

SISTEMA PID EM MOTOR SEM ESCOVAS COM ARDUINO E IOT

Eduardo Bejamin da Silva*; Jean Carlos Schneider*; Carlos Lavrado Filho**

*Acadêmico de Engenharia de Software - Faculdade Uniguaçu, Jeanschneider@icloud.com;
dudu15smi@gmail.com.

**Mestre em Engenharia Aeroespacial - ITA/MAI, eng.lavrado@gmail.com.

INFORMAÇÕES

Histórico de submissão:

Recebido em: 27 maio 2025
Aceite: 08 jun. 2025
Publicação online: jun. 2025

RESUMO

Este artigo apresenta um estudo sobre sistemas de controle PID, com foco na implementação de um protótipo de motor brushless controlado via Arduino e IoT (Internet das Coisas). O objetivo principal é comparar diferentes configurações de parâmetros PID para otimizar a resposta do motor. Para isso, foram utilizados um Arduino Uno, um controlador eletrônico de velocidade (ESC), um motor brushless de 1000KV, uma hélice, uma bateria LiPo 3S 2200mAh 30C e um módulo ESP8266. A metodologia envolveu o desenvolvimento do algoritmo PID, a realização de testes práticos e a análise da resposta do motor. Os resultados visam demonstrar a viabilidade da integração entre controle PID e IoT, destacando os benefícios da automação remota.

Palavras-chave: Controle, PID, IoT, Arduino, Motor Brushless.

ABSTRACT

This article presents a study on PID control systems, focusing on the implementation of a brushless motor prototype controlled via Arduino and IoT (Internet of Things). The main objective is to compare different PID parameter configurations to optimize the motor's response. For this purpose, an Arduino Uno, an electronic speed controller (ESC), a 1000KV brushless motor, a propeller, a 3S 2200mAh 30C LiPo battery, and an ESP8266 module were used. The methodology involved the development of the PID algorithm, practical testing, and analysis of the motor's response. The results aim to demonstrate the feasibility of integrating PID control with IoT, highlighting the benefits of remote automation.

Keywords: Control, PID, IoT, Arduino, Brushless Motor.

Copyright © 2025, Eduardo Bejamin da Silva; Jean Carlos Schneider; Carlos Lavrado Filho. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Citação: DA SILVA Eduardo Bejamin; SCHNEIDER, Jean Carlos; LAVRADO FILHO, Carlos. Sistema PID em motor sem escovas com arduino e IoT. *Iguazu Science*, São Miguel do Iguazu, v. 3, n. 7, p. 125-134 jun. 2025.

INTRODUÇÃO

O controle eficiente de sistemas dinâmicos é uma das áreas mais importantes e desafiadoras dentro da engenharia, especialmente na engenharia elétrica, eletrônica e de controle e automação. Esses sistemas estão presentes em uma grande variedade de aplicações modernas, como veículos autônomos, linhas de produção industriais, sistemas robóticos, equipamentos médicos e aparelhos domésticos inteligentes. Em todos esses contextos, a necessidade de manter o comportamento do sistema dentro de limites desejáveis, mesmo diante de perturbações externas ou variações internas, torna indispensável o uso de estratégias de controle robustas.

Dentre as diversas abordagens desenvolvidas ao longo das décadas para controlar sistemas dinâmicos, o controlador proporcional-integral-derivativo (PID) destaca-se como uma das mais utilizadas no meio acadêmico e industrial (Brito; Ferreira, 2021).

Isso se deve à sua relativa simplicidade conceitual e à sua eficácia prática em uma ampla gama de sistemas lineares e levemente não lineares. O controlador PID atua reduzindo o erro entre a variável de processo e o valor de referência (setpoint), aplicando correções proporcionais ao erro atual (P), acumuladas ao longo do tempo (I) e baseadas na taxa de variação do erro (D). A correta combinação desses três efeitos permite ao controlador alcançar boa estabilidade, resposta rápida e mínima oscilação,

desde que os parâmetros de ganho (K_p , K_i e K_d) sejam devidamente ajustados.

Entretanto, o sucesso do controle PID depende diretamente do ajuste fino desses parâmetros. Um pequeno desvio nos valores dos ganhos pode levar a um sistema com comportamento instável, lento ou com excesso de oscilação. Tradicionalmente, o ajuste desses parâmetros é feito de forma manual ou empírica, exigindo conhecimento técnico e tempo. Esse processo pode se tornar ainda mais complexo quando o sistema a ser controlado está sujeito a condições variáveis, como alterações de carga, temperatura ou interferência externa, comuns em ambientes reais de operação.

Nesse cenário, a Internet das Coisas (IoT – Internet of Things) tem surgido como uma tecnologia transformadora, trazendo novas possibilidades para o desenvolvimento e operação de sistemas de controle inteligentes. A IoT permite a interconexão de dispositivos físicos à rede, possibilitando a coleta de dados em tempo real, o monitoramento remoto e o envio de comandos a distância.

A combinação de microcontroladores acessíveis, como o Arduino, com módulos de comunicação sem fio, como o ESP8266, tem proporcionado uma plataforma poderosa e de baixo custo para a prototipagem de soluções baseadas em IoT. O Arduino, por sua arquitetura aberta, facilidade de programação e ampla compatibilidade com sensores e atuadores, permite a implementação de algoritmos de controle embarcados com grande flexibilidade. Já o ESP8266, por incorporar conectividade Wi-Fi, expande significativamente as capacidades do sistema, tornando possível o envio e recebimento de dados através da internet em tempo real.

