NGHIÊN CỨU THIẾT KẾ CHẾ TẠO HỆ THỐNG GIÁM SÁT ĐIỀU KHIỂN CÁNH TAY ROBOT TÍCH HỢP IoT TRONG CÔNG NGHIỆP

RESEARCH ON DESIGN AND DEVELOPMENT OF AN IoT-INTEGRATED ROBOT ARM MONITORING AND CONTROL SYSTEM IN INDUSTRY

Phạm Trung Thiên, Đỗ Văn Tỉnh, Nguyễn Ngọc Thể, Đặng Huy Hoàn

Khoa Cơ khi, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp

Đến Tòa soạn ngày 15/3/2025, chấp nhận đăng ngày 15/4/2025

Tóm tắt: Xu hướng hiện nay tập trung vào việc kết hợp các phương pháp điều khiển thông minh, bao gồm mạng nơron nhân tạo, học tăng cường và các thuật toán điều khiển thích nghi, nhằm nâng cao tính chính xác và hiệu quả vận hành của cánh tay robot 6 bậc tự do. Đồng thời, việc ứng dụng công nghệ IoT và hệ thống SCADA trong giám sát từ xa giúp tối ưu hóa quy trình điều khiển và bảo trì hệ thống, mở ra nhiều tiềm năng ứng dụng trong công nghiệp 4.0. Bài báo này tập trung vào xây dựng hệ thống điều khiển và giám sát cánh tay robot 06 bậc tự do ứng dụng công nghệ IoT truyền thông công nghiệp. Hệ thống có thể giám sát theo thời gian thực bằng kết nối không dây hiển thị trạng thái lên màn hình giám sát và điều khiển các tác vụ của robot chính xác. Chương trình điều khiển được thiết kế trên nền tảng bộ điều khiển lập trình khả dĩ logic PLC và truyền thông qua module IoT 2040 đáp ứng ngay tức thì với độ chính xác điều khiển lên tới 0,5 mm khi thao tác trên vật kích thước 100 x 100 x 30 mm có trọng lượng 300 gram.

Từ khóa: Cánh tay robot 06 bậc tự do, công nghệ IoT, điều khiển giám sát.

Abstract: The current trend focuses on integrating intelligent control methods, including artificial neural networks, reinforcement learning, and adaptive control algorithms, to enhance the accuracy and operational efficiency of the 6-degree-of-freedom (6-DOF) robotic arm. At the same time, the application of IoT technology and SCADA systems for remote monitoring optimizes control processes and system maintenance, unlocking significant potential for Industry 4.0 applications. This paper focuses on developing a 6-DOF robotic arm control and monitoring system utilizing industrial IoT communication technology. The system enables real-time monitoring via wireless connectivity, displaying the robot's status on a monitoring screen while ensuring precise task execution. The control program is designed on a Programmable Logic Controller (PLC) platform and communicates through an IoT 2040 module, providing immediate responsiveness with a control accuracy of up to 0.5 mm when handling objects of 100mm × 100mm × 30mm in size and 300 grams in weight.

Keywords: (6-DOF) Robotic Arm, IoT Technology, Control and Monitoring.

1. ĐẶT VẤN ĐỀ

Cánh tay robot 6 bậc tự do (6-DOF robotic arm) là một trong những nền tảng tự động hóa then chốt, giữ vai trò không thể thiếu trong nhiều lĩnh vực. Với khả năng mô phỏng chuyển động linh hoạt của cánh tay con người, loại robot này mở ra tiềm năng ứng dụng vượt trội trong những nhiệm vụ đòi hỏi độ chính xác và sự khéo léo cao. Nghiên cứu về điều khiển và giám sát cánh tay robot 6 bậc

