

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

THIAGO HENRIQUE TROVA

**DESENVOLVIMENTO DE MÓDULO DE INTERFACEAMENTO DE
DISPOSITIVOS DE ENTRADA E SAÍDA PARA USO DIDÁTICO EM
DISCIPLINAS DE ENGENHARIA**

CAMPO MOURÃO

2023

THIAGO HENRIQUE TROVA

**DESENVOLVIMENTO DE MÓDULO DE INTERFACEAMENTO DE
DISPOSITIVOS DE ENTRADA E SAÍDA PARA USO DIDÁTICO EM
DISCIPLINAS DE ENGENHARIA**

**Development of an input and output device interface module for didactic
use in engineering courses**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Bacharel em Engenharia Eletrônica
do Curso de Bacharelado em Engenharia
Eletrônica da Universidade Tecnológica Federal
do Paraná.

Orientador: Prof. Dr. Lucas Ricken Garcia

CAMPO MOURÃO

2023



[4.0 Internacional](https://creativecommons.org/licenses/by-nc/4.0/)

Esta licença permite remixe, adaptação e criação a partir do trabalho, para fins não comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

THIAGO HENRIQUE TROVA

**DESENVOLVIMENTO DE MÓDULO DE INTERFACEAMENTO DE
DISPOSITIVOS DE ENTRADA E SAÍDA PARA USO DIDÁTICO EM
DISCIPLINAS DE ENGENHARIA**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do
título de Bacharel em Engenharia Eletrônica
do Curso de Bacharelado em Engenharia
Eletrônica da Universidade Tecnológica Federal
do Paraná.

Data de aprovação: 27/Novembro/2023

Lucas Ricken Garcia
Título (Doutorado)
Universidade Tecnológica Federal do Paraná

Eduardo Giometti Bertogna
Título (Doutorado)
Universidade Tecnológica Federal do Paraná

Marcio Rodrigues da Cunha
Título (Doutorado)
Universidade Tecnológica Federal do Paraná

**CAMPO MOURÃO
2023**

AGRADECIMENTOS

Gostaria de expressar meus sinceros agradecimentos a Deus, fonte de orientação e sabedoria, por me guiar nesta jornada acadêmica que culmina com a conclusão deste trabalho.

À minha amada família, meu pai Jair Trova e minha mãe Velisete Melnik, minha gratidão é profunda pelo apoio incondicional e amor que permearam cada etapa deste percurso.

Um agradecimento especial ao meu orientador, Lucas Ricken Garcia, cuja experiência e orientação foram pilares para a realização deste trabalho. Seus insights valiosos contribuíram significativamente para o meu crescimento acadêmico.

Também quero estender meu agradecimento ao meu padrinho Paulo Trova e minha madrinha Cristina Trova, cujo apoio e foram fundamentais para este sucesso, sem eles eu não poderia estar aqui.

Agradeço aos amigos, colegas e demais familiares que contribuíram com palavras de incentivo e conhecimento.

Com profunda gratidão, Thiago Henrique Trova.

No cerne da revolução tecnológica, a eletrônica embarcada desempenha um papel vital na moldagem de nosso mundo moderno. Como ressaltado por Jack Kilby, o inventor do circuito integrado, "A eletrônica embarcada é a arte de unir a imaginação humana à capacidade de processamento das máquinas."(KILBY, Jack, 1958). Essa afirmação atemporal ecoa através das décadas, lembrando-nos que a eletrônica embarcada não é apenas um campo técnico, mas uma fusão entre criatividade humana e potencial computacional.

RESUMO

Neste trabalho apresenta-se o desenvolvimento de um módulo de interfaceamento de dispositivos de entrada e saída universal para utilização com placas de desenvolvimento em microcontroladores e sistemas embarcados. Atualmente, para um bom entendimento e melhor didática de disciplinas na engenharia que envolvam sistemas embarcados, microcontroladores e placas de desenvolvimento, se observa a carência de um módulo de uso simples, que possua uma variedade de dispositivos e seja de fácil utilização independentemente da arquitetura de microcontroladores utilizada. Neste sentido, foi desenvolvido um módulo impresso de interfaceamento de sensores e dispositivos de entrada e saída, tendo como base módulos eletrônicos já existentes e usuais no mercado. Para isso foram realizadas (i) análises e levantamentos de circuitos usuais e populares no desenvolvimento de projetos com placas de desenvolvimento, (ii) a seleção efetiva dos circuitos, (iii) a validação dos circuitos eletrônicos junto aos *datasheets* dos respectivos componentes, (iv) testes em bancada de esquemas elétricos e de funcionalidade e, assim, (v) confecção e validação do módulo desenvolvido. Como resultado deste trabalho obteve-se uma placa de circuito impresso funcional, permitindo o uso de circuitos diversos como conversores analógico/digitais, diferentes protocolos de comunicação, modulador por largura de pulso, dispositivos de entrada e saída visuais e comuns a desenvolvedores e à academia. O módulo foi projetado com um *design* que favoreça o desenvolvimento de jogos ou possua uma interação similar a um video-game portátil. Este processo de diversificação em sala de aula pode ser uma estratégia interessante para aumentar o engajamento dos alunos na execução dos projetos.

Palavras-chave: microcontroladores; sistemas embarcados; ensino; placa de circuito impresso.

ABSTRACT

This work presents the development of an interfacing module for universal input and output devices for use with development boards in microcontrollers and embedded systems. and development boards, there is a lack of a simple-to-use module that has a variety of devices and is easy to use regardless of the microcontroller architecture used. In this sense, a printed interface module for sensors and input and output devices will be developed, based on electronic modules that already exist and are common in the market. For this, analyzes and surveys of usual and popular circuits will be carried out in the development of projects with development boards; determine the effective selection of circuits; determine the effective selection of circuits; validate the project with the datasheets of the respective components; validate the project with the datasheets of the respective components; perform bench tests of electrical schematics and functionality and, thus, manufacture and validate the developed module. As a result of this work, a functional board is expected, where there is integration with any development board, allowing the use of different circuits such as analog/digital converters, different communication protocols, pulse modulator, visual output and input devices common to developers. and to the academy. With the realization of this project proposal, I concluded that the module allowing access to some resources along with development boards, can be a facilitator for teaching microcontrollers and embedded systems. Due to its easy-to-use feature, it can allow quick access to resources that could previously require time and equipment to build and adapt these modules in a rudimentary way, making the development environment less visually polluted and reducing possible practical problems in the interfacing process.

Keywords: microcontrollers; embended systems; teaching; printed circuito board.

LISTA DE FIGURAS

Figura 1 – Setores e <i>softwares</i> de sistemas microcontrolados e embarcados. . . .	13
Figura 2 – Relatório da pesquisa sobre o mercado brasileiro de sistemas embarcados e iot do ano de 2021.	14
Figura 3 – <i>Software</i> utilizado.	24
Figura 4 – Diagrama de disposição dos módulos representado o kit de interfaces desenvolvido	25
Figura 5 – Guia visual do kit desenvolvido.	26
Figura 6 – Esquemático dos circuitos RTC, MAX485, LEDs E USB	29
Figura 7 – Esquemático dos circuitos de botões, chaves, PWM, optoacoplador e joystick	30
Figura 8 – Esquemático dos circuitos Regulador de tensão 3,3 V e 5 V, alimentação e barra de pinos	30
Figura 9 – Esquemático dos circuitos de <i>display</i> , conversor ADC e pino seletor de tensão	30
Figura 10 – Vista superior PCB 2D	31
Figura 11 – Vista 3D da PCB projetada lado A e B.	32
Figura 12 – Fluxograma prática de laboratório 01	37
Figura 13 – Fluxograma prática de laboratório 02	39
Figura 14 – Fluxograma prática de laboratório 03	40
Figura 15 – Fluxograma prática de laboratório 04	42
Figura 16 – Fluxograma prática de laboratório 05	44
Figura 17 – Esquemático	69

LISTA DE FOTOGRAFIAS

Fotografia 1 – <i>Hardware</i> da prática de laboratório criação de jogo.	17
Fotografia 2 – Estação meteorologica.	18
Fotografia 3 – Controle de acesso de locais físicos.	18
Fotografia 4 – Resultado do projeto proposto por Rocha da Silva.	19
Fotografia 5 – Projeto de calculadora realizado por alunos do ensino médio.	19
Fotografia 6 – Placa Sanusbino.	20
Fotografia 7 – Robô de baixo custo desenvolvido por alunos.	21
Fotografia 8 – Placa sem componentes - lado A e B.	35
Fotografia 9 – Placa com componentes - lado A e B.	36
Fotografia 10 – Imagem demonstrativa da 1º prática proposta.	38
Fotografia 11 – Imagem demonstrativa da 2º prática proposta.	40
Fotografia 12 – Imagem demonstrativa da 3º prática proposta demonstrando o valor do resistor dependente de luz.	41
Fotografia 13 – Imagem demonstrativa da 3º prática proposta demonstrando o valor do potenciômetro.	41
Fotografia 14 – Imagem demonstrativa da 4º prática proposta.	43
Fotografia 15 – Imagem demonstrativa da 5º prática proposta.	45

SUMÁRIO

1	INTRODUÇÃO	10
1.1	Objetivos	11
1.1.1	Objetivo geral	11
1.1.2	Objetivos específicos	11
1.2	Justificativa	11
2	FUNDAMENTACAO TEÓRICA	13
2.1	Crescimento dos sistemas microcontrolados e embarcados	13
2.2	Tendências destacando <i>softwares</i> em sistemas embarcados	15
2.3	Ensino e formação de profissionais em sistemas microcontrolados e embarcados	16
2.4	Projetos propostos para sistemas microcontrolados e embarcados	17
2.5	Trabalhos relacionados a ensino de sistemas embarcados e microcontrolados	19
2.6	Protocolos de comunicação serial comumente usados	21
2.6.1	Protocolo de comunicação I2C	22
2.6.2	Comunicação TTL	22
2.6.3	Protocolo de comunicação RS-485.	23
3	METODOLOGIA	24
3.1	Visão Geral do Módulo de Interfaceamento de Dispositivos de Entrada e Saída	24
3.2	Guia de conexões da placa de expansão de interfaces desenvolvida	25
3.2.1	Conexão para interfaceamento de <i>Display</i>	26
3.2.2	Botões Direcionais, Chaves e Botão Analógico	26
3.2.3	Sinalizações visuais	27
3.2.4	Módulo ADC	27
3.2.5	Interfaces Seriais	27
3.2.6	Módulo RTC	28
3.2.7	Driver para acionamento de cargas via PWM	28
3.2.8	Alimentação do circuito projetado	29
3.3	Projeto do Esquemático elétrico	29

3.4	Projeto da placa de circuito impresso	30
3.5	Proposição de Prática de laboratório e elaboração de manual de uso . .	32
3.5.1	Proposta Prática de Laboratório 01	32
3.5.2	Proposta Prática de Laboratório 02	32
3.5.3	Proposta Prática de Laboratório 03	33
3.5.4	Proposta Prática de Laboratório 04	33
3.5.5	Proposta Prática de Laboratório 05	34
4	RESULTADOS	35
4.1	Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 1	36
4.2	Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 2	38
4.3	Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 3	39
4.4	Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 4	40
4.5	Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 5	43
5	CONCLUSÃO	46
	REFERÊNCIAS	47
	APÊNDICE A ALGORITMO PRATICA DE LABORATÓRIO 01^o.	51
	APÊNDICE B ALGORITMO PRATICA DE LABORATÓRIO 02^o.	54
	APÊNDICE C ALGORITMO PRATICA DE LABORATÓRIO 03^o.	58
	APÊNDICE D ALGORITMO PRATICA DE LABORATÓRIO 04^o.	62
	APÊNDICE E ALGORITMO PRATICA DE LABORATÓRIO 05^o.	66
	APÊNDICE F ESQUEMÁTICO ELÉTRICO DO PROJETO DESENVOL- VIDO.	69

1 INTRODUÇÃO

Quando se observa o desenvolvimento da engenharia eletrônica nas últimas décadas, nota-se que a utilização de sistemas embarcados é perceptível em quase todas as áreas, com desenvolvimento de soluções para aplicações específicas, além da redução física desses sistemas. Em termos de aplicabilidade, sistemas embarcados e microcontrolados possuem uma ampla gama de aplicações abrangendo equipamentos clínicos, automação agrícola, equipamentos de uso diário (Kato, 2013), tais como roteadores, forno micro-ondas e refrigeradores. Além disso, observa-se uma crescente ascensão de equipamentos de automação residencial, apresentado por Caetano (2021) que utiliza-se de uma Raspberry Pi 3 como recurso de processamento de informações; ou por Lima, Nobre e ALENCAR (2015) que utilizam de uma plataforma Arduino Mega para o desenvolvimento de dispositivos para automação residencial.

Neste contexto, com o aumento da demanda por dispositivos eletrônicos microcontrolados e embarcados, cresce também a demanda pela adaptação e aumento de ações pedagógicas para um ensino superior voltado a práticas e aplicações com base em problemas reais. Nota-se que um ensino voltado à prática é uma verdadeira necessidade nos cursos de graduação (MEEK *et al.*, 2003), assim como é reconhecido o aumento contínuo do uso de *kits* de desenvolvimento para o setor educacional (Kato, 2013).

Desde o surgimento dos microcontroladores, a área educacional tem se empenhado em desenvolver métodos de ensino para esses dispositivos. Um exemplo é o Cricket (Resnick, 1998), criado pelo MIT em 1993, que consiste em um kit de componentes eletrônicos e peças de Lego para criar pequenos robôs. Esse kit usava uma linguagem simples baseada em Logo. No entanto, com o advento da plataforma Arduino em 2005, o ensino de microcontroladores se tornou mais flexível e acessível devido à facilidade de conexão com diversos módulos (Kato, 2013).

Em Ma *et al.* (2010) foi proposta a utilização um kit de desenvolvimento com um microcontrolador AVR. No kit são ofertados mais 16 módulos, no qual suas conexões são realizadas a partir de “*jumpers*” (fios provendo comunicação de um ponto a outro). Os principais módulos usados são: *display* LCD, comunicação serial, teclado básico e LEDs (diodo emissores de luz) (Ma *et al.*, 2010) (Kato, 2013).

Na Universidade North Florida, nos EUA, em meados dos anos 2008 fora descrito o ensino de microcontroladores no curso de Engenharia Elétrica com um kit da Freescale com Debugger CodeWarrior utilizando o microcontrolador CSM12C32. A proposta usada na época engloba a construção dos módulos pelos discentes, para posterior utilização, junto ao desenvolvimento do aprendizado em programação. Módulos usados frequentemente são os *displays* LCD (*Liquid Crystal Display*), LEDs, comunicação serial, PWM (*Pulse-width Modulation*) e conversores ADC (Analogico-Digital e Digital-Analogico) (Choi, 2008).

Todas estas soluções possuem um ponto em comum: o desenvolvimento de um kit de desenvolvimento com módulos genéricos focado em uma arquitetura de microcontroladores

específica. Ou seja, em grande parte estes kits de desenvolvimento possuem módulos para interação entre usuário e máquina, tais como LEDs, *displays*, teclados e botões, recursos para conversão de sinais analógicos em digitais e digitais para analógicos, e, dependendo do interesse maior do kit, circuitos para comunicação serial, acionamento de cargas por meio de relés e outras interações com o mundo real por meio de sensores e atuadores específicos, porém, com todos estes módulos já interfaceados a um microcontrolador específico.

Observando esta característica em focar em apenas uma única arquitetura para o desenvolvimento de um recurso para ensino de microcontroladores e sistemas embarcados, este trabalho vislumbra a possibilidade da criação de um kit de desenvolvimento genérico, sem a definição de um microcontrolador específico. A criação de um kit nestes moldes pode aumentar a flexibilidade de utilização de um mesmo conjunto de módulos de entrada e saída, porém, com diferentes microcontroladores (ou recursos de processamento) e em diferentes disciplinas da graduação.

1.1 Objetivos

1.1.1 Objetivo geral

O trabalho propõe o desenvolvimento de um módulo de interfaces de entrada e saída a partir da criação de uma Placa de Circuito Impresso (do inglês "*Printed Circuit Board*" - PCB), com funcionalidades como *display* LCD, botões, chaves seletoras, modulador por largura de pulso, conversor ADC e interfaces de comunicação serial, com o intuito de auxiliar na implementação de métodos de ensino voltados à prática em sistemas micro controlados e embarcados.

1.1.2 Objetivos específicos

Os objetivos específicos deste trabalho são listados a seguir:

- Levantar a demanda didática com relação aos circuitos a serem usados;
- Selecionar os circuitos que irão compor o esquemático elétrico;
- Estudar os circuitos obtidos a partir de *datasheets* dos CI (Circuitos Integrados);
- Realizar testes de bancada circuito a circuito;
- Prototipagem e validação da placa de circuito impresso;
- Criação do guia de uso.

