

# TÌM HIỂU PHƯƠNG THỨC GIAO TIẾP GIỮA CÁNH TAY ROBOT MITSUBISHI RV-2AJ VÀ MÁY TÍNH THÔNG QUA THIẾT BỊ HAPTIC

## *Study on the mode of communication between the robot arm Mitsubishi RV-2AJ and computer through the use of a haptic device*

Võ Minh Trí<sup>1</sup>, Lê Công Khanh<sup>1</sup>, Khuru Hữu Nghĩa<sup>1</sup>, Lê Hoàng Sơn<sup>1</sup>

<sup>1</sup>Trường Đại học Cần Thơ

e-Mail: [vmtri@ctu.edu.vn](mailto:vmtri@ctu.edu.vn), [lecongkhanh382@gmail.com](mailto:lecongkhanh382@gmail.com),  
[nghia\\_cantho01@yahoo.com](mailto:nghia_cantho01@yahoo.com), [hoangsonle27@gmail.com](mailto:hoangsonle27@gmail.com)

### Tóm tắt

Ngày nay, robot được nhắc đến như một công cụ hỗ trợ hết sức đắc lực cho con người trong những hoạt động sản xuất kinh doanh, y tế hay giáo dục. Bằng cách tìm hiểu phương thức giao tiếp giữa cánh tay robot RV-2AJ và máy tính thông qua việc sử dụng một thiết bị haptic – Bamboo<sup>TM</sup>Pad, bài báo này trình bày một ứng dụng mới trong việc điều khiển robot để vẽ ký họa. Để thực hiện được việc lập trình cho robot vẽ, cần trải qua bốn giai đoạn như: i) Thực hiện mô phỏng robot RV-2AJ trên phần mềm Matlab; ii) Giải mã tín hiệu thông tin của robot RV-2AJ thông qua việc sử dụng phần mềm Serial Port Minotor để thu thập tập lệnh qua cổng RS232; iii) Thiết kế phần mềm điều khiển robot thông qua giao tiếp RS232 và phím chạm Bamboo<sup>TM</sup>Pad; iv) Kiểm chứng hoạt động của robot và phần mềm thiết kế. Kết quả cho thấy, robot có thể thực hiện việc vẽ ký họa trên một mặt phẳng bất kỳ trong không gian hoạt động của robot.

**Từ khóa:** Cánh tay robot RV-2AJ vẽ ký họa, Thiết bị haptic, Phần mềm Matlab

### Abstract:

Nowadays, robots are known as a supported tool very effectively for people in the industrial activities, business activities, medical or educational fields. By understanding the mode of communication between the robot arm RV-2AJ and computers through the use of a haptic device - Bamboo<sup>TM</sup>Pad, this paper presents a new robot application towards the application of robot to do the sketches. To implement the mode to do the sketches, four specific phases were carried out: i) Simulating robot RV-2AJ on Matlab software; ii) Decoding information signals of RV-2AJ robot through the use of Serial Port Monitor software to collect instructions through RS232 port; iii) Designing the robot's control software by using RS232 port and Bamboo<sup>TM</sup>Pad; iv) Checking the operation of the robot and designed software. As a result, the robot can be able to stimulate the sketch on a certain plane within its reachable space.

**Keywords:** Doing the sketch by Robot arm RV-2AJ, Haptic devices, Matlab software.

### Ký hiệu

Ký hiệu	Đơn vị	Ý nghĩa
$a_i$	m	Tham số động học Denavit – Hartenberg
$\alpha_i$	Độ	Tham số động học Denavit – Hartenberg
$d_i$	m	Tham số động học Denavit – Hartenberg
$\theta_i$	Độ	Tham số động học Denavit – Hartenberg
$D_q$		Vector sai số
$\vec{z}$		Vector 5 phần tử tương ứng với 5 khớp của robot
$J^T$		Ma trận jacobian được chuyển vị
$M_{rf}$		Ma trận biến đổi thuần nhất từ hệ tọa độ công tác về hệ tọa độ gốc
$i,j,n$		Vector đơn vị của trục x,y,z đặt trên mặt phẳng công tác
O		Vector tọa độ gốc O trên mặt phẳng công tác
pkt		Vector tọa độ của bút trên touchpad
P		Vector trung gian
$\rho$		Vector tọa độ cần đạt được trong hệ tọa độ gốc Robot

### Chữ viết tắt

PMAC	Programmable Multi-Axies Controller
FMS	Flexible Manufacturing System

## 1. Phần mở đầu

Một cách tự nhiên, chúng ta thường nghĩ ngay đến một cỗ máy, hình dạng tương tự con người và có những hoạt động tương tự con người khi có ai đó đề cập đến chủ đề robot. Nhưng trên thực tế, từ “robot” bao hàm rộng hơn rất nhiều. Robot là tất cả những máy điện hay máy cơ có khả năng thực hiện một chuỗi hành động phức tạp một cách tự động hay bán tự động (Oxford Dictionaries, 2014).

Thuật ngữ “Robot” được nhắc đến lần đầu tiên trong vở kịch “Rossum’s Univesal Robot” của Karel Capek năm 1922[1]. Năm 1961 robot công nghiệp đầu tiên được ứng dụng trong nhà máy lắp ráp ô tô của General Motor - USA [1]. Từ năm 1970 đến nay đã có

hơn 20 báo cáo chính thức về sứ mệnh chinh phục không gian của các robot được gửi đi để thám hiểm Hệ mặt trời [2].

Ngày nay công nghệ robot hầu như được áp dụng lên mọi lĩnh vực trong xã hội. Trong tương lai, robot có thể thay thế con người thực hiện các công việc phức tạp như điều khiển ô tô tham gia giao thông[3]. Các robot hiện đại có cấu trúc đặc biệt có thể tự tách ra hoặc kết hợp lại để cùng phối hợp thực hiện nhiều tính năng khác nhau theo yêu cầu công nghệ cụ thể[4].

