

**UNIVERSIDADE FEDERAL DE ITAJUBÁ – UNIFEI**

**ELTD13A – LABORATÓRIO DE MICROCONTROLADORES E  
MICROPROCESSADORES – TURMA P2**

**CONTROLADOR DE FAN POR MODULAÇÃO DE LARGURA DE PULSO (PWM)**

Discente:  
Pedro Andrade Gomes – 2022006926

# CONTROLADOR DE FAN POR MODULAÇÃO DE LARGURA DE PULSO (PWM)

## 1. INTRODUÇÃO

Este projeto consiste no desenvolvimento de um controlador de fan por modulação de largura de pulso (PWM). O equipamento apresenta um estado inicial que indica na linha superior de um display LCD 2x16 as opções de ações e operações disponíveis no equipamento. Na linha de cima, a velocidade atual do fan indicada em RPM (rotações por minuto) pode ser observada no canto esquerdo, já no canto direito é possível visualizar a porcentagem atual do PWM. O usuário poderá interagir com o equipamento através de 6 teclas de um teclado, sendo estes: quatro teclas de navegação (esquerda, direita, cima e baixo), uma tecla de confirmar e uma tecla de salvar.

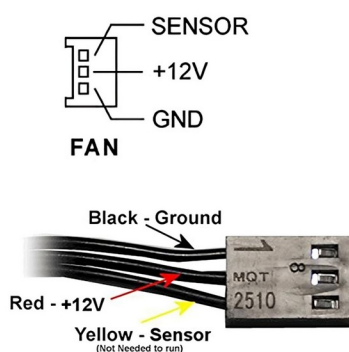
Dentre as operações e funcionalidades disponíveis no equipamento, têm-se:

- Tela inicial para visualizar o RPM atual, PWM atual e RPM máximo do fan conectado;
- Definição da velocidade-alvo em RPM;
- Estipulação do passo de mudança do PWM;
- Estipulação do intervalo de processamento do módulo de controle;
- Seleção de velocidade pré-definida em PWM: 25%, 50%, 75% ou 100% do PWM;
- Alternar o controle da velocidade do fan entre PWM/potenciômetro;

A placa de desenvolvimento possui 8 LEDs, os quais serão utilizados para indicar a porcentagem do PWM aplicado a uma escala de 12%. Isso significa que, se dois LEDs estiverem acessos, a porcentagem do PWM está entre:  $13\% < \text{PWM} \leq 24\%$ .

### 1.1 MÉTODO DE MEDIÇÃO DE VELOCIDADE DO FAN

Para medir a velocidade do fan será utilizado um pino do microcontrolador para ler um sinal do sensor de revolução do fan, geralmente integrado com o equipamento. Considere a imagem abaixo com as conexões dos pinos do fan.



*Figura 1: Conexão dos pinos do fan; Fonte: [6]*

A cada revolução do fan, um sinal analógico de aproximadamente 0,2v é emitido. Esse valor será amplificado por um circuito amplificador, constituído de um AmpOp MCP6002 com um ganho

de 31, lido e, em conjunto com um temporizador interno do microcontrolador, contado em um período de um a cinco segundo. A partir disso cálculos serão realizados para estimar a quantidade de revoluções por minuto (RPM) do referente fan.