1.2 Justificativa

O ensino de microcontroladores e sistemas embarcados tem, naturalmente, um grande viés prático, exigindo que o discente tenha contato com diferentes microcontroladores e dispo-

tivos de entrada e saída. Dependendo da abordagem da disciplina em um curso de Engenharia, este processo de aprendizagem pode focar na utilização de kits de desenvolvimento para facilitar o entendimento do processo de abstração tão comum à rotina de programação, como também pode ser utilizado para um maior aprofundamento sobre a arquitetura de um determinado microcontrolador, utilizando seus periféricos e compreendendo quais são as formas adequadas para interfaceamento de dispositivos de entrada e saída.

Por exemplo, segundo Araújo, Ferreira e Oliveira (2016) o uso da ferramenta Arduino como apoio no ensino de eletrônica e programação nas engenharias é um elo para desmistificação de conceitos abstratos, fazendo analogias a problemas reais e colocando em prática conceitos da eletrônica e programação.

Com a globalização e adoção crescente de tecnologias disruptivas, amplia-se o número de recursos aplicados na educação, tornando-se crucial a implementação dessas novas tecnologias no caminho de aquisição e criação do conhecimento (Milani; Milani, 2010). Neste sentido, o ensino de microcontroladores também exige um processo de atualização relativamente rápido, o que impacta também na atualização dos kits de desenvolvimento.

Pensando na necessidade constante do uso de *kits* de desenvolvimento no ensino de eletrônica e programação, sabe-se que os periféricos em grande parte sempre seguem um padrão, envolvendo conversores ADC (Conversor Analógico Digital), PWM (*Pulse Width Modulation*), RTC (*Real Time Clock*), LEDs, *displays*, botões entre outros dispositivos externos ao módulo de controle, enquanto que o módulo de controle sofre atualizações a medida que novas tecnologias são desenvolvidas, tais como Arduino, Raspberry, ESP 32, microcontroladores ARM, entre outros à escolha do usuário. No entanto, observa-se na literatura, tais como (Silva *et al.*, 2019), (Cardoso *et al.*, 2020), (Amorim; Monlevade, 2016), (Koslowski, 2014), (Casara; Zamparete; Beloli, 2014), que as soluções empregadas geralmente são focadas em microprocessadores ou microcontroladores específicos, de acordo com o projeto.

Desta forma, observa-se que pode-se buscar uma maior flexibilidade em relação ao módulo de controle se desenvolvido um kit de desenvolvimento com uma abordagem mais genérica, mais focada nos dispositivos de entrada e saída do que no módulo de controle em si, inserindo inúmeras interfaces como as citadas anteriormente no foco do desenvolvimento didático do discente, podendo, assim, ter uma aplicabilidade no ensino de diferentes disciplinas da engenharia que envolvem programação, microcontroladores e sistemas embarcados.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os tópicos essenciais para fundamentar o desenvolvimento de um módulo de interfaceamento de dispositivos de entrada e saída para uso didático em disciplinas de engenharia.

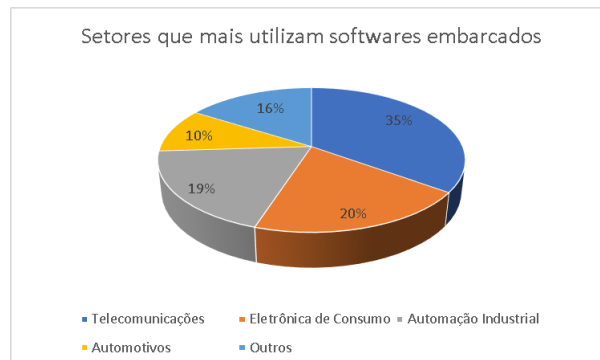
2.1 Crescimento dos sistemas microcontrolados e embarcados

O crescimento das aplicações da eletrônica embarcada é notável, principalmente o aumento em aplicações de uso específico, sendo produzidos bilhões de sistemas deste tipo, sendo mais eficazes, claros para os usuários além de acessíveis. Com a melhoria, também é visível a gradação de complexidade nos sistemas. Em quase todos casos, o emprego de sistemas embarcados se faz através de um gerenciamento e computação dos dados após a aquisição dos mesmos, na atualidade os dados tendem a serem valores analógicos que necessitam de conversão para que possam ser digitalizados e processados (Barros; Cavalcante, 2010).

De acordo com o relatório publicado pela The Brainy Insights, o mercado global de sistemas embarcados está projetado para crescer de 88,35 bilhões de dólares em 2020 para 138,45 bilhões em 2028, em uma taxa de crescimento anual composta de 5,73 por cento durante o período de previsão 2021-2028 (THE BRAINY INSIGHTS, 2022). Segundo Zelenovsky e Mendonça (1999) as aplicações de maior crescimento são nas áreas de telecomunicações, equipamentos médicos, automotivos, aviação e equipamentos de uso diário.

Assim é visível a digitalização das grandezas analógicas, fazendo com que o *hardware* seja cada vez mais dependente de *softwares* modernos e de alta qualidade. É mensurado que 50 % dos *recalls* do setor automotivo seja por conta de problemas em relação ao *software* (Pereira *et al.*, 2011). Na Figura 1, é demonstrado em forma de gráfico os setores que mais utilizam os sistemas embarcados.

Figura 1 – Setores e softwares de sistemas microcontrolados e embarcados.

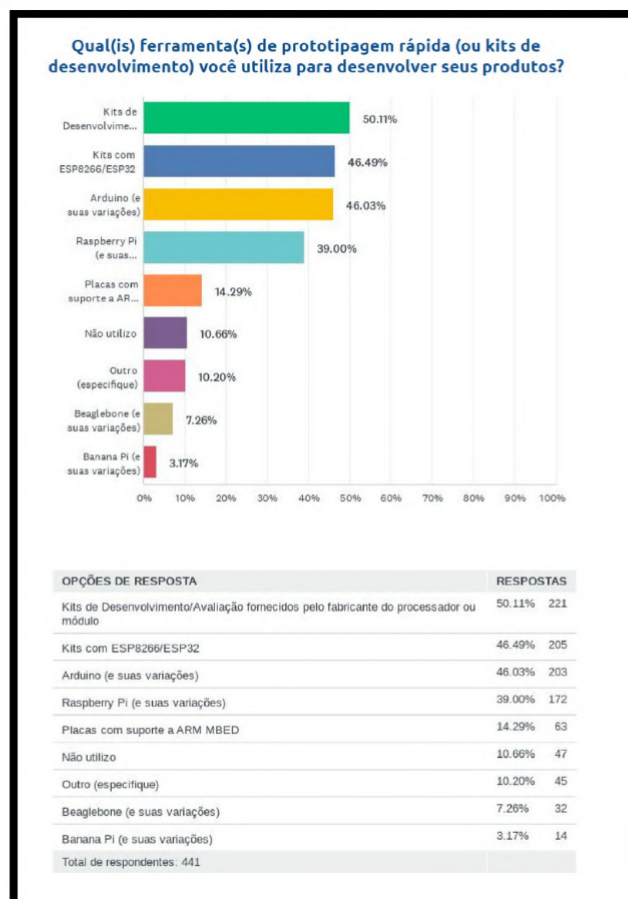


Fonte: Adaptado de Pereira *et al.* (2011, p. 07).

No Brasil, um dos portais de relevância na área de embarcados é o *embarcados.com.br*. Segundo este portal, foi avaliado o relatório da pesquisa sobre o mercado brasileiro de sistemas

embarcados e IOT (*Internet of Things*) do ano de 2021, de acordo com a pesquisa realizada, dois pontos importantes para o trabalho em questão foram levantados no relatório. Primeiro, foram avaliadas as principais áreas de aplicação de sistemas embarcados. No caso, foram 480 respostas obtidas considerando a possibilidade de assinalar múltiplas opções. As que mais se destacaram foram aplicações IoT (48,7%), 37,1% para Sistemas Industriais e 23,4% para Cidades Autônomas. A pesquisa completa pode ser encontrada em (PORTAL EMBARCADOS, 2021). Um segundo ponto avaliado foi a utilização de ferramentas e kits de desenvolvimento para auxiliar o processo de prototipagem de produtos. Neste caso foram 441 respostas que são apresentadas na Figura 2 (PORTAL EMBARCADOS, 2021). Nota-se que há uma grande variabilidade no interesse dos participantes da pesquisa, desde kits de desenvolvimento para ESP32, Arduino, Raspberry e soluções para ARM em geral.

Figura 2 – Relatório da pesquisa sobre o mercado brasileiro de sistemas embarcados e iot do ano de 2021.



Fonte: Adaptado de Portal Embarcados (2021).

Com base nos dados da pesquisa, torna-se evidente a eficácia do dispositivo abordado neste estudo, juntamente com a compreensão do avanço contínuo no campo de dispositivos

embarcados. Além disso, como a solução proposta corresponde ao módulo de interfaceamento genérico, isto possibilitará que diferentes soluções em microcontroladores/microprocessadores utilizem deste kit para desenvolvimento de projetos e testes em geral.

2.2 Tendências destacando *softwares* em sistemas embarcados

Nesta seção são abordadas as previsões e tendências do mercado de sistemas embarcados, mostrando que cada vez mais os *softwares* são necessários e demandam precisão, assim os tornando mais refinados e complexos (Taurion, 2005).

O *software* de um sistema embarcado tem uma diferenciação pois tem ênfase primariamente na interatividade entre meio físico onde está colocado, com isso ele requer alguns traços deste meio como consumo de energia, *timing* para executar, eficiência, entre outros (Lee, 2002).

Com crescimento significativo dos sistemas embarcados e aumento do tamanho dos códigos junto à grande capacidade de processamento e armazenamento dos *hardwares*, como foi previsto o crescimento pela Lei de Moore em 1965 (Moore *et al.*, 1965), permite assim aplicações com *hardware* que visam criações de funções onde o poder computacional e memória não são mais um grande problema, tornando fundamental a qualidade do *software* embarcado (Prado, 2012).

Os processadores do mundo até meados de 2005, que foram consumidos, mais de 98 % dos processadores tiveram o seu emprego em sistemas deste tipo e, se passa imperceptível aos usuários finais, não se dando conta de estar a usar um dispositivo embarcado, com *softwares* complexos (Taurion, 2005). Há tendências de uso de sistemas embarcados em aplicações que carecem de precisão, segurança e confiabilidade, assim para garantir essas exigências, os *softwares* são desenvolvidos especificamente para a aplicação em questão (Carrion; Werner, 2012). Simultaneamente com a evolução da capacidade dos *hardwares* os *softwares* evoluíram, porém, com uma taxa menor, assim havendo uma lacuna a ser preenchida com desenvolvimento de novos *softwares* gerando os menores códigos possíveis, para o fim de economizar em memória e dissipação de potência, em busca de tornar o sistema mais otimizado o possível (Prado, 2012).

É notável que a evolução nos cuidados médicos e o avanço na área são diretamente ligado a sistemas embarcados, e principalmente sistemas de monitoramento em tempo real, além do crescimento no setor da telemedicina, aumentando conectividade dos aparelhos, onde cada vez mais será requerido, *softwares* de alto grau de confiabilidade e complexidade elevada, pelo fato de que vidas podem ser perdidas em caso de falhas. Além desse setor, outros como o área de telecomunicações, segmentos industriais, militares e aeroespaciais tem grande destaque nas tendências de aplicações dos sistemas embarcados, como exemplo indústrias de telecomunicações ou eletrônica de consumo, no geral tem suas equipes de desenvolvimento

formadas em pelo menos 75 % de desenvolvedores de *softwares* para aplicação em sistemas embarcados (Taurion, 2005).

Como comentado anteriormente, a indústria da aviação é um exemplo de setor extremamente dependente de tecnologia, como se tem no Brasil uma indústria inovativa e imersa na lógica global e produção (Araújo; Oliveirall, 2020), é de suma importância levantar dados no âmbito da aviação, pois os produtos aéreos modernos incorporam ampla utilização de *Software Embarcado Aeronáutico*(SEA), onde a asseguarção da qualidade e segurança é intrínseca. Em cenários regulamentados, como a aviação, os elementos humanos e técnicos desempenham um papel crucial na garantia da segurança e qualidade. Nota-se que no contexto dos desenvolvedores de *Software Embarcado Aeronáutico* há deficiências identificadas e ligadas aos componentes técnicos, dando maior atenção para "ausência de familiaridade com os procedimentos de Engenharia de *Software* (ES)"(Yelissety *et al.*, 2021).

2.3 Ensino e formação de profissionais em sistemas microcontrolados e embarcados

Um aluno ao sair da graduação tem, em média, uma carreira de 40 a 50 anos. Logo as considerações sobre evolução da tecnologia e tendências de mercado devem ser levadas em conta. Mesmo não podendo afirmar sobre períodos muito a frente, as tendências de curto prazo, devem, sim, ser consideradas para que o curso de engenharia não se torne algo ortodoxo e inerte. Segundo Chen *et al.* (2007), a rápida evolução das tecnologias de *hardware* gerou um impacto considerável nos sistemas microcontrolados e embarcados, impactando, também, no papel da programabilidade dos *softwares*.

Assim como é descrito por Koslowski (2014), uma vez que o mercado está em constante mudança e evolução, os kits de desenvolvimento se tornam obsoletos ou apresentam uma limitação de recursos a medida que o tempo passa. Desta forma, torna-se recorrente a necessidade de atualização desses kits, o que pode ser inviável de se fornecer pelo provedor do curso. Com isso, há relativa importância para o aluno ter acesso a um kit de valor acessível que expande as funcionalidades do kit de desenvolvimento usado, bem como adaptar um kit já existente com novos recursos e funcionalidades (Chen *et al.*, 2007).

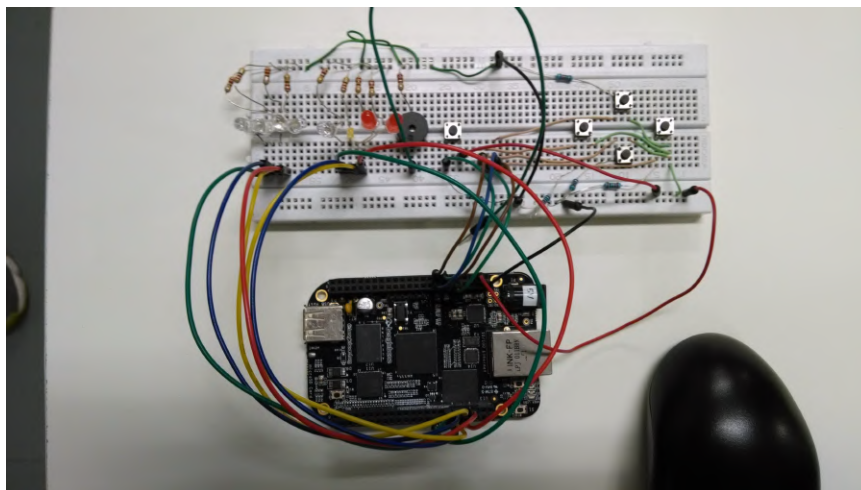
Torna-se então fundamental fazer escolhas de qual área enfatizar no decorrer do curso, considerando essa perspectiva, a utilização de uma placa expansora de funcionalidades, traz uma economia de tempo na questão da construção dos módulos periféricos, usada junto com o kits de desenvolvimento, tornando o tempo hábil do aluno superior para que possa se dedicar em desenvolvimento apenas de *software* ou *firmware*, o que é mais proveitoso para uma grade com enfoque em desenvolvimento da parte programacional.

2.4 Projetos propostos para sistemas microcontrolados e embarcados

Nesta seção serão apresentadas algumas práticas comuns no ensino de engenharia e apontamentos sobre como um módulo de interfaceamento de entradas e saídas pode ser interessante para facilitar o dia a dia do professor em sala de aula.

Amorim e Monlevade (2016) apresentam o projeto de controle de jogo com requisitos de tempo real, usando uma interface simples como LEDs e botões. A placa de desenvolvimento utilizada neste projeto foi a Beagle-Bone Black, composta por um ARM® Cortex-A8 de 1 GHz e uma SGX 3D *Graphics Engine* (sua descrição mais precisa pode ser encontrada no site oficial do fabricante na área referente a este modelo, <https://www.beagleboard.org/boards/beaglebone-black>). O resultado do projeto pode ser verificado na Fotografia 1.

Fotografia 1 – Hardware da prática de laboratório criação de jogo.