Cùng với sự tiến bộ của kỹ thuật robot trên thế giới, công nghệ robot tại Việt Nam cũng có những bước đi đáng ghi nhận. Tính từ năm 2006, đã có nhiều bài báo trong nước và nước ngoài ghi nhận sự tiến bộ của kỹ thuật robot tại Việt Nam với nhiều đại diện như: Robot chiến trường của Học viện kỹ thuật quân sự, Robot hút bụi thông minh, Robot nhảy múa TOSY, Robot đánh bóng bàn TOPIO, đại diện cho Robocon Việt Nam đã đạt 05 huy chương vàng các giải Robocon quốc tế [5]. Đến nay, nước ta đã nhập khẩu và triển khai nhiều robot phẫu thuật phục vụ điều trị cho các bệnh đòi hỏi cao về sự chính xác - như các ca phẫu thuật cột sống- tại các tuyến trung ương [6]. Robot hàn, robot lắp ráp, robot phân loại và nhiều chủng loại robot công nghiệp khác cũng được đầu tư nghiên cứu nghiêm túc [7].

Cùng với sự phát triển của công nghệ sản xuất phần cứng và thuật toán điều khiển số, robot hiện đại không chỉ là một khối thiết bị cứng nhắc mà còn được trang bị các cảm biến như là các giác quan. “Haptic control” là thuật ngữ chỉ một mô hình điều khiển mà trong đó người điều khiển và thiết bị có tương tác trực tiếp hoặc gián tiếp với nhau. Mô hình này tái tạo lại toàn bộ hay một phần cảm giác của con người khi tham gia tương tác với thiết bị[8]. Gần đây, Wacom cũng đã cho ra đời một thiết bị haptic thương mại có thể mã hóa được tọa độ và lực tác động của bút vẽ trên một mặt cảm ứng Bamboo™Pad như hình 1.



H. 1 Bamboo™Pad của hãng Wacom

Nhằm khai thác các thiết bị hiện có và cố gắng bắt kịp sự phát triển của khoa học công nghệ thế giới, nhóm tác giả đề xuất một giải pháp mới điều khiển cánh tay robot RV-2AJ trong dây chuyền sản xuất linh hoạt FMS của hãng FESTO thông qua việc sử dụng thiết bị Haptics - Bamboo™Pad để điều khiển robot vẽ ký họa. Kết quả cho thấy, robot có thể vẽ ký họa lên một mặt phẳng bất kỳ đặt trong không gian làm việc của robot.

## 2. Nội dung chính

### 2.1 Sơ lược về robot RV-2AJ

Robot RV-2AJ là cánh tay robot 5 bậc tự do của hãng Mitsubishi – Nhật Bản. Các thành phần chính bao gồm:

- Một cánh tay máy 5 bậc tự do – Main Body.
- Bộ điều khiển CR1-571.
- Bộ điều khiển và lập trình vị trí bằng tay - Teaching box R28TB.

Tổng quan robot được mô tả trong hình 2.



H. 2 Cánh tay robot Mitsubishi RV-2AJ

Các thông số cơ bản của Robot RV-2AJ:

- Số bậc tự do: 5 bậc.
- Kiểu cấu trúc: Đứng, toàn khớp quay.
- Động cơ dẫn động: AC servo.
- Độ chính xác:  $\pm 0.02\text{mm}$ .
- Khả năng mang tải: Tĩnh 2kg, động 1,5kg.
- Số ngõ In/Out số: 16 In/ 16 Out.
- Cổng giao tiếp: RS232/RS422.
- Điện áp sử dụng: 220VAC.
- Ngôn ngữ lập trình: MELFA-BASIC IV hoặc MOVEMASTER COMMANDS.

#### 2.1.1 Bộ điều khiển (Controller)

Là thành phần quan trọng quyết định khả năng linh hoạt và độ chính xác của Robot. Bộ điều khiển của robot MJ-2AV được trang bị CPU kiến trúc RISC/DSP 64bit[9] cùng các ngõ In/Out số có thể lập trình được. Cổng truyền thông cơ bản RS232/RS422. Bộ điều khiển có thể nâng cấp Card truyền thông qua mạng LAN.

#### 2.1.2 Ngôn ngữ lập trình

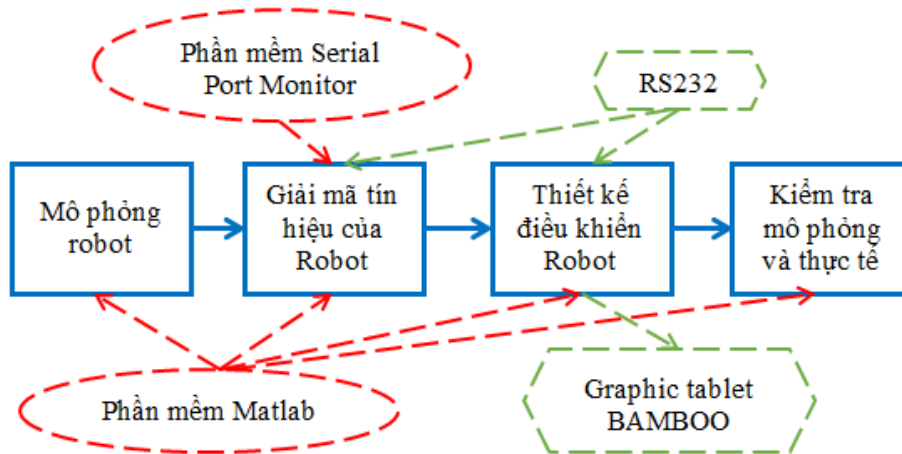
Ngôn ngữ lập trình cho robot là MELFA-BASIC IV hoặc MOVEMASTER COMMANDS[10]. Phần mềm giao diện đi kèm là COSIROP dùng để giao tiếp giữa robot với máy tính và lập trình điều khiển.

