

LỜI CAM ĐOAN

Chúng tôi xin cam đoan đề tài nghiên cứu khoa học này là công trình nghiên cứu của chúng tôi. Các kết quả nghiên cứu do chủ nhiệm đề tài và những người tham gia thực hiện.

Chúng tôi xin cam đoan rằng các thông tin trích dẫn trong đề tài này đều được chỉ rõ nguồn gốc.

Chúng tôi xin chịu trách nhiệm về nghiên cứu của mình.

Hải Phòng, ngày 17 tháng 01 năm 2013

Chủ nhiệm đề tài

Ngô Quang Vĩ

DANH MỤC CÁC CHỮ VIẾT TẮT

| Chữ viết tắt | Giải thích |
|---------------------|-------------------|
| RBCN | Robot công nghiệp |
| TĐH | Tự động hóa |

MỤC LỤC

| | Trang |
|---|-----------|
| MỞ ĐẦU | 1 |
| CHƯƠNG 1: CÁC KHÁI NIỆM CƠ BẢN VÀ PHÂN LOẠI ROBOT | 3 |
| 1.1. Các khái niệm cơ bản..... | 3 |
| 1.1.1. Robot và Robotics..... | 3 |
| 1.1.2. Robot công nghiệp (RBCN)..... | 5 |
| 1.2. Cấu trúc cơ bản của RBCN..... | 6 |
| 1.2.1. Kết cấu chung..... | 6 |
| 1.2.2. Kết cấu của tay máy | 8 |
| 1.3. Phân loại robot | 11 |
| 1.3.1. Phân loại theo kết cấu | 11 |
| 1.3.2. Phân loại theo điều khiển..... | 11 |
| 1.3.3. Phân loại theo ứng dụng..... | 12 |
| 1.4. Bài toán thuận của động học tay máy | 13 |
| 1.4.1. Mô tả quy tắc Denavit-Hartenberg..... | 15 |
| 1.4.2. Một số ví dụ áp dụng quy tắc Denavit-Hartenberg | 18 |
| 1.4.3. Vùng hoạt động của phân công tác | 20 |
| 1.5. Bài toán ngược của động học tay máy..... | 22 |
| 1.5.1. Cơ cấu 3 khâu phẳng..... | 23 |
| 1.5.2. Cơ cấu cầu..... | 24 |
| CHƯƠNG 2: XÂY DỰNG VÀ ĐIỀU KHIỂN CÁN H TAY ROBOT BA BẬC TỰ DO..... | 26 |
| 2.1. Xây dựng phần cơ khí của robot | 26 |
| 2.1.1. Cấu tạo của cánh tay robot..... | 26 |
| 2.1.2. Cấu tạo của tay kẹp..... | 29 |
| 2.1.3. Truyền động khí nén..... | 29 |

| | | |
|--------|--|----|
| 2.1.4. | <i>Truyền động điện cơ</i> | 30 |
| 2.2. | Bộ điều khiển cho cánh tay robot ba bậc tự do | 31 |
| 2.2.1. | <i>Mở đầu</i> | 31 |
| 2.2.2. | <i>Mạch điều khiển</i> | 32 |
| 2.2.3. | <i>Sử dụng phần mềm LabVIEW để viết giao diện điều khiển</i> | 38 |
| 2.3. | Hoạt động của cánh tay robot | 48 |
| 2.3.1. | <i>Chế độ bằng tay</i> | 49 |
| 2.3.2. | <i>Chế độ tự động</i> | 50 |
| | KẾT LUẬN VÀ KIẾN NGHỊ | 53 |
| | TÀI LIỆU THAM KHẢO | 54 |

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Ngành công nghiệp robot trên thế giới đã đưa được sản phẩm là robot công nghiệp để phục vụ sản xuất, thậm chí phục vụ nhu cầu giải trí cũng như chăm sóc con người. Với ngành công nghiệp của Việt Nam thì robot chưa được xuất hiện nhiều trong các dây chuyền sản xuất. Vì sản phẩm này còn quá đắt đối với thị trường Việt Nam.

Nhằm nội địa hóa sản phẩm, cũng như nghiên cứu chuyên sâu về robot, tôi chọn đề tài “Thiết kế, chế tạo và điều khiển cánh tay robot 3 bậc tự do”. Đề tài này hướng tới có thể thay thế các bộ điều khiển của các công ty nước ngoài và xây dựng thuật điều khiển tối ưu cho các đối tượng sản xuất, mà các đối tượng này thích hợp với điều kiện sản xuất ở nước ta.

Với các phòng thí nghiệm, đây là một mô hình để sinh viên thực nghiệm và nghiên cứu, để hướng tới cho các bạn sinh viên một cái nhìn cụ thể, thực tiễn hơn về robot.

2. Mục đích nghiên cứu của đề tài

Mục đích của đề tài này là nghiên cứu về cấu tạo và các phương pháp điều khiển thích hợp trên cơ sở ứng dụng các kỹ thuật tiên tiến và xây dựng những giải pháp phần cứng cũng như phần mềm để chế tạo bộ điều khiển cánh tay robot ba bậc tự do. Nhằm làm chủ kỹ thuật chế tạo robot, có thể áp dụng vào phòng thí nghiệm của các trường cao đẳng, đại học cũng như ứng dụng trong sản xuất công nghiệp.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài này là phương trình động học của robot để đưa ra thuật điều khiển tối ưu cho robot; phần mềm LabVIEW, CodeVisionAVR để điều khiển cánh tay robot ba bậc tự do và phần cơ khí để chế tạo cánh tay robot.

Nghiên cứu này chỉ giới hạn trong phạm vi nghiên cứu, chế tạo bộ điều khiển cánh tay robot ba bậc tự do với các phần mềm điều khiển nêu trên.

4. Phương pháp nghiên cứu của đề tài

Nghiên cứu cơ sở khoa học và thực tiễn về chế tạo, điều khiển robot.

Phương pháp nghiên cứu thực tiễn:

- Nghiên cứu về phương trình động học ngược của robot;
- Nghiên cứu và ứng dụng các phần mềm LabVIEW, CodeVisionAVR để điều khiển cánh tay robot ba bậc tự do;
- Nghiên cứu và ứng dụng các phân gia công cơ khí để chế tạo cánh tay robot.

5. Ý nghĩa khoa học và thực tiễn của đề tài

Đề tài này nghiên cứu cụ thể về điều khiển vị trí dùng thuật toán điều khiển PID. Sử dụng phần mềm LabVIEW để điều khiển cánh tay. Kết hợp với bộ điều khiển nhỏ gọn và giao tiếp thành công với robot.

Đề tài có thể sử dụng làm mô hình thí nghiệm cũng như sử dụng cho một khâu hay giai đoạn nào đó của sản xuất.

Chương 1:

CÁC KHÁI NIỆM CƠ BẢN VÀ PHÂN LOẠI ROBOT

1.1. Các khái niệm cơ bản

1.1.1. Robot và Robotics

Từ thời cổ xưa, con người đã mong muốn tạo ra những vật giống như mình để bắt chúng phục vụ cho bản thân mình. Ví dụ, trong kho thần thoại Hy Lạp có chuyện người khổng lồ Prometheus đúc ra con người từ đất sét và truyền cho họ sự sống, hoặc chuyện tên nô lệ Talus khổng lồ được làm bằng đồng và được giao nhiệm vụ bảo vệ hoang đảo Crete.

Đến năm 1921, từ "*Robot*" xuất hiện lần đầu trong vở kịch "*Rossum's Universal Robots*" của nhà viết kịch viễn tưởng người Sec, Karel Capek. Trong vở kịch này, ông dùng từ "*Robot*", biến thể của từ gốc Slavơ "*Rabota*", để gọi một thiết bị - lao công do con người (nhân vật Rossum) tạo ra.

Vào những năm 40 nhà văn viễn tưởng Nga, Issac Asimov, mô tả robot là một chiếc máy tự động, mang diện mạo của con người, được điều khiển bằng một hệ thần kinh khả trình *Positron*, do chính con người lập trình. Asimov cũng đặt tên cho ngành khoa học nghiên cứu về robot là *Robotics*, trong đó có 3 nguyên tắc cơ bản:

1. Robot không được xúc phạm con người và không gây tổn hại cho con người.
2. Hoạt động của robot phải tuân theo các quy tắc do con người đặt ra. Các quy tắc này không được vi phạm nguyên tắc thứ nhất.
3. Một robot cần phải bảo vệ sự sống của mình, nhưng không được vi phạm hai nguyên tắc trước.

Các nguyên tắc trên sau này trở thành nền tảng cho việc thiết kế robot.

Từ sự hư cấu của khoa học viễn tưởng, robot dần dần được giới kỹ thuật hình dung như những chiếc máy đặc biệt, được con người phỏng tác theo cấu tạo và hoạt động của chính mình, dùng để thay thế mình trong một số công việc xác định.

Để hoàn thành nhiệm vụ đó, robot cần có khả năng *cảm nhận* các thông số trạng thái của môi trường và tiến hành các *hoạt động* tương tự con người.

Khả năng hoạt động của robot được đảm bảo bởi *hệ thống cơ khí*, gồm cơ cấu vận động để đi lại và cơ cấu hành động để có thể làm việc. Việc thiết kế và chế tạo hệ thống này thuộc lĩnh vực khoa học về cơ cấu truyền động, chấp hành và vật liệu cơ khí.

Chức năng cảm nhận, gồm thu nhận tín hiệu về trạng thái môi trường và trạng thái của bản thân hệ thống, do các cảm biến (*sensor*) và các thiết bị liên quan thực hiện. Hệ thống này được gọi là hệ thống thu nhận và xử lý tín hiệu, hay đơn giản là *hệ thống cảm biến*.

Muốn phối hợp hoạt động của hai hệ thống trên, đảm bảo cho robot có thể tự điều chỉnh "Hành vi" của mình và hoạt động theo đúng chức năng quy định trong điều kiện môi trường thay đổi, trong robot phải có *hệ thống điều khiển*. Xây dựng các hệ thống điều khiển thuộc phạm vi điện tử, kỹ thuật điều khiển và công nghệ thông tin.

Robotics được hiểu là một ngành khoa học có nhiệm vụ nghiên cứu, thiết kế, chế tạo các robot và ứng dụng chúng trong các lĩnh vực hoạt động khác nhau của xã hội loài người, như nghiên cứu khoa học, kỹ thuật, kinh tế, quốc phòng và dân sinh [3,tr.8].

Từ hiểu biết sơ bộ về chức năng và kết cấu của robot, chúng ta hiểu, *Robotics* là một khoa học liên ngành, gồm cơ khí, điện tử, kỹ thuật điều khiển và công nghệ thông tin. Theo thuật ngữ hiện nay, robot là sản phẩm của ngành cơ - điện tử (*Mechatronics*).

Khía cạnh nhân văn và khía cạnh khoa học - kỹ thuật của việc sản sinh ra robot thống nhất ở một điểm: thực hiện hoài bão của con người, là tạo ra thiết bị thay thế mình trong những hoạt động không thích hợp với mình, như:

- Các công việc lặp đi lặp lại, nhàm chán, nặng nhọc: vận chuyển nguyên vật liệu, lắp ráp, lau cọ nhà,...
- Trong môi trường khắc nghiệt hoặc nguy hiểm: như ngoài không vũ trụ, trên chiến trường, dưới nước sâu, trong lòng đất, nơi có phóng xạ, nhiệt độ cao,...
- Những việc đòi hỏi độ chính xác cao, như thông tắc mạch máu hoặc các ống dẫn trong cơ thể, lắp ráp các cấu tử trong vi mạch,...

Lĩnh vực ứng dụng của robot rất rộng và ngày càng được mở rộng thêm. Ngày nay, khái niệm về robot đã mở rộng hơn khái niệm nguyên thủy rất nhiều. Sự phỏng

tác về kết cấu, chức năng, dáng vẻ của con người là cần thiết nhưng không còn ngự trị trong kỹ thuật robot nữa. Kết cấu của nhiều "con" robot khác xa với kết cấu các bộ phận của cơ thể người và chúng cũng có thể thực hiện được những việc vượt xa khả năng của con người.

1.1.2. Robot công nghiệp (RBCN)

Mặc dù, như định nghĩa chung về robot đã nêu, không có gì giới hạn phạm vi ứng dụng của robot, nhưng có một thực tế là hầu hết robot hiện đang có đều được dùng trong công nghiệp. Chúng có đặc điểm riêng về kết cấu, chức năng, đã được thống nhất hoá, thương mại hoá rộng rãi. Lớp robot này được gọi là *Robot công nghiệp (Industrial Robot - IR)*.

Kỹ thuật tự động hoá (TĐH) trong công nghiệp đã đạt tới trình độ rất cao: không chỉ TĐH các quá trình vật lý mà cả các quá trình xử lý thông tin. Vì vậy, TĐH trong công nghiệp tích hợp công nghệ sản xuất, kỹ thuật điện, điện tử, kỹ thuật điều khiển tự động trong đó có TĐH nhờ máy tính.

Hiện nay, trong công nghiệp tồn tại 3 dạng TĐH:

- TĐH cứng (*Fixed Automation*) được hình thành dưới dạng các thiết bị hoặc dây chuyền chuyên môn hoá theo đối tượng (sản phẩm). Nó được ứng dụng có hiệu quả trong điều kiện sản xuất hàng khối với sản lượng rất lớn các sản phẩm cùng loại.

- TĐH khả trình (*Programmable Automation*) được ứng dụng chủ yếu trong sản xuất loạt nhỏ, loạt vừa, đáp ứng phần lớn nhu cầu sản phẩm công nghiệp. Hệ thống thiết bị dạng này là các thiết bị vạn năng điều khiển số, cho phép dễ dàng lập trình lại để có thể thay đổi chủng loại (tức là thay đổi quy trình công nghệ sản xuất) sản phẩm.

- TĐH linh hoạt (*Flexible Automation*) là dạng phát triển của TĐH khả trình. Nó tích hợp công nghệ sản xuất với kỹ thuật điều khiển bằng máy tính, cho phép thay đổi đối tượng sản xuất mà không cần (hoặc hạn chế) sự can thiệp của con người. TĐH linh hoạt được biểu hiện dưới 2 dạng: tế bào sản xuất linh hoạt (*Flexible Manufacturing Cell - FMC*) và hệ thống sản xuất linh hoạt (*Flexible Manufacturing System - FMS*).

RBCN có 2 đặc trưng cơ bản:

- Là thiết bị vạn năng, được TĐH theo chương trình và có thể lập trình lại để đáp ứng một cách linh hoạt, khéo léo các nhiệm vụ khác nhau.

- Được ứng dụng trong những trường hợp mang tính công nghiệp đặc trưng, như vận chuyển và xếp dỡ nguyên vật liệu, lắp ráp, đo lường,...

Vì thể hiện 2 đặc trưng cơ bản trên của RBCN, hiện nay định nghĩa sau đây về robot công nghiệp do Viện nghiên cứu robot của Mỹ đề xuất được sử dụng rộng rãi:

RBCN là tay máy vạn năng, hoạt động theo chương trình và có thể lập trình lại để hoàn thành và nâng cao hiệu quả hoàn thành các nhiệm vụ khác nhau trong công nghiệp, như vận chuyển nguyên vật liệu, chi tiết, dụng cụ hoặc các thiết bị chuyên dùng khác.

Ngoài các ý trên, định nghĩa trong ΓOCT 25686-85 còn bổ sung cho RBCN chức năng *điều khiển* trong quá trình sản xuất:

RBCN là máy tự động được đặt cố định hay di động, bao gồm thiết bị thừa hành dạng tay máy có một số bậc tự do hoạt động và thiết bị điều khiển theo chương trình, có thể tái lập trình để hoàn thành các chức năng vận động và điều khiển trong quá trình sản xuất.

Chức năng *vận động* bao gồm các hoạt động "cơ bắp" như vận chuyển, định hướng, xếp đặt, gá kẹp, lắp ráp,... đối tượng. Chức năng *điều khiển* ám chỉ vai trò của robot như một phương tiện điều hành sản xuất, như cung cấp dụng cụ và vật liệu, phân loại và phân phối sản phẩm, duy trì nhịp sản xuất và thậm chí cả điều khiển các thiết bị liên quan.