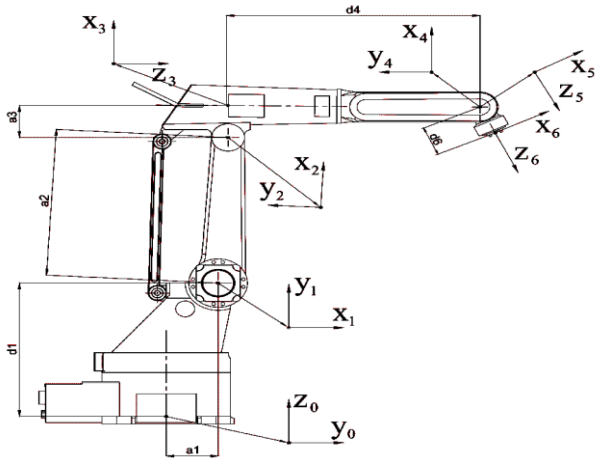
tự do hiện đang là một hướng đi trọng tâm của ngành kỹ thuật điều khiển và tự động hóa. Mục tiêu là không chỉ nâng cao hiệu suất làm việc mà còn tăng cường độ chính xác, độ ổn định và khả năng thích ứng linh hoạt của robot trong môi trường vận hành thực tế ngày càng phức tạp. Trên thế giới, điều khiển và giám sát cánh tay robot 6 bậc tự do đã được nghiên cứu sâu rộng với nhiều phương pháp tiên tiến. Siciliano và cộng sự [9] đã trình bày tổng quan về điều khiển động học và động lực học của cánh tay robot, trong đó nhấn mạnh phương pháp điều khiển tối ưu dựa trên động lực học ngược. Một nghiên cứu khác của Wang et al. [10] đã áp dụng trí tuệ nhân tạo để điều khiển cánh tay robot trong môi trường không xác định, giúp hệ thống thích nghi linh hoạt với các điều kiện làm việc khác nhau. Ngoài ra, hệ thống giám sát dựa trên thị giác máy tính kết hợp với Deep Learning đã được đề xuất bởi Hongqian Deng và các cộng sự [11] giúp nâng cao khả năng nhận diện và kiểm soát các thao tác của robot trong môi trường sản xuất công nghiệp.

Tại Việt Nam, nghiên cứu về điều khiển cánh tay robot 6 bậc tự do đã có nhiều tiến bộ đáng kể. Huỳnh Thế Hiển và cộng sự [1] đã nghiên cứu thuật toán điều khiển phản hồi bám quỹ đạo sử dụng bộ điều khiển PID kết hợp với mạng nơron nhân tạo nhằm nâng cao độ chính xác khi cánh tay robot thực hiện các tác vụ phức tạp. Trong khi đó, Vũ Thị Yến và cộng sự [2] đã triển khai phương pháp điều khiển trượt thích nghi để nâng cao độ ổn định và giảm sai số vị trí trong môi trường có nhiễu. Một số nghiên cứu khác cũng đã tập trung vào xây dựng hệ thống giám sát dựa trên công nghệ IoT và SCADA để theo dõi hoạt động của cánh tay robot theo thời gian thực của nhóm tác giả Đỗ Trí Nhựt [3]. Tổng hợp các nghiên cứu trong và ngoài nước cho thấy, việc điều khiển và giám sát cánh tay robot 6 bậc tự do không chỉ là một bài toán kỹ thuật mà còn liên quan đến nhiều lĩnh vực như trí tuệ nhân tạo, tự động hóa và hệ thống điều khiển tiên tiến. Bài báo tập trung xây dựng hệ thống điều khiển giám sát IoT công nghiệp để tích hợp cánh tay robot vào dây truyền đáp ứng công nghệ 4.0.

2. BÀI TOÁN ĐỘNG HỌC

2.1. Động học thuận

Hệ thống từ mô hình robot 6 bậc tự do, gán hệ tọa độ lên từng khâu của robot theo phương pháp D-H.