Trên dây chuyền FMS, Các tọa độ vị trí và chuyển động của robot RV-2AJ được lập trình trước bằng phần mềm COSIROP và nạp vào bộ điều khiển. Để mở rộng phương thức điều khiển, nhóm tác giả đề xuất một mô hình mới là điều khiển robot trực tiếp từ thiết bị haptic thông qua chương trình điều khiển linh hoạt trên nền ngôn ngữ MATLAB.

#### 2.2 Bố trí thí nghiệm

Như đã giới thiệu, nhóm tác giả sử dụng phần mềm MATLAB lập trình giao diện sự kiện, đọc dữ liệu từ thiết bị haptic, phân tích, tính toán dữ liệu và xuất dữ liệu điều khiển thông qua cổng truyền thông RS232

của bộ điều khiển. Sơ đồ bố trí thí nghiệm được mô tả trong hình 3.

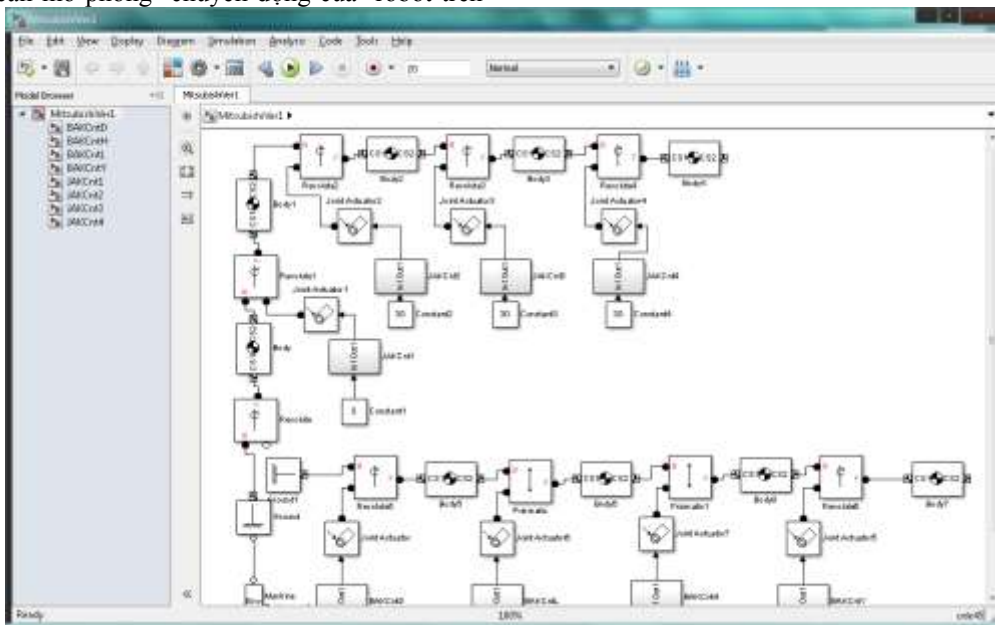


**H. 3** Sơ đồ khối bố trí các thí nghiệm

**2.2.1 Mô phỏng robot RV-2AJ trên Matlab**

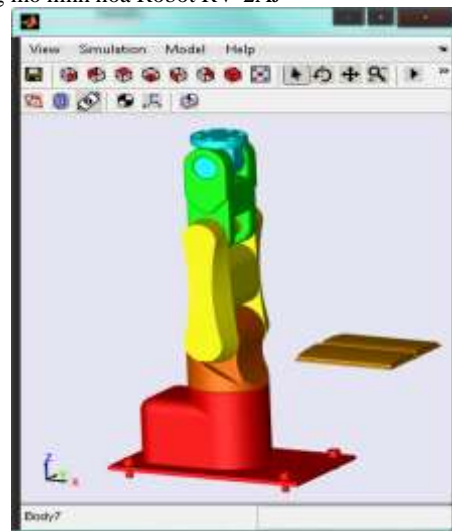
- Để hiểu rõ hơn chuyển động của robot, trước tiên cần mô phỏng chuyển động của robot trên

MATLAB-Simulink. Hình 4 trình bày chương trình mô phỏng robot trên MATLAB-Simulink.



**H. 4** Chương trình Matlab-Simulink trong mô hình hóa Robot RV-2AJ

Sau khi mô hình hóa robot dùng công cụ MATLAB-Simulink, kết quả được mô phỏng được trình bày trong hình 5.



**H. 5** Robot RV-2AJ được mô phỏng trên Matlab

Sau khi mô phỏng thành công robot trên Matlab, một trong những thí nghiệm quan trọng và cũng là mấu chốt ở nghiên cứu này là đọc và giải mã tín hiệu điều khiển của robot RV-2AJ để có thể điều khiển robot thực hiện những yêu cầu công việc đặt ra.

