

DESENVOLVIMENTO DE UM WEBLAB PARA CONTROLE DE VELOCIDADE DE MOTOR IMPLEMENTADO EM SISTEMA MICROCONTROLADO EMPREGANDO CONCEITOS DE INTERNET DAS COISAS

Guilherme De Gaspari Ximenes ¹; Alessandra Dutra Coelho²;
Luis Guilherme de Souza Munhoz ¹; Wânderson de Oliveira Assis ²;
Fernando de Almeida Martins ^{2,3}; Rogério Cassares Pires ³

¹ Aluno de Iniciação Científica do Instituto Mauá de Tecnologia (IMT);

² Professor(a) do Instituto Mauá de Tecnologia (IMT);

³ Engenheiro do Centro de Pesquisas do Instituto Mauá de Tecnologia (IMT).

Resumo. *A crescente evolução da tecnologia da informação e o surgimento de novas ferramentas computacionais propiciam consideráveis avanços nos projetos que englobam conceitos de Internet das Coisas, sistemas embarcados e tecnologias afins. Os laboratórios remotos (WebLabs) são exemplos claros de aplicação de ferramentas computacionais na educação, mas tipicamente direcionados para o desenvolvimento de ferramentas para ensino à distância, acessando remotamente pela Internet equipamentos instalados em laboratório. Nesse sentido a proposta do trabalho é criar um WebLab para controle de velocidade de motores com emprego de tecnologias de Internet das Coisas similares às utilizadas no projeto desenvolvido por MUNHOZ et al. (2022). Adicionalmente, pretende-se avaliar a oportunidade de desenvolvimento de uma plataforma universal, constituída de hardware padronizado, que permita a conexão de diversos tipos de sensores, de forma a viabilizar a rápida criação de interfaces para monitoramento de variáveis em aplicações industriais aplicando conceitos de Internet das Coisas.*

Introdução

O emprego crescente de ferramentas tecnológicas, aplicativos e laboratórios remotos é uma tendência e aumentou consideravelmente após a ocorrência da pandemia do COVID-19 (Assis e Coelho, 2022). Novas aplicações vêm sendo desenvolvidas fazendo o uso de recursos cada vez mais sofisticados, sobretudo no ensino do curso de Engenharia.

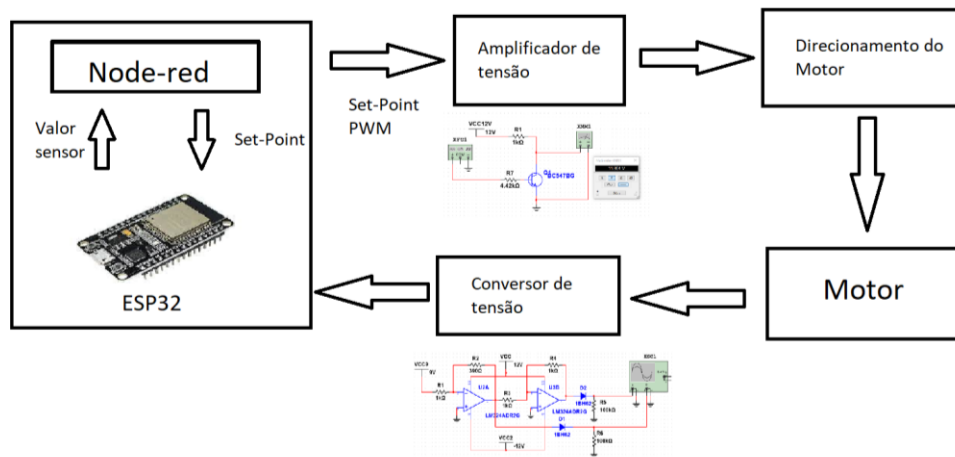
Este trabalho propõe o uso de ferramentas tecnológicas e laboratórios remotos visando o ensino na área de controle de processos. Com o custo elevado de softwares de controle que permitem a integração com aplicações controladas pela *web* e suas limitações na questão de adaptabilidade a diferentes dispositivos MUNHOZ et al. (2022) propôs uma solução responsiva e interativa relacionada ao controle de nível em um reservatório. Essa solução está integrada ao Smart Campus do IMT.

Nesse sentido, e dando continuidade à proposta, pretende-se desenvolver um WebLab de Controle de Velocidade de Motor.

Deseja-se desenvolver conceitos similares aos propostos por MUNHOZ et al. (2022) produzindo uma solução responsiva incluindo *back-end* e *front-end* que permita o controle usando qualquer dispositivo computacional, incluindo *smartphones*. Pretende-se adicionalmente avaliar ferramentas e tecnologias que podem ser empregadas para produzir uma plataforma universal, compatível com diversos sistemas e que permita a criação rápida de aplicações de IoT similares.

