



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO  
AMAZONAS  
CAMPUS MANAUS DISTRITO INDUSTRIAL  
ENGENHARIA DE CONTROLE E AUTOMAÇÃO**

**BARTOLOMEU HEBERT PEREIRA DE SÁ BARBOSA**

**COMPUTADOR DE BORDO PARA BICICLETA ALIMENTADO COM  
ALTERNADOR**

**MANAUS - AM**

**2019**

**BARTOLOMEU HEBERT PEREIRA DE SÁ BARBOSA**

**COMPUTADOR DE BORDO PARA BICICLETA ALIMENTADO COM  
ALTERNADOR**

Monografia apresentada ao programa de Graduação em 2019 do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Manaus Distrito Industrial – IFAM/CMDI, para obtenção do título de Bacharel em Engenharia de Controle e Automação.

Orientador: Prof. Me. Gilberto Andrade da Silva.

**MANAUS - AM  
2019**

## Dados Internacionais de Catalogação na Publicação (CIP)

---

B238c Barbosa, Bartolomeu Hebert Pereira de Sá.

Computador de bordo para bicicleta alimentado com alternador. /  
Bartolomeu Hebert Pereira de Sá Barbosa. – Manaus, 2019.  
72 f. : il. color

Monografia (Graduação em Engenharia de Controle e Automação) –  
Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, *Campus*  
Manaus Distrito Industrial, 2019.

Orientador: Prof. Me. Gilberto Andrade da Silva

1. Alternador. 2. Computador de bordo. 3. Bicicleta. 4. Sensor HALL.  
5. ESP32 I. Silva, Gilberto Andrade da (Orient.) II. Instituto Federal de  
Educação, Ciência e Tecnologia do Amazonas. III. Título.

CDD 629.89

---

Elabora por Fc<sup>a</sup>. Amélia Frota, registro n.858 (CRB11)

**BARTOLOMEU HEBERT PEREIRA DE SÁ BARBOSA**

**COMPUTADOR DE BORDO PARA BICICLETA ALIMENTADO  
COM ALTERNADOR**

Trabalho de Conclusão de Curso (TCC) apresentado ao curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Manaus Distrito Industrial (IFAM/CMDI), como requisito para obtenção do Título de Bacharel em Engenharia de Controle e Automação.

Aprovado em 29 de Novembro de 2019.

**BANCA EXAMINADORA**



---

**Prof. Me. Gilberto Andrade da Silva**  
Orientador



---

**Prof. Me. Ricardo Brandão Sampaio**  
Professor Avaliador



---

**Prof. Me. Marlos André Silva Rodrigues**  
Professor Avaliador

*Dedico este trabalho...*

*Aos meus pais, Adão e Dalila, e aos meus irmãos, Mateus e Clawbert, pelo incentivo e apoio constante ao longo dessa jornada.*

# Agradecimentos

Agradeço em primeiro lugar à Força Suprema, por ter me guiado e protegido durante esta jornada ao conhecimento.

À minha família que em todos os momentos esteve presente prestando apoio a todas as atividades propostas e desenvolvidas por mim.

Ao Instituto Federal de Educação, Ciência e Tecnologia do Amazonas (IFAM), pela oportunidade a mim concedida para a obtenção de conhecimento.

Aos professores do IFAM que de alguma forma contribuíram na construção do meu conhecimento, em especial, ao Prof. Me. Hillerman Osmídio Ferreira Lima pelas excelentes aulas práticas e projetos de automação em que trabalhamos juntos; ao Prof. Me. Adriano Bruno dos Santos Frutuoso por todo o seu conhecimento em Controle de Sistemas compartilhado de forma assídua e prática; e ao Prof. Me. Gilberto Andrade da Silva por ter dedicado seu tempo para orientar-me neste trabalho.

À comunidade pelo pagamento de impostos destinados à União que possibilitaram os investimentos em materiais, pesquisas científicas e o sustento da instituição.

BARBOSA, Bartolomeu Hebert Pereira de Sá. **Computador de bordo para bicicleta alimentado com Alternador**: 2019. Trabalho de Conclusão de Curso (Engenharia de Controle e Automação) – Instituto Federal de Educação, Ciência e Tecnologia do Amazonas - CMDI, Manaus, 2019.

## Resumo

Este trabalho apresenta uma proposta de um computador de bordo embutido em uma bicicleta. O sistema embarcado é responsável por medir os seguintes parâmetros da bicicleta: velocidade, quilometragem total e gasto calórico. O projeto do computador de bordo é disposto de microprocessador ESP32, sensor de efeito HALL e aplicativo instalado no celular. Além disso, será implantado um ímã no aro do pneu para que seja efetuada a leitura do sensor magnético. O computador de bordo disporá, ainda, de um sistema capaz de realimentar a bateria, sendo composto de alternador e um circuito controlador de corrente e tensão que fará o recarregamento da bateria. Dessa forma, quando o usuário iniciar uma corrida de bicicleta, será capaz de visualizar no aplicativo instalado no celular as variáveis velocidade e a quilometragem total, além do gasto calórico, dispondo de um sistema com bateria recarregável durante suas atividades sobre a bicicleta.

**Palavras-chave:** Alternador. Computador de Bordo. Bicicleta. Sensor HALL. ESP32.

BARBOSA, Bartolomeu Hebert Pereira de Sá. **Bicycle on-board computer powered by alternator**: 2019. Trabalho de Conclusão de Curso (Engenharia de Controle e Automação) – Instituto Federal de Educação, Ciência e Tecnologia do Amazonas - CMDI, Manaus, 2019.

## **ABSTRACT**

This paper presents a proposal for an onboard computer embedded in a bicycle. The embedded system is responsible for measuring the following bicycle parameters: speed, total mileage and caloric expenditure. The design of the onboard computer is disposed of ESP32 microprocessor, HALL effect sensor and application installed on the mobile phone. In addition, a magnet will be implanted in the rim of the tire for the magnetic sensor to be read. The on-board computer will also have a system capable of recharging the battery, being composed of dynamo and a voltage and current controller circuit that will recharge the battery. In this way, when the user starts a bicycle race, he / she will be able to see in the application installed in the cell phone the speed variables and the total mileage, besides the caloric expenditure, having a system with rechargeable battery during his activities on the bicycle.

**Keywords:** Alternator. Board computer. Bicycle. HALL sensor. ESP32.

## Lista de Ilustrações

Figura 1 – Placa NodeMCU (ESP32).....	18
Figura 2 – Coexistência dos canais BLE e Wi-Fi.....	20
Figura 3 – Topologia Star-bus BLE.....	20
Figura 4 – Composição Sensor Reed-Switch.....	22
Figura 5 - Princípio de funcionamento do Reed-Switch.....	23
Figura 6 – Módulo TP4056.....	24
Figura 7 – Curva de carga da Bateria 18650.....	25
Figura 8 – Célula de carga 18650.....	26
Figura 9 – Ponte de Graetz.....	27
Figura 10 – Circuito durante o semiciclo positivo.....	28
Figura 11 – Circuito durante o semiciclo negativo.....	28
Figura 12 – Módulo Step Down LM2596.....	30
Figura 13 – Estrutura básica da bicicleta.....	31
Figura 14 – Mecanismos do Alternador.....	32
Figura 15 – Composição do Alternador de Bicicleta.....	33
Figura 16 – Elementos do projeto.....	35
Figura 17 – Fluxograma da estrutura do projeto.....	36
Figura 18 – Comandos para adição do Flutter através do GitHub.....	39
Figura 19 – Fluxograma do Processador.....	41
Figura 20 – Variáveis do sistema.....	42
Figura 21 – Interrupções externas e internas.....	43
Figura 22 – Loop do Núcleo 2 do processador.....	43
Figura 23 – Tarefa da interrupção do sensor Hall da roda.....	44
Figura 24 – Sinal de entrada de uma interrupção no osciloscópio.....	45
Figura 25 – Tarefa de Interrupção Sensor de Efeito Hall do Pedal.....	46
Figura 26 – Incremento do tempo a cada intervalo de velocidade.....	47
Figura 27 – Incremento do Gasto Calórico.....	47
Figura 28 – Layout do aplicativo.....	49
Figura 29 – Notificação de alerta BLE.....	50
Figura 30 – Busca por dispositivo escravo.....	50
Figura 31 – Monitoramento de Gasto Calórico.....	51

Figura 32 - Animação da Bateria.....	51
Figura 33 – Função Map.....	52
Figura 34 - Conversores AC-DC e DC-DC.....	53
Figura 35 – Circuito de recarga da bateria de Lítio.....	54
Figura 36 – Circuito Completo.....	55
Figura 37 – Montagem do circuito.....	57
Figura 38 – Configurações da PCB.....	58
Figura 39 – Comando Begin-Routing.....	58
Figura 40 – Simulação Circuito Impresso ARES.....	59
Figura 41 – Protótipo 3D da PCI.....	59
Figura 42 – Suporte para o celular no inventor.....	60
Figura 43 - Desenho no inventor do Sensor Reed Switch.....	61
Figura 44 – Desenho no inventor Suporte para o botão.....	61
Figura 45 – Desenho da caixa para PCI no inventor.....	62
Figura 46 – Visão superior da Placa de Controle.....	67
Figura 47 - Visão inferior da Placa de Controle.....	68
Figura 48 – Bateria de Lítio.....	68
Figura 49 - Alternador.....	69
Figura 50 – Sensor de efeito Hall paralelo ao Pedal.....	69
Figura 51 – Sensor de Efeito Hall paralelo ao Pneu.....	70
Figura 52 – Suporte de apoio para o celular.....	70
Figura 53 – Suporte para o botão.....	71
Figura 54 – Caixa para PCI.....	71
Figura 55 – Computador de bordo anexo à bicicleta.....	72

## Lista de Tabelas

Tabela 1 – Características ESP32.....	18
Tabela 2 – Conceitos GATT.....	21
Tabela 3 – Especificações do Reed Switch.....	23
Tabela 4 - Característica da Bateria de Lítio 18650.....	26
Tabela 5 – Operação dos diodos durante o semiciclo positivo.....	28
Tabela 6 - Operação dos diodos durante o semiciclo negativo.....	29
Tabela 7 – Características de operação LM2596.....	29
Tabela 8 – Peças da estrutura da bicicleta.....	31
Tabela 9 – Quantidade de calorias gasta por atividades.....	34
Tabela 10 – Cronograma de Atividades.....	37
Tabela 11 – Protocolo de comunicação ESP32-Applicativo.....	50
Tabela 12 – Lista de Componentes eletrônicos.....	56

# Lista de Abreviaturas e Siglas

APP – Aplicativo Móvel

GATT – *General Agreement on Tariffs and Trade*

IDE – *Integrated Development Environment*

PCB – *Printed Circuit Board*

PCI – Placa de Circuito Impresso

SDK – *Software Development Kit*

SIG – *Special Interest Group*

UART – *Universal Asynchronous Receiver/Transmitter*

USB – *Universal Serial Bus*

MET – *Metabolic Equivalent Task*

IOS – *Apple Operating System*

IFAM – Instituto Federal do Amazonas

AC – Alternative Current

DC – Direct Current

IR – Receptor Infravermelho

IoT – *Internet of Things*

CAN – *Controller Area Network*

DAC – *Digital-to-Analog Converter*

BLE – *Bluetooth Low Energy*

UUID – *Universally Unique Identifier*

SPI – *Serial Peripheral Interface*

I2C – *Inter-Integrated Circuit*

SDIO – *Secure Digital Input Output*

GPIO – *General Purpose Input/Output*

PWM – *Pulse Width Modulation*

RAM – *Random Access Memory*

ROM – *Read-Only Memory*

# Sumário

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>14</b>
1.1	<b>OBJETIVOS.....</b>	15
1.1.1	Objetivo Geral.....	15
1.1.2	Objetivos Específicos.....	15
1.2	<b>JUSTIFICATIVA.....</b>	16
<b>2</b>	<b>REVISÃO DA LITERATURA.....</b>	<b>17</b>
2.1	<b>FLUTTER E DART.....</b>	17
2.2	<b>NODEMCU ESP32.....</b>	17
2.2.1	Bluetooth Low Energy.....	19
2.3	<b>SENSORES.....</b>	22
2.3.1	Chave Magnética de efeito Hall (Reed Switch).....	22
2.4	<b>MÓDULO CARREGADOR TP4056.....</b>	23
2.4.1	Bateria de Lítio (LiCoO2) 18650.....	25
2.5	<b>CONVERSOR AC-DC.....</b>	27
2.5.1	Ponte de Graetz (Graetz Bridge).....	27
2.6	<b>CONVERSOR DC-DC.....</b>	29
2.6.1	Step Down Lm2596.....	29
2.7	<b>BICICLETA.....</b>	30
2.8	<b>GERADORES ELÉTRICOS E ALTERNADORES.....</b>	32
2.9	<b>GASTO CALÓRICO NA PRÁTICA DE EXERCÍCIOS FÍSICOS.....</b>	33
<b>3</b>	<b>PLANEJAMENTO DO SISTEMA.....</b>	<b>34</b>
3.1	<b>VISÃO GERAL.....</b>	34
3.2	<b>METODOLOGIA.....</b>	37
<b>4</b>	<b>DESENVOLVIMENTO DO PROJETO.....</b>	<b>38</b>
4.1	<b>AMBIENTE DE DESENVOLVIMENTO.....</b>	38
4.1.1	Ambiente de Desenvolvimento do Firmware.....	38
4.1.2	Ambiente de Desenvolvimento mecânico.....	39
4.1.3	Ambiente de Desenvolvimento do Aplicativo.....	39
4.1.4	Ambiente de Desenvolvimento Elétrico.....	39
4.2	<b>DESCRIÇÃO DO PROJETO DE FIRMWARE.....</b>	40
4.2.1	Processador Esp32.....	40
4.2.2	Tarefa de Interrupção Sensor de Efeito Hall 1.....	44
4.2.3	Tarefa de Interrupção Sensor de Efeito Hall 2 .....	46

4.2.4	Tarefa de Interrupção do Timer.....	46
4.2.5	Aplicativo Android.....	48
<b>4.3</b>	<b>DESCRIÇÃO DO PROJETO ELÉTRICO.....</b>	<b>52</b>
4.3.1	Funcionamento do Circuito.....	52
4.3.2	Montagem Prática.....	55
4.3.3	Confecção da Placa de Circuito Impresso (PCI).....	57
<b>4.4</b>	<b>DESCRIÇÃO DO PROJETO MECÂNICO.....</b>	<b>60</b>
4.4.1	Suporte para o celular.....	60
4.4.2	Suporte do Sensor Reed Switch.....	60
4.4.3	Suporte para o botão.....	61
4.4.4	Caixa da PCI.....	61
<b>5</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>63</b>
	<b>REFERÊNCIAS.....</b>	<b>64</b>
	<b>APÊNDICE A – Elementos elétricos do computador de bordo.....</b>	<b>67</b>
	<b>APÊNDICE B – Peças mecânicas.....</b>	<b>70</b>

# 1 Introdução

Devido à explosão em popularidade do ciclismo no Brasil, a bicicleta passou a ser utilizada como meio transporte para atividades diárias e diversão em grande escala, visto que, proporciona um transporte sem altos custos financeiros ao usuário, sem emissão de poluentes no meio ambiente e com relativa segurança.

Diante desses benefícios, a bicicleta passou a ser instrumento de política de conscientização e mobilização de diversas esferas ligadas à questão socioambiental, de forma a instigar o maior uso deste meio de transporte em substituição aos transportes poluentes movidos à combustão.

O incentivo à mobilidade por bicicleta pode trazer benefícios para os usuários e para o meio ambiente urbano. Para tornar esta afirmativa uma prática corrente é preciso enfrentar as dificuldades estruturais e buscar a mudança de comportamento. É possível promover mudanças, desde que haja vontade política, planejamento, distribuição equitativa dos espaços de circulação e educação para o trânsito (MINISTÉRIO DAS CIDADES, 2007b).

Segundo levantamento feito em 2017 pela da ABRACICLO – Associação Brasileira de Fabricantes de Motocicletas, Ciclomotores, Motonetas, Bicicletas e Similares, o Brasil possui uma frota de mais de 70 milhões de bicicletas. Anualmente produz cerca de 2,5 milhões de unidades, sendo contemplado como o 4º maior produtor mundial de bicicletas. No entanto, figura em um cenário onde vem apresentando queda nas vendas em decorrência do aumento do uso de veículos motorizados à combustão.

Visando esse mercado e buscando-se contribuir para uma maior adesão do seu uso, propôs-se agregar tecnologia às bicicletas, utilizando meios eletrônicos de baixo custo e aproveitando a popularidade do celular para utilizá-lo como interface ao usuário no projeto. Desse modo, surge o projeto do computador de bordo para bicicleta.

---

## 1.1 OBJETIVOS

### 1.1.1 Objetivo Geral

Desenvolver um sistema embarcado para automação da bicicleta.

### 1.1.2 Objetivos Específicos

Para desenvolvimento do projeto, propôs-se executar as seguintes etapas:

- Desenvolver aplicativo Android para apresentação dos parâmetros da bicicleta: velocidade, quilometragem total, Gasto Calórico e Carga da Bateria.
- Desenvolver circuito eletrônico de conversão de corrente alternada do alternador para corrente contínua e seu posterior carregamento da bateria.
- Desenvolver *firmware* gravável na placa ESP32, da *Expressif Systems*, através da IDE do Arduino.
- Desenvolver peças através do *software Autodesk Inventor* e posterior impressão em impressora 3D, para fixação de sensores na bicicleta.
- Realizar testes de simulação no *software Proteus Professional*.
- Projetar o circuito elétrico na Placa de Circuito Impresso.

## 1.2 JUSTIFICATIVA

O computador de bordo, dotado de funcionalidades como medição de velocidade, medição de quilometragem e aferição de gastos calóricos, surge como um atrativo para usuários que utilizam a bicicleta como forma de diversão ou para fins de condicionamento físico, visto que, através da interface do aplicativo o ciclista faz o controle de suas atividades, além de propiciar um maior entretenimento.

Outro fator atrativo é o sistema de carregamento de bateria proporcionado pelo alternador junto ao circuito de tratamento e armazenamento de energia. Essa aplicabilidade do computador de bordo assegura ao usuário o privilégio de não necessitar recarregar a bateria através de matrizes energéticas externas, além de promover uma política de conscientização ambiental ao se aproveitar a energia cinética da bicicleta para gerar a própria energia elétrica consumida pelo sistema embarcado. Dessa forma, dispensa-se o carregamento externo vindo de matrizes energéticas como usinas hidrelétricas ou termelétricas, assim como, despreza-se a troca de bateria por uma nova ao fim de cada descarga completa.

