

Universidade do Estado da Bahia - UNEB
Departamento de Ciências Exatas e da Terra
Colegiado de Sistema de Informação

ROGÉRIO DO CARMO CONCEIÇÃO

**UM AMBIENTE ABERTO BASEADO EM
MICROCONTROLADOR PARA
EXPERIMENTOS DIDÁTICOS DE FÍSICA:
CONVERSÃO FOTOVOLTAICA**

Salvador
2017

ROGÉRIO DO CARMO CONCEIÇÃO

**UM AMBIENTE ABERTO BASEADO EM
MICROCONTROLADOR PARA EXPERIMENTOS
DIDÁTICOS DE FÍSICA: CONVERSÃO
FOTOVOLTAICA**

Monografia apresentada à banca do curso de Bacharelado em Sistemas de Informação da Universidade do Estado da Bahia – UNEB, como requisito parcial para obtenção do grau de bacharel em Sistemas de Informação, área de concentração: Sistemas Digitais.

Orientador: Prof. Dr. Cláudio Alves de Amorim

Salvador

2017

ROGÉRIO DO CARMO CONCEIÇÃO

**UM AMBIENTE ABERTO BASEADO EM
MICROCONTROLADOR PARA EXPERIMENTOS
DIDÁTICOS DE FÍSICA: CONVERSÃO
FOTOVOLTAICA**

Monografia apresentada à banca do curso de Bacharelado em Sistemas de Informação da Universidade do Estado da Bahia – UNEB, como requisito parcial para obtenção do grau de bacharel em Sistemas de Informação, área de concentração: Sistemas Digitais.

Orientador: Prof. Dr. Cláudio Alves de Amorim

Trabalho aprovado. Salvador, 29 de novembro de 2017:

Prof. Dr. Cláudio Alves de Amorim
Universidade do Estado da Bahia - UNEB

Prof. Dr. Leandro Santos Coelho de Souza
Universidade do Estado da Bahia - UNEB

Prof. MSc. Julian Hermogenes Quezada Celedon
Universidade do Estado da Bahia - UNEB

*Este trabalho é dedicado à minha amada esposa, Tereza Cristina, e aos meus amigos que
compartilharam comigo a aventura de tornar um sonho em realidade.*

Agradecimentos

Sou grato a Tereza Cristina, minha esposa e companheira, por sempre ter estado ao meu redor no decorrer de toda a minha jornada na universidade, protegendo-me, na vida, de todas adversidades e incentivando-me a ser uma pessoa construtiva que sempre tenta fazer o seu melhor em tudo. E, além disso, por ter sido extremamente paciente e acolhedora mediante o hábito que desenvolvi e mantive durante esse último ano em que estive trabalhando neste projeto, falar sobre este projeto em todas as oportunidades. Expresso, também, a minha gratidão aos meus amigos e amigas, irmãos e irmãs do grupo Haras (Luiz Eduardo, Lorena Pereira, Débora Oliveira, Matheus Moreira, Íris Ribeiro, Thiago Andrade, Danilo Simas e Calison Santos), os quais, tornaram-se minha família, por terem dividido comigo os melhores e piores momentos da minha jornada acadêmica e por sempre terem demonstrado disposição para ajudar quando necessário.

Agradeço ao Prof. Dr. Cláudio Alves de Amorim, meu orientador neste projeto, por ter sido fonte de inspiração, aconselhamento e orientação, sempre demonstrando, através de suas ações, seu interesse sincero em que o projeto ocorresse da forma mais eficiente possível. Sou grato, também, pelo esmero com o qual atuou na idealização, planejamento, execução e validação deste projeto, fazendo-se presente em todos os momentos necessários, e quando não fisicamente, o fez de maneira irrestrita através dos meios de comunicação eletrônicos. Fico agradecido, por todos os conselhos oferecidos, principalmente aqueles direcionados para a vida fora da universidade, os quais, foram imprescindíveis e ajudaram-me, não somente a prosseguir e concluir este projeto, mas também, a enxergar o mundo com outros olhos. Ao amigo Luiz Eduardo, escrevo os meus mais sinceros agradecimentos por todas as vezes em que se dispôs a sentar e discutir nuances deste projeto, por ter ajudado no desenvolvimento do Sistema Gestor do Experimento, por ter ficado 8 horas seguidas dentro de uma sala, no dia de nosso aniversário (pois nascemos no mesmo dia do mês de novembro), preparando e realizando experimentos e, no dia seguinte, de forma voluntária, ter retornado para participar da análise dos resultados, ajudando com suas observações sempre pertinentes.

Agradeço de forma especial ao Coordenador do curso de Bacharelado em Sistemas de Informação, Prof. Dr. Leandro Santos Coelho de Souza, que com sua dedicação irrestrita ao processo de construção do aparato, tornou possível o uso da impressão 3D com recursos da universidade, ademais, por ter feito a modelagem em 2D do segundo projeto do suporte pra fonte luminosa (jamais esquecerei esse dia), utilizada pelo aluno Marcelo Alcântara – a quem dirijo também minha gratidão – para modelar e imprimir em 3D, no espaço Mandacaru Lab, e por sua participação ativa no processo de montagem do aparato experimental, oferecendo sugestões acerca de materiais, modos de utilização dos recursos

disponíveis e orientações acerca da melhor maneira de realizar a montagem, assim como, por ter ajudado na execução, propriamente dita, da montagem desse aparato, motivos pelos quais, este trabalho pôde ser concluído. Estendo esse agradecimento à Prof.^a Msc. Débora Alcina Rego Chaves por ter intermediado a negociação junto ao colegiado do curso de Design a fim de obter o material de insumo para realização da impressão em 3D.

Por fim, agradeço ao aluno Matheus Tanure, integrante do Mandacaru Lab, por ter disposto de tempo, dada a sua agenda de viagens e demais projetos que realiza pelo Mandacaru Lab no período em que este trabalho foi realizado, além das suas obrigações acadêmicas, para discutir comigo e modelar em 3D o primeiro projeto de suporte para fonte luminosa.

“Uma coisa que aprendi numa longa vida: que toda a nossa ciência, confrontada com a realidade, é primitiva e infantil - e no entanto é o que temos de mais precioso.”

(Albert Einstein)

Resumo

O presente trabalho foi realizado com o objetivo de desenvolver e validar um ambiente aberto (software e hardware) para experimento didático em física usando o dispositivo microcontrolado Arduino e a plataforma computacional Raspberry Pi, sob a condição de que esse ambiente deve ser capaz de minimizar o esforço com o monitoramento dos experimentos e registro dos dados, automatizando essas tarefas. Como prova de conceito dessa concepção de aparato experimental aberto e automatizado, foram projetados e construídos: um aparato experimental, material de baixo custo e impressão 3D, um sistema de monitoramento e coleta de dados executado no arduino e, por fim, um sistema de gestão do experimento executado no Raspberry Pi. Comunicando-se de forma serial em tempo real. Para validar essa concepção de aparato experimental foram realizados experimentos de conversão fotovoltaica utilizando o aparato construído, e os resultados obtidos foram aferidos com a teoria. Após a análise dos resultados obtidos, constatou-se que o objetivo deste trabalho foi alcançado e, do ponto de vista didático, demonstrou-se que a utilização de aparato experimental aberto e automatizado no processo de experimentação, permite revelar nuances expressivas acerca do fenômeno estudado. Ademais, ficou demonstrado como prova de conceito o gerenciamento do experimento em ambiente amigável em computador de baixíssimo custo.

Palavras-chave: Experimento em física. Microcontroladores. Arduino. Raspberry Pi. Laboratório didático. Automação de experimentos.

Abstract

This work was executed with the goal of developing and validating an open environment (software e hardware) for didactic experiment in physics using the Arduino microcontrolled device and the computational platform Raspberry Pi, under the condition that: this environment should be able to minimize the effort with monitoring experiments and recording data by automating them. As proof of concept of this open and automated experimental apparatus design, the following were designed and constructed: an experimental apparatus using modern techniques, low cost material and 3D printing, a system for monitoring and collecting data, performed on arduino and an experiment management system running on Raspberry Pi, where, these two systems communicate serially in real time. In order to validate this project were performed photovoltaic conversion experiments using the apparatus constructed and the results obtained were checked with the theory. After analyzing the results obtained, it was verified that the goal of this work was achieved and, from a didactic point of view, it has been shown that the use of open and automated experimental apparatus in testing process, allows to reveal significant hints about the studied phenomenon. Moreover, it was demonstrated as proof of concept the management of the experiment in friendly environment in computer of very low cost.

Key-words: Physics Teaching. Experiment in Physics. Microcontrollers. Arduino. Raspberry Pi. Didactic laboratory. Automation of experiments.

Lista de ilustrações

Figura 1 – Estrutura da célula fotovoltaica	21
Figura 2 – Raspberry Pi 3 Model B	24
Figura 3 – Limites mínimos e máximos dentro da faixa de tensões para que a tensão no circuito B represente um dígito binário.	29
Figura 4 – Formato apresentado pelo pulso ideal e o formato de um pulso real. . .	30
Figura 5 – Circuito RC antes e durante o processo de carga.	31
Figura 6 – Diagrama de bloco do sistema de conversão analógico-digital.	32
Figura 7 – Gráfico do processo de quantização.	33
Figura 8 – Placa Arduino UNO R3	35
Figura 9 – Arquitetura do aparato experimental.	39
Figura 10 – Diagrama esquemático do circuito RC.	40
Figura 11 – Tela principal do SGE	45
Figura 12 – Tela de feedback da execução do experimento	46
Figura 13 – Tela de Resultado do SGE	46
Figura 14 – Gráfico típico da curva de carga do capacitor no circuito RC	54
Figura 15 – Compilação de fotos do Aparato Experimental	62
Figura 16 – Fluxo de execução do SMCD.	66

Lista de tabelas

Tabela 1 – Comparativo dos aspectos das tecnologias utilizadas pelas usinas h liotérmicas	18
Tabela 2 – Comprimentos de onda do espectro visível e suas respectivas cores	20
Tabela 3 – Valores de tensão e corrente para diferentes resistores	52
Tabela 4 – Média dos valores da bateria de experimentos com $2k\Omega$	67
Tabela 5 – Média dos valores da bateria de experimentos com $12k\Omega$	68
Tabela 6 – Conjunto de ensaios da primeira bateria de experimentos	69
Tabela 7 – Conjunto de ensaios da segunda bateria de experimentos	71

Lista de abreviaturas e siglas

DC	<i>Direct Current</i>
AC	<i>Alternating Current</i>
SGE	Sistema Gestor de Experimentos
SMCD	Sistema de Monitoramento e Coleta de Dados
Pi	Raspberry Pi
CMOS	<i>Complementary Metal Oxide Semiconductor</i>
ADC	<i>Analog-to-Digital Converter</i>
A/D	<i>Analógico-Digital</i>

Lista de símbolos

Ω	Ohm
$k\Omega$	Kiloohm
V	Volt
A	Ampere
W	Watt
F	Farad
μF	Microfarad
nF	Nanofarad
pF	Picofarad
λ	Comprimento de Onda
K	Temperatura Termodinâmica
m	Metro
cm	Centímetro
nm	Nanómetro
θ	Função Trabalho da Superfície Metálica
eV	Elétron-volt
°C	Grau Celsius
C	Capacitância
τ	Constante de Tempo do Circuito RC

Sumário

1	INTRODUÇÃO	15
2	CONVERSÃO DE ENERGIA LUMINOSA EM ENERGIA ELÉTRICA	17
2.1	Formas de conversão	17
2.1.1	Conversão Indireta	17
2.1.2	Conversão Direta	18
2.2	Efeito fotoelétrico	19
2.3	Células fotovoltaicas	21
3	RASPBERRY PI EM AMBIENTES DIDÁTICOS	23
4	CIRCUITOS ELETRÔNICOS	26
4.1	Componentes eletrônicos	26
4.2	Circuitos digitais	28
4.3	Circuito Analógico RC	30
4.4	Conversão analógico-digital	32
5	DISPOSITIVO MICROCONTROLADO ARDUINO	34
6	AMBIENTE ABERTO PARA EXPERIMENTO DE CONVERSÃO FOTOVOLTAICA	38
6.1	Desenho do Experimento	38
6.1.1	Circuito RC e Arduino	39
6.1.2	Desenvolvimento do Sistema de Monitoramento e Coleta de Dados	40
6.2	Gestão do Experimento e Apresentação dos Resultados	45
6.2.1	Desenvolvimento do Sistema Gestor de Experimentos	46
7	EXPERIMENTAÇÃO E VALIDAÇÃO DOS RESULTADOS	52
7.1	Testes Preliminares	52
7.2	Baterias de Experimentos	53
8	ANÁLISE DOS RESULTADOS	56
9	CONCLUSÃO E TRABALHOS FUTUROS	58
	REFERÊNCIAS	60

APÊNDICE A – APARATO EXPERIMENTAL	62
APÊNDICE B – ROTINAS DO SISTEMA DE MONITORAMENTO E COLETA DE DADOS	63
APÊNDICE C – FLUXOGRAMA DO SMCD	66
APÊNDICE D – DADOS DOS EXPERIMENTOS COM RESISTOR DE $2\text{k}\Omega$	67
APÊNDICE E – DADOS DOS EXPERIMENTOS COM RESISTOR DE $12\text{k}\Omega$	68
APÊNDICE F – BATERIA DE EXPERIMENTOS COM RESISTOR DE $2\text{k}\Omega$	69
APÊNDICE G – BATERIA DE EXPERIMENTOS COM RESISTOR DE $12\text{k}\Omega$	71

1 Introdução

Segundo Ramalho, Ferraro e Soares (2007, p. 3):

[...] a Física pode ser definida como uma ciência que busca descrever os fenômenos que ocorrem na Natureza e prever a sua ocorrência, procurando atualmente não mais oferecer uma imagem da Natureza, mas sim uma imagem da relação do ser humano com a Natureza.

O ensino da Física se divide entre aulas teóricas, voltadas à compreensão dos conceitos, leis e teorias, e aulas práticas voltadas à observação experimental dos fenômenos e coleta de dados. Os fenômenos descritos na forma de leis matemáticas são o objeto central do estudo teórico, que representa uma fonte importante para a compreensão dos fenômenos naturais. Por outra via, os experimentos realizados de forma didática reproduzem os fenômenos sob os critérios estabelecidos nas leis, e permitem avaliar a teoria estudada. A ciência experimental realizada em laboratórios didáticos difere daquela feita nos de pesquisa. Enquanto os experimentos realizados por cientistas profissionais tem por objetivo produzir ou validar conhecimento, aqueles desenvolvidos por alunos e professores no ambiente didático visam reforçar a compreensão das teorias e leis já validadas. Portanto, no contexto educacional, a análise feita sobre a empiria por meio do confronto com a teoria, destina-se tanto à identificação das convergências e divergências entre ambas, como para proporcionar ao aluno a oportunidade de pensar nas possíveis razões para ocorrência de inexatidões e imprecisões. Quando os laboratórios didáticos apresentam grande ênfase nos resultados, acabam por induzir os estudantes a realizarem os experimentos ignorando a relevância do processo de investigação científica, o que concorre para o não desenvolvimento do pensamento crítico, assim como das habilidades analíticas e investigativas, dado que, os estudantes assumem uma postura passiva onde reproduzem os experimentos de maneira sistematizada e inflexível, com vistas aos resultados almejados. Segundo Thornton (1987), além dessas consequências negativas oriundas da ênfase nos resultados, esse tipo de prática experimental não é capaz de mudar os equívocos e o entendimento simplista dos estudantes acerca da física, e a falta de técnicas sofisticadas de laboratório somada ao esforço empreendido para que os alunos obtenham respostas para questões já conhecidas, estimulam a prática experimental “cookbook”¹ nos laboratórios didáticos de física. Dessa forma, todo aprendizado científico vivenciado pelo estudante, sob essas condições, limita-se a permanecer no ambiente didático e não surte efeito sobre sua forma de pensar acerca do mundo físico que o cerca. A fim de evitar isso, é fundamental que os alunos e professores disponham de um aparato experimental sobre o qual possuam amplo domínio e que lhes

¹ Realizar, desprovido de qualquer pensamento crítico acerca do processo, o experimento seguindo instruções prontas a fim de obter resultados já esperados.

permita manipular todas as variáveis de seu experimento, condicionando-os a participar ativamente do processo de investigação científica e afastando-os da simples reprodução do experimento pela busca dos resultados, ao mesmo tempo que os aproxima da prática profissional, ou seja, do real processo de se fazer ciência.

Face ao exposto, o presente trabalho tem por objetivo desenvolver e validar um ambiente aberto (software e hardware) para experimentos didáticos em física usando o dispositivo microcontrolado Arduino e a plataforma computacional Raspberry Pi. Esse ambiente deve ser capaz de minimizar o esforço com o monitoramento dos experimentos e registro dos dados, automatizando-os. Com isso, alunos e professores terão mais tempo para se debruçarem sobre as questões científicas que são o objetivo de sua investigação. Pretende-se, também, favorecer uma abordagem inquisitiva, oferecendo aos realizadores do experimento pleno controle sobre as variáveis de seu aparato experimental, nos aspectos de hardware e software. Com a realização deste trabalho, almeja-se contribuir para enriquecer o processo de experimentação em laboratório didático de física (possibilitar que professores e alunos interfiram de diversas formas no arranjo experimental), aproximando-o da prática profissional, assim como, para ampliar a versatilidade do hardware e software do aparato experimental em comparação com um similar comprado pronto.

Esta monografia está organizada em nove capítulos. O capítulo 2 discorre acerca da conversão de energia luminosa em energia elétrica, das possíveis formas de conversão, assim como dos princípios e tecnologias que dão suporte a essa conversão. O 3º capítulo apresenta a plataforma computacional Raspberry Pi e sua aplicação em ambientes didáticos. No capítulo 4, os circuitos eletrônicos são introduzidos, com ênfase no circuito RC e no processo de conversão analógico-digital, importantes para a compreensão do aparato experimental proposto. O capítulo 5 aborda os dispositivos microcontrolados, apresentando a placa arduino. No capítulo 6, o ambiente aberto proposto neste trabalho para o desenvolvimento do experimento de conversão fotovoltaica é apresentado em detalhes. Encontram-se descritos, no capítulo 7, o processo de experimentação e a validação do ambiente proposto neste trabalho. No 8º capítulo é apresentada a análise dos resultados alcançados e, por fim, o capítulo 9 trata das conclusões e trabalhos futuros.

2 Conversão de Energia Luminosa em Energia Elétrica

2.1 Formas de conversão

Não há apenas um método para converter energia solar em elétrica, e embora não existam em abundância, as técnicas conhecidas permitem realizar essa conversão de forma direta, ou indireta, com um grau relativamente aceitável de eficiência energética, o que as tornam potenciais soluções para geração de energia de baixo impacto ambiental. Além disso, existem pesquisas em andamento cujos objetivos são tanto a descoberta de novas formas de converter a energia solar, como aprimorar o processo e sua eficiência – a exemplo da fotossíntese artificial que é estudada pelo grupo de pesquisadores do Instituto de Química (IQ) da Universidade Estadual de Campinas (Unicamp) ([ALISSON, 2014](#)). Some-se a isso, o esforço por parte dos órgãos e institutos especializados em energia solar, tal como o Centro de Referência para Energia Solar e Eólica Sérgio Brito – Cresesb, para aprimorar os métodos que são de conhecimento da comunidade científica e dos entes públicos e privados produtores de energia elétrica, tais como a conversão fotovoltaica e a conversão heliotérmica.

