



Tạp chí điện tử
Khoa học và Công nghệ Giao thông
Trang website: <https://jstt.vn/index.php/vn>



Article info

Type of article:

Scientific information

DOI:

<https://doi.org/10.58845/jstt.utt.2026.vn.6.7.1-16>

*Corresponding author:

Email address:

vu.le@ut.edu.vn

Received: 22/05/2026

Received in Revised Form:
18/08/2026

Accepted: 24/08/2026

Design and simulation of a 3-DOF SCARA Robotic arm model using ROS 2

Nguyen Trong Khang, Nguyen Le Dang Khoa, Le Van Thong, Le Cong Danh, Le Anh Uyen Vu*

Strategic Technology Board, Ho Chi Minh City University of Transport, No.2 Vo Oanh Street, Thanh My Tay Ward, Ho Chi Minh City, Vietnam

Abstract: Amid the profound scientific and technological transformations brought about by the Fourth Industrial Revolution, learners are required not only to master theoretical knowledge but also to develop important practical competencies such as critical thinking, problem-solving, and teamwork. The disparity between high theoretical scores and the lack of practical application skills among the younger generation highlights an urgent need to innovate teaching methodologies. Specifically, education must shift towards interdisciplinary integration rather than isolated subject teaching. This paper analyzes the essential role of STEM education and its correlation with Industrial Robotics. Consequently, the study proposes the design of a 3-DOF SCARA Robotic arm model, integrated with visual simulation using Digital Twin technology on the ROS 2 platform. This model aims to facilitate and enhance the quality of student scientific research operations at Ho Chi Minh City University of Transport.

Keywords: STEM, ROS 2, SCARA, Robot kinematics, Digital Twin.



Thiết kế và mô phỏng mô hình cánh tay Robot SCARA 3-DOF sử dụng ROS 2

Thông tin bài viết

Dạng bài viết:

Thông tin khoa học

DOI:

<https://doi.org/10.58845/jstt.utt.2026.vn.6.7.1-16>

*Tác giả liên hệ:

Địa chỉ Email:

vu.le@ut.edu.vn

Ngày nộp bài: 22/05/2026

Ngày nộp bài sửa: 18/08/2026

Ngày chấp nhận: 24/08/2026

Nguyễn Trọng Khang, Nguyễn Lê Đăng Khoa, Lê Văn Thông, Lê Công Danh, Lê Anh Uyên Vũ*

Ban Công nghệ chiến lược, Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh, số 2 đường Võ Oanh, phường Thạnh Mỹ Tây, TP. Hồ Chí Minh, Việt Nam

Tóm tắt: Trước những biến đổi mạnh mẽ về khoa học và công nghệ mà Cách mạng Công nghiệp lần thứ tư (CMCN 4.0) mang lại, người học không chỉ cần vững kiến thức lý thuyết, mà còn phải rèn luyện những năng lực thực tiễn quan trọng như tư duy phản biện, khả năng giải quyết vấn đề và kỹ năng làm việc nhóm. Sự chênh lệch giữa điểm số lý thuyết và kỹ năng ứng dụng thực tế đang đặt ra yêu cầu cấp thiết đối với việc đổi mới phương pháp giảng dạy theo hướng tích hợp liên môn thay vì tách rời từng lĩnh vực. Bài báo phân tích vai trò cốt lõi của giáo dục STEM và mối liên hệ thực tiễn với bộ môn Robot Công nghiệp. Qua đó, nhóm nghiên cứu đề xuất thiết kế mô hình cánh tay Robot SCARA 3 bậc tự do, kết hợp với mô phỏng trực quan ứng dụng công nghệ Bản sao số (Digital Twin) trên nền tảng ROS 2. Mô hình này được xây dựng nhằm phục vụ và nâng cao chất lượng hoạt động nghiên cứu khoa học của sinh viên tại trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh.

Từ khóa: STEM, ROS 2, SCARA, động học robot, bản sao số.

1. Giới thiệu

1.1. Sự tương quan môn học STEM và Robot Công nghiệp

1.1.1. Mô hình giáo dục STEM trên thế giới và tại Việt Nam

Giáo dục STEM (Science, Technology, Engineering, and Mathematics – Khoa học, Công nghệ, Kỹ thuật và Toán học) khởi nguồn từ khái niệm SMET (Science, Mathematics, Engineering, and Technology) (thập niên 1990) và được NSF (National Science Foundation) Hoa Kỳ chuẩn hóa năm 2001 với mục tiêu tăng cường nhân lực khoa học - kỹ thuật trong bối cảnh cạnh tranh toàn cầu. Trên thế giới, mỗi khu vực triển khai STEM theo cách riêng: Hoa Kỳ đầu tư bài bản qua chính sách quốc gia [1]; châu Âu tích hợp liên ngành để giải

quyết vấn đề xã hội và môi trường; trong khi các nước châu Á như Singapore, Hàn Quốc, Nhật Bản đưa STEM vào từ bậc tiểu học như một công cụ chiến lược thúc đẩy đổi mới và công nghiệp hóa. UNESCO cũng xác định STEM là "chìa khóa" thực hiện các mục tiêu phát triển bền vững. Ngày nay, STEM không chỉ đào tạo kỹ sư hay nhà khoa học mà còn hướng đến phát triển năng lực thích ứng và tư duy giải quyết vấn đề phức hợp [2]. Tại Việt Nam, STEM có xuất phát điểm khác biệt, không từ chính sách hệ thống mà từ các sân chơi Robot do doanh nghiệp tổ chức, khiến việc tiếp cận ban đầu còn manh mún, thiếu đồng bộ. Tuy nhiên, với Chiến lược phát triển giáo dục đến năm 2030 đặt mục tiêu nâng tỷ trọng sinh viên khối STEM lên 35% [3], Việt Nam đang dần định hình một nền

tăng STEM có hệ thống, khẳng định vai trò chiến lược của lĩnh vực này trong phát triển nguồn nhân lực quốc gia.

1.1.2. Đề xuất phương án triển khai STEM tại trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh

Hòa cùng xu hướng đổi mới giáo dục toàn quốc, Nhà trường đã chủ động thúc đẩy phong trào nghiên cứu khoa học (NCKH) cho năm học 2025 - 2026 thông qua các kế hoạch đăng ký nhiệm vụ đổi mới sáng tạo và sự kiện “Ngày nghiên cứu khoa học sinh viên”, giúp hoạt động này gắn liền với quá trình học tập thực tế. Bám sát định hướng của Chính phủ tại Quyết định 1131/QĐ-TTg ngày 12/06/2025 [4] về việc ưu tiên làm chủ các công nghệ cốt lõi như Robot, trí tuệ nhân tạo (AI - Artificial Intelligence) và bản sao số (Digital Twin), đồng thời hiện thực hóa mục tiêu phát triển giáo dục đến năm 2030 nhằm nâng cao chất lượng đào tạo khối ngành STEM [2]. Nhóm nghiên cứu đề xuất mô hình kỹ thuật phục vụ lồng ghép học phần Robot công nghiệp vào mô hình giáo dục liên ngành. Phương án này nhằm chuyển đổi môn học truyền thống thành hoạt động NCKH thường xuyên cho sinh viên các ngành Kỹ thuật điều khiển và Tự động hóa tại Nhà trường. Nghiên cứu này tập trung vào khía cạnh thiết kế và triển khai hệ thống kỹ thuật, trình bày việc ứng dụng bản sao số trên nền tảng ROS 2 kết hợp xây dựng mô hình vật lý cánh tay Robot SCARA, nhằm tạo ra môi trường thực hành trực quan giúp nâng cao năng lực ứng dụng thực tiễn cho sinh viên [2].

1.2. Tổng quan tài liệu

1.2.1. Bản sao số (Digital Twin) trong các bài toán Robot

Bản sao số (Digital Twin) đang trở thành một xu hướng quan trọng trong lĩnh vực Robot và tự động hóa hiện nay. Đây là mô hình số hóa của Robot hoặc toàn bộ hệ thống, cho phép mô phỏng và giám sát song song giữa môi trường ảo và hệ thống vật lý. Cùng với sự phát triển của công nghệ bán dẫn, các phần mềm mô phỏng như MATLAB/Simulink hay ROS ngày càng hỗ trợ xây dựng bản sao số với độ chính xác cao hơn. Trong

các bài toán Robot, công nghệ này giúp kiểm chứng thuật toán điều khiển trước khi triển khai thực tế, từ đó giảm rủi ro và chi phí thử nghiệm. Ngoài ra, bản sao số còn hỗ trợ phân tích động học, động lực học và tối ưu quỹ đạo chuyển động. Trong các nghiên cứu gần đây về Robot công nghiệp, Robot tự hành và thiết bị bay không người lái [5-7], mô hình song sinh đã được ứng dụng để mô phỏng môi trường làm việc, giải bài toán lập quỹ đạo, phát hiện va chạm và tối ưu hiệu suất thời gian thực. Tuy nhiên, phần lớn các nghiên cứu này tập trung vào quy mô công nghiệp hoặc nghiên cứu chuyên sâu, sử dụng nền tảng phần cứng/phần mềm có chi phí cao, chưa hướng đến mục tiêu triển khai ở quy mô giáo dục đại học với chi phí thấp.