Diante desse contexto tecnológico promissor, este estudo propõe o desenvolvimento, a implementação e a análise de um sistema de controle PID aplicado ao controle de um motor brushless (sem escovas), utilizando o Arduino como plataforma de controle e o ESP8266 como interface de comunicação com a rede. O objetivo principal é criar uma solução funcional e acessível para o controle preciso de motores, com monitoramento remoto via IoT, possibilitando ajustes dinâmicos dos parâmetros PID e a visualização contínua do desempenho do sistema.

Esse tipo de abordagem visa não apenas validar a viabilidade técnica da integração entre controle clássico e conectividade moderna, mas também demonstrar como ferramentas acessíveis podem ser empregadas em projetos de automação avançada, com potencial de aplicação em educação tecnológica, pesquisa acadêmica, desenvolvimento de protótipos industriais e inovação em startups. Através desse estudo, espera-se contribuir para o avanço do conhecimento prático sobre controle PID em ambientes reais e conectados, e abrir caminho para o

uso de IoT como facilitador de sistemas autônomos e adaptativos no mundo físico.

FUNDAMENTAÇÃO TEÓRICA

O controle PID é um método amplamente utilizado em sistemas de automação para garantir que a saída de um sistema físico siga um valor de referência (setpoint). Ele atua com base na diferença entre o valor atual e o valor desejado, aplicando correções proporcionais ao erro, à sua integral acumulada e à sua taxa de variação.

O motor brushless, por sua vez, requer um controlador eletrônico (ESC) que converte sinais PWM em comutação adequada para seu funcionamento. O sinal PWM, variando entre 1000 e 2000 microssegundos, define a velocidade do motor. A combinação desses elementos com o ESP8266 permite não apenas o controle local do motor, mas também o ajuste remoto via rede Wi-Fi, viabilizando aplicações em sistemas distribuídos e conectados.

PRINCÍPIO DE FUNCIONAMENTO DO MOTOR BRUSHLESS

Os motores brushless, também conhecidos como motores BLDC (Brushless Direct Current), são motores de corrente contínua que, diferentemente dos motores tradicionais, não utilizam escovas para realizar a comutação elétrica. Em vez disso, a troca das correntes nos enrolamentos é feita eletronicamente, por meio de um controlador externo. Essa arquitetura resulta em menor desgaste mecânico, maior eficiência energética, operação mais silenciosa e menor necessidade de manutenção, o que torna os motores brushless ideais para aplicações modernas em robótica, drones, veículos elétricos e sistemas de ventilação de precisão.

Os motores brushless mais comuns possuem configuração trifásica, composta por três enrolamentos dispostos no estator e ímãs permanentes no rotor. O princípio básico de funcionamento está na criação de um campo magnético rotativo no estator através da aplicação de correntes defasadas nas três fases. À medida que essas correntes variam, o campo magnético resultante se move, atraindo e repelindo os polos do rotor e gerando movimento rotacional contínuo.

Como a comutação é eletrônica, torna-se necessário o uso de um circuito de controle externo que defina com precisão a sequência e o tempo de energização das bobinas. Essa tarefa é realizada por um componente chamado ESC (Electronic Speed Controller), que atua como ponte entre os sinais de controle digitais (geralmente PWM) e os pulsos de corrente alternada que alimentam os enrolamentos do motor.

ESC – CONTROLADOR ELETRÔNICO DE VELOCIDADE

O ESC é um dispositivo eletrônico especializado que interpreta sinais PWM (Pulse Width Modulation) enviados por microcontroladores, como o Arduino, e os converte em pulsos elétricos apropriados para controlar a velocidade, torque e sentido de rotação do motor brushless (GARCIA, 2020). Sua função é crítica para a operação do motor, pois ele realiza o chaveamento dos transistores de potência de forma sincronizada com a posição do rotor, garantindo o correto funcionamento da máquina.

Como os motores brushless requerem correntes relativamente altas para operar, o ESC é alimentado por uma fonte externa, geralmente uma bateria de lítio ou fonte de corrente contínua de alta capacidade. Além disso, muitos ESCs modernos contam com funcionalidades adicionais, como frenagem regenerativa, detecção de sobrecorrente, proteção contra superaquecimento e modos de partida suave.

ARDUINO UNO COMO PLATAFORMA DE CONTROLE

O Arduino UNO, baseado no microcontrolador ATmega328P, é amplamente utilizado no desenvolvimento de sistemas embarcados devido à sua arquitetura simples, vasta documentação e grande comunidade de suporte. No contexto deste projeto, o Arduino é responsável pela implementação do algoritmo PID em tempo real. Ele calcula os valores de correção com base no erro entre a velocidade desejada e a velocidade medida (ou estimada) do motor e envia comandos na forma de sinais PWM ao ESC.

Além do controle PID, o Arduino também desempenha a função de interface com o módulo ESP8266, possibilitando a comunicação com redes Wi-Fi para que os parâmetros do sistema possam ser monitorados e ajustados remotamente. Essa capacidade aumenta significativamente a flexibilidade e a eficiência do sistema, permitindo adaptações dinâmicas e testes em tempo real.

INTEGRAÇÃO COM ESP8266 – CONECTIVIDADE IOT

O ESP8266 é um microcontrolador com conectividade Wi-Fi integrada, amplamente utilizado em projetos de Internet das Coisas (IoT) devido ao seu baixo custo, consumo reduzido de energia e facilidade de integração com sistemas embarcados. Neste projeto, o ESP8266 foi utilizado como interface de comunicação entre o sistema de controle PID e a rede local, permitindo o monitoramento e a interação remota com o motor brushless.

Por meio do ESP8266, informações como o estado do sistema, o acionamento de funções específicas (como estabilizar ou inclinar o motor em 45 graus) e os ajustes necessários podem ser enviados e recebidos através de uma página web simples e responsiva. A comunicação é realizada via Wi-Fi, possibilitando que

o usuário interaja com o sistema em tempo real, sem necessidade de conexão física direta.

