Position Control of an Experimental Three-Degree-of-Freedom Actuated Articulated Robot Arm utilizing PID Controller

Dinh-Hieu Vo [1,*], Nam-Chau Le [2], Thi-Y-Nhi Nguyen [3], Tran-Phuong Huynh [4], Nhat-Truong Huynh [5], Kim-Huy Tran [6], Ba-Chinh Nguyen [7], Truong-Giang Do [8], Gia-Huy Tran [9], Van-Dong-Hai Nguyen [10]
[1, 3, 4, 5, 6, 7, 8, 9, 10] Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology and Education (HCMUTE), Ho Chi Minh City (HCMC), Vietnam
[2] Faculty of Economics, Ho Chi Minh City University of Technology and Education (HCMUTE), Ho Chi Minh City (HCMC), Vietnam
Email: [1] 22151084@student.hcmute.edu.vn, [2] 22151054@student.hcmute.edu.vn, [3] 21125252@student.hcmute.edu.vn, [4] 21142153@student.hcmute.edu.vn, [5] 21142199@student.hcmute.edu.vn, [6] 21142110@student.hcmute.edu.vn, [7] 21142498@student.hcmute.edu.vn, [8] 20151355@student.hcmute.edu.vn, [9] 21151109@student.hcmute.edu.vn, [10] hainvd@hcmute.edu.vn
*Corresponding Author

***Abstract*—In this paper, we present an experimental three-degree-of-freedom (3-DOF) articulated robot arm. A hardware setup using Arduino – a popular and cheap processor- is utilized to make an experimental model suitable for a small laboratory. PID control is applied to position control for our robot for set-point positions. The experimental results prove that PID controller is suitable for this real model. Besides, calibration of PID parameters also proves that the experimental PID suits the PID theory. Thence, this research is a reference for individuals who are implementing a real model for a small laboratory.**

Keywords*—*PID Control; 3-DOF Robot Arm; Manipulator; Arduino

I. INTRODUCTION

Robot arms are popular models in both academic and industry. Their popularity makes them a criterion for standardizing them [1]. However, the flexibility in designing and controlling them still stimulates research. In [2][3], 2-DOF robot arms are presented. PID control [2] and LQR control [3] are used successfully for these robot arms. With 2-DOF, robot arms cannot work in 3D space. Because of the requirement of developing a mechanical structure, other links are added to create a 3-DOF robot arm. A design is shown in [4]. Applications of the implementation for PLC are suggested in [5] and [6]. However, using PLC, there is no algorithm in control and PLC is so expensive for students to practice. The stability of PLC is based on hardware technology. In our opinion, with only simple hardware, the development of the algorithm can be used to increase the quality control of the system. In [7], kinetic inverse and forward calculation of 3-DOF robot arm are done, and a simulation of using PID control for 3-DOF robot arm is proved, but no experiment is described. The need to apply algorithms in hardware still exists.

The development of Arduino opens a direction for cheap and easy-to-make robot models for the laboratory [8]. Thence, building a 3-DOF robot arm based on Arduino is a suitable direction for widening cheap models for kinds of people for training and research.

Besides many flexible methods for robots, such as sliding control [9] or intelligent control [10], PID controller, which is a single loop controller introduced in 1940 [11], is still the most popular due to its simple structure. Its application is shown in [12] for chemical process control calibration and in [13] for temperature control. In the laboratory, this control method is popularly used to a DC motor [14]. Using DC motors to apply for other kinds of systems, such as a submarine model [15]. However, this is only a simple single input-multi output (SISO) system. Thence, the challenge in control is limited. A more complicated multi-input-multi-output (MIMO) system, such as a 3-DOF robot arm, gives a better solution for increasing the challenge in training control theory for students in the laboratory.

In this research, we develop an experiment for a 3-DOF articulated robot arm based on our real model in HCMUTE. Hardware platform is described and proved through PID control for set-point position. Calibration is tested to confirm its suitability to theory. Thence, our research is a reference for the idea of building a simple and effective model for a control robot arm for laboratories.

II. MODEL

Mathematical structure of 3-DOF robot arm is shown in Fig. 1.This robot imitates a human arm, designed to perform basic actions through the integration of DC motor (Fig. 2) and Matlab/Simulink for control in section III.B. The First motor is used for right and left motion. Second and third motors are used for the elevation of the arm [2]. Forward kinematics analysis is used to compute the location and orientation of each joint in the robotic arm.

