura em relação ao cabeamento e definição de equipamentos ainda enfrentam dificuldades quanto à padronização. Devido à diversidade de protocolos de comunicação existentes, foram criadas normas que delimitam as especificações exigidas. As principais normas relacionadas à projetos de cabeamento estruturado são:

- Norma ANSI/TIA/EIA 570 – A: Padrão para cabeamento residencial de telecomunicações;
- Norma ANSI/TIA/EIA 568 – B.1: Requerimentos gerais do cabeamento estruturado;
- Norma ANSI/TIA/EIA 568 – B.2: Componentes UTP do cabeamento estruturado;
- Norma ANSI/TIA/EIA 568 – B.3: Componentes ópticos do cabeamento estruturado;
- Norma ISO/TEC 11801: Especificações de sistemas de cabeamento estruturado;
- Norma Técnica *NBR14565ABNT*: Procedimento básico para elaboração de projetos de cabeamento de telecomunicações para rede interna estruturada.

Para a implementação da automação em residências, é necessário atender as mais diversas particularidades. Apesar de existirem diferentes diretrizes para os projetos domóticos, essas normas técnicas como a ANSI/TIA/EIA-570-A, para cabeamento de telecomunicações em residências, definem então os diversos subsistemas presentes, por exemplo, no cabeamento, tipo de cabos, conectores, espaçamento e encaminhamento de distribuição.

Em relação aos recursos e equipamentos disponíveis, o mercado tem se mostrado inovador e repleto de soluções para as mais variados sistemas.

### 2.4.1 Iluminação

A aplicação da domótica em sistemas de iluminação atribui inúmeras características à residência, que vão desde a possibilidade de destaques dos detalhes arquitetônicos à

criação de cenários específicos para cada momento. Além apenas do acionamento, as lâmpadas podem ter sua intensidade controlada e tempo de funcionamento determinado de acordo com sua solicitação ou não, entre várias outras funções. O controle de iluminação consegue agregar um considerável consumo de energia à residência. Com presença da automação é possível diminuir os gastos a partir do acendimento de lâmpadas segundo horários previstos e programáveis, de acordo com a estação do ano, tipo de compartimento e a previsão de horas de ocupação (PESSOA, GHISI, LAMBERTS, 2013).

Manuais técnicos como da *Philips*, indicam que são basicamente cinco as maneiras de controle de iluminação atualmente, sendo elas: liga e desliga manual, controle manual do fluxo, liga e desliga automático, controle gradativo automático e controle automático de fluxo.

No que se refere à lâmpadas, existem várias características que são levadas em consideração no estudo da luminotécnica, sendo elas:

- Fluxo Luminoso: quantidade de energia radiante capaz de sensibilizar o olho humano. Essa grandeza possui unidade de medição em *Lúmens*, correspondente à essa energia captada por segundo;
- Eficiência Energética: razão entre fluxo luminoso emitido e a potência elétrica consumida. Em outras palavras, uma maior eficiência significa mais luz com menor consumo de energia. Sua unidade é o *Lúmen/Watt*;
- Intensidade Luminosa: é o fluxo luminoso emitido em uma única direção. A fórmula que expressa essa grandeza é a *Candela* e corresponde à razão do fluxo sobre  $\pi^4$  esferoradianos;
- Iluminância: razão entre fluxo luminoso emitido e a área da superfície iluminada a certa distância. A unidade é o *Lux*;
- Luminância: razão entre intensidade luminosa em determinada direção e a superfície aparente para o observador. A unidade dessa grandeza é dada em *Candela/m<sup>2</sup>*. A luminância é o que produz a sensação de clareza, sendo que terá valor máximo quando o olho estiver na perpendicular da superfície;
- Índice de reprodução de cores: índice utilizado para mensurar a qualidade de reprodução de cores de um objeto sob a incidência de uma luz artificial em comparação com condições ideais de luz natural. Sua unidade é adimensional variando de 0 a 100%;

- Temperatura de Cor: expressa a aparência da cor da luz, sendo que quanto mais alta sua temperatura, mais branca é. Sua unidade é o *Kelvin*, onde luzes quentes possuem valor em média de 3000 e as frias de 6000;
- Vida Útil: números de horas decorrido quando se atinge 70% da quantidade de luz inicial em recorrência da diminuição do fluxo luminoso. Existe ainda a definição de vida média, equivalente à média aritmética do tempo de sua duração.

Uma grande variedade na solução de lâmpadas encontra-se ao alcance do consumidor. As lâmpadas em si podem ser agrupadas em três famílias básicas, sendo elas: incandescentes (comuns, halógenas e halógenas dicróicas), fluorescentes (circulares, tubulares e composição integrada) e de descarga (vapor metálico, de mercúrio e de sódio). No mercado existem ainda lâmpadas especiais que, após sua conexão em um bocal comum, é possível as ligar, desligar e até mesmo controlar a intensidade ou cor. Lâmpadas acionadas através da energia solar, que dispensam o uso de rede elétrica, também já são uma realidade (PETRILLI, SILVA, SILVA, 2012).

De todas as soluções presentes no mercado, as lâmpadas LED (*Light Emitting Diode* ou Diodo Emissor de Luz) tem se mostrado as mais promissoras nas aplicações e maneiras de iluminar ambientes e objetos. As principais vantagens se baseiam em maior vida útil, baixa manutenção, baixo consumo, alta eficiência, não emissão de raios ultravioletas ou radiação infravermelha, além de alta resistência à impactos e vibrações (BLEY, 2012). A Figura 2.3 apresenta um comparativo de parâmetros entre o LED e outros tipos de lâmpadas.

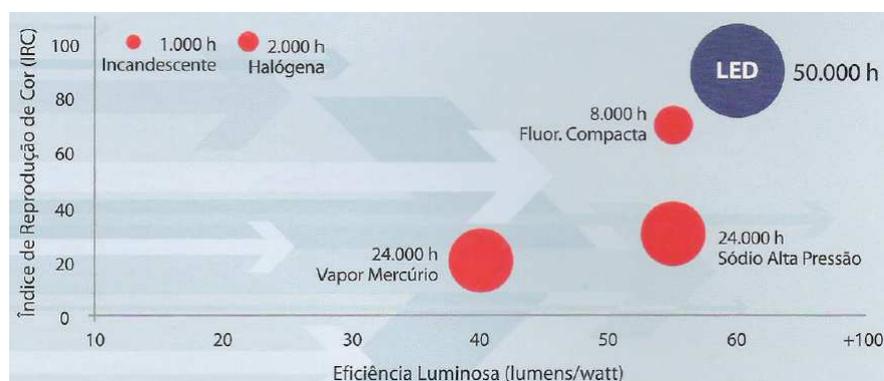


Figura 2.3: Eficiência, IRC e vida útil de lâmpadas (BLEY, 2012).

Nota-se que o LED é considerado como tipo de fonte luminosa bem próxima ao ideal. Isso porque esse tipo de lâmpada apresenta IRC e eficiência luminosa bem próximos de 100%, o que indica um alto nível de fidelidade à luz natural. Além disso, conforme

informado, o LED possui vida útil bem maior que os demais tipos, variando em uma escala de 20000 a 50000 horas.

O estudo de caso realizado por Greggianin *et. al* (2013) apresenta uma comparação de custo entre lâmpadas incandescentes, fluorescentes compactas e o LED. As Tabelas 2.2 e 2.3 contém a análise de custo comparativo entre os três tipos de lâmpadas.

Tabela 2.2: Análise de custos comparativos entre três tipos de lâmpadas (1).

Modelo	Potência (W)	Horas	Vida Útil (horas)	Lâmpadas
Incandescente	40,70	35.000	750	47
Fluorescente	11,97	35.000	8.000	5
LED	5,81	35.000	35.000	1

Tabela 2.3: Análise de custos comparativos entre três tipos de lâmpadas (2).

Modelo	Energia mensal	Custo unitário	Custo reposição	Custo total
Incandescente	R\$683,65	R\$2,40	R\$112,80	R\$796,45
Fluorescente	R\$201,06	R\$12,90	R\$64,50	R\$265,56
LED	R\$97,59	R\$79,5	R\$79,50	R\$117,09

Existem diversos dispositivos luminotécnicos disponíveis no mercado atualmente. São dimmers, fotocélulas, minuterias, sensores de presença, temporizadores, sequenciadores, controles remotos e os simples interruptores, como mostra a Figura 2.4.



Figura 2.4: Diagrama com os diversos tipos de controle de iluminação (Adaptado de NISKIER, 2000).

- *Dimmer*:

Dispositivo que permite variar o fluxo luminoso emitido por fontes de luz artificiais. Atua exclusivamente sob as lâmpadas de funcionamento incandescente, já que

os equipamentos auxiliares das fluorescentes como reatores eletromagnéticos não respondem à variação de luminosidade. Tendo em vista que a vida útil de uma lâmpada incandescente é de 1000 horas, uma sub tensão de apenas 5% aplicada acarreta à duplicação desse valor, enquanto 5% de sobre tensão pode ser responsável por diminuir esse tempo pela metade. Além disso, o filamento dimerizado pode sofrer diminuição gradativa da temperatura de cor, promovendo alteração nas cores do ambiente;

- *Sensores:*

Equipamentos que ativam ou desativam lâmpadas de acordo com o sinal recebido. Os tipos mais comuns são:

sensores de presença: são ativados pela aproximação de pessoas, sendo do tipo infravermelho ou ultra-sônico;

sensores de luz: captam a luminosidade do ambiente, basicamente constituído de semicondutores que habilitam corrente ao ser iluminado;

sensores magnéticos: acionados quando há qualquer tipo de vibração da porta em que estão implementados;

- *Fotocélula*

Análoga à um sensor de luz, são dispositivos sensíveis a luz possuindo ajustes internos indicando valores mínimos de luz para que os contatos fiquem acionados até que o nível fique abaixo do estabelecido;

- *Temporizador:*

Equipamento para ligar e desligar lâmpadas em horários pré-programados;

- *Minuteria:*

Equipamento eletrônico micromotorizado que desliga lâmpadas após um período predeterminado. Quando seu interruptor é acionado, um cronômetro dispara e permite que a corrente seja interrompida automaticamente segundos depois;

- *Interruptor:*

Dispositivo que pode interromper ou reestabelecer a continuidade num circuito elétrico, a partir da ação sob o condutor fase da instalação. É o equipamento mais comum em iluminação residencial.

- *Controle Remoto:*

Produto composto por unidades emissoras e receptoras que permitem o controle distante.

### 2.4.2 Refrigeração

Os sistemas de refrigeração têm como finalidade a climatização homogênea de ambientes. Termostatos são projetados a fim de se obter a máxima precisão e desempenho garantindo o controle da temperatura com alta precisão. As empresas têm buscado a implementação de um sistema de ar condicionado em que toda a casa resfrie na mesma velocidade. Essa facilidade evita que o morador experimente quedas bruscas de temperatura entre os cômodos. Além disso, com uma base de comando central há uma redução considerável no custo de instalação e manutenção, uma vez que não há mais a necessidade de instalar diferentes equipamentos em cada parte da casa separadamente.

### 2.4.3 Segurança

O sistema de segurança em residências se estabelece a partir do monitoramento e vigilância eletrônica. No combate às invasões, o uso de alarmes é bastante disseminado. Alarmes são basicamente sensores que pode ser de presença, temperatura e de perímetro (a laser). Quando acionados podem emitir sinais sonoros, enviar mensagens para os proprietários ou até mesmo para a polícia, além de poder acionar tarefas pré-programadas que automaticamente fecharão janelas e portas a fim de tentar deter o infrator. Além disso, existe o CFTV (Circuito Fechado de TV). Por meio de câmeras monitoradas remotamente, o morador têm acesso visual completo dos domínios de sua propriedade podendo identificar com facilidade qualquer movimento nas proximidades. Nesse contexto, câmeras com visão noturna e sensores de temperatura são recursos disponíveis no mercado.

### 2.4.4 Entretenimento

O entretenimento pode ser considerado um dos setores mais característicos da nova era na domótica. A presença da tecnologia de automação em equipamentos de áudio (home-theater) e vídeo (DVD, TV, Cd player, videokê, etc) garante uma enorme possibilidade de conforto ao usuário. A linha de pesquisa nessa área considera ajustes automáticos realizados pelo próprio controlador. Dessa forma, por exemplo, se a televisão for ligada as luzes da sala reduzem sua intensidade de forma a garantir um ambiente personalizado. Além disso, com caixas de som propositalmente posicionadas, pode-se escolher e melhor música para cada situação, reproduzindo inclusive os sons ambientes.

## 2.5 Redes

O desenvolvimento do avanço tecnológico tem possibilitado um avanço, tanto na criação de novos equipamentos e sistemas automáticos, quanto na infraestrutura de comunicação de dados. No que se refere à organização do cabeamento estruturado, existem determinadas topologias de distribuição de sistemas. Segundo Marin (2009), são três as principais configurações comumente utilizadas: em estrela (Figura 2.5a), anel (Figura 2.5b) e barramento (Figura 2.5c).

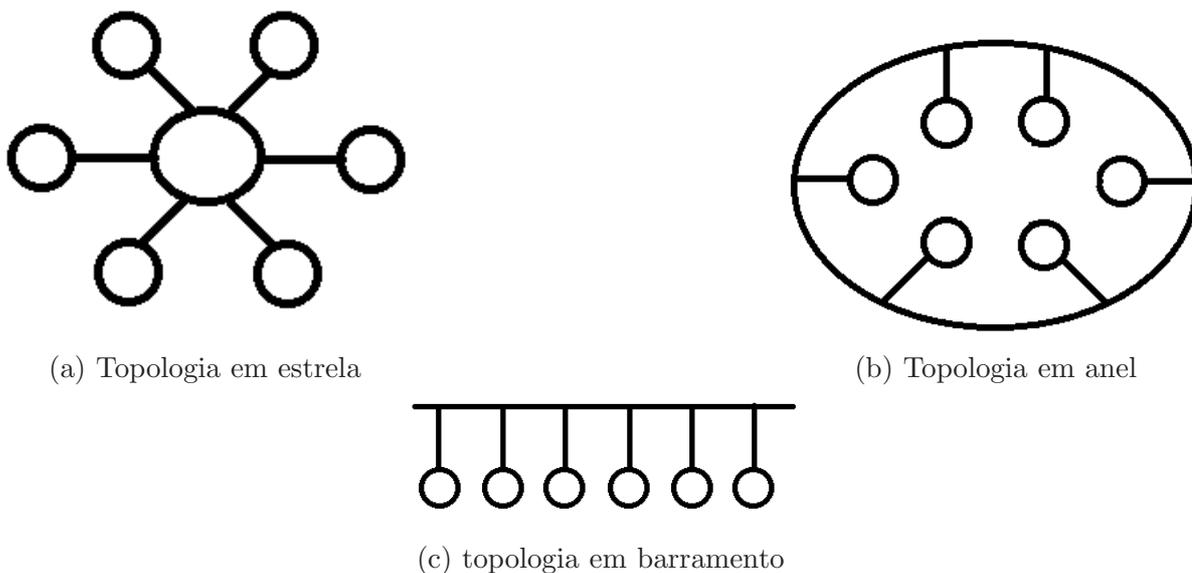


Figura 2.5: Topologias de cabeamento de comunicação em residências (Adaptado de MARIN, 2009).

Pela necessidade de uma interligação entre os diversos sistemas de uma casa que os permitam estar integrados e operar em conjunto, a topologia física comumente utilizada é a topologia em estrela. Considerada padrão de distribuição de cabos e característica em projetos de automação residencial, essa configuração apresenta os periféricos dispostos em torno de um elemento central, responsável pelo controle e gerenciamento do sistema, conforme a Figura 2.6. Essa disposição estrutural com hierarquia bem definida garante inúmeras características que justificam seu uso.

A premissa de integração apresenta melhores condições de operabilidade e possibilidade de aprimoramento. Segundo Muratori e Bó (2013), a flexibilidade do sistema sugere uma manutenção mais fácil e prática quando se comparada com as topologias em barramento e anel, pelo fato de já considerar um elemento central de comando.

Em um projeto de domótica, os equipamentos comunicam com a central de comando

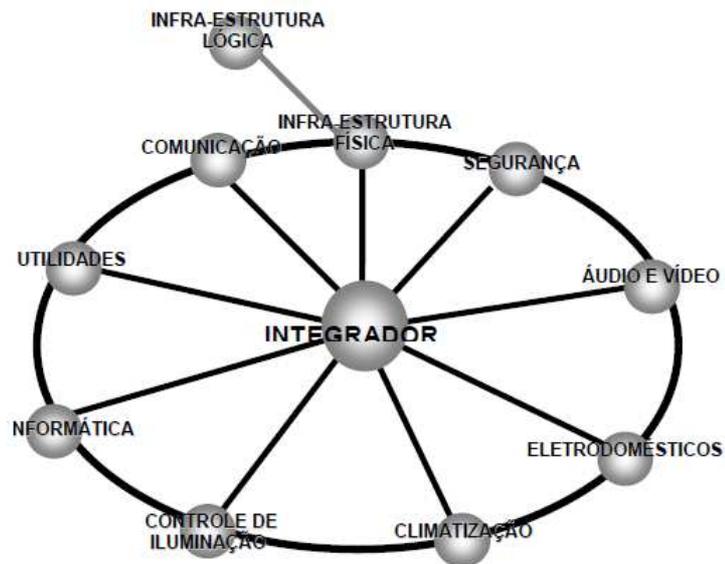


Figura 2.6: Integração de sistemas residenciais em topologia tipo estrela (Adaptado de DIAS e PIZZOLATO, 2009).

e processamento seguindo um determinado protocolo, cuja transição de dados será específica de acordo com o padrão escolhido (TANENBAUM, 2003). Nesse contexto, redes domiciliares são sistemas de comunicação que promovem a interconexão de dispositivos em curta distância (SENA, 2005). A Figura 2.7 apresenta um fluxograma indicando os principais tipos de redes e sua função.

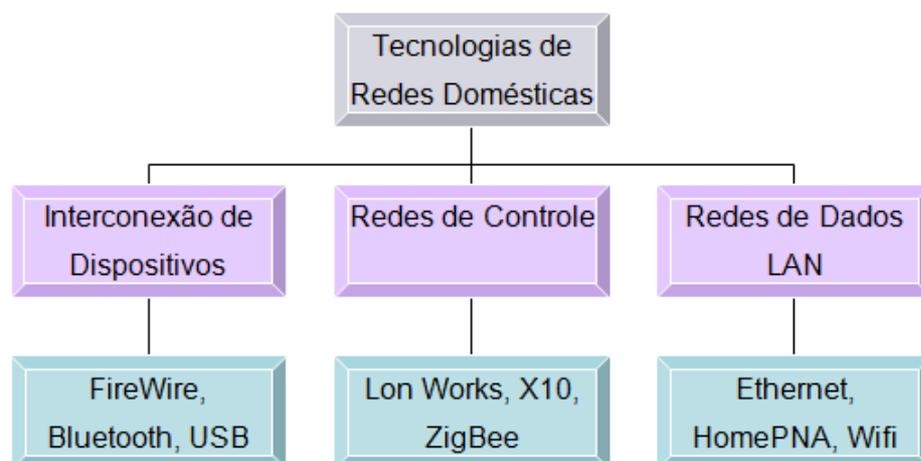


Figura 2.7: Tecnologias de redes domésticas (Adaptado de SENA, 2005).

As tecnologias de rede aplicadas no âmbito domiciliar podem ser classificadas basicamente em três grupos, que diferem entre si de acordo com a forma como serão tratados os dados. Eles podem ser usados para a interconexão de dispositivos, controle de equipamentos ou transmissão de dados. É possível notar que dentre os diversos protocolos

de comunicação, cada um deles é adequado para uma função específica. De acordo com Macedo (2009), alguns parâmetros são considerados importantes para avaliar e selecionar a melhor escolha para determinada situação:

- Estado atual do padrão;
- Características da rede;
- Processo de configuração oferecido;
- Suporte de interoperabilidade.

### 2.5.1 Wi-Fi

Wi-Fi é a abreviação de Wireless Fidelity e se refere ao protocolo de comunicação sem fio baseado no conjunto de padrões 802.11 desenvolvido pelo *Institute of Electrical and Electronic Engineers*: IEEE. Consiste em uma tecnologia para redes locais (WLANs) sem a necessidade do uso de fios com o intuito principal de prover com segurança a interoperabilidade de dispositivos móveis. Dentre suas várias versões, pode operar na faixa de  $5GHz$  e transferir dados a  $54Mbps$  aproximadamente. Possui boa margem de alcance, em torno de 100 metros, e ótima taxa de transmissão, mas deixa a desejar em seu alto consumo de energia (ZEINDIN *et al.*, 2003).

### 2.5.2 Bluetooth

O *Bluetooth* é outra tecnologia para comunicação sem fio para uso de redes pessoais (PANs e WPANs). Esse padrão tem como especialidade conectar até 255 dispositivos em uma distância que pode variar até 100 metros. Opera em uma faixa de frequência de  $2.4GHz$  e é capaz de realizar transferência de dados até  $1Mbps$ . Pode ser considerado barato, robusto e de baixo consumo, mas apresenta uma baixa qualidade de transmissão, o que delimita a sua utilização (JÚNIOR, 2012).

### 2.5.3 ZigBee

O padrão *ZigBee* é uma tecnologia de comunicação sem fio baseado no protocolo IEEE 802.15.4 para uso sem redes pessoais sem fio (WPLANs). Foi estruturado com o intuito de monitoramento e sensoriamento de dispositivos que sejam alimentados por baterias, sensores e atuadores em algumas dezenas de metros. Pelo fato das redes serem capazes de formar malhas é possível estabelecer um grande alcance ponto a ponto, apesar do pequeno alcance de cada nó individualmente. Apresenta um baixo consumo de energia, onde a vida

útil de sua bateria pode durar até 1000 dias. No entanto, umas de suas desvantagens é a baixa transferência de dados, tipicamente abaixo dos  $100Kbps$ .

### 2.5.4 Comparação das redes sem fio

Analisando as possíveis formas de comunicação *wireless* disponíveis no mercado, em comparação entre as tecnologias, o *ZigBee* possui foco de aplicação em controle e monitoramento, destacando-se em parâmetros como alcance superior a 100 metros e duração da bateria de até 1000 horas. O *Wi-Fi* também se apresenta como solução alternativa, já que garante uma largura de banda de aproximadamente 11000 Kbps, como demonstra a Tabela 2.4 (PEREIRA, 2007).

Tabela 2.4: Comparação entre os três padrões mais comuns de tecnologias de comunicação de redes sem fio (PEREIRA, 2007).

<b>Padrão (Nome de Mercado)</b>	<b>802.15.1 (Bluetooth)</b>	<b>802.11b (Wi-Fi)</b>	<b>802.15.4 (ZigBee)</b>
Foco de aplicação	Eliminação de cabos	Web, e-mail, vídeo	Controle e monitoramento
Largura de banda (Kbps)	1000 – 3000	11000	20 – 250
Alcance ( <i>m</i> )	100+	100+	20 – 70
Número máximo de nós	7	32	2 * 64
Duração da bateria (dias)	1 – 7	0.5 – 5	100 – 1000+
Consumo (transmissão)	150mA	300mA	30mA
Memória necessária	50+	70+	40

### 2.5.5 Ethernet

O protocolo *Ethernet* é um dos padrões mais utilizados ao acesso à internet usado para a transferência de dados multiponto usando cabos coaxiais. Possui altas taxas de comunicação digital com baixa vulnerabilidade a ruídos, garantindo-lhe uma alta confiabilidade. A partir de sua codificação *manchester*, que impossibilita o uso de protocolos simultâneos

no mesmo barramento, a frequência de comunicação acontece a  $125MHz$  com frames entre 72 e 1526bytes. Cada frame possui 7bytes de sincronização do receptor (preâmbulo), 1byte de início de frame (Sof), 12 bytes para endereçamento único necessários para qualquer aparelho se comunicar na rede (de envio e destino), 2 bytes de comprimento de pacote, uma fixa de 46 a 1500bytes destinados a payload e 4bytes de trailer CRC. A Tabela 2.5 apresenta a composição de um frame unitário.