2.1.1 Conversão Indireta

Quando o sistema conversor necessita realizar uma etapa intermediária de conversão – transformar a energia em um tipo distinto do esperado ao final do processo – antes que, de fato, converta a energia em sua entrada no tipo pretendido na saída, ele realiza conversão indireta. No caso da conversão de energia solar em elétrica, a usina heliotérmica é um exemplo de conversão indireta, uma vez que, o seu processo transforma a energia proveniente do sol em térmica antes que possa gerar energia elétrica. Essas usinas usam a irradiação solar como fonte de energia para seu funcionamento, o que as difere das termoelétricas, e são equipadas com um sistema mecânico responsável pela geração de eletricidade. A captura desse combustível tem início com a criação de um campo solar capaz de concentrar energia térmica, para isso é construído um sistema de espelhos (denominado de helióstato ou coletor, dependendo do mecanismo de concentração solar) que acompanham a posição do sol ao longo do dia e refletem os raios solares para um foco onde está o receptor que os transforma em calor. Esse é o primeiro processo de conversão que a energia solar sofre nesse sistema e, uma vez que, o calor seja transmitido para o fluido térmico, a energia térmica é transformada em elétrica pelo sistema mecânico ([IBICT, s.d.](#)). A Tabela 1 mostra um comparativo com valores aproximados para conversão de energia

solar em térmica nos diferentes tipos de tecnologias adotadas nas usinas heliotérmicas.

Tabela 1 – Comparativo dos aspectos das tecnologias utilizadas pelas usinas heliotérmicas

Nome	Foco	Arranjo dos Espelhos	Grau de Concentração	Temperatura de Operação no Foco
Calha Cilindro-Parabólico	Linha	Fileiras de espelhos linear-parabólicos	≈ 100 vezes	$\approx 450^\circ\text{C}$
Coletor Linear Fresnel	Linha	Fileiras de espelhos linear-planos	≈ 100 vezes	$\approx 450^\circ\text{C}$
Torre Solar	Ponto	Unidades de discos de espelhos parabólicos	≥ 1000 vezes	700 °C a 1.000 °C
Disco Parabólicos	Ponto	Campo de helióstatos de espelhos planos	≥ 1000 vezes	700 °C a 1.000 °C

Fonte: Adaptada de ([YOUNG; FREEDMAN, 2009a](#))

Segundo [IBICT \(s.d.\)](#), nesse sistema, o fluído térmico aquece a água – armazenada em um reservatório conectado a uma turbina – até ela transformar-se em vapor, o qual move a turbina que, por sua vez, está ligada a um gerador elétrico capaz de converter a energia mecânica proveniente das rotações da mesma, em elétrica. O vapor excedente nesse processo é direcionado para um sistema de refrigeração responsável por condensá-lo na forma líquida e redirecioná-lo para o estoque de água. Essa é a descrição de um ciclo heliotérmico simplificado, há, ainda, a possibilidade de adicionar um mecanismo de armazenamento do calor para manter a capacidade de geração de energia pela usina durante a noite, assim como, em dias nublados.

2.1.2 Conversão Direta

A conversão fotovoltaica é o processo capaz de transformar a energia solar diretamente em energia elétrica, isso através da célula fotovoltaica – sem partes móveis – que está no centro de todo o processo. O material dessa célula passa por um tratamento especial durante o processo de fabricação, o que torna sua superfície frontal mais receptível aos raios solares que a atinge, por consequência, esse tratamento facilita a liberação dos elétrons que migram naturalmente para superfície. Quando a luz incide sobre a célula, podem ocorrer três coisas com essa luz: ser refletida, passar diretamente pela célula, ou ser absorvida. Dessas três possibilidades, apenas a última é capaz de prover energia para o processo de conversão energética, pois os fôtons são absorvidos pelos elétrons que conseguem se desvincular dos átomos do material. As células fotovoltaicas podem ser utilizadas em diversos dispositivos eletrônicos que demandem um baixo consumo de energia, como calculadoras de bolso, relógios de pulso, brinquedos entre outros. Nesses casos uma única célula, em tamanho reduzido, consegue oferecer a potência necessária para o funcionamento desses eletrônicos, contudo, quando pretende-se alimentar circuitos

maiores (por exemplo, uma casa ou um veículo movido a luz solar) é indispensável a adoção de módulos. O conjunto de células fotovoltaicas conectadas eletricamente é denominado de módulo, e esse arranjo tem por finalidade aumentar a potência gerada pela conversão. Para os casos onde apenas um módulo não seja o suficiente para atender às necessidades do sistema, é possível conectar módulos entre si para atingir a potência desejada. O tipo de corrente gerada pelas células fotovoltaicas é direta (DC), que é utilizada para alimentar, ou carregar, pequenos equipamentos eletrônicos. Haja vista a maioria dos dispositivos elétricos, eletrônicos e eletroeletrônicos funcionarem com corrente alternada (AC) – isso se aplica aos utilizados na indústria tanto quanto nas residências –, a adoção de sistemas fotovoltaicos para fins comerciais ou em conjunto com a rede elétrica local, deve ser feita com o uso de inversores, ou conversores de corrente DC para AC ([CRESCEB, 2007](#), p. 21). Embora seja necessária a conversão da corrente em casos específicos, a instalação dos módulos é rápida, relativamente simples (uma vez que, não é necessário nenhum sistema de geração mecânico que ocuparia mais espaço) e o tamanho de seu arranjo pode ser adequado para áreas grandes, ou restritas; ademais, esse sistema tem baixo impacto ambiental e dispensa o uso de outros recursos naturais que não seja a luz solar, tal como a água. Essa forma de conversão é objeto de estudo do presente trabalho, onde será analisada a eficiência de seu processo para diferentes tipos de fonte luminosa, assim como, tipos de células fotovoltaicas.

2.2 Efeito fotoelétrico

Em 1887, o físico alemão Heinrich Hertz observou – enquanto realizava experimentos com ondas eletromagnéticas – o efeito fotoelétrico. Esse fenômeno apresentava as seguintes características: quando a luz incidia sobre uma superfície metálica, os elétrons livres próximos a essa superfície eram capazes de absorver a energia luminosa e conseguiam superar a atração provocada pelos íons cátions, o que lhes permitia deslocarem-se para áreas vagas próximas. Os físicos alemães Wilhelm Hallwachs e Philipp Lenard investigaram esse fenômeno e contribuíram de forma relevante para a melhor compreensão do mesmo. Dentre essas contribuições podemos citar a identificação da frequência de corte, a qual determina a frequência mínima que a luz deve ter para que ocorra a emissão de elétrons, em outras palavras, nenhum elétron é emitido quando a luz que incide sobre a superfície do material possui frequência inferior à de corte. Em contrapartida, frequências superiores à de corte provocam a emissão dos elétrons com uma velocidade inicial elevada. Existe, ainda, a função trabalho – designada por θ – que é outro conceito associado ao efeito fotoelétrico e determina a quantidade de energia mínima que um elétron deve absorver para que consiga superar a atração que o mantém na superfície ([YOUNG; FREEDMAN, 2009b](#), p. 181). Esse conceito, assim como a frequência de corte, sofre variação de seu valor em função da mudança do tipo de material sobre o qual a luz é incidida.

Segundo YOUNG e FREEDMAN (2009b), a região ultravioleta, cujos comprimentos de onda estão entre 200 nm e 300 nm, é a frequência de corte para a maioria dos metais. Contudo, isso muda quando se trata dos óxidos de potássio e de célio, para os quais, essa frequência passa a constar na região visível do espectro - λ entre 400 nm e 700 nm. O espectro de comprimento e frequência das ondas eletromagnéticas é extremamente amplo, de forma que, é necessário o uso da escala logarítmica para representar todas as bandas. Dessa amplitude, o olho humano consegue enxergar um pequeno seguimento denominado de luz visível, onde as variações nos comprimentos de onda da luz são percebidos como cores distintas. A Tabela 2 apresenta (de forma muito aproximada) os valores de cada comprimento de onda e, a cor percebida pela visão humana para o mesmo.

Tabela 2 – Comprimentos de onda do espectro visível e suas respectivas cores

Comprimento	Cor
De 400 nm a 440 nm	Violeta
De 440 nm a 480 nm	Azul
De 480 nm a 560 nm	Verde
De 560 nm a 590 nm	Amarelo
De 590 nm a 630 nm	Laranja
De 630 nm a 700 nm	Vermelho

Fonte: Adaptada de (YOUNG; FREEDMAN, 2009a)

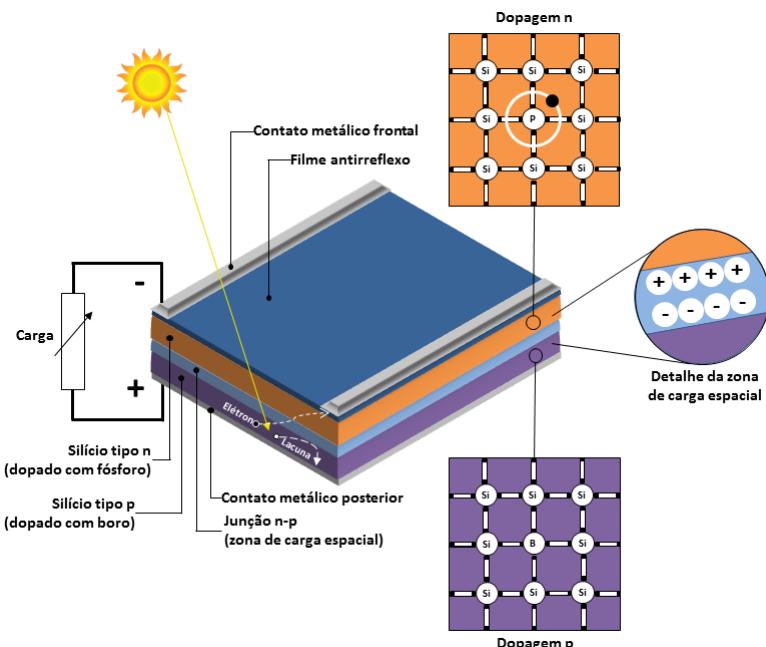
Os resultados dos experimentos realizados pelos cientistas Hertz, Hallwachs e Lenard apresentaram evidências que a física clássica não conseguiu explicar o porquê de suas ocorrências, tal como a frequência de corte. YOUNG e FREEDMAN (2009b) explicam que, a frequência da onda eletromagnética não influencia na sua intensidade, logo, uma luz de qualquer frequência seria capaz de oferecer a um elétron energia suficiente para que o mesmo pudesse escapar da superfície, o que tornaria impossível a existência da frequência de corte. Devido às incoerências entre a teoria e o que se evidenciava com a prática, surgiram lacunas nas explicações acerca do efeito fotoelétrico, as quais, foram preenchidas em 1905 quando Albert Einstein analisou corretamente esse fenômeno. Einstein desenvolveu suas ideias a partir da hipótese levantada cinco anos antes por Planck sobre a emissão intermitente da luz, e postulou a quantização da energia, onde afirmou que um feixe de luz era composto por pequenos pacotes de energia denominados de fôtons ou quantas, o que explicou a ocorrência da emissão imediata dos elétrons do metal, uma vez que, a energia da radiação estaria concentrada nos fôtons e não distribuída sobre a onda. Além disso, ele demonstrou que a velocidade dos elétrons ao deixarem a superfície não está relacionada com a quantidade de fôtons que a atingem, mas sim com a frequência da onda eletromagnética que transporta esses fôtons. Para tanto, Einstein utilizou a lei da conservação de energia e descobriu que a diferença entre a função trabalho e a energia absorvida pelo elétron determina a energia cinética máxima do mesmo. Segundo

Einstein, o número de fótons contidos em um feixe luminoso mantém uma relação de proporcionalidade com a intensidade da luz (energia por unidade de tempo e unidade de área) e, portanto, determina o número de elétrons emitidos do metal. A relevância desse trabalho desenvolvido por Albert Einstein foi reconhecida oficialmente no ano de 1921, quando ele foi laureado com o Prêmio Nobel de Física por suas contribuições à física teórica.

2.3 Células fotovoltaicas

A célula fotovoltaica é um dispositivo que converte diretamente energia luminosa em energia elétrica. Ela é composta de um material semicondutor caracterizado por possuir uma banda de valência preenchida por elétrons e outra banda de condução “vazia” de elétrons¹. A estrutura básica de uma célula fotovoltaica é composta por: par de contatos metálicos (frontal e posterior), filme antirreflexo em sua superfície, duas camadas sobrepostas e constituídas de material semicondutor (uma com dopagem n, e a outra com dopagem p), e entre essas, forma-se a zona de carga espacial e campo elétrico – junção pn. Devido à existência dessa junção, os portadores de cargas são separados gerando o denominado efeito fotoelétrico. Na figura 1 é apresentada a visão parcial de uma célula fotovoltaica com destaque de sua estrutura básica.

Figura 1 – Estrutura da célula fotovoltaica



Fonte: Adaptada de (PINHO; GALDINO, 2014).

Segundo PINHO e GALDINO (2014, p. 113), a energia de um fóton está associada

¹ Essa condição ocorre a 0 K

a cor da luz e quando a célula fotovoltaica é atingida por fótons com energia superior ao gap² ocorre a geração dos pares eletrón-lacuna nos semicondutores de sua estrutura. Esse fenômeno pode ocorrer tanto dentro quanto fora da zona espacial, o que influência no comportamento dos portadores de carga gerados no interior da fotocélula. Os pares gerados dentro da zona espacial dão origem a corrente elétrica no sentido da região *n* para região *p*, isso ocorre devido à aceleração que os portadores sofrem no campo eléctrico, onde os elétrons são acelerados para o lado *n* e as lacunas para o lado *p*. Quando os portadores são gerados fora da zona espacial, ainda podem vir a contribuir para a corrente elétrica gerada, mas para isso, os elétrons presentes na região *p* e as lacunas que estão na região *n* – denominados de portadores minoritários – devem possuir um comprimento de difusão que os permita alcançarem a zona espacial para serem coletados. Caso contrário, sofreram recombinação do tipo direta ou indireta. Quando o portador de carga negativa vai diretamente da banda de valência para a de condução, podendo emitir energia sob forma de fóton em intensidade na frequência de infravermelho, ocorre a recombinação direta. De modo distinto, quando o elétron passa por níveis intermediários de energia no *gap* devido às possíveis falhas na estrutura cristalina da camada, denomina-se recombinação indireta.

Existem diversos tipos de células fotovoltaicas que apresentam desempenhos distintos de acordo com o processo de fabricação e o material semicondutor do qual são constituídas. Segundo PINHO e GALDINO (2014), são fabricadas e comercializadas células construídas com filmes finos de telureto de cádmio, silício amorfo hidrogenado, silício microcristalino, silício monocristalino e policristalino entre alguns outros. Dessas, as de silício cristalino (monocristalino e policristalino) haviam atingido cerca de 87.7% da produção mundial de células em 2011 e são as mais utilizadas na fabricação dos módulos fotovoltaicos. Além do tipo monocristalina e policristalina, o tipo amorfo faz parte dos modelos principais de células fotovoltaicas. Dentre esses tipos, o amorfo possui o processo de fabricação mais simples e menos custoso, contudo, seu desempenho é inferior quando comparado com os demais, devido ao nível de desordem na estrutura dos átomos na célula e seu rendimento elétrico varia de 8% a 10% (podendo chegar a 13% em laboratório). O tipo policristalino apresenta rendimento elétrico de 18% em laboratório e, fora dele, de 11% até 13%, além disso, seu processo de fabrico demanda menor quantidade de energia que o processo da monocromática, porém, maior do que o da amorfo. Por fim, o monocristalino apresenta o maior custo e complexidade de fabrico dentre os principais tipos, contudo, demonstra o maior desempenho devido ao grau de pureza que esse processo de produção alcança, cerca de 99.99%. Em laboratório, esse tipo de célula alcança até 23%, e no ambiente externo seu rendimento aproxima-se dos 16% (ENERGIA, 2011).

² Zona proibida entre as bandas de energia dos semicondutores que possui valor máximo de 3 eV (três elétron-volts)

3 Raspberry Pi em Ambientes Didáticos

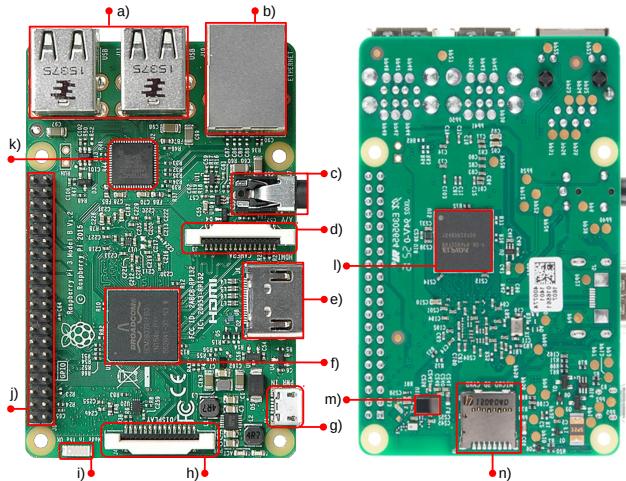
Raspberry Pi é um computador surgido no Reino Unido e criado pela *Raspberry Pi Fundation*, responsável pela manutenção e difusão do projeto Raspberry pi. A proposta da fundação é prover computadores de baixo custo e alta performance para uso no ensino, na solução de problemas e no entretenimento. Desde 2012 , ano em que iniciou a comercialização do Pi para o público em geral, esse projeto expandiu o número de versões distintas da placa e, em 2017, lançou o sexto modelo comercial do projeto com preço de venda fixado em \$10 , o Raspberry Pi Zero W. Dentre esses modelos, o escolhido para compor o ambiente aberto proposto neste trabalho foi o Raspberry Pi 3 Model B, que destaca-se da geração anterior por possuir processador de 1.2 GHz 64-bit quad-core ARMv8 , wireless 802.11 n, bluetooth 4.1 e BLE (*Bluetooth Low Energy*), ademais, apresentou melhor desempenho em relação ao inicialmente testado, Raspberry Pi model B. o Pi é composto por outros hardwares além dos citados, e todos estão integrados numa única placa cujas dimensões assemelham-se às de um cartão de crédito¹. Trata-se da porta Full HDMI, das quatro portas USB, da porta Ethernet, da interface para câmera, da interface para display, do slot para cartão micro SD, do processador gráfico e do grupo de quarenta pinos digitais de entrada e saída. A integração desses hardwares pode ser vista na Figura 2, que exibe a parte superior e inferior do Raspberry Pi 3 model B com destaque para o local ocupado na placa por esses componentes.

Esse arranjo de hardware, a dimensão física e o custo de aquisição do Pi concorrem para torná-lo uma ferramenta poderosa, versátil e atrativa. Contudo, não são os responsáveis isolados pelo número expressivo de projetos educacionais desenvolvidos com base nessa plataforma computacional. Soma-se a isso, a diversidade de materiais disponíveis e aplicações desenvolvidas tanto pela comunidade mundial, que presta suporte ao projeto Raspberry Pi, quanto os esforços empreendidos pela *Raspberry Pi Fundation*, que mantém em seu site oficial uma área dedicada ao uso do Pi no contexto educacional, onde vasto material (esquemas, treinamentos, projetos completos etc.) direcionado a professores, estudantes e entusiastas da tecnologia encontra-se disponível para acesso. Além disso, a fundação é parceira na produção da revista para educadores *Hello World* que realiza três publicações anuais e as disponibiliza gratuitamente para download no formato PDF sob licença *Creative Commons*². A quantidade expressiva de pesquisas publicadas, que utilizam o Raspberry Pi como ferramenta a serviço do aprendizado, revela que a comunidade acadêmica ao redor do mundo está atenta para possibilidade de seu emprego

¹ Segundo a ISO 7810 a dimensão do cartão de crédito é 8.56 cm × 5.39 cm, enquanto a do Pi é 8.50 cm × 5.60 cm.

² Licença que permite a cópia e compartilhamento com menores restrições em relação as licenças que reservam todos os direitos aos autores.