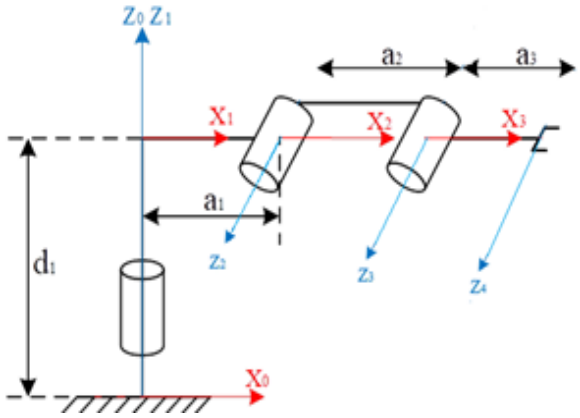


Fig. 1. Kinematic Frame of 3-DOF robot arm

Fig. 2. DC motor

In this platform, 12V DC sources are used to supply all DC motors and the Arduino. In an experimental model, these three DC motors get six wires. Red and black colors are the two wires of the motor power. Green color is GND of encoder. Blue color is VCC of encoder (3.5 20V). Yellow and white colors are the output of A and B of the encoder. Motor is controlled by PWM signals [2].

Structure of PID control is shown in Fig. 3. PID controller in governor designs appeared in the 1890s [16]. It was later developed in automatic ship steering systems. PID is a classical method of automation and control theory. It is popular due to its easy calibration of trial control mode [1].

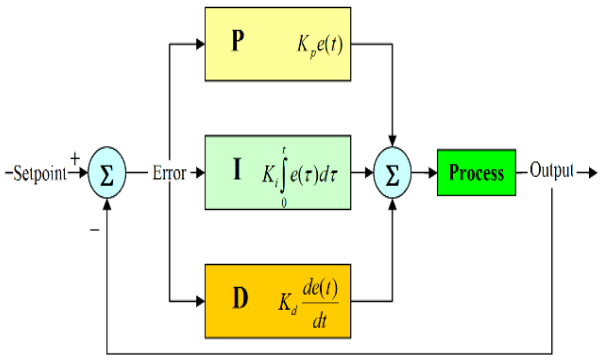


Fig. 3. Block diagram of a PID controller [6]

Three main components of PID control are P, I, D. P determines reaction to current error, I determines reaction to the sum of recently appeared errors, D determines reaction according to the rate of error changing. Sum of these three parts contributes to the control mechanism, such as speed control of a motor, in which P value depends upon current error, I also depend on the accumulation of previous error, and D predicts future error based on the current rate of change [6]. Theory of calibration of PID parameters is shown in Table 1.

Table 1. Manual P, I, and D parameters tuning

Gain	Rising time	Over-shoot	Settling time	Error Steady-State
Increasing K_p	Decrease	Increase	Tiny increase	Decrease
Increasing K_i	Tiny decrease	Increase	Increase	Large decrease
Increasing K_d	Tiny decrease	Decrease	Decrease	No change

PID formula blocks are described by formula (1) to formula (4)

$$U_P(t) = K_p \cdot e(t) \tag{1}$$

$$U_I(t) = K_I \int_0^t e(t)dt \tag{2}$$

$$U_D(t) = K_D \frac{d}{dt} e(t) \tag{3}$$

$$U(t) = K_p \cdot e(t) + K_I \int_0^t e(t)dt + K_D \frac{d}{dt} e(t) \tag{4}$$

III. Experiment

A. Hardware

The robot's movements are intended to mimic those of a human arm. The section describes 3D modeling, hardware using SolidWorks 2022 (Fig. 4). Fig. 5 and Fig. 6 illustrate the overall structure of the three-dimensional real experimental model. All of the robotic arm's links are individually designed using software, allowing for precise customization and modularity. This approach enables the arm to attach various screwdriver bits to the end effector, making it versatile for tasks like removing multiple screws in different configurations. Once the individual components are designed, they are assembled into a cohesive system, enhancing the understanding of the arm's mechanics and functionality. This modular design not only improves operational efficiency but also simplifies maintenance and upgrades.