Tabela 2.5: Frame unitário do protocolo Ethernet (Adaptado de MIZUSAKI, 2009)

Preâmbulo	SoF	Destino	Envio	Tamanho	Payload	Trailer
7Bytes	1Byte	6Bytes	6Bytes	2Bytes	46 – 1500Bytes	4Bytes

O alcance máximo gira em torno de 100 metros e uma de suas principais vantagens diz respeito ao cabeamento estruturado Cat5, onde vários dispositivos podem ser conectados através de hubs, criando sub-redes e evitando a colisão de dados. A Figura 2.8 apresenta o par de fios trançados Cat5 usado no padrão *Ethernet*.

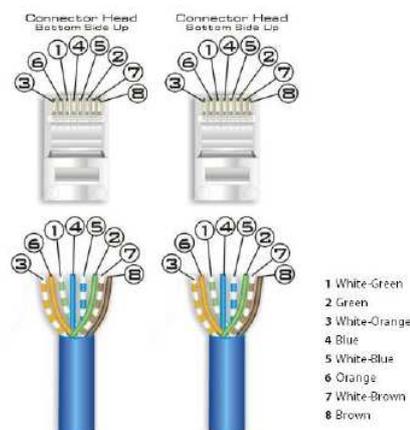


Figura 2.8: Cabo Cat5 usado no padrão Ethernet (MIZUSAKI, 2009).

## 2.5.6 A Internet

A internet pode ser definida como sistema global de redes de computadores que interliga milhões de dispositivos computacionais espalhados por todo o mundo. Esses dispositivos são geralmente computadores pessoais, estações de trabalho ou servidores, que armazenam e transmitem informações. A expansão dessa tecnologia demonstra ainda o quanto a população está se tornando público para a domótica. A tabela 2.4 apresenta o

crescimento percentual do número de usuários da internet ao redor do mundo.

Os principais protocolos utilizados para o acesso à internet são o TCP (*Transmission Control Protocol*) e o IP (*Internet Protocol*). De certa forma, o envio de dados pela internet pode ser organizado em quatro camadas de informações, sendo elas a camada de aplicação, transporte, de rede e camada de enlace. A Figura 2.9 apresenta a pilha de protocolos da internet.

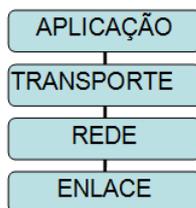


Figura 2.9: Pilha de protocolos da Internet (TANENBAUM, 2003).

A primeira delas, a camada de aplicação, é responsável por gerenciar e servir ao usuário todas as ferramentas e sistemas destinados a ele, tendo como função servir de terminal para as operações que ocorrem em uma rede. Em outras palavras, existe a comunicação entre os programas e o padrão de transporte. Por exemplo, quando um endereço de um website é digitado no navegador para visualizar uma página na internet, uma das subcamadas da aplicação, como o subprotocolo HTTP, será acionado e por isso todas as páginas começam com 'http://' (GUERBER, 2008).

Já a camada de transporte controla a comunicação entre os dispositivos conectados à rede. O emissor recebe as mensagens e as fragmenta e as encapsula em unidade de dados de protocolo, conhecidos como segmentos que são repassados à rede e assim incorporados aos datagramas. Nesse ponto, é importante a definição do TCP. Esse protocolo oferece a multiplexação e de multiplexação de aplicações assim como um mecanismo de detecção de erros, o que garante uma transmissão bem sucedida. Seu papel no processo de transmissão de dados consiste em conectar o host emissor e receptor, fazendo com que eles possam trocar informações. Sua principal função consiste em manipular os dados recebidos pela camada de aplicação e dirigi-los para o buffer de envio, onde serão fragmentados e enviados (CANTÚ, 2013).

A terceira camada, conhecida como camada de rede, tem a função de transmitir a informação pelas redes locais, em uma espécie de interconexão. Nesse ponto ocorre a determinação da rota que será tomada, desde o ponto de origem até o destino a partir

de algoritmos de roteamento e também o chaveamento de datagramas. Dessa forma, o protocolo IP é responsável por estabelecer a rota pela qual seguirá cada pacote de dados na malha de roteadores da internet. Essa ação é tomada tendo como base o endereço de destino, conhecido como endereço de IP (CARNEIRO, BRAGA, JÚNIOR,1999).

O roteamento é a principal função do protocolo IP. Consiste no processo de encaminhar os datagramas entre as redes encontradas. O protocolo sempre assume que um host é capaz de enviar esses dados para outro host e se isso não for possível entra em cena o roteador. São equipamentos que possuem uma interface de rede baseada em uma tabela de roteamento cujos dados informam de onde e para onde ocorrerá o envio de informação.

Por último se encontra o protocolo de enlace que consiste basicamente no canal de comunicação entre os equipamentos, chamados de nós, ao longo de uma rota. Sendo assim, para haver o envio de uma datagrama contendo os dados de determinada informação desde sua origem até o destino final, o percurso passa por cada enlace individual entre os diversos nós. Portanto, após todo esse processo, a informação chega ao ponto final. Em comparação com o modelo OSI, existe ainda uma camada física. Nesse ponto, encontra-se a parte do hardware em si onde é possível estabelecer a conexão com outros equipamentos e transmitir a informação adiante. Essa etapa trata-se do uso das tecnologias para conexões, tais como *Ethernet* e roteador.

## 2.6 Comunicação Serial

A troca de dados entre dispositivos eletrônicos ocorre segundo determinadas interfaces de comunicação. Existem basicamente duas formas de se conectar equipamentos microprocessados a outros equipamentos, sendo elas pela forma paralela ou serial. Na interface paralela, todos os bits de um pacote de dados são transmitidos simultaneamente, exigindo então que cada canal possua diversos fios. Esse tipo de comunicação possui maior rapidez e simplicidade das interfaces envolvidas, implicando porém no uso excessivo de fios e maior susceptibilidade à ruídos e perda de sincronismo (CUGNASCA, HIRAKAWA,2006).

Por outro lado, em uma interface serial os dados são transmitidos sequencialmente no tempo, em um único meio físico. Isso garante a redução do número de fios do sistema fazendo com que seja possível uma comunicação a longas distâncias. Em consequência a troca de informações possui menor velocidade e as interfaces envolvidas tem um significativo grau de complexidade aumentado.

As características a serem levadas em conta em uma interface serial são:

- Taxa de transmissão (*Baud rate*): medida de velocidade de comunicação, relativa ao número de bits transmitidos por segundo;
- Bits de dados (*Data bits*): medida do número de bits de dados vinculados em uma transmissão;
- Bits de parada (*Stop bits*): sinalizadores de fim de comunicação para cada pacote;
- Paridade: verificação de erro.

A comunicação serial pode ser então subdivida em dois tipos, síncrona e assíncrona. Na comunicação síncrona, emissor e receptor compartilham um sinal de *clock* de mesma frequência e fase de forma a manter uma sincronia durante a transmissão. Além dos próprios dados transmitidos, a cada bloco de informação, o emissor envia um sinal de *strobe* contendo caracteres de sincronismo e um *check* para verificação de erro. Esse tipo de comunicação serial possui então alto índice de eficiência porém com custo elevado, pois há a necessidade de implementação de circuito e cabeamento para compartilhamento de *clock*.

O protocolo I2C (*Inter-Integrated Circuit* ou Circuito Inter Integrado) é um exemplo de padrão que faz uso de transmissão de dados de forma síncrona. Inventado pela Philips na década de 90, esse protocolo é utilizado para conectar periféricos de baixa velocidade à placa mãe e microcontroladores em geral. O seu barramento é composto por apenas dois fios, SDA e SCL, além de alimentação, tipicamente 3V ou 5V (CAMARA, 2013).

Já no que se refere à comunicação serial assíncrona, os dados são transmitidos intermitentemente em um fluxo estável sem a presença de um sinal de *clock*. Cada bloco de dados inclui então bits de informação de controle, conhecidos como *flags* de forma que o microcontrolador possa identificar quando começa ou acaba o bloco de dados, além da sequência correta da informação transmitida. Esse tipo de comunicação possui custo de implementação mais barato, já que não necessita de um *clock* externo de sincronismo. Além disso, por não depender desse sinal constante, transmissor e receptor não precisam ser sincronizados todo o tempo, permitindo então uma taxa variável de bits, transmitidos apenas quando exigido.

O protocolo UART (*Universal Asynchronous Receiver Transmitter* ou Transmissor e Receptor Assíncrono Universal) é um exemplo de padrão que transmite os dados de forma

assíncrona. É um componente essencial nos subsistemas de comunicação, com a capacidade de converter dados paralelos em seriais e vice versa. O seu barramento é composto pelas entradas e saídas dos dados bem como os fios de alimentação TTL, ou seja, 3V ou 5V (FREESCALE, 2011).

## 2.7 Linguagens de Programação

Uma linguagem de programação é um conjunto de regras sintáticas e semânticas utilizadas para definir instruções que serão associadas de forma a compor um programa para resolução de uma tarefa específica. As linguagens de programação podem ser classificadas em níveis de acordo com a proximidade entre a linguagem interpretada pelo processador e o usuário. Nesse âmbito, existem as linguagens de baixo e alto nível. As de baixo nível são analisadas diretamente pelo sistema, gerando resultados rápidos, mas com um alto nível de complexidade, o que as torna difíceis e incômodas de serem manipuladas.

Por outro lado, as linguagens de alto nível são mais fáceis de trabalhar, já que os comandos são representados por palavras, geralmente em inglês, selecionadas para facilitar a memorização e a lógica. O código fonte não é interpretado diretamente pelo computador, pelo fato de não estar em base binária, o que justifica a utilização de um compilador que realizará essa tradução ao processador.

As principais etapas de desenvolvimento de programas são:

1. Codificação  
Geração do código fonte, no qual é realizada a escrita de instruções baseadas em linguagens de programação em um editor;
2. Tradução  
Geração do código objeto através da tradução do código fonte para o sistema de numeração binária por compiladores;
3. Incorporação  
Rearranjamento de todas partes isolas do código inicial em um código executável através de editores de ligação;
4. Verificação  
Busca de erros na rotina de execução do programa por meio de depuradores e debuggers.

No que se refere ao processo de execução de programas, é preciso sempre que o programa seja transferido para a memória principal. A maioria dos programas fica armazenada em disco, no entanto a CPU não os pode executar diretamente de lá. Para isso, o sistema operacional permanece o tempo todo ativo na memória principal aguardando o comando de execução de programas. Quando isso ocorre, o próprio SO procura no disco o arquivo referente ao comando e o carrega na memória RAM. Nesse ponto então o controle da CPU é transferido para o programa até a finalização de sua rotina. É importante ressaltar que no momento em que o computador é ligado, o sistema operacional está armazenado no disco. Nesse momento a memória RAM está vazia e então a CPU executa o BIOS, responsável por carregar o SO, processo conhecido como BOOT (WILLRICH, 2015).

### 2.7.1 HTML

HTML é uma abreviação para *Hypertext Markup Language* - Linguagem de Marcação de Hipertexto - desenvolvida originalmente por Tim Berners-Lee na década de 90. Uma página em HTML é um conjunto de comandos que determinam a publicação de conteúdo (texto, imagem, vídeo e áudio) na *World Wide Web*, ou simplesmente WWW. É baseada no conceito de hipertexto, que são conjuntos de elementos ou nós ligados por conexões. As informações são interligadas umas às outras de uma forma complexa e não sequencial através de chaves relacionadas, criando uma grande rede de informação.

Para a distribuição da informação de uma maneira global, é necessária uma linguagem que seja entendida universalmente por diversos meios de acesso. O HTML propõe ser essa padronização de linguagem por vários fatores. A começar por sua interpretação, que se apresenta independente de plataformas e/ou meios de acesso. Isso garante que o código possa ser lido por todos os browsers (navegadores) disponíveis, seja qual for sua versão. Existem algumas versões que possuem incompatibilidade, no entanto essa fração é mínima quando se comparado com as que são compatíveis. Além disso, por ser considerada uma interpretada, a partir do momento em que o arquivo é salvo, o mesmo já pode ser aberto pelo navegador, já que é o software é capaz de determinar se o documento foi escrito conforme os padrões da linguagem. O navegador consegue então apresentar se o que está na tela é realmente o que foi desejado quando o código foi escrito.

O documento em HTML é um arquivo composto por elementos da tabela ASCII que pode ser escrito através de qualquer editor de texto comum, possuindo a extensão .html ou .htm. Os comandos de formatação da linguagem são conhecidos como tags e todas

possuem o mesmo formato: começam com um sinal de menor < e acabam com um sinal de maior >. Cada elemento é então formado por uma tag, cada qual com sua função e que pode ser alterada com atributos e valores. A estrutura básica de um código é composta por três tags (*html*, *head* e *body*).

### 2.7.2 C e C++

C é uma linguagem de programação compilada de propósito geral para implementação de sistemas. Desenvolvida em 1972 por Dennis Ritchie, sua estrutura permite uma compilação que forneça acesso de baixo nível à memória e baixos requerimentos de hardware. É considerada um linguagem de alto nível, quando se comparada, por exemplo, com o *Assembly*, uma vez que facilita a criação de programas extensos com menos erros a partir de uma complexidade consideravelmente simples. Por apresentar uma proximidade ao código de máquina, a linguagem C é muito utilizada para desenvolvimento de sistemas e softwares de base, bem como programas de computadores. Além disso, sua lógica é baseada principalmente em estruturas, ponteiros e vetores.

Analogamente ao C, encontra-se a Linguagem C++. Criado por Bjarne Stroustrup em 1980, essa nova solução atribui características de orientação à objetos e foi desenvolvida sob a base C, pelo fato de propiciar o uso genérico, rápido e portátil para diversas plataformas. Sendo assim, o C++ possui a mesma base de desenvolvimento da linguagem C adicionada de novos recursos, como a comunicação de sistemas distribuídos ou de memória compartilhada e o controle de seus *drivers*. Considerada uma certa evolução ao C, o C++ suporta vários paradigmas. Além da programação estruturada que permite a compreensão da lógica sob a visão de sub-estruturas envolvendo a criação de várias funções pequenas e simples dentro de um programa, ela suporta outros paradigmas como programação procedural, genérica, abstração de dados e a programação orientada à objeto. Dentre elas, a mais utilizada é programação orientada à objetos, ou OOP (*Object Oriented Programming*).

### 2.7.3 Java

Java é uma linguagem de programação orientada a objeto na década de 90 pela empresa *Sun Microsystems*. A sua sintaxe deriva-se do C++ porém com um modelo bem mais simples. Todo código é escrito dentro de classes e a compilação lógica ocorre segundo a execução por uma máquina virtual java (MVJ). Dentre suas principais características, pode-se citar uma arquitetura neutra e portátil que apresenta solidez e segurança que

pode ser utilizada em diversos sistemas operacionais.

Muito se especula em respeito entre a comparação do Java com o *JavaScript*. A interpretação correta é que são distintas linguagens de programação. O *JavaScript* é uma linguagem de *scripts* utilizada para acessar objetos dentro de outras aplicações. Sua utilização acontece disseminadamente em páginas da *Internet*.

## 2.8 Microcomputadores

Segundo Silva e Carvalho (2011), as principais partes de um microcomputador, dispostas na Figura 2.10, geralmente são:



Figura 2.10: Partes básicas de um microcomputador (SILVA, CARVALHO, 2011).

- *Microprocessador*

Também conhecido como CPU, corresponde à inteligência do sistema. Executa as instruções de seu programa principal gravado na memória do sistema, sendo responsável por analisar as informações recebidas de seus periféricos e processá-las a partir da interpretação, manipulação, cálculos lógicos e matemáticos em ação ao sistema;

- *Memória Volátil*

Memória para armazenamento de informações, responsáveis pelas operações de leitura e escrita. As informações contidas são perdidas sempre quando o equipamento é desenergizado. São elas: RAM, SRAM e DRAM;

- *Memória Não-Volátil*

Memória de armazenamento de programa, utilizada apenas para operações de leitura e conseqüentemente impossível de realizar escrita. As informações são preservadas mesmo quando o sistema é desenergizado. São elas: ROM, PROM, EPROM e EEPROM;

- *Entradas e Saídas*

Correspondem à interface entre o processamento e o mundo externo. Geralmente são circuitos integrados capazes de ler as informações externas e interpretar ou armazená-las (e.g. teclado e porta seriais), bem como definir as informações do microprocessador a serem aplicadas (e.g. monitor e auto-falantes).

Um microcontrolador consiste basicamente em um microcomputador implementado em um único circuito integrado, como é apresentado na Figura 2.11. No entanto, existem algumas diferenças básicas quando se comparam microcontroladores e microprocessadores. As principais delas talvez digam respeito, intuitivamente, à capacidade de processamento e controle periférico. O circuito controlador possui altas taxas de interpretações, manipulações e cálculos lógicos e matemáticos de forma que as informações de programa podem ser executadas rapidamente. De forma análoga, o controlador comunica-se bem com seus periféricos de entrada e saída oferecendo o suporte necessário de atuação, como suprimento de corrente elétrica.

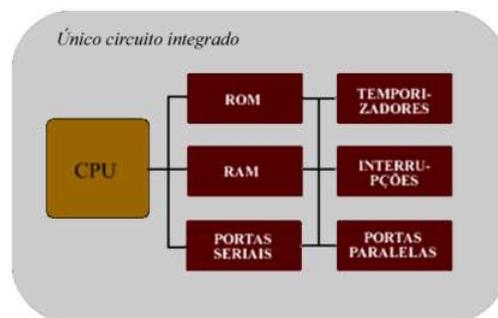


Figura 2.11: Diagrama de blocos simples de um microcontrolador (SILVA, CARVALHO, 2011).

## Metodologia

Nesse capítulo são apresentados os materiais, métodos e ferramentas a serem utilizados durante o desenvolvimento do projeto. A Figura 3.1 apresenta a metodologia a ser seguida.

A execução do presente trabalho de conclusão de curso ocorreu conforme um planejamento sequencial. Inicialmente foi realizada uma fundamentação teórica a respeito da domótica, de forma a promover o conhecimento necessário para as atividades subsequentes. Foram analisadas todas as variáveis envolvidas no processo e identificados os possíveis pontos de automação relevantes.

Posteriormente, definidos os sistemas residenciais a serem trabalhados, foi realizada a seleção e escolha dos módulos e equipamentos a serem utilizados. Encontram-se no mercado inúmeras soluções que, apesar de desempenhar a mesma função, diferenciam no preço e na plataforma de desenvolvimento. Sendo assim, foram escolhidos os microcontroladores, microprocessadores e módulos, selecionando os componentes e materiais necessários para cada sistema.

Feitas todas as escolhas e definidos os parâmetros de construção, passou-se para a etapa de projeto em si. Nesse ponto, foram projetados todos os sistemas residenciais interligados por meio da topologia de conexão pré-determinada. Deu-se início então a primeira fase de testes e simulações, onde as variáveis observadas são a taxa transmissão de dados e resolução da interface ao usuário. Caso os resultados obtidos fossem satisfatórios, seria iniciada a fase de programação dos microcontroladores e montagem das placas eletrônicas. Caso contrário, se fossem verificados equívocos quanto ao protocolo de comunicação, no que se refere à forma como os dados são tratados, justificando a escolha de cabos por exemplo, seria necessário o redimensionamento do projeto. Assim, uma nova análise a respeito das decisões tomadas deveria ser feita levando em consideração os parâmetros e

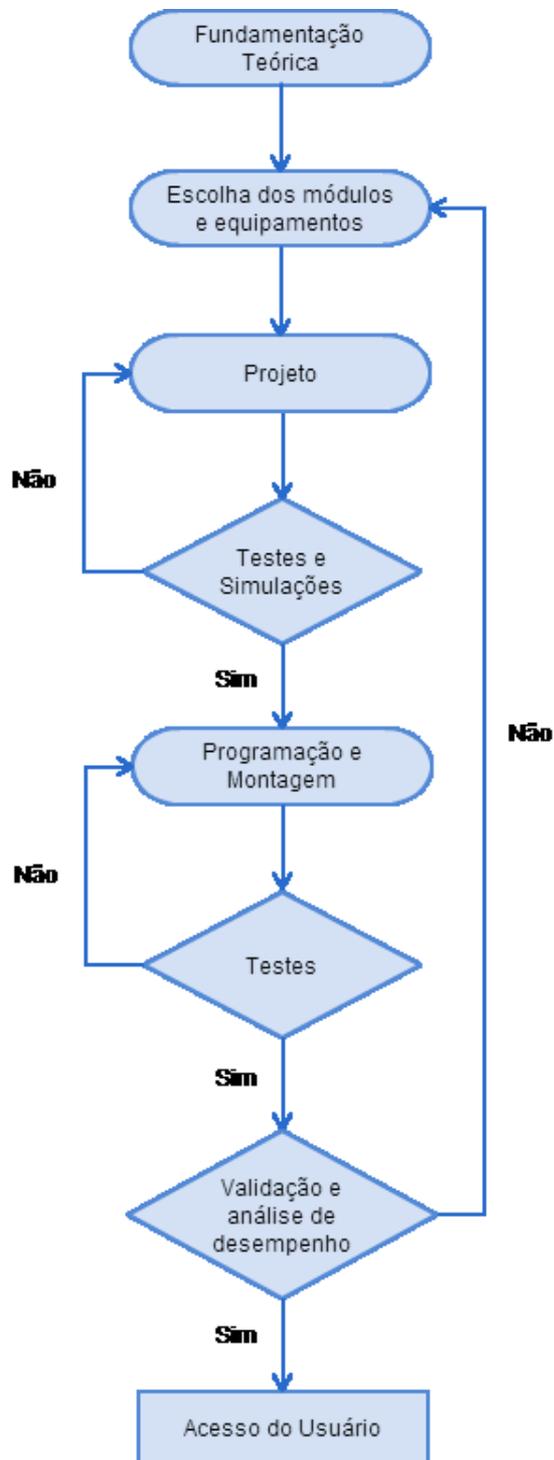


Figura 3.1: Fluxograma da metodologia do projeto.

---

equipamentos pré-selecionados.

Com a montagem física dos sistemas, se tornou possível uma segunda fase de testes, dessa vez prática. Nessa etapa todas as conexões foram testadas e os valores medidos na saída analisadas, avaliando se as montagens e a programação foram bem sucedidas. Caso o resultado fosse aceitável, todo o sistema prosseguia para a validação dos resultados e análise de desempenho. Se os valores de saída medidos obtivessem amplitudes razoáveis dentro das margens de erro estabelecidas, significaria que o projeto foi bem dimensionado e estaria a um passo de estar disponível ao usuário. Caso contrário, seria preciso voltar novamente à fase de escolha dos módulos e equipamentos e reavaliar todo o processo, a fim de se identificar as prováveis melhorias e corrigir os devidos equívocos.

Ao final, com a validação do sistema apresentando margens de erro admissíveis, os sistemas automatizados estão disponíveis ao usuário sob uma interface de comando remota, na qual é possível controlar os sistemas residenciais em questão com apenas um toque.

O desenvolvimento do projeto inicialmente foi programado para oito meses, sendo quatro referentes à disciplina de TCC 1 e os outros quatro ao TCC 2. O cronograma de trabalho e a idealização das tarefas, bem como os prazos estimados, estão apresentados na Tabela 3.1.

Tabela 3.1: Cronograma de trabalho para projeto disciplina de TCC.

