

INSTITUTO FEDERAL DE SANTA CATARINA

Dionata Fermino

Luiz Felipe de Souza

CONSTRUÇÃO DE UM KIT DIDÁTICO COM RASPBERRY PI 3B PARA O ENSINO  
DE AUTOMAÇÃO

Joinville

2022

Dionata Fermino

Luiz Felipe de Souza

CONSTRUÇÃO DE UM KIT DIDÁTICO COM RASPBERRY PI 3B PARA O ENSINO  
DE AUTOMAÇÃO

Monografia apresentada ao  
Curso Superior de  
Tecnologia em Mecatrônica  
do Campus de Joinville do  
Instituto Federal de Santa  
Catarina para a obtenção  
do diploma de Tecnólogo  
em Mecatrônica Industrial.

Orientador: Michael Klug

Joinville

2022

Fermino, Dionata.

Construção de um kit didático com Raspberry PI 3B para o ensino de automação / Dionata Fermino, Luiz Felipe de Souza. – Joinville, SC, 2022.

39 p.

Trabalho de Conclusão de Curso (Graduação) - Instituto Federal de Educação Ciência e Tecnologia de Santa Catarina, Curso de Tecnologia em Mecatrônica Industrial, Joinville, 2022.

Orientador: Dr. Eng. Michael Klug.

1. Kit didático. 2. Raspberry PI. 3. Acessórios eletrônicos. I. Souza, Luiz Felipe de. II. Instituto Federal de Educação Ciência e Tecnologia de Santa Catarina. III. Título.

Dionata Fermino  
Luiz Felipe de Souza

CONSTRUÇÃO DE UM KIT DIDÁTICO COM RASPBERRY PI 3B PARA O ENSINO  
DE AUTOMAÇÃO

Este trabalho foi julgado adequado para obtenção do título em Tecnólogo em  
Mecatrônica Industrial, pelo Instituto Federal de Educação, Ciência e Tecnologia de  
Santa Catarina, e aprovado na sua forma final pela comissão avaliadora  
abaixo indicada.

Joinville, 20 de Dezembro de 2022.

---

Prof. Michael Klug, Dr. Eng.

Orientador

Instituto Federal de Santa Catarina Campus Joinville

---

Prof. José Flavio Dums, Dr. Eng.

Instituto Federal de Santa Catarina Campus Joinville

---

Prof. Rubens Hesse, Dr. Eng.

Instituto Federal de Santa Catarina Campus Joinville

Dedicamos este trabalho a nossas famílias e amigos,  
que sempre estiveram nos apoiando nessa etapa de nossas vidas,  
acreditando e incentivando e nunca duvidando de nossas capacidades,  
e que agora colheram junto conosco os frutos de todo esse esforço.

## AGRADECIMENTOS

Agradecemos a nossas famílias que nos deram todo o suporte necessário para nosso crescimento durante toda essa etapa das nossas vidas e sem eles ao nosso lado, não teríamos chegado até o fim deste trabalho.

Agradecemos aos nossos amigos que estiveram presentes, e aos professores do IFSC, em especial ao nosso orientador Michael Klug, que nos instruiu no desenvolvimento deste trabalho.

## RESUMO

Este trabalho apresenta estudo teórico e prático da construção de um kit didático, que utiliza como base uma placa de Raspberry Pi construído em uma estrutura de acrílico, com entradas e saídas através de bornes com engate rápido para facilitar o manuseio, além de conexões USB, HDMI, ethernet e uma interface LCD *touch screen* que deverá auxiliar no desenvolvimento de projetos.

O kit foi desenvolvido para pessoas que tenham ou não conhecimentos em linguagem de programação, auxiliando novos estudantes das áreas de mecatrônica, engenharia elétrica e afins, possam criar e executar atividades como programar e controlar dispositivos diversos como relés, *LEDs*, sensores, botões, etc, em um único dispositivo.

Palavras Chaves: *Kit Didático, Raspberry Pi, Acessórios eletrônicos, Linguagem Python.*

## **ABSTRACT**

This work presents a theoretical and practical study of the construction of a didactic kit, which uses as a base a Raspberry Pi board built in an acrylic structure, with inputs and outputs through terminals with quick connection to facilitate handling, in addition to USB connections, HDMI, ethernet and an LCD touch screen interface that should help in project development.

The kit was developed for people with or without knowledge of programming language, helping new students in the areas of mechatronics, electrical engineering and the like, to create and execute activities such as programming and controlling various devices such as relays, LEDs, sensors, buttons, etc., on a single device.

**Keywords:** Didactic Kit, Raspberry Pi, Electronic accessories, Python language.



## LISTA DE ILUSTRAÇÕES

Figura 1 - Tipos de pinos do GPIO encontrado no Raspberry.....	15
Figura 2 – Detalhes dos componentes da placa Raspberry PI 3B. ....	16
Figura 3 - Funcionamento do sistema resistivo <i>Touch Screen</i> .....	19
Figura 4 - Base estrutural em acrílico do kit em desenvolvimento. ....	21
Figura 5 - Dimensões totais e visualização do pré-projeto do kit didático. ....	22
Figura 6 - Placa Raspberry Pi 3B. ....	23
Figura 7 - Tela <i>Touch Screen</i> 7".....	23
Figura 8 - Borne de Pressão 4 Vias. ....	24
Figura 9 – Tela Touch Screen, Bornes de Pressão e Caneta. ....	24
Figura 10 – Conector Sindal e Prensa Cabos $\frac{3}{4}$ .....	25
Figura 11 – Fonte Ajustável de 5V. ....	25
Figura 12 – Cabos HDMI.....	26
Figura 13 – Cabo de Alimentação e Botão Liga/Desliga.....	26
Figura 14 – Cabos unifilar e Jumpers.....	27
Figura 15 – Etiquetas de identificação das GPIO nos bornes de pressão do kit didático.....	27
Figura 16 – Caixa de testes do kit didático.....	28
Figura 17 – Vista frontal da montagem concluída do kit didático.....	28
Figura 18 – Kit didático em funcionamento.....	29
Figura 19 - Diagrama Elétrico de alimentação do Kit. ....	29
Figura 20 – Lógica de programação para acionamento dos 4 leds por botões físicos. ....	31
Figura 21 – Setup do acionamento dos 4 leds por botões físicos. ....	32
Figura 22 – <i>Layout</i> do <i>software</i> de criação da <i>interface 4 leds</i> na tela <i>touch screen</i> . ....	33
Figura 23 – Lógica de programação para acionamento dos 4 leds pela <i>tela touch screen</i> . ....	33
Figura 24 – Setup do acionamento dos 4 leds pela tela <i>touch screen</i> . ....	34
Figura 25 - Placa de circuito com 16 optoacopladores.....	35
Figura 26 - Optoacoplador Bornes de entrada. ....	35
Figura 27 - Optoacoplador Bornes de saída.....	36

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>12</b>
<b>1.1</b>	<b>Objetivos .....</b>	<b>13</b>
1.1.1	Objetivo geral .....	13
1.1.2	Objetivo específico .....	13
<b>2</b>	<b>REVISÃO DA LITERATURA.....</b>	<b>14</b>
2.1	Raspberry PI .....	14
2.1.1	O que é.....	14
2.1.2	GPIO.....	15
2.1.3	Acessórios .....	16
2.1.4	Aplicações da placa .....	16
2.2	Linguagem de Programação .....	17
2.2.1	Definição .....	17
2.2.2	Python .....	18
2.2.3	C++ .....	18
2.3	Componentes e Acessórios.....	18
2.3.1	Tela Touch Screen .....	18
2.3.2	Fonte.....	20
2.3.2.1	Fontes de alimentação CA/CC .....	20
2.3.2.2	Tipos de Fontes.....	20
<b>3</b>	<b>DESENVOLVIMENTO DO KIT DIDÁTICO .....</b>	<b>21</b>
3.1	Estrutura do Kit.....	21
3.2	Lista de Componentes .....	22
3.3	Diagrama Elétrico do Kit Didático .....	29
3.4	Custo do Projeto.....	30
<b>4</b>	<b>DEMONSTRAÇÃO DO FUNCIONAMENTO DO KIT DIDÁTICO .....</b>	<b>30</b>
4.1	Exemplo 1.....	30
4.1.1	Programação .....	30
4.1.2	Funcionamento.....	31
4.2	Exemplo 2.....	32
4.2.1	Programação .....	32

4.2.2	Funcionamento.....	34
<b>5</b>	<b>SUGESTÃO DE MELHORIA.....</b>	<b>34</b>
5.1	Circuito de Proteção contra surtos nas GPIO .....	34
<b>6</b>	<b>CONCLUSÃO.....</b>	<b>37</b>
	<b>REFERÊNCIAS.....</b>	<b>38</b>

## 1 INTRODUÇÃO

Estamos na era da globalização e da informatização, um dos traços característicos da nossa época é a velocidade das transformações sociais e da criação de novas tecnologias que geram avanço da sociedade como um todo ou até mesmo na simples atualização de equipamentos e máquinas utilizadas no dia a dia. Enfim, as novas tecnologias e seu impacto na vida das pessoas acabam sendo um fio condutor importante para interpretar os avanços no desenvolvimento social e, mais ainda, na qualidade de vida, que é considerada uma área multidisciplinar por excelência (GUTIERREZ e ALMEIDA, 2006).