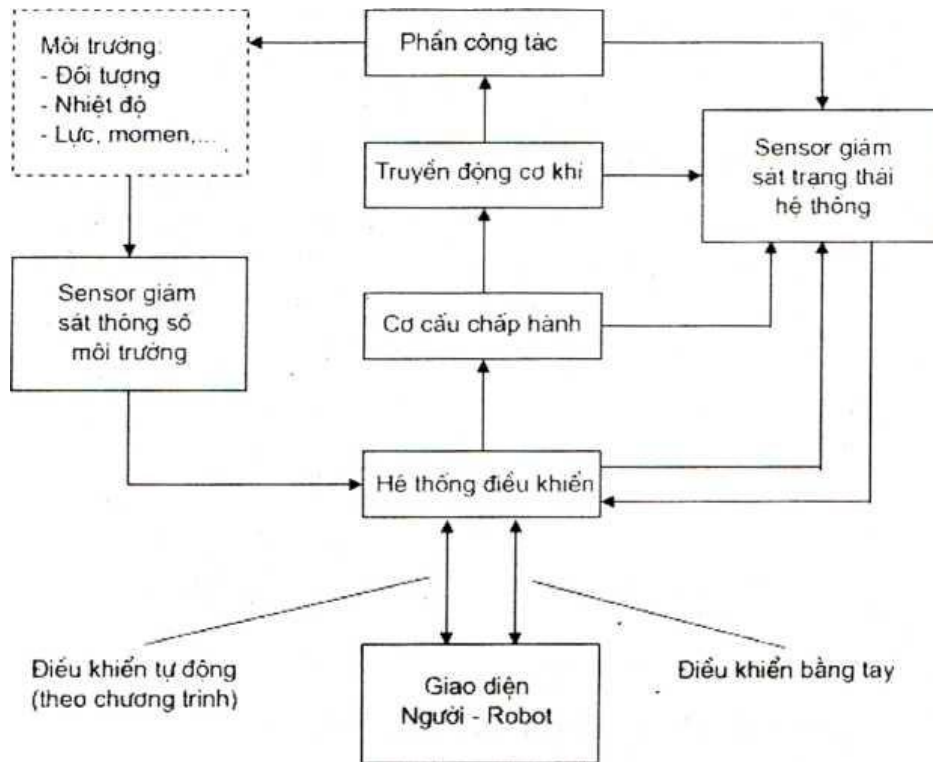
Với đặc điểm có thể lập trình lại, RBCN là thiết bị TĐH khả trình và ngày càng trở thành bộ phận không thể thiếu được của các tế bào hoặc hệ thống sản xuất linh hoạt.

1.2. Cấu trúc cơ bản của RBCN

1.2.1. Kết cấu chung

Một RBCN được cấu thành bởi các hệ thống sau (hình 1.1):

- Tay máy (*Manipulator*) là cơ cấu cơ khí gồm các khâu, khớp. Chúng hình thành *cánh tay* để tạo các chuyển động cơ bản, *cổ tay* tạo nên sự khéo léo, linh hoạt và bàn tay (*End Effector*) để trực tiếp hoàn thành các thao tác trên đối tượng.



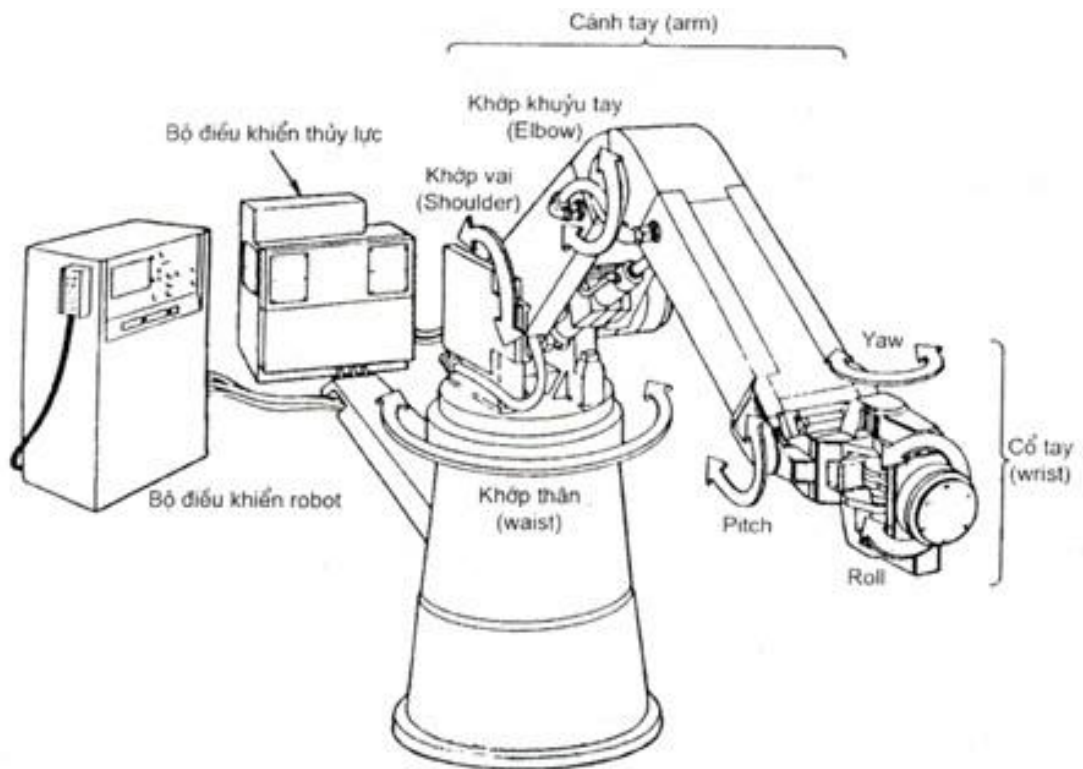
Hình 1.1: Sơ đồ khối của RBCN

- Cơ cấu chấp hành tạo chuyển động cho các khâu của tay máy. Nguồn động lực của các cơ cấu chấp hành là động cơ các loại: điện, thủy lực, khí nén hoặc kết hợp giữa chúng.

- Hệ thống cảm biến gồm các sensor và thiết bị chuyển đổi tín hiệu cần thiết khác. Các robot cần hệ thống sensor trong để nhận biết trạng thái của bản thân các cơ cấu của robot và các sensor ngoài để nhận biết trạng thái của môi trường.

- Hệ thống điều khiển (*Controller*) hiện nay thường là máy tính để giám sát và điều khiển hoạt động của robot.

Sơ đồ kết cấu chung của robot như trong hình 1.2.



Hình 1.2: Sơ đồ kết cấu chung của RBCN

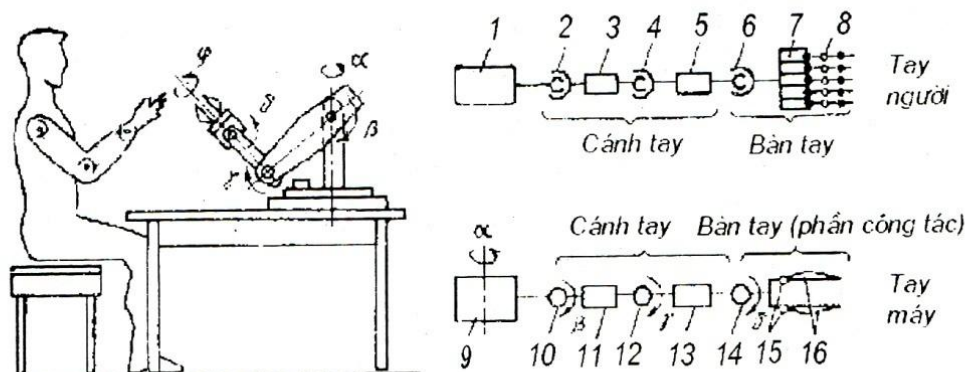
1.2.2. Kết cấu của tay máy

Tay máy là phần cơ sở, quyết định khả năng làm việc của RBCN. Đó là thiết bị cơ khí đảm bảo cho robot khả năng chuyển động trong không gian và khả năng làm việc, như nâng hạ vật, lắp ráp,... Ý tưởng ban đầu của việc thiết kế và chế tạo tay máy là phỏng tác cấu tạo và chức năng của tay người (hình 1.3). Về sau, đây không còn là điều bắt buộc nữa. Tay máy hiện nay rất đa dạng và nhiều loại có dáng vẻ khác rất xa với tay người. Tuy nhiên, trong kỹ thuật robot người ta vẫn dùng các thuật ngữ quen thuộc, như vai (*Shoulder*), cánh tay (*Arm*), cổ tay (*Wrist*), bàn tay (*Hand*) và các khớp (*Articulations*),... để chỉ tay máy và các bộ phận của nó.

Trong thiết kế và sử dụng tay máy, người ta quan tâm đến các thông số có ảnh hưởng lớn đến khả năng làm việc của chúng, như:

- Sức nâng, độ cứng vững, lực kẹp của tay,...
- Tầm với hay vùng làm việc: kích thước và hình dáng vùng mà phần công tác có thể với tới;

- Sự khéo léo, nghĩa là khả năng định vị và định hướng phần công tác trong vùng làm việc. Thông số này liên quan đến số bậc tự do của phần công tác.



Hình 1.3: Sự tương tác giữa tay người và tay máy

Để định vị và định hướng phần công tác một cách tùy ý trong không gian 3 chiều nó cần có 6 bậc tự do, trong đó 3 bậc tự do để định vị, 3 bậc tự do để định hướng. Một số công việc như nâng hạ, xếp dỡ,... yêu cầu số bậc tự do ít hơn 6. Robot hàn, sơn thường có 6 bậc tự do. Trong một số trường hợp cần sự khéo léo, linh hoạt hoặc cần tối ưu hoá quỹ đạo,... người ta có thể dùng robot với số bậc tự do lớn hơn 6.

Các tay máy có đặc điểm chung về kết cấu là gồm có các khâu, được nối với nhau bằng các khớp để hình thành một chuỗi động học hở, tính từ thân đến phần công tác. Các khớp được dùng phổ biến là khớp trượt và khớp quay. Tùy theo số lượng và cách bố trí các khớp mà có thể tạo ra tay máy kiểu tọa độ đề các, tọa độ trụ, tọa độ cầu, SCARA và kiểu tay người (*Anthropomorphic*).

Tay máy kiểu tọa độ đề các (hình 1.4), còn gọi là kiểu chữ nhật, dùng 3 khớp trượt, cho phép phần công tác thực hiện một cách độc lập các chuyển động thẳng, song song với 3 trục tọa độ. Vùng làm việc của tay máy có dạng hình hộp chữ nhật. Do sự đơn giản về kết cấu, tay máy kiểu này có độ cứng vững cao, độ chính xác được đảm bảo đồng đều trong toàn bộ vùng làm việc, nhưng ít khéo léo. Vì vậy, tay máy kiểu đề các được dùng để vận chuyển và lắp ráp.

Tay máy kiểu tọa độ trụ (hình 1.5) khác với tay máy kiểu đề các ở khớp đầu tiên: dùng khớp quay thay cho khớp trượt. Vùng làm việc của nó có dạng hình trụ rỗng.

Khớp trượt nằm ngang cho phép tay máy "thò" được vào khoang rỗng nằm ngang. Độ cứng vững cơ học của tay máy trụ tốt, thích hợp với tải nặng, nhưng độ chính xác định vị góc trong mặt phẳng nằm ngang giảm khi tầm với tăng.

Tay máy kiểu toạ độ cầu (hình 1.6) khác kiểu trụ do khớp thứ hai (khớp trượt) được thay bằng khớp quay. Nếu quỹ đạo chuyển động của phần công tác được mô tả trong toạ độ cầu thì mỗi bậc tự do tương ứng với một khả năng chuyển động và vùng làm việc của nó là khối cầu rỗng. Độ cứng vững của loại tay máy này thấp hơn 2 loại trên và độ chính xác định vị phụ thuộc vào tầm với. Tuy nhiên, loại này có thể "nhặt" được cả vật dưới nền.

SCARA (hình 1.7) được đề xuất lần đầu vào năm 1979 tại Trường đại học Yamanashi (Nhật bản) dùng cho công việc lắp ráp. Đó là một kiểu tay máy có cấu tạo đặc biệt, gồm 2 khớp quay và 1 khớp trượt, nhưng cả 3 khớp đều có trục song song với nhau. Kết cấu này làm tay máy cứng vững hơn theo phương thẳng đứng nhưng kém cứng vững (*Compliance*) theo phương được chọn (*Selective*), là phương ngang. Loại này chuyên dùng cho công việc lắp ráp (*Assembly*) với tải trọng nhỏ, theo phương thẳng đứng. Từ SCARA là viết tắt của "*Selective Compliance Assembly Robot Arm*" để mô tả các đặc điểm trên. Vùng làm việc của SCARA là một phần của hình trụ rỗng, như trong hình 1.7.

Tay máy kiểu tay người (Anthropomorphic), như được mô tả trong hình 1.8, có cả 3 khớp đều là các khớp quay, trong đó trục thứ nhất vuông góc với 2 trục kia. Do sự tương tự với tay người, khớp thứ hai được gọi là khớp vai (*Shoulder joint*), khớp thứ ba là khớp khuỷu (*Elbow joint*), nối *cẳng tay* với *khủy tay*. Với kết cấu này, không có sự tương ứng giữa khả năng chuyển động của các khâu và số bậc tự do. Tay máy làm việc rất khéo léo, nhưng độ chính xác định vị phụ thuộc vị trí của phần công tác trong vùng làm việc. Vùng làm việc của tay máy kiểu này gần giống một phần khối cầu.

Toàn bộ dạng các kết cấu tả ở trên mới chỉ liên quan đến khả năng định vị của phần công tác. Muốn định hướng nó, cần bổ sung phần *cổ tay*. Muốn định hướng một cách tùy ý phần công tác, cổ tay phải có ít nhất 3 chuyển động quay quanh 3 trục vuông góc với nhau. Trong trường hợp trục quay của 3 khớp gặp nhau tại một điểm thì ta gọi đó là khớp cầu (hình 1.9). Ưu điểm chính của khớp cầu là tách được thao tác

định vị và định hướng của phần công tác, làm đơn giản việc tính toán. Các kiểu khớp khác có thể đơn giản hơn về kết cấu cơ khí, nhưng tính toán tọa độ khó hơn, do không tách được 2 loại thao tác trên.

Phần công tác là bộ phận trực tiếp tác động lên đối tượng. Tùy theo yêu cầu làm việc của robot, phần công tác có thể là tay gắp (*Gripper*), công cụ (súng phun sơn, mỏ hàn, dao cắt, chìa vặn ốc,...).

1.3. Phân loại robot

Thế giới robot hiện nay đã rất phong phú và đa dạng, vì vậy phân loại chúng không đơn giản. Có rất nhiều quan điểm phân loại khác nhau. Mỗi quan điểm phục vụ một mục đích riêng. Tuy nhiên, có thể nêu ra đây 3 cách phân loại cơ bản: theo kết cấu, theo điều khiển và theo phạm vi ứng dụng của robot.

1.3.1. Phân loại theo kết cấu

Theo kết cấu (hay theo hình học), người ta phân robot thành các loại: đề các, trụ, cầu, SCARA, kiểu tay người và các dạng khác nữa (xem các hình từ 1.4 đến hình 1.9). Điều này đã được trình bày trong mục 1.2.2.

1.3.2. Phân loại theo điều khiển

Có 2 kiểu điều khiển robot: điều khiển hở và điều khiển kín.

Điều khiển hở, dùng truyền động bước (động cơ điện hoặc động cơ thủy lực, khí nén,...) mà quãng đường hoặc góc dịch chuyển tỷ lệ với số xung điều khiển. Kiểu điều khiển này đơn giản, nhưng đạt độ chính xác thấp.

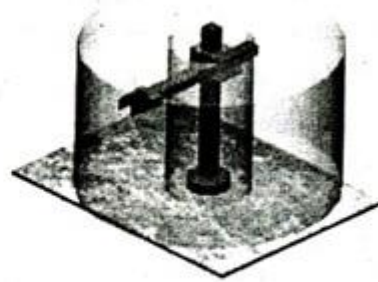
Điều khiển kín (hay điều khiển servo), sử dụng tín hiệu phản hồi vị trí để tăng độ chính xác điều khiển. Có 2 kiểu điều khiển servo: điều khiển điểm - điểm và điều khiển theo đường (contour).

Với kiểu điều khiển điểm - điểm, phần công tác dịch chuyển từ điểm này đến điểm kia theo đường thẳng với tốc độ cao (không làm việc). Nó chỉ làm việc tại các điểm dừng. Kiểu điều khiển này được dùng trên các robot hàn điểm, vận chuyển, tán đinh, bắn đinh,...

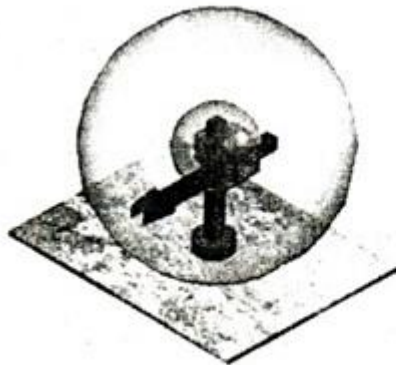
Điều khiển contour đảm bảo cho phần công tác dịch chuyển theo quỹ đạo bất kỳ, với tốc độ có thể điều khiển được. Có thể gặp kiểu điều khiển này trên các robot hàn hồ quang, phun sơn.