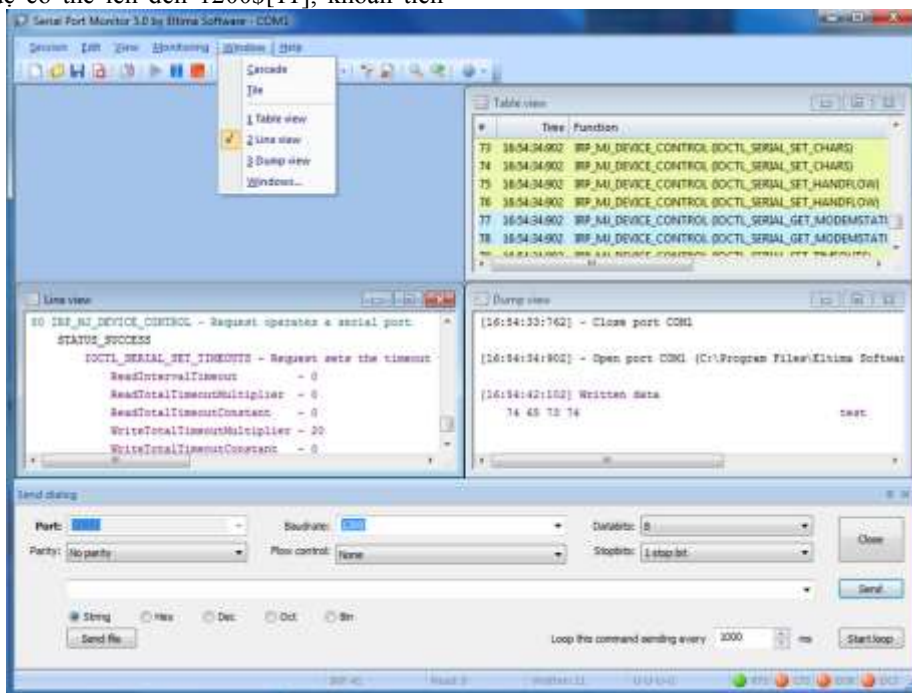
**2.2.2 Giải mã tín hiệu của Robot để điều khiển**

Đa số robot chế tạo sẵn trong thời điểm hiện tại đều được đóng gói cùng với phần mềm biên dịch và bộ điều khiển. Để điều khiển robot từ một phần mềm ứng dụng khác, cần có được thông tin mà phần mềm giao tiếp truyền qua cổng giao tiếp. Điều này đã được thực hiện bởi đơn vị như công ty Nation Instruments[11], các nghiên cứu sinh ở đại học Quindio – Amenia – Colombia[12], và một số đơn vị khác. Nhưng điều đáng lưu tâm là tiền mua bản quyền sở hữu trí tuệ có thể lên đến 1200\$[11], khoản tiền

này vượt xa kinh phí nghiên cứu khoa học mà đơn vị có thể cung cấp. Vì thế chúng tôi đã tìm đến những giải pháp ít tốn kém hơn nhằm mục đích “lọc” lấy thông tin hai chiều giữa máy tính và robot.

Sau khi tìm hiểu nhiều phần mềm lọc tín hiệu. Nhóm tác giả quyết định chọn chương trình Serial Port Monitor của Eltima vì tính ổn định, nhiều chức năng hữu ích như lưu thông tin dưới dạng txt, lọc sự kiện, hiển thị các ngắt thời gian, ngắt sự kiện.

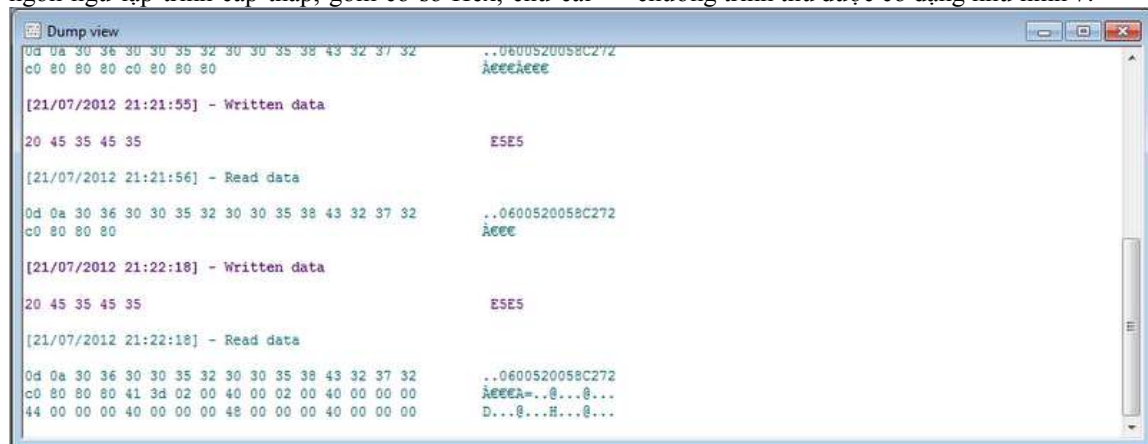
Như vậy có thể nói, thí nghiệm giải mã tín hiệu robot ở đây chính là việc thu thập tập lệnh truyền nhận giữa robot và phần mềm COSIROP qua cổng giao tiếp RS232 thông qua việc sử dụng phần mềm Serial Port Monitor. Giao diện phần mềm Serial Port Monitor được mô tả trong hình 6.



**H. 6** Giao diện điều khiển của phần mềm Serial Port Monitor

Thông tin thu thập được tương tự như hợp ngữ, một ngôn ngữ lập trình cấp thấp, gồm có số Hex, chữ cái

viết tắt, dấu câu và nhiều phần tử khác. Một đoạn chương trình thu được có dạng như hình 7.



**H. 7** Một đoạn chương trình điều khiển robot RV-2AJ được thu thập

Chúng ta biết rằng thông tin này có một quy luật nhất định. Tuy nhiên để hiểu được thông tin này, nhóm tác giả đã phải thiết lập rất nhiều thí nghiệm liên quan. Mỗi thí nghiệm phải tập trung vào một chi tiết nhất

định, thay đổi các thông số nhất định để nhận ra sự khác biệt. Từng thông số thiết lập có thể nhìn thấy, độ dài chương trình, thay đổi câu lệnh cũng như thứ tự của các câu lệnh, tọa độ cũng như thay đổi các thuật



toán được hỗ trợ. Sau khi xác định được quy luật của các câu lệnh và kết hợp với ngôn ngữ lập trình MELFA-BASIC IV nhóm tác giả đã giải mã được tập lệnh của phần mềm COSIROP điều khiển robot và chuyển sang giai đoạn thiết kế chương trình điều khiển.