E quando se fala em tecnologia as automatizações andam juntas, sendo automatização a execução automática de tarefas sem intervenção humana, fazendo com que trabalhos sejam mais fáceis, sem a necessidade de execução manual, abrindo caminho para padronização de processos os deixando mais eficientes, além de aperfeiçoar a gestão do tempo, concentrando esforços operacionais em outras áreas e criando profissionais mais capacitados.

Sendo assim, o objetivo com este trabalho é a criação de um kit didático que seja versátil em suas aplicações, e auxilie na execução de pessoas que desejam criar e executar atividades como programar e controlar dispositivos diversos como relés, *LEDs*, sensores, botões em um único dispositivo, facilitando a interação com novas tecnologias.

## 1.1 Objetivos

### 1.1.1 Objetivo geral

O presente trabalho objetiva desenvolver através de uma placa de Raspberry PI, um kit didático, para auxiliar na aprendizagem de pessoas que desejam programar, controlar e automatizar algum equipamento.

### 1.1.2 Objetivo específico

- A. Apresentar aplicações didáticas utilizando a placa Raspberry PI como plataforma.
- B. Dimensionar todos os equipamentos e componentes, que serão utilizados no kit didático.
- C. Simular algumas funcionalidades utilizando a plataforma Raspberry PI.
- D. Apresentar um protótipo de kit didático, que otimiza e facilita na aplicação dessas funcionalidades.

## 2 REVISÃO DA LITERATURA

Este capítulo tem como objetivo realizar uma abordagem introdutória sobre os componentes, *softwares* e métodos aplicados, para o desenvolvimento do kit didático, descrevendo definições e exemplos de funcionamento do mesmo. Essa análise busca contextualizar a parte teórica e estruturar o desenvolvimento do projeto.

### 2.1 Raspberry Pi

Em 2006, professores e pesquisadores da Universidade de Cambridge, no Reino Unido, se reuniram para criar um computador mais acessível e portátil que os outros disponíveis. Mas apenas em fevereiro de 2012, após 6 anos de pesquisa e implementação, que ocorreu o lançamento do primeiro Raspberry Pi.

Nos anos subsequentes, a Fundação Raspberry Pi continuou a melhorar seus produtos e a lançar novos microcomputadores, cada vez menores e mais potentes.

#### 2.1.1 O que é

Segundo (RASPBERRYPI, 2021) Raspberry Pi é um computador do tamanho de um cartão de crédito e que possui funcionalidades semelhantes a de um computador pessoal, sendo portátil embora menos potente. Fisicamente é uma placa pequena com entradas e saídas digitais e que possui todas as funcionalidades de um computador. O sistema operacional utilizado geralmente é o Linux, porém nos modelos mais novos já são capazes de utilizar sistemas operacionais mais comuns, como o Windows.

Como todo computador, nessa placa também é possível criar programas de diversas linguagens de programação sendo que os mais utilizados no momento são Python e C++, que podem também ser utilizadas para controlar entradas e saídas digitais (GPIO - *General Purpose Input/Output*).

## 2.1.2 GPIO

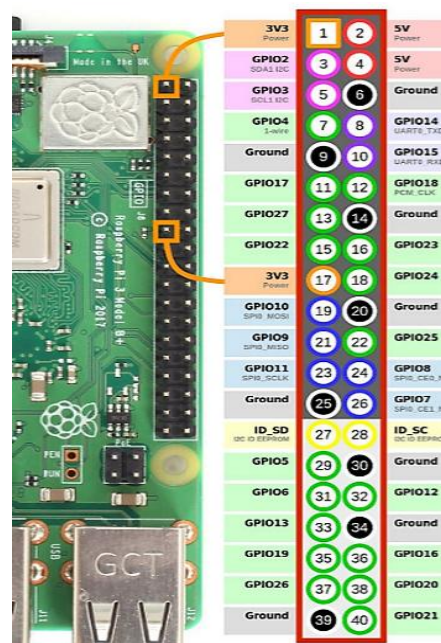
De acordo com Valeri (2022), GPIO (*General Purpose Input/Output*), é um conjunto de pinos que está presente em placas-mãe ou placas adicionais como, por exemplo, a do Raspberry Pi.

Diferente das portas *USB*, que possuem pinos/conexões com uma função específica determinada por um órgão regulador, o GPIO pode ter seus pinos utilizados de diferentes maneiras.

No Raspberry Pi, por exemplo, pode -se encontrar os seguintes tipos de pinos (Figura 1):

- Pinos que fornecem energia em voltagens típicas, como 3,3 V ou 5 V. Isso serve para alimentar dispositivos conectados que não possuem fonte de alimentação própria, como um simples *LED*.
- Pinos de aterramento que não fornecem energia, mas são necessários para completar alguns circuitos.
- Pinos GPIO, que podem ser configurados para enviar ou receber sinais elétricos.
- Pinos com propósito especial, que variam de acordo com o GPIO específico em questão.

Figura 1 - Tipos de pinos do GPIO encontrado no Raspberry.

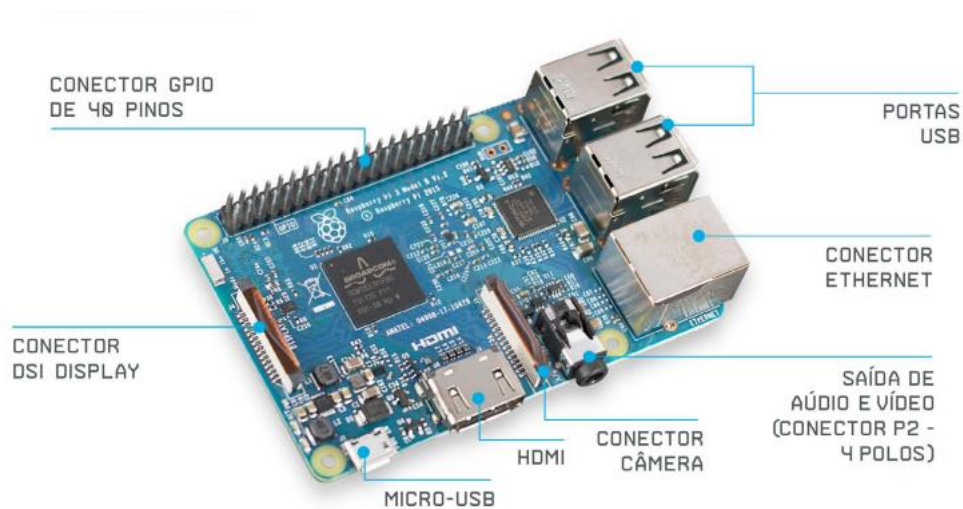


Fonte: raspberrypi-spy

### 2.1.3 Acessórios

Os acessórios mais utilizados são periféricos como teclado e mouse, podendo ser conectados a entradas *USB* localizados na lateral da placa, e para o caso do monitor utiliza-se a saída *HDMI*, como também possui entrada para cabo ethernet, saída P2 de áudio e vídeo, conector para display DSI, alimentação através de micro *USB* e conectores GPIO, como mostra a (Figura 2).

Figura 2 – Detalhes dos componentes da placa Raspberry PI 3B.



Fonte: <https://www.daeletrica.com.br/blog/2019/06/16/primeiros-passos-com-raspberry-pi-e-linux/>

Além destes, o microcomputador também suporta câmeras conectadas na porta (CSI - *Câmera serial interface*), e também suporta expansão *Gertboard*, que é uma placa eletrônica com pinos de entradas e saídas, que permite a interação do computador com sinais analógicos, por exemplo, *LEDs*, interruptores e sensores.

### 2.1.4 Aplicações da placa

O Raspberry Pi é um computador que possui diversas finalidades, segundo Mello (2022), este microcomputador tem como objetivo principal incentivar que pessoas realizem projetos de comunicação a treinarem programação com a placa. Geralmente é mais procurado por programadores e profissionais da computação, porém é importante lembrar que o Raspberry Pi é menos potente do que um computador completo.



Já na área de IoT (*Internet of Things*, que traduzindo para o português, Internet das Coisas) é outra que se beneficia muito do Raspberry Pi, pois possui acesso à internet, podendo utilizar o microcomputador como o centro de controle das operações dentro de uma casa, especialmente utilizando uma *Gertboard*, permitindo-o se comunicar diretamente com sensores e aparelhos inteligentes.

A seguir algumas aplicações que podem ser realizadas com placa Raspberry Pi:

- Ponto de acesso wireless, expandindo a capacidade da sua rede;
- Sistema de monitoramento com câmera;
- Media Center, reproduzindo músicas, vídeos e fotos;
- Estação meteorológica, enviando até mesmo os dados pela internet;
- Console de videogame “retrô”;
- Print Server wireless;
- Servidor de Arquivos / File Server;
- Web Server;
- Comandar Pinos GPIO, através e linguagem de programação;

## 2.2 Linguagem de Programação

### 2.2.1 Definição

Segundo Mendes (2022), linguagem de programação é um conjunto de regras e instruções que um programador constrói para gerar programas e *softwares* que serão processados por um computador, dispositivo móvel ou outro equipamento. Quando estruturada, essa linguagem forma um código fonte de um software, que informa a uma ferramenta quais ações ela deve realizar.