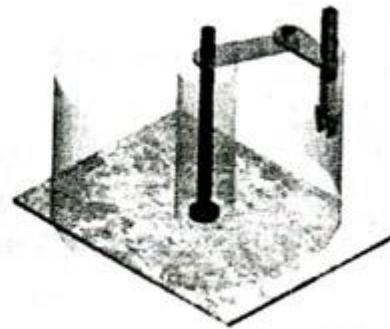
Hình 1. 4: Tay máy kiểu tọa độ đề các



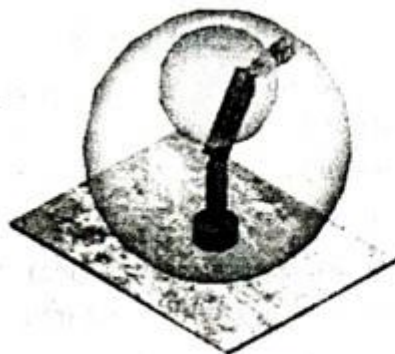
Hình 1. 5: Cấu trúc tay máy kiểu tọa độ trụ



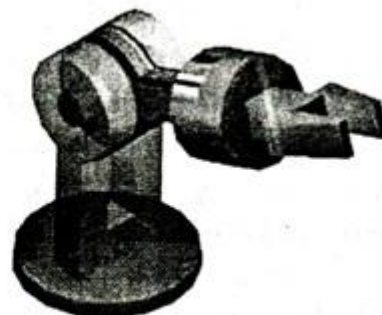
Hình 1. 6: Tay máy kiểu cầu



Hình 1. 7: Tay máy kiểu SCARA



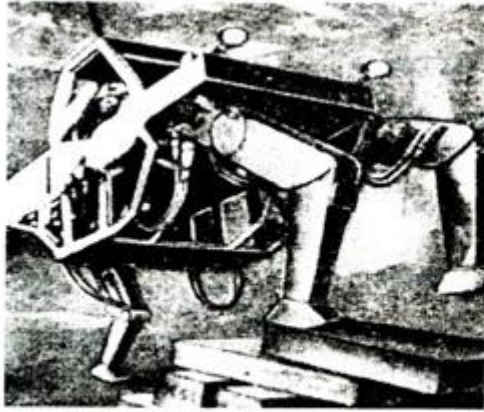
Hình 1. 8: Tay máy kiểu tay người



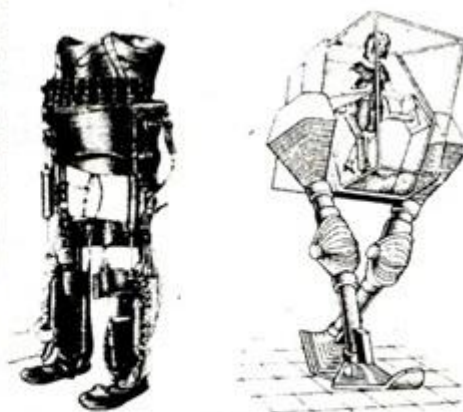
Hình 1. 9: Khớp cổ tay

1.3.3. Phân loại theo ứng dụng

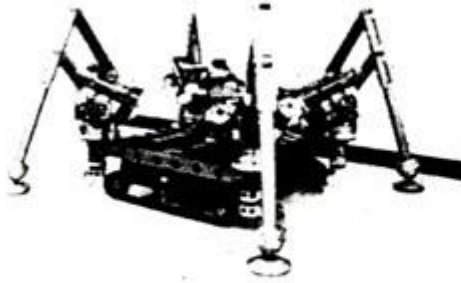
Cách phân loại này dựa vào ứng dụng của robot. Ví dụ, có robot công nghiệp, robot dùng trong nghiên cứu khoa học, robot dùng trong kỹ thuật vũ trụ, robot dùng trong quân sự,... (hình 1.10).



Robot leo cầu thang (General Electric - USA)



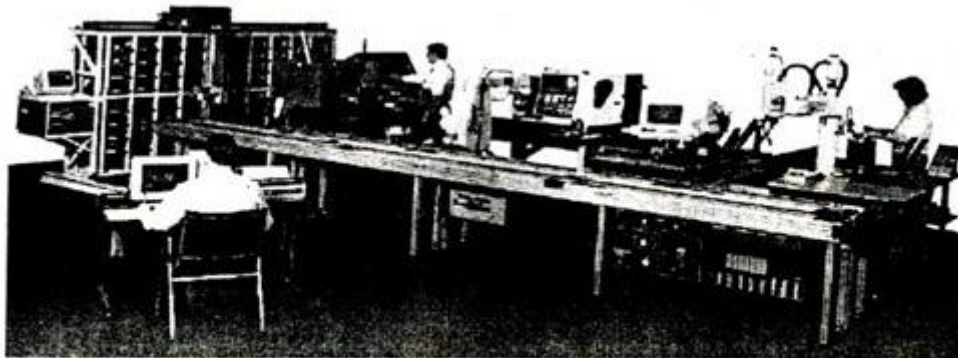
"Chân máy"



Robot chân nhện (Mech. Laboratory-Japan)



Robot làm việc dưới nước (OCA)



Robot trong hệ thống sản xuất linh hoạt

Hình 1.10: Một số loại robot được ứng dụng trong thực tế

1.4. Bài toán thuận của động học tay máy

Trong đại đa số các trường hợp, tay máy là một chuỗi động hờ, được cấu tạo bởi một số khâu (*Links*), được nối với nhau nhờ các khớp. Một đầu của chuỗi nối với giá (*Base*), còn đầu kia nối với phần công tác. Mỗi khâu hình thành cùng với khớp phía

trước nó một cặp khâu - khớp. Tùy theo kết cấu của mình mà mỗi loại khớp đảm bảo cho khâu nối sau nó các khả năng chuyển động nhất định.

Mỗi khớp (thực chất là cặp khâu - khớp) được đặc trưng bởi 2 loại thông số:

- Các thông số không thay đổi giá trị trong quá trình làm việc của tay máy được gọi là *tham số*.
- Các thông số thay đổi khi tay máy làm việc được gọi là các biến khớp.

Hai loại khớp thông dụng nhất trong kỹ thuật tay máy là *khớp trượt* và *khớp quay*. Chúng đều là loại khớp có một bậc tự do.

Bài toán thuận nhằm mô tả *thế (vị trí và hướng) của phần công tác* dưới dạng *hàm số của các biến khớp*. Giả sử có một tay máy với $n+1$ khâu và n khớp (hình 2.13).

Thế của phần công tác so với hệ tọa độ gốc $O_0 - x_0 y_0 z_0$ được mô tả bằng vector định vị p^0 và hướng của các vector chỉ phương n, s, a . Phép chuyển đổi tọa độ được biểu diễn bằng ma trận chuyển đổi thuận nhất:

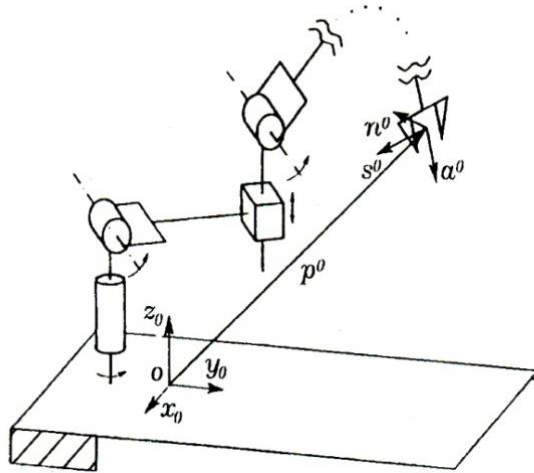
$$T^0(q) = \begin{bmatrix} n^0(q) & s^0(q) & a^0(q) & p^0(q) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.32)$$

Trong đó, q là vector n phần tử, gồm các biến khớp; p là vector định vị; n, s, a là các vector chỉ phương của phần công tác, cũng chính là vector đơn vị của các trục tọa độ. Nếu phần công tác là tay gấp thì gốc tọa độ đặt vào tâm quay; vector a đặt theo phương tiến đến vật; s nằm trong mặt phẳng trượt của hàm kẹp; n vuông góc với a và s theo quy tắc bàn tay phải.

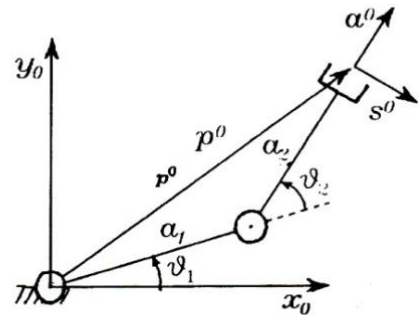
Một trong những phương pháp giải bài toán thuận là dùng trực tiếp hình học giải tích. Ví dụ, đối với trường hợp cơ cấu 2 khâu phẳng (hình 2.14), ta có:

$$T^0 q = \begin{bmatrix} n^0 & s^0 & a^0 & p^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & s_x^0 & a_x^0 & p_x^0 \\ 0 & s_y^0 & a_y^0 & p_y^0 \\ 1 & s_z^0 & a_z^0 & p_z^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & s_{12} & c_{12} & a_1 c_1 + a_2 c_2 \\ 0 & -c_{12} & s_{12} & a_1 s_1 + a_2 s_{12} \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Phương pháp tính toán trực tiếp chỉ áp dụng được cho các cơ cấu đơn giản. Để có thể giải các bài toán tổng quát cần một thuật giải chung. Một trong những thuật giải như vậy xuất phát từ *quy tắc Denavit-Hartenberg*, được Denavit và Hartenberg xây dựng vào năm 1955. Đó là quy tắc thiết lập hệ thống toạ độ trên các cặp khâu - khớp trên tay máy. Dựa trên hệ toạ độ này có thể mô tả các cặp bằng hệ thống các tham số, biến khớp và áp dụng một dạng phương trình tổng quát cho bài toán động học tay máy.



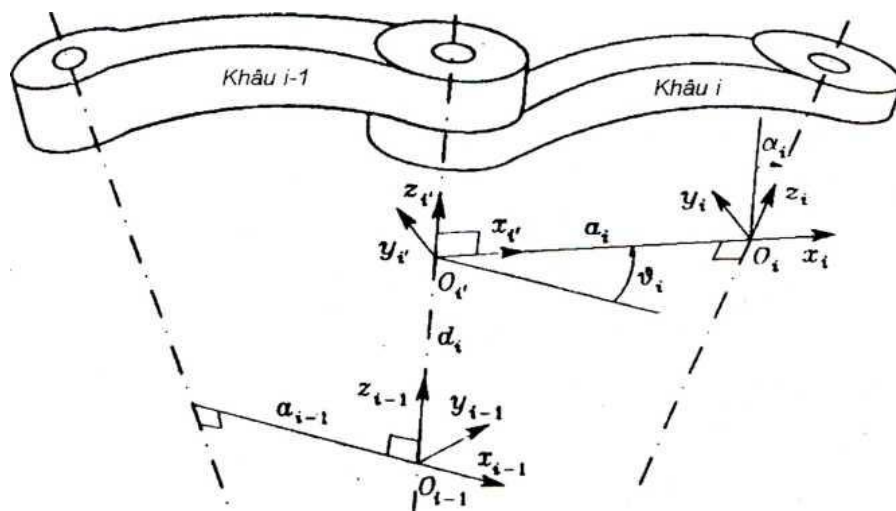
Hình 2.13: Mô tả thể của phần công tác



Hình 2.14: Chuỗi phẳng 2 khâu

1.4.1. Mô tả quy tắc Denavit-Hartenberg

Giả sử trong chuỗi động học của tay máy có n khâu, khâu thứ i nối khớp thứ i với khớp thứ $i+1$ (hình 2.15).



Hình 2.15: Biểu diễn các thông số động học theo quy tắc Denavit-Hartenberg

Theo quy tắc Denavit-Hartenberg thì hệ tọa độ được gắn lên các khâu, khớp như sau:

- Đặt trục tọa độ z_j dọc theo trục của khớp sau (thứ $i+1$).
- Đặt gốc tọa độ O_i tại *giao điểm giữa* z_i và pháp tuyến chung nhỏ nhất của trục z_i và z_{i-1} . Giao điểm của pháp tuyến chung với trục z_{i-1} là gốc O_i của hệ $O_i - x_i y_i z_i$.

Quy tắc *Denavit-Hartenberg* có một số trường hợp đặc biệt, cho phép đơn giản hoá thủ tục tính toán:

- Đối với hệ tọa độ gốc chỉ có phương của trục z_0 là xác định. Góc $0,1$ và trục x_j có thể *chọn* tùy ý.
- Đối với hệ thứ n , chỉ có phương của trục x_i là xác định. Trục z_i có thể *chọn* tùy ý.
- Khi 2 khớp liền nhau có trục song song, vị trí của pháp tuyến chung có thể lấy bất kỳ.
- Khi trục của 2 khớp liền nhau có trục cắt nhau, phương của trục x_i có thể *chọn* bất kỳ.

- Khi khớp thứ i là khớp trượt thì chỉ có phương của trục z_{i-1} là xác định.
- Đặt trục tọa độ x_i theo phương pháp tuyến chung giữa z_{i-1} và z_i hướng *từ khớp thứ i đến khớp thứ $i+1$*
- Trục y_i , vuông góc với x_i , và z_i theo quy tắc bàn tay phải.

Sau khi được thiết lập, vị trí của hệ $O_i - x_i y_i z_i$ so với hệ $O_{i-1} - x_{i-1} y_{i-1} z_{i-1}$ hoàn toàn xác định nhờ các thông số sau:

- $a_i = O_i O_i'$: khoảng cách giữa 2 khớp liên tiếp theo phương x_i
- $d_i = O_{i-1} O_i'$: khoảng cách giữa 2 khớp liên tiếp theo phương z_{i-1}
- α_i : góc quay quanh trục x_i , giữa z_{i-1} và z_i
- θ_i : góc quay quanh trục z_{i-1} giữa x_{i-1} và x_i

Trong 4 thông số trên thì a_i , và α_i , chỉ phụ thuộc vào kết cấu của khâu thứ i . Nếu là khớp quay thì θ_i là biến, còn $d_i = const$. Với khớp trượt thì θ_i là biến, còn $\theta_i = const$.

Đến đây, có thể mô tả phép chuyển tọa độ giữa hệ i và hệ $i-1$, như sau:

- Tịnh tiến hệ $O'_{i-1}x'_iy'_iz'_i$ dọc theo trục z_{i-1} một khoảng d_i , sau đó quay một góc để nhận được hệ $O_{i-1}x_iy_iz_i$. .. Ma trận chuyển đổi thuần nhất tương ứng là:

$$A_i^{i-1} = \begin{bmatrix} c_{g_i} & -s_{\alpha_i} & 0 & 0 \\ -s_{g_i} & c_{g_i} & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

- Tịnh tiến hệ $O'_{i-1}x'_iy'_iz'_i$ vừa nhận được một khoảng a_i dọc trục x_i , sau đó quay nó quanh trục x_i một góc α_i , để nhận được hệ $O_{i-1}x_iy_iz_i$. Ma trận chuyển đổi thuần nhất tương ứng là:

$$A_i^{i'} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Ma trận tổng hợp nhận được bằng cách nhân hai ma trận trên:

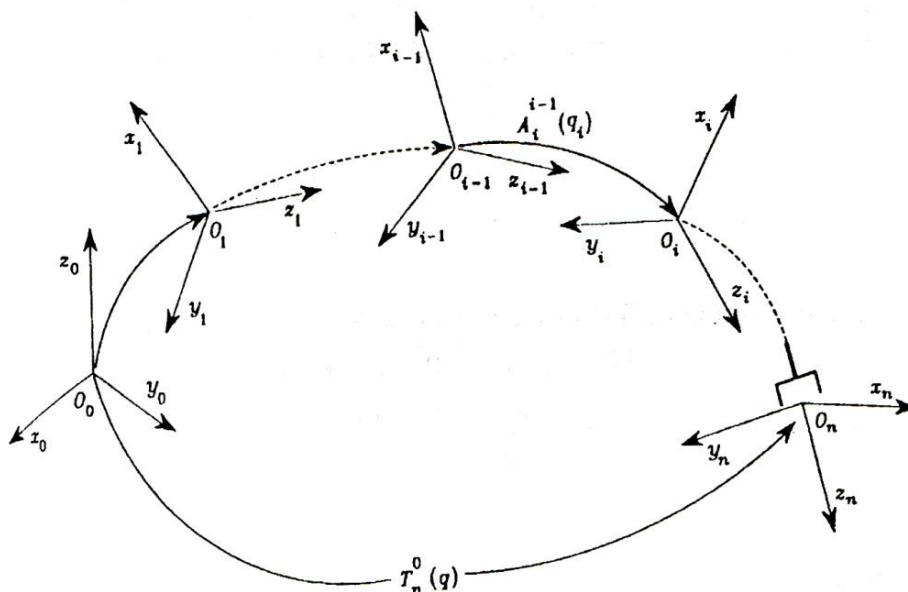
$$A_i^{i-1} q_i = A_i^{i-1} A_i^{i'} = \begin{bmatrix} c_{g_i} & -s_{g_i}c_{\alpha_i} & s_{g_i}c_{\alpha_i} & a_i c_{g_i} \\ s_{g_i} & c_{g_i}c_{\alpha_i} & -c_{g_i}c_{\alpha_i} & a_i s_{g_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.33)$$

Chú ý rằng, ma trận chuyển vị từ hệ i đến hệ $i-1$ là hàm của các biến khớp q_i , (nếu khớp thứ i là khớp quay) hoặc d_i (nếu khớp thứ i là khớp trượt).