Figura 2 – Raspberry Pi 3 Model B



a) 4 Portas USB. b) Entrada Ethernet. c) Conector de Áudio 3.5mm combinado com Vídeo Composto. d) Interface Câmera. e) Saída Full HDMI. f) Processador 1.2GHz 64-bit quad-core ARMv8. g) Conector Fonte de Energia 5V. h) Interface Display. i) Antena SMD. j) 40 Pinos GPIO. k) Processador Gráfico VideoCore IV 3D. l) Chip 1 GB SDRAM. m) Chip BCM43438 (Wireless802.11n, Bluetooth 4.1 e BLE). n) Slot Cartão Micro SD. Fonte: Adaptada de ([WIKIPÉDIA, 2017](#)).

em ambientes didáticos. A exemplo, [Ali et al. \(2013\)](#) afirma que a criação do Pi gerou novas oportunidades para aprimoramento das ferramentas educacionais, e apresenta o design, desenvolvimento e montagem de um protótipo utilizando o Raspberry Pi como ferramenta educacional no ensino de ciências nas escolas de Uganda, assim como, a análise das implicações sociais da implementação dessa solução.

[Reck e Sreenivas \(2015\)](#) descrevem o desenvolvimento do kit modular e portável para laboratório, capaz de substituir equipamentos caros com uma alternativa de baixo custo e atrativa, principalmente, para universidades cujo orçamento seja um limitador perante suas demandas de equipamentos para laboratório. Esse kit foi montado para o curso introdutório de controle no programa de Engenharia Geral da *University of Illinois* localizada na *Urbana-Champaign* e consiste de um Raspberry Pi, um motor DC e vários circuitos necessários ao *drive* do motor para medir a velocidade, posição e executar a identificação do sistema, além de outro computador para processar e exibir os resultados. O projeto desse kit estava submetido a algumas restrições de projeto, tais como orçamento total em torno de \$100 e hardware de baixo custo, tamanho portável e integrável com o Simulink³ para monitoramento. Segundo [Reck e Sreenivas \(2015\)](#), essas restrições somadas aos benefícios do Pi (flexibilidade de uso, potencial para processar algoritmos mais complexos e possibilidade de monitoramento por vídeo), foram os motivadores para sua escolha como componente responsável por monitorar e coletar os dados, em vez da

³ Ferramenta desenvolvida pela MathWorks para modelagem, simulação e análise de sistemas dinâmicos.

placa ARDUINO. Contudo, cabe ressaltar que a estrutura montada para monitorar a velocidade de rotação (foto interruptor e roda impressa em 3D conectada ao eixo do motor) utiliza sinal digital, o que minimiza o impacto dessa escolha sobre a complexidade do circuito e possibilita o uso das GPIOs do Raspberry Pi para essa tarefa. Do contrário, ou seja, medir a grandeza como sinal analógico, traria complexidade adicional para o circuito e a necessidade de adicionar componentes externos porque o Pi não possui estrutura física embutida para trabalhar com esse tipo de sinal. Embora a adoção do Raspberry Pi seja um ponto em comum entre (RECK; SREENIVAS, 2015) e este trabalho, as metodologias diferem tanto na finalidade de seu emprego quanto no tipo de sistemas utilizados. Pois neste projeto o Pi é utilizado para gerir o experimento, processar os dados e exibir os resultados por meio de software livre – desenvolvido como parte integrante do projeto –, dispensando, portanto, a necessidade de outro computador para executar essas tarefas.

4 Circuitos Eletrônicos

Eletrônica, em sua plenitude, pode ser enquadrada em duas grandes áreas: eletrônica analógica e eletrônica digital, as quais diferenciam-se pelas suas aplicações e pela forma como representam as grandezas.

4.1 Componentes eletrônicos

Os componentes eletrônicos, dispositivos transmissores de corrente elétrica, são utilizados nos circuitos eletrônicos onde encontram-se obrigatoriamente interligados entre si. Esses dispositivos podem ser classificados em passivos ou ativos, do mesmo modo que, pela forma como conduzem a eletricidade – por meio condutor, semicondutor ou vácuo. Classifica-se como passivo o componente incapaz de prover eletricidade para o circuito, fornecer ganho para corrente ou redirecioná-la, por outra via, denomina-se ativos os componentes cujo comportamento é oposto ao apresentado pelos passivos. Embora os componentes eletrônicos possam ser distinguidos pelas características citadas, a atuação desses dispositivos assemelha-se por modificar o comportamento da corrente nos circuitos, porém, o efeito dessa mudança difere de acordo com o componente (e.g. o cristal de quartzo faz a corrente oscilar, ao passo que, o resistor cria uma resistência elétrica à corrente no circuito e provoca queda da tensão).

Nesta pesquisa foram adotados componentes eletrônicos na construção do circuito RC integrante do ambiente aberto proposto, a saber: resistor, capacitor, transistor e célula fotovoltaica (descrita em capítulo anterior). O resistor impõem uma resistência, medida em ohm, ao meio condutor e regula a passagem da corrente elétrica no circuito, como consequência, parte da energia elétrica que o atravessa é convertida em energia térmica. Esse componente pode ter resistência ajustável (potenciômetro), ou que sofre mudança por influência do ambiente onde está inserido (termistores e *Light Dependent Resistor-LDR*), ou ter resistência fixa (resistores de filme metálico).

Outro componente presente no circuito RC deste projeto é o semicondutor denominado de transistor. Esse componente é capaz de realizar um conjunto variado de atividades nos circuitos eletrônicos (e.g. amplificação e interrupção de sinais elétricos) e sua estrutura é composta por três regiões semicondutoras – cujos materiais são o metal germânio, ou o semimetal silício – que formam as junções internas nesse componente. A maioria dos transistores apresentam três terminais externos denominados Coletor (*Collector*), Base (*Base*) e Emissor (*Emitter*) nos transistores bipolares, assim como, Fonte (*Source*), Portão (*Gate*) e Dreno (*Drain*) nos tipos de efeito de campo. Embora existam diversos modelos de transistores, todos baseiam-se basicamente em dois tipos, o transistor de junção bipolar

(BJT - *Bipolar Junction Transistor*) e o transistor de efeito de campo (FET - *Field Effect Transistor*), que se distinguem por diversas características. Enquanto a produção de corrente nos transistores de efeito de campo ocorre utilizando apenas um tipo de Portador de Cargas, nos transistores bipolares, a corrente é produzida utilizando tanto electrões quanto lacunas (motivo pelo qual foram denominados de bipolares).

Modelos de transistores bipolares dividem-se em NPN e PNP. Essas duas categorias indicam como o arranjo das camadas desse componente formam as junções, onde NPN significa uma camada de polaridade positiva (P) entre duas camadas de polaridade negativa (N), enquanto PNP indica uma região N entre duas regiões P. Essa disposição das camadas possibilita que o transistor opere em situação de corte, de saturação e de amplificação linear conforme a polarização aplicada a cada uma dessas regiões. Quando um transistor funciona alternando entre estado de corte e estado de saturação, esse componente assume o comportamento característico de uma chave eletrônica no circuito. Essa alternância é possível porque o transistor controla a corrente que o atravessa (no sentido convencional dos NPN, de coletor para emissor) através da quantidade de corrente na base. Quando saturado, o transistor funciona como chave fechada e permite que a corrente passe por completo no sentido convencional, de modo contrário, quando em estado de corte, o transistor comporta-se como chave aberta e não há fluxo de corrente entre coletor e emissor¹. Tendo em vista que o transistor utilizado na construção do circuito RC deste projeto desempenha a função de chave eletrônica – controlada pela porta digital do Arduino UNO – e, ademais, adotar componentes de baixo custo foi uma diretiva deste projeto, optou-se, portanto, pelo uso do tipo BJT por adequar-se aos limites de fornecimento de tensão e correntes do Arduino e por seu custo de aquisição ser inferior ao do tipo FET.

O último componente do circuito supracitado é o capacitor. Esse componente eletrônico possui estrutura interna composta por duas placas condutoras (armaduras), dispostas em paralelo, e separadas por um material isolante denominado dielétrico. Essa arquitetura acumula as cargas que chegam até esse componente quando há tensão elétrica entre seus terminais, de modo que, enquanto a tensão do capacitor não se igualar a diferença de potencial da fonte a qual está conectado, a armadura conectada ao polo positivo da fonte acumula cargas $+Q$ e a armadura ligada ao polo negativo, cargas $-Q$ – contudo, o valor absoluto das cargas é o mesmo (BRAGA, s.d.). A essa capacidade de armazenar energia elétrica atribui-se o nome de capacitância (C) e sua unidade de medida no Sistema Internacional de Unidades (SI) é o F. Na prática, a maioria dos capacitores fabricados possuem capacitância com valores inferiores a 1 F (devido ao valor de capacitância demandado pela maioria dos circuitos presentes nos equipamentos eletrônicos ser muito inferior a esse montante) e adotam os submúltiplos do F como unidades de medidas, dos quais, μF , nF e pF são os mais utilizados. Além da variação de capacitância,

¹ "O Transistor TBJ como Chave". Disponível em: https://www.youtube.com/watch?v=nRAk_fCV6Ew

causada pelo aumento ou diminuição da distância entre as armaduras e pela mudança do material isolante que as separa, os capacitores podem ter formatos distintos e apresentar propriedades que, conforme natureza do dielétrico entre as armaduras, os tornam propícios para determinados propósitos. A exemplo disso, os capacitores eletrolíticos apresentam melhor funcionamento quando empregados em circuitos onde a corrente é contínua e a frequência é baixa, pois esses componentes são fabricados utilizando folhas de alumínio como armaduras e uma camada superfina de óxido (formada por meio de um processo eletrolítico) que atua como dielétrico entre as folhas. Diante disso, e pelo fato do circuito RC presente neste projeto ter sido projetado para ser alimentado com corrente contínua, optou-se pelo uso do capacitor eletrolítico nesse circuito.

4.2 Circuitos digitais

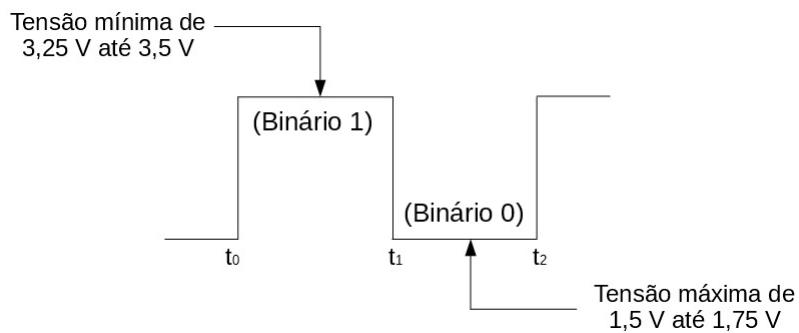
Os circuitos digitais utilizam sinais cujos valores assumidos são sempre discretos, por outra via, na analógica os sinais são sempre contínuos e podem assumir quaisquer valores entre dois limites. Embora a maioria das grandezas que se pode mensurar na natureza sejam contínuas (e.g. luminosidade, intensidade do som, velocidade de um objeto), os sistemas digitais podem apresentar vantagens em relação aos analógicos na manipulação desses valores. Isso não significa que a eletrônica digital seja melhor que a analógica, ou até mesmo, que irá substituí-la em todas as suas possíveis aplicações, por exemplo, na amplificação do sinal de áudio. Essas vantagens revelam-se quando há a necessidade de armazenar, processar e transmitir os dados medidos – principalmente se o volume de dados for relativamente alto.

Os circuitos digitais fazem parte da eletrônica digital, a qual, adota o sistema de numeração binária, e para representar os dois dígitos desse sistema emprega-se tensões distintas, de modo que, a maior tensão representa o dígito 1, a menor, o dígito 0 e são denominadas de nível lógico alto e nível lógico baixo, respectivamente. Na prática, os circuitos digitais utilizam uma faixa de tensões para representar cada nível lógico, em vez de apenas um único valor de tensão. Essa faixa é relativa ao tipo de tecnologia e a tensão de alimentação do circuito, por exemplo, para a tecnologia CMOS(*complementary metal-oxide-semiconductor*)², o nível lógico será alto quando a tensão mínima da entrada variar de 65 % até 70 % da tensão de alimentação, e o mesmo será baixo - para essa mesma fonte de alimentação – quando a variação da tensão máxima da entrada situar-se entre 30 % e 35 % ([SACCO, 2014](#)). Para essa tecnologia, um circuito B alimentado com 5 V apresenta nível lógico baixo para tensão máxima entre 1.5 V e 1.75 V, assim como, nível alto para tensão mínima entre 3.25 V e 3.5 V. As tensões situadas na faixa entre o valor máximo para nível lógico baixo (1.75 V) e o valor mínimo para nível lógico alto (3.25 V)

² Tecnologia utilizada no processo de fabricação de circuitos integrados.

não são admitidas pelo circuito. A Figura 3 exibe o comportamento do pulso digital nesse circuito quando a tensão varia nos limites supracitados.

Figura 3 – Limites mínimos e máximos dentro da faixa de tensões para que a tensão no circuito B represente um dígito binário.

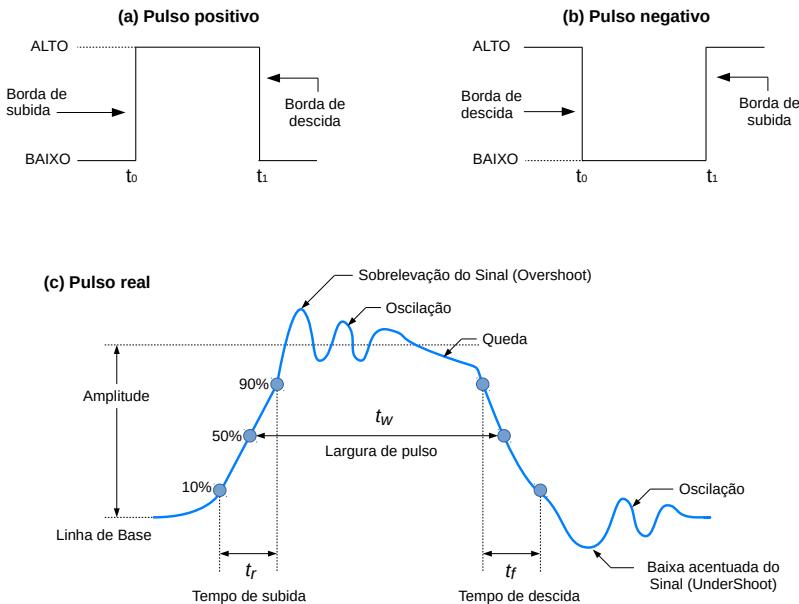


Fonte: Autor

A tensão no circuito digital pode alternar entre os níveis lógicos ao longo do tempo, de modo que, dessa comutação surge o pulso. Todo pulso apresenta duas bordas distintas: subida e descida, onde, a ordem de ocorrência das mesmas indica se o pulso é positivo ou negativo. Quando a primeira borda do pulso é de subida, a segunda será de descida e o pulso é positivo (a tensão comutou do nível lógico baixo para alto e retornou para baixo), caso contrário, a primeira borda é de descida e a segunda será de subida (a tensão mudou do nível lógico alto para baixo e retornou para alto), logo, o pulso é negativo. Todo pulso tem duração finita de tempo – largura do pulso (t_w – *pulse width*) – e a mudança de nível lógico que ocorre em cada banda dura um instante de tempo t , denominado de tempo de subida (t_r – *rise time*) quando a mudança ocorre do nível baixo para o alto e, tempo de descida (t_f – *fall time*) quando ocorre o inverso. Floyd (2009, p. 23) afirma que na eletrônica digital os pulsos são considerados ideais – a transição de uma borda para outra no pulso ocorre de forma instantânea, num tempo zero, e sem distorções. Contudo, na prática os mesmos podem portar-se de modo não esperado e apresentar alguns tipos de distorções (oscilações, sobrelevação e/ou baixa do sinal etc.) que podem ser causadas por efeito de indutância e capacidade parasitas³. Isso significa que uma onda de formato digital, ao contrário daquilo que normalmente se espera, pode ser formada por pulsos não ideais e concorrer para o mau funcionamento do circuito digital. Na Figura 4 é possível observar a diferença entre o pulso ideal e o não ideal, assim como, as características apresentadas por ambos.

³ Surge quando elementos cujo comportamento deveria ser não capacitivo introduzem capacidade não desejada no circuito.

Figura 4 – Formato apresentado pelo pulso ideal e o formato de um pulso real.



Fonte: Adaptada de (FLOYD, 2009).

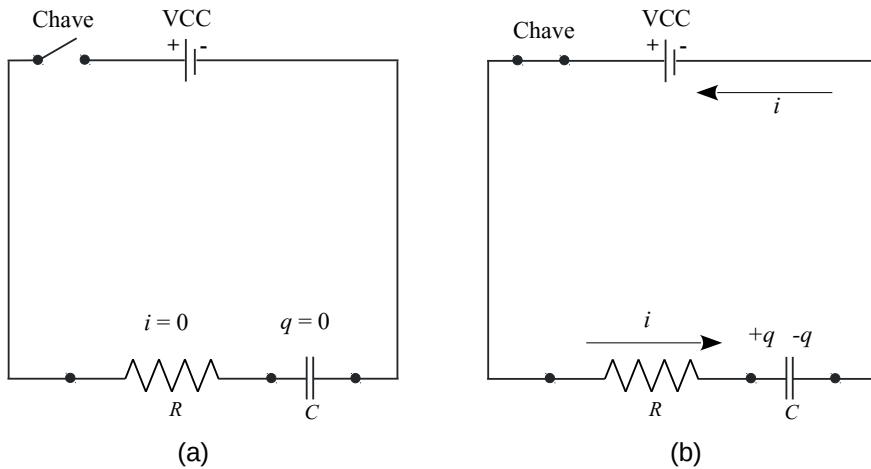
4.3 Circuito Analógico RC

O circuito RC (Resistor-Capacitor) é um dos arranjos fundamentais dos componentes eletrônicos, usado em osciladores, filtros de frequência, geradores de sinal e temporizadores. Para fins experimentais, ele permite estudar o comportamento da corrente e da tensão em relação ao tempo (GALERIU; LETSON; ESPER, 2015). No regime transitório, um circuito RC simples pode consistir em resistor, capacitor e fonte de tensão, onde o polo positivo do capacitor encontra-se conectado em série com o resistor, que por sua vez, é conectado ao polo positivo dessa fonte, por outra via, os polos negativos da fonte de tensão e do capacitor encontram-se conectados, o que cria uma diferença de potencial entre os terminais do capacitor. A fim de explicar o comportamento de carga do capacitor, consideremos um circuito RC simples que possui uma chave entre os polos positivos do capacitor e da fonte de tensão, conforme exibido na 5. Inicialmente a chave encontra-se aberta e o capacitor está descarregado com tensão nula entre seus terminais, ao se fechar a chave no instante de tempo $t = 0$, temos o seguinte estado no circuito: a tensão que transpassa o resistor é a mesma da fonte de tensão e a diferença de potencial do capacitor é nula (YOUNG; FREEDMAN, 2009b, p. 182).

A partir desse instante, o processo de carga inicia-se no capacitor e a diferença de potencial (ddp) entre seus terminais aumenta com o passar do tempo, pois à medida que o capacitor acumula cargas positivas em sua armadura de polaridade positiva, ele perde cargas negativas pelo seu terminal negativo⁴ capacitor aumenta é mais rápida no início desse

⁴ “Me Salva! RLC02 - Circuito RC - Condições iniciais e finais”. Disponível em: <https://www.youtube.com/watch?v=KJzXWzgkVwU>

Figura 5 – Circuito RC antes e durante o processo de carga.