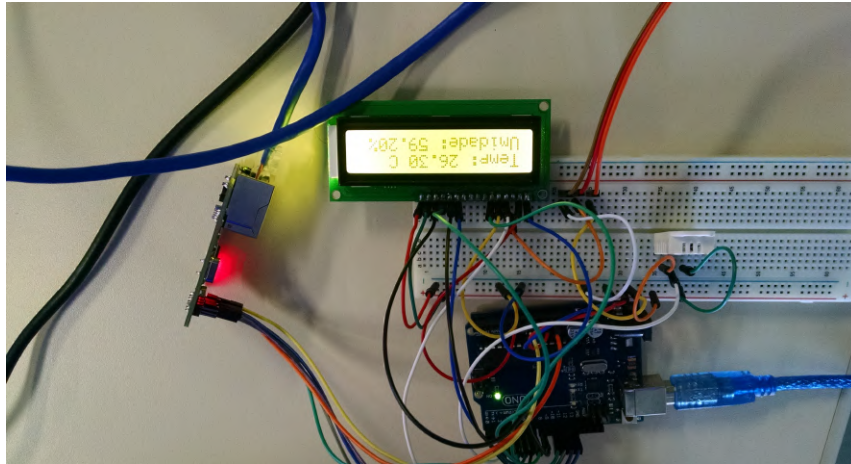
Fonte: Adaptado de Amorim e Monlevade (2016, p. 06).

Observa-se que o protótipo foi desenvolvido utilizando *protoboard*, realizando o interfaceamento com a BeagleBone a partir de *jumpers*. Esta estratégia não é usual para utilização cotidiana em sala de aula. Entretanto, a ideia de aproximar os discentes de um contexto mais gamificado pode aumentar o engajamento no desenvolvimento das tarefas.

Já na Fotografia 2 é demonstrado o desenvolvido de uma proposta de prática de laboratório com conexão a internet em que os sensores captam variações físicas e estas informações coletadas são apresentadas em um *display* LCD em uma *protoboard*, o projeto no caso faz o uso de um Arduino. Novamente, observa-se a grande quantidade de conexões e a necessidade de uma conexão de dispositivos eletrônicos utilizando uma placa de prototipação.

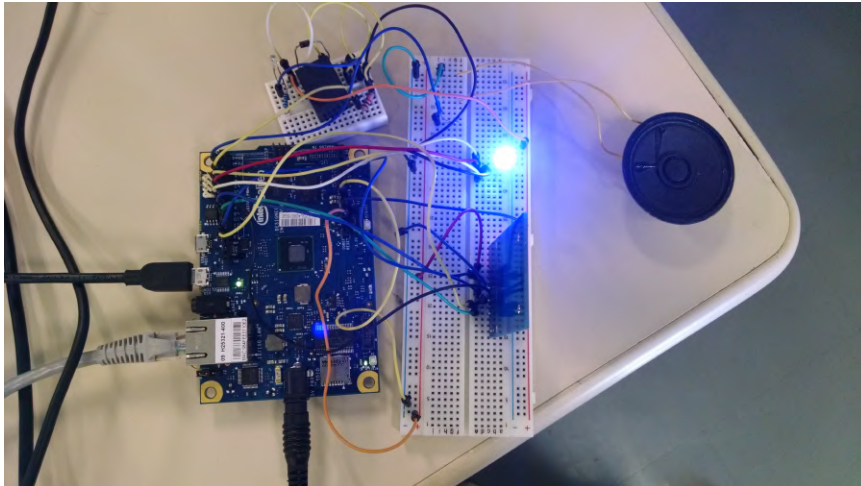
Os mesmos autores apresentam outra prática laboratorial comum no ensino de microcontroladores: o desenvolvimento de um controle de acesso, Fotografia 3. No caso, a tecnologia empregada é o NFC, onde é realizado o monitoramento e gestão do acesso via web. A placa de desenvolvimento usada para esta aplicação foi o Arduino Uno R1 que possui um microcontrolador ATmega328P.

Fotografia 2 – Estação meteorologica.



Fonte: Adaptado de Amorim e Monlevade (2016, p. 06).

Fotografia 3 – Controle de acesso de locais físicos.

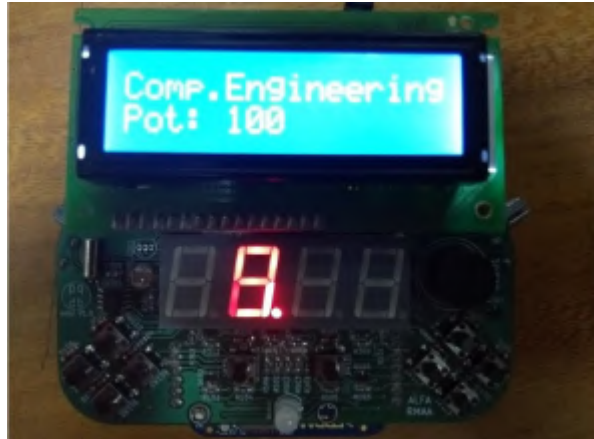


Fonte: Adaptado de Amorim e Monlevade (2016, p. 06).

Em outra perspectiva, Silva *et al.* (2019) projetaram uma placa para desenvolver atividades com alunos tal como uma *shield* para *arduino*. Para fazer o controle das funcionalidades oferecidas pela placa, como pode ser observado na Fotografia 4, os resultados da construção da placa e seu uso foram satisfatórios e de real valor em sala de aula (Silva *et al.*, 2019).

Embasado nos dados trazidos anteriormente percebe-se que é visualmente mais limpo o circuito proposto por Silva *et al.* (2019) em comparação às práticas realizadas com alunos de Amorim e Monlevade (2016). Observa-se que o uso do kit de expansão de placas de desenvolvimento para aulas de sistemas microcontrolados e embarcados seria vantajoso, além do foco do desenvolvimento ser mais voltado para criação de *softwares* e *firmwares* com intuito de evoluir o aprendizado nesta área, assim o tempo hábil pode ser efetivado com o uso de uma placa de interfaces já pronta.

Fotografia 4 – Resultado do projeto proposto por Rocha da Silva.



Fonte: Adaptado de Silva *et al.* (2019, p. 09).

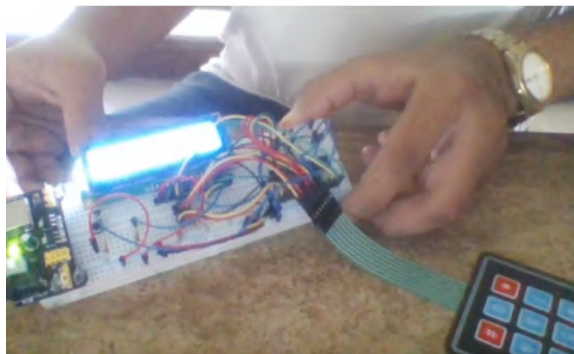
2.5 Trabalhos relacionados a ensino de sistemas embarcados e microcontrolados

Em vista dos tempos atuais, é fundamental que se estimule o entusiasmo do discente para que já cedo tenha contato com tecnologias novas e faça a familiarização dessas ferramentas. Neste sentido a Universidade Federal do Ceará, realizou um projeto focado em levar a disciplina de microcontroladores a escolas profissionais de ensino médio, por saber que microcontroladores são a base de automação e robótica, assim, desenvolvendo aptidões de desenvolvimento em jovens antes mesmo de saírem da escola (Alves; Costa; Bezerra, 2018).

Notou-se que foram abordados os seguintes tópicos nessa iniciativa de (Alves; Costa; Bezerra, 2018):(i) Introdução aos paradigmas de programação em C; (ii) Configuração e Programação de Periféricos I/O; (iii) *Display* LCD; (iv) Comunicação serial USART; (v) Leitura de sinais analógicos com conversor ADC e (v) controle de servomotores com PWM, tudo controlado por um circuito usando um PIC18F4550.

Neste contexto, foi percebida a curiosidade dos alunos pela área de microcontroladores. Com a noção da importância social deste conhecimento, é esperado, que práticas como esta, influenciem positivamente no desenvolvimento do país como um todo (Alves; Costa; Bezerra, 2018).

Fotografia 5 – Projeto de calculadora realizado por alunos do ensino médio.

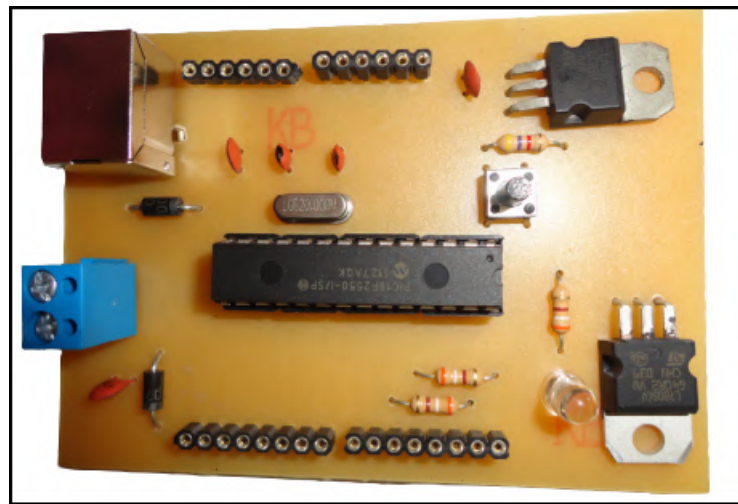


Fonte: Adaptado de Alves, Costa e Bezerra (2018, p. 07).

Um segundo trabalho realizado no mesmo sentido educacional é proposto por Silva *et al.* (2012). Neste trabalho são desenvolvidos projetos envolvendo robótica para prover uma consciência tecnológica na geração egressa das escolas. O projeto em questão tem a ênfase de criar um ambiente de conhecimento, criatividade e lúdico, possibilitando o contato do aluno com tecnologias usadas ao redor do mundo, fomentando o aprendizado de conceitos teóricos e práticos de ocasiões reais. Assim, possibilita o crescimento e aplicações de técnicas pedagógicas diferentes em projetos, evoluindo as destrezas e capacidades lógicas dos alunos.

A placa criada para controlar o projeto de robótica é mostrada na Fotografia 6. Apelidada de Sanusbino, esta placa é formada com base em um microcontrolador da família PIC18F2550. Segundo Silva *et al.* (2012) mostrou-se funcional para o uso em diversas turmas que trabalharam com as aulas usando kit de robótica.

Fotografia 6 – Placa Sanusbino.



Fonte: Adaptado de Silva *et al.* (2012, p. 03).

Na Fotografia 7 é apresentado o resultado do projeto de Silva *et al.* (2012) com o desenvolvimento de um robô seguidor de labirintos criado junto a escolas públicas utilizando a placa Sanusbino e materiais recicláveis.

Fotografia 7 – Robô de baixo custo desenvolvido por alunos.



Fonte: Adaptado de Silva *et al.* (2012, p. 03).

Portanto, observa-se que dependendo do interesse maior da disciplina, do projeto educacional ou do viés de ensino do professor, uma placa de desenvolvimento em microcontroladores pode focar em periféricos mais genéricos, bem como em soluções que possuem um tema específico, tal como robótica, acionamento de dispositivos e desenvolvimento de jogos.

2.6 Protocolos de comunicação serial comumente usados

Independentemente do viés de um kit de desenvolvimento em microcontroladores - ou de soluções mais genéricas como o caso de um módulo para interfaceamento de dispositivos de entrada e saída - é comum a necessidade de interfaceamento com dispositivos externos utilizando algum tipo de comunicação serial.

A comunicação serial é um método de transferência de dados que envolve a transmissão sequencial de bits através de um único canal. Os dados são representados binariamente e organizados em bytes. A taxa de transferência, medida em bits por segundo (bps), define a velocidade da comunicação. Em comunicações assíncronas, um bit de início (*start bit*) é enviado antes de cada byte para a sincronização. Após o byte, é transmitido um bit de parada (*stop bit*). Algumas vezes, é usado um bit de paridade para detecção de erros. Os dispositivos podem comunicar em *full duplex*, permitindo transmissão simultânea, ou *half duplex*, alternando entre transmissão e recepção (ROBOCORE, 2021).

A comunicação serial exige um canal físico e um protocolo para definição de regras. Protocolos desse tipo como por exemplo RS-232 (serial padrão para comunicação entre dispositivos DTE (*Data Terminal Equipment*) e DCE (*Data Circuit-terminating Equipment*)), RS-485 (para redes multiponto), TTL (*Transistor-Transistor Logic*), I2C (comunicação bidireccional) e SPI (interface serial periférica) (ROBOCORE, 2021).

2.6.1 Protocolo de comunicação I2C

A comunicação I2C, conhecida como *Inter-Integrated Circuit*, é um protocolo serial usado para interconectar dispositivos eletrônicos em sistemas integrados. Desenvolvido pela Philips, agora NXP Semiconductors, o protocolo I2C é amplamente utilizado devido à sua eficiência e simplicidade. (I2C-bus. . . , 2021)

A comunicação I2C envolve dois tipos de dispositivos: mestres e escravos. O mestre inicia e coordena a comunicação, enquanto os dispositivos escravos respondem a suas solicitações.

Dois fios são utilizados para a comunicação I2C: SDA (*Serial Data Line*) e SCL (*Serial Clock Line*). O SCL é um sinal de *clock* que sincroniza a comunicação, enquanto o SDA transporta os dados.

A comunicação começa com um sinal de início (*START*) emitido pelo mestre e termina com um sinal de parada (*STOP*). Esses sinais delimitam a transmissão de dados. O mestre envia um byte com o endereço do dispositivo escravo pretendido. Cada dispositivo possui um endereço único. O bit menos significativo do endereço determina se a operação é de leitura (bit 1) ou escrita (bit 0). Depois de enviar o endereço, o mestre pode iniciar a transmissão de dados. Se o bit de leitura ou escrita for 0, o dispositivo responderá com um *acknowledge* (ACK); se for 1, um *not-acknowledge* (NACK) será enviado. Os bytes de dados são transmitidos sequencialmente com sinais de *backnowledge* intermediários.

Os dispositivos escravos podem atrasar o sinal de *clock* temporariamente se precisarem de mais tempo para processar dados, conhecido como "*clock stretching*". O mestre pode emitir um sinal de reinício (*REPEATED START*) para iniciar uma nova comunicação sem liberar o barramento (I2C-bus. . . , 2021).

2.6.2 Comunicação TTL

O TTL (*Transistor-Transistor Logic*) é um padrão de comunicação para transferir dados digitais entre dispositivos eletrônicos. Ele estabelece os níveis de tensão que representam os estados lógicos "0"(baixo) e "1"(alto), bem como a interpretação desses sinais. O TTL é amplamente empregado em sistemas eletrônicos e microcontroladores para comunicação serial, como em conexões entre microcontroladores e sensores, *displays* e outros periféricos (Horowitz, 2015).

Este protocolo opera por meio de níveis lógicos. O "0"(baixo) é representado por uma tensão próxima a 0 V, indicando inatividade ou desligamento, enquanto o "1"(alto) é representado por uma tensão próxima à alimentação do sistema, geralmente 5 V. Além disso, o TTL ou lógica transistor-transistor é compatível com tecnologias similares, permitindo conexões diretas entre dispositivos. O TTL também exibe boa capacidade de "*fan-out*", possibilitando a alimentação direta de diversos dispositivos de entrada por um dispositivo de saída TTL. Apesar de

sensível ao ruído devido aos níveis lógicos bem definidos, o TTL apresenta imunidade razoável a interferências, em virtude da notável diferença entre os estados lógicos (Axelson, 1998).

A arquitetura do TTL envolve o uso de transistores bipolares para amplificar e regenerar sinais, promovendo uma transição nítida entre os estados lógicos. Além disso, é comum seu uso em comunicação serial assíncrona, possibilitando a transmissão sequencial de um bit de início, seguido por bits de dados (Horowitz, 2015).

2.6.3 Protocolo de comunicação RS-485.

O protocolo de comunicação RS-485 é um padrão utilizado para a transmissão de dados em redes seriais. Esse padrão possibilita a conexão de vários dispositivos a uma mesma linha de comunicação, permitindo a troca de informações entre eles. O RS-485 é reconhecido pela sua habilidade de transmitir dados em longas distâncias e por sua resistência a interferências. (Freitas, 2014)

No funcionamento do protocolo RS-485, ocorre uma transmissão diferencial de dados. Isso significa que os dados são transmitidos como uma diferença de tensão entre dois fios, denominados A e B. Quando um dispositivo envia um sinal lógico "1", o fio A mantém um nível de tensão superior ao fio B. Já quando o dispositivo envia um sinal lógico "0", o fio A fica com um nível de tensão inferior ao fio B (Freitas, 2014).

Essa abordagem diferencial confere ao RS-485 uma resistência natural a interferências e ruídos, tornando-o especialmente adequado para ambientes industriais ou locais com uma grande quantidade de dispositivos eletrônicos. Além disso, o RS-485 oferece suporte a diferentes topologias de rede, como barramento e *multidrop*, onde vários dispositivos podem compartilhar a mesma linha de comunicação (Freitas, 2017).