O fluxograma mostrado na figura 1 facilita o entendimento do funcionamento do projeto como um todo. Essa etapa foi crucial para a progressão do projeto, uma vez que permitiu uma visão clara e organizada das operações em curso.

Figura 1 - Fluxograma do Projeto

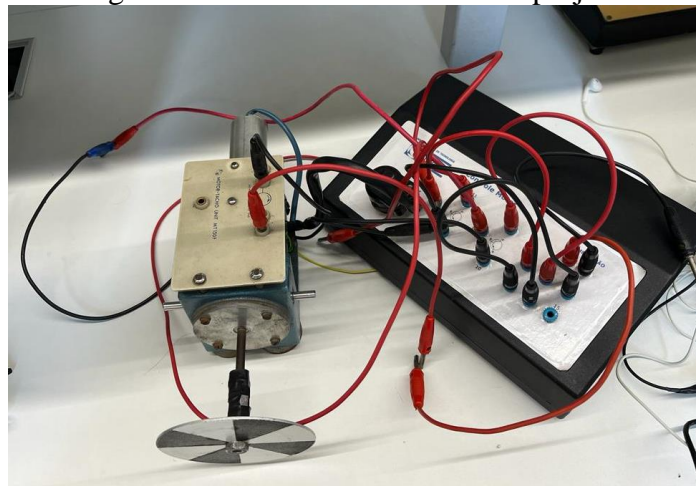


Materiais e Métodos

- Hardware

Sendo o projeto de iniciação científica amplamente baseado no anterior desenvolvido pelo aluno Luis Guilherme de Souza Munhoz, foi possível adiantar grande parte do projeto. Assim o trabalho pode ser iniciado em etapa avançada. Entretanto foram necessários ajustes para tornar viável a utilização do ESP32 de forma a ativar o motor (figura 1) com a velocidade necessária e monitorar sua velocidade. A variação de tensão do microcontrolador é de 0 a 3,3 Volts, para ativar o motor em sua velocidade máxima seriam necessários 12 Volts. Para ler o sensor de velocidade de acordo com a tensão recebida, precisaria que o ESP32 suportasse até 9 Volts em suas entradas analógicas.

Figura 2 - Motor DC referência do projeto.

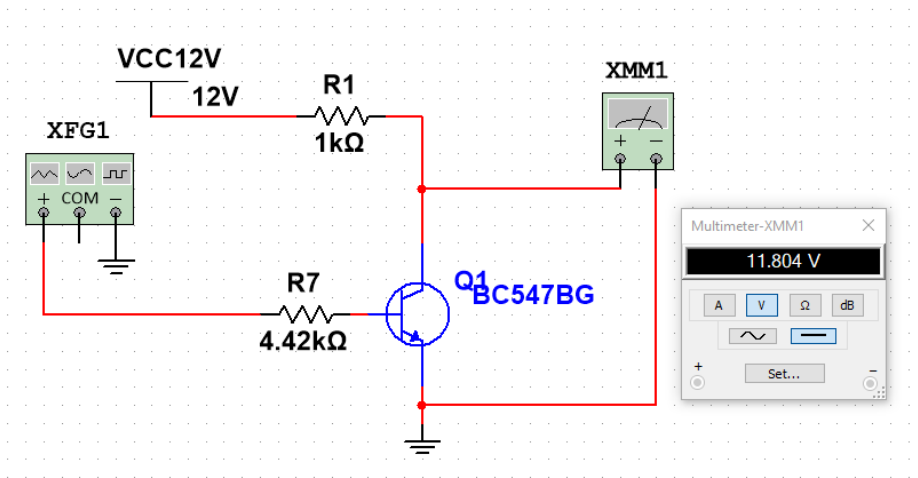


- Sistema de ativação do motor

Os trabalhos Assis et al, 2020 e Assis et al, 2011 serviram como referência para o estudo das soluções de controle aplicadas a motores DC. Para o projeto em questão, primeiramente foi necessário ampliar a voltagem de saída do ESP32 de 0 a 3,3 Volts para 0 a 12 Volts de tensão. Além disso precisaria de duas saídas PWM visto que para o motor funcionar em duas direções é necessário que a fonte de energia varie de -12 a +12 Volts. Assim foi criado um simples circuito (figura 2) para amplificar as duas saídas do ESP que possibilitou atingir a tensão

necessária utilizando uma fonte externa de 12 Volts, um transistor BC547 e dois resistores de 1k Ohms e 4.42k Ohms.

Figura 3 - Circuito de amplificador de tensão.