A utilização da linguagem de programação se popularizou, pois gera produtividade e facilidades para os usuários em relação ao uso do código de máquinas, que são as instruções que o computador compreende de fábrica.

A linguagem é composta por símbolos, palavras-chave, regras semânticas e sintáticas e são classificadas em níveis. Existem diversas linguagens de programação, sendo que as mais populares são Java Script, Python, Java, PHP, CSS, C#, C++ e C, porém as linguagens que foram utilizadas para exemplificar a funcionalidade do Kit foram Python e C++.

### 2.2.2 Python

Segundo Yamamoto (2017), Python é uma linguagem de programação de alto nível e de propósito generalista. Sua filosofia de design enfatiza a legibilidade do código com a proposta de ser de fácil compreensão com o uso de palavras-chave em inglês, que permite uma leitura fácil e gera menos linhas de código em comparação com outras linguagens.

Devido a essas características muitas coisas feitas em Python ficaram conhecidas como: Youtube, Google, Instagram, Dropbox, Pinterest e Spotify.

### 2.2.3 C++

C++ é uma linguagem de programação baseada na linguagem C, e segundo Pacievitch (2022), possui uma enorme variedade de códigos, pois além de seus códigos, pode contar com vários da linguagem C. Esta variedade possibilita a programação em alto e baixo níveis. O C++ apresenta grande flexibilidade, embora seja bom, este fato faz com que a programação seja muito mais cuidadosa para não terem erros.

Muitos dos mais conhecidos programas são feitos em C++, ou parte dos seus códigos são nessa linguagem, alguns deles são: Adobe Photoshop, MySQL, Mozilla Firefox, Internet Explorer, Microsoft Windows, entre vários outros.

## 2.3 Componentes e Acessórios

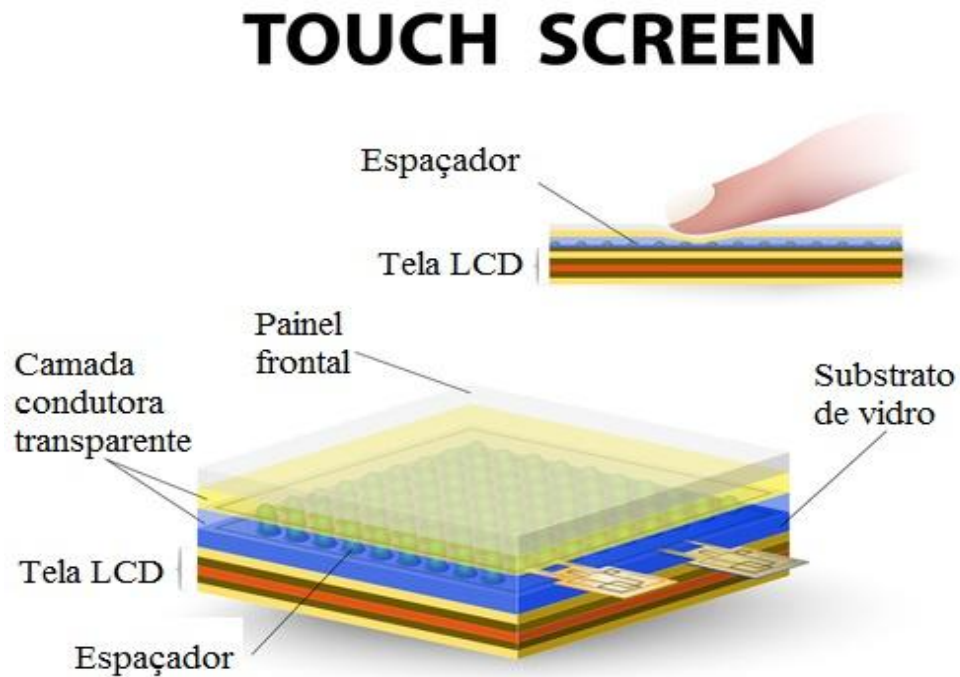
### 2.3.1 Tela Touch Screen

Segundo Teixeira (2022), *Touch Screen* significa tela sensível ao toque, é um *display* eletrônico capaz de detectar o toque em uma determinada posição sobre a tela. Hoje encontra-se esta tecnologia em quase todos os aparelhos eletrônicos como celulares, tvs, tablets, painéis de equipamento industriais, etc. Quando se refere a tipos de sistemas de funcionamento de *Touch Screen*, os mais conhecidos são os sistemas resistivo e o capacitivo.

De acordo com Quinafelex (2013), as telas resistivas possuem menor precisão no reconhecimento do toque, já que são compostas por várias camadas. A superfície frontal é de plástico, resistente a riscos, com material condutor do lado

interno. Abaixo, vem uma segunda camada que geralmente é de vidro, mas por vezes é feita de plástico rígido. Conforme pressionamos a tela, estas camadas fazem contato que informam ao sistema qual sua posição em coordenadas (Figura 3).

Figura 3 - Funcionamento do sistema resistivo *Touch Screen*.



Fonte: <https://brasilescola.uol.com.br/fisica/touch-screen.htm>

Segundo o mesmo Quinafelex (2013), as telas capacitivas são um tipo de tela que usa um capacitor, com duas placas separadas por um isolante (que pode ser o ar). Assim que o dedo toca esse tipo de tela, um capacitor monitorado pelo sistema informa exatamente onde foi acionado, através de uma troca de elétrons, e por causa dessa troca que canetas comuns não funcionam nesse tipo de tela, apenas as especiais e o próprio dedo.

Uma das grandes vantagens do display capacitivo é conseguir trabalhar com as informações de vários toques ao mesmo tempo, o que possibilita ao dispositivo reconhecer movimentos diferenciados, como a pinça. A precisão dessa tela pode chegar a 100% e é de alta durabilidade. As telas capacitivas estão presentes na maior parte dos dispositivos mais modernos, e costumam ter um tempo de resposta muito pequeno e uma usabilidade bem confortável.

## 2.3.2 Fonte

### 2.3.2.1 Fontes de alimentação CA/CC

São as fontes de alimentação da maioria dos dispositivos do dia a dia, como os carregadores de celulares e dispositivos eletrônicos. A fonte de alimentação converte a energia elétrica que pode se apresentar em tensão e/ou corrente alternada para tensão e/ou corrente contínua, ajustando na saída os parâmetros necessários à aplicação, segundo a Direct Industry (2020).

### 2.3.2.2 Tipos de Fontes

As fontes de alimentação podem ser divididas em duas categorias:

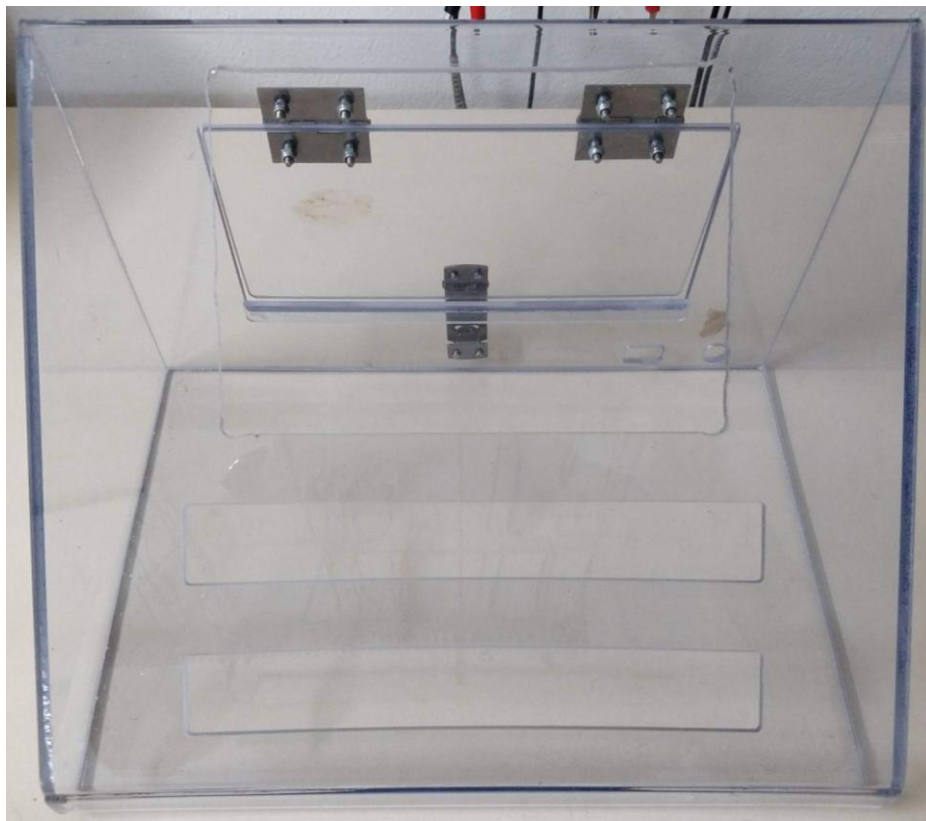
- **Fonte de alimentação linear:** De acordo com Direct Industry (2020), este tipo de fonte de alimentação fornece uma ou mais tensões contínuas estabilizadas e constantes, independentemente das variações de tensão da rede. Uma fonte de alimentação linear é composta por um transformador, um retificador, um filtro e um regulador (ou circuito de controle). O transformador baixa o nível da tensão da rede, o retificador converte a tensão alternada em tensão contínua, o filtro armazena a energia de modo a tornar mais constante e linear a tensão obtida à saída do retificador e, por fim, o regulador estabiliza e regula a tensão de saída.
- **Fonte de alimentação comutada, ou fonte de alimentação chaveada:** Segundo o mesmo Direct Industry (2020), com este tipo de fonte de alimentação, a regulação é efetuada por componentes eletrônicos de potência, como os transístores, responsáveis pela comutação (também chamada chaveamento). Ao contrário das fontes lineares, certas fontes de alimentação de modo comutado transmitem instantaneamente a potência. Este tipo de fonte de alimentação é adequado para dispositivos eletrônicos comuns, tais como computadores, televisores, carregadores de *smartphones*, etc.