## 2. JUSTIFICATIVA

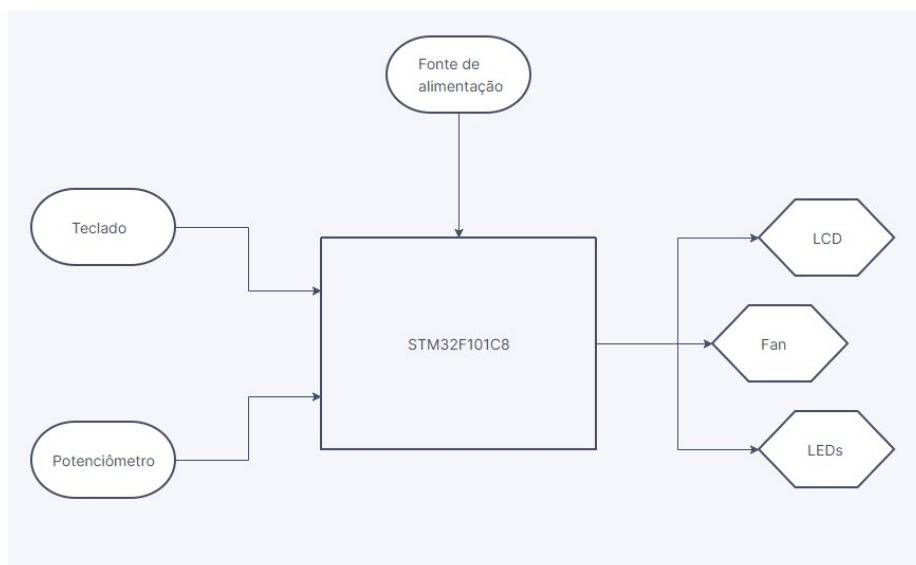
A justificativa para o desenvolvimento deste projeto se dá a partir da premissa que: um sistema dinâmico de controle de resfriamento abre a possibilidade de uma integração com um módulo de sensor de temperatura, fazendo com que seja possível ajustar a rotação dos fans de um sistema para que se adéquem a temperatura lida. Isso possibilita que o sistema seja dinâmico, evita desgaste desnecessário nos eixos de rotação dos fans do sistema, além de promover eficiência energética e tornar o sistema mais silencioso em termos do ruído gerado pela rotação das hélices.

Além disso, se for observado que o sistema sempre está com uma alta velocidade de rotação, constantemente, isso pode indicar que o sistema de refrigeração não está sendo suficiente. Desta forma, o equipamento também pode ajudar a prever e/ou detectar problemas no sistema de refrigeração, da forma como foi dito.

É importante ressaltar que esta é uma justificativa para o **desenvolvimento do módulo**, e não de sua aplicação individual. Este projeto se limita ao desenvolvimento de todo o sistema de controle e configuração do mesmo, e não da sua aplicação efetiva e individual no mundo real.

## 3. COMPONENTES FÍSICOS DO EQUIPAMENTO

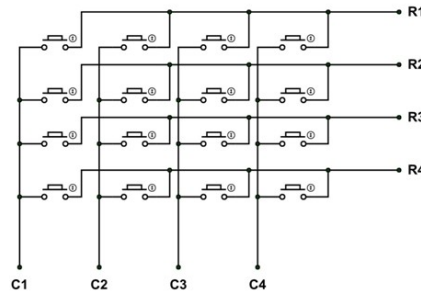
O modulador possui os seguintes componentes: Microcontrolador, fonte de alimentação, teclado com pelo menos 6 botões, display LCD 2x16 e um fan. O diagrama abaixo ilustra o esquema de montagem do sistema:



*Figura 2: Diagrama de componentes do equipamento; Fonte: Autor*

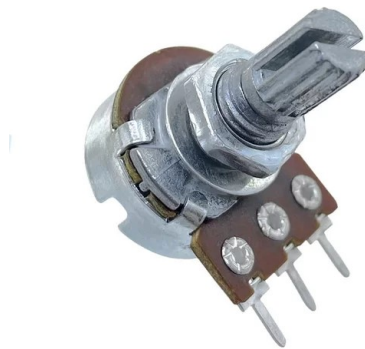


O teclado, por sua vez, é o periférico responsável pela interação do usuário, com o LCD e, por sua vez, o controle geral do equipamento. O teclado possui um esquema elétrico genérico associado, o qual está disponível na imagem abaixo. O seu tipo é *pull-down*.