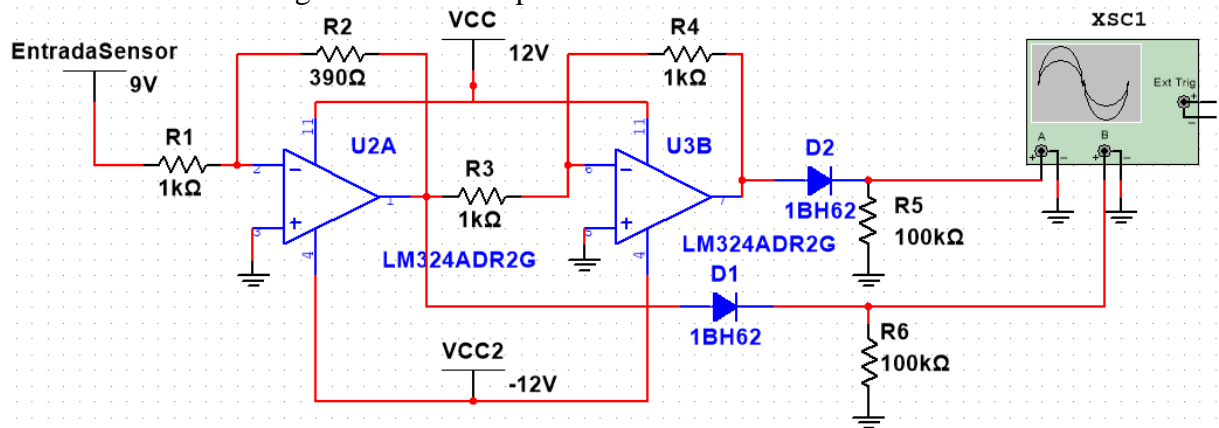


Como demonstrado na figura 3 o componente XFG1, um gerador de função, faz o papel do PWM do ESP que na simulação está em duty-cycle máximo, ou seja, enviando uma tensão de 3,3 Volts enquanto na saída temos uma tensão de 11,8 Volts possibilitando que o ESP envie a tensão necessária ao motor.

- Sistema de monitoramento de velocidade do motor

Para o monitoramento da velocidade do motor foi necessário um circuito capaz de ler a tensão de -9 a 9 Volts (figura 4) produzida pelo sensor de velocidade, foi utilizado uma fonte externa capaz de fornecer -12 a +12 Volts, um CI LM324, dois diodos e 6 resistores, sendo três deles de 1k Ohms, dois de 100k Ohms e um de 390 Ohms.

Figura 4 - Circuito para monitoramento de velocidade



Com o circuito demonstrado acima é possível registrar a velocidade do sensor do motor. Cada saída ligada ao osciloscópio XSC1 envia tensão de 0 a 3,3 Volts, sendo que uma envia a tensão quando o motor está produzindo -9 a 0 volts e a outra quando o motor produz 0 a 9 Volts. Esses sinais são aplicados ao ESP32 por duas entradas analógicas, o programa identifica qual das portas de entradas está recebendo a tensão e indica a direção e velocidade do motor.

- Desenvolvimento do Sistema de Controle Velocidade

A implementação dos circuitos mencionados anteriormente proporcionou avanços significativos no desenvolvimento do projeto.

- Softwares

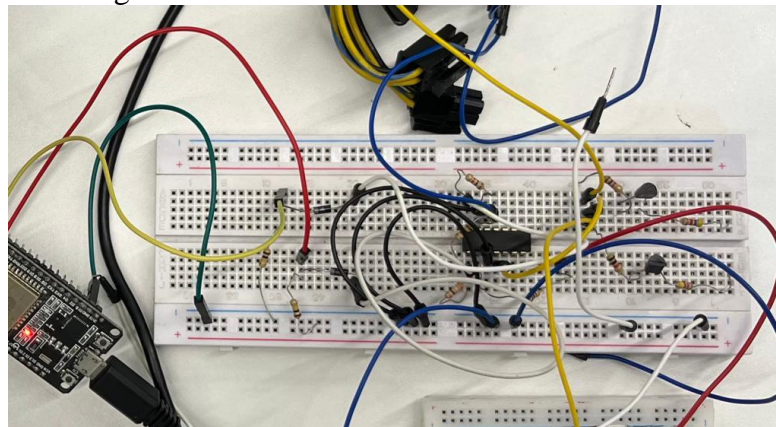
A condução do projeto WebLab foi estruturada em duas componentes fundamentais: o back-end e o front-end, conforme destacado por Silva et al. em 2019. O back-end desempenha um papel crucial na gestão das interações entre a interface e o hardware, sendo também responsável pela execução do código diretamente no hardware. Por outro lado, o front-end concentra-se na otimização da experiência do usuário (UX) e na criação da interface do usuário (UI).

No tocante ao desenvolvimento do front-end, a escolha recaiu sobre o Node-RED, uma ferramenta inovadora inicialmente concebida pela IBM para facilitar a conexão entre dispositivos de hardware, APIs e serviços online, sendo especialmente relevante no contexto da Internet das Coisas.