### 3 DESENVOLVIMENTO DO KIT DIDÁTICO

#### 3.1 Estrutura do Kit

A base estrutural do kit é de acrílico, e possui formato retangular com vista frontal inclinada, onde se localiza a tela *touch screen* e as entradas e saídas da placa, já na vista posterior há uma abertura do tipo portinhola, para acesso e manutenção nos componentes internos e da placa da Raspberry Pi (Figura 4).

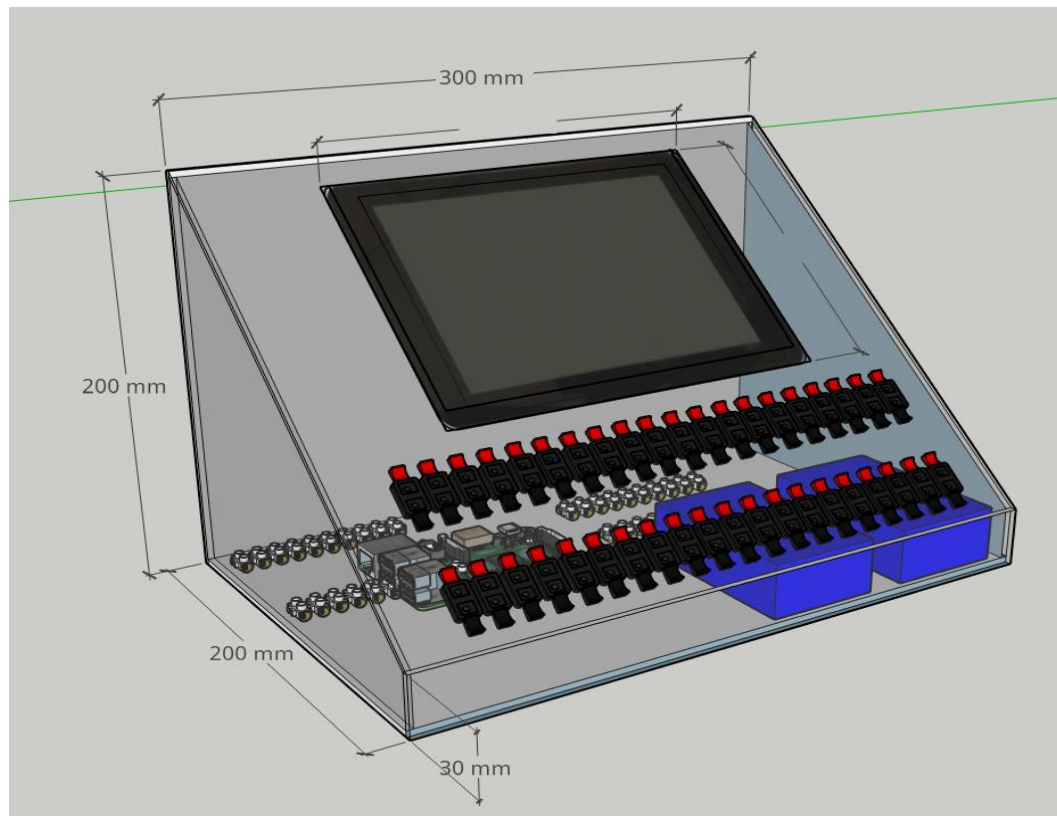
**Figura 4 - Base estrutural em acrílico do kit em desenvolvimento.**



Fonte: o autor.

Na (Figura 5) mostra as etapas para elaborar o projeto estrutural, que foi através de um software de modelagem onde é possível ter a real noção do que seria o projeto.

**Figura 5 - Dimensões totais e visualização do pré-projeto do kit didático.**



Fonte: o autor.

### 3.2 Lista de Componentes

A seguir será descrito os principais componentes que formam o kit didático, dentre eles estão:

- **Placa Raspberry PI 3B:** Componente principal do kit didático, é uma placa de circuito integrado, possui processador de 1.4GHz, memória RAM de 1GB, conexão *Ethernet*, *Wi-Fi* e *Bluetooth* integrado, entre outras funcionalidades como GPIOs, portas *USB*, *HDMI*, (Figura 6);

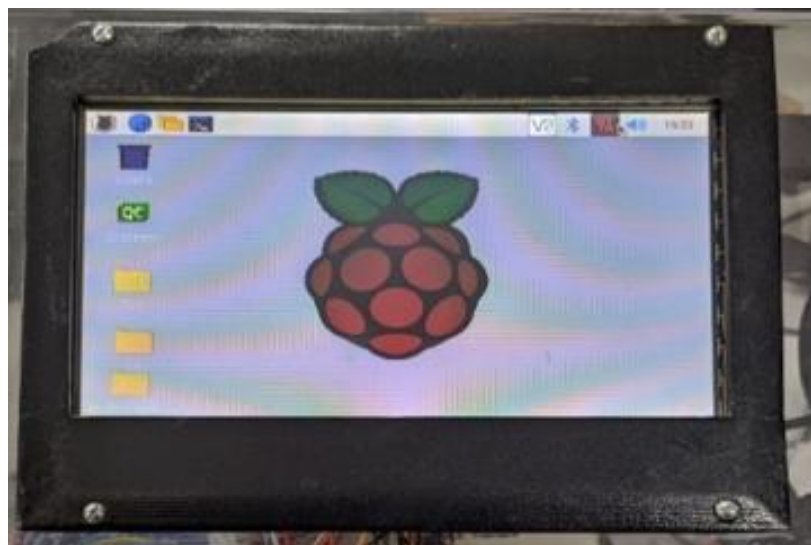
Figura 6 - Placa Raspberry Pi 3B.



Fonte: o autor.

- **Tela *Touch Screen 7"***: A tela que está fixada na parte frontal, e será utilizada para mostrar todas as funções do Raspberry Pi, assim como, programar diretamente no kit didático, alterar configurações de conexão do Raspberry Pi, acessar internet e realizar pesquisas e também utilizar a tela como painel de comandos de equipamentos, (Figura 7);

Figura 7 - Tela *Touch Screen 7"*.



Fonte: o autor.

- **Conectores Borne de Pressão 4 Vias – 16 unidades:** Também serão unidos e fixados na parte frontal do kit, vão ser conectados todas as saídas GPIOs da placa Raspberry Pi, estes conectores são práticos e uteis para realização de testes e projetos eletrônicos, (Figura 8);

Figura 8 - Borne de Pressão 4 Vias.



Fonte: o autor.

- **Caneta para tela Touch Screen:** Será utilizada para manusear os comandos na *tela touch*, com mais praticidade (Figura 9).

Figura 9 – Tela Touch Screen, Bornes de Pressão e Caneta.



Fonte: o autor.



- **Conector Sindal:** Será utilizada para conexão dos fios que saem das GPIO da placa, até os conectores bornes de caixa que dão acesso a parte externa, (Figura 10).
- **Prensa cabo:** Utilizado para passagem do cabo de alimentação da fonte, e de outros periféricos do kit didático (Figura 10).

Figura 10 – Conector Sindal e Prensa Cabos <sup>3</sup>/<sub>4</sub>



Fonte: o autor.

- **Fonte 5V:** Utilizada para alimentar a placa Raspberry Pi. Ela ajusta a tensão de entrada de 220v (CA) para saída de 5v (CC), (Figura 11).