Dentre outros fatores benéficos que podem ser destacados sobre os veículos elétricos são a inexistência de emissões de gases de combustão, um melhor aproveitamento da matriz energética brasileira (maior parte hidrelétrica), no que diz respeito aos horários fora de pico do sistema, que poderiam ser liberados para recarga de frotas desses veículos elétricos, maximizando o uso do sistema(SILVA;MURADÁS;BARBOSA, 2007).

O implemento do computador de bordo à bicicleta passa a ser um atrativo para usuários de outros meios de transporte, como ônibus ou carro, visto que, a bicicleta computadorizada torna-se um transporte seguro, de baixo custo, e contempla a necessidade dos usuários em monitorar variantes comuns ao dia-a-dia, como, distância, tempo e velocidade. Dessa forma, este projeto passa a ser um vetor de incentivo à mudança de meio de transporte de usuários, contribuindo na causa socioambiental.

A bicicleta é o meio de transporte que apresenta menor consumo de energia primária e é ideal para deslocamentos urbanos de curtas distâncias. Seus benefícios são consideráveis tanto para comunidade urbana quanto para seus usuários (SILVEIRA, 2010).

---

## 2 REVISÃO DA LITERATURA

### 2.1 FLUTTER E DART

O ambiente FLUTTER é um SDK de código aberto para o desenvolvimento de aplicativos para dispositivos móveis, *Desktop* e *WEB* criado pelo Google. Possui um conjunto de ferramentas para desenvolvimento *Android* e *IOS*, onde se apropria de *Widgets* prontos, que constituem os componentes da versão do aplicativo.

A programação com o FLUTTER é realizada através da linguagem de programação *Dart*, uma linguagem de *Script* voltada à *WEB*, tendo sua primeira versão desenvolvida pelo Google em 2011.

O objetivo do *Dart* foi, inicialmente, substituir o *JavaScript* como linguagem principal embutida nos Navegadores. Com o decorrer dos anos, passou a ser bastante utilizado no desenvolvimento de *APPs mobile*.

Recentemente, em 2018, foi lançado o *Dart 2.0* otimizado para o desenvolvimento *WEB* e dispositivos móveis.

### 2.2 NODEMCU ESP32

Segundo a fabricante *Expressif Systems* (2018), o *NodeMCU ESP32* é um dispositivo embarcado de alta performance composto de um *chip* capaz de executar aplicações que envolvam *Wi-Fi*, *Bluetooth Serial* e *Bluetooth Low Energy* (BLE).

O ESP32 possui muitos recursos que o torna ideal para a IoT (*Internet of the Things*), dado que a presença de mais periféricos permite a sua integração com mais dispositivos. Dentre as interfaces de comunicação, ele possui suporte a SPI, UART e I2C (protocolos relativamente comuns), como também tem suporte a Infravermelho (IR) e SDIO (para interface com cartão de memória), e como diferenciação recente, CAN, Ethernet, DAC, Sensor de Toque, e I2S, que é uma interface de comunicação útil para comunicar com dispositivos de áudio. Possui também hardware para aceleração de criptografia embutido. (GIGLIO et al, 2018).

Apresenta-se na Tabela 1 outras características de Hardware do ESP32.

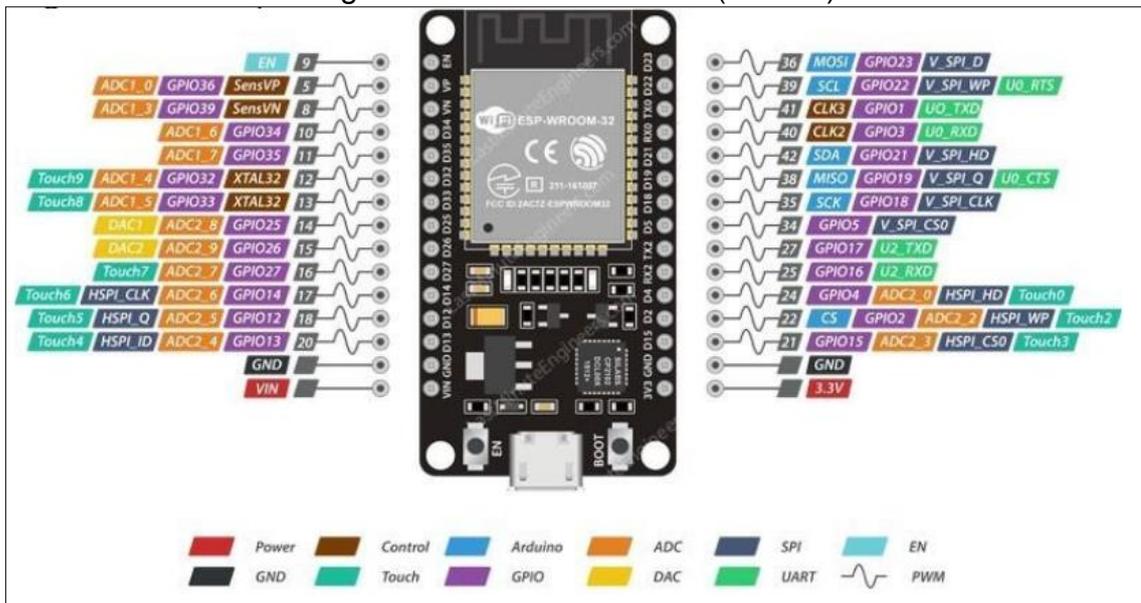
Tabela 1 – Características ESP32

Características Principais	
Microprocessador	Xtensa Dual-Core 32-bit LX6
Memória Flash Programável	32Mb
Memória RAM	512 KBytes
Memória ROM	448 KBytes
Clock	80 até 240 MHz
GPIO	36 (PWM / I2C e SPI)
Resolução do PWM	Até 16 bits
Wireless	802.11 B/G/N - 2.4GHz
Tensão de Alimentação Externa	3.3 V

Fonte: KOYANAGI (2017)

Apresenta-se na Figura 1 a placa de desenvolvimento *NodeMCU*, que embarca o *chip* ESP32, bem como, mostra os GPIOs com suas respectivas funções.

Figura 1 – Placa NodeMCU (ESP32)



Fonte: Aita (2017)

As principais vantagens e desvantagens do *NodeMCU ESP32* em relação a outras placas de desenvolvimento dispostas no mercado, segundo Aita (2017), são:

Vantagens:

- Baixo custo, sendo de fácil aquisição para desenvolvimento de novas aplicações.
- Suporte nativo a rede Wi-Fi 802.11 b/g/n.
- Suporte nativo de Bluetooth 4.2, uma das suas últimas versões.
- Baixo consumo de energia, e com um alto desempenho.

Desvantagens:

- A necessidade de aprender uma nova linguagem de programação.
- Utilizar uma pinagem reduzida.

### 2.2.1 Bluetooth Low Energy

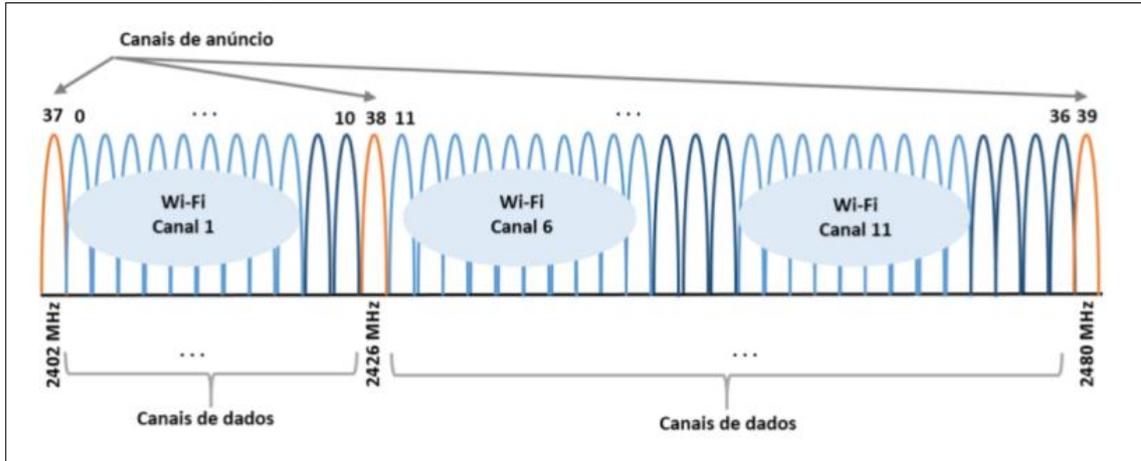
O *Bluetooth Low Energy* é uma tecnologia de rede sem fio de curto alcance, projetada e comercializada pela *Bluetooth SIG*, capaz de fazer transmissão de dados entre dois dispositivos em tempo real.

Segundo a *Bluetooth SIG* (2019), o BLE é caracterizado por oferecer um compartilhamento de dados utilizando-se baixo consumo de energia. De resto, tem muito em comum com as versões anteriores de Bluetooth, pelo fato de ambas serem de baixo custo, robustas e operarem na mesma banda de frequências.

Conforme descreve Maio (2014), O BLE opera na banda de frequências de 2,4 GHz e define 40 canais de Rádio Frequência (RF), divididos entre canais de anúncios e canais de dados. Os canais de anúncios são 3, sendo utilizados para descobrir dispositivos e estabelecer conexões, enquanto que os 37 canais de dados restantes são usados para comunicação bidirecional entre os dispositivos conectados.

A existência de apenas 3 canais de anúncio implica no aumento da probabilidade de escolha do mesmo canal por parte de outras tecnologias de transmissão, como por exemplo, o Wi-Fi. Na Figura 2 expõe-se o mapeamento das frequências BLE para os canais de dados e anúncios.

Figura 2 – Coexistência dos canais BLE e Wi-Fi

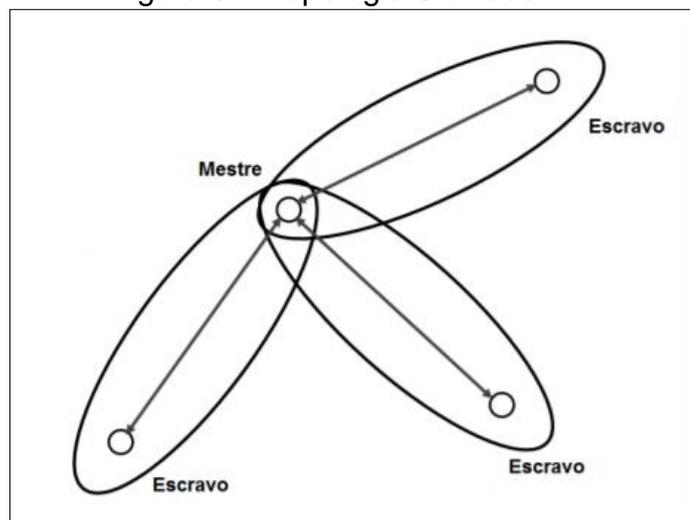


Fonte: Maio (2014)

Segundo Reck (2016), o BLE consegue enviar pacotes de dados a uma distância de até 50 metros e sua taxa de transferência é de 1 MBit/s. Além disso, possui um tempo total de 3 ms para enviar pequenos pacotes de dados, muito abaixo do 100 ms do Bluetooth Clássico.

Em relação à topologia, no *Bluetooth Low Energy* cada escravo se comunica com o mestre em uma camada física separada. Dessa forma, um mestre irá reconhecer os pedidos de conexão do escravo e estabelecer-se-á, caso os parâmetros de conexão estejam em acordo (Reck, 2016). A topologia em questão é chamada de *Star-bus*, ilustrada na Figura 3.

Figura 3 – Topologia Star-bus BLE



Fonte: Adaptado de Laird (2016)

Sua comunicação é baseada no *Generic Attribute Profile* (GATT), que é uma camada com conjuntos de características e relacionamentos com outros serviços que encapsulam o comportamento de parte de um dispositivo (*Bluetooth SIG, 2019*).

O GATT é composto pelos conceitos ilustrados na Tabela 2.

Tabela 2 – Conceitos GATT

<b>Conceitos</b>	<b>Funções</b>
Cliente	É o dispositivo que inicia comandos e solicitações do GATT, por exemplo, um smartphone.
Servidor	É o dispositivo que recebe os comandos e solicitações e retorna uma resposta, por exemplo, ESP32.
Característica	É o valor de dados transferido entre o cliente e o servidor.
Serviço	É o conjunto de características executadas para uma função específica.
Descritor	É o conjunto de dados adicionais à característica, sendo dessa forma, opcional.

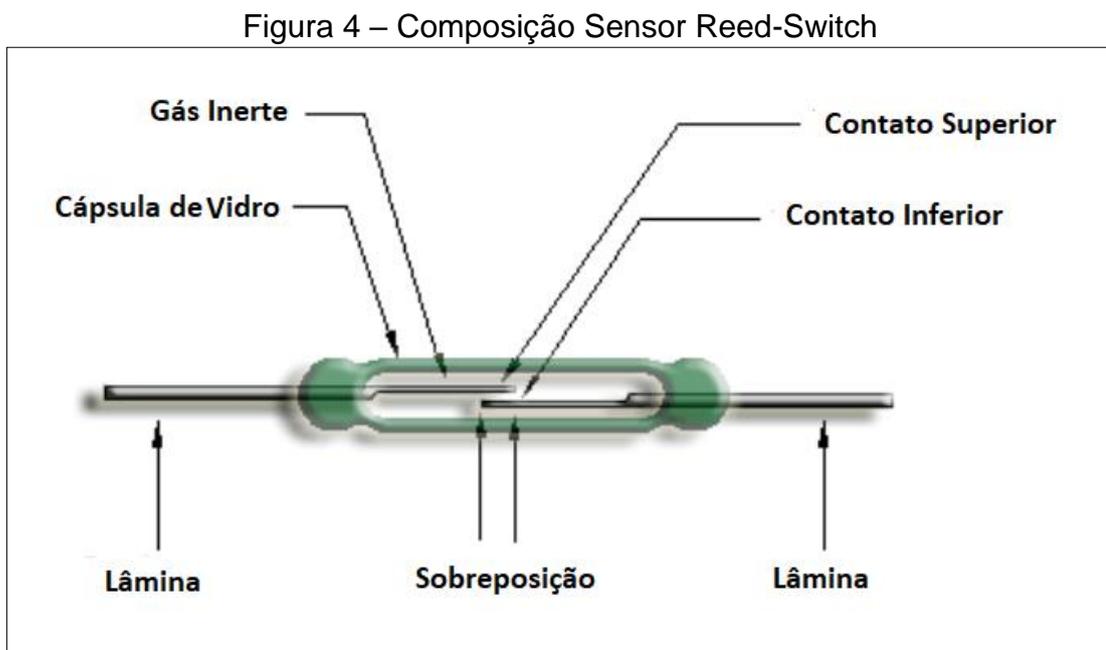
**Fonte:** Autoria Própria (2019)

As informações obtidas no GATT são armazenadas em estruturas de dados chamadas de atributos. Consoante expõe Maio (2014), cada atributo é composto por 4 elementos: 16 bits para a handle que define unicamente um atributo; um UUID (*Universal Unique Identifier*) de 16 ou 32 bits que define o tipo de atributo; a sua descrição e por fim o seu valor de tamanho variável. Dessa forma, tem-se o atributo como a identidade dos dispositivos.

## 2.3 SENSORES

### 2.3.1 Chave Magnética de efeito Hall (Reed Switch)

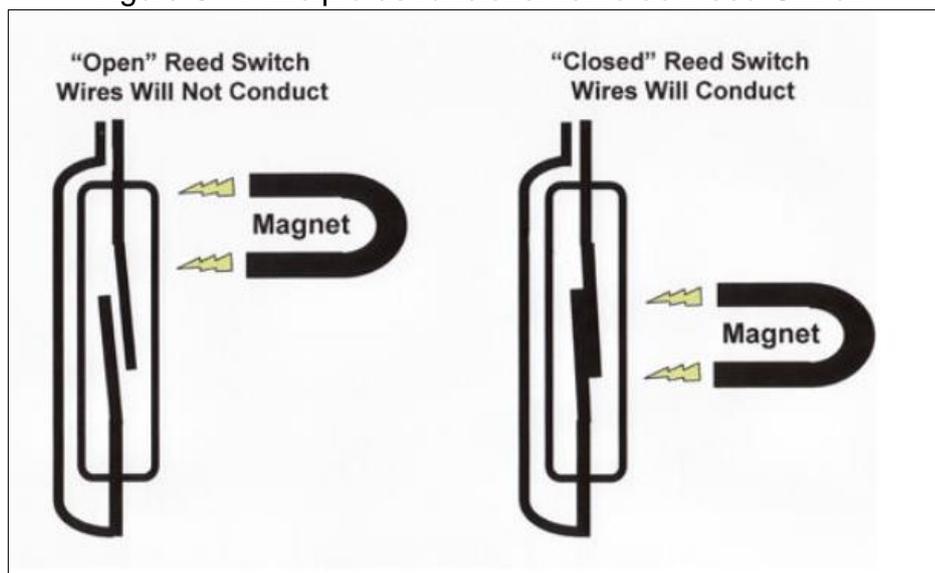
O sensor *Reed-Switch* ou interruptor de lâminas consiste em um dispositivo formado por um bulbo de vidro que contém em seu interior lâminas flexíveis feitas de material ferromagnético (ligas de níquel e ferro) que pode sofrer ação de campos magnéticos, conforme ilustrado na Figura 4.



Fonte: Adaptado de Stroski (2019)

O reed switch funciona como uma chave magnética: quando é aproximado de um campo magnético, os contatos metálicos em seu interior se fecham e conduzem corrente se estiverem conectados a um circuito (PATSKO, 2006). Internamente, o bulbo de vidro é composto por um gás inerte de modo a proteger as lâminas de uma ação corrosiva.

Conforme ilustrado na Figura 5, o princípio de funcionamento desse tipo de sensor que, após a presença de campo magnético, fecha o contato mecânico e passa a conduzir corrente.

Figura 5 - Princípio de funcionamento do *Reed-Switch*

Fonte: Wendling (2010)

Exibe-se na Tabela 3 as especificações elétricas e mecânicas de uma chave magnética *Reed-Switch* comercial.