- Desenvolvimento

As primeiras etapas do desenvolvimento foram para garantir a funcionalidade do hardware. Para realizar os testes os componentes foram montados em uma protoboard, fontes externas forneceram a tensão para simular a funcionalidade do sensor do motor enquanto o ESP32 efetuava a leitura analógica para confirmar os dados. Como pode ser visto na figura 5.

Figura 5 - Protoboard com o hardware montado.



Após o efetivo funcionamento do hardware, o próximo passo foi o desenvolvimento dos softwares.

- Desenvolvimento do *back-end* do sistema de controle de velocidade

Node-RED é uma plataforma de desenvolvimento visual de código aberto, projetada para facilitar a criação de aplicativos para a Internet das Coisas (IoT) e automação. Desenvolvida originalmente pela IBM, o Node-RED oferece uma abordagem intuitiva e baseada em fluxo para conectar dispositivos, APIs e serviços de maneira eficiente. Sua interface gráfica simplificada permite que desenvolvedores e entusiastas criem fluxos de trabalho complexos sem a necessidade de uma extensa codificação manual. As figuras 6-9 mostram o desenvolvimento do back-end do sistema no Node-RED.

Figura 6 - Back-end do Node-RED responsável de receber a mensagem enviada pelo ESP32 via WIFI.



Figura 7 - Back-end responsável por preparar a informação para enviá-la ao Dashboard pelos blocos em azul.

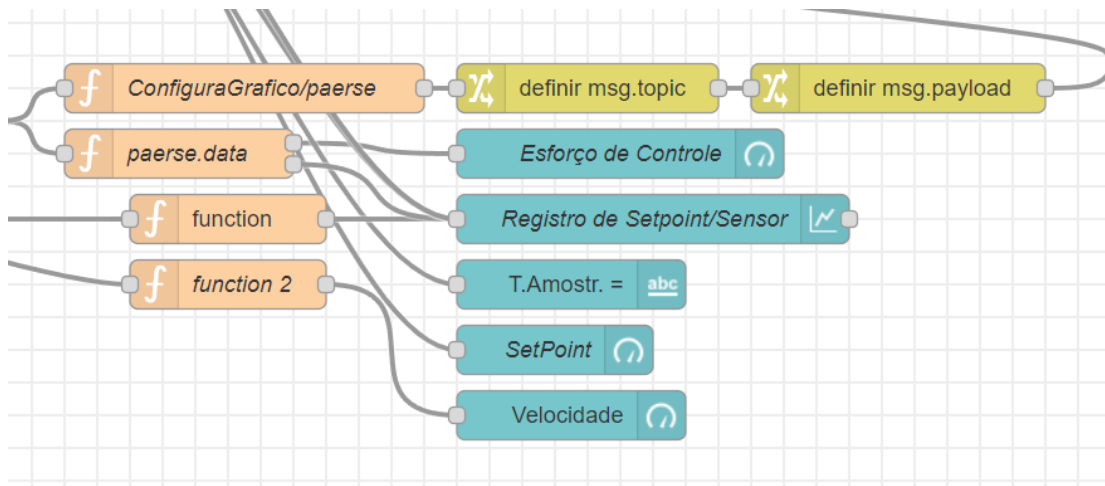


Figura 8 - Back-end responsável por receber as configurações desejadas do PID e inseri-las a função PID.

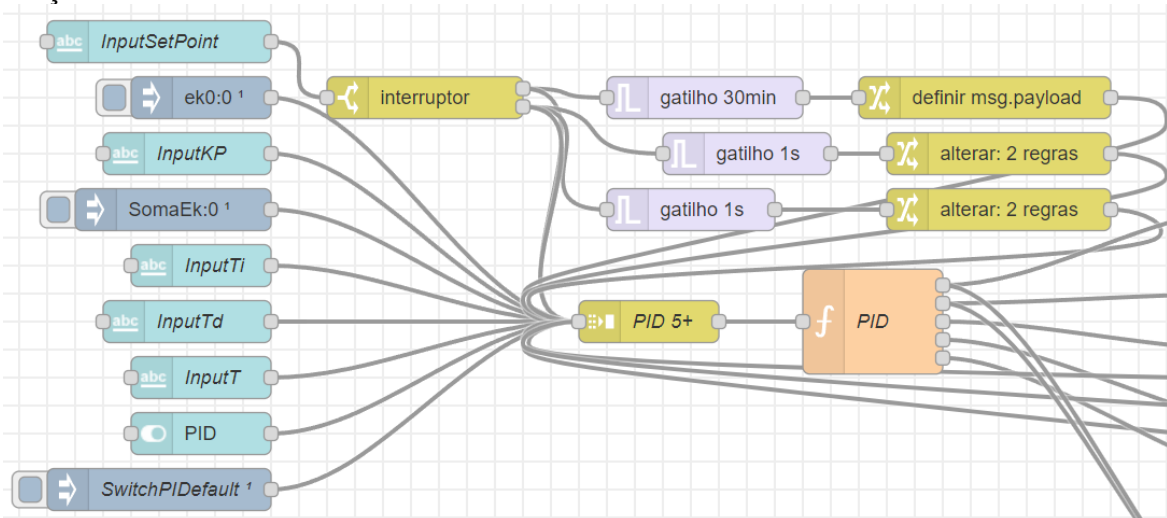
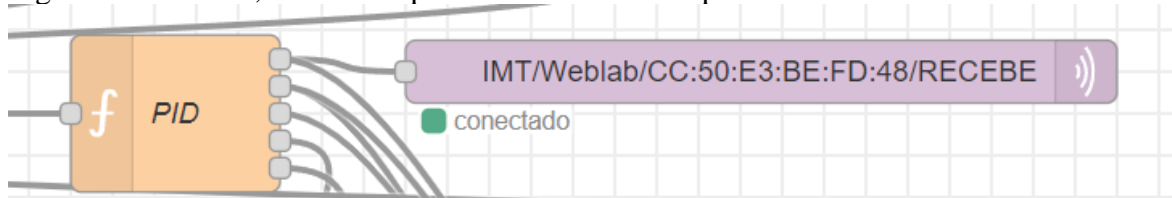


