

# PROPOSTA DE PLATAFORMA DIDÁTICA PARA IDENTIFICAÇÃO E MODELAGEM DE MOTORES DE CORRENTE CONTÍNUA USANDO ARDUÍNO PARA AQUISIÇÃO DOS DADOS

## PROPUESTA DE PLATAFORMA DIDÁCTICA PARA LA IDENTIFICACIÓN Y MODELADO DE MOTORES DE CORRIENTE DIRECTA CON ARDUINO PARA LA ADQUISICIÓN DE DATOS

### PROPOSAL OF TEACHING PLATFORM FOR IDENTIFICATION AND MODELING OF DIRECT CURRENT MOTORS USING ARDUINO FOR DATA ACQUISITION

**Wallace Pereira Neves dos Reis\***  
wallace.reis@ifrj.edu.br

**Giselle Elias Couto\*\***  
giselle.couto@cefet-rj.br

**Orides Morandin Junior\*\*\***  
orides@ufscar.br

\*Instituto Federal de Educação, Ciência e Tecnologia do Rio de Janeiro, Volta Redonda/RJ, Brasil

\*\*Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, Itaguaí/RJ, Brasil

\*\*\*Universidade Federal de São Carlos, São Carlos/SP, Brasil

---

#### Resumo

A modelagem e identificação de sistemas dinâmicos é uma etapa importante para o projeto de controladores e projeto de produtos. A modelagem de sistemas dinâmicos pode ser realizada por meio da interpretação dos fenômenos físicos envolvidos, mas também pela medição da saída do sistema a partir da aplicação de sinais de entrada conhecidos. Assim, esse trabalho propõe uma plataforma didática com sensores e circuito de acionamento para a medição das variáveis de motores de corrente contínua, possibilitando a identificação e modelagem do sistema dinâmico. A proposta mede a tensão de entrada, proveniente de um circuito com acionamento por modulação de largura de pulso (PWM), a corrente e a velocidade do motor. A estrutura também permite o ajuste da carga acoplada ao eixo do motor, para medição do comportamento do sistema sob condições de carga. A plataforma é construída com dispositivos facilmente encontrados no mercado e utiliza um Arduino como sistema principal de controle do motor e aquisição de dados. O trabalho apresenta o método utilizado para aquisição dos dados, medidos em três diferentes condições. Os resultados mostram o potencial da proposta para utilização na identificação de sistemas dinâmicos, mas sua utilização também pode ser estendida para outros experimentos e atividades de ensino e pesquisa.

**PALAVRAS CHAVE:** Modelo Dinâmico, Identificação de Sistemas, Aquisição de dados, Robótica.

#### Resumen

Modelar e identificar sistemas dinâmicos es un paso esencial en el diseño de controladores y diseño de productos. Un modelo de sistema dinámico incluye la interpretación de los fenómenos físicos que gobiernan el sistema o la medición de los resultados de salida del sistema de la aplicación de señales de entrada conocidas. Así, este trabajo propone una plataforma didáctica con sensores y un circuito de accionamiento para medir las variables de los motores de CC para la identificación y modelado del sistema dinámico. La propuesta mide el voltaje de entrada de un circuito impulsado por modulación de ancho de pulso (PWM), la corriente y la velocidad del motor. La estructura también permite ajustar la carga acoplada al eje del motor para medir el comportamiento del sistema en condiciones de carga. La plataforma está construida con dispositivos que se encuentran fácilmente en el mercado y utiliza un Arduino como sistema principal de control de motor y adquisición de datos. El trabajo presenta el método utilizado para la adquisición de datos, medidos bajo tres condiciones diferentes. Los resultados muestran el potencial de la propuesta para ser utilizada en el proceso de identificación de sistemas

dinâmicos y ser utilizada en otros experimentos y otras actividades de docencia e investigación.

**PALABRAS CLAVE:** Modelo Dinámico, Identificación de Sistemas, Adquisición de Datos, Robótica.

### **Abstract**

Modeling and identifying dynamic systems is an essential step in controller design and product design. A dynamic system modeling includes interpreting the physical phenomena that govern the system or the measurement of the system output results from applying known input signals. Thus, this work proposes a didactic platform with sensors and a drive circuit for measuring the variables of DC motors for the identification and modeling of the dynamic system. The proposal measures the input voltage from a circuit driven by pulse width modulation (PWM), the current, and the motor speed. The structure also allows adjusting the load coupled to the motor shaft to measure the system's behavior under load conditions. The platform is built with devices easily found on the market and uses an Arduino as the primary motor control and data acquisition system. The work presents the method used for data acquisition, measured under three different conditions. The results show the proposal's potential to be used in the dynamic systems' identification process and be used in other experiments and other teaching and research activities.

**KEYWORDS:** Dynamic Model, Systems Identification, Data Acquisition, Robotics.

---

## **1. INTRODUÇÃO**

Em robótica móvel, modelos cinemáticos são encontrados em um grande número de trabalhos no domínio de problema, especialmente no caso de veículos autoguiados, do Inglês, *Automated Guided Vehicles* (AGVs) (AUTOR1 e AUTOR3, ANO). Modelos cinemáticos, entretanto, descrevem o movimento do veículo considerando suas restrições de movimentos, mas sem considerar a aceleração necessária e consequente para a variação do movimento. Dessa forma, controles puramente cinemáticos apresentam baixo desempenho em situações de variação de velocidade e em condições diferentes de carga, por exemplo.

Modelos dinâmicos são mais complexos que modelos puramente cinemáticos, pois a quantidade de fenômenos envolvidos aumenta substancialmente e o comportamento desses fenômenos nem sempre são conhecidos, além de ser difícil modelar a interação entre elementos do veículo. Entretanto, um controle dinâmico consegue atuar em condições que o modelo cinemático não o faz. Descrever o modelo dinâmico de um robô móvel parte da descrição de seus atuadores, em grande parte, motores de corrente contínua (CC). Assim, esse trabalho tem o objetivo de propor e descrever uma plataforma didática para testes e medições de variáveis de um motor CC de forma a possibilitar o uso de técnicas de identificação do sistema para a definição de seu modelo matemático.

A aquisição de dados é realizada por um Arduino Mega que lê as informações de um encoder rotativo incremental e um sensor de corrente por efeito hall, além de medir a tensão de saída do circuito de acionamento do motor por meio de dois circuitos de divisores de tensão com resistores. O motor é acionado por um circuito de potência utilizando ponte H. A proposta permite a variação das condições de atuação do motor, a partir da atuação da ponte H pelo Arduino, usando diferentes tipos de sinal. Permite também a alteração das condições de carga acoplada no eixo, sendo apresentados ensaios com cargas de 1, 2 e 3 kg. Além do uso para modelagem e identificação de motores CC, a plataforma pode ser usada ainda para experimentos de controle de sistemas dinâmicos, estudos de aquisição e processamento de sinais digitais e aulas de programação em Arduino.

O presente artigo está organizado de acordo com as seguintes seções: na Seção 2 é apresentada a fundamentação teórica do trabalho e uma breve revisão da literatura com os principais trabalhos relacionados; na Seção 3 são indentificados os materiais e métodos utilizados na construção da plataforma e a estrutura de software utilizada para a aquisição dos dados e medição das variáveis de interesse; na Seção 4 são apresentados e discutidos os resultados obtidos em três diferentes cenários de testes e uma

aplicação da plataforma proposta é ilustrada com a identificação do sistema do motor a partir dos dados gerados em um dos experimentos; e, por fim, na Seção 5 são feitas as considerações finais do artigo.

## **2. FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS**

A modelagem matemática de um sistema dinâmico é conhecida como modelagem física ou caixa branca (AGUIRRE, 2004). O modelo extraído baseia-se em leis e princípios físicos. Entretanto, para a correta modelagem, todos os fenômenos e parâmetros devem ser previamente conhecidos. Para um sistema dinâmico complexo, como um robô móvel, nem todas as interações são conhecidas e podem ser modeladas, gerando simplificações.

No outro extremo da identificação de sistemas dinâmicos figura o método caixa preta, em que nenhum conhecimento do funcionamento do sistema está previamente disponível e seu modelo se baseia na análise da resposta do sistema a entradas conhecidas (AGUIRRE, 2004). Apesar de não demandar a quantidade de informações de uma modelagem caixa branca e, por isso, ser mais simples, essa modelagem pode levar a um modelo restrito do sistema devido às entradas, técnicas de modelagem e estrutura do modelo usadas.

O meio-termo é a chamada modelagem caixa cinza, que, segundo Aguirre (2004) busca combinar as vantagens dos procedimentos anteriores. Assim, são usados os conhecimentos prévios do sistema, como parâmetros conhecidos, o modelo relacionando alguns fenômenos envolvidos, e dados de entrada e saída obtidos em experimentos com o sistema. Os procedimentos de identificação caixa preta e caixa cinza utilizam métodos numéricos para obter os modelos.

