DESIGN OF DC MOTOR CONTROLLER BASED ON ARM-M3

Pham Thi Thanh Loan, Dao Hieu
Hanoi University of Mining and Geology, Hanoi, Vietnam

Abstract: This document aims to design a control strategy for direct current (DC) motor, including speed and position controllers. ARM Coretex-M3 is used to develope the driver allowing the users to switch to stepping drivers without changing control systems. The transfer function of the system is obtained by determining motor parameters in tuning mode. In the paper, the hardware construction of the driver and control algorithms are presented. The PID controllers with different combinations of proportional, integral and derivative are proposed to identify the optimal strategy. The simulation results are compared with the outputs of the actual model to verify the effectiveness of the proposed system and the research method for DC motor control. This work contributes a simple technique for designing speed and position controllers based on overshoot value, steady-state error, and response time. Additionally, the solution using ARM-M3 helps reduce the cost of a motor driver for low-power applications.

Keywords: DC Motor, Closed-loop, Speed controller, Position controller, PID controller.

I. INTRODUCTION

DC motors play an important role in automation systems, converting electrical energy into mechanical energy. They operate on the principle that when a curent flows through a conductor placed in a magnetic field, a force is generated on the conductor making it spin. DC motor offer advantages over alternating current (AC) motor in terms of overload capabilities, speed performance and high starting torque. These motors are widely used in robotics, mechatronic systems, CNC (computer Numerical Control) machines and various other appliances. Due to the fact that they have a rather simple internal structure, DC motors can be easily controlled using analog eclectronics. However, for higher precision and more complex algorithms, numerical controllers are preferred.

Many studies have achieved achieved optimal controller performance [1][2]. The research on using microcontroller to control brushless DC motor has significant practical applications. The simple hardware structure with PIC microcontroller and protection functions allows for secondary development while maintaining cost efficiency. The PI algorithm for speed and current loop is conveniently implemented using the processor's available resources [3].

Current DC motor drives are expensive and difficult to customize. Furthermore, the software is not opened along with complicated program making them difficult for users during design process and implement system. The DC motor controller intergrated Model Based Design (MBD) concept based on Arduino has solved above issues. The hardware is combined of basic equipment, including Arduino, input/output interfaces, an encoder and other peripherals. The software is developed in Simulink enabling PID algorithm design, simulation, real-time data observation... This system meets the technical and functional requirements of a motor controller for training and experimentation [4].

In recent years, many effective and reliable morden controllers have been made due to the rapid development of technology. The availability of various solid-state switching devives and various analog/digital chips in controlling circuits make them more powerful regarding the ability to perform in the digital domain. A DC motor drive system using Arduino and an Android-based application with NoteMCU has been successfully implemented. It is a Bluetooth-based control equipment that wirelessly controls DC motor using PWM techniques. In fact, loads of electrical systems are left ON in most of time even when unnecessary. The units connected to the Internet can be observed and regulated online helping the wastage of electrical energy to be reduced [5].

Along with hardware advancements, DC motor control algorithms have been proposed and successfully implemented. The PI controllers are used to adjust current and speed independently or they can be combined in two loops for cascade construction. In the study of using Research using MATLAB and LabVIEW has demonstrated the effectiveness of different controllers. Calculations and state-space controllers are implemented in MATLAB, while the PID controller is executed in LabVIEW. Results show that the state-space controller has the fastest response and the lowest settling times [6].

The fuzzy logic controller (FLC) has also been applied to DC motor in another research [7]. A modified adaptive bats sonar algorithm (MABSA) was compared with standard FLC and particle swarm optimization algorithm (PSO) in term of simulation. Matlab was used to design FLC and model DC motor. Parameters including settling time, rise time and maximum overshoot were analyzed. The results prove that MABSA-FLC algorithm had advantages compare to FLC and PSO-FLC.

Intelligent control techniques using siding mode control (SMC) for the speed control loop have been explored. By constrasting the output signal of FLC with PID controller and SMC, the optimal gain is selected regarding overshoot and stability criteria. Transient response studies conclude that SMC gives lower overshoot as well as faster response times,

Contact author: Pham Thi Thanh Loan
Email: thanhloanbkhn@gmail.com
Manuscript received: 4/2025, revised: 5/2025, accepted: 6/2025.

making it superior to standard FLC and classical PID controllers [8].

This paper aims to develop a DC motor control system using ARM Coretex-M3 to reduce costs and advance technological mastery. Besides, PI controller are designed for velocity and position control loops, as they maintain high control quality and are suitable for experimental models.