Figura 9 - Em roxo, o bloco responsável a enviar a resposta do controlador e o Set-Point.



A principal vantagem do Node-RED reside na representação visual das lógicas de programação, utilizando uma abordagem de "arrastar e soltar" para conectar nodes que executam funções específicas. Essa característica torna o Node-RED acessível mesmo para aqueles com pouca experiência em programação, ao mesmo tempo em que oferece flexibilidade para projetos mais avançados.

Com uma biblioteca expansiva de nodes prontos para uso e suporte para uma variedade de protocolos e dispositivos, o Node-RED se destaca como uma ferramenta versátil, acelerando o desenvolvimento de soluções IoT e proporcionando uma experiência eficaz no design de automações e integrações.

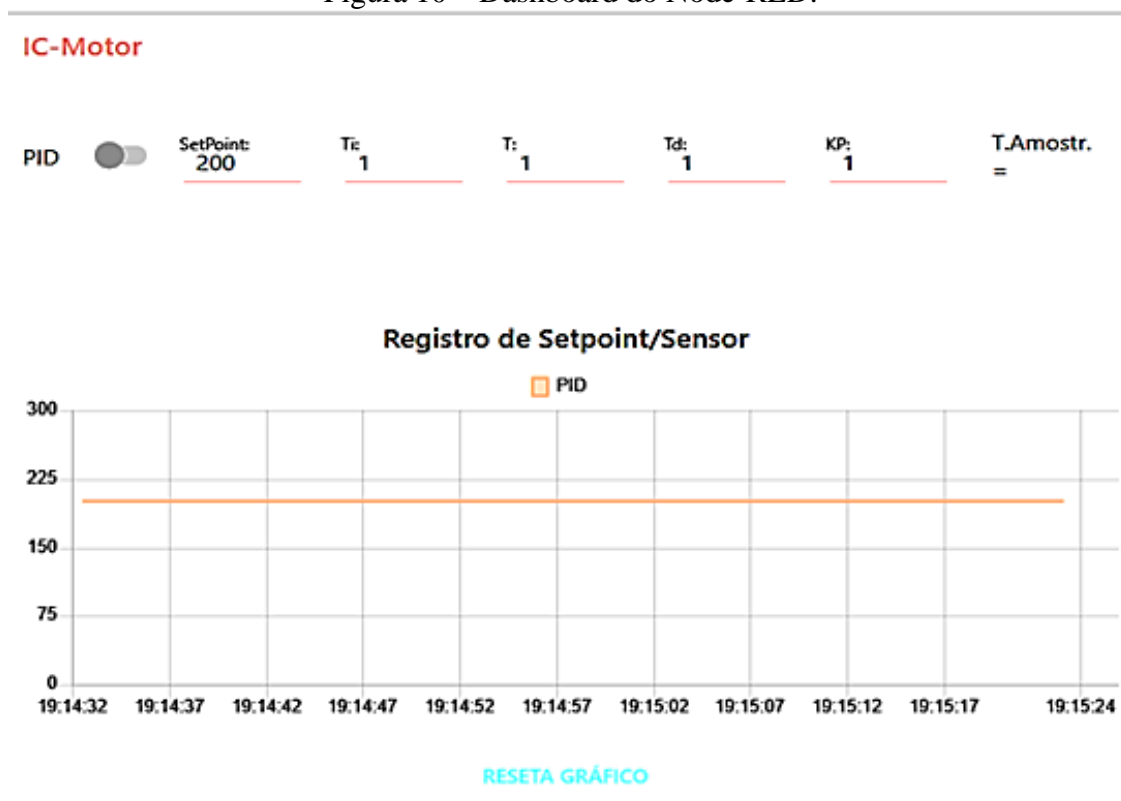
Como já citado anteriormente este trabalho é baseado em uma iniciação científica do ano anterior retirando a necessidade de desenvolver o back-end do zero, foi realizada a adaptação do desenvolvimento anterior para a aplicação atual. Desta forma, foi realizado um estudo detalhado do código existente no Node-Red para assim fazer uma adaptação otimizada para o projeto.