Etapas	Mês							
	1	2	3	4	5	6	7	8
1 - Revisão literária.	x	x	x	x				
2 - Elaboração do projeto.	x	x						
3 - Simulação eletrônica (virtual).		x	x					
4 - Orçamento.			x					
5 - Aquisição de materiais.			x	x				
6 - Redação do Relatório Parcial			x	x				
7 - Desenvolvimento dos algoritmos de controle e da interface ao usuário				x	x	x		
8 - Programação do microcontrolador				x	x			
9 - Programação do microprocessador				x	x			
10 - Projeto e Layout das placas eletrônicas					x			
11 - Confeção das placas eletrônicas						x	x	
12 - Testes e simulações eletrônicas							x	
13 - Validação do sistema por controle remoto							x	
14 - Simulações e testes finais						x	x	
15 - Redação do Relatório Final						x	x	x

Os procedimentos experimentais foram realizados nas dependências do CEFET-MG *campus* Divinópolis. Por se tratar de um projeto que possui como áreas de atuação a eletrônica e programação, foi necessária a utilização dos seguintes laboratórios:

- Laboratórios de Eletrônica: Para realização da montagem das placas, foi necessário utilizar protoboards, ferramentas e conectores. Em relação à medição de sinais e testes em geral foram usados osciloscópios, geradores de sinais, multímetros, entre outros equipamentos eletrônicos;
- Laboratórios de Programação: Os laboratórios de programação e informática ofereceram o suporte para o desenvolvimento dos softwares e algoritmos a serem desenvolvidos. Foi possível encontrar computadores disponíveis que possuíam plataforma de desenvolvimento e conexão com fornecimento de sinais;
- Laboratórios de Elétrica: Encontram-se disponíveis painéis elétricos contendo elementos como relés, fusíveis, contadores e também certos atuadores como lâmpadas, interruptores e sinalizadores.

Tendo em vista que o desenvolvimento do projeto visa o baixo custo, a Tabela 3.2 apresenta a lista dos os materiais e equipamentos utilizados e como foram concebidos.

Tabela 3.2: Orçamento: Lista de equipamentos e recursos utilizados.

Nome do Equipamento	Preço Médio de Mercado	Fonte de Obtenção (Disponível: Sim/Não)
Microprocessador <i>Raspberry pi</i> módulo B	R\$200,00	Autor (S)
Microcontrolador <i>Atmega 328</i>	R\$20,00	Autor (N)
Roteador	R\$50,00	Autor (N)
Lâmpadas incandescentes CA	R\$20,00	Laboratório (S)
Lâmpada incandescente CC	R\$4,00	Laboratório (S)
Lâmpada LED CC	R\$12,00	Laboratório (S)
Boquilhas	R\$15,00	Laboratório (S)
Interruptor e tomada	R\$15,00	Laboratório (S)
Cooler	R\$20,00	Autor (S)
Módulo dimer	R\$20,00	Autor (N)
Módulo relé	R\$20,00	Autor (N)
Módulo controle CC	R\$20,00	Autor (N)
Sensor de presença para iluminação	R\$30,00	Autor (N)
Medidor de corrente	R\$50,00	Autor (N)
Extensor USB	R\$30,00	Autor (N)
Fonte de alimentação	R\$50,00	Autor (N)
Madeira para maquete	R\$50,00	Autor (N)
Estojo de proteção dos módulos	R\$20,00	Autor (N)

Analisando a Tabela 3.2, o valor de custo total do projeto gira em torno de aproximadamente R\$576,00. Com uma margem de erro e variação de preços, essa quantia pode ser considerada entre R\$500,00 e R\$600,00. Os materiais mais caros adquiridos foram o Raspberry, que pode ser adquirido apenas através de importação, e a fabricação das placas eletrônicas dos módulos de controle, como o dimer e relé.

Esse valor pode ser considerado como custo relativo já que certos materiais e equipamentos não serão devidamente comprados. Alguns itens na lista puderam ser adquiridos de forma indireta ou estavam disponíveis nos laboratórios. É necessário considerar que existem instrumentos auxiliares que foram indispensáveis para o desenvolvimento do projeto. Dentre esses instrumentos podem ser citados: *smartphone*, monitor e teclado (para programar o microprocessador), fios conectores entre equipamentos, entre outros.

## 3.1 Materiais e Métodos

### 3.1.1 Configuração do Sistema

O sistema é composto por células microcontroladoras remotas gerenciadas por uma central microprocessada, onde os sinais de controle são enviados remotamente por meio de uma rede sem fio. A Figura 3.2 apresenta o esquema de conexão proposto.

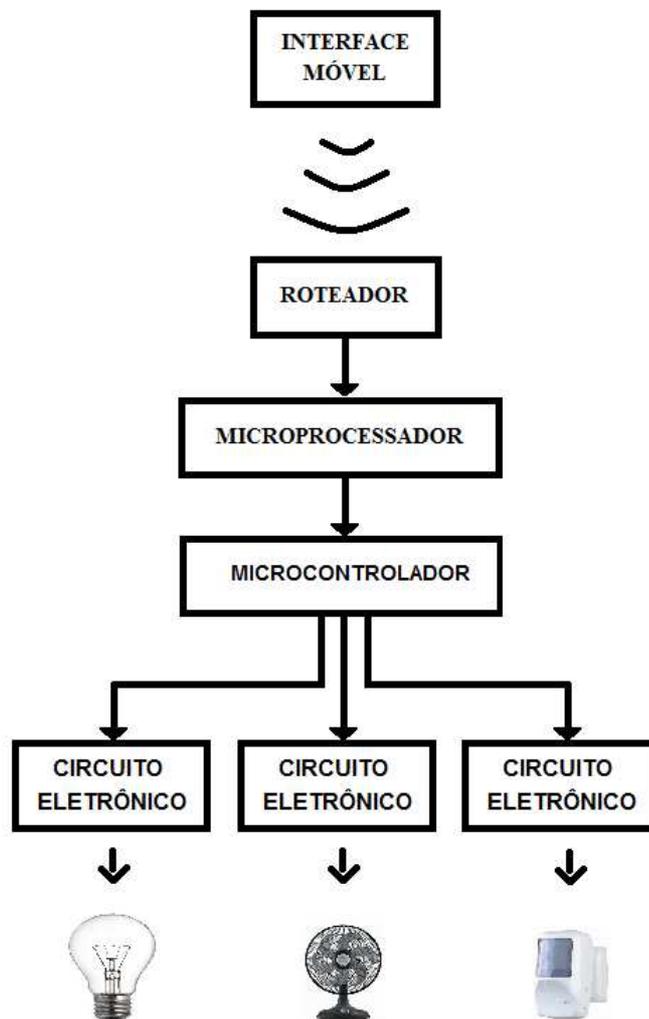


Figura 3.2: Esquema de conexão do projeto.

O *Raspberry pi* é o microprocessador utilizado. Apresentando um sistema operacional próprio contendo 512MB de memória RAM, sistema de processamento igual a 700MHz e periféricos adequados, esse equipamento garante ao projeto uma capacidade efetiva de tratamento de dados. O consumo de uma quantidade de energia consideravelmente desprezível determina que esse dispositivo fique energizado 24 horas por dia enviando os sinais de comando quando acionado. A sua aquisição é justificada por apresentar um preço mé-

dio de mercado de aproximadamente cinco vezes menor que as soluções concorrentes.

Todos os sinais enviados pelo microprocessador são repassados ao microcontrolador, e o escolhido foi o *Atmega 328*. Apresentando alto desempenho e também um baixo consumo de energia, esse equipamento recebe os sinais de comando e ativa o nível lógico de suas saídas. Assim como o *Raspberry pi*, o *Atmega* precisa ser programado apenas uma vez e opera então sistematicamente quando exigido.

Inicialmente, o objetivo do trabalho corresponde ao acionamento e controle de intensidade de equipamentos como lâmpadas incandescentes e ventiladores. Isso poderia ser facilmente implementado utilizando-se apenas um microcontrolador, já que o *Atmega 328* consegue interpretar os sinais de entrada e controlar suas saídas de acordo com seu código fonte. No entanto essa configuração não demonstrou ser suficientemente efetiva durante uma análise inicial por vários motivos.

Em primeiro lugar, o microcontrolador não possui memória não volátil suficiente para armazenar todos os códigos. Nessa configuração a qualidade da interface do usuário seria necessariamente simples, uma vez que o ambiente de desenvolvimento integrado de programação deveria conter toda a configuração das páginas da *WEB*. De certa forma a lógica de comunicação contém códigos em *Java*, *JavaScript*, *JavaServer*, *HTML* e *C/C++*, cada qual em um arquivo diferente e, mesmo se tratando de apenas alguns *Mega Bytes*, não haveria espaço suficiente. Outro fato que contribuiu para o descarte dessa configuração diz respeito ao processamento de dados e às possíveis futuras implementações. Incorporando-se o *Raspberry pi* é possível desenvolver o controle de sistemas que possam fazer a transmissão e interpretação de dados em tempo real apresentando uma interface bem amigável e atrativa ao usuário.

Quanto ao envio de instruções relacionadas aos comandos para a realização de uma tarefa, é utilizado um meio de comunicação por rede sem fio. A *Internet* será o meio de comunicação padrão entre o sistema e o usuário, através de uma interface aplicável em navegadores da *WEB*. Com o uso do sistema global de computadores, torna-se possível o envio de sinais, através de *smartphones*, *tablets* ou *laptops* independente de sua localização.

É importante ressaltar a resiliência apresentada pela configuração quanto às possíveis interferências. Caso haja uma falha na conexão com a *Internet* ou mesmo se o usuário encontrar-se dentro de sua própria residência, o dispositivo móvel pode conectar-se na rede sem fio local gerada pelo roteador. Isso é possível graças à capacidade de ser conec-

tado e também conectar-se à redes *wireless*. Dessa forma, os sistemas continuam sendo controlados de forma remota sem a necessidade de conexão *online*. Essa característica garante uma independência quanto ao sinal de *Internet*, que pode ser suscetível à ruídos e indisponibilidade, como em casos de tempestades e fortes chuvas.

Além disso, como configuração padrão, o gerenciamento manual e convencional de interruptores continua funcionando independentemente da acessibilidade do dispositivo móvel. Em caso de uma possível inconveniência que impeça a comunicação sem fio, o acionamento por interruptores e disjuntores permanecerá como devido. Sendo assim a residência considerada inteligente, conforme Rodrigues (2012), pode ser considerada uma residência comum acrescentada de dispositivos inteligentes que jamais lhe retirarão suas características básicas.

No que se refere à forma de transição de dados dentro da própria residência, com o conseqüente envio dos sinais via rede sem fio, um roteador é implementado e, através do padrão *Ethernet*, os dados recebidos são repassados ao microprocessador. Através de um cabo Cat5, os comandos serão interpretados pela central de processamento e posteriormente repassados aos microcontroladores.

Em relação ao funcionamento dos equipamentos, todos os sinais de saída do microcontrolador são enviados aos circuitos específicos de cada tarefa. Como os pinos de saída do *Atmega 328* operam entre 0 e 5 *Volts*, são feitas a utilização de módulos relés, *dimers* e de controle CC, de forma que a energia suficiente seja aplicada ao atuador. Para o acionamento simples, o módulo relé é a escolha mais aplicável. Isso porque a construção de um módulo é possível com uso apenas de um próprio relé, transistor, diodo e resistências. No entanto, esse circuito apenas aplica a energia da rede através de um sinal de comando, sendo possível apenas ligar e desligar equipamentos.

Quando o controle de intensidade é necessário, é utilizado um circuito dimerizado que por definição, é um circuito que permite a variação do fluxo energético. No projeto serão abordadas duas variações na configuração de um dimer. A primeira corresponde ao acionamento de cargas puramente resistivas, como as lâmpadas incandescentes. A segunda se destina ao comando de motores, onde a carga possui impedância complexa, caracterizada pela presença da parte indutiva. Para isso, será preciso adicionar à um dimer convencional um resistor de potência, de forma a compensar a parte real da impedância de saída.

### 3.1.2 Microprocessador Raspberry Pi

O Raspberry Pi, Figura 3.4, é um mini microcomputador desenvolvido pela *The Raspberry Foundation*, uma instituição filantrópica do Reino Unido. Esse equipamento é provido de processadores de arquitetura ARM, garantindo que os sistemas de controle e unidades gerem menos calor e gastem menos energia. A sua operabilidade ocorre através do sistema operacional baseado em *Linux*, permitindo ao usuário a conexão de mouse e teclado comum. Por não apresentar disco rígido próprio (memória não volátil), o sistema operacional deve ser instalado em um cartão de memória SD. Atualmente, existem 5 distribuições disponíveis para instalação do SO, sendo elas: *Raspbian*, *Pidora*, *Openelec*, *Raspbmc* e *Risc OS*.

Para o projeto em questão, está sendo utilizada a distribuição *Raspbian*. Esse sistema operacional é otimizado para o Raspberry apresentando um código aberto e contendo cerca de 35.000 pacotes pré-compilados, o que torna sua instalação simples. A Figura 3.3 apresenta o processo em que o raspberry esteve sendo configurado para o sistema proposto. O **Apêndice A** desse documento apresenta o procedimento de instalação e configuração inicial para o dispositivo.



Figura 3.3: Processo de configuração do Raspberry PI.

São vários os motivos que foram decisivos na escolha de utilização do Raspberry. Primeiramente, em comparação com as soluções análogas e concorrentes, as opções se resumiam essencialmente ao uso de computadores e *laptops*. Em uma breve análise de mercado, estima-se que o preço médio de um computador pessoal seja de R\$1500,00 enquanto o preço de conversão do Raspberry é de aproximadamente R\$200,00, na cotação

atual do dólar. Isso indica que a obtenção de um equipamento com recursos semelhantes pode gerar uma economia de 750%. Além disso, o termo portabilidade e viabilidade de execução se propuseram importantes. De certa forma, para a instalação dentro uma residência é preciso um equipamento pequeno e que possa ficar ligado interruptamente. Usuários domésticos não necessitam de interfaces de controle direto, por isso um monitor, teclado e mouse são indispensáveis apenas no momento de assistência técnica e não para o dia a dia.



Figura 3.4: Microprocessador *Raspberry pi* Modelo B (ADAFRUIT, 2014).

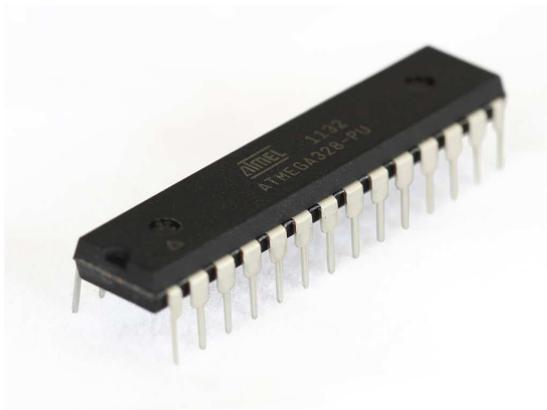
Esse equipamento possui as seguintes especificações:

- 512MB de memória RAM;
- Processador de 700MHz (*Broadcom BC2835 ARM11*);
- 4 processadores GPU capaz de reproduzir 1080p em Full-HD;
- Sistema Operacional GNU/*Linux Debian*;
- 2 Portas USB;
- Saída de Vídeo HDMI;
- Saída de Vídeo RCA;
- Porta *Ethernet* 10/100Mb;
- 5V Micro USB para entrada de energia;

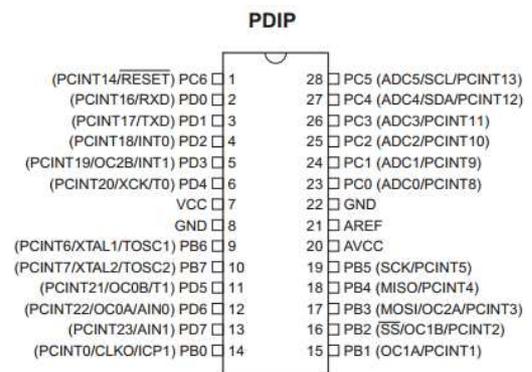
- *Slot* para cartão de memória SD, MMC, SDIO *flash*;
- 26 pinos que permitem periféricos e placas de expansão;
- Consumo de energia de 3.5W (700mA)

### 3.1.3 Microcontrolador Atmega 328

O *Atmega 328*, Figura 3.5a, é um chip microcontrolador desenvolvido pela *Atmel* pertencente à família *megaAVR*. Possui alta performance e baixo consumo de energia e foi desenvolvido na tecnologia CMOS e arquitetura RISC, cuja pinagem é apresentada na Figura 3.5b.



(a) Controlador *Atmega 328*



(b) Pinagem do controlador *Atmega 328*

Figura 3.5: Microcontrolador *Atmega 328* (ATMEL, 2009).

As principais características e parâmetros do *Atmega 328* são:

- 28 pinos;
- CPU 8 bits;
- 16 canais variáveis;
- 32x8 registradores de trabalho de propósito geral;
- Operação até 16M de instruções por segundo a 16MHz;
- 130 instruções em um único ciclo de clock;
- 23 entradas e saídas programáveis;
- 3 canais PWM;

- 6 canais A/D com precisão de 8 bits;
- Watchdog Timer com oscilador interno separado;
- Tensão entre 1,8 e 5,5 Volts.

Uma forma alternativa de conexão para o microcontrolador Atmega 328 consiste em utilizar o Arduino como plataforma embarcada. O Arduino é um *hardware* de prototipagem eletrônica *open source* com suporte embutido de entradas e saídas. Uma vantagem em seu uso diz respeito à fácil manipulação de sua IDE, bastando conectá-lo a um computador via USB e prosseguir com a alteração de sua programação, que ocorre essencialmente em C/C++. A Figura 3.6 apresenta a equivalência da pinagem entre o circuito integrado Atmega 328 e as I/O's da plataforma Arduino.



Figura 3.6: Equivalência de pinagem entre Atmega 328 e Plataforma Arduino.

### 3.1.4 Módulo Dimer

Para o controle de intensidade da potência dos equipamentos em corrente alternada foi projetado um circuito dimer, conforme a configuração apresentada na Figura 3.7. O circuito pode ser compreendido em duas etapas:

**Etapa 1:** a parte superior da configuração possui a função de contar as vezes que a onda alternada passa pelo zero. Inicialmente utilizam-se resistores de  $30k\Omega$  com potência de dissipação igual a  $0.5\text{ Watts}$  para a diminuição da corrente provinda da rede de alimentação. Posteriormente segue-se um sistema retificador BR1, obtido através de 4 diodos 1N4007 dispostos em ponte, responsável por obter apenas semiciclos positivos. Com a presença de um opto acoplador 4N25, que isola fisicamente o circuito, um transistor faz

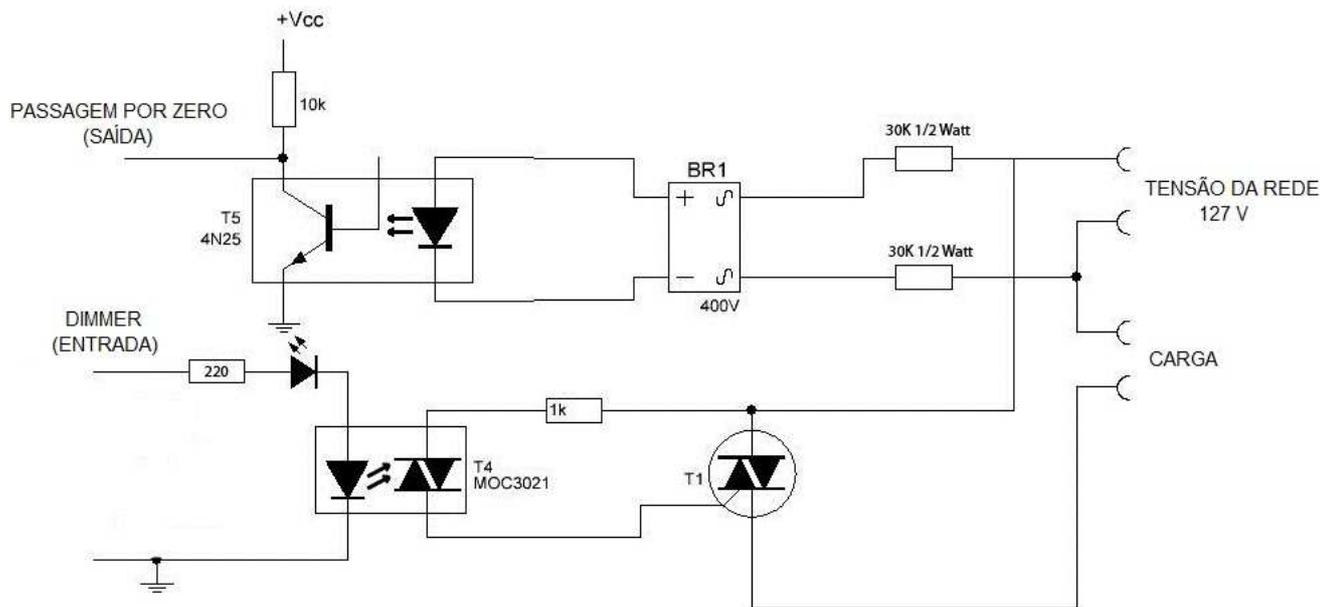
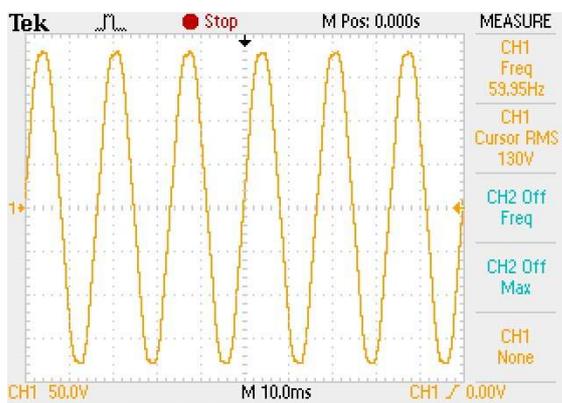


Figura 3.7: Configuração do dimer

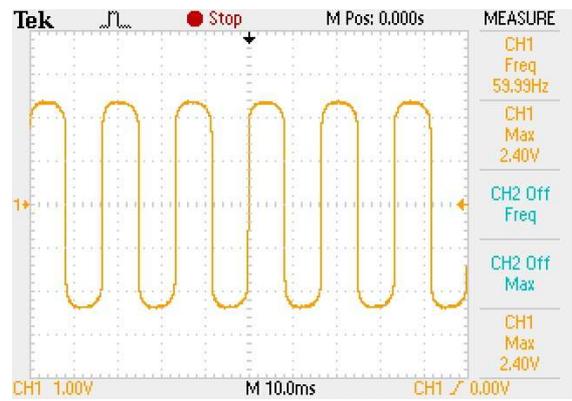
o papel central de identificação de zeros. Isso acontece porque toda vez em que o sinal alternado passa por 0, o fotodiodo não emite luz, o transistor não conduz e desse modo a tensão na saída (*Zero Crossing*) é 5V. Esse sinal é então enviado ao microcontrolador como aviso de amplitude nula. Sua repetição acontece 120 vezes por segundo, já que a senóide tem valor igual a zero duas vezes a cada semiciclo, indicando intervalos de 8.33ms. A Figura 3.8 apresenta os sinais medidos pelo osciloscópio do sistema projetado descritos na etapa 1, referente à passagem e detecção de zero.

A Figura 3.8a representa o sinal provindo da rede de alimentação e por isso possui valor RMS de tensão igual a 130V. Após a passagem pelos resistores de potência, o sinal tem sua amplitude diminuída a 2.4V, como é mostrado em 3.8b. O retificador gera uma onda contendo apenas semicilcos positivos, bem como apresentado em Figura 3.8c. Por fim, a cada passagem por zero, são emitidos pulsos à 120Hz com amplitude de 4.4V de acordo com a Figura 3.8d.