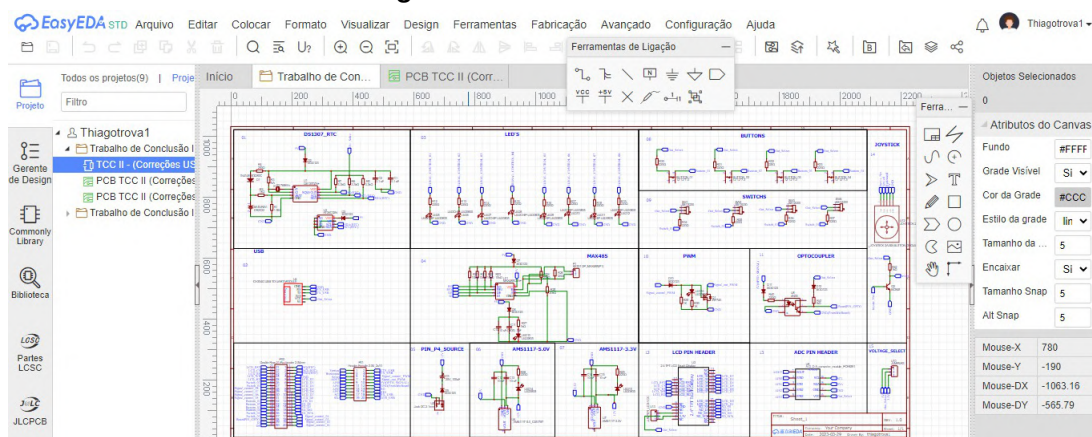
A operação do RS-485 envolve um transmissor e um receptor. Quando um dispositivo deseja transmitir dados, ele ativa o transmissor, ajustando os níveis de tensão nos fios A e B conforme os bits a serem transmitidos. O dispositivo receptor, por sua vez, compara as tensões presentes nos fios A e B para determinar os bits que foram recebidos (Freitas, 2014)(Axelson, 1998).

3 METODOLOGIA

Neste capítulo serão apresentados os materiais e *softwares* utilizados para o projeto do módulo de interfaceamento apresentado. Além disso, serão apresentados quais circuitos foram utilizados e os procedimentos para testes e desenvolvimento de material para dar suporte à sua utilização.

Para o desenvolvimento do diagrama esquemático e posteriormente o projeto da placa de circuito impresso, foi utilizado o *software online* EasyEDA (<https://easyeda.com/pt>). A escolha desta ferramenta foi baseada na quantidade de bibliotecas oferecida pelo *software*, gratuidade, facilidade de uso e a possibilidade de já realizar a encomenda do projeto através da própria plataforma. A Figura 3 apresenta a tela do EasyEDA para criação/edição do diagrama esquemático.

Figura 3 – Software utilizado.



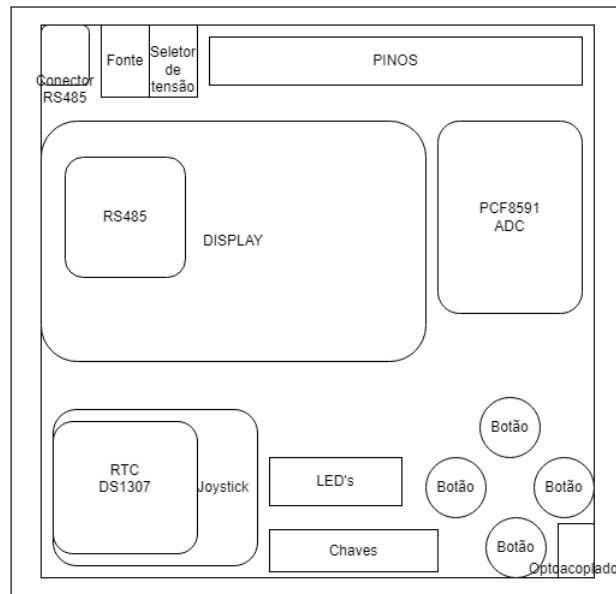
Fonte: (EasyEDA..., 2023).

O projeto do kit foi desenvolvido com base em módulos já existentes no mercado. Alguns foram integrados ao projeto e outros foram incorporados a partir de conexões padrões dispostas em regiões apropriadas para sua conexão. Para o projeto da PCB foi definido um limite máximo de 10x10 cm. Isto se deve ao preço promocional para confecção de placas até esta especificação de dimensão. Acima destas dimensões o valor por unidade se torna consideravelmente mais elevado.

3.1 Visão Geral do Módulo de Interfaceamento de Dispositivos de Entrada e Saída

A Figura 4 apresenta a concepção inicial do módulo desenvolvido. Observa-se que o foco foi o desenvolvimento de uma solução que pudesse ser utilizada de forma similar a um videogame portátil, possibilitando ao usuário/estudante desenvolver projetos que utilizam *display*, botões direcionais, botões de configuração e cursores analógicos. Este conjunto de periféricos pode propiciar o desenvolvimento de minijogos, aumentando o engajamento do discente em relação à tarefa.

Figura 4 – Diagrama de disposição dos módulos representado o kit de interfaces desenvolvido



Fonte: Autoria própria (2023).

Além dos periféricos básicos para o desenvolvimento de um minijogo, foram incorporados outros periféricos que podem ser úteis para o aprendizado de microcontroladores, tais como conversor AD/DA, relógio em tempo real (RTC), comunicação RS485, comunicação USB (não representado na imagem por estar na parte traseira da placa) e driver para acionamento de cargas via PWM.

Todos estes periféricos são elencados abaixo, descrevendo características, circuito-base, modificações e especificações.

3.2 Guia de conexões da placa de expansão de interfaces desenvolvida

O guia abrangente dos pinos de entrada e saída do kit desenvolvido proporciona uma visão da disposição de cada elemento. Isso resulta em maior agilidade e facilidade de uso, simplificando o acesso aos recursos do kit, apresentado na Figura 5.

Figura 5 – Guia visual do kit desenvolvido.



Fonte: Autoria própria (2023).

3.2.1 Conexão para interfaceamento de *Display*

Projetou-se na PCB uma conexão via barra de pinos destinada ao interfaceamento de *displays* TFT ou LCD. No caso, observou-se que no mercado há uma gama de *displays* que são compatíveis com a disposição de pinos do Arduino, tais como o *Display LCD TFT 2.4" Touchscreen Shield* e o módulo *Display LCD 16x2*. Desta forma, visando uma maior flexibilização, as conexões para os *displays* seguiram a disposição dos pinos do Arduino UNO R3. Estes pinos são então disponibilizados na barra de pinos geral (pinos ímpares de 9 a 31) e (pinos pares de 32 a 40), podendo o usuário realizar as conexões necessárias de acordo com o projeto e microcontrolador escolhido.

3.2.2 Botões Direcionais, Chaves e Botão Analógico

Algumas interfaces de comando para simulações de diversas situações em projetos interativos foram adicionadas à proposta do trabalho, um *Joystick* ou botão direcional analógicos, chaves seletoras e botões. Na PCB foi adicionado à barra de pinos três conexões para comunicação do *Joystick*, Vertical, Horizontal e *Select* (pinos 41, 43 e 45).

Nas chaves seletoras há a possibilidade de manter um acionamento constante conforme for a demanda da proposta, 4 chaves foram integradas à PCB onde seu acesso se faz na barra de pinos, sendo o (pino 28) relacionado a chave 1, (pino 30) à chave 2, (pino 8) à chave 3 e (pino 10) à chave 4.

Para simular acionamentos discretos ou constantes projetou-se 4 botões integrados à placa, onde seu acesso é realizado pela barra de pinos nos seguintes pontos, botão 1 (pino 18), botão 2 (pino 16), botão 3 (pino 14) e botão 4 (pino 12).

3.2.3 Sinalizações visuais

No sentido de sinalizações destinou-se 12 LEDs para realizar esta função, onde as sinalizações são divididas em duas, os 8 LEDs do primeiro tipo de sinalização são relacionados a barra de pinos do kit, onde podem ser controlados através do kit de desenvolvimento conectado, esses são os LEDs de 5 a 12 conforme a serigrafia da placa, na barra de pinos são correspondente as posições pinos ímpares de 1 a 7 e pinos pares de 20 a 26. Os outros 4 LEDs também exercem funções de sinalização, desta vez indicando o funcionamento correto de determinado circuito. O LED 13 corresponde ao correto funcionamento do circuito conversor RS485, já o LED 14 está relacionado ao regulador de tensão de 3,3 V e o LED 15 ao regulador de 5 V, por fim o LED 16 que sinaliza o funcionamento do transistor IRF520, selecionado por ser um transistor de alta potência, recomendado para aplicações que envolvem o chaveamento de correntes elevadas, como acionamento de solenoides e relés, controle de motores, amplificadores de áudio e diversas outras aplicações.

3.2.4 Módulo ADC

Para este recurso foi interfaceado à placa um módulo conversor AD/DA comercial baseado no CI PCF8591. Este circuito integrado possui uma conversão de dados analógico-digital de 8 bits com quatro canais de entrada e um conversor digital-analógico, além de sua comunicação ocorrer via barramento I2C. Neste módulo comercial alguns circuitos básicos já são acoplados para testar os canais do conversor ADC, tal como um sensor NTC (*Negative Temperature Coefficient*), um LDR (*Light Dependent Resistor*) e um potenciômetro. A escolha deste modelo fez sentido para este projeto, pois já é utilizado em aulas de sistemas embarcados na UTFPR-CM (Universidade Tecnológica Federal do Paraná, campus Campo Mourão).

3.2.5 Interfaces Seriais

Tendo em vista que alguns microcontroladores não são compatíveis com uma comunicação USB, disponibilizando apenas uma comunicação UART (*universal asynchronous recei-*

ver/transmitter), foi inserido um módulo conversor USB para TTL. Este módulo tem como base o CI CH340 que pode operar em 3,3 ou 5 V. Isto facilitará a comunicação de dados de um microcontrolador com um computador.

Outra interface que foi integrada à PCB é o módulo RS485 baseado no CI MAX485. O uso do protocolo RS485 é usado para transmissão de dados em longas distâncias em ambientes ruidosos, podendo-se comunicar com mais de um ponto simultaneamente. Protocolo muito usado em aplicações de automação industrial (Perez; Lima, 2007).

3.2.6 Módulo RTC

Foi incorporado à placa um circuito RTC (*Real Time Clock*) que é capaz de gerar através de um cristal oscilador uma temporização e assim criar um módulo que é capaz de contar o tempo e armazenar a data e horas. Com um chip DS1307, tem uma SRAM de 56 Bytes. Ele se comunica com um microcontrolador através da interface I2C e pode ser configurado para alarmes e interrupções. A bateria de *backup* garante que o RTC continue a funcionar mesmo quando a energia principal é desligada.

O DS1307 é um exemplo de um circuito integrado que implementa um módulo RTC.

O DS1307 se comunica com um microcontrolador (ou outro dispositivo) usando uma interface I2C (*Inter-Integrated Circuit*). O microcontrolador atua como mestre na comunicação I2C e envia comandos de leitura ou escrita para os registradores do DS1307, assim o DS1307 responde de acordo, fornecendo os dados solicitados ou executando as operações desejadas.

Como complemento a este circuito integrado, é prática comum adicionar uma memória EEPROM para armazenar informações e manter o dispositivo com a data e hora corretas. No presente caso, optou-se pela utilização do modelo 24C32, amplamente reconhecido e empregado em circuitos dedicados a essa finalidade.

3.2.7 Driver para acionamento de cargas via PWM

O Módulo PWM é um circuito de modulação por largura de pulso, amplamente usado na eletrônica como um exemplo de aplicação do controle da velocidade de motores elétricos, este modelo simples é construído com MOSFET IRF520 além de componentes básicos.

Onde sinais enviados do kit de desenvolvimento irão controlar o ciclo de trabalho do PWM ou seu (*Duty Cycle*), que é basicamente a porcentagem do ciclo de trabalho que está com sinal desligado.

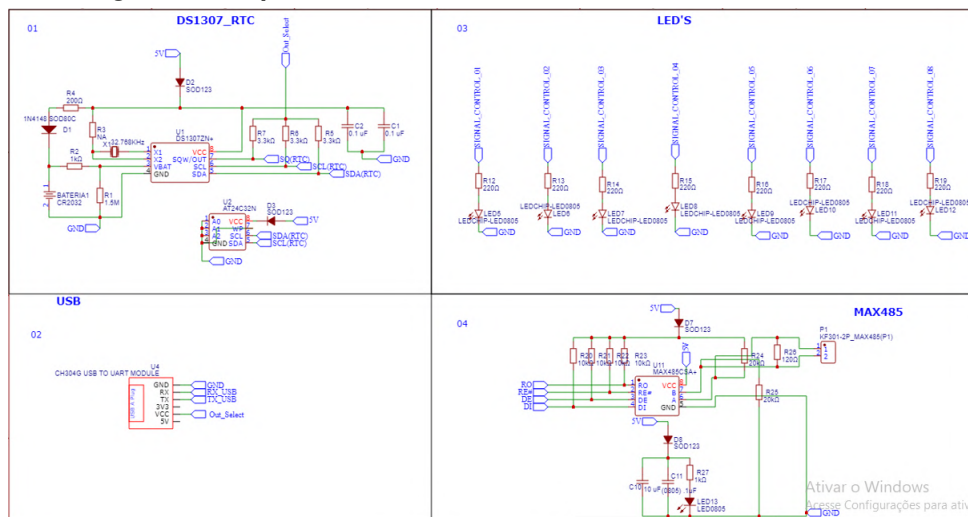
3.2.8 Alimentação do circuito projetado

Com relação ao circuito de alimentação da placa de expansão de interfaces, são necessárias duas tensões diferentes para alimentar distintos circuitos menores da placa. Ela deve ser alimentada por uma fonte de no mínimo 2.5 A e 12 V, via um conector P4, com a corrente máxima de operação em 2 A, ao utilizar os periféricos da placa, como cada módulo selecionado já é conhecido, a consulta para saber o consumo pode ser realizada via pesquisa na internet. A regulação de tensão da entrada para 3,3 V e 5 V é realizada usando o regulador MS1117 e outros componentes básicos, porém o MS1117 também pode ser substituído pelo LM1117 operando da mesma maneira. Desta forma, estes dois barramentos são disponibilizados e a configuração é realizada com os três pinos ao lado do conector P4, onde um *jumper* faz a seleção da tensão desejada, na serigrafia da placa tem-se a descrição da tensão de cada pino sendo o mais externo da placa 5 V e o voltado para o lado interno da placa 3 V onde o pino central é ligado a trilha que ira alimentar a placa com o valor desejado, adapta-se então o nível lógico do microcontrolador utilizado. Desta forma aumenta-se o nível de praticidade da solução desenvolvida.

3.3 Projeto do Esquemático elétrico

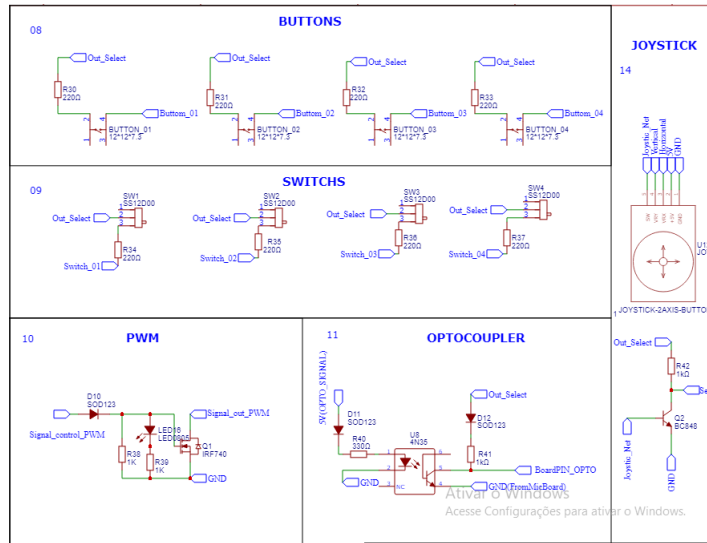
O esquemático dos circuitos foi dividido para melhor observação demonstrados nas Figuras 6, 7, 8 e 9.

Figura 6 – Esquemático dos circuitos RTC, MAX485, LEDs E USB



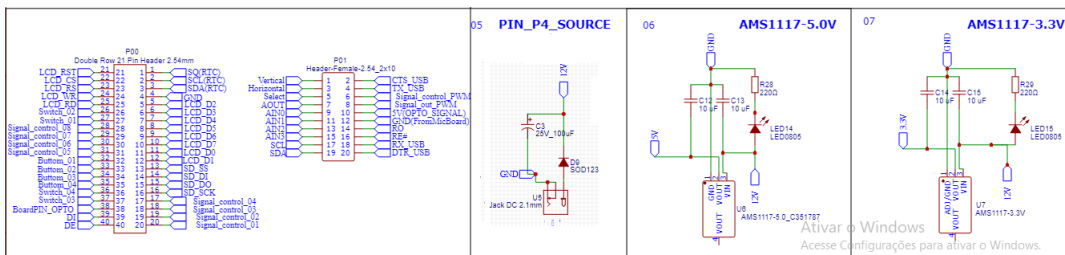
Fonte: Autoria própria (2023).