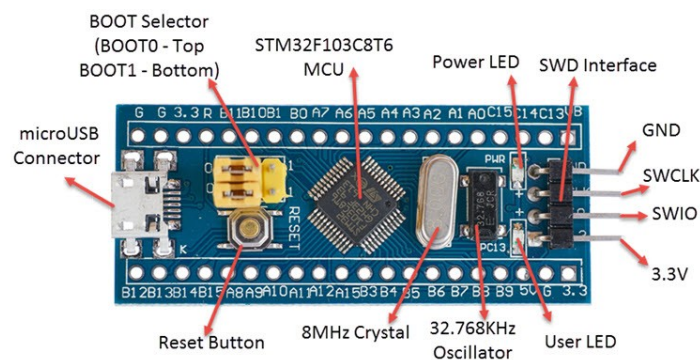
*Figura 6: Esquema elétrico genérico associado ao teclado;  
Fonte: [3]*

O potenciômetro é um componente eletrônico de natureza analógica com a característica da variação da sua resistência elétrica em função da angulação de sua haste interna. Segue abaixo uma imagem de um potenciômetro genérico:



*Figura 7: Potenciômetro genérico;  
Fonte: [8]*

Os LEDs são componentes associados à placa de desenvolvimento. A fonte, por sua vez, é uma fonte genérica associada a alimentação da placa de desenvolvimento. O microcontrolador é o STM32F103, associado à placa comercial denominada *blue-pill*, que pode ser observada abaixo.



*Figura 8: Placa blue-pill; Fonte: [4]*

O fan por sua vez é um fan genérico do tipo utilizado em sistemas de refrigeração de computadores *desktop*. A obtenção deste equipamento é de íntegra responsabilidade do autor deste projeto, uma vez que a instituição de ensino no qual o mesmo compartilha vínculo **não** dispõe deste tipo de componente eletrônico. Segue a imagem ilustrativa abaixo de um fan genérico muito semelhante ao que será, de fato, utilizado:



Figura 9: Fan genérico; Fonte: [5]

O circuito de acionamento é composto por um optoacoplador 4N33, um estágio de acionamento composto por um conversor buck e um estágio de captura de sinal composto por um MCP6002, como ilustra o esquemático abaixo:

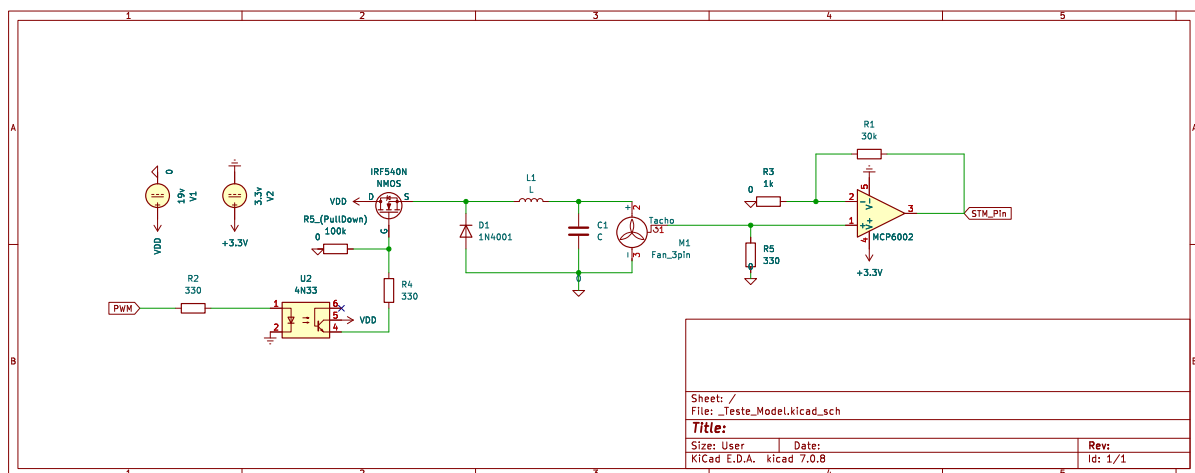


Figura 10: Driver de acionamento do fan e circuito de tratamento do sinal; Fonte: Autor