Independentemente do modelo escolhido, uma etapa importante é a validação do mesmo. Essa etapa visa medir o nível de representação em relação ao sistema real. Assim, para Aguirre (2004), o modelo deve ser testado sem nenhum ajuste adicional em um conjunto de dados extraído de experimentos diferentes daqueles usados para a estimação dos parâmetros e sintonia do modelo.

Nelles (2020) e Tangirala (2015) defendem o uso de uma técnica chamada de validação cruzada para validação de modelos, usada principalmente quando o conjunto de dados não é grande o suficiente para que seja particionado em um conjunto para estimação de parâmetros, outro para validação e outro para testes do modelo. Nessa técnica, o conjunto de dados é dividido em  $X$  partes e então  $(X-1)$  partes são usadas para a estimação de parâmetros e a parte restante para a validação. Esse processo é repetido até que as  $X$  combinações possíveis sejam usadas.

Assim, para a utilização de métodos numéricos para a identificação de um sistema é necessário ter dados desse sistema e quanto mais dados em diferentes condições, melhor. Por isso, o uso de uma plataforma para aquisição dos dados do motor CC mostra-se essencial para submeter o sistema a diferentes condições e medir o seu comportamento. O uso de técnicas de identificação de sistemas para identificação de motores CC é relevante na literatura. Alguns exemplos são: Akgün *et al.* (2019), Naung *et al.* (2018), Tang *et al.* (2017), Mokhlis *et al.* (2019), Faieghi e Azimi (2010) e Kara e Eker (2004).

Além da importância no projeto do sistema de controle de robôs móveis, outro fator a ser considerado na modelagem do veículo e de seus componentes é a aplicação em simuladores. Autor1 e Autor3 (ANO) apontam para um grande número de pesquisas que propõem sistemas de controle para AGVs e apresentam a validação dos resultados por meio de simulações.

O uso de simuladores pode reduzir o consumo de tempo e de recursos no desenvolvimento de um projeto, além de possibilitar o estudo de situações que seriam impraticáveis com o veículo real, seja por espaço, seja pelo tempo necessário para o experimento (ECHEVERRIA *et al.*, 2011). Entretanto, para obtenção de resultados mais próximos do comportamento real do AGV, é importante considerar o modelo mais completo possível e, conseqüentemente, considerar o modelo dinâmico de seus atuadores.

## **3. MATERIAIS E MÉTODOS**

Essa seção descreve os materiais que compõem a plataforma proposta, suas conexões e os sensores

utilizados. Além disso, apresenta as especificações do motor CC ensaiado, que será utilizado como atuador de um AGV em fase de projeto. Em relação aos métodos, é apresentado o fluxograma do código embarcado no Arduino Mega responsável pela aquisição dos dados e os cálculos realizados para o tratamento dos dados e a efetiva medição das variáveis do motor.

### 3.1. Sensores e dispositivo de atuação do motor CC

A plataforma proposta conta com sensores para medição da tensão, velocidade e corrente do motor. A Tabela 1 descreve os sensores e suas principais especificações.

Tabela 1. Sensores utilizados na plataforma proposta.

Sensor	Grandeza Medida	Especificações
<b>Módulo baseado em sensor de efeito Hall - ACS712</b>	Corrente [Ampère]	Tensão de alimentação: 5,0 V Saída analógica: 66mV/A Faixa de medição: -30 a +30 A Tempo de resposta: 5 ms
<b>Encoder Rotativo Incremental OMRON E6B2-CWZ6C</b>	Velocidade [RPM]	Tensão de alimentação: 5 - 24V Pulsos por rotação: 1800 Tipo de saída: NPN Sinais de saída: A, B, Z
<b>Circuito divisor de tensão usando resistores</b>	Tensão [Volt]	Dois circuitos de divisores de tensão com três resistores de 100 k $\Omega$ , de forma a adequar o nível de tensão de entrada do motor ao nível de tensão de entrada analógica do Arduino. Variação de tensão: 0 V e 4 V Resolução: 4,88 mV

Fonte: Allegro Microsystems (S.d.) e OMRON (S.d.).

A Tabela 2 apresenta as especificações do circuito de acionamento e do motor utilizados na proposta.

Tabela 2. Especificações do circuito de acionamento e do motor CC utilizados nos ensaios.

Sensor	Especificações
<b>Ponte H VNH2SP30 Sparkfun Monster Moto Shield</b>	Tensão de alimentação: 5 a 16 V Corrente de pico: 30 A Corrente nominal: 14 A Frequência máxima: 20 kHz Desligamento automático por tensão abaixo ou acima do permitido e por temperatura elevada
<b>Motor CC - Argedo DC Gear Motor</b>	Tensão de alimentação: 12 V Velocidade a vazio: 470 RPM Velocidade nominal: 400 RPM Torque nominal: 5 kgf.cm Corrente nominal: 1,6 A Torque máximo: 17 kgf.cm Corrente máxima: 5 A Redução: 17:1

Fonte: Sparkfun (S.d.).

Um circuito contador rápido de quadratura baseado no circuito integrado (CI) LS7366R é usado para contagem dos pulsos do encoder. A comunicação com o Arduino é realizada por protocolo SPI (*Serial Peripheral Interface*). O circuito opera com tensões de alimentação de 3 a 5 V, possui contador programável e registradores de 32 bits e oscilador de 40 MHz, faz a leitura de fases A e B de encoders e do pino Z, também chamado de índice, que informa a ocorrência de uma revolução completa do encoder. Além disso, tem quatro diferentes modos configuráveis para leitura dos pulsos. A atuação do motor é realizada por meio de um circuito de potência que utiliza circuitos integrados de ponte H e saída à MOSFETs, possibilitando alto níveis de corrente de atuação.

### 3.2. Construção da Plataforma

A Figura 1 mostra o esquemático do circuito eletrônico, com a representação dos dispositivos utilizados e a ligação de cada item ao Arduino Mega, onde é possível observar os sensores de corrente, tensão e velocidade do motor, assim como as conexões da ponte H, circuito de potência utilizado para acionar o motor CC.