Tabela 3 – Especificações do Reed Switch

Variável	Dimensão
Contato	Normalmente Aberto
Tensão (v)	0 ~ 100
Corrente (A)	0.5
Diâmetro do Bulbo (mm)	3.0
Comprimento do Bulbo (mm)	50

Fonte: Autoria Própria (2019)

Apesar do baixo custo e bom desempenho de funcionamento, os sensores mecânicos apresentam algumas desvantagens em relação aos elétricos, entre elas, o fato de terem peças móveis sujeitas à quebra e desgaste, além da inércia natural que limita a velocidade de ação (Wendling 2010).

## 2.4 MÓDULO CARREGADOR TP4056

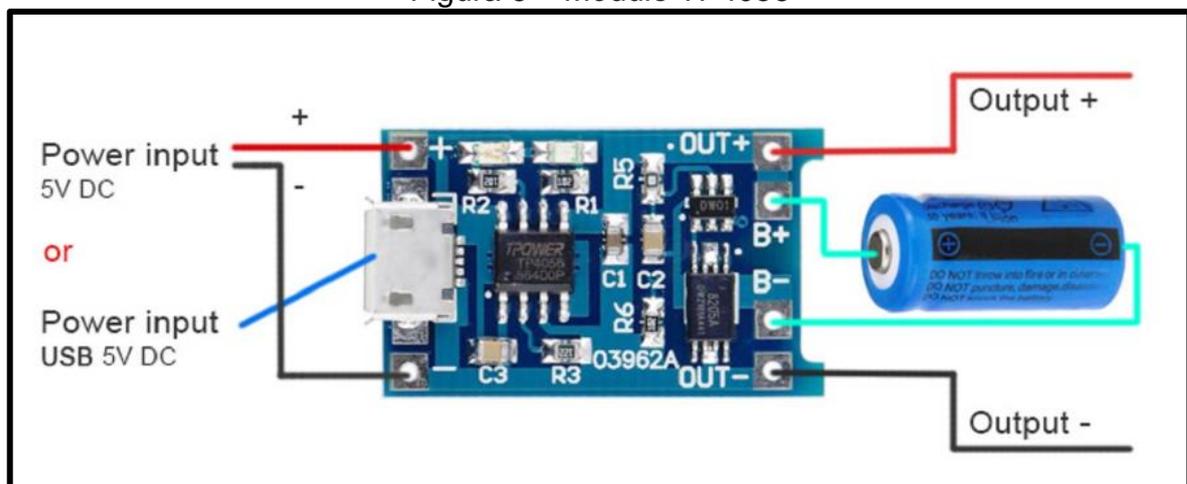
Segundo Turatti et al (2018), o carregamento de células necessita de um circuito de proteção para que sua integridade seja mantida e, sobretudo, não haja

sobrecarga, descarga excessiva e sobrecorrentes. Desta forma, o módulo TP4056 dispõe de um circuito eletrônico capaz de controlar a carga e descarga das células.

O TP4056 é um carregador linear de corrente constante/tensão constante completo para uma única célula baterias de íões de lítio. Seu pacote SOP e baixa contagem de componentes externos fazem o TP4056 idealmente adequado para aplicações portáteis. Além disso, o TP4056 pode trabalhar dentro de USB e parede adaptador. (Datasheet, 2018).

Na Figura 6 apresenta-se o carregador TP4056, onde ilustra-se a entrada (*Power Input*) e saída (OUT+ e OUT-) para o circuito externo. Além disso, a placa dispõe dos pinos (B+ e B-) para alimentação da bateria.

Figura 6 – Módulo TP4056



Fonte: Adaptado de qqtranding (2018)

Conforme descreve Wollinger (2010), o carregamento da célula de lítio polímero (Li-Po) é baseado em três estágios característicos. O primeiro estágio consiste no carregamento por corrente constante de 0.1 C, ou seja, com corrente equivalente a 10% da capacidade nominal da célula. Neste estágio a célula se encontra com tensão de 2.7 v.

Ainda segundo o autor, o segundo estágio pode ser definido como o estágio de corrente constante, onde baterias que possuem grandes taxas de descarga são carregadas com taxas de 2 à 10 C e baterias destinadas a alta durabilidade são carregadas com taxas de 0.2 a 1 C. O terceiro estágio é comumente compreendido pela Tensão Constante, efeito presente quando a célula atinge sua tensão máxima de

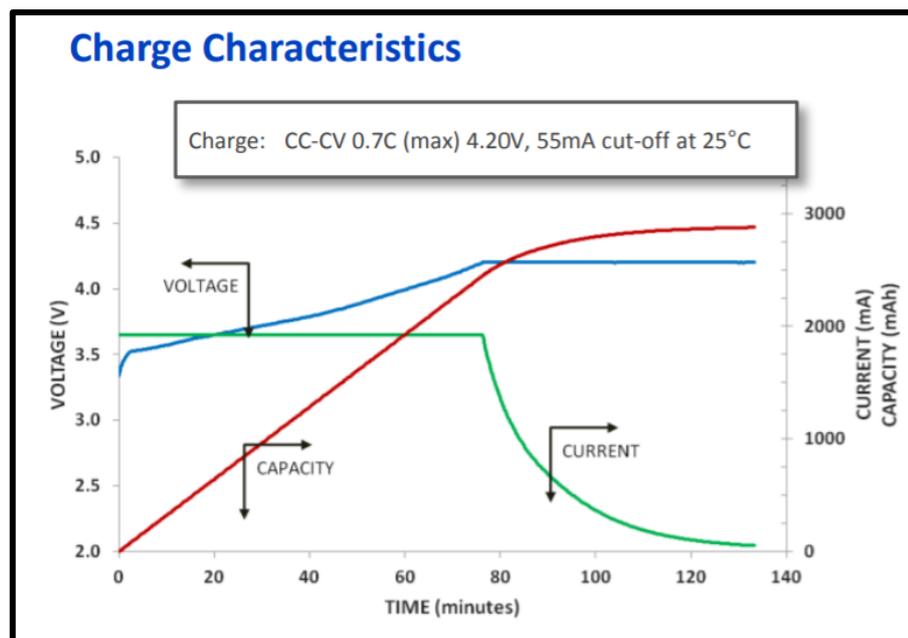
carregamento, 4.2 V. O fim do carregamento ocorre quando a corrente de carga fica abaixo de 0.0002 C.

Segundo Freitas e Júnior (2018), as principais características do módulo carregador são:

- Corrente de 1ª ajustável;
- Precisão de carregamento de 1,5%;
- Tensão de entrada de 4,5V até 5,5V;
- Tensão máxima de carregamento 4,2V;
- LED indicativo de carregamento e de carregamento máximo;
- Dimensões (2,6 cm x 1,9 cm x 1,0 cm);

A Figura 7 ilustra os estágios de carga, tensão constante e corrente constante.

Figura 7 – Curva de carga da Bateria 18650



Fonte: PANASONIC – Datasheet Bateria NCR18650 (2019)

#### 2.4.1 Bateria de Lítio (LiCoO<sub>2</sub>) 18650

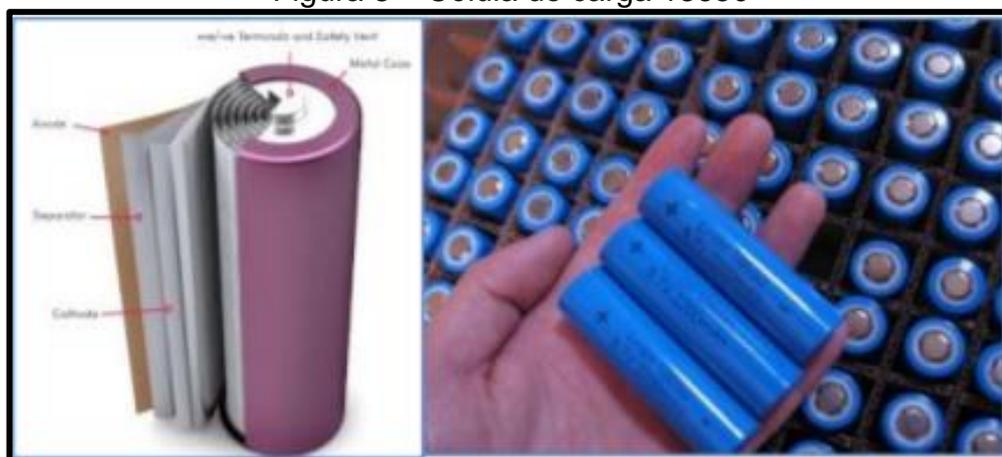
A bateria ou pilha 18650, é uma célula de lítio recarregável com dimensões de 18.5mm de diâmetro por 65.3mm de comprimento. Segundo a fabricante PANASONIC (2017), a bateria 18650 possui uma capacidade de armazenamento de

corrente de 2700 mAh com uma tensão nominal de 3.7 V, mas quando completamente carregada chega a 4.2 V, e pode descarregar até o limite de 2.7 V.

Se ultrapassado o limite superior, a célula pode sofrer desde redução da vida útil, até uma dilatação ou explosão. Se ultrapassado o limite inferior, a célula pode sofrer uma redução da vida útil, podendo perder a capacidade de ser recarregada novamente, caso caia abaixo de 2.7V a célula necessita passar por um estágio de pré-carga até ultrapassar este limiar. (Wollinger, 2018).

Demonstra-se na Figura 8 a célula de carga 18650, que leva em média 3 horas para carregamento total.

Figura 8 – Célula de carga 18650



Fonte: Panasonic – Datasheet (2019)

Ilustra-se na Tabela 4 as informações retiradas do *Datasheet* disposto pela fabricante: Panasonic.

Tabela 4 - Característica da Bateria de Lítio 18650

Peso (gramas)	46.5
Temperatura (°Celsius)	Carga: 0 a +45 Descarga: -20 a +60 Armazenamento: -20 a +50
Tensão Nominal (Volts)	3.7
Carregamento	CC, Std. 1925mA, 4.20V, 3.0 hrs

Fonte: Panasonic (2019)

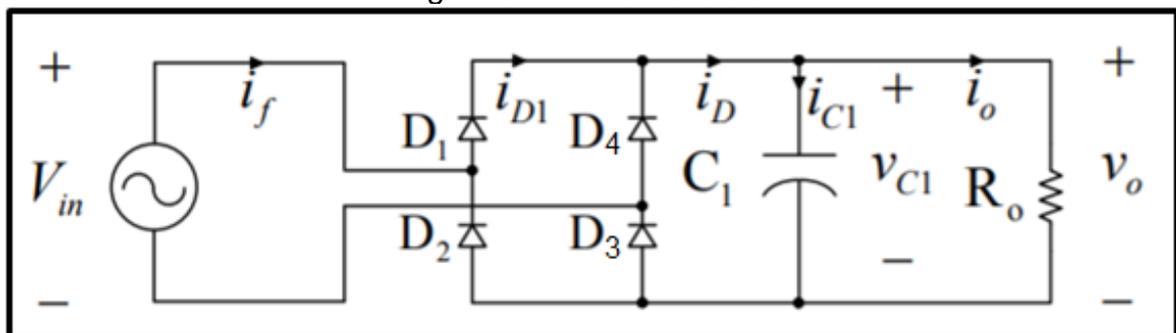
## 2.5 CONVERSOR AC-DC

### 2.5.1 Ponte de Graetz (Graetz Bridge)

Segundo Petry (2012), o circuito retificador monofásico de onda completa, conhecido como ponte de Graetz, é um circuito conversor que converte a corrente monofásica em corrente retificada, tornando-se apenas com um tipo de alternância: positiva ou negativa.

Conforme ilustrado na Figura 9, a ponte de Graetz utiliza quatro diodos semicondutores junto a um filtro com capacitor e transfere para a carga uma onda retificada.

Figura 9 – Ponte de Graetz

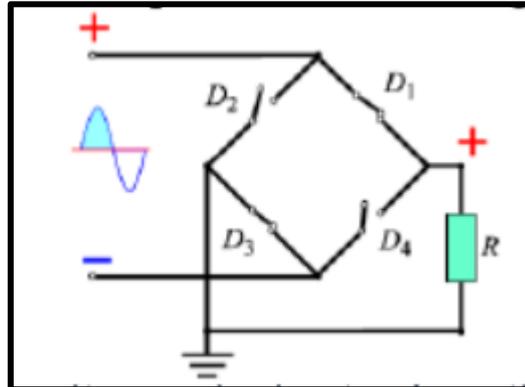


Fonte: Adaptado de Petry (2012)

O princípio de funcionamento pode ser descrito em duas etapas: semiciclos positivo e negativo.

Como pontua Beslaquem et al (2018), o semiciclo positivo pode ser descrito através da Tabela 5 onde, durante esse ciclo, ocorre o tráfego da corrente pelos diodos D1 e D3, enquanto os diodos D2 e D4 funcionam como circuito aberto, Figura 10.

Figura 10 – Circuito durante o semiciclo positivo



Fonte: Belasquem et al (2018)

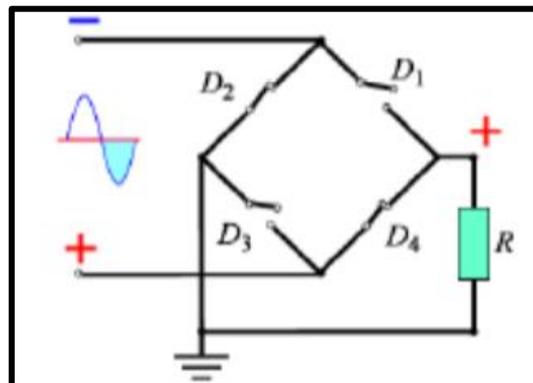
Tabela 5 – Operação dos diodos durante o semiciclo positivo

Diodo	Polarização	Regime de Operação
D1	Ânodo positivo em relação ao cátodo	Condução
D2	Cátodo positivo em relação ao ânodo	Bloqueio
D3	Cátodo negativo em relação ao ânodo	Condução
D4	Ânodo negativo em relação ao cátodo	Bloqueio

Fonte: Autoria própria (2019)

De modo análogo, o semiciclo negativo pode ser descrito através da Tabela 6, onde demonstra-se as operações dos diodos durante o ciclo negativo. Conforme verifica-se na Figura 11, os diodos D2 e D4 conduzem, corrente enquanto os diodos D1 e D3 funcionam como circuito aberto.

Figura 11 – Circuito durante o semiciclo negativo



Fonte: Belasquem et al (2018)

Tabela 6 - Operação dos diodos durante o semiciclo negativo

<b>Diodo</b>	<b>Polarização</b>	<b>Regime de Operação</b>
D1	Ânodo negativo em relação ao cátodo	Bloqueio
D2	Cátodo negativo em relação ao ânodo	Condução
D3	Cátodo positivo em relação ao ânodo	Bloqueio
D4	Ânodo positivo em relação ao cátodo	Condução

Fonte: Autoria Própria (2019)

## 2.6 CONVERTOR DC-DC

Um módulo *Step Down* é um dispositivo que tem por finalidade proporcionar um degrau descendente de potência em um sinal de entrada ou saída, seja este um sinal de alimentação ou transmissão de dados (Filho, 2018).

### 2.6.1 Step Down Lm2596

O módulo Step Down Lm2596, desenvolvido pela Texas Instruments, é um conversor DC-DC ajustável aplicado em projetos eletrônicos onde são necessárias diferentes tensões e correntes para o acionamento de componentes eletrônicos.

Segundo Filho (2018), os reguladores LM2596 atuam por meio de um processo de “liga e desliga”, onde comutam a continuidade de passagem de corrente até 150 mil vezes por segundo. O efeito de comutação permite variar a condução de corrente o que resulta na alteração de potência entregue ao circuito externo. Dessa forma, a corrente exigida para funcionamento é quase constante, e a variável alterada é a tensão para resultante do valor de potência final, o que segue a lei de Ohm. Apresenta-se na Tabela 7 as principais características elétricas do módulo conversor.

Tabela 7 – Características de operação LM2596

Característica	Grandeza
Tensão de entrada máxima	45 VDC
Tensões de saída fixa	3V3, 5V, 12V e 15V
Tensão de saída ajustável	1,23 a 37VDC
Corrente máxima de saída	3 A

Temperatura de operação	-40°C a 85°C
-------------------------	--------------

Fonte: <https://www.onsemi.com/pub/Collateral/LM2596-D.PDF>

Figura 12 – Módulo Step Down LM2596

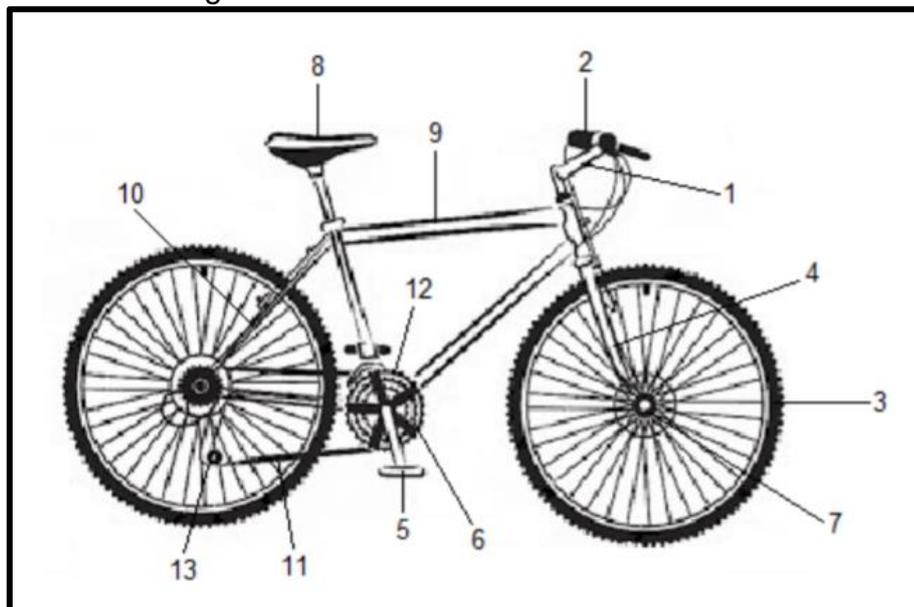


Fonte: <http://www.institutodigital.com.br/pd-140cdb-modulo-lm2596-regulador-de-tensao-ajustavel-dc-dc.html>

## 2.7 BICICLETA

A bicicleta é um veículo mecânico movido à propulsão humana sobre duas rodas criada por volta 2.500 anos atrás na China. De acordo com Gonzales (2014), existem diversos modelos de bicicleta que, no entanto, em sua maioria possuem a mesma estrutura básica: um quadro sobre duas rodas. Ilustra-se na Figura 13 as principais peças da estrutura de uma bicicleta.

Figura 13 – Estrutura básica da bicicleta.



Fonte: Adaptado de Gonzales (2014)

A Tabela 8 descreve as peças enumeradas na figura acima.