É relevante mencionar que a referência da entrada inversora diz respeito ao GND da BluePill, e o pull-down da entrada não inversora diz respeito ao GND da fonte de 19v, tornando assim a alta impedância do AmpOp um elemento isolante entre a saída do circuito de potência e a entrada do microcontrolador. Um segundo optoacoplador seria utilizado para intermediar a saída do amplificador, porém, por limitações de disponibilidade, isso foi inviabilizado.

#### 4. FLUXOGRAMA DO PROGRAMA DESENVOLVIDO

O programa embarcado associado ao equipamento será desenvolvido em linguagem Assembly pelo discente autor deste documento. O programa, por sua vez, seguirá o modelo que é representado abaixo por meio de um diagrama:

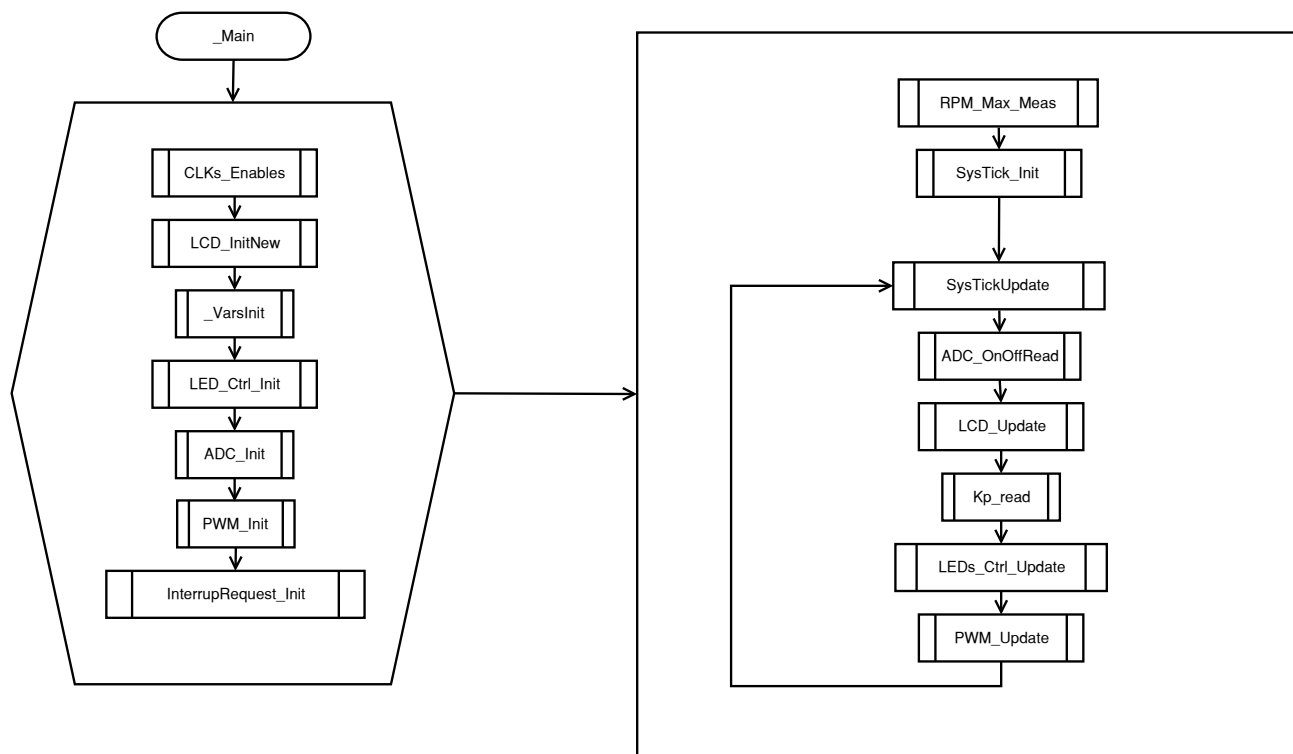


Figura 11: Fluxograma do programa desenvolvido; Fonte: Autor;

O bloco da esquerda é composto por um arquivo `_Main` que chama uma sequência de sub-rotinas de inicialização, contidas dentro do bloco de inicialização.