Một cách tổng quát, quy tắc Denavit-Hartenberg cho phép tổ hợp các ma trận chuyển vị riêng rẽ thành một ma trận chuyển vị thuần nhất, biểu diễn vị trí và hướng của khâu n so với khâu cơ sở.

$$T_n^0 q = \begin{bmatrix} n_x^0 & s_x^0 & a_x^0 & p_x^0 \\ n_y^0 & s_y^0 & a_y^0 & p_y^0 \\ n_z^0 & s_z^0 & a_z^0 & p_z^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_1^0(q_1)A_2^1(q_2) \dots A_n^{n-1}(q_n) \quad (2.34)$$

Quy tắc này có thể áp dụng cho chuỗi hờ bất kỳ trong kết cấu tay máy, như biểu diễn trong hình 2.16.

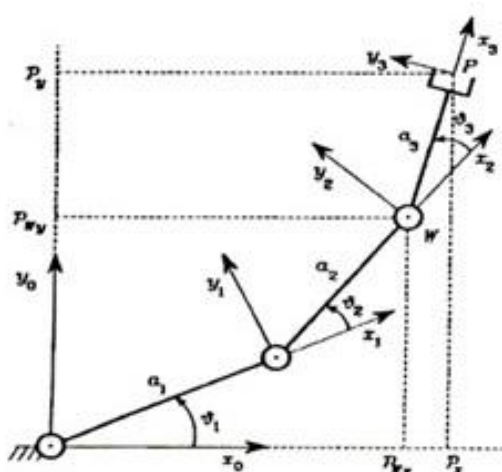


Hình 2.16: Sơ đồ chuyển vị của phần công tác so với cơ sở

1.4.2. Một số ví dụ áp dụng quy tắc Denavit-Hartenberg

Cơ cấu 3 khâu phẳng

Cơ cấu có 3 khớp quay với các trục song song. Đặt trục x_i dọc theo phương của các khâu, còn các tham số $d_i = 0$. Các biến khớp là các góc quay ϑ_i . Sơ đồ động học và bảng tham số Denavit-Hartenberg như trên hình 2.17.



Bảng thông số Denavit-Hartenberg

| Khâu | a_i | α_i | d_i | ϑ_i |
|------|-------|------------|-------|---------------|
| 1 | a_1 | 0 | 0 | ϑ_1 |
| 2 | a_2 | 0 | 0 | ϑ_2 |
| 3 | a_3 | 0 | 0 | ϑ_3 |

Hình 2.17: Cơ cấu ba khâu phẳng

Vì các cặp khâu - khớp có kết cấu tương tự nhau, nên từ (2.34) có thể viết cả 3 ma trận chuyển đổi thuần nhất dưới dạng như nhau:

$$A_i^{i-1}(\mathcal{G}_i) = \begin{bmatrix} c_i & -s_i & 0 & a_i c_i \\ s_i & c_i & 0 & a_i s_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{với } i = 1, 2, 3)$$

Ma trận chuyển vị (2.34) bây giờ trở thành:

$$T_3^0(q) = A_1^0 A_2^1 A_3^2 = \begin{bmatrix} c_{123} & -s_{123} & 0 & a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ s_{123} & c_{123} & 0 & a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.35)$$

trong đó $q = \mathcal{G}_1 \mathcal{G}_2 \mathcal{G}_3^T$

Cơ cấu tọa độ cầu

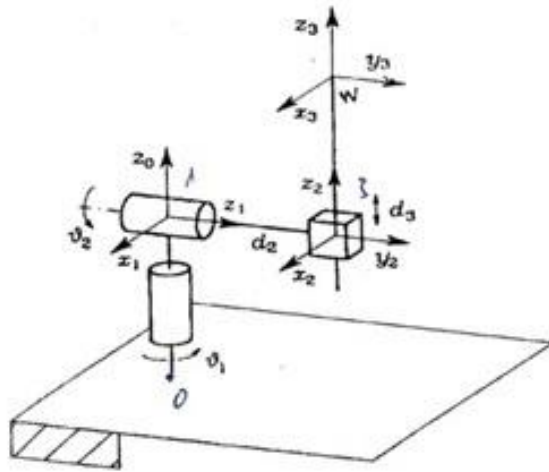
Cơ cấu tay máy cầu và bảng tham số của nó được cho trong hình 2.18. Vì z_0 và z_1 cắt nhau, nên $d_1 = 0$.

Từ (2.34), có thể viết các ma trận chuyển vị thành phần như sau:

$$A_1^0(\mathcal{G}_1) = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^0(\mathcal{G}_2) = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & c_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^0(\mathcal{G}_3) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Bảng tham số Denavit-Hartenberg

| Khớp | a_i | α_i | d_i | ϑ_i |
|-------|-------|------------|-------|---------------|
| A_1 | 0 | $-\pi/2$ | 0 | ϑ_1 |
| A_2 | 0 | $\pi/2$ | d_2 | ϑ_2 |
| A_3 | 0 | 0 | d_3 | 0 |

Hình 2.18: Tay máy kiểu tọa độ cầu

Ma trận chuyển vị tổng hợp

$$T_3^0(q) = A_1^0 A_2^1 A_3^2 = \begin{bmatrix} c_1 c_2 & -s_1 & c_1 s_2 & c_1 s_2 d_3 + s_1 d_2 \\ s_1 c_2 & c_1 & s_1 s_2 & s_1 s_2 d_3 + c_1 d_2 \\ -s_2 & 0 & c_2 & c_2 d_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.36)$$

trong đó $q = \vartheta_1 \vartheta_2 \vartheta_3^T$

1.4.3. Vùng hoạt động của phần công tác

Như đã nói ở trên, vị trí của phần công tác được đại diện bởi vị trí của gốc tọa độ gắn trên nó so với hệ tọa độ chung, nghĩa là bởi vector p . Tương tự, hướng của phần công tác được mô tả thông qua bộ các tham số MRO và, một cách hình thức, ta biểu diễn bằng vector ϕ . Tổng hợp lại, thế của phần công tác được biểu diễn bằng vector

$m \times 1$, với $m \leq$ số biến khớp (n):

$$x = \begin{bmatrix} p \\ \phi \end{bmatrix} \quad (2.37)$$

Biểu thức này dùng một số lượng tối thiểu các thông số độc lập nhau để mô tả thế của phần công tác. Nó cũng biểu diễn vùng, trong đó tay máy có thể hoạt động theo đúng chức năng của nó, gọi là vùng hoạt động.

Vector $n \times 1$ biểu diễn miền giá trị của các biến khớp q_i , gọi là *không gian khớp*:

$$q = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} \quad (2.38)$$

trong đó, $q_i = \mathcal{G}_i$ cho khớp quay; $q_i = d_i$, cho khớp trượt.

Bằng cách này có thể viết phương trình động học của tay máy dưới dạng khác:

$$x = k(q) \quad (2.39)$$

Ví dụ, với cơ cấu 3 khâu phẳng (hình 2.17), có thể nhận thấy vị trí của phần công tác được xác định nhờ 2 tọa độ p_x, p_y , còn hướng của phần công tác được xác định nhờ góc ϕ giữa nó với trục x_0 . Đối chiếu với (2.35), có thể biểu diễn vị trí của phần công tác thông qua 2 phần tử đầu của cột thứ tư, còn hướng của nó qua góc $\phi = \mathcal{G}_1 = \mathcal{G}_1 + \mathcal{G}_2 + \mathcal{G}_3$

$$x = \begin{bmatrix} p_x \\ p_y \\ \phi \end{bmatrix} = k(q) = \begin{bmatrix} a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ \mathcal{G}_1 + \mathcal{G}_2 + \mathcal{G}_3 \end{bmatrix}$$

Đó cũng chính là trường hợp riêng của (2.37), biểu diễn vùng hoạt động của tay máy 3 khâu phẳng.

Một trong những thông số động học quan trọng của tay máy là *vùng làm việc (Workspace)* của nó. Đó là không gian mà gốc tọa độ của phần công tác có thể với tới được, tức là không tính đến sự định hướng của phần công tác. Đôi khi người ta phân biệt vùng làm việc nói trên (gọi là *Reachable workspace*) với vùng làm việc có tính đến sự định hướng của phần công tác (*Dexterous Workspace*).

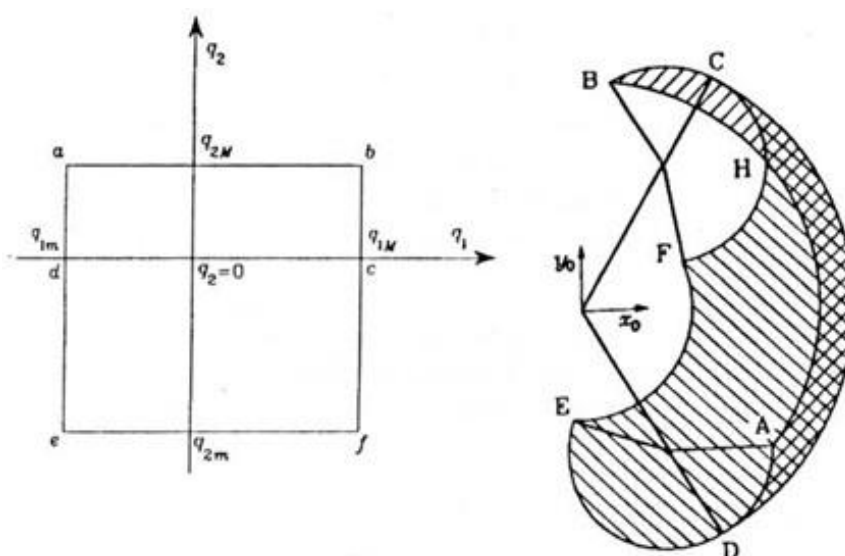
Thể tích và hình dạng của vùng làm việc phụ thuộc vào kết cấu của tay máy và giới hạn (miền giá trị) của các biến khớp. Đối với tay máy có n bậc tự do, vùng làm việc là tập hợp mọi vị trí có thể của phần công tác, như mô tả trong phương trình động học:

$$p = p(q); q_{im} \leq q_{iM}; i = 1 \dots n.$$

trong đó, q_{im} (q_{iM}) là giá trị giới hạn dưới (trên) của mỗi biến khớp. Vùng làm việc này có các tính chất: *cố giới hạn, khép kín* và *liên thông*.

Biểu diễn hình học vùng công tác của tay máy là điều phức tạp. Tuy nhiên, chúng ta có thể hình dung về vùng làm việc của tay máy thông qua một ví dụ đơn giản. Đó là cơ cấu 2 khâu phẳng (hình 2.19).

Cấu hình và thể của cẳng tay được thể hiện bằng hình chữ nhật kín $abcfeda$. Đoạn ab tương ứng với $q_2 = q_{2M}$, còn q_1 biến thiên từ q_{1m} đến q_{1M} . Ta vẽ được cung AB tương ứng trong hình bên phải. Cung BF tương ứng đoạn bf : $q_1 = q_{1M}$ còn q_2 giảm từ q_{2M} đến q_{2m} . Tương tự, ta vẽ được các cung FE , EA . Vùng công tác còn được hình thành bởi cung CD ứng với $q_2 = 0$, còn q_1 biến thiên từ q_{1m} đến q_{1M} .



Hình 2.19: Phương pháp xây dựng vùng làm việc của cơ cấu hai khâu phẳng

1.5. Bài toán ngược của động học tay máy

Bài toán thuận của động học tay máy cho phép xác định thể của phần công tác và có thể cả vùng làm việc của nó theo quan hệ với các thông số động học của các cặp khâu - khớp. Bài toán ngược nhằm xác định bộ thông số động học để đảm bảo chuyển động cho trước của phần công tác. Theo biểu thức (2.34), nếu có bộ các thông số xác định thì có thể xác định $T_n^0(q)$ một cách đơn trị. Đối với bài toán ngược thì không hẳn như vậy, vì:

- Các phương trình có dạng phi tuyến và siêu việt, thường không cho lời giải đúng.

- Có thể có nhiều lời giải.
- Có thể gặp nghiệm vô định, vì các liên kết thừa (giống như kết cấu siêu tĩnh).
- Có thể có nghiệm tìm được bằng toán học lại không chấp nhận được về mặt vật lý, do các ràng buộc về kết cấu.

Tính đa nghiệm của bài toán ngược không chỉ phụ thuộc vào số các biến khớp (tức là số bậc tự do) mà cả vào số lượng các tham số khác không trong kết cấu. Nói chung, số lượng này càng lớn thì số lời giải chấp nhận được càng nhiều. Ví dụ, tay máy số bậc tự do có thể có tới 16 nghiệm. Điều đó đòi hỏi phải có các điều kiện phụ về cơ cấu để giảm số nghiệm này.

Việc tìm kiếm một nghiệm phù hợp đòi hỏi ở người thiết kế một trực giác về toán học và về kết cấu để dự đoán những điểm hoặc khu vực khả dĩ giảm được số nghiệm cần chọn lựa.

1.5.1. Cơ cấu 3 khâu phẳng

Cơ cấu 3 khâu phẳng được hình dung như trong hình 2.17. So sánh phương trình động học của nó (2.35) với phương trình động học tổng quát (2.34) ta có thể xác định tọa độ của điểm W (gốc tọa độ của khớp 2) [3, tr.42]:

$$\left. \begin{aligned} P_{w_x} &= p_x - a_3 c_\phi = a_1 c_1 + a_2 c_{12} \\ P_{w_y} &= p_y - a_3 s_\phi = a_1 s_1 + a_2 s_{12} \end{aligned} \right\} \quad (2.40)$$

Mặt khác, từ hình học của cơ cấu, có giới hạn về góc nghiêng của phần công tác với thân:

$$\phi = \vartheta_1 + \vartheta_2 + \vartheta_3 \quad (2.41)$$

Bình phương rồi cộng 2 vế của (2.40) sẽ rút ra được

$$c_2 = \frac{p_{w_x}^2 + p_{w_y}^2 - a_1^2 - a_2^2}{2a_1 a_2}$$

Tất nhiên, c_2 phải thỏa mãn điều kiện $-1 \leq c_2 \leq 1$

Từ đó ta có $s_2 = \pm \sqrt{1 - c_2^2}$

Trong đó, dấu cộng ứng với thế bàn tay hướng lên, còn dấu trừ ứng với thế bàn tay hướng xuống. Vậy:

$$\mathcal{G}_2 = A \tan 2(s_2, c_2)$$

Thay \mathcal{G}_2 vào (2. 40) rồi giải hệ phương trình đó, nhận được

$$s_1 = \frac{(a_1 + a_2 c_2) p_{w_y} - a_2 s_2 p_{w_x}}{p_{w_x}^2 + p_{w_y}^2}$$

$$\mathcal{G}_1 = A \tan 2(s_1, c_1)$$

Cuối cùng, tính được: $\mathcal{G}_3 = \phi - \mathcal{G}_1 - \mathcal{G}_2$

1.5.2. Cơ cấu cầu

Cơ cấu cầu (hình 2.18) và phương trình động học của nó như (2.36). Cần xác định giá trị của các biến khớp $\mathcal{G}_1, \mathcal{G}_2, d_3$ tương ứng vị trí xác định của điểm gốc tọa độ W trên phần công tác. Để thuận tiện, vị trí của W được xác định theo hệ 1. Vậy, từ (2.36) có biểu thức

$$A_1^0{}^{-1} T_3^0 = A_2^1 A_3^2 = \begin{bmatrix} c_2 & 0 & s_2 & s_3 s_2 \\ s_2 & 0 & -c_2 & -d_3 c_2 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Bình phương 3 phần tử đầu của cột thứ 4 của ma trận trên 2 vế, được biểu thức chỉ phụ thuộc \mathcal{G}_2 và d_3

$$p^1 w = \begin{bmatrix} p_{w_x} c_1 + p_{w_y} s_1 \\ -p_{w_z} \\ -p_{w_x} s_1 + p_{w_y} c_1 \end{bmatrix} = \begin{bmatrix} d_3 s_2 \\ -d_3 c_2 \\ d_2 \end{bmatrix} \quad (2.42)$$

Để giải phương trình, đặt

$$t = \tan(\mathcal{G}_1/2)$$

để cho $c_1 = \frac{1-t^2}{1+t^2}$ và $s_1 = \frac{2t}{1+t^2}$

Thay chúng vào vế trái của (2.42), được phương trình

$$(d_2 + p_{w_x})t^2 + 2p_{w_x}t + d_2 - p_{w_y} = 0$$

Giải ra, ta được

$$t = \frac{-p_{w_x} \pm \sqrt{p_{w_x}^2 + p_{w_y}^2 - d_2^2}}{d_2 + p_{w_y}}$$

với điều kiện biệt thức phải dương. Hai nghiệm tương ứng với 2 thể khác nhau của bàn tay là

$$\mathcal{G}_1 = 2A \tan 2(-p_{w_x} \pm \sqrt{p_{w_x}^2 + p_{w_y}^2 - d_2^2}, d_2 + p_{w_y})$$

Từ hai phân tử đầu của (2.42), nhận được

$$\frac{p_{w_x}c_1 + p_{w_y}s_1}{-p_{w_z}} = \frac{d_3s_2}{-d_3c_2}$$

Từ đó: $\mathcal{G} = A \tan 2(p_{w_x}c_1 + p_{w_y}s_1, p_{w_z})$

Cuối cùng, bình phương và cộng 2 phân tử đầu của (2.42), được:

$$d_3 = \sqrt{(p_{w_x}c_1 + p_{w_y}s_1)^2 + p_{w_z}^2}$$

với điều kiện $d_3 > 0$

Chương 2:

XÂY DỰNG VÀ ĐIỀU KHIỂN

CÁNH TAY ROBOT BA BẬC TỰ DO

Các chương trước của đề tài đã tập trung giải quyết các vấn đề thuộc về cấu trúc, động học, động lực học, thiết kế quỹ đạo,... của tay máy. Chương này sẽ đề cập đến hệ thống điều khiển, nếu không có nó thì robot không hoạt động được.