### 2.2.3 Thiết kế chương trình điều khiển robot

Yêu cầu hiện tại của phần mềm điều khiển gồm có:

- Kết nối, ngắt kết nối.
- Dừng robot.
- Reset lỗi.
- Khởi động robot.
- Đóng/mở tay gấp.
- Đọc vị trí, góc quay.
- Trả robot về vị trí home.
- Tăng/giảm vị trí tay gấp, di chuyển robot.
- Tăng/giảm góc quay các khớp, di chuyển robot.
- Mô phỏng trong không gian ảo ba chiều.

- Đọc thông số các khớp.

Theo yêu cầu đã đề ra, và với sự hỗ trợ của Matlab-guide, mỗi đơn vị nút bấm trên giao diện hình 8 sẽ thực hiện một chức năng chính và kết hợp mới một số chức năng khác:

- Chức năng reset lỗi sẽ thực thi sau chức năng dừng robot.
- Chức năng trả về vị trí home sẽ thực thi sau khi khởi động robot.
- Chức năng tăng/giảm vị trí sẽ đi kèm với vòng lặp đã đề cập ở phần Động học ngược để tiến hành mô phỏng robot.
- Chức năng đọc thông số khớp thực thi trước khi lệnh di chuyển robot được gửi đi để đảm bảo tọa độ tiếp theo vẫn nằm trong không gian làm việc của robot.

Hình 8 trình bày giao diện điều khiển robot ảo:



H. 8 Giao diện điều khiển robot ảo trên Matlab Guide

Giao diện trình bày trong hình 8 cho phép tạo chuyển động cho robot từ vị trí hiện tại đến vị trí mới bằng cách thay đổi góc quay của các khớp hay thay đổi vị trí khâu tác động cuối.

Khi chuyển sang chế độ kết nối với robot, lệnh từ giao diện điều khiển sẽ truyền đến cánh tay robot thông qua cổng RS232

#### a) Giao tiếp qua RS232

Để thực hiện giao tiếp với RS232, nhóm tác giả đã tạo ra một hàm điều khiển vị trí trên Matlab có dạng như sau:

```
function [output_args]=Robot_Mitsubishi
( s,xx,yy,zz,rol,pitch,yar )
```

Hàm này được tạo ra với 5 mục đích chính:

- Kết nối MATLAB với cổng truyền thông RS232.
- Điều khiển khớp cổ tay của robot đến vị trí xác định trong không gian.
- Đóng kết nối với cổng RS232.

```
PressureTest_GraphicTablet.m
wPtr = Screen('OpenWindow', 0, 0, [0 0 30 30]); %Tạo ra một cửa sổ kích thước 30x30 để phục vụ
thu thập dữ liệu
WinTabMex(0, wPtr); %Kích hoạt driver, kết nối vào cửa sổ 'wPtr'.
WinTabMex(2); % Bắt đầu ghi nhận và xử lý số liệu.
pkt=WinTabMex(5); %Truy suất dữ liệu theo phương thức FIFO
Return_Value=pkt; %Ghi dữ liệu thu thập được vào biến trả về
while(~isempty(pkt)) %Vòng lặp loại bỏ các điểm thừa ghi nhận được từ tablet
    pkt = WinTabMex(5);
    if(~isempty(pkt))
```

- Kết hợp với các vòng lặp tạo thành chu trình điều khiển.

- Thử tốc độ truyền thông với robot.

Chúng ta có thể can thiệp vào hàm này khi cần thay đổi tốc độ truyền thông bằng cách thêm các Init-commands cần thiết cho Robot.

#### b) Giao tiếp với Graphic Tablet sử dụng Psychtoolbox:

Psychtoolbox là một hộp công cụ được tạo ra cho Matlab giúp cho việc trình bày các kích thích và thu thập phản ứng hành vi trở nên dễ dàng hơn. Trong nghiên cứu này, nhóm tác giả sử dụng Psychtoolbox để ghi nhận chuyển động của bút và áp lực của đầu bút trên Bamboo™Pad.

Dưới đây là đoạn chương trình được tạo ra trên MATLAB:

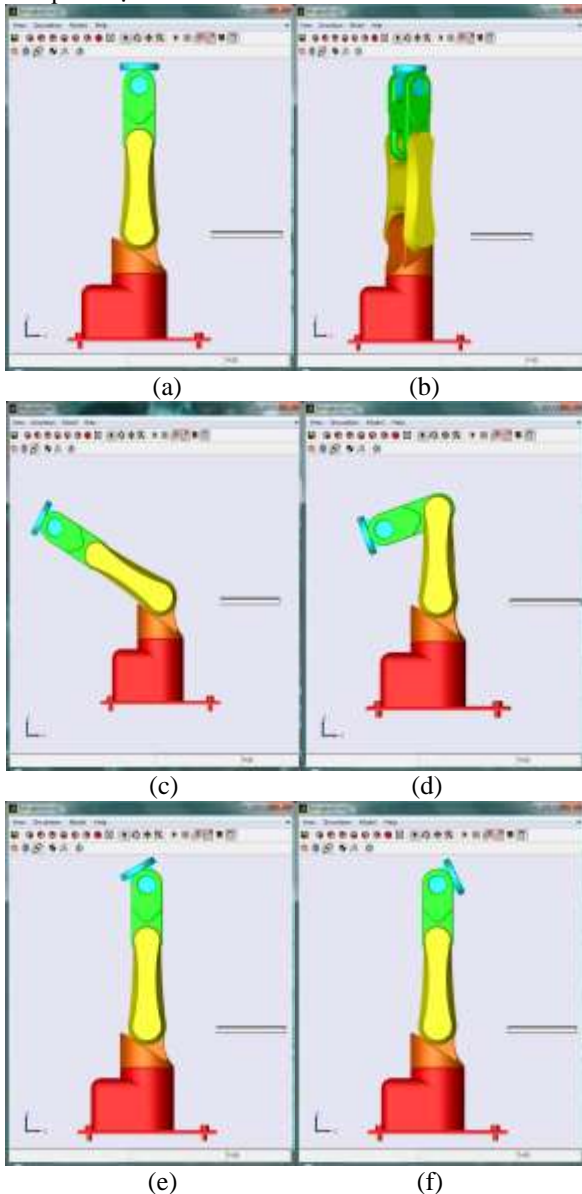
```

Return_Value=pkt;
end
end
if(~isempty(Return_Value)) %Trả về vị trí mới nhất người dùng tác động lên Tablet
output_args=[Return_Value([1 2],1);Return_Value(9,1)];
%disp(Return_Value([1 2],1))
end
WinTabMex(3); % Stop/Pause ghi nhận và xử lý số liệu.
WinTabMex(1); % Shutdown driver.
Screen('CloseAll');
    
```