O bloco da direita, por sua vez, é composto por mais duas sub-rotinas de inicialização e 6 sub-rotinas de ciclo de programa, isto é, sub-rotinas estas que serão o loop de programa.

A sub-rotina `RPM_Max_Meas` é responsável por mensurar o RPM máximo do fan conectado. Ela faz isso setando o PWM em 100% durante 4 segundos assim que o programa é inicializado. A sub-rotina de inicialização do SysTick Timer é acompanhada após a execução de `RPM_Max_Meas` para evitar a interrupção gerada pelo SysTick Timer.

#### 4.11 – CICLO DE PROGRAMA: SysTickUpdate



Figura 12: Sub-rotina  
SysTickUpdate; Fonte: Autor;

Esta sub-rotina é responsável por verificar se houve requisição de alteração de tempo de processamento, se houver, atualiza o registro `stk_load`, se não houver, pula a execução.

#### 4.12 – CICLO DE PROGRAMA: ADC\_OnOffRead

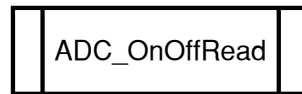


Figura 13: Sub-rotina  
ADC\_OnOffRead; Fonte: Autor;

Esta sub-rotina é responsável por verificar se o modo ANALOG foi selecionado, se sim, ele liga o ADC em modo contínuo e realiza uma nova medição de forma independente do tempo de processamento vigente. O controle do PWM neste modo é independente das variáveis de controle de tempo. Se o modo ANALOG não estiver setado e se o ADC estiver ligado, ele é, nestas condições, desligado.



#### 4.13 – CICLO DE PROGRAMA: LCD\_Update

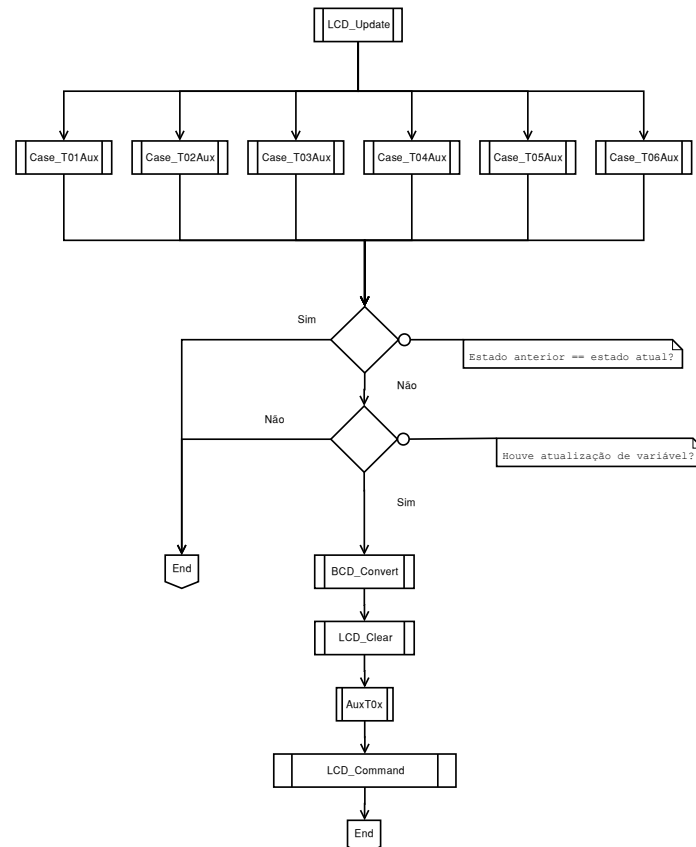


Figura 14: Fluxograma de LCD\_Update; Fonte: Autor;

Esta sub-rotina é responsável por atualizar o display LCD. Se o estado do LCD for igual ao anterior, o display não é atualizado. Se o estado do LCD for diferente do anterior e houver atualização em variáveis, o LCD é atualizado.