Tabela 8 – Peças da estrutura da bicicleta

Peça	Nome	Descrição
1	Mesa	Peça que conecta o guidão ao garfo.
2	Guidão	Peça tubular fixada na parte de cima do garfo, com o propósito de direcionar os movimentos da bicicleta.
3	Rodas	São compostas dos seguintes elementos: Pneu, aro, raios e cubo.
4	Garfo Dianteiro	Peça semelhante a uma forquilha, que aloja a roda dianteira e faz parte do conjunto de direção da bicicleta.
5	Pedal	É conectado ao eixo do movimento central através da pedivela.
6	Coroa	Aro dentado que transmite o movimento da pedalada para a corrente.
7	Cubo	Elemento central da roda, em volta do qual a roda gira e onde são fixados o garfo e os raios.
9	Selim	Assento do ciclista.
10	Quadro	Peça de grande importância da bicicleta, onde se fixam todas as outras partes.
11	Corrente	Conjunto de elos metálicos flexível, que transmite o movimento da coroa para o pinhão.
12	Câmbio Dianteiro	Peça que direciona a corrente na coroa.
13	Câmbio Traseiro	Peça que direciona a corrente no pinhão.

Fonte: Adaptado de Gonzales (2014)

## 2.8 GERADORES ELÉTRICOS E ALTERNADORES

Segundo a Revista Redação Indústria (2014), um gerador elétrico ou alternador é uma máquina que tem por função transformar energia mecânica em energia elétrica por meio de indução magnética.

A indução magnética é explicada através da Lei de Lenz, que estabelece: “o sentido da corrente induzida é oposto à variação do campo magnético que a gera”. Dessa forma, a variação de fluxo magnético em uma espira induz uma tensão elétrica entre os seus terminais. A partir desse princípio, desenvolveu-se o alternador que é composto por dois mecanismos: Estator e Rotor.

O estator é a parte fixa do gerador que se mantém fixo à carcaça e tem como função suportar o conjunto de bobinas enroladas no núcleo magnético. O rotor é o elemento rotacional que conduz o fluxo magnético através do seu giro. Os dois mecanismos são ilustrados na Figura 14.

Figura 14 – Mecanismos do Alternador

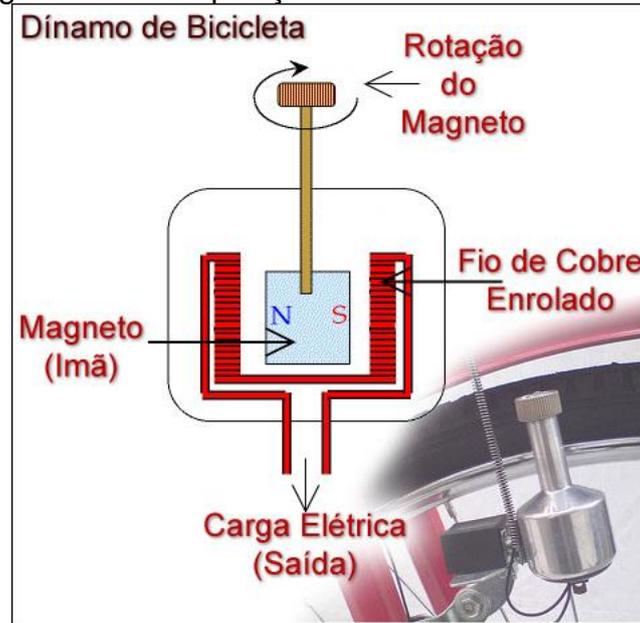


Fonte: A autoria Própria (2019)

Através da Indução eletromagnética, a corrente passa pelo rotor e cria um campo magnético que gera o movimento dos elétrons no estator, originando-se a corrente alternada. Após cada giro, o ciclo se repete.

O princípio da indução eletromagnética aproveita-se da repulsão e atração natural de cargas elétricas funcionando como um ímã. Na Figura 15 é ilustrada a estrutura interna do Alternador, onde são apresentados os elementos que o compõe.

Figura 15 – Composição do Alternador de Bicicleta



Fonte: Adaptado de Ismayatim (2012)

## 2.9 GASTO CALÓRICO NA PRÁTICA DE EXERCÍCIOS FÍSICOS

O corpo humano mantém um balanço energético entre a ingestão alimentar e o gasto energético. Dessa forma, torna-se fundamental o monitoramento da perda de calorias em razão da ingestão de alimentos.

O gasto calórico, expresso em Kcal, obtido durante a prática de exercícios físicos pode ser calculado através da Equação [ 1 ].

$$\text{Gasto Calórico Total} = \text{MET} * \text{Peso} * \text{Tempo em Minutos} \quad [ 1 ]$$

As variáveis “Peso” e “Tempo em minutos”, dispostos na equação acima, são referentes ao peso do usuário e o tempo em minutos da atividade praticada, respectivamente.

Os METs (Equivalente Metabólico da Tarefa) são índices pré-definidos da quantidade de calorias gasta de acordo com a atividade física praticada e sua respectiva intensidade.

A Tabela 9 exemplifica os valores de METs obtidos na prática do ciclismo, onde se tem uma variação do índice de acordo com a velocidade do ciclista e seu esforço físico empregado.

Tabela 9 – Quantidade de calorias gasta por atividades.

<b>Atividade</b>	<b>Intensidade</b>	<b>MET</b>
Ciclismo	0 < Pedalando < 15 Km/h – lazer, trabalho.	0.066
Ciclismo	16 < Pedalando < 19 Km/h – esforço leve.	0.100
Ciclismo	20 < Pedalando < 22 Km/h – esforço moderado.	0.133
Ciclismo	23 < Pedalando < 25 Km/h – esforço vigoroso, competição.	0.166
Ciclismo	26 < Pedalando < 30 Km/h – esforço vigoroso, competição.	0.200
Ciclismo	Pedalando > 30 Km/h – esforço muito vigoroso, competição.	0.266

Fonte: adaptado de Gil (2015)

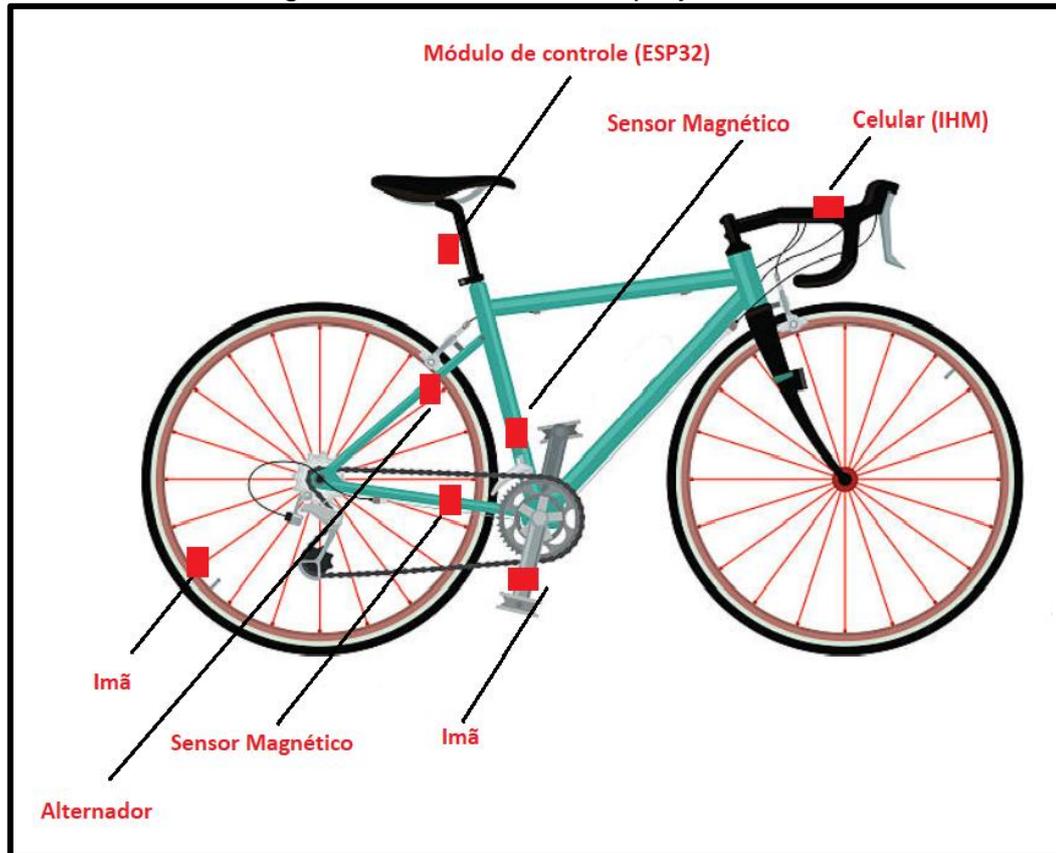
### 3 PLANEJAMENTO DO SISTEMA

#### 3.1 VISÃO GERAL

O projeto demonstra uma metodologia baseada em uma arquitetura simples, no qual se propõe um sistema de computador de bordo para bicicleta alimentado por um alternador, capaz de processar informações e repassá-las para o usuário através de um aplicativo de celular.

Fisicamente, o projeto será constituído de sensores magnéticos tipo *Reed-Switch*, um alternador, módulo de controle (ESP32) e o celular que fará a *Interface Homem-Máquina (IHM)*, conforme ilustra-se na Figura 16.

Figura 16 – Elementos do projeto.



Fonte: Adaptado de gettyimages (2019).

Serão dois sensores *Read Switch* fixados junto ao quadro, que verificarão as rotações por minuto da roda traseira e do pedal através da passagem dos ímãs de neodímio fixados na roda e pedal, respectivamente.

Abaixo do selim localiza-se o módulo de controle: Placa de Circuito Impresso (PCI) embutida de microcontrolador ESP32, conversor AC/DC, módulo regulador DC/DC *Step Down* e carregador de bateria (módulo TP4056). Justaposta ao módulo de controle estará a bateria de Lítio de 3.7 V, que fará a alimentação do sistema embarcado.

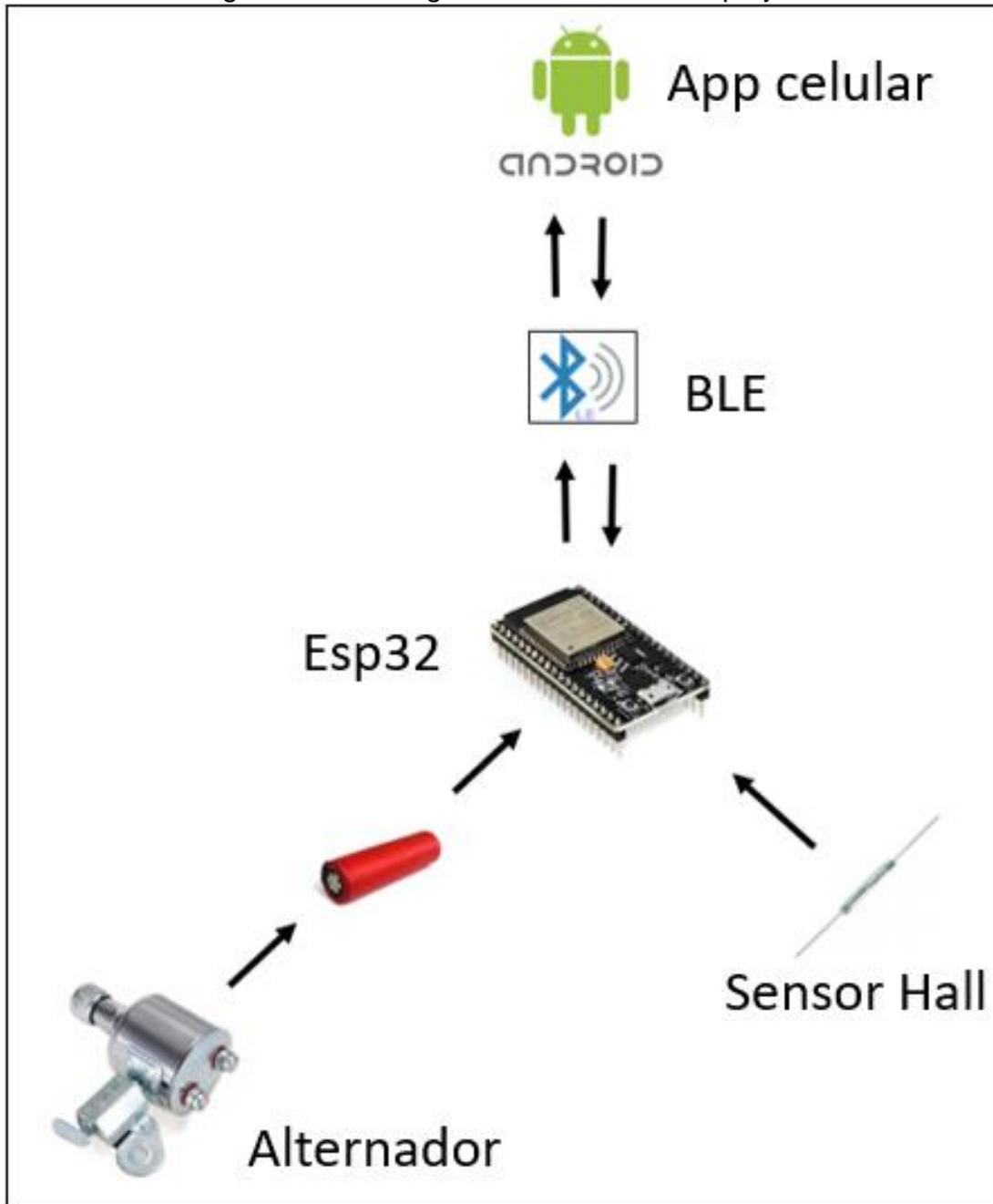
O sistema embutido na bicicleta é dotado, ainda, de um alternador que ficará alicerçado junto ao quadro, paralelamente à roda traseira. Dessa forma, o eixo central do alternador entrará em contato com o pneu que ocasionará no giro do rotor do alternador.

E, finalmente, no guidão da bicicleta constará o celular que fará a interface homem máquina. Com isso, o usuário terá o controle completo do sistema e fará a

supervisão dos parâmetros disponíveis pelo computador de bordo, além de executar os comandos disponíveis.

Ilustra-se na Figura 17 o diagrama em blocos da estrutura do projeto do computador de bordo para bicicleta, onde são apresentados os elementos que o compõe, bem como a comunicação entre eles.

Figura 17 – Fluxograma da estrutura do projeto



Fonte: Autoria Própria (2018)

### 3.2 METODOLOGIA

Para atingir os objetivos propostos no projeto, tornou-se necessário separar as atividades em etapas, consoante indica o cronograma disposto na Tabela 10. Dessa forma, o primeiro passo foi realizar um levantamento bibliográfico para que fosse possível obter conhecimento sobre os elementos que compõem o computador de bordo para bicicleta.

Realizadas as pesquisas, o passo seguinte foi fazer a aquisição da bicicleta e dos componentes eletrônicos necessários para o desenvolvimento do *hardware* e, de forma conjunta, trabalhar no desenvolvimento do *software* do sistema.

A segunda etapa, realizada entre Janeiro e Abril, foi ingressar no desenvolvimento do *firmware* do aplicativo. Para isso, realizou-se o estudo da linguagem *Dart*, necessária para a programação do app Android na plataforma *Visual Studio Code*. Para tal utilizou-se o Flutter, SDK desenvolvido pelo Google com *widgets* para desenvolvimento de aplicativos *mobile*.

De forma paralela à etapa anterior, deu-se início à programação do *software* escrito no módulo ESP32 na linguagem de programação C. Nesse período, de forma consequente, deu-se início também à montagem do *hardware*, composto por sensores e o circuito de controle na protoboard para realização de testes e validações.

A quarta etapa foi trabalhar no desenvolvimento dos desenhos mecânicos referentes aos suportes dos sensores e placas de controle fixados na bicicleta. Para isso, utilizou-se o software *Autodesk Inventor Professional* e, posteriormente, imprimiu-se as peças na impressora 3D.

A última etapa no processo de desenvolvimento do projeto foi realizar a montagem do circuito eletrônico na Placa de Circuito Impresso e, sequencialmente, realização de testes práticos com o projeto montado na bicicleta.

Tabela 10 – Cronograma de Atividades.

Atividades	2018		2019						
	Novembro	Dezembro	Janeiro	Fevereiro	Março	Abril	Maiο	Junho	Julho
Levantamento de referências bibliográficas.	X	X	X	X	X	X	X	X	
Aquisição de componentes a serem utilizados no projeto.		X	X	X	X	X			

Desenvolvimento do Firmware executado no ESP32.			X	X	X	X			
Desenvolvimento do código do Aplicativo.				X	X	X	X		
Desenvolvimento do desenho mecânico das peças utilizadas no suporte dos sensores.					X	X	X		
Montagem do circuito de controle de corrente do Alternador e realimentação da bateria.			X	X	X	X	X	X	
Desenvolvimento da monografia.				X	X	X	X	X	X

Fonte: Autoria Própria (2019)

## 4 DESENVOLVIMENTO DO PROJETO

### 4.1 AMBIENTE DE DESENVOLVIMENTO

O projeto foi desenvolvido com o auxílio dos *Softwares: Proteus Professional 8.6*, que realiza a simulação elétrica do circuito; *Arduino IDE 1.8.2*, responsável pelo desenvolvimento do *firmware* gravado no ESP32; *Autodesk Inventor 2016*, para criação de protótipos virtuais tridimensionais; e da IDE *Visual Studio Code*, onde programa-se o código do aplicativo.

#### 4.1.1 Ambiente de Desenvolvimento do Firmware

Para desenvolver-se a programação do sistema embarcada ao microcontrolador ESP32, foi necessária a instalação da IDE para a plataforma Arduino, disponível no site: [www.arduino.cc](http://www.arduino.cc). A versão utilizada foi a 1.8.2.

Além disso, foi necessário o uso de bibliotecas adicionais para uso do Bluetooth Low Energy, disponíveis no site: [https://github.com/nkolban/ESP32\\_BLE\\_Arduino/tree/master/src](https://github.com/nkolban/ESP32_BLE_Arduino/tree/master/src), as quais seguem abaixo:

✓ BLEDevice.h

- ✓ BLEServer.h
- ✓ BLEUtils.h
- ✓ BLE2902.h

#### 4.1.2 Ambiente de Desenvolvimento mecânico

Para o desenvolvimento do projeto mecânico do sistema foi necessário o uso do programa *Autodesk Inventor*, disponível no site: <https://www.autodesk.com.br/products/inventor/>. A versão utilizada foi a *Autodesk Inventor 2016* na versão estudante.