Hình 1. Hệ tọa độ gắn theo bảng D-H

Bảng 1. Bảng D-H

Joint	θ_i (rad)	d_i (m)	a_i (m)	α_i (rad)
1	θ_1	d1	a1	$\pi/2$
2	θ_2	0	a2	0
3	θ_3	0	a3	$\pi/2$
4	θ_4	d4	0	$-\pi/2$
5	θ_5	0	0	$\pi/2$
6	θ_6	d6	0	0

Ta nhận được các ma trận truyền biến đổi tọa độ ${}^{i-1}A_i$:

$${}^0A_1=\begin{bmatrix} c_1 & 0 & s_1 & a1c_1 \\ s_1 & 0 & -c_1 & a1s_1 \\ 0 & 1 & 0 & d1 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^1A_2=\begin{bmatrix} c_2 & -s_2 & 0 & a2c_2 \\ s_2 & c_2 & 0 & a2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3=\begin{bmatrix} c_3 & 0 & s_3 & a3c_3 \\ s_3 & 0 & -c_3 & a3s_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^3A_4=\begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & d4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4A_5=\begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^5A_6=\begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ma trận truyền biến đổi tọa độ thuần nhất của khâu cuối là:

$${}^0A_6(q)={}^0A_1\,{}^1A_2\,{}^2A_3\,{}^3A_4\,{}^4A_5\,{}^5A_6 \quad (1)$$

- Vị trí của điểm tác động cuối E:

$$\begin{cases} x_E = A14 \\ y_E = A24 \\ z_E = A34 \end{cases} \quad (2)$$

- Hướng của khâu tác động cuối EE:

$$\begin{cases} \sin\beta = A13 \\ \cos\beta = \pm\sqrt{1-\sin^2\beta} \\ \sin\alpha = -\dfrac{A23}{\cos\beta} \\ \cos\alpha = \dfrac{A33}{\cos\beta} \\ \sin\eta = -\dfrac{A12}{\cos\beta} \\ \cos\eta = \dfrac{A11}{\cos\beta} \end{cases} \quad (3)$$

2.2. Bài toán động học ngược

$${}^0A_1\,{}^1A_2\,{}^2A_3\,{}^3A_4\,{}^4A_5\,{}^5A_6=T_6 \quad (4)$$

$${}^0A_6(t)=T_6=\begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Trong đó $R^*=\begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}$ là ma trận chỉ hướng của điểm tác động cuối;

$r^*=\begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T$ là ma trận vị trí của điểm tác động cuối.

$$\begin{aligned} T_6 &= {}^0A_1\,{}^1A_2\,{}^2A_3 T(z_3,d_4) \\ &\times R(z_3,\theta_4)R(x_4,\alpha_4) \\ &\times R(z_4,\theta_5)R(x_5,\alpha_5) \\ &\times R(z_5,\theta_6)T(z_5,d_6) \end{aligned} \quad (5)$$

3. THIẾT KẾ HỆ THỐNG

3.1. Nguyên lý hoạt động



Hình 2. Sơ đồ mô tả hệ thống điều khiển - giám sát tích hợp IOT

Hệ thống trong sơ đồ khối trên là một hệ thống điều khiển và giám sát từ xa dựa trên IoT (Internet of Things), sử dụng PLC, HMI, Node-RED và Webserver để quản lý và vận hành thiết bị. Hệ thống bao gồm 4 khối chính: PLC (Programmable Logic Controller) là bộ

điều khiển chính, chịu trách nhiệm xử lý tín hiệu và điều khiển hoạt động của hệ thống; HMI (Human Machine Interface) cung cấp giao diện trực quan để người vận hành giám sát và điều khiển hệ thống; các tín hiệu từ cảm biến và thiết bị đầu cuối sẽ được gửi về PLC để xử lý, sau đó hiển thị trên HMI cho người dùng.

Giao diện giám sát điều khiển trực tuyến trên Web để có thể giám sát và điều khiển từ xa, hệ thống sẽ thiết kế một giao diện web cho phép người dùng theo dõi và thao tác trên hệ thống. Dữ liệu từ PLC sẽ được thu thập và hiển thị theo thời gian thực thông qua công nghệ IoT.