Tóm lại, trong bối cảnh Cách mạng Công nghiệp lần thứ tư, việc áp dụng bản sao số không chỉ giúp nâng cao hiệu quả thiết kế và điều khiển Robot mà còn mở ra hướng tiếp cận mới trong tối ưu hóa, giám sát và ra quyết định dựa trên dữ liệu, góp phần thúc đẩy sự phát triển của các hệ thống Robot thông minh và tự chủ.

1.2.2. Giới thiệu tổng quan về ROS 2

ROS 2 là nền tảng phần mềm trung gian (middleware) mã nguồn mở, đóng vai trò kết nối các thành phần của hệ thống Robot thành một thể thống nhất. Kế thừa từ ROS 1 nhưng được tái thiết kế hoàn toàn, ROS 2 khắc phục hạn chế điểm lỗi đơn lẻ (Single Point of Failure - SPOF) của phiên bản trước, nơi toàn bộ hệ thống phụ thuộc vào một nút trung tâm duy nhất (ROS Master), dẫn đến rủi ro gián đoạn mạng giao tiếp khi nút này gặp sự cố. Thay vào đó, kiến trúc phân tán dựa trên chuẩn DDS (Data Distribution Service) cho phép các thành phần giao tiếp trực tiếp ngang hàng, mang lại khả năng xử lý thời gian thực (real-time), tăng cường khả năng chịu lỗi và đảm bảo tính ổn định trong các điều kiện vận hành phức tạp [8]. Hệ sinh thái của ROS 2 tích hợp nhiều tính năng chuyên sâu: cơ chế QoS (Quality of Service-Chất lượng dịch vụ) cho phép tinh chỉnh băng thông và độ trễ theo từng yêu cầu ứng dụng; micro-ROS mở rộng khả năng tích hợp xuống các vi điều khiển có tài

nguyên hạn chế như STM32 hay ESP32; trong khi RViz2 và Gazebo cung cấp công cụ trực quan hóa và mô phỏng vật lý 3D, giúp kiểm thử toàn bộ hệ thống Robot trên môi trường ảo trước khi triển khai trên phần cứng thực tế [8]. Nhờ đó, ROS 2 đang là nền tảng tiêu chuẩn được áp dụng rộng rãi từ Robot công nghiệp, Robot phẫu thuật, xe tự hành đến thiết bị bay không người lái, bởi cả cộng đồng nghiên cứu lẫn doanh nghiệp công nghệ.

1.2.3. Khoảng trống nghiên cứu

Mặc dù bản sao số và ROS 2 đã được ứng dụng rộng rãi trong nghiên cứu Robot công nghiệp và tự động hóa quy mô lớn, các mô hình hiện có thường đòi hỏi hạ tầng phần cứng phức tạp và chi phí đầu tư cao, gây khó khăn khi triển khai trong môi trường giáo dục đại học, nơi yêu cầu cân bằng giữa tính trực quan, chi phí hợp lý và khả năng thực hành cho sinh viên. Bên cạnh đó, các nghiên cứu về giáo dục STEM tại Việt Nam hiện vẫn thiếu mô hình kỹ thuật cụ thể tích hợp đồng thời cả nền tảng mô phỏng số (ROS 2) và mô hình vật lý thực nghiệm, phục vụ trực tiếp cho giảng dạy học phần Robot công nghiệp. Để giải quyết khoảng trống này, nghiên cứu đề xuất giải pháp thiết kế và chế tạo hệ thống cánh tay Robot SCARA chi phí thấp tích hợp bản sao số trên nền tảng ROS 2 đóng vai trò là công cụ thực hành trực quan phục vụ đào tạo STEM tại Trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh.

1.3. Đóng góp của nghiên cứu

Trên cơ sở khoảng trống nghiên cứu đã nêu ở mục 1.2.3, nghiên cứu của nhóm tác giả tập trung vào các nội dung cụ thể và có được các đóng góp như sau:

- Xây dựng hoàn chỉnh mô hình động học cho cánh tay Robot SCARA 3 bậc tự do: thiết lập bảng tham số Denavit-Hartenberg, giải bài toán động học thuận và động học ngược, đồng thời kiểm chứng bằng thực nghiệm mô phỏng với sai số định vị nhỏ (khoảng 0.020–0.060 theo các trường hợp kiểm thử tại Bảng 3, Bảng 4), khẳng định độ tin cậy của mô hình toán trước khi mở rộng sang điều khiển phần cứng thực tế.

- Triển khai quy trình mô phỏng trực quan trên nền tảng ROS 2 mã nguồn mở: đóng gói toàn bộ quy trình từ thiết lập không gian làm việc, xây dựng mô tả URDF, đến giao diện điều khiển đồ họa (control_gui.py) cho phép nhập góc khớp và tính toán động học thuận/ngược trực tiếp, hiển thị và kiểm chứng chuyển động trên RViz2 tạo thành một bộ công cụ thực hành khép kín, dễ tiếp cận cho sinh viên chưa có nền tảng chuyên sâu về ROS.

- Kết hợp quy trình thiết kế số và chế tạo thực nghiệm: sử dụng SolidWorks để thiết kế, lắp ráp và kiểm tra tương thích cơ khí giữa các cụm cơ cấu (nâng hạ trục Z, khớp xoay - truyền động đai răng, cơ cấu chấp hành/tay gấp hai ngón) trước khi gia công, giúp loại bỏ phương pháp thử-sai truyền thống và tối ưu chi phí, thời gian nghiên cứu.

- Cung cấp học liệu thực hành tiêu chuẩn phục vụ giáo dục STEM: Đề tài đã giải quyết thành công khoảng trống về mô hình kỹ thuật chi phí thấp tích hợp đồng thời nền tảng mô phỏng số và thực nghiệm tại Việt Nam. Kết quả nghiên cứu đóng vai trò là học liệu thực hành trực quan, hỗ trợ đổi mới phương pháp dạy học liên ngành STEM, phục vụ trực tiếp học phần Robot Công nghiệp và nâng cao chất lượng hoạt động nghiên cứu khoa học của sinh viên tại trường Đại học Giao thông vận tải Thành phố Hồ Chí Minh [2].

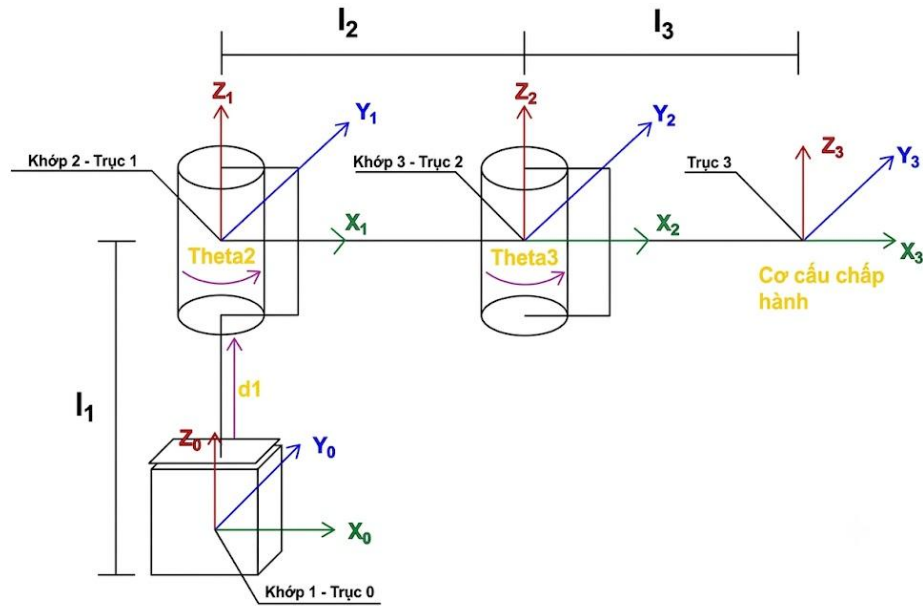
2. Thiết kế, chế tạo cánh tay Robot SCARA

2.1. Bài toán động học thuận, ngược cánh tay Robot SCARA

2.1.1. Bài toán động học thuận cánh tay Robot SCARA

Bài toán động học thuận của cánh tay Robot SCARA được xây dựng trên cơ sở phân tích giản đồ động học của cánh tay Robot (Hình 1) và xác định bảng D-H tương ứng với giản đồ động học đó (Bảng 1) [9, 10].

Từ bảng D-H (Bảng 1) và công thức tổng quát của ma trận thuần nhất (1) thu được các ma trận biến đổi thuần nhất giữa các hệ tọa độ (2-4) [9]. Từ đó giúp xác định ma trận chuyển vị của điểm làm việc cuối với tọa độ gốc (5).



Hình 1. Giải đồ động học cánh tay Robot SCARA

Bảng 1. Bảng tham số D-H của cánh tay Robot SCARA

Joint i	θ_i	α_i	r_i	d_i
1	0°	0°	0	$l_1 + d_1$
2	θ_2	0°	l_2	0
3	θ_3	0°	l_3	0

$$H_i^{i-1} =$$

$$\begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & r_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & r_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$H_1^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_1 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$H_2^1 = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & l_2 \cdot \cos \theta_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & l_2 \cdot \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$H_3^2 = \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & l_3 \cdot \cos \theta_3 \\ \sin \theta_3 & \cos \theta_3 & 0 & l_3 \cdot \sin \theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Ta có:

$$H_3^0 = H_1^0 \cdot H_2^1 \cdot H_3^2 =$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_1 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & l_2 \cdot \cos \theta_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & l_2 \cdot \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & l_3 \cdot \cos \theta_3 \\ \sin \theta_3 & \cos \theta_3 & 0 & l_3 \cdot \sin \theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & l_3 \cdot \cos \theta_3 \\ \sin \theta_3 & \cos \theta_3 & 0 & l_3 \cdot \sin \theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_3^0 =$$

$$\begin{bmatrix} \cos(\theta_2 + \theta_3) & -\sin(\theta_2 + \theta_3) & 0 & l_2 \cos \theta_2 + l_3 \cos(\theta_2 + \theta_3) \\ \sin(\theta_2 + \theta_3) & \cos(\theta_2 + \theta_3) & 0 & l_2 \sin \theta_2 + l_3 \sin(\theta_2 + \theta_3) \\ 0 & 0 & 1 & l_1 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Suy ra:

$$d_3^0 = \begin{bmatrix} l_2 \cos \theta_2 + l_3 \cos(\theta_2 + \theta_3) \\ l_2 \sin \theta_2 + l_3 \sin(\theta_2 + \theta_3) \\ l_1 + d_1 \end{bmatrix} \quad (5)$$

2.1.2. Bài toán động học ngược cánh tay Robot SCARA

Bài toán động học ngược cánh tay Robot SCARA được giải quyết thông qua phương pháp hình học với hình chiếu bằng (Hình 2a) và hình chiếu đứng (Hình 2b) [11].

Từ bài toán thuận ta có tọa độ điểm B (X_b , Y_b , Z_b).

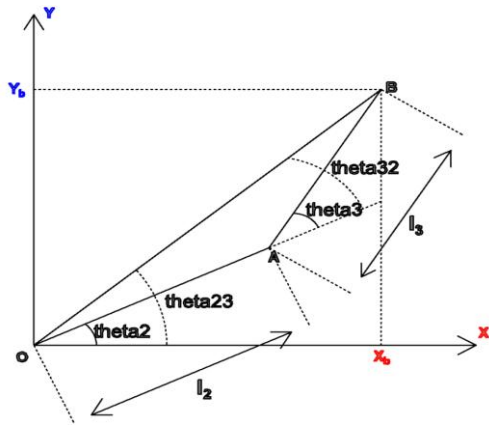
$$\begin{cases} X_b = l_2 \cos \theta_2 + l_3 \cos(\theta_2 + \theta_3) \\ Y_b = l_2 \sin \theta_2 + l_3 \sin(\theta_2 + \theta_3) \\ Z_b = l_1 + d_1 \end{cases}$$

Đặt:

$$Z_b = OB = l_1 + d_1$$

$$OA = l_2$$

$$BA = l_3$$



a)

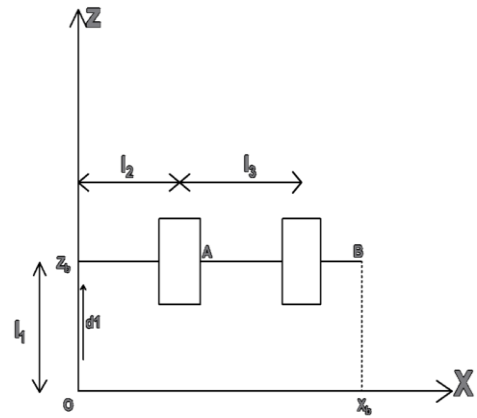
Ta có (6):

$$Z_b = l_1 + d_1 \quad (6)$$

$$\rightarrow d_1 = Z_b - l_1$$

Xét tam giác vuông OXB (7):

$$\begin{aligned} OB^2 &= OX_b^2 + BX_b^2 \\ \rightarrow OB &= \sqrt{OX_b^2 + BX_b^2} \end{aligned} \quad (7)$$



b)

Hình 2. Hình chiếu bằng (a) và hình chiếu đứng (b) của cánh tay Robot SCARA

Xét tam giác OAB. Ta có:

$$OB^2 = OA^2 + AB^2 - 2.OA.AB. \cos \widehat{OAB}$$

$$\rightarrow OB^2 = OA^2 + AB^2 - 2.OA.AB. \cos(180^\circ - \theta_3)$$

$$OB^2 = OA^2 + AB^2 + 2.OA.AB. \cos \theta_3$$

$$\cos \theta_3 = \frac{OB^2 - OA^2 - AB^2}{2.OA.AB}$$

Suy ra (8):

$$\cos \theta_3 = \frac{OB^2 - l_2^2 - l_3^2}{2.l_2.l_3} \quad (8)$$

Ta có (9) - (10):

$$\sin^2 \theta_3 + \cos^2 \theta_3 = 1$$

$$\sin \theta_3 = \pm \sqrt{1 - \cos^2 \theta_3} \quad (9)$$

Để tính θ_3 chính xác, tránh lỗi miền giá trị của hàm Arccos, dùng hàm Arctan (10):

$$\theta_3 = \text{Arctan}(\sin \theta_3, \cos \theta_3) \quad (10)$$

Ta có (11) - (13):

$$\theta_2 = \theta_{23} - \theta_{32}$$

$$\theta_{23} = \text{Arctan}\left(\frac{BX_b}{OX_b}\right) \quad (11)$$

$$\theta_{32} = \text{Arctan}\left(\frac{l_3 \sin \theta_3}{l_2 + l_3 \cos \theta_3}\right) \quad (12)$$

$$\rightarrow \theta_2 = \text{Arctan}\left(\frac{BX_b}{OX_b}\right) - \text{Arctan}\left(\frac{l_3 \sin \theta_3}{l_2 + l_3 \cos \theta_3}\right) \quad (13)$$

2.2. Mô hình hóa cánh tay Robot SCARA với Rviz

Việc mô hình hóa cánh tay Robot SCARA trong nghiên cứu này được thực hiện thông qua các bước sau: Cài đặt hệ điều hành Ubuntu; Tạo môi trường làm việc; Xây dựng mô hình với URDF; Vận hành mô hình.

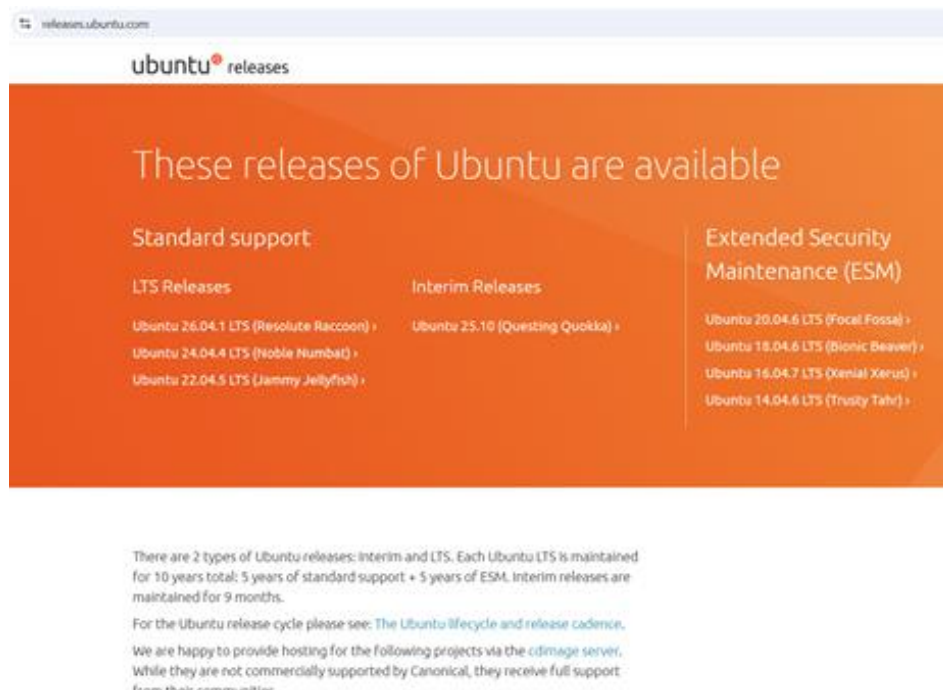
2.2.1. Cài đặt Linux Ubuntu

Đầu tiên, tải File VirtualBox từ trang chủ <https://www.virtualbox.org/> [12] sẽ được một file cài đặt (Hình 3a). Vào trang <https://ubuntu.com/> [13] để tải và lựa chọn những phiên bản Ubuntu phù hợp với cấu hình máy tính (Hình 3b).

Kết quả trong nghiên cứu này được xây dựng với phiên bản Ubuntu 22.04 LTS (Jammy Jellyfish). Sau khi vào trang của phiên bản Ubuntu 22.04 LTS (Jammy Jellyfish), click vào *64-bit PC (AMD64) desktop image* để tải phiên bản này xuống. Khuyến nghị nên chọn phiên bản hỗ trợ dài hạn (Long Term Support - LTS) vì tính ổn định và hỗ trợ cao cho ROS.



a)



b)

Hình 3. Quy trình cài đặt Linux Ubuntu: a) Trang chủ cài đặt; b) Lựa chọn phiên bản hệ điều hành Ubuntu phù hợp.

2.2.2. Quy trình tạo mô hình trên ROS 2

Sau khi cài đặt hệ điều hành Ubuntu, việc tiếp theo là cần cài ROS 2 trên hệ điều hành này. Mỗi phiên bản Ubuntu chỉ hỗ trợ tốt nhất cho một phiên bản ROS nhất định (Bảng 2). Với hệ điều hành Ubuntu 22.04 LTS đang sử dụng, phiên bản cài đặt phù hợp nhất chính là ROS 2 Humble. Để cài đặt ROS 2 Humble, có thể vào trang hỗ trợ và

theo các bước hướng dẫn cụ thể để tiến hành cài đặt [8, 13].

Sau khi cài đặt xong, tiến hành kiểm tra phiên bản ROS 2 đã được cài đặt bằng câu lệnh: “`echo $ROS_DISTRO`” (Hình 4). Kết quả như trên cho thấy hệ thống cài đặt ROS 2 Humble. Tiếp theo, có thể kiểm tra đã cài đặt phiên bản ROS 2 với câu lệnh: “`ls /opt/ros/humble`” (Hình 5).

Bảng 2. Hệ điều hành Ubuntu và ROS 2 tương ứng

STT	Hệ điều hành Ubuntu	ROS 2
1	18.04	Dashing Diademata
2	20.04	Foxy Fitzroy
3	22.04	Humble Hawksbill
4	24.04	Jazzy Jalisco

```

ubuntu@ubuntu: ~
ubuntu@ubuntu: ~ 38x24
ubuntu@ubuntu:~$ echo $ROS_DISTRO
humble
ubuntu@ubuntu:~$

```

Hình 4. Xác nhận phiên bản ROS 2

```

ubuntu@ubuntu: ~
ubuntu@ubuntu: ~ 38x24
ubuntu@ubuntu:~$ ls /opt/ros/humble
bin          opt
cmake        setup.bash
include      setup.sh
lib          setup.zsh
local        share
local_setup.bash src
local_setup.sh test
_local_setup_util.py tools
local_setup.zsh

```

Hình 5. Cấu trúc thư mục cài đặt ROS 2

2.2.3. Tạo môi trường cơ bản cho cánh tay Robot

Khởi động Ubuntu trước, mở Terminator. Đầu tiên, nhập `mkdir --parents prr_ws/src + enter` để tạo không gian làm việc cho cánh tay Robot. Nhập `cd ~/prr_ws + enter` để vào thư mục vừa tạo, sử dụng lệnh `colcon build` kết thúc quá trình tạo (Hình 6).

```

colcon build' successful
/home/ubuntu/prr_ws

ubuntu@ubuntu: ~/prr_ws 64x24
ubuntu@ubuntu:~$ mkdir --parents prr_ws/src
cd ~/prr_ws
colcon build
Starting >>> prr_bringup
Starting >>> prr_control
Starting >>> prr_description
Finished <<< prr_bringup [0.34s]
Finished <<< prr_description [0.36s]
Finished <<< prr_control [0.91s]
Summary: 3 packages finished [1.15s]

```

Hình 6. Tạo không gian làm việc ban đầu

Lệnh `source install/setup.bash; source /opt/ros/humble/setup.bash;source ~/prr_ws/install/setup.bash` là một lệnh rất quan trọng khi làm việc với ROS 2 có nhiệm vụ thiết lập môi trường làm việc cho ROS 2 trong terminal – Cửa sổ dòng lệnh hiện tại. Vì vậy, cần sử dụng lệnh này để cập nhật môi trường cho terminal hiện tại.

Như vậy, phần thiết lập môi trường cho cánh tay Robot đã cơ bản hoàn tất, đảm bảo sẵn sàng cho các bước triển khai mô phỏng và điều khiển tiếp theo (Hình 7 và Hình 8).

```

ubuntu@ubuntu: ~/prr_ws
ubuntu@ubuntu: ~/prr_ws 64x24
ubuntu@ubuntu:~/prr_ws$ source install/setup.bash
source /opt/ros/humble/setup.bash
source ~/prr_ws/install/setup.bash

```

Hình 7. Thiết lập môi trường làm việc cho ROS 2

```

Starting >>> prr_bringup
Starting >>> prr_control
Starting >>> prr_description
Finished <<< prr_bringup [0.34s]
Finished <<< prr_description [0.36s]
Finished <<< prr_control [0.91s]

Summary: 3 packages finished [1.15s]

```

Hình 8. Đã thiết lập xong không gian làm việc

2.2.4. Tích hợp mô hình URDF của cánh tay Robot và các thành phần liên quan

Đầu tiên tạo các thư mục chính để chứa các file tạo nên mô hình Robot và điều khiển Robot trong ROS 2. Trong thư mục `prr_description` chứa tệp mô tả mô hình cánh tay Robot SCARA.

Đoạn mã URDF (Hình 9b) mô tả cấu trúc vật lý và động học của một Robot mang cấu hình PRR (Tịnh tiến - Xoay - Xoay) với 3 bậc tự do (3-DOF), kết hợp cùng một cơ cấu chấp hành cuối. Về cơ

bản, hệ thống được cấu thành từ các phần chính sau:

Đế và khung dẫn hướng: Bao gồm bộ đỡ cố định liên kết với một giá đỡ thẳng đứng, đóng vai trò làm điểm tựa và trục dẫn hướng cho toàn bộ chuyển động của cánh tay.

Hệ thống truyền động (3-DOF): Khớp 1 (Khớp tịnh tiến - Prismatic): Cho phép toàn bộ cụm cánh tay trượt lên/xuống dọc theo giá đỡ thẳng đứng để thay đổi độ cao; Khớp 2 và Khớp 3 (Khớp xoay - Revolute): Là hai khớp bản lề nối tiếp nhau trên mặt phẳng ngang, giúp cánh tay Robot mở rộng tầm với và định vị chính xác điểm cuối trong không gian làm việc.

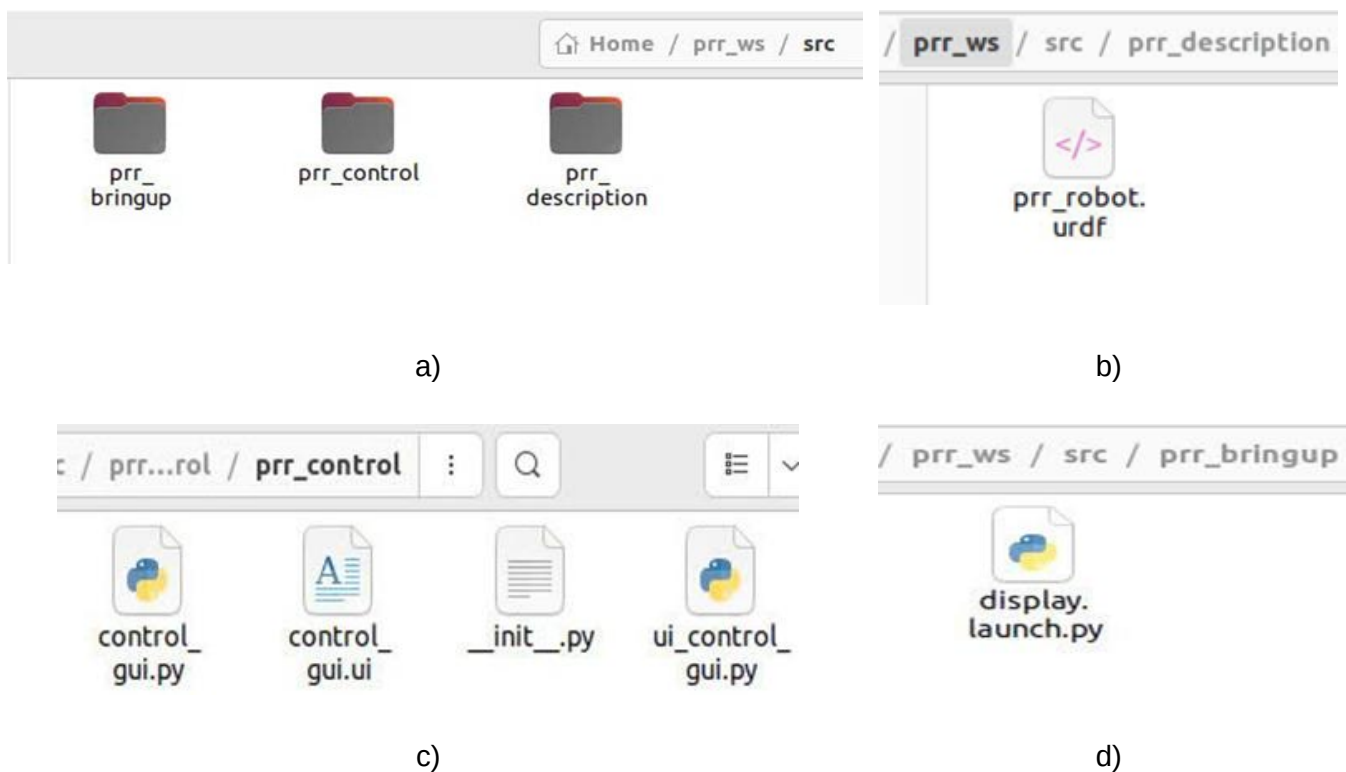
Cơ cấu tay gấp (Gripper): Được gắn ở khâu cuối cùng của Robot, bao gồm hai ngón kẹp đối xứng sử dụng cơ chế khớp xoay để thực hiện thao tác đóng/mở kẹp vật thể.

Giao diện điều khiển (ros2_control): Mô hình được tích hợp sẵn cấu hình hệ thống điều khiển

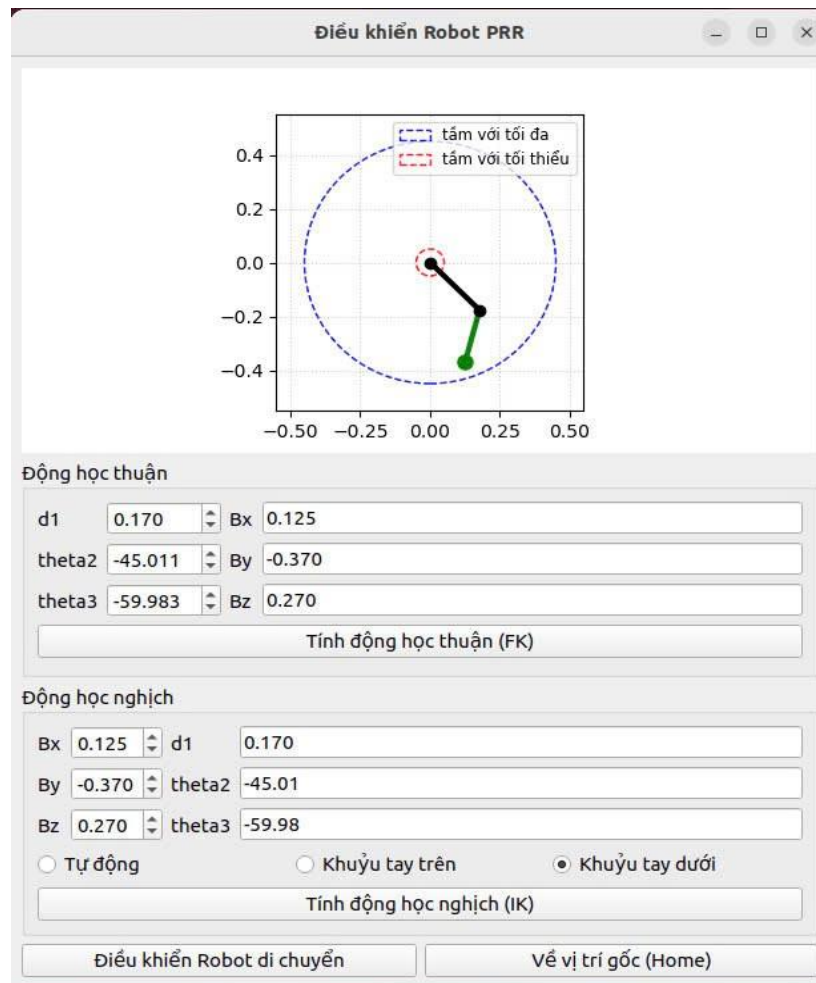
giả lập của ROS 2. Trong đó, cả ba khớp chính đều được thiết lập giao diện nhận lệnh và phản hồi theo vị trí (position interface), sẵn sàng cho các bài toán mô phỏng quỹ đạo và tính toán động học ngược (Inverse Kinematics).

Tiếp theo (Hình 9c) là thư mục chứa các file tạo ra bảng điều khiển các góc xoay và tính toán động học thuận và động học nghịch của cánh tay Robot. Cụ thể, file `control_gui.py` đóng vai trò là file chính để khởi tạo và hiển thị bảng điều khiển; trong khi đó, các file `ui_control_gui.py` và `control_gui.ui` được sử dụng để xây dựng và thiết kế giao diện người dùng của hệ thống điều khiển (Hình 10).

Tại (Hình 9d) là thư mục chứa file launch, được sử dụng để khởi động và quản lý đồng thời nhiều phần mềm (node) của hệ thống Robot chỉ bằng một câu lệnh duy nhất. Thư mục đóng vai trò như một bộ điều khiển trung tâm trong quá trình khởi động hệ thống, cho phép người dùng tự động nạp các thông số cấu hình (parameters).



Hình 9. Cấu trúc thư mục workspace ROS 2 cho Robot SCARA: (a) Workspace chính của mô hình cánh tay Robot SCARA; (b) File URDF mô tả mô hình Robot SCARA trong ROS 2; (c) Thư mục chứa giao diện tính toán động học; (d) Thư mục chứa file launch khởi động hệ thống Robot



Hình 10. Giao diện điều khiển Robot bằng tính toán động học

2.2.5. Chạy mô hình trên ROS 2

Trước tiên, để mở mô hình, truy cập vào package prr_ws bằng lệnh `cd ~/prr_ws + enter`. Sau đó cần build và source lại môi trường trước khi cho Robot chạy bằng lệnh `rm -rf build install log; colcon build; source install/setup.bash`. Tiếp theo nhập lệnh `ros 2 launch prr_bringup display.launch.py` để khởi tạo mô hình Robot trong ROS 2 (Hình 11).

Khi chạy mô hình trên ROS 2

Tại bảng động học thuận nhập các giá trị: $d_1 = 25$ cm, $\theta_2 = 30^\circ$, $\theta_3 = 70^\circ$ (Hình 12) và tính được các giá trị động học ngược $X_b = 0.182$, $Y_b = 0.322$, $Z_b = 0.350$ (Bảng 3).

Bảng động học ngược đã tính ra các giá trị: $X_b = 0.182$, $Y_b = 0.322$, $Z_b = 0.350$, sau đó trả về các giá trị động học thuận $d_1 = 25$ cm, $\theta_2 = 30^\circ$,

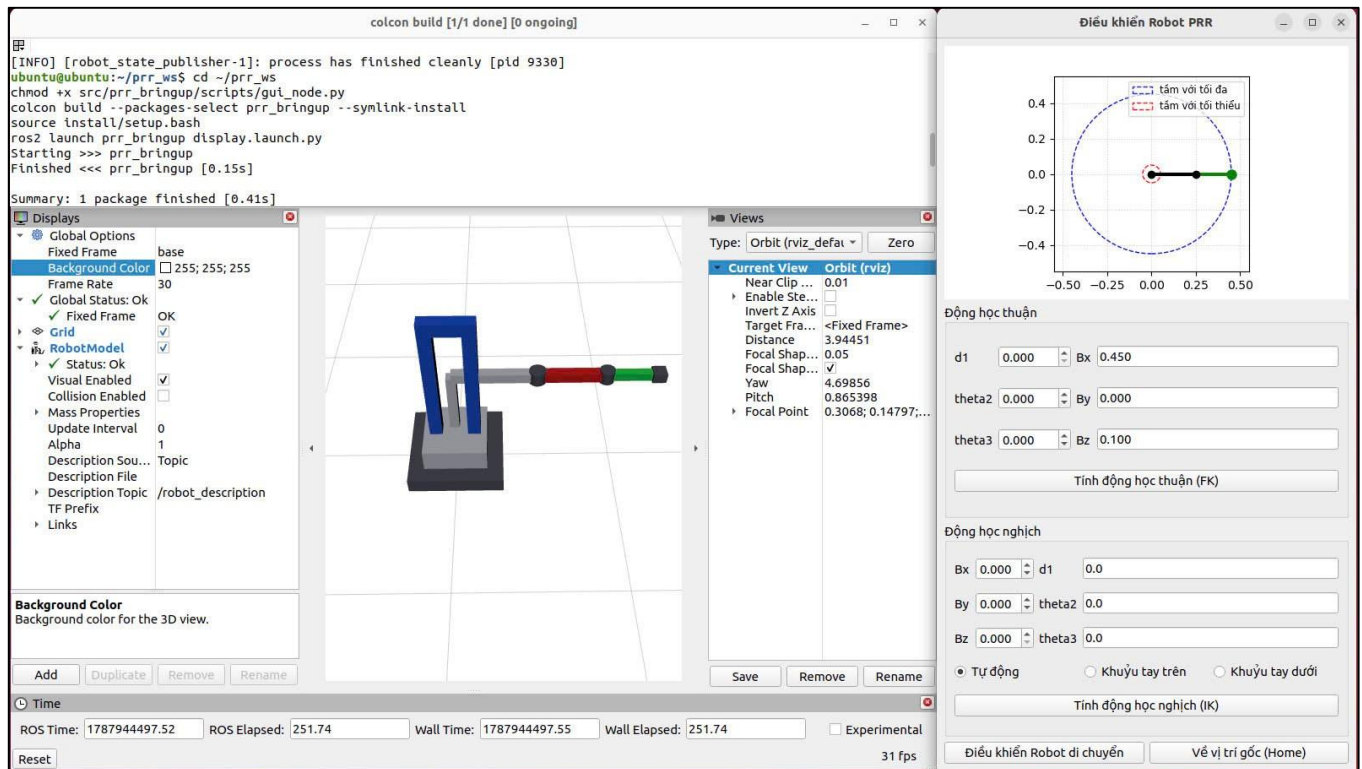
$\theta_3 = 69.94^\circ$ (Bảng 4).

Bảng động học thuận nhập các giá trị: $d_1 = 17$ cm, $\theta_2 = 23.27^\circ$, $\theta_3 = 59.5^\circ$ (Hình 13) và tính được các giá trị động học ngược $X_b = 0.255$, $Y_b = 0.297$, $Z_b = 0.270$ (Bảng 3).

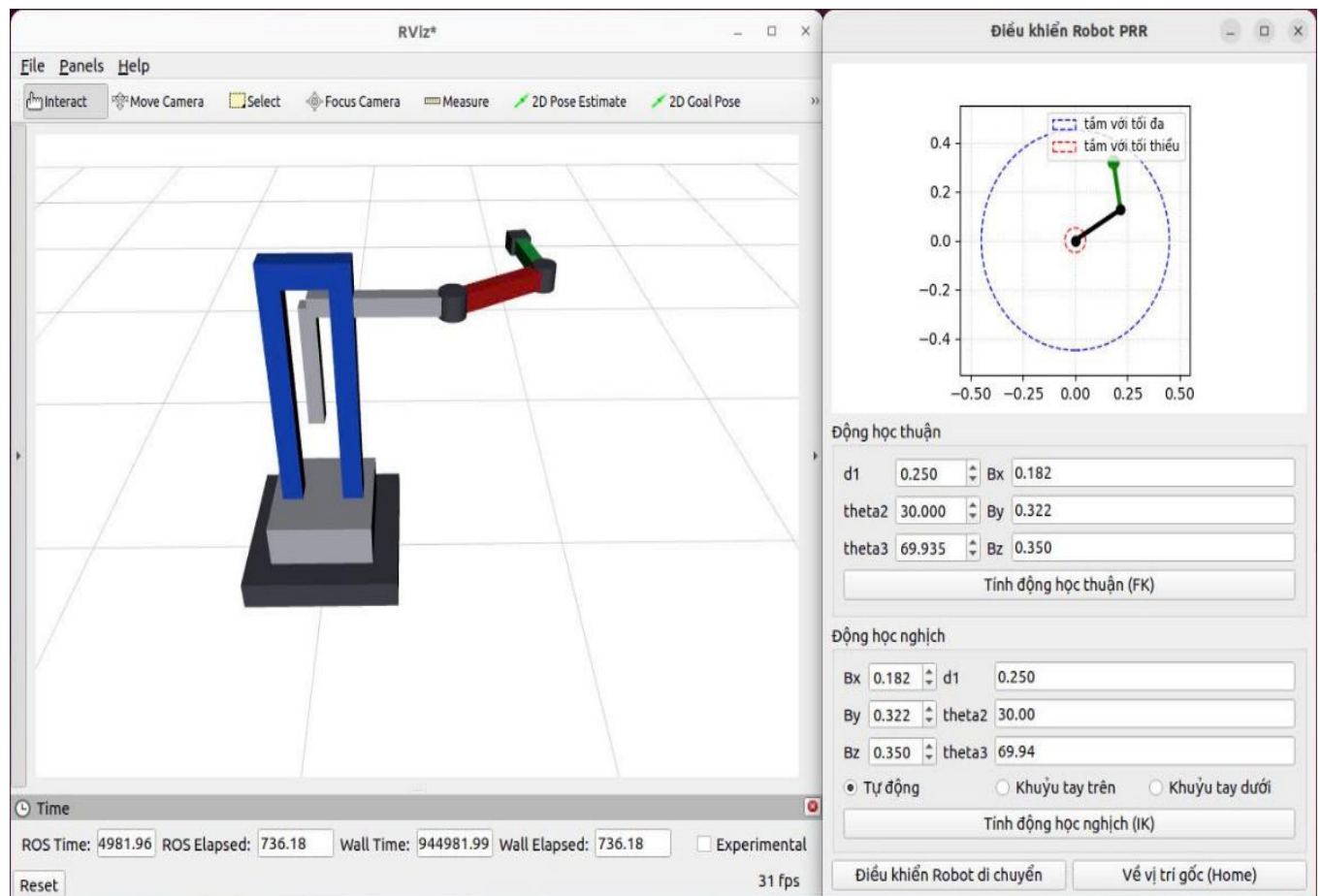
Bảng động học ngược đã tính ra các giá trị: $X_b = 0.255$, $Y_b = 0.297$, $Z_b = 0.270$ sau đó trả về các giá trị động học thuận $d_1 = 17$ cm, $\theta_2 = 23.23^\circ$, $\theta_3 = 59.51^\circ$ (Bảng 4).

Bảng động học thuận nhập các giá trị: $d_1 = 17$ cm, $\theta_2 = -45^\circ$, $\theta_3 = -60^\circ$ (Hình 14) và tính được các giá trị động học ngược $X_b = 0.125$, $Y_b = -0.37$, $Z_b = 0.27$ (Bảng 3).

Bảng động học ngược đã tính ra các giá trị: $X_b = 0.125$, $Y_b = -0.37$, $Z_b = 0.27$, sau đó trả về các giá trị động học thuận $d_1 = 17$ cm, $\theta_2 = -45.01^\circ$, $\theta_3 = -59.98^\circ$ (Bảng 4).



Hình 11. Giao diện sau khi khởi tạo mô hình Robot trong ROS 2



Hình 12. Nhập các giá trị động học (giá trị dương)

Bảng 3. Đánh giá sai số động học thuận

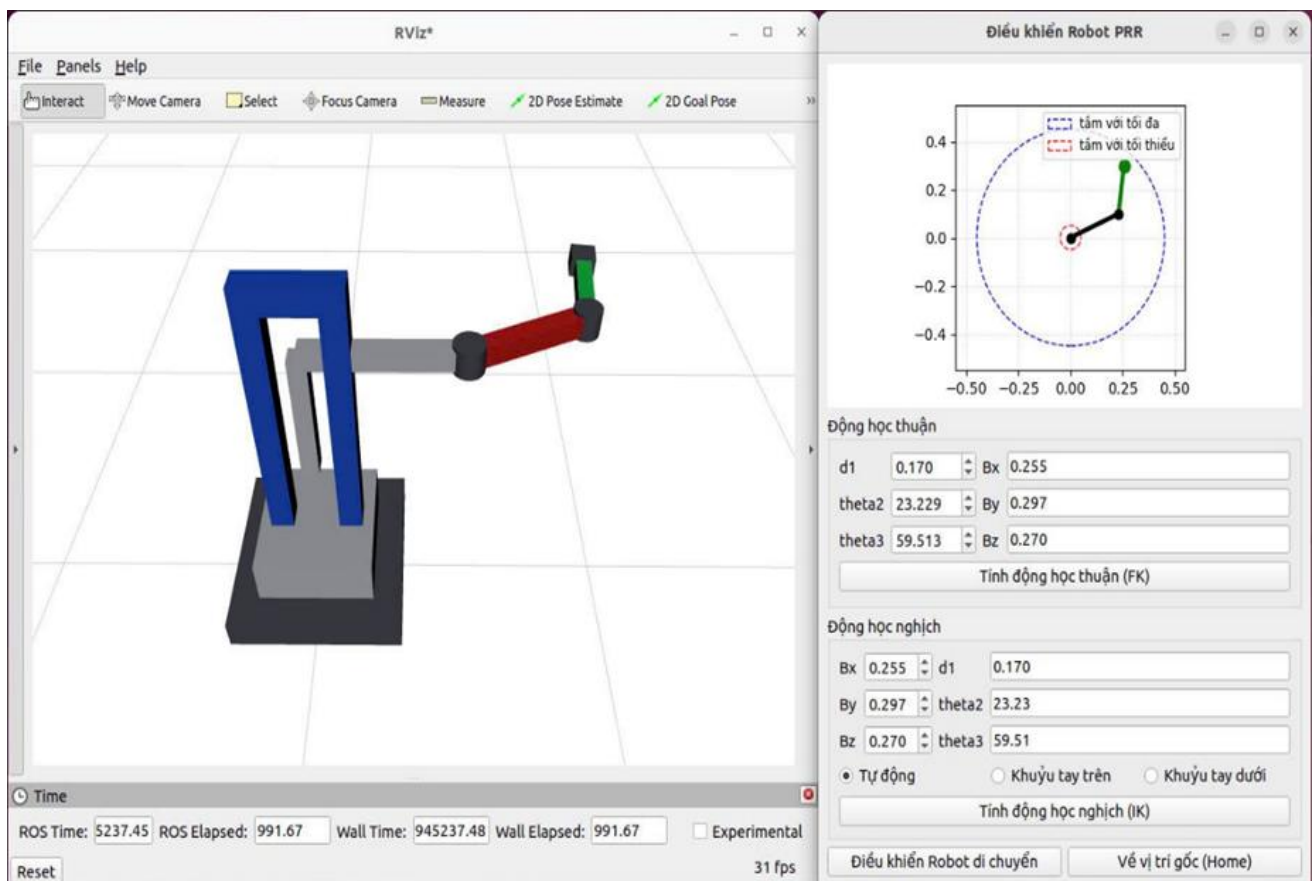
Trường hợp kiểm thử	Thông số (d_1, θ_2, θ_3)	Tọa độ (x, y, z)	Sai số
Hình 12	(25cm, 30^0 , 70^0)	(0.182m, 0.322m, 0.35m)	Lệch $\approx 0.06^0$
Hình 13	(17cm, 23.27^0 , 59.5^0)	(0.255m, 0.297m, 0.27m)	Lệch $\approx 0.04^0$
Hình 14	(17cm, -45^0 , -60^0)	(0.125m, -0.37m, 0.27m)	Lệch $\approx 0.02^0$

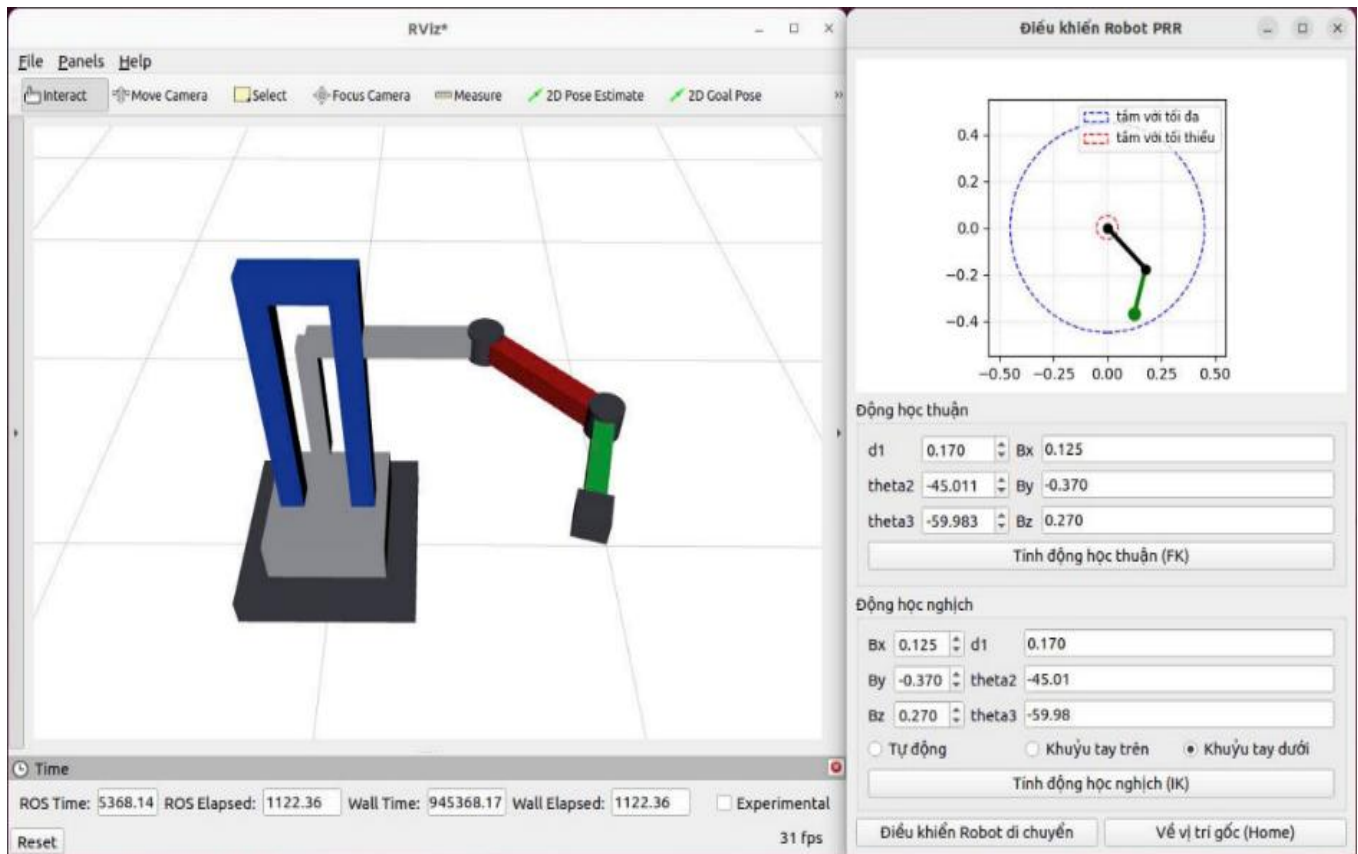
Bảng so sánh sai số FK (Thuận) và IK (Ngược)

Kết quả tại Bảng 3 cho thấy sai số động học thuận dao động $0,02^0$ – $0,06^0$ trên cả ba trường hợp kiểm thử (Hình 12–14), nhỏ hơn dung sai góc quay thường gặp ở động cơ servo trong mô hình giáo dục ($0,1^0$ – $0,5^0$), cho thấy mô hình D-H đủ chính xác làm nền tảng điều khiển, dù chưa đạt ngưỡng khắt khe của Robot công nghiệp (dưới $0,01^0$). Sai số này chủ yếu đến từ (1) làm tròn số khi chuyển đổi tọa độ Descartes sang góc khớp trong giải động học ngược bằng phương pháp hình học, và (2) độ phân giải hiển thị góc trên giao diện control_gui.py chưa đạt độ chính xác thập phân đầy đủ, có thể khắc phục bằng cách tăng độ phân

giải tính toán và bổ sung thuật toán hiệu chỉnh ở giai đoạn sau.

So với các nghiên cứu mô hình bản sao số bằng các phần mềm khác như RoboDK, MATLAB... ROS 2 tập trung vào đồng bộ hai chiều thời gian thực giữa Robot công nghiệp/cộng tác và môi trường ảo, mô hình trong nghiên cứu này mới dừng ở mức mô phỏng động học thuần túy, chưa tích hợp động lực học hay phản hồi cảm biến thực [14]. Do đó, sai số tại Bảng 3, Bảng 4 chỉ phản ánh độ chính xác thuật toán và mô hình hình học, chưa bao gồm sai số động lực học thực tế (quán tính, ma sát, độ trễ cơ khí), đây là giới hạn cần được lưu ý khi so sánh kết quả của nghiên cứu này với các mô hình bản sao số hoàn chỉnh đã công bố.

**Hình 13.** Nhập các giá trị động học (giá trị góc lẻ)



Hình 14. Nhập các giá trị động học (giá trị góc âm).

Bảng 4. Đánh giá chất lượng động học ngược.

Trường hợp kiểm thử	Tọa độ (x, y, z)	Chế độ lựa chọn trên GUI	Góc tính ra bởi ROS 2 (d_1, θ_2, θ_3)	Nhận xét Thuật toán
Góc dương	(0.182m, 0.322m, 0.35m)	ELBOW - UP	(25cm, 30°, 70°)	$\sin \theta_2 > 0$. Tay gấp về phía trái/lên. Chuẩn xác.
Góc âm	(0.125m, -0.37m, 0.27m)	ELBOW - DOWN	(17cm, -45°, -60°)	$\sin \theta_2 < 0$. Tay gấp về phía phải/xuống. Cùng tới 1 điểm. Chuẩn xác.
Nằm ngoài tầm với	(0.255m, 0.297m, 0.27m)	Không khả dụng	Không chạy	Cảnh báo được vùng cấm

Tóm lại, trong không gian làm việc của mô hình SCARA, tại một vị trí tọa độ (x, y, z) bất kỳ, phương trình giải động học ngược luôn trả về hai bộ nghiệm góc khớp θ_2, θ_3 tương đương với hai vế $\pm$ của hàm $\sin \theta_2$. Hệ quả hình học của hai bộ nghiệm này là hai cấu hình tay máy: Elbow-up (Khớp khuỷu gấp theo chiều dương) và Elbow-

down (Khớp khuỷu gấp theo chiều âm). Trong giao diện điều khiển nhúng trên ROS 2, nhóm nghiên cứu đã thiết lập thành công cơ chế cho phép lựa chọn bộ nghiệm. Khi cần tiếp cận vị trí có chướng ngại vật nhất định, việc linh hoạt thay đổi giữa hai cấu hình này giúp không gian làm việc của Robot an toàn và tối ưu hơn, tránh hiện tượng va chạm

cơ khí (Bảng 4).

2.3. Hướng xây dựng phần cứng thực tế

Để tối ưu hóa quá trình phát triển, nhóm nghiên cứu áp dụng quy trình từ thiết kế mô phỏng đến chế tạo thực nghiệm, sử dụng phần mềm SolidWorks với các giai đoạn sau:

2.3.1. Thiết kế mô hình 3D

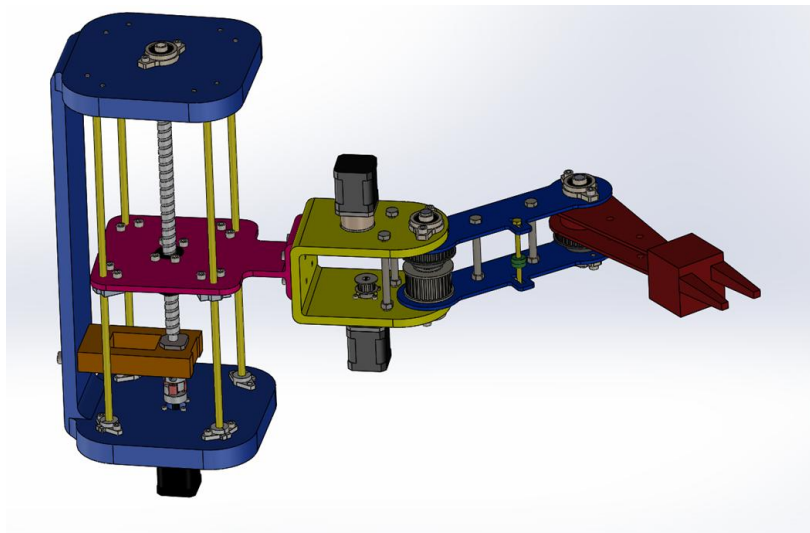
SolidWorks được dùng để xây dựng và lắp ráp chi tiết các cụm cơ cấu của Robot (Hình 15):

Cơ cấu nâng hạ (Trục Z): Sử dụng thanh dẫn hướng và trục vít me dẫn động bằng động cơ, giúp

bộ Robot di chuyển tịnh tiến theo phương thẳng đứng.

Khớp xoay và truyền động: Các khớp được liên kết bằng giá đỡ kim loại (bracket). Thiết kế sử dụng bộ truyền đai răng (timing belt) nhằm đảm bảo chuyển động xoay ổn định và mượt mà.

Cơ cấu chấp hành: Tích hợp tay gấp hai ngón (gripper) ở đầu cuối để gấp và thao tác với phôi. Việc thiết kế 3D giúp kiểm soát chính xác kích thước và khả năng tương thích của các chi tiết trước khi gia công.



Hình 15. Mô hình 3D của Robot SCARA trên SolidWorks

2.3.2. Mô phỏng động học

Mô hình hoàn chỉnh được chạy mô phỏng chuyển động trên phần mềm để đánh giá không gian làm việc (workspace) và phát hiện sớm các rủi ro va chạm cơ khí. Bước này giúp giảm thiểu tối đa sai sót thiết kế, tiết kiệm vật tư và thời gian gia công.

2.3.3. Gia công và lắp ráp thực tế

Từ mô hình số, nhóm nghiên cứu xuất bản vẽ kỹ thuật 2D để tiến hành gia công cơ khí (phay CNC, in 3D...). Các linh kiện sau đó được lắp ráp, đấu nối mạch điều khiển và chạy thử nghiệm. Dữ liệu vận hành thực tế sẽ được so sánh với kết quả mô phỏng để tinh chỉnh độ chính xác.

Kết luận: Sự kết hợp giữa thiết kế số và chế tạo thực tế cho phép kiểm tra trước tương thích cơ khí giữa các cụm chi tiết, qua đó giảm số lần thử-sai trong giai đoạn gia công thực tế và hỗ trợ trực

quan hóa quá trình thiết kế phục vụ giảng dạy.

3. Kết luận và hướng phát triển

Đóng góp của nghiên cứu: Bài báo đã thiết kế và tính toán động học thành công mô hình cánh tay Robot SCARA 3 bậc tự do. Trên nền tảng mã nguồn mở ROS 2, nhóm tác giả đã đóng gói mô hình thành một quy trình trực quan, bao gồm các bước từ thiết lập Workspace, thông số URDF đến giao diện điều khiển GUI kiểm chứng trên RViz. Kết quả nghiên cứu đóng vai trò như một học liệu thực hành tiêu chuẩn, đáp ứng trực tiếp nhu cầu phương pháp giáo dục STEM cho sinh viên NCKH tại trường đại học [2].

Hạn chế của nghiên cứu: Tuy mô phỏng đạt độ chính xác về mặt quỹ đạo, mô hình hiện tại mới chỉ dừng lại ở mức mô phỏng chuyển động thuần túy (Kinematics Simulation). Do chưa có cơ chế đồng bộ tín hiệu hai chiều (Bi-directional

synchronization) và chưa tích hợp vật lý động lực học từ phần mềm Gazebo, hệ thống chưa đo lường được các sai số thực tế và chưa đáp ứng đầy đủ tiêu chuẩn của một bản sao số hoàn chỉnh [14].

Hướng phát triển tiếp theo: Để khắc phục các nhược điểm trên, hệ thống sẽ tiếp tục được nâng cấp với ba mục tiêu chính ở giai đoạn sau:

- Chế tạo phần cứng: Lắp ráp mô hình vật lý hoàn chỉnh nhằm chạy thực nghiệm và đo lường định lượng các sai số định vị.

- Mô phỏng động lực học: Tích hợp môi trường vật lý đa hệ Gazebo (Gazebo Physics Simulation) để tính toán lực quán tính, ma sát, độ trễ và trọng lượng thực tế.

- Giám sát khép kín: Sử dụng trình điều khiển ros 2_control để cập nhật tín hiệu phản hồi từ cảm biến thực (Sensor Feedback) về lại không gian ảo, hoàn thiện môi trường bản sao số 2 chiều.

Lời cảm ơn

Nghiên cứu này được tài trợ từ chương trình nghiên cứu khoa học sinh viên của Trường Đại học Giao thông Vận tải Thành phố Hồ Chí Minh. Nhóm tác giả chân thành cảm ơn sự hỗ trợ tài chính và thiết bị thí nghiệm của Ban Công nghệ Chiến lược, Trường Đại học Giao thông Vận tải Thành phố Hồ Chí Minh cho đề tài “Thiết kế, chế tạo mô hình cánh tay Robot hệ cấu trúc SCARA kết hợp mô phỏng kỹ thuật số sử dụng ROS 2 nhằm phục vụ công tác giảng dạy học phần Robot công nghiệp và xây dựng câu lạc bộ STEM tại UTH” mã số KHSV2616.

Tuyên bố về việc sử dụng AI tạo sinh

Trong quá trình chuẩn bị bài báo này, nhóm tác giả đã sử dụng NotebookLM và Gemini AI nhằm cải thiện khả năng đọc hiểu và diễn đạt ngôn ngữ của bản thảo. Sau khi sử dụng các công cụ này, nhóm tác giả đã rà soát và chỉnh sửa nội dung khi cần thiết, đồng thời chịu hoàn toàn trách nhiệm về nội dung của bài báo được công bố.

Tài liệu tham khảo

[1] U.S. Department of Education, Office of Innovation and Improvement. (2016). STEM 2026: A Vision for Innovation in STEM

Education, Washington, DC, USA. https://www.ed.gov/sites/ed/files/2016/09/AIR-STEM2026_Report_2016.pdf (truy cập ngày 07/05/2026)

[2] Partnership for 21st Century Learning. (2015). P21 Framework Definitions. Tucson, AZ, USA. https://www.marietta.edu/sites/default/files/documents/21st_century_skills_standards_book_2.pdf (truy cập ngày 07/05/2026)

[3] V. Hà. (2025). Đột phá theo Nghị quyết 57 – nâng tỷ lệ sinh viên theo học các ngành STEM. *Thông tấn xã Việt Nam*. <https://baotintuc.vn/giao-duc/dot-pha-theo-nghi-quyet-57-nang-ty-le-sinh-vien-theo-hoc-cac-nganh-stem-20250131104621528.htm> (truy cập ngày 07/05/2026).

[4] Thủ tướng Chính phủ. (2025). Quyết định số 1131/QĐ-TTg ngày 12/06/2025 của Thủ tướng Chính phủ ban hành Danh mục công nghệ chiến lược và sản phẩm công nghệ chiến lược. <https://vanban.chinhphu.vn/?pageid=27160&docid=213929> (truy cập ngày 07/05/2026)

[5] D. Kawshan, Q. Peng. (2026). Digital Twin Framework for Robot Path Planning and Real-Time Execution Using Unity-ROS Integration: Systems Architecture and Experimental Validation. *Machines*, 14(4), 387. <https://doi.org/10.3390/machines14040387>

[6] E. Coronado, T. Ueshiba, I.G. Ramirez-Alpizar. (2024). A Path to Industry 5.0 Digital Twins for Human–Robot Collaboration by Bridging NEP+ and ROS. *Robotics*, 13(2), 28. <https://doi.org/10.3390/robotics13020028>

[7] H. Sekkat, O. Moutik, B.El Kari, Y. Chaibi, T. Ait Tchakoucht, A.El Hachimi Alaoui. (2024). Beyond simulation: Unlocking the frontiers of humanoid Robot capability and intelligence with Pepper's open-source digital twin. *Heliyon*, 10(14), e34456. <https://doi.org/10.1016/j.heliyon.2024.e34456>

- [8] Open Robotics. (2022). Robot Operating System 2 (ROS 2) Humble Hawksbill Documentation. <https://docs.ros.org/en/humble/index.html> (truy cập ngày 07/05/2026)
- [9] J. Denavit, R.S. Hartenberg. (1955). A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices. *Journal of Applied Mechanics*, 22(2), 215–221. <https://doi.org/10.1115/1.4011045>
- [10] J.J. Craig. (2018). Introduction to Robotics: Mechanics and Control, 4th ed. Pearson.
- [11] M.W. Spong, S. Hutchinson, M. Vidyasagar. (2020). Robot Modeling and Control, 2nd ed. John Wiley & Sons.
- [12] Oracle. (2024). Oracle VM VirtualBox User Guide for Release 7.0. <https://docs.oracle.com/en/virtualization/virtualbox/7.0/user/EN-VBOX-7-0-USER.pdf> (truy cập ngày 07/05/2026)
- [13] Canonical. (2022). Ubuntu 22.04 LTS (Jammy Jellyfish) Release Notes, Canonical Ltd, <https://documentation.ubuntu.com/release-notes/22.04/> (truy cập ngày 07/05/2026)
- [14] W. Chang, W. Sun, P. Chen, H. Xu. (2025). Construction and Validation of a Digital Twin-Driven Virtual-Reality Fusion Control Platform for Industrial Robots. *Sensors*, 25(13), 4153. <https://doi.org/10.3390/s25134153>