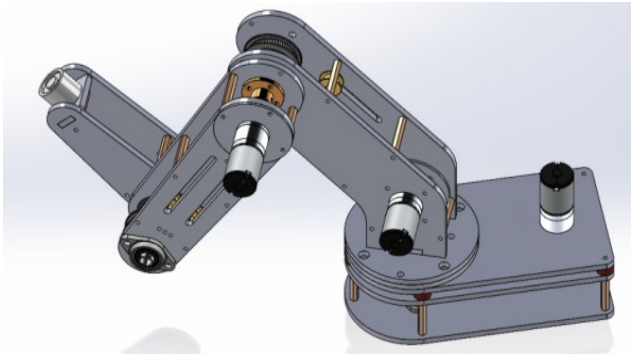


Fig. 4. Designed 3D model Architecture of 3-DOF robotic arm

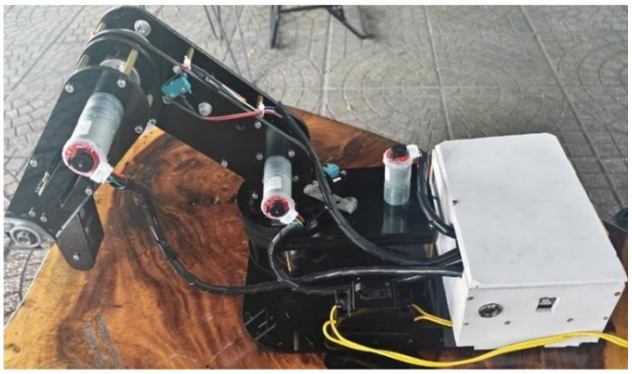


Fig. 5. One side of a realistic full robot

Fig. 6. The other side of a realistic full robot

In this project, the robot was designed using SolidWorks 2022 software. Thence, an experimental model is shown in Fig. 5 and Fig. 6. This model is then used to perform simulations according to the block diagram in Fig. 7. And, the total structure of connection is shown in Fig. 8.

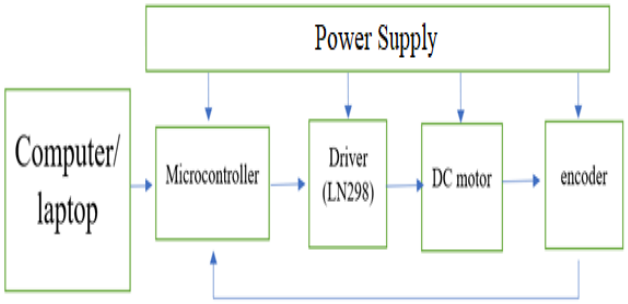


Fig. 7. General block diagram of hardware of robot arm

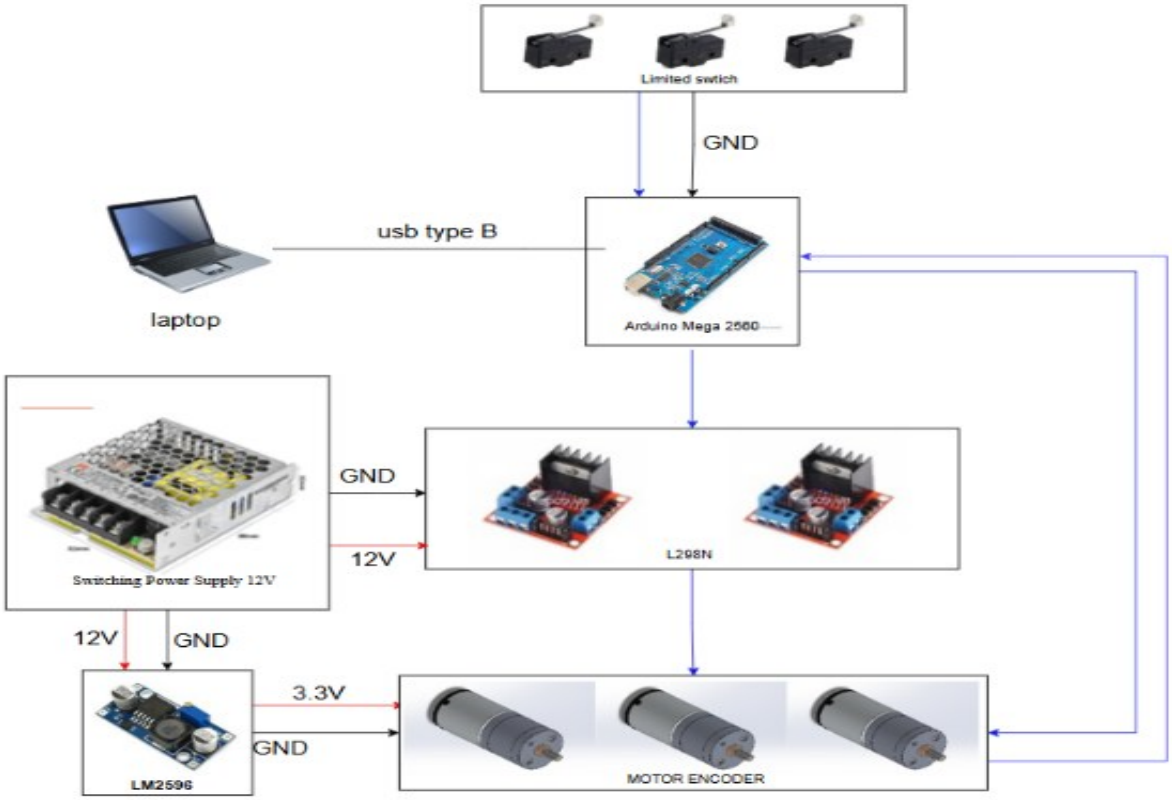


Fig. 8. System structure of hardware

- **Power Supply Block**: Provides DC power to the components that require it.
- **PC/Laptop Block**: Sends control signals and receives feedback signals from the microcontroller.
- **Microcontroller Block**: Executes algorithms to control the robot and processes sensor signals.
- **Driver Block**: Controls the motors.
- **Encoder Block**: Reads signals from sensors and sends them to the microcontroller for processing.