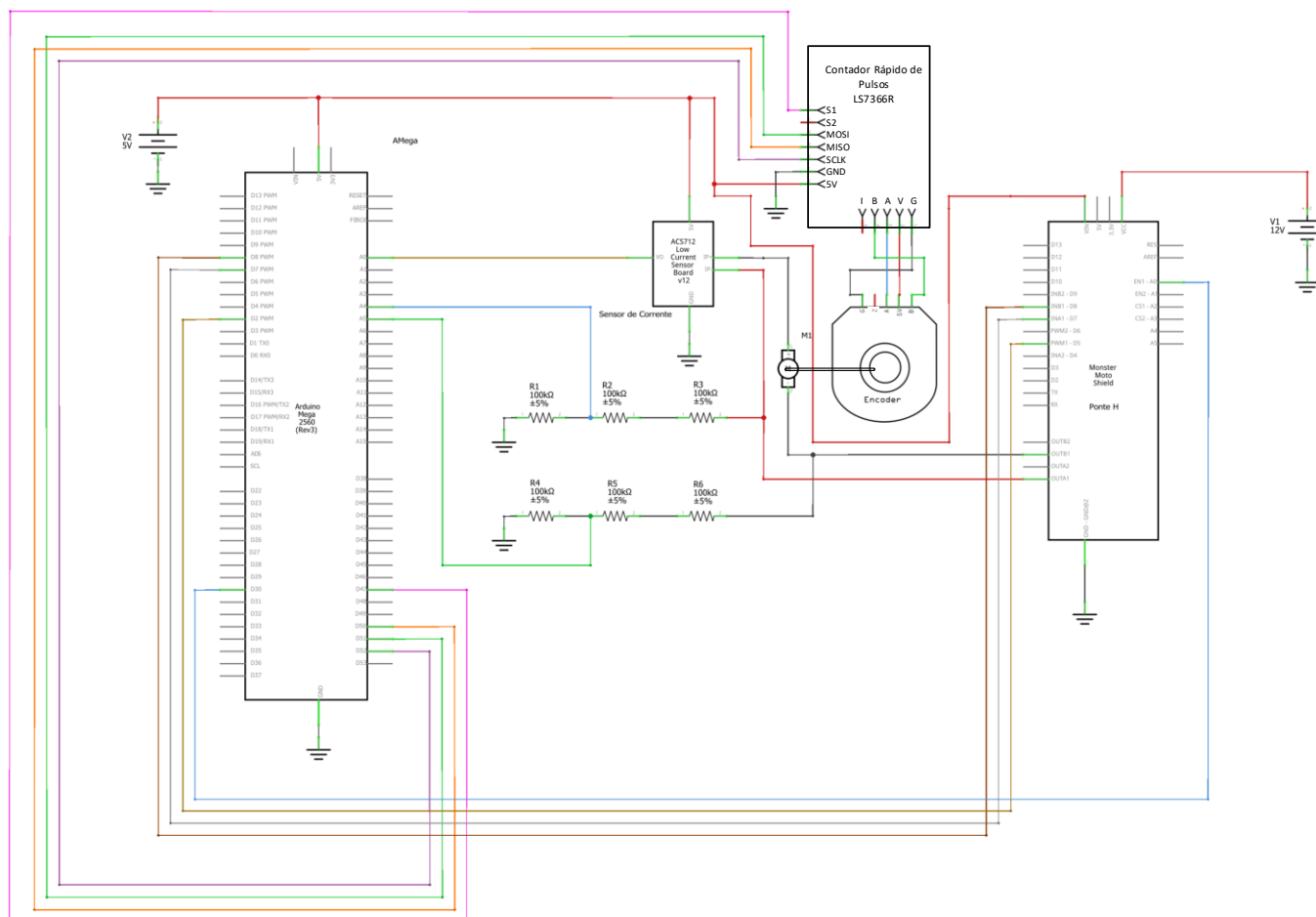
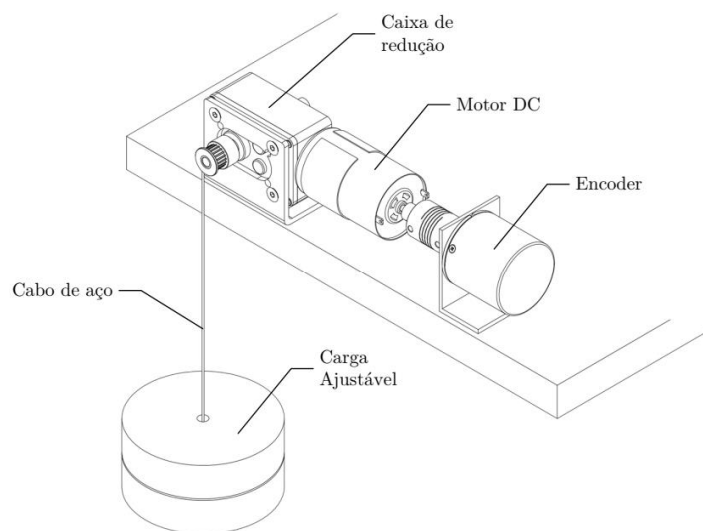
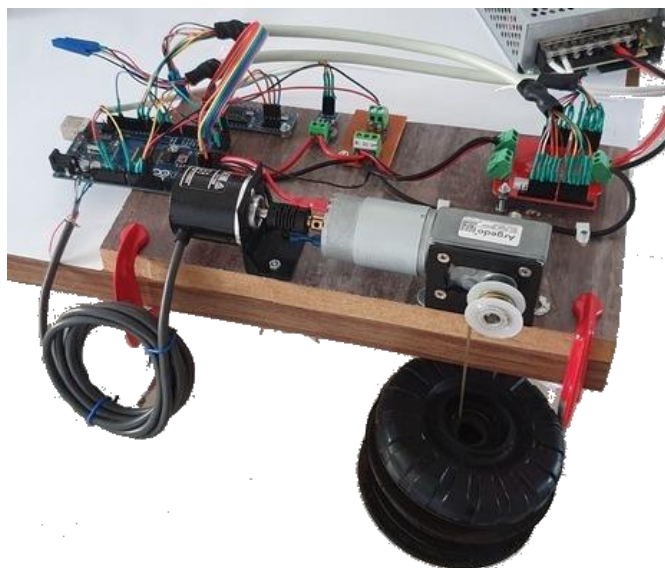


Figura 1. Esquemático do circuito eletrônico e conexão dos componentes da plataforma didática proposta. Fonte: Autoria própria (2021).

A construção física da plataforma é mostrada na Figura 2 (a) e a montagem real é mostrada na Figura 2 (b).



(a)



(b)

Figura 2. Ilustração da plataforma proposta. (a) Ilustração do aparato mecânico para o ensaio dos motores CC com o ajuste de carga para identificação do modelo do sistema. (b) Montagem real. Fonte: Autoria própria (2021).

### 3.3. Aquisição dos dados

O código para aquisição de dados foi desenvolvido no ambiente de desenvolvimento do Arduino, versão 1.8.15, e tem como dependências as bibliotecas SPI e ACS712, para comunicação via rede SPI e leitura do sensor de corrente, respectivamente. A Figura 3 mostra o fluxograma do código para aquisição dos dados. A primeira raia, denominada Declarações Iniciais, engloba a inclusão de bibliotecas, definição da numeração dos pinos utilizados pelos sensores e pelo circuito atuador do motor, configurações iniciais dos sensores de corrente e tensão e a declaração de variáveis utilizadas ao longo do código.

A seguir, na raia Rotina de Configuração, são realizadas as configurações necessárias para a execução da aquisição de dados, como configuração do modo dos pinos do Arduino, inicialização de sensores e do temporizador (timer) utilizado para contar o tempo de interrupção. Para esse trabalho, foi definida a interrupção por tempo. Dessa forma, o estouro do timer, ou seja, quando este atinge a contagem de 10 ms, é o que dispara a interrupção, que é a rotina responsável por executar a medição dos sensores e a atualização do comando para a ponte H.

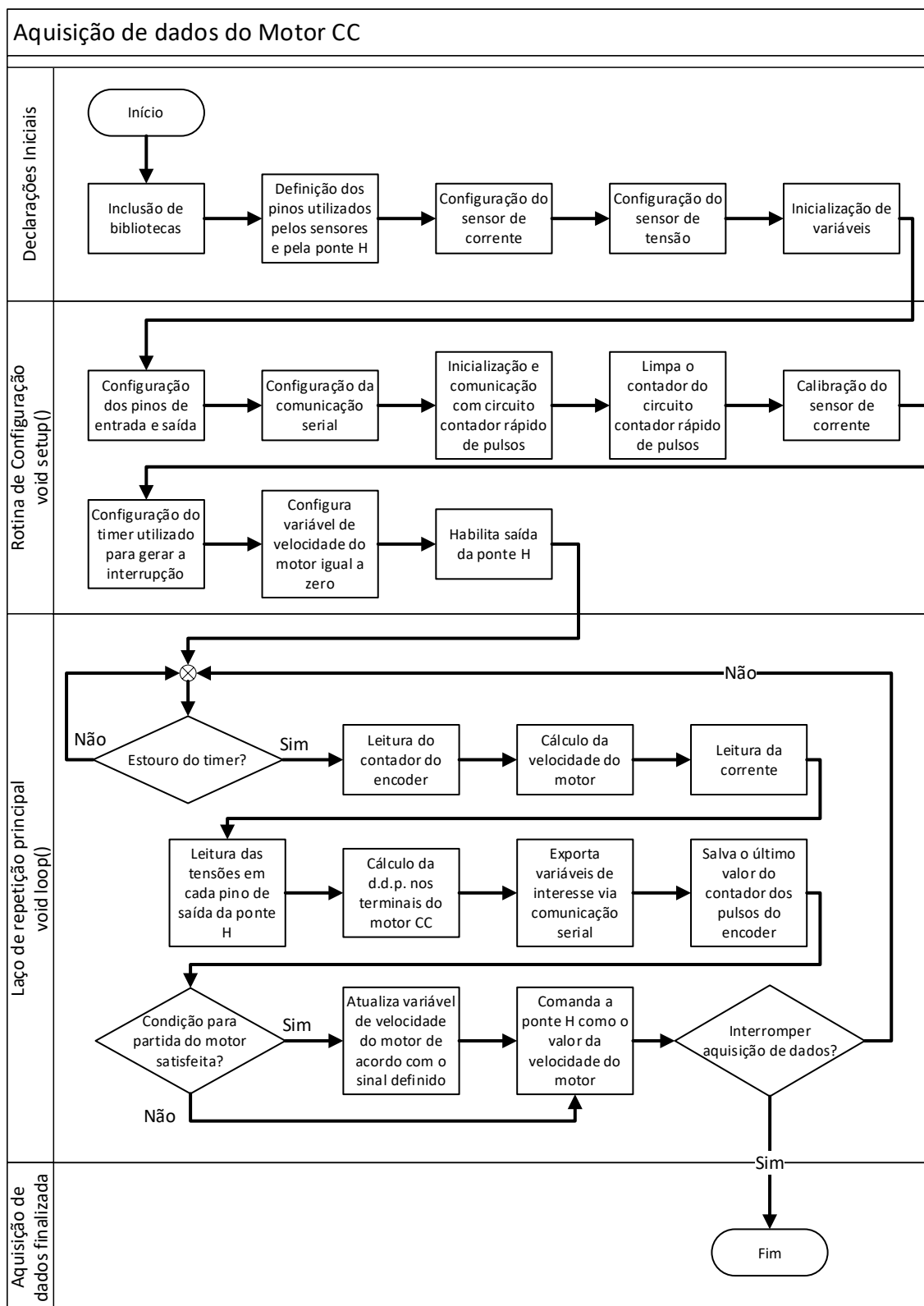


Figura 3. Fluxograma do código embarcado no Arduino para a aquisição de dados. Fonte: Autoria própria (2021).

Ainda na Figura 3, a raia denominada Laço de repetição principal representa a execução da aquisição de dados. A cada ocorrência de interrupção, os sensores são lidos, os cálculos necessários são realizados, as variáveis de interesse são transmitidas via comunicação serial e a saída de comando da ponte H é atualizada. Diferentes sinais podem ser gerados pelo Arduino para o comando do motor, como, por exemplo, um pulso ou uma rampa de tensão.