#### 4.1.3 Ambiente de Desenvolvimento do Aplicativo

O desenvolvimento do aplicativo foi realizado através da IDE *Visual Studio Code*, disponível no endereço eletrônico: <https://code.visualstudio.com/>.

Nas dependências da IDE, utilizou-se o Kit de Desenvolvimento de Software (SDK) *Flutter*, obtido através do repositório do *GitHub* clonando o canal principal e, em seguida, realizando-se o *download* das dependências do SDK, consoante ilustra-se na Figura 18.

Figura 18 – Comandos para adição do Flutter através do GitHub

```
$ git clone -b master https://github.com/flutter/flutter.git
$ ./flutter/bin/flutter --version
```

Fonte: <https://flutter.dev/docs/development/tools/sdk/releases>

#### 4.1.4 Ambiente de Desenvolvimento Elétrico

O ambiente de desenvolvimento do circuito elétrico simulacional e do *layout* da placa de circuito impresso (PCB), foi o *software Proteus Professional*, disponível no endereço eletrônico: <https://www.labcenter.com/downloads/>. Para isso, utilizou-se a versão 8, que dispõe de ferramentas modernas para o desenvolvimento de circuitos elétricos.

---

## 4.2 DESCRIÇÃO DO PROJETO DE FIRMWARE

O projeto do computador de bordo é baseado em uma arquitetura integrada de *Hardware* e *Software*. Diante disso, propôs-se dividir o projeto nessas duas frentes de trabalho.

### 4.2.1 Processador ESP32

O microcontrolador ESP32 dispõe de dois núcleos de processadores que podem realizar tarefas de forma paralela. Diante disso, dividiu-se as operações do sistema para os dois núcleos.

Ao iniciar-se a execução do ESP32, no núcleo 1 executam-se as seguintes tarefas:

- São inicializadas 2 interrupções externas, correspondentes aos 2 sensores de efeito *Hall* (*Reed Switch*) e 1 interrupção interna para gerar um contador que auxiliará nos cálculos que serão vistos nas próximas seções deste trabalho.
- Habilita-se a comunicação *Bluetooth Low Energy* do ESP32. Dessa forma, ele passa a ser um dispositivo escravo e fica à espera de um dispositivo mestre (Aplicativo no celular) para iniciar uma comunicação.

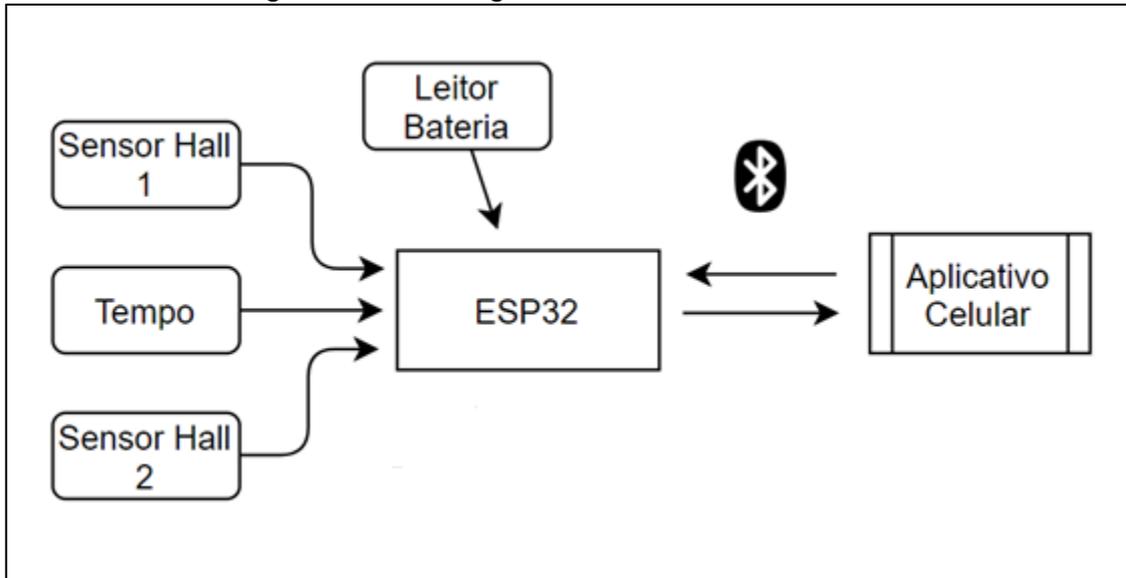
De modo análogo, ao executar-se o ESP32, o núcleo 2 fica responsável por realizar a seguinte tarefa:

- Enviar e receber dados do dispositivo mestre assim que a comunicação for estabelecida. Dessa forma, ao comunicar-se com o aplicativo no celular, o ESP32 passa a enviar as seguintes informações: velocidade, quilometragem, tensão da bateria e gastos calóricos para o dispositivo móvel. Concomitantemente, o microcontrolador fica à espera de comandos do aplicativo, como por exemplo, o peso do usuário, utilizado para cálculo de gastos calóricos, conforme veremos adiante.

Descreve-se na Figura 19 o processo de execução do processador. No sistema em questão possuem-se entradas – sensores de efeito hall, interrupção de tempo, tensão da bateria e dados via *bluetooth* através do aplicativo de celular – e

apenas uma de saída: dados via *bluetooth* para o aplicativo de celular. Conforme observa-se, o ESP32 capta todas essas informações para serem usadas durante a execução das linhas de código e, de forma paralela, as transmite via *bluetooth* para o aplicativo do celular.

Figura 19 – Fluxograma do Processador.



Fonte: Autoria Própria (2019).

Abaixo encontram-se fragmentos do código, onde visualiza-se, inicialmente na Figura 20, a declaração de algumas das variáveis utilizadas no código de *firmware*.

Figura 20 – Variáveis do sistema.

```
24 #define PortSensor1 19
25 #define PortSensor2 18
26 #define PortSensor3 32
27 double energia;
28 int energialimpa;
29 int flagSource;
30 char valor1[10];
31 volatile int quilometragem;
32 volatile int velocidade;
33 volatile int velocidade2;
34 volatile int peso;
35 volatile int zerador=0;
36 volatile int distancia = 2;
37 volatile int tempatual = 0;
38 int entrada = 0;
```

Fonte: Autoria Própria (2019).

Na sequência, é perceptível através da Figura 21, a declaração das interrupções externas referentes aos sensores de efeito *Hall* e a inicialização do *Timer* (interrupção interna) necessário para criação de um contador.

Figura 21 – Interrupções externas e internas.

```

void setup() {
  Serial.begin(115200);
  pinMode(PortSensor1, INPUT);
  pinMode(PortSensor2, INPUT);
  pinMode(PortSensor3, INPUT);
  attachInterrupt(digitalPinToInterrupt(PortSensor1), contador, RISING);
  attachInterrupt(digitalPinToInterrupt(PortSensor2), contador_Cal, RISING);
  startTimer();

  xTaskCreatePinnedToCore(
    coreTaskZero, /* função que implementa a tarefa */
    "coreTaskZero", /* nome da tarefa */
    8192, /* número de palavras a serem alocadas p
    NULL, /* parâmetro de entrada para a tarefa (p
    1, /* prioridade da tarefa (0 a N) */
    NULL, /* referência para a tarefa (pode ser NU
    0); /* Núcleo que executará a tarefa */

    delay(10); //tempo para a tarefa iniciar

```

Fonte: Autoria Própria (2019).

As linhas de código do *loop* executado no núcleo 2 do ESP32 são expressas na Figura 22, onde demonstra-se a montagem do buffer de dados de resposta enviado via *bluetooth low energy* para o aplicativo instalado no celular.

Figura 22 – Loop do Núcleo 2 do processador.

```

//----- NÚCLEO 2 -----
void coreTaskZero(void*z){
  while(1)
  {
    if (deviceConnected)
    {
      char humidityString[2];
      char temperatureString[2];
      char testando[2];
      char bateria[2];

      dtostrf(velocidade, 1, 2, humidityString);
      dtostrf(quilometragem, 1, 2, temperatureString);
      dtostrf(GastoCalorico, 1, 2, testando);
      dtostrf(energialimpa, 1, 2, bateria);

      char dhtDataString[16];
      sprintf(dhtDataString, "%d,%d,%d,%d", velocidade, quilometragem, GastoCaloricoInt, energialimpa);

      pCharacteristic->setValue(dhtDataString);
      pCharacteristic->notify(); // Envia o valor para o aplicativo!
    }
    delay(1000);
  }
}

```

Fonte: Autoria Própria (2019)

#### 4.2.2 Tarefa De Interrupção Sensor De Efeito Hall 1

O sensor de efeito *hall* funciona como um botão normalmente aberto, conforme visto na seção 2.3.1 deste trabalho.

Quando o ímã, que está preso ao aro, passa perto do sensor que está preso ao garfo, o circuito interno do sensor fecha e envia o sinal *HIGH* para a porta de interrupção do *ESP32*.

Ao receber o sinal no pino de interrupção, o *ESP32* para a execução do código para atender à interrupção com a maior prioridade e executar a tarefa referente àquela interrupção. Nesta tarefa, acrescenta-se à variável quilometragem o valor do comprimento do pneu, calculado através da Equação [ 2 ] .

$$D = 2 * \pi * r \quad [2]$$

Onde *r* é o raio do pneu.

Conforme apresenta a Figura 23, após o incremento à variável quilometragem, é realizado o cálculo do parâmetro velocidade através da equação [3], onde se calcula a velocidade média a cada giro do pneu.

Figura 23 – Tarefa da interrupção do sensor Hall da roda

```
void contador(){
  int temporizador = millis();
  if(temporizador - tempatual > 20)
  {
    quilometragem+=distancia;
    velocidade = ((distancia*3,6*1000)/(temporizador - tempatual));

    if((velocidade-velocidade2)>15) velocidade = velocidade2;

    if(velocidade < 2) velocidade = 0;
    tempatual = temporizador;

    velocidade2 = velocidade;
  }
}
```

Fonte: Autoria Própria (2019)

$$V_m = \frac{\Delta S}{\Delta T} = \frac{S - S_o}{T - T_o} = \frac{D}{T - T_o} \quad [3]$$

Onde,

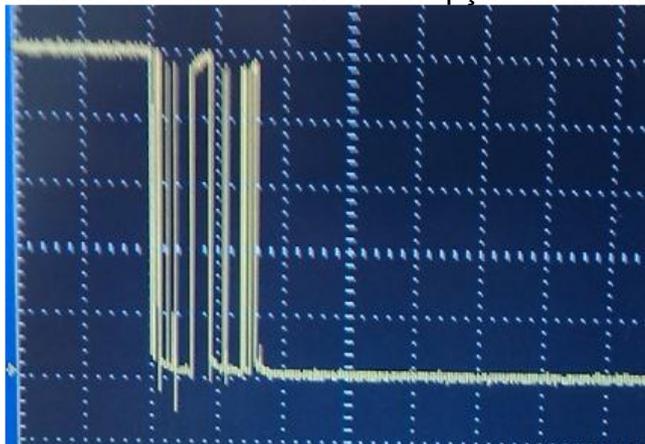
**D** é o comprimento da circunferência do Pneu.

**T** é o tempo final (Tempo registrado instantaneamente na volta anterior).

**T<sub>o</sub>** é o tempo inicial (Tempo registrado instantaneamente na volta atual).

A execução da rotina de interrupção do sensor leva, em média, 1 milissegundo. Entretanto, ocorre um fenômeno conhecido como bouncing em que o sensor lê várias atenuações de voltagem em apenas uma interrupção, fazendo várias contagens em uma única interrupção. A Figura 24 mostra este fenômeno.

Figura 24 – Sinal de entrada de uma interrupção no osciloscópio



Fonte: Autoria Própria (2019)

Para contornar esta situação, utilizou-se, via *software*, a técnica de atraso temporal conhecida como *debounce*, em que o sistema só deve entender uma nova interrupção após 20 milissegundos da primeira. Para isso, registra-se o tempo em que o sensor emite o primeiro pico de sinal de nível alto e, durante os próximos milissegundos definidos, o processador ignora novos registros até que se decorra os 20 milissegundos.

Contudo, o sensor é capaz de atender 50 interrupções por segundo. Com isso, torna-se possível o cálculo da velocidade máxima do sistema através da equação [4].

$$V_{m\acute{a}x} = 2 * \pi * r * 50 \quad [4]$$

$$V_{m\acute{a}x} \cong 314 * r[m / s] \quad [5]$$

O sistema permite chegar a uma velocidade máxima de aproximadamente 100 km/h, considerando-se um raio máximo de 32 cm.

#### 4.2.3 Tarefa de interrupção Sensor de Efeito Hall 2.

Com funcionamento análogo ao sensor apresentado na seção anterior, o sensor instalado no pedal tem como objetivo primordial verificar se o usuário está pedalando.

Nessa perspectiva, sempre que o sensor verifica a passagem do ímã instalado no pedal, inicia-se uma tarefa de interrupção externa no ESP32,

Figura 25.

Figura 25 – Tarefa de Interrupção Sensor de Efeito Hall do Pedal.

```
void contador_Cal() {  
    zerador = 0;  
}
```

Fonte: Autoria Própria (2019)

A função básica dessa tarefa é zerar o contador que será apresentado na próxima seção.

#### 4.2.4 Tarefa de Interrupção do Timer

A rotina de interrupção do Timer no ESP32 tem como função primordial auxiliar no cálculo do gasto calórico. Solicitada a cada 1 s, a tarefa se inicia com o incremento da variável “zerador” que tem como encargo contar os intervalos de tempo em que o usuário fica sem pedalar.

Sequencialmente, ocorre o incremento das variáveis de tempo usadas no cálculo do Gasto Calórico. Para isso, há duas condições para que o tempo comece a ser implementado: o usuário necessita informar o seu peso e deve estar pedalando. Dessa forma, o peso tem que ser maior que 0 e o valor do contador tem que ser menor que 10, conforme apresenta a Figura 26.

Figura 26 – Incremento do tempo a cada intervalo de velocidade

```
void func_timer() {
    zerador++;
    flagSource++;
    if (zerador > 50){ zerador = 11;}

    if(zerador < 10 && pesoreal > 0)
    {

        if(velocidade>= 0 && velocidade < 15){ C1++; }

        if(velocidade>=15 && velocidade<19) { C2++; }

        if(velocidade>=19 && velocidade<22) { C3++; }

        if(velocidade>=22 && velocidade<25) { C4++; }

        if(velocidade>=25 && velocidade<30) { C5++; }

        if(velocidade>=30){ C6++; }

    }
}
```

Fonte: Autoria Própria (2019)

Baseado na equação [1], explicitada na seção 2.8 deste trabalho, verifica-se que se o usuário estiver pedalado nos últimos 5 segundos, será contabilizado o cálculo de gasto calórico para cada intervalo de velocidade e, ao fim, somam-se todos, consoante ilustrado na Figura 27.

Figura 27 – Incremento do Gasto Calórico

```
if(zerador<5) {

    GastoCalorico1 = MET1 * pesoreal * (C1/60);
    GastoCalorico2 = MET2 * pesoreal * (C2/60);
    GastoCalorico3 = MET3 * pesoreal * (C3/60);
    GastoCalorico4 = MET4 * pesoreal * (C4/60);
    GastoCalorico5 = MET5 * pesoreal * (C5/60);
    GastoCalorico6 = MET6 * pesoreal * (C6/60);

}
```

Fonte: Autoria Própria (2019)

---

Ressalta-se que a leitura do sensor é realizada através da mudança de estado: nível baixo para nível alto, fenômeno conhecido como *RISING*. Dessa forma, caso o imã fique estacionário em frente ao sensor, não ocorrerá a falsa impressão de que o ciclista esteja pedalando, visto que, após o *RISING* o estado continua em nível alto. Para que novas leituras do sensor ocorram, é necessário que o estado volte para o nível baixo de forma a possibilitar um novo *RISING*.

#### **4.2.5 Aplicativo Android**

O aplicativo, que funcionará como Interface Homem Máquina (IHM), foi desenvolvido através da IDE *Visual Studio Code* utilizando-se a linguagem *Dart*, conforme apresentado na seção 4.1.3 deste trabalho.

No aplicativo desenvolvido (Figura 28), são apresentados os seguintes parâmetros: Velocidade, em quilômetros por hora; Distância, em metros; Gasto Calórico, em quilocalorias; Tensão de carga da bateria, em V.

Figura 28 – Layout do aplicativo



Fonte: Autoria Própria (2019)

Através das variáveis do sistema dispostas acima, é permitido ao ciclista verificar a velocidade instantânea da bicicleta ao longo de seu trajeto, de forma que o permita manter o controle de sua velocidade. Além disso, o sistema embarcado faz o registro da quilometragem total de vida da bicicleta de forma a auxiliar o usuário nas manutenções periódicas que necessitam ser feitas de acordo com sua rotação.

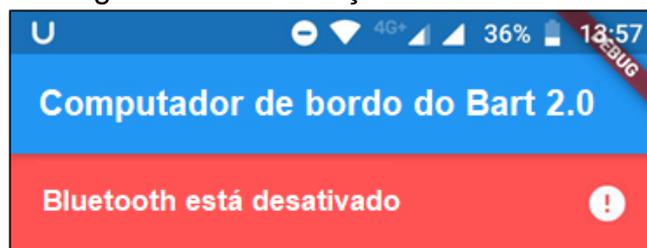
Ao usuário é possível, ainda, fazer a contabilização de gasto calórico durante seu percurso sobre a bicicleta, sendo necessário que seja informado o peso. O sistema dispõe de um sensor de efeito hall no pedal com o objetivo de detectar movimento no pedal, ou seja, o gasto calórico só será contabilizado se o ciclista estiver pedalando. Em situações onde o usuário percorra em terreno com declive, sem pedalar, não será contabilizado o gasto calórico.

Outra funcionalidade do aplicativo é a supervisão do nível de carga da bateria. Ao usuário é exequível acompanhar a porcentagem de carga da célula durante o processo de carga e descarga através de uma animação.

## Modo de operação

Ao iniciar, o aplicativo realiza uma verificação sobre o *status* do *bluetooth* do aparelho celular. Se o bluetooth estiver desativado, o aplicativo emite uma notificação, conforme ilustrado na Figura 29.