Muốn điều khiển được robot thì phải có 4 yếu tố:

- Hệ thống phát động và chấp hành
- Hệ thống cảm biến (Sensors)
- Bộ điều khiển (Controller), gồm phần cứng và phần mềm hệ thống
- Chương trình điều khiển.

Nó nhằm tạo cho người sử dụng robot một cơ sở thực tiễn cho phù hợp với yêu cầu công nghệ để thiết kế chế tạo và điều khiển cánh tay robot.

2.1. Xây dựng phần cơ khí của robot

Xây dựng phần cơ khí phải căn cứ vào sức nâng của tay máy, đó là khối lượng lớn nhất của vật mà robot có thể nâng được. Thiết kế robot cũng tuân theo những nguyên tắc chung của việc thiết kế máy. Trong phần này tác giả trình bày về cơ cấu cơ khí nhưng cũng phải bám theo yêu cầu về công nghệ để đưa ra một con robot với yêu cầu kỹ thuật như sau:

- Hoạt động linh hoạt
- Chắc chắn, gọn gàng về cơ khí
- Đảm bảo hài hòa hợp giữa robot với môi trường
- Sự hòa hợp giữa robot với người dùng
- Thiết kế có xu hướng sản xuất

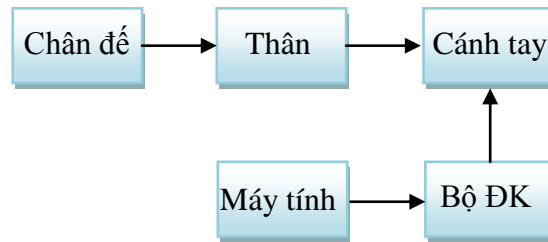
Với yêu cầu về khối lượng nâng cũng như yêu cầu công nghệ, tác giả xin trình bày về cấu tạo cánh tay robot do tác giả tự chế tạo bao gồm các bộ như sau:

2.1.1. Cấu tạo của cánh tay robot

Cấu tạo của cánh tay robot được chia làm 3 phần (Hình 2.1):

- Phần chân đế

- Thân robot
- Phần cánh tay



Hình 2.1: Sơ đồ khối của cánh tay robot

+ Phần chân đế:

Được làm bằng hộp gỗ với độ dày là 10mm, chiều cao là 100mm với kết cấu cứng cáp và chân đế này cũng là nơi ta để bộ điều khiển (Hình 2.2)



Hình 2.2: Kết cấu phần chân đế của robot

+ Phần thân robot:

Phần thân là một hình trụ tròn có đường kính 100mm, bề dày là 3mm, chiều cao 300mm (Hình 2.3)



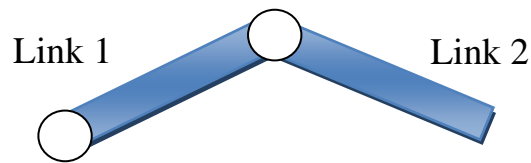
Hình 2.3: Kết cấu phần thân robot

+ Phần cánh tay:

Được cấu tạo bởi những tấm mika có độ dày 10mm, chiều dài Link 1 là 10mm, chiều dài Link 2 là 95mm. Được gia công bằng tia lazer. Với kết cấu nhẹ nhàng và đẹp mắt nhưng vẫn đảm bảo được quá trình làm việc cho cánh tay robot khi robot gấp vật (Hình 2.4)

Cánh tay robot với 3 khớp chuyển động, gồm 2 khớp quay và 1 khớp tịnh tiến, trong đó 2 khớp quay được gắn 2 động cơ DC và gắn encoder, khớp tịnh tiến được điều khiển bởi hệ thống xilanh 9.

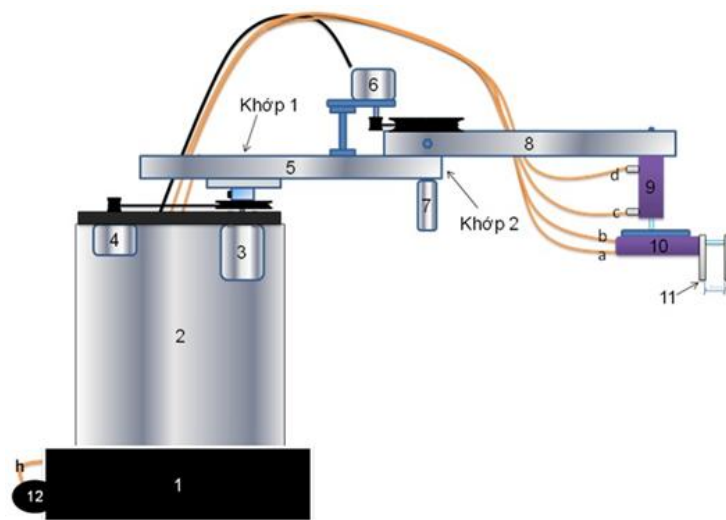
Bàn kẹp 11 được điều khiển bởi xilanh 10. Với kết cấu là khí nén nên đảm bảo khi gấp vật thì sẽ chắc chắn và độ ổn định cao.



Hình 2.4: Kết cấu phần cánh tay của robot

+ Tổng thể của cánh tay robot

Sau khi thiết kế riêng lẻ từng bộ phận của robot ta lắp ráp được 1 cánh tay robot hoàn thiện như sau (Hình 2.5)



Hình 2.5: Kết cấu cơ khí của cánh tay robot

+ Chú thích:

1. Chân tay máy
2. Thân tay máy
3. Động cơ dẫn động cho khớp 1
4. Encoder phản hồi vị trí cho động cơ 3
5. Link 1
6. Encoder phản hồi vị trí cho động cơ 7
7. Động cơ dẫn động cho khớp 2
8. Link 2
9. Xilanh tịnh tiến

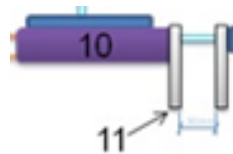
10. Xilanh kẹp

11. Tay kẹp

2.1.2. Cấu tạo của tay kẹp

Phần công tác của robot rất đa dạng. Trên các robot chuyên dùng thì phần công tác cũng là thiết bị chuyên dùng. Ở đề tài này thì cái với tay kẹp thì không đòi hỏi quá cao về cơ cấu chuyển động nhưng cũng cần phải hoạt động và làm việc đáng tin cậy: bắt đúng đối tượng, giữ chắc, nhưng không làm hỏng đối tượng. Ngoài ra, còn gọn nhẹ, tác động nhanh.

Với yêu cầu công nghệ như trên thì tay kẹp được chế tạo đơn giản nhằm mục đích giữ và nắm chặt đối tượng cần di chuyển (Hình 2.6).



Hình 2.6: Kết cấu cơ khí của tay kẹp

Cấu tạo của tay kẹp: Với 2 bản kẹp là bằng mika (11). Chuyển động được dẫn động bởi xilanh 10. Dẫn động bằng khí nén, nên chuyển nhanh gọn và chính xác.

2.1.3. Truyền động khí nén

Truyền động khí nén là loại có giá thành thấp nhất, thường dùng cho các thao tác lắp đặt chi tiết trên dây truyền lắp ráp. Đặc điểm của loại này là trang bị đơn giản và dễ điều khiển. Việc thiết kế và lắp đặt loại này đơn giản nhanh gọn, chuyển động độc lập được thực hiện bởi các xilanh riêng lẻ (Hình 2.7) hay bởi các mô – đun khí nén chuyên dùng. Điểm lưu ý ở loại này là độ chính xác vị trí lặp lại thấp.

Trong hầu hết các ứng dụng cơ cấu tác động khí nén hoạt động với 2 vị trí hơn là sử dụng hồi tiếp để đạt chế độ điều khiển tỉ lệ bởi vì do tính chất của khí nén, khi điều khiển ở cuối hành trình thì độ chính xác vị trí cũng như độ lặp khá tốt nhưng khoảng giữa hành trình thì rất khó đạt được.

Đề tài ở đây dùng loại hệ thống khí nén có áp suất là 9 Mpa luôn được giữ ổn định trong quá trình làm việc của robot.



Hình 2.7: Hệ thống khí nén

Ưu điểm:

- Giá thành không cao
- Khí thải không gây ô nhiễm môi trường
- Nguồn khí nén phổ biến trong công nghiệp
- Thích hợp cho các thiết kế dạng robot dạng mô – đun
- Cơ cấu tác động có thể dừng mà không hư hỏng

Nhược điểm:

- Áp suất khí nén giới hạn sự điều khiển và độ chính xác
- Khí xả gây ồn
- Khí bị rò rỉ gây trở ngại cho hệ thống
- Cần phải có bộ lọc làm khô nguồn khí nén
- Khó điều khiển tốc độ

2.1.4. Truyền động điện cơ

Với những công việc đòi hỏi chính xác, loại robot với truyền động điện tỏ ra đặc dụng nhất vì chúng cho phép bảo đảm được độ chính xác dịch chuyển cao và khả năng thực hiện những thao tác phức tạp (Hình 2.8).

Ưu điểm:

- Cơ cấu tác động nhanh và chính xác

- Có khả năng áp dụng kỹ thuật điều khiển phức tạp cho các chuyển động
- Giá thành không cao
- Thời gian triển khai hệ thống robot mới nhanh
- Nhiều động cơ có mô – men quay cao, trọng lượng giảm và thời gian đáp ứng

nhanh

Nhược điểm:

- Bản chất đã là tốc độ cao
- Khe hở bộ truyền bánh răng làm giảm độ chính xác
- Gây quá nhiệt hệ thống bị dừng hoạt động khi quá tải
- Cần phải có thắng để ghim các vị trí các khớp

+ Nhóm thứ nhất:

Dùng động cơ bước (stepper motor) để thực hiện những chuyển dịch góc chính xác dưới tác dụng của các xung điện áp. Dịch chuyển góc của robot của các động cơ bước đạt được độ chính xác cao nếu mômen tải trọng không vượt quá mômen giới hạn của động cơ.

+ Nhóm thứ hai:

Dùng động cơ phụ trợ với nguồn điện DC. Trong trường hợp này nhất thiết phải có liên hệ ngược giữa nguồn dẫn động (động cơ) với phần dịch chuyển (chấp hành) của robot.



Hình 2.8: Động cơ DC

2.2. Bộ điều khiển cho cánh tay robot ba bậc tự do

2.2.1. Mở đầu

Trong phần này tác giả sẽ đi sâu nghiên cứu vào bộ điều khiển cánh tay robot 3 bậc tự do. Bộ điều khiển robot thường cấu thành từ các bộ phận cơ bản tương tự như máy tính, gồm bộ vi xử lý, bộ nhớ và bộ xuất nhập, kết hợp với máy tính để thể hiện

các lệnh khi lập trình và đồng thời theo dõi sự thay đổi tọa độ trong dịch chuyển của các khâu, được sắp xếp theo từng mô-đun gồm các bo mạch điện tử. Bên cạnh đó, để lập trình một cách thuận tiện cho robot, tác giả sử dụng phần mềm LabVIEW để có thể thay đổi vị trí gắp và nhả vật kết hợp với bộ mô phỏng 3D để tiện quan sát cánh tay robot. Bộ điều khiển là bộ phận thể hiện những đặc điểm kỹ thuật ưu việt của robot, vì vậy ở đây sẽ trình bày cụ thể hơn về từng bộ phận trong hệ thống điều khiển robot.

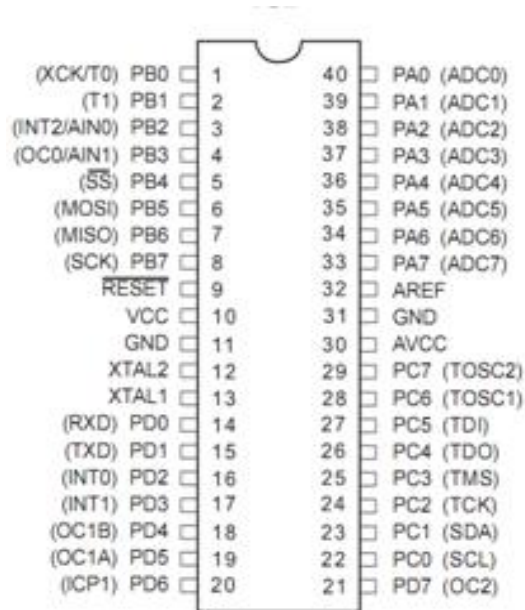
Chủ động thiết kế được bộ điều khiển do vậy có thể hoàn toàn chủ động trong lĩnh vực công nghệ. Bộ điều khiển có thể kết nối được với máy tính thông qua cổng USB. Tạo sự linh động trong quá trình thao tác cũng như vận hành và có thể dễ dàng cài đặt lại các thông số cho robot. Sau đây ta đi tìm hiểu về các đối tượng trong bộ điều khiển.

2.2.2. Mạch điều khiển

+ Vi điều khiển

Khối vi điều khiển được viết trên dòng vi điều khiển ATmega (ATmega 16). Dòng vi điều khiển này là loại 40 chân. Dòng vi điều khiển này thuộc họ vi điều khiển AVR. Đặc điểm của họ vi điều khiển AVR như sau:

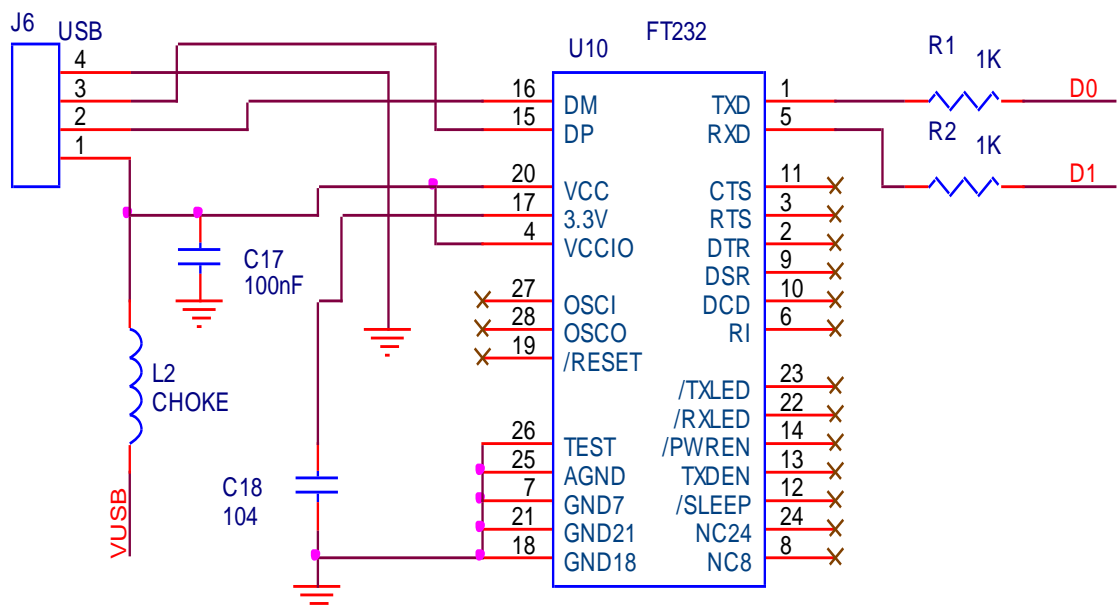
- AVR do hãng Atmel chế tạo ra, là họ vi điều khiển 8bit theo công nghệ mới với những tính năng rất mạnh được tích hợp trong trong chip.
- AVR ổn định hơn rất nhiều so với dòng vi điều khiển 8051
- Tính năng mới của họ AVR
 - Giao diện SPI đồng bộ, giao tiếp I2C, USART.
 - Các lối vào ra lập trình được.
 - Tích hợp bộ biến đổi ADC 10bit.
 - Có sẵn các kênh băm xung PWM.
 - Bộ Timer/Counter 8bit và 16bit rất mạnh.
 - Bộ định thời Watchdog.
 - Bộ nhớ EEPROM.