**2.2.4 Kiểm tra hoạt động của mô phỏng và thực tế**

**a) Kiểm tra không có kết nối**

Đầu tiên, mở phần mềm ứng dụng và rút cáp kết nối, kết quả được mô tả như hình 9.



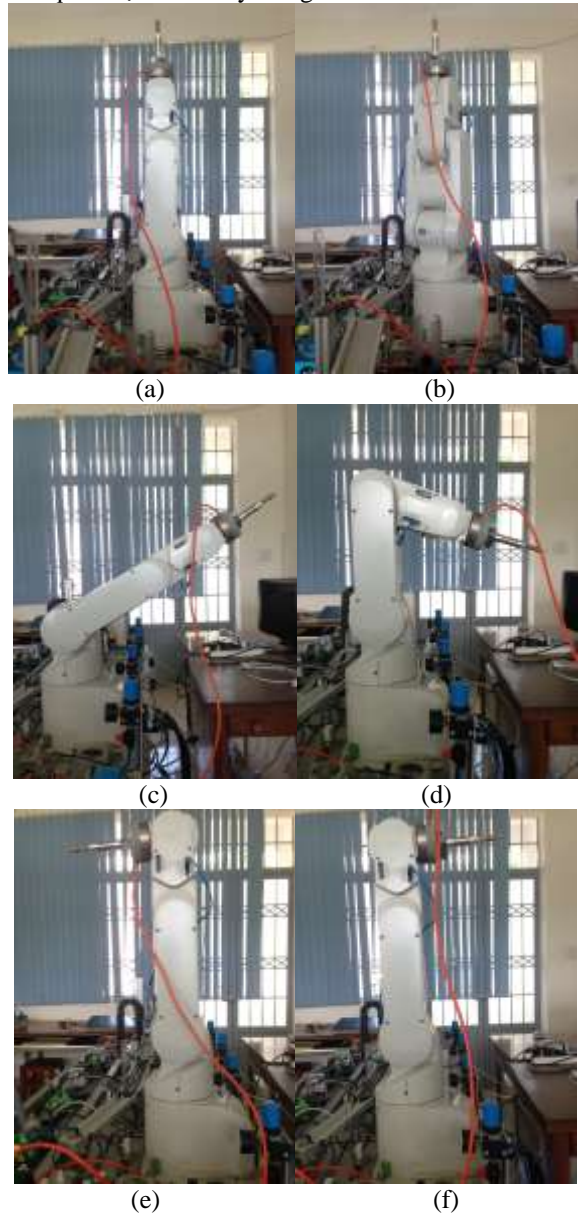
**H. 9** Quan sát các chuyển động của robot trên phần mềm mô phỏng

- (a) - Robot tại vị trí home; (b) - Giới hạn một bên của khớp 1; (c) - Giới hạn trên của khớp thứ 2; (d) - Giới hạn trên của khớp thứ 3; (e) - Giới hạn trên của khớp thứ 4; (f) - Giới hạn dưới của khớp thứ 4

Sau khi tiến hành kiểm tra, nhóm tác giả nhận thấy rằng ứng dụng mô phỏng robot hoạt động đúng với dự kiến.

**b) Kiểm tra có kết nối**

Khi tiến hành kiểm tra có kết nối, nhóm đã bố trí ít nhất hai người để vừa giám sát phần mềm vừa giám sát phần cứng, dừng khẩn cấp robot khi có dấu hiệu bất thường. Trong chế độ đã kết nối, tọa độ của robot được cập nhật lên giao diện theo thời gian định trước. Kết quả được trình bày trong hình 10.



**H. 10** Quan sát các chuyển động ngoài thực tế của robot

- (a) - Robot tại vị trí home; (b) - Giới hạn một bên của khớp 1; (c) - Giới hạn trên của khớp thứ 2; (d) - Giới hạn trên của khớp thứ 3; (e) - Giới hạn trên của khớp thứ 4; (f) - Giới hạn dưới của khớp thứ 4

Sau khi tiến hành kiểm tra có kết nối, nhận thấy robot hoạt động đúng như khi giả lập.

**2.3 Ứng dụng robot RV-2AJ vẽ ký họa**

Robot RV-2AJ được biết đến rộng rãi là một Robot công nghiệp. Phương thức điều khiển thông thường là lập trình cho robot một chuỗi các thao tác tại các tọa độ cố định biết trước. Trong nghiên cứu này, Nhóm tác giả muốn mở rộng ứng dụng của robot này trong điều kiện thực tế. Và như đã nêu, nhóm đã nghiên cứu sâu hơn phương pháp điều khiển robot nhằm phục vụ công tác đào tạo, thúc đẩy tinh thần học tập nghiên cứu của sinh viên tại trường. Nghiên cứu này sẽ làm tiền đề cho các ứng dụng tiếp theo phục vụ nhiều mục tiêu nghiên cứu trong tương lai.

Ứng dụng mô hình Haptic control vào thực tế điều khiển robot sẽ mở ra một hướng ứng dụng mới theo xu thế của thời đại. Ứng dụng robot vẽ ký họa chỉ là bước đầu trong chuỗi các ứng dụng tiếp theo như điều khiển lực, điều khiển tổng trở, điều khiển mềm dẻo. Kết quả của nghiên cứu sẽ được ứng dụng vào việc thiết lập một phương thức mới trong việc điều khiển robot liên tục từ thiết bị thu nhận dữ liệu.