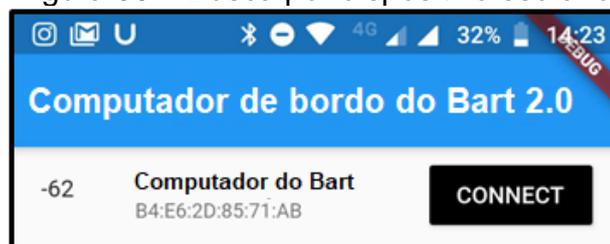
Figura 29 – Notificação de alerta BLE



Fonte: Autoria Própria (2019)

Ao se ativar o bluetooth do aparelho, torna-se possível realizar a busca por dispositivos escravos que possuem *bluetooth low energy* ativos, sobretudo o ESP32, nomeado de “Computador de Bordo do Bart 2.0” (Figura 30). Dessa forma, faz-se a conexão entre o aplicativo e o módulo de controle (ESP32).

Figura 30 – Busca por dispositivo escravo.



Fonte: Autoria Própria (2019).

Após o pareamento com o ESP32, o aplicativo passa a receber um *buffer* de dados que segue o protocolo de comunicação demonstrado na Tabela 11. A cada novo *buffer* recebido, o app faz atualizações no layout da tela e modifica o valor das *labels* e animações.

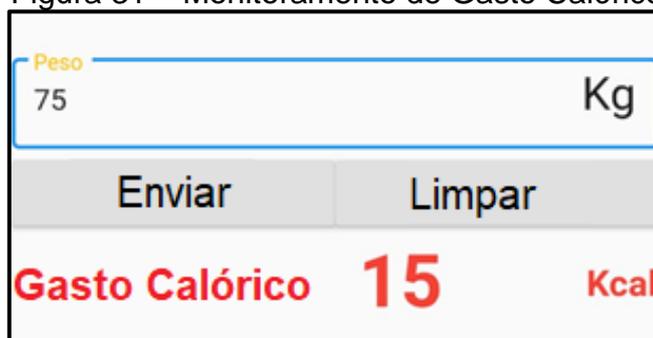
Tabela 11 – Protocolo de comunicação ESP32-Applicativo

Tamanho	4 bytes	4 bytes	4 bytes	4 bytes
Variável	velocidade	quilometragem	GastoCalorico	Energialimpa

Fonte: Aatoria Própria (2019)

Para iniciar a contagem de gasto calórico é necessário que o usuário insira seu peso na caixa *Peso*, em quilogramas, e pressione o botão “Enviar”. O valor inserido será enviado via BLE para o ESP32. O microcontrolador, por sua vez, realizará os cálculos de gasto calórico e retornará o valor calculado para o aplicativo via BLE, que o apresentará na tela, coerente ao ilustrado na Figura 31.

Figura 31 – Monitoramento de Gasto Calórico



Peso	75	Kg
Enviar	Limpar	
Gasto Calórico	15	Kcal

Fonte: Aatoria Própria (2019)

Em situação oportuna, pode-se realizar o reinício da contagem de gastos calóricos. Para tanto, basta que seja pressionado o botão “Limpar”. Com isso, o aplicativo limpará os *labels* do app e enviará um código chave para o ESP32, que por sua vez, incrementará valor nulo a todas as variáveis referentes ao cálculo de gasto calórico.

A supervisão de carga da bateria é apresentada de dois modos ao usuário: a partir do valor numérico da tensão e através de uma barra de animação, Figura 32.

Figura 32 - Animação da Bateria



Fonte: Aatoria Própria (2019)

Para que fosse possível fazer a animação, desenvolveu e implementou-se a função *map*, que estipula valores de 0 a 100 por cento de acordo com a variação de 3.50 a 4.60 volts da tensão da bateria, convergente ao demonstrado na Figura 33.

Figura 33 – Função Map

```
double map(double x, double in_min, double in_max, double out_min, double out_max) {  
    return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;  
}
```

Fonte: Autoria Própria (2019)

### 4.3 DESCRIÇÃO DO PROJETO ELÉTRICO

O projeto elétrico é constituído de um sistema de recarga de bateria composto por alternador, conversor AC/DC, Regulador de tensão e módulo carregador de bateria, bateria, microcontrolador ESP32 e componentes adjacentes.

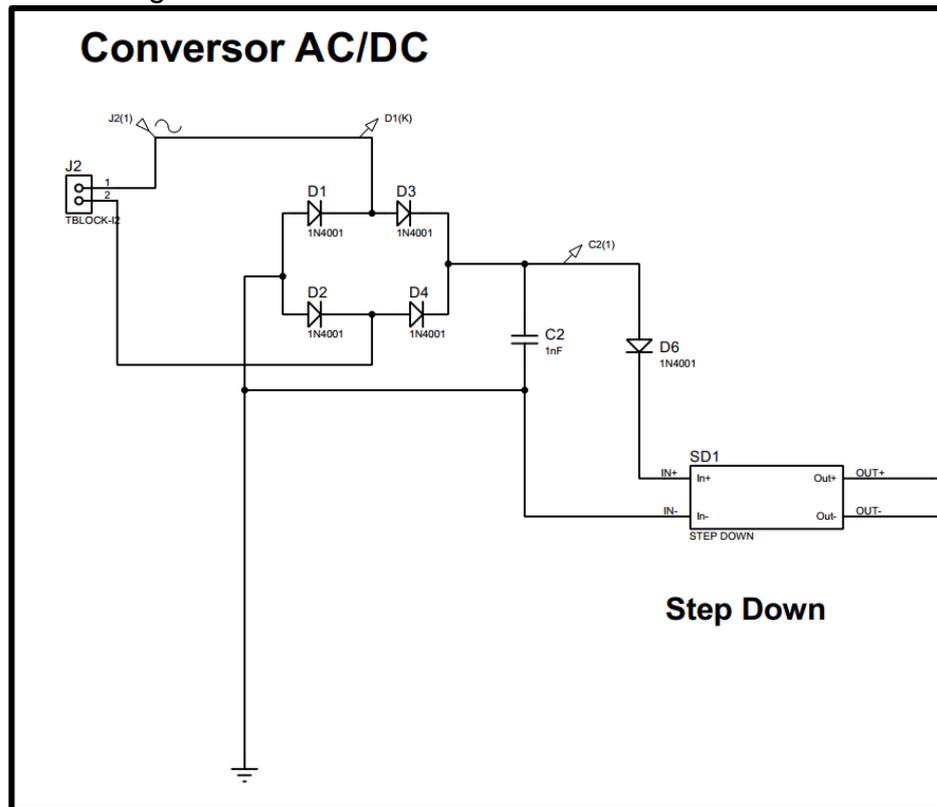
#### 4.3.1 Funcionamento do Circuito

O Alternador é a fonte principal de alimentação do sistema embarcado. Dessa forma, ele passa a ser o fornecedor de corrente continuamente para os dispositivos periféricos. Para seu uso, torna-se necessário o desenvolvimento de um circuito de conversão e regulação de tensão, visto que este dispositivo gera tensão alternada de 0 a 12 volts e o carregador de bateria (TP4056) necessita de uma tensão contínua de 5 volts. A Figura 49, no Apêndice A, mostra o alternador utilizado para gerar energia.

Dessa forma, incrementou-se ao sistema uma ponte de diodos retificadora em conjunto com um capacitor eletrolítico, para que houvesse a conversão de corrente alternada para corrente contínua, através da retificação do sinal e sua posterior filtragem.

Realizada a conversão AC/DC, a etapa seguinte foi regular a tensão de 12 V no capacitor para 5 V. Para esse fim, utilizou-se o conversor DC-DC módulo Step-Down Lm2596, que é capaz de fornecer uma tensão de saída ajustada de 1,5 a 35 V para uma entrada variante de 3,2 a 40 V, segundo a fabricante Texas Instruments (2016). Ilustra-se na Figura 34 os dois conversores utilizados primariamente no circuito.

Figura 34 - Conversores AC-DC e DC-DC

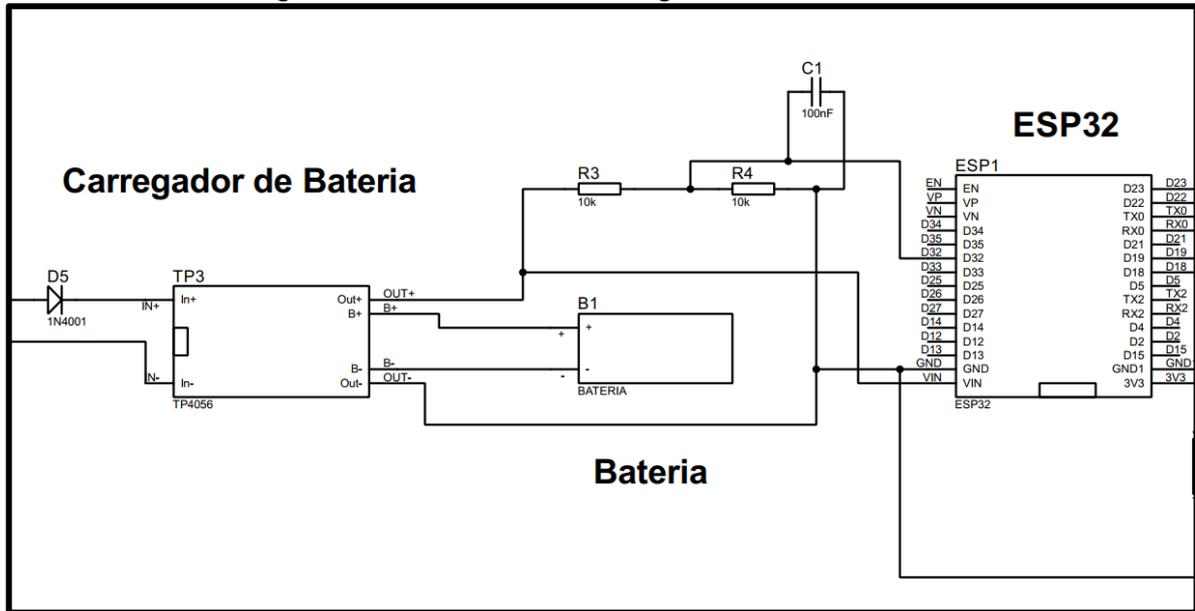


Fonte: Autoria Própria (2019)

Regulada a tensão em 5 V através do conversor DC-DC, realizou-se a conexão do módulo TP4056. Diante disso, o módulo TP4056 passa a fazer a recarga da bateria de lítio acoplada a ele, e paralelamente, passa a alimentar o microcontrolador ESP32 e periféricos.

A fonte de energia secundária é atribuída à bateria, que passa a ser solicitada em um eventual cessar de fornecimento de energia do alternador. Portanto, se o alternador deixar de fornecer corrente ao sistema, a bateria passa a ser a fonte de corrente, de forma que o microcontrolador não resete e perca informações em sua memória *flash*.

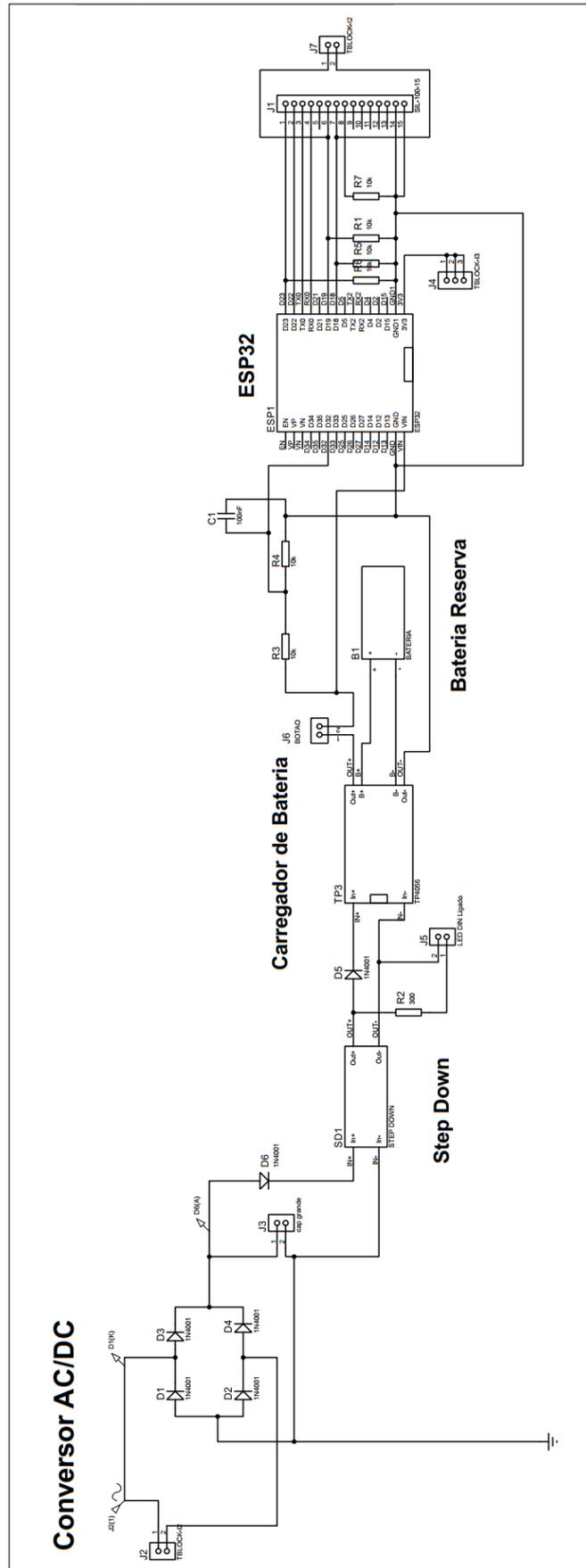
Figura 35 – Circuito de recarga da bateria de Lítio



Fonte: Autoria Própria (2019)

A partir da ilustração acima, verifica-se o implemento de um divisor de tensão representado pelos resistores R3 e R4 e o capacitor C1. Objetiva-se com esse circuito fazer o monitoramento da tensão da bateria para que possa ser feito o acompanhamento da porcentagem de carga, conforme descrito na seção 4.2.4. O circuito completo é ilustrado na Figura 36.

Figura 36 – Circuito Completo



Fonte: Autoria Própria (2019)

### 4.3.2 Montagem prática

Após a realização de simulações computacionais no software *Proteus Professional*, consoante comentado nas seções anteriores, projetou-se a montagem física do sistema. Para isso, gozou-se do uso dos componentes apresentados na Tabela 12.

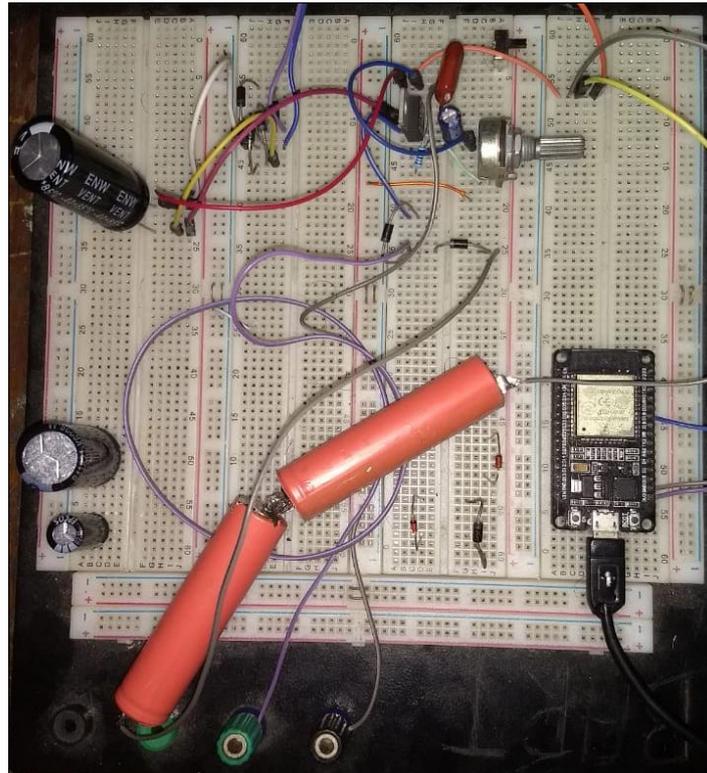
Tabela 12 – Lista de Componentes eletrônicos

Quantidade	Componentes	Valor
1 m	Cabeamento de fio de cobre	R\$ 4,00
1 u	Filamento PLA (250g)	R\$ 30,00
1 u	Alternador	R\$ 34,38
2 u	Sensores de Efeito Hall + Imãs	R\$ 4,00
2 u	Imãs	R\$ 5,00
1 u	Bateria Recarregável	R\$ 6,87
1 u	ESP32	R\$ 18,00
-	Componentes Eletrônicos	R\$ 9,20
1 u	Step Down	R\$ 2,67
1 u	TP4056	R\$ 4,94
1 u	Placa Fenolite	R\$ 6,60
6 u	Parafusos	R\$ 5,00
-	Total	R\$ 125,66

Fonte: Autoria Própria (2019)

Inicialmente foram realizados testes de funcionamento nos sensores, bem como, testes de eficiência no alternador. De forma progressiva, adicionou-se circuitos adjacentes, como, os conversores e divisor de tensão. Após a finalização da montagem do circuito eletrônico e da validação a partir de testes de funcionamento, o passo seguinte foi produzir a placa de circuito impresso. Ilustra-se na Figura 37 a montagem física na placa de testes *protoboard*.

Figura 37 – Montagem do circuito



Fonte: Autoria Própria (2019)

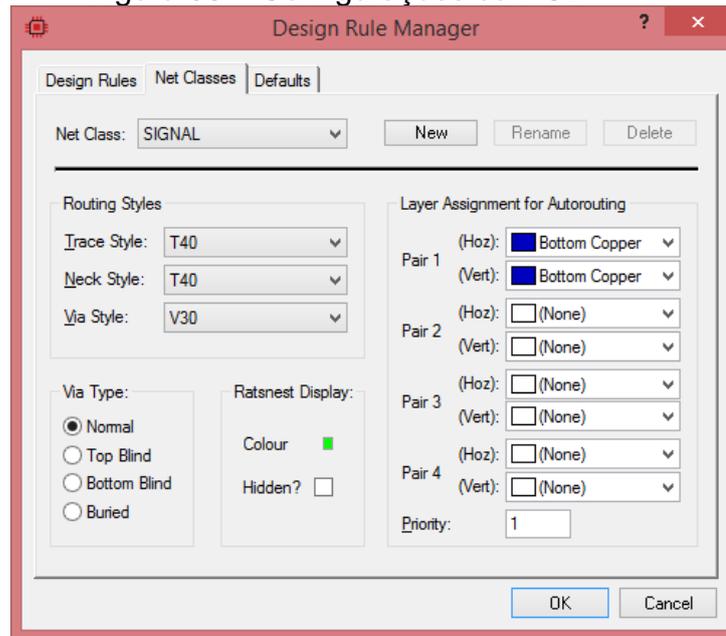
#### 4.3.3 Confeção da Placa de Circuito Impresso (PCI)

A placa de Circuito Impresso, projetada em ambiente ISIS-Proteus, foi desenvolvida a partir do circuito demonstrado na Figura 36.