Para a integração com o ESP32 foi necessário utilizar as bibliotecas: “WiFi.h”, “PubSubClient.h”, “Wire.h” e “ArduinoJson.h”.

- Desenvolvimento do *front-end* do sistema de controle de velocidade

A concepção da interface foi meticulosamente planejada para oferecer uma experiência imersiva, responsiva e interativa aos usuários. Nesse ambiente, os usuários têm a capacidade de inserir valores nas variáveis de controle e realizar uma variedade de experimentos, desde a análise da relação linear entre velocidade e tensão medida no sensor até a modelagem do sistema em resposta a degraus e a sintonização de controladores PID discretos. Essas funcionalidades foram implementadas por meio do robusto conjunto de dashboards (figura 10) proporcionado pelo Node-RED.

Figura 10 – Dashboard do Node-RED.



Destacando-se como a principal vantagem, o Node-RED oferece um pacote de dashboards que possibilitou a criação de uma interface de usuário rica em funcionalidades, com uma variedade de botões, entradas de dados e modelos visuais. A implementação dos templates envolveu a utilização de funções em Javascript, HTML e CSS, permitindo não apenas a representação gráfica dos tanques, mas também a incorporação em tempo real de imagens da câmera, enriquecendo ainda mais a experiência do usuário.

Resultados e Discussão

Os resultados obtidos representam um avanço promissor na implementação do sistema. A leitura e processamento do sinal do ESP32, seguidos pela transmissão da ordem de PWM para o motor, demonstraram eficácia inicial. No entanto, confrontamo-nos com uma possível limitações na capacidade de processamento do Node-RED, que pode estar tornando inviável a aplicação contínua do controle PID por ele. Com o tempo, observa-se a manifestação de atrasos no dashboard e falhas no controle, culminando eventualmente no congelamento completo do programa.

Diante desse cenário, torna-se imperativo migrar a lógica do controle PID para o código do ESP32, aproveitando sua capacidade de processamento superior. Essa abordagem não apenas soluciona as questões identificadas, mas também preserva o papel do Node-RED para a apresentação eficiente de dados, sem sobrecarregar sua capacidade computacional.

Ao trilhar esse caminho, a integração do Controlador do Motor no Smart Campos do IMT emerge como uma inevitabilidade. A transferência do controle PID para o ESP32 não apenas atende às demandas de desempenho, mas também pavimenta o caminho para uma integração mais harmoniosa e efetiva no contexto mais amplo do Smart Campos.

Em síntese, a conclusão é que, ao migrar o controle PID para o ESP32, estamos não apenas resolvendo desafios operacionais, mas também preparando terreno para uma integração robusta e eficiente do controlador do motor no ecossistema do Smart Campos do IMT. Essa evolução é uma questão de tempo, representando um passo essencial para o aprimoramento contínuo e a eficácia do sistema como um todo.