Essa integração permite um controle mais flexível e acessível, viabilizando testes, ajustes e operações remotas do sistema. A aplicação da IoT, com auxílio do ESP8266, demonstra como tecnologias acessíveis podem ser empregadas para aumentar a eficiência e a usabilidade de sistemas de automação e controle.

METODOLOGIA

A metodologia adotada neste projeto foi estruturada em etapas sequenciais, visando o desenvolvimento, a implementação prática e a validação de um sistema de controle PID aplicado a um motor brushless, com monitoramento e ajuste remoto por meio de conectividade IoT. As atividades foram divididas em fases de planejamento, montagem do hardware, programação do sistema embarcado, integração com a rede e testes experimentais.

Inicialmente, foram definidos os componentes eletrônicos necessários, incluindo o microcontrolador ESP8266, o motor brushless e o ESC. Em seguida, foi realizada a montagem do circuito em bancada, conectando o ESC ao motor e ao ESP8266, além do sensor responsável por fornecer o sinal de realimentação para o controle PID.

A programação foi desenvolvida em linguagem C++, utilizando a IDE do Arduino. O controle PID foi implementado diretamente no microcontrolador, recebendo o valor do sensor e ajustando dinamicamente o sinal PWM enviado ao ESC para manter a estabilidade ou atingir um ângulo de referência.

Para a interface de controle, foi criada uma página web hospedada no próprio ESP8266, com três comandos principais: estabilizar, levantar 45° e parar o motor. A comunicação entre o usuário e o sistema embarcado é realizada via Wi-Fi, sem necessidade de roteador externo, utilizando o modo Access Point da placa.

Por fim, foram conduzidos testes experimentais com o sistema montado, validando o funcionamento do controle PID, a resposta do motor e a eficácia da interface web para comandos remotos. Os dados observados permitiram analisar a estabilidade, precisão e tempo de resposta do sistema.

DEFINIÇÃO DO ESCOPO E REQUISITOS DO SISTEMA

Inicialmente, foi realizado um levantamento dos requisitos funcionais e técnicos necessários para o desenvolvimento do sistema. Foram definidos como objetivos principais: o controle preciso da rotação de um motor brushless, a implementação de um algoritmo PID no microcontrolador Arduino UNO (Tiago; Costa, 2018) e a comunicação remota com um

módulo ESP8266, responsável por enviar e receber dados via rede Wi-Fi (Oliveira, 2022).

Além disso, estabeleceu-se a necessidade de registrar o comportamento dinâmico do sistema, permitindo a realização de ajustes em tempo real nos parâmetros PID e a execução de comandos remotos, como estabilização ou inclinação do motor, possibilitando a observação direta da influência dessas ações sobre a resposta do sistema.

SELEÇÃO E MONTAGEM DOS COMPONENTES DE HARDWARE

Com base nos requisitos identificados, os seguintes componentes foram selecionados e integrados:

- Motor brushless trifásico: com ímãs permanentes no rotor e três bobinas no estator (Figura 1);

Figura 1. Motor Brushless



Fonte: Mercado Livre

- ESC (Electronic Speed Controller): responsável por converter sinais PWM em pulsos de corrente trifásica alternada para controle do motor (Figura 2);

Figura 2: ESC



Fonte: Mercado Livre

- Arduino UNO (ATmega328P): para execução do algoritmo PID, geração dos sinais PWM e comunicação serial com o ESP8266 (Figura 3);

Figura 3: Arduino UNO



Fonte: Mercado Livre

- Módulo ESP8266: para gerenciamento da comunicação sem fio com a rede Wi-Fi (Figura 4);

Figura 4: Módulo ESP8266



Fonte: Mercado Livre

- Fonte externa de alimentação: necessária para fornecer a corrente adequada ao ESC e ao motor brushless; foi utilizada uma bateria LiPo de 2200 mAh, 3S (11,1 V), 30C, garantindo energia suficiente para o funcionamento estável do sistema.

Figura 5: Bateria LiPo



Fonte: Mercado Livre

Após a aquisição dos componentes, foi realizada a montagem do circuito em uma protoboard e estrutura de teste com organização dos fios, conexões e isolamento adequado das linhas de potência e controle.

Com a montagem finalizada, realizamos os testes necessários para verificar o desempenho dos componentes em conjunto e verificar o melhor posicionamento para aproveitar ao máximo o pequeno espaço disponível (Figura 6).

Figura 6: Projeto montado



Fonte: Proprio Autor

IMPLEMENTAÇÃO DO ALGORITMO PID NO ARDUINO

A próxima etapa consistiu no desenvolvimento e upload do código-fonte para o Arduino, utilizando a IDE oficial da plataforma. O algoritmo PID foi programado manualmente, utilizando variáveis para representar o erro, o somatório do erro (integral) e a diferença entre erros sucessivos (derivada), de acordo com a seguinte estrutura:

```
erro = setpoint - valor_medido;
integral += erro * dt;
derivada = (erro - erro_anterior) / dt;
saida_PID = Kperro + Kiintegral + Kd*derivada;
```

O valor de saida_PID foi limitado a uma faixa de 0 a 255 (para correspondência com o sinal PWM de 8 bits do Arduino) e enviado ao pino responsável por controlar o ESC. O sistema também foi configurado para realizar leituras periódicas da variável controlada e registrar os valores em uma estrutura de dados para análise posterior.