Many studies have been conducted in the field of controller design and tuning using deep learning algorithms to enhance control system performance. Some research has compared the effectiveness of traditional PID controllers with artificial neural network (ANN)-based controllers. In one study, the PID controller was tuned to balance stability and response speed for a DC motor, followed by training and optimization of a neural network controller for the same motor model. Simulation results in Matlab/Simulink demonstrated that the neural network controller performed better when faced with changes in rotor inertia without requiring re-tuning. Graph analysis and root mean square error (RMSE) calculations also confirmed that controllers utilizing machine learning algorithms have advantages in long-term control systems [9].

Control methods in electric motor and powertrain systems for new energy vehicles are analyzed by Xiaogang Wu [10]. Results show that permanent magnet synchronous motors perform best, especially when combined with SiC MOSFET converters, enhancing efficiency and driving range. Various control algorithms are compared, demonstrating superior performance over traditional PID control and optimizing electric powertrain performance. Hybrid powertrains benefit from adaptive and intelligent control strategies, improving fuel efficiency.

Advancements in motor control strategies to enhance energy efficiency are also expored [11]. It analyzes how improved control methods, alongside new materials and designs, contribute to better motor performance. Various motor types are reviewed, highlighting recent developments in their control techniques. A comparative study shows that modern control approaches surpass traditional PID control in efficiency and adaptability. Finally, the research presents future directions for optimizing motor control to further improve energy savings.

II. HARDWARE DESIGN FOR CONTROL SYSTEM

This section dedicates an overview on the ARM (Advanced RISC Machine) Cortex-M3 and the driver construction.

The Cortex-M3 processor based on the ARMv&-M is specifically designed to achieve high performance in energy and cost efficient embedded application.

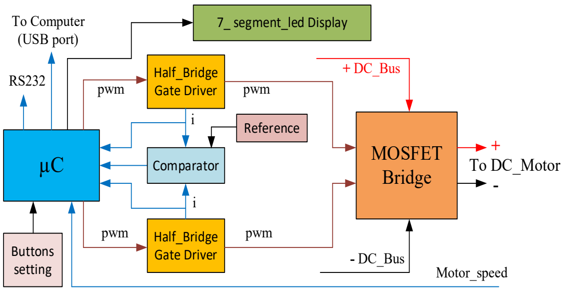

Figure 1. The structure of Drive Systems

The structural diagram of the DC motor control module, as shown in Figure 1, consists of three main parts: a MOSFET H-bridge power circuit, two driver units that control the operation of the bridge (each controlling half of the bridge), and a microcontroller. Additionally, there are auxiliary components such as an overcurrent protection circuit (for output short circuits or motor overload), setup buttons, a display screen, and communication ports for the encoder, computer, and RS232 [12], [13].

A. Power MOSFET Brigde

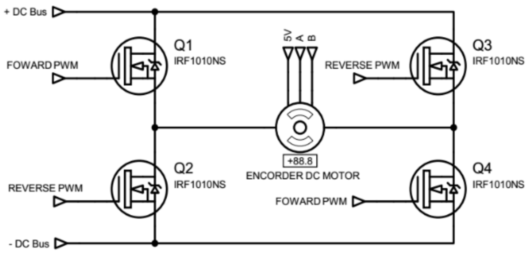

Figure 2. MOSFET bridge

The MOSFET bridge uses four N-channel MOSFET power components to control the speed and rotation direction of the DC motor. With a DC Bus voltage of 12VDC, the selected MOSFETs are KND3306B, which can handle currents up to 80A and withstand voltages up to 60V.

The motor rotates forward when the control signal is sent to the FORWARD PWM port, causing power switches Q1 and Q4 to turn on, supplying DC power from the DC Bus to the motor terminals. Conversely, when the control signal is sent to the REVERSE PWM port, power switches Q2 and Q3 turn on, supplying power to the motor in the opposite direction, making it rotate in reverse.