Figura 7 – Esquemático dos circuitos de botões, chaves, PWM, optoacoplador e joystick



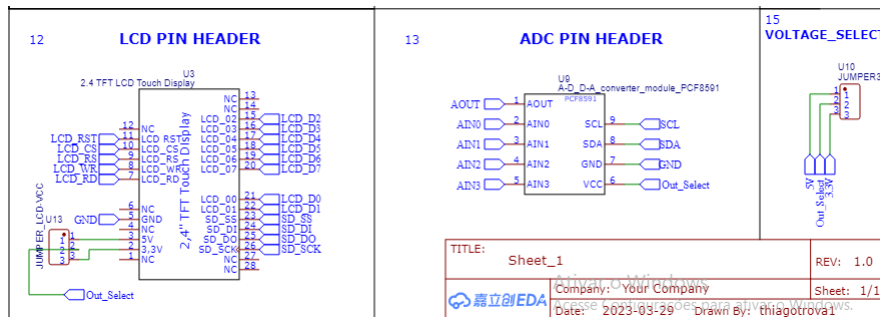
Fonte: Autoria própria (2023).

Figura 8 – Esquemático dos circuitos Regulador de tensão 3,3 V e 5 V, alimentação e barra de pinos



Fonte: Autoria própria (2023).

Figura 9 – Esquemático dos circuitos de display, conversor ADC e pino seletor de tensão



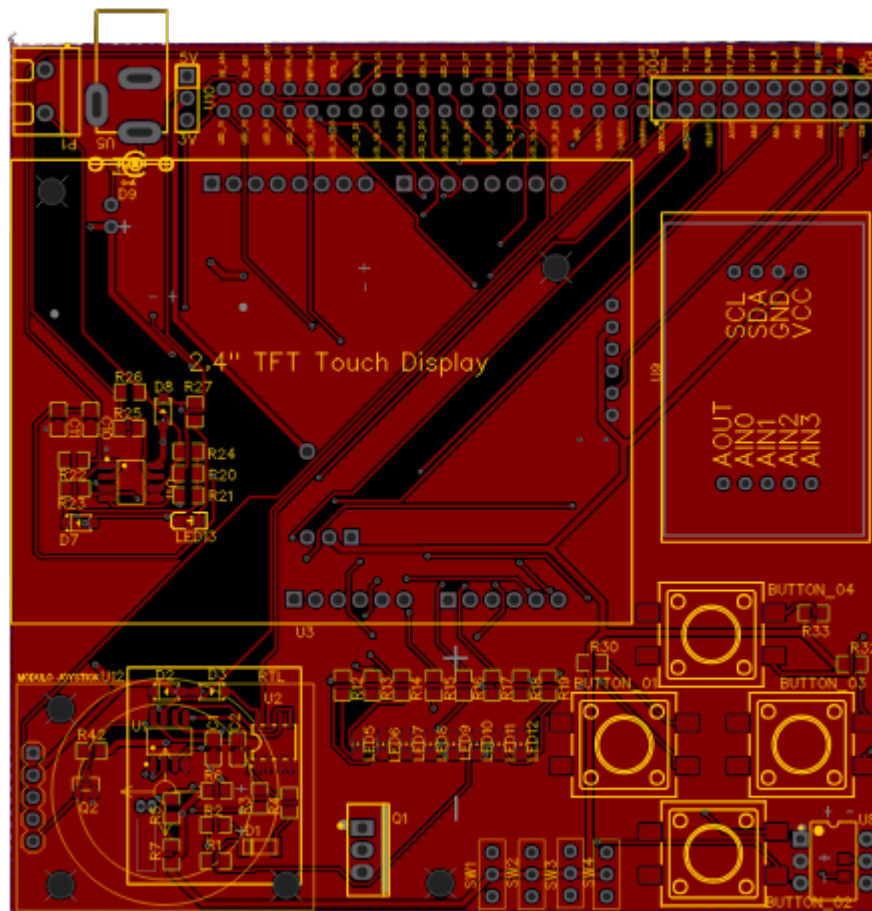
Fonte: Autoria própria (2023).

3.4 Projeto da placa de circuito impresso

Para o desenvolvimento da placa de circuito impresso, foram selecionados os recursos de *hardware* descritos na seção anterior. Posteriormente, com uma consulta nos *datasheets* de cada periférico e em módulos comerciais projetou-se o esquemático elétrico.

Realizado o projeto (similar ao idealizado anteriormente na Figura 4) obteve-se a placa 2D do projeto, apresentado na Figura 10, além dela foi possível desenvolver uma imagem ilustrativa do projeto em 3D, como mostra na Figura 11, nele é observado a disposição final dos componentes e dos conectores onde serão acoplados os módulos usados em conjunto a placa desenvolvida. Para desenvolvimento do protótipo foi usado componentes mistos tanto PTH (*Pin through hole*) quanto SMD (*Surface Mounting Technology*) onde a grande maioria é SMD. A criação dos arquivos Gerber resultou na produção da placa, ocorrida na fabricante internacional PCBWay.

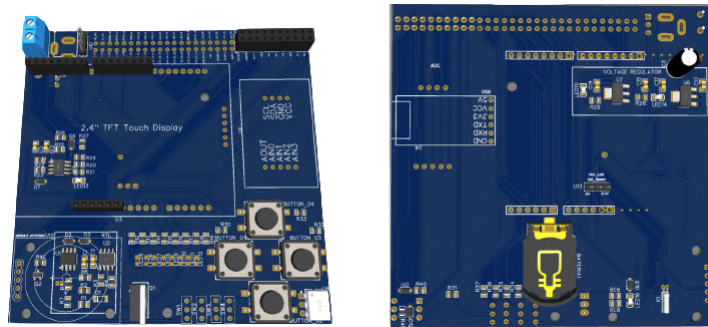
Figura 10 – Vista superior PCB 2D



Fonte: Autoria própria (2023).

A Figura 11, demonstra a PCB fabricada, exibindo os desenho técnico da área de contato e a serigrafia da placa, sem a fixação dos componentes, tanto na face principal quanto na posterior.

Figura 11 – Vista 3D da PCB projetada lado A e B.



Fonte: Autoria própria (2023).

3.5 Proposição de Prática de laboratório e elaboração de manual de uso

Para uma validação do funcionamento da placa, assim como já expor um exemplo prático do uso da mesma em sala de aula, a proposição de práticas de laboratório se faz justificada. Alguns dos desafios impostos pelas práticas de laboratório é realizar a aquisição de um sensor analógico, usada em conjunto um conversor analógico digital, para que haja o real entendimento de protocolos de comunicação, e o funcionamento do conversor analógico digital, além de testar a capacidade do aluno em relação criação de bibliotecas específicas observando o funcionamento do dispositivo em questão. O laboratório pode ser solucionado junto a qualquer kit de desenvolvimento, com finalidade de usar as interfaces do circuito proposto no trabalho, assim foi selecionado o kit Arduino Uno, para realizar as soluções dos laboratórios descritos a escolha do kit Arduino se deu pelo fato de já estar em mãos e sua facilidade de uso.

3.5.1 Proposta Prática de Laboratório 01

Na primeira prática de laboratório proposta no trabalho, iniciamos com uma rotina que ilustra o funcionamento dos LEDs usando os LEDs 1, 2, 3 e 4. Na etapa subsequente, empregaremos quatro botões, cada um associado a um LED específico. Quando um botão é pressionado, o LED correspondente se acenderá. Simultaneamente, no Serial Monitor do Arduino fornecerá uma mensagem indicando quais botões estão soltos e quais estão pressionados. Esse método simplifica a compreensão e permite a observação do correto funcionamento da placa que foi desenvolvida, incluindo os testes de funcionalidade dos botões e LEDs do dispositivo.

3.5.2 Proposta Prática de Laboratório 02

Na segunda prática de laboratório, o procedimento é muito semelhante ao da primeira prática, com a diferença de que, em vez de usar botões, empregam-se chaves seletoras de posição. São utilizadas quatro chaves, cada uma correspondendo a um LED específico, semelhante à prática anterior. Nesta etapa, é apresentada uma rotina de testes para os LEDs, que

segue o mesmo padrão da prática anterior, mas agora testando os LEDs 5, 6, 7 e 8. Isso permite demonstrar o funcionamento adequado de todos os LEDs na placa, bem como o das chaves seletoras.

3.5.3 Proposta Prática de Laboratório 03

Uma das propostas à prática de laboratório 03 foi a utilização de Conversor ADC para leitura e mostrar os dados através de *display* com o kit de interface. O objetivo desta prática de laboratório é capacitar os alunos a utilizar um conversor ADC para ler dados de sensores de potenciômetro e luz . Os dados adquiridos serão então processados por meio de *software* para converter valores em grandezas conhecidas. Posteriormente, os resultados serão exibidos em um *display* LCD.

O procedimento da configuração do *Hardware*, deve conectar os sensores de temperatura e luz aos canais analógicos do ADC, conectar o *display* LCD ao microcontrolador, estabelecendo a comunicação adequada.

Na programação deve-se implementar o código para inicialização do ADC, configuração dos canais e aquisição de dados dos sensores, criar a lógica para exibir os resultados no *display* LCD, possibilitando a intercalação dos dados de luz e do potenciômetro.

E por fim, para os testes e depuração, deve testar o sistema, usando a ferramenta Serial Monitor da IDE do Arduino ,verificando se os valores lidos dos sensores estão sendo enviados corretamente, verificar se os resultados exibidos no *display* estão de acordo.

Para a documentação necessária, deve haver um registro detalhado do processo de montagem, programação e testes. Com isso, os alunos deverão ser capazes de desenvolver um sistema que utiliza um ADC para ler dados de sensores de luz e potenciômetro, apresentar esses dados em grandezas digitais, e exibir os resultados em um *display* LCD. Além disso, o sistema deverá permitir o revezamento dos dados a serem exibidos no *display* LCD.

Essa prática proporcionará aos alunos a oportunidade de aplicar conhecimentos teóricos em eletrônica e programação para construir um sistema prático e funcional. Além disso, eles ganharão experiência na integração de diferentes componentes e na implementação de uma interface de controle intuitiva.

3.5.4 Proposta Prática de Laboratório 04

A quarta prática de laboratório proposta para testar o kit de desenvolvimento envolve o uso do módulo de *joystick*. Nessa prática, o objetivo é utilizar o módulo de *joystick* em conjunto com o módulo LCD. Isso permitirá representar a posição do *joystick* no visor LCD. Ao mesmo tempo, a ferramenta *Serial Monitor* no ambiente de desenvolvimento do Arduíno mostrará os valores analógicos recebidos em relação à posição horizontal ou vertical do *joystick*. Esses

valores fornecem informações sobre a direção para a qual o *joystick* está apontando. Dessa forma, essa prática possibilita realizar testes tanto no módulo de *joystick* quanto no módulo LCD, que pode ser conectado ao dispositivo desenvolvido no trabalho em questão.

3.5.5 Proposta Prática de Laboratório 05

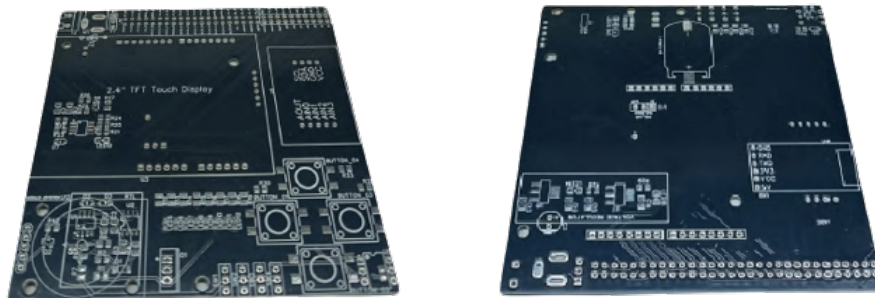
A quinta prática de laboratório proposta envolve a utilização do DS1307 presente no kit de interface deste projeto. Consiste em adquirir os dados do DS1307 por meio do protocolo de comunicação I2C. Após a aquisição dos dados, o kit de desenvolvimento, em conjunto com o kit de expansão de interfaces, processará as informações e as apresentará em um *display* LCD conectado ao kit de interface. Isso permitirá realizar testes e demonstrar o perfeito funcionamento de mais um dos módulos do kit. Assim como nas práticas anteriores, a data e a hora serão exibidas simultaneamente no *display* LCD e também disponibilizadas para visualização no *Serial Monitor* da IDE do Arduíno.

4 RESULTADOS

Através dos estudos realizados neste trabalho, desenvolveu-se uma placa de circuito impresso, com objetivo de unir funcionalidades que expandem o uso de kits de desenvolvimento. A placa auxilia tanto desenvolvedores, como alunos de cursos que envolvam sistemas embarcados ou microcontrolados, contornando a necessidade de montar placas confusas em *Proto-boards*, assim o tempo gasto nos projetos é melhor aproveitado, com maior foco no desenvolvimento da parte programacional. Assim o kit de expansão desenvolvido fomenta o aprendizado de jovens e entusiastas, e assim se entrega mais segurança no desenvolvimento e praticidade.

Na Fotografia 8 estão demonstradas as placas finais sem os componentes e na Fotografia 9 com o componentes montados.

Fotografia 8 – Placa sem componentes - lado A e B.



Fonte: Autoria própria (2023).

Conforme evidenciado na Fotografia 9, no lado B, é perceptível a presença de um fio visível. Essa medida foi adotada como solução para adequar a placa ao seu correto funcionamento. O referido fio alimenta o pino de 3.3V do *display*. Esse ajuste foi necessário devido a uma interpretação equivocada, na qual se supôs ser uma ou outra tensão. No entanto, o *display* TFT utiliza ambas simultaneamente.

Uma segunda adaptação necessária consistiu na inversão dos pinos SCL e SDA da memória utilizada em conjunto com o DS1307. Este equívoco passou despercebido durante a análise final da placa de circuito impresso, contudo, já foi corrigido no esquemático.

Fotografia 9 – Placa com componentes - lado A e B.

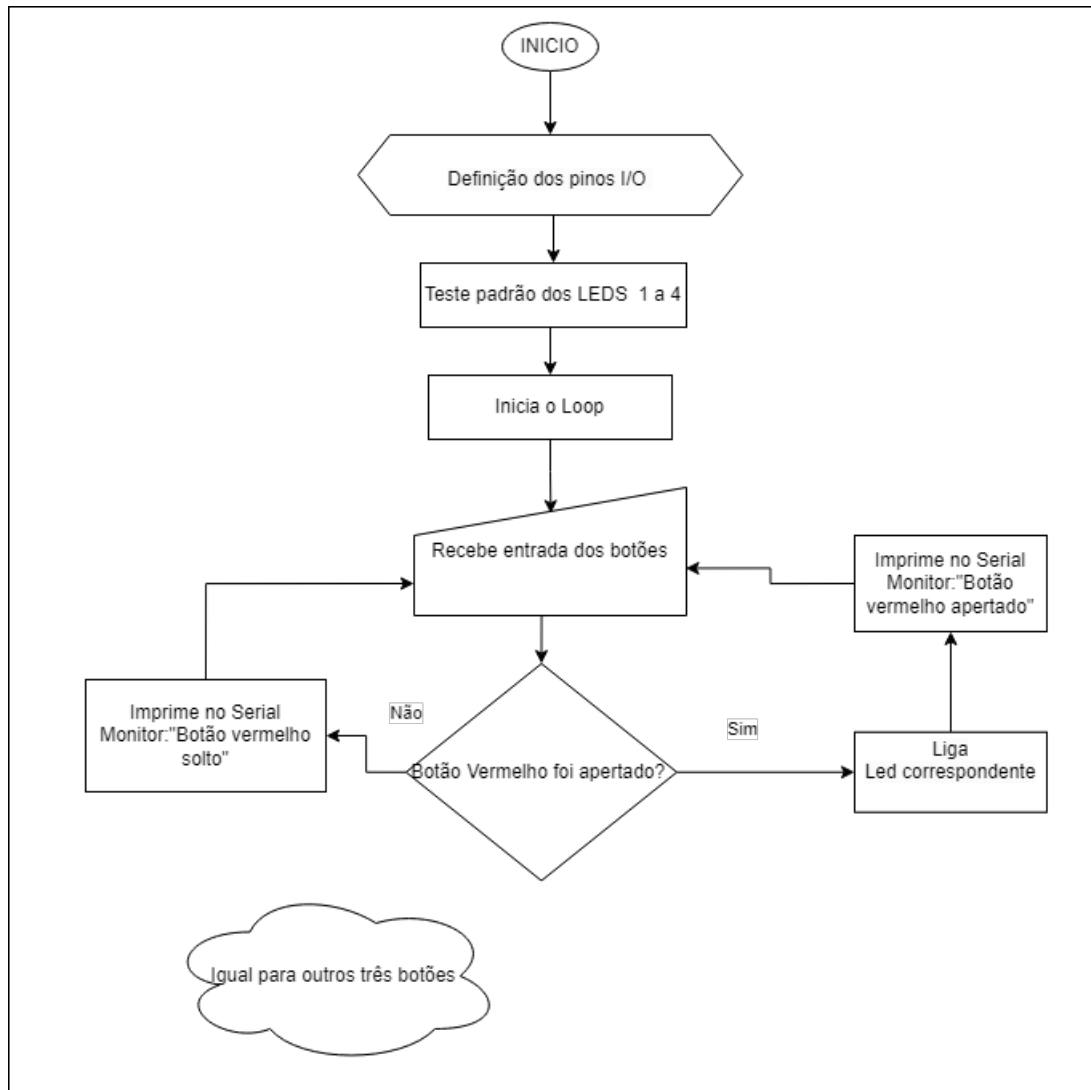


Fonte: Autoria própria (2023).