(a) Devido à chave está aberta, não há corrente i atravessando o resistor R e a carga q no capacitor descarregado é nula. (b) Quando a chave encontra-se fechada, passa a existir uma corrente i de VCC em direção a C , atravessando o resistor R , assim como, outra corrente i de C para VCC. Fonte: Adaptada de (YOUNG; FREEDMAN, 2009b, p. 182).

processo e tornar-se mais lenta na proporção que a ddp no capacitor aproxima-se da ddp na fonte de tensão, até que, essas tensões sejam praticamente iguais e o fluxo de corrente no circuito cesse (na prática essa corrente tende a zero). Logo, independente da tensão nominal do capacitor, o valor máximo de tensão que esse componente eletrônico conseguirá atingir num circuito RC encontra-se limitado à tensão da fonte, ou seja, para fins práticos, considera-se que o capacitor está carregado quando ele não mais conseguir acumular cargas devido à sua ddp ter alcançado a ddp da fonte. Isso demonstra o comportamento da tensão nos circuitos RC em série durante a carga do capacitor. Quanto ao comportamento da corrente nesses circuitos, observa-se que no inicio do processo de carga, a corrente que atravessa o resistor é a máxima para o circuito, pois a voltagem que passa pelo resistor é igual à tensão da fonte. Ocorre que, à medida que a tensão no capacitor aumenta ao longo do tempo, a tensão através do resistor diminui, concorrendo para redução da corrente que o atravessa. Dessa forma, observa-se que o comportamento da corrente é inverso ao comportamento da tensão no circuito RC durante o processo de carga do capacitor, onde no início desse processo, a corrente atinge seu valor máximo e a tensão seu valor mínimo, e ao final do processo alcançam, respectivamente, seu valor mínimo e máximo. Outro aspecto do processo de carga do capacitor é a denominada constante de tempo. Segundo YOUNG e FREEDMAN (2009b, p. 184), essa constante – também denominada tempo de relaxação do circuito – indica a medida da velocidade durante o processo de carga do capacitor através do produto entre a resistência e a capacitância (RC), e sua representação é feita com o uso da letra τ . Cabe ressaltar que, ao adotar as medidas Ohm e Farad para

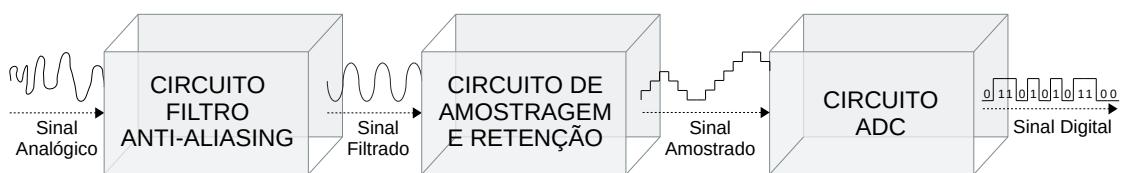
resistência e capacidade, respectivamente, a medida de tempo obtida estará em segundos. Podemos analisar τ da seguinte maneira: caso a resistência seja grande, o tempo para carregar o capacitor será maior, no caso contrário, o tempo será menor.

Para o experimento desenvolvido nesta pesquisa, o processo de carga do capacitor num circuito RC constitui parte relevante na observação experimental, uma vez que, possibilita observar, dentro de condições experimentais controladas, o comportamento da fotocélula ao gerar tensão para carregar o condensador.

4.4 Conversão analógico-digital

Quando é necessário processar sinais analógicos em sistemas digitais, deve-se primeiro convertê-los numa representação digital equivalente. Chama-se conversor analógico-digital (A/D) a qualquer um dos diferentes tipos circuito usado com essa finalidade. Em linhas gerais, um conversor A/D é composto pelos circuitos de filtro *anti-aliasing*, de amostragem e retenção – S/H (*Sample and Hold*) e de conversão, propriamente dita (*Analog-to-Digital Converter*). Segundo Floyd (2009), esses circuitos relacionam-se através de suas entradas e saídas, onde o filtro *anti-aliasing* recebe o sinal analógico em sua entrada e, por ser um filtro passa-baixas, gera na saída um sinal analógico livre das componentes de frequência que excedem à frequência de Nyquist⁵. Esse sinal serve de entrada para o circuito S/H que o transforma, por meio do processo de amostragem e retenção, numa forma de onda semelhante a degraus e dispõe esse sinal amostrado como entrada para o ADC, que converte cada valor amostrado em código binário ao executar o processo de quantização. O diagrama de bloco apresentado na Figura 6 demonstra o arranjo desses circuitos no sistema de conversão analógico-digital.

Figura 6 – Diagrama de bloco do sistema de conversão analógico-digital.



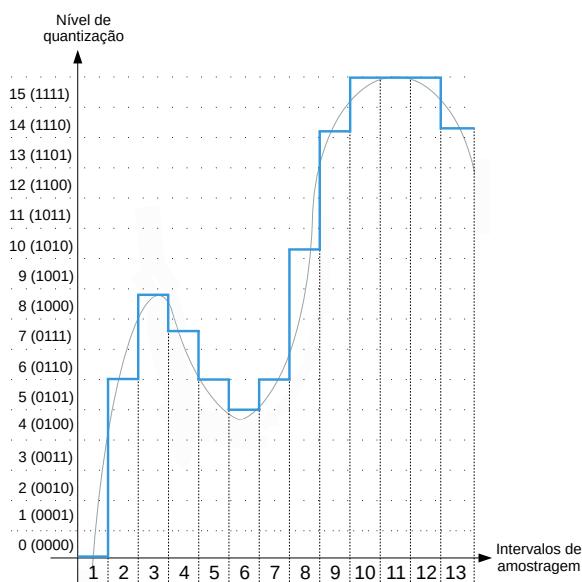
As formas de ondas no diagrama foram elaboradas arbitrariamente para fins de ilustração. Fonte: Adaptada de (FLOYD, 2009, p. 760).

A quantidade de bits utilizado pelo circuito ADC para realizar a quantização dos valores amostrados é o principal fator a determinar a precisão de um conversor A/D: quanto maior for o número de bits do conversor, maior será a precisão, pois a magnitude

⁵ O teorema da frequência de Nyquist determina que, para uma correta amostragem do sinal analógico, a componente de maior frequência desse sinal não pode ser maior que a metade da frequência do sinal de amostragem.

do sinal analógico encontrará representação digital mais aproximada. O número de níveis que o quantizador possui é função exponencial da quantidade de bits do conversor e pode ser expresso pela função: $q = 2^n$, onde q é o número de níveis do quantizador e n é o número de bits de precisão do ADC. Portanto, para um conversor com 4 bits de precisão, haverá 16 níveis de quantização, conforme mostrado na Figura 7. Outro aspecto do circuito ADC são os distintos métodos de conversão existentes (p. ex. flash e dupla rampa), os quais distinguem-se por suas características técnicas. Segundo [Floyd \(2009\)](#), o método de aproximação sucessiva é o mais utilizado e apresenta como características técnicas, maior velocidade de conversão quando comparado com o método dupla rampa, contudo, mais lenta em relação ao flash, e seu tempo de conversão é fixo independente da tensão do sinal analógico percebida na entrada. O método de aproximações sucessivas é o utilizado pelo microcontrolador presente na placa Arduino, usada no presente trabalho.

Figura 7 – Gráfico do processo de quantização.



A partir da análise do gráfico é possível identificar os códigos binários gerados pelo processo de quantização a cada intervalo de amostragem. No primeiro intervalo, o código binário será 0000, no segundo será 0101 e assim por diante até o décimo terceiro intervalo, onde o código será 1110. Fonte: Adaptada de ([FLOYD, 2009](#), p. 766).

5 Dispositivo Microcontrolado Arduino

Arduino é um dispositivo microcontrolado criado na *Ivrea Interaction Design Institute* – localizada no norte da Itália – cujo projeto baseia-se no conceito *easy-to-use*¹ e busca produzir placas de prototipagem rápida cujas funcionalidades sejam de fácil aprendizado e utilização ([ARDUINO, s.d.c](#)). Para isso, o projeto Arduino disponibiliza uma IDE onde é possível escrever programas, em linguagem de alto nível, e transferi-los para o microcontrolador da placa. Esse ambiente de desenvolvimento integrado encapsula rotinas que manipulam diretamente o microcontrolador e as disponibiliza através das bibliotecas de código com interfaces definidas e prontas para uso. Isso concorre para diminuição da complexidade envolvida no processo de configuração do microcontrolador, de forma que, reduz o esforço necessário para manipular suas funcionalidades (interrupções, configurações das portas de entrada e saída etc.) e, por consequência, a escrita do programa torna-se relativamente mais simples e rápida.

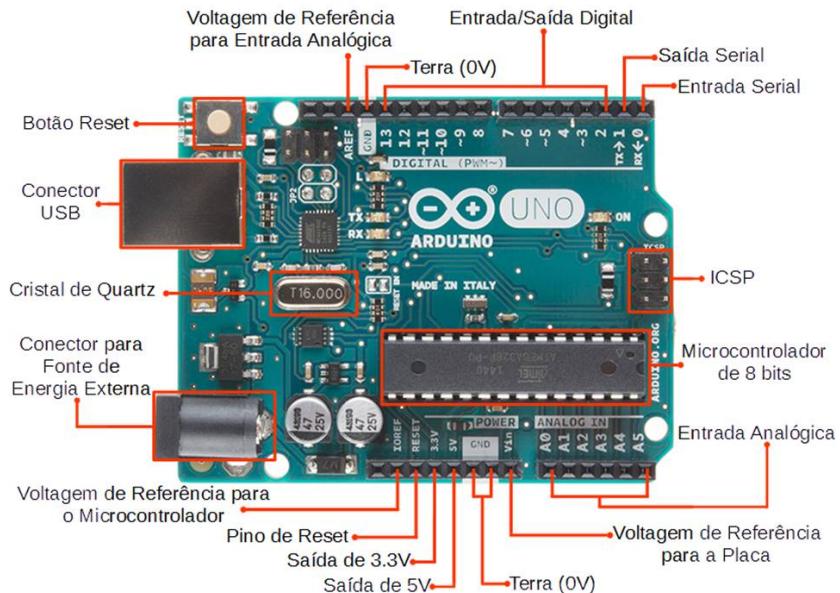
Existem diversos modelos de Arduino, onde cada tipo de placa apresenta uma arquitetura diferente das demais e adequa-se melhor a determinadas aplicações que as outras, por exemplo, o Arduino GEMMA foi desenvolvido com o propósito de atender a projetos onde as placas necessitam ser embutidas em vestimentas², o que justifica seu tamanho reduzido e hardware limitado (número muito pequeno de portas digitais – 3 no total –, pouca memória flash entre outras.). Já o modelo adotado no presente trabalho, Arduino UNO, consegue atender a aplicações mais complexas em relação ao que o GEMMA é capaz, além de ser o mais documentado e utilizado dentre todas as placas Arduino já produzidas ([ARDUINO, s.d.b](#)). A placa UNO funciona com o microcontrolador de 8 bits ATmega328P que possui 32 kB de memória flash, 2 kB de SRAM, 1 kB de EEPROM, conversor A/D que utiliza o método de aproximação sucessiva com até 10 bits de resolução entre outras configurações. Esse microcontrolador pode funcionar com clock máximo de 20 MHz, contudo, o cristal externo dessa placa opera com velocidades de até 16 MHz. O Arduino UNO possui 14 pinos digitais de entrada e saída, 6 pinos analógicos de entrada, uma porta USB e um conector tipo *jack* ([SOUZA, 2013c](#)). Essas configurações e outros componentes que auxiliam no funcionamento da placa são apresentados na Figura 8.

Os pinos digitais (também conhecidos como portas digitais) do UNO podem ser utilizados tanto para entrada quanto para saída de sinais. A configuração desses pinos é feita durante a escrita do programa de controle, onde define-se quais portas serão utilizadas e o modo de funcionamento de cada uma delas. Quando configurado como modo de entrada, o pino digital apresenta alta impedância e consome pouquíssima corrente do

¹ Termo utilizado para denotar que uma dada tecnologia é de fácil compreensão e uso.

² “Lilypad Arduino Projects”. Disponível em: (<https://www.youtube.com/watch?v=L4a89n4ZJ5w>)

Figura 8 – Placa Arduino UNO R3



Fonte: Adaptada de ([SPARKFUN, 2017](#)).

circuito conectado a si. Por outra via, um pino configurado como saída possui baixa impedância e consegue fornecer para o circuito externo uma corrente máxima de 40 mA, contudo, a corrente fornecida por todos os pinos de saída não pode ultrapassar o limite de 200 mA ([SOUZA, 2013b](#)). A sintaxe para configuração dos pinos é `pinMode(pino, modo)`. O primeiro parâmetro dessa função é o número do pino que se deseja utilizar (os pinos digitais são numerados de 0-13) e o segundo, refere-se ao modo de funcionamento desse pino. É possível passar um dos três seguintes valores para esse parâmetro: INPUT, INPUT_PULLUP e OUTPUT, os quais significam, respectivamente, entrada digital, entrada digital com resistor *pull-up*³ interno habilitado e saída digital. Há ainda duas funções que permitem “escrever” e “ler” valores lógicos com as portas digitais. A `digitalWrite(pino, valor)` é a interface presente na IDE que possibilita atribuir um valor lógico (LOW ou HIGH) no pino de saída indicado no primeiro parâmetro dessa função. E a leitura do valor presente na porta digital é feita através da função de sintaxe `digitalRead(pino)`, que apresenta como retorno um valor lógico.

Conforme citado, o UNO possui 6 entradas para sinais analógicos que podem ser manipuladas via software e, diferente das portas digitais, somente operam como entradas, dispensando portanto, que seja definido seu modo de operação. É por meio do ADC, presente em seu microcontrolador, que essa placa consegue traduzir e manipular os sinais analógicos que chegam às entradas analógicas, denominadas de A0 a A5. As interfaces para fazer uso das portas e do conversor analógico-digital fazem parte da IDE e possibilitam a configuração da tensão de referência que o ADC deverá utilizar, assim como, ler valores

³ Resistor que liga o pino ao VCC para garantir que seu estado lógico seja HIGH.

em um dos 6 canais de entradas multiplexados do conversor. Souza (2013a) descreve que a interface, responsável por fazer a leitura dos valores analógico, seleciona o canal de entrada conforme a porta indicada como parâmetro da função. A interface para manipular a tensão de referência é `analogReference(tipo)`, onde o parâmetro *tipo* pode assumir – na placa UNO – o valor `DEFAULT` (tensão de alimentação da placa, 5 V), ou `INTERNAL` (tensão interna de 1.1 V), ou `EXTERNAL` (tensão aplicada no pino AREF que pode variar entre 0 V e 5V). A leitura do canal ocorre por meio da função `analogRead(pino)` que retorna um valor inteiro de 0 e 1023 e recebe como parâmetro o número da porta analógica a ser lida, que no UNO pode ser de 0 a 5. Cabe ressaltar que o tempo necessário para essa função ler o valor da porta é de aproximadamente 100 microssegundos, portanto a frequência máxima de leitura é 10 kHz (SOUZA, 2013a). Dessa forma, a utilização da interface presente na IDE para perceber os valores analógicos monitorados pelo aparato experimental não proporciona riscos de incoerência (perda de um valor não lido em tempo hábil), pois esses valores variam numa frequência inferior à máxima apresentada pela `analogRead(pino)`.

Algumas vezes, durante a execução do programa de controle, torna-se necessário interromper seu fluxo sequencial a fim de responder a algum tipo de modificação percebida no sistema monitorado, ou deseja-se que o microcontrolador realize determinada tarefa de forma autônoma em intervalos de tempos definidos e, após isso, retorne para o fluxo normal de execução. Isso é possível com o uso dos recursos presentes no microcontrolador da placa UNO, trata-se dos *Timers* e das *Interruptions*. O *timer* é um registrador presente nos microcontroladores e funciona como um contador que incrementa seu valor a cada pulso de clock, até que ultrapasse seu limite de armazenamento e reinicie a contagem (ROBOTFREAK, 2014). O microcontrolador do UNO possui 3 *timers* denominados de timer0, timer1 e timer2 que podem ser manipulados através da biblioteca nativa da IDE (Timer) ou bibliotecas de terceiros, por exemplo, a MsTimer2 – utilizada no SMCD deste projeto – que utiliza interrupção com timer2 do microcontrolador. Cada registrador de tempo possui uma resolução em bits que determina o seu valor limite de armazenamento. Para o timer0 e timer2, essa resolução é de 8bits, portanto, seu armazenamento limita-se a valores de 0 a 255, enquanto o timer1 consegue atingir de 0 a 65535, pois possui 16bits de resolução (ROBOTFREAK, 2014). Existem dois tipos de interrupções na placa UNO, a externa cujo acionamento está relacionado com o estado do pino 2 ou 3, e a interna que utiliza os temporizadores internos para acionar a interrupção. As interrupções paralisam o fluxo sequencial do programa para executar uma tarefa determinada pelo usuário, e logo após encerrar essa tarefa, retorna para a próxima instrução após o ponto onde ocorreu a interrupção. A biblioteca MsTimer2 abstrai a complexidade de configuração do timer2 e da interrupção, de modo que, o usuário apenas precisa fornecer o intervalo de tempo em milissegundos e a função a ser executada durante a interrupção como parâmetros. MsTimer2 possui apenas três interfaces, `MsTimer2::set(unsigned long ms,`

função) configura a interrupção e o temporizador, MsTimer2::start() habilita a interrupção e MsTimer2::stop() desabilita a interrupção.

Embora a MsTimer2 torne o processo de configuração da interrupção mais simples, o uso de interrupções no Arduino requer certos cuidados. A função executada durante a interrupção é uma rotina do serviço de interrupção (do inglês, Interrupt Service Routine – ISR) e possui limitações que a difere das funções comuns. Essa rotina, obrigatoriamente, não deve ter parâmetros e nem retornar dados. Funções que utilizam interrupções internamente podem não funcionar, como acontece com delay() e millis(), ou apresentar comportamento anormal quando chamadas dentro da rotina, por exemplo, micros(). Uma vez que, não é permitido passar parâmetros para as ISRs, é comum utilizar variáveis globais dentro dessa função, contudo, deve-se utilizar o modificador *volatile* na declaração dessas variáveis quando pretende-se alterar seu valor durante a execução da rotina para evitar erros de inconsistência. ([ARDUINO, s.d.a](#)).

6 Ambiente Aberto para Experimento de Conversão Fotovoltaica

O experimento didático de conversão fotovoltaica proporciona ao aluno, do mesmo modo que ao professor, a oportunidade de debruçar-se sobre o assunto contemporâneo e de relevância mundial: conversão de energia luminosa em energia elétrica. Esse experimento foi desenvolvido neste trabalho como parte do objetivo proposto pelo mesmo: criar um ambiente aberto de hardware e software, baseado no dispositivo microcontrolado Arduino e na plataforma computacional Raspberry Pi para experimentos em física.