- Experimentos

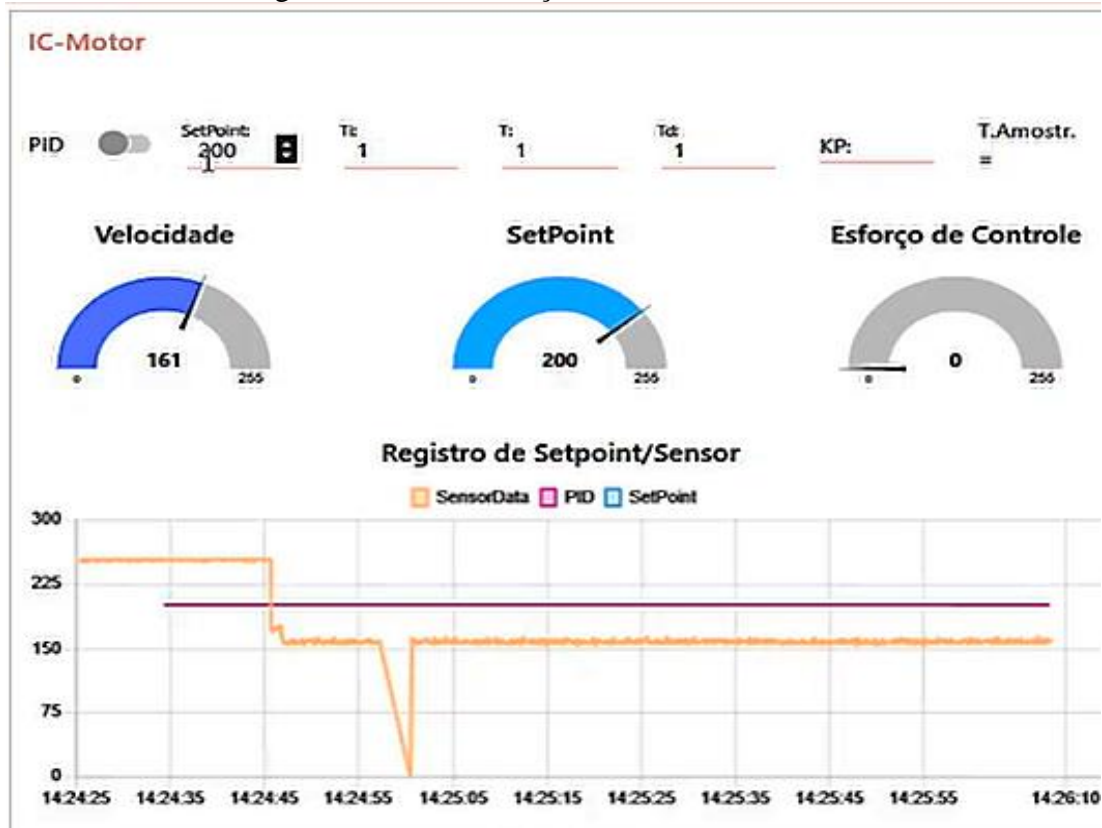
Os primeiros experimentos realizados demonstraram os problemas causados pelo processamento utilizando o Node-red. Felizmente foi possível coletar dados suficiente do funcionamento do PID e da resposta enviada para o motor.

Figura 11 - Experimento ativando o PID.



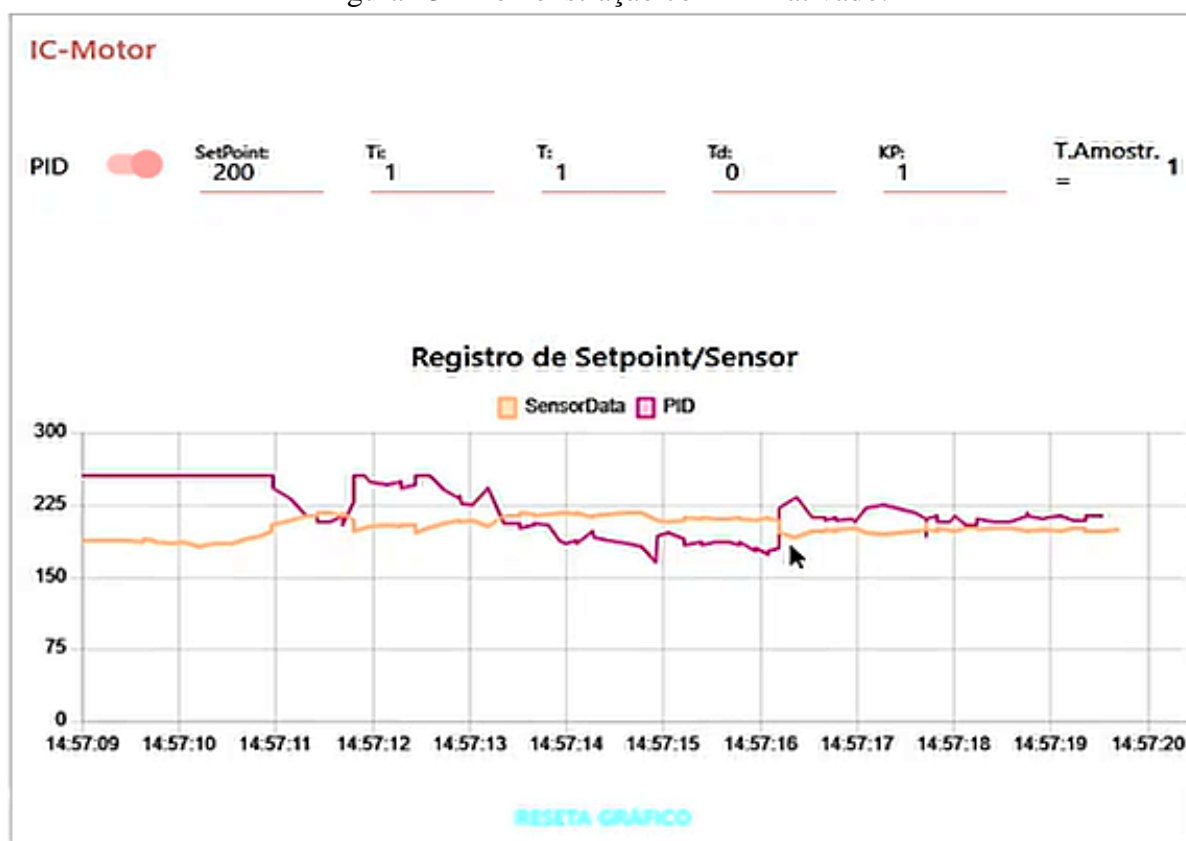
Na imagem acima (figura 11) é possível identificar três momentos importantes, antes do acionamento do controle, durante o funcionamento do controle enquanto set point estava no seu objetivo e após a velocidade do motor ficar abaixo do set point. No primeiro momento em que o PID está desativado, o set point é mantido constante em relação a velocidade do motor independente da variação do sensor de velocidade como demonstrado na figura 12.