Em ambiente ARES-Proteus, necessitou-se criar componentes referentes ao módulo Step-Down, TP4056 e bateria, visto que o ARES não possui esses componentes em sua biblioteca.

Após montar-se o esquemático do circuito no ISIS do Proteus, projetou-se o layout da placa de circuito impresso no toolbox ARES. Para isso, necessitou-se fazer algumas configurações no comando *Design Rule Manager*, onde foram definidas as configurações apresentadas na Figura 38, de acordo com as necessidades do projeto.

Figura 38 – Configurações da PCB

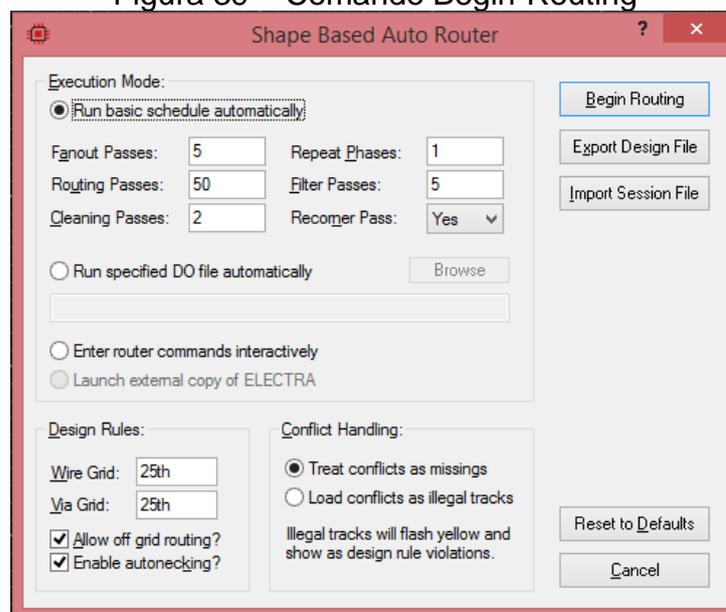


Fonte: Autoria Própria (2019)

Após configurar-se as trilhas, foi feita a projeção do circuito de forma automática, utilizando-se o comando *auto-router*, clicando-se no botão *Begin Routing*,

Figura 39.

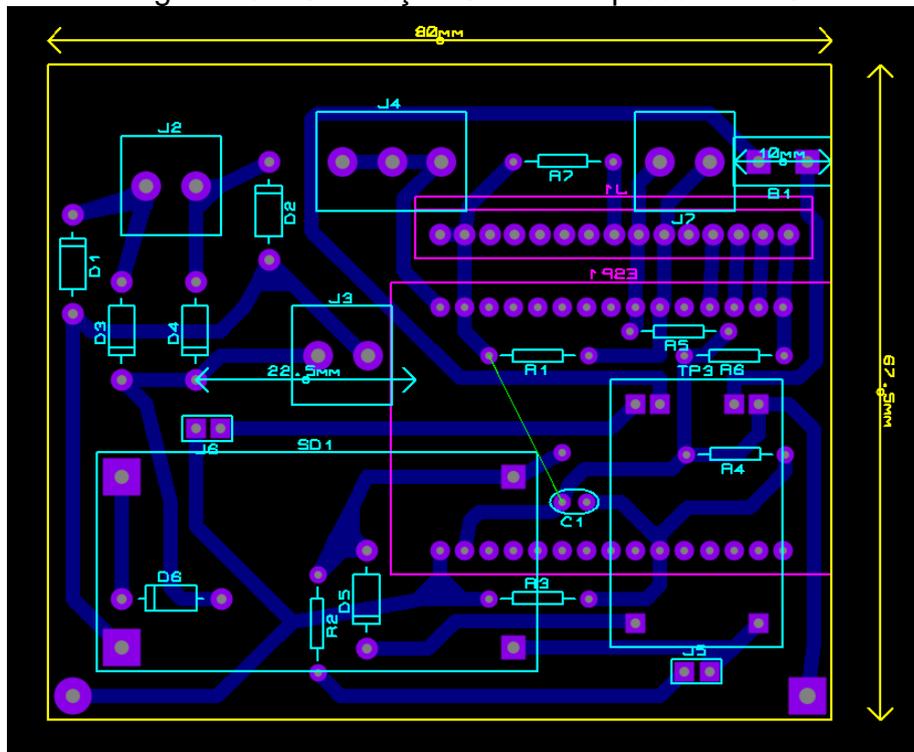
Figura 39 – Comando Begin-Routing



Fonte: Autoria Própria (2019)

Após a emulação do circuito através das etapas anteriores, chegou-se ao layout do circuito impresso representado na Figura 40.

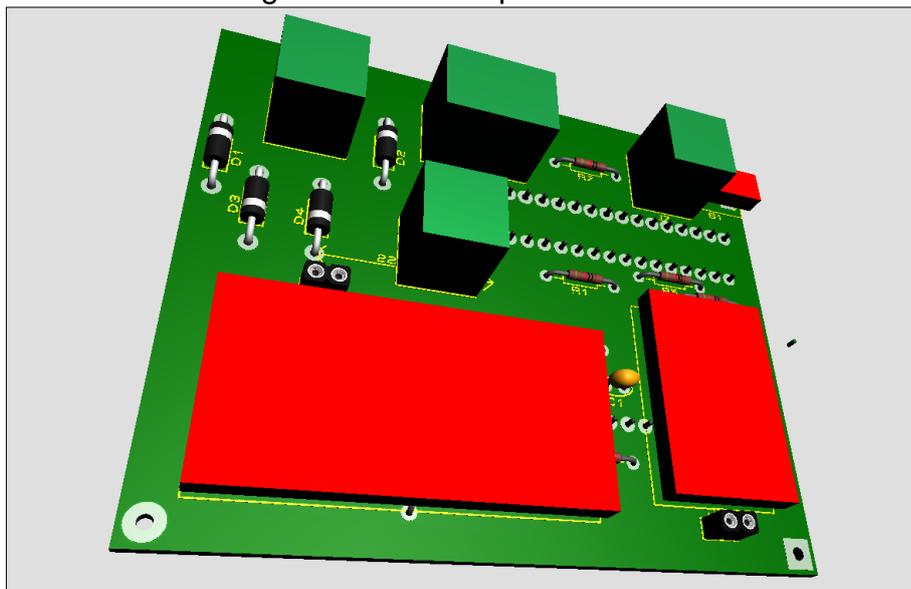
Figura 40 – Simulação Circuito Impresso ARES



Fonte: Autoria Própria (2019)

Ilustra-se na Figura 41 a projeção da placa em 3 dimensões, com realidade aproximada da placa de circuito impresso produzida.

Figura 41 – Protótipo 3D da PCI



Fonte: Autoria Própria (2019)

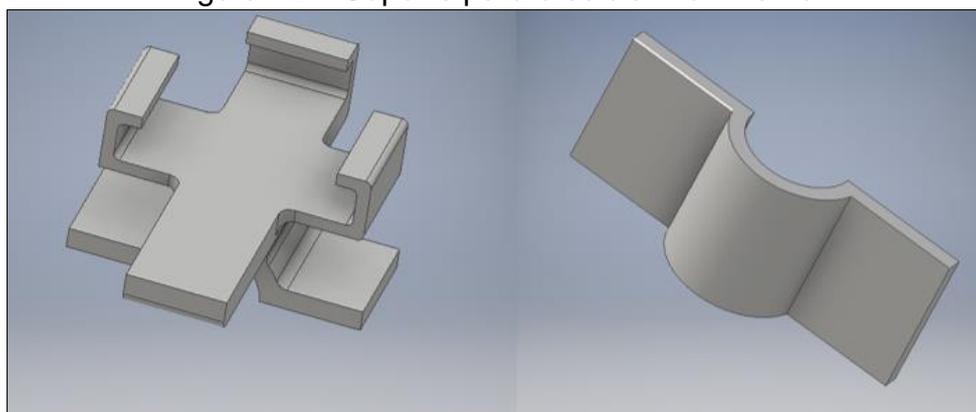
## 4.4 DESCRIÇÃO DO PROJETO MECÂNICO

O desenvolvimento das peças foi realizado no *software Inventor Professional*, conforme descrito na seção 4.1.2. Objetivou-se produzir suportes de fixação à bicicleta para placa de circuito impresso, sensores, celular e peças auxiliares.

### 4.4.1 Suporte para o celular

Desenvolveu-se uma peça de apoio ao celular a fim de que o mesmo pudesse ser fixado à bicicleta. Demonstra-se na Figura 42 o desenho desenvolvido no *software* para impressão 3D.

Figura 42 – Suporte para o celular no inventor



Fonte: Autoria Própria (2019)

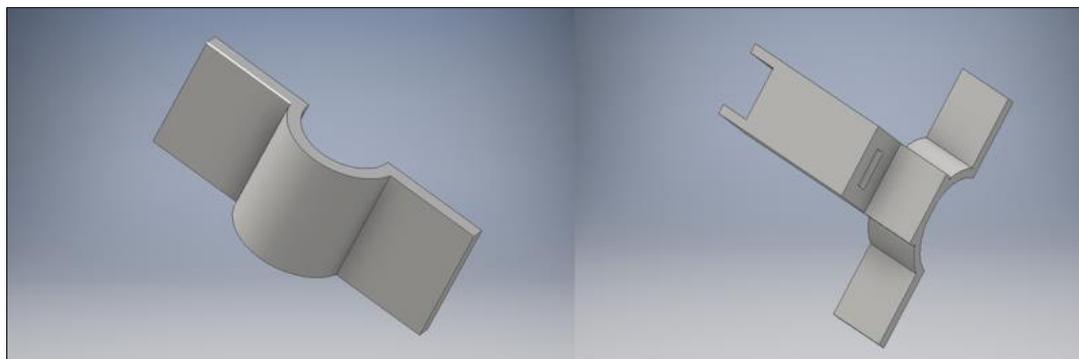
Depois de impresso, colocou-se a peça na bicicleta no guidão e parafusou-se. Para isso, foram utilizados 4 parafusos (Parafuso r. maq. cab. redonda #5-40x1/4 304 (a2)). Logo após todo processo, colocou-se o celular de modelo moto geração 5 plus, que contém espessura de 144.4 x 72 x 9.9 mm.

### 4.4.2 Suporte do Sensor Reed Switch

A peça foi desenvolvida para o sensor magnético *Reed Switch*. Utilizou-se 2 parafusos (Parafuso r. maq. Cab. Redonda #5-40x1/4 304 (a2)) para fixação da peça no aro da bicicleta. Ilustra-se na Figura 43 o desenho da peça em 3 dimensões

desenvolvido no *software*. Exibe-se na Figura 51, Anexo A, a peça impressa fixada ao quadro da bicicleta.

Figura 43 - Desenho no inventor do Sensor Reed Switch

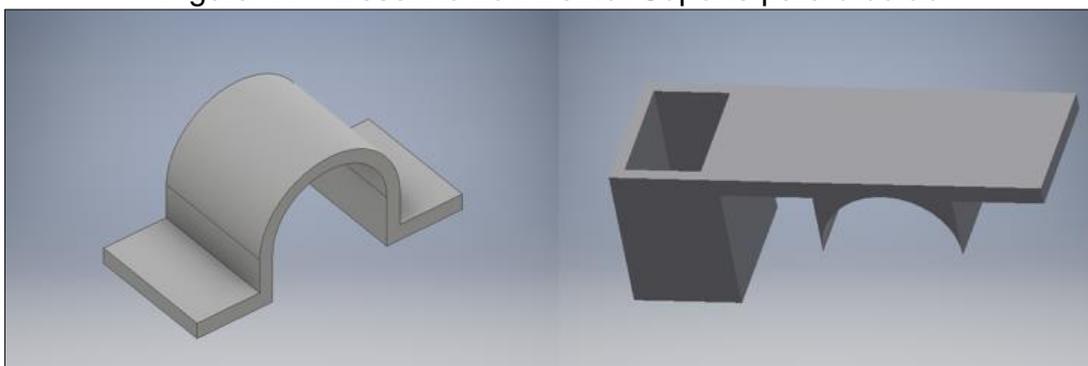


Fonte: Autoria Própria (2019)

#### 4.4.3 Suporte para o botão

Desenvolveu-se uma caixa para apoio do botão de ligação geral do computador de bordo. Anexou-se o botão no suporte e depois se colocou o suporte no lado esquerdo do guidão da bicicleta, entre a troca de marchas e o apoio da mão, parafusando-se a peça com 2 parafusos (Parafuso r. maq. cab. redonda #5-40x1/4 304 (a2)), como mostra a Figura 53. Demonstra-se na Figura 44 o desenho desenvolvido no *software* Inventor.

Figura 44 – Desenho no inventor Suporte para o botão



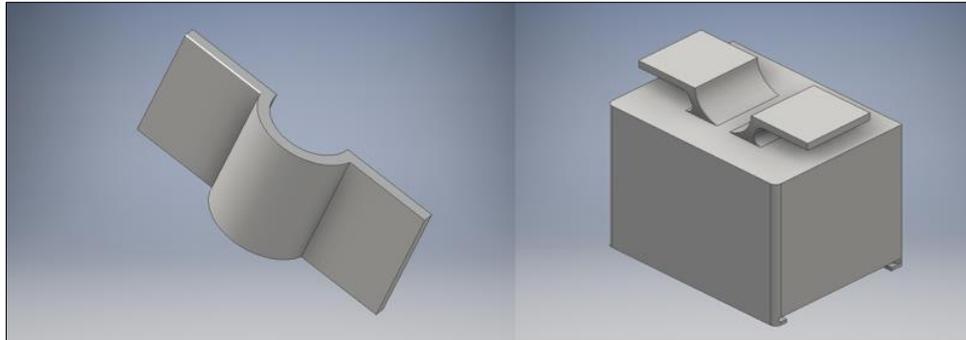
Fonte: Autoria Própria (2019)

#### 4.4.4 Caixa da PCI

A Figura 45 ilustra a caixa da PCI desenvolvida no *software* Inventor. Depois de impressa, acoplou-se ao garfo. Para tal, utilizou-se 4 parafusos (Parafuso r. maq.

cab. redonda #5-40x1/4 304 (a2)) para fixar a peça. Exibe-se na Figura 54 a posição da caixa para a PCI fixada ao quadro da bicicleta.

Figura 45 – Desenho da caixa para PCI no inventor



**Fonte:** Autoria Própria (2019)

## 5 CONSIDERAÇÕES FINAIS

O projeto do computador de bordo para bicicleta atende às necessidades do usuário de controlar sua atividade na bicicleta, visto que, o sistema embarcado apresenta informações sobre a velocidade instantânea, a quilometragem total da bicicleta, o gasto calórico gerado durante a viagem e monitora a carga da bateria do sistema. Além disso, possibilita que o usuário utilize seu aparelho *smartphone* como interface homem máquina, diminuindo o custo do projeto.

O processador utilizado no sistema dispõe de um processamento em alta velocidade, possibilitando ao usuário a leitura das informações em tempo real. Além disso, os dois núcleos de processador, que dividem a execução de tarefas, torna o sistema mais robusto, o que diminui a probabilidade de travamento e bugs do sistema.

De modo análogo, o aplicativo mobile desenvolvido para celular, feito através do SDK Flutter, possui alta qualidade de desempenho visto que foi programado através de código nativo da plataforma Android. Com isso, o app possui robustez na execução do *firmware*, garantindo o bom desempenho de funcionamento.

Além disso, o sistema de alimentação do computador de bordo, traz ao usuário a comodidade e economia financeira ao desprezar o uso de matrizes energéticas externas para recarregar a bateria, tornando o usuário um contribuinte com a sustentabilidade ambiental.

Para trabalhos futuros e aprimoramento da aplicação torna-se interessante a inclusão de luzes de alerta e iluminação, buzina, sensores de temperatura, humidade e força.

Sugere-se, também, a utilização do Wi-Fi do ESP32 para armazenamento de informações em banco de dados. Além disso, fica como sugestão a troca de componentes PTH por componentes SMD na placa de circuito impresso, a fim de que haja diminuição no tamanho físico da placa de controle.

## REFERÊNCIAS

SILVA, Ana Bastos; SILVA, João Pedro. **A bicicleta como modo de transporte sustentável**. Universidade de Coimbra e Instituto Politécnico de Leiria, v. 7, 2005.

MINISTÉRIO DAS CIDADES, 2007<sup>a</sup>; **PlanMob – Caderno de Referência para Elaboração de Plano de Mobilidade Urbana, Secretaria Nacional de Transporte e da Mobilidade Urbana**; Brasília, 2007.

ABRACICLO, 2017; **Indústria de bicicletas inicia 2018 com aumento de 49,8% na produção**; São Paulo, 2018.

DA SILVEIRA, Mariana Oliveira. **Mobilidade Sustentável: A bicicleta como um meio de transporte integrado**. 2010. Tese de Doutorado. Universidade Federal do Rio de Janeiro.

SILVA, V. M. L; MURADÁS, J. L. M; BARBOSA, L. F. W. **Circuito regenerador de energia elétrica para a recarga de baterias de um futuro triciclo híbrido movido a tração humana e elétrica**. São José dos Campos, 2007.

PATRICIO, Luis Claudio Brito; KRUSZIELSKI, Leandro. Dia de Bicicleta ao Trabalho: Uma Potencial Ferramenta para Planejamento e Promoção da Mobilidade Sustentável. **Revista de Gestão Ambiental e Sustentabilidade-GeAS**, v. 5, n. 3, p. 135-151, 2016.

CAVILL, N., et al., 2009, “**Economic analyses of transport infrastructure and policies including health effects related to cycling and walking: A systematic review**”, *Transport Policy*, doi: 10.1016/j.tranpol.2008.11.001.

AITA, Ricardo Hahn. **Sistema de irrigação localizada e automatizada**. Pontifícia Universidade Católica do Rio Grande do Sul, 2017

GIGLIO, G. P. M et al., 2018. **Desenvolvimento de um protótipo de solução mobile para acessibilidade e inclusão escolar para deficientes visuais**.