The motor speed is controlled by the duty cycle of the PWM signal. The motor's speed and rotation direction are determined by the ENCODER, which sends data back to the microcontroller

B. Half_Brigde driver

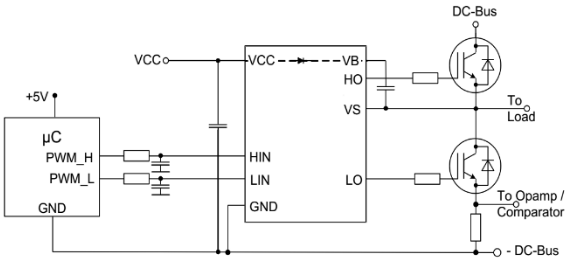

Figure 3. Half Brigde driver ELD05I06

Controlling a DC motor using an H-bridge circuit (Figure 2) poses a risk of DC Bus short circuits if both switches in the same branch turn on simultaneously, especially during switching phases when reversing the motor's direction. When switching from forward to reverse rotation, Q1 and Q4 turn off while Q2 and Q3 turn on. The switching process occurs within a short time interval Δt. During this period, one switch may not have fully turned off while the other is in the process of turning on, leading to cross-conduction a condition where both switches in the same branch conduct simultaneously, creating a direct current path from +DC Bus to -DC Bus, effectively short-circuiting the DC Bus.

To prevent cross-conduction, the module uses drivers to control the MOSFET bridge branches. These drivers introduce dead time, ensuring that one switch in a branch fully turns off before the other turns on. The selected driver for this module is ELD05I06B (Figure 3). Since each driver controls one branch of the bridge, the module requires two drivers.

C. Microcontroller Module

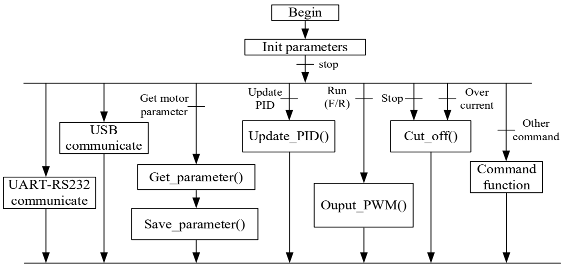

Figure 4. Microcontroller Module Control Algorithm

The microcontroller acts as the CPU of the module, responsible for receiving configuration parameters to control the motor speed. During operation, it monitors the motor speed using feedback from the encoder. The microcontroller is designed with a digital PID controller, allowing users to adjust control parameters as needed. Before operation, the motor parameters are automatically determined by the software and sent to the computer via USB communication. The microcontroller's operation and processing algorithm is illustrated in Figure 4.

The module operates based on control commands from a computer or another microcontroller via USB and standard RS232 UART communication. Two shunt resistors are connected in series with the lower MOSFET power switches to measure the current flowing through the motor (Figure 5). The motor current and angular velocity serve as feedback signals for the microcontroller to execute the PID control algorithm. Additionally, current feedback is used as a safety measure to protect the module from short circuits or output overcurrent conditions.

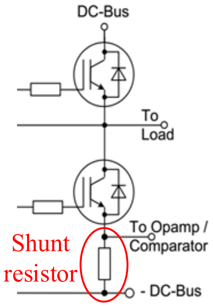

Figure 5. Curent Feedback Signal by Shunt resistor

D. Additional Features

Several auxiliary features have also been updated to enhance usability. A 7-segment LED display and onboard buttons allow users to set parameters and control the module directly without needing external control devices. During motor operation, speed and current parameters can be transmitted via standard RS232 UART communication, depending on the module's configuration

III. SYSTEM SYNTHESIS

A. Control System Structure

The drive control system consists of torque, speed and position control loops (*Figure 6*).

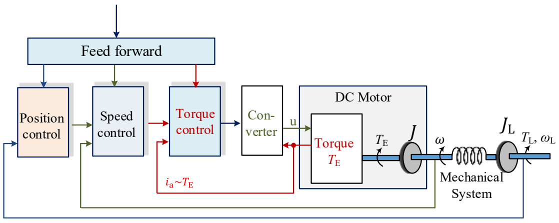

Figure 6. The structure of Drive Systems

k_m: Machine specific parameter;
$\varPhi$: magnetic flux;
e_a: Induced electromagnetic force (EMF)
i_a: Armature current;
T_E: Torque of the machine.
T_L: Load torque;
J, J_L: Inertia of the machine and the load.
ω: angular velocity