Hình 2.9: Vi điều khiển ATmega 16

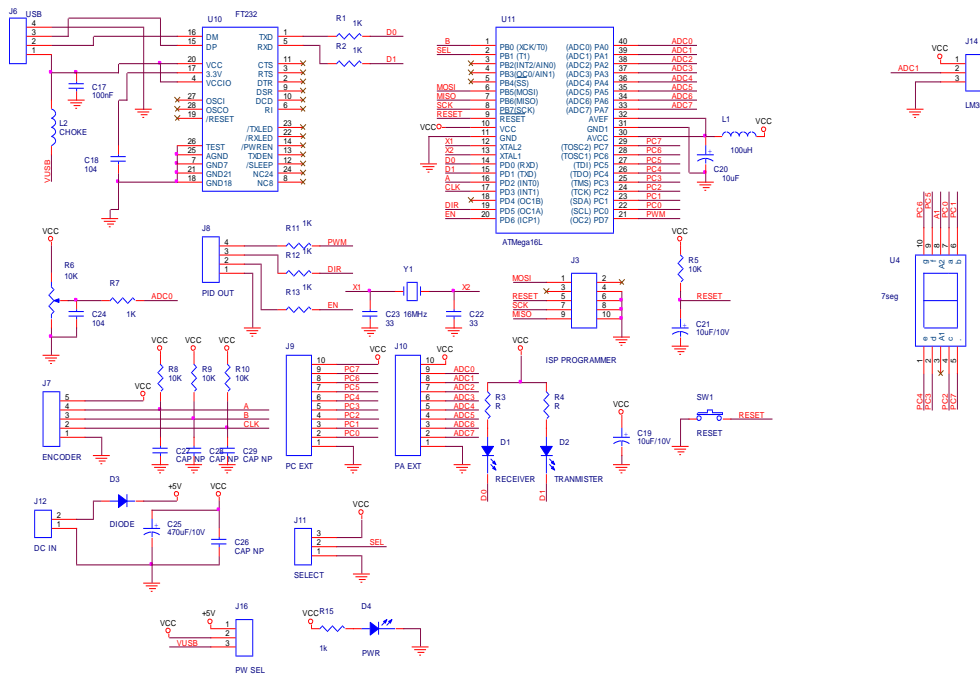
+ Sơ đồ FT232:

- Mạch đầu vào là cổng USB được nối với các 4,15,16,17,20 các chân là đầu vào của FT232. Trong đó có 2 chân tín hiệu DM và DP là 2 chân truyền dữ liệu từ USB.
- Mạch đầu ra theo chuẩn RS232 bao gồm 2 chân TXD, RXD
- Sau khi cài đặt thì máy tính sẽ hiểu đây là một COM ảo



Hình 2.10: Sơ đồ mạch FT232

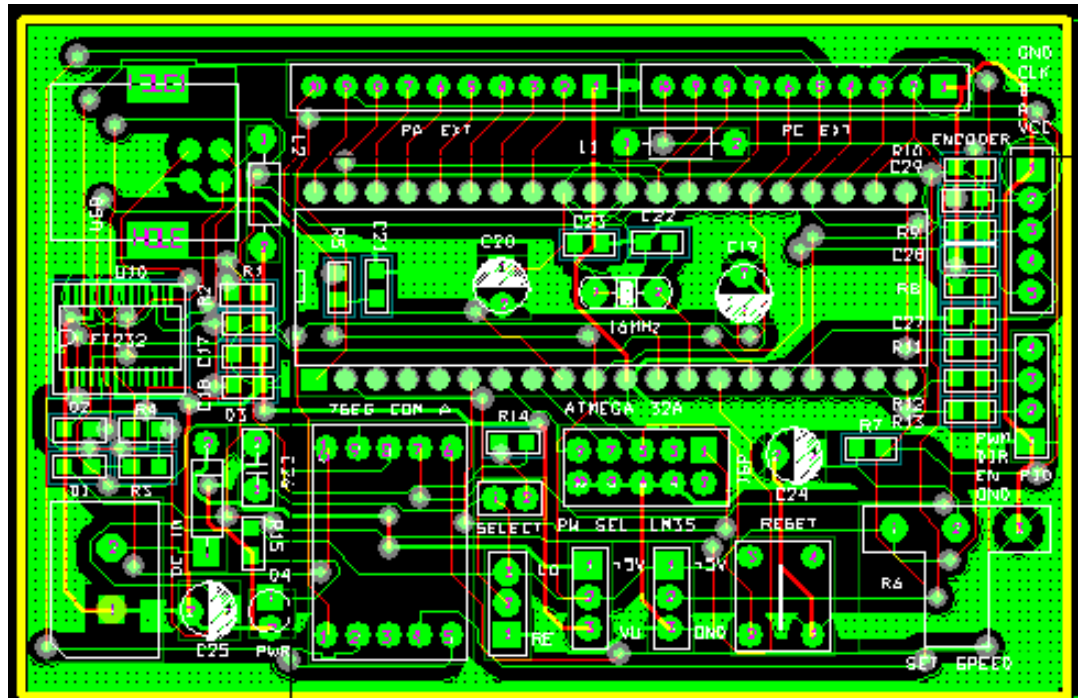
+ Sơ đồ nguyên lý:



| | | | |
|------|----------------------------|-------|------------|
| File | <Title> | Rev | <Rev Code> |
| Size | Document Number | | |
| B | <Doc> | | |
| Date | Thursday, October 04, 2012 | Sheet | 1 of 1 |

Hình 2.11: Sơ đồ mạch nguyên lý của bộ điều khiển

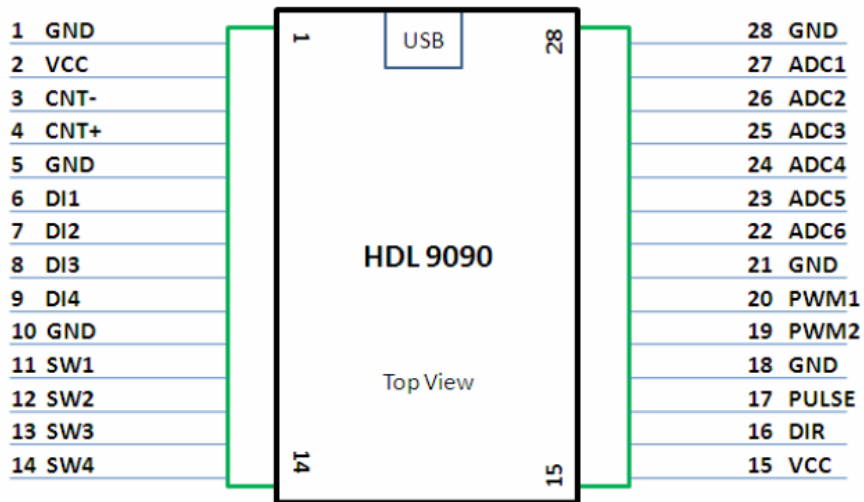
+ Sơ đồ mạch in:



Hình 2.12: Sơ đồ mạch in

+ Card HDL 9090

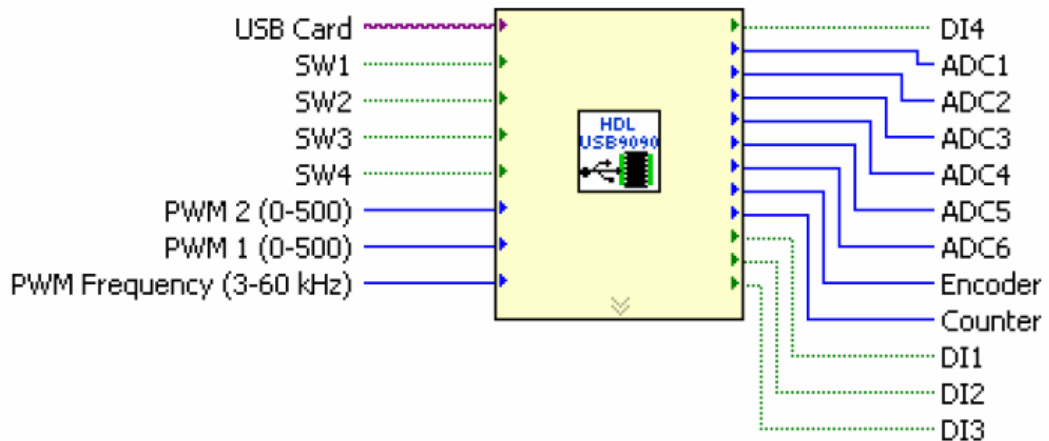
Sử dụng thêm card HDL 9090 để làm bộ điều khiển cho cánh tay robot.



Hình 2.13: Card 9090

Bảng 2.1: Bảng mô tả chức năng chân của Card 9090

| Ký hiệu | Mô tả | Loại |
|---------------|--|--------|
| GND | Ground – chân mát | Nguồn |
| VCC | Nguồn 5V lấy từ USB | Nguồn |
| CNT+/CNT- | Chân đếm Counter | Input |
| DI1-DI4 | Tín hiệu vào dạng số 0-5V | Input |
| SW1-SW4 | Tín hiệu ra dạng số 0-5V | Output |
| DIR | Set bộ đếm xung đếm lên(5V) hay đếm xuống (0V) | Input |
| PULSE | Đếm xung tín hiệu 0-5V | Input |
| PWM1- PWM2 | Xuất tín hiệu PWM để điều khiển | Output |
| ADC1- ADC6 | Nhận tín hiệu vào tương tự (Analog) 0-5V | Input |



Hình 2.14: Thư viện của Card 9090

Bảng 2.2: Bảng mô tả các đầu vào, đầu ra của thư viện

| Chân | Loại | Mô tả |
|-------------------------|-----------|---|
| USB Card | Control | Tạo control tại chân này để chọn thiết bị USB HDL 9090 |
| SW1-SW4 | Control | Nối giá trị Boolean (TRUE-FALSE) vào các chân này để phần cứng HDL 9090 xuất ra tín hiệu số tương ứng (TRUE 5V, FALSE 0V) |
| PWM1-PWM2 | Control | Nối giá trị số nguyên 0-500 vào để phần cứng xuất xung PWM có duty cycle tương ứng là 0-100% |
| PWM Frequency (3-60kHz) | Control | Nối giá trị từ 3000-60000 vào để chọn tần số tín hiệu PWM card xuất ra. |
| DI1-DI4 | Indicator | Giá trị tín hiệu số đọc được từ chân DI1-DI4 trên card. Giá trị mặc định là TRUE (5V) |
| ADC1-ADC6 | Indicator | Giá trị ADC (0-1023) đọc được từ các chân ADC tương ứng trên card |
| Encoder | Indicator | Giá trị encoder 0-100000 |

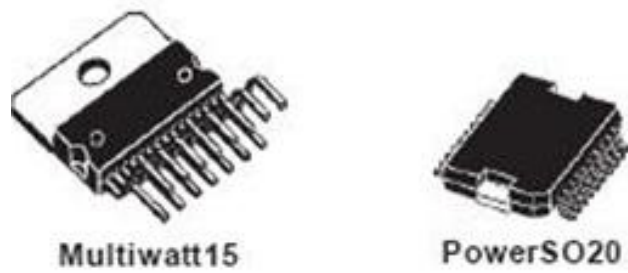
+ Diver DC:

Đây là mạch L298 dùng cho việc điều khiển động cơ DC, với cánh tay robot ta sử dụng 2 mạch này. L298 với dòng làm việc là 2A dùng để điều khiển cho 2 động cơ, nhưng với mạch của cánh tay robot thì chỉ dùng cho 1 động cơ DC vì khả năng chịu dòng làm việc của L298 là 2A vậy nên nó được đấu song song. Mỗi một mạch điều khiển 1 động cơ DC.

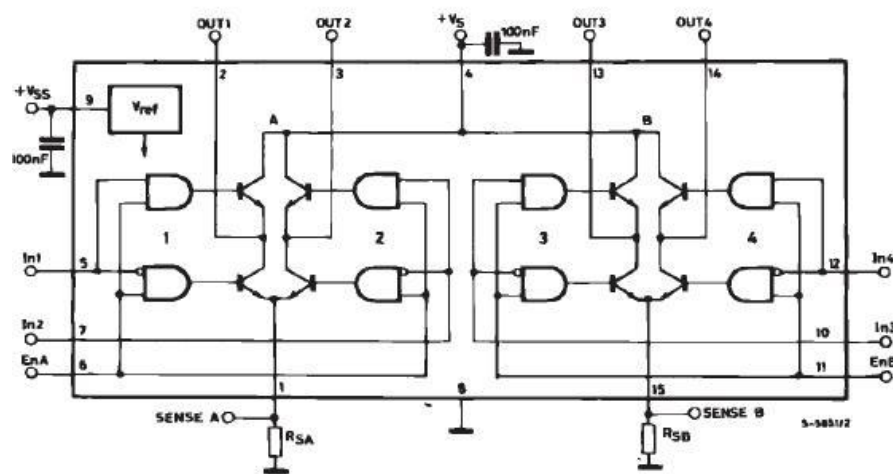
Hoạt động của L298:

Có 3 chân tín hiệu đầu vào là EnA, In1, In2. Hai chân tín hiệu ra là Out1, Out2 được nối với động cơ DC. Muốn động cơ quay thuận thì cho phép tín hiệu In1 (chân In1 được nối với chân của vi điều khiển). Chân EnA cũng được nối với chân của vi

điều khiển. Khi có 2 tín hiệu này thì động cơ sẽ quay thuận. Còn muốn quay ngược ta chỉ cần cho tín hiệu In2 (Hình 2.15, Hình 2.16, Hình 2.17).



Hình 2.15: L298



Hình 2.16: Sơ đồ khối L298



Hình 2.17: Driver L298

2.2.3. Sử dụng phần mềm LabVIEW để viết giao diện điều khiển

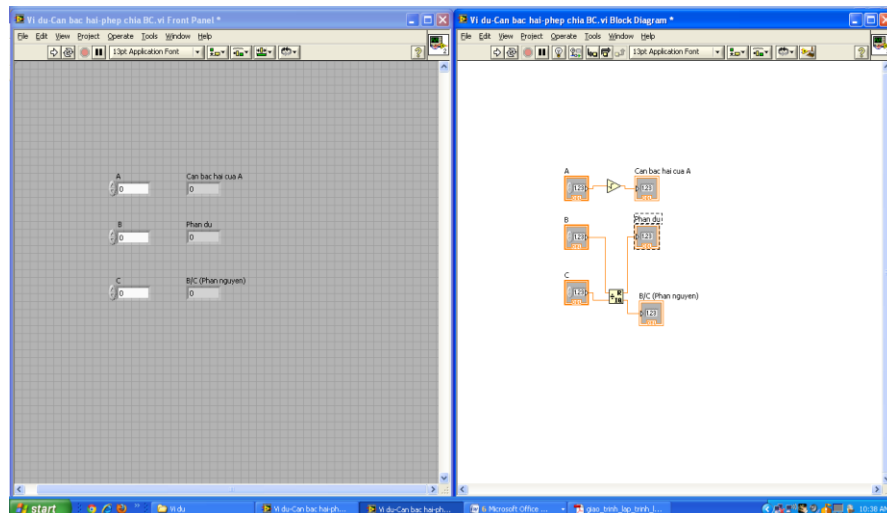
Nguyên cứu về LabVIEW

LabVIEW viết tắt của Laboratory Virtual Instrumentation Engineering Workbench là một môi trường nền và phát triển cho một ngôn ngữ lập trình đồ họa của hãng National Instruments.

LabVIEW được phát triển từ năm 1986, ban đầu LabVIEW được sử dụng rộng rãi để nhận dữ liệu, điều khiển thiết bị và tự động hóa trong công nghiệp chạy trong nhiều nền hệ điều hành khác nhau như Windows, Unix, Linux và Mac OS.

LabVIEW ngày càng được sử dụng rộng rãi trong đo lường công nghiệp, thí nghiệm và giáo dục cũng như trong các ứng dụng tự động hóa dựa trên cơ sở lập trình đồ họa (graphical programming). Khác với lập trình văn bản (textual programming), lập trình đồ họa trực giác hơn. Tuy nhiên LabVIEW vẫn hỗ trợ lập trình văn bản. Trong LabVIEW có rất nhiều hàm dùng để phân tích, thiết kế và biểu diễn dữ liệu bằng đồ thị và các dụng cụ đo lường ảo phong phú.