4.1 Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 1

Na primeira prática de laboratório, inicia com uma demonstração do funcionamento dos LEDs 1, 2, 3 e 4. Em seguida, incorpora quatro botões, cada um controlando um LED específico. Quando um botão é pressionado, o LED correspondente acende, enquanto o *Serial Monitor* do Arduino exibe informações sobre o estado dos botões. No Apêndice B é apresentado o código utilizado para implementar o Laboratório 01 e na Figura 12 é apresentado o fluxograma deste código e na Fotografia 10 demonstrou-se através de uma imagem do funcionamento do circuito.

Figura 12 – Fluxograma prática de laboratório 01

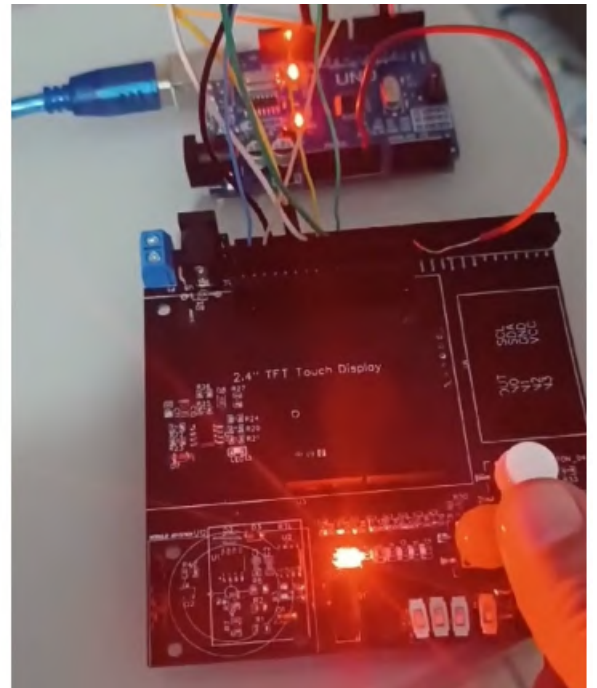


Fonte: Autoria própria (2023).

No instante da captura da imagem, como é evidenciado na saída exibida pela ferramenta *Serial Monitor* do IDE Arduino, é perceptível que o botão vermelho estava sendo pressionado, o que resultou na ativação do LED correspondente.

Fotografia 10 – Imagem demonstrativa da 1ª prática proposta.

```
Botão 01 Vermelho Apertado  
Botão 02 Azul Solto !!  
Botão 03 Preto Solto !!  
Botão 04 Branco Solto !!
```



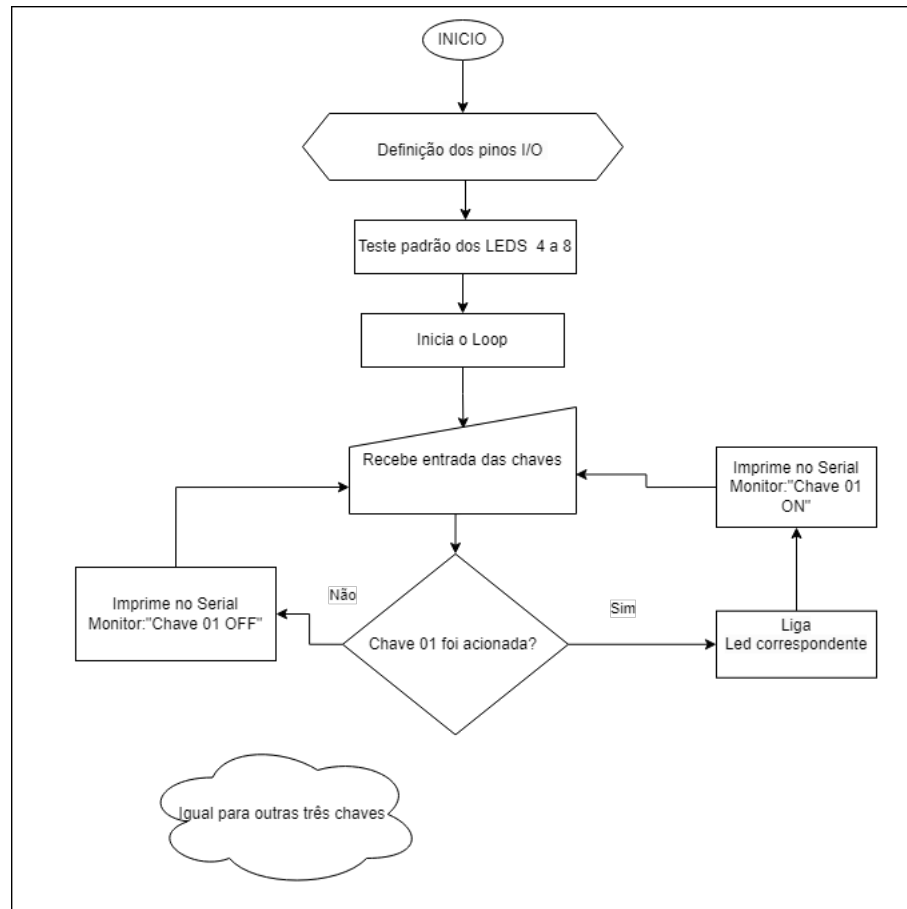
Fonte: Autoria própria (2023).

4.2 Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 2

Na terceira prática de laboratório, o procedimento é semelhante à primeira, exceto pelo uso de chaves seletoras de posição em vez de botões. Quatro chaves controlam LEDs específicos, seguindo o padrão da prática anterior. Uma rotina de testes é apresentada para os LEDs 5, 6, 7 e 8, demonstrando o funcionamento desses LEDs e das chaves seletoras. No Apêndice C é apresentado o código utilizado para implementar o Laboratório 03 e na Figura 13 é apresentado o fluxograma deste código.

Observa-se na Fotografia 11 uma imagem do funcionamento deste código. No caso, o momento da captura demonstra que as Chaves 01 a 04 estão acionadas e os LEDs correspondentes estão ligados.

Figura 13 – Fluxograma prática de laboratório 02



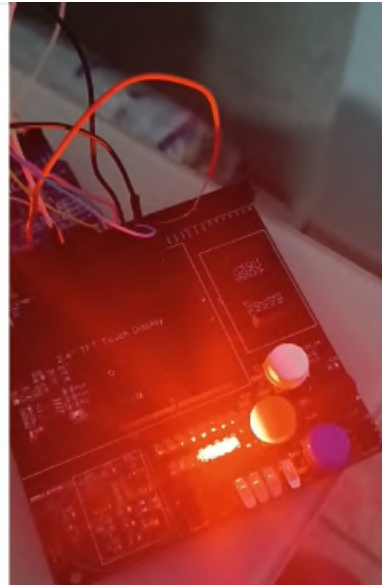
Fonte: Autoria própria (2023).

4.3 Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 3

A prática de laboratório 03 envolve a utilização de um Conversor ADC para ler dados de sensores, como potenciômetros e sensor de luz. Os sinais de entrada analógicos são lidos e convertidos para uma informação digital, que são apresentados no *display* LCD. No Apêndice A é apresentado o código utilizado para implementar o Laboratório 01 e na Figura 14 é apresentado o fluxograma deste código e na Fotografia 12 e Fotografia 13 apresenta-se imagens do funcionamento do circuito.

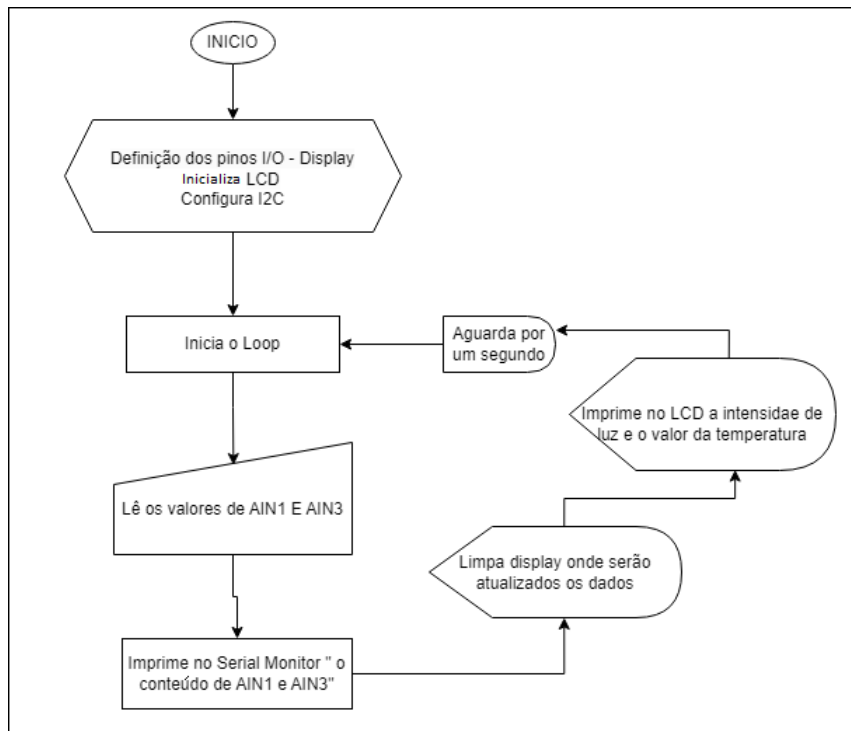
Fotografia 11 – Imagem demonstrativa da 2ª prática proposta.

```
Chave 1 ON !!
Chave 2 ON !!
Chave 3 ON !!
Chave 4 ON !!
```



Fonte: Autoria própria (2023).

Figura 14 – Fluxograma prática de laboratório 03



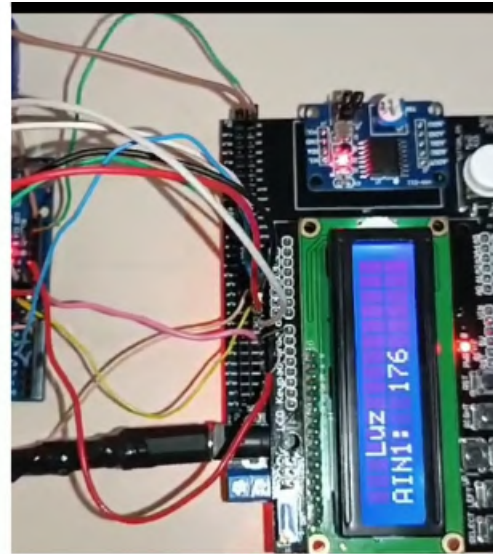
Fonte: Autoria própria (2023).

4.4 Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 4

Na quarta prática de laboratório, propõe-se testar o kit de desenvolvimento, envolvendo o uso do módulo de *joystick*. O objetivo é combinar o módulo de *joystick* com o módulo LCD para exibir a posição do *joystick* no visor. Simultaneamente, a ferramenta *Serial Monitor* no ambiente de desenvolvimento do Arduino apresentará os valores analógicos correspondentes à

Fotografia 12 – Imagem demonstrativa da 3ª prática proposta demonstrando o valor do resistor dependente de luz.

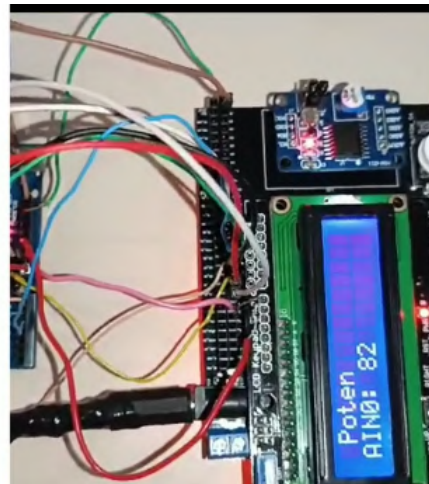
```
-> Poten: 85  
-> Luz: 176
```



Fonte: Autoria própria (2023).

Fotografia 13 – Imagem demonstrativa da 3ª prática proposta demonstrando o valor do potenciômetro.

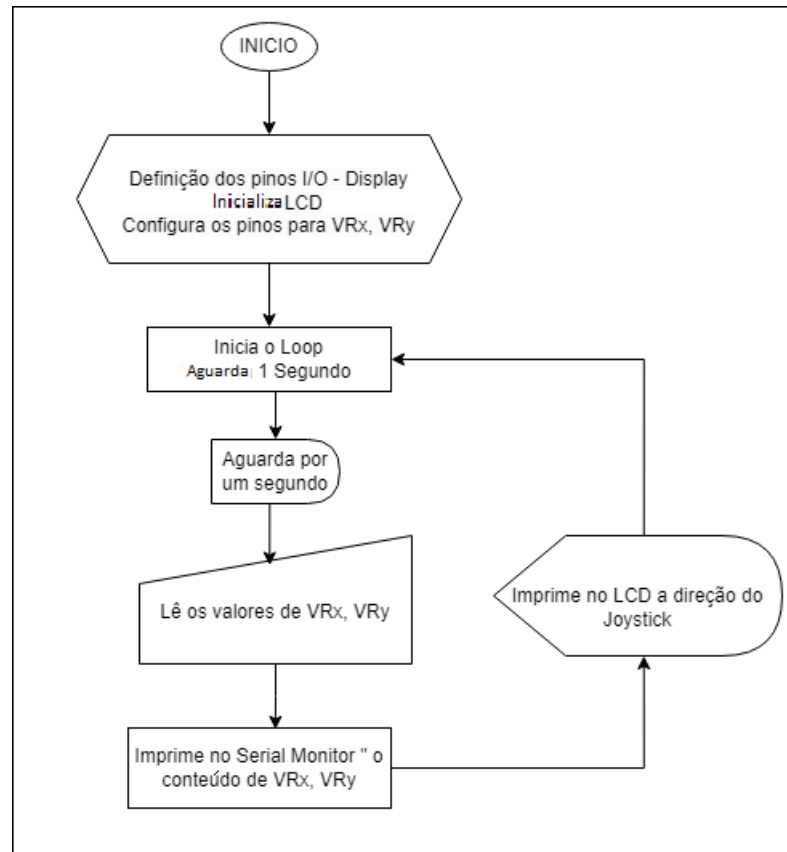
```
-> Luz: 176  
-> Poten: 82
```



Fonte: Autoria própria (2023).

posição horizontal ou vertical do *joystick*. No Apêndice D é apresentado o código utilizado para implementar o Laboratório 04 e na Figura 15 é apresentado o fluxograma deste código e na Fotografia 14 o funcionamento do mesmo.

Figura 15 – Fluxograma prática de laboratório 04



Fonte: Autoria própria (2023).

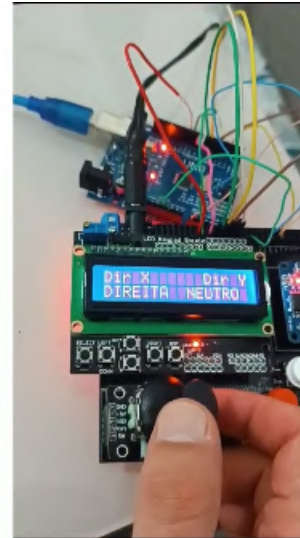
Na Fotografia 14, é visível o *joystick* sendo movido para a direita. Por meio da saída no *display* LCD, o código indica claramente em qual plano, seja horizontal ou vertical, o *joystick* está apontando, e isso pode ser confirmado com precisão por meio da ferramenta *Serial Monitor* que demonstra o valor analógico lido do *joystick* e colocado em um intervalo entre 0 e 180, onde 90 é o botão centralizado em ambos eixos, 0 representa esquerda no eixo x e baixo no eixo y, assim como 180 representa direita para o eixo x e cima para o eixo y.

