

Projeto - Centro de Controle de Motores Máquinas Elétricas e Acionamentos

Ana Carolina Pela Tofetti

Gabriel Yamashita

Lila Takahashi Hadba

Raphael Jacob Butori

Rodrigo Carvalho Monteiro de Barros

Professores: Antonio F. Maiorquim e Silvio Szafir

Introdução

O uso dos CCMs popularizou-se por volta de 1950, durante a Terceira Revolução Industrial, visto que o uso de motores elétricos ficou cada vez mais popular no uso em aplicações industriais e comerciais, principalmente no setor automobilístico. O CCM é capaz de controlar os motores à distância por dentro de uma sala com ambiente controlado, e pode ser acompanhado de relés de proteção, que nesse caso, não foi o intuito do projeto.

O Projeto de Centro de Controle de Motores de Máquinas Elétricas e Acionamentos teve por intuito simular um CCM, através de uma IHM, criada pelo usuário, juntamente com a comunicação serial, entre ambos o IHM e o microcontrolador, que aciona independentemente dois motores de passo através de drivers.

Elaboração

Como todo o projeto, requisitos devem ser atendidos, esses foram:

- (i) i. Acionar e desacionar (Enable/Disable) o funcionamento do motor;
- (ii). Variar a rotação do motor entre uma velocidade mínima e máxima cadastrada nas configurações do motor no CCM;
- (iii). Escolher o sentido de rotação;
- (iv). Controlar o tempo de acionamento e/ou número de passos e/ou revoluções que o motor desenvolve.

Componentes Utilizados

- 1 fonte 12V DC
- 1 placa NUCLEO STM32F302R8
- 2 placas ST X-NUCLEO-IHM-4A1
- 2 Motor de Passo AK17/1.10 F6LN1.8

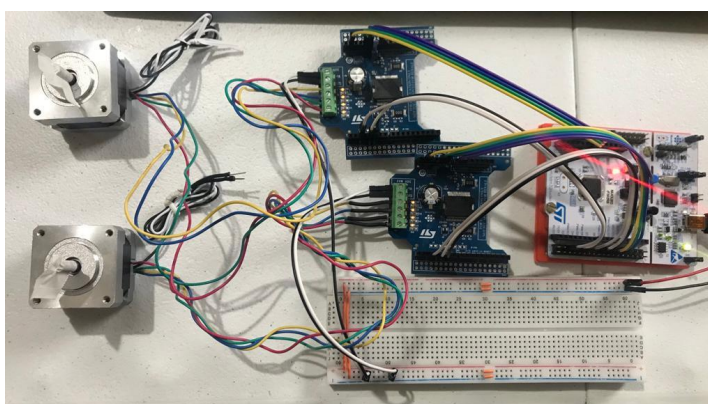


Figura 1 - Esquemático da Montagem

Hardware

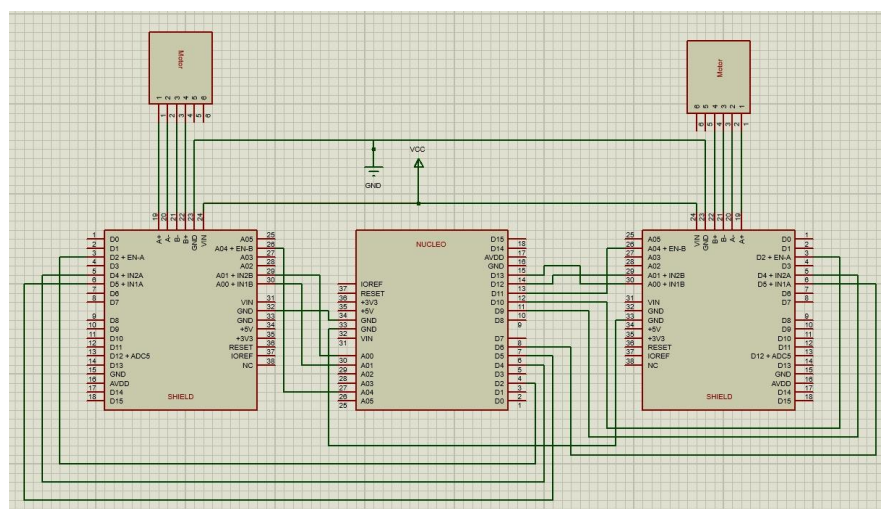


Figura 2 - Hardware do CCM

Software

Em suma temos 3 códigos diferentes, um que serve para controlar a IHM do motor, mandar e receber comandos pela serial feito em python. Os dois últimos são feitos em C ++, um que seria o código principal, que recebe os comandos pela serial e cuida dos acionamentos de cada motor, e por último um código que seria a classe de acionamento do motor de passo.