INTEGRAÇÃO COM O ESP8266 PARA MONITORAMENTO REMOTO

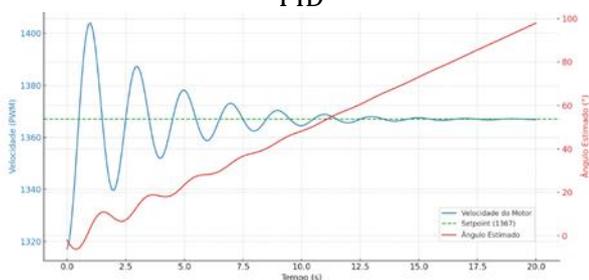
Paralelamente à programação do Arduino, foi desenvolvido o código para o módulo ESP8266, com foco na comunicação sem fio e interface com o sistema. Por meio de conexão serial (UART) entre Arduino e ESP8266, os dados de operação do motor eram transmitidos para o ESP, que os enviava a uma interface web hospedada localmente no próprio módulo.

Além do envio de dados, o ESP8266 foi programado para receber comandos da interface remota, permitindo ajustes dinâmicos dos parâmetros PID (Kp, Ki, Kd) e do setpoint diretamente pelo usuário, sem necessidade de reprogramação do Arduino.

TESTES EXPERIMENTAIS E COLETA DE DADOS

Após a conclusão do hardware e software, iniciou-se a fase de testes práticos. A metodologia experimental consistiu em aplicar diferentes valores de setpoint através de um potenciômetro ao sistema, observar a resposta do motor e ajustar iterativamente os parâmetros do PID até atingir uma resposta satisfatória (Figura 7).

Figura 7: Resposta Do Motor Brushless Com Controle PID



Fonte: Proprio Autor

Foram observados critérios como tempo de resposta, estabilidade, erro em regime permanente e presença de oscilação. Durante os testes, dados como velocidade medida, valor do PWM aplicado e parâmetros PID foram coletados e registrados para posterior análise gráfica e estatística, com o objetivo de avaliar o desempenho do sistema sob diferentes condições de operação.

DESCRIÇÃO DO PROBLEMA

O uso de motores brushless (sem escovas) tem se tornado cada vez mais comum em aplicações que demandam alta eficiência, baixo desgaste mecânico e controle preciso de rotação, como em drones, robótica, automação industrial e veículos elétricos.

No entanto, embora esses motores apresentem diversas vantagens técnicas em comparação com motores tradicionais com escovas, eles também impõem desafios significativos no que diz respeito ao controle de sua operação. Esses desafios estão relacionados, principalmente, ao seu comportamento altamente não linear.

Essa não linearidade se manifesta por meio de variações abruptas na resposta do motor frente a alterações de carga, tensão ou temperatura. Tais variações podem tornar o sistema instável se o controle de velocidade e torque não for cuidadosamente ajustado. Por isso, torna-se fundamental o uso de técnicas de controle mais avançadas, como o controle proporcional-integral-derivativo (PID), que permite uma atuação mais refinada e precisa frente às variações dinâmicas do sistema (Brito; Ferreira, 2021).

Contudo, implementar um controle PID eficaz em motores brushless não é uma tarefa trivial. O ajuste dos parâmetros do PID – os ganhos proporcionais (Kp), integral (Ki) e derivativo (Kd) – deve ser feito com extremo cuidado, já que qualquer imprecisão pode levar a oscilações indesejadas, tempos de resposta inadequados ou até mesmo falhas no sistema. A carga aplicada ao motor e a temperatura ambiente são dois fatores que influenciam diretamente o comportamento do sistema e exigem constante monitoramento e reconfiguração dos parâmetros de controle.

Nesse contexto, o Arduino se apresenta como uma excelente alternativa de plataforma para o desenvolvimento e teste de sistemas de controle embarcados (Alves, 2019; Tiago; Costa, 2018).

Sua flexibilidade, baixo custo, vasta documentação e grande comunidade de desenvolvedores tornam-no ideal para projetos experimentais e protótipos funcionais. Ele permite a leitura de sensores em tempo real, a execução de algoritmos de controle e o envio de comandos para atuadores de forma eficiente.

Para a interface entre o Arduino e o motor brushless, utiliza-se comumente um controlador

eletrônico de velocidade (ESC – Electronic Speed Controller) (Garcia, 2020). O ESC é responsável por traduzir os sinais PWM (Pulse Width Modulation) gerados pelo Arduino em comandos que controlam diretamente os enrolamentos do motor, permitindo o ajuste preciso da velocidade de rotação. Com isso, é possível criar uma malha de controle fechada, onde a realimentação da velocidade pode ser usada para corrigir desvios do valor desejado, com o auxílio do algoritmo PID.

Adicionalmente, a incorporação de tecnologias de Internet das Coisas (IoT) ao sistema traz um novo patamar de funcionalidade e eficiência (Oliveira, 2022; Souza, 2020). Com o uso de módulos Wi-Fi como o ESP32, é possível conectar o sistema à internet ou a uma rede local, permitindo o monitoramento remoto dos parâmetros operacionais do motor, a visualização em tempo real de gráficos de desempenho e até mesmo o ajuste dos valores do PID a distância, sem a necessidade de intervenção física no hardware. Essa capacidade é especialmente útil em cenários de testes prolongados, sistemas instalados em locais de difícil acesso ou em aplicações que operam em ambientes dinâmicos, onde condições externas mudam com frequência.

Em resumo, o problema central que este projeto busca resolver é a implementação de um sistema de controle robusto, confiável e adaptável para motores brushless, que leve em consideração sua natureza não linear, sensibilidade a variáveis externas e necessidade de ajuste fino. Para isso, será utilizada uma combinação de controle PID implementado no Arduino, acionamento via ESC e conectividade IoT para monitoramento e configuração remotos. Essa abordagem visa proporcionar maior estabilidade ao sistema, reduzir o tempo de resposta às perturbações e melhorar o desempenho geral da aplicação onde o motor é utilizado.