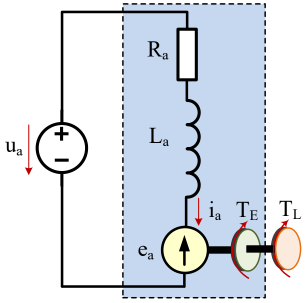

Figure 7. The model of DC motor

Mathematical model of DC motor as follow:
The armature electrical equation:

$$u_\mathrm{a} = i_\mathrm{a} \cdot R_\mathrm{a} + L_\mathrm{a} \cdot \frac{\mathrm{d}i_\mathrm{a}}{\mathrm{d}t} + e_\mathrm{a} \qquad (1)$$

$$e_\mathrm{a} = k_\mathrm{m} \cdot \varPhi \cdot \omega \qquad (2)$$

Torque equation:

$$T_\mathrm{E} = k_\mathrm{m} \cdot \varPhi \cdot i_\mathrm{a} \qquad (3)$$

Torque balance equation:

$$J \cdot \frac{\mathrm{d}\omega}{\mathrm{d}t} = T_\mathrm{E} - T_\mathrm{L} \qquad (4)$$

Identifying the mathematical model of a DC motor using an experimental approach is an important solution for system control and monitoring. Due to the difficulty in directly measuring parameters such as resistance, inductance, time constants, or moment of inertia, the experimental method is used to determine these parameters through output responses when applying input signals to the motor. By collecting data on current, speed, and voltage, and then using system identification techniques such as regression or optimization methods, an approximate model of the motor can be developed, helping to improve control performance. In this study, the acquired data is the speed of DC motor for open_loop.

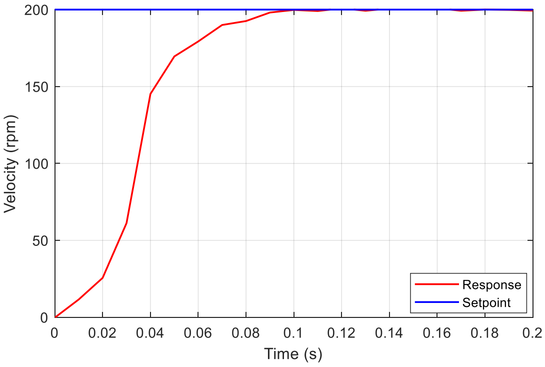

Figure 8. Experimental speed characteristic of the DC motor

The mathematical model of the DC motor can be represented as a first-order inertial system, expressed by the following transfer function:

$$G_M(s) = \frac{k}{Ts+1} \qquad (5)$$

where k is the system gain and T denotes the time constant of the system. With input in the form of a unit step, the output will equal the input value as t approaches infinity. One important characteristic of such an exponential response curve is that at $t = T$ the value of output is 0.632, or the response has reached 63.2% of its total change. In addition, the slope of the response curve is decresed from $1/T$ at $t = 0$ to zero at $t = \infty$. Based on experimental data, the transfer function of the DC motor was identified as:

$$G_M = \frac{7.6628}{0.048s+1} = \frac{1}{0.0063s+0.1305} \qquad (6)$$

B. Velocity Control Constructions

Figure 9 shows a velocity control structure of the DC motor, which consists of a proportional–integral (PI) controller in conjunction with the motor's transfer function. The parameters K_1 and K_2 denote the proportional and integral gains of the velocity controller, respectively.

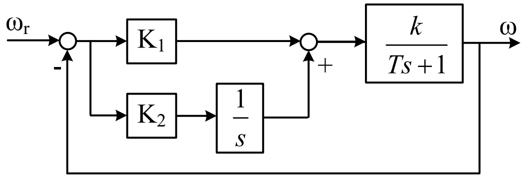

Figure 9. The velocity control structure

The transfer function of the resulting closed-loop system is given by:

$$G_V(s) = \frac{k(K_1s+K_2)}{Ts^2+(1+K_1\cdot k)s+K_2\cdot k} \qquad (7)$$

This transfer function exhibits the form of a standard second-order system, which is typically expressed as:

$$G_S(s) = \frac{\omega_n^2}{s^2+2\xi\omega_n s+\omega_n^2} \qquad (8)$$

where $\boldsymbol{\omega_n}$ is the undamped natural frequency; $\boldsymbol{\xi}$ is damping ratio.

The transient response of the system depends on the value of. For $\mathbf{0 < \xi < 1}$, the system is underdamped, exhibiting oscillatory behavior with complex-conjugate poles located in the left-half of the complex s-plane. If $\boldsymbol{\xi = 0}$, the system becomes a harmonic oscillator. When $\boldsymbol{\xi = 1}$, the system is critically damped, behaving as a second-order inertial stage. For $\boldsymbol{\xi > 1}$, the system is overdamped, with no oscillations in the transient response.