**Bảng 1:** Các thông số Denavit-Hartenbergs

STT khớp	$a_i$ [m]	$\alpha_i$ [°]	$d_i$ [m]	$\theta_i$ [°]
1	0	-90	0.3	0
2	0.25	0	0	-90
3	0.16	0	0	90
4	0	-90	0	-90
5	0	0	0	0

**Phân biệt loại bài toán động học:**

- Bài toán động học thuận: Từ thông số của các khớp (góc/độ dài) tìm ra được vị trí và hướng của tay robot (end-effector)[13].
- Bài toán động học ngược: Biết tọa độ và hướng cần đạt được của tay robot, xác định thông số cần thiết của các khớp[13].

Dễ dàng thấy rằng để thực hiện thao tác vẽ ký họa lên thiết bị haptic và sau đó robot sẽ thực hiện theo chính là một bài toán động học ngược.

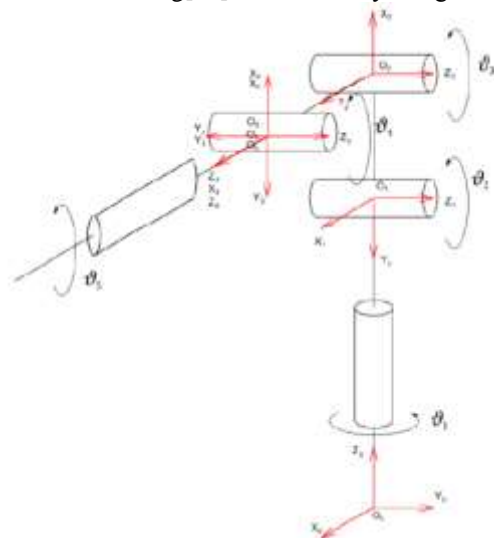
Có nhiều phương pháp có thể sử dụng để giải bài toán này, tuy nhiên để thuận tiện cho việc lập trình và mô phỏng trên máy tính thì phương pháp tính động học ngược sử dụng ma trận chuyển đổi Jacobian là một lựa chọn tốt. Sử dụng phương pháp này, góc quay của robot sẽ cập nhật trong vòng lặp đến khi sai số giữa góc mong muốn và góc hiện tại là chấp nhận được. Nếu đặt vector cột các góc quay của khớp thứ k là  $\theta$  thì lượng thay đổi:

$$Dq = a J^T e \quad (1)$$

Trong đó  $\vec{e}$  là vector 5 phần tử tương ứng với 5 khớp của robot,  $Dq$  biểu hiện cho sai số giữa vị trí hiện tại và vị trí mong muốn robot đạt được,  $a$  được chọn rất gần cực tiểu của  $\vec{e}$  và  $J^T$  là ma trận jacobian được chuyển vị [14].

**2.3.1 Xác định và thiết lập bài toán động học phù hợp**

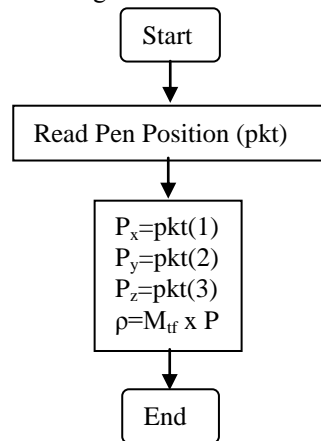
Để tính toán các bài toán động học, cần phải đặt vào các khớp robot một hệ tọa độ mới theo quy tắc Denavit-Hartenberg[13] như trình bày trong hình 11.



**H. 11** Hệ tọa độ đặt cho các khớp

**2.3.2 Tính toán tọa độ tuyệt đối của điểm vẽ trên hệ tọa độ robot**

Thuật toán tính toán tọa độ cần di chuyển của bút vẽ được mô tả trong lưu đồ hình 12.



**H. 12** Lưu đồ thuật toán tính toán tọa độ bút vẽ trong hệ tọa độ của Robot

$M_{tf}$  Là ma trận biến đổi thuần nhất từ hệ tọa độ của mặt phẳng bảng vẽ về hệ tọa độ gốc của robot[13].



$$[M_{tf}] = \begin{bmatrix} i_x & j_x & n_x & O_x \\ i_y & j_y & n_y & O_y \\ i_z & j_z & n_z & O_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$i, j, n$  lần lượt là vector đơn vị trục  $x, y, z$  và  $O$  là vector tọa độ gốc  $O$  trên bảng vẽ.

Kết thúc lưu đồ, chương trình sẽ thu được tọa độ các điểm mà bút vẽ cần di chuyển đến trong không gian công tác của Robot. Kết quả này sẽ được xuất sang bộ điều khiển để điều khiển robot thông qua công RS232[9]. Kết quả thu được trình bày trên hình 13.



H. 13 Robot RV-2AJ thực hiện thao tác ký tên

### 3. Kết luận

Từ kết quả nghiên cứu phương thức giao tiếp giữa robot và máy tính thông qua việc sử dụng thiết bị haptic – Bamboo™Pad, nhóm đã thiết lập một ứng dụng điều khiển robot mới theo hướng nghiên cứu, phát triển phục vụ công tác đào tạo và thực hành, đặc biệt là việc ứng dụng robot để vẽ kỹ họa. Kết quả này chỉ là một bước khiêm tốn trong công cuộc đưa thiết bị kỹ thuật hiện đại vào thực tế, nhưng nhóm tác giả tin tưởng đề tài này đã sẵn sàng công bố để thể thu hẹp khoảng cách ứng dụng tiến bộ khoa học vào thực tiễn. Đặt nền móng cho tinh thần nghiên cứu học hỏi và mở rộng ứng dụng các thiết bị sẵn có trong mỗi sinh viên Việt Nam. Kết quả nghiên cứu có thể tiếp tục kế thừa để phát triển các ứng dụng khác quan trọng hơn, hiệu quả hơn, có lợi ích to lớn hơn nhằm phục vụ cho cộng đồng.