O sinal de comando para a ponte H só é enviado quando o critério de partida do motor é satisfeito. Para este trabalho, o critério foi um intervalo de tempo de 2 s, ou seja, o motor só parte depois de terem se passado 2 s desde que o Arduino foi iniciado. Antes disso, o motor recebe apenas uma velocidade nula, significando que o duty cycle é mínimo e a saída da ponte H está desligada. Por fim, se a aquisição de dados necessária foi realizada, a execução é interrompida.

O cálculo da velocidade do motor é realizada com base na variação da contagem dos pulsos do encoder. O circuito contador rápido registra ininterruptamente os pulsos do encoder, contando um pulso a cada subida de nível, ou seja, quando o sinal vai do nível baixo de tensão para o nível alto, de forma incremental. Assim, a cada ocorrência da interrupção por tempo do Arduino é realizada a leitura atual da contagem dos pulsos e, a partir da Equação 1, a velocidade é calculada. Em que  $P_n$  é o valor atual do contador,  $P_{n-1}$  o valor registrado na medição anterior, 1800 é o valor dos pulsos por revolução do encoder utilizado,  $\Delta t$  é o intervalo de tempo, em segundos, entre as medições da contagem de pulsos,  $1/17$  é a relação de redução da caixa de redução do motor dado que a velocidade é medida no rotor do motor e a velocidade de atuação é a do eixo da caixa de redução. A relação de  $60 \text{ s}/1 \text{ min}$  é feita para conversão das unidades, de forma que a unidade da velocidade resultante  $V_r$  seja rotações por minuto, ou RPM.

$$V_r = \frac{1}{17} \left( \frac{(P_n - P_{n-1}) \cdot \frac{60 \text{ s}}{1 \text{ min}}}{\Delta t \text{ s}} \right) \quad (1)$$

A medição da corrente é realizada por meio da leitura da tensão analógica, que varia de 0 a 5 V, de saída do sensor ACS712. Como sua faixa de medição é de -30 a 30 A, o zero é deslocado. Isso significa que o Arduino lê no pino de entrada analógico um valor central em sua escala quando a corrente medida pelo sensor é zero. Isso deve ser considerado para que o valor de corrente calculado não esteja deslocado. Dada a resolução de 10 bits da entrada analógica do Arduino, a entrada analógica tem 1023 valores possíveis, com resolução de 4,8 mV. Assim, o valor central da escala é 512. Valores maiores que 512 significam correntes positivas e valores menores, correntes negativas. A biblioteca para Arduino ACS712.h realiza os cálculos necessários e disponibiliza o método usado para medir a corrente com o sensor utilizado.

Como o acionamento do motor é realizado por um circuito ponte H, a diferença de potencial (d.d.p.) no motor depende da diferença de potencial entre os terminais de saída da ponte H, o que determina o sentido de giro do motor. Cada saída do acionamento, OUTA1 e OUTB1 na Figura 1(c), é independente. Dessa forma, dois circuitos divisores de tensão são usados para medir cada uma das tensões e calcular, assim, a d.d.p. aplicada ao motor CC.

Como mostrado na Figura 2, os circuitos divisores de tensão dividem a tensão do motor, até 12 V, por três, pois são três resistores de resistências iguais. O valor de tensão lido pelo Arduino varia, então, entre 0 e 4 V. Assim, utilizando a Equação 2 o valor de tensão  $v_{motor}$  é calculado na escala 0 a 12 V.

$$v_{motor} = L_A \left( \frac{5V}{1023} \right) \cdot \left( \frac{R_T}{R_1} \right) - L_B \left( \frac{5V}{1023} \right) \cdot \left( \frac{R_T}{R_4} \right) \quad (2)$$

Em que  $L_A$  é o valor da leitura do pino analógico do Arduino referente à saída OUTA1 do circuito de acionamento, variando de 0 a 1023. De forma análoga,  $L_B$  é a leitura referente à saída OUTB1.  $5V/1023$  é utilizado para cálculo da resolução da entrada analógica,  $R_T$  é a resistência total do circuito de medição e  $R_1$  ou  $R_4$  é o resistor a qual a tensão é efetivamente medida, pela Figura 2,  $R_1$  no caso da saída da ponte H OUTA1 e  $R_4$  para a saída da ponte H OUTB1.



## 4. RESULTADOS E DISCUSSÕES

Para demonstrar o uso da plataforma, são apresentados a seguir os resultados de três condições experimentais: entrada pulso com diferentes amplitudes e entrada em rampa com diferentes inclinações, em ambos os casos, aplicadas ao motor a vazio, e medição de corrente e velocidade do motor sob condição de carga acoplada ao eixo, com cargas de 1, 2 e 3 kg. Serão apresentados os dados brutos medidos diretamente pelo sistema e, para os dois primeiros casos, o processamento dos dados para melhor visualização dos resultados.

### 4.1. Entrada pulso

Nesse cenário, um pulso de tensão definida por meio do software embarcado no Arduino e com duração de 1,5 segundos é aplicado ao motor. Foram definidos cinco níveis de tensão conforme a saída PWM do Arduino, que possui resolução de 8 bits e, por isso, admite valores entre 0 e 255, sendo 0 equivalente ao duty cycle mínimo e 255 ao duty cycle máximo. Assim, o intervalo entre 0 e 255 foi dividido em cinco níveis iguais, ou seja, com um incremento de 51 unidades a cada nível.

Os níveis de tensão foram nomeados de V1 a V5, representando V1 o duty cycle máximo e a maior tensão média, ou seja, a escrita no pino analógico do Arduino do valor 255, e V5 o menor duty cycle ensaiado, portanto, a menor tensão média testada, significando a escrita no pino analógico do Arduino do valor 51. Quanto à duração do pulso, a mesma foi determinada de forma a possibilitar que o sistema atinja o regime permanente e o comportamento do motor possa ser observado em sua completude.

A Figura 4 mostra os resultados brutos dos experimentos, sendo representadas todas as entradas e saídas agrupadas no mesmo gráfico, para comparação. A tensão média aplicada ao motor foi calculada e inserida nas legendas dos gráficos pois a atuação via PMW apresenta variações da tensão ao longo do tempo, não deixando explícito o valor médio aplicado. É possível observar que, apesar de não afetar a velocidade do motor, a corrente medida acompanha as variações da tensão de entrada.

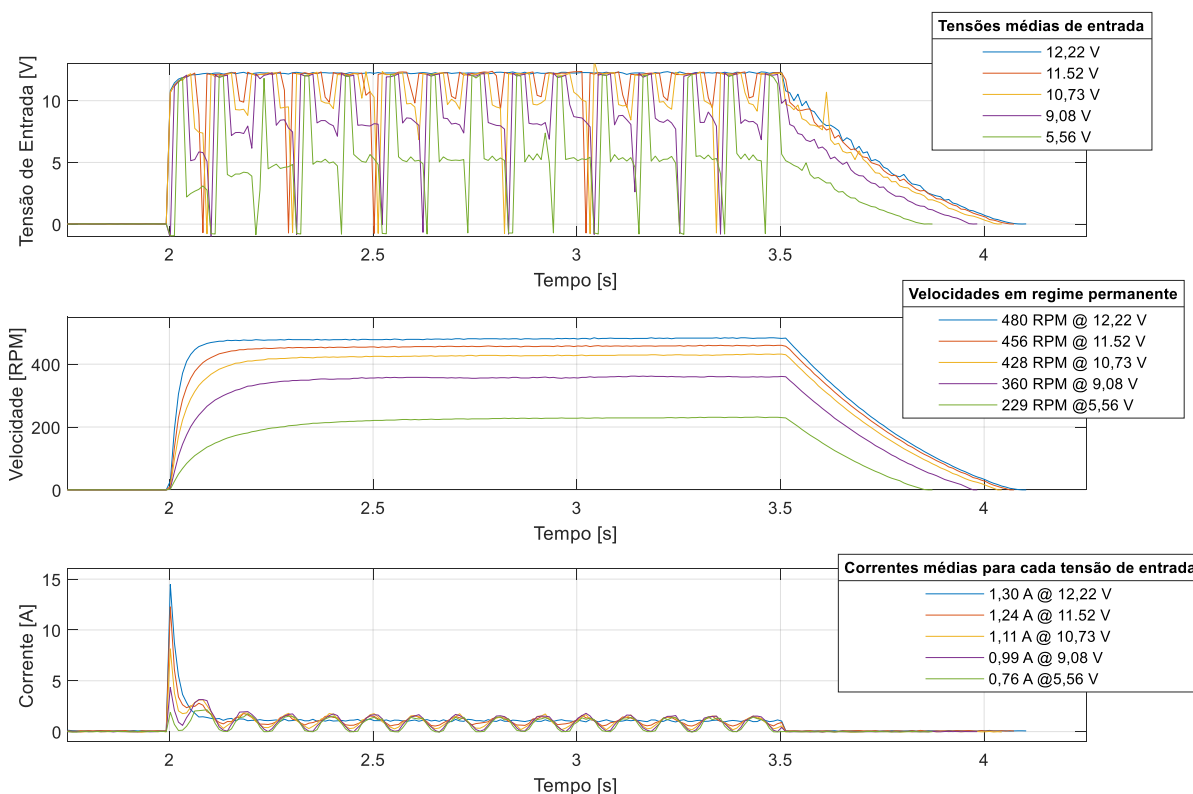


Figura 4. Gráficos de tensão de entrada, velocidade e corrente resultantes do motor para um pulso de entrada com diferentes valores de acionamento da ponte H, resultando em diferentes tensões médias de entrada. Fonte: Autoria própria (2021).