The controller gains K_1 and K_2 are selected based on the desired steady-state error e_{ss}, as defined in [14]:

$$K_1 = 2\xi\sqrt{K_2\cdot T/k} - 1/k \qquad (9)$$

$$K_2 = 1/(k\cdot e_{ss}) \qquad (10)$$

C. Cascade Controller Constructions

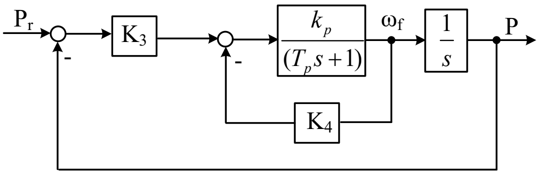

Figure 10. The cascade control structure

The open-loop transfer function of the position control system is obtained by synthesizing the velocity control structure presented in Figure 9. In this context P_r and P represent the reference and actual positions, respectively, and K_3 and K_4 the proportional and integral gains of the position control loop. The transfer function of the resulting closed-loop system, as illustrated in Figure 10, is given by:

$$G_P(s) = \frac{K_3\cdot k_p}{T_ps^2+(1+k_p\cdot K_4)s+K_3\cdot k_p} \qquad (11)$$

The peak time t_p and the undamped natural frequency ω_n are related through the following expressions:

$$t_p = \frac{\pi}{\omega_d};\ \omega_n = \frac{\omega_d}{\sqrt{1-\xi^2}} \qquad (12)$$

The parameters K_3 and K_4 are then determined as:

$$K_3 = \frac{\omega_d^2}{1-\xi^2};\ K_4 = 2\xi\sqrt{K_3T_p/k_p} - 1/k_p \qquad (13)$$

where t_p is peak time and ω_d is the damped natural frequency. These parameters are selected to meet the desired transient response specifications of the position control system.

D. Case study

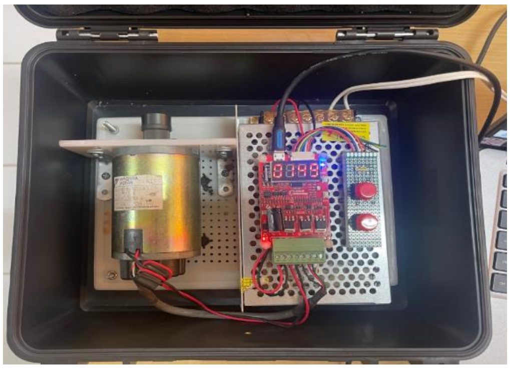

Figure 11. The hardware set up

Figure 11 shows the hardware configuration of the DC motor control system. To evaluate the effectiveness of the proposed control algorithm, various scenarios related to velocity and position control at different speed levels are analyzed in the subsequent subsections.

1). Velocity control system

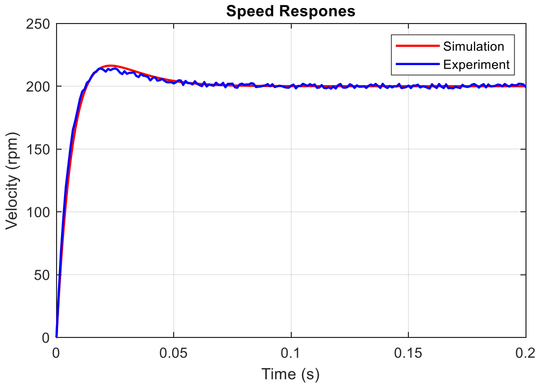

Figure 12. Speed response with a reference of 200 rpm

The graph in Figure 12 represents velocity (rpm) over time (s). The two curves are nearly identical, indicating that the simulation results closely match the actual data, demonstrating the model's high accuracy in capturing the system's dynamics. Initially, the velocity rapidly increases from 0 to approximately 200 rpm within a very short time, reflecting a fast system response. However, a slight oscillation occurs before reaching stabilization, which may be attributed to system inertia or transient response characteristics of the control system.

Upon closer analysis, the system stabilizes at around 200 rpm after approximately 0.1s, with small oscillations gradually diminishing. The discrepancy between simulation and actual results is minimal, reinforcing the validity of the model. The small initial discrepancy could stem from real-world measurement noise or minor assumptions in the simulation, such as idealized component characteristics or neglected external disturbances.