B. Programm

Fig. 9 represents the simulation of the control algorithms using MATLAB/Simulink software 2023B.

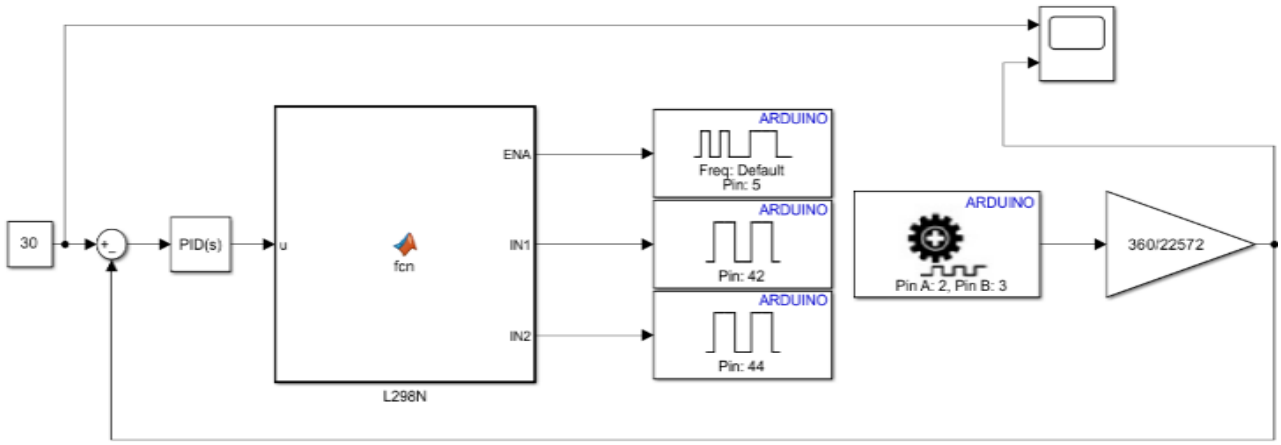


Fig. 9. MATLAB program to control each link of the robot arm

Encoder block (Fig. 10) outputs tick count from a quadrature encoder on a rotating motor connected to an Arduino board. Every increment in tick count of the encoder indicates that the motor is rotating clockwise. Every decrement in tick count of encoder indicates that the motor is rotating counterclockwise. Total tick count represents the incremental position of the rotating motor [17]. PWM block (Fig. 11) generates square pulses of varying duty cycle depending on input value sent to block on the Arduino hardware pin. This block enables a digital output to provide a range of different power levels, similar to that of an analog output [17].

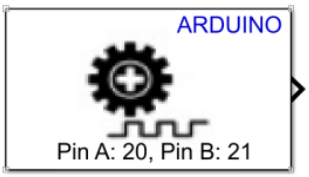


Fig. 10. Encoder block

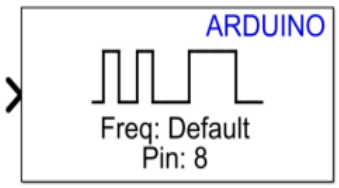


Fig. 11. PWM block

C. Experiment Result

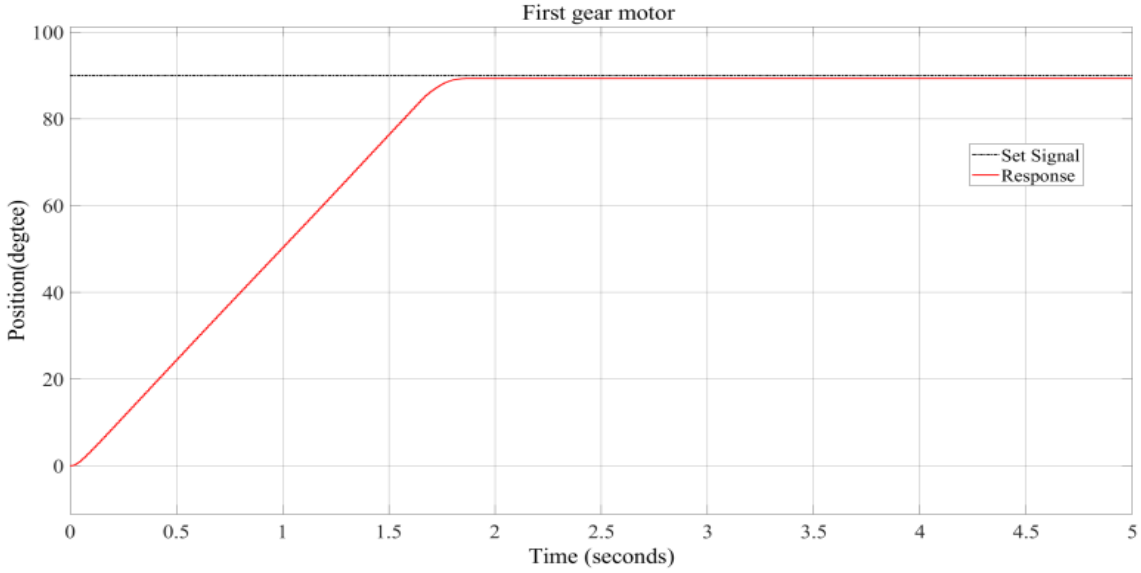
Based on the experimental model, the embedded program is based on Simulink using MATLAB 2023b software. The program, as shown in Fig. 15, includes blocks for reading output values, the main program, and the PID. The PID control parameters are shown in Table 2, with sampling time of 0.001s.