Os sinais medidos de tensão e corrente apresentam variações devido ao acionamento do motor por um sinal PWM. Para interpretação dos dados, é possível aplicar técnicas de processamento de sinais posteriores à aquisição dos dados. A Figura 5 mostra os sinais de tensão, nos gráficos de (a) a (e), brutos e filtrados com um filtro de média móvel com uma janela de 30 amostras. De forma similar, os gráficos de (f) a (j) mostram as medições de corrente em cada situação filtradas por um filtro de média móvel com uma janela de 20 amostras. As correntes de I1 a I5 representam as correntes resultantes da aplicação das tensões de entrada de V1 a V5, respectivamente. Desconsiderando-se o atraso inserido no sinal devido ao número de amostras, os valores filtrados possibilitam uma interpretação mais direta do comportamento do motor.

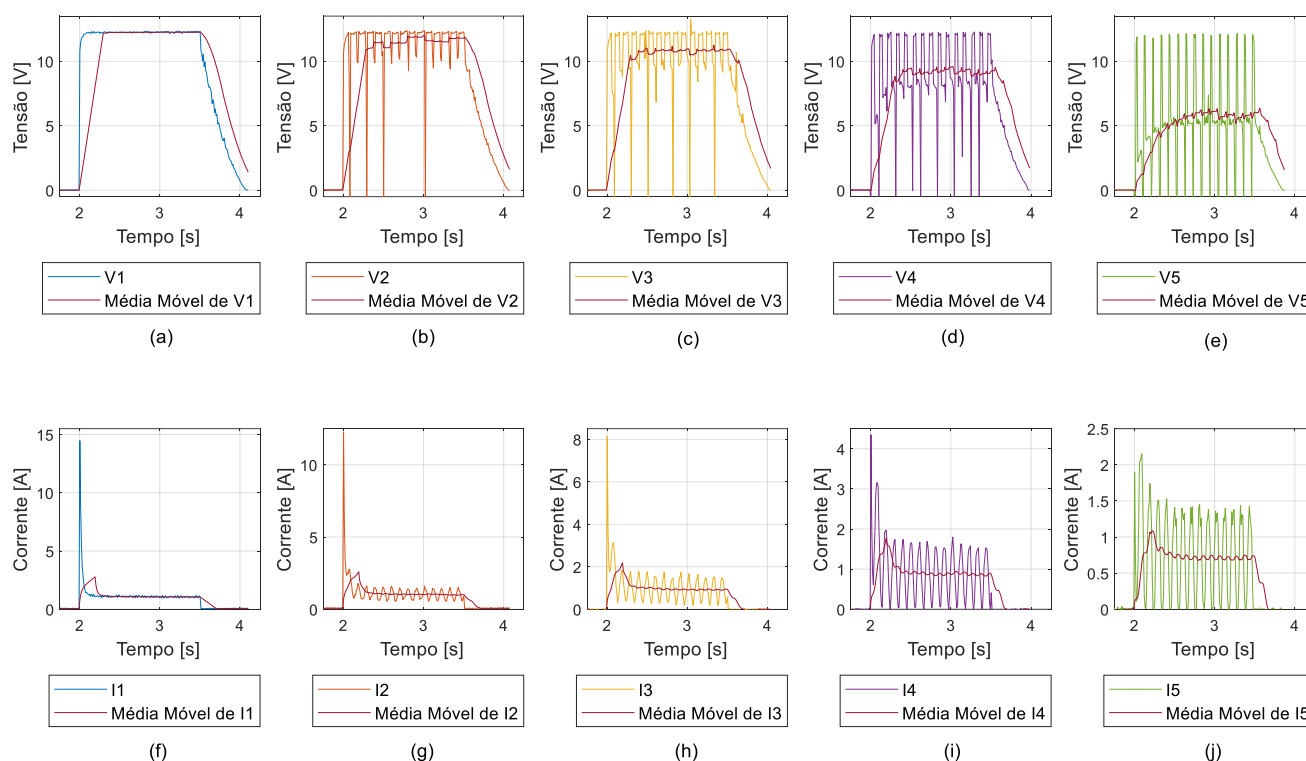


Figura 5. Gráficos detalhados da tensão de entrada e da corrente resultante do motor para cada pulso de tensão. Os gráficos de (a) a (e) mostram as tensões de entrada medidas e as respectivas médias móveis com janela de 30 amostras. Os gráficos de (f) a (j) mostram as correntes resultantes medidas e as respectivas médias móveis com janela de 20 amostras. Fonte: Autoria própria (2021).

O comportamento do conjunto, ou seja, motor e circuito de acionamento ponte H, mostra sua não linearidade, visto que os níveis de tensão definidos via software são lineares, mas o comportamento do sistema não. Esse comportamento também pode ser observado nos resultados da Seção 4.2.

## 4.2. Entrada Rampa

Para aplicar uma entrada de tensão em rampa no motor CC, o valor do duty cycle da saída PWM do Arduino foi incrementado com três diferentes intervalos de tempo, gerando três rampas com inclinações distintas. A rampa representada por V1 tem uma inclinação suave e leva 14 segundos para atingir o valor máximo. A rampa V2 tem uma inclinação mais acentuada e leva metade do tempo de V1 para atingir o valor máximo de tensão aplicada no motor CC. Já rampa V3 tem a inclinação mais acentuada e variação menos suave da tensão em comparação com as anteriores, levando 4 segundos para atingir o valor máximo.

A Figura 6 mostra os resultados de tensão, velocidade e corrente medidas a partir de cada situação. No gráfico, a Velocidade,  $V_r1$  é a velocidade resultante da entrada V1 e  $V_r2$  e  $V_r3$  são as velocidades resultantes relativas às tensões de entrada V2 e V3. De forma análoga, no gráfico de Corrente, I1, I2 e I3 representam as correntes resultantes dadas as tensões de entrada V1, V2 e V3, respectivamente.