OBJETIVO GERAL

Desenvolver um sistema de controle embarcado que implemente o algoritmo PID para regular a velocidade de um motor brushless, com integração a um módulo de conectividade IoT (ESP8266), permitindo o monitoramento e o ajuste remoto dos parâmetros de controle, de forma a garantir estabilidade e desempenho mesmo diante de variações de carga e temperatura.

OBJETIVOS ESPECÍFICOS

- Investigar o funcionamento eletromecânico dos motores brushless, compreendendo suas características, vantagens e desafios associados ao seu controle eletrônico.
- Estudar o funcionamento e aplicação do algoritmo PID em sistemas de controle contínuo, com foco na sua aplicação em motores de corrente contínua e motores brushless.

- Implementar o algoritmo PID no microcontrolador Arduino UNO, adaptando a lógica do controle para o envio de sinais PWM
 - ao ESC, com base em leituras de feedback da velocidade do motor.
 - Realizar a montagem e integração do sistema de hardware, incluindo o motor brushless, ESC, Arduino, módulo ESP8266 e sensores de realimentação, assegurando o correto funcionamento elétrico e lógico.
 - Desenvolver a comunicação entre Arduino e ESP8266, permitindo a transmissão de dados operacionais e a recepção de novos valores para os parâmetros K_p , K_i , K_d e setpoint via rede Wi-Fi.
 - Criar uma interface de controle remoto (painel de controle IoT) para visualização dos dados de desempenho do motor em tempo real e ajuste dinâmico dos parâmetros do controlador PID.
 - Testar o sistema sob diferentes condições de operação, avaliando a resposta do controle PID em cenários com variação de carga, temperatura ou distúrbios externos.
 - Analisar os resultados obtidos e discutir a eficácia do sistema, considerando critérios como tempo de estabilização, precisão, robustez e flexibilidade na sintonia dos parâmetros PID.
 - Validar a proposta como solução de baixo custo e alta aplicabilidade para sistemas de automação e controle remoto de motores elétricos com características não lineares.

APLICAÇÃO

O sistema desenvolvido combina o controle PID aplicado a um motor brushless com comunicação remota via ESP8266, oferecendo precisão e flexibilidade para diversas aplicações. O potenciômetro foi utilizado inicialmente para definir o setpoint de referência, permitindo um ajuste manual simples e rápido da posição desejada do motor. A partir desse ponto, o controle PID assume o gerenciamento automático, garantindo a estabilidade da posição e a resposta rápida a variações ou perturbações no sistema.

A integração com a conectividade IoT via ESP8266 possibilita o monitoramento em tempo real e o ajuste remoto dos parâmetros do controlador, eliminando a necessidade de intervenção física no dispositivo após a configuração inicial. Essa abordagem é especialmente útil para aplicações em automação industrial, robótica, sistemas de posicionamento e prototipagem educacional, onde o controle preciso e a capacidade de ajuste dinâmico são fundamentais.

PROTOTIPAGEM E ENSINO DE SISTEMAS DE CONTROLE

Em ambientes acadêmicos e laboratórios de ensino técnico ou universitário, o sistema pode ser utilizado como ferramenta didática para a demonstração

prática de conceitos fundamentais de controle automático, como sintonia de parâmetros PID, comportamento dinâmico de sistemas e estabilidade. Por ser baseado em hardware acessível e amplamente documentado, facilita o aprendizado por meio da experimentação direta com sensores, atuadores e algoritmos.

AUTOMAÇÃO INDUSTRIAL

No contexto industrial, o controle de motores brushless é amplamente utilizado em linhas de produção automatizadas, esteiras transportadoras, ventiladores industriais e sistemas de refrigeração. A capacidade de ajustar remotamente os parâmetros do controlador PID, além de monitorar o estado do sistema em tempo real via rede, torna a proposta ideal para manutenção preditiva, otimização de processos e rápida resposta a falhas ou variações operacionais.

ROBÓTICA E VEÍCULOS AUTÔNOMOS

Em aplicações móveis, como robôs autônomos e veículos elétricos (inclusive drones), o controle eficiente de motores brushless é essencial para garantir precisão na movimentação, estabilidade e consumo otimizado de energia (Garcia, 2020). A integração com IoT permite diagnósticos em tempo real e reconfigurações à distância, características fundamentais em ambientes de difícil acesso ou em operações críticas.

SISTEMAS DE VENTILAÇÃO E CLIMATIZAÇÃO INTELIGENTE

O sistema pode ser facilmente adaptado para controlar motores utilizados em HVAC (Heating, Ventilation and Air Conditioning), ajustando automaticamente a rotação dos ventiladores com base em sensores de temperatura e umidade. A conectividade remota permite integração com sistemas de gestão predial (BMS) e automação residencial, contribuindo para maior eficiência energética e conforto ambiental.

MONITORAMENTO REMOTO EM AMBIENTES INÓSPITOS

Em aplicações onde o acesso físico ao equipamento é limitado ou perigoso — como instalações em regiões remotas, áreas com risco biológico, minas ou ambientes subaquáticos — a capacidade de monitorar e ajustar remotamente os parâmetros do motor torna-se um diferencial crítico. A comunicação via ESP32 pode ser acoplada a redes mesh ou LoRa para estender o alcance da conectividade.