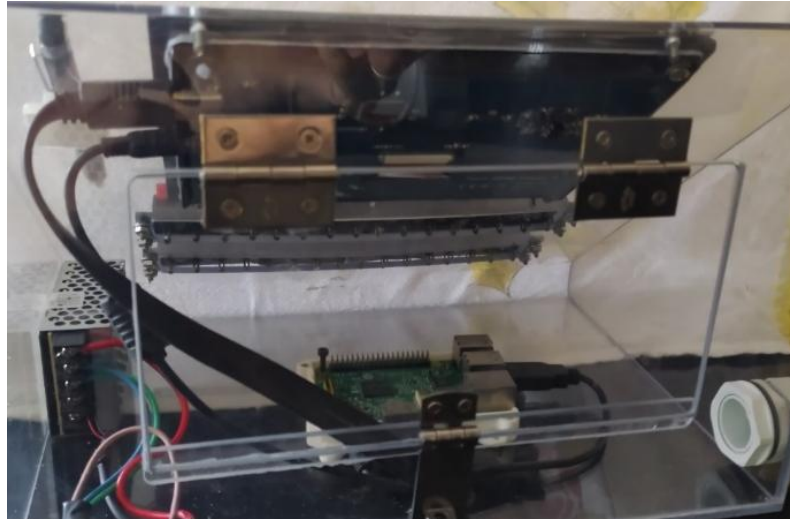
Figura 11 – Fonte Ajustável de 5V.



Fonte: o autor.

- **Cabo HDMI:** Para conectar a saída *HDMI* da placa Raspberry Pi a tela *touch screen* (Figura 12).

Figura 12 – Cabos HDMI



Fonte: o autor.

- **Botão liga / desliga:** Um interruptor para ligar e desligar o kit didático quando plugado o cabo na tensão de 220v (Figura 13).

Figura 13 – Cabo de Alimentação e Botão Liga/Desliga

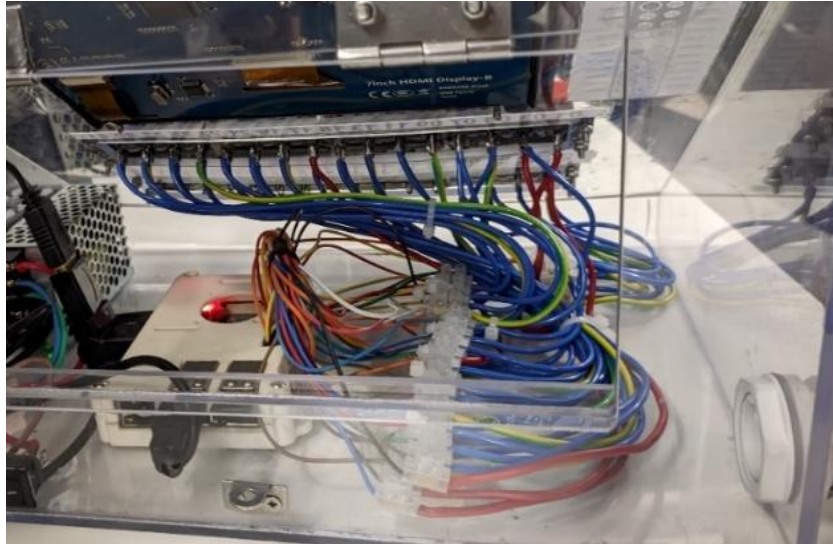


Fonte: o autor.

- **Cabo unifilar 1 mm<sup>2</sup>:** Cabos utilizados para conectar os bornes até os conectores sindal;

- **Cabos Jumper 20cm:** Cabos utilizados para conectar GPIO da placa Raspberry Pi aos conectores sindal (Figura 14).

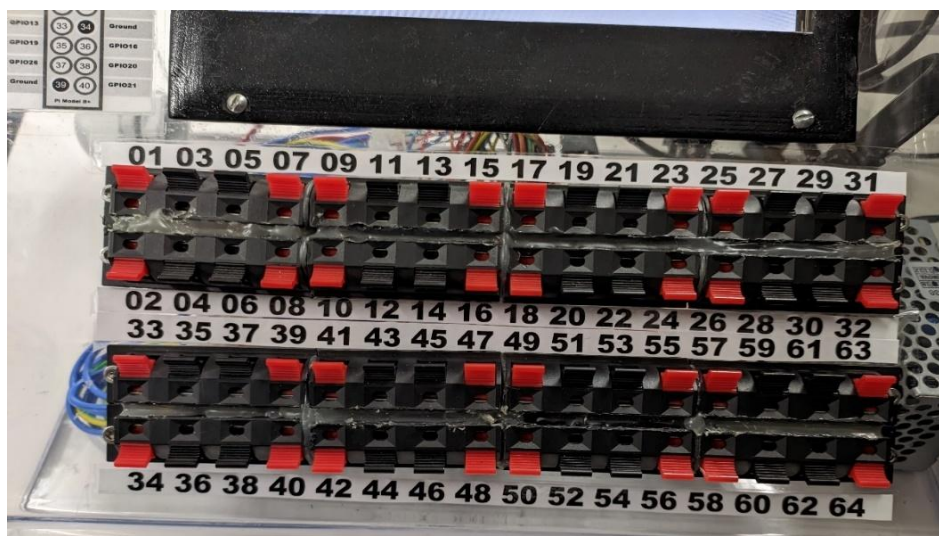
Figura 14 – Cabos unifilar e Jumpers



Fonte: o autor.

- **Etiquetas de Identificação:** Etiquetas para identificar posição das GPIOs nos bornes de caixa, e também identificando a função de cada GPIO da placa Raspberry Pi, ambas estão fixadas na parte frontal do kit (Figura 15).

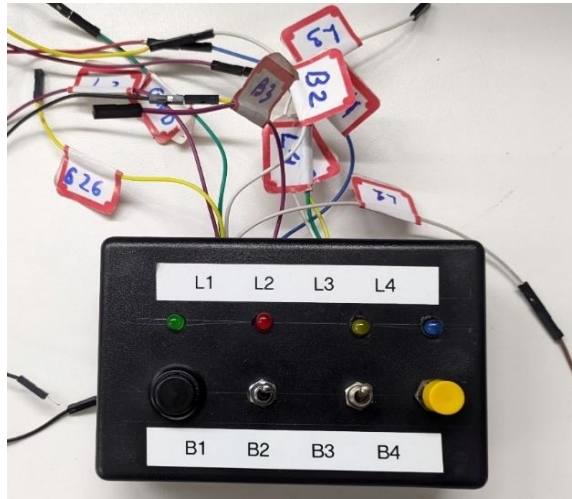
Figura 15 – Etiquetas de identificação das GPIO nos bornes de pressão do kit didático



Fonte: o autor.

- **Caixa com Leds e Botões para testes:** Foi confeccionado uma caixa com 4 botões e 4 Leds com Resistores de  $220\Omega$  em série com cada Led, para conectar as entradas e saídas GPIO. O objetivo é demonstrar o funcionamento do kit didático utilizando os comandos do botão como sinal de entrada nos exemplos, e os Leds como sinal de saída, (Figura 16).

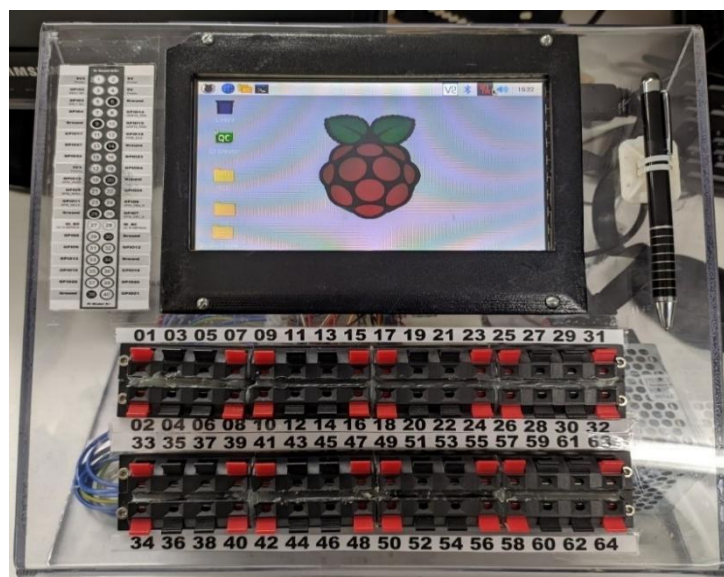
Figura 16 – Caixa de testes do kit didático



Fonte: o autor.

A (Figura 17), mostra a montagem concluída do kit didático, na (Figura 18), é visto o kit didático em funcionamento com a caixa de testes de Leds, testando as GPIOs nos bornes de pressão:

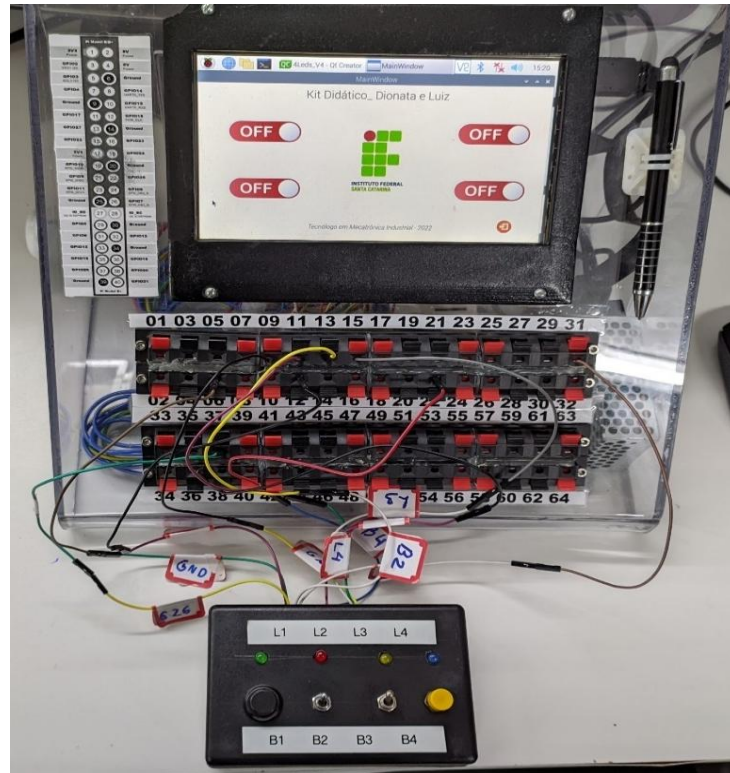
Figura 17 – Vista frontal da montagem concluída do kit didático



Fonte: o autor.