#### 4.14 – CICLO DE PROGRAMA: Kp\_read

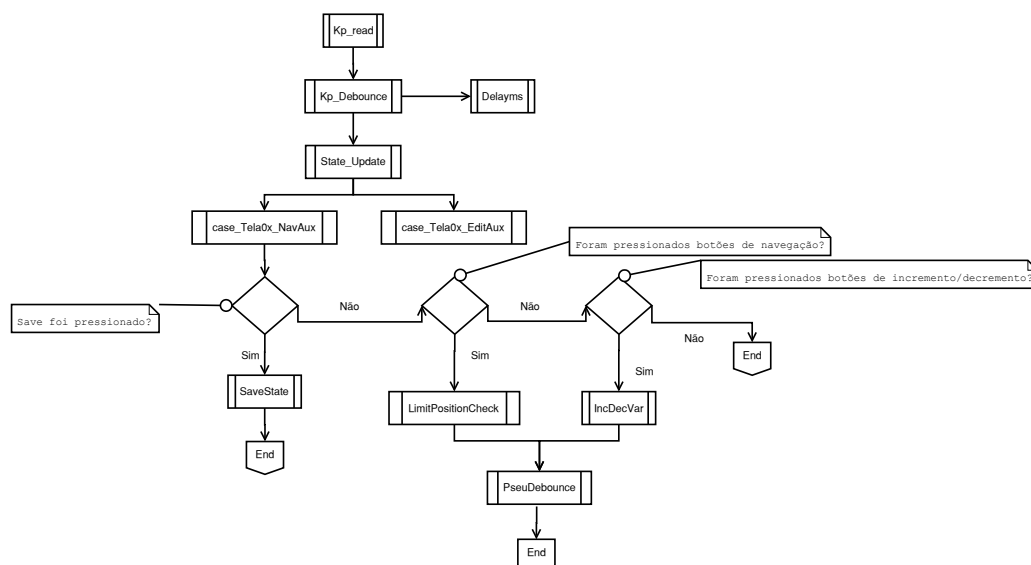


Figura 15: Fluxograma de Kp\_read; Fonte: Autor;

Esta sub-rotina é responsável por verificar se uma tecla foi pressionada. Se foi, verifica qual, e com base na tecla e no estado atual, toma as devidas providências.

#### 4.15 – CICLO DE PROGRAMA: LEDs\_Ctrl\_Update

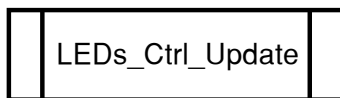


Figura 16: Sub-rotina  
LEDs\_Ctrl\_Update; Fonte: Autor;

Esta sub-rotina é responsável por verificar a porcentagem do PWM e realizar, assim, um controle de LEDs com base nesta informação.

#### 4.16 – CICLO DE PROGRAMA: PWM\_Update



Figura 17: Sub-rotina  
PWM\_Update; Fonte: Autor;

Esta sub-rotina é responsável por fazer o controle do valor do PWM com base no RPM desejado pelo usuário. Só executa após o intervalo de tempo de processamento definido pelo usuário.

#### 4.17 – CICLO DE PROGRAMA: DETALHES DE PROGRAMA

O controle de tempo de ciclo de programa é controlado através de uma interrupção gerada pelo SysTick Timer. Este tempo pode ser controlado pelo usuário em uma faixa que varia de 1 a 5 segundos.