Fotografia 14 – Imagem demonstrativa da 4ª prática proposta.

```

Valor VRx: 89
Valor VRy: 88
Valor VRx: 89
Valor VRy: 88
Valor VRx: 89
Valor VRy: 88
Valor VRx: 90
Valor VRy: 89

```



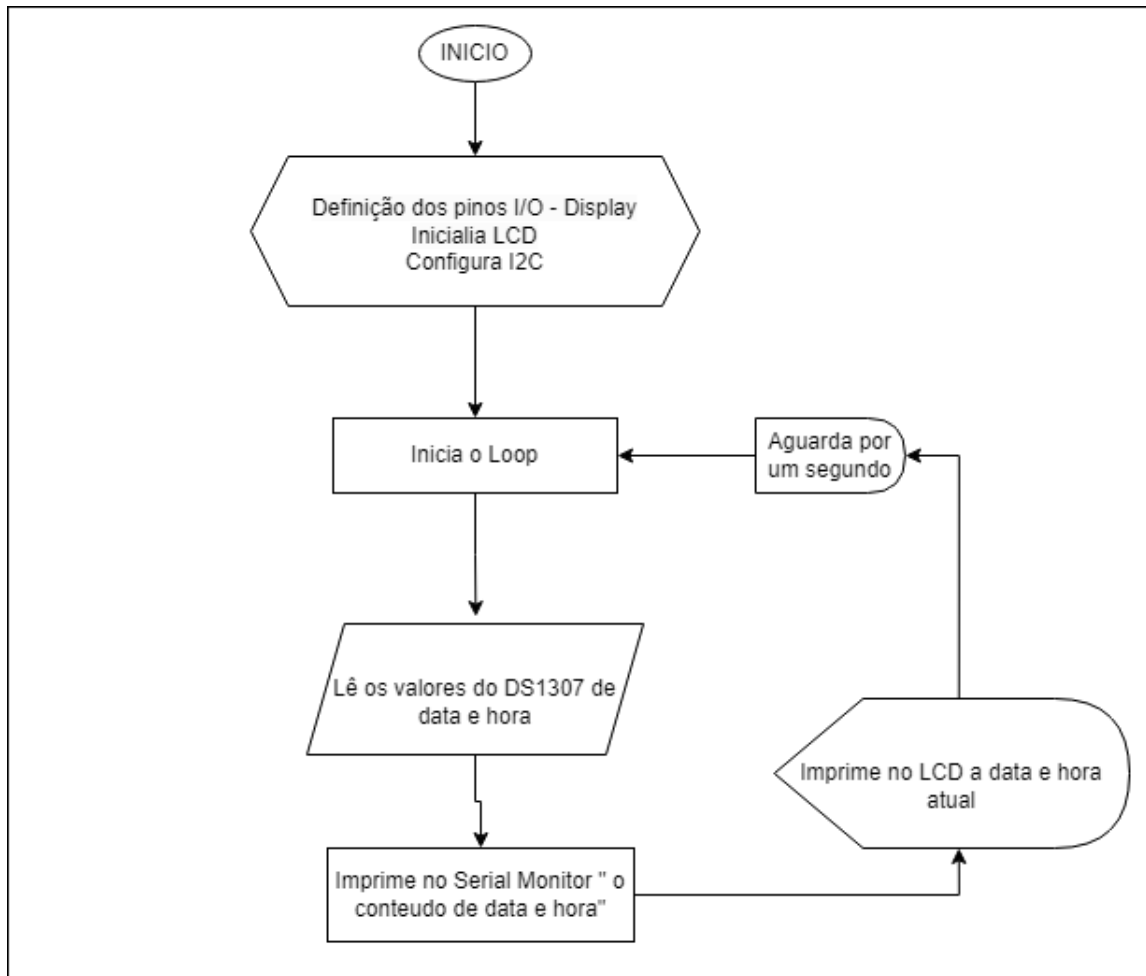
Fonte: Autoria própria (2023).

4.5 Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório 5

Na quinta prática de laboratório, foi proposto a utilização do DS1307. Esta prática, é focada na aquisição de dados do DS1307 através do protocolo I2C. Após a coleta dos dados, o kit de desenvolvimento, se encarregará de processar essas informações e exibi-las no *display* LCD conectado ao kit de interface. No Apêndice E é apresentado o código utilizado para implementar o Laboratório 05 e na Figura 16 é apresentado o fluxograma deste código e na Fotografia 15 o projeto em funcionamento.

Na Fotografia 15 da prática de laboratório, é evidente a capacidade do kit em exibir a hora em tempo real, gerada pelo DS1307. Os dados adquiridos são processados e apresentados no *display*, mostrando tanto a data quanto a hora. Os mesmos valores são refletidos no *Serial Monitor*, indicando que o processo iniciou no segundo 0 e chegou ao segundo 4 até momento da captura da imagem.

Figura 16 – Fluxograma prática de laboratório 05

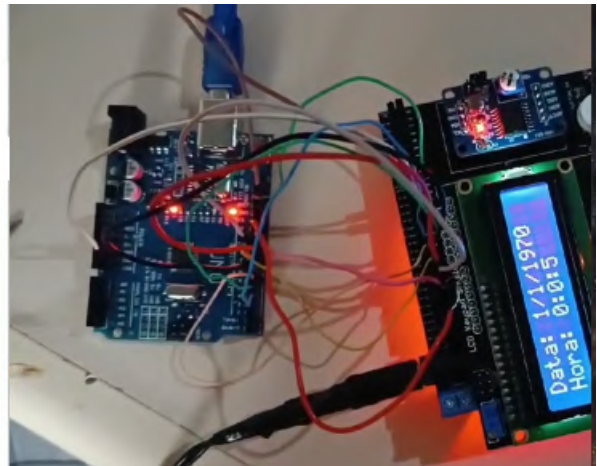


Fonte: Autoria própria (2023).

Um vídeo demonstrando o funcionamento de cada um das práticas de laboratório, para que possa ser feita uma comprovação visual do funcionamento da solução está disponível em (Trova, 2023).

Fotografia 15 – Imagem demonstrativa da 5ª prática proposta.

```
Data: 0/0/1970 Hora: 0:0:0  
Data: 1/1/1970 Hora: 0:0:1  
Data: 1/1/1970 Hora: 0:0:2  
Data: 1/1/1970 Hora: 0:0:3  
Data: 1/1/1970 Hora: 0:0:4
```



Fonte: Autoria própria (2023).

5 CONCLUSÃO

Através dos estudos realizados neste trabalho, foi possível desenvolver uma placa de circuito impresso com o objetivo de unir funcionalidades que expandam o uso de kits de desenvolvimento. A placa auxilia tanto desenvolvedores, como alunos de cursos que envolvam sistemas embarcados ou microcontrolados, contornando a necessidade da montagem de circuitos em *Protoboards*. Desta forma, espera-se que o tempo gasto no desenvolvimento de projetos seja melhor aproveitado, com maior foco no desenvolvimento do *software* para sistemas embarcados.

Ao longo deste estudo, explorou-se o cenário em constante evolução da engenharia eletrônica, destacando a crescente importância dos sistemas embarcados e microcontroladores em diversas aplicações, desde equipamentos clínicos até a automação residencial. Foi evidenciada a necessidade de adaptar abordagens de ensino superior para incorporar práticas e aplicações do mundo real.

A pesquisa ressaltou a relevância das abordagens práticas no ensino de microcontroladores e sistemas embarcados, com destaque para a flexibilidade e acessibilidade proporcionadas por kits de desenvolvimento, como o Arduíno. Esses kits desempenham um papel crucial na formação de engenheiros eletrônicos e demais profissionais da área tecnológica.

Neste sentido, foi introduzida a ideia de um kit de expansão de interfaces genérico para kits de desenvolvimento, que oferece uma ampla variedade de módulos de entrada e saída, tornando-o versátil para diferentes disciplinas de graduação. Isso amplia as possibilidades no ensino de microcontroladores e sistemas embarcados, permitindo que os alunos se familiarizem com diversos dispositivos e tecnologias.

Os objetivos deste trabalho levaram ao desenvolvimento de um módulo de interfaceamento que inclui funcionalidades como *display* LCD, botões, chaves seletoras, modulador por largura de pulso, conversor ADC e interfaces de comunicação serial. Esse módulo representa uma ferramenta valiosa para alunos e desenvolvedores, economizando tempo na montagem de circuitos complexos e permitindo um foco maior no desenvolvimento de *software*.

Os projetos de laboratório demonstraram a eficácia do kit de interfaces em várias aplicações, desde a leitura de sensores analógicos até o controle de LEDs e dispositivos por meio de botões e chaves seletoras. Esses projetos destacam o potencial do kit de desenvolvimento em promover a aprendizagem prática e o desenvolvimento de habilidades em sistemas microcontrolados e embarcados.

Em síntese, este estudo contribui para o avanço do ensino de microcontroladores e sistemas embarcados, fornecendo uma ferramenta versátil e prática que pode ser aplicada em diversas disciplinas de engenharia eletrônica.

Em trabalhos futuros pode-se verificar a eficácia da solução desenvolvida em sala de aula, bem como desenvolver materiais didáticos e manuais para auxiliar no uso da placa desenvolvida.

REFERÊNCIAS

- ALVES, M. L.; COSTA, J. R. de F.; BEZERRA, C. I. M. Um relato de experiência: Ensinando robótica por meio de microcontroladores em uma escola profissional de ensino médio. *In: SBC. Anais do XXVI Workshop sobre Educação em Computação*. [S.l.], 2018.
- AMORIM, V. J. P. de; MONLEVADE, J. Ensino de sistemas computacionais embarcados através de projetos práticos em uma universidade pública brasileira. *In: II Fórum de Educação em Engenharia de Computação*. [S.l.: s.n.], 2016.
- ARAÚJO, M. N.; FERREIRA, W. T.; OLIVEIRA, R. de. Sistema embarcado arduino como ferramenta de apoio ao ensino de eletrônica básica e programação nas engenharias e na área de computação. *ERIMT*, p. 36, 2016.
- ARAÚJOI, C. G.; OLIVEIRAI, F. H. F. Embracer: um caso de imitação à inovação. *Economia e Desenvolvimento*, v. 32, p. e6–e6, 2020.
- AXELSON, J. Serial port complete: COM ports, usb virtual com ports, and ports for embedded systems. *Lakeview Research LLC Madison*, v. 3, p. 47–49, 1998.
- BARROS, E.; CAVALCANTE, S. *Introdução aos sistemas embarcados*. [S.l.], 2010. 36 p. Universidade Federal de Pernambuco.
- CAETANO, V. P. *Automação Residencial Utilizando Raspberry Pi 3 B*. 2021.
- CARDOSO, O. E. *et al.* Desenvolvimento de uma plataforma de aprendizagem de sistemas embarcados baseada em esp32. *In: Brazilian Congress of Engineering Education*. [S.l.: s.n.], 2020.
- CARRION, A. P.; WERNER, C. *Software embarcado e para web–tendencia de mercado para alta tecnologia*. [S.l.]: UNIPAR, 2012.
- CASARA, V. P.; ZAMPARETTE, R. L. B.; BELOLI, A. S. R. Utilização da plataforma arduino como forma de contribuição para a melhoria da qualidade da formação acadêmica dos alunos e introdução de novas práticas pedagógicas no curso de engenharia elétrica. *In: Anais: XLII–Congresso Brasileiro de Educação de Engenharia. Juiz de Fora, MG*. [S.l.: s.n.], 2014.
- CHEN, T. *et al.* Embedded system education in zhejiang university. *In: 7th Conference on 7th WSEAS International Conference on Applied Informatics and Communications*. [S.l.: s.n.], 2007. p. 337–344.
- CHOI, C. A microcontroller applications course and the freescale’s microcontroller student learning kit. *In: 2008 Annual Conference & Exposition*. [S.l.: s.n.], 2008. p. 13–58.
- EASYEDA - A simple and powerful electronic circuit design tool. 2023. Disponível em: <https://easyeda.com>. Acesso em: 10 nov. 2023.
- FREITAS, C. M. **Modbus: fundamentos e aplicações -Saiba mais sobre o Protocolo Modbus**. 2014. Disponível em: <https://embarcados.com.br/protocolo-modbus/>. Acesso em: 10 nov. 2023.
- FREITAS, C. M. **Saiba mais sobre redes de comunicação em RS-485 - Embarcados**. 2017. Disponível em: <https://embarcados.com.br/redes-de-comunicacao-em-rs-485/>. Acesso em: 10 nov. 2023.

- HOROWITZ, H. W. P. The art of electronics. **New York: Cambridge University Press**, v. 3, p. 704–718, 2015.
- I2C-BUS specification and user manual. [S.l.], 2021. Disponível em: <https://www.nxp.com/docs/en/user-guide/UM10204.pdf>. Acesso em: 10 nov. 2023.
- KATO, E. R. R. Método de ensino da disciplina de “laboratório de microcontroladores e aplicações” do curso de engenharia de computação da ufscar. 2013.
- KOSLOWSKI, M. A. **Kit didático de baixo custo para o ensino de microcontroladores da família MCS-51 USB**. 2014. Dissertação (Trabalho de Conclusão de Curso (Graduação)) — Universidade Tecnológica Federal do Paraná, Curitiba, 2014.
- LEE, E. A. **Embedded Software” Advances in Computers, M Zelkowitz, ed., Vol. 56**. [S.l.]: Academic Press, 2002.
- LIMA, E. M. S.; NOBRE, A. Y. M.; ALENCAR, R. A. E. d. Automação residencial de baixo custo com arduino mega e ethernet shield. **Curso de Sistemas de Informação no Centro Universitário Estácio do Ceará**, 2015.
- MA, C. *et al.* Low cost avr microcontroller development kit for undergraduate laboratory and take-home pedagogies. *In: IEEE. 2010 2nd International Conference on Education Technology and Computer*. [S.l.], 2010. v. 1, p. V1–35.
- MILANI, F. R.; MILANI, M. P. C. **Utilização de recursos tecnológicos no ensino superior**. [S.l.], 2010.
- MOORE, G. E. *et al.* **Cramming more components onto integrated circuits**. [S.l.]: McGraw-Hill New York, 1965.
- PEREIRA, L. A. M. *et al.* Software embarcado, o crescimento e as novas tendências deste mercado. **Revista de Ciências Exatas e Tecnologia**, v. 6, n. 6, p. 85–94, 2011.
- PEREZ, C. R.; LIMA, D. d. O. Proposta de protocolos de comunicação para um ambiente de computação ubíqua para automação residencial. *In: XIII Congreso Argentino de Ciencias de la Computación*. [S.l.: s.n.], 2007.
- PORTAL EMBARCADOS. **Relatório da Pesquisa sobre o Mercado Brasileiro de Sistemas Embarcados e IoT 2021**. 2021. Disponível em: <https://embarcados.com.br/relatorio-da-pesquisa-sobre-o-mercado-brasileiro-de-sistemas-embarcados-e-iot-2021>. Acesso em: 10 nov. 2023.
- PRADO, B. O. P. **Construção e geração de software dependente de hardware de modelos de sistema de alto nível**. 2012. Tese (Doutorado) — Universidade Federal de Pernambuco, Recife, 2012.
- RESNICK, M. Technologies for lifelong kindergarten. **Educational technology research and development**, Springer, v. 46, n. 4, p. 43–55, 1998.
- ROBOCORE. **Comparação Entre Protocolos de Comunicação Serial**. 2021. Disponível em: <https://www.robocore.net/tutoriais/comparacao-entre-protocolos-de-comunicacao-serial.html>. Acesso em: 10 nov. 2023.
- SILVA, F. R. da *et al.* Desenvolvimento de plataforma embarcada para ensino. **Research, Society and Development**, Grupo de Pesquisa Metodologias em Ensino e Aprendizagem em Ciências, v. 8, n. 4, p. e2084757, 2019.

SILVA, P. I. S. da *et al.* Robótica educacional como ferramenta interdisciplinar desenvolvendo uma consciência tecnológica em escolas públicas. **Mostra Nacional de Robótica**, 2012. Disponível em: <http://www.sistemaolimpo.org/midias/uploads/78b8e0e65c68015697543e14252c755f.pdf>. Acesso em: 10 nov. 2023.

TAURION, C. **Software embarcado-a nova onda da Informática**. [S.l.]: Brasport, 2005.

THE BRAINY INSIGHTS. **Embedded system market to hit USD 138.45 billion by 2028 - Advancements in consumer electronics, technologies and IoT solutions, business segments, market share analysis regional growth**. [S.l.], 2022. Disponível em: <https://www.globenewswire.com/news-release/2022/02/03/2378667/0/en/Embedded-System-Market-to-hit-USD-138-45-billion-by-2028>. Acesso em: 21 out. 2023.

TROVA, T. H. **(39) TCC 2 Thiago Trova Demonstração de Funcionamento - YouTube**. 2023. (Video de Demonstração do Kit). Disponível em: https://www.youtube.com/watch?v=tzCPSYd9yo4&ab_channel=THIAGOHENRIQUETROVA. Acesso em: 10 nov. 2023.