Figura 18 – Kit didático em funcionamento

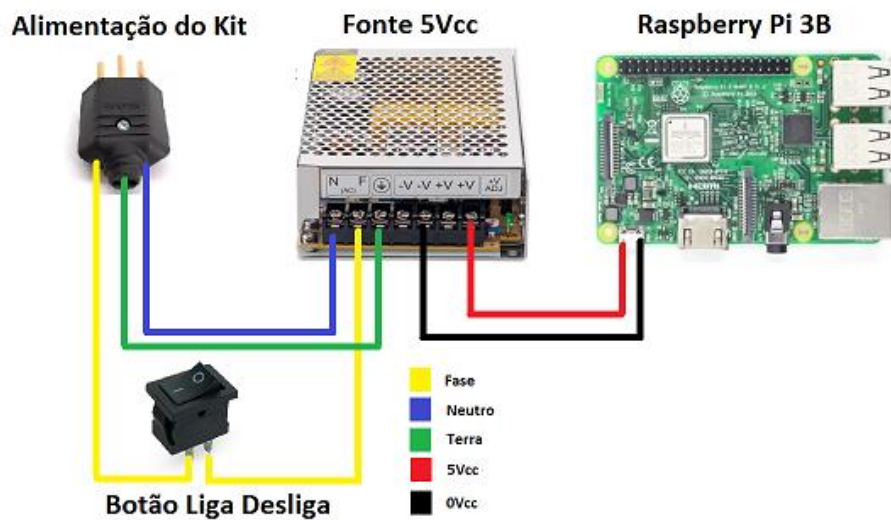


Fonte: o autor.

### 3.3 Diagrama Elétrico do Kit Didático

A seguir na (Figura 19), está representado o diagrama elétrico de alimentação do Kit didático:

Figura 19 - Diagrama Elétrico de alimentação do Kit.



Fonte: o autor.

### 3.4 Custo do Projeto

Na (Tabela 1) a seguir, está listado o custo do projeto para produzir o kit didático:

**Tabela 1 – Descrição do Custo do Projeto**

Item	Quantidade	Custo
Caixa de Acrílico	1	R\$ 300,00
Tela Touch Screen 7"	1	R\$ 211,54
Conectores Borne Para Caixa Acustica 4 Vias Pressão	16	R\$ 80,00
Caneta para tela Touch Screen	1	R\$ 16,50
Placa raspberry PI 3	1	R\$ 607,00
Conector Sindal	4	R\$ 19,76
Prensa cabo	1	R\$ 4,33
Fonte 5V	1	R\$ 33,29
Cabo HDMI	1	R\$ 27,99
Cabo de alimentação para Placa Raspberry pi 3	1	R\$ 12,00
Botão liga / desliga	1	R\$ 1,56
Cabo unifilar 1 mm <sup>2</sup> - em metros	4	R\$ 3,74
Cabos Jumper 20cm conjuntos com 10und cada	5	R\$ 61,10
<b>Total</b>		<b>R\$ 1.378,81</b>

Fonte: o autor.

## 4 DEMONSTRAÇÃO DO FUNCIONAMENTO DO KIT DIDÁTICO

Para demonstrar o funcionamento do Kit Didático, foram elaborados dois exemplos simples de aplicação:

### 4.1 Exemplo 1

Acionamento de 4 leds através de 4 botões físicos, utilizando a linguagem de programação Python.

#### 4.1.1 Programação

Para esse exemplo, foi criada uma lógica de programação utilizando a linguagem de programação Python, como mostra a (Figura 20).

Figura 20 – Lógica de programação para acionamento dos 4 leds por botões físicos.

```

1  import RPi.GPIO as gpio
2  import time
3
4  gpio.setmode(gpio.BCM)
5  gpio.setwarnings(False)
6
7  gpio.setup(6, gpio.OUT)
8  gpio.setup(18, gpio.OUT)
9  gpio.setup(22, gpio.OUT)
10 gpio.setup(17, gpio.OUT)
11
12 gpio.setup(26, gpio.IN, gpio.PUD_UP)
13 gpio.setup(27, gpio.IN, gpio.PUD_UP)
14 gpio.setup(16, gpio.IN, gpio.PUD_UP)
15 gpio.setup(25, gpio.IN, gpio.PUD_UP)
16
17 while True:
18     if gpio.input(26) == gpio.LOW:
19         gpio.output(6, gpio.HIGH)
20     else:
21         gpio.output(6, gpio.LOW)
22
23     if gpio.input(27) == gpio.LOW:
24         gpio.output(18, gpio.HIGH)
25     else:
26         gpio.output(18, gpio.LOW)
27
28     if gpio.input(16) == gpio.LOW:
29         gpio.output(22, gpio.HIGH)
30     else:
31         gpio.output(22, gpio.LOW)
32
33     if gpio.input(25) == gpio.LOW:
34         gpio.output(17, gpio.HIGH)
35     else:
36         gpio.output(17, gpio.LOW)
37
38
39 gpio.cleanup()

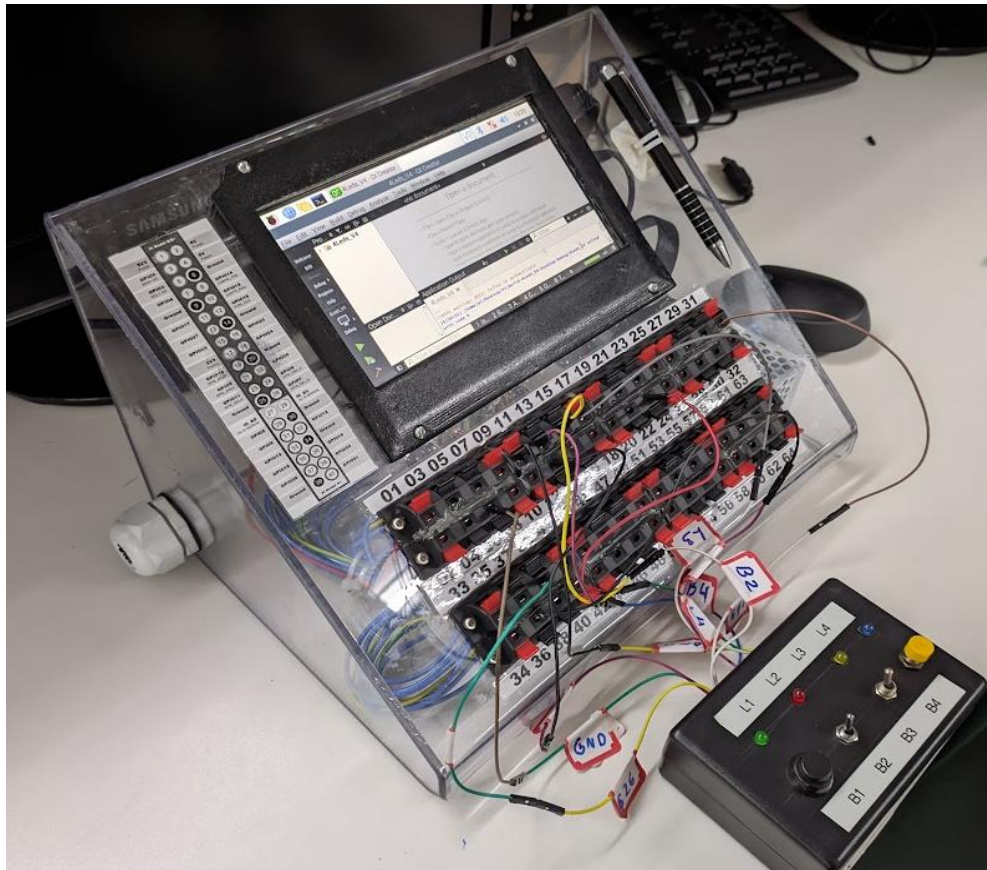
```

Fonte: o autor.

#### 4.1.2 Funcionamento

Após executado a lógica de programação, ao pressionar o botão “B1” aciona o *Led* “L1”, e assim respectivamente para os botões “B2”, “B3”, “B4” e para os *leds* “L2”, “L3”, “L4”, como mostra a (Figura 21).

Figura 21 – Setup do acionamento dos 4 leds por botões físicos.