Table 2. The parameters of the PID controller

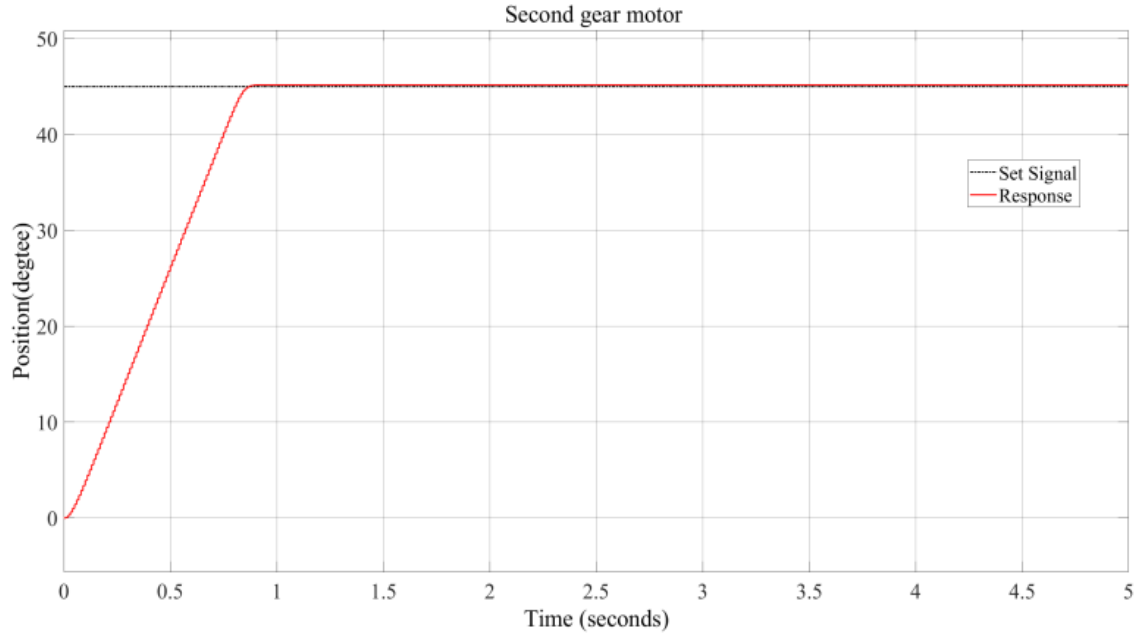
DC motor	K_p	K_i	K_d
1	108	0.01	4.5
2	108	0.01	4.5
3	108	0.01	4.5

Fig. 12 show the response graphs of rotational joints 1, 2, and 3, respectively. The set angles are represented by the black dashed line, while the response angles are shown by the red line.

