

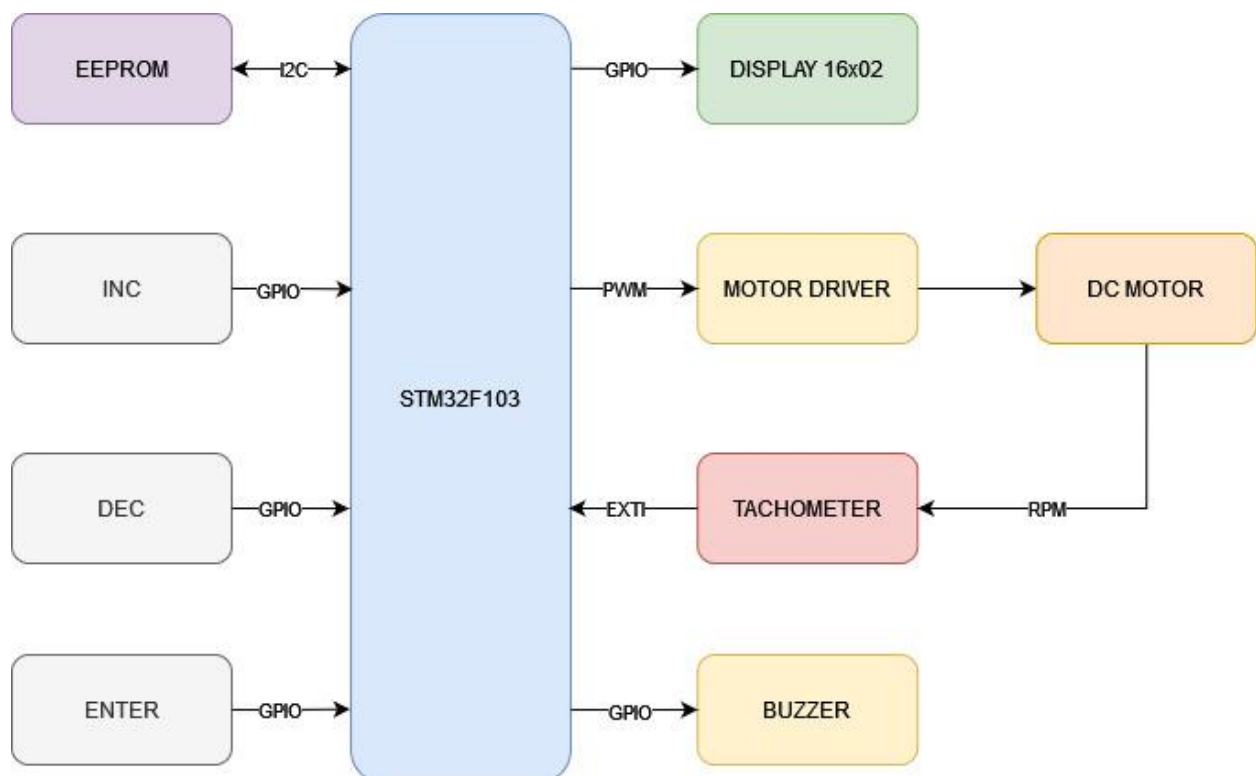
Projeto 01 – Controlador de Motor DC

O objetivo do projeto é criar um sistema que controle a velocidade de um motor DC, permitindo a configuração do seu funcionamento por meio de uma Interface Homem Máquina (I.H.M).

Overview

O diagrama em blocos abaixo, apresenta uma ideia do hardware firmware do projeto, todos os blocos e periféricos apresentados são apenas uma sugestão, sendo assim o desenvolvedor fica livre para utilizar outros métodos para atingir os requisitos e funcionamento apresentados ao longo deste documento.

Figura 1. Diagrama em blocos do projeto.

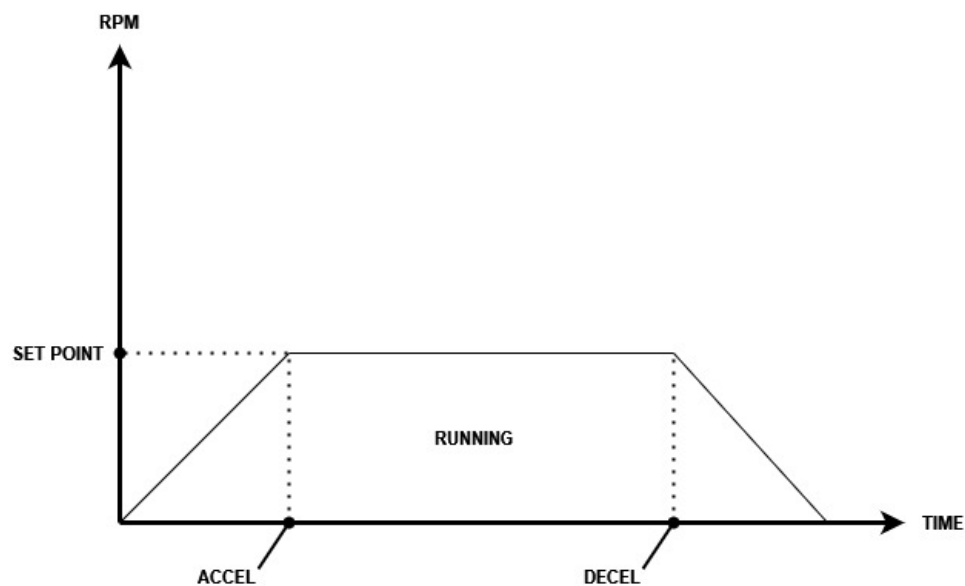


Funcionamento

Ao iniciar o procedimento, o motor deve acelerar gradualmente, atingindo a rotação por minuto (**RPM**) configurada no tempo predeterminado.