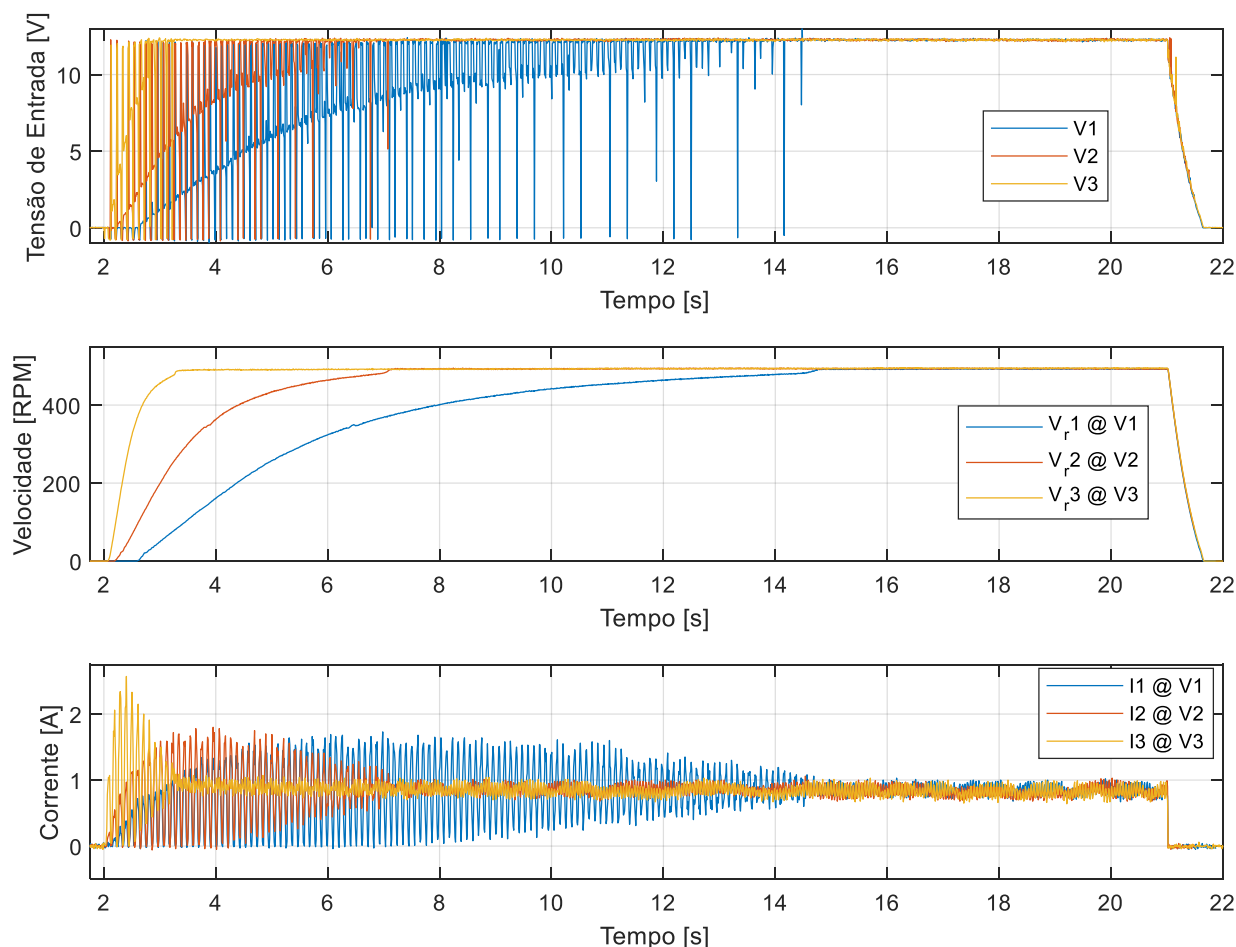


Figura 6. Gráficos de tensão de entrada e velocidade e corrente resultantes do motor para diferentes funções rampa. Fonte: Autoria própria (2021).

A variação gradual da tensão aplicada no motor CC resulta em uma variação também gradual de sua velocidade, porém, apresenta uma medição de tensão e corrente com alto nível de variação, que não é, necessariamente, ruído. Apesar disso, é possível observar que as variáveis convergem para o mesmo valor no regime permanente.

Nos gráficos de (a) a (c) da Figura 7 é possível observar cada uma das tensões de entrada aplicadas comparadas com o resultado do sinal filtrado usando média móvel com uma janela de 30 amostras e com a rampa de referência definida no pino de saída analógica do Arduino. É possível observar novamente a não linearidade do sistema, principalmente no gráfico (a), onde é possível observar a zona morta do sistema, ou seja, o intervalo em que a variação da tensão de entrada não causa variação no sistema.

Já os gráficos de (d) a (f) da Figura 7 mostram a corrente referente a cada rampa de entrada e o mesmo sinal filtrado com uma média móvel com uma janela de 20 amostras para comparação. É possível observar no gráfico (f) que mesmo o sinal de entrada sendo uma rampa, a variação brusca ainda causou um pico de corrente, se comparado com o valor do regime permanente.

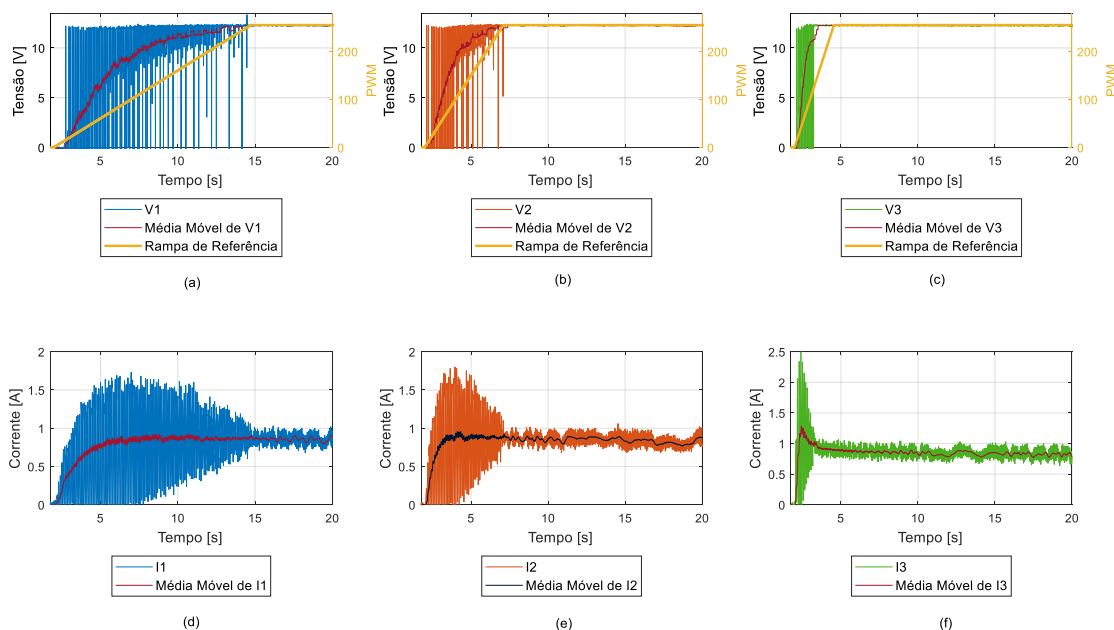


Figura 7. Gráficos detalhados de tensão e corrente do motor para diferentes funções rampa. Os gráficos de (a) a (c) representam, respectivamente, a tensão de entrada para rampas de tensão com diferentes inclinações. Os gráficos de (d) a (f) representam a corrente resultante no motor em cada situação. Fonte: Autoria própria (2021).

### 4.3. Medição sob diferentes condição de cargas

Para essa condição, aplica-se um pulso de tensão com duração de 1 segundo com o valor máximo do circuito acionador do motor, sendo apenas a carga ajustada para cada ensaio. A Figura 8 mostra os gráficos resultantes do experimento. Para o mesmo valor de tensão de entrada, o comportamento da velocidade é inversamente proporcional ao aumento da carga, enquanto a corrente é diretamente proporcional à mesma.

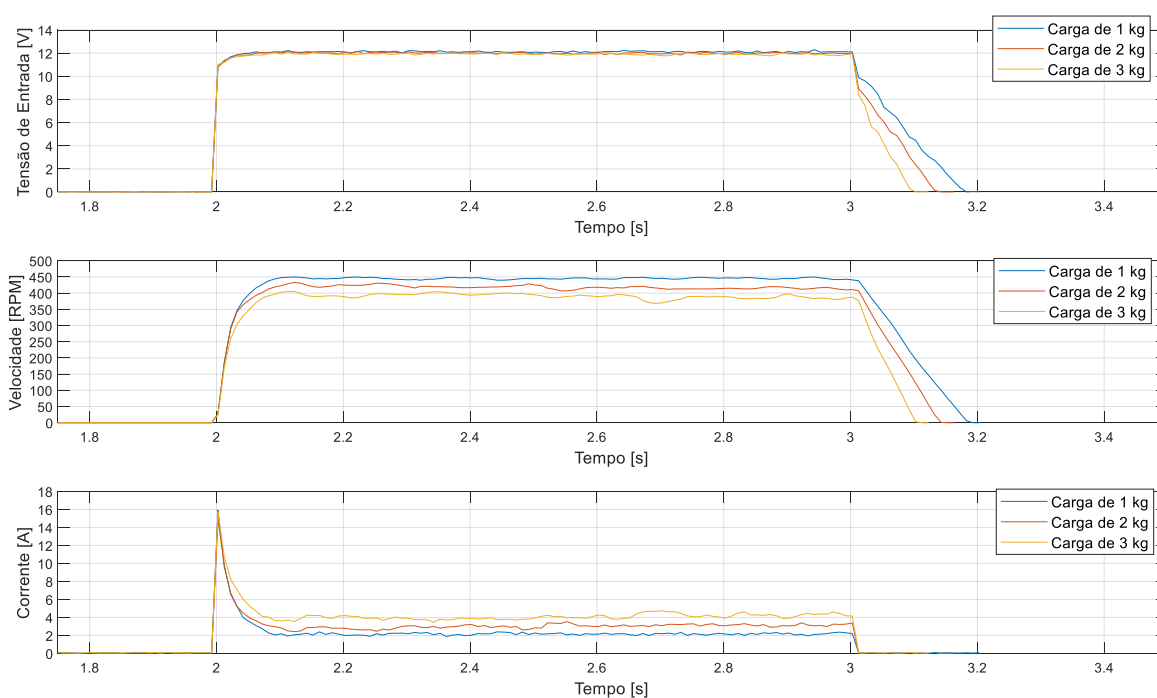


Figura 8. Gráficos de tensão de entrada e velocidade e corrente resultantes do motor para diferentes valores de carga acoplada ao eixo. Fonte: Autoria própria (2021).

#### 4.4. Aplicações da plataforma proposta

A plataforma proposta tem o objetivo principal de realizar medições de três variáveis de um motor CC para a modelagem e estimação de parâmetros. Um exemplo dessa aplicação é a utilização do *toolbox System Identification* do MATLAB®. Essa ferramenta permite o uso de diferentes técnicas e tipos de modelos para gerar modelos matemáticos a partir de sinais de entrada e saída de sistemas reais (LJUNG, 2012).