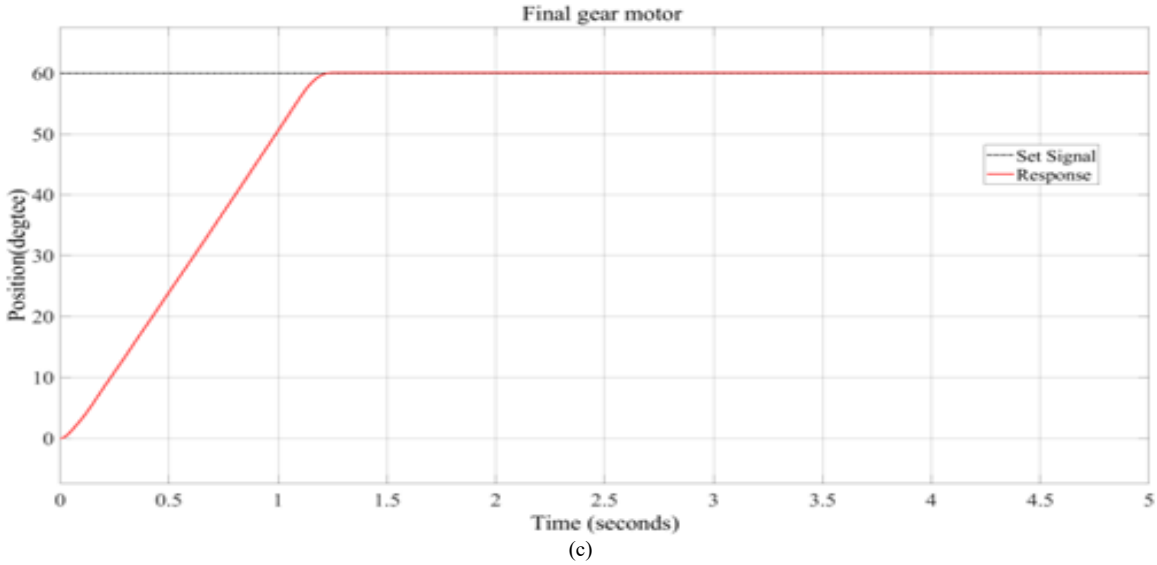
From control parameters in Table 3, response of the system is shown in Fig. 13. It is observed that if K_p is low, the output response does not reach the set angle value. Conversely, if K_p is too high, it causes overshoot. The set angles are represented by the black dashed line, while the red, blue, and purple lines show the response graphs of the first rotational joint when K_p is 10, 40, and 500, respectively. It can be seen that when K_p = 10 and 40, the steady-state errors are approximately 7 and 2, respectively. On the other hand, when K_p = 500, there is a small overshoot, and steady-state error is zero.



(a)



(b)



(c)

Fig. 12. The output responses of the robotic arm are shown for (a) joint 1, (b) joint 2, and (c) joint 3

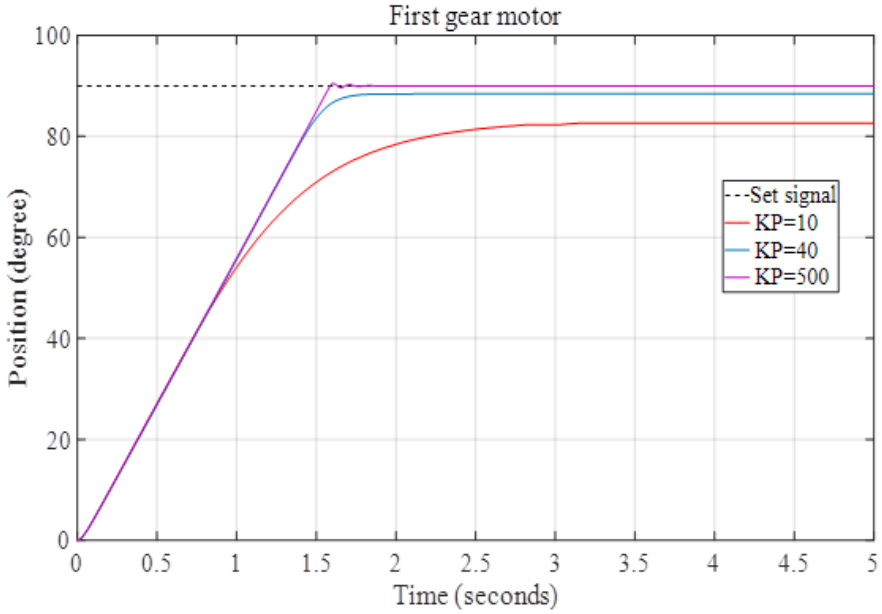


Fig. 13. The output responses of the robotic arm under K_p=10, K_p=40, and K_p=500.

Table 3. The parameters of the PID controller

Case	K_p	K_i	K_d
1	10	0.01	4.5
2	40	0.01	4.5
3	500	0.01	4.5

From control parameters in Table 4, response of the system is shown in Fig. 14. It is observed that as K_d increases, both overshoot and settling time decrease. The set angles are represented by black dashed line, while the red, blue, and purple lines show the response graphs of the first rotational joint when K_d is increased from 0.5 to 2 and 8. It can be observed that when K_d = 0.5 and 2, the overshoot is greater than zero and gradually decreases as K_d increases.