Uma vez alcançado o RPM definido como ponto de referência, o motor deve manter essa rotação constante durante o período operacional pré configurado. Após o término desse período, a desaceleração deve ser executada, levando o motor de volta de sua RPM atual a zero dentro do tempo previamente definido.

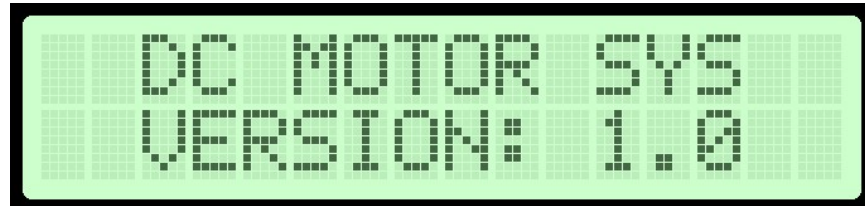
Figura 2. Controle de velocidade para motor DC.



Interface homem máquina

Ao ligar o equipamento, uma tela deverá ser apresentada com o nome do produto e versão de firmware, essa tela deve aparecer por **3 segundos**.

Figura 3 - Nome do produto e versão de firmware.

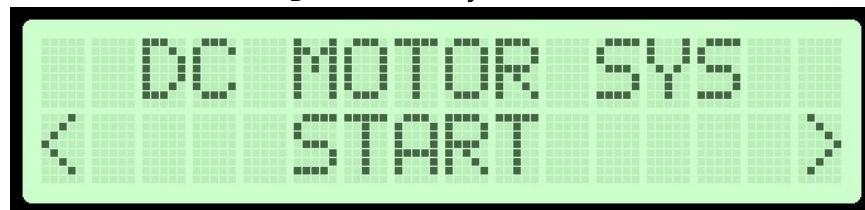


Após a apresentação do nome do produto, vamos para a tela principal onde o usuário através das teclas de navegação poderá navegar entre 2 opções : **START** e **MENU**.

START

Ao selecionar a função de START, o usuário irá dar início ao processo de controle de velocidade, caso o usuário não tenha selecionado previamente um perfil de configuração, o funcionamento deverá seguir a configuração padrão.

Figura 4 - Função START.



A IHM deve informar ao usuário o RPM(Rotações por minuto) do motor em tempo real, bem como, o tempo de execução de cada etapa do processo (aceleração, funcionamento e desaceleração). As figuras a seguir demonstram todas as etapas de funcionamento do sistema.

Figura 5 - Tela exemplo da etapa de aceleração.

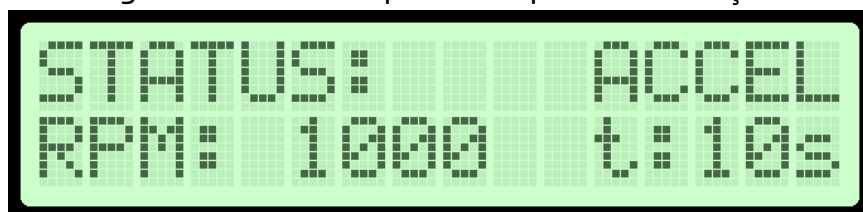


Figura 6 - Tela exemplo da etapa de funcionamento .

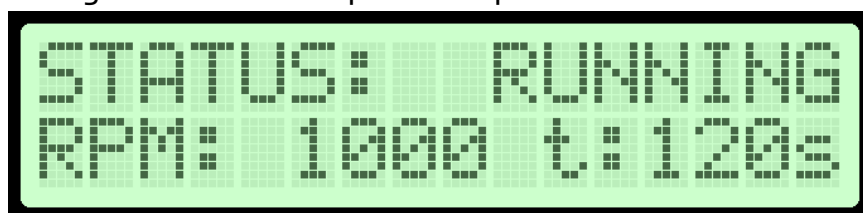
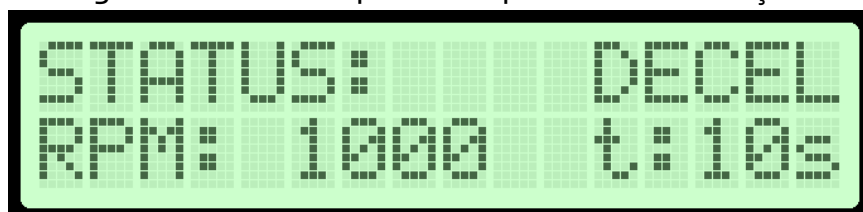


