

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC NÔNG LÂM THÀNH PHỐ HỒ CHÍ MINH

KHÓA LUẬN TỐT NGHIỆP

THIẾT KẾ, CHẾ TẠO VÀ ĐIỀU KHIỂN TAY MÁY
THEO KIỂU LẬP MẪU

Sinh viên thực hiện: **HOÀNG THANH VƯƠNG 17153080**

NGUYỄN TRỌNG TIẾN 17153070

Ngành : **Công nghệ kỹ thuật cơ điện tử**

Niên khóa: **2017-2021**

THIẾT KẾ, CHẾ TẠO VÀ ĐIỀU KHIỂN TAY MÁY THEO KIỂU LẬP MẪU

Tác giả

HOÀNG THANH VƯƠNG 17153080

NGUYỄN TRỌNG TIẾN 17153070

Khóa luận được đệ trình để đáp ứng yêu cầu cấp bằng

Cử nhân ngành công nghệ kỹ thuật cơ điện tử

Giáo viên hướng dẫn:

Thạc sĩ: Nguyễn Đăng Khoa

Tháng 8 năm 2021

LỜI MỞ ĐẦU

Lời cảm ơn tới ban tổ chức khoa cơ khí công nghệ ĐH Nông Lâm đã cho chúng em cơ hội được tham gia vào đề tài khóa luận tốt nghiệp hết sức đặc biệt này.

Cảm ơn những đóng góp ý kiến, chỉ điểm từ giảng viên phụ trách đề tài của chúng em, Ths. Nguyễn Đăng Khoa. Cùng với khát vọng được trình bày, phản biện một cách sôi nổi ở buổi bảo vệ luận án sắp tới.

Với những kiến thức, kinh nghiệm qua chương trình đào tạo của nhà trường như nền tảng, tiếp thêm sự tự tin cho chúng em trong tương lai. Vì vậy chúng em xin trân trọng những giá trị về mặt tinh thần lẫn vật chất mà nhà trường đã cung cấp, hỗ trợ chúng em trong suốt chương trình học vừa qua.

Sự nhiệt huyết, chu đáo của các giảng viên không thể không được chúng em nhắc đến trong những dòng tri ân này. Đặc biệt trân trọng những giá trị mà các thầy cô phụ trách đề tài tiểu luận và khóa luận bộ môn cơ điện tử trao cho chúng em trong những năm tháng cuối của đời sinh viên.

Gửi lời chúc đến những người bạn tri kỉ chốn giảng đường, chúc các bạn thành công trong con đường sự nghiệp cũng như trong hạnh phúc gia đình.

Hậu thuẫn to lớn từ gia đình là động lực để chúng em tiến bước vào đời. Cảm ơn sự hi sinh cao cả của cha mẹ và anh chị em đã giúp đỡ chúng em trong những tháng năm cấp sách tới trường.

TÓM LƯỢC

Bài báo cáo này bao gồm việc thiết kế và điều khiển mô hình cánh tay máy năm bậc tự do. Mô hình với mục đích chính là dùng trong học tập, đào tạo hay nghiên cứu. Đề tài với sự kế thừa về thiết kế cũng như trong chương trình điều khiển, và với đôi chút cải tiến khiến cho việc sử dụng thao tác trở nên dễ dàng hơn.

Qua việc nghiên cứu, khảo sát thị trường trong những năm gần đây cho thấy thị trường robot cộng tác đang có xu hướng phát triển dù số lượng không nhiều qua các năm. Rút ra kết luận về mục tiêu của đề tài là thỏa nhu cầu về vấn đề đào tạo nhân công cho robot cộng tác với các tính năng hữu ích, dễ sử dụng.

Mô hình điều khiển gồm các linh kiện module có sẵn trên thị trường: Servo MG996R, MG90S, màn hình LCD 16x2, I2C LCD, chiết áp, bộ cấp nguồn AC-DC.

Các chi tiết tay máy được thiết kế qua trình Solidworks và được chế tạo bằng phương pháp in 3D nhờ file stl xuất từ phần mềm.

Đề tài sử dụng mạch Arduino với trình biên tập Arduino IDE, cung cấp đầy đủ các thư viện và cú pháp trực quan dễ sử dụng, để lập trình cho chương trình điều khiển.

Tính hiệu quả và trực quan dễ sử dụng của đề tài được thể hiện qua sự tối giản của bộ điều khiển với 3 chế độ: chế độ điều khiển và lưu thao tác(record), chế độ chạy tay máy(play), chế độ trở về vị trí tay máy (reset). Ngoài ra còn có thể xóa chương trình đã lưu để thực hiện thao tác mới.

Vì mục đích chính là để cho việc nghiên cứu, đào tạo nên mô hình cũng không tránh khỏi những nhược điểm về độ chịu tải và tính chính xác của tay máy và hạn hẹp về các chức năng.

Với bài toán động học thuận và động học ngược đã có trong bài thì sẽ có ích việc nâng cấp, cải tiến cho các tính năng mới của mô hình sau này.

MỤC LỤC

LỜI MỞ ĐẦU	2
TÓM LƯỢC	3
DANH MỤC HÌNH ẢNH.....	6
DANH MỤC CÁC BẢNG.....	8
Chương 1: MỞ ĐẦU	9
1.1 ĐẶT VẤN ĐỀ.....	9
1.1.1 Khảo sát thị trường :	9
1.1.2 Nhu cầu hiện nay:	10
1.1.3 Tại sao lại chọn phương pháp điều khiển mẫu lặp làm đề tài:	12
1.2 TÍNH CẤP THIẾT CỦA ĐỀ TÀI:	12
1.3 MỤC TIÊU VÀ NỘI DUNG CỦA ĐỀ TÀI NGHIÊN CỨU:	13
1.3.1 Mục tiêu của đề tài nghiên cứu:.....	13
1.3.2 Nội dung nghiên cứu:	13
1.4 HẠN CHẾ CỦA ĐỀ TÀI:	13
Chương 2: TỔNG QUAN	14
2.1 SƠ LƯỢC VỀ CÁC LOẠI ROBOT CỘNG TÁC:	14
2.2 TỔNG QUAN VỀ ĐỀ TÀI NGHIÊN CỨU.....	18
2.2.1 Cấu tạo chung của cánh tay máy 5 bậc tự do:	18
2.2.1.1 Kết cấu tay máy	18
2.2.1.2 Động học tay máy	19
2.2.2 Linh kiện, thiết bị sử dụng cho mô hình:.....	23
2.2.2.1 Động cơ servo mg996r và g90s:	23
2.2.2.2 Màn hình hiển thị 16x2 LCD:.....	25
2.2.2.3 Bộ giao tiếp I2C LCD:.....	27
2.2.2.4 Arduino Mega 2560:.....	28
2.2.2.5 Chiết áp	29
2.2.2.6 Bộ cấp nguồn AC-DC:.....	30
2.2.3 Phần mềm sử dụng:.....	30

2.2.3.1 Solidworks	30
2.2.3.2 Arduino IDE	30
Chương 3: NỘI DUNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU	31
3.1 NỘI DUNG ĐỀ TÀI NGHIÊN CỨU:.....	31
3.2 CƠ SỞ THIẾT KẾ HỆ THỐNG:.....	31
3.2.1 Cơ sở thiết kế các chi tiết tay máy:.....	31
3.2.2 Cơ sở thiết kế mạch điều khiển:	31
3.2.3 Cơ sở thiết kế chương trình điều khiển:	32
3.3 PHƯƠNG PHÁP SỬ DỤNG:.....	32
3.3.1 Thiết kế, chế tạo cánh tay máy	32
3.3.2 Mạch điều khiển cho hệ thống tay máy	33
3.3.3 Chương trình điều khiển cho hệ thống	33
Chương 4: KẾT QUẢ VÀ THẢO LUẬN.....	34
4.1 THIẾT KẾ MÔ HÌNH TAY MÁY	34
4.2 THIẾT KẾ CHẾ TẠO MẠCH ĐIỀU KHIỂN:	36
4.3 SƠ ĐỒ KHỐI CỦA HỆ THỐNG ĐIỀU KHIỂN:.....	37
4.4 LƯU ĐỒ THUẬT TOÁN:.....	38
4.5 KHẢO NGHIỆM, ĐÁNH GIÁ HOẠT ĐỘNG HỆ THỐNG:	39
Chương 5: KẾT LUẬN VÀ ĐỀ NGHỊ	40
5.1 NHỮNG KẾT QUẢ ĐẠT ĐƯỢC:.....	40
5.2 HẠN CHẾ CỦA ĐỀ TÀI:	41
5.3 HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI:	41
TÀI LIỆU THAM KHẢO	42
PHỤ LỤC	44

DANH MỤC HÌNH ẢNH

Hình 1.1	Số lượng robot công nghiệp lắp đặt ở 15 thị trường lớn nhất năm 2019.....	9
Hình 1.2	Mẫu cobot GoFa™ của ABB's Robotics	11
Hình 2.1	GoFa™ CRB 15000.....	14
Hình 2.2	UR3e Collaborative robot arm.....	15
Hình 2.3	Bộ xử lý OmniCore C30& OmniCore C90XT	16
Hình 2.4	Bảng điều khiển và bộ xử lý Universal Robots	16
Hình 2.5	Mô hình cánh tay 5 bậc tự do - đồ án robot 2-2020.	17
Hình 2.6	Logo của ứng dụng Blynk.	17
Hình 2.7	Hệ thống điều khiển tay máy 5 bậc tự do.	18
Hình 2.8	Bản phác thảo cánh tay robot.....	19
Hình 2.9	Phân tích hệ tọa độ các liên kết tay máy.....	20
Hình 2.10	Mô tả hệ tọa độ vật gắp và tay gắp	21
Hình 2.11	Sơ đồ dây và hình mô tả chu kỳ hoạt động của sóng PWM.....	23
Hình 2.12	Động cơ servo MG996R cùng phụ kiện kèm theo.	24
Hình 2.13	Động cơ servo MG90S.	25
Hình 2.14	Bảng vẽ kỹ thuật của màn hình LCD 16x2.....	25
Hình 2.15	Màn hình LCD 16x2 cùng sơ đồ chân kết nối.....	26
Hình 2.16	Bộ giao tiếp I2C LCD	27
Hình 2.17	Arduino ATmega2560 cùng sơ đồ chân kết nối.	28
Hình 2.18	Chiết áp.	29
Hình 2.19	Bộ cấp nguồn AC-DC 5V-30A.....	30

Hình 3.1 Quá trình hoàn thiện mô hình tay máy.	32
Hình 4.1 Nguyên lý cấu tạo mô hình gồm cánh tay máy và cánh tay điều khiển.	34
Hình 4.2 Mô hình thực tế của hệ thống	35
Hình 4.3 Mạch điều khiển được mô tả lại qua phần mềm proteus	36
Hình 4.4 Bộ điều khiển của mô hình.	37
Hình 4.5 Sơ đồ khối hệ thống điều khiển	37
Hình 4.6 Lưu đồ thuật toán của chương trình điều khiển.	38
Hình 5.1 Hệ thống tay máy đang hoạt động	40
Hình 5.2 Biểu thị trạng thái khởi động, đang sẵn sàng hoạt động và thông báo không có thao tác nào đã lưu, ngưng hoạt động.....	40
Hình 5.3 Biểu thị đang ghi thao tác 2-3; Thông báo có hai bước đã lưu; Hiện thị trạng thái đang chạy thao tác; Về vị trí tay máy; Thông báo đang xóa các thao tác lưu.....	41

DANH MỤC CÁC BẢNG

Bảng 2.1 Tổ hợp các thông số Denavit-Hartenberg.....	20
Bảng 2.2 Mô tả giao diện các chân kết nối của LCD 16x2.	26
Bảng 2.3 Tóm lược thông số kỹ thuật của ArduinoMega 2560.....	29
Bảng 4.1 Khảo nghiệm độ phản hồi của các khớp tay máy chính so với tay điều khiển với 10 lần thử nghiệm trên mỗi khớp.	39

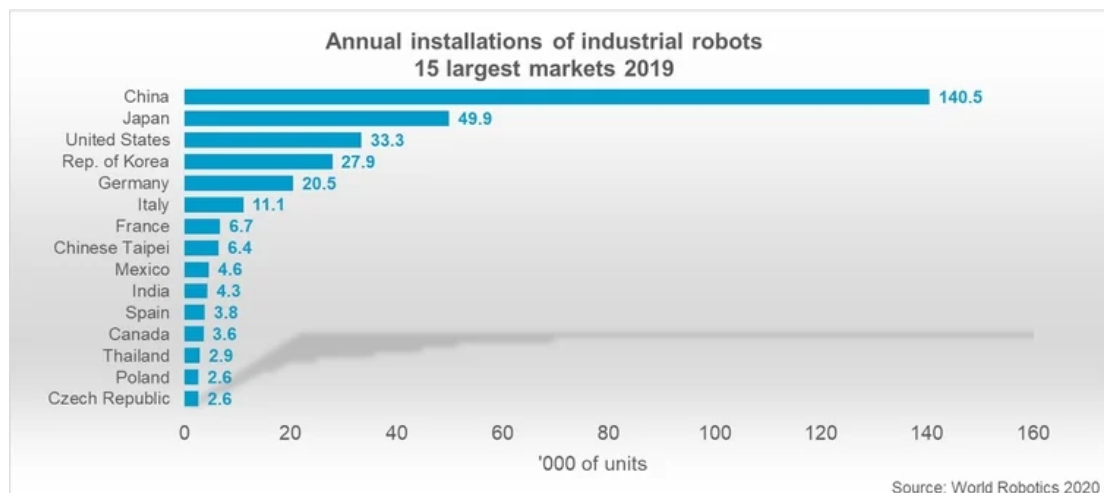
Chương 1: MỞ ĐẦU

1.1 ĐẶT VẤN ĐỀ

1.1.1 Khảo sát thị trường :

Cách mạng công nghệ lần thứ ba đang diễn ra và sự bùng nổ của cách mạng số (thay đổi cách thức mã hóa và tiếp cận thông tin) khiến cho việc nhiều công ty, doanh nghiệp có nhu cầu đổi mới quá trình sản xuất với dây chuyền tự động. Và để phát huy, cải thiện chất lượng cũng như năng suất thì không thể thiếu các tay robot công nghiệp với khả năng vận hành liên tục chính xác.

Theo dữ liệu thống kê của IFR, liên đoàn robot thế giới, trong bài báo cáo tháng 9 năm 2020 đã liệt kê con số kỷ lục: 2,7 triệu robot công nghiệp đang hoạt động trong các nhà máy trên khắp thế giới [1].



Hình 1.1 Số lượng robot công nghiệp lắp đặt ở 15 thị trường lớn nhất năm 2019

Như vậy theo số liệu như hình 1.1 thì trong 15 thị trường lớn có tới 6 quốc gia thuộc khu vực châu Á Thái Bình Dương gồm Trung Quốc, Nhật Bản, Đài Loan, Hàn Quốc, Thái Lan.

Nghiên cứu gần đây chỉ ra rằng cứ một robot trên mỗi một nghìn nhân công sẽ làm giảm 0,2 phần trăm tỷ lệ người có việc làm trên tổng dân số và tiền công cũng giảm đi 0,42 phần trăm [2] (thông số đối với thị trường lao động Mỹ). Vì vậy hầu hết các công ty muốn đáp ứng nhu cầu thị trường(giá thành rẻ, sản phẩm chất lượng,...) nên nhu cầu tăng cường tự động hóa để tăng năng suất, độ chính xác, chất lượng sản phẩm cũng tăng theo.

Theo đó việc các doanh nghiệp ồ ạt áp dụng robot công nghiệp hoàn toàn vào trong dây chuyền sản xuất của các công ty nhà máy lớn ảnh hưởng đến nền kinh tế nội địa làm cho tỉ lệ thất nghiệp tăng kèm theo nhu cầu về nguồn nhân công có kỹ năng phù hợp với công việc mới.

Bằng những ưu điểm vượt trội như tính chính xác, linh hoạt và năng suất cao cánh tay robot công nghiệp thường có vị trí và nhiệm vụ cố định, chỉ phục vụ sản xuất cho những ngành công nghiệp nặng với khối lượng công việc khổng lồ như sản xuất ô tô, công nghiệp chế tạo như đúc, gia công chi tiết lớn...

Hơn nữa với tuổi thọ cho mỗi chu kỳ sản phẩm trên thị trường vào khoảng 10 năm khiến cho việc thay đổi dây chuyền và lập trình tinh chỉnh cho phù hợp với hệ thống mới sẽ trở nên phức tạp, cần số vốn khổng lồ để đầu tư.

1.1.2 Nhu cầu hiện nay:

Hiện nay để hoàn thành mục tiêu tự động hóa hoàn toàn đối với các nhà máy lớn thì không thể thiếu các cánh tay robot cộng tác, được viết tắt là cobot (collaborative robot). Đối với các cánh tay công nghiệp lớn thì chỉ lập trình cho một mục đích chuyên dụng nhất định thì cobot thực hiện được các thao tác thủ công, nhiều mục đích và dễ lập trình.

- Cobot hoạt động an toàn trong môi trường làm việc với con người - với các thao tác và tốc độ an toàn, được lập trình, trang bị cảm biến để ngừng khẩn cấp khi phát hiện va chạm không mong muốn.
- Đối với các doanh nghiệp vừa và nhỏ có nhu cầu cải thiện qui trình công nghệ tạo môi trường làm việc hiện đại, chuyên nghiệp thì cobot sẽ là cái tên phù hợp vì số

vốn bỏ ra ít, kết hợp linh hoạt giữa phân bố nhân công xen kẽ tự động hóa nhằm tối ưu chi phí.

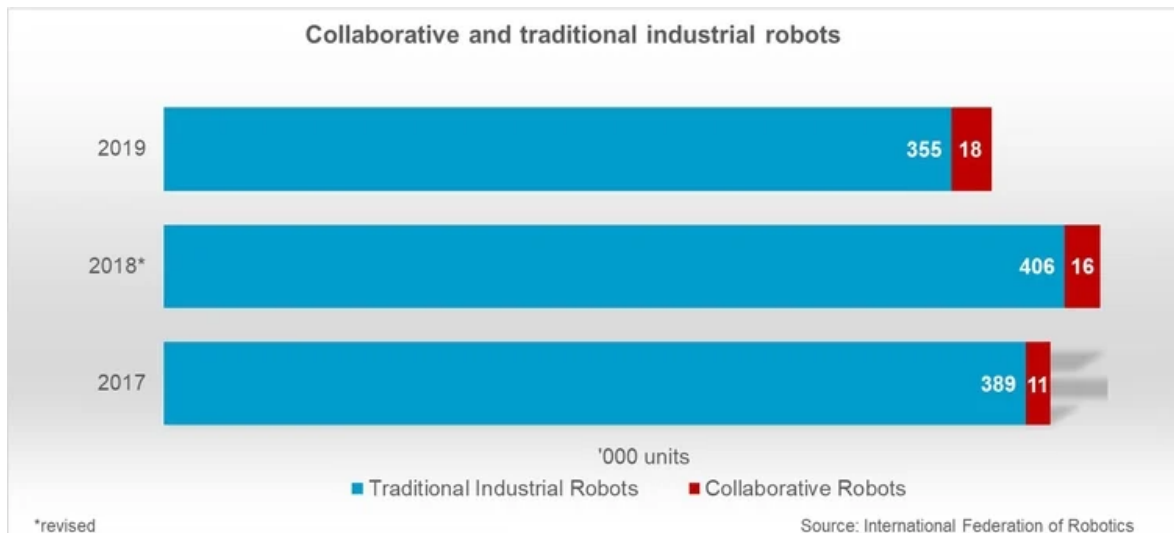
- Hơn nữa trong tương lai các cánh tay máy phụ trợ đầy tiềm năng này sẽ góp mặt trong hầu hết các ngành nghề như y tế, công nghệ hóa học, trong giáo dục hay chế biến thực phẩm...

” Làm cho robot trở nên dễ tiếp cận hơn, dễ sử dụng hơn và trực quan hơn là chìa khóa chỉ nam cho tầm nhìn của chúng tôi để tạo ra những con robot quen thuộc ở nơi làm việc như chiếc laptop ngày nay” [3] - Sami Atiya, chủ tịch công ty ABB's Robotics.

Do đó với sự đổi mới phá cách trong thiết kế cùng với tầm nhìn đi trước thời đại đã giúp cho sản phẩm mới của ABB: cobot GoFa™ giành lấy giải thưởng danh giá “ Best of the Best Red Dot award” vào tháng 4 năm 2021 (Hình 1.2).



Hình 1.2 Mẫu cobot GoFa™ của ABB's Robotics



Hình 1.3 Số lượng robot cộng tác và robot truyền thống được lắp đặt trong ba năm gần đây.

Số liệu thống kê qua biểu đồ của IFR cho thấy mặc dù số lượng cobots được lắp đặt tuy tăng đều qua các năm nhưng sản lượng không cao, chứng tỏ rằng mảng cobots này tuy phát triển nhanh nhưng vẫn còn sơ khai [1].

1.1.3 Tại sao lại chọn phương pháp điều khiển mẫu lặp làm đề tài:

Mặc dù cobot có rất nhiều điểm đặc biệt trong thiết kế về phần cứng nhưng chúng tôi lại chọn phương pháp điều khiển làm trọng tâm nghiên cứu vì những mục đích sau:

- Nghiên cứu, hiểu biết thêm về phương pháp điều khiển mới.
- Phương pháp lập trình trực quan bằng cách mô hình hóa thực cánh tay điều khiển, phù hợp làm mô hình nghiên cứu, đào tạo nhân công.
- Đây cũng là bước đầu để tiếp cận, làm quen với phương pháp lập trình “record and play” hay thuần việt hơn là ” ghi mẫu lặp”, một phương thức lập trình dễ dàng và trực quan là tính năng ưu việt của nhiều loại cobot trên thị trường.

1.2 TÍNH CẤP THIẾT CỦA ĐỀ TÀI:

Các ngành công nghiệp vừa và nhỏ ở Việt Nam với nhu cầu thay đổi dây chuyền sản xuất với công nghệ hiện đại, môi trường làm việc kết hợp gồm robot và nhân công nhằm tối ưu năng suất và chi phí vận hành.

Với xu hướng rằng cobot sẽ là tiền đề cho sự phát triển kinh tế trong thế kỷ 21 của Việt Nam thì việc nghiên cứu cách thức điều khiển hay về các tính năng thân thiện hữu ích với người dùng là nhu cầu cấp thiết trong chương trình đào tạo hiện nay.

1.3 MỤC TIÊU VÀ NỘI DUNG CỦA ĐỀ TÀI NGHIÊN CỨU:

1.3.1 Mục tiêu của đề tài nghiên cứu:

Nghiên cứu, lập trình cho phương pháp mẫu lặp, phương pháp điều khiển tương ứng trên các loại cobot phổ biến hiện nay với nhiều cái tên khác nhau như “record and play”, “direct teaching”, “lead through”..., với các tính năng hữu ích, dễ sử dụng.