Fonte: o autor.

## 4.2 Exemplo 2

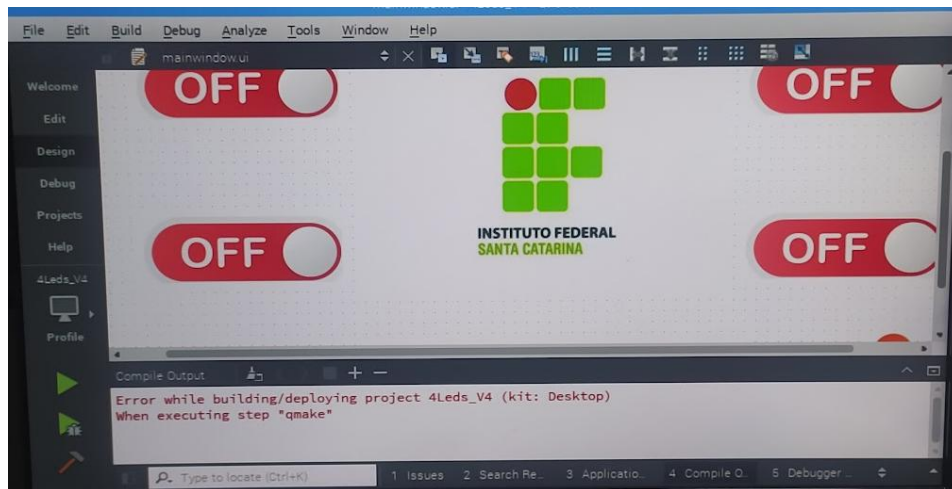
Acionamento de 4 *leds* através da tela *touch screen*, utilizando a linguagem de programação C++.

### 4.2.1 Programação

Para o exemplo 2, foi criado um executável utilizando a linguagem de programação C++, através de uma IDE - (*Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado), multiplataforma para desenvolvimento de *software* QT Creator. Na (Figura 22), mostra o *layout* do *software* de criação da interface, e na (Figura 23), mostra lógica de programação utilizando a linguagem de programação C++.



Figura 22 – Layout do software de criação da interface 4 leds na tela touch screen.



Fonte: o autor.

Figura 23 – Lógica de programação para acionamento dos 4 leds pela tela touch screen.

```

1 #include "mainwindow.h"
2 #include "ui_mainwindow.h"
3 #include <wiringPi.h>
4
5 int pino_Botao_1 = 22;
6 int pino_Botao_4 = 0;
7 int pino_Botao_2 = 1;
8 int pino_Botao_3 = 3;
9
10 MainWindow::MainWindow(QWidget *parent)
11     : QMainWindow(parent)
12     , ui(new Ui::MainWindow)
13 {
14     ui->setupUi(this);
15 }
16
17 MainWindow::~MainWindow()
18 {
19     delete ui;
20 }
21
22
23 void MainWindow::on_Botao_1_clicked()
24 {
25     digitalWrite(pino_Botao_1, !digitalRead(pino_Botao_1));
26 }
27
28 void MainWindow::on_Botao_2_clicked()
29 {
30     digitalWrite(pino_Botao_2, !digitalRead(pino_Botao_2));
31 }
32
33 void MainWindow::on_Botao_3_clicked()
34 {
35     digitalWrite(pino_Botao_3, !digitalRead(pino_Botao_3));
36 }
37
38 void MainWindow::on_Botao_4_clicked()
39 {
40     digitalWrite(pino_Botao_4, !digitalRead(pino_Botao_4));
41 }
42

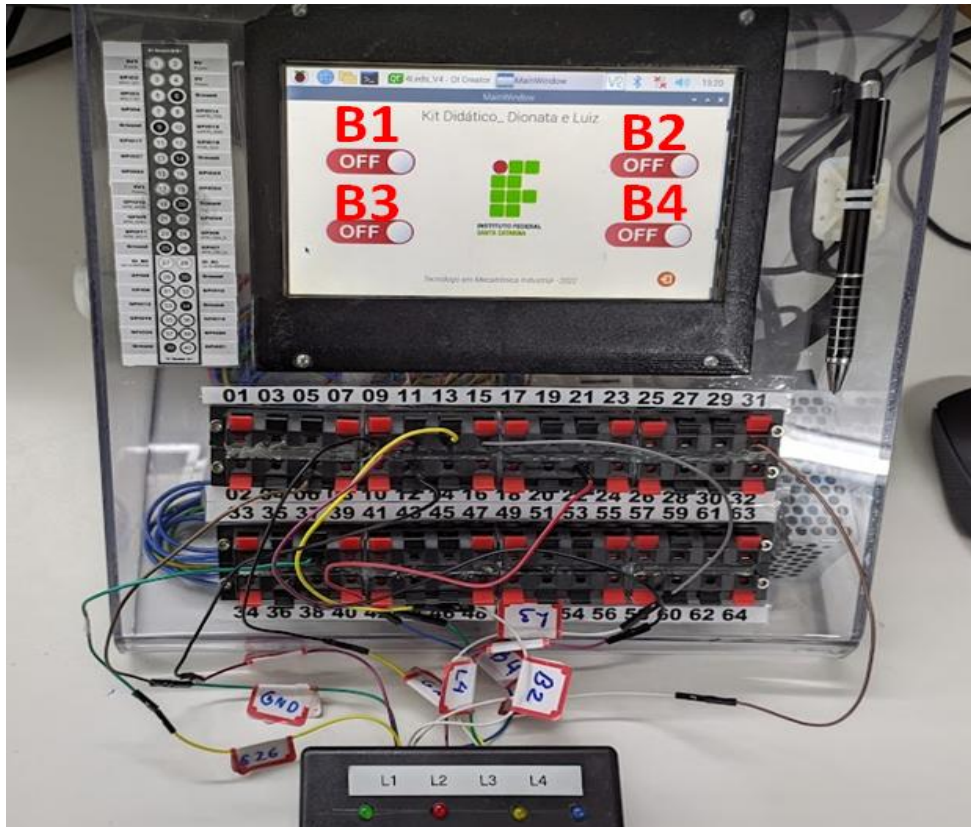
```

Fonte: o autor.

#### 4.2.2 Funcionamento

Após executado a lógica de programação, ao pressionar o botão na *tela touch* “B1” aciona o *led* “L1”, e assim respectivamente para os botões “B2”, “B3”, “B4” e para os *leds* “L2”, “L3”, “L4”, como mostra a (Figura 24).

Figura 24 – Setup do acionamento dos 4 *leds* pela *tela touch screen*.



Fonte: o autor.

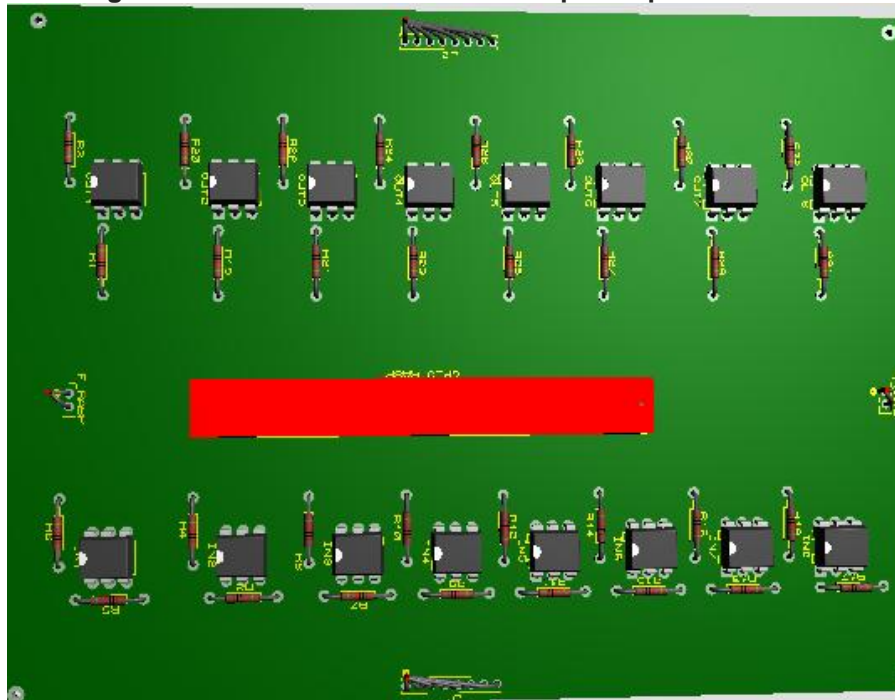
## 5 SUGESTÃO DE MELHORIA

### 5.1 Circuito de Proteção contra surtos nas GPIO

A sugestão de melhoria para o kit didático, é a implementação de um circuito de proteção contra curto circuito nas GPIOs, protegendo a placa Raspberry Pi. A equipe desenvolveu um circuito com optoacopladores, mas o tamanho da placa de circuito impresso ficou grande, e não foi possível montar dentro da estrutura do kit.

Foi projetado uma placa de circuito impresso com 16 optoacopladores, sendo que 8 para pinos de entradas e 8 para pinos de saída (Figura 25).