Os principais modelos lineares são o modelo auto-regressivo com entradas externas, do Inglês *autoregressive with exogenous inputs* (ARX), o modelo auto-regressivo com média móvel e entradas externas, do Inglês *autoregressive moving average with exogenous inputs* (ARMAX) e o modelo de *Box-Jenkins* (BJ).

O modelo ARX é o mais usado e outros modelos só são usados se ele falhar na representação linear do sistema. Seus parâmetros podem ser estimados usando uma técnica de determinação de mínimos quadrados. Já o modelo ARMAX, em comparação com o ARX, é mais flexível, pois possui um modelo de ruído ampliado. O modelo de *Box-Jenkins* tem funções de transferência sem parâmetros em comum e independentemente parametrizadas. Isso faz com que o ele possa filtrar distúrbios de qualquer ruído colorido (NELLES, 2020).

A Figura 9 mostra a interface de usuário da ferramenta *System Identification*. Do lado esquerdo, três sinais foram carregados, os dados medidos nos experimentos com pulsos de tensão de entrada V1, V3 e V5 da Seção 4.1. Os modelos gerados têm como única entrada a tensão do motor e como única saída a velocidade do motor. Nesse caso, a corrente medida não é considerada.

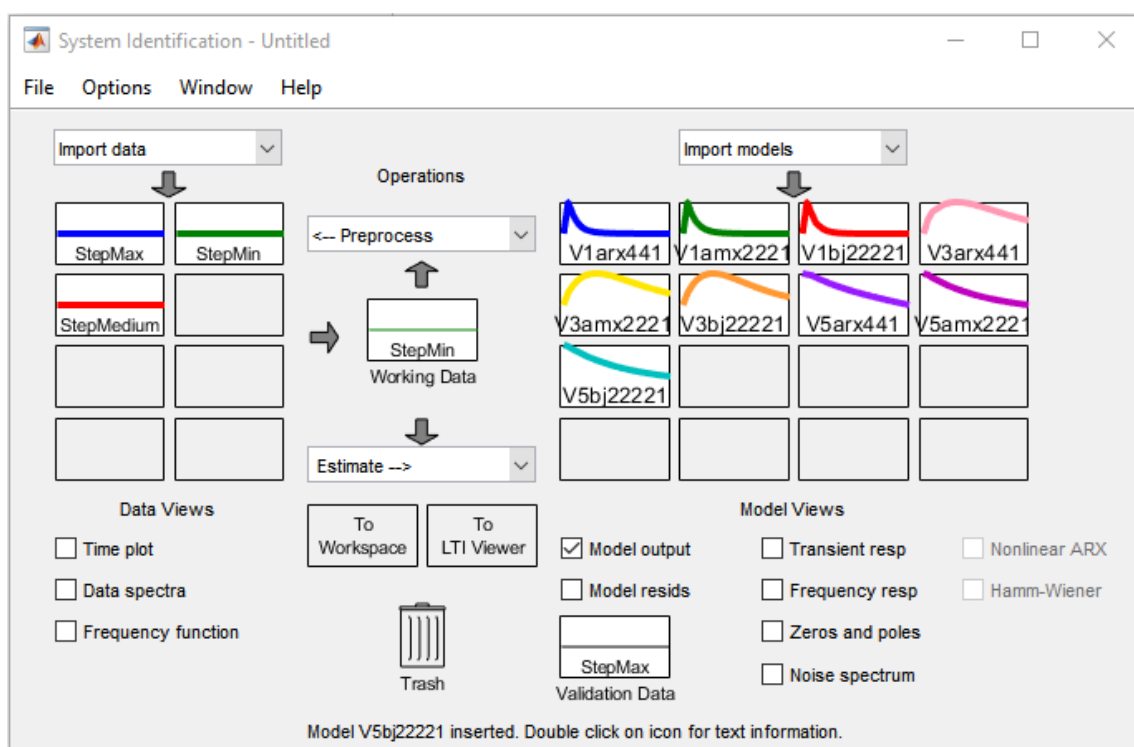


Figura 9. Gráficos de tensão de entrada e velocidade e corrente resultantes do motor para diferentes valores de carga acoplada ao eixo. Fonte: Próprios autores (2021).

Como exemplo, foram gerados três modelos a partir de cada dado de entrada. A partir dos dados de tensão e velocidade medidas do experimento com o pulso denominado StepMax (V1), foram gerados os modelos V1ARX441, V1ARMAX2221 e V1BJ22221. De forma similar, usando as medições de StepMedium (V3), foram gerados os modelos V3ARX441, V3ARMAX2221 e V3BJ22221 e para StepMin (V5), os modelos V5ARX441, V5ARMAX2221 e V5BJ22221.

A própria ferramenta permite uma comparação dos modelos gerados com dados de validação, como mostrado na Figura 10. Todos os modelos gerados são comparados com a velocidade medida a partir da entrada V1. Os gráficos gerados são uma estimativa da velocidade a partir de cada um dos modelos gerados, dada a entrada V1. Ao lado do gráfico a ferramenta *System Identification* ordena os modelos de acordo com o grau de exatidão, em porcentagem, em relação aos dados de validação. Assim, quanto maior o valor, a velocidade estimada pelo modelo está mais próxima da velocidade real medida. Como é de se esperar, os modelos gerados com os dados de V1 têm os melhores resultados. Por isso, é importante ter dados de validação distintos dos dados utilizados para a construção do modelo.

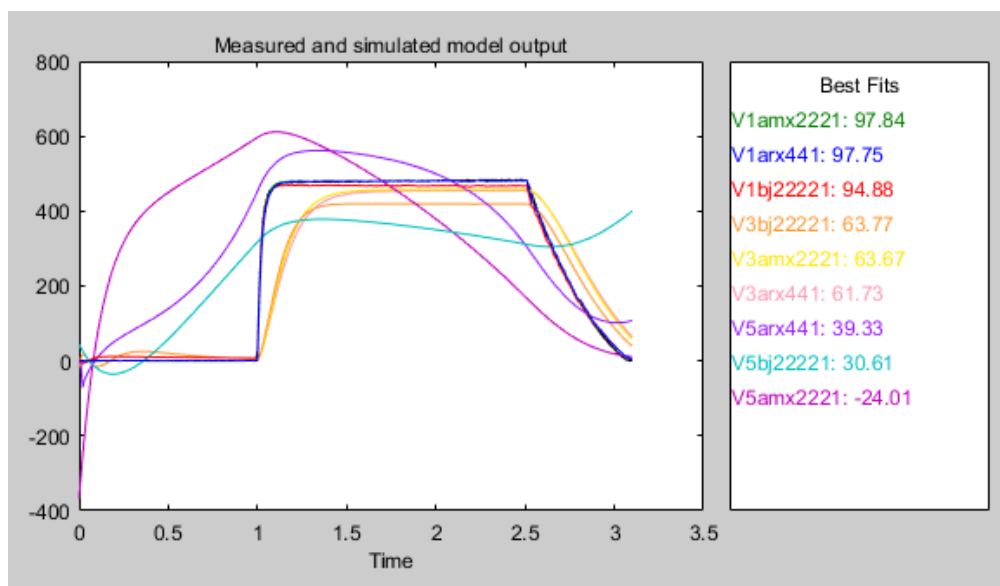


Figura 10. Comparação entre os modelos gerados usando os dados do experimento com V1 como dados de validação, tendo como variável de saída a velocidade do motor. Fonte: Autoria própria (2021).

As Figuras 11 e 12 mostram os resultados da estimativa dos modelos considerando os dados das outras duas condições como dados de validação para comparação. Os modelos gerados podem ser utilizados para a simulação do sistema e para o projeto de controladores, por exemplo.

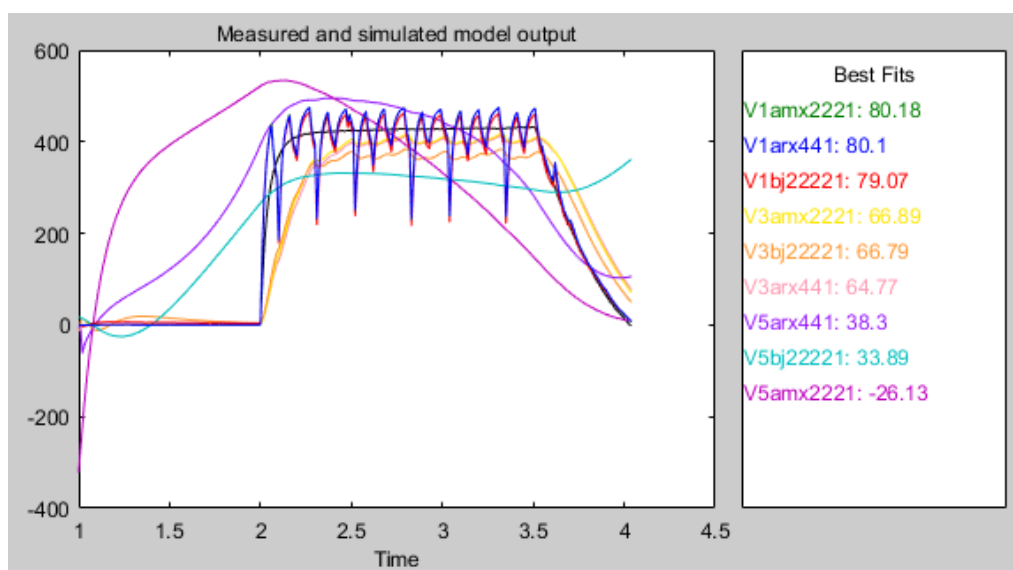


Figura 11. Comparação entre os modelos gerados usando os dados do experimento com V3 como dados de validação, tendo como variável de saída a velocidade do motor. Fonte: Autoria própria (2021).