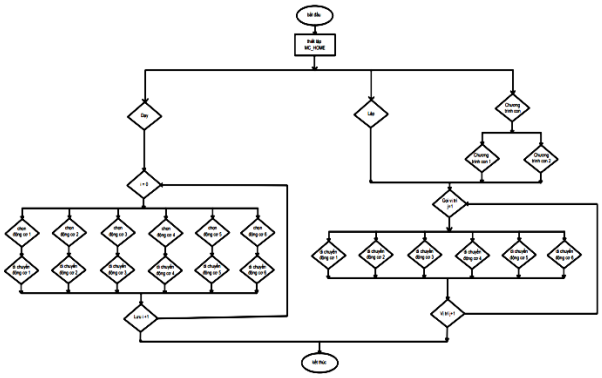
Lập trình giao diện Node-RED là một công cụ lập trình trực quan giúp kết nối dữ liệu từ các thiết bị công nghiệp với hệ thống giám sát. Webserver đóng vai trò trung gian, giúp người dùng có thể truy cập hệ thống từ bất kỳ đâu thông qua internet. Node-RED được sử dụng để thiết lập luồng dữ liệu giữa PLC, webserver và giao diện giám sát.

Truyền thông giám sát (Node-RED - PLC, Webserver - Node-RED) thông qua giao thức truyền thông Modbus TCP. Dữ liệu này sẽ được xử lý và gửi đến Webserver. Webserver đóng vai trò lưu trữ và hiển thị dữ liệu từ Node-RED lên giao diện web. Người dùng có thể gửi lệnh điều khiển từ Webserver đến Node-RED, sau đó chuyển tiếp đến PLC để thực hiện các thao tác.

3.2. Thiết kế hệ thống điều khiển PLC-HMI

Thiết lập vị trí gốc cho robot, sau đó chọn chế độ của chương trình (chương trình con hoặc chương trình auto).

Dạy học cho robot: bấm chọn động cơ cần dạy học (dc 1-2-3-4-5-6), Tín hiệu từ phím điều khiển gửi qua PLC xuống động cơ chạy khối lênh MoveJog để điều khiển các động cơ vừa chọn đến vị trí mong muốn. Khi có lệnh lưu các vị trí từ bộ điều khiển, các giá trị hiện tại góc quay của động cơ sẽ lưu vào bộ nhớ theo các mảng data.



Hình 3. Sơ đồ thuật toán lập trình điều khiển

Khi có lệnh điều khiển chạy chương trình auto, tín hiệu điều khiển từ bàn phím HMI hoặc giao diện điều khiển gửi xuống PLC bật khối MC_Power, PLC gọi bộ nhớ các mảng data đã lưu ra. Mỗi giá trị mảng data là một vị trí ứng với góc quay của 06 động cơ và khâu thao tác cuối. Khối MC_MoveAbsolute điều khiển động cơ quay đúng giá trị được gán với tốc độ được lập trình. Chương trình sẽ thực hiện chạy lặp đi lặp lại cử động robot được dạy cho đến khi dừng chương trình auto robot mới dừng hoạt động. Trong chương tình auto có hai hoặc nhiều chương trình con để robot có thể làm việc nhiều chương trình khác nhau khi được gọi thao tác.

3.3. Thiết kế giao diện giám sát



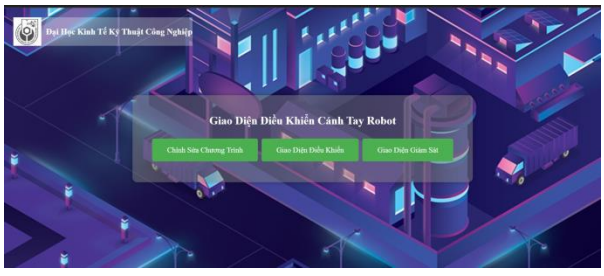
Hình 4. Giao diện người dùng trên Node-Red

Người dùng điều khiển từ xa thông qua giao diện Web. Webserver nhận lệnh và gửi đến Node-RED. Node-RED chuyển tiếp lệnh đến PLC để điều khiển thiết bị. PLC thực hiện điều khiển và cập nhật trạng thái hoạt động. Dữ liệu phản hồi được gửi về Webserver thông qua Node-RED. Người dùng theo dõi giám sát trạng thái hệ thống trên giao diện Web.