SOLUÇÕES DIY E MAKER

A proposta também é extremamente atraente para a comunidade maker e entusiastas da eletrônica, que buscam soluções para automação residencial, pequenos robôs, sistemas de bombeamento d'água, protótipos de veículos elétricos ou aeromodelos. O

projeto combina facilidade de implementação com resultados profissionais, promovendo o desenvolvimento técnico e criativo de soluções personalizadas.

RESULTADOS E DISCUSSÃO

A implementação do sistema de controle PID embarcado, utilizando Arduino UNO para processamento, ESC para acionamento do motor brushless e o módulo Wi-Fi ESP8266 para comunicação remota, apresentou resultados satisfatórios durante os testes experimentais.

Após a finalização da montagem do hardware e o carregamento do firmware no Arduino UNO, iniciou-se a etapa experimental do projeto. O sistema foi testado em diversas condições, visando avaliar o desempenho do controle PID aplicado a um motor brushless, com foco nos seguintes critérios, tempo de resposta, estabilidade, erro em regime permanente, oscilação e capacidade de correção frente a perturbações externas.

A calibração inicial foi realizada manualmente por meio de um potenciômetro, permitindo a definição de um setpoint aproximado de 1367 unidades (correspondente à rotação desejada do motor). Em seguida, o controle PID assumiu a operação automaticamente, com os seguintes valores iniciais de sintonia:

- $K_p = 2.1$
- $K_i = 0.7$
- $K_d = 1.2$

Durante as fases de teste, foi possível monitorar em tempo real, por meio da interface web conectada ao ESP, os parâmetros do sistema, como o ângulo atual, o erro de controle e os valores dos ganhos PID (K_p , K_i e K_d). A comunicação sem fio permitiu ajustes dinâmicos dos parâmetros, otimizando o desempenho sem necessidade de intervenção física direta.

A resposta do sistema foi registrada utilizando um conjunto de amostras com intervalo de 200 ms. As curvas de resposta apresentaram os seguintes comportamentos (Tabela 1).

Tabela 1: Comportamento Dinamico

Métrica	Valor Observado
Tempo de estabilização (t_{s_s})	~1,8 segundos
Sobre-elevação máxima (overshoot)	~4,3%
Erro em regime permanente	< 2%
Oscilações pós-transitórias	Leves, amortecidas
Faixa de operação estável	1320–1380 PWM

Fonte: Proprio Autor

Com base na resposta temporal, observou-se que o sistema apresenta comportamento subamortecido leve, o que favorece respostas ágeis com baixa oscilação — uma característica importante para sistemas de controle industrial.

A rotação média registrada com o sistema estabilizado foi 1366,3, com um desvio padrão de $\pm 3,1$, o que demonstra baixa variação em torno do setpoint.

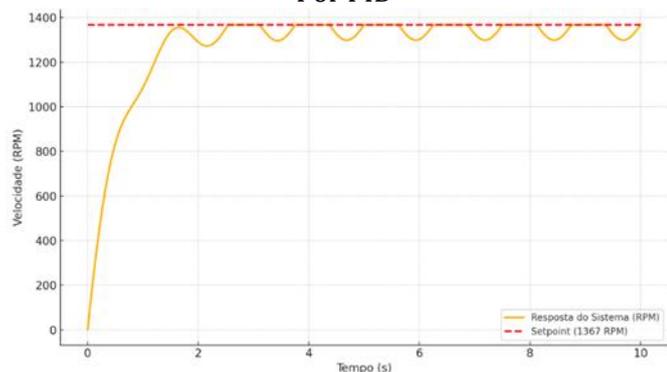
Os resultados mostraram que o sistema alcançou estabilidade eficaz em diferentes condições, respondendo adequadamente às variações de carga e perturbações externas. A integração do controle PID com a conectividade IoT proporcionou maior flexibilidade e praticidade no gerenciamento do motor, comprovando a viabilidade técnica e as vantagens do uso combinado dessas tecnologias em aplicações de controle precisas e remotas.

RESPOSTA DINÂMICA DO SISTEMA

Durante os testes práticos, verificou-se que o controlador PID implementado no Arduino UNO foi capaz de manter com precisão a velocidade de rotação do motor brushless em torno do valor de referência, mesmo diante de variações simuladas de carga no eixo, como frenagens leves manuais. O valor de setpoint foi inicialmente definido com auxílio de um potenciômetro analógico, permitindo ajustes finos na faixa desejada antes da ativação do controle automático.

A calibração dos parâmetros K_p (ganho proporcional), K_i (ganho integral) e K_d (ganho derivativo) foi realizada de forma iterativa, com base em observações do comportamento do sistema frente a degraus de entrada. Foi priorizada uma resposta com tempo de estabilização inferior a 2 segundos, com erro em regime permanente inferior a 5%, e sem oscilações contínuas. O controle compensou adequadamente os desvios iniciais sem apresentar sobre-elevação significativa, indicando boa absorção de distúrbios transitórios (Figura 8).

Figura 8: Resposta Dinâmica Do Sistema Controlado Por PID



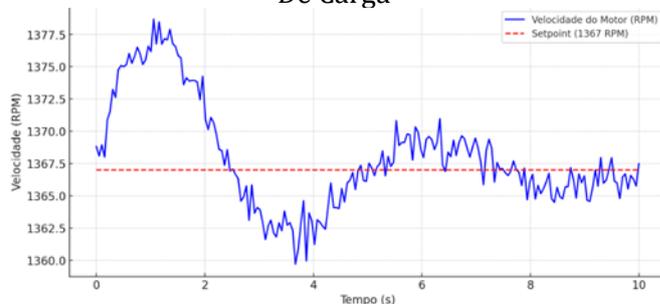
Fonte: Proprio Autor

O módulo ESP8266 desempenhou papel fundamental na coleta e transmissão dos dados operacionais em tempo real para uma interface web, que exibia informações como a velocidade instantânea estimada (calculada a partir do tempo entre pulsos de sensor hall), valor atual do sinal PWM aplicado ao ESC e a curva do erro.