### Tài liệu tham khảo

[1] Thiện Phúc N. “Robot Công Nghiệp” GS. TSKH, Nhà xuất bản Khoa Học Kỹ Thuật, Hà Nội, 2006.  
 [2] Aravind and Schafer, 2010. *Robot mobility systems for planetary surface exploration state-*

*of-the-art and future outlook: a literature survey*. Intech Publ. Germany. 30pp.

- [3] Tom Vanderbilt, 2012. *Let the Robot Drive: The Autonomous Car of the future is Here*.  
 [4] Mark Yim, Wei-Min Shen, Behnam Salemi, Daniela Rus. *Modular Self-Reconfigurable Robot Systems*. Corel Corp. 10pp.  
 [5] Tuổi trẻ online, 2015. *Nhip sống trẻ*. <http://tuoitre.vn/tin/nhip-song-tre/20150823/viet-nam-lai-dang-quang-robocon-chau-a-thai-binh-duong/957171.html> Truy cập ngày 24/8/2015  
 [6] TTXVN, 2014. *Ứng dụng Robot định vị chính xác trong phẫu thuật*. <http://www.vietnamplus.vn/photo-ung-dung-robot-dinh-vi-chinh-xac-trong-phau-thuat/267585.vnp> Truy cập ngày 27/9/2014  
 [7] Tô Tâm, 2004. *Việt Nam chế tạo thành công robot công nghiệp*. <http://vietbao.vn/Khoa-hoc/Viet-Nam-che-cao-thanh-cong-robot-cong-nghiep/45126229/188/> Truy cập ngày 27/9/2014  
 [8] Miriam George. *Haptics: The New Wave Of Tactics For Customer Experience*. International Journal of Information & Futuristic Research, Vol.2, Iss. 8, pp. 2606-2611, 2015  
 [9] Kamil ŽIDEK, Jozef SVETLÍK, Vladislav MAXIM. *Remote control of industrial robot*. Slovakia 2012  
 [10] Mitsubishi Industrial Robot (2007), CR1/CR2/CR3/CR4/CR7/CR8/CR9 Controller, *Detailed explanations of functions and operations*, 2007, Tokyo.  
 [11] National Instruments, 2014. *Robotics Library for Mitsubishi – DigiMetrix GmbH*. <http://sine.ni.com/nips/cds/view/p/lang/en/nid/211070> Truy cập ngày 04/10/2014  
 [12] Buitrago, J.A, 2011. *Remote access lab for Mitsubishi RV-2AJ robot*. <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6086802&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F6078719%2F6086789%2F06086802.pdf%3Farnumber%3D6086802> Truy cập ngày 04/10/2014  
 [13] Trường Đại Học Công Nghiệp TP.HCM “*Giáo Trình Kỹ Thuật Robot*” Khoa Công Nghệ Điện, TP.HCM, tháng 12 2008  
 [14] Samuel R.Buss, 2009. *Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods*. University of California, San Diego La Jolla.





**Võ Minh Trí** nhận bằng Kỹ sư Cơ khí nông nghiệp tại Trường Đại Học Cần Thơ năm 1993, bằng Thạc sỹ Cơ khí nông nghiệp tại Trường Đại Học Quốc Gia Tp. Hồ Chí Minh năm 1998, và nhận bằng Tiến sỹ Cơ điện tử tại Trường Đại Học Katholieke Universiteit Leuven, Vương quốc Bỉ năm

2010.

Tiến sỹ Võ Minh Trí tham gia giảng dạy tại Trường Đại Học Cần Thơ từ năm 1993 đến nay. Hiện anh đang là Giảng Viên Chính thuộc Bộ môn Tự Động Hóa, Khoa Công Nghệ; giữ chức vụ Trưởng bộ môn Tự động hóa, Khoa Công Nghệ, Trường Đại Học Cần Thơ.



**Lê Công Khanh** sinh năm 1992 tại An Giang, tốt nghiệp Đại học Cần Thơ năm 2014, chuyên ngành Cơ điện tử, tham gia tập huấn về Nghiên cứu phân tích dữ liệu định tính tại Đại học An Giang vào tháng 10 năm 2014. Hiện tại, Lê Công Khanh đang theo học bằng Thạc sỹ ngành Tự động hóa tại

Đại học Cần Thơ.



**Lê Hoàng Sơn** sinh năm 1993 tại Vĩnh Long, tốt nghiệp ngành kỹ sư Cơ điện tử của Đại học Cần Thơ vào tháng 8 năm 2015. Hoàn thành chương trình học bổng của tập đoàn Yanmar Nhật Bản tại Nhật Bản từ tháng 6 đến tháng 8 năm 2015. Tham gia khóa tập huấn Phương pháp phân tích dữ liệu định tính tại

Đại học An Giang vào tháng 10 năm 2014.



**Khuru Hữu Nghĩa** sinh năm 1980 tại Bạc Liêu. Nhận bằng Kỹ sư Điện Tử tại trường Đại Học Cần Thơ năm 2003. Đang theo học bằng Thạc sỹ ngành Kỹ Thuật Điều Khiển và Tự Động Hóa tại Đại Học Cần Thơ. Hiện tại anh đang công tác tại Truyền Tải Điện Miền Tây. Lĩnh vực công tác: Thí nghiệm, bảo trì thiết bị nhất thứ và nhị thứ trạm

biến áp truyền tải điện áp đến 500KV. Lĩnh vực quan tâm: Relay bảo vệ hệ thống điện, Robot, Tự động hóa sản xuất, SCADA, CNC.