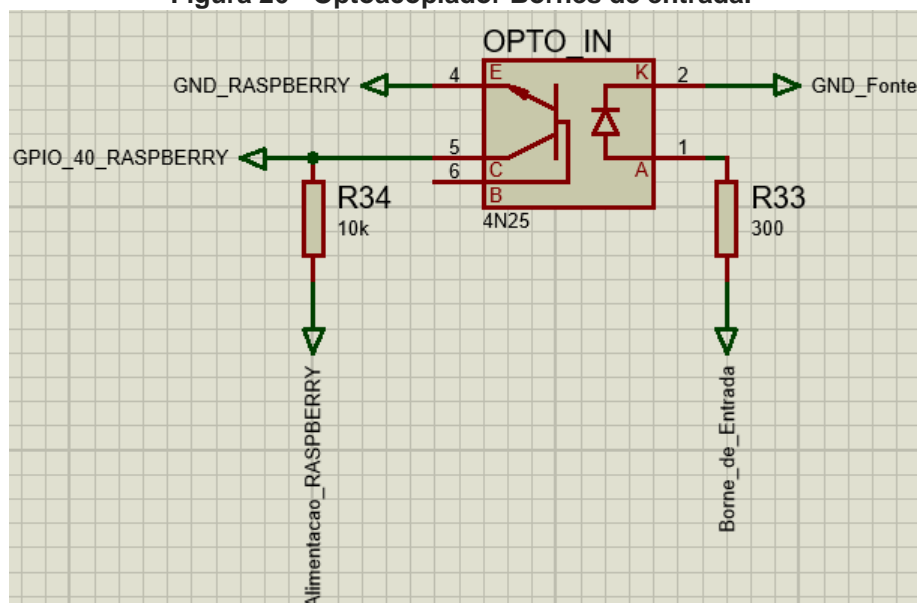
Figura 25 - Placa de circuito com 16 optoacopladores.



Fonte: o autor.

Circuito de ligação de proteção com Optoacoplador nos Bornes de Entrada (Figura 26).

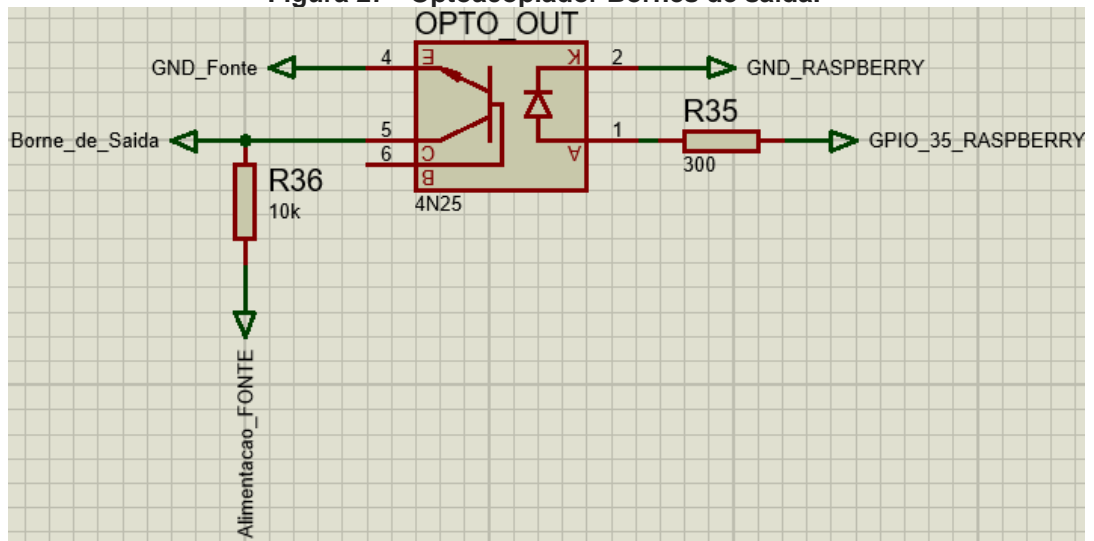
Figura 26 - Optoacoplador Bornes de entrada.



Fonte: o autor.

Circuito de ligação de proteção com Optoacoplador nos Bornes de Saída (Figura 27).

Figura 27 - Optocoplador Bornes de saída.



Fonte: o autor.

## 6 CONCLUSÃO

O kit em si proporciona a funcionalidade que o discente necessita o tornando prático em sua execução, ainda que algumas melhorias possam ser implementadas. O mesmo cumpriu com as expectativas que era de integrar as multitarefas que a placa *Raspberry* proporciona em um único equipamento.

Foram abordadas em detalhes as características da placa e da montagem do kit, como também as programações que foram utilizadas neste trabalho demonstram que o kit didático, é de fácil compreensão e bastante versátil como nos dois exemplos da programação em *Python* e *C++*, e dos controles dos *leds* tanto por botões físicos quanto pela tela *touch screen*, que faz deste kit um atrativo para as escolas, integrando a sociedade com o ramo da tecnologia, auxiliando jovens e adultos a ingressarem e conhecerem melhor o *Raspberry Pi*.

O objetivo é de que este kit didático possa divulgar e propagar ainda mais o conhecimento da placa *Raspberry Pi* e a programação, pois já se sabe da grandiosidade do *Arduino* e o quanto é disseminado o seu conhecimento dentre as escolas, e poder mostrar aos discentes a versatilidade da placa *Raspberry Pi* e a praticidade da programação, criando infinitas ferramentas e possibilidades de novas ideias e aplicações tecnológicas.

## REFERÊNCIAS

BAUERMEISTER, GIOVANNI. Programação em Linguagem C com Raspberry Pi. Disponível em: <<https://www.filipeflop.com/blog/linguagem-c-com-raspberry-pi/>>. Acesso em: Abril de 2022.

BAUERMEISTER, GIOVANNI. Interface Gráfica com Qt e Raspberry Pi. Disponível em: <<https://www.filipeflop.com/blog/interface-grafica-com-qt-e-raspberry-pi/>>. Acesso em: Abril de 2022.

DIRECT INDUSTRY. Quais os diferentes tipos de fontes de alimentação. Disponível em: <<https://guide.directindustry.com/pt/que-fonte-de-alimentacao-escolher/#:~:text=Fontes%20de%20alimenta%C3%A7%C3%A3o%20AC%2FDC,ten%C3%A3o%20%C3%A0s%20necessidades%20do%20dispositivo>>. Acesso em: Novembro de 2022.

GUIMARÃES, FLÁVIO. Como Programar Botões na GPIO - RaspberryPi Primeiros Passos. Disponível em: <<https://www.youtube.com/watch?v=E1f0IDiJyI8>>. Acesso em: Abril de 2022.

GUTIERREZ, GUSTAVO E ALMEIDA, MARCO. Conflito E Gestão Da Qualidade De Vida Nas Organizações. Vilarta, Roberto (Orgs.). Qualidade De Vida E Fadiga Institucional. Campinas: Ipes, 2006.P.85-96.

MENDES, TATYANE. Linguagem de programação: conheça os principais conceitos e ferramentas desse universo. Disponível em: <<https://www.napratica.org.br/linguagem-de-programacao/>>. Acesso em: Setembro de 2022.

YAMAMOTO, CLEBER. Quais são as aplicações mais famosas feitas em Python?. Disponível em: <<https://oraculoti.com.br/2017/03/24/quais-sao-as-aplicacoes-mais-famosas-feitas-em-python/>>. Acesso em: Dezembro de 2022.

PACIEVITCH, YURI. Linguagem de programação C++.

Disponível em: <<https://www.infoescola.com/informatica/cpp/>>. Acesso em: Dezembro de 2022.

PENHA, LUIZ GUSTAVO SILVA DA. Como fazer Interface Gráfica QT Creator e Raspberry Pi 4 em C/C++.

Disponível em: <[https://lgustavorobotica.kpages.online/jetson\\_arduino](https://lgustavorobotica.kpages.online/jetson_arduino)>. Acesso em: Abril de 2022.

RASPBERRY PI FOUNDATION. raspberry pi, 2018. Democratização de Tecnologia.

Disponível em: <<https://www.raspberrypi.com/products/raspberry-pi3-model-b-plus/>>.

TEIXEIRA, MARIANE MENDES. Touch screen.

Disponível em: <<https://brasilecola.uol.com.br/fisica/touch-screen.htm>>. Acesso em: Setembro de 2022.

QUINAFELEX, RODOLFO. Qual é a diferença entre tela capacitiva e tela resistiva.

Disponível em: < <https://www.techtudo.com.br/noticias/2013/06/qual-e-diferenca-entre-tela-capacitiva-e-tela-resistiva-entenda.ghtml>>. Acesso em: Janeiro de 2023.

VALERI, VITOR. GPIO.

Disponível em: <<https://www.oficinadanet.com.br/hardware/40552-o-que-e-gpio>>. Acesso em: Dezembro de 2022.

MELLO, MARCIO. O que é Raspberry Pi, para que serve e principais modelos no Brasil.

Disponível em: <<https://victorvision.com.br/blog/o-que-e-raspberry-pi/>>. Acesso em: Dezembro de 2022.