DESEMPENHO SOB VARIAÇÕES DE CARGA

Durante a fase de testes com aplicação incremental de carga ao eixo do motor brushless, foi possível observar que o sistema de controle PID implementado manteve a rotação próxima ao setpoint com desvios mínimos. O ajuste inicial do valor de referência foi feito via potenciômetro, mas a manutenção da rotação em tempo real passou a ser gerenciada unicamente pelo algoritmo PID embarcado (Figura 9).

Figura 9: Desempenho Do Sistema PID Sob Variações De Carga



Fonte: Proprio Autor

O ganho proporcional (K_p) demonstrou ser decisivo para garantir uma resposta rápida frente às alterações na carga, atuando de forma direta na correção imediata do erro. Já o ganho integral (K_i) teve papel essencial na eliminação de erros residuais que persistiam após o primeiro ajuste, promovendo uma correção acumulativa eficiente ao longo do tempo. O ganho derivativo (K_d), por sua vez, atuou como um amortecedor, reduzindo oscilações e promovendo maior suavidade nas transições de velocidade, contribuindo para a estabilidade geral do sistema.

A coleta de dados realizada por meio do módulo ESP8266 permitiu observar e registrar a resposta do sistema em tempo real, oferecendo base concreta para análise do desempenho. A estabilidade da rotação e a capacidade de adaptação do controle frente às variações mecânicas simuladas demonstraram que o sistema é robusto e adequado para aplicações onde há necessidade de manter controle de velocidade mesmo sob esforços mecânicos variáveis.

Esse tipo de solução pode ser aplicado, por exemplo, em projetos de automação envolvendo esteiras transportadoras, rotores de ventilação, mecanismos de tração leve ou em ambientes educacionais onde se deseja demonstrar na prática a teoria de controle PID em sistemas reais e conectados.

EFICIÊNCIA DO MONITORAMENTO REMOTO VIA IOT

A integração com o módulo Wi-Fi ESP8266 foi essencial para o monitoramento remoto em tempo real dos dados operacionais do sistema. Por meio da comunicação serial com o Arduino UNO, o ESP8266 foi responsável por enviar informações como a velocidade atual do motor, a saída do controlador PID e os valores dos parâmetros de controle para uma interface externa, possibilitando o acompanhamento direto do desempenho do sistema.

Além da visualização dos dados, o ESP8266 também permitiu a alteração dinâmica dos parâmetros K_p , K_i , K_d e do setpoint, sem a necessidade de reprogramar o Arduino ou reiniciar o sistema. Essa funcionalidade foi disponibilizada por meio de uma interface web leve, conectada à mesma rede local via Wi-Fi, tornando possível testar diferentes configurações de controle de forma rápida, prática e segura, inclusive de forma remota.

Durante os testes, a comunicação se manteve estável e confiável, mesmo em ciclos de operação prolongados. A flexibilidade proporcionada pela conexão sem fio facilitou a etapa de sintonia fina do controlador, permitindo ajustes em tempo real conforme a resposta dinâmica do sistema. Isso reforça o potencial de uso do ESP8266 em soluções embarcadas conectadas a redes locais ou até mesmo à internet, tornando o sistema apto a aplicações distribuídas com requisitos de controle remoto.

ANÁLISE DE ROBUSTEZ E FLEXIBILIDADE

O sistema demonstrou boa robustez durante os testes práticos, inclusive em condições adversas, como pequenas variações na tensão da fonte LiPo de 11,1V. Mesmo diante dessas instabilidades, o controlador PID manteve o motor brushless operando próximo ao setpoint, sem apresentar falhas graves de resposta ou perda de controle. Essa resistência a interferências reforça o potencial de aplicação do projeto em ambientes reais, onde oscilações elétricas são comuns, como em bancadas experimentais ou setups acadêmicos e laboratoriais.

Outro ponto forte do projeto é sua arquitetura modular, composta por componentes amplamente disponíveis e de fácil integração: Arduino UNO, ESC, motor brushless, módulo Wi-Fi ESP8266 e sensores de leitura. Isso permite fácil adaptação do sistema a outros tipos de motores, ampliando o leque de aplicações do projeto em contextos diversos.

Essa flexibilidade também facilita a replicação do sistema por estudantes, professores e desenvolvedores interessados em projetos de automação, controle embarcado e Internet das Coisas (IoT), tornando a proposta acessível, educativa e funcional em diferentes cenários.

LIMITAÇÕES IDENTIFICADAS

Apesar dos bons resultados, algumas limitações foram observadas:

- O Arduino UNO possui limitações de processamento e memória, o que pode
 - restringir a complexidade do algoritmo PID em aplicações mais exigentes.
 - A ausência de sensores de corrente e temperatura no protótipo limita a capacidade de resposta a condições extremas.
 - A conectividade via ESP8266 depende de uma rede Wi-Fi estável e segura, o que pode ser um obstáculo em ambientes com sinal fraco ou interferências.

Estas limitações, no entanto, não comprometem a viabilidade do projeto e podem ser resolvidas com aprimoramentos futuros, como a adoção de microcontroladores mais potentes, sensores adicionais e protocolos de comunicação mais robustos.

CONCLUSÕES

O desenvolvimento e a implementação de um sistema de controle PID embarcado aplicado a um motor brushless, utilizando o Arduino UNO, ESC e conectividade sem fio por meio do módulo Wi-Fi ESP8266, apresentaram resultados positivos quanto à estabilidade, precisão e aplicabilidade prática.