Ngày nay, LabVIEW có nhiều công cụ và các môđun khác nhau làm cho LabVIEW có chức năng tính toán mạnh trong lập trình phân tích, thiết kế hệ thống điều khiển, xử lý số liệu, nhận dạng hệ thống, toán học, mô phỏng, và nhiều chức năng khác. Ngoài ra, LabVIEW còn hỗ trợ rất nhiều phần cứng như bảng giao diện dữ liệu vào ra (bảng nhận dữ liệu, DAQ, data acquisition), bus truyền thông dữ liệu CAN dùng trong đo lường và tự động hóa; các bộ điều khiển mềm (drivers) và hỗ trợ các tiêu chuẩn giao tiếp như CAN bus, OPC, Modbus, GPIB, ... (Hình 2.18)



Hình 2.18: Mã nguồn của LabVIEW

LabVIEW được sử dụng tại hơn 30.000 ngàn công ty và hàng ngàn phòng thí nghiệm trên thế giới. Hầu hết các phòng thí nghiệm ở các châu lục đặc biệt là tại châu Mỹ, Hàn quốc, Nhật bản, Trung quốc. Các viện nghiên cứu cao cấp như máy giao tốc lớn nhất thế giới, thu thập và truyền nhận dữ liệu tại cơ quan vũ trụ NASA. Các công ty về đo lường, kiểm định, xử lý tín hiệu, và điều khiển công nghiệp ví dụ, các dây truyền kiểm tra lỗi sản phẩm tại Hàn quốc, các hệ thống thu thập và điều khiển công nghiệp trong ngành tàu biển, giao thông vận tải tại Nhật Bản, các công ty, nhà máy kiểm tra điện thoại tại Singapore...

Tại Việt Nam hiện nay, nhiều công ty đã sử dụng phần mềm này để điều khiển các dây chuyền tự động mà hầu hết được nhập khẩu từ nước ngoài. Tuy nhiên, các nghiên cứu về ứng dụng của LabView trong giảng dạy thì chưa nhiều. Nghiên cứu các ứng dụng của phần mềm LabView để đưa vào giảng dạy sẽ giúp sinh viên tiếp cận, nắm bắt được phần mềm mới và sử dụng trong thực tiễn sản xuất khi ra ngoài làm việc.

Bên cạnh đó, LabView là ngôn ngữ lưu đồ họa, diễn đạt cú pháp thông qua các hình ảnh trực quan trong môi trường soạn thảo giúp người lập trình có phương pháp trực quan hơn trong việc tự động hóa các hệ thống đo lường và điều khiển. Điều này giúp giảng viên trình bày cho sinh viên một cách trực quan, sinh động, dễ hiểu.

Ngôn ngữ lưu đồ kết hợp với khối I/O gắn liền và điều khiển giao diện người sử dụng tương tác cùng đèn chỉ báo làm cho LabVIEW trở thành một sự lựa chọn lí tưởng.

Ứng dụng của LabVIEW

LabVIEW được ứng dụng trong các lĩnh vực đo lường, tự động hóa, cơ điện tử, robot, vật lý, toán học, công nghệ sinh học vv...

- LabVIEW giúp kỹ sư kết nối bất kỳ với cảm biến nào, bất kỳ cơ cấu chấp hành nào với máy tính.
- LabVIEW có thể xử lý các dữ liệu như các tín hiệu tương tự (analog), tín hiệu số (digital), âm thanh (audio).
- LabVIEW hỗ trợ các giao thức giao tiếp khác nhau như RS232, RS485, TCP/IP, PCI, PXI.

- Bạn cũng có thể tạo ra các thực thi độc lập và các thư viện chia sẻ (ví dụ như thư viện liên kết động DLL), bởi vì LabVIEW là một trình biên dịch 32bit (Hình 2.19 và Hình 2.20).



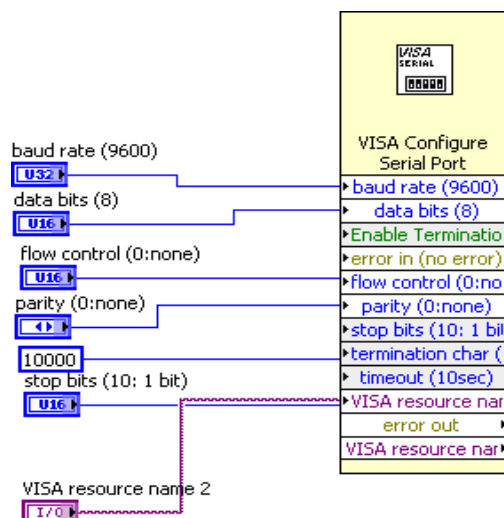
Hình 2.19: Thu thập dữ liệu tại Cơ quan hàng không vũ trụ - NASA



Hình 2.20: Điều khiển động cơ theo thuật toán PID

Truyền thông trên LabVIEW

- Khối VISA Configure Serial Port (Hình 2.21)



Hình 2.21: Mô tả chân của khối VISA Configure Serial Port

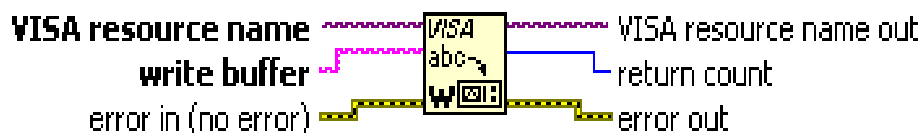
Kết nối với mạch qua chuẩn nối tiếp RS232. Với sơ đồ khối VISA Configure Serial Port, với các tín hiệu như sau (Bảng 2.3).

Bảng 2.3: Bảng mô tả tên chân của khối VISA Configure Serial Port

| Tên chân | Chức năng |
|------------------------|--|
| VISA resource name 2 | Chọn thông số cổng Com |
| Baud rate (9600) | Chọn tốc độ baud. Mặc định là 9600 |
| Data bits (8) | Số bit dữ liệu vào. Mặc định là 8 bit |
| Parity (0:none) | Bit chẵn lẻ - dùng khi được truyền - nhận. Mặc định là 0 |
| Error in | Thông báo lỗi trước khi chức năng này hoạt động. Mặc định là No Error |
| Stop bits (1.0) | Một bit Stop |
| Flow control | Điều khiển lưu lượng truyền dữ liệu. Mặc định là 0 |
| Millisecond multiple | Khoảng thời gian truyền thông |
| VISA resource name out | Là toàn bộ giá trị của VISA resource name |
| Error out | Nếu lỗi ngõ vào chứa thông tin lỗi trước khi VI chạy thì lỗi ngõ ra sẽ mang thông tin lỗi đó |

- Khối VISA Write Function (Hình 2.22)

VISA Write Function: Ghi dữ liệu từ Write Buffer tới thiết bị hay giao diện được chỉ bởi VISA resource name (Bảng 2.4).



Hình 2.22: Mô tả chân của khối VISA Write Function

Bảng 2.4: Bảng mô tả chức năng chân của khối VISA Write Function

| Tên chân | Chức năng |
|------------------------|---|
| VISA resource name | Chọn thông số cổng COM |
| Write buffer | Chứa dữ liệu sẽ được ghi tới thiết bị |
| Error in | Thông báo lỗi trước khi chức năng này hoạt động. Mặc định là No Error. |
| VISA resource name out | Là toàn bộ giá trị của VISA resource name |
| Return count | Chứa những byte thực sự được ghi ra |
| Error out | Nếu lỗi ngõ vào chứa thông tin lỗi trước khi VI chạy thì lỗi ngõ ra sẽ mang thông tin lỗi đó. |

Với hai khối hoạt động ở trên, chúng ta sẽ dùng hai khối này cho toàn bộ các bài thí nghiệm còn lại. Cách khai báo hoàn toàn giống nhau.

+ Viết chương trình trên nền LabVIEW

Chương trình của LabVIEW được chia làm 2 phần:

Viết trên Front Panel và Block Diagram

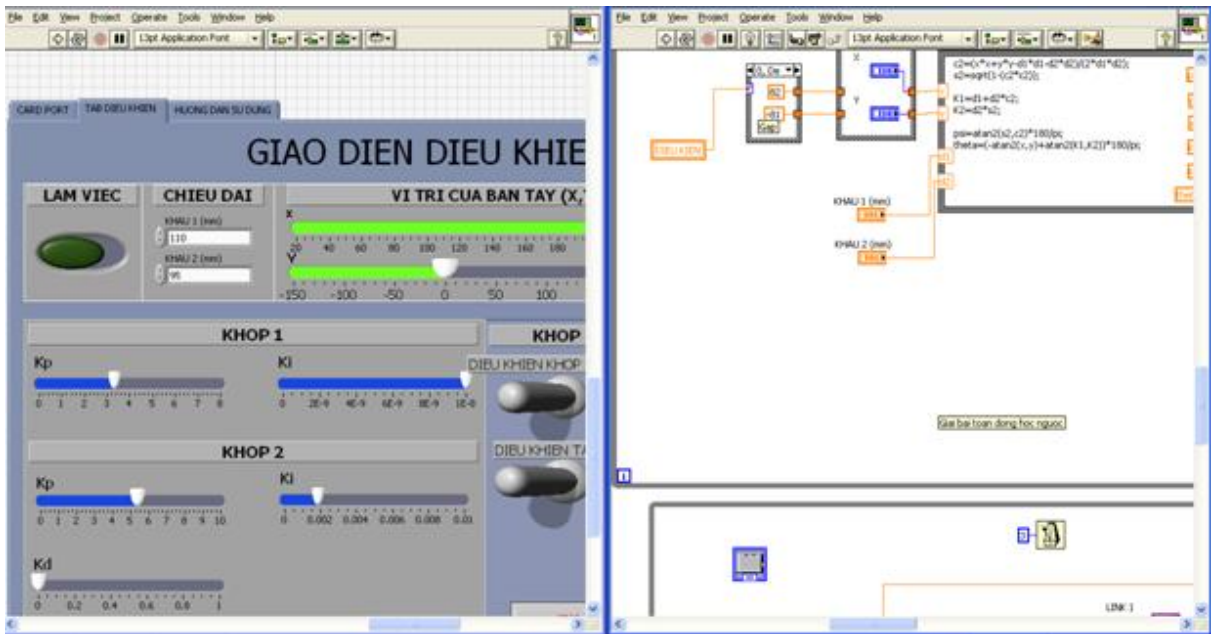
Phần Front Panel là phần giao diện thực thi chương trình điều khiển, tạo cách lệnh điều khiển đối tượng của mình, là phần Control để điều khiển cánh tay.

- Control là các đối tượng đặt trên Front Panel để cung cấp dữ liệu cho chương trình. Nó tương tự như đầu vào cung cấp dữ liệu.

- Indicator là đối tượng được đặt trên Front Panel dùng để hiển thị kết quả nó tương tự như một đầu vào của chương trình.

Phần Block Diagram là phần sơ đồ khối, tạo các khối chương trình lỗi để điều khiển (Hình 2.23). Block Diagram là phần để viết thuật giải cho robot, các phương trình động học ngược, thuật giải PID cho các động cơ và cách mô phỏng 3D cho cánh tay robot.

- Terminal: là các cổng dữ liệu truyền qua giữa Block Diagram và Front Panel và giữa các Node trong Block Diagram. Các Terminal nằm ở dưới dạng các Icon của các Function.
- Node: là các phần tử thực thi chương trình, chung tương tự các mệnh đề, toán tử hàm và các chương trình con trong các ngôn ngữ lập trình thông thường.
- Wises: là các dây nối dữ liệu giữ các node

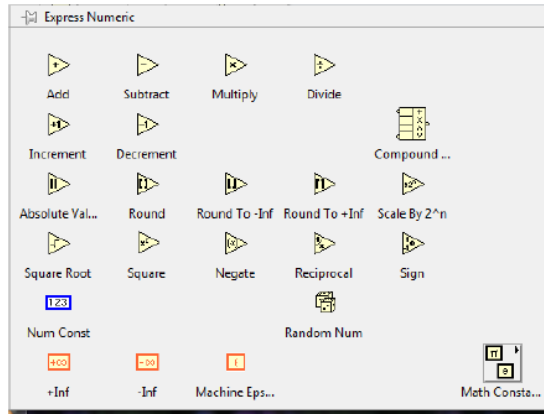


Hình 2.23: Chương trình điều khiển Font Panel và Block Diagram

+ Giao diện điều khiển bằng máy tính

Các khối được sử dụng trong chương trình

- Các kí hiệu toán học: Functions/Express/Arithmetic & Comparson/ Numeric (Hình 2.24)



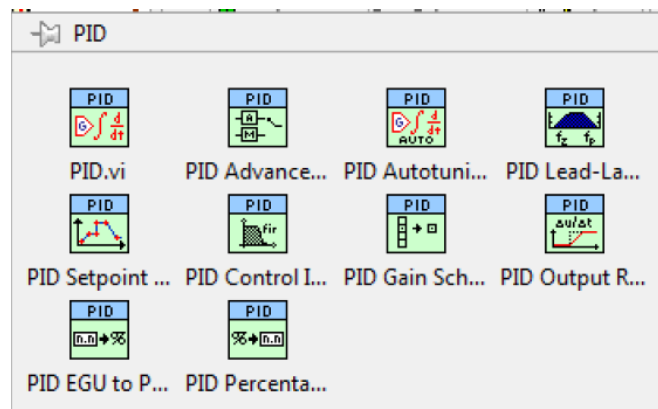
Hình 2.24: Khối ký hiệu toán học

- Các toán tử so sánh (Hình 2.25)



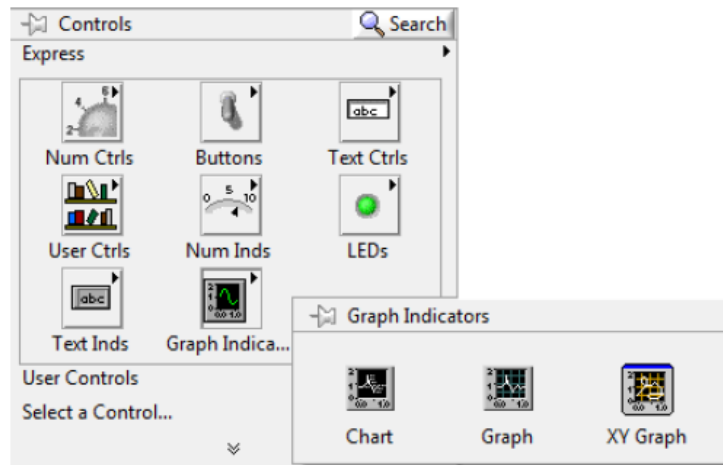
Hình 2.25: Khối toán tử so sánh

- Khối PID: control design & Simulation / PID (Hình 2.26)



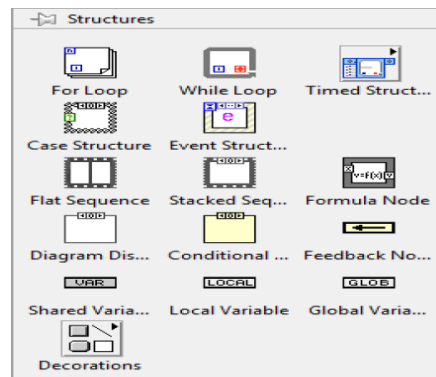
Hình 2.26: Khối PID

- Khối biểu diễn đồ thị: Controls /Grap Indicators (Hình 2.27).



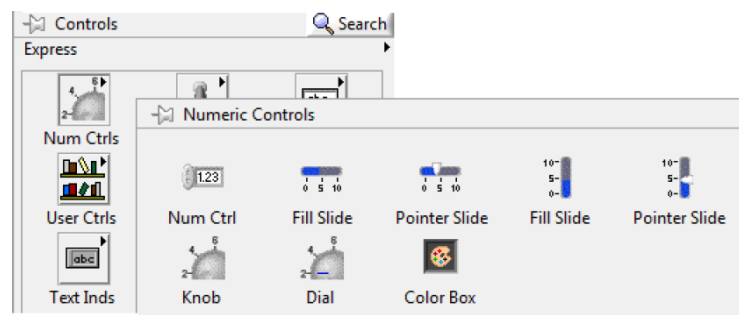
Hình 2.27: Khởi biểu diễn đồ thị

- Các hàm: Functions/ Programming/Structures (Hình 2.28)



Hình 2.28: Khởi các hàm

- Các biến: Controls/Numeric controls (Hình 2.29)



Hình 2.29: Khởi các biến

- Định dạng kiểu dữ liệu cho các biến: Nhấp phải chuột vào biến numeric control/Representation/chọn kiểu dữ liệu (Hình 2.30)



Hình 2.30: Khối các kiểu dữ liệu

+ Thuật giải và code điều khiển

Dựa vào mạch giao tiếp và mạch công suất để điều khiển động cơ đã được trình bày ở các phần trên, ta thiết kế được code (Block Diagram) như các hình sau đây.

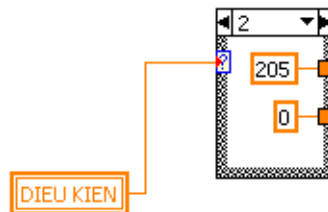
Yêu cầu đặt ra là tạo được phương trình động học ngược để điều khiển cánh tay robot. Dùng các hàm toán học để xây dựng thuật toán để điều khiển vị trí của cánh tay.