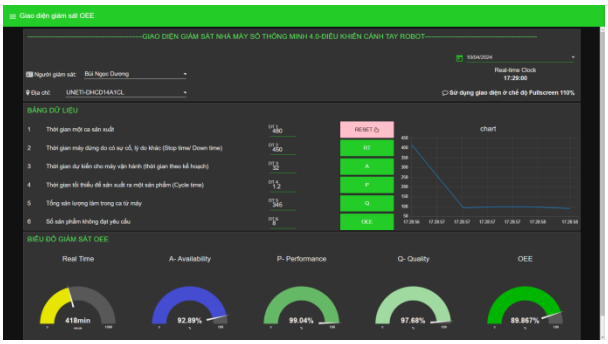
Hình 5. Giao diện diều khiển các động cơ

Web Server là máy chủ lưu trữ và phân phối dữ liệu web đến người dùng thông qua giao thức HTTP/HTTPS. Khi người dùng truy cập hệ thống từ xa, trình duyệt gửi yêu cầu (request) đến Web Server. Web Server xử lý yêu cầu, truy vấn dữ liệu từ Node-RED hoặc cơ sở dữ liệu, sau đó phản hồi kết quả về trình duyệt để hiển thị giao diện giám sát. Nếu người dùng gửi lệnh điều khiển, Web Server tiếp nhận, chuyển lệnh đến Node-RED, sau đó Node-RED gửi tín hiệu đến PLC để thực thi thao tác.

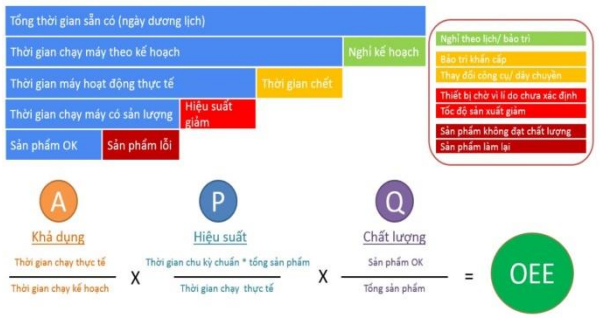


Hình 6. Giao diện Web server

Hệ thống tích hợp đánh giá hiệu suất tổng thể OEE (Overall Equipment Effectiveness) với thời gian, chất lượng và tốc độ vận hành qua đó đánh giá và đưa ra giải pháp để cải tiến hiệu quả.



Hình 7. Giao diện giám sát OEE



Hình 8. Phương thức tính toán OEE

4. KẾT QUẢ VÀ THẢO LUẬN

Xây dựng giao diện điều khiển giám sát bằng Node-RED Dashboard cho phép người dùng lập trình và thiết kế nên giao diện điều khiển giám sát riêng cho hệ thống thiết bị của mình. Giao diện trực quan và dễ sử dụng, giúp người dùng dễ dàng trong việc sử dụng các chức năng mà không cần phải viết các mã lệnh phức tạp, giúp tạo ra giao diện nhanh chóng và dễ dàng. Sử dụng các node chuẩn có sẵn trong Node- RED hoặc các node riêng được thêm từ các thư viện được Node- RED hỗ trợ cho phép người dùng thực hiện các công việc như đọc dữ liệu từ cảm biến, thu thập dữ liệu điều khiển từ các bộ điều khiển lập trình logic (PLC) được sử dụng trong hệ thống thiết bị của mình. Các node được sử dụng dạng kéo thả và liên kết với nhau để hình thành nên giao diện riêng cho người lập trình. Dựa trên bộ điều khiển trực tiếp từ màn hình HMI triển

khai giao diện điều khiển trực tuyến điều khiển hệ thống thiết bị một cách trực quan. Đồng thời giám sát quá trình vận hành của thiết bị. Dữ liệu được truyền thông giữa các thiết bị thông qua giao thức profinet sử dụng Ethernet/IP.