Figura 12 - Demonstração com PID desativado.



No segundo momento, após o PID ser ativado é possível observar que o setpoint do PID se desloca para valor 0 por conta do setpoint estar atingido, porém, como demonstra o terceiro momento, assim que o sensor do motor demonstra velocidade inferior a escolhida, o PID começa a mandar sinal para aumentar a tensão do motor e assim fazer sua velocidade voltar ao alvo, como demonstrado na figura 13.

Figura 13 - Demonstração com PID ativado.



Conclusões

Com base nos resultados obtidos até o momento, fica evidente que a implementação do controle PID diretamente no código do ESP32 é uma medida necessária para garantir a estabilidade e eficiência do sistema. Os testes revelaram que a capacidade de processamento do Node-Red, embora útil para a leitura e processamento inicial do sinal do ESP32, aparenta insuficiente para a aplicação contínua e precisa do controle PID ao longo do tempo.

Ao transferir a responsabilidade do controle PID para o ESP32, que possui maior poder de processamento, espera-se superar os desafios observados, como atrasos e falhas na aplicação do controle. Essa abordagem não apenas otimiza o desempenho do sistema, mas também permite que o Node-Red cumpra seu papel principal de apresentação de dados no dashboard, sem sobrecarregar sua capacidade de processamento.

A migração do controle PID para o código do ESP32 não apenas resolve as questões identificadas, mas também abre caminho para uma integração mais eficaz do controlador do motor no Smart Campos do IMT. A centralização do controle de motor no ESP32 proporciona uma solução mais robusta e autônoma, contribuindo para a confiabilidade do sistema como um todo.

Em resumo, a conclusão é que a otimização do sistema envolve a transferência do controle PID para o ESP32, maximizando assim a capacidade de processamento e garantindo um funcionamento estável e eficiente, enquanto o Node-Red continua desempenhando um papel na apresentação de dados no Smart Campos do IMT.

Referências Bibliográficas

- Assis, W. O., Coelho, A. D. (2022). Uso de Tecnologias Acessíveis Remotamente como Estratégia de Ensino-Aprendizagem na Engenharia. *The Journal of Engineering and Exact Sciences*, 8(2), 13948–01e. <https://doi.org/10.18540/jcecvl8iss2pp13948-01e>
- Assis, W. O., Coelho, A. D.; Gonçalves, H. S. B. WebLabs: Remote Access Experiments for Teaching Process Control in Engineering Courses. *XIV Technologies Applied to Electronics Teaching Conference (TAAE)*, Porto, Portugal, 2020, pp. 1-7, doi: <https://doi.org/10.1109/TAAE46915.2020.9163667>.
- Assis, W. O.; Fernandes, T.; Pacheco, V. G.; Souza Júnior, J. C.; Coelho, A. D.; Romano, R. A. Um WebLab para Ensino de Controle em Cursos de Engenharia. COBENGE 2011. Disponível em: https://r.search.yahoo.com/_ylt=AwrEry8uIcZj_N8Zlg_z6Qt.;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1673957807/RO=10/RU=http%3a%2f%2fwww.abenge.org.br%2fcobenge%2flegado%2farquivos%2f8%2fseessoestec%2fart1583.pdf/RK=2/RS=WPB15cBbBI7nDoo02BMx8nx803g-
- Munhoz, L. G. S.; Assis, W. O.; Coelho, A. D.; Martins, F. A.; Pires, R. C. Desenvolvimento de Laboratório Remoto na Área de Controle Integrado ao Smart Campus. Anais do 14º. Seminário Mauá de Iniciação Científica. 2022. Disponível em: <https://maua.br/files/munhoz-assis-1670870820.pdf>
- Silva, G. T.; Assis, W. O.; Coelho, A. D. Uma Nova Concepção de Laboratório Remoto e sua Utilização como Ferramenta Didática na Implementação de Sistema de Controle de Nível. In: *SEMINÁRIO MAUÁ DE INICIAÇÃO CIENTÍFICA*, São Caetano do Sul, SP, 2019. <https://maua.br/files/122019/uma-nova-concepcao-laboratorio-remoto-sua-utilizacao-como-ferramenta-didatica-implementacao-sistema-controle-nivel-261216.pdf>