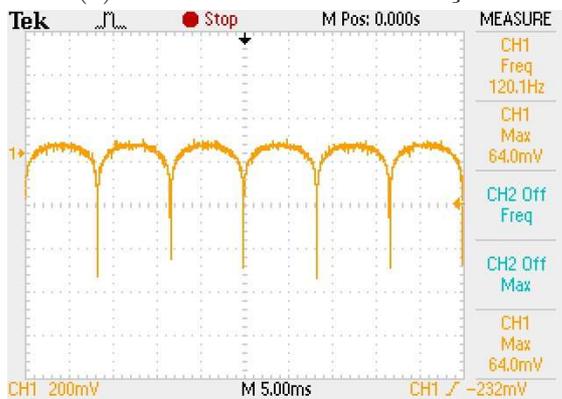
**Etapa 2:** a parte inferior atua sobre o ângulo de disparo do dimmer. A partir do sinal de 5V, o microcontrolador emite um outro sinal de tensão que ativa o fotodiodo do opto acoplador/opto isolador MOC3021. Esse sinal acontece em um determinado tempo previamente calculado de acordo com o ângulo de disparo desejado. O tempo de disparo de acordo com o ângulo pode ser calculado pela Equação 3.1:



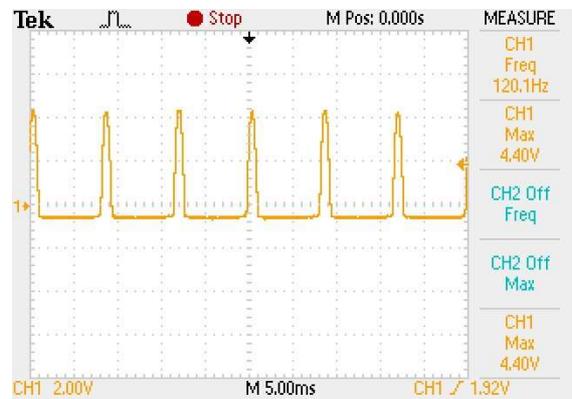
(a) Sinal da rede de alimentação.



(b) Sinal reduzido pelos resistores de potência.



(c) Sinal retificado.



(d) Pulsos de passagem por zero.

Figura 3.8: Medição dos processos da etapa de passagem por zero do dimer.

$$T_{disp} = \frac{Iz - Gc}{Re} \quad (3.1)$$

Esse momento para disparo do triac corresponde então à razão da diferença entre o tempo de intervalo entre os zeros do sinal e o tempo de garantia de condução, sobre a escala ou resolução do sistema. Dessa forma, operando com 127V a 60Hz e desejando uma escala que varia de 0 a 100, tem-se que:

$$T_{disp} = \frac{8.33ms - 8.33us}{100} = 83.2\mu s$$

Esse valor indica que a cada 83.2  $\mu$  segundos decorridos, o ângulo de disparo aumenta em 1 grau, correspondente a cada valor de intensidade. Posteriormente, de acordo com o valor de energia desejado, esse ângulo de disparo é enviado ao triac, em uma escala correspondente em Volts. Finalmente, o sinal enviado ao triac o faz conduzi-lo em diferentes momentos, aplicando-se então a tensão desejada na Carga. A Figura 3.9 apresenta a saída para a carga de acordo com os pulsos emitidos pelo triac, em razão do ângulo de disparo determinado.

Nota-se que o valor da intensidade está diretamente relacionado com o momento de condução do triac. Quanto mais tardio ele acontece, menor é a área da onda conduzida e consequentemente também é menor a potência de saída do dimer, uma vez que a potência é calculada pela integral da área ativada. Na Figura 3.9a, para intensidade a 0%, o sinal é praticamente nulo, enquanto para saídas intermediárias como as Figuras 3.9b, 3.9c e 3.9d o sinal aumenta proporcionalmente ao momento em que o triac conduz. A Figura 3.9e conduz toda energia de entrada para a saída e dessa forma, percebe-se que o pulso de condução é aplicado praticamente no mesmo instante em que a amplitude torna-se não-nula.

Para a construção da PCI (placa de circuito impresso), utilizou-se o software Proteus para dimensionamento e simulação, conforme a Figura 3.10.

O módulo dimer desenvolvido é apresentado na Figura 3.11. As dimensões referentes da placa são 75mm por 50mm com uma altura máxima de 25mm. Nota-se que o projeto enfatizou a redução dimensional do circuito pela possibilidade de implementação de forma prática e que seja o menos notável possível ao usuário.

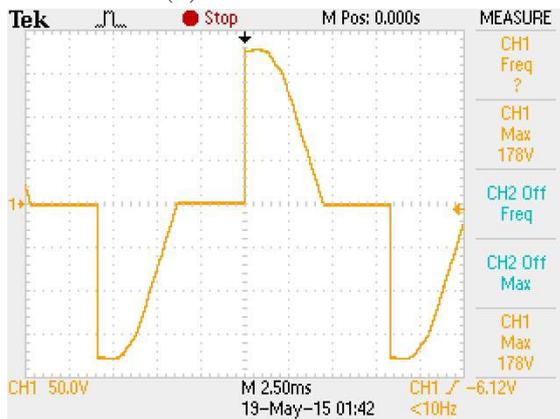
**Controle de cargas indutivas:** Cargas como motores possuem impedância complexa devido à presença da parte indutiva. Para o acionamento desses equipamentos



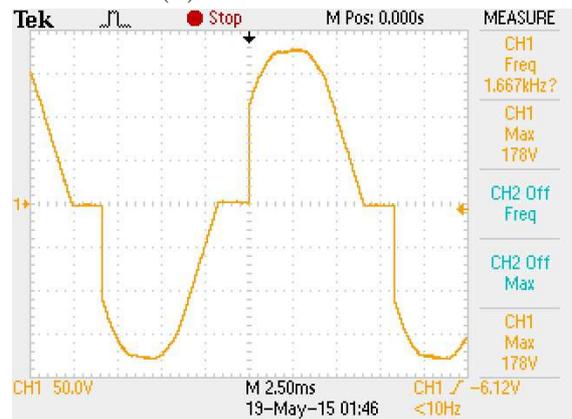
(a) Intensidade 0%.



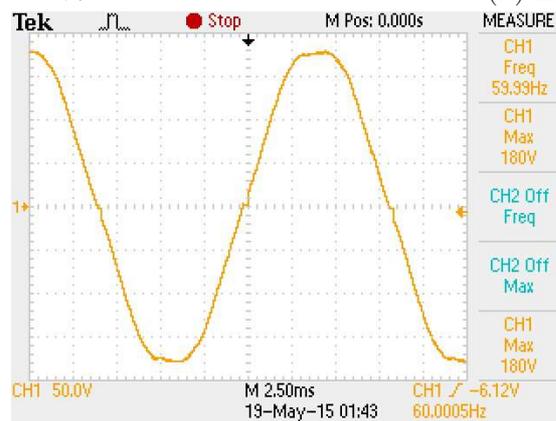
(b) Intensidade 25%.



(c) Intensidade 50%.



(d) Intensidade 75%.



(e) Intensidade 100%.

Figura 3.9: Medição das saídas para carga de acordo com o ângulo de disparo referente à determinada intensidade.

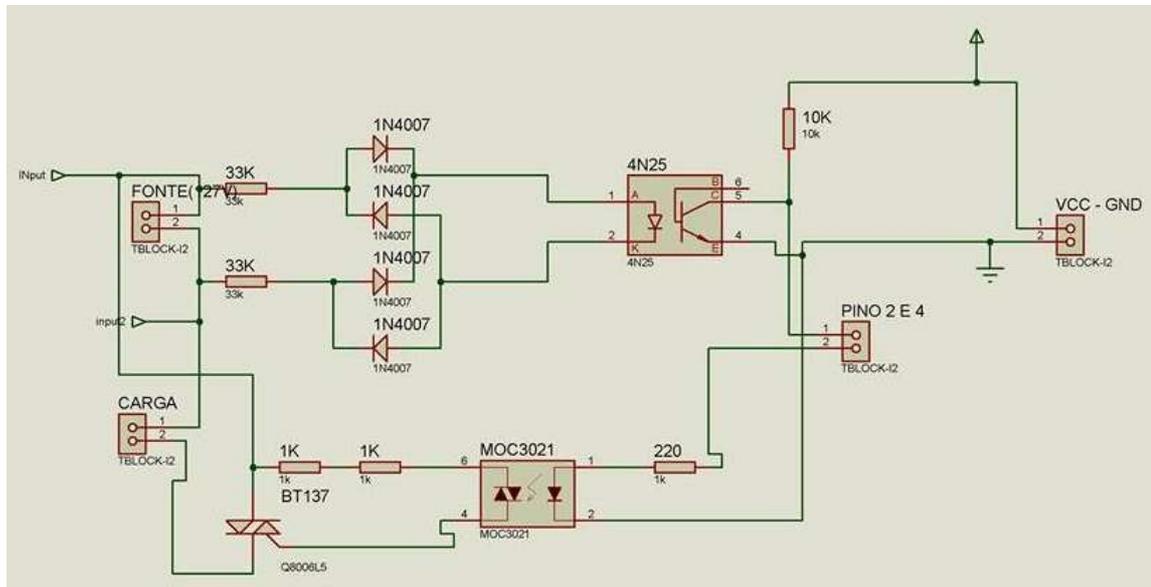


Figura 3.10: Projeto esquemático do dimer em Proteus.

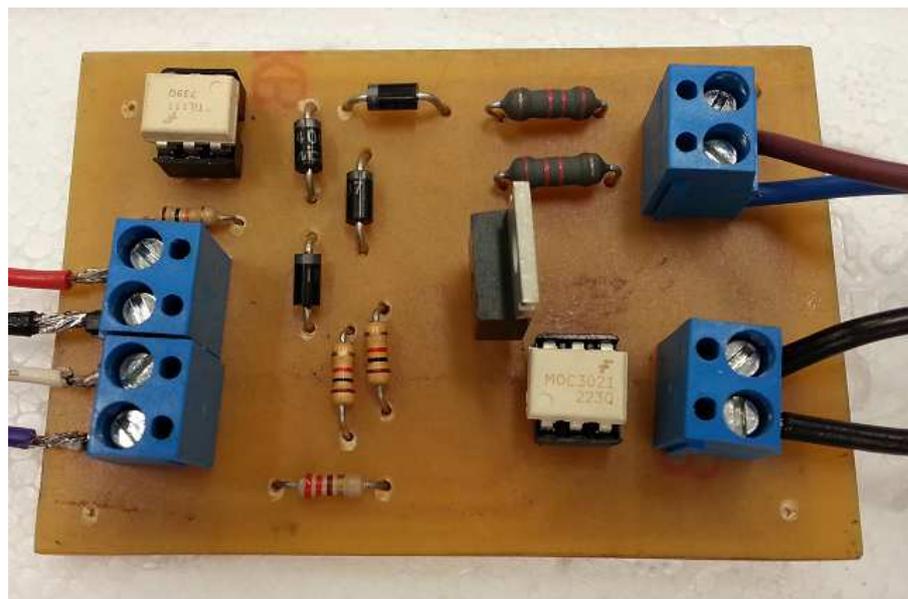


Figura 3.11: Módulo dimer desenvolvido.

deve-se ser feita uma pequena modificação no circuito do dimer. Essa modificação consiste no simples acréscimo de um resistor de potência em série com a saída para a carga. A justificativa para essa ação consiste no fato de proporcionar uma maior carga resistiva para o controle de intensidade. Dimers atuam diretamente sobre a resistência de atuadores, como lâmpadas e resistências elétricas, e por isso podem não funcionar corretamente sobre equipamentos como secadores de cabelos e ventiladores. A especificação para esse resistor depende do tipo de carga a ser controlada, mas em média pode ser utilizado de 30K a 50K à em média 1KW de dissipação.

### 3.1.5 Módulo Relé

Para o acionamento dos equipamentos em que não há a necessidade de controle de potência, projetou-se um módulo relé, conforme a Figura 3.12. Um relé consiste em um dispositivo eletromecânico que funciona basicamente como um interruptor. Seu princípio de funcionamento baseia-se na movimentação física de contatos quando a sua bobina é energizada a partir de corrente elétrica, a partir da criação de um campo magnético. Sua utilização é viável quando deseja-se acionar circuitos que trabalhem com potências superiores aos do sinal de comando. No trabalho em questão, por exemplo, os sinais de comando operam de 3 a 5 Volts, enquanto os dispositivos trabalham na faixa de 127 ou 220 Volts.

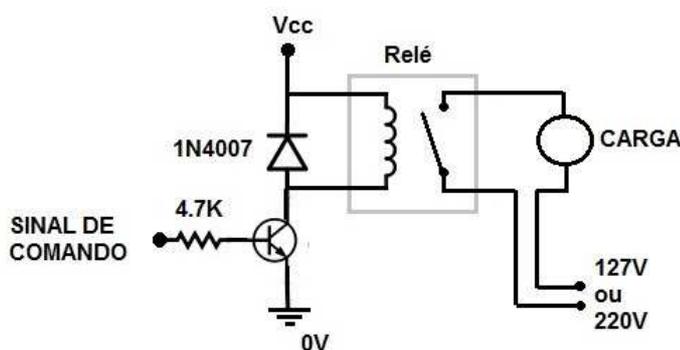


Figura 3.12: Configuração do relé.

As Figuras 3.13 apresenta o projeto em Proteus de dimensionamento e simulação da PCI do módulo.

O módulo relé desenvolvido é apresentado na Figura 3.14. As dimensões da placa são 75mm por 35mm com uma altura máxima de 20mm.

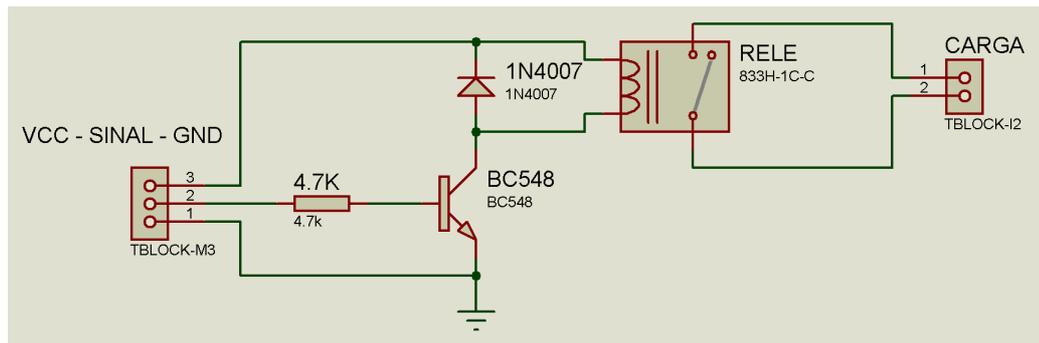


Figura 3.13: Projeto esquemático do relé em Proteus.

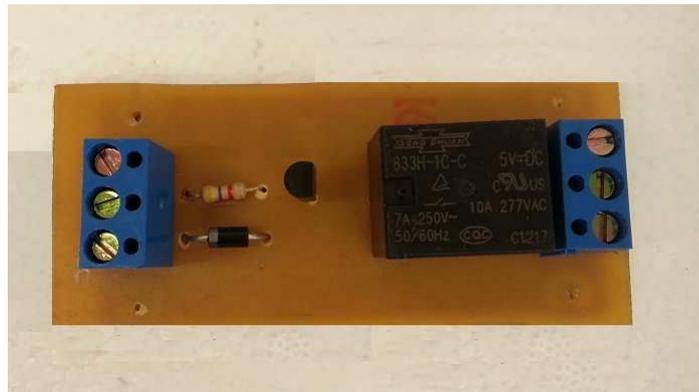


Figura 3.14: Módulo relé desenvolvido.

### 3.1.6 Módulo Controle CC

Para o controle de potência de equipamentos que funcionam em corrente contínua, projetou-se um módulo específico cujo princípio de funcionamento concentra-se no mosfet IRF540. A Figura 3.15 o circuito em questão.

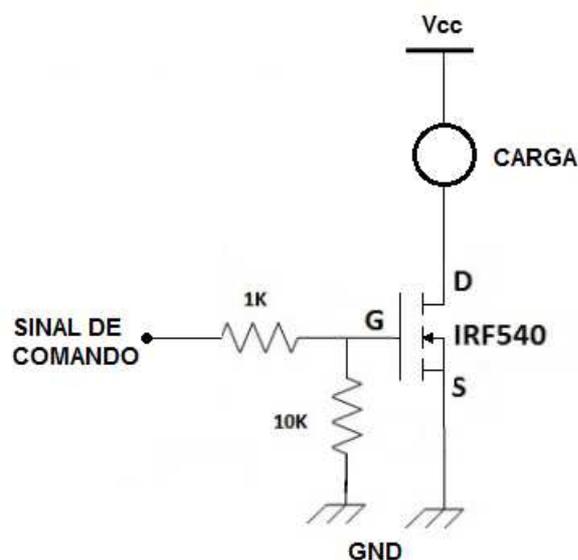


Figura 3.15: Configuração do controle CC.

Analisando a Figura 3.15 nota-se que o sinal de comando é conectado ao pino *Gate* do mosfet através de uma resistor de 1K juntamente com outro de 10K que vai para o referencial do circuito. Dependendo do valor aplicado nesse ponto, numa escala de 0V a 5V, a saída na carga varia seu valor de tensão de 0V a  $V_{cc}$  de forma proporcional às curvas de cada mosfet. Isso acontece porque o princípio de funcionamento implica que a corrente de dreno é função da tensão de comporta. Dessa forma, quando a tensão de entrada é 0V, a de saída também é nula, enquanto se o valor aplicado for de 5V, a saída adquire tensão igual a  $V_{cc}$  conduzindo o valor máximo de corrente. Nota-se que o dreno deve ser conectado ao pólo negativo da carga enquanto o pólo positivo liga-se em  $V_{cc}$ .

As Figuras 3.16 apresenta o projeto em Proteus de dimensionamento e simulação da PCI do módulo.

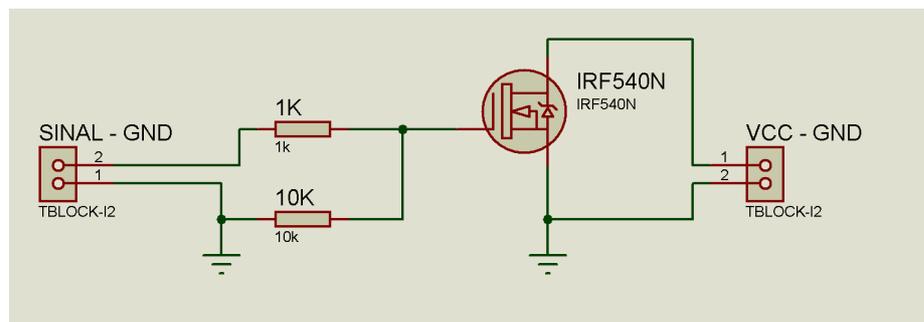


Figura 3.16: Projeto esquemático do circuito de controle CC em Proteus.

O módulo controle CC desenvolvido é apresentado na Figura 3.17. As dimensões da placa são 45mm por 25mm com uma altura máxima de 25mm.

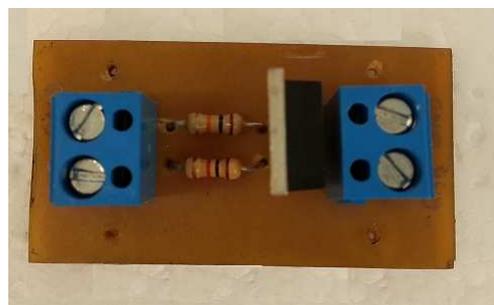


Figura 3.17: Módulo controle de CC desenvolvido.

### 3.1.7 Medidor de Corrente

A fim de se medir a corrente drenada por cada equipamento, foi projetado um circuito medidor de corrente, Figura 3.19. Ele é baseado no sensor ACS714 do fabricante Allegro,

possuindo quatro entradas (duas de sinal de alimentação e duas de potência) e uma saída, conforme a Figura 3.18. Através do efeito Hall, esse CI isola eletricamente a corrente e responde linearmente à variação de corrente referente à entrada. O sistema suporta entrada de corrente bidirecional de  $-30\text{A}$  a  $+30\text{A}$  e opera à uma sensibilidade de  $66\text{mV/A}$ . Esse sensor é alimentado com  $4.5/5.5\text{V}$  e tem sua saída centrada em  $2.5\text{V}$ , variando assim  $66\text{mV}$  positivamente se a corrente aumentar e negativamente se diminuir. A sua escolha baseia-se no fato de apresentar boa precisão e confiabilidade, com erro total típico de  $1.5\%$  em temperatura ambiente, tensão offset estável, histerese magnética desprezível e é calibrado de fábrica.

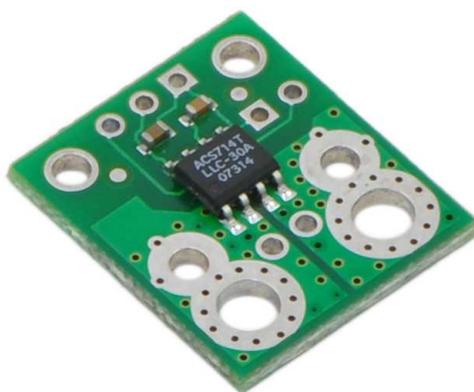


Figura 3.18: ACS714.

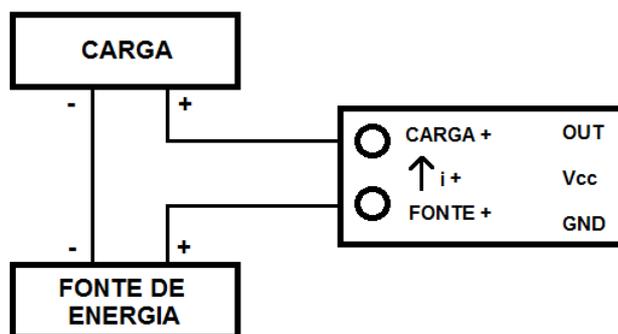


Figura 3.19: Circuito medidor de corrente.

Esse sensor está ligado em série com a alimentação geral do sistema. Dessa forma é possível perceber a variação de corrente a cada momento em que determinado equipamento é acionado, informando em tempo real essa variação.

### 3.1.8 Fonte de Corrente Contínua

Para o fornecimento de energia em corrente contínua, utilizou-se uma fonte de alimentação para computadores, Figura 3.20. Esse equipamento consiste em um sistema cuja

função principal é converter a tensão alternada (associada à corrente alternada - CA) fornecida pela rede elétrica em tensão contínua (associada à corrente contínua - CC).



Figura 3.20: Fonte de alimentação CC.

A escolha em utilizar uma fonte de computador se justifica pela sua fácil obtenção e pelo fato de converter os 110V ou 220V, à 60Hz, da rede elétrica em tensões contínuas iguais a +3.3V, +5V, +12V e -12V. Além disso, a potência disponível igual a 240W supre os gastos do sistema, a partir de corrente elétrica máxima de 5A. A Tabela 3.3 apresenta as saídas CC e a cor do fio correspondente.

Tabela 3.3: Saídas CC da fonte de alimentação

Cabo	Marron	Vermelho	Branco	Amarelo	Azul	Preto
Tensão	+3.3V	+5V	-5V	+12V	-12V	0V
Corrente	19A	14A	0.3A	7.5A	0.15A	-

Para testar o funcionamento de uma fonte de alimentação basta identificar o cabo de alimentação da placa mãe, que normalmente é o barramento de maior tamanho, e conectar o fio verde à qualquer fio preto, bem como apresentado na Figura 3.21. Feito isso, o cooler deve ser acionado instantaneamente, indicando que a fonte foi ligada. No entanto isso não garante ainda que a fonte está funcionando perfeitamente. Para isso, um teste de medição

com o multímetro deve ser feito entre os fios energizados, verificando correspondência de leitura bem como apresentado na Tabela 3.3. Para o projeto utilizou-se apenas tensões de 12V para o acendimento de lâmpadas e coolers. Para isso, foi preciso usar a tensão entre os fios amarelos e pretos, bem como no barramento apresentado na Figura 3.22.

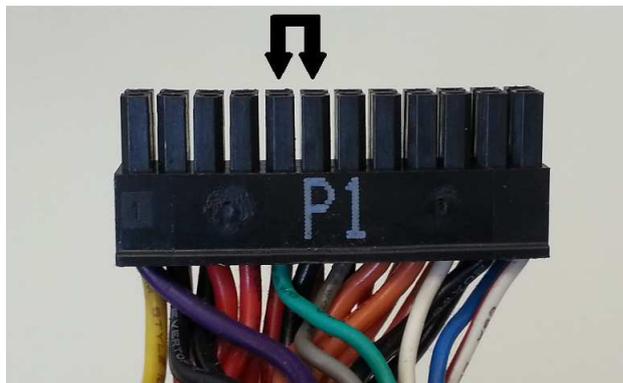


Figura 3.21: Conector principal de alimentação da placa mãe da fonte CC.