Sau 30 lần tiến hành thực nghiêm cho robot chạy tự động gắp vật từ vị trí A thả sang vị trí B theo quỹ đạo được dạy và vị trí được dạy. Kết quả thực nghiệm cho thấy robot hoạt động ổn định sai số vị trí thả phôi lệch so với trục x từ 1 đến 2mm và lệch góc nghiêng với trục ox 2^o. Tuy nhiên một số lần chạy có sự trễ giữa tín hiệu điều khiển được truyền từ giao diện điều khiển tới hệ thống thiết bị và ngược lại tín hiệu phản hồi từ thiết bị tới giao diện giám sát khoảng 01÷02 s. Lý do có sự trễ đến từ một số yếu tố:

- Độ trễ mạng (Network Latency): Điều này có thể do các yếu tố như kết nối mạng không ổn định, tải lưu lượng mạng cao, hoặc độ trễ trên mạng.
- Thời gian xử lý (Processing Time): PLC cần thời gian để xử lý và thực thi các tín hiệu điều khiển mới nhận được. Do chương trình PLC phức tạp hoặc tải lượng công việc lớn, thời gian này có thể tăng lên, gây ra sự delay trong việc phản hồi tới tín hiệu điều khiển mới.
- Độ trễ của giao thức (Protocol Overhead): Giao thức Ethernet/IP có thể có một độ trễ nhất định khi trao đổi dữ liệu giữa Node-RED và PLC. Điều này có thể được ảnh hưởng bởi việc mã hóa và giải mã dữ liệu, cũng như các thao tác xác thực và xử lý lỗi.
- Tải lưu lượng mạng (Network Traffic Load): Nếu có nhiều thiết bị khác sử dụng cùng một mạng Ethernet/IP và tạo ra lưu lượng mạng cao, điều này có thể làm tăng độ trễ trong việc truyền tải tín hiệu điều khiển giữa Node-RED và PLC.

Cách khắc phục:

- Kiểm tra kết nối mạng: Đảm bảo rằng kết nối mạng giữa Node-RED và PLC là ổn định và đủ băng thông. Kiểm tra dây cáp, cổng kết nối, và thiết lập mạng trên cả hai thiết bị để đảm bảo không có vấn đề kết nối nào xảy ra.
- Tối ưu hóa giao thức và cấu hình mạng: Kiểm tra và cấu hình lại các thiết lập giao thức và mạng trên cả Node-RED và PLC để tối ưu hóa hiệu suất truyền dẫn. Thiết lập địa chỉ IP cố định cho cả hai thiết bị để tránh vấn đề khi cấp phát địa chỉ IP động.
- Giảm xung điện và nhiễu: Sử dụng các thiết bị giảm nhiễu hoặc các bộ lọc nhiễu để giảm thiểu ảnh hưởng của xung điện và nhiễu từ môi trường xung quanh lên tín hiệu truyền đi.
- Kiểm tra và cập nhật firmware: Đảm bảo rằng cả Node-RED và PLC đều sử dụng phiên bản firmware mới nhất và cập nhật.
- Đánh giá và tối ưu hóa quy trình tích hợp: Đánh giá và tối ưu hóa lại quy trình tích hợp giữa Node-RED và PLC để tối đa hóa hiệu suất và giảm thiểu độ trễ.

Bằng cách thực hiện những biện pháp trên, có thể giảm thiểu sự delay và đảm bảo rằng tín hiệu điều khiển được truyền đi và xử lý một cách nhanh chóng và đáng tin cậy giữa giao diện người dùng của Node-RED và PLC sử dụng Ethernet/IP. Kết quả thực nghiệm được cải thiện đáng kể, thời gian delay giảm xuống chậm nhất chỉ còn 1s.