Further experimental analysis was conducted by varying the set speed within the range of 100–300 rpm to observe the system's response under different operating conditions. Additionally, the steady-state error was maintained within the recommended range of 0.0018 to 0.0023, ensuring minimal deviation between actual and simulated results. The damping ratio was varied between 0.85 and 1 to assess the system's overshoot behavior, allowing for an evaluation of control stability and transient response characteristics. These variations provided insights into how controller parameters influence system performance, as summarized in Table 1.

The results suggest that the chosen control strategy effectively minimizes steady-state error while maintaining a well-damped response. The system's ability to closely follow the set velocity with minimal overshoot and rapid stabilization confirms the effectiveness of the applied control methodology.

Table 1. The controller parameters in different cases

Parameter	1	2	3	4	5
e_{ss}	0.0021	0.0023	0.0019	0.002	0.0018
ξ	1	0.9	0.95	1	0.85
K_1	1.109	0.87	0.88	0.79	0.772
K_2	61.01	56.05	60	48.73	54.65

The steady-state error varies within a small range but does not significantly affect the trend of the control parameters. When decreases, ξ the system's oscillation level increases; therefore, K_1 tends to decrease gradually to prevent excessive oscillations.

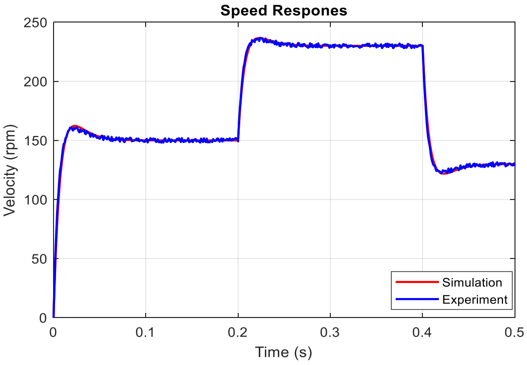

Figure 13. Speed response with different references

The results in Figure 13 illustrate the system's response to sequential changes in the reference speed, specifically from 150 rpm to 230 rpm and then down to 130 rpm. The response demonstrates that the system is capable of tracking the reference values effectively, with the actual speed aligning with the setpoint after approximately 50 ms.

A closer examination of the response dynamics reveals that during each transition, the system exhibits a rapid acceleration or deceleration phase followed by a brief settling period. The rise time, defined as the time taken for the velocity to reach 90% of its final value, remains consistently low, indicating a fast response characteristic. Additionally, the overshoot observed during transitions is minimal, suggesting that the control strategy effectively mitigates excessive deviation beyond the desired value.

The close agreement between the simulated and actual results further confirms the accuracy of the system model. Any minor discrepancies may arise from real-world factors such as sensor noise, friction variations, or slight nonlinearities in the motor and powertrain. The smooth convergence of the actual response to the reference speed without prolonged oscillations highlights the system's stability and robustness. Moreover, the results indicate that the control algorithm maintains a balance between fast response time and minimal steady-state error. The controller parameters, such as proportional and integral gains, appear to be well-tuned, ensuring that the system avoids excessive lag while preventing instability. The ability of the system to accurately follow varying reference speeds further reinforces its suitability for applications requiring precise speed control

2). Position control system

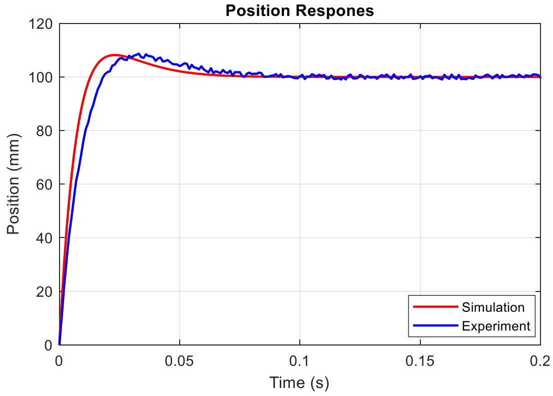

Figure 14. Position response with a reference of 100mm

The system response with the position controller (for setpoints within 100–200 mm) demonstrates fast tracking with minimal delay, reaching the desired value (~100%) quickly. Initial oscillations occur with an overshoot of 8 ÷ 12%, which is within an acceptable range for precise positioning applications. The system stabilizes within approximately 0.1 seconds, confirming its rapid convergence and stability. The close agreement between the simulation and actual results validates the accuracy of the model in representing system dynamics.