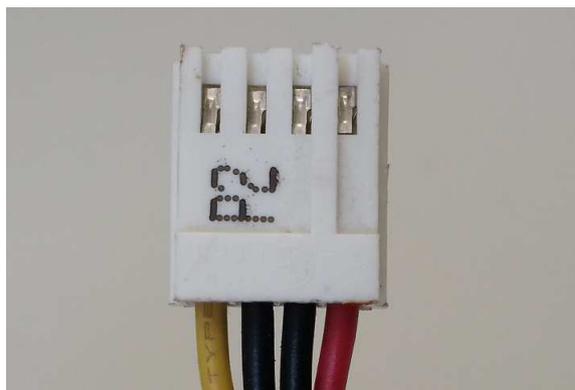


Figura 3.22: Conector de alimentação da fonte CC.

### 3.1.9 Topologia de Instalação

Para realizar a instalação física do projeto, foram propostas configurações específicas de ligação entre os dispositivos envolvidos. Primeiramente, consideraram-se os fundamentos de uma instalação elétrica convencional e a partir daí, foi desenvolvida a instalação apropriada.

Assim como abordado por Muratori e Bó (2013), em um sistema centralizado todos os retornos das cargas devem ser levados para o quadro de automação. Partindo desse pressuposto, projetou-se uma configuração tal que, o sistema apresente um nível considerável de resiliência no que se diz respeito ao controle do usuário. Dessa forma, foi preciso

determinar o acendimento de uma carga remotamente e que ao mesmo tempo atenda aos comandos cotidianos como o pressionamento de interruptores.

No que se refere à iluminação, a solução para essa questão consiste na disposição em paralelo entre interruptor e o atuador de acionamento (relé ou dimer). Percebe-se que em uma ligação residencial comum, o retorno sai diretamente do interruptor e segue para um dos contatos da lâmpada e posteriormente para o quadro de distribuição de circuitos (QDC), conforme a Figura 3.23. Por outro lado, em um edifício inteligente onde há a presença de um quadro de Automação (QA), o retorno é ligado no pino comum do relé, e esse conectado ao interruptor, conforme a Figura 3.24. Essa configuração cria dois caminhos para a corrente de acordo com as duas chaves. Dessa forma, caso um dos dois seja acionado, a lâmpada pode ser desativada pelo outro. Isso garante que, em uma possível queda do sistema de controle, a carga continua podendo ser ligada ou desligada de forma independente.

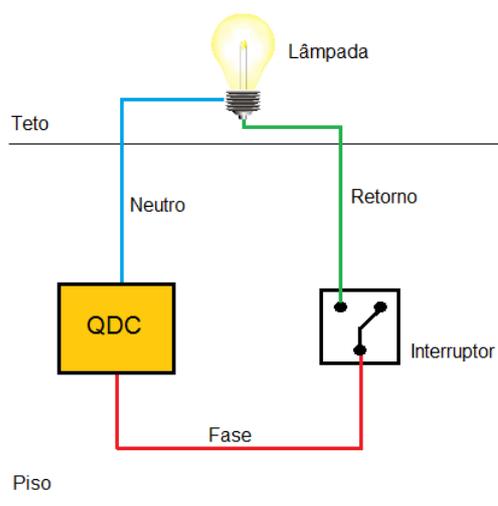


Figura 3.23: Instalação Convencional de lâmpadas

No que se refere ao controle de tomadas, o pressuposto continua o mesmo. Em uma instalação convencional, a alimentação da tomada é provida diretamente pela fase e neutro do QDC, conforme a Figura 3.25. Já o análogo se aplica quando há a presença de automação, Figura 3.26. O retorno é enviado ao QA e dessa forma pode-se ligá-la ou desligá-la a partir de um relé ou dimer.

De uma forma geral, nota-se que a disposição inicial dos contatos do relé deve ser levada em consideração. Para que o acionamento de uma lâmpada a partir de um interruptor ou de uma tomada aconteça de forma independente, é preciso que os contatos desses

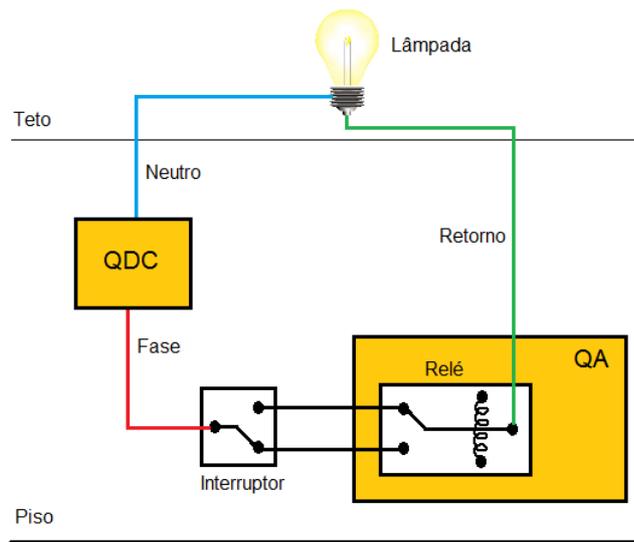


Figura 3.24: Instalação automatizada de lâmpadas

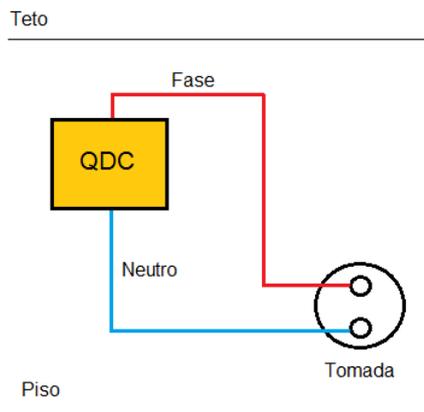


Figura 3.25: Instalação Convencional de tomadas

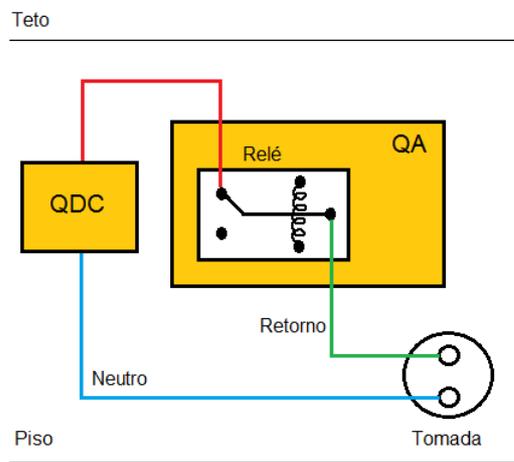


Figura 3.26: Instalação automatizada de tomadas

atuadores estejam inicialmente configurados como contatos normalmente abertos (NA). Isso porque permite a sincronização de estado entre carga e dispositivo.

### 3.1.10 Planta do Sistema

Para a verificação do funcionamento do sistema foi realizada a montagem física em miniatura da planta de uma residência. Similar à uma maquete sem a presença de paredes, essa planta consiste na fixação de todos os módulos desenvolvidos e dos dispositivos utilizados em uma mesa de madeira, conforme a Figura 3.27.

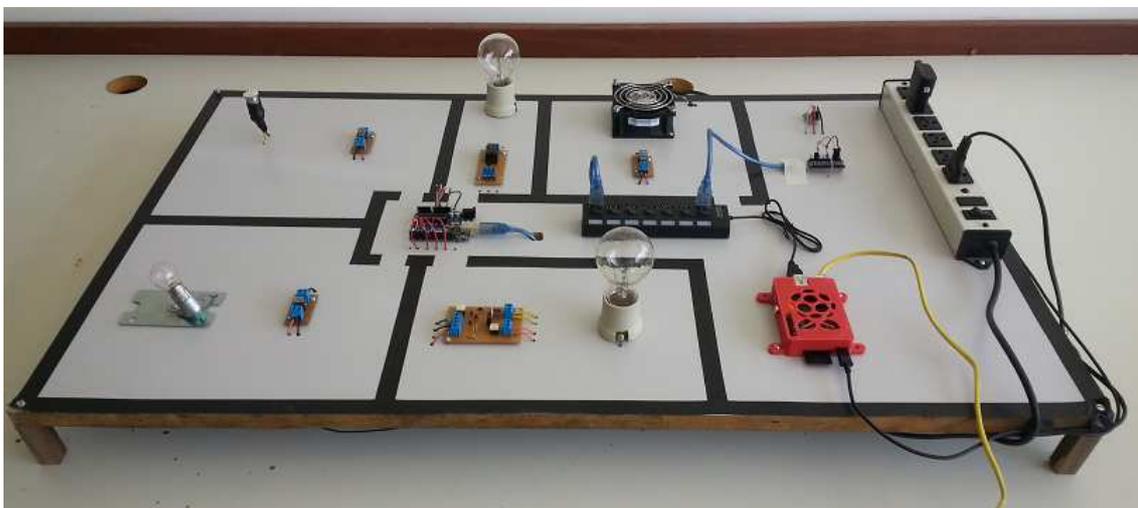


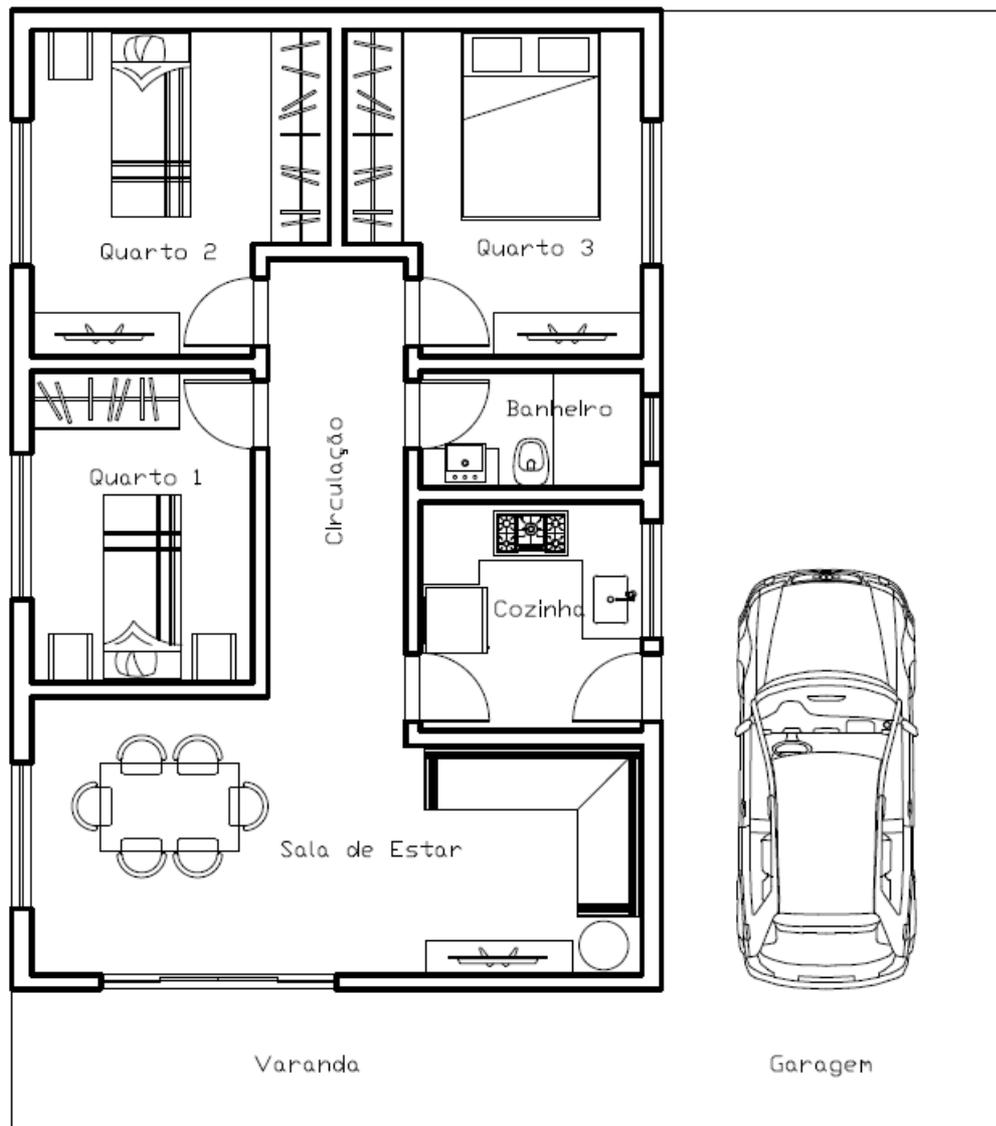
Figura 3.27: Montagem da planta desenvolvida.

A planta desenvolvida possui dimensões 900mm x 55mm x 20mm. Na parte superior estão concentrados todos os dispositivos presentes no sistema, sendo que toda fiação foi passada pela parte de baixo da tábua por motivos de visualização. Nota-se que as delimitações em preto representam as paredes externas e as que separam os cômodos. A planta original utilizada como modelo está apresentada na Figura 3.28.

## 3.2 Ferramentas

Conforme o planejamento de trabalho, após a elaboração do projeto e a devida obtenção dos recursos necessários, realizou-se o desenvolvimento dos algoritmos de controle e a programação do microcontrolador e microprocessador.

O ambiente de desenvolvimento integrado, ou *Integrated Development Environment* (IDE), usado para a construção dos códigos do sistema foi o *NetBeans* 8.0.2. A sua escolha é justificada por apresentar plataforma aberta para linguagens Java, C, C++, PHP e



PLANTA  
Esc.: 1/75

Figura 3.28: Planta residencial utilizada na planta.

diversas outras, além de sua capacidade em ser executado nos sistemas operacionais *Linux* e *Windows*.

O sistema de controle do projeto é composto por códigos programados nas linguagens HTML, JAVA, *JavaScript* (JS), *JavaServerPage* (JSP) e C/C++. Sua interligação ocorre segundo um fluxo de transmissão de informação de acordo o dado em questão. O percurso da informação pode ser esquematizado conforme apresentado na Figura 3.29.

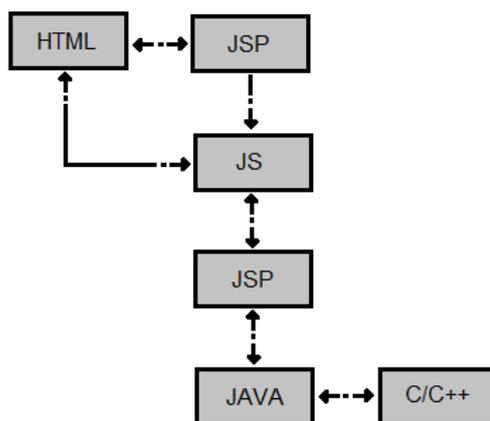


Figura 3.29: Fluxo de informação dos algoritmos do sistema.

Analisando a Figura 3.29 nota-se que algumas vias são bidirecionais. Isso significa que os dados podem ser escritos e lidos simultaneamente na execução do sistema. Seguindo o fluxo, começando com o código em HTML, primeiramente o usuário acessa o portal e tem acesso à uma página em HTML. Os botões presentes na página irão direcionar às páginas em JS que tratarão os dados seguindo a lógica dos arquivos em JSP. Posteriormente, os comandos são então repassados à outra página JSP que determinará quais rotinas devem ser executadas. Os dados escritos ou lidos, por precisarem estar sempre ativos nos canais de comunicação, necessitam da presença de uma classe em JAVA, que desempenha essa função. As informações são, por fim, repassadas à lógica dos microcontroladores programadas em C/C++, numa orientação à objetos. O **Apêndice A** apresenta a sintaxe cada documento.

### 3.2.1 Interface Principal

A página HTML, nomeada Controle de Sistemas Residenciais, realiza a interface principal com o usuário, como apresenta a Figura 3.30. Contendo elementos da versão HTML 5, essa página contém a planta da residência em questão e todos os dispositivos a serem controlados ilustrados. As informações apresentadas na página são o título (Sistemas Do-

móticos), o nome do autor, do professor orientador e o logo da instituição de ensino.

A página em HTML contém figuras de lâmpadas, ventiladores e sensores que possuem *hiperlink* que irão redirecionar o usuário para suas respectivas páginas de controle. Elas são todas JS, cada qual com uma caracterização específica, onde é possível acionar, controlar a intensidade e selecionar modos de operação pré-determinados. Essas páginas recebem os algoritmos regidos por páginas JSP, agindo de forma interativa com a lógica em questão.

### 3.2.2 Páginas de Controle

Clicando-se na lâmpada do Quarto 1 na Figura 3.30, abre-se uma nova página direcionada pelo *hiperlink* para sua interface de controle, como apresentado na Figura 3.31.

O sistema possui então três tipos de variáveis de entrada: um botão de acionamento, uma barra de valores e botões de cenário. O botão ligar/Desligar liga e desliga a lâmpada, a barra de valores quando arrastada aplica a intensidade correspondente, enquanto os botões de cenários aplicam intensidades pré-selecionadas na saída. Dessa forma há então três variáveis de saída: o mostrador de status, que pode assumir o valor *OFF* (desligado) e *ON* (ligado), a figura de uma lâmpada que fica amarela quando a lâmpada está ligada e cinza quando está desligada, e também um indicador de intensidade que varia de 0 a 100%.

O sistema inicia-se com a lâmpada desligada, conforme a Figura 3.32a, onde é possível notar o status igual a *OFF*, o desenho está desligado e a intensidade é 0. Quando o botão ‘Ligar’ é acionado, Figura 3.32d, a variável ‘STATUS’ comuta para *ON*, o desenho é de uma lâmpada ligada, a intensidade é 100% e o próprio botão se dispõe para ‘Desligar’. Caso o usuário não queira luminosidade total, é possível determinar a sua intensidade deslocando-se a barra ‘VALOR INTENSIDADE’. O valor da barra varia de 0 a 100 referentes a uma escala de 0 a 100% da luminosidade incandescente.

Para facilitar a escolha de intensidades diferentes, foram criados cenários de iluminação, sendo eles: romântico, cinema, leitura e festa. O cenário romântico equivale à lâmpada com intensidade igual a 50% do valor nominal, conforme a Figura 3.32c. O cenário cinema possui intensidade igual a 10%, Figura 3.32d, enquanto o cenário leitura equivale a 80%, Figura 3.32e, e festa 30%, Figura 3.32f. Para acessar os cenários basta clicar no botão referente ao desejado. Para desligá-lo é preciso clicar no botão de desligar ou então selecionar outro cenário desejado.

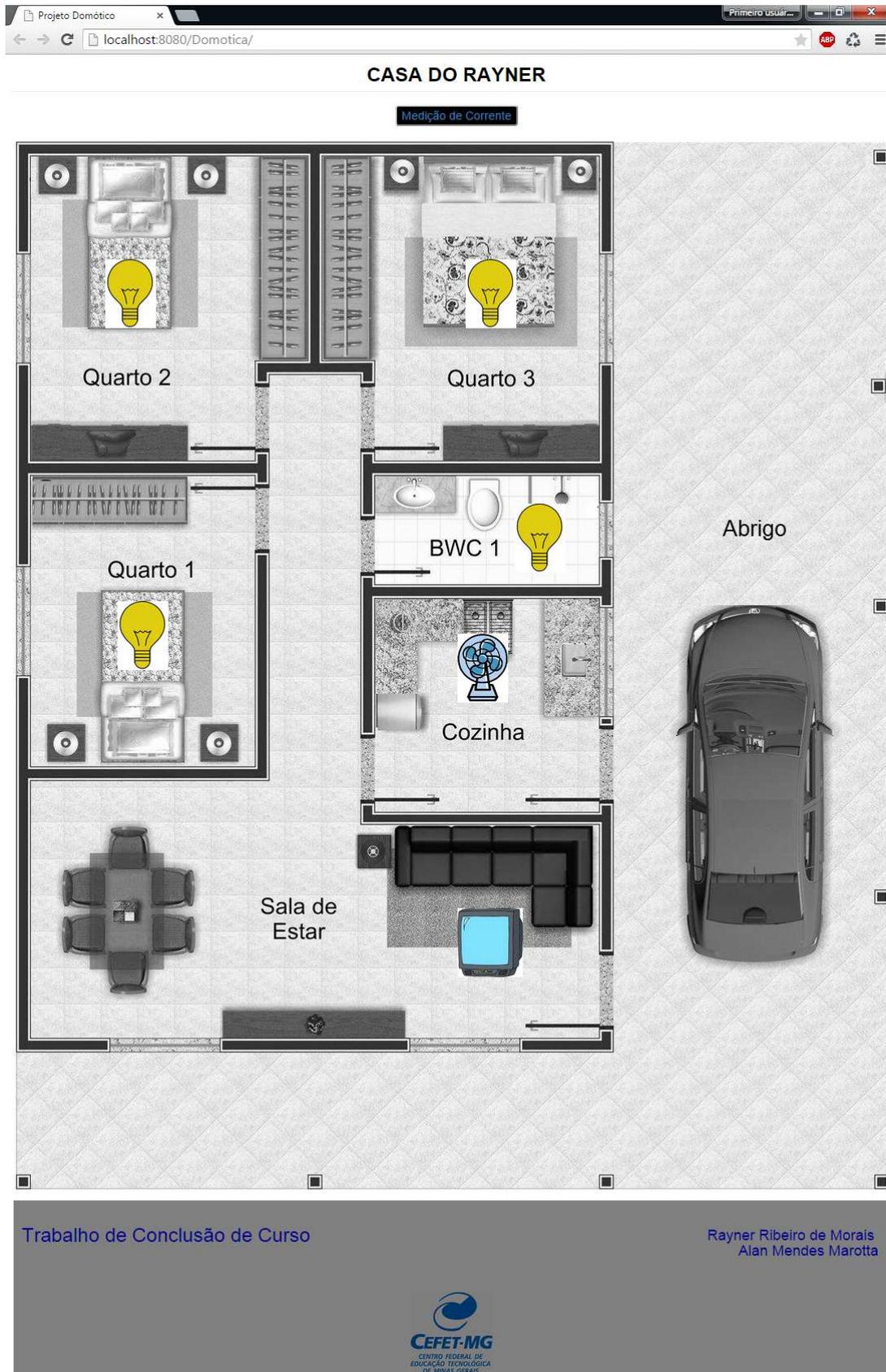


Figura 3.30: Página HTML - Controle de Sistemas Residenciais.



Figura 3.31: Página *JavaServer* - Controle de Iluminação.

Selecionando-se as lâmpadas dos quartos 2 e 3 da Figura 3.30 o mesmo processo se repete, abrindo então uma nova página por onde o sistema será controlado, conforme Figura 3.33. Nesse caso, a imagem ícone utilizada faz referência à uma lâmpada de corrente contínua. Dessa forma, a lâmpada presente no quarto 2 consiste em uma lâmpada led de 12V CC enquanto no quarto 3 encontra-se uma lâmpada incandescente também de 12V CC.

De forma análoga, o sistema começa com a lâmpada desligada, Figura 3.33a, com o status *OFF*, imagem ‘apagada’ e intensidade 0. Ao pressionar o botão ligar, Figura 3.33b, têm-se o status igual a *ON*, imagem ‘acesa’ e intensidade de 100%. A barra de intensidade também está disponível, Figura 3.33c. Nota-se que não foram criados cenários de iluminação para as lâmpadas de corrente contínua. A razão para essa decisão baseia-se no fato de promover uma comparação de rendimento em ambos casos, cujos resultados serão apresentados no Capítulo 4.

Clicando-se no ícone do ventilador que está situado na cozinha o mesmo processo se repete, abrindo então uma nova página por onde será controlado, conforme Figura 3.34.

