Nghiên cứu thiết kế bộ điều khiển và xây dựng giải thuật điều khiển cho robot 6 bậc tự do ứng dụng trong đào tạo

Phùng Trí Công*, Nguyễn Tấn Tiến, Nguyễn Tấn Đạt, và Nguyễn Ngọc Sơn

***Tóm tắt*—Bài báo này trình bày một phương pháp thiết kế bộ điều khiển và xây dựng giải thuật điều khiển cho các robot công nghiệp nhằm mục đích xây dựng các bài thí nghiệm phục vụ cho đào tạo. Robot được sử dụng trong bài báo là robot SV3X của hãng MOTOMAN. Trong nghiên cứu này, nhóm sẽ trình bày phương pháp thiết kế và phục hồi bộ điều khiển cho một robot đã qua sử dụng. Một giao diện mới điều khiển robot cũng được thiết kế dựa trên nền tảng ngôn ngữ C#. Kế đến nhóm sẽ đề xuất các giải thuật có thể thực hiện được dựa trên bộ điều khiển thiết kế. Hai giải thuật được kiểm chứng là lập trình điều khiển robot bám theo quỹ đạo cho trước và hoạch định quỹ đạo cho quá trình gắp và thả vật. Cuối cùng, độ chính xác lặp lại của robot cũng như các giải thuật đề nghị sẽ được kiểm chứng bằng thực nghiệm.**

***Từ khóa*— robot công nghiệp, bộ điều khiển, giải thuật điều khiển, bám quỹ đạo, gắp vật**

1 GIỚI THIỆU

Tự động hóa các dây chuyền sản xuất trong các nhà máy và xí nghiệp là một nhu cầu tất yếu của nước ta hiện nay. Trong đó các robot công nghiệp hay các tay máy đóng vai trò quan trọng trong nền công nghiệp tự động hóa. Vì vậy việc làm quen với các robot công nghiệp trong quá trình học đại học là rất cần thiết đối với các sinh viên đại học. Những kiến thức này sẽ rất hữu ích với những kỹ sư điều khiển robot sau này.

Tuy nhiên giá thành của các robot công nghiệp hiện nay là khá cao so với khả năng mua sắm ở các trường học. Để giải quyết vấn đề này thì có 2 phương án được đề xuất: thứ nhất là tìm mua các robot công nghiệp đã qua sử dụng và phục hồi bộ điều khiển cho robot và thứ hai là chế tạo mới các mô hình robot mô phỏng lại hoạt động của các robot công nghiệp. Trong bài báo này sẽ tập trung vào phương án thứ nhất. Nhóm nghiên cứu tiếp cận bài toán theo hướng tự thiết kế bộ điều khiển cho robot và xây dựng giao diện điều khiển dựa trên ngôn ngữ C# là ngôn ngữ lập trình cơ bản.

Một ưu điểm nữa của phương án tự thiết kế bộ điều khiển là tăng cường khả năng tiếp nhận và xử lý dữ liệu từ môi trường bên ngoài. Đối với các robot công nghiệp hiện nay, khả năng can thiệp sâu vào driver của robot không được hỗ trợ bởi các nhà sản xuất. Các lệnh hỗ trợ trong ngôn ngữ lập trình thường bị giới hạn, đặc biệt là khi muốn nhận tín hiệu từ cảm biến bên ngoài để xử lý.

Đã có nhiều nghiên cứu trên thế giới tìm hiểu về việc thiết kế bộ điều khiển mới cho robot công nghiệp. Các nhà khoa học ở Đại học Brown đã phát triển hệ thống SIERA cho phép điều khiển trực tiếp các khớp của robot từ các servo drive [1]. Các nhà khoa học ở Đại học Toronto thiết kế một bộ điều khiển mới dựa trên nền tảng là một máy tính TUNIS để thay thế cho bộ điều khiển của robot PUMA [2]. Các nhà nghiên cứu ở Đại học Jaen xây dựng một bộ điều khiển mới có khả năng kết nối với bộ điều khiển cũ của robot công nghiệp STAUBLi RX60 [3].

Về tình hình trong nước thì có công ty TOSY

Ngày nhận bản thảo: 03-4-2018, ngày chấp nhận đăng: 25-8-2018, ngày đăng: 30-11-2018.

Nghiên cứu được tài trợ bởi Đại học Quốc gia Thành phố Hồ Chí Minh (ĐHQG-HCM) trong khuôn khổ Đề tài mã số C2017-20-04.

Phùng Trí Công hiện đang công tác tại Trường Đại học Bách Khoa – ĐHQG-HCM, 268 Lý Thường Kiệt, P. 14, Q. 10, TP. Hồ Chí Minh, Việt Nam (e-mail: ptcong@hcmut.edu.vn).

Nguyễn Tấn Tiến hiện đang công tác tại Trường Đại học Bách Khoa – ĐHQG-HCM, 268 Lý Thường Kiệt, P. 14, Q. 10, TP. Hồ Chí Minh, Việt Nam (e-mail: nttien@hcmut.edu.vn).

Nguyễn Tấn Đạt hiện đang công tác tại Công ty TNHH Bosch Việt Nam, Đường số 8, Khu công nghiệp Long Thành, Xã Tam An, huyện Long Thành, Đồng Nai, Việt Nam (e-mail: nguyentandat295@gmail.com).