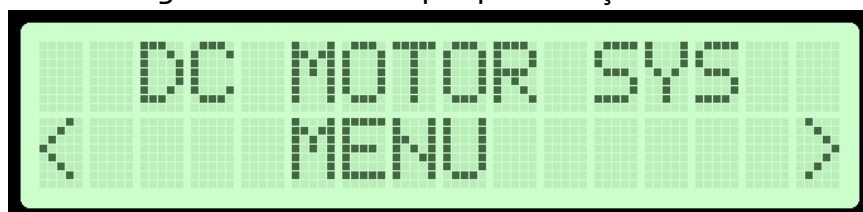
Figura 7 - Tela exemplo da etapa de desaceleração.



MENU

Ao seleccionar a tela de MENU, o usuário irá configurar o perfil de funcionalidade do sistema onde o mesmo irá poder alterar os seguintes parâmetros: **aceleração**, **funcionamento** e **desaceleração**.

Figura 8 - Tela exemplo para função menu.



Todas as alterações feitas devem ser salvas na EEPROM externa, e ao ligar o dispositivo devem ser carregadas ao acionar o **START**.

Os parâmetros mencionados devem respeitar os seus limites de configuração definidos abaixo:

- **Tempo de aceleração.**
 - Máximo: 80s, Mínimo 5s.
- **Tempo de funcionamento.**
 - Máximo: 180s, Mínimo 5s.
- **Tempo de desaceleração.**
 - Máximo: 80s, Mínimo 5s.

Quando é um equipamento e nenhum parâmetro foi salvo pelo usuário, o sistema deverá seguir a configuração padrão.

Configuração padrão :

- Tempo de aceleração = 10s.
- Tempo de funcionamento = 120s.
- Tempo de desaceleração = 10s.

Fluxo de configuração

Ao entrar no menu configuração serão apresentados os parâmetros a serem editados, e através dos botões **INC** e **DEC** o usuário poderá alterar cada um deles entre seus valores máximos e mínimos, e ao pressionar o botão **ENTER**, será levado ao próximo parâmetro e ao finalizar a configuração o usuário será levado à tela principal.

Sempre que um botão for acionado, o **buzzer** deve ser acionado para confirmar ao usuário que o mesmo foi pressionado e entendido pelo sistema.

Figura - Ajuste do tempo de aceleração.

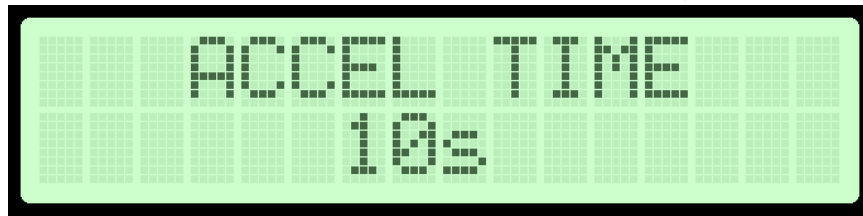


Figura 5 - Ajuste do tempo de funcionamento.

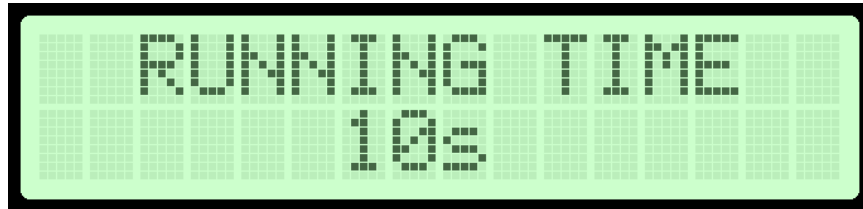
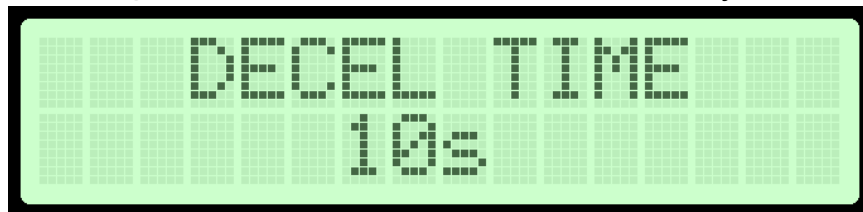


Figura 5 - Ajuste do tempo de desaceleração.



Publicação

Todo o projeto deve ser disponibilizado em um repositório público do github, e todos os commits feitos devem ser feitos em branches e depois incluídos no master/main.

Materiais Necessários

- 1x STM32F103C8T6
- 1x Mosfet canal N (IRF540)
- 1x Display 16x02
- 3x Chaves Táctil
- 7x Resistor 10K 1/4W
- 4x Capacitor de Cerâmica 100nF
- 1x Sensor de velocidade
- 1x Buzzer
- 1x Transistor NPN (BC548)
- 1x EEPROM 24C04
- [Sensor de velocidade](#)