As variáveis ‘STATUS’, ‘LIGAR’, ‘DESLIGAR’ e a barra de intensidade se compor-



(a) Lâmpada desligada (0%).



(b) Lâmpada ligada (100%).



(c) Cenário Romântico: lâmpada ligada (50%).



(d) Cenário Cinema: lâmpada ligada (10%).



(e) Cenário Leitura: lâmpada ligada (80%).



(f) Cenário Festa: lâmpada ligada (30%).

Figura 3.32: Controle de Iluminação de lâmpada incandescente.



(a) Lâmpada desligada (0%).



(b) Lâmpada ligada (100%).



(c) Lâmpada ligada (50%).

Figura 3.33: Controle de Iluminação de lâmpada de corrente contínua.

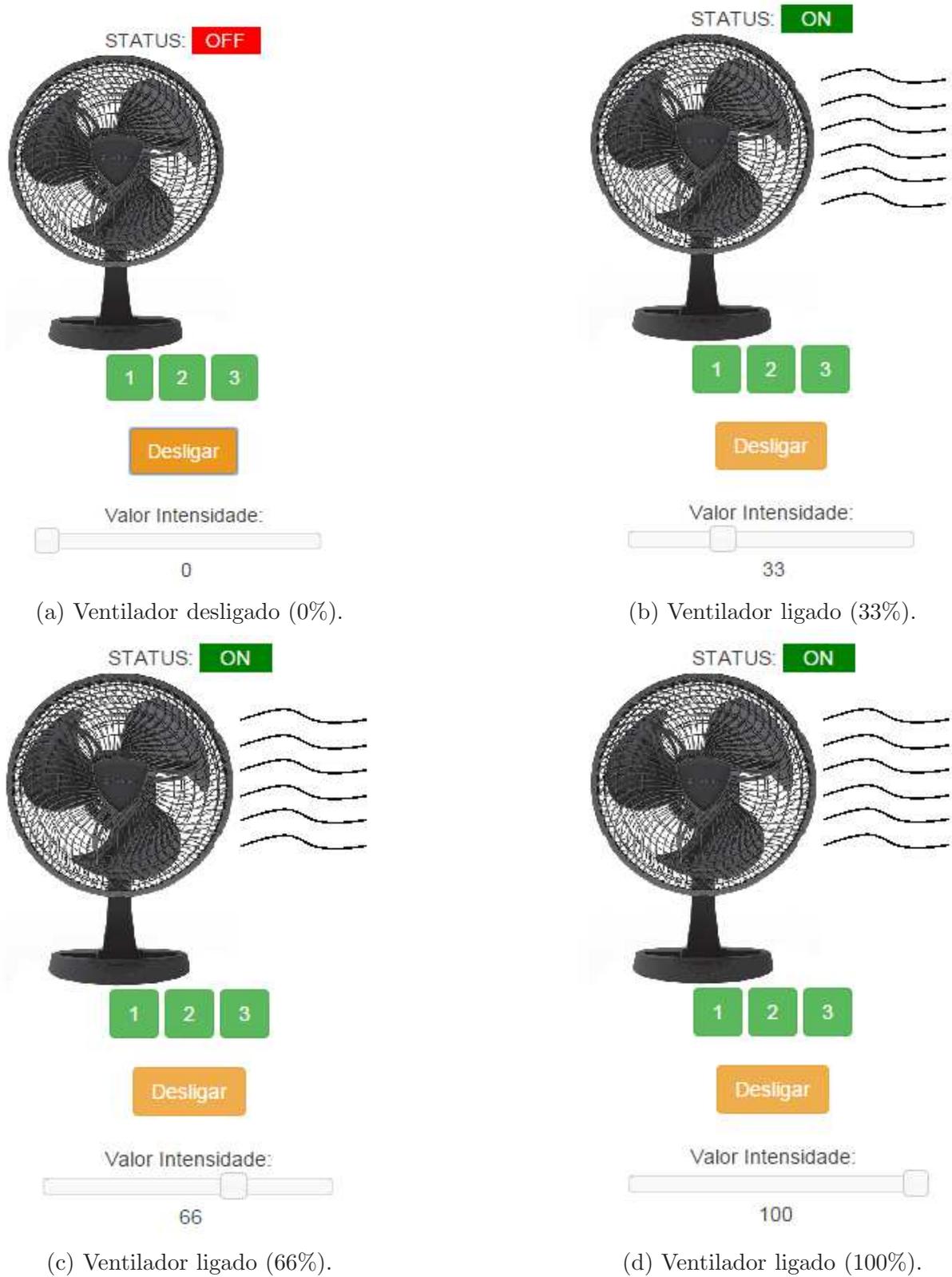


Figura 3.34: Controle de refrigeração.

tam da mesma forma que o apresentado nas análises anteriores. O ponto relevante no controle do ventilador está nos botões de potência nomeados “1”, “2” e “3”. Começando com o sistema desligado, Figura 3.34a, quando pressionado o botão ‘1’ é selecionada a intensidade de 33%, Figura 3.34b. O botão ‘2’ aciona o equipamento em 66%, Figura 3.34c, enquanto o botão ‘3’ se refere a 100%, Figura 3.34d.

Clicando-se no ícone da televisão que está situado na sala, abre-se a página de controle de entretenimento, conforme Figura 3.35.

### CONTROLE DE ENTRETENIMENTO



(a) Televisão desligada (0%).

### CONTROLE DE ENTRETENIMENTO



(b) Televisão ligada (100%).

Figura 3.35: Controle de entretenimento.

No que se refere ao entretenimento, pode-se ligar e desligar uma televisão. Existem apenas as variáveis ‘STATUS’, ‘LIGAR’ e ‘DESLIGAR’, desempenhando uma ação simples ao ser pressionado o botão liga/desliga. Existem apenas dois estados comutáveis para essa página, começando com o sistema desligado, Figura 3.35a. Quando o botão ‘Ligar’ é

pressionado, o ícone referente ao de uma imagem na tela comuta indicando que a televisão está ligada, Figura 3.35b.

Quando se deseja verificar o valor atual da corrente drenada pelo circuito, o usuário pode clicar no botão ‘Medição de Corrente’ que se encontra logo acima da planta, em posição de destaque. Quando requerido, o sistema direciona o navegador para outra página onde um gráfico irá aparecer, conforme a Figura 3.36.

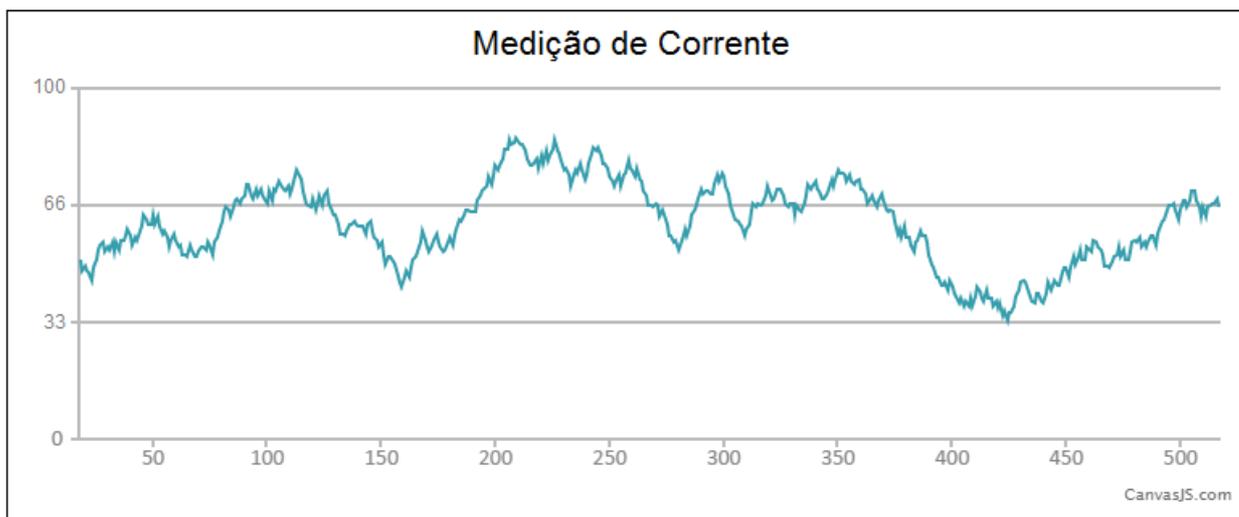


Figura 3.36: Página de medição de corrente.

Esse gráfico apresenta o valor da corrente ao longo do tempo. Todos os equipamentos do sistema estão em série com esse sensor de medição de corrente, fazendo com que qualquer variação de entrada seja demonstrada diretamente na saída. Isso garante que toda vez que determinado equipamento for ligado ou desligado, o gráfico irá ter sua forma de onda alterada, informando de forma perceptível ao usuário que houve variação no consumo de energia.

### 3.2.3 Controle pela Internet

Para a transmissão dos dados remotamente pela Internet, é necessário estabelecer um canal seguro, onde seja possível executar os aplicativos gráficos, transferir os arquivos e encapsular os protocolos. Isso é possível graças à criação de um tunel reverso SSH, onde todas as informações transitadas no processo vão diretamente do roteador para o ponto de acesso do usuário ou para a central de comando.

O grande desafio em estabelecer um portal pela Internet se baseia no fato de que o IP do sistema é dinâmico. Isso indica que, toda vez que a conexão com a *Web* é reinicializada, o IP muda automaticamente, indicando um endereço diferente. De certa forma há duas maneiras de se conseguir um IP fixo para o sistema:

- Contratação do serviço pelo provedor da Internet;
- Comprar (alugar) um *host* através de um distribuidor, por exemplo, o *website* Amazon.com

A primeira opção ainda é um recurso um tanto quanto inviável no país, já que os serviços populares de distribuição da Internet são caros e garantem a comunicação com uma qualidade bem abaixo da esperada. Já a segunda opção é de fato a mais sensata no que se trata de manutenção e acessibilidade. É possível alugar um *host* online, que pode ser pago mensalmente de acordo com a demanda do usuário.

Dessa forma, para estabelecer uma comunicação de IP variável entre o terminal e uma central com IP fixo, é necessário criar um canal no qual o trânsito das informações esteja sempre ativo. Isso é possível graças à criação de túnel SSH reverso, descrito no **Apêndice B** dessa monografia.

O processo consiste basicamente em criar um máquina virtual abrindo uma conexão SSH para uma porta remota e encaminhar então as informações para o destino especificado. Nas linhas de códigos, especifica-se a porta remota e a porta local, bem como o endereço de IP referente. Assim é possível acessar a porta 8080 de uma máquina que está na rede interna pela porta 80 do servidor remoto.

## Resultados e Discussões

Analisando as premissas do projeto e realizando uma comparação entre a metodologia atual e a fundamentação teórica, foi possível observar a viabilidade do sistema proposto por razões de economia de energia, redução de custo de implementação, menor retorno de investimento, aumento da durabilidade de equipamentos e alta capacidade de transferência de dados. Razões estas discriminadas a seguir:

### 4.1 Economia de Energia

No aspecto sustentável, a viabilidade do sistema proposto pode ser observada pela redução do consumo de energia, no que se refere à variação de intensidade, de uma lâmpada LED CC em comparação com uma incandescente CC, tal qual notado na comparação entre as Tabelas 4.1 e 4.2.

Tabela 4.1: Amostragem experimental da lâmpada de LED 12V CC.

Tensão(V)	Corrente(A)	Potência(W)	Intensidade(L)
1.2	0.05	0.06	2
2.4	0.10	0.05	5
3.6	0.16	0.57	7
4.8	0.20	0.96	10
6.0	0.26	1.56	13
7.2	0.30	2.16	15
8.4	0.37	3.18	18
9.6	0.42	4.03	20
10.8	0.50	5.40	23
12.0	0.65	7.80	26

Tabela 4.2: Amostragem experimental da lâmpada incandescente 12V CC.

Tensão(V)	Corrente(A)	Potência(W)	Intensidade(L)
1.2	0.76	0.91	0
2.4	1.13	2.71	0
3.6	1.41	5.07	0
4.8	1.70	8.16	2
6.0	1.94	11.64	6
7.2	2.14	15.40	15
8.4	2.32	19.32	21
9.6	2.51	24.09	38
10.8	2.65	28.62	50
12.0	2.83	33.96	76

Nota-se que para obter uma luminância de 15 lúmens, são necessários 15.40 Watts usando-se uma lâmpada incandescente enquanto a mesma intensidade é garantida com 2.16 Watts referentes ao de uma lâmpada LED. Com esses dados é possível perceber uma economia de energia igual a 7.12 vezes menor apenas utilizando-se o LED. De fato, a lâmpada incandescente tem um poder maior no que se refere à quantia de iluminação máxima, sendo que aos 100% oferece no mínimo 76 lúmens contra 26 do LED. No entanto, é perceptível o maior consumo de energia em qualquer patamar quando em comparação direta com o primeiro tipo de lâmpada.

Para comparar o rendimento relativo de uma lâmpada entre a potência consumida e luminosidade oferecida, realizou-se um experimento com os dois tipos de lâmpadas, conforme a Figura 4.1. A lâmpada de LED corresponde a uma lâmpada de um pólo 12V de 22 leds branco, Figura 4.1a, enquanto a incandescente é de 12V com 20W/A, Figura 4.1b.



(a) Lâmpada LED 12V.



(b) Lâmpada Incandescente 12V.

Figura 4.1: Lâmpadas de corrente contínua 12V de 1 pólo.

A experimentação consistiu em uma montagem utilizando uma haste para erguer a lâmpada até determinada altura e, a partir da variação de sua intensidade pelo do módulo de controle cc, medir sua potência e luminosidade consequente. Para neutralizar a influência do ambiente, o experimento foi realizado à noite em uma sala sem janelas, onde a penetração de luz exterior é praticamente desprezível. Essa condição garantiu que quando o sistema estivesse desligado, o valor de referência do luxímetro fosse igual a zero lúmens. A Figura 4.2 apresenta o esquema proposto e a Figura 4.3, a montagem propriamente realizada.

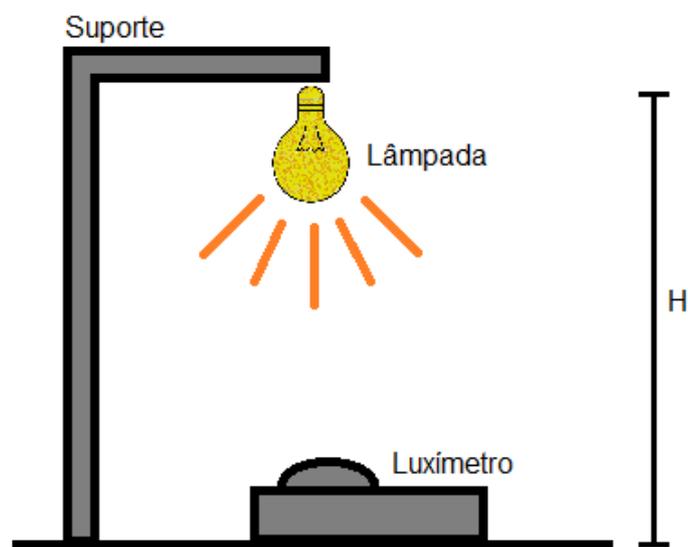


Figura 4.2: Esquema da montagem experimental.

Notou-se a partir da experimentação que a lâmpada LED possui maior eficiência relativa em comparação com a incandescente. Isso porque para determinada intensidade luminosa, o LED consegue consumir menos energia que a incandescente. Os resultados obtidos afirmam a fundamentação teórica, onde se afirma que a incorporação da automação em um residência representa ao morador uma economia considerável na conta de energia. Na experimentação realizada essa economia, apenas com iluminação, gira em torno de 85% sendo dependente da variação de intensidade da lâmpada. Em comparação com revisão da literatura, segundo Filho (2011), a redução total de energia é estimada em aproximadamente 30%. Esse valor condiz com os números obtidos, já que a iluminação corresponde à apenas 20% do consumo residencial e dessa forma, os 85% representam apenas 17%, acrescidos da economia em outros setores, como refrigeração e entretenimento.

De acordo com a Agência Internacional de Energia, IEA (*International Energy Agency* em inglês), 19% da produção global de energia é destinada à iluminação. Intuitivamente é



Figura 4.3: Montagem experimental.

possível perceber a importância no desenvolvimento de soluções de automação aplicadas à luminotécnica. Os recursos convencionais utilizados para auxiliar na gestão de energia dentro de uma residência variam desde sensores de presença à dimers que regulam a intensidade luminosa de lâmpadas, como o módulo que foi projetado. O controle programado de atividades evita que os equipamentos fiquem ligados quando não é necessário, garantindo que tudo aquilo que não está sendo utilizado desligue automaticamente.

Segundo a Aureside (2014), correspondendo à porcentagem de demanda, 50% dos projetos de automação são específicos ao controle de iluminação, que além do conforto e praticidade adquiridos, garantem uma significativa redução na conta de energia ao final do mês.

## 4.2 Redução do Custo de Implementação

O sistema proposto também se mostrou viável quanto ao aspecto financeiro, no que diz respeito à diminuição do custo de implementação. Considerando-se uma residência com as seguintes características:

- Equipamentos com comandos baseados em dados de redes sem fio;
- Central de gerenciamento;
- Iluminação: acendimento simples de lâmpadas e programação de cenários de luz;
- Entretenimento: ligar televisão e aparelhos de som quando o cliente chegar em casa;
- Segurança: enviar sinal ao usuário caso a porta da sala for aberta quando o alarme estiver acionado.

De acordo com a Tabela 3.2, o trabalho proposto apresenta um custo que varia em uma faixa que vai de R\$550,00 a R\$650,00. Esse valor, no entanto, pode aumentar segundo a necessidade de utilização de fios, conectores e também criação de módulos protetores (e.g. caixas de madeira) para que o circuito não fique exposto. Considerando que esses adicionais acrescentem R\$350,00, o preço total do trabalho sobe para R\$1.000,00. É preciso ressaltar ainda que esses valores não são de produção em larga escala, mas sim de produção artesanal.

A partir do levantamento de um orçamento digital realizado pelo portal de empresas especializadas na área, como a *Neocontrol* e *Myhouse Automação*, entre diversas outras,

o custo médio de implementação com as características acima listadas gira em torno de R\$4.000,00.

Comparando-se então as duas perspectivas, induz-se que o projeto em questão possui custo de implementação de aproximadamente 25% ao referente de uma empresa especializada. Para aqueles que demonstram interesse em automatizar a própria residência sem a intenção de recorrer à terceiros, essa estatística apresenta ser uma solução um tanto quanto viável.

No entanto, é preciso considerar como interpretação dessa análise fatores como mão de obra necessária, acesso ao cliente, divulgação e ferramentas utilizadas. As empresas prestadoras desse serviço certamente possuem uma qualidade de serviço refinada quando se comparada com os módulos artesanais fabricados aqui e também uma melhor habilidade em instalar e modificar as instalações elétricas em questão.

Logo, o interessante é investir nesse projeto caso o haja o devido interesse na redução do custo de implementação. O sistema proposto pode ser então incorporado à residências e também empresas que buscam um certo nível de automação. É possível, graças ao à utilização do raspberry uma alta taxa de transferência de dados por um baixo custo. Isso implica que vários sistema possa ser incrementados no sistema sem a necessidade de alteração na central de comando, ou seja, sem o aumento extra de custo.

## 4.3 Retorno de Investimento

O retorno de investimento é outro fator que justifica a incorporação do projeto desenvolvido à residências convencionais. Com um prazo de retorno de investimento igual a 20 meses, o projeto possui alto índice de rendimento e justifica os gastos de implementação investido pelo cliente em um tempo menor que 2 anos.

Considerando que a média de preço dos projetos de automação residencial esteja entre 2% e 3% do valor da residência e que a economia de energia calculada seja de 20% e 30%, segundo Tremel para a Folha de São Paulo (2013), pode-se calcular o retorno de investimento de um projeto da forma apresentada a seguir.

O investimento total (com ajustes de implementação e mão de obra) do projeto é de aproximadamente R\$1000,00. Além disso, considerando-se que a conta de energia média

de uma residência comum seja de R\$200,00, tem-se que a redução de gastos com a presença da domótica seja cerca de 25% ou R\$50,00.

O SEBRAE (2014), Serviço Brasileiro de Apoio às Micro e Pequenas Empresas, define ainda o Prazo de Retorno de Investimento, obtido sob a forma de unidade de tempo, conforme a Equação 4.1:

$$PRI = \frac{It}{B} \quad (4.1)$$

O PRI, corresponde à razão entre Investimento total aplicado e o valor dos benefícios mensais gerados pela aplicação. Para o projeto em questão, tem-se então que o PRI seja calculado como sendo:

$$PRI = \frac{1000}{50} = 20$$

Esse número indica que em vinte meses o gasto inicial investido seja convertido de volta ao bolso do cliente.

O cálculo feito na automação de uma residência, pode ainda ser quantificado pela análise ROI (*Return of Investment*). Ele consiste no retorno ou benefício total dividido pelo custo total aplicado, onde o retorno consiste no ganho subtraído do próprio custo, conforme a Equação 4.2.

$$ROI = \frac{L - Ci}{Ci} \quad (4.2)$$

O ROI corresponde à razão entre a diferença do lucro e o custo total de investimento pelo próprio custo de investimento. Assim, tem-se que o ROI para 2 anos após a implementação do projeto seja de:

$$ROI = \frac{50 * 24 - 1000}{1000} = 0.2$$

Esse valor indica que após dois anos de investimento o retorno financeiro terá sido de 20% de lucro acima do valor investido. Esse no entanto, representa um cálculo financeiro direto considerando a cotação atual do dólar e não levando em conta a inflação recorrente no período calculado.

De acordo com Greenwell (2011), dependendo das facilidades aplicadas e tecnologias selecionadas, estima-se que o ROI da automação residencial é considerado razoavelmente satisfatório. De forma análoga ao PRI, estima-se que em projetos atuais prestados por empresas sejam necessários aproximadamente 5 anos para que o cliente tenha retorno de investimento.

Dessa forma, estima-se então que o prazo de retorno de investimento do projeto ocorre de mais rapidamente do que o previsto pela revisão bibliográfica, resultando em um bom aproveitamento financeiro do sistema.

## 4.4 Durabilidade de Equipamentos

Com a premissa do controle automático e variação de intensidade, observou-se a diminuição da potência consumida pelos equipamentos, o que induz intuitivamente um aumento em sua vida útil. As lâmpadas incandescentes, quando acionadas e gerenciadas por um dimer, são submetidas à tensões menores do que a nominal, fato que lhe atribui diversos benefícios, conforme a Tabela 4.3.

Tabela 4.3: Características de lâmpadas incandescentes de acordo com a tensão (Adaptado de SANTOS, 2001).

Tensão (V)	Intensidade (%)	Potência (%)	Vida útil (%)	Eficiência relativa
120	100	100	100	1.00
110	75	88	300	0.85
100	55	76	1000	0.72
90	38	64	4000	0.59

É possível notar que a simples diminuição de tensão em pouco menos de 10% implica em uma vida útil três vezes maior do que quando operando a 100%. Em cenários onde é exigida uma baixa luminosidade, românticos, por exemplo, se for utilizada apenas 38%

da potência total, permite-se que essa durabilidade estenda-se em aproximadamente 40 vezes mais. No entanto, deve-se levar em consideração que a eficiência relativa da lâmpada diminui com a diminuição da tensão aplicada. Essa eficiência corresponde à razão entre luminosidade fornecida e potência consumida e, apesar de apresentar redução significativa, está sempre acima ao valor análogo da intensidade.

Além do controle de intensidade de uma lâmpada, o acionamento regulado por sensores de movimento garante que uma lâmpada ficará ligada apenas quando for necessário, ou seja, caso haja alguém no cômodo. Essa limitação de atuação reduz o tempo de atividade de trabalho dos equipamentos, fazendo com que se reduza o consumo médio.

Em corredores de circulação, onde o fluxo de transição de pessoas ocorre intermitentemente durante o dia e ocasionalmente à noite, a energia consumida pode de fato atingir reduções significativas em seu valor conforme citado por Muratori e Bó (2013).