Agora em mais detalhes, o primeiro código, cuida da parte de construir e montar a tela da IHM, através da biblioteca PyGame, além da lógica de se conectar com a serial, com a biblioteca PySerial, e mandar os comandos certos de cada motor, de acordo com cada botão. O segundo código cuida da parte de inicializar o código, juntamente com a biblioteca Mbed, com as variáveis globais de acionamentos de cada motor, tais como o Enable, Direção, Emergência, Velocidade, entre outras. Esse código também cuida da parte de acionamento de cada motor, a partir de cada comando que vem pela serial mandado pelo código citado acima. Por último o terceiro código, é uma classe do motor de passo, responsável para simplificar os acionamentos de cada motor, e conta com métodos, tais como, mover o motor em somente um passo, definir as bobinas de acionamento, definir a velocidade de cada motor, e fazer a conversão da velocidade do motor de RPM para ms, que é usado para definir a velocidade que o motor irá rotacionar.

É possível ver os fluxogramas do código, que ajudaram na estruturação e criação do código.

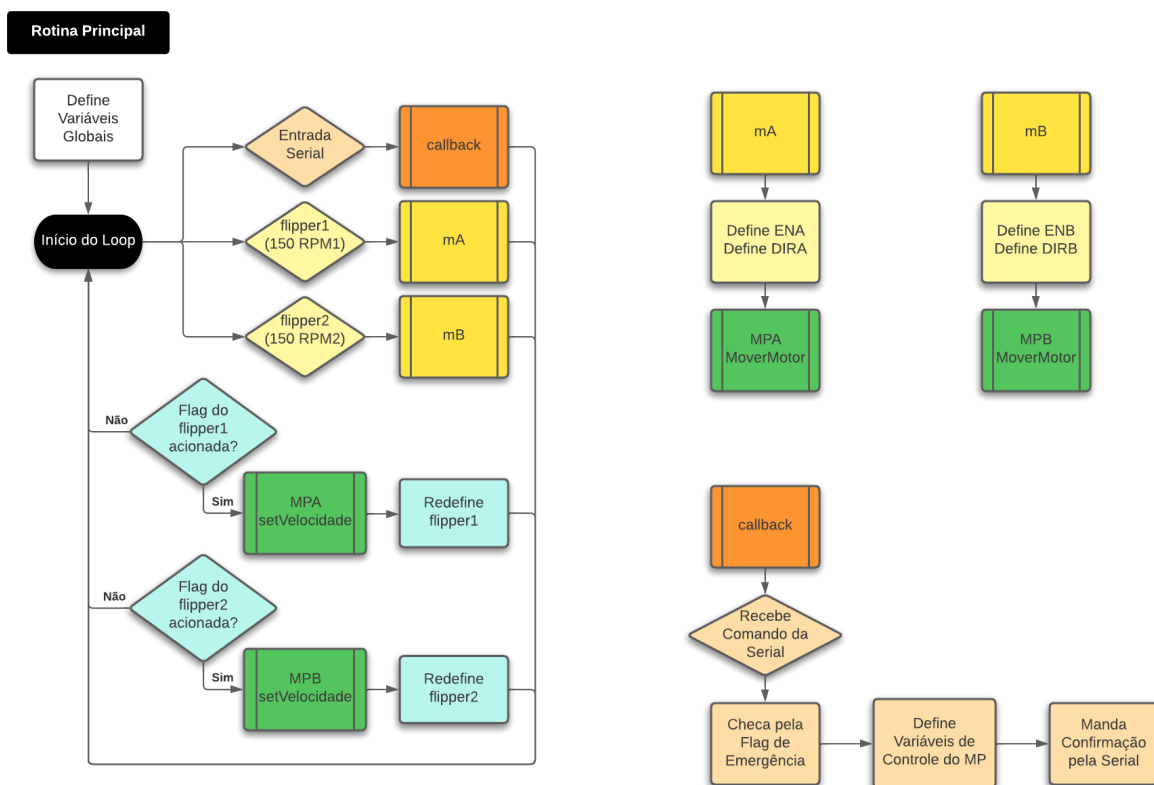


Figura 3 - Fluxograma do main.cpp

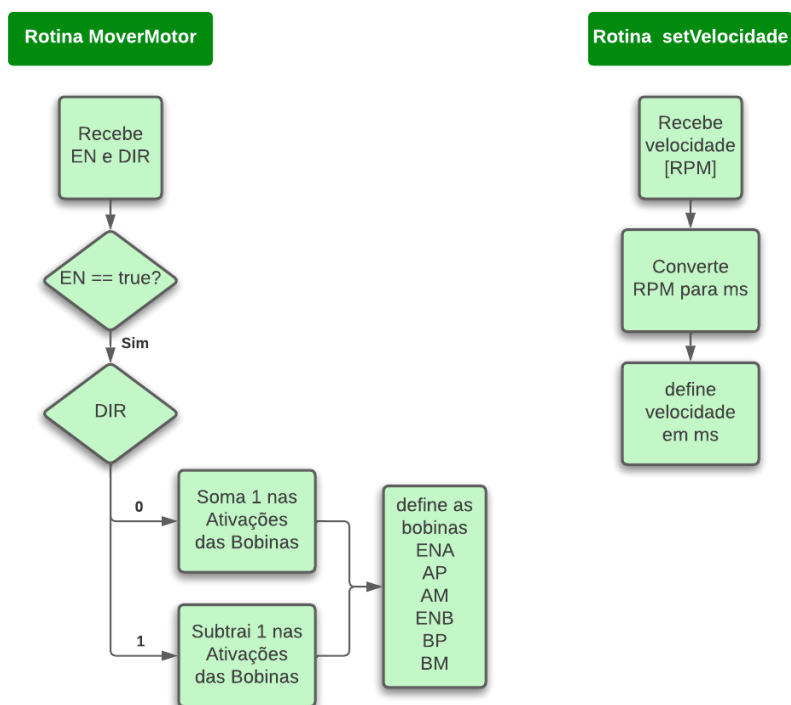


Figura 4 - Fluxograma do MotorPasso.cpp

Para uma melhor leitura, como os códigos, são grandes e para manter o relatório mais organizado, segue o link para o github onde é possível encontrar o código completo de todas as partes. Github: [Projeto CCM](#)

IHM

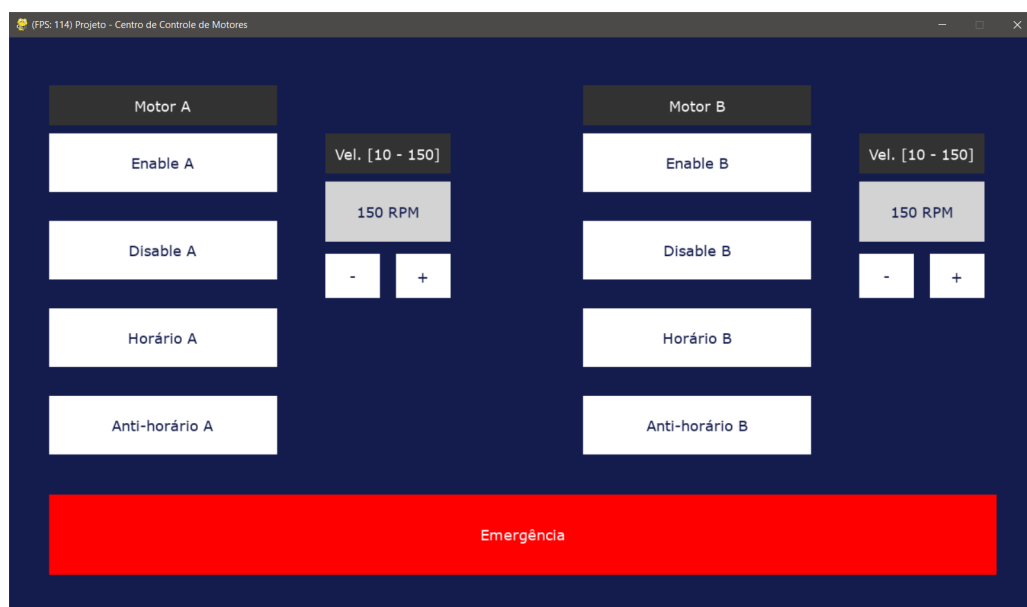


Figura 5 - IHM

Como é possível ver acima, essa foi a IHM, criada e utilizada para os acionamentos de cada motor. Nela é possível acionar e desabilitar cada um dos motores, além de controlar os sentidos e as velocidades em RPM também. Além disso, ela possui o botão de emergência, junto com a taxa de frames no título da aplicação.

Referências

1. [Slides de Aula de Máquinas Elétricas e Acionamentos](#)
2. [Wikipedia - Centro de Controle de Motores](#)
3. [Github - CCM](#)