6.1 Desenho do Experimento

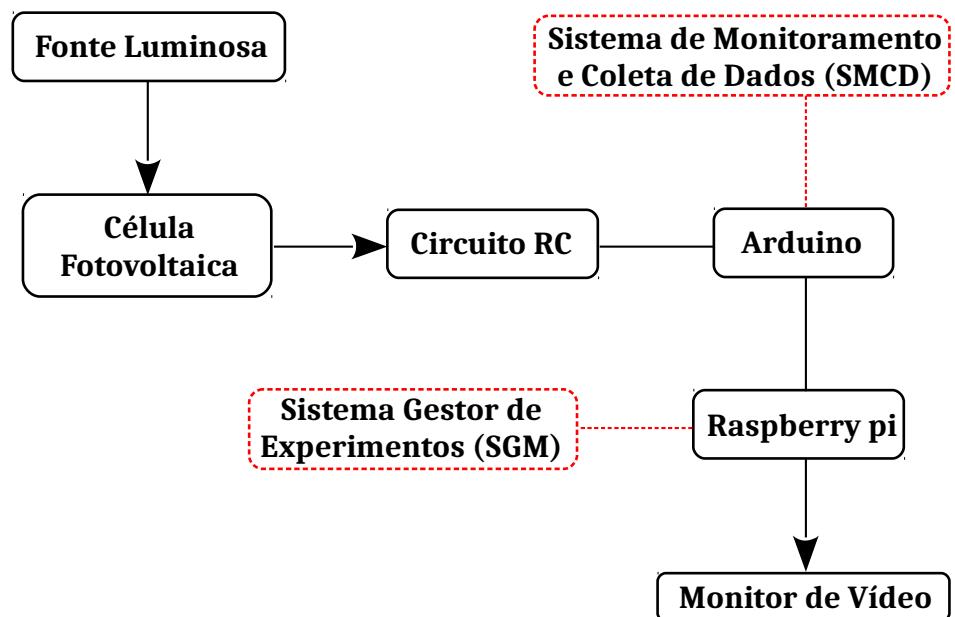
O experimento de conversão fotovoltaica, realizado no presente projeto, foi elaborado com o propósito de analisar o comportamento de uma célula fotovoltaica sob diferentes condições de iluminação, durante o processo de carga do capacitor num circuito RC. As variáveis manipuladas no experimento foram: temperatura de cor das lâmpadas utilizadas como fonte luminosa, resistores do circuito RC e a distância entre fonte luminosa e célula fotovoltaica. O monitoramento desse experimento foi realizado de modo automatizado, através da plataforma Arduino UNO, que coletou os dados gerados ao longo do tempo de execução e os enviou para o Sistema Gestor de Experimentos.

O experimento realizado com uso do aparato instrumental elaborado neste trabalho consiste em utilizar uma lâmpada, com determinada potência nominal, para incidir luz sobre uma célula fotovoltaica que produzirá energia para alimentar o circuito RC e carregar o capacitor ao longo tempo. Essa tensão gerada pela célula será monitorada através da porta analógica do UNO por um período equivalente a 5τ , a partir desse instante de tempo o monitoramento encerra-se e a tensão em τ será analisada a fim de identificar o comportamento apresentado pela fotocélula. Cabe ressaltar que, optou-se por analisar o comportamento da fotocélula no instante τ , devido a esse valor ser referência na maioria dos circuitos, como em circuitos osciladores.

A arquitetura do ambiente aberto desenvolvido neste trabalho é composta por hardware (fonte luminosa, suporte para lâmpada luminosa impresso em 3d, célula fotovoltaica, circuito eletrônico RC, Arduino UNO, Raspberry Pi modelo B e monitor de vídeo) e software (SGE e SMCD). O arranjo e a interação entre os “elementos” dessa arquitetura ocorrem da seguinte forma: a fonte luminosa incide luz sobre a célula fotovoltaica, que por sua vez, alimenta o circuito RC com a tensão proveniente do processo de conversão fotovoltaica. Esse circuito está conectado a uma das portas analógicas do Arduino, e

seu monitoramento é feito pelo sistema SMCD que coleta os dados a cada instante de tempo t e os envia para a plataforma Raspberry Pi através da porta serial do UNO. Os dados coletados são armazenadas, processadas e exibidas por meio do Raspberry Pi. A Figura 9 apresenta uma visão geral da arquitetura desse ambiente e a interação entre seus componentes. Ademais, uma compilação de imagens do aparato experimental montado, encontra-se disponível no apêndice A.

Figura 9 – Arquitetura do aparato experimental.



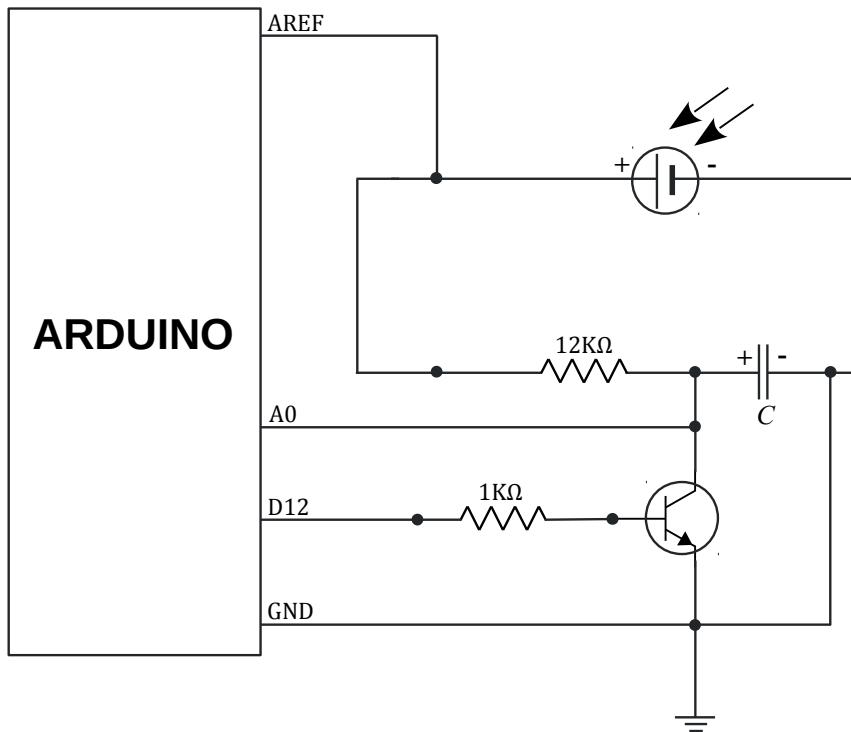
Fonte: Autor.

6.1.1 Circuito RC e Arduino

Conforme mostrado na Figura 9, existe no aparato experimental um circuito RC composto por capacitor, resistor, transistor de junção bipolar e fotocélula. Nesse circuito, a fotocélula atua como fonte de tensão, cujo polo positivo é conectado ao capacitor por intermédio do resistor e o polo negativo ao terminal negativo do capacitor. O fluxo da corrente que atravessa o resistor e chega ao capacitor pode ser modificado pelo transistor, que tem o coletor conectado à ligação do resistor com o terminal positivo do capacitor e o emissor conectado à referência comum de 0V (*GND*) do circuito RC. Ademais, o transistor atua como dreno das cargas acumuladas no capacitor e seu controle é feito pelo Arduino por intermédio da saída digital D12 (pino 12), conectada à base do transistor. Quando a fonte de tensão está ativa no circuito e o valor lógico na porta D12 é *HIGH*, a corrente no

circuito atravessa o resistor mas não chega ao capacitor, pois o transistor opera em estado de saturação, drenando a corrente desde o coletor até o *GND*, e garantindo que o capacitor seja descarregado. Por outro lado, quando o valor em D12 é *LOW*, a corrente chega ao capacitor porque o transistor opera em estado de corte. Desse modo, o Arduino controla o circuito, tanto para determinar quando a corrente passará a fluir até o capacitor quanto para descarregar o capacitor ao término do experimento. Isso é fundamental para garantir que a carga no capacitor será muito próxima de zero antes do início do experimento (ao desviar o fluxo da corrente) e após seu término (ao descarregar o capacitor pelo dreno). Além da conexão para exercer controle sobre o dreno, há uma ligação entre o pino analógico A0 e a conexão do resistor com capacitor, por meio da qual o Arduino mede a tensão do capacitor. A fim de definir o fundo de escala para conversão analógico-digital realizada pelo Arduino, existe uma ligação entre seu pino AREF e o terminal positivo da fotocélula. Por fim, existe uma conexão entre o *GND* do Arduino com a referência de 0 V do circuito, para que a tensão medida no Arduino possua a mesma referência que o circuito RC. O diagrama esquemático do circuito RC com o Arduino é mostrado abaixo na Figura 10:

Figura 10 – Diagrama esquemático do circuito RC.



Fonte: Autor.

6.1.2 Desenvolvimento do Sistema de Monitoramento e Coleta de Dados

Conforme citado, o monitoramento do experimento foi realizado pelo UNO e o nome atribuído ao programa de controle executado por seu microcontrolador foi Sistema de

Monitoramento e Coleta de Dados (SMCD). Esse sistema é composto por um conjunto de rotinas que torna o UNO capaz de controlar a passagem de corrente da célula fotovoltaica para o circuito RC, monitorar a tensão no capacitor em intervalos constantes de tempo, e comunicar-se com a plataforma computacional Raspberry Pi para troca de dados. Nesta seção, as rotinas do SMCD serão apresentadas e descritas.

As primeiras instruções do SMCD são as definições das constantes e das variáveis utilizadas pelo sistema. As constantes END_TRANSMISSION, START_EXPERIMENT e DURATION_EXPERIMENT são utilizadas como código de sinal para comunicação serial entre Sistema Gestor de Experimentos (SGE) e SMCD. Existem, ainda, as constantes RESISTOR, CAPACITOR e TAU que estão relacionadas com o circuito RC. E a última constante do sistema é RESET_PIN que refere-se à terceira porta digital do UNO. Quanto às variáveis, encontram-se presentes no sistema, o vetor *seconds_and_volts* que acumula os dados monitorados do experimento para posterior envio para o SGE e os dados que armazena são do tipo estruturado, denominado *Data_experiment*, composto por três variáveis: *second*, *separator* e *volt*; a *indice* utilizada para percorrer o vetor, a *time_* que serve para acumular o tempo transcorrido do experimento, a *previous_time* que armazena o instante de tempo anterior ao tempo em *time_*, a *volt* que armazena a tensão do capacitor e a *signal_code* que armazena o último sinal de código recebido do SGE. O trecho de código 6.1 apresenta o trecho de código acima descrito para as declarações de variáveis e constantes.

```

1 struct Data_experiment{
2     int second;
3     char separator = ';';
4     float volt;
5 };
6
7 const char END_TRANSMISSION = 'e';
8 const char START_EXPERIMENT = 's';
9 const char DURATION_EXPERIMENT = 't';
10 const float RESISTOR = 12e3;
11 const float CAPACITOR = 3300e-6;
12 const long TAU = 5*(RESISTOR*CAPACITOR);
13 const unsigned int RESET_PIN = 12;
14 Data_experiment seconds_and_volts[5];
15 int index = 0;
16 volatile int time_ = 0;
17 int previous_time = 0;
18 float volt = 0.0;
19 char signal_code = ' ';
```

Código 6.1 – Conjunto de variáveis e constantes do SMCD

Após essas definições, foram escritas as rotinas de configuração e de execução,

e as sub-rotinas de interrupção, de conversão do valor de tensão e de verificação do código de sinal. As configurações foram escritas no escopo interno da rotina *setup()*, que é executada apenas uma única vez ao carregar o programa para o microcontrolador e seu uso é obrigatório, ainda que seja com escopo vazio. Nessa função foram definidas as seguintes configurações: modo de operação da porta RESET_PIN como saída, a taxa de transmissão da comunicação serial em 9600 bits por segundo (*baund*), o intervalo de 1000 milissegundos para interrupção e a ISR *measurement_time* como parâmetros da função *MsTimer2::set* e o valor HIGH no RESET_PIN para manter o dreno da corrente no capacitor aberto, desse modo garante-se que a corrente nele será muito próxima de zero no instante em que iniciar o experimento. O trecho de código 6.2 apresenta o trecho de código da rotina *setup()*.

```

1 void setup() {
2   pinMode( RESET_PIN, OUTPUT );
3   Serial.begin( 9600 );
4   MsTimer2::set( 1000, measurement_time );
5   digitalWrite( RESET_PIN, HIGH );
6 }
```

Código 6.2 – Rotina *setup()*

Os códigos da rotina que deve ser executada durante todo o tempo em que a placa UNO permanecer ligada foram escritos na rotina *loop()* (de uso obrigatório, assim como, a *setup()*), a qual funciona como uma iteração sem fim, ou seja, as instruções dentro do escopo dessa função são executadas até que o Arduino seja desligado. Existem três blocos de tomada de decisão no escopo da rotina *loop()*. No fluxo principal do primeiro bloco, o SMCD aguarda que o SGE envie o sinal de código solicitando o tempo de duração do experimento, após receber esse código o SMCD envia o tempo de duração para o SGE e torna a esperar que o SGE envie o código para iniciar o experimento. Após receber o sinal para iniciar o experimento, o SMCD fecha o dreno da corrente do capacitor, habilita a interrupção e inicia a contagem do tempo do experimento. Além disso, após esse primeiro bloco, é feito o *polling* na porta de entrada analógica A0, com uso da função *analogRead*, e essa função retorna a representação binária gerada pelo conversor ADC para o valor analógico percebido na porta A0. Esse valor binário é mapeado pela sub-rotina *convertIntoVolts* para sua respectiva representação dentro da faixa de tensão. Para fazer esse mapeamento, essa sub-rotina executa uma regra de três simples entre a tensão máxima da fonte de alimentação (definida pelo usuário com base na ddp dessa fonte), o maior valor binário que o ADC pode gerar e a representação binária da tensão no pino A0. O trecho de código 6.3 apresenta o trecho de código desse primeiro bloco da rotina *loop()*.

```

1 if ( signal_code != START_EXPERIMENT ) {
2  waitForRequest(DURATION_EXPERIMENT);
3   Serial.print(TAU);
4   waitForRequest(START_EXPERIMENT);
5   digitalWrite( RESET_PIN, LOW );
```

```

6     MsTimer2 :: start ();
7 }
```

Código 6.3 – Código da primeira estrutura de decisão da rotina *loop()*

O segundo bloco trata da transferência dos valores de tempo e tensão, monitorados pelo SMCD, para o SGE. Para isso, a estrutura de decisão verifica se o tempo de execução do experimento avançou de acordo com o intervalo estabelecido para contagem do tempo, em outras palavras, confere se o valor na variável *previous_time* é menor do que o valor na *time_*. Caso o retorno dessa comparação seja verdadeiro, o SMCD acumula no *seconds_and_volts* os valores contidos em *time_* e *volt*. Além disso, é verificado se o limite de armazenamento desse vetor foi atingido, e caso o tenha sido, envia-se todos os valores do vetor para o SGE acompanhados do caractere “|” como separador entre os dados estruturados e o valor de *index* é reiniciado, caso contrário, a variável *index* é incrementada em 1. Ademais, no final do segundo bloco, o valor de *previous_time* é atualizado com o valor de *time_*. O trecho de código 6.4 apresenta o trecho de código desse segundo bloco da rotina *loop()*.

```

1 if (previous_time < time_){
2     seconds_and_volts [ index ].second = time_ ;
3     seconds_and_volts [ index ].volt = volt ;
4     if (index < 4){
5         index += 1;
6     } else{
7         for (int i = 0; i <= index; i++){
8             Serial.print (seconds_and_volts [ i ].second);
9             Serial.print (seconds_and_volts [ i ].separator);
10            Serial.print (seconds_and_volts [ i ].volt);
11            Serial.print ('|');
12        }
13        index = 0;
14    }
15    previous_time = time_;
16 }
```

Código 6.4 – Código da segunda estrutura de decisão da rotina *loop()*

A terceira estrutura de decisão, destina-se a verificar se o tempo de execução do experimento alcançou o tempo de duração indicado pela constante Tau. Quando *time_* é igual à constante Tau, é verificada a existência de dados mensurados que não foram enviados para o SGE, caso existam, todos são enviados da mesma forma que ocorre na segunda estrutura de decisão, e após isso, a contagem do tempo é encerrada com o uso de *MsTimer2::stop()* - que desabilita a interrupção de tempo -, as variáveis *time_* e *signal_code* recebem seus valores iniciais novamente, o dreno da corrente do capacitor é aberto e o sinal de código que representa término do experimento é enviado para o SGE. O trecho de código 6.5 apresenta o trecho de código desse terceiro bloco da rotina *loop()*.

```

1  if (time_ == TAU){
2      MsTimer2::stop();
3      if (index != 0){
4          for (int i = 0; i < index; i++){
5              Serial.print(seconds_and_volts[i].second);
6              Serial.print(seconds_and_volts[i].separator);
7              Serial.print(seconds_and_volts[i].volt);
8              Serial.print(' | ');
9          }
10     }
11     time_ = 0;
12     previous_time = 0;
13     signal_code = ' ';
14     digitalWrite( RESET_PIN, HIGH );
15     //delay(500);
16     Serial.print(END_TRANSMISSION);
17     index = 0;
18 }
```

Código 6.5 – Código da terceira estrutura de decisão da rotina *loop()*

A sub-rotina de interrupção, a de conversão do valor de tensão e a de verificação do código de sinal, foram escritas fora do escopo da função *loop()*. Onde a sub-rotina *measurement_time* incrementa a variável *time_* em 1 a cada 1000 milissegundos (1 segundo), a sub-rotina *convertIntoVolts* transforma o valor lido da porta analógica A0 – que é um valor analógico convertido para um valor digital correspondente numa faixa que vai de 0 a 1023 – para um valor entre 0 e a ddp da fonte de alimentação do circuito e a sub-rotina *waitForRequest* aguarda até que o SGE envie um sinal para o SMCD e analisa se aquele sinal está correto para o escopo sendo executado no momento em que o sinal foi recebido. O trecho de código 6.6 apresenta o trecho de código das sub-rotinas citadas e o fluxograma que demonstra todos os fluxos de execução do Sistema de Monitoramento e Coleta de Dados, encontra-se disponível no apêndice C. Além disso, o código completo do SMCD foi incluído no apêndice B.

```

1 float convertIntoVolts( float valor ){
2     return ( valor *3.88 ) /1023;
3 }
4
5 void measurement_time() {
6     time_ += 1;
7 }
8
9 void waitForRequest( char code ){
10    while( signal_code != code ){
11        if( Serial.available() > 0 ){
12            signal_code = Serial.read();
```

```

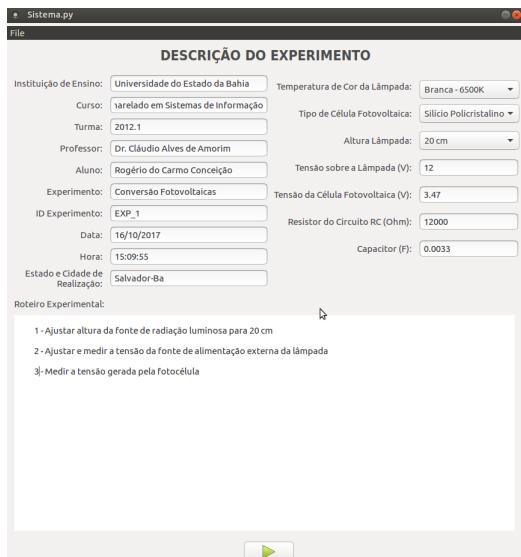
13     }
14 }
```

Código 6.6 – Sub-rotinas do SMCD

6.2 Gestão do Experimento e Apresentação dos Resultados

Para gerenciar a execução do experimento foi desenvolvido o Sistema Gestor do Experimento (SGE) que funciona na plataforma computacional Raspberry Pi, por intermédio do qual é possível iniciar o experimento e visualizar o resultado gerado ao término. Na primeira tela desse software, conforme mostrado na figura 11, devem ser preenchidos os dados cadastrais (nome da instituição, aluno, professor etc.) e os dados experimentais necessários à execução do experimento (temperatura de cor da lâmpada, energia consumida pela fotocélula etc.), após isso, com um clique, a execução do experimento é iniciada. A partir desse momento, o sistema gestor exerce controle sobre o andamento

Figura 11 – Tela principal do SGE

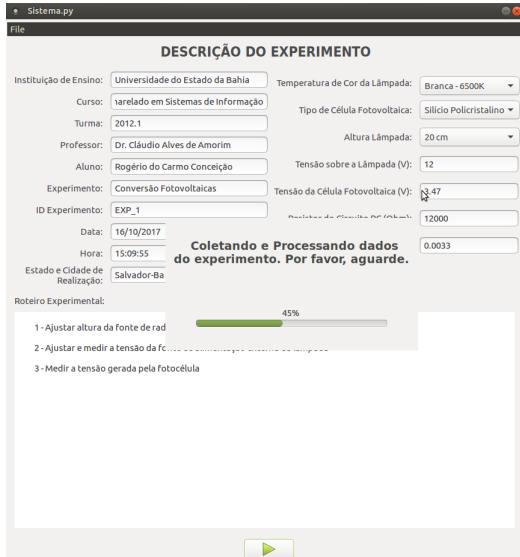


Fonte: Autor.

do experimento e passa a trocar mensagens com o sistema de monitoramento a fim de coordenar o processo de monitoramento do experimento. Essa comunicação ocorre da seguinte maneira: O SGE envia uma mensagem solicitando ao sistema de monitoramento que informe o tempo de duração do experimento, ao receber essa mensagem, o SMCD remete a informação solicitada e aguarda a próxima mensagem. Após obter a informação solicitada, o SGE ordena que o sistema de monitoramento inicie a coleta dos dados e aguarda pela chegada desses dados para armazená-los e processá-los até que o último seja enviado pelo SMCD. O feedback acerca do processo da coleta de dados é fornecido ao usuário pelo SGE através da tela de espera. Essa tela, exibida na figura 12, apresenta uma

pequena barra de porcentagem que indica o avanço da execução do experimento, e ao término, exibe um botão para a terceira tela do sistema, a tela de resultados. Os dados

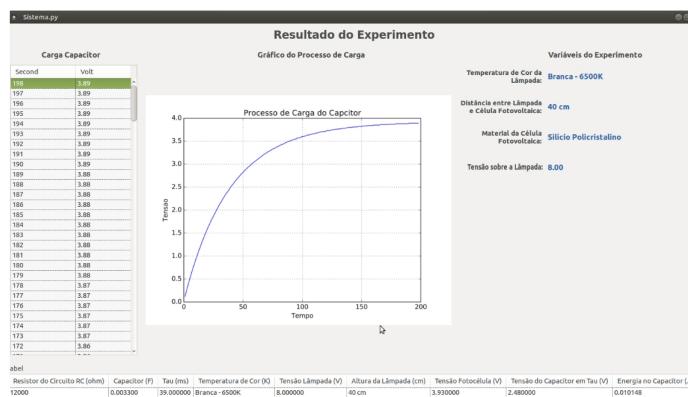
Figura 12 – Tela de feedback da execução do experimento



Fonte: Autor.

de tempo e tensão do capacitor, coletados pelo SMCD, o gráfico da curva de carga do capacitor gerado pelo SGE através da biblioteca matplotlib e um painel com informações das variáveis do experimento são exibidos na tela de resultados, conforme é possível observar na figura 13.