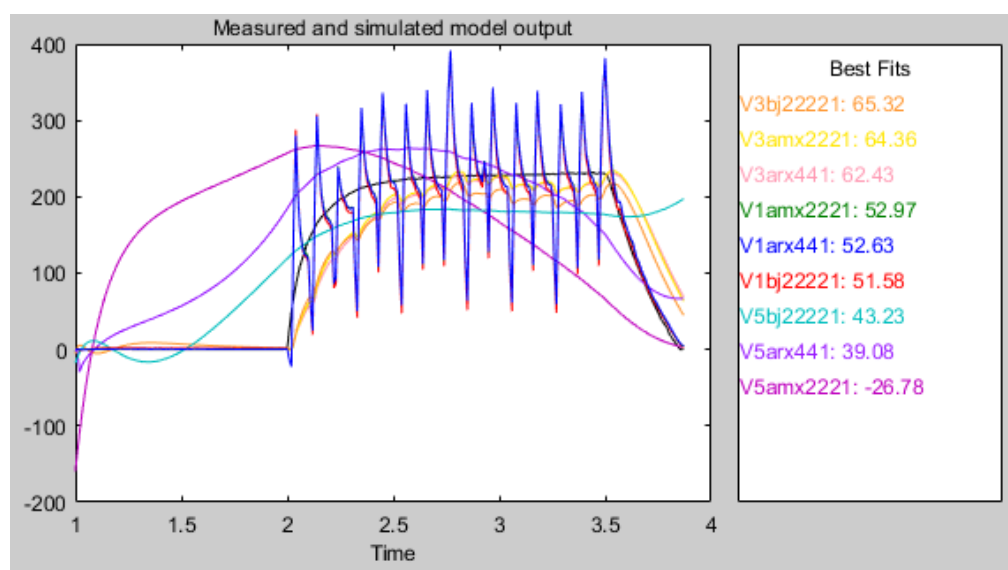


Figura 12. Comparação entre os modelos gerados usando os dados do experimento com V5 como dados de validação, tendo como variável de saída a velocidade do motor. Fonte: Autoria própria (2021).

Como é possível observar, o uso de modelos lineares faz com que os modelos gerados se adéquem apenas a condições específicas. Assim, modelos mais complexos, não-lineares e até adaptativos fazem-se necessários caso seja preciso representar um intervalo maior de condições de operação do motor.

Além disso, outras atividades de ensino e pesquisa podem ser realizadas na plataforma, como o estudo de técnicas de controle de sistemas dinâmicos, usando o motor como planta (CABRÉ et al., 2021), e também o estudo de processamento digital de sinais, a partir das medições das variáveis do motor (AL-BARGOTHI et al., 2019). No primeiro caso, o Arduino pode ser programado para funcionar como um controlador, fechando a malha de controle do motor, que pode ser de velocidade e/ou de corrente. Assim, diferentes técnicas de controle podem ser testadas. Para o estudo de processamento digital de sinais, as medições de tensão e corrente do motor, como vistas nas Figuras 4 e 6, apresentam um alto nível de variação devido ao acionamento do motor via PWM. Assim, técnicas de filtragem digital de sinais podem ser estudadas tanto aplicadas posteriormente à medição, como o realizado nesse trabalho, mas também em tempo de execução, embarcando o filtro digital no Arduino.

## 5. CONCLUSÕES

O artigo apresenta uma proposta de plataforma didática para o ensaio de motores CC com o objetivo de realizar medições das variáveis tensão, corrente e velocidade para a aplicação de técnicas de identificação de sistemas. Apesar de ser muitas vezes desconsiderada, a correta e adequada modelagem matemática dos atuadores em robôs móveis têm papel essencial no projeto de controladores e na representação do sistema em simuladores.

Foram apresentadas três condições de ensaios: entrada pulso, entrada rampa e medição com diferentes cargas acopladas ao eixo do motor. É exemplificado, a partir dos dados de um dos ensaios, a identificação do sistema usando o *toolbox System Identification* do MATLAB®. Além disso, o artigo discute brevemente as possíveis aplicações da plataforma, além do uso para identificação de sistemas. Os resultados mostram a versatilidade da plataforma do ponto de vista didático, cobrindo diversas áreas de cursos técnicos e superiores na área de Engenharia.

## Referências

AGUIRRE, L. A. **Introdução à identificação de sistemas**: Técnicas lineares e não-lineares aplicadas a sistemas reais. 2. ed. Editora UFMG, Belo Horizonte, Minas Gerais, Brasil 730 p., 2004.



AL-BARGOTHI, Sarah N.; QARYOUTI, Ghazi M.; JABER, Qazem M. Speed control of DC motor using conventional and adaptive PID controllers. **Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)**, v. 16, n. 3, p. 1221-1228, 2019.

Allegro Microsystems. ACS712 - Fully Integrated, Hall-Effect-Based Linear Current Sensor IC (Ficha Técnica). Disponível em: < <https://datasheetspdf.com/pdf/570845/AllegroMicroSystems/ACS712/1>>. Acesso em: 26 de julho de 2021.

AKGÜN, G. *et al.* System identification using LMS, RLS, EKF and Neural Network. **Proceedings of 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES)**, Cairo, Egito, p. 1–6, 2019.

Autor1; Autor3. Título do Artigo. **The International Journal of Advanced Manufacturing Technology**, Springer, ANO.

CABRÉ, Tomàs Pallejà *et al.* Didactic platform for DC motor speed and position control in Z-plane. **ISA Transactions**, v. 118, p. 116-132, 2021.

ECHEVERRIA, Gilberto *et al.* Modular open robots simulation engine: Morse. **Proceedings of 2011 IEEE International Conference on Robotics and Automation**, Shanghai, China, p. 46-51, 2011.

FAIEGHI, M. R.; AZIMI, S. M. Design an optimized PID controller for brushless DC motor by using PSO and based on NARMAX identified model with ANFIS. **Proceedings of the 12th International Conference on Computer Modelling and Simulation**, Cambridge, Reino Unido, p. 16–21, 2010.

KARA, T.; EKER, I. Nonlinear modeling and identification of a DC motor for bidirectional operation with real time experiments. **Energy Conversion and Management**, Elsevier, v. 45, n. 7-8, p. 1087–1106, 2004.

LJUNG, Lennart; SINGH, Rajiv. Version 8 of the MATLAB system identification toolbox. **IFAC Proceedings Volumes**, v. 45, n. 16, p. 1826-1831, 2012.

MOKHLIS, S.; SADKI, S.; BENSASSI, B. System identification of a dc servo motor using ARX and ARMAX models. **Proceedings of 2019 International Conference on Systems of Collaboration Big Data, Internet of Things & Security (SysCoBIoTS)**, Casablanca, Marrocos, p. 1–4, 2019.

NAUNG, Y.*et al.* Implementation of data driven control system of dc motor by using system identification process. **Proceedings of 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)**, Moscou e São Petersburgo, Rússia, p. 1801–1804, 2018.

NELLES, O. **Nonlinear System Identification: From Classical Approaches to Neural Networks, Fuzzy Models, and Gaussian Processes**. 2. ed. Springer International Publishing, Berlin, Heidelberg, Alemanha, 1825 p., 2020.

OMRON Industrial Automation. E6B2-CWZ6C 1800P/R 0.5M – Incremental rotary encoder (Ficha Técnica). Disponível em: <https://www.ia.omron.com/product/item/2460/>. Acesso em: 26 de julho de 2021.

Sparkfun: SparkFun Monster Moto Shield (Ficha Técnica). Disponível em: < <https://www.sparkfun.com/products/retired/10182>>. Acesso em: 26 de julho de 2021.

TANG, W.-J.; LIU, Z.-T.; WANG, Q. DC motor speed control based on system identification and PID auto tuning. **Proceedings of 36th Chinese Control Conference (CCC)**, Dalian, China, p. 6420–6423, 2017.

TANGIRALA, A. K. **Principles of system identification: theory and practice**. CRC Press, Boca Raton, Florida, EUA, 908 p. 2015.

Recebido em: 30/07/2021

Aceito em: 10/02/2023

Endereço para correspondência

Nome: Wallace Pereira Neves dos Reis

E-mail: wallace.reis@ifrj.edu.br



Esta obra está licenciada sob uma [Licença Creative Commons Attribution 4.0](https://creativecommons.org/licenses/by/4.0/)