TURATTI, A. et al. **Relatório Técnico Help**. Universidade Tecnológica Federal do Paraná. (2018, pg. 23)

REDAÇÃO INDÚSTRIA. Disponível em: <<https://industria hoje.com.br/o-que-e-um-alternador-automotivo>> Acesso em: 12 de julho de 2019.

PANASONIC. Datasheet da bateria NCR18650. Disponível em: <<https://engineering.tamu.edu/media/4247819/ds-battery-panasonic-18650ncr.pdf>> Acesso em: junho de 2019.

WENDLING, M., 2010. **Sensores**. Universidade Estadual Paulista – UNESP, 2010.

FREITAS, L. M.; JÚNIOR, I. S. Q. **Prototipagem e construção de um acoplador para óculos com comunicação bluetooth para fins de automação industrial e transmissão de dados**.

PETRY, C. A. **Metodologia de projeto de retificadores com filtro capacitivo**. Revista Ilha Digital, Instituto Federal de Santa Catarina, 2012. Disponível em <<http://ilhadigital.florianopolis.ifsc.edu.br/>> Acesso em: Julho de 2019.

Bluetooth SIG. (2013). Specification of Bluetooth System (Covered Core Package version: 4.0) [Online]. Página disponível em: <https://www.bluetooth.org/en-us/specification/adopted-specifications>.

MAIO, A. J. F. **Bluetooth Low Energy para Monitorização da Postura no Ciclismo**. Universidade do Minho, Braga, Portugal, 2014.

RECK, M. S. **Beacons BLE – BLUETOOTH LOW ENERGY – Design e análise de um sistema de localização indoor**. Universidade de Caxias do Sul, 2016.

LAIRD. Disponível em: <http://www.summitdata.com/blog/ble-overview/>. Acesso em: 19 maio 2019.

STROSKI, P. N. Disponível em: <http://www.electricalibrary.com/2018/02/25/conheca-o-reed-switch/>. Acesso em: 01 de Julho de 2019.

QQTRADING. Disponível em: <http://qqtrading.com.my/battery-charger-tp4056-w-discharge-protection-for-single-cell-li-po-18650>. Acesso em: 26 de junho de 2019.

ISMAYATIM. Disponível em: <http://minielectricscooter.blogspot.com/2012/02/>. Acesso em: 26 de junho de 2019.

ALIEXPRESS. Disponível em: [https://pt.aliexpress.com/item/32965972569.html?spm=a2g03.search0302.3.35.48bd104b7FgULn&ws\\_ab\\_test=searchweb0\\_0%2Csearchweb201602\\_0\\_10084\\_10083\\_10547\\_10887\\_10307\\_321\\_453\\_10548\\_322\\_454\\_10618\\_536\\_10065\\_317\\_537\\_10068\\_319\\_10059\\_10103\\_10884\\_10696%2Csearchweb201603\\_0%2CppcSwitch\\_0&algo\\_pvid=9d0066c1-015d-4873-bcd2-37285152a459&algo\\_expid=9d0066c1-015d-4873-bcd2-37285152a459-4](https://pt.aliexpress.com/item/32965972569.html?spm=a2g03.search0302.3.35.48bd104b7FgULn&ws_ab_test=searchweb0_0%2Csearchweb201602_0_10084_10083_10547_10887_10307_321_453_10548_322_454_10618_536_10065_317_537_10068_319_10059_10103_10884_10696%2Csearchweb201603_0%2CppcSwitch_0&algo_pvid=9d0066c1-015d-4873-bcd2-37285152a459&algo_expid=9d0066c1-015d-4873-bcd2-37285152a459-4). Acesso em: 05 de Julho de 2019.

GETTYIMAGES. Disponível em: <https://www.gettyimages.pt/detail/ilustração/bicycle-ilustração-royalty-free/>. Acesso em 01 de julho de 2019.

TEXAS INSTRUMENTS. Disponível em: <http://www.ti.com/lit/ds/symlink/lm2596.pdf>. Acesso em: 11 de julho de 2019.

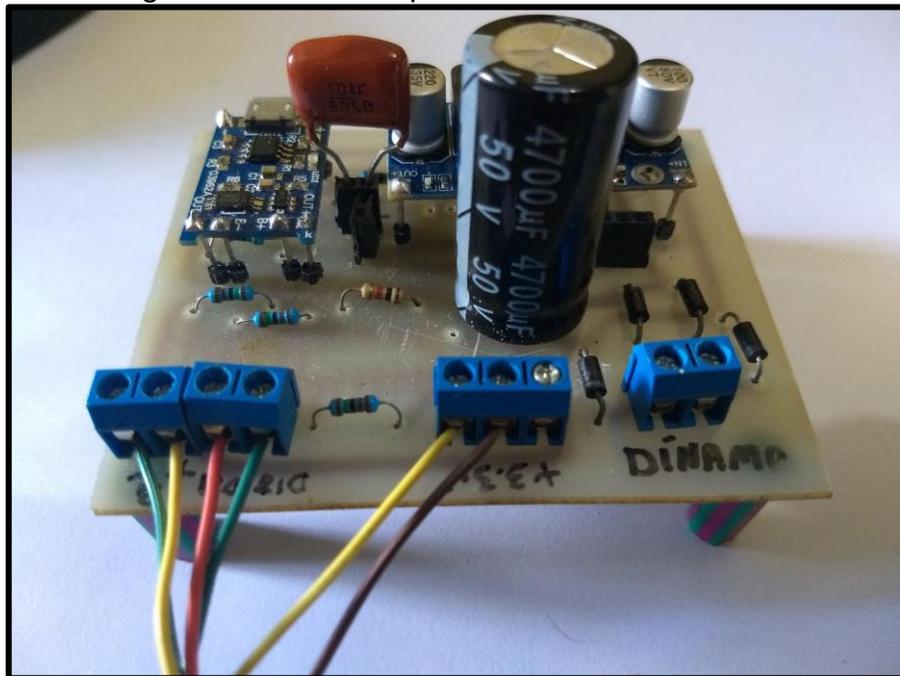
---

FILHO, A. J. C. **Automação via browser de rede industrial modular serial rs485 conectada via usb à servidor web.** Centro Universitário de Maringá – UNICESUMAR, 2018.

BELASQUEM, B. R. et al. **Retificador de onda completa com dois diodos e ponte de dois diodos.** Faculdade Estácio de Curitiba, 2018.

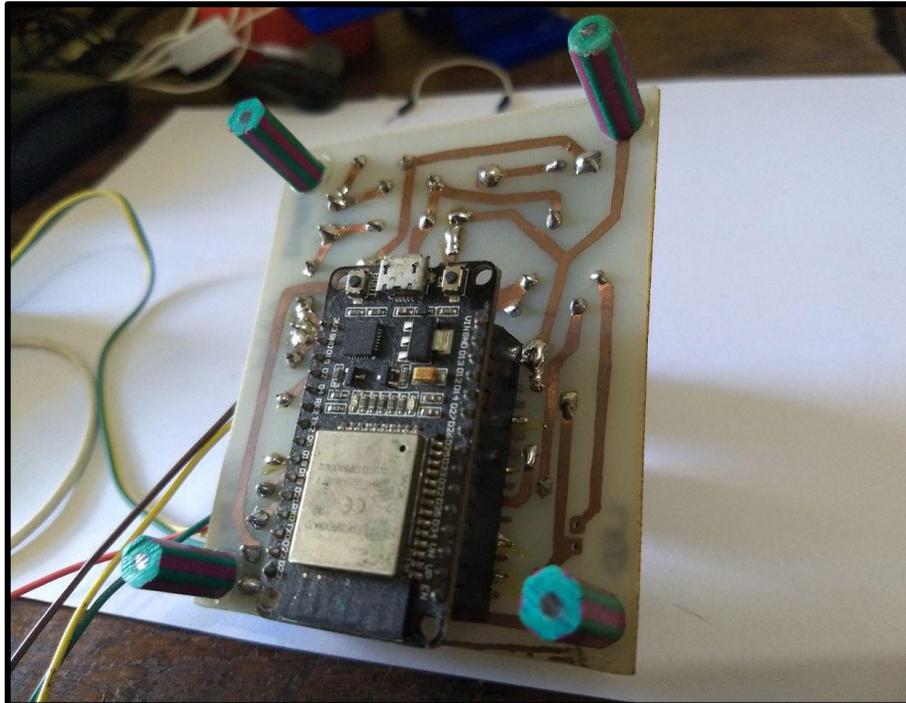
## APÊNDICE A – Elementos Elétricos do Computador de Bordo

Figura 46 – Visão superior da Placa de Controle



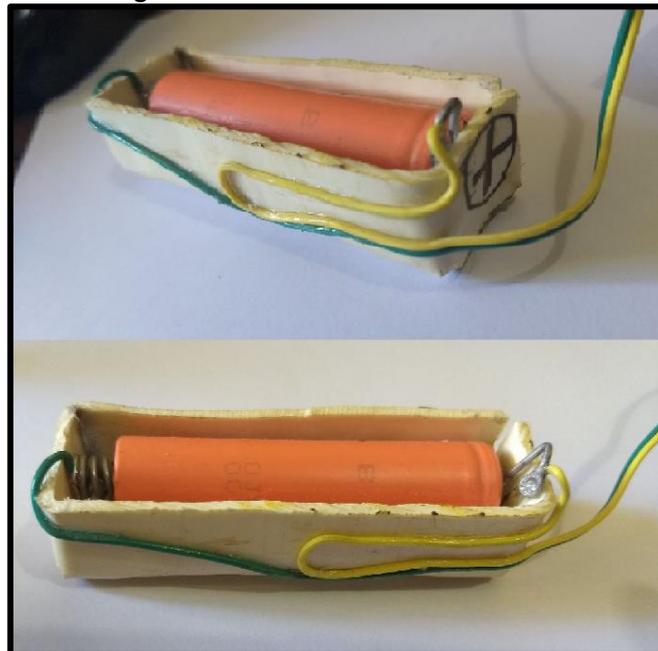
Fonte: Autoria Própria (2019)

Figura 47 - Visão inferior da Placa de Controle



Fonte: Autoria Própria (2019)

Figura 48 – Bateria de Lítio



Fonte: Autoria Própria (2019)

Figura 49 - Alternador



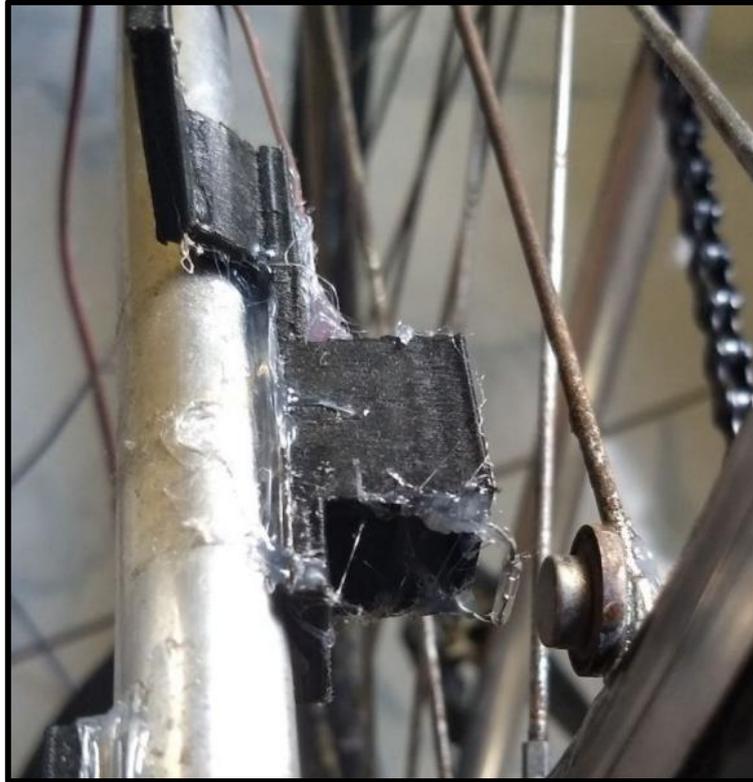
Fonte: Aatoria Própria (2019)

Figura 50 – Sensor de efeito Hall paralelo ao Pedal



Fonte: Aatoria Própria (2019)

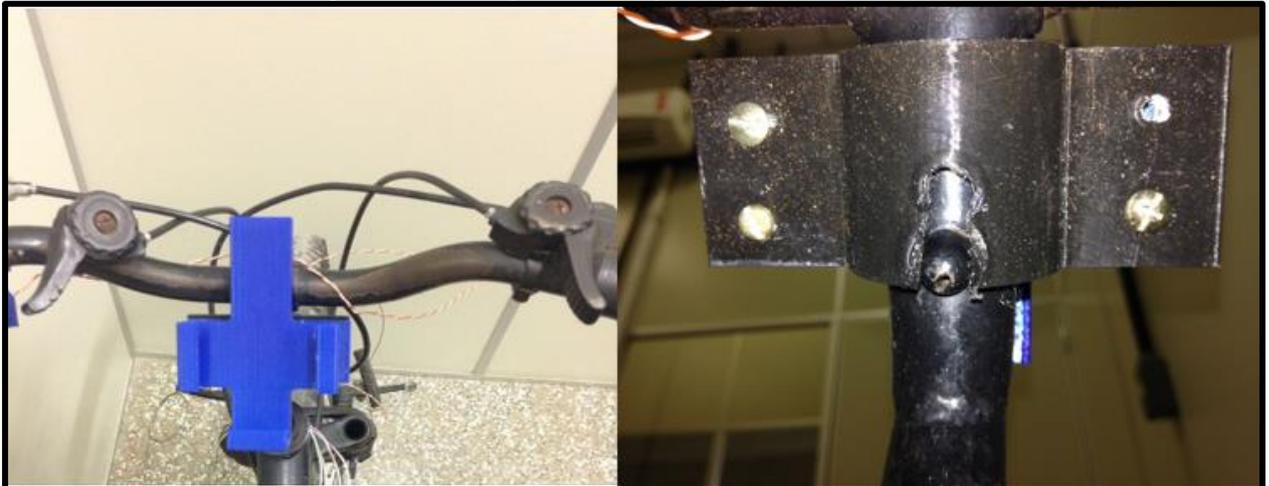
Figura 51 – Sensor de Efeito Hall paralelo ao Pneu



Fonte: Aatoria Própria (2019)

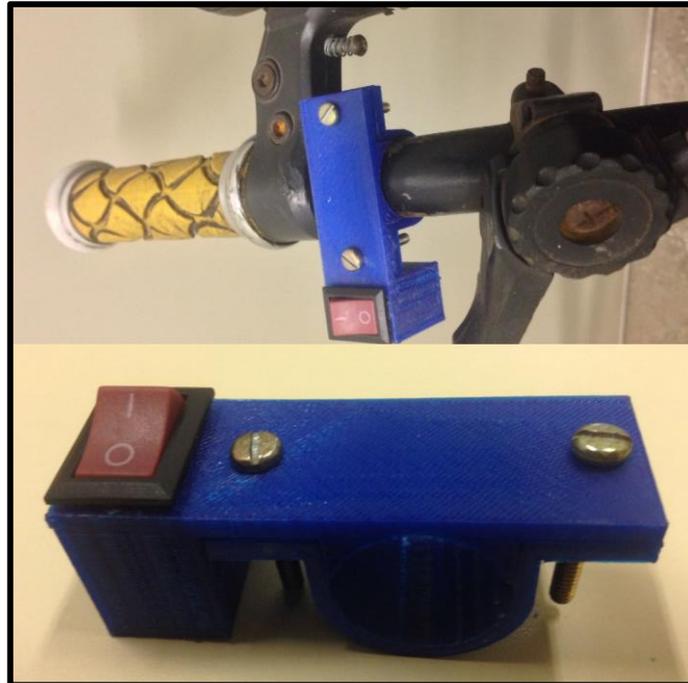
## APÊNDICE B – Peças mecânicas

Figura 52 – Suporte de apoio para o celular



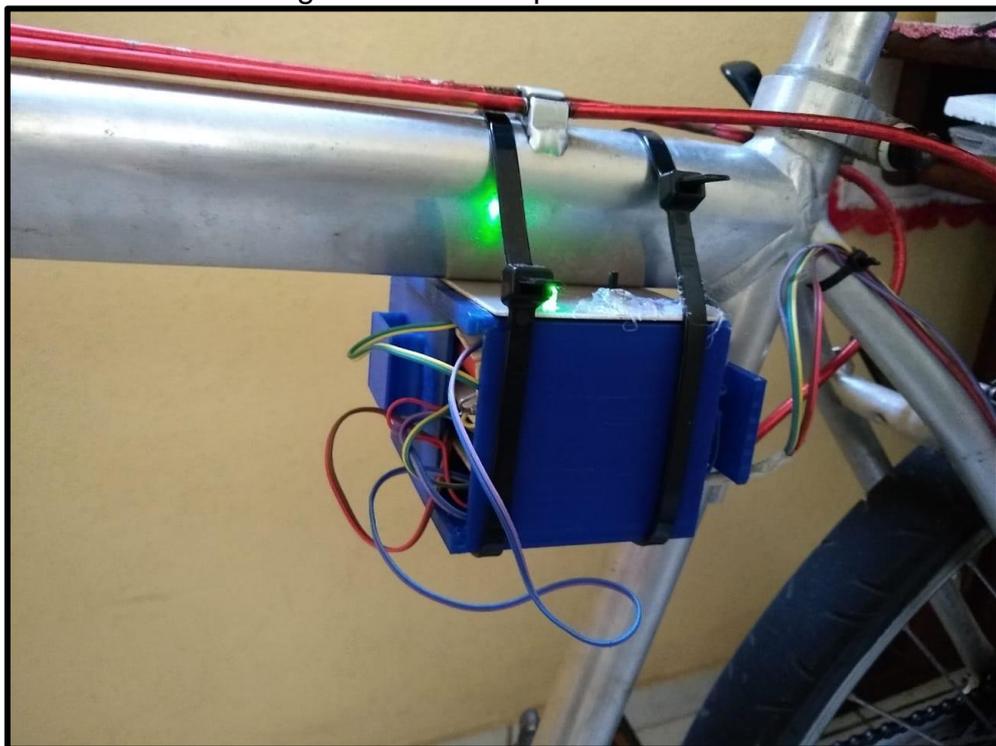
Fonte: Aatoria Própria (2019)

Figura 53 – Suporte para o botão



Fonte: Aatoria Própria (2019)

Figura 54 – Caixa para PCI



Fonte: Aatoria Própria (2019)

Figura 55 – Computador de bordo anexo à bicicleta.



Fonte: Autoria Própria (2019)