Table 4. The parameters of the PID controller

Case	K_p	K_i	K_d
1	108	0.01	0.5
2	108	0.01	2
3	108	0.01	8

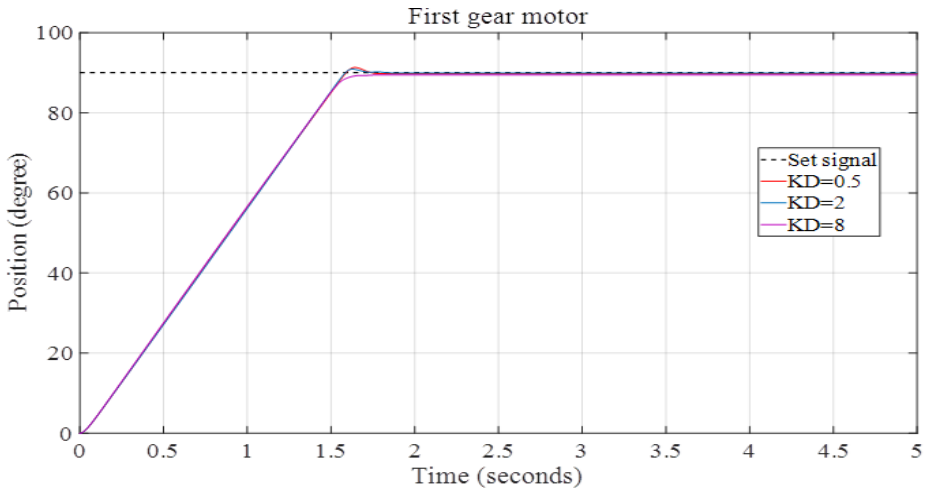


Fig. 14. The output responses of the robotic arm are shown K_d=0.5, K_d=2, and K_d=8

From control parameters in Table 5, response of the system is shown in Fig. 15. It is observed that as K_i increases, the output value exceeds the set value. The set angles are represented by the black dashed line, while the red, green, and purple lines show the response graphs of the first rotational joint when K_i is increased from 1 to 3 and 6. It can be seen that when K_i exceeds the value of 1, the steady-state error becomes negative.

Table 5. The parameters of the PID controller

Case	K_p	K_i	K_d
1	108	1	4.8
2	108	2	4.8
3	108	3	4.8

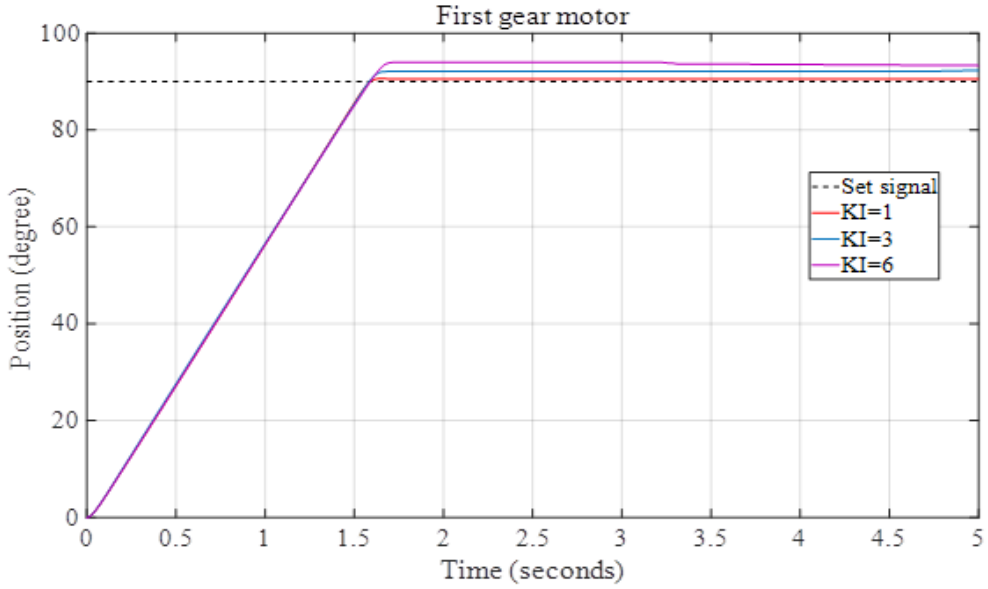


Fig. 15. The output responses of the robotic arm are shown K_i=0.5, K_i=2, and K_i=6

IV. CONCLUSION

In this paper, we focus on position control for a 3-DOF articulated robot arm at set-point positions. This research implements for the case that inverse and reverse kinetic calculations were provided. We present a hardware platform for this model. In our research, a MATLAB-embedded program for Arduino is shown. Through this experimental system, PID control for MIMO system is utilized and proved to control well each link at set-point positions. Also, PID parameters adjustments are operated to test the PID calibration rule. Thence, through adjustment, K_p is confirmed to decrease settling error and shorten the settling time and increase overshoot; K_d in this case is confirmed to increase overshoot; K_i in this case is confirmed to make the system more vibrate. The K_p calibration suits the theory of PID method. In experimental, increasing K_i gives more overshoot. It suits the theory. However, the mechanical part of DC motor makes the system delayed. Thence, it takes more time for the system to be stabilized at set-point. The characteristics suit the theory, but the simulation time should be longer for a better view. Bigger value of K_d makes overshoot smaller. And, this main characteristic suits the PID theory.