1.3.2 Nội dung nghiên cứu:

a. Thiết kế chế tạo mô hình tay máy :

- Thiết kế bằng phần mềm SolidWork.
- Chế tạo bằng phương pháp in 3D.

b. Lắp đặt mạch điều khiển:

- Dùng servo MG996R làm động cơ cho các khớp của tay máy.
- Sử dụng mạch Arduino Mega.
- Sử dụng mạch LCD giao tiếp với Arduino qua module I2C.
- Kết nối tín hiệu của các công tắc điều khiển với mạch arduino.

c. Xây dựng chương trình và lưu đồ thuật toán:

- Chương trình xây dựng dựa trên phương pháp điều khiển record and play.
- Tích hợp các chức năng cơ bản, dễ sử dụng và làm quen.

d. Phương pháp khảo nghiệm đối với đề tài:

Kiểm tra độ phản hồi của các khớp tay máy.

1.4 HẠN CHẾ CỦA ĐỀ TÀI:

- Đề tài chỉ xoay quanh phương pháp điều khiển và các tính năng của hệ thống.
- Chưa khai thác được khả năng kết nối, điều khiển, lưu trữ và truyền dữ liệu .

Chương 2: TỔNG QUAN

2.1 SƠ LƯỢC VỀ CÁC LOẠI ROBOT CỘNG TÁC:

Thị trường robot công nghiệp ngày nay không chỉ tập trung vào các mảng robot chuyên dụng như CNC, tay máy robot công nghiệp mà còn mở rộng sang các loại robot cộng tác phù hợp với nhu cầu của các công xưởng sản xuất mang tính tự động hóa cao hay nhu cầu từ các ngành công nghiệp nhẹ tới các ngành dịch vụ (y tế, nhà hàng, khách sạn,...).

- **Thị trường quốc tế gồm có một số loại robot cộng tác tiêu biểu:**

ABB GoFa™ CRB 15000 (Hình 2.2)



Hình 2.1 GoFa™ CRB 15000

CRB 15000 được tích hợp các cảm biến mô men xoắn ở mỗi sáu khớp của tay máy. Ngoài sức mạnh vượt trội từ động cơ CRB 15000 còn có giới hạn hiệu suất lực tác động bởi cánh tay với độ phản hồi nhanh nhạy và chính xác. Tích hợp các tính năng trên giúp cho cánh tay máy ngăn chặn mọi rủi ro khi cảm nhận được bất kỳ tương tác ngoài dự tính

Với tính năng Lead Through, người dùng dễ dàng điều chỉnh các khớp của tay máy và lưu hành trình thao tác của robot vào bộ nhớ, giúp cho việc cài đặt thiết lập đơn giản hơn [4].

Universal Robot UR3e (Hình 2.3)



Hình 2.2 UR3e Collaborative robot arm

Cánh tay robot với ưu điểm kiểu dáng nhỏ gọn, phù hợp cho không gian làm việc chật hẹp. UR3e cho phép người dùng lập trình nhanh chóng, sử dụng bảng điều khiển để lập trình trực quan hơn bằng cách di chuyển các khâu, khớp đến các điểm tham chiếu mong muốn hoặc chọn các chế độ lập trình sẵn trên bảng điều khiển.

Cùng với cảm biến momen xoắn tích hợp mang lại độ chính xác tối đa và giúp việc lập trình UR3e cho các ứng dụng có độ nhạy cao [5].

Vậy qua các loại robot cộng tác được liệt kê trên thì hầu hết đều sử dụng phương pháp lập trình bằng cách ghi nhớ vị trí thao tác một cách trực quan dễ tiếp cận và đều có các chức năng bảo vệ người dùng trong trường hợp rủi ro không mong muốn.

Tương tự như laptop hay smartphone thì các bộ điều khiển của các loại cobot đều có điểm chung là dễ sử dụng với phần mềm, đồ họa cải tiến hỗ trợ cho việc điều khiển, lập trình, thay đổi các thông số cho các ứng dụng, chức năng linh hoạt của cobot. Đảm bảo độ ổn định chính xác trong thao tác của người dùng tránh nhầm lẫn gây thiệt hại.

Hình 2.4 với các bộ điều khiển cho các dòng robot của ABB gồm OmniCore C30 (2) OmniCore C90XT (1), bảng điều khiển FlexPendant (3)



Hình 2.3 Bộ xử lý OmniCore C30& OmniCore C90XT

Bảng điều khiển và bộ xử lý (Hình 2.7) của Universal Robots.



Hình 2.4 Bảng điều khiển và bộ xử lý Universal Robots

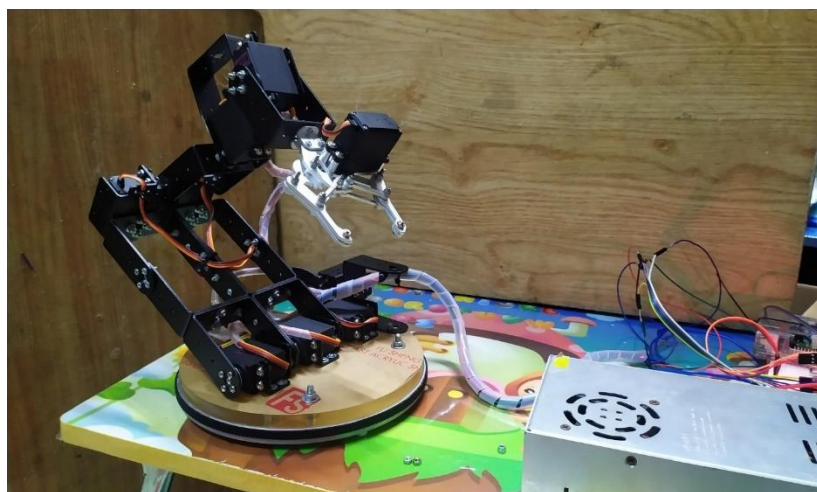
Hơn nữa việc trang bị đầy đủ các cổng kết nối truyền dữ liệu (Dây Bus, mạng Lan, Wifi..) giúp cho việc điều hành, quản lý hệ thống trở nên chặt chẽ, hiệu quả hơn.

Như vậy thị trường cobot trên thế giới đã phát triển đa dạng nhiều loại mẫu mã và kích thước, cùng với độ hoàn thiện về thiết kế phần cứng cũng như những chức năng (bảo vệ

người dùng, dễ lập trình cũng như thiết lập, cài đặt..) hữu ích cho từng môi trường làm việc.

Tuy nhiên thì việc phổ cập các loại tay máy cho các cơ sở đào tạo nhân công, hay các trường ở Việt Nam còn khá hạn chế vì giá thành cao và thiếu hụt các linh kiện phụ tùng thay thế, bảo trì bảo dưỡng.

- **Các đề tài nghiên cứu trước:**



Hình 2.5 Mô hình cánh tay 5 bậc tự do - đồ án robot 2-2020.

Mô hình của các đề tài trước với những ưu điểm là nhỏ gọn linh hoạt về vị trí làm việc ngoài ra nó còn chắc chắn, đáng tin cậy đối với các tác vụ cần trọng tải từ 100g-250g. Cánh tay được điều khiển không dây bằng mạng wifi qua ứng dụng IOT Blynk.

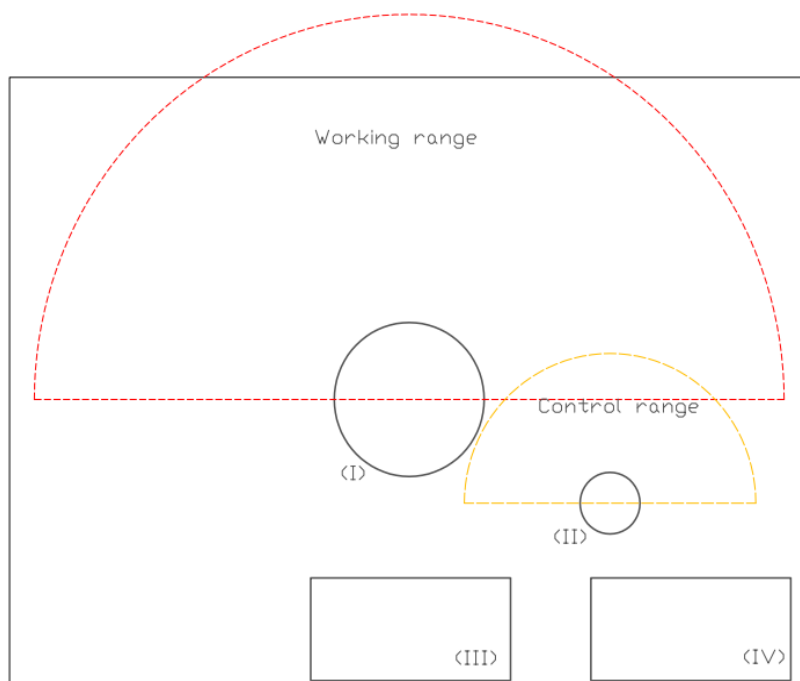


Hình 2.6 Logo của ứng dụng Blynk.

Với mạch xử lý raspberry pi cho tốc độ xử lý cao, đa nhiệm cung cấp nhiều tiềm năng cho việc phát triển hệ thống sau này như xử lý ảnh cũng như tìm hiểu thêm về trí tuệ nhân tạo hay machine learning giúp hệ thống hoạt động thông minh, hiệu quả hơn.

2.2 TỔNG QUAN VỀ ĐỀ TÀI NGHIÊN CỨU

Hệ thống gồm tay máy chính (I) , tay máy điều khiển (II), bộ xử lý (III), và nguồn cấp cho động cơ (IV) (Hình 2.8).



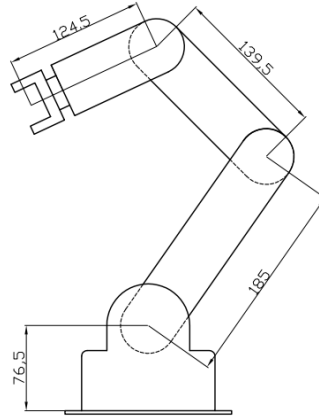
Hình 2.7 Hệ thống điều khiển tay máy 5 bậc tự do.

2.2.1 Cấu tạo chung của cánh tay máy 5 bậc tự do:

2.2.1.1 Kết cấu tay máy

Tay máy 5 bậc tự do thuộc loại mô phỏng theo cánh tay người với cả 5 khớp đều là khớp quay. Với 1 khớp xoay 1 khớp vai, 1 khớp khuỷu, nối cẳng tay với khuỷu tay và hai khớp cổ tay.

Với phần công tác thì tùy theo yêu cầu làm việc, phần công tác có thể là tay gắp (Gripper), công cụ (súng phun sơn, mỏ hàn, dao cắt...).



Hình 2.8 Bản phác thảo cánh tay robot

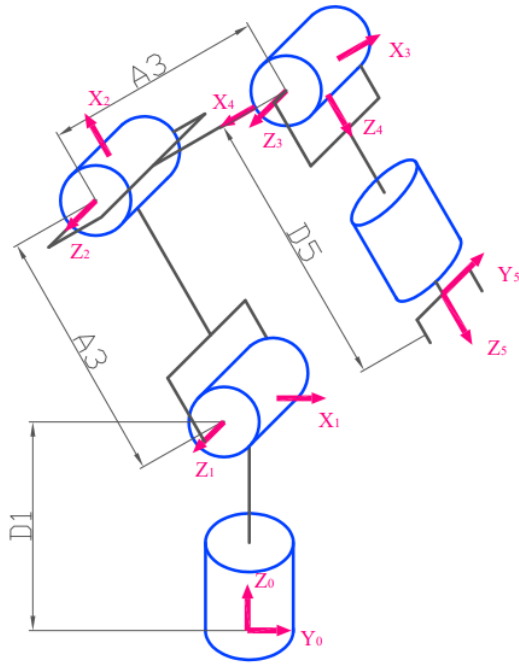
Tay máy dùng động cơ RC servo với nguồn là điện một chiều DC. Tay máy thuộc dạng điều khiển kín(hay điều khiển servo), sử dụng tín hiệu phản hồi để tăng độ chính xác điều khiển.

2.2.1.2 Động học tay máy

Một tay máy có thể biểu diễn bằng một chuỗi động học hở, gồm các khâu, liên kết với nhau bằng các khớp quay hoặc khớp trượt. Một đầu của chuỗi được gắn lên thân, còn đầu kia nối với phần công tác. Thao tác trong quá trình làm việc đòi hỏi phần công tác phải được định vị và định hướng chính xác trong không gian. Động học tay máy giải quyết 2 lớp bài toán:

- Lớp bài toán thuận căn cứ vào các biến khớp để xác định vùng làm việc của phần công tác và mô tả chuyển động của phần công tác trong vùng làm việc của nó.
- Lớp bài toán ngược, xác định các biến khớp để đảm bảo chuyển động cho trước của phần công tác.

Bài toán động học thuận và động học ngược được giải trên cơ sở quy tắc Denavit-Hartenberg. Là quy tắc thiết lập hệ tọa độ trên các cặp khâu khớp trên cánh tay máy, dựa trên hệ tọa 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 [6].



Bảng 2.1 Tổ hợp các thông số Denavit-Hartenberg

Liên kết	θ_i	d_i	a_i	α_i
1	$\theta_1 + \frac{\pi}{2}$	d_1	0	$\frac{\pi}{2}$
2	$\theta_2 + \frac{2\pi}{3}$	0	a_2	0
3	$\theta_3 - \frac{\pi}{2}$	0	a_3	0
4	$\theta_4 - \pi$	0	0	$-\frac{\pi}{2}$
5	θ_5	d_5	0	0

Hình 2.9 Phân tích hệ tọa độ các liên kết tay máy

a. Động học thuận:

Ta tính được ma trận chuyển đổi thuần nhất như sau:

$$T_5^0 = \begin{bmatrix} (S_1 C_5 S_{234} \frac{2\pi}{3} - S_1 C_5) & (-S_1 S_5 S_{234} \frac{2\pi}{3} - C_1 C_5) & S_1 C_{234} \frac{2\pi}{3} & P_x \\ (-C_1 C_5 S_{234} \frac{2\pi}{3} - S_1 S_5) & (C_1 S_5 S_{234} \frac{2\pi}{3} - S_1 C_5) & -C_1 C_{234} \frac{2\pi}{3} & P_y \\ C_5 C_{234} \frac{2\pi}{3} & -S_5 C_{234} \frac{2\pi}{3} & -S_{234} \frac{2\pi}{3} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

$$P_x = S_1 (d_5 C_{234} \frac{2\pi}{3} - a_3 S_{23} \frac{2\pi}{3} - a_2 C_{2} \frac{2\pi}{3}) \quad (2.2)$$

$$P_y = -C_1 (d_5 C_{234} \frac{2\pi}{3} - a_3 S_{23} \frac{2\pi}{3} - a_2 C_{2} \frac{2\pi}{3}) \quad (2.3)$$

$$P_z = -d_5 S_{234} \frac{2\pi}{3} - a_3 C_{23} \frac{2\pi}{3} + a_2 S_{2} \frac{2\pi}{3} + d_1 \quad (2.4)$$

Với S_1 tương ứng với $\sin(\theta_1)$ tương tự với S_2, S_3, \dots

C_1 ứng với $\cos(\theta_1)$ tương tự với C_2, C_3, \dots

Biên $C_{234:\frac{2\pi}{3}}$ ứng với $\cos(\theta_2 + \theta_3 + \theta_4 + \frac{2\pi}{3})$ tương tự như vậy với $C_{2:\frac{2\pi}{3}}, S_{23:\frac{2\pi}{3}}, \dots$

b. Động học ngược với các tham số cho trước:

Vị trí cần đến P_x, P_y, P_z .

Góc hợp bởi phần công tác so với phương ngang $\varnothing = \theta_2 + \theta_3 + \theta_4 - \frac{\pi}{3}$ (1.2).

Hệ số góc: α, β (Hình 2.11).

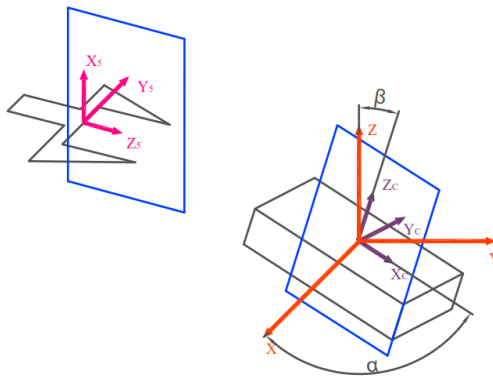
Giới hạn góc quay của các khớp tay máy: $\theta_1 \in [-\pi; 0], \theta_2 \in [-\frac{2\pi}{3}; \frac{\pi}{3}], \theta_3 \in [0; \pi],$

$$\theta_4 \in [0; \pi], \theta_5 \in [-\frac{\pi}{2}; \frac{\pi}{2}].$$

Từ biểu thức (2.2) và (2.3) ta được:

$$\theta_1 = \text{Atan2}(-P_x/P_y) \quad (2.5)$$

Để gắp được vật thì mặt phẳng gắp của tay kẹp và mặt gắp của vật (với hệ tọa độ vật cho trước) phải có mối quan hệ song song với nhau (Hình 2.11) (chi tiết xem thêm phần phụ lục I).



Hình 2.10 Mô tả hệ tọa độ vật gắp và tay gắp

Ta có hai trường hợp như sau:

Trường hợp $\overrightarrow{Y_s}$ và $\overrightarrow{Y_c}$ cùng chiều ta tính được:

$$C_5 = S_{\alpha-1} C_\beta \quad (2.6) .$$

$S_{\alpha-1}$ tương ứng với $\sin(\alpha - \theta_1)$ tương tự $S_{1-\alpha}$ tương ứng với $\sin(\theta_1 - \alpha) \dots$

$$\emptyset = \text{Atan2}\left(-\frac{C_{\alpha-1}C_{\beta}}{S_{\beta}}\right) (2.7).$$

$$S_5 = \pm \sqrt{S_{\beta}^2 + C_{\beta}^2 C_{1-\alpha}^2} (2.8).$$

$$\text{Vậy } \theta_5 = \text{Atan2}\left(\frac{S_5}{C_5}\right) (2.9).$$

Trường hợp $\vec{Y_5}$ và $\vec{Y_c}$ ngược chiều ta có:

$$C_5 = S_{1-\alpha} C_{\beta} (2.10).$$

$$\emptyset = \text{Atan2}\left(-\frac{C_{\alpha-1}C_{\beta}}{S_{\beta}}\right) (2.7).$$

$$S_5 = \pm \sqrt{S_{\beta}^2 + C_{\beta}^2 C_{1-\alpha}^2} (2.8).$$

Với S_5 và C_5 ta tính được θ_5 theo biểu thức (2.9).

Ta có biểu thức:

$$S_3 = \frac{P_{w_x}^2 + P_{w_z}^2 - a_2^2 - a_3^2}{2a_2a_3} (2.10) \rightarrow C_3 = \pm \sqrt{1 - S_3^2}$$

C_3 có giá trị âm dương tương ứng với hai tư thế hoạt động của phần công tác.

$$\text{Trong đó } P_{w_x} = \frac{P_x}{S_1} + d_5 C_{\emptyset} (2.11); P_{w_z} = P_z - d_1 - d_5 S_{\emptyset} (2.12)$$

Có S_3 và C_3 ta tính được $\theta_3 = \text{Atan2}(S_3/C_3)$

$$\text{Từ } C_3 \text{ và } S_3 \text{ tính được: } S_{2:\frac{2\pi}{3}} = \frac{P_{w_z}(a_3 S_3 + a_2) - a_3 C_3 P_{w_x}}{P_{w_x}^2 + P_{w_z}^2} (2.13)$$

$$C_{2:\frac{2\pi}{3}} = -\frac{P_{w_x}(a_3 S_3 + a_2) + a_3 C_3 P_{w_z}}{P_{w_x}^2 + P_{w_z}^2} (2.14)$$

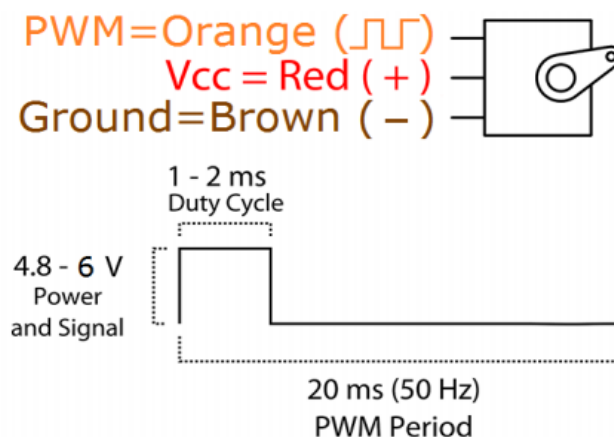
$$\text{Vậy } \theta_2 = \text{Atan2}\left(\frac{S_{2:\frac{2\pi}{3}}}{C_{2:\frac{2\pi}{3}}}\right) - \frac{2\pi}{3} (2.15)$$

$$\text{Từ biểu thức (1.2) ta tính được } \theta_4 = \frac{\pi}{3} + \emptyset - (\theta_2 + \theta_3) (2.16)$$

2.2.2 Linh kiện, thiết bị sử dụng cho mô hình:

2.2.2.1 Động cơ servo mg996r và g90s:

Hai động cơ servo sau đây thuộc loại RC servo quay từ 0 cho đến xấp xỉ 180° tùy thuộc vào chu kỳ hoạt động của sóng PWM cấp cho chân tín hiệu của servo (Hình 2.12).



Hình 2.11 Sơ đồ dây và hình mô tả chu kỳ hoạt động của sóng PWM.

Trong trình biên tập Arduino IDE thì ở thư viện servo.h có hai câu lệnh cấp xung cho servo như sau:

- [tên servo điều khiển].write([góc cần đến]);
- [tên servo điều khiển].writeMicroseconds ([chỉ số xung cấp cho servo]);

Ở hai trường hợp trên thì ở câu lệnh đầu việc bấm số góc điều khiển sẽ trở nên khó khăn vì góc cần đến chỉ nằm trong khoảng 0-180 nên sẽ không tránh trường hợp các chỉ số góc bấm bị lẻ và thiếu chính xác. Với câu lệnh thứ hai thì chỉ số nhập vào dao động từ 1000-2000 μ s (tương ứng với 1-2ms) khi biên tập trong IDE, thì việc bấm xung điều khiển cho servo sẽ dễ dàng và chính xác hơn vì không có số lẻ và chỉ nhập số nguyên.

Sơ đồ đấu dây của servo tiêu chuẩn:

- Màu cam ứng với dây tín hiệu.
- Màu đỏ ứng với dây nguồn (V+).

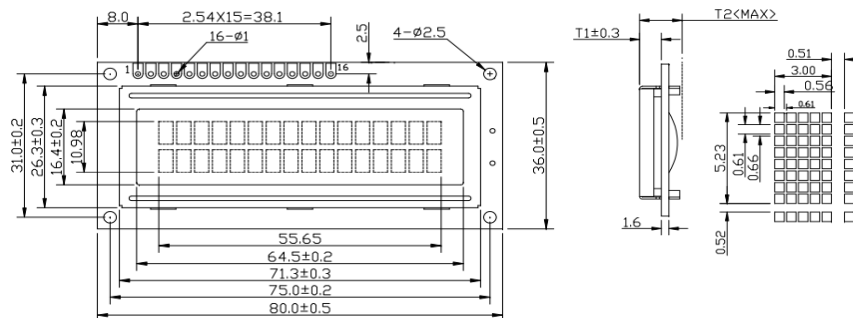
- Kích thước: 22.5 x 12 x 35.5 mm
- Điện áp hoạt động: 4.8V-6V
- Mô-men xoắn: 1.8 kgf*cm (4.8V) 2.2kgf*cm (6V)
- Tốc độ hoạt động: 0.1 s/60° (4.8V), 0.08 s/60° (6V)
- Dải dung sai độ rộng PWM (dead band width): 5 μ s



Hình 2.13 Động cơ servo MG90S.

2.2.2.2 Màn hình hiển thị 16x2 LCD:

Màn hình hiển thị LCD [9] gồm 16 cột và 2 dòng. Hiển thị đồng thời 32 ký tự với mỗi ký tự được hình thành từ 5x8 điểm ảnh.

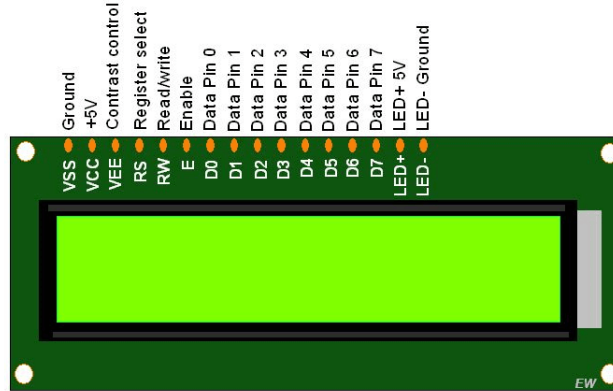


Hình 2.14 Bảng vẽ kỹ thuật của màn hình LCD 16x2

Thông số kỹ thuật:

- Điện áp hoạt động: 4.7V - 5.3V
- Dòng tiêu thụ: 1mA (không có đèn nền)
- Hiển thị chữ, số và các ký tự đặc biệt
- Gồm 2 dòng, mỗi dòng có thể in 16 ký tự
- Có thể hoạt động ở hai chế độ: 8-bit và 4-bit

- Có thể hiển thị bất cứ ký hiệu tùy chỉnh
- Có sẵn đèn nền màu xanh lục hoặc xanh lam



Hình 2.15 Màn hình LCD 16x2 cùng sơ đồ chân kết nối

Pin No:	Pin Name:	Mô tả
1	Vss (Ground)	Chân GND nối với GND của hệ thống.
2	Vdd (+5V)	Cấp nguồn cho LCD (4.7V-5.3V).
3	VE (Contrast V)	Ảnh hưởng đến độ tương phản của màn hình. Nối GND để được độ tương phản tối đa.
4	RS (Register Select)	Kết nối với vi điều khiển để chuyển đổi giữa thanh ghi lệnh hoặc thanh ghi dữ liệu.
5	R/W(Read/Write)	Được dùng để đọc hay ghi dữ liệu. Thông thường được nối đất (GND) để ghi dữ liệu cho LCD
6	Enable	Được kết nối với chân của vi điều khiển- nhận tín hiệu chuyển đổi giữa 0-1 , để xác nhận giữ liệu.
7	Data Pin 0	Các chân dữ liệu từ 0-7 hình thành nên một chuỗi dữ liệu 8-bit. Chúng được dùng để chuyển đổi dữ liệu giữa MPU và LCM.
8	Data Pin 1	
9	Data Pin 2	
10	Data Pin 3	
11	Data Pin 4	
12	Data Pin 5	
13	Data Pin 6	
14	Data Pin 7	
15	LED Positive	Chân cực dương đèn nền LED
16	LED Negative	Chân cực âm đèn nền LED

Bảng 2.2 Mô tả giao diện các chân kết nối của LCD 16x2.

2.2.2.3 Bộ giao tiếp I2C LCD:

Module I2C [10] LCD sử dụng driver HD44780(LCD 16x2, LCD 20x4...) và tương thích với hầu hết các bộ vi điều khiển hiện nay.



Hình 2.16 Bộ giao tiếp I2C LCD

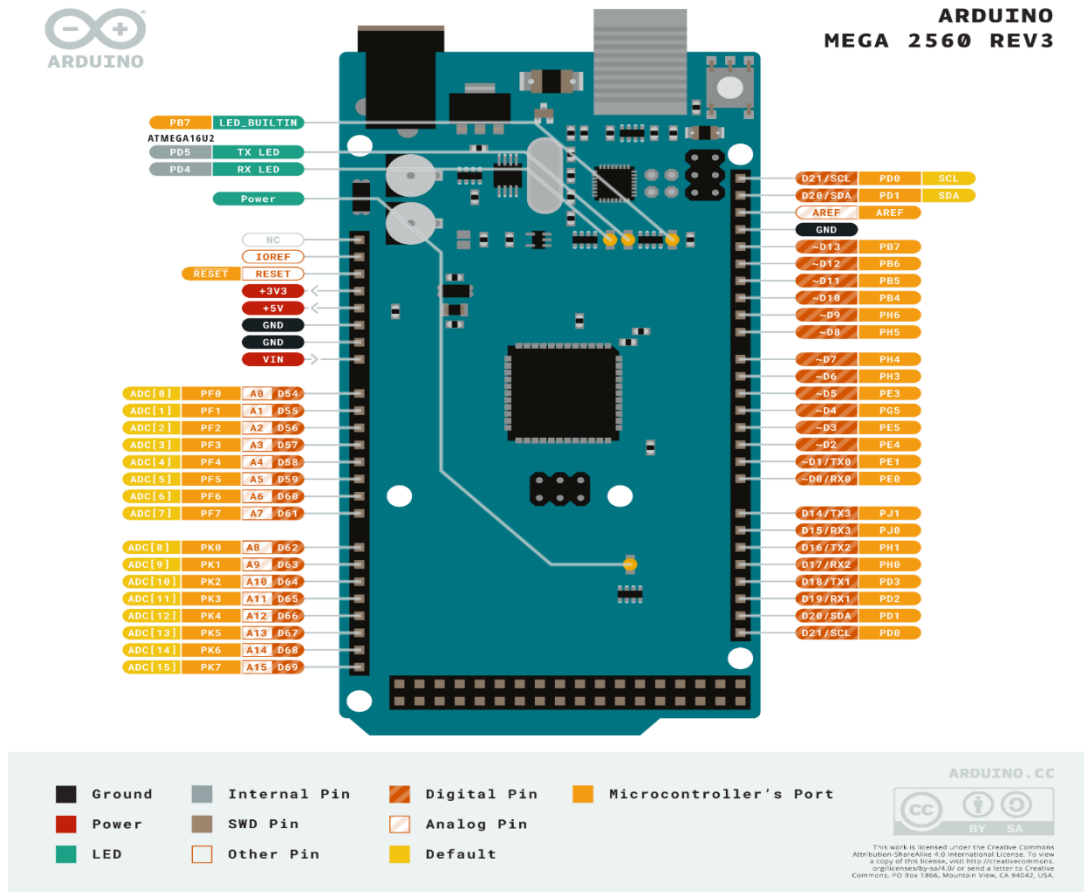
Thông số kỹ thuật:

- Điện áp hoạt động: 2.5-6V DC
- Tương thích với các màn hình LCD: 1602,1604,2004 (driver HD44780)
- Giao tiếp: I2C
- Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2)
- Kích thước: 41.5mm(L)x19mm(W)x15.3mm(H)
- Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt
- Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD

Bộ giao tiếp I2C rút gọn số chân kết nối với vi điều khiển. Từ 16 chân kết nối thu gọn còn hai chân nguồn và hai chân tín hiệu.

2.2.2.4 Arduino Mega 2560:

Arduino Mega 2560 [11] là bo mạch vi điều khiển dựa trên Atmega2560. Mạch vi điều khiển với đầy đủ các phương thức giao tiếp như I²C, USART, USB và các cổng I/O, PWM, Analog... hỗ trợ cho nhu cầu học tập nghiên cứu.



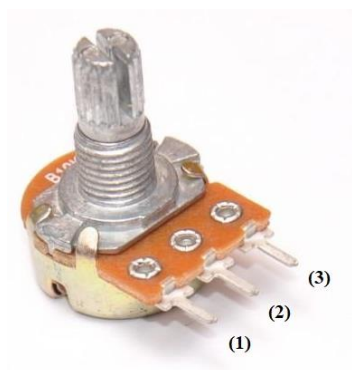
Hình 2.17 Arduino ATmega2560 cùng sơ đồ chân kết nối.

Vi điều khiển	Atmega2560
Điện áp hoạt động	5V
Điện áp đầu vào (khuyên dùng)	7-12V
Điện áp đầu vào (giới hạn)	6-20V
Số chân I/O-Digital	54(trong đó gồm 14 chân PWM)
Số chân nhận tín hiệu Analog	16
Dòng DC cho mỗi chân I/O	40mA (tối đa)
Dòng DC cho chân 3.3V	50mA (tối đa)
Bộ nhớ Flash	256 KB trong đó 8KB cho bootloader
SRAM	8KB
EEPROM	4KB
Tốc độ xử lý	16MHz

Bảng 2.3 Tóm lược thông số kỹ thuật của ArduinoMega 2560.

2.2.2.5 Chiết áp

Chiết áp hay còn gọi là biến trở với giá trị 10K Ohm được lựa chọn để sử dụng trong mô hình(lắp đặt trong các khớp của cánh tay điều khiển).



Chân số (1) và (3) nối với hai điện cực của nguồn điện.

Chân số (2) là chân cho ra điện áp (tín hiệu analog) điều chỉnh được.

Hình 2.18 Chiết áp.

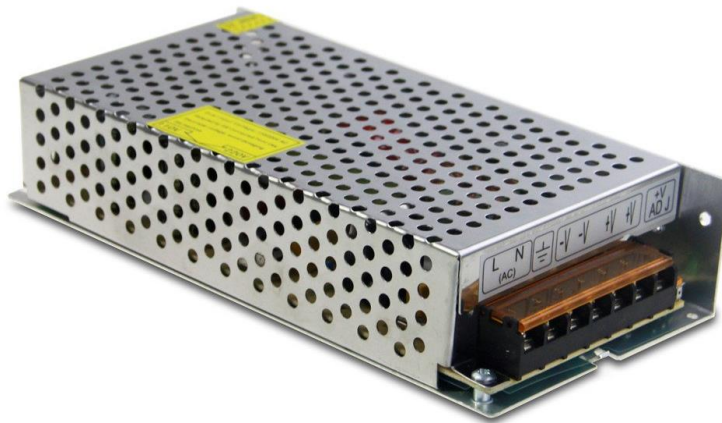
Thông số kỹ thuật:

- Kiểu chiết áp: dạng xoay hay còn gọi là biến trở volume
- Giá trị điện trở: 10Kohm

2.2.2.6 Bộ cấp nguồn AC-DC:

Thông số kỹ thuật:

- Điện áp đầu vào: 110/220V
- Điện áp đầu ra: 5-6V
- Công suất tối đa: 150 W



Hình 2.19 Bộ cấp nguồn AC-DC 5V-30A

2.2.3 Phần mềm sử dụng:

2.2.3.1 Solidworks

Solidworks thuộc loại phần mềm CAD, phần mềm hỗ trợ đầy đủ các công cụ thiết kế mô hình 3D.

Phần mềm thiết kế ngoài Solidworks còn có Autocad, Inventor By Autodesk,... với các chức năng, và tính ưu việt riêng.

2.2.3.2 Arduino IDE

Arduino IDE [12] là phần mềm đa nền tảng (Windows, MacOS, Linux), phần mềm này gồm các hàm được viết bởi ngôn ngữ C, C++.

Phần mềm dùng để biên tập và tải các chương trình lên các mạch điều khiển Arduino tương thích.

Chương 3: NỘI DUNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU

3.1 NỘI DUNG ĐỀ TÀI NGHIÊN CỨU:

- Xây dựng mô hình điều khiển gồm cánh tay máy năm bậc tự do, mạch điều khiển.
- Lập trình cho các chức năng điều khiển của tay máy.
- Khảo nghiệm tính hiệu quả, trực quan của mô hình đối với người sử dụng.

3.2 CƠ SỞ THIẾT KẾ HỆ THỐNG:

3.2.1 Cơ sở thiết kế các chi tiết tay máy:

- Dựa vào mục đích, tính chất của công việc hay chức năng mà lựa chọn mẫu tay máy, vật liệu và động cơ phù hợp.
- Mô hình được chọn lựa sử dụng trong đề tài: cánh tay máy 5 bậc tự do. Với mục đích chính là nghiên cứu phương thức điều khiển nên việc lựa chọn động cơ và vật liệu phải tối ưu chi phí:
 - Vật liệu sử dụng: nhựa pla
 - Động cơ: 5 động cơ Servo MG996R và 2 động cơ MG90S
- Lựa chọn động cơ phù hợp với tính chịu tải của các khâu, khớp tay máy.
- Lựa chọn các chi tiết lắp như ổ bi, bi sắt hay ốc vít có tính phổ biến rộng trên thị trường.

3.2.2 Cơ sở thiết kế mạch điều khiển:

- Lựa chọn hay thiết kế mạch điều khiển phụ thuộc vào các tính năng của hệ thống, số lượng các chân và các loại tín hiệu vào ra trong hệ thống.
- Mô hình dùng mạch điều khiển Arduino Mega với đa dạng các cổng truyền tín hiệu và số lượng các cổng đáp ứng được nhu cầu của đề tài.

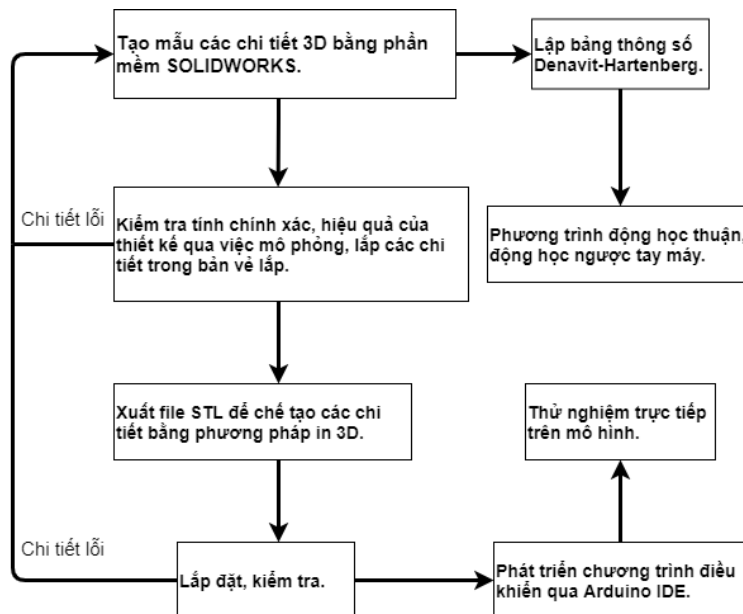
- Nguồn năng lượng cấp cho tay máy và vi điều khiển phải đủ điện áp, công suất hoạt động của hệ thống
- Để tránh việc nhiễu tín hiệu thì hệ thống gồm hai nguồn cấp riêng cho tay máy và mạch điều khiển.
- Đảm bảo tính linh hoạt, tối giản không gian làm việc của hệ thống.

3.2.3 Cơ sở thiết kế chương trình điều khiển:

- Việc thiết kế chương trình điều khiển cho hệ thống dựa vào trải nghiệm người dùng.
- Chương trình dễ thao tác, điều khiển. Đầy đủ các hướng dẫn, chú thích.
- Đảm bảo tính ổn định và tốc độ phản hồi của chương trình, tránh bị lỗi khi đang thao tác, điều khiển tay máy.

3.3 PHƯƠNG PHÁP SỬ DỤNG:

3.3.1 Thiết kế, chế tạo cánh tay máy



Hình 3.1 Quá trình hoàn thiện mô hình tay máy.

- Thiết kế các chi tiết dựa vào các kích thước của động cơ và các khâu đã xác định ở bảng phác thảo ở Hình 2.9.

- Dùng phương pháp in 3d (hay công nghệ bồi đắp vật liệu) để chế tạo ra các chi tiết tay máy.

3.3.2 Mạch điều khiển cho hệ thống tay máy

- Arduino Mega làm mạch điều khiển chính.
- Mạch điều khiển nhận tín hiệu analog từ biến trở và chuyển đổi thành xung điều khiển cho các động cơ servo của tay máy.
- Sử dụng các nút nhấn cho việc chuyển đổi các chế độ điều khiển hay dừng khẩn cấp.
- Dùng kết nối I2C để giao tiếp, hiển thị trạng thái hoạt động lên màn LCD.

3.3.3 Chương trình điều khiển cho hệ thống

- Các thư viện hỗ trợ sử dụng trong trình biên tập Arduino IDE:
 - Servo.h
 - Wire.h
 - LiquidCrystal_I2C.h
- Hàm timer millis() và micros() là công cụ chính, là khung xương để thực hiện các chức năng của hệ thống.
- Chương trình với các dòng lệnh với vòng lặp liên tục để kiểm tra trạng thái của các nút điều khiển và quá trình nhận, xuất tín hiệu điều khiển.

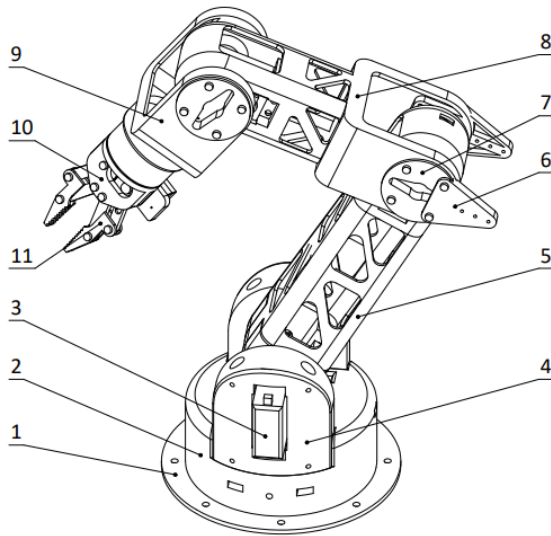
Chương trình chính khởi động và chạy trong vòng lặp liên tục void loop(){ } nên các dòng lệnh kiểm tra trạng thái của các công tắc với tốc độ phản hồi cao chỉ trong vài mili giây hay dịch chuyển servo với độ dài xung 1000-2000 μ s chỉ trong micro giây.

- Dùng các mảng ma trận để lưu trữ, chuyển đổi xung điều khiển của các khớp.

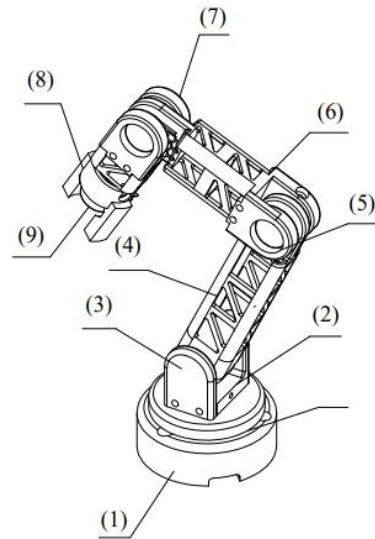
VD: biến joint0[arrayStep] với arrayStep là biến lưu trữ giá trị các mảng trong khoảng từ 1 đến 180 bước (tùy theo nhu cầu sử dụng), thì mảng joint0 có thể lưu đến 180 giá trị của xung điều khiển cho 1 khớp nhất định và khi khai báo nhiều mảng như vậy sẽ chiếm đáng kể bộ nhớ của module arduino.

Chương 4: KẾT QUẢ VÀ THẢO LUẬN

4.1 THIẾT KẾ MÔ HÌNH TAY MÁY



- 1-Đế cố định; 2-Đế xoay; 3-Servo MG996R;
4-Cụm đế khớp vai; 5-Khâu bắp tay;
6-Giá lò xo; 7-Gá khớp xoay-II;
8-Khâu khuỷu tay-I; 9-Khâu khuỷu tay-II;
10-Cụm lắp tay kẹp; 11-Cụm tay kẹp.

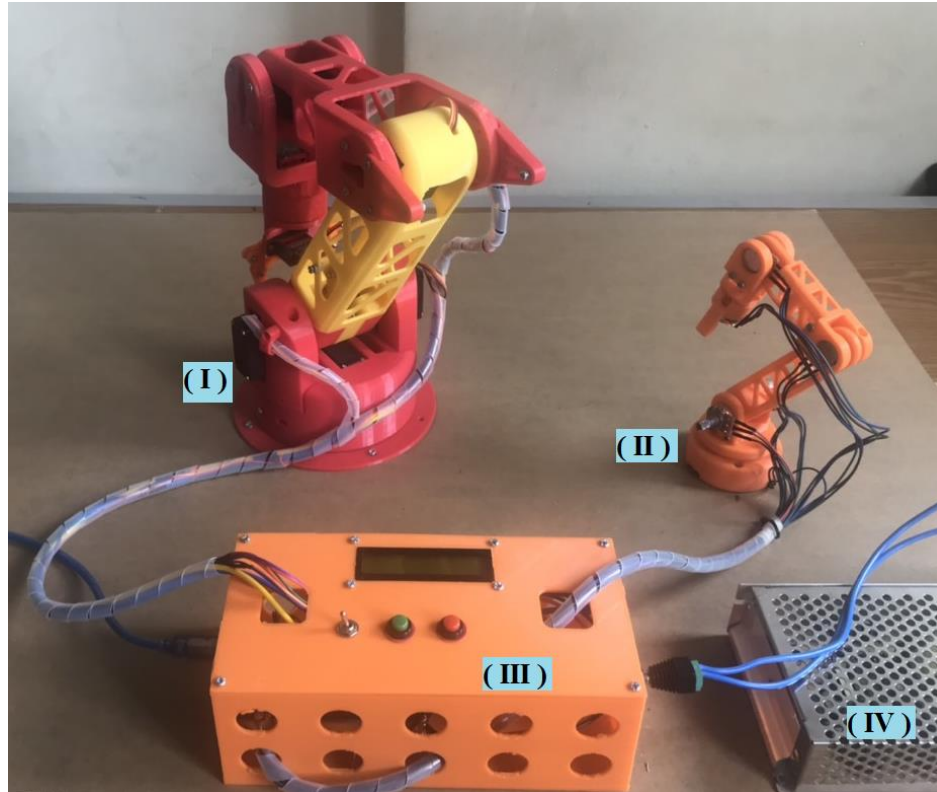


- (1)Đế cố định;(2) Đế xoay;(3)Giá đỡ trục;
(4) Khâu I;(5)Gá trục biến trở;(6) Khâu II;
(7) Khâu III; (8) Tay xoay;(9) biến trở

Hình 4.1 Nguyên lý cấu tạo mô hình gồm cánh tay máy và cánh tay điều khiển.