The slight overshoot and transient oscillations are influenced by the proportional-integral (PI) control action, where the proportional gain (K_1) primarily affects response speed, while the integral gain (K_2) helps minimize steady-state error. The damping ratio, ranging from 0.85 to 1, ensures a well-damped response without excessive oscillations. By adjusting K_1 and K_2, the overshoot and peak time can be fine-tuned, allowing for better control over transient behavior.

These results highlight the PI controller's effectiveness in achieving fast, accurate, and stable positioning. The ability to modify control parameters provides flexibility to optimize performance for different operating conditions, ensuring precision and reliability in dynamic positioning tasks

Table 2. The controller parameters in different cases

Parameter	1	2	3	4	5
M_p	0.088	0.012	0.09	0.11	0.13
t_p	0.025	0.02	0.03	0.028	0.019
K_3	159.1	193.93	83.56	75.212	198.23
K_4	1.095	1.027	0.64	0.538	0.9696

Increasing K_3 generally helps reduce the peak time t_p, but it can either increase or decrease the overshoot M_p depending on the value of K_4. A lower K_4 tends to reduce M_p but may also slow down the system. To optimize the system, it is essential to balance response speed and overshoot by properly adjusting K_3 and K_4.

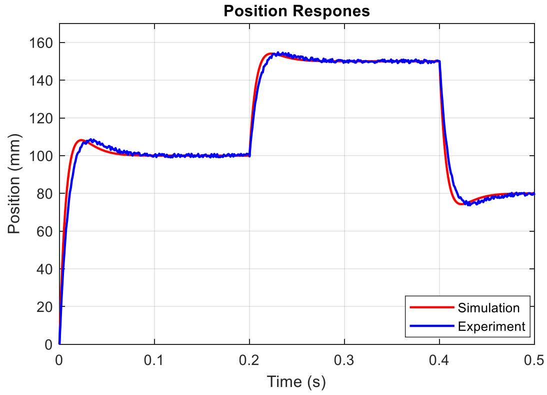

Figure 15. Position response with different references

The results in Figure 15 show that when the reference value changes sequentially from 100, 150, and 80 mm, the response follows the set value after approximately 80ms. The simulation results closely match the actual results, confirming the accuracy of the model.

IV. CONCLUSION

This study presents an effective control strategy for DC motor speed and position regulation using an ARM Cortex-M3 microcontroller. By designing and implementing PID controllers with various parameter combinations, we analyze system performance through output responses to different setpoint values. The simulation and experimental results confirm the reliability of the proposed method, demonstrating minimal deviation between theoretical and real-world outcomes.

The findings indicate that adjusting controller parameters, particularly and, significantly influences overshoot, steady-state error, and response time. While increasing generally reduces peak time, its effect on overshoot varies depending on. Lower values tend to minimize overshoot but may slow down the system. Therefore, achieving optimal performance requires a careful balance between response speed and overshoot.

Additionally, utilizing ARM Cortex-M3 enhances cost efficiency, making this approach suitable for low-power applications. The research contributes a practical and adaptable solution for DC motor control, facilitating further advancements in automation and embedded systems

REFERENCES

[1] J. Holtz, W. Lotzkat, and A. M. Khambadkone. "On continuous control of PWM inverters in the overmodulation range including the six-step mode," IEEE Trans. Power Electron., vol. 8, no. 4, pp. 546–553, 1993. https://doi: 10.1109/63.261026.

[2] Dong- Choon Lee and G-Myoung Lee "A novel overmodulation technique for space-vector PWM inverters," IEEE Trans. Power Electron., vol. 13, no. 6, pp. 1144–1151, 1998. https:// doi: 10.1109/63.728341.

[3] T.M. Jahns, "Flux-Weakening Regime Operation of an Interior Permanent-Magnet Synchronous Motor Drive," IEEE Trans. Ind. Appl. vol. IA-23, no. 4, pp. 681–689, 1987. https://doi: 10.1109/TIA.1987.4504966.

[4] Xu Hu, Peng Lu, Yan Yang, Chunpeng Pan, Guoguo Wu, and Yan Ping, "Design of DC motor controller based on MBD," Journal of Physics: Conference Series 2125, 2021. doi:10.1088/1742-6596/2125/1/012069.

[5] T. Ahmad, "Design and Construction of DC Motor Speed Controller Uisng Android," Department of EEE, World Iniversity of Bangladesh, 2022.