Figura 13 – Tela de Resultado do SGE



Fonte: Autor.

6.2.1 Desenvolvimento do Sistema Gestor de Experimentos

O Sistema de Gestão do Experimento (SGE) foi desenvolvido para automatizar as atividades de gerenciamento dos dados experimentais e exibição dos resultados do

experimento, atuando como ferramenta de suporte ao processo experimental. As funcionalidades desse software foram implementadas utilizando a linguagem de programação Python, dadas as facilidades técnicas que essa linguagem oferece por ter suporte nativo em todos os sistemas operacionais disponíveis para Raspberry Pi e por possuir bibliotecas estáveis e livres, como a pyserial e a matplotlib, mas também, conforme identificado através de leituras realizadas, ser considerada uma linguagem apropriada – do ponto de vista didático – para o aprendizado inicial de programação com Raspberry Pi (p. ex. Projeto Code Club¹). Ademais, cabe ressaltar que o Python é uma linguagem muito utilizada para análise de dados. As interfaces gráficas foram desenvolvidas em GTK+² através da IDE Glayde. O código-fonte completo do SGE, e os arquivos xml das interfaces, foram armazenados no GitHub, plataforma de hospedagem de código-fonte com controle de versionamento utilizando Git, e pode ser acessado livremente no [repositório³](#) online. Esse foi o primeiro passo para tornar o software livre, os demais procedimentos exigidos⁴ para colocar o sistema sob a licença GNU General Public License (GPL) serão realizados após a conclusão deste trabalho. O Sistema de Gerenciamento do Experimento foi desenvolvido utilizando programação orientada a objetos com aplicação do padrão de desenvolvimento MVC (Model-View-Controller) com vistas a diminuir o acoplamento entre as classes do sistema. Visto que, o número de classes presentes no SGE torna inviável a apresentação e discussão de todas neste trabalho, nesta seção será dada ênfase aos trechos de códigos referentes à comunicação serial entre SGE e SMCD e à geração do gráfico com base nos dados coletados, pois uma funcionalidade permite demonstrar como foi feita a comunicação entre plataformas de diferentes arquiteturas, e a outra, o modo como os gráficos foram criados com o uso da biblioteca livre matplotlib.

O tempo de execução de um experimento varia de acordo com o arranjo experimental utilizado, de modo que, um ensaio pode durar algumas dezenas de segundos, ou, até mesmo, algumas dezenas de minutos. Ora, quando o usuário interage com uma aplicação que demora cinco segundos para responder as solicitações feitas, esse usuário pode ter a sensação de que algo não está correto no funcionamento do sistema. A fim de evitar isso, o processo de comunicação entre a plataforma Raspberry Pi e o arduino foi implementado para ocorrer em paralelo com a tela onde o estado atual do processo experimental é exibido em tempo real, em conformidade com a comunicação serial em ambiente. Esse mecanismo foi desenvolvido através do uso de uma *thread* responsável pelas interações do SGE com o SMCD, a qual, possui quatro métodos, além do construtor: *run()*, *start_measurement_experiment()*, *get_measurement_data()* e *split_data(data)*. Quando a *thread* é criada, em tempo de execução, *run()* é o único método executado, por isso os demais métodos são chamados em seu

¹ <https://www.codeclubbrasil.org.br/sobre-o-projeto/>

² Conjunto de ferramentas multiplataforma para desenvolvimento de interfaces gráficas

³ [System for Management of Experiments in Physics Didactic Labs](#)

⁴ Os procedimentos necessários para utilizar a licença GNU GLP estão disponíveis em <http://www.gnu.org/licenses/gpl-howto.html>

escopo, conforme exibido no trecho de código 6.7.

```

1  def run(self):
2      self.start_measurement_experiment()
3      time.sleep(0.4)
4      wait_screen.get_duration_experiment()
5      self.get_measurement_data()

```

Código 6.7 – Trecho de Código do método *run()*

A primeira troca de mensagens entre as plataformas, ocorre para que o SGE obtenha o tempo de duração do experimento, a fim de configurar o valor limite para a barra de status da tela de *feedback*. Isso é executado através do método *start_measurement_experiment()*, onde o SGE cria uma porta de comunicação serial com o SMCD, envia um sinal de requisição do tempo de duração e, após recebê-lo, esvazia o *buffer* para garantir que na próxima vez em que o SMCD enviar dados pelo canal, não haverá ”lixo” no *buffer*, e configura a barra de status. O trecho de código 6.8 exibe o escopo interno desse método.

```

1  def start_measurement_experiment(self):
2      self.port_serial = SerialCommunication()
3      self.port_serial.request_duration_experiment()
4
5      # Verifica se o buffer contem dados, caso nao haja aguarda 0.3 antes de
6      # realizar a leitura
7      if(self.port_serial.get_channel().in_waiting == 0):
8          time.sleep(0.3)
9
10     self.qnt_byte = self.port_serial.get_channel().in_waiting
11     self.duration_of_experiment = self.port_serial.receive(self.qnt_byte)
12     self.port_serial.get_channel().flush()
13
14     // Adiciona o tempo de duracao a tela de Loading
15     wait_screen.set_duration_experiment(self.duration_of_experiment)

```

Código 6.8 – Trecho de Código do método *start_measurement_experiment()*

No segundo método, resumidamente, ocorre a parametrização do tempo em que o SGE deverá gerenciar o experimento e o estabelecimento da comunicação serial através da porta que foi aberta entre as plataformas, até esse momento, o monitoramento dos dados pelo SMCD ainda não foi iniciado. Quando o método *run()* faz uma chamada a *def get_measurement_data()* a coleta dos dados é iniciada e o SGE passa a monitorar o canal de comunicação serial e armazena os dados recebidos do SMCD. A primeira ação a ocorrer nesse processo é o envio do sinal para que o SMCD inicie o monitoramento e coleta dos dados, após isso, o SGE passa a monitorar constantemente a porta serial e verifica se há dados no *buffer*, caso não haja, o sistema aguarda 3 segundos antes de verificar novamente. Quando são encontrados dados ocorre o seguinte:

1. o tamanho dos bytes presentes no *buffer* é recuperado e utilizado para ler exatamente todos os dados do *buffer*.
2. Os dados lidos são armazenados numa *list* e, por segurança, o *buffer* é esvaziado.
3. Dado que o SMCD enviar um conjunto de cinco pares de medidas tempo e tensão, o último dado armazenado é recuperado e passado como parâmetro para o quarto método da *thread*, onde o conjunto de medidas são separadas e armazenadas em uma tupla denominada de *times_measured*.
4. Após separados, o dado de tempo presente no último valor da tupla é utilizado é repassado para a classe responsável por atualizar a barra de espaço.
5. Por fim, esse mesmo dado de tempo é comparado com o tempo de duração do experimento para verificar se o experimento chegou ao fim e, caso o resultado da verificação seja positivo, o *buffer* é esvaziado, a comunicação serial e o monitoramento são encerrados.

O trecho de código 6.9 mostra o descrito acerca do terceiro método:

```

1  #Metodo responsavel por manter o polling na porta serial e pegar os dados
2  #enviados pelo arduino
3  def get_measurement_data(self):
4      self.port_serial.start_arduino()
5      self.index = 0
6
7      while True:
8          if(self.port_serial.get_channel().in_waiting != 0):
9              time.sleep(3)
10
11         # Pega a quantidade de bytes presente no buffer
12         self.qnt_bytes = self.port_serial.get_channel().in_waiting
13
14         # Ler do buffer a quantidade exata de dados presentes nele
15         set_results.add_measurement(self.port_serial.receive(self.qnt_bytes))
16         # Limpa o buffer por segurança
17         self.port_serial.get_channel().flush()
18
19         # Recupera a ultima tupla adicionada e remove o caractere separador
20         # entre as medidas enviadas
21         self.split_data(set_results.get_measurements()[self.index])
22
23         # Pega o ultimo par tempo-tensao, separa-os e adiciona o tempo como
24         # novo valor da razao entre tempo atual do experimento e o tempo total do
25         # experimento

```

```

23         wait_screen.set_new_fraction(int(self.times_measureds[len(
24             self.times_measureds)-2].split(';')[0]))
25
26     # Index utilizado para recuperar a ultima tupla de pares tempo-
27     # tensao
28     self.index += 1
29
30     # Verifica se o ultimo par tempo-tensao foi recebido e, caso seja
31     # verdade encerrar o polling no buffer
32     if(self.times_measureds[len(self.times_measureds)-2][0] ==
33         wait_screen.get_duration_experiment()):
34         self.port_serial.get_channel().flush()
35         self.port_serial.close_connection()
36         break
37
38     def split_data(self, data):
39         self.times_measureds = data.split('|')

```

Código 6.9 – Trecho de Código do método *get_measurement_data()*

Após encerrar a gestão da comunicação, o SGE possui todos os dados coletados, armazenados em uma *list* que pertence a um objeto da classe *SetofResult*, conforme descrito anteriormente. Logo após encerrar o monitoramento o SGE permite ao usuário acessar a tela de resultados, a lógica implementada para geração do gráfico utilizando os dados coletados foi implementada na classe *WindowResultExperiment*. A biblioteca utilizada na construção dessa lógica foi a matplotlib, o que permitiu gerar os gráficos utilizando poucos comandos conforme demonstrado no trecho de código 6.10. A fim de inserir os valores de tempo e tensão, coletados com a realização do experimento, no gráfico foram criadas duas listas, a lista *axis_x* recebeu o conjunto de dados de tempo e, a *axis_y*, o conjunto de valores de tensão. Dos módulos disponíveis no matplotlib, foi utilizado o pyplot que cria a figura e os eixos de forma implícita e automática. Após os gráficos terem sido gerados, uma imagem é criada e salva na pasta raiz do SGE, para então, ser carregada na tela de resultados.

```

1 self.index = 0
2     self.axis_x = []
3     self.axis_y = []
4
5     while (self.index < len(self.set_of_results.get_measurements())):
6         self.times_and_volts = self.set_of_results.
7         get_specific_measurement(self.index).split('|')
8
9         for i in range(0, len(self.times_and_volts)-1):
10            self.times_and_volts[i] = self.times_and_volts[i].split(';')
11            self.insert_data_table(self.times_and_volts[i][0], \

```

```
12      '',
13      add measurements datas to vector
14      '',
15      self.axis_x.append( self.times_and_volts[ i ][ 0 ] )
16      self.axis_y.append( self.times_and_volts[ i ][ 1 ] )
17
18      self.index += 1
19      #Faz o grafico para os dados coletados e insere os rotulos e uma grade
19      de fundo.
20
21      plt.plot( self.axis_x , self.axis_y )
22      plt.xlabel( 'Tempo' )
23      plt.ylabel( 'Tensao' )
24      plt.title( 'Processo de Carga do Capacitor' )
25      plt.grid( True )
26
27      #Cria uma figura e salva na pasta raiz do SGE
28      plt.savefig( "curva_capacitor.jpeg" , dpi = 800 )
29      graphic = self.window.get_object_from_window( "graphic" )
30
31      #Carrega a imagem salva na tela de resultados
32      self.pixbuf = GdkPixbuf.Pixbuf.new_from_file_at_scale( filename =
32      "curva_capacitor.jpeg" , width=700 , height=500 , preserve_aspect_ratio=True
32      )
33      graphic.set_from_pixbuf( self.pixbuf )
```

Código 6.10 – Trecho de Código do arquivo *WindowResultExperiment*

7 Experimentação e Validação dos Resultados

7.1 Testes Preliminares

Analizar a tensão e corrente gerados pela célula fotovoltaica foi a primeira atividade do processo de experimentação. Para isso, foram utilizados: parte do aparato experimental (fotocélula, lâmpadas e estrutura para variar a distância da incidência de radiação luminosa), resistores de $12\text{k}\Omega$, $2\text{k}\Omega$, $1\text{k}\Omega$ e 100Ω , fonte de tensão ajustável e um multímetro. A finalidade dessa análise era determinar quais resistores seriam utilizados na futura bateria de experimentos. Desse modo, a célula fotovoltaica foi estimulada (sob condições controladas) a gerar tensão e corrente através de resistores de diferentes resistências e os valores gerados foram medidos com uso do multímetro. Os valores obtidos para cada teste realizado é apresentado na tabela 3.

Tabela 3 – Valores de tensão e corrente para diferentes resistores

Tensão sobre a Lâmpada (V)	Temperatura de Cor (K)	Distância da fonte luminosa (cm)	Resistor (Ω)	Tensão Célula Fotovoltaica (V)	Corrente (mA)	Potência (mW)
12	3.000	20	12.000	4,13	0,34	1,404
			2.000	2,21	1,06	2,343
			1.000	1,37	1,31	1,795
			100	0,18	1,76	0,317
		40	12.000	3,11	0,26	0,809
			2.000	1,07	0,55	0,589
			1.000	0,59	0,61	0,360
			100	0,06	0,67	0,040
	6.500	20	12.000	4,10	0,35	1,435
			2.000	2,79	1,40	3,906
			1.000	1,66	1,48	2,457
			100	0,18	2,19	0,394
		40	12.000	3,32	0,27	0,896
			2.000	1,30	0,64	0,832
			1.000	0,71	0,72	0,511
			100	0,07	0,80	0,056

Fonte: Autor.

Durante a realização dos testes, foi utilizada uma fonte de alimentação que forneceu uma tensão de 12V e uma corrente de 170mA (confirmados com uso do multímetro) para

as lâmpadas, ou seja, a potência de cada lâmpada utilizada foi de 2,04W. A fotocélula utilizada nos testes é do tipo amorfo, sua tensão máxima é 5V e sua potência indicada é de 0,8W (ambos valores nominais), contudo, esse valor não foi alcançado pela fotocélula em nenhum dos testes, os quais, não envolveram avaliar a fotocélula sob luz solar direta. Os dois maiores valores de potência apresentados pela célula fotovoltaica foram com o uso dos resistores de $2\text{k}\Omega$ e de $1\text{k}\Omega$, entretanto, notou-se que a tensão gerada pela fotocélula tornou-se instável – dentro das condições de testes – de $2\text{k}\Omega$ para baixo. Desse modo, optou-se por adotar os valores de resistência $12\text{k}\Omega$ (no lugar de $1\text{k}\Omega$) e $2\text{k}\Omega$ para a bateria de experimentos com o circuito RC.

7.2 Baterias de Experimentos

O modelo experimental adotado consiste em utilizar uma célula fotovoltaica para carregar um capacitor sob diferentes condições e analisar o comportamento da fotocélula por intermédio da curva de carga do capacitor. Para estabelecer essas condições, as variáveis experimentais foram manipuladas da seguinte maneira: definiu-se dois pontos, 20 cm e 40 cm, a partir dos quais a incidência de radiação luminosa foi emitida sobre a fotocélula, as tensões de 6 V, 8 V, 10 V e 12 V foram utilizadas para alterar a intensidade das duas lâmpadas utilizadas (alternadamente) como fonte luminosa, as quais, possuem a mesma potência nominal e temperaturas de cor diferentes, a resistência do circuito RC foi alternada entre os valores de $2\text{k}\Omega$ e $12\text{k}\Omega$ e, por fim, foi estabelecido um intervalo de 60 segundos entre a realização de cada ensaio, com vistas a minimizar a perda da capacidade de geração de energia, sofrida pela fotocélula, em decorrência do aumento da temperatura ambiente ocasionada pela dissipação de calor da lâmpada. Desse modo, tornou-se possível avaliar o comportamento da fotocélula sob 32 cenários diferentes.

A experimentação foi realizada em duas baterias de ensaios experimentais, onde cada bateria gerou um conjunto de resultados que foram dispostos e organizados numa tabela com 12 colunas, conforme mostrado no apêndice G. Para cada ensaio realizado a tabela relaciona, resistência e capacitância utilizadas no circuito RC, constante de tempo τ , intervalo entre os ensaios, temperatura de cor da lâmpada, tensão sobre a lâmpada, distância da fonte luminosa, tensão da fotocélula, tensão prevista do capacitor em τ , tensão do capacitor em τ , percentual da diferença entre as tensões do capacitor em τ e razão entre as quantidades de energia armazenadas no capacitor em τ .