Para realizar o controle do RPM e sua estipulação, uma interrupção do tipo *Rising edge* foi configurada no pino PB7 da placa RedPill e conectado ao circuito de tratamento de sinal, apresentado na [10](#). A cada sinal de revolução completa emitido pelo fan uma interrupção é gerada, e, na sub-rotina de interrupção, é incrementado a variável de contagem de revolução. Após o tempo de processamento definido o RPM é calculado em função deste tempo e do número de revoluções contabilizadas. Após o cálculo, essa variável é resetada.

Ademais, o controle do RPM por incremento/decremento de passo de PWM possui uma faixa de tolerância de 100 RPM para mais ou para menos. Isso significa que, se o RPM desejado for 1200, e o atual estiver em 1300, o controle assumirá que o ponto de controle foi atingido.

#### 4.20 – APRESENTAÇÃO DOS ARQUIVOS DE PROJETO:

- **\_STM32F103C8\_Address:** Arquivo que contém a definição de todas as constantes utilizadas no projeto;
- **\_Vars:** Arquivo que contém a definição de todas as variáveis utilizadas no projeto;
- **ADC\_Lib:** Arquivo que contém as sub-rotinas referentes ao controle do conversor analógico/digital;
- **BCD\_Converter:** Arquivo que contém as sub-rotinas de conversão BCD e conversão BCD/ASCII\_hexadecimal;
- **CLKs\_Enables:** Arquivo que contém a sub-rotina de controle geral do RCC;
- **Delayms:** Arquivo que contém a sub-rotina de controle de delay para controle de tempo geral utilizado em várias seções do programa;
- **Event\_Update:** Arquivo que contém as sub-rotinas referentes ao controle de fluxo de programa, como Kp\_read, Kp\_Debounce e State\_Update;
- **Interrupt\_Request\_Lib:** Arquivo que contém as sub-rotinas de controle de interrupção do pino PB7;
- **LCD\_Lib:** Arquivo que contém todas as sub-rotinas referentes à inicialização e controle padrão do LCD;
- **LCD\_Invert\_Lib:** Arquivo que contém todas as sub-rotinas de controle, inicialização, definição e manipulação da inversão dos eixos X e Y gráficos do LCD;
- **LCD\_Updat\_Aux:** Arquivo que contém algumas sub-rotinas de controle de envio de dados das respectivas telas pré definidas do programa para o LCD;
- **LCDUpdate:** Arquivo que contém todas as sub-rotinas de controle pré envio de informações ao LCD. Chama LCD\_Updat\_Aux quando precisa atualizar as informações no LCD;
- **LED\_Ctrl:** Arquivo que contém todas as sub-rotinas de controle dos LEDs, que agem em função da porcentagem do PWM;
- **PWM\_Lib:** Arquivo que contém todas as sub-rotinas de controle do sistema do PWM, incluindo RPM\_Max\_Meas, RPM\_Max\_Meas e PWM\_Init;
- **SysTick\_Lib:** Arquivo que contém todas as sub-rotinas de controle só SysTick Timer;
- **\_Main:** Arquivo principal do programa, reúne e organiza as principais sub-rotinas necessárias para que o fluxo do programa funcione de forma eficiente;

## 5.1 - DESCRITIVO DE IMPLEMENTAÇÃO DO PROJETO

Os passos que procedem à implementação real do projeto se dão por: cálculos de valores e obtenção de componentes, implementação física do esquemático elétrico desenvolvido, testes de depuração de partes-chave do código e correções posteriores. Todos estes passos serão comentados a frente.

O cálculo e de valores e obtenção de componentes resultou na seguinte lista de componentes eletrônicos:

- Optoacoplador 4N33;
- MOSFET IRF540N;
- Diodo schottky 1N4001;
- Indutor de 1mH;
- Capacitor de 470uF de 100v;
- Resistores de valores: 100k, 30k, 1k e 330 ohms;
- Fonte de 19v;
- AmpOp MCP6002;

É importante dizer que alguns destes componentes não remetem aos planejados inicialmente, mas sim à disponibilidade deles acerca do autor do projeto.