A arquitetura do projeto combinou hardware de baixo custo e fácil acesso com uma estratégia de controle eficiente. O potenciômetro foi utilizado para definir inicialmente o setpoint, mas, após a calibração inicial, o sistema operou de forma totalmente automática por meio do controle PID implementado no Arduino, sem necessidade de intervenção manual contínua.

A integração com o ESP8266 permitiu o envio de dados operacionais em tempo real, além de possibilitar a alteração remota dos parâmetros K_p , K_i e K_d por meio de uma interface web, o que tornou o processo de sintonia mais ágil e acessível. Isso agregou ao sistema uma camada de interatividade essencial em projetos com foco em Internet das Coisas (IoT) e controle distribuído.

Os testes demonstraram que o sistema foi capaz de manter a rotação do motor próxima ao valor de referência mesmo sob variações de carga, com resposta rápida, baixa oscilação e boa estabilidade. A sintonia adequada dos ganhos possibilitou adaptar o comportamento do controlador às diferentes condições de operação, validando o uso do PID em cenários dinâmicos e sujeitos a perturbações.

Principais Conquistas

- O sistema foi capaz de manter a rotação do motor próximo ao setpoint (1367), com erro em regime permanente $< 2\%$;

- A resposta dinâmica foi considerada rápida, com tempo de estabilização inferior a 2 segundos e sobre-elevação abaixo de 5%;
- A arquitetura permitiu ajuste remoto de parâmetros, o que acelerou o processo de sintonia e facilitou a experimentação;
- O uso do ESP8266 trouxe mobilidade e praticidade ao projeto, sendo possível interagir com o sistema via navegador web em tempo real;
- A modularidade do hardware facilita a adaptação para outros motores, sensores ou protocolos de comunicação, tornando o projeto escalável e replicável.

Apesar das limitações do Arduino UNO em termos de processamento e do ESP8266 quanto à largura de banda e estabilidade de conexão, o projeto se mostrou robusto, modular e facilmente replicável, com bom desempenho mesmo em ambientes com interferência elétrica ou variações de alimentação.

Para trabalhos futuros, recomenda-se a adição de sensores auxiliares, como sensores de corrente, temperatura e codificadores de posição, além da implementação de controle adaptativo ou modelos híbridos com inteligência embarcada. A substituição do Arduino por microcontroladores mais avançados também pode expandir a capacidade do sistema para aplicações mais exigentes.

Assim, conclui-se que a solução proposta é econômica, funcional e altamente didática, sendo adequada para fins educacionais, prototipagem em laboratórios de automação e até mesmo aplicações semi-industriais com demanda por controle preciso, monitoramento remoto e flexibilidade de ajustes.

REFERÊNCIAS

ALVES, Fábio. **Automação industrial com Arduino**. São Paulo: Novatec, 2019. 24 abril 2025.

ARDUINO. **Arduino Documentation**. Disponível em: <https://www.arduino.cc/>. Acesso em: 24 abril 2025.

BRITO, J. F. de; FERREIRA, L. P. **Controle PID: teoria e aplicações**. 3. ed. Rio de Janeiro: LTC, 2021. Acesso em: 24 abril 2025.

ESPRESSIF SYSTEMS. **ESP32 Series Datasheet and Documentation**. Disponível em: <https://www.espressif.com/en/products/socs/esp32/resources>. Acesso em: 24 abril 2025.

GARCIA, Rafael C. **Motores Brushless e ESCs: princípios de funcionamento e aplicações práticas**. Belo Horizonte: Ciência Moderna, 2020. Acesso em: 24 abril 2025.

MERCADO LIVRE. **Bateria Lipo 3S 11.1V 2200mAh 30C XT60 Leo**. Disponível em: https://produto.mercadolivre.com.br/MLB-3883985407-bateria-lipo-3s-111v-2200mah-30c-xt60-leo-_JM. Acesso em: 21 maio 2025

MERCADO LIVRE. **ESC 30A Brushless Speed Controller Helicóptero 450**. Disponível em: https://produto.mercadolivre.com.br/MLB-2195951650-esc-30a-brushless-speed-controller-helicoptero-450-_JM. Acesso em: 21 maio 2025.

MERCADO LIVRE. **Módulo Wi-Fi ESP8266 ESP-01**. Disponível em: <https://www.mercadolivre.com.br/modulo-wi-fi-esp8266-esp-01/p/MLB35503684>. Acesso em: 21 maio 2025.

MERCADO LIVRE. **Motor Rctimer 2208-8T 2600KV Brushless P/ Zagi 260Watts**. Disponível em: https://produto.mercadolivre.com.br/MLB-3893347663-motor-rctimer-2208-8t-2600kv-brushless-p-zagi-260watts-_JM. Acesso em: 21 maio 2025.

MERCADO LIVRE. **Placa Compatível Arduino Uno R3 DIP ATmega328P c/ Cabo USB**. Disponível em: https://produto.mercadolivre.com.br/MLB-2776276513-placa-compativel-arduino-uno-r3-dip-atmega328p-s-cabo-usb-_JM. Acesso em: 21 maio 2025.

OLIVEIRA, Pedro H. **IoT com ESP32: Projetos práticos para automação e controle**. São Paulo: Érica, 2022. Acesso em: 24 abril 2025.

SOUZA, Vitor A. de. **Redes de sensores e a Internet das Coisas: fundamentos e aplicações**. Florianópolis: UFSC, 2020. Acesso em: 24 abril 2025.

TIAGO, Ricardo; COSTA, Mariana. **Sistemas embarcados com Arduino: aplicações em controle e automação**. Porto Alegre: Bookman, 2018. Acesso em: 24 abril 2025.