[6] N. Bacac, V. Slukic, M. Puškarić, B. Stih, E. Kamenar and S. Zelenika, "Comparison of different DC motor positioning control algorithms", 2014 37th International Convention on Information and Communication Technology Electronics and Microelectronics (MIPRO), pp. 1654-1659, 2014

[7] Elias N, Yahya NM, "Comparison of DC motor position control simulation using MABSA-FLC and PSO-FLC," In: 15th international colloquium on signal processing & its applications (CSPA), Penang, pp 39–42, 2019.

[8] A. Almawla, M J. Hussein, A T. Abdullah, "A Comparative Study of DC Motor Speed Control Techniques Using Fuzzy, SMC and PID," Journal Européen des Systèmes Automatisés, 2024. https://doi: 10.18280/jesa.570209.

[9] Jeen Ann Abraham, "DC Motor speed control using Machine Learning Algorithm," International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181, 2018.

[10] Xiaogang Wu, Minghao Zhou, Yafei Liang & Yujin Wang, "Review and Development of Electric Motor Systems and Electric Powertrains for New Energy Vehicles," Springer nature link, Open access, Volume 4, pages 3–22, 2021.

[11] Jawad Faiz & Farbod Parvin, "Trends and Technical Advancements on High-Efficiency Electric Motors: A Review," Springer nature link, Open access, pp 81–95, 2022.

[12] Infineon, "2EDL05 family Datasheet," 2022. http://www.infineon.com/gdHalfBridge.

[13] Xu Hu et al, [2021]. Design of DC motor controller based on MBD. Journal of Physics: Conference Series.

[14] Katshuhiko Ogata, "Modern Control Engineering", Prentice Hall PTR, Upper Saddle River, NJ, United States, 2001.

THIẾT KẾ BỘ ĐIỀU KHIỂN ĐỘNG CƠ MỘT CHIỀU DỰA TRÊN VI ĐIỀU KHIỂN ARM-M3

Tóm tắt: Bài báo này nhằm thiết kế một chiến lược điều khiển cho động cơ điện một chiều (DC), bao gồm bộ điều khiển tốc độ và vị trí. Vi điều khiển ARM Cortex-M3 được sử dụng để phát triển bộ điều khiển, cho phép người dùng chuyển đổi sang chế độ điều khiển động cơ bước mà không cần thay đổi hệ thống. Hàm truyền của hệ thống được thiết lập thông qua việc xác định các thông số của động cơ trong chế độ hiệu chỉnh. Cấu trúc phần cứng của bộ điều khiển và các thuật toán điều khiển được trình bày chi tiết trong bài báo. Bộ điều khiển PID với các tổ hợp khác nhau giữa thành phần tỉ lệ, tích phân và vi phân được sử dụng nhằm xác định chiến lược tối ưu. Kết quả mô phỏng được so sánh với đầu ra của mô hình thực tế để kiểm chứng hiệu quả của hệ thống đề xuất và phương pháp nghiên cứu điều khiển động cơ DC. Nghiên cứu này đóng góp một kỹ thuật đơn giản trong việc thiết kế bộ điều khiển tốc độ và vị trí dựa trên các giá trị quá điều chỉnh, sai số tĩnh và thời gian đáp ứng. Ngoài ra, giải pháp sử dụng ARM-M3 giúp giảm chi phí bộ điều khiển động cơ cho các ứng dụng công suất thấp.

Từ khóa: Động cơ một chiều, điều khiển vòng kín, Bộ điều khiển tốc độ, Bộ điều khiển vị trí, Bộ điều khiển PID.

Dr. Pham Thi Thanh Loan earned her Bachelor's, Master's, and Ph.D. degrees in Control Engineering and Automation in 2003, 2005, and 2015, respectively, from Hanoi University of Science and Technology (HUST). In 2003, she worked as an Automation Engineer at the High-Tech Center – Institute of Machinery and Industrial Equipment. Since 2004, Dr. Loan has been a lecturer at the Automation Department, Faculty of Electromechanics, Hanoi University of Mining and Geology. Her research interests include process control, predictive control, modeling, renewable energy and artificial intelligence (AI).

Dr. Dao Hieu earned his Bachelor's, Master's, and Ph.D. degrees in Control Engineering and Automation in 2005, 2009, and 2023, respectively, from Hanoi University of Mining and Geology (HUMG). From 2005 to 2007, he worked as an Automation Engineer in various industrial plants. Since 2008, Dr. Hieu has been a lecturer at the Automation Department, Faculty of Electromechanics, Hanoi University of Mining and Geology. His research interests include renewable energy, intelligent control systems and IoT.

.