Após a obtenção dos componentes, a implementação do mesmo via protoboard e jumpers foi realizada e verificada. Análises em laboratório utilizando multímetros e osciloscópio foram realizadas para validar os diferentes estágios do circuito de controle e de tratamento de sinal. Todas as devidas correções e ajustes nos valores, principalmente, dos resistores foram realizadas após esta etapa, resultando nos valores aqui apresentados.

Etapas de teste para verificar a correta detecção das interrupções geradas no pino PB7 também foram realizadas e validadas. Alguns ajustes no cálculo do RPM via software também foram realizados em função de ruídos externos que não puderam ser eliminados pelo hardware, tendo a versão final do projeto um bom desempenho nesse quesito.

## 5.2 – CONCLUSÃO GERAL

De modo geral, o projeto desenvolvido atingiu as metas inicialmente propostas. Com relação ao desempenho e qualidade efetiva do projeto desenvolvido, é possível afirmar que, como esperado, implicitamente, limitações foram encontradas. A precisão em termos de medição de RPM é bastante limitada devido a geração de ruídos provenientes do circuito de potência. A limitação em termos de componentes e equipamentos precisos e de qualidade resultaram em uma aplicação aceitável, em termos de aprendizado acadêmico, porém, não muito aplicável em termos de aplicação de mercado e uso comercial.

Em suma, o projeto desenvolvido possui limitações de precisão em função das limitações do autor em ter acesso a componentes e equipamentos precisos e de qualidade, porém, apresentou um resultado satisfatório e aplicável em meio acadêmico.

## 6. REFERÊNCIAS

- [1] - Disponível em: <[https://www.usinainfo.com.br/1019423-thickbox\\_default/display-lcd-16x2-com-fundo-azul.jpg](https://www.usinainfo.com.br/1019423-thickbox_default/display-lcd-16x2-com-fundo-azul.jpg)>. Acesso em: 5 maio. 2024.
- [2] - Disponível em: <<https://www.electronicwings.com/storage/PlatformSection/TopicContent/197/description/LCD16x2%20Pins.png>>. Acesso em: 5 maio. 2024.
- [3] - Disponível em: <<https://components101.com/sites/default/files/inline-images/4x4-Keypad-Internal-Structure.png>>. Acesso em: 5 maio. 2024.
- [4] - Disponível em:  
<[https://www.teachmemicro.com/wp-content/uploads/2020/10/img\\_5f769959db0e5.jpeg](https://www.teachmemicro.com/wp-content/uploads/2020/10/img_5f769959db0e5.jpeg)>. Acesso em: 5 maio. 2024.
- [5] - Disponível em: <<https://cdn.awsli.com.br/2500x2500/1919/1919257/produto/217382364/Foto-Digital-2.jpg>>. Acesso em: 5 maio. 2024.
- [6] - ((Disponível em: <<https://m.media-amazon.com/images/I/41iPV+JIYCL.jpg>>. Acesso em: 21 maio. 2024.))
- [7] - Disponível em:  
<<https://cdn.awsli.com.br/600x700/468/468162/produto/19414409/106e8da5c7.jpg>>. Acesso em: 23 maio. 2024.
- [8] - Disponível em: <[https://global.cdn.magazord.com.br/proesi/img/2024/01/produto/5960/potenciometro-l15-para-placa.jpeg?ims=fit-in/600x600/filters:fill\(white\)](https://global.cdn.magazord.com.br/proesi/img/2024/01/produto/5960/potenciometro-l15-para-placa.jpeg?ims=fit-in/600x600/filters:fill(white))>. Acesso em: 23 maio. 2024.

*Itajubá – Junho de 2024*

*ELTD13 – Laboratório de microcontroladores e microprocessadores*