Nguyễn Ngọc Sơn hiện đang công tác tại Công ty Cổ phần Tập đoàn Hoa Sen, Số 9, Đại lộ Thống Nhất, Khu công nghiệp Sóng Thần II, phường Dĩ An, thị xã Dĩ An, Bình Dương, Việt Nam (e-mail: nguyenngocson9399@gmail.com).

và công ty Chế tạo máy AKB là 2 công ty dẫn đầu về thiết kế và chế tạo mới robot. Tuy nhiên nghiên cứu về xây dựng các bài thí nghiệm cho sinh viên đại học là rất hạn chế. Ở Việt Nam nhóm nghiên cứu chưa thấy nghiên cứu nào về việc tự thiết kế bộ điều khiển và xây dựng giao diện điều khiển dựa trên ngôn ngữ C# để phục vụ cho các bài thí nghiệm về robot cho sinh viên.

Mục tiêu bài báo là thiết kế bộ điều khiển và xây dựng giải thuật điều khiển cho một robot công nghiệp. Mục 2 giới thiệu về robot công nghiệp MOTOMAN SV3X, là loại robot công nghiệp được sử dụng trong bài báo, và phương án phục hồi robot. Mục 3 trình bày phương án thiết kế bộ điều khiển cho robot. Mục 4 trình bày giao diện điều khiển cũng như các giải thuật điều khiển robot. Mục 5 trình bày kết quả thực nghiệm các giải thuật trên cũng như kiểm tra độ chính xác lặp lại của robot.

2 PHƯƠNG ÁN PHỤC HỒI ROBOT MOTOMAN SV3X

Robot MOTOMAN SV3X do hãng Yaskawa chế tạo và sản xuất. Khác với những nhà sản xuất khác, các robot của Yaskawa được xây dựng và chế tạo dựa trên nền tảng các thiết bị truyền động nổi tiếng do chính Yaskawa cung cấp. Do vậy các robot của Yaskawa đạt được độ chính xác, tốc độ di chuyển và độ tin cậy rất cao và đã được ứng dụng không chỉ ở Việt Nam mà còn trên toàn thế giới.

2.1 Thực trạng robot MOTOMAN SV3X

Robot MOTOMAN SV3X được sử dụng trong bài báo có 6 bậc tự do, hoạt động hoàn toàn bằng hệ thống điện. Tải trọng cho phép tối đa của robot là 3 kg và độ chính xác lặp lại của robot là 0,03mm. Tuy nhiên, robot được sản xuất khá lâu và đã được dùng trong thực tế nên đã bị hư hao đáng kể. Phần cơ khí của robot tuy đã cũ nhưng vẫn còn sử dụng được mặc dù đã có sai số ở các bộ truyền động.

Phần điện công suất của robot MOTOMAN SV3X gồm 6 động cơ AC servo đi kèm với 2 bộ servopack, trong đó mỗi bộ servopack điều khiển 3 động cơ. Tuy nhiên, các servopack này đã hư và không thể xuất tín hiệu điều khiển được nên cần thay thế. Bộ điều khiển trung tâm của robot là một board máy tính mini và cũng không còn hoạt động nên cũng cần thay thế. Ngoài ra, phần mềm điều khiển của robot cũng không còn được hỗ trợ nên cần được xây dựng mới.

2.2 Phương án phục hồi robot

Trong phần này sẽ trình bày phương án phục hồi bộ điều khiển cho robot, cả phần cứng lẫn phần mềm.

Ở phần điện công suất, nhóm nghiên cứu thay thế các động cơ cũ bằng 6 động cơ AC servo của hãng Yaskawa đi kèm với 6 bộ servopack độc lập. Mỗi động cơ được gắn liền với một servopack tương ứng.

Ở phần điều khiển, nhóm sử dụng một máy tính cá nhân làm bộ điều khiển trung tâm. Máy tính này sẽ tính toán và xử lý các giải thuật để điều khiển robot. Sau đó, máy tính sẽ truyền tín hiệu điều khiển xuống 1 vi điều khiển Master và vi điều khiển Master này có nhiệm vụ xử lý và truyền tín hiệu đến các vi điều khiển Slave. Mỗi vi điều khiển Slave sẽ phụ trách điều khiển độc lập một động cơ AC servo.

Ở phần mềm điều khiển, nhóm xây dựng một giao diện sử dụng dựa trên ngôn ngữ C#. Trong đó, sẽ có những chức năng cơ bản nhất cho việc điều khiển một robot công nghiệp. Hai chế độ người sử dụng có thể dùng là điều khiển bằng tay hoặc tự động.

2.3 Bài toán động học robot

Bài toán động học thuận robot được thể hiện trong mục này. Hình 1 thể hiện sơ đồ đặt hệ tọa độ lên robot MOTOMAN SV3X. Các thông số DH của robot được thể hiện trong bảng 1.

Ma trận chuyển đổi từ hệ tọa độ công tác về hệ tọa độ toàn cục được thể hiện trong biểu thức sau:

$$ {}^{0}T_7 = {}^{0}T_1\,{}^{1}T_2\,{}^{2}T_3\,{}^{3}T_4\,{}^{4}T_5\,{}^{5}T_6\,{}^{6}T_7 \quad (1) $$

$$ {}^{0}T_7 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2) $$

Trong đó các thành phần chỉ hướng được thể hiện như sau:

$$ \begin{aligned} r_{11} &= s_1\left(s_4c_5c_6 + c_4s_6\right) + c_1\left(c_{23}\left(c_4c_5c_6 - s_4s_6\right) - s_{23}s_5c_6\right) \\ r_{21} &= -c_1\left(s_4c_5c_6 + c_4s_6\right) + s_1\left(c_{23}\left(c_4c_5c_6 - s_4s_6\right) - s_{23}s_5c_6\right) \\ r_{31} &= c_{23}s_5c_6 + s_{23}\left(c_4c_5c_6 - s_4s_6\right) \\ r_{12} &= s_1\left(c_4c_6 - s_4c_5s_6\right) - c_1\left(c_{23}\left(s_4c_6 + c_4c_5s_6\right) - s_{23}s_5s_6\right) \\ r_{22} &= -c_1\left(c_4c_6 - s_4c_5s_6\right) - s_1\left(c_{23}\left(s_4c_6 + c_4c_5s_6\right) - s_{23}s_5s_6\right) \\ r_{32} &= -s_{23}\left(c_4c_5s_6 + s_4c_6\right) - c_{23}s_5s_6 \\ r_{13} &= s_1s_4s_5 + c_1\left(c_{23}c_4s_5 + s_{23}c_5\right) \\ r_{23} &= -c_1s_4s_5 + s_1\left(c_{23}c_4s_5 + s_{23}c_5\right) \\ r_{33} &= s_{23}c_4s_5 - c_{23}c_5 \end{aligned} \quad (3) $$

Bảng 1
Thông số DH của robot MOTOMAN SV3X

i	a_i (mm)	α_i (độ)	d_i (mm)	θ_i (độ)
1	150	90	300	θ_1
2	260	0	0	θ_2
3	60	90	0	θ_3
4	0	-90	260	θ_4
5	0	90	0	θ_5
6	0	0	0	θ_6
7	0	0	330	0

Vị trí của đầu công tác trong hệ tọa độ toàn cục là:

$$X = d_7 x_1 + c_1\left(a_1 + a_3 c_{23} + a_2 c_2\right) + d_4 s_{23} c_1$$
$$Y = d_7 y_1 + s_1\left(a_1 + a_3 c_{23} + a_2 c_2\right) + d_4 s_{23} s_1$$
$$Z = d_7\left(s_{23} c_4 s_5 - c_{23} c_5\right) + d_1 - d_4 c_{23} + a_3 s_{23} + a_2 s_2 \quad (4)$$

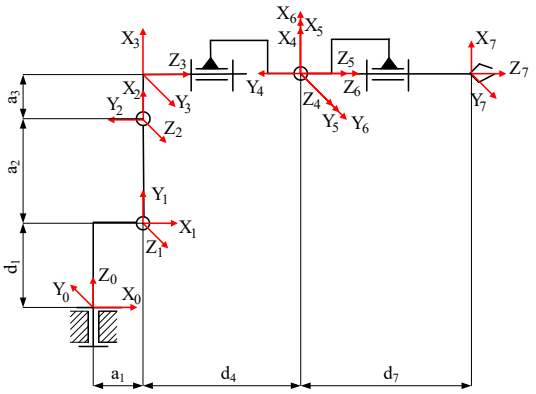
Với:

$$x_1 = c_1\left(s_{23} c_5 + c_{23} c_4 s_5\right) + s_1 s_4 s_5$$
$$y_1 = s_1\left(s_{23} c_5 + c_{23} c_4 s_5\right) - c_1 s_4 s_5$$

Trong đó:

$$s_i = \sin\theta_i, c_i = \cos\theta_i$$
$$s_{23} = \sin\left(\theta_2 + \theta_3\right), c_{23} = \cos\left(\theta_2 + \theta_3\right)$$

Từ kết quả của bài toán động học thuận chúng ta có thể giải bài toán động học ngược tương ứng của robot MOTOMAN SV3X. Đối với loại tay máy 6 khớp có 3 khớp cuối đồng quy tại một điểm này, chúng ta có thể tách bài toán động học ngược thành 2 bài toán đơn giản hơn là động học ngược về vị trí và động học ngược về hướng. Tức là chúng ta sẽ tìm vị trí giao điểm các trục cổ tay (tâm cổ tay) và tìm hướng của cổ tay. Từ đó chúng ta sẽ tìm được các giá trị tương ứng tại các khớp của robot.



Hình 1. Sơ đồ đặt hệ tọa độ trên robot MOTOMAN SV3X

3 THIẾT KẾ BỘ ĐIỀU KHIỂN ROBOT

Trong phần này, nhóm sẽ trình bày cụ thể phần thiết kế mạch công suất và phần điều khiển của robot. Nhóm sẽ thay thế các thiết bị hư bằng các thiết bị mới hoạt động bình thường. Đồng thời nhóm sẽ xây dựng phần điều khiển cho các thiết bị mới này.

3.1 Phục hồi phần mạch công suất cho robot

Nhóm nghiên cứu đã tiến hành thay thế các động cơ cũ của robot bằng các động cơ AC servo mới đi kèm với các servopack điều khiển độc lập. Động cơ AC servo được sử dụng để phục hồi robot là các loại động cơ SGMPH và động cơ SGMAH. Các loại servopack đi kèm với các động cơ trên được sử dụng trong bài báo gồm SGDA, SGDJ và SGDP.

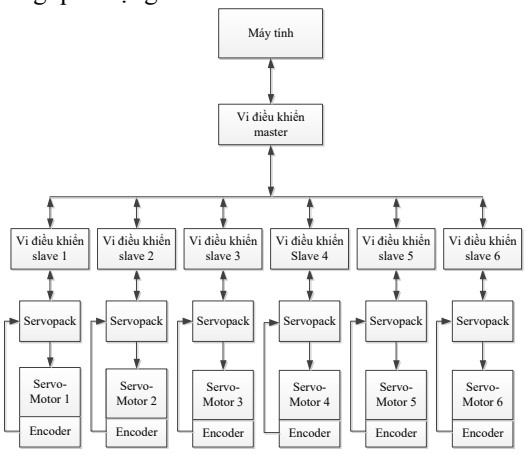
Nguyên lý điều khiển các bộ servopack đều giống nhau là đều điều khiển theo dạng cấp xung. Dạng xung có duty là 50% và vận tốc được quyết định bởi tần số xung. Khi tần số xung càng lớn thì động cơ xoay càng nhanh và ngược lại, khi tần số xung càng bé thì động cơ quay càng chậm. Việc điều khiển đáp ứng vận tốc của động cơ do bộ servopack đảm nhận. Vì vậy, ta có thể điều khiển vận tốc động cơ bằng cách điều khiển tần số cấp xung cho từng bộ servopack riêng biệt.

3.2 Thiết kế phần điều khiển cho robot

Phần điều khiển của robot bao gồm một máy tính cá nhân và các vi điều khiển (VĐK), được thể hiện trong hình 2. Máy tính đóng vai trò trung tâm trong việc điều khiển robot từ việc giải các bài toán động học thuận, động học ngược, động lực học đến việc xử lý tín hiệu từ các cảm biến ngoại vi và tính toán đưa ra tín hiệu điều khiển cho robot. Sau khi tính toán xong máy tính sẽ truyền tín hiệu điều khiển xuống một vi điều khiển Master. VĐK Master này nhận tín hiệu điều khiển từ máy tính và tiến hành xử lý dữ liệu. Sau đó VĐK Master sẽ truyền các tín hiệu điều khiển đến các VĐK Slave. Các VĐK Slave này sẽ xuất tín hiệu để điều khiển trực tiếp các động cơ AC servo.

Về vấn đề giao tiếp thì giao tiếp giữa máy tính chủ và VĐK Master là giao tiếp RS232. Máy tính chủ sẽ tính toán và truyền tín hiệu điều khiển xuống VĐK Master thông qua chuẩn giao tiếp nối tiếp này. Sau đó VĐK Master sẽ truyền tín hiệu điều khiển đến các VĐK Slave thông qua chuẩn giao tiếp CAN, là chuẩn giao tiếp công nghiệp được sử dụng phổ biến hiện nay. VĐK được lựa

chọn sử dụng trong nghiên cứu này là PIC18F4580. Đây là loại VĐK đã có sẵn module giao tiếp CAN tích hợp bên trong nên chúng ta chỉ cần sử dụng thêm IC MCP2551 để chuyển đổi mức điện áp tín hiệu phù hợp trong CAN bus. Hai chân của VĐK PIC18F4580 nối với IC MCP2551 là chân RB2/CANTX và chân RB3/CANRX. Cả VĐK PIC18F4580 và IC MCP2551 sẽ tạo thành một node CAN và sẽ giao tiếp với các VĐK khác thông qua mạng CAN bus.



Hình 2. Sơ đồ kết nối mạch điện của hệ thống

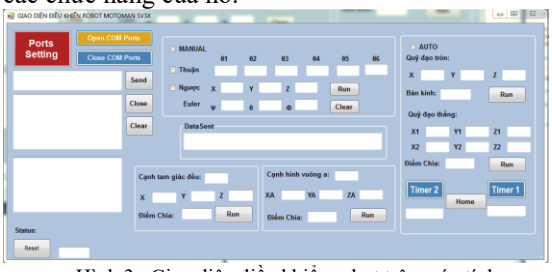
Trong nghiên cứu này, nhóm tác giả tập trung vào việc điều khiển vị trí của robot nên các VĐK Slave sẽ xuất xung để điều khiển các động cơ AC servo. Timer1 được sử dụng để cấp xung cho các servopack của động cơ AC servo. Cụ thể, chân RD0 dùng để cấp xung PWM và chân RD1 dùng để đảo chiều động cơ. Bản thân các servopack đã tích hợp các bộ điều khiển vòng kín để điều khiển chính xác vị trí động cơ. Do đó VĐK Slave chỉ cần cấp số xung tương ứng là có thể điều khiển giá trị quay tại các khớp của robot.

4 XÂY DỰNG GIAO DIỆN VÀ GIẢI THUẬT ĐIỀU KHIỂN ROBOT

Trong mục này, bài báo sẽ trình bày giao diện được sử dụng để điều khiển robot. Ngôn ngữ được sử dụng trong nghiên cứu này là ngôn ngữ C#, một trong những ngôn ngữ phổ biến hiện nay. Từ đó các giải thuật điều khiển robot 6 bậc tự do cũng được đề xuất.

4.1 Giao diện điều khiển robot

Bộ điều khiển robot mà nhóm nghiên cứu xây dựng là sự kết hợp giữa máy tính và vi điều khiển. Vì vậy, việc xây dựng chương trình điều khiển trên máy tính là cần thiết để đáp ứng yêu cầu của bộ điều khiển. Phần này sẽ đi sâu vào việc trình bày quá trình xây dựng giao diện điều khiển và các chức năng của nó.



Hình 3. Giao diện điều khiển robot trên máy tính

Hình 3 thể hiện giao diện điều khiển của robot trên máy tính. Giao diện điều khiển phải có chức năng kết nối cổng COM, trong quá trình điều khiển, dữ liệu từ máy tính liên tục được truyền xuống vi điều khiển Master. Đầu tiên người sử dụng nhấn nút "Ports Setting" để khởi tạo các thông số kết nối giữa vi điều khiển Master và máy tính như: lựa chọn cổng kết nối, khai báo tốc độ truyền, khai báo các thông số cần thiết cho việc truyền nhận dữ liệu nối tiếp.

Sau khi lựa chọn xong, người sử dụng nhấn nút "Open COM ports" để mở cổng COM, lúc này thanh trạng thái "Status" sẽ thông báo kết nối thành công, lúc này máy tính có thể truyền nhận dữ liệu với vi điều khiển Master.

Sau khi kết nối thiết bị xong, người sử dụng sẽ đi vào lựa chọn chế độ điều khiển. Có 2 chế độ điều khiển là Manual và Auto. Ở chế độ Manual, người sử dụng có thể điều khiển robot theo các phương trình động học thuận và ngược. Ở chế độ Auto người sử dụng có thể điều khiển robot bám theo quỹ đạo cho trước. Các dạng quỹ đạo cụ thể được xây dựng trên giao diện là: quỹ đạo đường thẳng, quỹ đạo hình vuông, quỹ đạo hình tròn, và quỹ đạo hình tam giác đều.

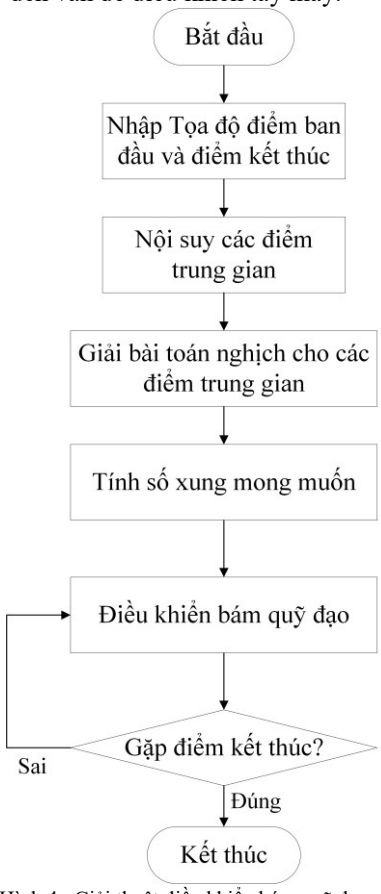
4.2 Cấu trúc Frame truyền của robot

Một vấn đề rất quan trọng của hầu hết các hệ thống điều khiển đó là việc truyền nhận dữ liệu. Đặc biệt trong hệ thống điều khiển robot, vấn đề giao tiếp giữa nhiều vi điều khiển hay giữa máy tính với vi điều khiển ảnh hưởng rất lớn đến kết quả đạt được. Phần này sẽ trình bày cách thức giao tiếp giữa máy tính và vi điều khiển thông qua chuẩn giao tiếp RS232, giao tiếp và truyền nhận dữ liệu giữa các vi điều khiển thông qua giao thức CAN.

Theo phương pháp điều khiển phân tích ở trên, máy tính sẽ đảm nhiệm toàn bộ công việc tính toán và giải quyết các thuật toán, sau đó sẽ truyền dữ liệu điều khiển xuống vi điều khiển Master, dữ liệu này chỉ gồm 1 byte tín hiệu hướng và 5 bytes

tín hiệu xung điều khiển động cơ. Với hệ thống có 6 vi điều khiển Slave tương ứng dùng để điều khiển 6 động cơ, như vậy, dữ liệu truyền xuống cho vi điều khiển Master sẽ có 36 bytes, cộng với 1 byte dữ liệu bắt đầu, tổng cộng Frame truyền có 37 bytes. Byte bắt đầu có thể là M hoặc A tương ứng với chế độ điều khiển là Manual hay Auto. Byte hướng là P hoặc N quy định chiều chuyển động của động cơ. Byte xung động cơ gồm 5 kí tự, đây là số xung để điều khiển động cơ.

Trong bộ điều khiển mà nhóm nghiên cứu đề xuất, có tất cả 7 vi điều khiển, tất cả các vi điều khiển này giao tiếp với nhau thông qua chuẩn giao tiếp CAN. Cấu trúc của hệ thống gồm 1 mạch Master (tương ứng với Node 0) và 6 mạch Slave (được đánh dấu theo thứ tự từ Node 1 đến Node 6) . Mạch Master là một mạch trung gian giữa máy tính và các mạch Slave (các mạch điều khiển động cơ). Khi frame truyền từ máy tính xuống Master, nó sẽ xử lí frame truyền đó thành dữ liệu để truyền lại cho các mạch Slave. Dữ liệu truyền cho mạch Slave được sử dụng để điều khiển tay máy. Việc truyền nhận dữ liệu giữa các node phải đảm bảo độ chính xác, vì quá trình này ảnh hưởng trực tiếp đến vấn đề điều khiển tay máy.

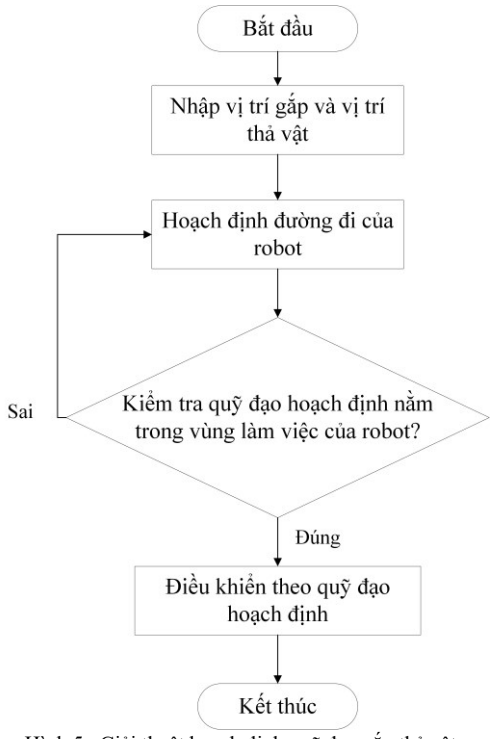


Hình 4. Giải thuật điều khiển bám quỹ đạo

4.3 Các giải thuật điều khiển robot

Trong mục này sẽ trình bày 2 giải thuật được áp dụng cho bộ điều khiển của robot. Giải thuật 1 là giải thuật điều khiển robot bám theo quỹ đạo cho trước. Giải thuật 2 là giải thuật hoạch định quỹ đạo robot cho việc gắp và thả vật.

Đối với giải thuật điều khiển robot bám theo quỹ đạo cho trước thì chúng ta cần chia quỹ đạo đó thành nhiều điểm trung gian. Sau đó chúng ta sẽ giải bài toán động học ngược cho robot để tìm ra giá trị tương ứng tại các khớp. Cuối cùng bộ điều khiển sẽ điều khiển các khớp đến các giá trị tương ứng. Hình 4 thể hiện lưu đồ giải thuật điều khiển robot bám theo một quỹ đạo cho trước. Trong nghiên cứu này, chúng tôi tập trung vào 2 quỹ đạo là quỹ đạo thẳng và quỹ đạo tròn. Đầu vào của bài toán bám quỹ đạo thẳng là tọa độ điểm đầu và điểm cuối của đường thẳng cần bám. Từ quỹ đạo thẳng, người sử dụng có thể phát triển điều khiển robot bám theo các quỹ đạo hình tam giác, hình vuông, hoặc một hình đa giác bất kỳ. Đối với bài toán bám quỹ đạo đường tròn thì đầu vào là bán kính và tâm của đường tròn cần bám. Đồng thời chúng ta cần xác định điểm bắt đầu và kết thúc của quỹ đạo tròn.



Hình 5. Giải thuật hoạch định quỹ đạo gắp thả vật

Giải thuật thứ 2 là giải thuật gắp và thả vật, một giải thuật được ứng dụng phổ biến trong điều khiển robot. Trong bài báo này, chúng tôi trình bày ứng dụng gắp và thả chai dầu nhớt MOTUL. Nội dung của giải thuật này là trình bày phương pháp hoạch định quỹ đạo cho robot trong quá trình gắp. Lưu đồ thực hiện của giải thuật được thể hiện trong hình 5. Trong giải thuật này, đầu vào là vị trí gắp và vị trí thả vật. Còn quỹ đạo trung gian đi từ điểm đầu đến điểm kết thúc là do người sử dụng hoạch định. Thông thường nó là những quỹ đạo đơn giản cho việc lập trình như đường thẳng hoặc đường tròn. Một vấn đề quan trọng nữa là quỹ đạo mà người sử dụng hoạch định phải nằm trong vùng làm việc của robot.

5 KẾT QUẢ THỰC NGHIỆM

Trong mục này, nhóm nghiên cứu sẽ trình bày các kết quả thực nghiệm mà nhóm nghiên cứu đã thực hiện đối với bộ điều khiển mới của robot. Đầu tiên nhóm nghiên cứu sẽ kiểm tra độ chính xác lặp lại của robot. Kế đến nhóm sẽ kiểm tra giải thuật bám quỹ đạo thẳng và quỹ đạo tròn cho robot. Cuối cùng nhóm tiến hành thực nghiệm hoạch định quỹ đạo cho việc gắp và thả chai dầu nhớt MOTUL.

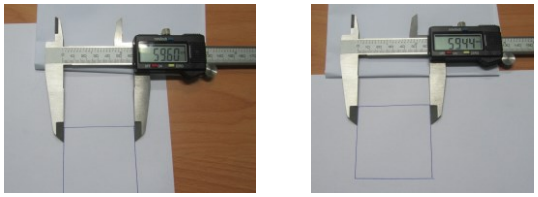
5.1 Độ chính xác lặp lại của robot

Phương pháp kiểm tra độ chính xác lặp lại của robot được thực hiện bằng cách di chuyển đầu công tác giữa hai điểm A và điểm B đã biết trước tọa độ. Đầu tiên nhóm nghiên cứu lấy dấu đầu công tác tại vị trí điểm A, sau đó di chuyển đầu công tác qua điểm B, rồi di chuyển đầu công tác quay lại A và lấy dấu một lần nữa. Độ chênh lệch giữa hai lần lấy dấu là sai số lặp lại. Khoảng cách giữa hai điểm A và B trong thí nghiệm này là 80 mm. Thực hiện 5 lần ta được kết quả trung bình là 0,44mm. Độ chính xác lặp lại của robot theo nhà sản xuất là 0,03mm. Điều này có thể giải thích do phần cơ khí của robot không còn hoạt động tốt như lúc còn mới. Tuy nhiên sai số này là hoàn toàn chấp nhận được cho mục đích học thuật và nghiên cứu.

5.2 Kết quả bám quỹ đạo thẳng và tròn

Trong thí nghiệm này, nhóm nghiên cứu gắn một đầu bút vào đầu công tác của robot và thực hiện việc điều khiển quỹ đạo robot lên một bảng thẳng đứng. Hình 6 thể hiện bố trí điều khiển robot bám theo quỹ đạo. Kết quả đạt được được thể hiện trong hình 7 và hình 8. Quỹ đạo hình vuông mà nhóm tiến hành vẽ có kích thước 60mm x 60mm và sau 5 lần vẽ thì nhận được quỹ đạo thực có sai số trung bình là 0,72mm. Quỹ đạo hình tròn mong muốn có bán kính 30mm và sau 3 lần thực hiện có sai số trung bình là 2,35mm.

Hình 6. Bố trí điều khiển robot bám theo quỹ đạo



a) Kích thước theo phương dọc b) Kích thước theo phương ngang

Hình 7. Quỹ đạo hình vuông

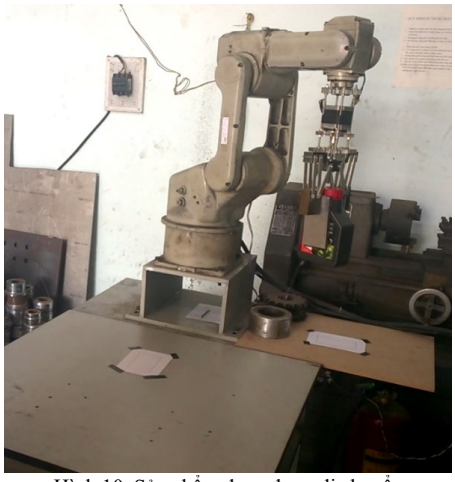
a) Kích thước theo phương dọc b) Kích thước theo phương ngang

Hình 8. Quỹ đạo hình tròn

5.3 Kết quả giải thuật gắp vật

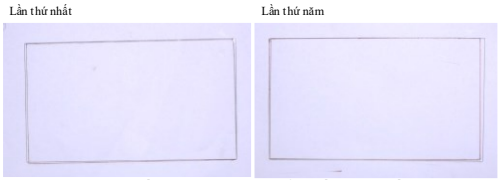
Hình 9. Robot gắp sản phẩm

Trong thí nghiệm này nhóm nghiên cứu thực hiện thực nghiệm gắp vật. Đối tượng được gắp là chai dầu nhớt MOTUL. Ban đầu robot ở vị trí Home, điểm end-effector của tay gắp có tọa độ (740; 0; 620). Robot được điều khiển di chuyển đến điểm gắp sản phẩm, lúc này điểm end – effector của tay gắp có tọa độ (410; 0; 15) và tọa độ vị trí tâm của đáy sản phẩm là (410; 0; -166). Tại vị trí này robot thực hiện thao tác gắp sản phẩm. Sau đó robot di chuyển và đặt sản phẩm tại vị trí (0; 410; -166), đồng thời lúc này điểm end – effector của tay gắp của robot ở vị trí có tọa độ (0; 410; 15). Hình 9 thể hiện lúc robot bắt đầu gắp sản phẩm. Hình 10 thể hiện lúc sản phẩm đang được di chuyển. Hình 11 thể hiện lúc robot đặt sản phẩm vào vị trí mong muốn. Hình 12 cho thấy kết quả khi đặt sản phẩm lần 1 và lần 5. Đường màu xanh là vị trí mong muốn, đường màu đỏ là do robot thực hiện. Sau 5 lần đặt thì sai lệch trung bình theo phương x là 0,964mm và sai lệch trung bình theo phương y là 1,86mm.



Hình 10. Sản phẩm đang được di chuyển

Hình 11. Sản phẩm được đặt vào vị trí



Hình 12. Kết quả đặt sản phẩm lần 1 và lần 5

6 KẾT LUẬN

Bài báo trình bày một phương pháp thiết kế và phục hồi bộ điều khiển cho một robot công nghiệp đã qua sử dụng. Phần mạch công suất, mạch điều khiển và giao diện điều khiển đều được xây dựng mới. Bộ điều khiển robot mới này là sự kết hợp giữa máy tính và các vi điều khiển thông qua chuẩn giao tiếp RS232 và CAN. Từ đó các giải thuật điều khiển được xây dựng gồm 2 giải thuật cơ bản là điều khiển robot bám theo quỹ đạo đã biết và hoạch định quỹ đạo cho robor gắp vật. Các giải thuật này có thể được dùng để xây dựng các bài thí nghiệm cho môn Kỹ thuật robot (ME3016) gồm 15 tiết và chia thành 3 buổi, mỗi buổi 5 tiết.

Kết quả nghiên cứu của bài báo hoàn toàn có thể ứng dụng cho những hướng nghiên cứu sâu hơn về robot, đặc biệt là việc tích hợp các tín hiệu cảm biến để điều khiển robot. Ngoài ra kết quả bài báo cũng có thể được ứng dụng cho việc chế tạo mới robot ở nước ta.

DANH MỤC TỪ VIẾT TẮT

VĐK: Vi điều khiển

CAN: Controller Area Network

Động cơ AC: Động cơ điện xoay chiều

SGMAH, SGMPH: Ký hiệu các dòng động cơ AC của hãng Yaskawa

SGDA, SGDJ, SGDP: Ký hiệu các dòng bộ điều khiển động cơ AC của hãng Yaskawa

TÀI LIỆU THAM KHẢO

[1]. P. Kazanzides, H. Wasti, and W. A. Wolovich, "A multiprocessor system for realtime robotic control: desgn and application," in *Proc. IEEE Int. Conf. Robotics Automation,* Boston, MA, USA, 1987, pp. 1903–1908.

[2]. A. A. Goldenberg, and L. Chan, "An Approach to Real-time Control of Robots in Task Space. Application to Control of PUMA 560 Without VAL-II," *IEEE Transactions on Industrial Electronics*, vol. 35, no. 2, pp. 231-238, May 1988.

[3] S. E. Shafiei, *Advanced Strategies for Robot Manipulators*. Sciyo, Croatia, 2010, pp. 281–396.

[4]. H. Bruyninckx, "Open Robot Control Software: the OROCOS project," in *Proc. IEEE Int. Conf. on Robotics and Automation,* 2001, pp. 2523–2528.

[5]. C. Cote, Y. Brosseau, D. Letourneau, C. Raievsky, and F. Michaud, "Robotic Software Integration Using MARIE," *International Journal of Advanced Robotic System*, pp. 55-60, 2006.

[6] J. Gamez, J. Gomez, L. Nieto, A. G. Sanchez, "Design and Validation of an open Architecture for an Industrial Robot Control," in *IEEE International Symposium on Industrial Electronics,* pp. 2004–2009, 2007.

[7]. K. Nilsson, and R. Johansson, "Integrated Architecture for Industrial Robot Programming and Control," *J. Robotics and Autonomous Systems*, pp. 205-226, 1999.

Phùng Trí Công nhận bằng kỹ sư Cơ điện tử của trường Đại học Bách Khoa, ĐHQG-HCM năm 2004, nhận bằng thạc sỹ năm 2007 và bằng tiến sĩ năm 2011 tại khoa Cơ khí của trường Đại học Sungkyunkwan, Hàn Quốc. Từ năm 2011 đến nay, anh là giảng viên tại Bộ môn Cơ điện tử, khoa Cơ khí, trường Đại học Bách Khoa, ĐHQG-HCM. Hướng nghiên cứu chính là: robot công nghiệp và các ứng dụng, giải thuật điều khiển mobile robot, ứng dụng vi điều khiển và các hệ thống điều khiển tự động.

Nguyễn Tấn Tiến sinh năm 1968, nhận bằng kỹ sư Cơ khí Chế tạo máy của trường Đại học Bách Khoa ĐHQG-HCM năm 1990, nhận bằng Thạc sỹ Cơ điện tử năm 1998 và Tiến sỹ Cơ điện tử năm 2001 của Trường ĐHQG Pukyong, Hàn quốc. Từ năm 1990 anh là giảng viên Bộ môn Thiết kế máy và từ năm 2005 là giảng viên Bộ môn Cơ điện tử, Trường Đại học Bách Khoa, ĐHQG-HCM. Lĩnh vực nghiên cứu hiện nay: lý thuyết điều khiển, humanoid robot, hệ thống cơ điện tử và ứng dụng trong lĩnh vực tự động hóa công nghiệp.

Nguyễn Tấn Đạt đã tốt nghiệp Trường Đại học Bách Khoa, ĐHQG-HCM ngành Kỹ thuật Cơ Điện Tử năm 2016. Hiện nay đang công tác tại Công ty TNHH Bosch Việt Nam, Đường số 8, Khu công nghiệp Long Thành, xã Tam An, huyện Long Thành, Đồng Nai, Việt Nam.

Nguyễn Ngọc Sơn đã tốt nghiệp Trường Đại Học Bách Khoa, ĐHQG-HCM ngành Kỹ thuật Cơ Điện Tử năm 2016. Hiện nay đang công tác tại Công ty Cổ phần Tập đoàn Hoa Sen, Số 9, Đại lộ Thống Nhất, Khu công nghiệp Sóng Thần II, phường Dĩ An, thị xã Dĩ An, Bình Dương, Việt Nam.

A study on designing controller and building control algorithms for 6dof robot applied in education

Tri Cong Phung*, Tan Tien Nguyen, Tan Dat Nguyen, and Ngoc Son Nguyen

Ho Chi Minh City University of Technology, VNU-HCM
Corresponding author: ptcong@hcmut.edu.vn

Received: 03-4-2018, Accepted: 25-8-2018, Published: 30-11-2018

***Abstract*—This paper proposes a method of designing controller and building control algorithms for an industrial robot to establish laboratory lectures for education. The robot is used in this paper is SV3X robot, a product of MOTOMAN. In this research, we present how to design and manufacture a controller for an old robot. After that, we propose labortory lectures based on this controller. A new graphic user interface (GUI) is also built based on C# language. Next, we propose algorithms that can be done. Two algorithms suggested were controlling robot to follow a desired trajectory and controlling robot to grasp a know object from started point to goal point. Finally, these algorithms are verified by experiments. Repeatability of robot is also checked.**

***Index term* — industrial robot, controller, control algorithm, trajectory tracking, grasping object**