YELISETTY, S. *et al.* Um levantamento de fatores humanos e técnicos em garantia de qualidade de projeto de software embarcado aeronáutico no brasil. *In*: SBC. **Anais do VI Workshop sobre Aspectos Sociais, Humanos e Econômicos de Software**. [S.l.], 2021. p. 1–10.

ZELENOVSKY, R.; MENDONÇA, A. **Introdução aos sistemas embutidos**. 1999. Disponível em: <http://www.mzeditora.com.br/artigos/embut.htm>. Acesso em: 21 out. 2023.

APÊNDICE A – Algoritmo pratica de laboratório 01º.

Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório

1.

```

include <Wire.h>
include <LiquidCrystal.h>

const int PCF8591Address = 0x48;// Endereço do PCF8591
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);// Pinos para ligar o display

void setup()

Serial.begin(9600);
  lcd.begin(16, 2); // Define o número de colunas e linhas do display
  Wire.begin(); // Inicializa a comunicação I2C
void loop()
// Lê os valores das entradas analógicas do PCF8591
int analogValue1 = readPCF8591(1); // Leitura do canal AIN1
int analogValue3 = readPCF8591(3); // Leitura do canal AIN3
// Exibe os valores no display LCD
lcd.clear(); // Limpa o display

  int Porta =0;

  switch (Porta)
case 0:
lcd.setCursor(0, 0);
  lcd.print("Poten ");
  lcd.setCursor(0, 1);
  lcd.print("AIN0: ");
  lcd.print(analogValue3);
  Serial.print("Poten: ");
  Serial.println(analogValue3);
  delay(2000);
  Porta = 1;
  case 1:
lcd.setCursor(0, 0);
  lcd.print("Luz ");
  lcd.setCursor(0, 1);
  lcd.print("AIN1: ");

```

```
    lcd.print(analogValue1);  
    Serial.print("Luz: "); Serial.println(analogValue1);  
    delay(2000);  
    Porta = 0;  
  
    delay(1000); // Aguarde 1 segundo antes de fazer uma nova leitura  
  
int readPCF8591(int channel)  
  
Wire.beginTransmission(PCF8591Address);  
    Wire.write(channel); // Seleciona o canal AIN0, AIN1, AIN2 ou AIN3  
    Wire.endTransmission();  
    Wire.requestFrom(PCF8591Address, 1);  
return Wire.read();  
  
Lesser General Public License (LGPL)
```

APÊNDICE B – Algoritmo pratica de laboratório 02°.

Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório

2.

```

const int chavePin6 = 6; //Pinodachave
const int chavePin7 = 7; //Pinodachave
const int chavePin8 = 8; //Pinodachave
const int chavePin9 = 9; //Pinodachave

const int ledPin2 = 2; //PinodoLED
const int ledPin3 = 3; //PinodoLED
const int ledPin4 = 4; //PinodoLED
const int ledPin5 = 5; //PinodoLED

//Auxiliarparatesteinicialdeledsintvelocidadeled = 115;
void setup()
    Serial.begin(9600);
    // Definição dos pinos I/O
    pinMode(chavePin6, INPUT_PULLUP); //Configuraopinodachavecomoentrada
    pinMode(chavePin7, INPUT_PULLUP); //Configuraopinodachavecomoentrada
    pinMode(chavePin8, INPUT_PULLUP); //Configuraopinodachavecomoentrada
    pinMode(chavePin9, INPUT_PULLUP); //Configuraopinodachavecomoentrada

    pinMode(ledPin2, OUTPUT); //ConfiguraopinodoLEDcomosada
    pinMode(ledPin3, OUTPUT); //ConfiguraopinodoLEDcomosada
    pinMode(ledPin4, OUTPUT); //ConfiguraopinodoLEDcomosada
    pinMode(ledPin5, OUTPUT); //ConfiguraopinodoLEDcomosada

    //Rotina testes inicial dos Leds
    for(int j = 0; j < 5; j++)

    velocidadeled = 15;
    for(int contador = 0; contador < 4; contador++)
        digitalWrite(ledPin2, HIGH);
        delay(velocidadeled);
        digitalWrite(ledPin2, LOW);
        delay(velocidadeled);
        digitalWrite(ledPin3, HIGH);
        delay(velocidadeled);
        digitalWrite(ledPin3, LOW);
        delay(velocidadeled);

```



```

digitalWrite(ledPin4, HIGH);
delay(velocidadeled);
digitalWrite(ledPin4, LOW);
delay(velocidadeled);

```

```

digitalWrite(ledPin5, HIGH);
delay(velocidadeled);
digitalWrite(ledPin5, LOW);
delay(velocidadeled);

```

```
//FimdaRotina
```

```

voidloop()
  if (digitalRead(chavePin6) == LOW)
    digitalWrite(ledPin2, HIGH); //LigaoLED

    Serial.println("Chave 1 ON !!");

  else

    digitalWrite(ledPin2, LOW); //DesligaoLED

    Serial.println("Chave 1 OFF !!");

  if (digitalRead(chavePin7) == LOW)
    digitalWrite(ledPin3, HIGH); //LigaoLED

    Serial.println("Chave 2 ON !!");

  else

    digitalWrite(ledPin3, LOW); //DesligaoLED

    Serial.println("Chave 2 OFF !!");

  if (digitalRead(chavePin8) == LOW)

```

```
digitalWrite(ledPin4, HIGH); //LigaoLED

Serial.println("Chave 3 ON !!");

else

digitalWrite(ledPin4, LOW); //DesligaoLED

Serial.println("Chave 3 OFF !!");

if (digitalRead(chavePin9) == LOW)
digitalWrite(ledPin5, HIGH); //LigaoLED

Serial.print("Chave 4 ON !!");

else
digitalWrite(ledPin5, LOW); //DesligaoLED

Serial.println("Chave 4 OFF !!");

delay(500);

Lesser General Public License (LGPL)
```

APÊNDICE C – Algoritmo pratica de laboratório 03º.

Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório

3.

```

const int chavePin6 = 6; //Pinodachave

const int chavePin7 = 7; //Pinodachave

const int chavePin8 = 8; //Pinodachave

const int chavePin9 = 9; //Pinodachave

const int ledPin2 = 2; //PinodoLED

const int ledPin3 = 3; //PinodoLED

const int ledPin4 = 4; //PinodoLED

const int ledPin5 = 5; //PinodoLED

// Auxiliar para teste inicial de leds

int velocidadeled = 115;

void setup()
Serial.begin(9600);
// Definição dos pinos I/O
pinMode(chavePin6, INPUT_PULLUP); //Configuraopinodachavecomoentrada
pinMode(chavePin7, INPUT_PULLUP); //Configuraopinodachavecomoentrada
pinMode(chavePin8, INPUT_PULLUP); //Configuraopinodachavecomoentrada
pinMode(chavePin9, INPUT_PULLUP); //Configuraopinodachave

pinMode(ledPin2, OUTPUT); //ConfiguraopinodoLEDcomosada
pinMode(ledPin3, OUTPUT); //ConfiguraopinodoLEDcomosada
pinMode(ledPin4, OUTPUT); //ConfiguraopinodoLEDcomosada
pinMode(ledPin5, OUTPUT); //ConfiguraopinodoLEDcomosada

//Rotina testes inicial dos Leds
for(int j =0; j<5; j++)
velocidadeled = 15;

```

```

    for (int contador = 0; contador<4; contador++)

        digitalWrite(ledPin2, HIGH);
    delay(velocidadeled);
    digitalWrite(ledPin2, LOW);
    delay(velocidadeled);

        digitalWrite(ledPin3, HIGH);
    delay(velocidadeled);
    digitalWrite(ledPin3, LOW);
    delay(velocidadeled);

        digitalWrite(ledPin4, HIGH);
    delay(velocidadeled);
    digitalWrite(ledPin4, LOW);
    delay(velocidadeled);

        digitalWrite(ledPin5, HIGH);
    delay(velocidadeled);
    digitalWrite(ledPin5, LOW);
    delay(velocidadeled);

//FimdaRotina

voidloop()
    if (digitalRead(chavePin6) == LOW)
        digitalWrite(ledPin2, HIGH); //LigaoLED
    Serial.println("Chave1ON!!");
    elsedigitalWrite(ledPin2, LOW); //DesligaoLEDSerial.println("Chave1OFF!!");

    if (digitalRead(chavePin7) == LOW)
        digitalWrite(ledPin3, HIGH); //LigaoLED

    Serial.println("Chave 2 ON !!");

    else

```

```
digitalWrite(ledPin3, LOW); //DesligaoLED
```

```
Serial.println("Chave 2 OFF !!");
```

```
if (digitalRead(chavePin8) == LOW)  
digitalWrite(ledPin4, HIGH); //LigaoLED
```

```
Serial.println("Chave 3 ON !!");
```

```
else
```

```
digitalWrite(ledPin4, LOW); //DesligaoLED
```

```
Serial.println("Chave 3 OFF !!");
```

```
if (digitalRead(chavePin9) == LOW)  
digitalWrite(ledPin5, HIGH); //LigaoLED
```

```
Serial.print("Chave 4 ON !!");
```

```
Serial.println();
```

```
Serial.println();
```

```
else
```

```
digitalWrite(ledPin5, LOW); //DesligaoLED
```

```
Serial.println("Chave 4 OFF !!");
```

```
Serial.println();
```

```
delay(500);
```

Lesser General Public License (LGPL)

APÊNDICE D – Algoritmo pratica de laboratório 04º.

Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório

4.

```

include <LiquidCrystal.h>

define pinVRx A0
define pinVRy A1

const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7; // Pinos para ligar o display
LiquidCrystal lcd(rs, en, d4, d5, d6, d7); // Define os pinos que serão usados para ligar o display

void setup()
Serial.begin(9600);
lcd.begin(16, 2); // Define o número de colunas e linhas do display

pinMode(pinVRx, INPUT);
pinMode(pinVRy, INPUT);
pinMode(pinSW, INPUT_PULLUP);

void loop()
delay(1000);

int valorVRx = analogRead(pinVRx);
int valorVRy = analogRead(pinVRy);
int statusSW = digitalRead(pinSW);

Serial.print("Valor VRx: ");
Serial.println(map(valorVRx,0,1023,0,180));

Serial.print("Valor VRy: ");
Serial.println(map(valorVRy,0,1023,0,180));

if ((valorVRx >= 450)(valorVRx <= 550))

lcd.setCursor(0, 1);
lcd.print();

lcd.setCursor(0, 0);

```



```
lcd.print("Dir X Dir Y");
lcd.setCursor(0, 1);
lcd.print("NEUTRO");

if ((valorVRx >= 551) (valorVRx <= 1080))

lcd.setCursor(0, 1);
lcd.print();

lcd.setCursor(0, 0);
lcd.print("Dir X Dir Y");
lcd.setCursor(0, 1);
lcd.print("DIREITA");

if ((valorVRx >= 0) (valorVRx <= 449))

    lcd.setCursor(0, 1);
    lcd.print();

    lcd.setCursor(0, 0);
    lcd.print("Dir X Dir Y");
    lcd.setCursor(0, 1);
    lcd.print("ESQUERDA");

    if ((valorVRy >= 450) (valorVRy <= 550))

        lcd.setCursor(8, 1);
        lcd.print();

        lcd.setCursor(0, 0);
        lcd.print("Dir X Dir Y");
        lcd.setCursor(9, 1);
        lcd.print("NEUTRO");

        if ((valorVRy >= 551) (valorVRy <= 1080))

            lcd.setCursor(8, 1);
            lcd.print();
```

```
        lcd.setCursor(0, 0);  
lcd.print("Dir X Dir Y");  
lcd.setCursor(9, 1);  
lcd.print("BAIXO");  
  
if ((valorVRy >= 0) (valorVRy <= 449))
```

```
lcd.setCursor(8, 1);  
lcd.print();
```

```
        lcd.setCursor(0, 0);  
lcd.print("Dir X Dir Y");  
lcd.setCursor(9, 1);  
lcd.print("CIMA");
```

Lesser General Public License (LGPL)

APÊNDICE E – Algoritmo pratica de laboratório 05°.

Projeto ilustrando o desempenho do kit de interfaces no contexto do laboratório

5.

```

    include <LiquidCrystal.h>
include <Wire.h>
include <TimeLib.h>
include <DS1307RTC.h>

    const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7; // Pinos para ligar o display
LiquidCrystal lcd(rs, en, d4, d5, d6, d7); // Define os pinos que serão usados para ligar o display

    void setup()
lcd.begin(16, 2); // Define o número de colunas e linhas do display
Serial.begin(9600);

    #ifndef ESP8266
while (!Serial); // Wait for serial port to connect. Needed for native USB
endif

setSyncProvider(RTC.get); // Sincronize o TimeLib com o DS1307RTC

// Defina a hora e a data iniciais (exemplo: 23:40:00 em 19/09/2023)
tmElements_t startTime;
    startTime.Second = 0;
    startTime.Minute = 0;
    startTime.Hour = 0;
    startTime.Day = 0;
    startTime.Month = 0;
    startTime.Year = 70;
    RTC.write(startTime);

    if (timeStatus() != timeSet)
Serial.println("Unable to sync with the RTC");
    else
Serial.println("RTC has set the system time");

    void loop()

```

```
time_t noww = now();

    lcd.clear(); // Limpa o display
lcd.setCursor(0, 0);
lcd.print("Data: ");
lcd.print(day(noww));
lcd.print('/');
lcd.print(month(noww));
lcd.print('/');
lcd.print(year(noww));

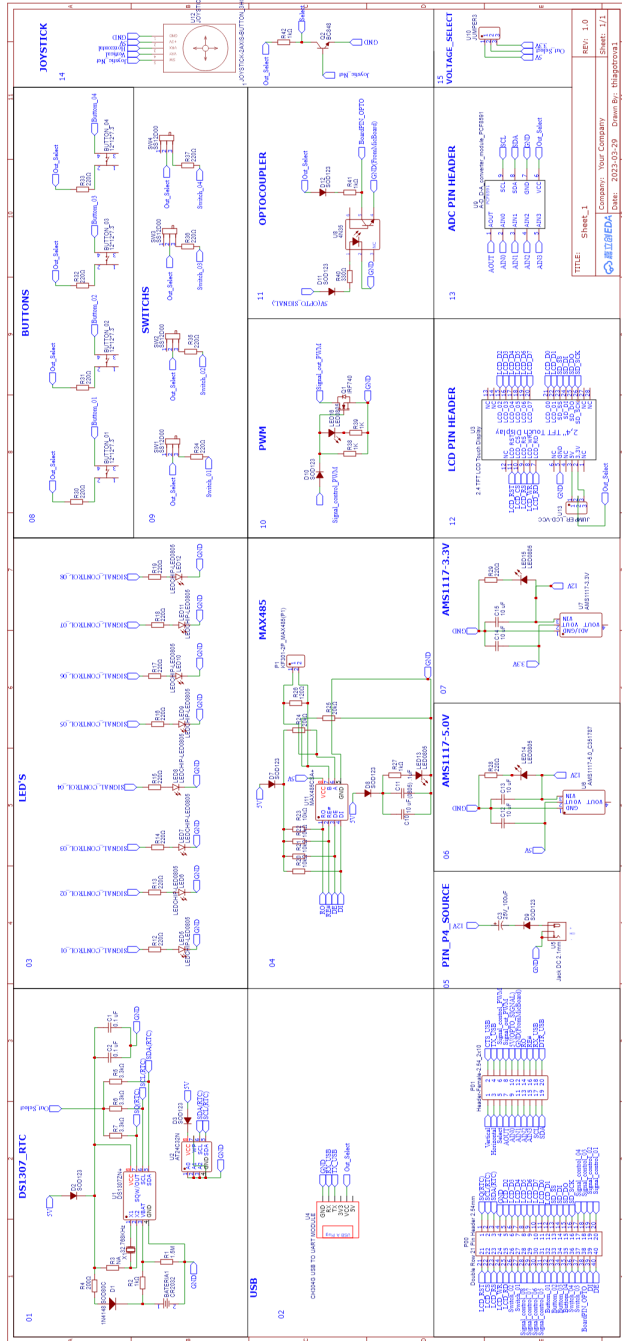
lcd.setCursor(0, 1);
lcd.print("Hora: ");
lcd.print(hour(noww));
lcd.print(':');
lcd.print(minute(noww));
lcd.print(':');
lcd.print(second(noww));

Serial.println(analogRead(0)); // Exibe a leitura do pino analógico A0 no monitor serial
delay(1000); // Aguarde 1 segundo
```

Lesser General Public License (LGPL)

APÊNDICE F – Esquemático elétrico do projeto desenvolvido.

Figura 17 – Esquemático



Fonte: Autoria própria (2023).