Việc thiết kế chế tạo mô hình ngoài việc xác định kích thước các khâu, khớp còn phải lựa chọn động cơ với độ chịu tải phù hợp(chi tiết xem phụ lục phần 2).

Mô hình bao gồm cánh tay máy 5 bậc tự do(I), cánh tay điều khiển(II), hộp điều khiển(III) và bộ nguồn(IV) (Hình 4.1).

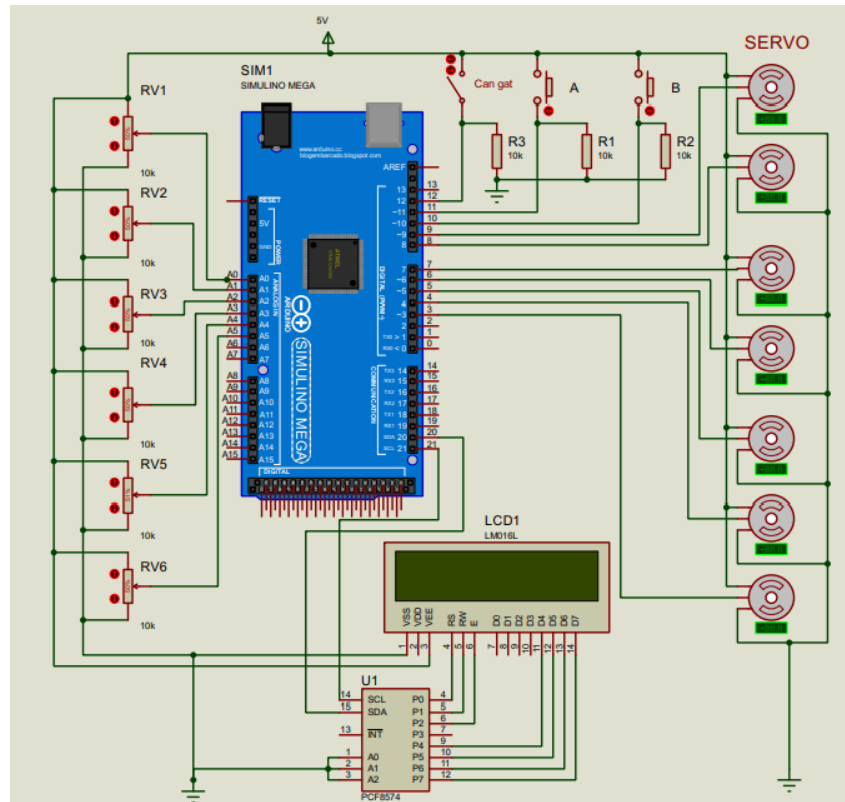


Hình 4.2 Mô hình thực tế của hệ thống .

Hệ thống gồm hai bộ nguồn cấp cho mạch điều khiển và cánh tay máy. Nguồn của mạch điều khiển được cấp từ bên ngoài (sạc dự phòng, laptop, adapter..). Cánh tay máy được duy trì bởi bộ nguồn 5V 30A.

Bộ điều khiển sẽ nhận tín hiệu từ cánh tay điều khiển hay các công tắc để, điều khiển, chuyển đổi giữa các chế độ của cánh tay máy.

4.2 THIẾT KẾ CHẾ TẠO MẠCH ĐIỀU KHIỂN:



Hình 4.3 Mạch điều khiển được mô tả lại qua phần mềm proteus

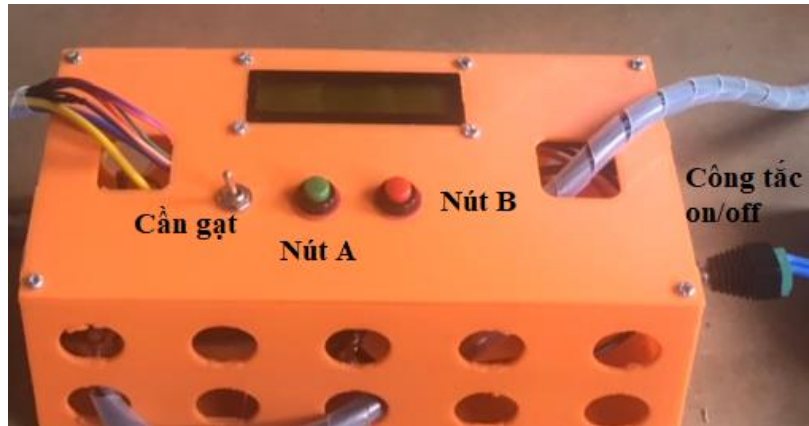
Bộ vi xử lý sẽ nhận tín hiệu analog từ các biến trở RV1, RV2... ở các khớp của cánh tay điều khiển rồi chuyển đổi thành xung điều khiển cho các servo. Các xung điều khiển này dùng để điều khiển cánh tay máy, đồng thời được lưu vào mảng theo thứ tự thao tác.

Khi chế độ chạy thao tác được kích hoạt thì các thao tác máy đã lưu được tính toán và xuất tín hiệu điều khiển cho các servo tại các khớp của tay máy.

Các công tắc điều khiển gồm nút A (XANH) nút B (DO), công tắc gạt và công tắc on/off.

- Nút A để lưu thao tác (nhấn 1 lần) hay chạy thao tác lưu (nhấn đúp).
- Nút B dùng để về vị trí tay máy so với cánh tay điều khiển (nhấn đơn) hay để xóa các thao tác đã lưu(nhấn đúp)(lưu ý: sau khi lưu hay xóa thao tác ta có thể nạp lại bằng nút A) (Hình 4.5)

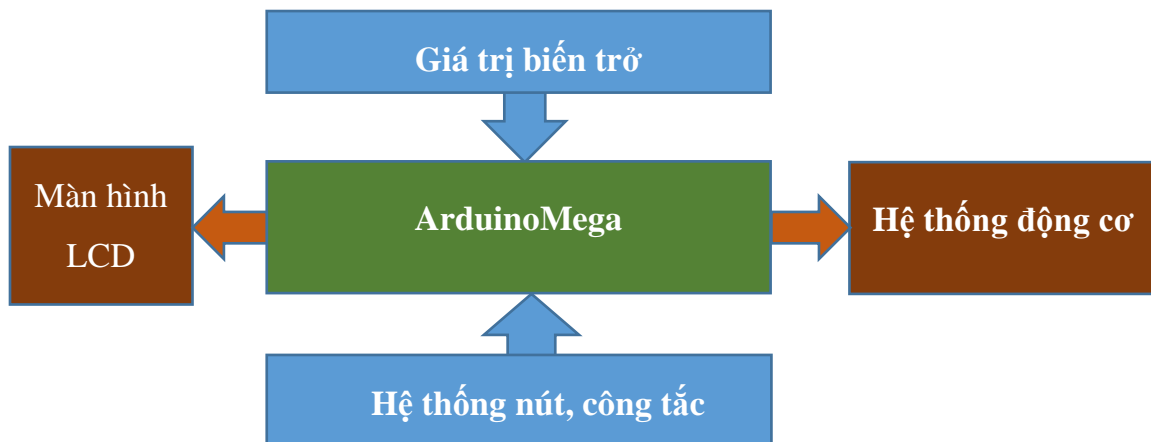
- Cần gạt để dừng tay máy khẩn cấp và công tắc on/off dùng để dẫn hay ngắt nguồn riêng của tay máy khi khởi động chương trình hay nạp chương trình mới.



Hình 4.4 Bộ điều khiển của mô hình.

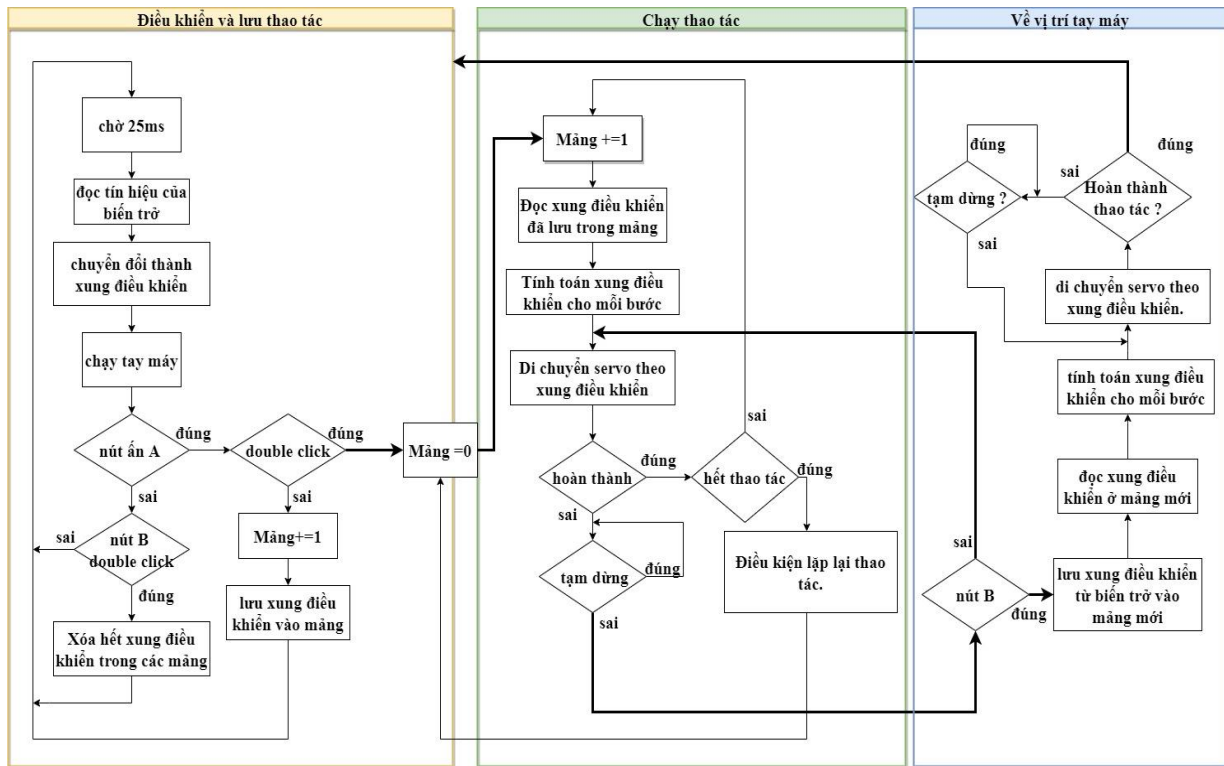
4.3 SƠ ĐỒ KHỐI CỦA HỆ THỐNG ĐIỀU KHIỂN:

Hệ thống gồm mạch điều khiển làm trung tâm, nhận tín hiệu điều khiển từ biến trở và công tắc điều khiển rồi xuất tín hiệu điều khiển ra màn hiển thị LCD và động cơ servo.



Hình 4.5 Sơ đồ khối hệ thống điều khiển

4.4 LƯU ĐỒ THUẬT TOÁN:



Hình 4.6 Lưu đồ thuật toán của chương trình điều khiển.

Hệ thống chương trình điều khiển gồm ba chức năng chính tương ứng với ba khung chương trình của lưu đồ thuật toán.

Ở trạng thái điều khiển và lưu thao tác thì chương trình luôn đọc tín hiệu điều khiển từ các biến trở để xuất xung điều khiển cho tay máy đồng thời liên tục quét trạng thái của các nút để lưu trữ các xung điều khiển của các khớp cho từng bước (Mảng).

Khi trạng thái chạy tay máy được kích hoạt thì biến “Mảng” được cài về không để chạy theo thứ tự các mảng đã lưu từ đầu chương trình. Mảng được lấy ra để tính toán băm xung cho các khớp tay máy chạy một cách đồng bộ .

Khi vòng lặp chạy tay máy kết thúc (nút B) thì chương trình đọc xung điều khiển xuất ra từ các biến trở hiện tại rồi tính toán xuất tín hiệu cho các khớp tay máy về vị trí như cánh tay điều khiển.

Khi nhấn đúp nút B thì quá trình reset các mảng đã lưu trước đó được thực thi để sẵn sàng bắt đầu cho phiên làm việc mới.

4.5 KHẢO NGHIỆM, ĐÁNH GIÁ HOẠT ĐỘNG HỆ THỐNG:

Bố trí khảo nghiệm độ phản hồi của cánh tay chính đối với cánh tay điều khiển bằng cách chuyển động từng khớp của cánh tay điều khiển.

Số lần Khớp	1	2	3	4	5	6	7	8	9	10
Hông	T	T	T	T	T	T	T	T	T	T
Vai	T	T	T	T	T	T	T	T	T	T
Khuỷu I	T	T	T	T	T	T	T	T	T	T
Khuỷu II	T	T	T	T	T	X	T	T	X	T
Cổ tay	T	T	T	T	T	T	T	T	T	T

Bảng 4.1 Khảo nghiệm độ phản hồi của các khớp tay máy chính so với tay điều khiển với 10 lần thử nghiệm trên mỗi khớp.

T: khớp tay máy phản hồi tốt.

X: khớp tay máy phản hồi không như ý muốn-như bị delay hay chạy loạn.

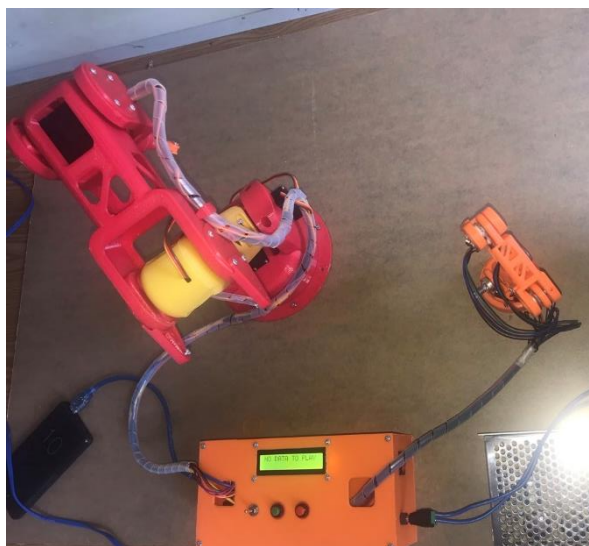
Vậy theo bảng khảo nghiệm trên thì các khớp hông, vai, khuỷu I và khớp cổ tay có độ phản hồi là 100%, tuy nhiên khớp khuỷu II độ phản hồi chỉ có 80%.

Độ chính xác về góc của cánh tay chính so với tay điều khiển thì theo nhà sản xuất dung sai của hai loại servo trên là 5 μ s so với khoảng xung điều khiển 1000 μ s cho cấp chính xác 0,5%. Vậy sai số đối với các khớp sử dụng MG996R là 0,6° với khoảng chuyển động tối đa 0°-120°. Đối với động cơ servo mg90s thì sai số là 0,9° với khoảng chuyển động tối đa là 0°-180°.

Chương 5: KẾT LUẬN VÀ ĐỀ NGHỊ

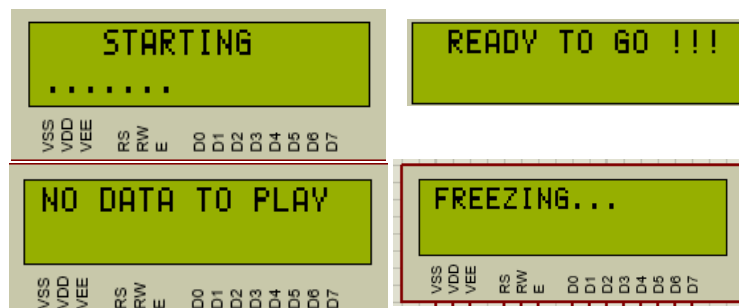
5.1 NHỮNG KẾT QUẢ ĐẠT ĐƯỢC:

- Hoàn thiện được mô hình cánh tay máy cũng như mô hình điều khiển.
- Giao diện người dùng tối giản, dễ tiếp cận.



Hình 5.1 Hệ thống tay máy đang hoạt động

- Tích hợp được nhiều chức năng điều khiển: điều khiển và lưu thao tác, chạy thao tác, về vị trí tay máy.
- Chương trình hoạt động tốt, hiển thị đầy đủ được trạng thái hoạt động của tay máy.



Hình 5.2 Biểu thị trạng thái khởi động, đang sẵn sàng hoạt động và thông báo không có thao tác nào đã lưu, ngưng hoạt động.



Hình 5.3 Biểu thị đang ghi thao tác 2-3; Thông báo có hai bước đã lưu; Hiển thị trạng thái đang chạy thao tác; Về vị trí tay máy; Thông báo đang xóa các thao tác lưu.

5.2 HẠN CHẾ CỦA ĐỀ TÀI:

- Kết cấu tay máy không chắc chắn ít tin cậy trong trường hợp tải trọng vật gấp lớn.
- Không áp dụng được động học thuận, động học ngược trong chương trình điều khiển.
- Vì điều khiển bằng tay nên độ chính xác thấp, không đáng tin cậy.
- Chương trình vận hành đơn giản, không có ứng dụng kết nối không dây, Internet.

5.3 HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI:

Vì đề tài còn nhiều hạn chế nên việc khắc phục những hạn chế trên là một trong những hướng phát triển của đề tài này.

Ngoài ra phương pháp điều khiển mẫu lặp trên còn có hướng phát triển tối ưu là tích hợp quá trình phản hồi analog trực tiếp từ các khớp của tay máy mà không qua cánh tay điều khiển. Hơn nữa việc ứng dụng các cảm biến mô men lên các khớp của tay máy cũng là một hướng đi vượt trội, bắt kịp xu thế.

TÀI LIỆU THAM KHẢO

- [1] F. “IFR presents World Robotics Report 2020,” IFR Press Room, 2020.
- [2] D. A. và P. R. , “Robots and Jobs: Evidence from US Labor Markets,” tập 128, 2020.
- [3] “ABB,” ABB Asea Brown Boveri Ltd, 12 tháng 4 2021. [Trực tuyến].
Available: <https://new.abb.com/news/detail/76652/prsrl-abbs-new-gofa-cobot-wins-prestigious-best-of-the-best-red-dot-award>.
[Đã truy cập tháng 4 2021].
- [4] “ABB,” ABB Asea Brown Boveri Ltd, 2021. [Trực tuyến].
Available: <https://new.abb.com/products/robotics/collaborative-robots/crb-15000>.
[Đã truy cập tháng 4 2021].
- [5] UR3e, “Universal Robots,” 2021. [Trực tuyến].
Available: <https://www.universal-robots.com/products/ur3-robot/>.
[Đã truy cập tháng 4 2021].
- [6] “Động học tay máy,” trong Kỹ Thuật robot, Hà Nội, Nhà xuất bản khoa học và kỹ thuật, 2004.
- [7] “Components101,” 03 April 2019. [Trực tuyến].
Available: <https://components101.com/motors/mg996r-servo-motor-datasheet>.
[Đã truy cập tháng 4 2021].

- [8] "Components101," March 2019. [Trực tuyến].
Available: <https://components101.com/motors/mg90s-metal-gear-servo-motor#>.
[Đã truy cập tháng 4 2021].
- [9] "Components101," August 2017. [Trực tuyến].
Available: <https://components101.com/displays/16x2-lcd-pinout-datasheet>.
[Đã truy cập tháng 4 2021].
- [10] "ARDUINO KIT," December 2020. [Trực tuyến].
Available: <https://arduinokit.vn/giao-tiep-i2c-lcd-arduino/>.
[Đã truy cập tháng 4 2021].
- [11] "RobotShop," [Trực tuyến].
Available:
<https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf>.
[Đã truy cập tháng 4 2021].
- [12] "Wikipedia," [Trực tuyến].
Available: https://en.wikipedia.org/wiki/Arduino_IDE.
[Đã truy cập tháng 4 2021].

PHỤ LỤC

I/ Chi tiết phần động học ngược tay máy chương 2:

$$\text{Atan2}\left(\frac{y}{x}\right)=\begin{cases} \arctan\left(\frac{y}{x}\right) \text{ khi } x > 0 \\ \arctan\left(\frac{y}{x}\right) + \pi \text{ khi } x < 0; y \geq 0 \\ \arctan\left(\frac{y}{x}\right) - \pi \text{ khi } x < 0; y < 0 \\ \frac{\pi}{2} \text{ khi } x = 0; y > 0 \\ -\frac{\pi}{2} \text{ khi } x = 0; y < 0 \\ \text{không xác định khi } x = 0; y = 0 \end{cases}$$

Để hai mặt phẳng song song thì véc tơ pháp tuyến của hai mặt phẳng phải cùng phương. Khi ứng dụng vào phương trình mặt phẳng ta có điều kiện sau:

Cho hai mặt phẳng A và B tương ứng với hai phương trình

$$A: ax+by+cz+d=0$$

$$B: a'x+b'y+c'z+d'=0$$

Với $a, b, c; a', b', c'$ lần lượt là các thành phần của hai véc tơ pháp tuyến thuộc hai mp.

Để hai mặt phẳng A và B có quan hệ song song thì

$$\frac{a}{a'} = \frac{b}{b'} = \frac{c}{c'} = |k|$$

Với k là hệ số tỉ lệ, hai vecto cùng chiều thì $k > 0$ và ngược lại.

Vậy để mặt phẳng gấp của tay kẹp và mặt phẳng gấp của vật song song nhau thì véc tơ pháp của hai mặt phẳng tương ứng với vecto đơn vị y_c trong ma trận quay của vật hay y_5 trong ma trận chuyển đổi thuần nhất của tay máy.

Ta có ma trận quay của vật gấp trong không gian được mô tả bởi các phép quay:

Quay hệ tọa độ gốc quanh trục Oz một góc α ta được hệ tọa độ $Ox_1y_1z_1$.

Tiếp tục quay quanh trục O_1x_1 một góc β ta được hệ tọa độ $Ox_2y_2z_2$.

Quay hệ $Ox_2y_2z_2$ quanh trục Oy_2 một góc φ ta được hệ $Ox_cy_cz_c$.

$$R = \begin{bmatrix} C_\alpha & -S_\alpha & 0 \\ S_\alpha & C_\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C_\beta & -S_\beta \\ 0 & S_\beta & C_\beta \end{bmatrix} \begin{bmatrix} c_\varphi & 0 & s_\varphi \\ 0 & 1 & 0 \\ -s_\varphi & 0 & c_\varphi \end{bmatrix} = \begin{bmatrix} r_{11} & -s_\alpha c_\beta & r_{13} \\ r_{21} & c_\alpha c_\beta & r_{23} \\ r_{31} & s_\beta & r_{33} \end{bmatrix}$$

Ta có $y_c(-s_\alpha c_\beta, c_\alpha c_\beta, s_\beta)$; $y_5(S_1 S_5 S_\emptyset - C_1 C_5, -C_1 S_5 S_\emptyset - S_1 C_5, S_5 C_\emptyset)$

Vì y_c và y_5 đều là vectơ đơn vị nên hệ số $k=1$.

$$\frac{-s_\alpha c_\beta}{S_1 S_5 S_\emptyset - C_1 C_5} = \frac{c_\alpha c_\beta}{-C_1 S_5 S_\emptyset - S_1 C_5} = \frac{s_\beta}{S_5 C_\emptyset} = |1|.$$

Từ biểu thức trên ta có hệ phương trình:

$$\begin{cases} S_1 S_5 S_\emptyset - C_1 C_5 = -s_\alpha c_\beta & (p1) \\ -C_1 S_5 S_\emptyset - S_1 C_5 = c_\alpha c_\beta & (p2) \\ S_5 C_\emptyset = s_\beta & (p3) \end{cases} \quad (E1) \text{ (với trường hợp hai vector cùng chiều).}$$

$$\begin{cases} S_1 S_5 S_\emptyset - C_1 C_5 = s_\alpha c_\beta \\ -C_1 S_5 S_\emptyset - S_1 C_5 = -c_\alpha c_\beta \\ S_5 C_\emptyset = -s_\beta \end{cases} \quad (E2) \text{ (trường hợp hai vector ngược chiều).}$$

Giải hệ (E1)

Từ (p2) ta được $C_5 = \frac{-C_\alpha c_\beta - C_1 S_5 S_\emptyset}{S_1}$ (p2') thế (p2') vào (p1) ta được:

$$S_5 S_\emptyset = -C_\beta S_1 S_\alpha - C_\beta C_1 C_\alpha \quad (p4).$$

Chia hai vế của hai pt (p4) và (p3) ta có $\frac{S_\emptyset}{C_\emptyset} = -\frac{C_\beta C_{\alpha-1}}{S_\beta}$.

Cộng bình phương hai vế của (p3) và (p4) ta được: $S_5^2 = S_\beta^2 + C_\beta^2 C_{1-\alpha}^2$.

Thay thế giá trị $S_5 S_\emptyset$ ở (p4) vào (p1) hoặc (p2) ta được $C_5 = C_\beta S_{\alpha-1}$.

Với hệ (E2) thì cách giải tương tự như hệ (E1)

Với các giá trị góc θ_2 , θ_3 và θ_4 ta xác định như trường hợp cánh tay robot cơ cấu ba khâu phẳng.

Ta giải các biến θ_2 , θ_3 trong mặt phẳng Oxz (hay trong mp Oyz)

$$\begin{cases} P_z = d_5 S_{\emptyset} - a_3 C_{23:\frac{2\pi}{3}} + a_2 S_{2:\frac{2\pi}{3}} + d_1 \\ P_x = S_1 (-d_5 C_{\emptyset} - a_3 S_{23:\frac{2\pi}{3}} - a_2 C_{2:\frac{2\pi}{3}}) \end{cases}$$

$$\text{Đặt } P_{w_x} = \frac{P_x}{S_1} + d_5 C_{\emptyset}, P_{w_z} = P_z - d_1 - d_5 S_{\emptyset}.$$

$$\text{Ta được hệ } \begin{cases} P_{w_x} = -a_3 S_{23:\frac{2\pi}{3}} - a_2 C_{2:\frac{2\pi}{3}} \\ P_{w_z} = -a_3 C_{23:\frac{2\pi}{3}} + a_2 S_{2:\frac{2\pi}{3}} \end{cases} \quad (F1)$$

Cộng bình phương hai vế của hệ trên ta được:

$$S_3 = \frac{P_{w_x}^2 + P_{w_z}^2 - a_2^2 - a_3^2}{2a_2 a_3}$$

$$\rightarrow C_3 = \pm \sqrt{1 - S_3^2} \text{ (ứng với hai tư thế khác nhau của cánh tay máy).}$$

Ta biến đổi từ hệ (F1) được:

$$\begin{cases} P_{w_x} = -C_3 a_3 S_{2:\frac{2\pi}{3}} - C_{2:\frac{2\pi}{3}} (a_3 S_3 + a_2) \\ P_{w_z} = -C_3 a_3 C_{2:\frac{2\pi}{3}} + S_{2:\frac{2\pi}{3}} (S_3 a_3 + a_2) \end{cases}$$

Sau khi giải hệ trên ta được:

$$S_{2:\frac{2\pi}{3}} = \frac{P_{w_z} (a_3 S_3 + a_2) - a_3 C_3 P_{w_x}}{P_{w_x}^2 + P_{w_z}^2} \quad (2.13)$$

$$C_{2:\frac{2\pi}{3}} = - \frac{P_{w_x} (a_3 S_3 + a_2) + a_3 C_3 P_{w_z}}{P_{w_x}^2 + P_{w_z}^2} \quad (2.14)$$

Lưu ý rằng động học ngược tay máy trong bài chưa được chức thực tính chính xác, hiệu quả cho cánh tay máy.

II/ Tính toán xác định lực tác động lên các khớp tay máy

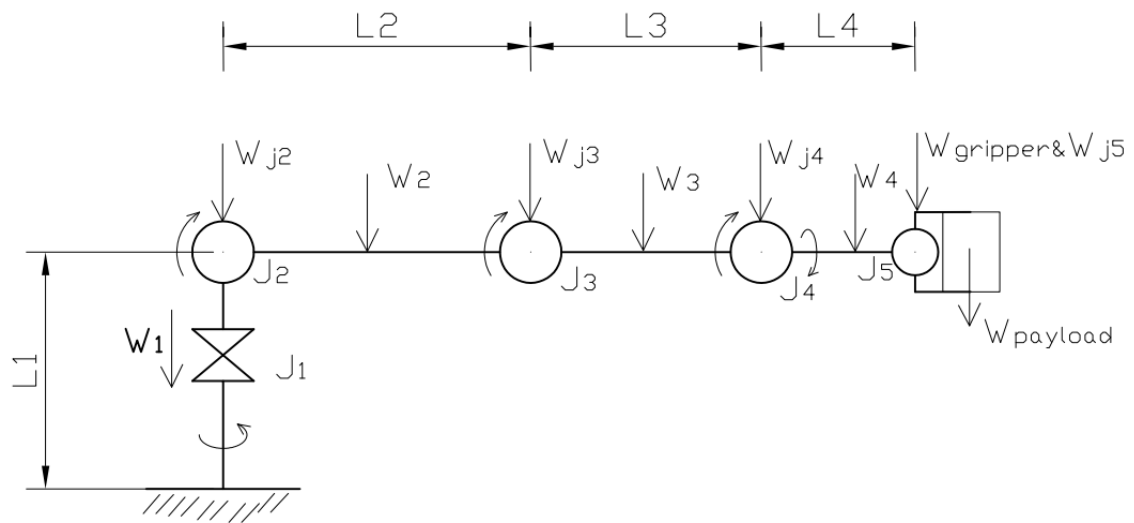
Để tạo nên chuyển động của các liên kết của một cánh tay máy thì trên mỗi khớp của cánh tay robot được tích hợp cố định một bộ truyền động(trong đề tài chúng tôi sử dụng động cơ servo). Bộ truyền động với mục đích gây mô men xoắn ở khớp xoay nhằm đánh bại lực cản của liên kết (hay còn gọi là khâu) và tạo nên chuyển động. Và lực cản của bất kỳ liên kết nào ảnh hưởng đến chuyển động của tay máy cũng đều chịu sự tác động của trọng lực và lực quán tính.

- Tác động của lực cản là trọng lực lên các khớp của cánh tay robot

Vì bản chất của trọng lực là tác động lên vật thể bằng cách kéo chúng về tâm của vật (cụ thể vật ta đang xét là Trái Đất) có khối lượng lớn hơn với gia tốc và khối lượng của chính vật (các liên kết của tay máy) bị tác động, do đó gây nên lực cản đối với các khớp của tay máy khi hoạt động. Chính vì lẽ đó mà lượng mô men xoắn từ bộ truyền động sử dụng tại các khớp của tay máy phải vượt qua được lực cản gây ra bởi trọng lực. Vì vậy việc cần thiết cho việc thiết kế và lựa chọn bộ truyền động phù hợp cho mỗi khớp xoay là tính toán, xác định lượng mômen cản mà trọng lực gây ra ở các liên kết đối với mỗi khớp xoay.

Mô men cản tác động lên khớp xoay phụ thuộc rất nhiều vào tư thế hoạt động của cánh tay máy. Trực quan ta thấy mô men cản do trọng lực tác động lên khớp vai lớn hơn nhiều khi tay máy hoạt động với tư thế duỗi dài theo phương ngang. Vì khi tính toán xác định thông số cho việc lựa chọn thiết kế ta cần lựa chọn thông số an toàn, thế nên ta tính toán trường hợp lượng mô men cản gây ra lớn nhất khi mà tay máy hoạt động trong điều kiện tệ nhất (tay máy với tư thế duỗi dài theo phương ngang) (Corke,2011).

Ở Hình 2.9 và 2.10 ta có bản ghi độ dài của các liên kết (các khâu của cánh tay máy) trong khi Ảnh 9.1 thể hiện giản đồ của cánh tay máy hoạt động với tư thế duỗi thẳng theo phương ngang.



Ảnh 9.1 Giản đồ phân tích lực tác động lên tay máy ở thể duỗi thẳng theo phương ngang.

Trong Ảnh 9.1 ta có:

J₁: Khớp trụ (khớp hông)

W₂: Trọng lượng của liên kết L₂.

J₂: Khớp vai.

W₃: Trọng lượng của liên kết L₃.

J₃: Khớp khuỷu I.

W₄: Trọng lượng của liên kết L₄.

J₄: Khớp khuỷu II.

W_{payload}: Trọng lượng của vật gắp.

J₅: Khớp xoay cổ tay.

W_{gripper}: Trọng lượng của tay kẹp.

W₁: Trọng lượng của liên kết L₁.

W_{j2}: Trọng lượng của khớp vai.

W_{j3}: Trọng lượng của khớp khuỷu I.

W_{j4}: Trọng lượng của khớp khuỷu II.

W_{j5}: Trọng lượng của khớp xoay cổ tay.

Việc tính toán mô men cản tác động lên từng khớp do tác dụng của trọng lực như sau:

Mô men cản tại khớp J₁ do tác động của trọng lực: T_{1g}

Mô men cản tại khớp J₂ do tác động của trọng lực: T_{2g}

Mô men cản tại khớp J_3 do tác động của trọng lực: T_{3g}

Mô men cản tại khớp J_4 do tác động của trọng lực: T_{4g}

Mô men cản tại khớp J_5 do tác động của trọng lực: T_{5g}

Với độ dài các liên kết ta có L_1, L_2, \dots và l_1, l_2, \dots tương ứng với tỉ lệ giữa chiều dài của liên kết với khoảng cách của trục đầu tiên (khớp nối đầu) của liên kết với trọng tâm của liên kết đó.

Ta có:

$T_{1g}=0$ Nm vì khi xoay khớp trụ(khớp hông) của tay máy thì không gây ra chuyển động của bất kỳ liên kết nào trong mặt phẳng đứng (tức là chống lại trọng lực).

$$T_{2g}=W_2\left(\frac{L_2}{l_2}\right)+W_{j3}L_2+W_3\left(L_2 + \frac{L_3}{l_3}\right)+W_{j4}(L_2+L_3)+W_4\left(L_2 + L_3 + \frac{L_4}{l_4}\right)+ \\ (W_{gripper}+W_5+W_{payload})(L_2+L_3+L_4).$$

$$T_{3g}= W_3\frac{L_3}{l_3}+W_{j4}L_3+W_4\left(L_3 + \frac{L_4}{l_4}\right)+ (W_{gripper}+W_5+W_{payload})(L_3+L_4).$$

$$T_{4g}= W_4\frac{L_4}{l_4} + (W_{gripper}+W_5+W_{payload})L_4.$$

$T_{5g}= 0$ Nm -vì chuyển động xoay của khớp cổ tay không bị tác động bởi trọng lực (với vật gắp đơn giản).

- **Tác động của lực cản là quán tính:**

Sự cản trở lên một vật khi vật bắt đầu thay đổi trạng thái từ trạng thái tĩnh chuyển sang chuyển động quay được gọi là mô men quán tính (I). Vậy lực cản tác động lên vật khi vật đang thay đổi trạng thái tĩnh sang chuyển động quay quanh một trục được đo lường bằng cách tính toán mô men quán tính của nó đối với trục cụ thể đó. Mô men quán tính thực hiện vai trò tương tự đối với chuyển động quay như cách mà khối lượng làm đối với chuyển động tịnh tiến. Mô men quán tính mô tả việc gia tốc góc được tạo ra bởi mô men xoắn tác dụng (Jha,2006).

Như vậy, với gia tốc góc bất kỳ của liên kết nào của tay máy, quán tính của liên kết dẫn đến mô men cản tác dụng lên khớp xoay, mỗi nối đầu của liên kết. Giá trị của mô men xoắn này (T_i) được tính bằng phép nhân giữa mô men quán tính của liên kết và gia tốc góc của nó, α (Jha, 2006).

$$T_i = I\alpha$$

Theo Benenson cùng cộng sự (2006), mô men quán tính của một vật phụ thuộc vào hình dáng và phân bố khối lượng cũng như hướng của trục quay, do đó đòi hỏi áp dụng cho mỗi liên kết tay máy một công thức tính khác nhau.

Vì các liên kết của tay máy với biên dạng hình học không đơn giản nên các công thức ứng với từng liên kết cũng trở nên phức tạp, do đó một quyết định được đưa ra là xác định mô men quán tính bằng cách sử dụng chương trình thiết kế với sự hỗ trợ của máy tính-CAM (Computer Aided Modelling). Biên dạng 3D của các chi tiết được thiết kế và phân tích bằng phần mềm SolidWorks. Mô men quán tính của các bộ phận khác nhau của tay máy đã được xác định bằng tính năng “Mass Properties” của SolidWorks.

Ảnh 9.2 minh họa việc sử dụng phần mềm CAD để xác định mô men quán tính của liên kết L2 đối với hệ tọa độ X, Y và Z (tức là I_{xx} , I_{yy} , I_{zz}) được chọn như trong hình.

Ta có các giá trị từ ảnh minh họa:

$$I_{xx} = 1475876.05 \text{ g} \cdot \text{mm}^2$$

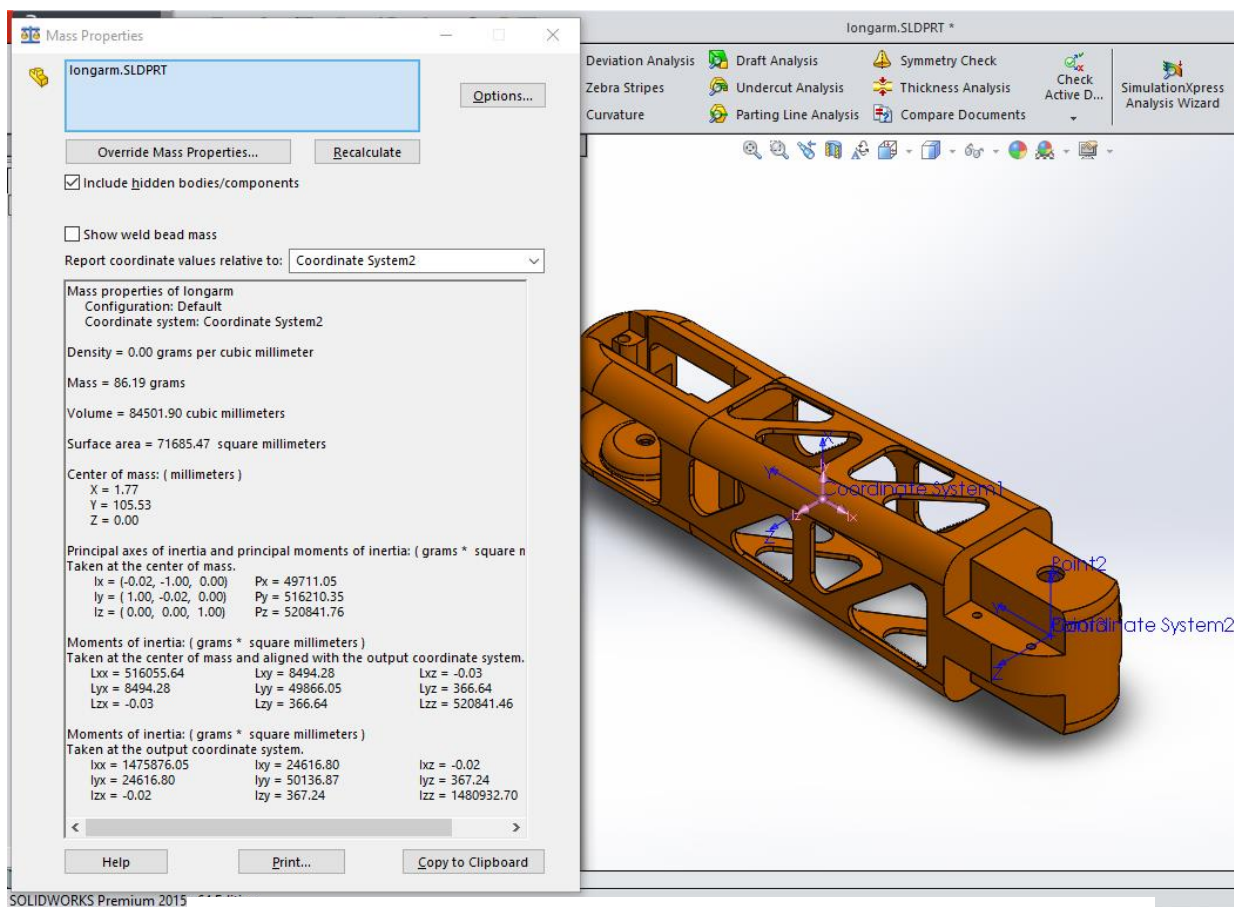
$$I_{yy} = 50136.87 \text{ g} \cdot \text{mm}^2$$

$$I_{zz} = 1480932.7 \text{ g} \cdot \text{mm}^2$$

Sử dụng cùng một quy trình ta xác định được I_{xx} , I_{yy} và I_{zz} cho các thành phần khác nhau của cánh tay robot.

Xác định việc thiết kế cánh tay máy với mỗi động cơ quay với tốc độ góc 1 độ trên mỗi 20 milli giây tương đương với vận tốc góc 50 độ/giây hay 0.873 rad/s. Mặc dù động cơ được thiết kế với tốc độ góc nhanh hơn nhưng với tốc độ vừa phải được chọn sẽ giúp giảm tỷ lệ hao mòn trên khớp cánh tay máy.

Ngoài ra, tay máy được lập trình thiết kế sao cho mỗi liên kết chuyển động từ trạng thái nghỉ (0 rad/s) sang trạng thái hoạt động với vận tốc thiết kế là 0.873 rad/s trong vòng 0.5s, chuyển sang gia tốc góc ta được 1.764 rad/s².



Ảnh 9.2 Minh họa tính năng Mass Properties để xác định mô men quán tính cho hệ tọa độ định trước của liên kết L2.

Việc tính toán mô men cản tác động lên từng khớp do quán tính được tiến hành như sau:

Mô men cản tác động lên khớp J1 do quán tính: T_{1i}

Mô men cản tác động lên khớp J2 do quán tính: T_{2i}

Mô men cản tác động lên khớp J3 do quán tính: T_{3i}

Mô men cản tác động lên khớp J4 do quán tính: T_{4i}

Mô men cản tác động lên khớp J5 do quán tính: T_{5i}

Mỗi khớp được gắn cố một động cơ. Vì vậy ta đặt:

Khối lượng của động cơ ở khớp vai(J2): m_{sm}

Khối lượng của động cơ ở khớp khuỷu I(J3): m_{em1}

Khối lượng của động cơ ở khớp khuỷu II(J4): m_{em2}

Khối lượng của động cơ ở khớp cổ tay(J5): m_{wm}

Khối lượng của động cơ ở tay kẹp: m_{gm}

m_{L1} : Khối lượng của liên kết L_1

m_{L2} : Khối lượng của liên kết L_2

m_{L3} : Khối lượng của liên kết L_3

m_{L4} : Khối lượng của liên kết L_4

$m_{gripper}$: Khối lượng của bộ lắp ráp tay kẹp.

I_{sm} : mô men quán tính của động cơ khớp vai

I_{em1} : mô men quán tính của động cơ khớp khuỷu I

I_{em2} : mô men quán tính của động cơ khớp khuỷu II

I_{wm} : mô men quán tính của động cơ khớp cổ tay

I_{gm} : mô men quán tính của động cơ ở tay kẹp

$I_{gripper}$: mô men quán tính của bộ lắp ráp tay kẹp

I_{base} : mô men quán tính của bộ quay và cụm giá đỡ động cơ.

I_{L2} : mô men quán tính của liên kết L_2 .

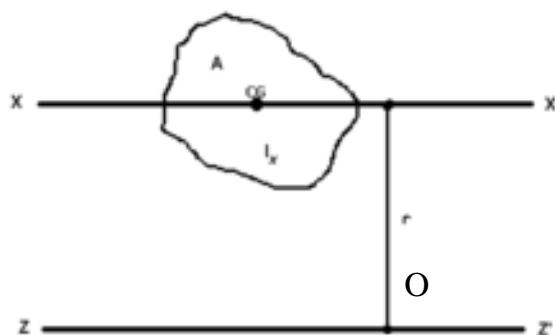
I_{L3} : mô men quán tính của liên kết L3.

I_{L4} : mô men quán tính của liên kết L4.

Các giá trị mô men quán tính ta thu thập từ phần mềm CAD là so với hệ trục đi qua trọng tâm của vật thể. Trong trường hợp cánh tay máy, mô men quán tính đến việc bộ phận đang xét quay quanh một điểm trục(cụ thể là khớp tay máy) nằm cách khối tâm một khoảng. Để định lượng mô men quán tính đối với các trục khớp thì định lý trục quay song song đã được áp dụng.

Giả sử trục quay z đi qua điểm O của vật rắn A, còn trục thứ hai đi qua khối tâm G và giả sử hai trục này song song với nhau(Ảnh 9.3). Gọi r là khoảng cách giữa hai trục, ta có công thức:

$$I_O = I_G + mr^2$$



Ảnh 9.3 Minh họa cho định lý trục quay song song

Trong đó :

I_O là mô men quán tính của vật đối với trục quay đi qua điểm O

I_G là mô men quán tính của vật đối với trục quay đi qua khối tâm G

m là khối lượng của vật

Từ định lý trên ta tính được mô men cản do quán tính của từng khớp như sau(Lưu ý rằng các giá trị sau được tính trong trường hợp cánh tay máy duỗi thẳng theo phương ngang):

$$T_{li}=[(I_{zz,base}+m_{base}r_{11}^2)+(I_{zz,sm}+m_{sm}r_{12}^2)+(I_{zz,L2}+m_{L2}r_{13}^2)+(I_{zz,em1}+m_{em1}r_{14}^2)+ \\ (I_{zz,L3}+m_{L3}r_{15}^2)+(I_{zz,em2}+m_{em2}r_{16}^2)+(I_{zz,L4}+m_{L4}r_{17}^2)+(I_{zz,wm}+m_{wm}r_{18}^2)+(I_{zz,gm}+m_{gm}r_{19}^2)+ \\ (I_{zz,gripper}+m_{gripper}r_{1a}^2)]\alpha.$$

r_{11} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của khối trụ.

r_{12} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của động cơ khớp vai.

r_{13} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của liên kết L_2 .

r_{14} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của động cơ khớp khuỷu I.

r_{15} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của liên kết L_3 .

r_{16} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của động cơ khớp khuỷu II.

r_{17} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của liên kết L_4 .

r_{18} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của động cơ khớp cổ tay.

r_{19} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của động cơ tay kẹp.

r_{1a} : khoảng cách vuông góc giữa trục xoay chính của khớp trụ- J_1 và trục z đi qua trọng tâm của bộ tay kẹp.

$$T_{2i} = [(I_{xx,L2} + m_{L2}r_{21}^2) + (I_{xx,em1} + m_{em1}r_{22}^2) + (I_{xx,L3} + m_{L3}r_{23}^2) + (I_{xx,em2} + m_{em2}r_{24}^2) + (I_{xx,L4} + m_{L4}r_{25}^2) + (I_{xx,wm} + m_{wm}r_{26}^2) + (I_{xx,gm} + m_{gm}r_{27}^2) + (I_{xx,gripper} + m_{gripper}r_{28}^2)]\alpha.$$

r_{21} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của liên kết L_2 .

r_{22} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của động cơ khớp khuỷu I.

r_{23} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của liên kết L_3 .

r_{24} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của động cơ khớp khuỷu II.

r_{25} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của liên kết L_4 .

r_{26} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của động cơ khớp xoay.

r_{27} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của động cơ tay kẹp.

r_{28} : khoảng cách vuông góc giữa trục xoay chính của khớp vai- J_2 và trục x đi qua trọng tâm của bộ tay kẹp.

$$T_{3i} = [(I_{xx,L3} + m_{L3}r_{31}^2) + (I_{xx,em2} + m_{em2}r_{32}^2) + (I_{xx,L4} + m_{L4}r_{33}^2) + (I_{xx,wm} + m_{wm}r_{34}^2) + (I_{xx,gm} + m_{gm}r_{35}^2) + (I_{xx,gripper} + m_{gripper}r_{36}^2)]\alpha.$$

r_{31} : khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu- J_3 và trục x đi qua trọng tâm của liên kết L_3 .

r_{32} : khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu- J_3 và trục x đi qua trọng tâm của động cơ khớp khuỷu II.

r₃₃: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₃ và trục x đi qua trọng tâm của liên kết L₄.

r₃₄: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₃ và trục x đi qua trọng tâm của động cơ khớp cổ tay.

r₃₅: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₃ và trục x đi qua trọng tâm của động cơ kẹp.

r₃₆: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₃ và trục x đi qua trọng tâm của bộ tay kẹp.

$$T_{4i} = [(I_{xx,L4} + m_{L4}r_{41}^2) + (I_{xx,wm} + m_{wm}r_{42}^2) + (I_{xx,gm} + m_{gm}r_{43}^2) + (I_{xx,gripper} + m_{gripper}r_{44}^2)]\alpha.$$

r₄₁: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₄ và trục x đi qua trọng tâm của liên kết L₄.

r₄₁: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₄ và trục x đi qua trọng tâm của liên kết L₄.

r₄₁: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₄ và trục x đi qua trọng tâm của liên kết L₄.

r₄₁: khoảng cách vuông góc giữa trục xoay chính của khớp khuỷu-J₄ và trục x đi qua trọng tâm của liên kết L₄.

$$T_{5i} = [(I_{yy,gm} + m_{gm}r_{51}^2) + (I_{yy,gripper} + m_{gripper}r_{52}^2)]\alpha.$$

r₅₁: khoảng cách vuông góc giữa trục xoay chính của khớp cổ tay J₅ và trục y đi qua trọng tâm của động cơ tay kẹp.

r₅₂: khoảng cách vuông góc giữa trục xoay chính của khớp cổ tay J₅ và trục y đi qua trọng tâm của bộ tay kẹp.

Như vậy với các thông số mô men cản do trọng lực và lực quán tính đã được phân tích tính toán thì ta có được tổng mô men cản tác động lên mỗi khớp: $T=T_1+T_g$.

Và với T tìm được ta có thể dễ dàng lựa chọn động cơ hay thiết kế bộ truyền động phù hợp.

VD: ta có tổng mô men cản của khớp vai = $T_{2g}+T_{2i}$

$$T_{2g}=0.5903\text{Nm}$$

$$T_{2i}=0.0152\text{Nm}$$

→ Tổng mô men cản của khớp vai = 0.6055Nm hay $6.055\text{kg}\cdot\text{cm}$

Vậy ta lựa chọn các loại động cơ hay thiết kế bộ truyền động sản sinh ra mô men xoắn lớn hơn $6.055\text{kg}\cdot\text{cm}$. Đề tài sử dụng servo MG996R với mô men xoắn $9.6\text{kg}\cdot\text{cm}$ ở điều kiện điện áp 4.8V(DC) .

Lưu ý rằng các số liệu trong ví dụ trên chỉ là giả lập. Độ chính xác phụ thuộc rất nhiều vào cách đo lường, thiết kế, gán các tính chất về vật liệu của các liên kết hay bộ phận tay máy cho chương trình CAD.

- **Lựa chọn động cơ servo cho tay máy:**

Với tay kẹp sử dụng động cơ MG90S cho bộ tay gấp- với vật gấp nhỏ gọn và khối lượng tối đa là 80g.

Với khớp cổ tay đề tài sử dụng động cơ MG90S.

Khớp khuỷu I và khuỷu II sử dụng động cơ MG996R.

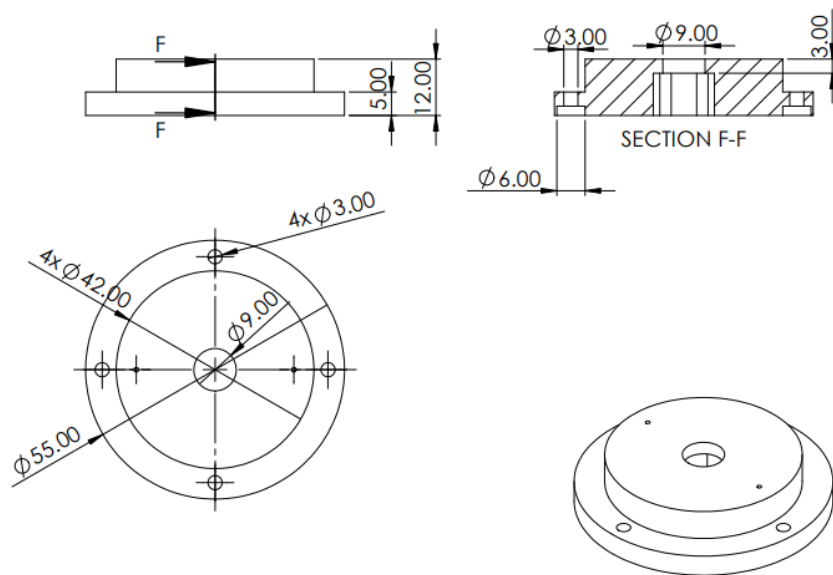
Ở khớp vai tích hợp hai động cơ MG996R.

Khớp hông sử dụng động cơ MG996R.

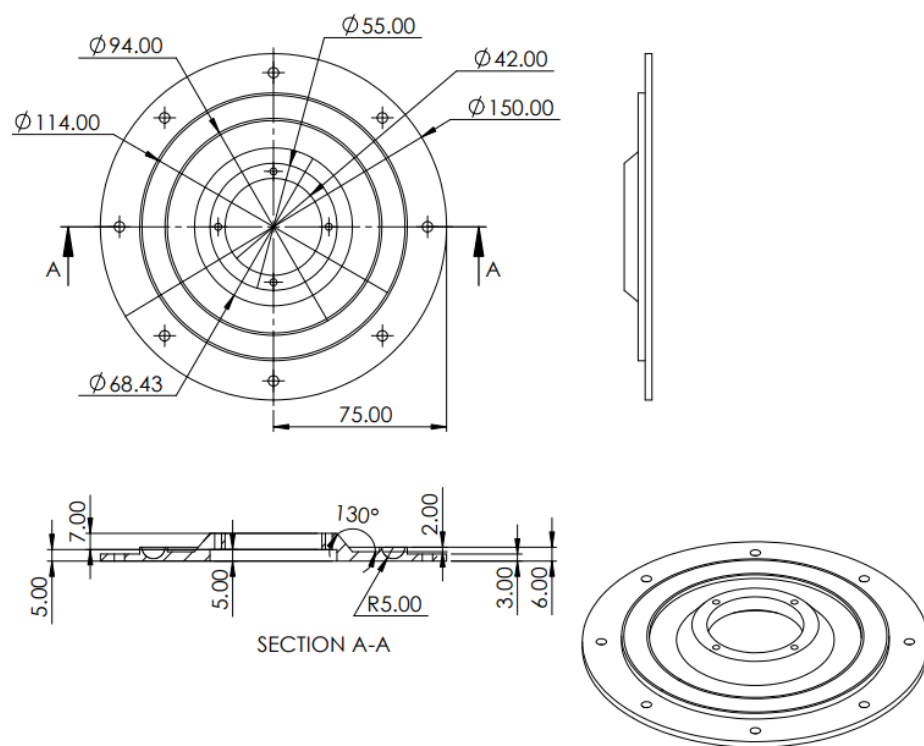
III/ Bản vẽ các chi tiết tay máy

STT	Tên chi tiết	Số lượng
1	Gá khớp xoay-I	1
2	Đế cố định	1
3	Vòng bi	1
4	Đế xoay	1
5	Cụm đế khớp vai	2
6	Cụm gá khớp vai	2
7	Khâu bấp tay	1
8	Khâu khuỷu tay-I	1
9	Khâu khuỷu tay-II	1
10	Gá khớp xoay-II	2
11	Trục đỡ khớp xoay	2
12	Vòng bi 6*17*6mm	2
13	Gá khớp xoay-III	1
14	Cụm lắp tay kẹp	1
15	Chi tiết kẹp 1	2
16	Chi tiết kẹp 2	2
17	Chi tiết kẹp 3	2
18	Chi tiết kẹp 4	2
19	Giá lò xo	2
20	Bi sắt 10mm	10
21	Động cơ servo MG996R	5
22	Động cơ servo MG90S	2

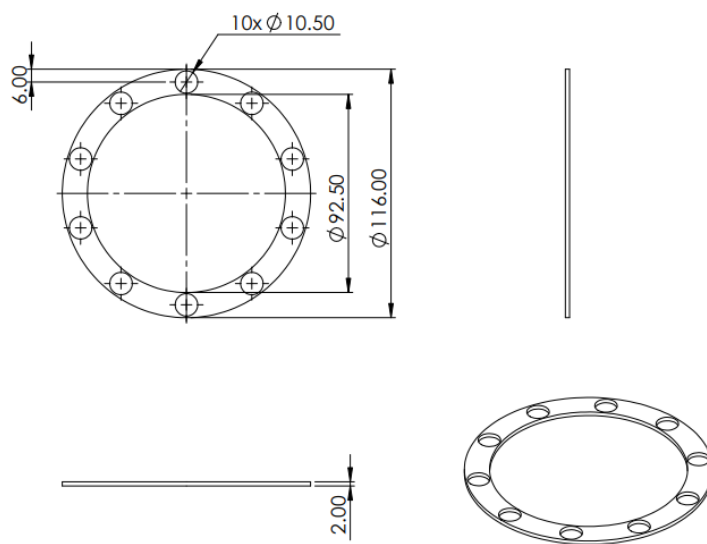
Bảng liệt kê các chi tiết cấu tạo tay máy



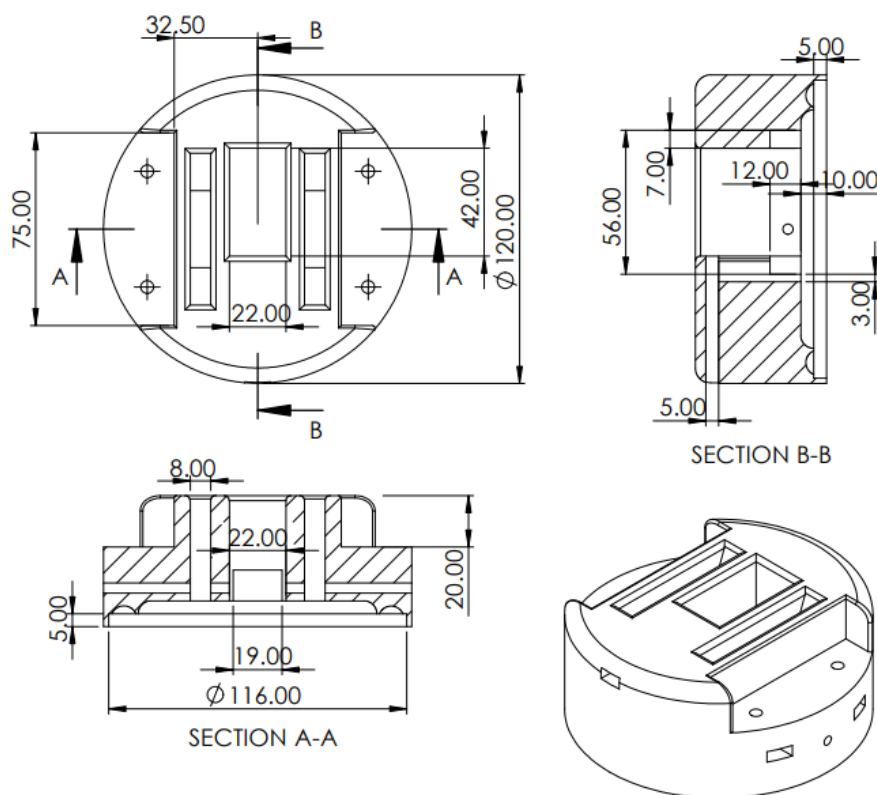
Chi tiết 1 Gá khớp xoay-I



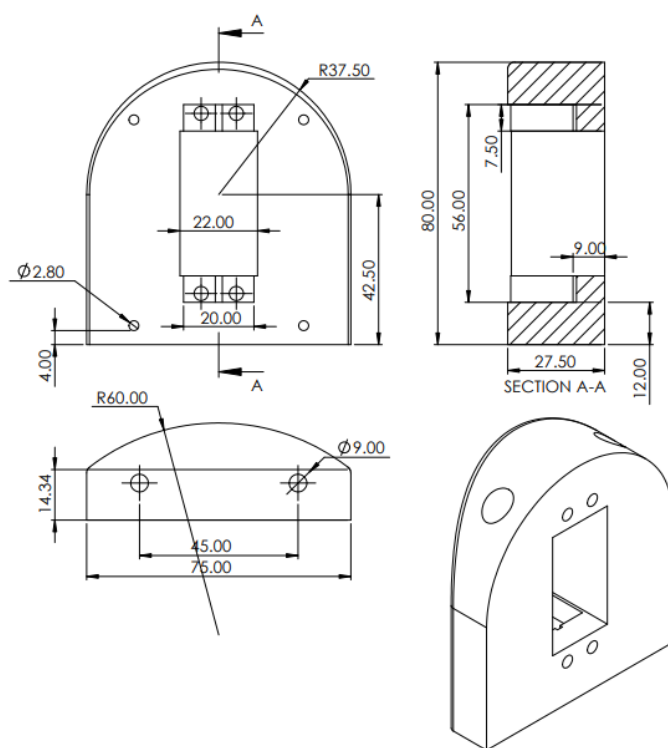
Chi tiết 2 Đế cố định



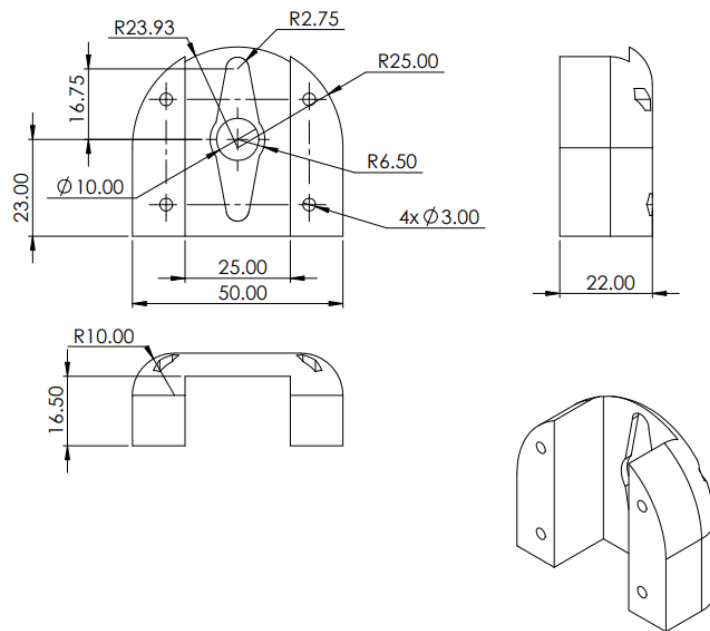
Chi tiết 3 Vòng bi



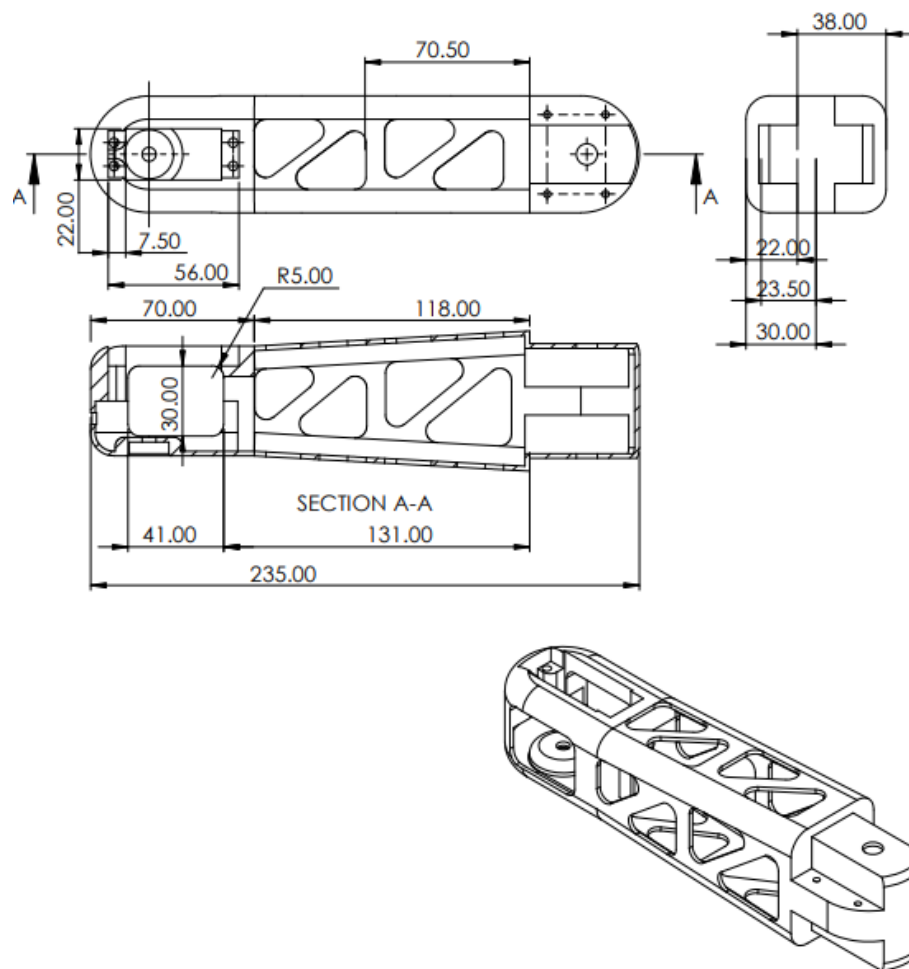
Chi tiết 4 Đế xoay



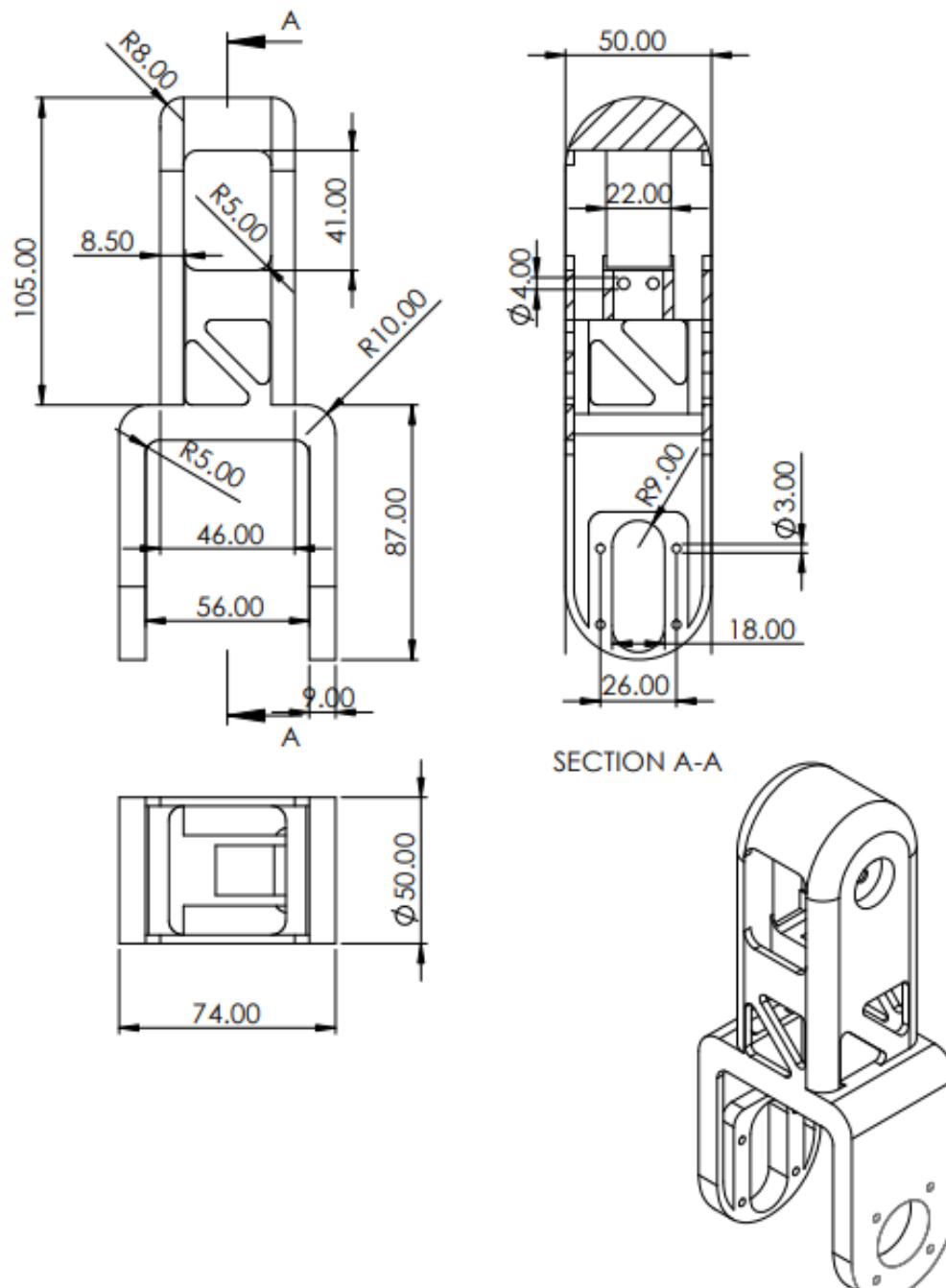
Chi tiết 5 Cụm đế khớp vai



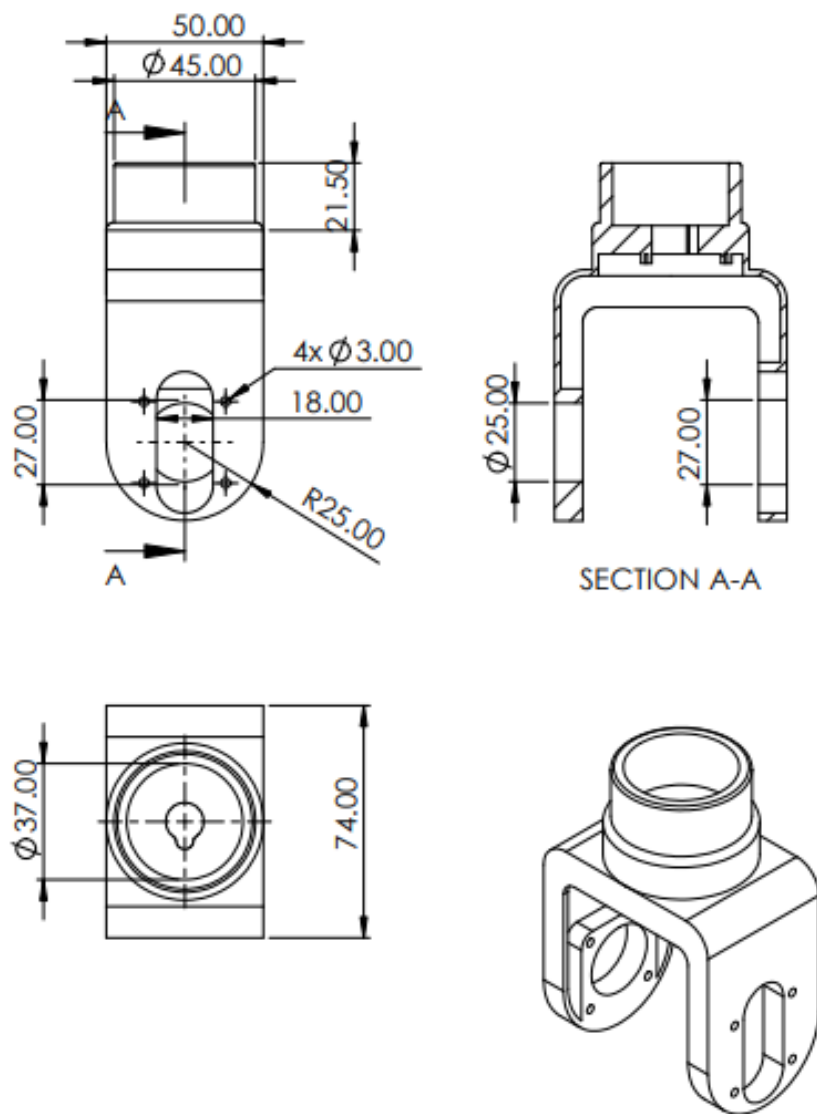
Chi tiết 6 Cụm gá khớp vai



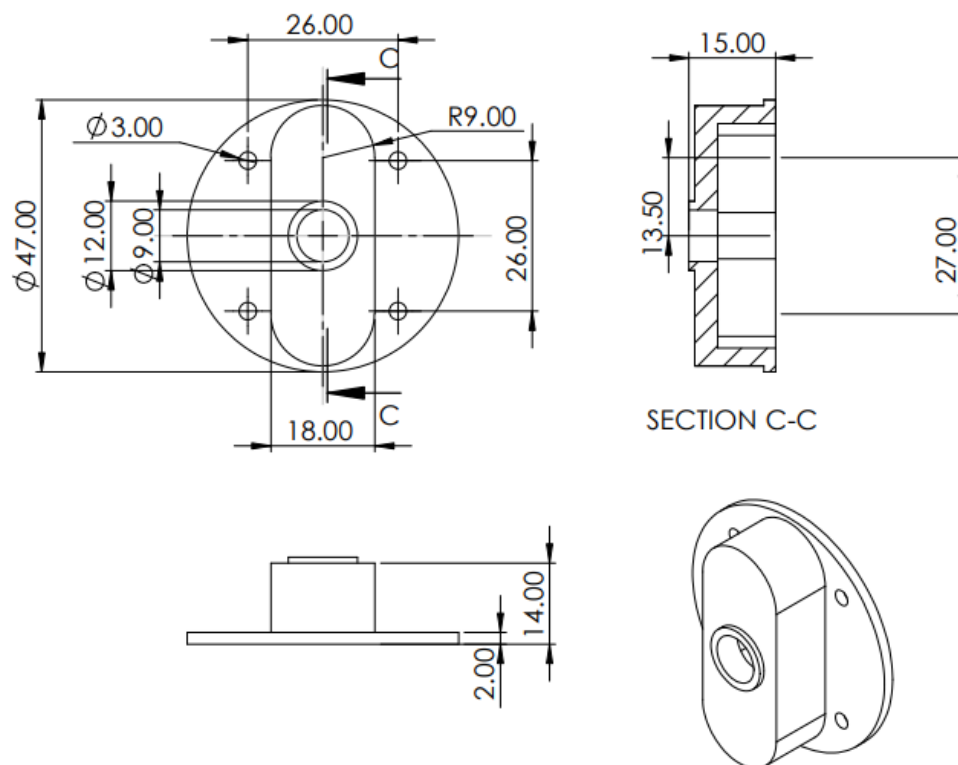
Chi tiết 7 Khâu bắt tay



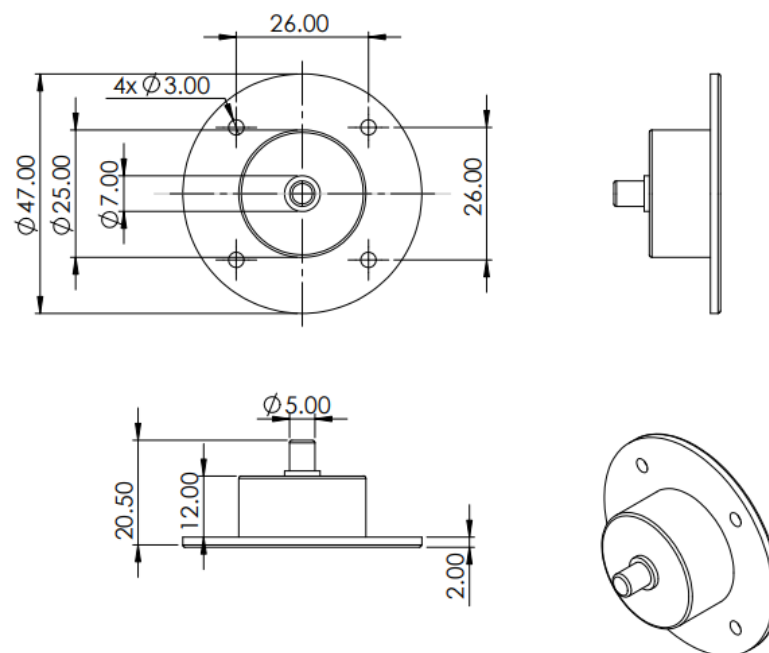
Chi tiết 8 Khâu khuỷu tay- I



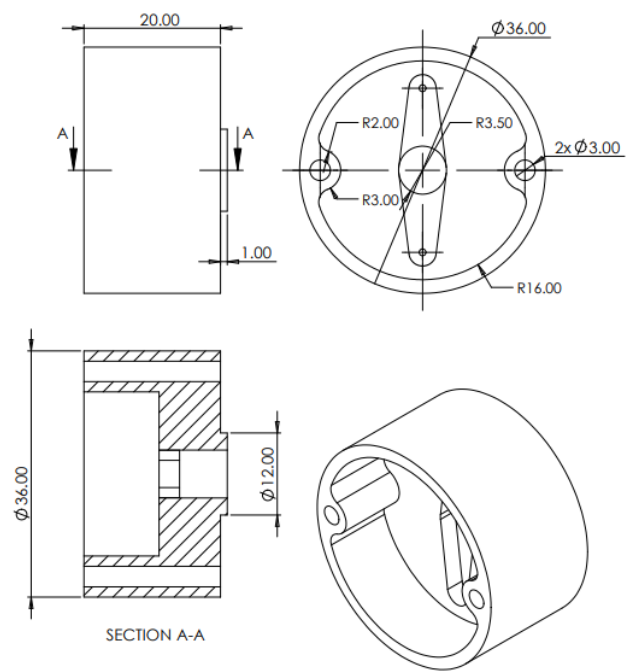
Chi tiết 9 Khâu khuỷu tay-II



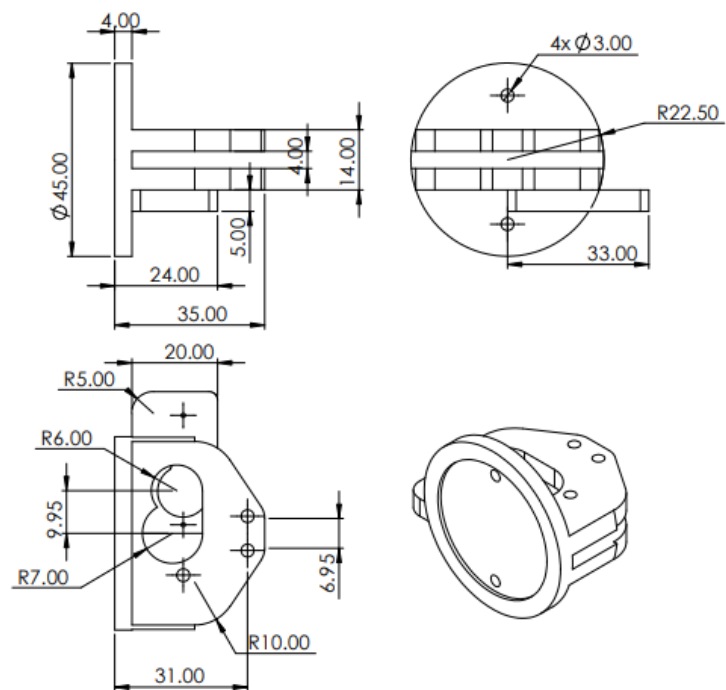
Chi tiết 10 Gá khớp xoay-II



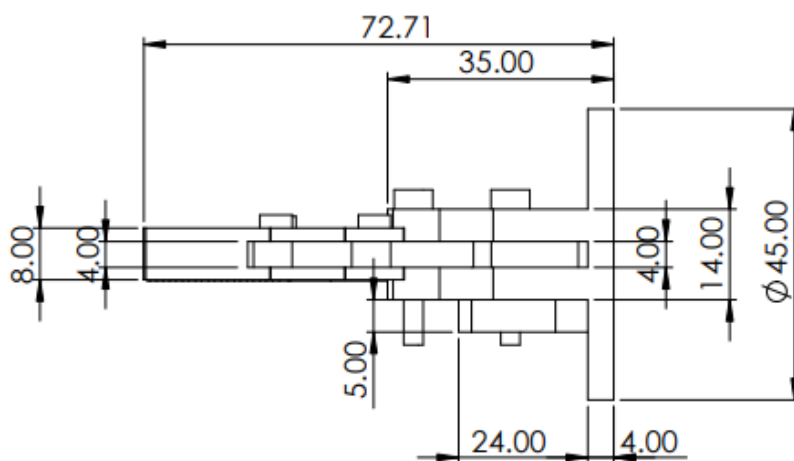
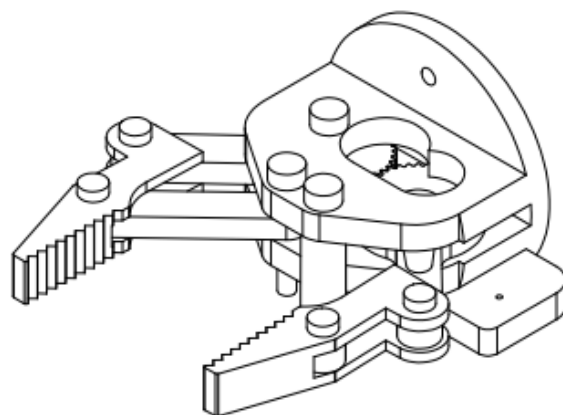
Chi tiết 11 Trục đỡ khớp xoay



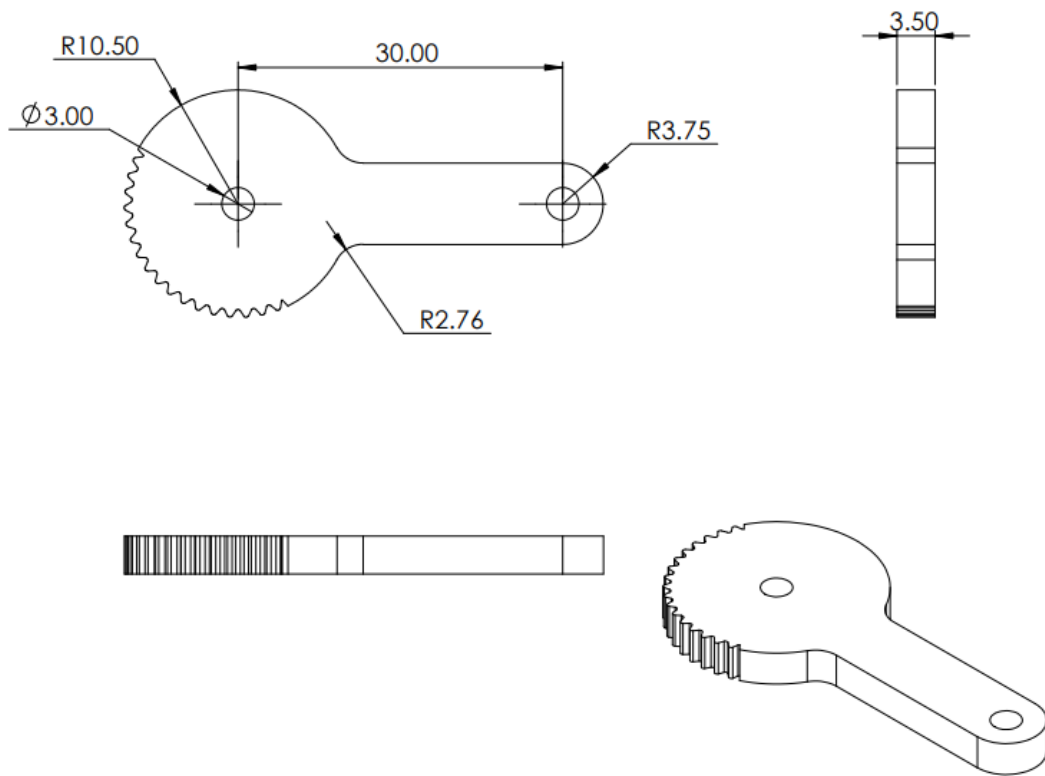
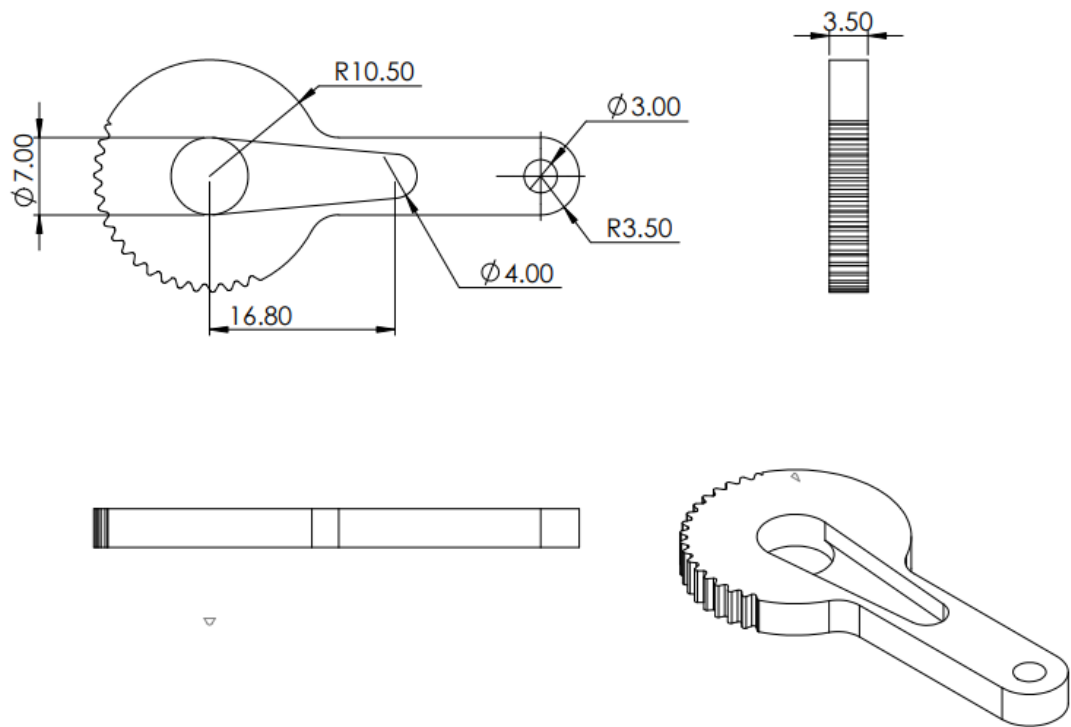
Chi tiết 13 Gá khớp xoay- III

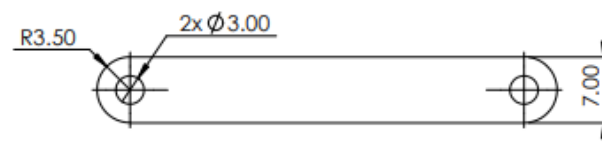
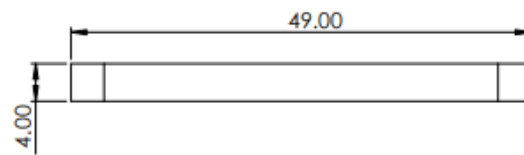
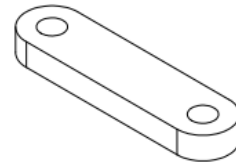
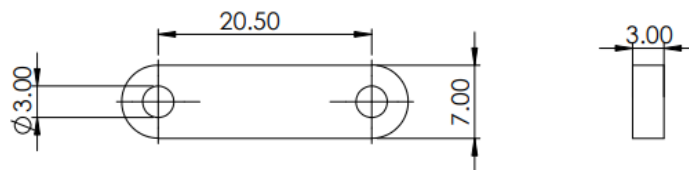


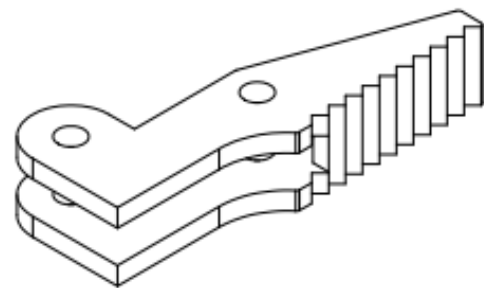
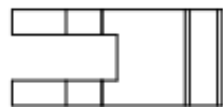
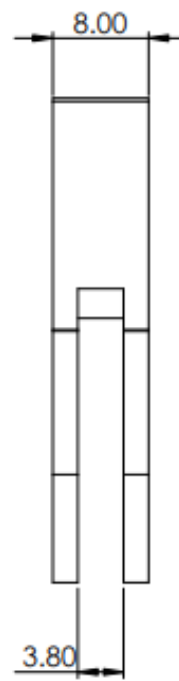
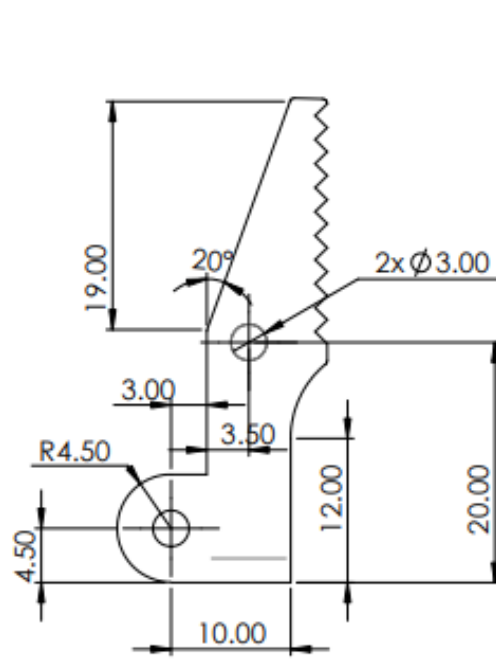
Chi tiết 14 Cụm lắp tay kẹp



Cụm tay kẹp







Đường dẫn chi tiết (cho file Word)

- [1] [IFR presents World Robotics Report 2020 - International Federation of Robotics](#)
- [2] Robots and Jobs: Evidence from US Labor Markets
[Journal of Political Economy | Vol 128, No 6 \(uchicago.edu\)](#)
- [13] https://www.yaskawa.eu.com/products/robots/collaborative/productdetail/product/hc10dt_681
- [3] <https://new.abb.com/news/detail/76652/prsrl-abbs-new-gofa-cobot-wins-prestigious-best-of-the-best-red-dot-award>
- [4] [GoFa CRB15000-datasheet digital 20210408 ms.pdf \(ctfassets.net\)](#)
- [5] [UR3e Robot \(knottscs.com\)](#)

- [7] [MG996R Tower-Pro \(components101.com\)](#)
- [8] [MG90S Micro Servo Motor Datasheet, Wiring Diagram & Features \(components101.com\)](#)
- [9] [16x2 LCD Module: Pinout, Diagrams, Description & Datasheet \(components101.com\)](#)
- [10] [Tổng quan LCD 16x2 và giao tiếp I2C LCD sử dụng Arduino - ARDUINO KIT arduinomega2560datasheet.pdf \(robotshop.com\)](#)
- [12] [Arduino IDE - Wikipedia](#)
<https://new.abb.com/news/detail/76652/prsrl-abbs-new-gofa-cobot-wins-prestigious-best-of-the-best-red-dot-award>
- [11] <https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf>

Chương trình điều khiển (source code)

```
/*  
  
serial config:  
  
Board: Arduiono MEGA  
  
Port: tty.usbseriala400eMNr  
  
Programmer: USBtinyISP  
  
*/  
  
// khai báo các thư viện hỗ trợ  
  
#include <Servo.h>  
  
#include <Wire.h>  
  
#include <LiquidCrystal_I2C.h>  
  
// cú pháp liên kết với màn hình sử dụng kết nối I2C  
  
LiquidCrystal_I2C lcd(0x27,16,2);  
  
// Khai tên các động cơ servo và biến trở điều khiển.  
  
Servo servo_0;  
  
Servo servo_1;  
  
Servo servo_2;  
  
Servo servo_3;  
  
Servo servo_4;  
  
Servo servo_5;  
  
Servo servo_6;  
  
int sensorPin0 = A4;
```

```

int sensorPin1 = A3;

int sensorPin2 = A2;

int sensorPin3 = A1;

int sensorPin4 = A0;

int sensorPin5 = A5;

int count0, arrayStep, arrayMax,arrayreset, countverz, Taster,Taster1, stepsMax, steps,
time = 1000, del = 1000, temp;

// arraystep = biến nhớ vị trí ( số thứ tự ghi thao tác) trong bộ nhớ

// arrayMax = các bước tối đa đã ghi được.

// Taster = trạng thái của nút A ( Taster1 trạng thái nút B)

// stepsMax = hành trình dài nhất mà servo đi được trong một chuyển động(di chuyển của
tay máy giữa hai vị trí lưu)

// steps = bước di chuyển giữa các vị trí đã ghi lại

// các biến định thời dùng trong chương trình

unsigned int verz = 0;

long previousMillis5 =0;

long previousMillis1 = 0;

long previousMillis2 = 0;

long previousMillis3 = 0;

long previousMillis4 = 0;

long previousMicros = 0;

long previousDelay=0;

unsigned long currentMillis=millis() ;

```

```

unsigned long currentMicros=micros() ;

// Khai báo mảng

int Delay[7] = {0,0,1,3,15,60,300}; //mảng gồm các biến với đơn vị giây dùng để gán cho
chiết áp

int SensVal[6]; // biến mảng dùng cho chiết áp

float dif[7],ist[7], sol[7], dir[7];

int joint0[180];// array for servo(s)

int joint1[180];// các mảng dùng để lưu trữ tín hiệu điều khiển của các khớp tay máy

int joint2[180];

int joint3[180];

int joint4[180];

int joint5[180];

int joint6[180];

int top = 179; // Giới hạn các biến tín hiệu lưu trong các mảng

// các biến lưu trạng thái

// Status

boolean playmode = false, Step = false,reset=false,Rstep;

// khởi tạo chương trình

void setup()

{

    lcd.begin();

    lcd.backlight();

    lcd.setCursor(3,0);

```

```

lcd.print("STARTING"); // hiển thị trạng thái đang khởi động lên màn LCD

pinMode(10, INPUT); // khởi tạo ba cổng 10, 11, 12 nhận tín hiệu từ các nút

pinMode(11, INPUT);

pinMode(12, INPUT);

pinMode(13, OUTPUT); // sets the digital pin 13 as output

digitalWrite(13, HIGH); // sets the LED on

servo_0.attach(3); // Kết nối các tên servo đã khai báo vào các chân điều khiển của arduino

servo_1.attach(4);

servo_2.attach(5);

servo_3.attach(6);

servo_4.attach(7);

servo_5.attach(8);

servo_6.attach(9);

Serial.begin(115200); // Tốc độ truyền tín hiệu qua cổng serial port

Serial.println("mini robot ready...");

//delay(1000);

lcd.setCursor(0,1);

for(int i=1;i<=16;i++)

{

    lcd.print(".");    //Hiệu ứng loading trên LCD

    delay(100);

}

}

```

```

void loop() // Vòng lặp chính của chương trình.

{

    currentMillis = millis(); // Định thời gian trong vòng lặp hiện tại với độ chính xác mili
giây

    currentMicros = micros(); // micro giây


    if( currentMillis<=3020)

    {

        lcd.clear();

        lcd.print(" READY TO GO !!!");

    }

    if ( currentMillis-previousMillis4 >10000 && playmode==false && reset==false )

    {

        lcd.clear();

        lcd.print(" READY TO GO !!!");

        previousMillis4=currentMillis;    // Hiển thị trạng thái của chương trình

        if((currentMillis/1000)%20==0)

        {

            lcd.clear();

            lcd.print(" READY TO GO !!!");

        }

        else

        checkdata();

```

```

}

//Đọc trạng thái nút nhấn

Button();

BUTTONR(); // chi tiết xem phần chương trình con

if (reset)

{

    if(Rstep)

    {

        Read();// đọc vị trí hiện tại của chiết áp

        calculate(); // tính xung điều khiển cho mỗi bước di chuyển servo

        Rstep=0;           //đặt lại vị trí tất cả các khớp theo cánh tay mẫu

    }

    else

        if(currentMicros-previousMicros>time)

        {

            previousMicros=currentMicros;    // thời gian chờ micro giây giữa các bước cấp xung

            điều khiển servo

            play_servo(); // ct con cấp xung điều khiển đã tính cho servo

        }

    }

    if(!playmode) // chế độ điều khiển

    {

        if(currentMillis - previousMillis1 > 25) // 25miliseconds until next manual mode update

```

```

{
  if (arrayStep < top)
  {
    previousMillis1 = currentMillis; //đặt lại biến thời gian cho vòng lặp sau
    readPot(); // get the value from potentiometers
    mapping(); // map to milliseconds for servos
    move_servo(); // setz newservo position
  } // Số lượng vị trí lưu < top
} // kiểm tra số lượng vị trí lưu trong ctr

else if(playmode && !reset ) // chạy tay máy tự động tuần tự theo các vị trí đã lưu
{
  if(arrayMax==0 )
  {
    Serial.println("khong co du lieu"); // Kiểm tra nếu trong ctr ko có vị trí đã lưu hoặc đã
    lưu ít hơn hai vị trí
    playmode=false;
  }
  if(arrayMax==1)
  { lcd.clear();
    lcd.print("Steps >= 2");
    delay(300);
    playmode=false;
  }
}

```

```

}

else if(arrayMax>=2)

{

if (Step) // Kiểm tra biến Step để chuyển đổi sang bước di chuyển tiếp theo

{

if (arrayStep < arrayMax) // we not reach the end from stored data

{

arrayStep += 1; // chuyển sang vị trí tay máy tiếp theo

Read(); // đọc dữ liệu từ các mảng

calculate(); // tính toán xung điều khiển cho các khớp trong chuyển động hiện tại

Step = 0; // reset biến nhớ Step cho lần lặp tới

}

else // Hoàn thành chu trình chuyển động

{

arrayStep = 0; // đặt arrayStep cho lần lặp sau

calc_pause(); // delay between moves read from potentiometer

countverz = 0; // used for the delay           Điều kiện lặp lại chu trình

while(countverz < verz) // verz = time getting from calc_pause();

{ // here we do loop and wait until next start over

countverz += 1;

calc_pause();

lcd.clear();

lcd.print("WAIT FOR "); lcd.print(verz-countverz); lcd.print(" secs");

```



```

    }

    lcd.clear();

    lcd.print("PLAYMODE.....");

}

//Serial.println(arrayStep);

}

else //Vòng lặp chạy một chu trình

{

    if (currentMicros - previousMicros > time) // here we do a single micro step

    { //

        previousMicros = currentMicros;

        play_servo();

    }

}

}

} // ende playmode

// -----Nút gạt vật lý ngừng tay
máy

// Ngừng tay máy và báo trạng thái hiện tại sau khi hoạt động lại

if(digitalRead(22)==true)

{

    lcd.clear();

    lcd.print("FREEZING...");

```

```

}

while (digitalRead(22) == true)

{
    previousMillis2=millis();

    Serial.println("aa");

    Serial.println(currentMillis);

    Serial.print(millis());

}

if( digitalRead(22) == false && millis()-previousMillis2<100)

{

    if(playmode==true && reset==0)

    {

        lcd.clear();

        lcd.print("PLAYMODE...");

    }

    else if (reset ==1 && playmode==false)

    {

        lcd.clear();

        lcd.print("RETURNING...");

    }

    if (reset==0 && playmode ==false)

    {

        lcd.clear();

```

```

        lcd.print("READY TO GO!!!");

    }

}

}

// ----- chương trình
con

void calc_pause() // read pot and map to usable delay time after a complete move is done
{
    // đọc biến trở và gán cho thời gian delay sau khi hoàn thành một chu trình
    readPot();

    temp = SensVal[4];

    if (temp < 0) temp = 0;

    temp = map(temp, 0, 1023, 0, 6);

    verz = Delay[temp]; // verz = delay in second

    Serial.print(temp);

    Serial.print(" ");

    Serial.print(verz);

    Serial.print(" ");

    Serial.println(countverz);
}

void readPot() // read analog inputs and add some offsets (mechanical corrections)
{
    // đọc tín hiệu analog và tính chỉnh vị trí của các khớp tay máy

    SensVal[0] = analogRead(sensorPin0); SensVal[0] += 0; // rotate

    SensVal[1] = analogRead(sensorPin1); SensVal[1] += -40; // Shoulder

```

```

    SensVal[2] = analogRead(sensorPin2); SensVal[2] += -120; // hand
    SensVal[3] = analogRead(sensorPin3); SensVal[3] += 0;
    SensVal[4] = analogRead(sensorPin4); SensVal[4] += 0;
    SensVal[5] = analogRead(sensorPin5); SensVal[5] += 0;
    Serial.print(SensVal[3]); Serial.print(" "); // CHECK
}

void mapping() // Chuyển đổi giá trị biên trở thành xung điều khiển servo
{
    ist[0] = map(SensVal[0], 153, 800, 2400, 600);
    ist[1] = map(SensVal[1], 170, 910, 2350, 600);
    ist[2] = map(SensVal[1], 170, 910, 600, 2350);
    ist[3] = map(SensVal[2], 90, 840, 550, 2400);
    ist[4] = map(SensVal[3], 520, 1000, 600, 2350);
    ist[5] = map(SensVal[4], 120, 840, 2350, 600);
    ist[6] = map(SensVal[5], 590, 1023, 600, 2350);
    //Serial.println(ist[2]); // CHECK
}

void record1() // Chương trình lưu xung điều khiển vào các mảng
{
    if(reset == 0)
    {
        joint0[arrayStep] = ist[0]; // write positions in servo array
        joint1[arrayStep] = ist[1];
    }
}

```

```

    joint2[arrayStep] = ist[2];

    joint3[arrayStep] = ist[3]; // ghi vị trí vào mảng của các khớp theo thứ tự arrayStep

    joint4[arrayStep] = ist[4];

    joint5[arrayStep] = ist[5];

    joint6[arrayStep] = ist[6];

}

else // gán giá trị biến trở trong trường hợp reset từ thể tay máy

{

    readPot();

    joint0[arrayreset] = map(SensVal[0], 0, 1023, 2400, 600);

    joint1[arrayreset] = map(SensVal[1], 170, 910, 2350, 600);

    joint2[arrayreset] = map(SensVal[1], 170, 910, 600, 2350);

    joint3[arrayreset] = map(SensVal[2], 90, 840, 550, 2400);

    joint4[arrayreset] = map(SensVal[3], 520, 1023, 600, 2350);

    joint5[arrayreset] = map(SensVal[4], 120, 840, 2350, 600);

    joint6[arrayreset] = map(SensVal[5], 590, 1023, 600, 2350);

}

}

void Read() // đọc giá trị điều khiển trong hai trường hợp

{

    if(reset==0)

    {

        sol[0] = joint0[arrayStep]; // read from the array

```

```

    sol[1] = joint1[arrayStep];
    sol[2] = joint2[arrayStep];
    sol[3] = joint3[arrayStep];
    sol[4] = joint4[arrayStep];
    sol[5] = joint5[arrayStep];
    sol[6] = joint6[arrayStep];
}
else
{
    sol[0] = joint0[arrayreset]; // read from the array
    sol[1] = joint1[arrayreset];
    sol[2] = joint2[arrayreset];
    sol[3] = joint3[arrayreset];
    sol[4] = joint4[arrayreset];
    sol[5] = joint5[arrayreset];
    sol[6] = joint6[arrayreset];

}
}

void move_servo()// cấp xung điều khiển cho các servo
{
    servo_0.writeMicroseconds(ist[0]);
    servo_1.writeMicroseconds(ist[1]);

```

```

servo_2.writeMicroseconds(ist[2]);

servo_3.writeMicroseconds(ist[3]);

servo_4.writeMicroseconds(ist[4]);

servo_5.writeMicroseconds(ist[5]);

servo_6.writeMicroseconds(ist[6]);

}

// bấm xung điều khiển cho từng chu trình chuyển động của tay máy

void calculate()
{
    // travel distance for each servo- khoảng cách di chuyển của mỗi động cơ
    dif[0] = abs(ist[0]-sol[0]);
    dif[1] = abs(ist[1]-sol[1]);
    dif[2] = abs(ist[2]-sol[2]);
    dif[3] = abs(ist[3]-sol[3]);
    dif[4] = abs(ist[4]-sol[4]);
    dif[5] = abs(ist[5]-sol[5]);
    dif[6] = abs(ist[6]-sol[6]);

    // biggest travel way from all servos - khoảng cách lớn nhất trong các khớp di chuyển
    stepsMax = max(dif[0],dif[1]);
    stepsMax = max(stepsMax,dif[2]);
    stepsMax = max(stepsMax,dif[3]);
    stepsMax = max(stepsMax,dif[4]);
    stepsMax = max(stepsMax,dif[5]);

```

```

stepsMax = max(stepsMax,dif[6]);

// stepsMax is the biggest distance a servo have to do between momentary position and
new pos read from the array

// Serial.println("StepsMax");

//Serial.println(stepsMax);

if (stepsMax < 500) // del(ay) between a single step is bigger is move is smaler. just
looks cool

    del = 2400;

else

    del = 1000;

// calculating single (micro) step for each servo - bãm chu trình thành các bước nhỏ
(micro steps) cho các servo

// need that to do move all servos in a loop (stepsMax times done) with different values.

// This makes all servos have done the traveling distance at same time- Làm cho các
khớp chuyển động đồng nhất.

if (sol[0] < ist[0]) dir[0] = (0-dif[0]/stepsMax); else dir[0] = dif[0]/stepsMax;

if (sol[1] < ist[1]) dir[1] = (0-dif[1]/stepsMax); else dir[1] = dif[1]/stepsMax;

if (sol[2] < ist[2]) dir[2] = (0-dif[2]/stepsMax); else dir[2] = dif[2]/stepsMax;

if (sol[3] < ist[3]) dir[3] = (0-dif[3]/stepsMax); else dir[3] = dif[3]/stepsMax;

if (sol[4] < ist[4]) dir[4] = (0-dif[4]/stepsMax); else dir[4] = dif[4]/stepsMax;

if (sol[5] < ist[5]) dir[5] = (0-dif[5]/stepsMax); else dir[5] = dif[5]/stepsMax;

```



```

    if (sol[6] < ist[6]) dir[6] = (0-dif[6]/stepsMax); else dir[6] = dif[6]/stepsMax;

    // Serial.println(dir4);
}

void play_servo() // Chạy tay máy bằng các bước nhỏ đã bấm
{
    steps += 1;

    if (steps < stepsMax) // sure we not reach the end from a move
    {
        time = del;// anfahr rampe

        if(steps == 20) time = del*4;      // chỉnh thời gian delay (trong micro giây) dựa vào
        các bước (steps) thực hiện

        else if(steps == 40) time = del*3;  // time is the delay in microsecns we wait in the
        mainloop until

        else if(steps == 80) time = del*2;  // a micro step will be done

        else if(steps == 100)time = del-1;

        if(steps == stepsMax-200) time = del*2;    // stop ramp down (200 microsteps before
        end time will be increased

        else if(steps == stepsMax-80) time = del*3;

        else if(steps == stepsMax-40) time = del*4;

        else if(steps == stepsMax-20) time = del*5;

        ist[0] += dir[0]; // set new pos - thêm xung điều khiển đã bấm vào xung điều khiển hiện
        tại

        ist[1] += dir[1];
    }
}

```

```

    ist[2] += dir[2];

    ist[3] += dir[3];

    ist[4] += dir[4];

    ist[5] += dir[5];

    ist[6] += dir[6];

    Serial.println("current step ist ");

    Serial.print(steps);

servo_1.writeMicroseconds(ist[1]);
servo_0.writeMicroseconds(ist[0]);
servo_2.writeMicroseconds(ist[2]); // xuất xung điều khiển cho các khớp
servo_3.writeMicroseconds(ist[3]);
servo_4.writeMicroseconds(ist[4]);
servo_5.writeMicroseconds(ist[5]);
servo_6.writeMicroseconds(ist[6]);

}

else

{

    Step = 1; // next step aus array lesen

    steps = 0; // servo zwischenschritte

    if (reset==true) // trường hợp reset tay máy

    {

        playmode=false;

```

```

    reset=false;

    Rstep=0;

    }

    }

}

void checkdata()

{

    if(arrayMax==0)

    {

        lcd.clear();

        lcd.print("NO DATA TO PLAY");

    }

    else

    {

        lcd.clear();

        lcd.print(arrayMax);

        lcd.print(" STEPS TO PLAY");

        if (arrayMax ==1)

        {

            lcd.clear();

```

```

    lcd.print("NOTICE: STEPS<2");

}

}

}

void Button() // check buttons for single and doubleclick
{
    if (digitalRead(26) == false)
    {
        delay(20);

        if (digitalRead(26) == true) // taster losgelassen
        {
            Taster1=0;

            if (Taster == 0)
            {
                Taster = 1;

                previousMillis3 = currentMillis;

                //Serial.print("Status record1 "); Serial.println(Taster);
            }

            else if ((Taster == 1) && (currentMillis - previousMillis3 < 250))
            {
                Taster = 2;

```

```

    //Serial.println(Taster);

}

}

}

// -----thực hiện các giao thức sau khi xác nhận trạng thái nút nhấn.

if ((Taster == 1) && (currentMillis - previousMillis3 > 1000))
{
    if (playmode==true && reset==0)
    {
        lcd.clear();

        lcd.print("PLEASE!!!!");

        delay(2000);

        lcd.clear();

        lcd.print("RETURN FIRST"); //Trường hợp nhấn record trong lúc tay máy chạy tự
động

        delay (2000);

        lcd.clear();

        delay(300);

        lcd.print("PLAYMODE.....");

    }

    else

    { // Ghi xung điều khiển vào mảng của các biến khớp và báo trạng thái qua lcd

```

```

    lcd.clear();

    delay(300);

    lcd.print("RECORD");

    arrayStep += 1;

    arrayMax = arrayStep;

    record1();

    Taster = 0;

    playmode = false;

    Serial.print("record1 Step: "); Serial.println(arrayStep);

    lcd.setCursor(7,0);

    lcd.print("Step: "); lcd.print(arrayStep);

    delay(300);

}

}

else if (Taster == 2)

{

    if (playmode==true && reset==0)

    {

        lcd.clear();

        lcd.print("PLAYMODE.....");

    }

    else

    {

```

```

    lcd.clear();

    delay(300);

    lcd.print("PLAYMODE.....");

    arrayStep = 0;

    playmode = true;      // set up trước khi chạy tự động

    Taster = 0;

    Step = 1;

    Serial.println("playmode ");

    //data_out();

    }

}

if (currentMillis - previousMillis3 > 2000) // button Status clear

{

    Taster = 0; // reset trạng thái nút

    //Serial.println("restart ");

    }

}

void BUTTONR()

{

    if (digitalRead(24)==false)

```

```

{
    delay(20);

    if(digitalRead(24)==true)
    {
        Taster=0;

        if (Taster1 == 0)
        {
            Taster1 = 1;

            previousMillis3 = currentMillis;

            //Serial.print("Status record1 "); Serial.println(Taster);
        }

        else if ((Taster1 == 1) && (currentMillis - previousMillis3 < 250))
        {
            Taster1 = 2;

            //Serial.println(Taster);
        }
    }
}

if ((Taster1 == 1)&& (currentMillis - previousMillis3>1000))
{
    arrayreset=arrayMax+1;

    reset=1;          // Ghi biến cho chu trình reset

```



```

        arrayStep=arrayMax;
record1();
// ++ arrayStep

    Rstep=1;

    steps=0;

    Taster1=0;

    lcd.clear();

    delay(300);

    lcd.print("RETURNING...");

    //Serial.print("reset shape ");
}

else if (Taster1 ==2)
{
    if (playmode==1 && !reset)
    {
        lcd.clear();

        delay(300);

        lcd.print("CAN'T CLEAR");

        delay(2000);

        lcd.clear();

        delay(300);

        lcd.print("RETURN FIRST");

        delay(2000);
    }
}

```

```

        lcd.clear();

        lcd.print("PLAYMODE.....");

    }

    else // Xóa các mảng của các khớp đã lưu

    {

        lcd.clear();

        delay(300);

        lcd.print("CLEAR ARRAY...");

        memset(joint0,0,sizeof(joint0));

        memset(joint1,0,sizeof(joint1));

        memset(joint2,0,sizeof(joint2));

        memset(joint3,0,sizeof(joint3));

        memset(joint4,0,sizeof(joint4));

        memset(joint5,0,sizeof(joint5));

        memset(joint6,0,sizeof(joint6));

        // Serial.print("clear ");

        Taster1=0;

        arrayStep=0;

        arrayMax=0;

        delay(2000);

        lcd.clear();

        delay(300);

        lcd.print("CLEAR DONE");

```

```
        delay (2000);

    }

}

if (currentMillis - previousMillis3 > 2000) // button Status clear

{

    Taster1=0;

    //Serial.println("restart ");

}

}
```