## 4.5 Compactidade com Alta Taxa de Transferência de Dados

No que se refere à montagem física, a viabilidade da proposta pode ser observada através da compactidade dos módulos utilizados com a alta taxa de transferência de dados referente. As dimensões dos módulos em milímetros são:

- Raspberry PI: 85 x 56 x 20
- Módulo dimer: 75 x 50 x 25
- Módulo relé: 75 x 35 x 20
- Módulo controle CC: 45 x 25 x 25
- Atmega 328 (UNO): 65 x 50 x 10

Os módulos dimer, relé e controle CC foram produzidos artesanalmente através da impressão do circuito em placas de fenolite. Os correspondentes disponíveis no mercado apresentam basicamente as mesmas dimensões e até mesmo menores. No entanto, quando refere-se ao raspberry, as dimensões dos *laptops* concorrentes no mercado são bem maiores, como demonstrado na Figura 4.4.

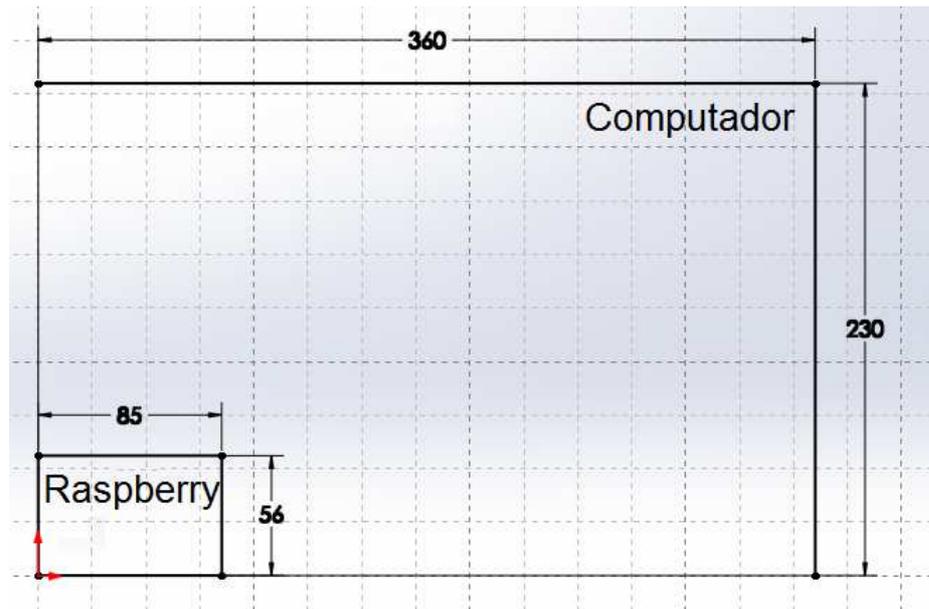


Figura 4.4: Dimensões de um Raspberry PI módulo B e Laptop de 17".

Em projetos de sistemas residenciais é necessário que a presença de circuitos e dispositivos aconteça da forma mais imperceptível possível. Isso porque qualquer exposição desnecessária pode resultar em acidentes, como na presença de crianças, e desconfiguração da montagem. Sendo assim, conforme comparado, o raspberry é a melhor opção quanto ao tamanho a ser instalado.

No que se refere ao poder de processamento de dados, o sistema embarcado também se mostra eficientemente capaz de suprir à necessidade de uma casa inteligente. Possuindo um sistema operacional próprio com processamento de 700MHz e 512MB de memória RAM, o raspberry consegue transmitir e manipular uma grande quantidade de dados em tempo real. Isso garante que a transmissão de vídeo em tempo real de câmeras de segurança pode estar disponíveis ao usuário, sendo que a qualidade da comunicação ficaria restrita apenas à resolução do roteador ou à velocidade de *upload* da *Internet*.

## Considerações Finais

Analisando as premissas do projeto e realizando uma comparação entre a metodologia atual e a fundamentação teórica, é possível apresentar as seguintes constatações.

### 5.1 Conclusão

Este trabalho apresentou o desenvolvimento de sistemas de domótica controlados por dispositivos móveis. O sistema apresentado faz referência aos módulos de controle existentes em residências, mas foi desenvolvido artesanalmente com o objetivo de suprir pequenas aplicações e com menor custo.

Nesse trabalho de conclusão de curso foi fundamentado, projetado e construído o sistema. Foram realizadas também a especificação de componentes eletrônicos necessários, construção dos módulos de controle, além de simulações eletrônicas e computacionais.

O intuito principal do trabalho foi de controlar sistemas residenciais individuais como o acionamento simples e de intensidade de lâmpadas, ventiladores e televisores, além de seu manuseio remoto através de dispositivos eletrônicos. Em conclusão aos testes realizados e à montagem concretizada, verificou-se que a utilização de um sistema embarcado como o raspberry apresenta tecnicamente a solução mais viável para a centralização de comando. A incorporação de objetos dentro de uma residência deve se apresentar da forma mais discreta possível e isso foi garantido utilizando-se esse mini microcomputador do tamanho de um telefone celular. Os módulos produzidos também possuem tamanho reduzido e que podem ser facilmente colocados dentro de paredes e próximos à caixas de passagem de fios e interruptores ou tomadas.

A implementação do sistema apresentou um grau satisfatório no que se refere à resiliência. É possível aderir o projeto desenvolvido à residências sem a perda de suas

características comuns. Com a configuração paralela entre atuadores e interruptores, é possível que o usuário tenha a comodidade em escolher de que forma deseja-se ligar os dispositivos de sua casa, não sendo necessariamente através da internet ou da rede local, mas também pelo próprio interruptor.

Algumas dificuldades foram encontradas para a construção do sistema. A primeira delas se refere ao cabeamento estruturado e a forma como as ligações físicas deveriam se dar. Apesar de definido a topologia em estrela como a mais eficiente na disposição dos equipamentos, a conexão entre os microcontroladores atmega ao raspberry por meio de portas USB se mostrou inviável em instalações reais. Isso acontece porque cabos USB a longas distâncias podem ser susceptíveis a ruídos e possuem alto custo de implementação.

## 5.2 Propostas de Continuidade

O desenvolvimento do projeto permitirá uma vasta lista de futuras implementações. Analisando a metodologia atual e realizando comparações entre os recursos utilizados e os disponíveis no mercado é possível apresentar algumas propostas para trabalhos futuros.

O sistema desenvolvido permite o usuário, com acesso à rede local, ligar, desligar e controlar a intensidade de equipamentos a partir de um ponto remoto, como celulares, *tablets* e *laptops*. Como primeira proposta, tem-se a implementação da Internet como rede principal de acesso pelo usuário. Correspondendo à premissa central do projeto em termos de controle remoto, a Internet é a rede universal de computadores e habilita o usuário em manusear os sistemas de sua casa a partir de qualquer lugar do mundo. É preciso salientar que a resiliência do sistema deve continuar, em outras palavras, deve-se poder controlar os sistemas a partir do acesso à mesma rede ou mesmo pelo interruptor. Isso garante que em casos de tempestades, por exemplo, possa conectar-se automaticamente à rede local.

No que se refere ao cabeamento estruturado, uma forma prática de se resolver a questão logística de disposição de cada equipamento, baseia-se em implementar uma interface sem fio na transmissão de dados entre raspberry e microcontroladores. O padrão *WiMax* pode ser uma solução, representando uma interface sem fio similar ao padrão *Wi-Fi*. Com sua utilização é possível uma velocidade de transferência de dados maior que 1 Gbit por segundo, oferecendo uma conexão de banda larga de alto desempenho (OLIVEIRA, VILLELA, MIYASHITA, 2012).

Com a presença do raspberry, possibilitou-se ainda uma grande capacidade de processamento de dados. Isso indica que grandes quantidades de dados tem a real oportunidade de serem enviados em tempo real, permitindo-se então que vídeos de câmeras de segurança possa estar disponíveis na tela do aparelho celular do usuário.

Além disso, como proposta de trabalho futuro, pode-se prever o estado atual do equipamento controlado. No momento, caso o usuário acesse o portal e a lâmpada de sua casa já estiver ligada, ela não aparece como ligada no sistema. Para ter certeza de que ela está ligada ou desligada deve-se pressionar o botão referente ao estado. Para que essa divergência seja solucionada, um sistema retificador pode ser colocado entre o ponto comum de um interruptor e o referencial do sistema. Dessa forma, quando o interruptor ou relé, ou dimer, estiver acionado, esse nó irá adquirir tensão não nula e enviará um sinal, depois de ser retificado, para o microcontrolador se referenciando ao estado no equipamento. Nota-se que esse circuito retificador é o mesmo utilizado no projeto do dimer, correspondendo à quatro diodos dispostos em ponte.

## Apêndice - Códigos

### A.1 Código HTML - Página Inicial

```
<HTML>
  <HEAD>
<title> Projeto Dom&oacute;tico </title>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1,
    maximum-scale=1, user-scalable=no"> <!-- Compatibilidade
    smartphone-->
  <script src="js/jquery.js"></script>
  <script src="js/jquery-ui.js"></script>
  <script src="js/bootstrap.min.js"></script>
  <link href="css/bootstrap.min.css" rel="stylesheet"/>
  <link href="css/jquery-ui.css" rel="stylesheet"/>
  <link href="css/jquery-ui.structure.css" rel="stylesheet"/>
  <link href="css/jquery-ui.theme.css" rel="stylesheet"/>
  </HEAD>

  <BODY bgcolor="#cfcfcf" align="center" >
<h1 align="center"> <b> <font size="5" face="arial" color="000000">
CASA DO RAYNER
</font> </b> </h1>
<hr size="2" width="100%" noshade>
<br>
  <div>
    <style scoped>
```

```

        .button-botao {
        color: white;
        border-radius: 4px;
        text-shadow: 0 1px 1px rgba(255, 255, 255, 0.2);
        background: rgb(0, 0, 0);
    }</style>
    <button class="button-botao pure-button"> <a href=
        "medidor.jsp">
        Medição de Corrente</a></button>
</div>
<br>

<br><br>
<div style="background-color:808080; color:00008b; padding:10px;">
<h3 align="left">
Trabalho de Conclusão de Curso
</h3>
<h4 align="right">
Rayner Ribeiro de Moraes
<br>
Alan Mendes Marotta
</h4>

</div>
<a href="quarto1.jsp">
<span style="position: absolute; top:720px; left:265px; width:5;
    height:5" >
<div class="css">
    
    </div>
</span>
    </a>
    <a href="quarto2.jsp">
<span style="position: absolute; top:300px; left:250px; width:5;
    height:5" >

```

```
<div class="css">
  
</div>
</span>
</a>
<a href="quarto3.jsp">
  <span style="position: absolute; top:300px; left:690px; width:5;
            height:5" >
    <div class="css">
      
    </div>
  </span>
</a>
<a href="banheiro.jsp">
  <span style="position: absolute; top:600px; left:750px; width:5;
            height:5" >
    <div class="css">
      
    </div>
  </span>
</a>
<a href="ventilador.jsp">
  <span style="position: absolute; top:760px; left:680px; width:5;
            height:5" >
    <div class="css">
      
    </div>
  </span>
</a>
<a href="televisao.jsp">
  <span style="position: absolute; top:1100px; left:680px; width:5;
            height:5" >
    <div class="css">
```

```

</div>
</span>
  </a>
</BODY>
```

## A.2 Código Java Server Page - Controle dos Sistemas Individuais

```
<!doctype html>
<html lang="en">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Controle dos Sistemas</title>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1,
      maximum-scale=1, user-scalable=no"> <!-- smartphone-->
    <script src="js/jquery.js"></script>
    <script src="js/jquery-ui.js"></script>
    <script src="js/bootstrap.min.js"></script>
    <script src="js/app.js"></script>
    <link href="css/bootstrap.min.css" rel="stylesheet"/>
    <link href="css/jquery-ui.css" rel="stylesheet"/>
    <link href="css/jquery-ui.structure.css" rel="stylesheet"/>
    <link href="css/jquery-ui.theme.css" rel="stylesheet"/>
    <link href="css/app.css" rel="stylesheet"/>
  </head>

  <body ALIGN="center">
    <H2>CONTROLE DOS SISTEMAS</H2>
    <BR>
    STATUS:
    <div id="status">OFF</div>
    <br>
    
    <br>
```

```

<button class="btn btn-success" id="ligar">Ligar</button>
<br><br>
Valor Intensidade:
<br>
<div id="barra"></div>
<br>
<div id="valorbarra"></div>
<br><br>
Cen&aacute;rios:
<br>
<button class="btn btn-warning" id="cenario1">CENARIO 1</button>
<br><br>
<button class="btn btn-warning" id="cenario2">CENARIO 2</button>
<br><br>
<button class="btn btn-warning" id="cenario3">CENARIO 3</button>
<br><br>
<button class="btn btn-warning" id="cenario4">CENARIO 4</button>
</body>
</html>

```

### A.3 Código Java Script - Tratamento do Controle

```

function desligageral () {
    $('#estado').attr('src', 'img/imagem_off.png');
    $('#status').html('OFF').css('background-color', 'red');
    $('#ligar').html('Ligar').removeClass("btn-danger").
        addClass("btn-success");
}

function ligageral () {
    $('#estado').attr('src', 'img/imagem_on.png');
    $('#status').html('ON').css('background-color', 'green');
    $('#ligar').html('Desligar').removeClass("btn-success").
        addClass("btn-danger");
}

$(document).ready(function () {
    window.light = false;

```

```
$('#ligar').click(function () {
    if (window.light) {
        jQuery.post("arduino.jsp", {
            estado: "A"
        },
        function(){
            desligageral();
            $("#barra").slider( "option", "value", 0 );
        });
    } else {
        jQuery.post("arduino.jsp", {
            estado: "a"
        },
        function(){
            $("#barra").slider( "option", "value", 100 );
            ligageral();
        });
    }
    window.light1 = !window.light1;
});
$('#barra').slider({max: 100, min: 0, animate: true});
$("#barra").slider({
    change: function (event, ui) {
        jQuery.post("arduino.jsp", {
            estado: 'S',
            valor: Math.ceil(ui.value * 0.94)+" "
        }, function (data) {
            console.log(ui.value);
            if (ui.value === 0) {
                if (window.light1)
                    $("#ligar").click();
            } else {
                desligageral();
                ligageral();
                window.light = true;
            }
        });
    });
```

```
        $("#valorbarra").html(ui.value);
    });
    CENARIO01.inten = false;
    $('#cenario1').click(function () {
        jQuery.post("arduino.jsp", {
            estado: "d"
        },
        function (data) {
            $("#barra").slider( "option", "value", 50 );
            console.log(data);
        });
    });
    CENARIO02.inten = false;
    $('#cenario2').click(function () {
        jQuery.post("arduino.jsp", {
            estado: "f"
        },
        function (data) {
            $("#barra").slider( "option", "value", 10 );
            console.log(data);
        });
    });
    CENARIO03.inten = false;
    $('#cenario3').click(function () {
        jQuery.post("arduino.jsp", {
            estado: "g"
        },
        function (data) {
            $("#barra").slider( "option", "value", 80 );
            console.log(data);
        });
    });
    CENARIO04.inten = false;
    $('#cenario4').click(function () {
        jQuery.post("arduino.jsp", {
            estado: "h"
        },
```

```
        function (data) {
            $("#barra").slider( "option", "value", 30 );
            console.log(data);
        });
    });
});
```

## A.4 Código Cascading Style Sheet - Visual dos Elementos

```
#status {
width: 50px;
display: inline-block;
background-color: red;
color: white;
}
#barra {
width: 200px;
background-color: grey;
display: inline-block;
}
```

## A.5 Código Java Server Page - Medição

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>
<html>

<head>
    <script src="js/jquery.js" type="text/javascript"></script>
    <script src="js/canvasjs.min.js" type="text/javascript"></script>
    <script src="js/app_med.js" type="text/javascript"></script>
</head>
<body>
<div id="chartContainer" style="height: 300px; width:100%;">
</div>

    <center>
    <input type='text' id='instante' />
```

```

    </center>
</body>
</html>

```

## A.6 Código Java Script - Gráfico de Consumo

```

$(function () {
    var dps = [];
    var x = 0;
    var chart = new CanvasJS.Chart("chartContainer", {
        title: {
            text: "Consumo de Energia"
        },
        data: [{
            type: "line",
            dataPoints: dps
        }]
    });
    setInterval(function () {
        $.get('arduino_3.jsp', function (data) {
            var current = data.current;
            if (current !== null && current !== undefined) {
                dps.push({
                    y: current,
                    x: x
                });
                $('#instante').val(current);
                x++;
                chart.render();
            }
        });
    }, 1000);
});

```

## A.7 Código Java Server Page - Comunicação com Microcontrolador

```

<%@page import="br.cefet.rayner.Arduino"%>

```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%
    Arduino arduino = new Arduino();
    String estado = (String) request.getParameter("estado");
    char c = estado.charAt(0);
    switch (c) {
        // ===== QUARTO 1 =====
        case 'a':
            arduino.write('a');
            break;
        case 'A':
            arduino.write('A');
            break;
        case 'S': {
            String valor = (String) request.getParameter("valor");
            int v = Integer.parseInt(valor);
            arduino.write('1');
            arduino.write((byte) v);
            break;
        }
        case 'd':
            arduino.write('d');
            break;
        case 'f':
            arduino.write('f');
            break;
        case 'g':
            arduino.write('g');
            break;
        case 'h':
            arduino.write('h');
            break;
        // ===== QUARTO 2 =====
        case 'Z':
            arduino.write('Z');
            break;
        case 'z':
```

```
        arduino.write('z');
        break;
case 'X': {
    String valor = (String) request.getParameter("valor");
    int v2 = Integer.parseInt(valor);
    arduino.write('2');
    arduino.write((byte) v2);
    break;
}
// ===== QUARTO 3 =====
case 'C':
    arduino.write('C');
    break;
case 'c':
    arduino.write('c');
    break;
case 'V': {
    String valor = (String) request.getParameter("valor");
    int v3 = Integer.parseInt(valor);
    arduino.write('3');
    arduino.write((byte) v3);
    break;
}
// ===== BANHEIRO =====
case 'o':
    arduino.write('o');
    break;
case '0':
    arduino.write('0');
    break;
// ===== VENTILADOR =====
case 'Q':
    arduino.write('Q');
    break;
case 'W': {
    String valor = (String) request.getParameter("valor");
    int v5 = Integer.parseInt(valor);
```

```

        arduino.write('5');
        arduino.write((byte) v5);
        break;
    }
    case 'e':
        arduino.write('e');
        break;
    case 'r':
        arduino.write('r');
        break;
    case 't':
        arduino.write('t');
        break;
    // ===== TELEVISAO =====
    case 'u':
        arduino.write('u');
        break;
    case 'U':
        arduino.write('U');
        break;
}
%>

```

## A.8 Código Java - Comunicação da porta serial

```

package br.cefet.rayner;
import java.util.logging.Level;
import java.util.logging.Logger;
import jssc.SerialPort;
import jssc.SerialPortException;
public class Arduino {
    private static SerialPort serial_port = null;
    public Arduino() {
        if (serial_port == null) {
            //serial_port = new SerialPort("COM3");
            serial_port = new SerialPort("/dev/ttyACM0");
            try {

```

```
        synchronized (this) {
            serial_port.openPort();
            serial_port.setParams(9600, 8, 1, 0);
            wait(2000);
        }
    } catch (InterruptedException | SerialPortException ex) {
        Logger.getLogger(Arduino.class.getName()).log(Level.SEVERE,
            null, ex);
    }
}

public void write(byte b) {
    try {
        serial_port.writeByte(b);
    } catch (SerialPortException ex) {
        Logger.getLogger(Arduino.class.getName()).log(Level.SEVERE,
            null, ex);
    }
}

public void write(char c) {
    write((byte) c);
}
}
```

## A.9 Código C++ - Microcontrolador 1

```
String inString = "";
volatile int zero = 0;
volatile int power = 0;
int dimer = 0;
int mosfet2 = 0;
int mosfet3 = 0;
int rele4 = 0;
int mosfet5 = 0;

void setup(){
    Serial.begin(9600);
```

```
pinMode(4, OUTPUT);
pinMode(5, OUTPUT);
pinMode(6, OUTPUT);
pinMode(9, OUTPUT);
pinMode(12, OUTPUT);
attachInterrupt(0, passagem_zero, RISING);
}

void loop(){
  if (zero == 1){
    int tempo_desparo =(83*(100-dimer));
    delayMicroseconds(tempo_desparo);
    digitalWrite(4, HIGH);
    delayMicroseconds(8.333);
    digitalWrite(4, LOW);
    zero = 0;
  }
  if (Serial.available()){
    power = Serial.read();
    //=====Dimer QUARTO 1=====
    if (power == 'a'){
      dimer = 93;
    }else if (power == 'A'){
      dimer = 0;
    }else if (power == 'd'){
      dimer = 50;
    }else if (power == 'f'){
      dimer = 10;
    }else if (power == 'g'){
      dimer = 80;
    }else if (power == 'h'){
      dimer = 30;
    }
    //=====Mosfet QUARTO 2=====
    else if (power == 'z'){
      analogWrite(5, 255);
    }else if (power == 'Z'){
```

```
    analogWrite(5, 0);
}
//=====Mosfet QUARTO 3=====
else if (power == 'c'){
    analogWrite(6, 255);
}else if (power == 'C'){
    analogWrite(6, 0);
}
//=====Mosfet BANHEIRO=====
else if (power == 'o'){
    digitalWrite(12, HIGH);
}else if (power == 'O'){
    digitalWrite(12, LOW);
}
//=====Mosfet VENTILADOR=====
else if (power == 'Q'){
    analogWrite(9, 0);
}else if (power == 'e'){
    analogWrite(9, 85);
}else if (power == 'r'){
    analogWrite(9, 170);
}else if (power == 't'){
    analogWrite(9, 255);
}
//=====INTENSIDADE=====
else if (teste(power))
{
    int lampada = power;
    int inChar = 0;
    delay(100);
    while(Serial.available()){
        power = Serial.read();
        if (power >= '0' && power <= '9'){
            inChar = (inChar*10) + (power - '0');
        }
    }
    if (lampada == '1'){
```

```
        dimer = inChar;
    }else if (lampada == '2'){
        mosfet2 = inChar*255/100;
        analogWrite(5, mosfet2);
    }else if (lampada == '3'){
        mosfet3 = inChar*255/100;
        analogWrite(6, mosfet3);
    }else if (lampada == '5'){
        mosfet5 = inChar*255/100;
        analogWrite(9, mosfet5);
    }
}
}
}
void passagem_zero(){
    zero = 1;
}
int teste (char carac){
    if (isDigit(carac)){
        return 1;
    }
    else{
        return 0;
    }
}
```

## A.10 Código C++ - Microcontrolador 2

```
int AnalogInputPin = A0;
void setup(){
    Serial.begin(9600);
}

void loop(){
    if (Serial.available() && Serial.read() == 'j'){
        double Current = currentSensor(analogRead(AnalogInputPin));
        printDouble(Current, 2);
    }
}
```

```

    }
}

void printDouble(double val, byte precision){
    Serial.print (int(val));
    if( precision > 0){
        Serial.print(".");
        unsigned long frac, mult = 1;
        byte padding = precision -1;
        while(precision--) mult *=10;
        if(val >= 0){
            frac=(val-int(val))*mult;
        }
        else{
            frac=(int(val)-val)*mult;
        }
        unsigned long frac1 = frac;
        while(frac1 /= 10) padding--;
        while(padding--) Serial.print("0");
        Serial.print(frac,DEC) ;
    }
}

double currentSensor(int RawADC){
    int    Sensitivity    = 66;
    long   InternalVcc    = readInternalVcc();
    double ZeroCurrentVcc = InternalVcc / 2;
    double SensedVoltage  = (RawADC * InternalVcc) / 1024;
    double Difference     = SensedVoltage - ZeroCurrentVcc;
    double SensedCurrent  = Difference / Sensitivity;
    return SensedCurrent;
}

long readInternalVcc(){
    long result;
    ADMUX = _BV(REFS0) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
    delay(2);
    ADCSRA |= _BV(ADSC);
    while (bit_is_set(ADCSRA,ADSC));
}

```

```
result = ADCL;
result |= ADCH<<8;
result = 1126400L / result;
return result;
}
```

## Apêndice - Configuração Raspberry Pi

Esse apêndice apresenta um guia prático de instalação e configuração do Raspberry Pi, Figura B.1.