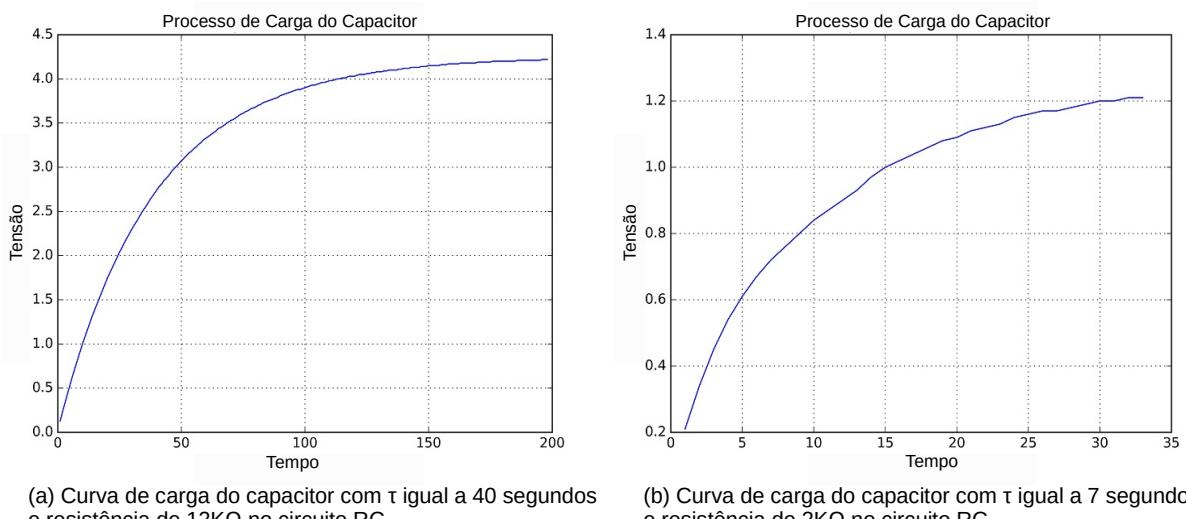
Para o cenário com, distância da fonte luminosa é 20 cm, resistor de $12\text{k}\Omega$, 12 V de tensão sobre a lâmpada e temperatura de cor igual a 6.500 K, a execução um ensaio experimental ocorre conforme os passos apresentados abaixo:

1. Iniciar o Sistema Gerenciador do Experimento.

2. Ajustar a distância da fonte luminosa para 20 cm.
3. Conectar a lâmpada com a temperatura de cor 6.500 K (caso não exista lâmpada conectada ou haja alguma de temperatura de cor diferente).
4. Conectar resistor de $12\text{ k}\Omega$ (Caso o resistor no circuito RC seja diferente).
5. Ajustar a fonte externa de alimentação da lâmpada para 12 V de tensão.
6. Medir com multímetro a tensão gerada pela fotocélula, inserir esse valor no código do SMCD e carregá-lo no Arduino.
7. Preencher as informações do experimento na tela principal do SGE e pressionar o botão *Iniciar Experimento*.
8. Aguardar a conclusão e pressionar o botão *Resultado* para ter acesso ao resultado.

A figura 14 mostra em (a) o gráfico típico da curva de carga do capacitor para o ensaio experimental acima, e em (b), o gráfico gerado para um cenário experimental similar ao descrito, com mudanças na temperatura de cor (3000 K em vez de 6500 K) e na resistência ($2\text{ k}\Omega$ em vez de $12\text{ k}\Omega$).

Figura 14 – Gráfico típico da curva de carga do capacitor no circuito RC



Fonte: Autor.

As baterias de experimentos foram divididas em relação à resistência utilizada no circuito RC. Para cada uma delas, os ensaios foram realizados de modo a explorar todas as combinações possíveis para as variáveis experimentais. Ademais, para cada tensão sobre a fonte luminosa foram executados seis ensaios, três para cada distância da fonte de radiação luminosa, a fim de examinar a consistência das medidas tomadas de forma automatizada.

Após a realização dos experimentos foram calculados o percentual da diferença entre tensão do capacitor medido em τ e tensão ideal esperada, e a razão entre a quantidade de energia armazenada no capacitor em τ , para distância de 20 cm e 40 cm. Cada bateria de experimentos possui o mesmo número de ensaios realizados, que somados totalizam 96 ensaios no intervalo de 480 minutos. Portanto, cada experimento tomou um tempo total de cinco minutos, na média. Os resultados obtidos com as baterias de experimentos encontram-se no apêndice F, para resistência de $2\text{ k}\Omega$, e para resistência de $12\text{ k}\Omega$, apêndice G. Para fins de análise dos resultados obtidos com as duas baterias de ensaios, optou-se por criar duas tabelas para sintetizar o conjunto de ensaios que compõe cada bateria. Em cada uma dessas tabelas é apresentado a média das medidas coletadas e dos resultados calculados com base nessas medições, conforme exibido na tabela 5 para os ensaios com resistência de $12\text{ k}\Omega$ e na tabela 4 para ensaios onde a resistência adotada foi $2\text{ k}\Omega$.

Conforme descrito no capítulo anterior, a tensão do capacitor em τ foi utilizada como parâmetro para análise do comportamento da fotocélula, além disso, esse parâmetro também indica se as medidas realizadas com uso do aparato são fidedignas e condizem ao que está sendo monitorado. A fim de validar o aparato experimental, os valores de tensão medidos em τ foram comparados com os valores esperados do ponto de vista teórico e o percentual da diferença entre esses valores foi calculado para cada ensaio realizado. A tabela de ensaios realizados com resistência de $12\text{ k}\Omega$ foi utilizada para validação do aparato. O percentual de erro dos valores medidos pelo aparato experimental ficou abaixo de aproximadamente 2 %, considerada aceitável, dadas imprecisões introduzidas no processo pelos componentes do aparato experimental, por exemplo, discrepância entre os valores nominais e reais dos resistores e do capacitor, assim como, a imprecisão de aproximadamente dois bits para mais, ou para menos, do conversor analógico-digital do microcontrolador. A bateria de experimentos com resistência de $2\text{ k}\Omega$ não pôde ser utilizada para avaliar o aparato experimental por ter apresentado resultados considerados anômalos, os quais, são discutidos na próxima seção, Análise dos Resultados.

8 Análise dos Resultados

A análise sobre os resultados obtidos com a realização das baterias de experimentos possibilita extrair algumas informações acerca do comportamento da fotocélula e das virtudes presentes na concepção de aparato experimental adotada. A partir da análise dos valores de tensão no instante de tempo τ , exibidos na tabela 4, é possível identificar um comportamento anômalo da fotocélula. Para todos os ensaios realizados com resistência de $2\text{ k}\Omega$ no circuito RC, a fotocélula apresentou instabilidade na geração de tensão, consequentemente, a corrente também era instável, e demonstrou dificuldades para carregar o capacitor conforme esperado do ponto de vista teórico. A possível explicação para esse comportamento da fotocélula seria a sua incapacidade em prover carga suficiente para o capacitor, quando a corrente deveria assumir o seu maior valor no circuito, pois o fluxo da corrente comporta-se de forma contrária ao da tensão no circuito RC, de modo que, seu pico ocorre nos momentos iniciais do processo de carga do capacitor. Cabe ressaltar que, a constante de tempo τ é estabelecida para um provedor de tensão ideal, que consequentemente proverá uma corrente ideal por intermédio de um resistor constante, logo, quando esse provedor não é ideal, a curva de carga do capacitor está comprometida.

Embora os resultados em τ para resistência de $2\text{ k}\Omega$ tenham sido anômalos, ainda é possível analisar a sensibilidade da fotocélula em relação ao aumento da incidência luminosa, dada pela variação na distância entre a fonte de incidência e a fotocélula. Os valores apresentados na 8^a coluna da tabela 4 são os valores médio das tensões geradas (medida com multímetro) pela fotocélula nas alturas de 20 cm e 40 cm, para cada uma das tensões aplicadas sobre a lâmpada e, o mesmo ocorre com a tabela 5. A análise dos valores contidos nessas colunas indicam que, a fotocélula reage de maneira distinta à variação da incidência luminosa para diferentes valores de resistência. Dentro das condições experimentais adotadas, a célula fotovoltaica demonstrou ser mais sensível ao aumento da incidência luminosa para resistência de $2\text{ k}\Omega$ e as tensões geradas mediante a diminuição da distância entre fonte de incidência e fotocélula, foram aproximadamente duas vezes maiores. Por outra via, com resistência de $12\text{ k}\Omega$, o comportamento da célula fotovoltaica ao gerar tensão demonstrou ser menos sensível à variação da incidência luminosa. Esse comportamento evidencia que a fotocélula é mais sensível à variação na incidência luminosa quando as cargas são mais altas, logo, ao restringir a corrente por intermédio do aumento da resistência, a sensibilidade diminui significativamente. Ademais, é possível notar que o ponto de saturação da fotocélula foi atingido ao variar a tensão sobre a fonte luminosa de 8 V para 10 V, ou seja, a partir desse ponto, a fotocélula está bastante próxima do seu limite para gerar tensão, consequentemente, o aumento na tensão sobre a lâmpada não provoca um aumento significativo da tensão gerada pela célula, isso para ambas

cores de luz, branca e amarela. Cabe frisar que, esse comportamento ocorreu dadas as condições experimentais e pode não se repetir para cores distintas de luz. Pois a incidência de radiação de uma lâmpada, não necessariamente, em cada caso, será proporcional à tensão posta sobre ela, é possível que a variação na tensão de 10 V para 12 V em uma lâmpada azul represente uma coisa, e de 10 V para 12 V numa lâmpada amarela represente outra coisa, uma vez que, LEDs de cores diferentes reagem de maneira distinta para uma mesma fonte de alimentação.

Conforme demonstrado acima, a partir da análise dos resultados pode-se extrair algumas informações acerca do comportamento da fotocélula. Além disso, quando essa análise é realizada a fim de identificar possíveis virtudes da concepção de utilização de aparato experimental aberto e automatizado, torna-se necessário avaliar os aspectos técnicos mediante a quantidade de experimentos realizados e a precisão dos dados coletados. Portanto, cabe destacar que foram realizados três ensaios para cada arranjo montado com um dos valores possíveis de cada variável do experimento, visto que existem duas lâmpadas com temperaturas de cor diferente, quatro tensões para alimentação da lâmpada, duas alturas para incidência de radiação luminosa e duas resistências para o circuito RC, foram realizados 96 experimentos, para os quais, conforme demonstrado no capítulo anterior, a coleta de dados ocorreu dentro da precisão esperada. Com base nisso, pode-se afirmar acerca da concepção experimental utilizada que:

1. Foi realizado o registro fidedigno dos dados, com erros de leitura estimados conforme as características do microcontrolador, o que eliminou possíveis erros de leitura e transcrição caso fossem executadas manualmente.
2. A quantidade de amostras tomadas, para cada experimento, possibilitou realizar o traçado fiel da curva de carga do capacitor, viabilizando analisá-las a fim de verificar sua coerência, ou não, com uma fotocélula funcionando em um regime adequado.
3. A precisão dos intervalos em que as amostras foram tomadas, a cada 1 segundo, é impossível de se alcançar manualmente, mesmo que, os intervalos fossem mais largos.

Em virtude dos fatos descritos, torna-se evidente que a realização do experimento de forma automatizada apresenta resultados cuja precisão seria impossível alcançar quando comparado com a execução de forma manual.

9 Conclusão e Trabalhos Futuros

Este trabalho foi realizado com o objetivo de desenvolver e validar um ambiente aberto (software e hardware) para experimento didático em física usando o dispositivo microcontrolado Arduino e a plataforma computacional Raspberry Pi, sob a condição de que esse ambiente deve ser capaz de minimizar o esforço com o monitoramento dos experimentos e registro dos dados, automatizando-as. O processo de desenvolvimento desse ambiente aberto foi descrito em maiores detalhes no capítulo 6 e, após a conclusão desse ambiente, foram realizados experimentos de conversão fotovoltaica conforme demonstrado no capítulo 7. Com base na observação dos resultados pode-se concluir que o experimento mostra:

1. Pequenas diferenças no comportamento da fotocélula quando estimulada a gerar tensão com lâmpadas de diferentes temperaturas de cor.
2. Diferença significativa do comportamento da fotocélula na geração de energia com a variação da incidência luminosa dada pela distância da lâmpada.
3. A facilidade de obtenção de resultados múltiplos em um tempo muito curto, comparativamente a registro manuais e cálculos manuais.
4. Para resistências de $2\text{k}\Omega$ para baixo, dadas as condições experimentais adotadas, a célula fotovoltaica não é capaz de manter estabilidade na geração de tensão e, consequentemente, adultera o traçado da curva de carga do capacitor (procedimento experimental adotado) que leva em consideração uma fonte de tensão constante.

Desse modo, conclui-se que o objetivo deste trabalho foi alcançado e, do ponto de vista didático, fica demonstrado quantas nuances a utilização da concepção de aparato experimental aberto e automatizado, no processo de experimentação, permite revelar. Ademais, está demonstrado como prova de conceito o gerenciamento do experimento em ambiente amigável em computador de baixíssimo custo.

Após a conclusão deste trabalho, foi realizada uma avaliação dos aspectos técnicos do aparato experimental com vistas a identificar possíveis pontos de melhoria, assim como de novas funcionalidades capazes de expandir as possibilidades de uso da concepção de aparato experimental aberto e automatizado. A listagem abaixo é o resultado desse processo de avaliação, a qual, é sugerida como possíveis trabalhos futuros:

1. Análise e aperfeiçoamento da comunicação serial entre dispositivos de arquiteturas distintas.

2. Concentrar os cálculos sobre o front-end, uma vez que, o microprocessador ARM do Raspberry Pi consegue fazer cálculos envolvendo ponto flutuante de forma nativa.
3. Concentrar toda parametrização no front-end, pois isso pode aumentar a agilidade do processo experimental, uma vez que, elimina a necessidade de parametrizar diretamente no sistema SMCD e ter de transferi-lo para o arduino a cada novo experimento.
4. Aprimorar a gestão dos experimentos através da criação de um sistema de armazenamento a fim de criar uma memória experimental com possibilidade de análises comparativas entre os resultados experimentais.
5. Automatização do aparato físico com vistas a realização de experimentos através de acesso remoto ao aparato experimental.
6. Modificar o modo de comunicação serial através de conexão física para comunicação wireless, o que tornaria possível aprimorar o sistema gestor para gerenciar mais de um experimento ao mesmo tempo e tornaria a central de gestão independente do aparato experimental do ponto de vista físico.
7. Generalizar o sistema gestor para torná-lo um software capaz de gerenciar diversos experimentos.

Referências

- ALI, M. et al. Technical development and socioeconomic implications of the raspberry pi as a learning tool in developing countries. In: IEEE. *Computer Science and Electronic Engineering Conference (CEEC), 2013 5th.* [S.l.], 2013. p. 103–108. Citado na página 24.
- ALISSON, E. *Novos materiais realizam fotossíntese artificial.* 2014. Disponível em: <http://agencia.fapesp.br/novos_materiais_realizam_fotossintese_artificial/18685/>. Acesso em: 01 mar. 2017. Citado na página 17.
- ARDUINO. s.d. Disponível em: <<https://www.arduino.cc/en/Reference/AttachInterrupt>>. Acesso em: 08 oct. 2017. Citado na página 37.
- ARDUINO. *ARDUINO UNO REV 3.* s.d. Disponível em: <<https://store.arduino.cc/usa/arduino-uno-rev3>>. Acesso em: 07 oct. 2017. Citado na página 34.
- ARDUINO. *What is Arduino.* s.d. Disponível em: <<https://www.arduino.cc/en/Guide/Introduction>>. Acesso em: 06 oct. 2017. Citado na página 34.
- BRAGA, N. C. *Como funciona o capacitor (MEC070).* s.d. Disponível em: <<http://www.newtoncbraga.com.br/index.php/como-funciona/3177-mec070>>. Acesso em: 30 set. 2017. Citado na página 27.
- CRESCEB, C. de Referência para Energia Solar e Eólica Sérgio de S. B. *ENERGIA SOLAR PRINCÍPIOS E APLICAÇÕES.* 2007. Disponível em: <http://www.cresesb.cepel.br/download/tutorial/tutorial_solar_2006.pdf>. Acesso em: 29 oct. 2017. Citado na página 19.
- ENERGIA, P. *Principais tipos de células fotovoltaicas constituintes de painéis solares.* 2011. Disponível em: <<https://www.portal-energia.com/principais-tipos-de-celulas-fotovoltaicas-constituintes-de-paineis-solares/>>. Acesso em: 25 out. 2017. Citado na página 22.
- FLOYD, T. *Sistemas digitais: fundamentos e aplicações.* [S.l.]: Bookman Editora, 2009. Citado 4 vezes nas páginas 29, 30, 32 e 33.
- GALERIU, C.; LETSON, C.; ESPER, G. An arduino investigation of the rc circuit. *The Physics Teacher*, AAPT, v. 53, n. 5, p. 285–288, 2015. Citado na página 30.
- IBICT, I. B. de Informação em Ciência e T. *Energia HélioTérmica: como funciona?* s.d. Disponível em: <<http://energiaheliotermica.gov.br/pt-br/energia-heliotermica/como-funciona>>. Acesso em: 29 oct. 2017. Citado 2 vezes nas páginas 17 e 18.
- PINHO, J. T.; GALDINO, M. A. Manual de engenharia para sistemas fotovoltaicos. Rio de Janeiro: Cepel-Cresesb, 2014. Disponível em: <http://www.cresesb.cepel.br/publicacoes/download/Manual_de_Engenharia_FV_2014.pdf>. Acesso em: 20 set. 2016. Citado 2 vezes nas páginas 21 e 22.
- RAMALHO, F.; FERRARO, N. G.; SOARES, P. A. de T. *Os fundamentos da física.* 9. ed. São Paulo: Moderna, 2007. Citado na página 15.

- RECK, R. M.; SREENIVAS, R. S. Developing a new affordable dc motor laboratory kit for an existing undergraduate controls course. In: IEEE. *American Control Conference (ACC), 2015*. [S.I.], 2015. p. 2801–2806. Citado 2 vezes nas páginas 24 e 25.
- ROBOTFREAK. *Arduino 101: Timers and Interrupts*. 2014. Disponível em: [⟨http://www.robotshop.com/letsmakerobots/arduino-101-timers-and-interrupts⟩](http://www.robotshop.com/letsmakerobots/arduino-101-timers-and-interrupts). Acesso em: 08 oct. 2017. Citado na página 36.
- SACCO, F. *Níveis e limites de tensões digitais*. 2014. Disponível em: [⟨https://www.embarcados.com.br/niveis-e-limites-de-tensoes-digitais/⟩](https://www.embarcados.com.br/niveis-e-limites-de-tensoes-digitais/). Acesso em: 27 jul. 2017. Citado na página 28.
- SOUZA, F. *Arduino - Entradas Analógicas*. 2013. Disponível em: [⟨https://www.embarcados.com.br/arduino-entradas-analogicas/⟩](https://www.embarcados.com.br/arduino-entradas-analogicas/). Acesso em: 07 oct. 2017. Citado na página 36.
- SOUZA, F. *Arduino- Entradas e Saídas digitais*. 2013. Disponível em: [⟨https://www.embarcados.com.br/arduino-entradasaidas-digitais/⟩](https://www.embarcados.com.br/arduino-entradasaidas-digitais/). Acesso em: 07 oct. 2017. Citado na página 35.
- SOUZA, F. *Arduino UNO*. 2013. Disponível em: [⟨https://www.embarcados.com.br/arduino-uno/⟩](https://www.embarcados.com.br/arduino-uno/). Acesso em: 07 oct. 2017. Citado na página 34.
- SPARKFUN. *Placa Arduino Uno*. 2017. Disponível em: [⟨https://cdn.sparkfun.com/assets/parts/6/3/4/3/11021-04c.jpg⟩](https://cdn.sparkfun.com/assets/parts/6/3/4/3/11021-04c.jpg). Acesso em: 28 fev. 2017. Citado na página 35.
- THORNTON, R. K. Tools for scientific thinking-microcomputer-based laboratories for physics teaching. *Physics Education*, IOP Publishing, v. 22, n. 4, p. 230, 1987. Citado na página 15.
- WIKIPÉDIA. *Raspberry Pi*. 2017. Disponível em: [⟨https://pt.wikipedia.org/wiki/Raspberry_Pi#/media/File:Raspberry_Pi_3_Model_B.png⟩](https://pt.wikipedia.org/wiki/Raspberry_Pi#/media/File:Raspberry_Pi_3_Model_B.png). Acesso em: 09 set. 2017. Citado na página 24.
- YOUNG, H.; FREEDMAN, R. *Física III, IV–Sears e Zemansky, 12^a edição*. [S.I.]: São Paulo: Addison Wesley, 2009. Citado 2 vezes nas páginas 18 e 20.
- YOUNG, H.; FREEDMAN, R. *Física IV: Ótica e Física Moderna. 12^o edição*. São Paulo: Editora Pearson Education do Brasil., 2009. Citado 4 vezes nas páginas 19, 20, 30 e 31.

APÊNDICE A – Aparato Experimental

Figura 15 – Compilação de fotos do Aparato Experimental



Fonte: Autor.

APÊNDICE B – Rotinas do Sistema de Monitoramento e Coleta de Dados

```

1 #include <MsTimer2.h>
2
3 struct Data_experiment{
4     int second;
5     char separator = ';';
6     float volt;
7 };
8
9 const char END_TRANSMISSION = 'e';
10 const char START_EXPERIMENT = 's';
11 const char DURATION_EXPERIMENT = 't';
12 const float RESISTOR = 12e3;
13 const float CAPACITOR = 3300e-6;
14 const long TAU = 5*(RESISTOR*CAPACITOR);
15 const unsigned int RESET_PIN = 12;
16 Data_experiment seconds_and_volts[5];
17 int index = 0;
18 volatile int time_ = 0;
19 int previous_time = 0;
20 float volt = 0.0;
21 char signal_code = ' ';
22
23 /*
24  * Configura o intervalo de tempo em que os dados serao enviados
25  * e ativa o reset para impedir que a corrente chegue ate o
26  * capacitor antes do usuario iniciar o experimento.
27 */
28 void setup() {
29     pinMode( RESET_PIN, OUTPUT );
30     Serial.begin( 9600 );
31     MsTimer2::set( 1000, measurement_time );
32     digitalWrite( RESET_PIN, HIGH );
33 }
34
35 void loop() {
36     /*
37      * estrutura de decisao que compara o signal_code com o codigo
38      * para iniciar o experimento. caso sejam diferentes a funcao
39      * waitForRequest aguarda o SGE enviar o codigo e. Quando o
40      * codigo e DURANTION_EXPERIMENT, envia o tempo de duracao.

```

```

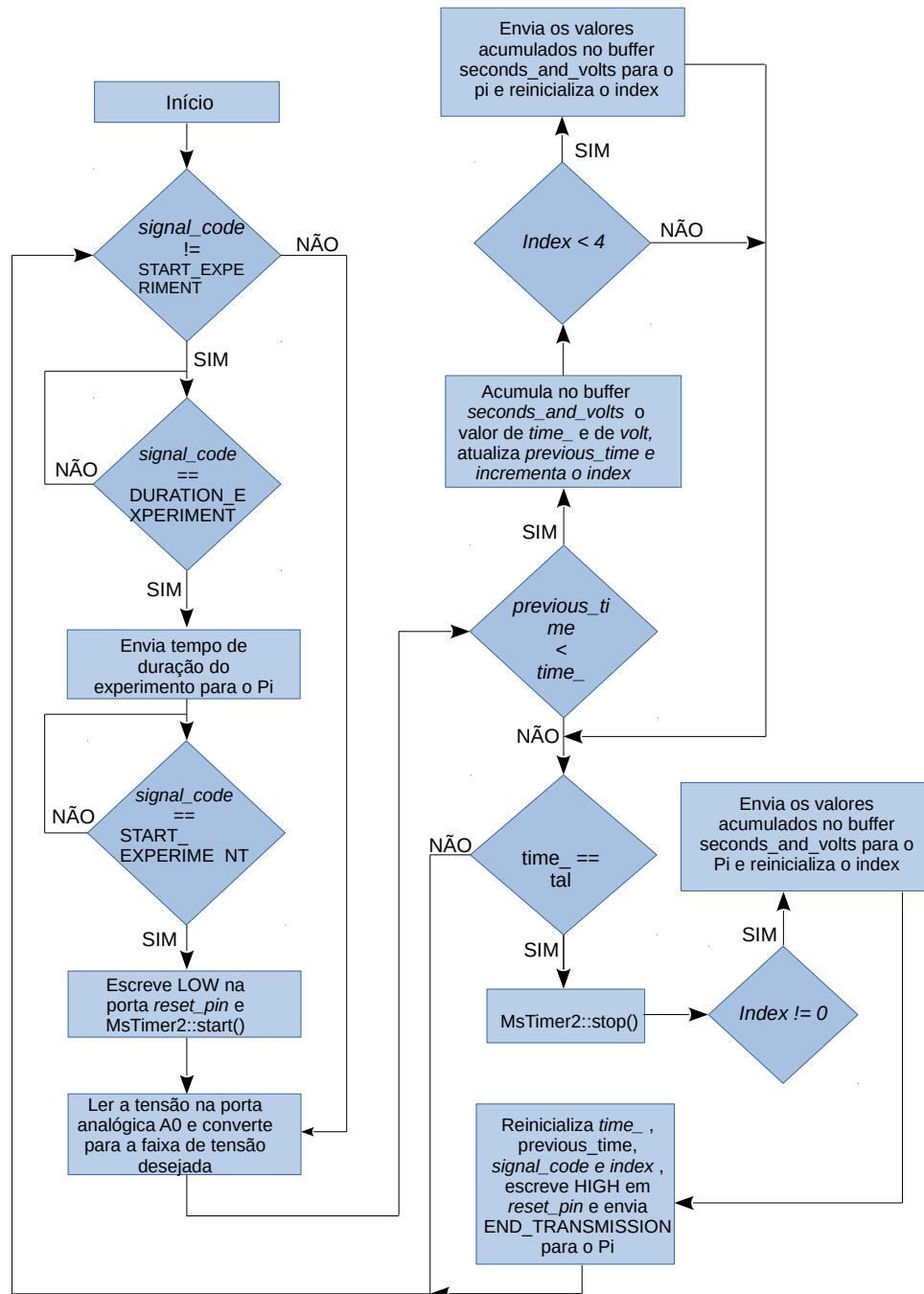
41     * Quando o codigo e START_EXPERIMENT, desliga o dreno e
42     * habilita a interrupcao
43     */
44     if( signal_code != START_EXPERIMENT){
45         waitForRequest(DURATION_EXPERIMENT) ;
46         Serial.print(TAU);
47         waitForRequest(START_EXPERIMENT) ;
48         digitalWrite( RESET_PIN, LOW ) ;
49         MsTimer2::start () ;
50     }
51     /*
52     * volt recebe o valor percebido com o polling feito na
53     * porta A0 e esse valor e convertido para a faixa de
54     * tensao pretendida
55     */
56     volt = analogRead( 0 );
57     volt = convertIntoVolts( volt );
58     /*
59     * estrutura de decisao que compara o tempo anterior com
60     * o tempo do experimento. caso tempo anterior seja menor
61     * os dados em time_ e volt sao enviados para o SGE
62     * e previous_time recebe o valor de time_
63     */
64     if( previous_time < time_){
65         seconds_and_volts [index].second = time_ ;
66         seconds_and_volts [index].volt = volt ;
67         if(index < 4){
68             index += 1;
69         }else{
70             for(int i = 0; i <= index; i++){
71                 Serial.print(seconds_and_volts [i].second);
72                 Serial.print(seconds_and_volts [i].separator);
73                 Serial.print(seconds_and_volts [i].volt);
74                 Serial.print('|');
75             }
76             index = 0;
77         }
78         previous_time = time_;
79     }
80     /*
81     * estrutura de decisao que compara o tempo do experimento
82     * com o tempo de duracao estimado. caso sejam iguais as
83     * variaveis time_ e signal_code recebem os valores iniciais
84     * o dreno e aberto e o sinal de encerramento e enviado p/
85     * o SMCD
86     */
87     if( time_ == TAU){
```

```

88     MsTimer2::stop();
89     if(index != 0){
90         for(int i = 0; i < index; i++){
91             Serial.print(seconds_and_volts[i].second);
92             Serial.print(seconds_and_volts[i].separator);
93             Serial.print(seconds_and_volts[i].volt);
94             Serial.print('|');
95         }
96     }
97     time_ = 0;
98     previous_time = 0;
99     signal_code = ' ';
100    digitalWrite( RESET_PIN, HIGH );
101    //delay(500);
102    Serial.print(END_TRANSMISSION);
103    index = 0;
104 }
105
106 }
107 /*
108  * Sub-rotina que converte o valor lido na porta A0 em
109  * outro valor dentro da faixa de 0 a 5
110  */
111 float convertIntoVolts( float valor ){
112     return (valor*3.88)/1023;
113 }
114 /*
115  * Rotina de interrupcao que realiza a contagem do tempo
116  * de execucao do experimento
117  */
118 void measurement_time() {
119     time_ += 1;
120 }
121 /*
122  * sub-rotina que recebe o codigo de sinal e verifica
123  * se ele esta de acordo com o esperado
124  */
125 void waitForRequest(char code){
126     while( signal_code != code ){
127         if( Serial.available() > 0 ){
128             signal_code = Serial.read();
129         }
130     }
131 }
```

APÊNDICE C – Fluxograma do SMCD

Figura 16 – Fluxo de execução do SMCD.



Fonte: Autor.

APÊNDICE D – Dados dos experimentos com resistor de $2\text{k}\Omega$

Tabela 4 – Média dos valores da bateria de experimentos com $2\text{k}\Omega$

Resistência (Ω)	Capacitância (μF)	Constante de tempo τ (s)	Intervalo entre os experi- mentos (s)	Temperatura de cor (K)	Tensão lâmpada (V)	Distância da fonte luminosa (cm)	Média da tensão da fotocélula (V)	Média da tensão do capacitor em τ (V) conforme teoria	Média da tensão do capacitor em τ (V)	Média do percentual da diferença entre as tensões do capacitor em τ	Média da razão entre as quantidades de energia armazenadas no capacitor em τ			
2.000	3.300	6,6	60	3.000	6.00	20	2,27	1,43	1,20	-15,87%	3,956			
						40	1,11	0,70	0,61	-13,50%				
					8.00	20	2,64	1,66	1,45	-12,62%	4,620			
						40	1,26	0,80	0,68	-14,99%				
				6.500	10.00	20	2,79	1,76	1,57	-10,79%	5,028			
						40	1,27	0,80	0,70	-12,09%				
					12.00	20	2,77	1,75	1,56	-10,42%	5,080			
						40	1,27	0,80	0,70	-12,93%				
	3.300				6.00	20	2,50	1,58	1,39	-12,07%	6,167			
						40	1,03	0,65	0,56	-13,74%				
	6.500			8.00	20	2,94	1,85	1,69	-8,58%	5,116				
					40	1,34	0,85	0,80	-6,88%					
				10.00	20	2,96	1,87	1,71	-8,40%	5,826				
					40	1,30	0,82	0,73	-11,32%					
				12.00	20	2,95	1,86	1,71	-7,99%	5,240				
					40	1,31	0,83	0,78	-6,14%					

APÊNDICE E – Dados dos experimentos com resistor de $12\text{k}\Omega$

Tabela 5 – Média dos valores da bateria de experimentos com $12\text{k}\Omega$

Resistência (Ω)	Capacitância (μF)	Constante de tempo τ (s)	Intervalo entre os experi- mentos (s)	Temperatura de cor (K)	Tensão lâmpada (V)	Distância da fonte luminosa (cm)	Média da tensão fotocélula (V)	Média da tensão do capacitor em τ (V) conforme teoria	Média da tensão do capacitor em τ (V)	Média do percentual da diferença entre as tensões do capacitor em τ	Média da razão entre as quantidades de energia armazenadas no capacitor em τ			
12.000	3.300	39,6	60	3.000	6.00	20	4,13	2,60	2,63	1,03%	1,946			
						40	2,95	1,86	1,85	-0,75%				
					8.00	20	4,25	2,68	2,70	0,76%	1,747			
						40	3,21	2,02	2,03	0,11%				
					10.00	20	4,25	2,68	2,69	0,26%	1,763			
						40	3,20	2,02	2,00	-0,40%				
					12.00	20	4,26	2,68	2,71	0,98%	1,750			
						40	3,22	2,03	2,02	-0,26%				
	6.500				6.00	20	4,13	2,60	2,62	0,82%	2,193			
						40	2,79	1,76	1,73	-1,46%				
					8.00	20	4,25	2,68	2,70	0,92%	1,825			
						40	3,14	1,98	1,97	-0,52%				
					10.00	20	4,24	2,67	2,70	1,00%	1,811			
						40	3,15	1,99	1,96	-1,17%				
					12.00	20	4,25	2,68	2,70	0,64%	1,745			
						40	3,22	2,03	2,07	1,90%				

APÊNDICE F – Bateria de Experimentos com Resistor de $2k\Omega$

Tabela 6 – Conjunto de ensaios da primeira bateria de experimentos

Resistência (Ω)	Capacitância (μF)	Constante de tempo τ (s)	Intervalo entre os experi- mentos (s)	Temperatura de cor (K)	Tensão lâmpada (V)	Distância da fonte luminosa (cm)	Tensão fotocélula (V)	Tensão do capacitor em τ (V)conforme teoria	Tensão do capacitor em τ (V)	Percentual da diferença entre as tensões do capacitor em τ	Razão entre as quantidades de energia armazenadas no capacitor em τ
2.000	3.300	6,6	60	3.000	6.00	20	2,32	1,46	1,25	-14,48%	4,340
						40	1,12	0,71	0,60	-14,97%	
						20	2,20	1,39	1,16	-16,82%	3,390
						40	1,11	0,70	0,63	-9,91%	
						20	2,29	1,44	1,20	-16,82%	4,137
						40	1,11	0,70	0,59	-15,63%	
					8.00	20	2,65	1,67	1,46	-12,55%	4,350
						40	1,27	0,80	0,70	-12,51%	
						20	2,64	1,66	1,45	-12,82%	4,684
						40	1,26	0,79	0,67	-15,60%	
						20	2,63	1,66	1,45	-12,49%	4,827
						40	1,26	0,79	0,66	-16,86%	
					10.00	20	2,89	1,82	1,63	-10,47%	5,919
						40	1,27	0,80	0,67	-16,26%	
						20	2,76	1,74	1,55	-10,86%	4,387
						40	1,27	0,80	0,74	-7,51%	

2.000	3.300	6,6	60	3.000	6.500	10.00	20	2,73	1,72	1,53	-11,04%	4,777
							40	1,27	0,80	0,70	-12,51%	
							20	2,76	1,74	1,56	-10,28%	4,326
							40	1,27	0,80	0,75	-6,26%	
							20	2,77	1,75	1,56	-10,61%	5,421
							40	1,27	0,80	0,67	-16,26%	
							20	2,78	1,75	1,57	-10,36%	5,491
							40	1,20	0,80	0,67	-16,26%	
							20	2,50	1,58	1,39	-11,75%	6,878
							40	1,00	0,63	0,53	-15,87%	
							20	2,51	1,58	1,39	-12,10%	5,550
							40	1,06	0,67	0,59	-11,65%	
							20	2,50	1,58	1,38	-12,38%	6,073
							40	1,03	0,65	0,56	-13,70%	
							20	2,93	1,85	1,69	-8,45%	6,557
							40	1,23	0,77	0,66	-14,83%	
							20	2,99	1,88	1,72	-8,69%	6,214
							40	1,23	0,77	0,69	-10,96%	
							20	2,90	1,83	1,67	-8,59%	2,578
							40	1,57	0,99	1,04	5,15%	
							20	2,96	1,86	1,70	-8,84%	3,732
							40	1,45	0,91	0,88	-3,67%	
							20	2,99	1,88	1,73	-8,16%	7,307
							40	1,21	0,76	0,64	-16,04%	
							20	2,94	1,85	1,70	-822%	6,438
							40	1,24	0,78	0,67	-14,23%	
							20	2,97	1,87	1,72	-8,08%	2,958
							40	1,45	0,91	1,00	9,47%	
							20	2,94	1,85	1,71	-7,68%	6,324
							40	1,24	0,78	0,68	-12,95%	
							20	2,94	1,85	1,70	-8,22%	6,438
							40	1,25	0,79	0,67	-14,92%	

Fonte: Autor.

APÊNDICE G – Bateria de Experimentos com Resistor de $12\text{k}\Omega$

Tabela 7 – Conjunto de ensaios da segunda bateria de experimentos

Resistência (Ω)	Capacitância (μF)	Constante de tempo τ (s)	Intervalo entre os experi- mentos (s)	Temperatura de cor (K)	Tensão lâmpada (V)	Distância da fonte luminosa (cm)	Tensão fotocélula (V)	Tensão do capacitor em τ (V)conforme teoria	Tensão do capacitor em τ (V)	Percentual da diferença entre as tensões do capacitor em τ	Razão entre as quantidades de energia armazenadas no capacitor em τ
12.000	3.300	39,6	60	3.000	6.00	20	4,14	2,61	2,66	1,99%	1,930
						40	2,98	1,88	1,87	-0,39%	
						20	4,13	2,60	2,61	0,31%	1,941
						40	2,95	1,86	1,84	-1,00%	
						20	4,11	2,59	2,61	0,80%	1,968
						40	2,93	1,85	1,83	-0,86%	
					8.00	20	4,26	2,68	2,70	0,60%	1,761
						40	3,21	2,02	2,01	-0,61%	
						20	4,26	2,68	2,71	0,98%	1,713
						40	3,24	2,04	2,06	0,92%	
						20	4,24	2,67	2,69	0,70%	1,767
						40	3,19	2,01	2,01	0,01%	
					10.00	20	4,26	2,68	2,65	-1,026%	1,772
						40	3,20	2,02	2,01	-0,30%	
						20	4,25	2,68	2,70	0,84%	1,764
						40	3,20	2,02	2,00	-0,79%	

12.000	3.300	39,6	60	3.000	6.500	10.00	20	4,25	2,68	2,71	1,21%	1,753
							40	3,21	2,02	2,02	-0,11%	
							20	4,26	2,68	2,71	0,98%	1,750
							40	3,22	2,03	2,01	-0,92%	
							20	4,26	2,68	2,01	0,98%	1,750
							40	3,22	2,3	2,02	-0,42%	
							20	4,26	2,68	2,71	0,98%	1,750
							40	3,22	2,03	2,04	0,56%	
							20	4,16	2,62	2,64	0,73%	2,192
							40	2,81	1,77	1,75	-1,15%	
							20	4,12	2,60	2,62	0,94%	2,155
							40	2,80	1,76	1,74	-1,36%	
							20	4,11	2,59	2,61	0,80%	2,234
							40	2,75	1,73	1,70	-1,88%	
							20	4,26	2,68	2,71	0,98%	1,829
							40	3,15	1,98	1,97	-0,73%	
							20	4,24	2,67	2,69	0,70%	1,823
							40	3,14	1,98	1,98	0,09%	
							20	4,24	2,67	2,70	-1,08%	1,823
							40	3,14	1,98	1,96	-0,92%	
							20	4,25	2,68	2,70	0,84%	1,809
							40	3,16	1,99	1,97	-1,04%	
							20	4,24	2,67	2,70	1,08%	1,812
							40	3,15	1,98	1,96	-1,23%	
							20	4,24	2,67	2,70	1,08%	1,812
							40	3,15	1,98	1,96	-1,23%	
							20	4,25	2,68	2,70	0,84%	1,679
							40	3,28	2,07	2,20	6,47%	
							20	4,25	2,68	2,70	0,84%	1,772
							40	3,20	2,02	2,00	-0,79%	
							20	4,26	2,68	2,69	0,23%	1,783
							40	3,19	2,01	2,01	0,01%	

Fonte: Autor.