ACKNOWLEDGEMENT

This paper belongs to the project for students of HCMUTE in year 2025 and it is funded by HCMUTE. We, authors, want to give thanks for that support. Video link of operation of system is: https://www.youtube.com/watch?v=IA029uSBq-Q

REFERENCES

[1] A. J. Vertut and A. Liégeois, "General design criteria for manipulators," *Mechanism and Machine Theory*, vol. 16, no. 1, pp. 65-70, 1981, https://doi.org/10.1016/0094-114X(81)90052-5.

[2] N. Nurnuansuwan *et al*., "A Prototyping of 2-DOF Robot Arm Using Feedback Control System," *5th International Conference on Control, Automation and Robotics (ICCAR)*, pp. 368-372, 2019, https://doi.org/10.1109/ICCAR.2019.8813726.

[3] S. M. Mahil and A. Al-Durra, "Modeling analysis and simulation of 2-DOF robotic manipulator," *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1-4, 2016, https://doi.org/10.1109/MWSCAS.2016.7870099.

[4] Ramish, S. B. Hussain and F. Kanwal, "Design of a 3 DoF robotic arm," *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, pp. 145-149, 2016, https://doi.org/10.1109/INTECH.2016.7845007.

[5] I. S. Karem, T. A. J. Wahab, and M. J. Yahyh, "Design and implementation for 3-DOF SCARA Robot based PL," *Al-Khwarizmi engineering journal*, vol. 13, no. 2, pp. 40-50, 2017, https://doi.org/10.22153/kej.2017.01.002.

[6] V. D. Cong, "Design and development of a cost-efficiency robot arm with plc-based robot controller," *FME Transactions*, vol. 52, no. 2, pp. 226-236, 2024, https://doi.org/10.5937/fme2402226C.

[7] W. W. Naing, M. Thanlyin, K. Z. Aung, and A. Thike, "Position control of 3-dof articulated robot arm using pid controller," *International Journal of Science and Engineering Applications*, vol. 7, no. 9, pp. 254-260, 2018, https://doi.org/10.7753/IJSEA0709.1001.

[8] Y. A. Badamasi, "The working principle of an Arduino," *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*, pp. 1-4, 2014, https://doi.org/10.1109/ICECCO.2014.6997578.

[9] C. Fallaha, M. Saad and H. Kanaan, "Sliding mode control with exponential reaching law applied on a 3 DOF modular robot arm," *2007 European Control Conference (ECC)*, Kos, Greece, 2007, pp. 4925-4931, 2007, https://doi.org/10.23919/ECC.2007.7068438.

[10] J. Kern, D. Marrero, and C. Urrea, "Fuzzy control strategies development for a 3-DoF robotic manipulator in trajectory tracking," *Processes*, vol. 11, no. 12, p. 3267, 2023, https://doi.org/10.3390/pr11123267.

[11] S. Bennett, "The past of PID controllers," *Annual reviews in control*, vol. 25, pp. 43-53, 2001, https://doi.org/10.1016/S1367-5788(01)00005-0.

[12] D. E. Seborg, T. F. Edgar, D. A. Mellichamp, and F. J. Doyle III. *Process dynamics and control*. John Wiley & Sons. 2016. https://books.google.co.id/books?hl=id&lr=&id=ZZVFEAAAQBAJ.

[13] N. Mehta, D. Chauhan, S. Patel, and S. Mistry, "Design of HMI based on PID Control of Temperature," *International Journal of Engineering Research and*, vol. 6, p. 05, 2017, https://doi.org/10.17577/IJERTV6IS050074.

[14] F. R. Septiawan, A. R. Al Tahtawi, and S. M. Ilman, "Control of Bidirectional DC-DC Converter with Proportional Integral

Derivative," *Journal of Fuzzy Systems and Control*, vol. 2, no. 3, pp. 164-169, 2024, https://doi.org/10.59247/jfsc.v2i3.241.

[15] Q. Ariyansyah and A. Ma'arif, "DC motor speed control with proportional integral derivative (PID) control on the prototype of a mini-submarine," *Journal of Fuzzy Systems and Control*, vol. 1, no. 1, pp. 18-24, 2023, https://doi.org/10.59247/jfsc.v1i1.26.

[16] S. Bennett, "Nicholas Minorsky and the automatic steering of ships," in *IEEE Control Systems Magazine*, vol. 4, no. 4, pp. 10-15, 1984, https://doi.org/10.1109/MCS.1984.1104827.

[17] M. Paluszek and S. Thomas. *MATLAB machine learning*. Apress.2016. https://books.google.co.id/books?hl=id&lr=&id=3kXODQAAQBAJ.