- Khối thiết lập tọa độ làm việc cho cánh tay robot

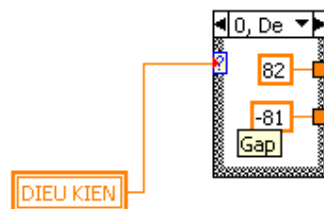
Hình 2.31 (a): Thiết lập tọa độ ban đầu cho cánh tay robot, nó có tọa độ là $O(205,0)$ với cấu trúc Case Structure.

Hình 2.31 (b): Thiết lập tọa độ gắp cho cánh tay robot, nó có tọa độ là $A(82,-81)$ cũng được viết trên cấu trúc Case Structure.

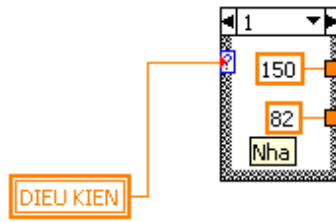
Hình 2.31 (c): Thiết lập tọa độ nhả cho cánh tay robot, nó có tọa độ $B(150,82)$ cũng được viết trên cấu trúc Case Structure.



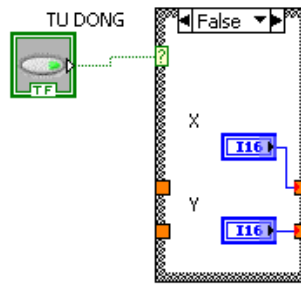
Hình 2.31 (a): Khai báo tọa độ ban đầu cho cánh tay robot



Hình 2.31 (b): Khai báo tọa độ gắp cho cánh tay robot

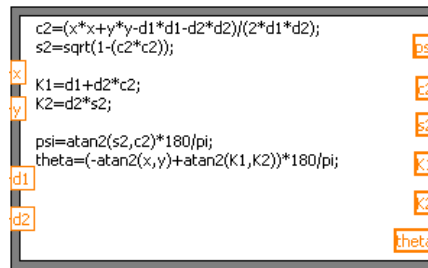


Hình 2.31 (c): Khai báo tọa độ nhà cho cánh tay robot



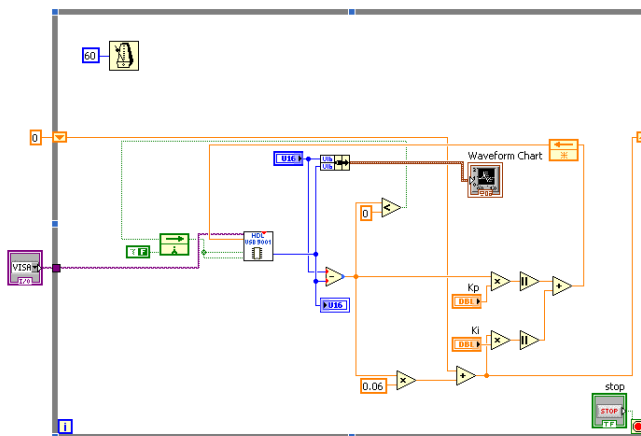
Hình 2.32: Khối xử lý các tọa độ của cánh tay robot

- Khối Case Structure dùng cho việc viết phương trình động học ngược cho robot (Hình 2.33) theo công thức 2.40 và 2.41 ở Chương 2.



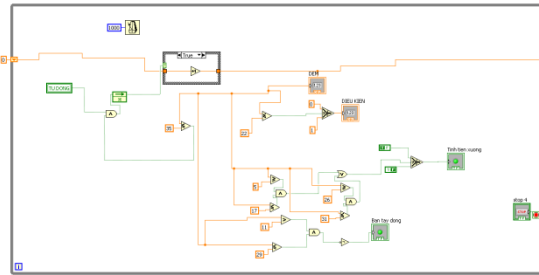
Hình 2.33: Khối xử lý bài toán động học ngược cho robot

- Khối Case Structure dùng cho việc viết thuật toán PID cho robot (Hình 2.34).



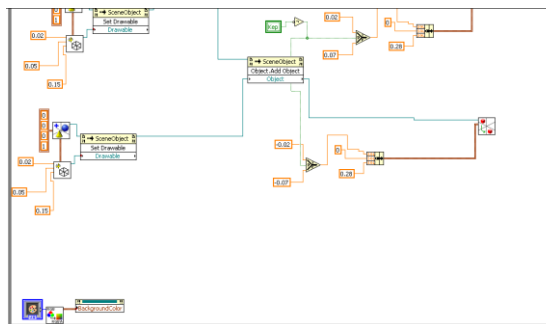
Hình 2.34: Điều khiển vị trí theo thuật toán PID

- Khối Case Structure dùng cho việc tạo delay (Hình 2.35)



Hình 2.35: Khối tạo delay cho cánh tay robot

- Khối Case Structure dùng cho việc tạo khối mô phỏng 3D cho robot (Hình 2.36)



Hình 2.36: Khối mô phỏng 3D cho robot

+ Code điều khiển cho vi điều khiển AVR (Hình 2.37)

```

Notes robot.c
22  *****/
23
24  #include <megal6.h>
25
26  // Declare your global variables here
27
28  void main(void)
29  {
30  // Declare your local variables here
31
32  // Input/Output Ports initialization
33  // Port A initialization
34  // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
35  // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
36  PORTA=0x00;
37  DDRA=0x00;
38
39  // Port B initialization
40  // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
41  // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
42  PORTB=0x00;
43  DDRB=0x00;
44
45  // Port C initialization
46  // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
47  // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

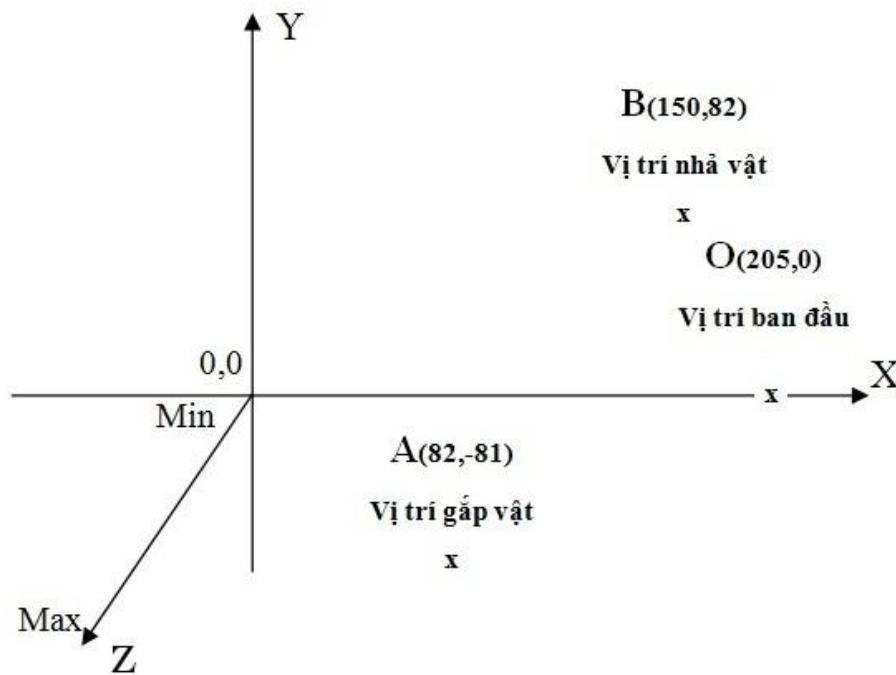
```

Hình 2.37: Code điều khiển cho vi điều khiển AVR

2.3. Hoạt động của cánh tay robot

Hoạt động của robot bám theo 3 trục đo là x,y,z. Điểm O có tọa độ ban đầu là O(205,0), từ vị trí điểm O này ta đi đến tọa độ điểm A có tọa độ là A(82,-81), đó là vị

trí gấp vật. Từ vị trí A ta đi đến vị trí B có tọa độ B(150,82) đó là vị trí nhà vật. Chu trình sẽ lặp đi lặp lại khi để ở chế tự động.



Hình 2.38: Tọa độ hoạt động của cánh tay robot

Mô hình có thể hoạt động ở 2 chế độ:

- Chế độ điều khiển bằng tay
- Chế độ điều khiển tự động

2.3.1. Chế độ bằng tay

Từ máy tính chọn các chế độ khác nhau để robot thực hiện các nhiệm vụ ở phần NHIỆM VỤ. Nó có 3 chế độ đó là (Quay về vị trí ban đầu, Di chuyển đến vị trí gấp vật, Di chuyển đến vị trí nhà vật) (Hình 2.39).

Modul 1: Đọc hướng dẫn sử dụng cho cánh tay robot

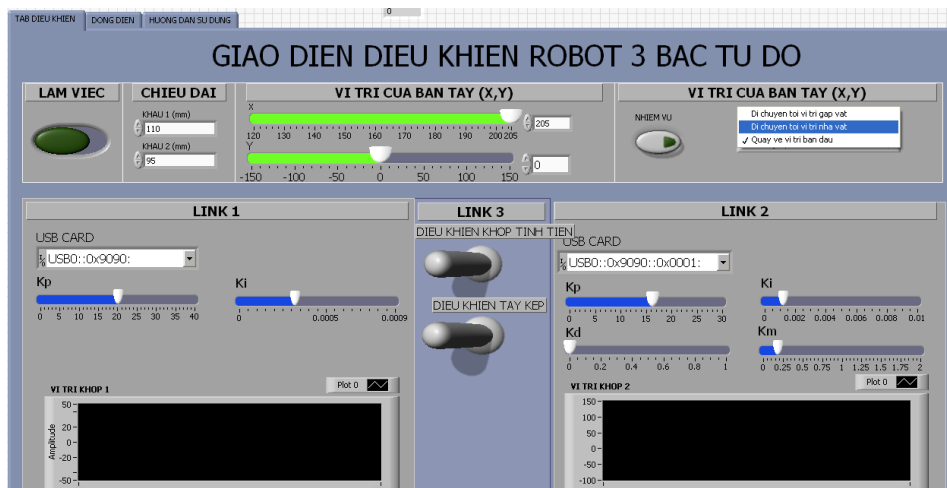
Modul 2: Chọn cổng USB

Modul 3: TAB điều khiển

Chương trình điều khiển cho Font Panel như sau:

- CARD PORT: phần chọn cổng để kết nối giữa máy tính và cánh tay robot. Phải chọn hai cổng cho Link 1 và Link 2. Link 1: cho khớp 1; Link 2: cho khớp 2.

Phần giao thức này cần phải chọn trước khi làm việc với robot.



Hình 2.39: Giao diện điều khiển robot ở chế độ bằng tay từ máy tính

2.3.2. Chế độ tự động

Modul 1: Đọc hướng dẫn sử dụng cho cánh tay robot

Modul 2: Chọn cổng USB

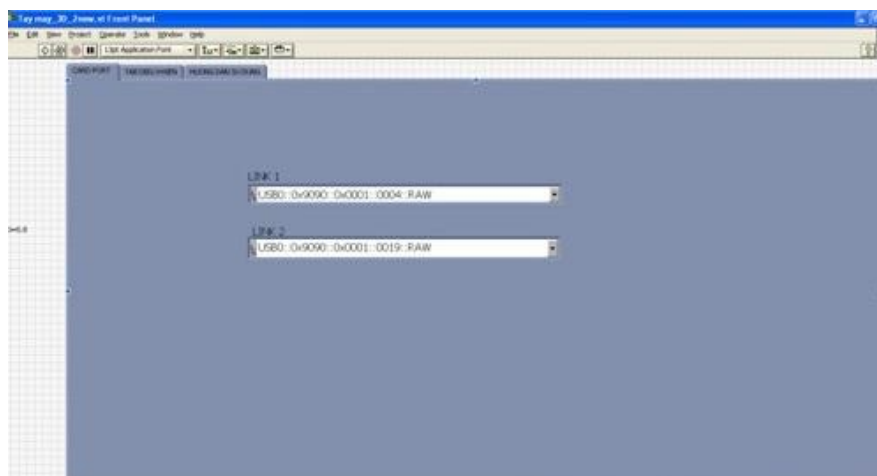
Modul 3: TAB điều khiển

Chọn nút **LAM VIEC** sau đó chọn chế độ **TU DONG** (Hình 2.40)

Chương trình điều khiển cho Font Panel như sau:

- **CARD PORT:** phần chọn cổng để kết nối giữa máy tính và cánh tay robot. Phải chọn hai cổng cho Link 1 và Link 2. Link 1: cho khớp 1; Link 2: cho khớp 2.

Phần giao thức này cần phải chọn trước khi làm việc với robot.



Hình 2.40: Phần chọn cổng để kết nối giữa máy tính và robot

- **TAB DIEU KHIEN:** được chia làm bốn phần:



Hình 2.41: Giao diện điều khiển robot ở chế độ tự động từ máy tính

- Nút khởi động: sau khi chọn Card Port, di chuyển sang TAB DIEU KHIEN và chọn nút LAM VIEC để cánh tay robot ở vị trí ban đầu có tọa độ O (Hình 2.41, Hình 2.42).
- Chọn các thông số K_i , K_p , K_d
- Công tắc điều khiển khớp tịnh tiến và tay kẹp
- Mô phỏng 3D



Hình 2.42: Giao diện điều khiển tự động

Sau đây là một số hình ảnh thực tế khi làm việc trên mô hình thật (Hình 2.43, Hình 2.44)

+ Một số hình ảnh hoạt động của cánh tay robot ngoài thực tế:



Hình 2.43: Vị trí gấp vật



Hình 2.44: Vị trí gấp nhà vật

KẾT LUẬN VÀ KIẾN NGHỊ

Trong quá trình thực hiện đề tài, tác giả luôn luôn nỗ lực và quyết tâm làm cho hoàn thiện nhất đề tài của mình. Đề tài đã đạt được một số yêu cầu như sau:

- Điều khiển bằng động cơ DC servo
- Tự thiết kế bộ điều khiển cho cánh tay robot
- Viết giao diện trên nền tảng LabVIEW
- Có mô phỏng cánh tay 3D trên LabVIEW
- Đề tài đã trở thành mô hình thí nghiệm của trường Đại học Dân lập Hải Phòng.

Có tính trực quan cho sinh viên, giảng viên để phục vụ cho công tác giảng dạy.

Tuy nhiên, do thời gian làm đề tài này có hạn và kinh phí còn quá eo hẹp nên đề tài còn có một số điểm hạn chế sau:

- Lý thuyết về điều khiển thông minh chưa thực hiện được
- Số bậc tự do còn thấp
- Mô hình cơ khí còn đơn giản

Kiến nghị

Mô hình cánh tay robot 3 bậc tự do được tác giả nghiên cứu, chế tạo, điều khiển có thể ứng dụng trong sản xuất công nghiệp. Trong thời gian tới, tác giả cũng sẽ phát triển đề tài, khắc phục những điểm còn hạn chế để hướng tới phục vụ sản xuất khi thời gian và kinh phí dồi dào hơn.

Vì vậy, tác giả rất mong muốn trong thời gian không xa sẽ được tiếp tục nghiên cứu sâu hơn về robot, cũng như nghiên cứu triển khai ứng dụng đề tài này vào thực tế sản xuất.

TÀI LIỆU THAM KHẢO

1. TS Nguyễn Mạnh Giang. 2006. Kỹ thuật ghép nối máy vi tính. Hà Nội: NXB Giáo dục.
2. TS Nguyễn Bá Hải. 2011. Lập trình LabVIEW. TpHCM: Trường Đại học Sư phạm kỹ thuật Tp.HCM.
3. PGS.TS Đào Văn Hiệp. 2006. Kỹ thuật robot. Hà Nội: NXB Khoa học và Kỹ thuật.
4. TS Nguyễn Văn Hòa. 2001. Cơ sở lý thuyết điều khiển tự động. Hà Nội: NXB Khoa học và Kỹ thuật.
5. GS.TSKH Thân Ngọc Hoàn, TS Nguyễn Tiến Ban. 2007. Điều khiển tự động các hệ thống truyền động điện. Hà Nội: NXB Khoa học và Kỹ thuật.
6. Bùi Quốc Khánh, Phạm Quốc Hải, Nguyễn Văn Liễn, Dương Văn Nghi. 1999. Điều chỉnh tự động truyền động điện. Hà Nội: NXB Khoa học và Kỹ thuật.
7. PGS.TS Tạ Duy Liêm. 2004. Robot và hệ thống công nghệ robot hóa. Hà Nội: NXB Khoa học và Kỹ thuật.
8. Quách Tuấn Ngọc. 2003. Ngôn ngữ lập trình C. Hà Nội: NXB Thống kê.
9. TS Lê Hoài Quốc, Chung Tấn Lâm. 2006. Nhập môn robot công nghiệp. Hà Nội: NXB Khoa học và Kỹ thuật.
10. Ngô Diên Tập. 2000. Kỹ thuật ghép nối máy tính. Hà Nội: NXB Khoa học và Kỹ thuật.
11. Ngô Diên Tập. 1999. Vi xử lý trong đo lường và điều khiển. Hà Nội: NXB Khoa học và Kỹ thuật.
12. Trần Quang Vinh. 2007. Nguyên lý phần cứng và kỹ thuật ghép nối máy tính. Hà Nội: NXB Giáo dục.