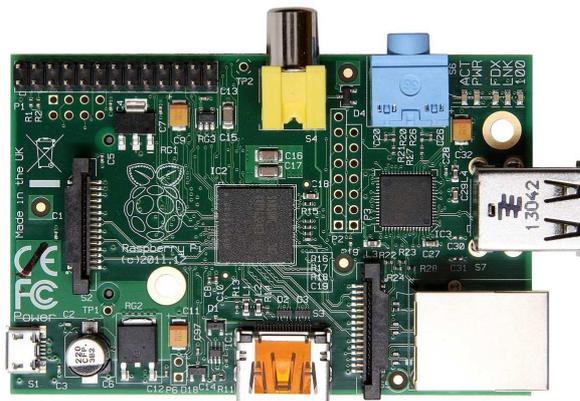


Figura B.1: Raspberry Pi.

### B.1 Especificações

O modelo B Rev. 2 do Raspberry possui as seguintes especificações:

- Sistema com processador Broadcom BCM2835 ARM11 de 700Mhz (desempenho similar ao de um Pentium 2 300 MHz);
- Videocore 4 Processadores Gráficos integrados Unit (GPU) capaz de reproduzir 1080p em Full-HD como um Blu-Ray em Alta qualidade de video;
- 512 MB de RAM
- Um livre, versátil e altamente revelador amigável GNU/Linux sistema operacional Debian
- 2 x Portas USB

- Saída de vídeo HDMI
- Saída de Vídeo RCA
- Saída de áudio Jack 3,5 milímetros
- Porta Ethernet 10/100Mb para acesso à Internet
- 5V Micro USB para entrada de energia
- SD, MMC, SDIO flash Slot para cartão de memória
- 26 pinos 2.54 milímetros, Slot de expansão Header (que permitem periféricos e placas de expansão)

## B.2 Hardware

O modelo B possui duas portas USB, entrada para cartão SD, saídas de áudio, saídas de vídeo RCA e HDMI, entrada ethernet, conectores gerais de entradas e saídas e conector de energia micro USB. A Figura B.2 apresenta um esquema do hardware com suas consequentes conexões.

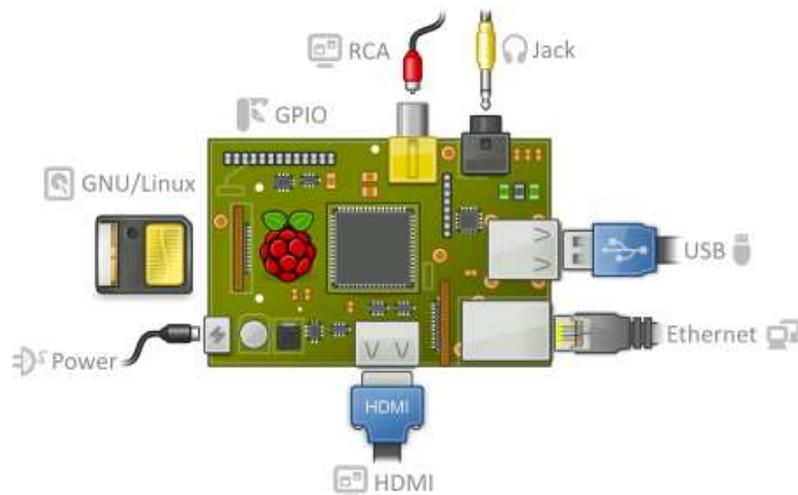


Figura B.2: Hardware do Raspberry Pi.

Dessa forma, para a configuração do Raspberry Pi são necessários os seguintes equipamentos:

- Cartão SD - recomendado cartões classe 4 com 8GB ou superior;
- Monitor - com entradas HDMI/DVI ou RCA;

- Teclado e mouse - com entrada USB;
- Fonte de Energia - carregador (e.g. celular) com saída 5V e 500mA;

É recomendado ter à disposição também:

- Conexão com a *Internet*, através de cabo de comunicação *Ethernet* ou adaptador USB Wi-Fi;
- Extensor (*Hub*) USB;
- Caixa protetora de acrílico;

## B.3 Instalação do Sistema Operacional

Para utilizar o equipamento é preciso instalar um sistema operacional (SO). Pelo fato de ser alimentado por um processador de arquitetura ARM existem alguns SO's específicos baseados em linux disponíveis gratuitamente na aba “*downloads*” pelo site oficial da franquia <[www.raspberrypi.org](http://www.raspberrypi.org)>. São eles:

- Raspbian (Debian Wheezy);
- Snappy Ubuntu Core;
- Openelec;
- RaspBMC;
- Pidora;
- Risc OS.

O projeto foi desenvolvido utilizando-se o **Raspbian** como SO por ser considerado o mais fácil de usar e simples de instalar, Figura B.1. Depois de ter sido concluído o download, recomenda-se que o arquivo seja movido para um diretório apropriado, clique com o botão direito e selecione “*Extrair tudo*” - isso irá extrair o arquivo imagem do arquivo comprimido.



Figura B.3: Raspbian.

Além disso, é preciso utilizar um programa que seja capaz de transferir o sistema operacional para o cartão de memória de forma que o raspberry possa inicializar a partir dele. Um programa recomendado é o **Win 32 Disk Imager** que pode ser obtido também gratuitamente.

Deve-se então inserir o cartão SD no computador e formatá-lo. É necessário que o formato do novo disco seja FAT-32 e que o tamanho da unidade de alocação seja de 4096 bytes. Posteriormente, abrindo o Win 32 Disk Imager, selecione o diretório do Raspbian que foi baixado e clique no arquivo imagem, tendo absoluta certeza que o driver selecionado é o próprio cartão de memória. Por fim, basta clicar em “Write” e aguardar o processo, que costuma levar alguns minutos. Ver Figura B.2.



Figura B.4: Programa Win 32 Disk Manager.

Ao final dessa etapa, o SO Raspbian foi devidamente instalado no cartão SD. No entanto, ele ainda não está configurado.

## B.4 Primeiro BOOT

Uma vez que o cartão de memória foi previamente preparado, conecte-o ao Raspberry juntamente com os outros periféricos. Ligando pela primeira vez, o sistema é iniciado no modo de configuração, conforme a Figura B.3. Nesse ponto é possível redimensionar o

tamanho da partição utilizada do cartão, selecionar o fuso horário, definir o tipo de teclado, entre outros. Caso o sistema não entre em modo de configuração automaticamente, digite-se o seguinte comando:

```
-> pi@raspberrypi ~ $ sudo rasp-config
```

```
Raspi-config
info Information about this tool
expand_rootfs Expand root partition to fill SD card
overscan Change overscan
configure_keyboard Set keyboard layout
change_pass Change password for 'pi' user
change_locale Set locale
change_timezone Set timezone
memory_split Change memory split
ssh Enable or disable ssh server
boot_behaviour Start desktop on boot?
update Try to upgrade rasp-config

<Select> <Finish>
```

Figura B.5: Modo de configuração do Raspberry Pi.

Nesse ponto, é possível reinicializar o sistema, sendo que na próxima inicialização todas as configurações estarão aplicadas. Para realizar o log in, usa-se as seguintes credenciais:

```
-> pi@raspberrypi ~ $ user: pi
-> pi@raspberrypi ~ $ password: raspberrypi
```

Após inserir o login, obtém-se acesso à linha de comandos. Para utilizar a interface gráfica de usuário ou GRU (*Graphical User Interface*) deve-se digitar:

```
-> pi@raspberrypi ~ $ startx
```

Com todas as configurações finalizadas, o sistema prossegue com sua inicialização. Ao fim do processo, o OS apresenta o ambiente de área de trabalho, ou LXDE (*Lightweight X11 Desktop Environment*), conforme a Figura B.4.



Figura B.6: Ambiente de área de trabalho do Raspberry Pi.

## B.5 Atualização do Sistema

Posteriormente à fase de instalação, recomenda-se que sejam feitas todas atualizações do sistema disponíveis afim de evitar falhas de incompatibilidade entre os softwares a serem utilizados. São digitados os seguintes comandos no LXTerminal:

```
-> pi@raspberrypi ~ $ sudo apt-get update
-> pi@raspberrypi ~ $ sudo apt-get upgrade
-> pi@raspberrypi ~ $ sudo apt-get install build-essential
```

O comando referente para desligar o sistema com segurança é:

```
-> pi@raspberrypi ~ $ sudo halt
```

ou

```
-> pi@raspberrypi ~ $ sudo shutdown -h
```

E para reiniciar:

```
-> pi@raspberrypi ~ $ sudo shutdown -r now
```

ou

```
-> pi@raspberrypi ~ $ sudo reboot
```

## B.6 Instalação de Programas

Todos os programas podem ser instalados por linhas de comandos através do terminal. É importante a ressalva de que antes de qualquer instalação deve-se atualizar o sistema.

### *Microcontrolador:*

Primeiramente será realizada a obtenção do software para o microcontrolador, no caso o Atmega 328. Uma das plataformas disponíveis de controle é o Arduino.

Para instalar a IDE do arduino:

```
-> pi@raspberrypi ~ $ sudo apt-get install arduino
```

Para instalar as bibliotecas:

```
-> pi@raspberrypi ~ $ sudo apt-get install avr-libc libftdi1 avrdude
    openjdk-6-jre librx-tx-java
```

### *Servidor:*

O servidor selecionado é o Apache Tomcat 7. Para sua utilização é preciso instalar também a plataforma Java.

```
-> pi@raspberrypi ~ $ sudo echo "deb http://ppa.launchpad.net/webupd8team/java/
    unbutu precise main" > /etc/apt/sources.list.d/webupd8team-java.list
-> pi@raspberrypi ~ $ sudo echo "deb-src http://ppa.launchpad.net/webupd8team/
    java/unbutu precise main" >> /etc/apt/sources.list.d/webupd8team-java.list
-> pi@raspberrypi ~ $ sudo apt-get key adv --keyserver keyserver.unbutu.com
    --recv-keys EEA14886
-> pi@raspberrypi ~ $ sudo apt-get install oracle-java7-installer
-> pi@raspberrypi ~ $ sudo java -version
```

Java foi instalado no diretório `/usr/lib/jvm/java-7-oracle`. Pode ser útil reservar um usuário reservado para utilizar o Tomcat, nesse caso:

```
-> pi@raspberrypi ~ $ sudo adduser \
    --system \
    --shell /bin/bash \
    --gecos 'Tomcat Java Servlet and JSP engine' \
```

```

--group \
--disable-password \
--home /home/tomcat \
tomcat

```

Realizando o download e descompactando o Tomcat:

```

-> pi@raspberrypi ~ $ cd /tmp
-> pi@raspberrypi ~ $ wget http://mirror.cogentco.com/pub/apache/tomcat/
    tomcat-7/v7.0.37/bin/apache-tomcat-7.0.37.tar.gz
-> pi@raspberrypi ~ $ tar xvzf ./apache-tomcat-7.0.37.tar.gz

```

Movendo a distribuição para /usr/share/tomcat7:

```

-> pi@raspberrypi ~ $ sudo mkdir /usr/share/tomcat7
-> pi@raspberrypi ~ $ /tmp/apache-tomcat-7.0.37 /usr/share/tomcat7

```

## B.7 Transferência de Arquivos (*DEPLOY*)

A programação das atividades do sistema foram desenvolvidas sob a plataforma Netbeans em um computador externo e posteriormente repassados ao raspberry. Os códigos em Java, JavaScript, JavaServer e Html, uma vez alterados podem ser implantados no sistema através do processo de *deploy* pelo gerenciador de aplicativos online quando conectados à mesma rede. Para isso, basta o usuário inserir no navegador a seguinte url:

```
http://[Endereço de IP do raspberrypi]:8080/manager/html
```

Para visualizar esse endereço de IP:

```
-> pi@raspberrypi ~ $ ifconfig
```

O processo de *deploy* baseia-se simplesmente em enviar o projeto para o raspberry sem a necessidade de um meio de transição física através das portas USB. Para isso deve-se, primeiramente, realizar o *undeploy* dos arquivos já existentes, ou seja, deletá-los do diretório. Caso isso não ocorra, o gerenciador informará que já existe um arquivo presente nesse destino. Feito isso, torna-se possível enviar e reenviar o projeto quando necessário, por exemplo em caso de alteração dos códigos. O arquivo a ser enviado encontra-se geralmente no seguinte diretório do computador hospedeiro do Netbeans:

```
C:\Users\PC-name\Documents\NetBeansProjects\Projeto\dist
```

O arquivo possui extensão `.war`, podendo ser interpretado como um arquivo compactado contendo todos os códigos interpretados. Ao recebê-lo, o raspberry irá descompactar todos os arquivos afiliados e salvá-los dentro do diretório apropriado ao Apache. O diretório correpondente consiste em:

```
-> pi@raspberrypi /var/lib/tomcat7/webapps
```

## B.8 Controle pela Internet

Para habilitar o controle remoto utilizando a Internet é preciso criar um túnel reverso em SSH de forma a estabelecer um canal de comunicação entre ponto de acesso e roteador. Para isso, basta utilizar uma única linha de comando como se segue:

```
-> pi@raspberrypi sudo ssh -R (porta remota):localhost:(porta local) sourceuser@endereço
```

No entanto, esse comando deve ser aplicado toda vez que a comunicação com a Internet seja interrompida. Pode-se estabelecer uma alternativa para esse canal de comunicação através de um controle automático através de outro comando, tal que:

```
-> pi@raspberrypi sudo autossh [-f] [ssh options]
```

Instala-se então um programa para começar um cópia do SSH e monitorá-lo, reiniciando o sistema sempre que necessário quando o sistema cair ou parar de transmitir as informações vinculadas.

## B.9 Execução do Projeto

O acesso do projeto é possível à todos usuários que estejam conectados à mesma rede do raspberry e possuam em mãos também o seu endereço de IP. Com isso, basta digitar no navegador a url:

```
http://[Endereço de IP do raspberry]:8080/Projeto
```

É possível estabelecer uma forma mais direta para esse acesso, no qual seja necessário entrar apenas como o endereço de IP. Para isso deve-se implementar um acesso proxy para o Apache. É realizado então um redirecionamento da porta 80, principal, para a 8080, onde se encontram os arquivos Apache como diretório padrão. Inicialmente deve-se inserir as seguintes linhas de comando:

```

-> pi@raspberrypi ~ $ cd /etc/apache2/sites-enabled          #diretório especificado
-> pi@raspberrypi ~ $ sudo a2enmod proxy                    #habilita acesso proxy
-> pi@raspberrypi ~ $ sudo vim 000-nome                     #edita e nomeia o proxy

```

Com isso, é possível alterar o proxy. Ele deve conter a seguinte estrutura:

```

<VirtualHost *:80>
    ProxyRequests On
    ProxyPreserveHost On
    <Proxy *>
        Require all granted                                #apenas para versão 2.4
    </Proxy>
    ProxyPass / http://127.0.0.1:8080/Domotica/
    ProxyPassReverse / http://127.0.0.1:8080/Domotica/
    <Location />
        Require all granted                                #apenas para versão 2.4
    </Location>
</VirtualHost>

```

Para sair do editor basta usar :wq e então reiniciar o servidor:

```

-> pi@raspberrypi ~ $ sudo service apache2 restart
-> pi@raspberrypi ~ $ sudo service tomcat7 restart

```

A partir desse ponto basta digitar-se então a seguinte url no navegador para rodar o projeto remotamente:

```
http://[Endereço de IP do raspberrypi]
```

É possível visualizar todas as ações de comando enviadas pelo usuário através do comando:

```
-> pi@raspberrypi ~ $ tail -f /var/log/apache2/*
```

Para visualizar todos os periféricos inseridos no sistema insere-se:

```
-> pi@raspberrypi ~ $ sudo ls /dev/tty*
```

## B.10 Comandos Úteis

Alteração de senha de acesso do usuário (recomendado antes de ligar o sistema à *Internet*):

```
-> pi@raspberrypi ~ $ sudo passwd pi          (inserir duas vezes)
-> pi@raspberrypi ~ $ sudo passwd root       (inserir duas vezes)
```

Informações do sistema:

```
-> pi@raspberrypi ~ $ cat /proc/version      (Versão do OS)
-> pi@raspberrypi ~ $ cat /proc/cpuinfo     (Informações sobre a CPU )
-> pi@raspberrypi ~ $ cat /proc/meminfo     (Informações sobre a memória)
-> pi@raspberrypi ~ $ cat /proc/partitions  (Memória de cada partição)
-> pi@raspberrypi ~ $ vcgencmd measure_temp (Temperatura da CPU)
-> pi@raspberrypi ~ $ lsusb                (Lista de periféricos conectados)
```

Procurar arquivos (caso não saiba o nome, basta digitar a extensão):

```
-> pi@raspberrypi ~ $ sudo find / -name *(nome do arquivo.extensão)
```

# Referências

- AURESIDE. Mercado de automação residencial deve atingir cinquenta bilhões de dólares ainda em dois mil e quatorze. *Associação Brasileira de Automação Residencial*, [S.l.], 2014.
- BARBOSA, A. F. V. Desenvolvimento de dispositivo de Domótica de baixo custo visando economia elétrica e segurança residencial. *Centro Federal de Educação Tecnológica de Minas Gerais*, [S.l.], 2014.
- BLEY, F. B. Leds versus lâmpadas convencionais viabilizando a troca. *Revista Especialize IPOG*, [S.l.], 2012.
- CAMARA, R. Protocolo I2C. *Universidade Federal do Vale do São Francisco*, [S.l.], 2013.
- CARNEIRO, L. F.; BRAGA, N. C.; JÚNIOR, N. A. Redes de Computadores. –, [S.l.], 1999.
- CUGNASCA, C. E.; HIRAKAWA, A. R. Comunicação Serial. *Universidade de São Paulo*, [S.l.], 2006.
- DELGADO, M. Five Smart Home Technologies that will Save you Money. *Freshhome Design and Architecture Magazine*, [S.l.], 2013.
- DIAS, C. L. A.; PIZZOLATO, N. D. Domótica Aplicabilidade e Sistemas de Automação Residencial. –, [S.l.], 2009.
- EMARKETER. Smartphones users worldwide will total 1.75 billion in 2014. *Emarketer*, [S.l.], 2014.
- ENERDATA. Global energy statistical yearbook 2014. *Enerdata*, [S.l.], 2014.
- FILHO, A. Sua Casa mais Moderna. *Jornal A Gazeta*, [S.l.], 2011.
- FILHO, A. V. Energia elétrica no Brasil contexto atual e perspectivas. *Revista Interesse Nacional*, [S.l.], 2013.

- FREESCALE, S. Multimedia Applications Processor Reference Manual. *Freescale Semiconductor*, [S.l.], 2011.
- GERMANY. Ar condicionado e aquecimento do setor no Brasil está crescendo. *Germany Trade and Invest*, [S.l.], 2013.
- GREENWELL, J. Smart building automation return of investment. *Revista Automate Buildings*, [S.l.], 2011.
- GREGGIANIN, C. A. *et al.* Estudo comparativo entre lâmpadas: incandescentes, fluorescentes compactas e led. *Espaço Energia*, [S.l.], 2013.
- GUERBER, C. Apostila TCP IP. *Laboratório de redes. Universidade do Contestado*, [S.l.], 2008.
- HEGGESTUEN, J. One in every 5 people in the world will own a smartphone. *Revista Business Insider*, [S.l.], 2013.
- JÚNIOR, A. L. C. Redes sem fio Protocolo Bluetooth aplicado em interconexão entre dispositivos. *Teleco Inteligência em Telecomunicações*, [S.l.], 2012.
- MACEDO, G. P. Domótica Uma análise de protocolos de automação predial residencial. *Universidade Federal de Ouro Preto*, [S.l.], 2009.
- MARIN, P. S. Automação residencial visão geral e aplicações. –, [S.l.], 2009.
- MEDEIROS, H. Casa do Futuro. *Revista Técnica*, [S.l.], 2009.
- MIRANDA, P. A integração pede Parâmetros. *Revista Capa*, [S.l.], 2008.
- MIZUSAKI, L. E. P. Comparação de Mecanismos de Comunicação para casa Inteligente. *Universidade do Rio Grande do Sul*, [S.l.], 2009.
- MORGAN, B. House of the Future: how automation tech is transforming the home. *Forbes Magazine*, [S.l.], 2013.
- MURATORI, J. R. Instalações Elétricas com Novas Abordagens. *Revista Setor Elétrico*, [S.l.], 2008.
- MURATORI, J. R.; Bó, P. H. Automação Residencial Conceitos e Aplicações. *Revista Capa*, Rio de Janeiro, 2013.
- NISKIER, J.; MACINTYRE, A. J. *Instalações Elétricas*. 4.ed. Rio de Janeiro: LTC, 2000.

- OLIVEIRA, S. R. A. L.; VILLELA, L. V.; MIYASHITA, R. Estudo da Difusão Tecnológica no setor de Telecomunicações WiMax. *Revista de Gestão e Operações Produtivas*, [S.l.], v.3, 2012.
- PEGN. Empresa fatura 1 milhão com serviços de automação residencial. *Pequenas Empresas Grandes Negócios*, [S.l.], 2013.
- PEREIRA, L. A. M. Automação Residencial rumo a um futuro pleno de novas soluções. *VII Congresso Internacional de Automação, Sistemas e Instrumentação*, São Paulo, 2007.
- PESQUISA ENERGÉTICA EPE, E. de. Anuário estatístico de energia elétrica de 2013. *Ministério de Minas e Energia*, [S.l.], 2014.
- PESSOA, J. L. N.; GHISI, E.; LAMBERTS, R. Estado da arte em eficiência energética: iluminação e envoltória. *Centro Brasileiro de Eficiência Energética em Edificações*, Florianópolis, 2013.
- PETRILLI, E. L.; SILVA, F. H.; SILVA, L. V. Tipos de lâmpadas. *Faculdade de Tecnologia de Tatuí*, Tatuí, 2012.
- PRADELLA, A. Automação residencial que cabe no bolso. *Revista Home Theater e Casa Digital*, [S.l.], 2013.
- REIS, G. H. R. Rede de Computadores. *Instituto Federal de Educação Tecnológica de Minas Gerais*, [S.l.], 2012.
- RODRIGUES, C. S. F. Análise da aplicação de um sistema domótico no desempenho térmico de frações autônomas de habitação face ao sistema de certificação energética. –, [S.l.], 2012.
- SANTOS, L. C. S. Sistema eletrônico de alto desempenho com baixa distorção harmônica para controle de intensidade luminosa de lâmpadas incandescentes de alta potência. *Universidade Federal de Santa Catarina*, Florianópolis, 2001.
- SENA, D. C. S. Automação residencial. *Departamento de Engenharia Elétrica Universidade Federal do Espírito Santo*, [S.l.], 2005.
- SILVA, I. V. F.; CARVALHO, S. S. Domótica uma abordagem sobre redes e soluções microprocessadas de baixo custo. –, [S.l.], 2011.
- TANENBAUM, A. S. *Redes de Computadores*. 4.ed. São Paulo: Elsevier, 2003.

- TEZA, V. R. Alguns aspectos sobre a automação residencial - Domótica. *Universidade Federal de Santa Catarina*, Florianópolis, 2002.
- TREMEL, D. Automação residencial quer ganhar morador pela economia. *Folha de São Paulo*, São Paulo, 2013.
- WILLRICH, R. Linguagens de Programação. *Universidade Federal de Lavras*, [S.l.], 2015.
- ZEINDIN, D. C. A. *et al.* A tecnologia do futuro Wi-Fi Wireless Fidelity. *Universidade Regional de Blumenau*, [S.l.], 2003.