5. KẾT LUẬN

Bài báo tập trung nghiên cứu thiết kế xây dựng hệ thống điều khiển giám sát tích hợp IoT lên cánh tay robot 06 bậc tự do. Hệ thống điều khiển xây dựng trên cơ sở giải phương

trình động học của cánh tay robot, bộ điều khiển PLC điều khiển 06 động cơ và khâu thao tác cuối hoạt động độc lập theo từng vị trí. Tín hiệu điều khiển được gửi từ màn hình HMI hoặc gửi từ giao diện điều khiển webserver tới PLC, PLC gửi xuống các động cơ và cơ cấu chấp hành. Toàn bộ tín hiệu được truyền thông qua module IoT 2040 lên websever. Web server vừa có thể giám sát vừa có thể điều khiển trực tiếp robot thông qua truyền thông không dây. Ngoài truyền thông giám sát điều khiển robot còn có thể tính OEE để đánh giá hiệu suất, tối ưu cho mục tiêu sử dụng. Robot được điều khiển làm việc chính xác khi điều khiển vị trí với sai lệch 0,5 mm và độ trễ truyền thông lớn nhất khi có sự cố là 1 s. Các nhược điểm được đề cập ở thảo luận và có hướng khắc phục, đó là cơ sở cho hướng nghiên cứu tiếp theo của nhóm tác giả.

TÀI LIỆU THAM KHẢO

[1] Huỳnh Thế Hiển, " Bộ điều khiển PID dựa trên mạng nơron hàm cơ sở xuyên tâm", Tạp chí Khoa học Đại học Cần Thơ, số 7 tập 54, (2018).

[2] Vũ Thị Yến*, Trần Đức Hiệp, Phạm Thế Vũ, Phạm Trung Hiếu. "Bộ điều khiển bền vững thích nghi nơron trên cơ sở của công nghệ cuốn chiếu cho tay máy," Tạp chí Khoa học và Công nghệ, số 4 vol 57, (2021).

[3] Đỗ Trí Nhựt, "Thiết kế hệ thống điều khiển từ xa cánh tay robot bằng công nghệ trên Iot" Tạp chí Khoa học và Công nghệ Trường Đại học Thủ Dầu Một, số 6 (67) (2023).

[4] J. Craig, Introduction to Robotics: Mechanics and Control, 3rd ed., Pearson, 2004. F. Fleuret and D. Geman. Coarse-to-fine face detection. Int. J. Computer Vision, (2001).

[5] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar, Robot Modeling and Control, Wiley, (2006).

[6] K. S. Fu, R. C. Gonzalez, and C. S. G. Lee, Robotics: Control, Sensing, Vision, and Intelligence, McGraw-Hill, (1987).

[7] M. Raibert and J. Craig, "Hybrid Position/Force Control of Manipulators," Journal of Dynamic Systems, Measurement, and Control, vol. 102, no. 2, pp. 126–133, (1981).

[8] P. Corke, Robotics, Vision and Control: Fundamental Algorithms in MATLAB, 2nd ed., Springer, (2017).

[9] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, Robotics: Modelling, Planning and Control, Springer, (2009).

[10] Shengzeng Huo, Fangyuan Wang, Luyin Hu, Peng Zhou, Jihong Zhu, Hesheng Wang, David Navarro-Alarcon, "A Dual-Arm Collaborative Framework for Dexterous Manipulation in Unstructured Environments with Contrastive Planning" Cornel University (2021).

[11] Hongqian Deng; Dan Xiong; Yiyong Huang; Liyong Fang; Wei Han; Hongwei Liu, "High-precision control of robotic arms based on active visual under unstructured scenes" IEEE International Conference on Robotics and Biomimetics, (2022).

Thông tin liên hệ: **Phạm Trung Thiên**

Điện thoại: 0963284444 - Email: ptthien.ck@uneti.edu.vn

Khoa Cơ khí, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp.