

BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH

KHOA CÔNG NGHỆ CƠ KHÍ



BÁO CÁO:

**THỰC HÀNH TỰ ĐỘNG HÓA VÀ ROBOT
NGÀNH: CÔNG NGHỆ KỸ THUẬT CƠ ĐIỆN
TỬ**

GV: LƯƠNG QUỐC VIỆT

NHÓM 4

HỌ TÊN SV:

2025220351- PHAN CHÍ BẢO

2025223256- NGUYỄN VĂN NHÂN

2025223633- TRẦN NGUYỄN TIẾN PHÁT

2025230075- TRẦN QUÂN HUY

THÀNH PHỐ HỒ CHÍ MINH, THÁNG 3 NĂM 2025

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH
KHOA CÔNG NGHỆ CƠ KHÍ



BÁO CÁO:
THỰC HÀNH TỰ ĐỘNG HÓA VÀ ROBOT
NGÀNH: CÔNG NGHỆ KỸ THUẬT CƠ ĐIỆN TỬ
GV: LƯƠNG QUỐC VIỆT
NHÓM 4

HỌ TÊN SV:

2025220351- PHAN CHÍ BẢO

2025223256- NGUYỄN VĂN NHÂN

2025223633- TRẦN NGUYỄN TIẾN PHÁT

2025230075- TRẦN QUÂN HUY

THÀNH PHỐ HỒ CHÍ MINH, THÁNG 3 NĂM 2025

PHẦN NHẬN XÉT TÓM TẮT CỦA CÁN BỘ HƯỚNG DẪN

1. Tinh thần thái độ của sinh viên trong quá trình làm đồ án.

.....

.....

.....

.....

2. Đánh giá chất lượng.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời tri ân sâu sắc đến giáo viên hướng dẫn là thầy **Lương Quốc Việt**. Trong quá trình tìm hiểu và thực hiện bộ môn thực hành tự động hóa và robot, em đã nhận được sự giảng dạy và hướng dẫn rất tận tình, tâm huyết trong suốt quá trình học tập. Nhờ vậy mà em đã hoàn thành được bộ môn này. Và quan trọng hơn, sau đó em đã học được thêm nhiều kiến thức mới, những kinh nghiệm thực tế trong quá trình học tập thực hành lập trình và làm việc với những thiết bị chuyên môn. Em xin chân thành cảm ơn thầy đã luôn kiên nhẫn giải đáp những thắc mắc của em, luôn động viên và khuyến khích em vượt qua những khó khăn, thử thách trong quá trình nghiên cứu và học tập. Sự nhiệt tình và sự cống hiến của thầy không chỉ giúp em hoàn thành tốt bài học, mà còn truyền cảm hứng cho em trong việc theo đuổi đam mê của mình trong lĩnh vực công nghệ.

Bên cạnh đó, em cũng muốn bày tỏ lòng biết ơn sâu sắc đối với thầy vì đã tạo điều kiện cho em tham gia vào các dự án thực tế, giúp em có cơ hội áp dụng những kiến thức đã học vào thực tiễn. Điều này không chỉ củng cố thêm kiến thức chuyên môn mà còn giúp em phát triển kỹ năng làm việc nhóm và khả năng giải quyết vấn đề. Kính chúc thầy dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp cao quý của mình. Mong rằng thầy sẽ tiếp tục dìu dắt nhiều thế hệ học trò, truyền đạt những tri thức quý báu và đam mê nhiệt huyết trong giảng dạy như thầy đã từng làm với chúng em.

Tp.HCM, ngày ... tháng ... Năm 2025

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

NHẬN XÉT GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP.HCM, Ngày....Tháng....Năm 2025

Giáo viên hướng dẫn

(Ký và ghi rõ họ tên)

MỤC LỤC

LỜI NÓI ĐẦU.....	5
TÓM TẮT MÔN HỌC.....	7
CHƯƠNG 1: GIỚI THIỆU CHUNG	8
1.1. Lý do chọn đề tài:.....	8
1.2. Mục tiêu của đề tài:	8
1.3. Giới thiệu các công cụ và ngôn ngữ:.....	8
Visual Studio:	8
Python:	9
Qt Designer:	9
CHƯƠNG 2: TỔNG QUAN VỀ CÁNHN TAY ROBOT 6 BẬC TỰ DO	9
2.1. Khái niệm bậc tự do (DOF):	9
2.2. Nguyên lý hoạt động:	18
2.3. Động học của robot:	18
2.3.1. Động học thuận (Forward Kinematics):	18
2.3.2. Động học ngược (Inverse Kinematics):	19
CHƯƠNG 3: PHẦN CỨNG VÀ PHẦN MỀM HỆ THỐNG	19
3.1. Phần cứng:.....	19
3.2. Phần mềm:.....	20
CHƯƠNG 4: QUY TRÌNH LẬP TRÌNH VÀ ĐIỀU KHIỂN	21
4.1. Sơ đồ khối hệ thống:	21
[Người dùng] → [Giao diện Qt] → [Python Logic] → [Serial COM] → [Vi điều khiển] → [Động cơ Robot].....	21
4.2. Thiết kế giao diện Qt:.....	21
<i>Hình 2: giao diện Qt điều khiển 4 bậc tự do</i>	<i>21</i>
4.3. Code điều khiển cánh tay robot:	22
4.3.1. Khởi tạo cổng Serial:	73
Code Python khởi động Serial	74
4.3.3. Liên kết GUI với logic:.....	76

cách hoạt động	78
4.4. Tích hợp Visual Studio:	78
CHƯƠNG 5: KẾT QUẢ, ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN.....	78
5.1. Kết quả đạt được:	78
5.2. Đánh giá:	78
5.3. Hướng phát triển:	79
KẾT LUẬN	79
TÀI LIỆU THAM KHẢO.....	79

LỜI NÓI ĐẦU

Môn học Thực hành và Tự động hóa Robot đóng vai trò quan trọng trong việc giúp sinh viên tiếp cận thực tiễn điều khiển và lập trình robot công nghiệp, từ đó nâng cao kỹ năng tư duy logic, khả năng ứng dụng công nghệ vào sản xuất hiện đại. Trong quá trình thực hiện các bài học và bài thực hành, sinh viên không chỉ được tiếp cận các hệ thống robot thực tế mà còn rèn luyện kỹ năng phân tích, xử lý tín hiệu và xây dựng giải pháp tự động hóa.

Báo cáo này tổng hợp những nội dung đã thực hiện, kết quả đạt được và những bài học rút ra trong suốt quá trình học tập. Mục tiêu nhằm củng cố kiến thức chuyên môn, đồng thời làm nền tảng cho việc nghiên cứu và phát triển trong lĩnh vực robot và tự động hóa trong tương lai.

Trong thời kỳ Cách mạng Công nghiệp 4.0, các dây chuyền sản xuất hiện đại ngày càng yêu cầu tính tự động hóa cao, giảm phụ thuộc vào nhân công thủ công, đồng thời tăng độ chính xác và ổn định của sản phẩm. Trong số các thiết bị tự động, cánh tay robot công

nghiệp đóng vai trò then chốt. Đặc biệt, cánh tay robot 6 bậc tự do (6 Degrees of Freedom – 6DOF) có khả năng mô phỏng hầu như mọi chuyển động cơ bản của tay người, từ di chuyển trong không gian 3D đến xoay, nghiêng, gấp, hàn, và lắp ráp.

Tuy nhiên, để khai thác tối đa tiềm năng của robot, cần phải xây dựng một hệ thống điều khiển thông minh, linh hoạt, trực quan và dễ vận hành. Trong báo cáo này, ta sẽ sử dụng kết hợp Visual Studio (môi trường phát triển tích hợp), Python (ngôn ngữ lập trình linh hoạt) và Qt Designer (công cụ thiết kế giao diện người dùng) để xây dựng một phần mềm điều khiển hoàn chỉnh cho cánh tay robot 6DOF.

TÓM TẮT MÔN HỌC

Môn *Thực hành Tự động hóa và Robot* trang bị cho sinh viên kiến thức và kỹ năng cơ bản trong việc lập trình, thiết kế giao diện và điều khiển hệ thống robot, đặc biệt là robot công nghiệp 6 bậc tự do. Trong khuôn khổ môn học, sinh viên sử dụng các công cụ phần mềm hiện đại như **Visual Studio**, **Git-SCM** và **Qt Designer** để xây dựng chương trình điều khiển, thiết kế giao diện người dùng (GUI), cũng như quản lý mã nguồn một cách khoa học.

Thông qua các bài thực hành, sinh viên được hướng dẫn cách kết nối phần mềm với cánh tay robot, viết mã điều khiển các chuyển động (quay, kẹp, dịch chuyển...), thiết kế giao diện trực quan để tương tác và điều khiển robot theo yêu cầu. Bên cạnh đó, việc sử dụng Git-SCM giúp sinh viên làm quen với quy trình phát triển phần mềm chuyên nghiệp, hỗ trợ làm việc nhóm và quản lý phiên bản hiệu quả.

Môn học không chỉ giúp củng cố kiến thức lý thuyết về robot và tự động hóa mà còn phát triển năng lực lập trình ứng dụng thực tế, từ đó tạo nền tảng vững chắc cho các dự án tự động hóa trong môi trường công nghiệp hiện đại.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Lý do chọn đề tài:

Việc lập trình robot công nghiệp thường yêu cầu những giải pháp phần mềm chuyên dụng, đôi khi phức tạp và đắt tiền. Với sự phát triển của **Python** cùng các công cụ mã nguồn mở, việc tạo ra một ứng dụng điều khiển robot có giao diện trực quan, chức năng mạnh mẽ và chi phí thấp trở nên khả thi. Đề tài này giúp:

- Tiết kiệm chi phí so với các phần mềm điều khiển thương mại.
- Dễ dàng mở rộng và tùy biến cho nhiều loại robot khác nhau.
- Giúp sinh viên và kỹ sư trẻ tiếp cận công nghệ robot mà không cần hệ thống phức tạp.

1.2. Mục tiêu của đề tài:

- **Xây dựng phần mềm** điều khiển cánh tay robot 6DOF bằng Python, tích hợp GUI từ Qt Designer.
- **Tích hợp Visual Studio** để lập trình, quản lý dự án, và gỡ lỗi.
- **Kết nối phần mềm** với phần cứng robot thông qua giao tiếp Serial (UART) hoặc các giao thức khác.
- Thực hiện điều khiển **theo thời gian thực**, độ trễ thấp, thao tác mượt mà.
- Có khả năng **lưu và chạy lại** các kịch bản chuyển động.
- Việc kết hợp các công cụ này giúp xây dựng hệ thống điều khiển có tính ổn định, giao diện thân thiện và dễ mở rộng.

1.3. Giới thiệu các công cụ và ngôn ngữ:

Visual Studio:

- IDE phổ biến hỗ trợ nhiều ngôn ngữ, trong đó có Python.
- Cung cấp trình gỡ lỗi (debugger), quản lý gói, quản lý file dự án.

- Hỗ trợ trực tiếp tích hợp Git, giúp lưu trữ và chia sẻ mã nguồn dễ dàng.

Python:

- Ngôn ngữ lập trình bậc cao, cú pháp rõ ràng, thư viện phong phú.
- Có nhiều thư viện hỗ trợ robot: pyserial (giao tiếp Serial), numpy (tính toán), matplotlib (vẽ đồ thị), PySide6/PyQt5 (GUI).
- Dễ dàng chạy trên nhiều hệ điều hành (Windows, Linux, macOS).

Qt Designer:

- Công cụ thiết kế GUI dạng kéo-thả.
- Cho phép thiết kế trực quan cửa sổ, nút, thanh trượt, bảng hiển thị.
- Xuất file .ui và chuyển đổi thành file Python để tích hợp logic.

CHƯƠNG 2: TỔNG QUAN VỀ CÁNH TAY ROBOT 6 BẬC TỰ DO

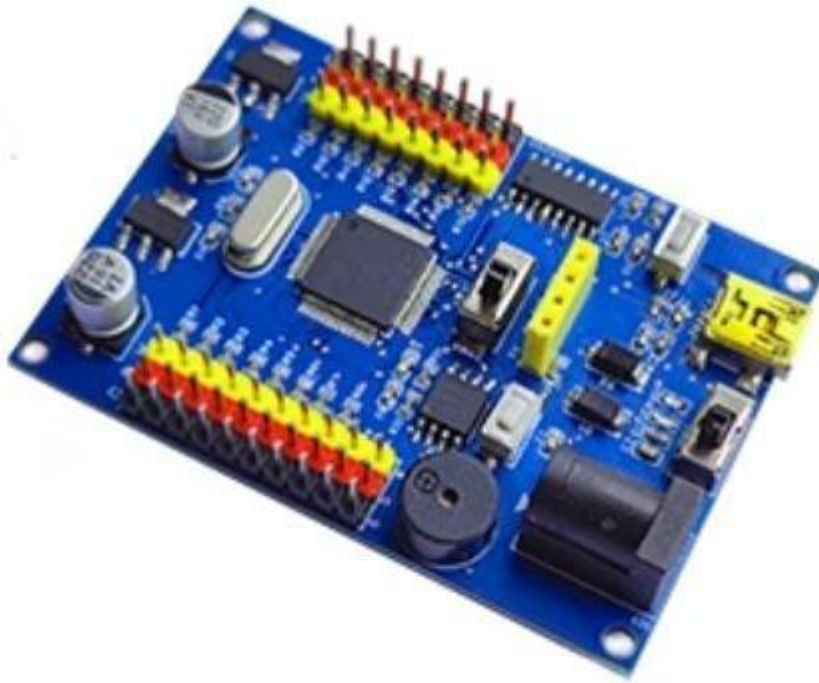
2.1. Khái niệm bậc tự do (DOF):



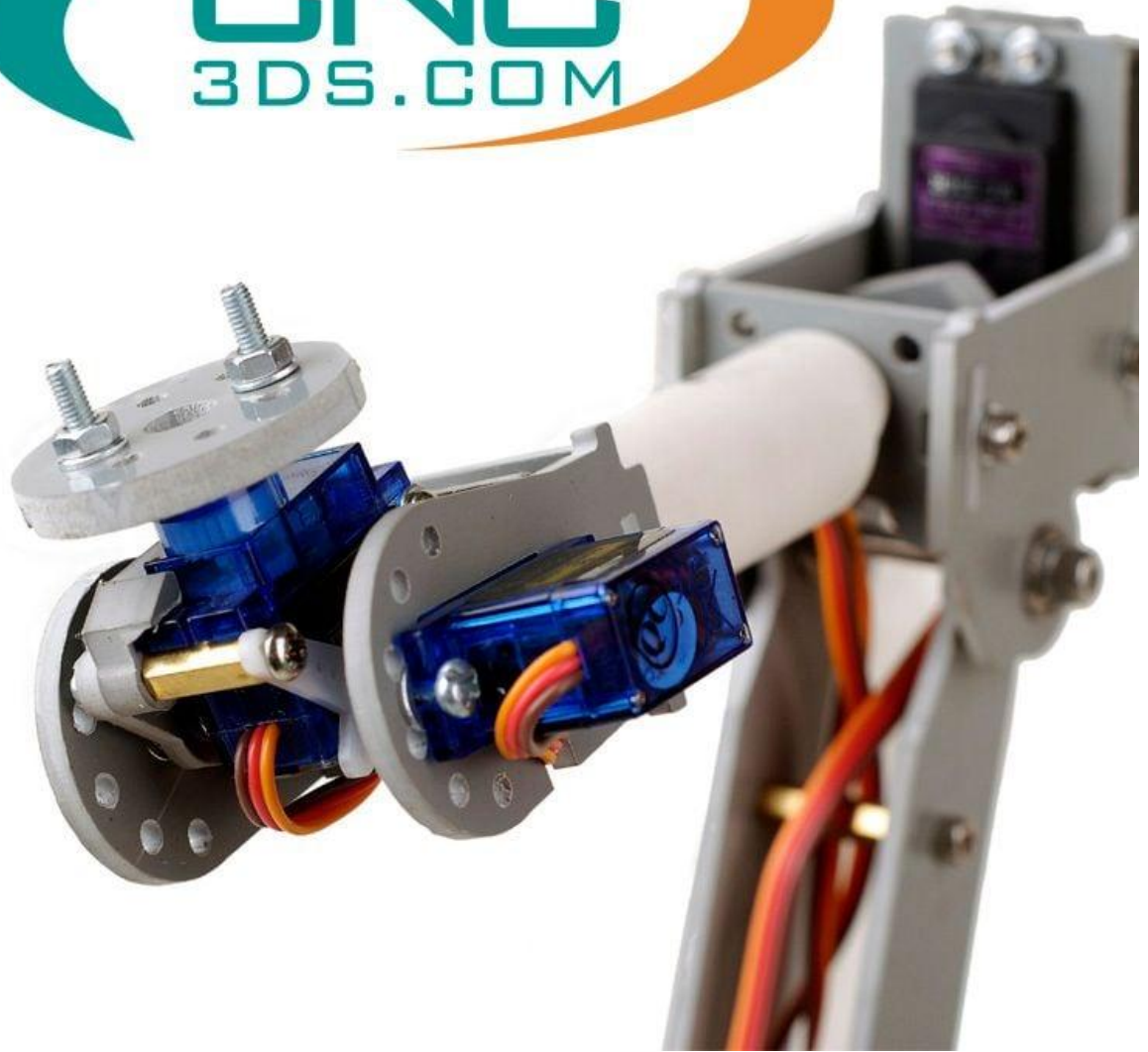
-
- Tính năng của cánh tay robot mini 6 bậc tự do AMZ-R3
- - Robot được trang bị 4 động cơ servo chống nước hoàn toàn do Đài Loan sản xuất và 2 động cơ servo nhỏ. Thân robot được làm hoàn toàn bằng thép không gỉ, xử lý bề mặt đẹp hơn, có độ cứng cao, ổn định và chắc chắn mà không bị biến dạng.

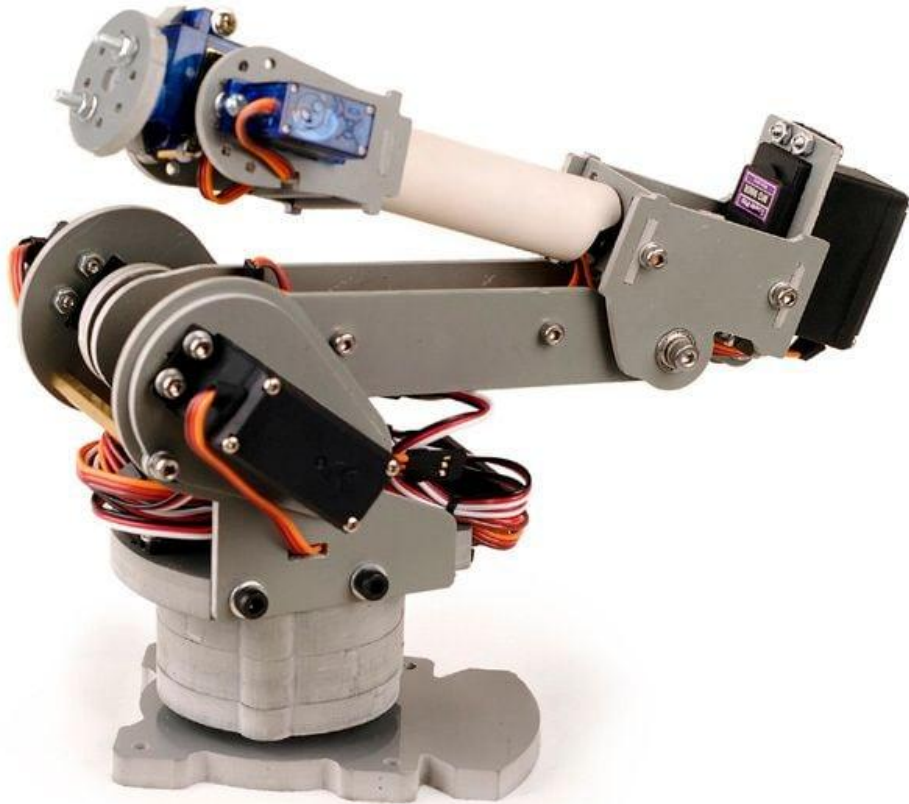


-
- - Bảng điều khiển servo 16 kênh rất phù hợp cho người mới bắt đầu và những người có kiến thức cơ bản về STM32. Nó có thể chỉnh sửa các hành động trực tiếp bằng đồ họa mà không cần viết mã. Nó hỗ trợ điều khiển từ xa không dây ps2 để điều khiển trực tiếp cánh tay robot.



-
- - Trọng lượng kẹp tối đa của cánh tay robot là 50g, phù hợp cho việc giảng dạy và kiểm soát lý thuyết xác minh, trình diễn địa điểm, DIY, v.v., và không phù hợp với dây chuyền sản xuất công nghiệp.





-
- - Cánh tay robot 6 bậc tự do thực sự: eo, căng tay, căng tay, xoay căng tay, cổ tay, xoay cổ tay, tất cả các khớp di chuyển được kết nối bằng vòng bi chất lượng cao.
- - Khung robot được gia công CNC. Các khớp chính là bánh răng kim loại kỹ thuật số MG996 và SG90.
-
- Thông số kỹ thuật cánh tay robot mini 6 bậc tự do AMZ-R3

- Các khớp chính là tay lái kỹ thuật số chống nước KS3518 và SG90. Sáu độ tham số tự do:

stt	part	Lý thuyết	Góc thực
1	Thân	0 - 180	5 - 175
2	Tay 1	Góc biến thiên	
3	Tay 2	Góc biến thiên	
4	Góc tay 2	0 - 180	5 - 175
5	Cổ tay	0 - 180	5 - 175
6	Cặp	0 - 180	5 - 175

- Ví dụ code lập trình

```
myspeed=500; ← Góc quay
for(pos = 0; pos <=myspeed; pos += 1)
{
  myservoA.write(int(map(pos,1,myspeed,66,90))); // 让A从66度旋转到90度 (可修改角度)
  myservoB.write(int(map(pos,1,myspeed,90,40))); // 让B从90度旋转到40度 (可修改角度)
  delay(1);
}
```

tên bậc góc bắt đầu Góc kết thúc

- Thông số kỹ thuật Servo MG996 55g:

- Kích thước: 40,7 * 19,7 * 42,9mm

Mô-men xoắn làm việc: 9,4-13kg / cm

Tốc độ quay phản hồi: 53-62R / M

Nhiệt độ làm việc: -30 ~ + 60 ° C

Vùng chết: 5us

Cắm: JR, Tutaba

Góc quay: 180 độ

Servo tương tự

Dòng điện làm việc: 100mA

Điện áp làm việc: 5-7.2V

Bánh răng kim loại, động cơ corless, hai ổ bi

Tốc độ hoạt động: 0,17s / 60 độ (4,8V); 0,14s / 60 độ (6.0V)

- Thông số kỹ thuật servo SG90 9G:
- Kích thước: 23 * 12,2 * 29mm

Cân nặng: 9g

Mô-men xoắn làm việc: 1.6kg / cm

Tốc độ quay phản hồi: 0,12-0,13s / 60 độ

Nhiệt độ làm việc: -30 ~ + 60 C

Vùng chết: 5ms

Cắm: JR, Tutaba

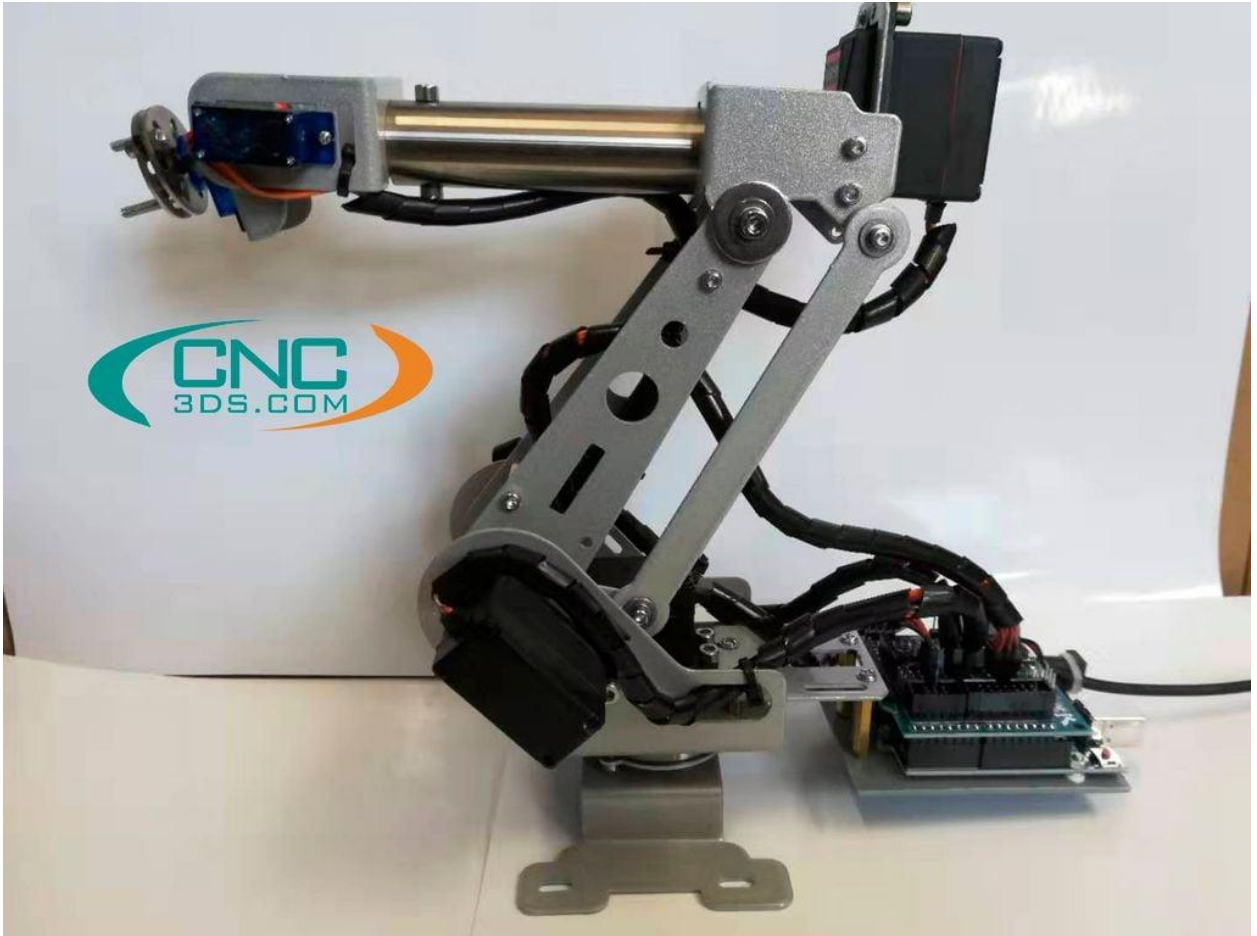
Góc quay: 180 độ

Servo tương tự

Dòng điện làm việc: 100mA

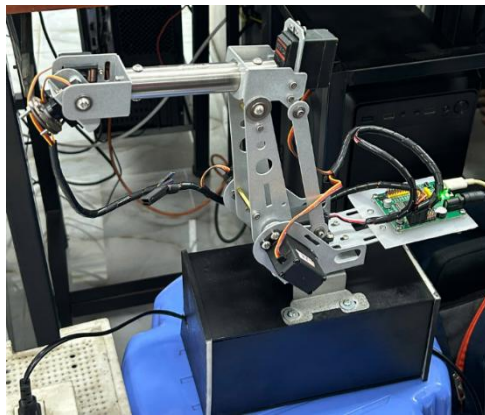
Điện áp làm việc: 3,5-6V

Bánh răng kim loại



-
-
- Ưu điểm của cánh tay robot mini 6 bậc tự do AMZ-R3
- Cánh tay robot 6 DOF là một bộ công cụ phát triển giáo dục của nhà sản xuất tích hợp việc dạy và học trong các trường cao đẳng và đại học. Nó được trang bị 6 bánh lái, tương đương với 6 khớp. Nó thực hiện xoay ba chiều để nắm các vật thể và trọng lượng nắm tối đa là 50g, nó không phù hợp với dây chuyền sản xuất công nghiệp.
- **Bậc tự do** là số lượng tham số độc lập cần thiết để xác định vị trí và hướng của một vật thể.

- Cánh tay robot 6DOF có 6 khớp quay (Revolute Joint) cho phép di chuyển tự do trong không gian 3D:
 1. Quay ngang (Base rotation)
 2. Nâng/hạ cánh tay (Shoulder pitch)
 3. Gập/duỗi khuỷu tay (Elbow pitch)
 4. Xoay cổ tay (Wrist roll)
 5. Gập cổ tay (Wrist pitch)
 6. Xoay đầu gắp (End effector rotation)



Hình 1: Cánh tay robot 6 bậc tự do

2.2. Nguyên lý hoạt động:

- Mỗi khớp được điều khiển bởi một động cơ (servo hoặc stepper).
- Vi điều khiển nhận lệnh góc quay từ phần mềm qua giao tiếp Serial.
- Động cơ quay đến vị trí mong muốn theo lệnh PWM hoặc xung điều khiển.

2.3. Động học của robot:

2.3.1. Động học thuận (Forward Kinematics):

- Mục tiêu: Tính vị trí và hướng của đầu gắp khi biết các góc khớp.
- Sử dụng tham số **Denavit–Hartenberg (DH)** để mô tả quan hệ giữa các khớp.
- Công thức tổng quát:

$$T=A1(\theta1) \cdot A2(\theta2) \cdot \dots \cdot A6(\theta6)$$

- Trong Python, có thể dùng numpy để xử lý ma trận.

2.3.2. Động học ngược (Inverse Kinematics):

- Mục tiêu: Xác định các góc khớp khi biết vị trí mong muốn của đầu gắp.
- Phức tạp hơn, có thể dùng giải pháp số (numerical) hoặc giải tích.

CHƯƠNG 3: PHẦN CỨNG VÀ PHẦN MỀM HỆ THỐNG

3.1. Phần cứng:

Trình bày giao diện điều khiển:

The interface is divided into two main sections: POSITION CONTROL and JOINT CONTROL.

POSITION CONTROL: Includes buttons for Z+, X-, 0, X+, and Z-. Below these are sliders for 'Bước duy chuyển' (set to 1) and 'Tốc độ duy chuyển' (set to 100).

JOINT CONTROL: Lists four joints (Khớp 1 to 4) with buttons for INC, DES, and a 90° target.

Data Table:

-0.00	0.00	0.00	-0.00
-0.00	0.00	1.00	-150.00
-1.00	0.00	-0.00	150.00
0.00	0.00	0.00	1.00

Note:

- Bước duy chuyển (độ)
- VD: bước duy chuyển = 1 => mỗi lần duy chuyển các khớp quay 1 độ.
- Tốc độ duy chuyển (ms)
- VD: 100 ms , mỗi bước động cơ quay trong khoảng thời gian 100ms (0,1s)

Buttons: NHẤN NHẢ, NHẤN GIỮ

Warning: Nếu chọn nhấn giữ thì sẽ không tùy chỉnh được tốc độ, bước duy chuyển.

1. Cánh tay robot 6DOF:

- Vật liệu: Nhôm hoặc nhựa.
- Động cơ: Servo MG996R hoặc động cơ bước NEMA17.

- Phạm vi góc quay: 0–180° (servo), hoặc tùy biến (stepper).

2. Bộ điều khiển:

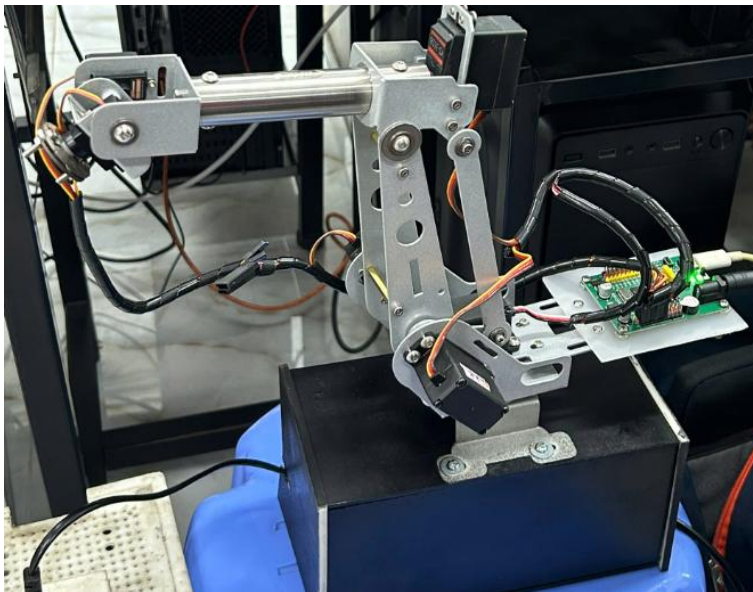
- Arduino Mega 2560: nhiều chân PWM, dễ lập trình.
- STM32: tốc độ cao, hỗ trợ nhiều giao thức.
- PLC (tùy chọn): cho môi trường công nghiệp.

3. Nguồn cấp:

- 5–6V DC cho servo.
- 12–24V DC cho stepper.

4. Máy tính:

- Windows 10/11, RAM \geq 4GB.
- Cài đặt Visual Studio, Python, Qt Designer.



Hình 1: cánh tay robot 6 bậc tự do

3.2. Phần mềm:

- **Visual Studio 2022** với Python extension.
- **Python**
- **Qt Designer** (tích hợp hoặc cài riêng).
- **Thư viện Python**

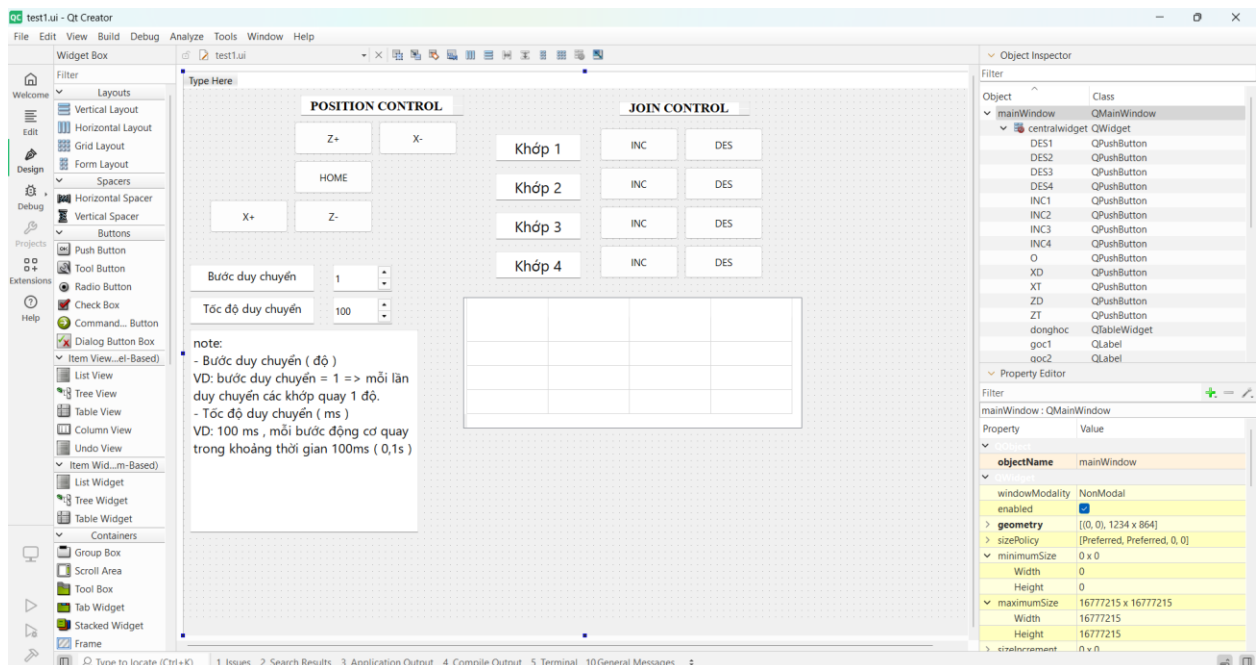
CHƯƠNG 4: QUY TRÌNH LẬP TRÌNH VÀ ĐIỀU KHIỂN

4.1. Sơ đồ khối hệ thống:

[Người dùng] → [Giao diện Qt] → [Python Logic] → [Serial COM] → [Vi điều khiển]
→ [Động cơ Robot]

4.2. Thiết kế giao diện Qt:

- **Thanh trượt (QSlider):** Điều chỉnh góc từng khớp.
- **Nút bấm (QPushButton):** Gửi lệnh di chuyển, lưu vị trí, chạy kịch bản.
- **Màn hình số (QLCDNumber):** Hiển thị góc hiện tại.
- **Nhãn (QLabel):** Thông tin trạng thái.



Hình 2: giao diện Qt điều khiển 4 bậc tự do

Lưu file .ui, chuyển sang .py:

```
pyside6-uic robot_gui.ui -o robot_gui.py
```

4.3. Code điều khiển cánh tay robot:

Code main

```
import sys, math, serial, time
```

```
from PyQt6 import QtWidgets
```

```
from test1 import Ui_mainWindow
```

```
# ===== CẤU HÌNH =====
```

```
SERIAL_PORT = 'COM9'
```

```
BAUD = 115200
```

```
# Map khớp -> kênh servo trên mạch Kejinghui (khớp1..4)
```

```
SERVO_CHANNEL = [[1], [3], [2], [4]]
```

```
# Lệch cơ khí
```

```
OFFSETS = [-4.0, 0.0, 0.0, 0.0]
```

```
# Giới hạn an toàn
```

```
MIN_ANGLE, MAX_ANGLE = 5.0, 175.0
```



```
# ===== Trục X/Z (nhấn-nhả, KHÔNG dùng spinbox) =====
```

```
AXIS_J2_STEP_DEG = 2.0 # Z+: mỗi lần click đổi khớp 2 ~2°
```

```
AXIS_J3_STEP_DEG = 2.0 # X+: mỗi lần click đổi khớp 3 ~2°
```

```
AXIS_T_MS = 350 # thời gian nội suy T cho lệnh trục
```

```
# Tham số động học (để tính r,z & điền bảng)
```

```
LINKS_MM = {"L1": 100.0, "L2": 150.0, "L3": 150.0, "L4": 100.0}
```

```
def clamp(v, lo, hi): return lo if v < lo else (hi if v > hi else v)
```

```
class MainWindow(QMainWindow):
```

```
    def __init__(self):
```

```
        super().__init__()
```

```
        self.ui = Ui_mainWindow()
```

```
        self.ui.setupUi(self)
```

```
        # Góc logic (deg): [t1, t2, t3, t4]
```

```
        self.goc = [90.0, 90.0, 90.0, 90.0]
```

```
        # Serial
```

```
        try:
```

```

        self.ser = serial.Serial(SERIAL_PORT, BAUD, timeout=0.5)

        print("Serial opened:", SERIAL_PORT)

    except Exception as e:

        self.ser = None

        print("Serial NOT opened:", e)

```

SpinBox (áp cho KHỚP, không áp cho trục X/Z)

```

self.ui.sodo.setRange(1, 10)

self.ui.tocdo.setRange(100, 2000)

```

Bảng 4x4

```

try:

    self.ui.donghoc.setRowCount(4)

    self.ui.donghoc.setColumnCount(4)

except:

    pass

```

Nút KHỚP (nhấn-nhả dùng spinbox)

```

self.ui.INC1.clicked.connect(lambda: self._joint_click(0, +1))

self.ui.DES1.clicked.connect(lambda: self._joint_click(0, -1))

self.ui.INC2.clicked.connect(lambda: self._joint_click(1, +1))

```

```

self.ui.DES2.clicked.connect(lambda: self._joint_click(1, -1))

self.ui.INC3.clicked.connect(lambda: self._joint_click(2, +1))

self.ui.DES3.clicked.connect(lambda: self._joint_click(2, -1))

self.ui.INC4.clicked.connect(lambda: self._joint_click(3, +1))

self.ui.DES4.clicked.connect(lambda: self._joint_click(3, -1))


# ===== GÁN NÚT TRỰC (đúng theo cơ cấu của bạn) =====

# X: điều khiển LÊN/XUỐNG => chỉ đôi KHỚP 3

self.ui.XD.clicked.connect(lambda: self._axis_x_click(+1)) # X+: lên
self.ui.XT.clicked.connect(lambda: self._axis_x_click(-1)) # X-: xuống

# Z: điều khiển THU/RA => chỉ đôi KHỚP 2

self.ui.ZD.clicked.connect(lambda: self._axis_z_click(+1)) # Z+: vươn ra
self.ui.ZT.clicked.connect(lambda: self._axis_z_click(-1)) # Z-: thu vào


# Reset

if hasattr(self.ui, "O"):

    self.ui.O.clicked.connect(self.reset_all)


# Mục tiêu hiện tại (để điền bảng)

self.target_r, self.target_z = self.forward_rz(self.goc)

```

```

# Hiện thị ban đầu

self._upd_angle_labels()

self.update_donghoc_table(self.build_T(self.goc))


# ----- KHỚP (dùng spinbox sodo/tocdo) -----

def _joint_click(self, k: int, direction: int):

    step_deg = float(self.ui.sodo.value()) * (1 if direction > 0 else -1)

    t_ms = int(self.ui.tocdo.value())

    new_deg = clamp(self.goc[k] + step_deg, MIN_ANGLE, MAX_ANGLE)

    if abs(new_deg - self.goc[k]) < 1e-6: return

    self.goc[k] = new_deg

    self._send_joint_t(k, t_ms=t_ms)

    self._post_update()


# ----- TRỤC X/Z (KHÔNG dùng spinbox) -----

def _axis_z_click(self, sign: int):

    """

    Z+: vươn ra (tăng r) — chỉ đổi KHỚP 2.

    Z-: thu vào (giảm r).

    Tự dò chiều: nếu tăng t2 mà r giảm thì tự đảo dấu.

    """

```

```

step = AXIS_J2_STEP_DEG * (1 if sign > 0 else -1)

r0, _ = self.forward_rz(self.goc)

test = self.goc[:]

test[1] = clamp(test[1] + step, MIN_ANGLE, MAX_ANGLE)

r1, _ = self.forward_rz(test)

want_increase = (sign > 0)

if (r1 > r0) != want_increase:

    step = -step # đảo chiều nếu lắp ngược

new2 = clamp(self.goc[1] + step, MIN_ANGLE, MAX_ANGLE)

if abs(new2 - self.goc[1]) < 1e-6: return

self.goc[1] = new2

self._send_joint_t(1, t_ms=AXIS_T_MS)

self._post_update()


def _axis_x_click(self, sign: int):

    """

    X+: lên (tăng z) — chỉ đổi KHỚP 3.

    X-: xuống (giảm z).

    Tự dò chiều: nếu tăng t3 mà z giảm thì đảo dấu.

    """

    step = AXIS_J3_STEP_DEG * (1 if sign > 0 else -1)

```

```

_, z0 = self.forward_rz(self.goc)

test = self.goc[:]

test[2] = clamp(test[2] + step, MIN_ANGLE, MAX_ANGLE)

_, z1 = self.forward_rz(test)

want_increase = (sign > 0)

if (z1 > z0) != want_increase:

    step = -step # đảo chiều nếu lắ ngược

new3 = clamp(self.goc[2] + step, MIN_ANGLE, MAX_ANGLE)

if abs(new3 - self.goc[2]) < 1e-6: return

self.goc[2] = new3

self._send_joint_t(2, t_ms=AXIS_T_MS)

self._post_update()


# ----- FK -----

def forward_rz(self, ang_deg):

    a1, a2, a3, a4 = [math.radians(x) for x in ang_deg]

    L1, L2, L3, L4 = LINKS_MM['L1'], LINKS_MM['L2'], LINKS_MM['L3'],
LINKS_MM['L4']

    r = (L2*math.cos(a2) + L3*math.cos(a2+a3) + L4*math.cos(a2+a3+a4))

    z = (L1 + L2*math.sin(a2) + L3*math.sin(a2+a3) + L4*math.sin(a2+a3+a4))

    return r, z

```

```

# ----- Gửi lệnh -----

def _send_joint_t(self, k: int, t_ms: int = 350):

    send_angle = clamp(self.goc[k] + OFFSETS[k], MIN_ANGLE, MAX_ANGLE)

    pulse = int(500 + 2000*(send_angle/180.0))

    for sid in SERVO_CHANNEL[k]:

        cmd = f"#{sid}P{pulse}T{int(t_ms)}\r\n"

        try:

            if self.ser: self.ser.write(cmd.encode())

        except Exception as e:

            print("Serial write error:", e)

        print("Sent:", cmd.strip())

def _send_all(self, t_ms=600):

    for k in range(4):

        self._send_joint_t(k, t_ms=t_ms)

# ----- Reset -----

def reset_all(self):

    self.goc = [90.0, 90.0, 90.0, 90.0]

    for _ in range(2):

```

```

        self._send_all(t_ms=700)

        time.sleep(0.05)

        self._post_update()

        print("Reset: 1–4 về 90°.")

# ----- UI / Bảng -----

def _upd_angle_labels(self):

    try:

        self.ui.goc1.setText(f'{int(round(self.goc[0]))}°')

        self.ui.goc2.setText(f'{int(round(self.goc[1]))}°')

        self.ui.goc3.setText(f'{int(round(self.goc[2]))}°')

        self.ui.goc4.setText(f'{int(round(self.goc[3]))}°')

    except:

        pass

def build_T(self, ang):

    a1, a2, a3, a4 = [math.radians(x) for x in ang]

    L1, L2, L3, L4 = LINKS_MM['L1'], LINKS_MM['L2'], LINKS_MM['L3'],
LINKS_MM['L4']

    r = (L2*math.cos(a2) + L3*math.cos(a2+a3) + L4*math.cos(a2+a3+a4))

    z = (L1 + L2*math.sin(a2) + L3*math.sin(a2+a3) + L4*math.sin(a2+a3+a4))

```



```

x = r*math.cos(a1); y = r*math.sin(a1)

yaw = a1; pitch = a2 + a3 + a4

cy, sy = math.cos(yaw), math.sin(yaw)

cp, sp = math.cos(pitch), math.sin(pitch)

R00 = cy*cp; R01 = -sy; R02 = cy*sp
R10 = sy*cp; R11 = cy; R12 = sy*sp
R20 = -sp; R21 = 0.0; R22 = cp
return [[R00, R01, R02, x],
        [R10, R11, R12, y],
        [R20, R21, R22, z],
        [0.0, 0.0, 0.0, 1.0]]

def update_donghoc_table(self, T=None):
    if T is None: T = self.build_T(self.goc)
    try:
        for i in range(4):
            for j in range(4):
                self.ui.donghoc.setItem(i, j, QtWidgets.QTableWidgetItem(f"{T[i][j]:.2f}"))
    except:
        pass

def _post_update(self):
    self._upd_angle_labels()
    try:
        self.target_r, self.target_z = self.forward_rz(self.goc)
    except:
        pass
    self.update_donghoc_table(self.build_T(self.goc))

```

```
if __name__ == "__main__":
    app = QtWidgets.QApplication(sys.argv)
    w = MainWindow()
    w.show()
    sys.exit(app.exec())
```

Code qt.py

```
# Form implementation generated from reading ui file 'test1.ui'
```

```
#
```

```
# Created by: PyQt6 UI code generator 6.9.1
```

```
#
```

```
# WARNING: Any manual changes made to this file will be lost when pyuic6 is
```

```
# run again. Do not edit this file unless you know what you are doing.
```

```
from PyQt6 import QtCore, QtGui, QtWidgets
```

```
class Ui_mainWindow(object):
```

```
    def setupUi(self, mainWindow):
```

```
        mainWindow.setObjectName("mainWindow")
```

```
        mainWindow.resize(1234, 864)
```

```
        self.centralwidget = QtWidgets.QWidget(parent=mainWindow)
```

```
        self.centralwidget.setObjectName("centralwidget")
```

```
self.ZD = QtWidgets.QPushButton(parent=self.centralwidget)

self.ZD.setGeometry(QtCore.QRect(170, 50, 121, 51))

self.ZD.setObjectName("ZD")

self.XT = QtWidgets.QPushButton(parent=self.centralwidget)

self.XT.setGeometry(QtCore.QRect(300, 50, 121, 51))

self.XT.setObjectName("XT")

self.XD = QtWidgets.QPushButton(parent=self.centralwidget)

self.XD.setGeometry(QtCore.QRect(40, 170, 121, 51))

self.XD.setObjectName("XD")

self.ZT = QtWidgets.QPushButton(parent=self.centralwidget)

self.ZT.setGeometry(QtCore.QRect(170, 170, 121, 51))

self.ZT.setObjectName("ZT")

self.O = QtWidgets.QPushButton(parent=self.centralwidget)

self.O.setGeometry(QtCore.QRect(170, 110, 121, 51))

self.O.setObjectName("O")

self.textEdit = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit.setGeometry(QtCore.QRect(180, 10, 251, 31))

self.textEdit.setObjectName("textEdit")

self.donghoc = QtWidgets.QTableWidget(parent=self.centralwidget)

self.donghoc.setGeometry(QtCore.QRect(430, 320, 521, 201))

self.donghoc.setObjectName("donghoc")
```

```
self.donghoc.setColumnCount(4)

self.donghoc.setRowCount(4)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setVerticalHeaderItem(0, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setVerticalHeaderItem(1, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setVerticalHeaderItem(2, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setVerticalHeaderItem(3, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setHorizontalHeaderItem(0, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setHorizontalHeaderItem(1, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setHorizontalHeaderItem(2, item)

item = QtWidgets.QTableWidgetItem()

self.donghoc.setHorizontalHeaderItem(3, item)

self.sodo = QtWidgets.QSpinBox(parent=self.centralwidget)

self.sodo.setGeometry(QtCore.QRect(230, 270, 91, 41))

self.sodo.setMinimum(1)
```

```
self.sodo.setMaximum(10)

self.sodo.setObjectName("sodo")

self.goc1 = QtWidgets.QLabel(parent=self.centralwidget)

self.goc1.setGeometry(QtCore.QRect(910, 60, 63, 51))

self.goc1.setText("")

self.goc1.setObjectName("goc1")

self.goc4 = QtWidgets.QLabel(parent=self.centralwidget)

self.goc4.setGeometry(QtCore.QRect(910, 240, 63, 51))

self.goc4.setText("")

self.goc4.setObjectName("goc4")

self.DES2 = QtWidgets.QPushButton(parent=self.centralwidget)

self.DES2.setGeometry(QtCore.QRect(770, 120, 121, 51))

self.DES2.setObjectName("DES2")

self.textEdit_6 = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit_6.setGeometry(QtCore.QRect(480, 70, 131, 41))

self.textEdit_6.setObjectName("textEdit_6")

self.INC3 = QtWidgets.QPushButton(parent=self.centralwidget)

self.INC3.setGeometry(QtCore.QRect(640, 180, 121, 51))

self.INC3.setObjectName("INC3")

self.DES4 = QtWidgets.QPushButton(parent=self.centralwidget)

self.DES4.setGeometry(QtCore.QRect(770, 240, 121, 51))
```

```
self.DES4.setObjectName("DES4")

self.DES1 = QtWidgets.QPushButton(parent=self.centralwidget)

self.DES1.setGeometry(QtCore.QRect(770, 60, 121, 51))

self.DES1.setObjectName("DES1")

self.textEdit_3 = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit_3.setGeometry(QtCore.QRect(480, 250, 131, 41))

self.textEdit_3.setObjectName("textEdit_3")

self.textEdit_4 = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit_4.setGeometry(QtCore.QRect(480, 190, 131, 41))

self.textEdit_4.setObjectName("textEdit_4")

self.textEdit_7 = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit_7.setGeometry(QtCore.QRect(670, 20, 201, 21))

self.textEdit_7.setObjectName("textEdit_7")

self.goc2 = QtWidgets.QLabel(parent=self.centralwidget)

self.goc2.setGeometry(QtCore.QRect(910, 120, 63, 51))

self.goc2.setText("")

self.goc2.setObjectName("goc2")

self.DES3 = QtWidgets.QPushButton(parent=self.centralwidget)

self.DES3.setGeometry(QtCore.QRect(770, 180, 121, 51))

self.DES3.setObjectName("DES3")

self.textEdit_5 = QtWidgets.QTextEdit(parent=self.centralwidget)
```

```
self.textEdit_5.setGeometry(QRect(480, 130, 131, 41))
self.textEdit_5.setObjectName("textEdit_5")
self.INC2 = QtWidgets.QPushButton(parent=self.centralwidget)
self.INC2.setGeometry(QRect(640, 120, 121, 51))
self.INC2.setObjectName("INC2")
self.INC4 = QtWidgets.QPushButton(parent=self.centralwidget)
self.INC4.setGeometry(QRect(640, 240, 121, 51))
self.INC4.setObjectName("INC4")
self.goc3 = QtWidgets.QLabel(parent=self.centralwidget)
self.goc3.setGeometry(QRect(910, 180, 63, 51))
self.goc3.setText("")
self.goc3.setObjectName("goc3")
self.INC1 = QtWidgets.QPushButton(parent=self.centralwidget)
self.INC1.setGeometry(QRect(640, 60, 121, 51))
self.INC1.setObjectName("INC1")
self.tocdo = QtWidgets.QSpinBox(parent=self.centralwidget)
self.tocdo.setGeometry(QRect(230, 320, 91, 41))
self.tocdo.setMinimum(100)
self.tocdo.setMaximum(2000)
self.tocdo.setObjectName("tocdo")
self.textEdit_2 = QtWidgets.QTextEdit(parent=self.centralwidget)
```

```

self.textEdit_2.setGeometry(QQtCore.QRect(10, 270, 191, 41))

self.textEdit_2.setObjectName("textEdit_2")

self.textEdit_8 = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit_8.setGeometry(QQtCore.QRect(10, 320, 191, 41))

self.textEdit_8.setObjectName("textEdit_8")

self.textEdit_9 = QtWidgets.QTextEdit(parent=self.centralwidget)

self.textEdit_9.setGeometry(QQtCore.QRect(10, 370, 351, 311))

self.textEdit_9.setObjectName("textEdit_9")

mainWindow.setCentralWidget(self.centralwidget)

self.menubar = QtWidgets.QMenuBar(parent=mainWindow)

self.menubar.setGeometry(QQtCore.QRect(0, 0, 1234, 26))

self.menubar.setObjectName("menubar")

mainWindow.setMenuBar(self.menubar)

self.statusbar = QtWidgets.QStatusBar(parent=mainWindow)

self.statusbar.setObjectName("statusbar")

mainWindow.setStatusBar(self.statusbar)


self.retranslateUi(mainWindow)

QtCore.QMetaObject.connectSlotsByName(mainWindow)


def retranslateUi(self, mainWindow):

```



```

_translate = QtCore.QCoreApplication.translate

mainWindow.setWindowTitle(_translate("mainWindow", "MainWindow"))

self.ZD.setText(_translate("mainWindow", "Z+"))

self.XT.setText(_translate("mainWindow", "X-"))

self.XD.setText(_translate("mainWindow", "X+"))

self.ZT.setText(_translate("mainWindow", "Z-"))

self.O.setText(_translate("mainWindow", "HOME"))

self.textEdit.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC \
//W3C//DTD HTML 4.0//EN\
\"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"

"<html><head><meta name=\
\"qrichtext\
\" content=\
\"1\
\" /><meta charset=\
\"utf-8\
\" /><style type=\
\"text/css\
\">\n"

"p, li { white-space: pre-wrap; }\n"

"hr { height: 1px; border-width: 0; }\n"

"li.unchecked::marker { content: \
\"\\2610\
\"; }\n"

"li.checked::marker { content: \
\"\\2612\
\"; }\n"

"</style></head><body style=\
\" font-family:\
\'Segoe UI\
\' font-size:9pt; font-weight:400;
font-style:normal;\
\">\n"

"<p align=\
\"center\
\" style=\
\" margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\
\"><span style=\
\" font-
family:\
\'Times New Roman\
\' font-size:12pt; font-weight:700; color:#000000;
background-color:#ffffff;\
\">POSITION CONTROL</span></p></body></html>"))

self.DES2.setText(_translate("mainWindow", "DES"))

```

```

        self.textEdit_6.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"

"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"
/><style type=\"text/css\">\n"

"p, li { white-space: pre-wrap; }\n"

"hr { height: 1px; border-width: 0; }\n"

"li.unchecked::marker { content: \"\\2610\"; }\n"

"li.checked::marker { content: \"\\2612\"; }\n"

"</style></head><body style=\" font-family:'Segoe UI'; font-size:9pt; font-weight:400;
font-style:normal;\">\n"

"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-
size:14pt;\">Khóp 1</span></p></body></html>"))

```

```

        self.INC3.setText(_translate("mainWindow", "INC"))

```

```

        self.DES4.setText(_translate("mainWindow", "DES"))

```

```

        self.DES1.setText(_translate("mainWindow", "DES"))

```

```

        self.textEdit_3.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"

"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"
/><style type=\"text/css\">\n"

"p, li { white-space: pre-wrap; }\n"

"hr { height: 1px; border-width: 0; }\n"

"li.unchecked::marker { content: \"\\2610\"; }\n"

```

```
"li.checked::marker { content: '\\2612\\'; }\\n"
```

```
"</style></head><body style=\" font-family:\\'Segoe UI\\'; font-size:9pt; font-weight:400; font-style:normal;\">\\n"
```

```
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:14pt;\">Khóp 4</span></p></body></html>"))
```

```
self.textEdit_4.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC  
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\\n"
```

```
"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"  
/><style type=\"text/css\">\\n"
```

```
"p, li { white-space: pre-wrap; }\\n"
```

```
"hr { height: 1px; border-width: 0; }\\n"
```

```
"li.unchecked::marker { content: '\\2610\\'; }\\n"
```

```
"li.checked::marker { content: '\\2612\\'; }\\n"
```

```
"</style></head><body style=\" font-family:\\'Segoe UI\\'; font-size:9pt; font-weight:400; font-style:normal;\">\\n"
```

```
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:14pt;\">Khóp 3</span></p></body></html>"))
```

```
self.textEdit_7.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC  
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\\n"
```

```
"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"  
/><style type=\"text/css\">\\n"
```

```
"p, li { white-space: pre-wrap; }\\n"
```

```

"hr { height: 1px; border-width: 0; }\n"

"li unchecked::marker { content: \"\\2610\"; }\n"

"li checked::marker { content: \"\\2612\"; }\n"

"</style></head><body style=\" font-family:\\'Segoe UI\\'; font-size:9pt; font-weight:400;
font-style:normal;\">\n"

"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-
family:\\'Times New Roman\\'; font-size:12pt; font-weight:700; color:#000000;
background-color:#ffffff;\">JOIN CONTROL</span></p></body></html>\"))

    self.DES3.setText(_translate("mainWindow", "DES"))

    self.textEdit_5.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"

"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"
/><style type=\"text/css\">\n"

"p, li { white-space: pre-wrap; }\n"

"hr { height: 1px; border-width: 0; }\n"

"li unchecked::marker { content: \"\\2610\"; }\n"

"li checked::marker { content: \"\\2612\"; }\n"

"</style></head><body style=\" font-family:\\'Segoe UI\\'; font-size:9pt; font-weight:400;
font-style:normal;\">\n"

"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-
size:14pt;\">Khóp 2</span></p></body></html>\"))

    self.INC2.setText(_translate("mainWindow", "INC"))

```

```

self.INC4.setText(_translate("mainWindow", "INC"))

self.INC1.setText(_translate("mainWindow", "INC"))

self.textEdit_2.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"

"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"
/><style type=\"text/css\">\n"

"p, li { white-space: pre-wrap; }\n"

"hr { height: 1px; border-width: 0; }\n"

"li.unchecked::marker { content: \"\\2610\"; }\n"

"li.checked::marker { content: \"\\2612\"; }\n"

"</style></head><body style=\" font-family:'Segoe UI'; font-size:9pt; font-weight:400;
font-style:normal;\">\n"

"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px;
margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-
size:11pt;\">Bước duy chuy  n</span></p></body></html>"))

self.textEdit_8.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"

"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"
/><style type=\"text/css\">\n"

"p, li { white-space: pre-wrap; }\n"

"hr { height: 1px; border-width: 0; }\n"

"li.unchecked::marker { content: \"\\2610\"; }\n"

"li.checked::marker { content: \"\\2612\"; }\n"

```

```
"</style></head><body style=\" font-family: 'Segoe UI'; font-size:9pt; font-weight:400; font-style:normal;\">\n"
```

```
"<p align=\"center\" style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:11pt;\">Tốc độ duy chuyển</span></p></body></html>"))
```

```
self.textEdit_9.setHtml(_translate("mainWindow", "<!DOCTYPE HTML PUBLIC  
\"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"
```

```
"<html><head><meta name=\"qrichtext\" content=\"1\" /><meta charset=\"utf-8\"  
</><style type=\"text/css\">\n"
```

```
"p, li { white-space: pre-wrap; }\n"
```

```
"hr { height: 1px; border-width: 0; }\n"
```

```
"li.unchecked::marker { content: \"\\2610\"; }\n"
```

```
"li.checked::marker { content: \"\\2612\"; }\n"
```

```
"</style></head><body style=\" font-family: 'Segoe UI'; font-size:9pt; font-weight:400; font-style:normal;\">\n"
```

```
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:12pt;\">note:</span></p>\n"
```

```
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:12pt;\">- Bước duy chuyển (độ)</span></p>\n"
```

```
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:12pt;\">VD: bước duy chuyển = 1 => mỗi lần duy chuyển các khớp quay 1 độ.</span></p>\n"
```

```
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:12pt;\">- Tốc độ duy chuyên ( ms ) </span></p>\n"
```

```
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-size:12pt;\">VD: 100 ms , mỗi bước động cơ quay trong khoảng thời gian 100ms ( 0,1s )</span></p>\n"
```

```
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:12pt;\"><br /></p>\n"
```

```
"<p style=\"-qt-paragraph-type:empty; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:12pt;\"><br /></p></body></html>"))
```

```
if __name__ == "__main__":
```

```
    import sys
```

```
    app = QtWidgets.QApplication(sys.argv)
```

```
    mainWindow = QtWidgets.QMainWindow()
```

```
    ui = Ui_mainWindow()
```

```
    ui.setupUi(mainWindow)
```

```
    mainWindow.show()
```

```
    sys.exit(app.exec())
```

Code qt.ui

```
<?xml version="1.0" encoding="UTF-8"?>

<ui version="4.0">

<class>mainWindow</class>

<widget class="QMainWindow" name="mainWindow">

<property name="geometry">

<rect>

<x>0</x>

<y>0</y>

<width>1234</width>

<height>864</height>

</rect>

</property>

<property name="windowTitle">

<string>MainWindow</string>

</property>

<widget class="QWidget" name="centralwidget">

<widget class="QPushButton" name="ZD">

<property name="geometry">

<rect>

<x>170</x>

<y>50</y>
```



```
<width>121</width>

<height>51</height>

</rect>

</property>

<property name="text">

  <string>Z+</string>

</property>

</widget>

<widget class="QPushButton" name="XT">

  <property name="geometry">

    <rect>

      <x>300</x>

      <y>50</y>

      <width>121</width>

      <height>51</height>

    </rect>

  </property>

  <property name="text">

    <string>X-</string>

  </property>

</widget>
```

```
<widget class="QPushButton" name="XD">
```

```
<property name="geometry">
```

```
<rect>
```

```
<x>40</x>
```

```
<y>170</y>
```

```
<width>121</width>
```

```
<height>51</height>
```

```
</rect>
```

```
</property>
```

```
<property name="text">
```

```
<string>X+</string>
```

```
</property>
```

```
</widget>
```

```
<widget class="QPushButton" name="ZT">
```

```
<property name="geometry">
```

```
<rect>
```

```
<x>170</x>
```

```
<y>170</y>
```

```
<width>121</width>
```

```
<height>51</height>
```

```
</rect>
```

```
</property>

<property name="text">

  <string>Z-</string>

</property>

</widget>

<widget class="QPushButton" name="O">

  <property name="geometry">

    <rect>

      <x>170</x>

      <y>110</y>

      <width>121</width>

      <height>51</height>

    </rect>

  </property>

  <property name="text">

    <string>HOME</string>

  </property>

</widget>

<widget class="QTextEdit" name="textEdit">

  <property name="geometry">

    <rect>
```

```

<x>180</x>

<y>10</y>

<width>251</width>

<height>31</height>

</rect>

</property>

<property name="html">

  <string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;;&gt;
&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;richtext&quot;
content=&quot;l&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style
type=&quot;text/css&quot;;&gt;

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: &quot;\2610&quot;; }

li.checked::marker { content: &quot;\2612&quot;; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;&quot;;&gt;

&lt;p align=&quot;center&quot; style=&quot; margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;&quot;;&gt;&lt;span style=&quot; font-family:'Times New Roman'; font-
size:12pt; font-weight:700; color:#000000; background-
color:#ffffff;&quot;;&gt;POSITION
CONTROL&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

```

```
</property>

</widget>

<widget class="QTableWidget" name="donghoc">

  <property name="geometry">

    <rect>

      <x>430</x>

      <y>320</y>

      <width>521</width>

      <height>201</height>

    </rect>

  </property>

  <row>

    <property name="text">

      <string/>

    </property>

  </row>

  <row>

    <property name="text">

      <string/>

    </property>

  </row>
```

```
<row>

  <property name="text">

    <string/>

  </property>

</row>

<row>

  <property name="text">

    <string/>

  </property>

</row>

<column>

  <property name="text">

    <string/>

  </property>

</column>

<column>

  <property name="text">

    <string/>

  </property>

</column>

<column>
```

```
<property name="text">

  <string/>

</property>

</column>

<column>

  <property name="text">

    <string/>

  </property>

</column>

</widget>

<widget class="QSpinBox" name="sodo">

  <property name="geometry">

    <rect>

      <x>230</x>

      <y>270</y>

      <width>91</width>

      <height>41</height>

    </rect>

  </property>

  <property name="minimum">

    <number>1</number>
```

```
</property>

<property name="maximum">

  <number>10</number>

</property>

</widget>

<widget class="QLabel" name="goc1">

  <property name="geometry">

    <rect>

      <x>910</x>

      <y>60</y>

      <width>63</width>

      <height>51</height>

    </rect>

  </property>

  <property name="text">

    <string/>

  </property>

</widget>

<widget class="QLabel" name="goc4">

  <property name="geometry">

    <rect>
```



```
<x>910</x>

<y>240</y>

<width>63</width>

<height>51</height>

</rect>

</property>

<property name="text">

  <string/>

</property>

</widget>

<widget class="QPushButton" name="DES2">

  <property name="geometry">

    <rect>

      <x>770</x>

      <y>120</y>

      <width>121</width>

      <height>51</height>

    </rect>

  </property>

  <property name="text">

    <string>DES</string>
```

```

</property>

</widget>

<widget class="QTextEdit" name="textEdit_6">

  <property name="geometry">

    <rect>

      <x>480</x>

      <y>70</y>

      <width>131</width>

      <height>41</height>

    </rect>

  </property>

  <property name="html">

    <string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;;&gt;
&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot;
content=&quot;l&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style
type=&quot;text/css&quot;;&gt;
p, li { white-space: pre-wrap; }
hr { height: 1px; border-width: 0; }
li.unchecked::marker { content: &quot;\2610&quot;; }
li.checked::marker { content: &quot;\2612&quot;; }

```

```
</style></head><body style=" font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;">
```

```
<p align="center" style=" margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;"><span style=" font-size:14pt;">Khó  
l</span></p></body></html></string>
```

```
</property>
```

```
</widget>
```

```
<widget class="QPushButton" name="INC3">
```

```
<property name="geometry">
```

```
<rect>
```

```
<x>640</x>
```

```
<y>180</y>
```

```
<width>121</width>
```

```
<height>51</height>
```

```
</rect>
```

```
</property>
```

```
<property name="text">
```

```
<string>INC</string>
```

```
</property>
```

```
</widget>
```

```
<widget class="QPushButton" name="DES4">
```

```
<property name="geometry">

<rect>

<x>770</x>

<y>240</y>

<width>121</width>

<height>51</height>

</rect>

</property>

<property name="text">

<string>DES</string>

</property>

</widget>

<widget class="QPushButton" name="DES1">

<property name="geometry">

<rect>

<x>770</x>

<y>60</y>

<width>121</width>

<height>51</height>

</rect>

</property>
```

```

<property name="text">

<string>DES</string>

</property>

</widget>

<widget class="QTextEdit" name="textEdit_3">

<property name="geometry">

<rect>

<x>480</x>

<y>250</y>

<width>131</width>

<height>41</height>

</rect>

</property>

<property name="html">

<string>&lt;!DOCTYPE HTML PUBLIC &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;

&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;richtext&quot;

content=&quot;l&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style

type=&quot;text/css&quot;&gt;

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: &quot;\2610&quot;; }

```

```

li.checked::marker { content: &quot;\2612&quot;; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;&quot;&gt;

&lt;p align=&quot;center&quot; style=&quot; margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;&quot;&gt;&lt;span style=&quot; font-size:14pt;&quot;&gt;Khó p
4&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

</property>

</widget>

<widget class="QTextEdit" name="textEdit_4">

<property name="geometry">

<rect>

<x>480</x>

<y>190</y>

<width>131</width>

<height>41</height>

</rect>

</property>

<property name="html">

<string>&lt;!DOCTYPE HTML PUBLIC &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;

```

```

<html><head><meta name="richtext"
content="1" /><meta charset="utf-8" /><style
type="text/css">

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: "\2610"; }

li.checked::marker { content: "\2612"; }

</style></head><body style="font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;">

<p align="center" style="margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;"><span style="font-size:14pt;">Khó p
3</span></p></body></html></string>

</property>

</widget>

<widget class="QTextEdit" name="textEdit_7">

<property name="geometry">

<rect>

<x>670</x>

<y>20</y>

<width>201</width>

<height>21</height>

</rect>

```

</property>

<property name="html">

<string><!DOCTYPE HTML PUBLIC " -//W3C//DTD HTML
4.0//EN" "http://www.w3.org/TR/REC-html40/strict.dtd";>
<html><head><meta name="richtext"
content="1" /><meta charset="utf-8" /><style
type="text/css";>

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: "\2610"; }

li.checked::marker { content: "\2612"; }

</style></head><body style="font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;";>

<p align="center" style="margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;";><span style="font-family:'Times New Roman'; font-
size:12pt; font-weight:700; color:#000000; background-color:#ffffff;";>JOIN
CONTROL</p></body></html></string>

</property>

</widget>

<widget class="QLabel" name="goc2">

<property name="geometry">

<rect>


```
<x>910</x>

<y>120</y>

<width>63</width>

<height>51</height>

</rect>

</property>

<property name="text">

  <string/>

</property>

</widget>

<widget class="QPushButton" name="DES3">

  <property name="geometry">

    <rect>

      <x>770</x>

      <y>180</y>

      <width>121</width>

      <height>51</height>

    </rect>

  </property>

  <property name="text">

    <string>DES</string>
```

```

</property>

</widget>

<widget class="QTextEdit" name="textEdit_5">

  <property name="geometry">

    <rect>

      <x>480</x>

      <y>130</y>

      <width>131</width>

      <height>41</height>

    </rect>

  </property>

  <property name="html">

    <string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;;&gt;
&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot;
content=&quot;l&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style
type=&quot;text/css&quot;;&gt;
p, li { white-space: pre-wrap; }
hr { height: 1px; border-width: 0; }
li.unchecked::marker { content: &quot;\2610&quot;; }
li.checked::marker { content: &quot;\2612&quot;; }

```

```
</style></head><body style=" font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;">
```

```
<p align="center" style=" margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;"><span style=" font-size:14pt;">Khó p
2</span></p></body></html></string>
```

```
</property>
```

```
</widget>
```

```
<widget class="QPushButton" name="INC2">
```

```
<property name="geometry">
```

```
<rect>
```

```
<x>640</x>
```

```
<y>120</y>
```

```
<width>121</width>
```

```
<height>51</height>
```

```
</rect>
```

```
</property>
```

```
<property name="text">
```

```
<string>INC</string>
```

```
</property>
```

```
</widget>
```

```
<widget class="QPushButton" name="INC4">
```

```
<property name="geometry">

<rect>

<x>640</x>

<y>240</y>

<width>121</width>

<height>51</height>

</rect>

</property>

<property name="text">

<string>INC</string>

</property>

</widget>

<widget class="QLabel" name="goc3">

<property name="geometry">

<rect>

<x>910</x>

<y>180</y>

<width>63</width>

<height>51</height>

</rect>

</property>
```

```

<property name="text">

<string/>

</property>

</widget>

<widget class="QPushButton" name="INC1">

<property name="geometry">

<rect>

<x>640</x>

<y>60</y>

<width>121</width>

<height>51</height>

</rect>

</property>

<property name="text">

<string>INC</string>

</property>

</widget>

<widget class="QSpinBox" name="tocdo">

<property name="geometry">

<rect>

<x>230</x>

```

```
<y>320</y>

<width>91</width>

<height>41</height>

</rect>

</property>

<property name="minimum">

  <number>100</number>

</property>

<property name="maximum">

  <number>2000</number>

</property>

</widget>

<widget class="QTextEdit" name="textEdit_2">

  <property name="geometry">

    <rect>

      <x>10</x>

      <y>270</y>

      <width>191</width>

      <height>41</height>

    </rect>

  </property>
```

```

<property name="html">

  <string>&lt;!DOCTYPE HTML PUBLIC &quot; -//W3C//DTD HTML
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;

  &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot;
content=&quot;l&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style
type=&quot;text/css&quot;&gt;

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: &quot;\2610&quot;; }

li.checked::marker { content: &quot;\2612&quot;; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;&quot;&gt;

&lt;p align=&quot;center&quot; style=&quot; margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;&quot;&gt;&lt;span style=&quot; font-size:11pt;&quot;&gt;Bước duy
chuyển&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

</property>

</widget>

<widget class="QTextEdit" name="textEdit_8">

  <property name="geometry">

    <rect>

      <x>10</x>

      <y>320</y>

```

```

<width>191</width>

<height>41</height>

</rect>

</property>

<property name="html">

    <string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;
&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;richtext&quot;
content=&quot;l&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style
type=&quot;text/css&quot;&gt;

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: &quot;\2610&quot;; }

li.checked::marker { content: &quot;\2612&quot;; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot;font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;&quot;&gt;

&lt;p align=&quot;center&quot; style=&quot;margin-top:0px; margin-bottom:0px;
margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
indent:0px;&quot;&gt;&lt;span style=&quot;font-size:11pt;&quot;&gt;Tốc độ duy
chuyển&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

</property>

</widget>

<widget class="QTextEdit" name="textEdit_9">

```



```

<property name="geometry">

<rect>

<x>10</x>

<y>370</y>

<width>351</width>

<height>311</height>

</rect>

</property>

<property name="html">

<string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML
4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;;&gt;
&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;richtext&quot;
content=&quot;1&quot; /&gt;&lt;meta charset=&quot;utf-8&quot; /&gt;&lt;style
type=&quot;text/css&quot;&gt;

p, li { white-space: pre-wrap; }

hr { height: 1px; border-width: 0; }

li.unchecked::marker { content: &quot;\2610&quot;; }

li.checked::marker { content: &quot;\2612&quot;; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Segoe UI'; font-size:9pt;
font-weight:400; font-style:normal;&quot;&gt;

&lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-
right:0px; -qt-block-indent:0; text-indent:0px;&quot;&gt;&lt;span style=&quot; font-
size:12pt;&quot;&gt;note:&lt;/span&gt;&lt;/p&gt;

```

<p style="margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;">- Bước duy chuyển (độ)</p>

<p style="margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;">VD: bước duy chuyển = 1 => mỗi lần duy chuyển các khớp quay 1 độ.</p>

<p style="margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;">- Tốc độ duy chuyển (ms) </p>

<p style="margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;">VD: 100 ms , mỗi bước động cơ quay trong khoảng thời gian 100ms (0,1s)</p>

<p style="margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:12pt;">
</p>

<p style="margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px; font-size:12pt;">
</p></body></html></string>

</property>

</widget>

</widget>

<widget class="QMenuBar" name="menubar">

<property name="geometry">

```

<rect>

<x>0</x>

<y>0</y>

<width>1234</width>

<height>26</height>

</rect>

</property>

</widget>

<widget class="QStatusBar" name="statusbar"/>

</widget>

<resources/>

<connections/>

</ui>

```

4.3.1. Khởi tạo cổng Serial:

xác định cổng COM trên máy tính và baud rate (tốc độ truyền) phù hợp với thiết bị bạn muốn kết nối. Sau đó, sử dụng phần mềm như Serial Monitor (trong Arduino IDE) hoặc các công cụ tương tự để kết nối và truyền dữ liệu.

xác định cổng serial mở **kết nối Serial** bằng lệnh `Serial.begin(baudrate)`

Chọn baud rate tương thích với thiết bị bạn kết nối. giá trị phổ biến là 9600, 115200, v.v.

với Arduino, sử dụng Serial Monitor trong Arduino IDE

```
void setup() {  
    Serial.begin(9600); // Khởi động cổng serial với baud rate 9600  
}  
  
void loop() {  
    if (Serial.available() > 0) {  
        // Đọc dữ liệu từ cổng serial  
        char incomingByte = Serial.read();  
        // Xử lý dữ liệu  
        Serial.print("Bạn đã nhập: ");  
        Serial.println(incomingByte);  
    }  
}
```

Cài thư viện PySerial

Mở terminal gõ: pip install pyserial

Code Python khởi động Serial

Ví dụ kết nối tới cổng **COM3** (Windows) hoặc **/dev/ttyUSB0** (Linux), tốc độ 9600 bps:

```
import serial  
  
# Khởi động cổng Serial
```

```

ser = serial.Serial(

    port='COM3',    # Cổng serial (Windows: COMx, Linux/Mac: /dev/ttyUSBx)

    baudrate=9600,  # Tốc độ truyền

    timeout=1       # Thời gian chờ đọc (giây)

)

if ser.is_open:

    print(f"Serial đã mở: {ser.port}")

else:

    print("Không thể mở cổng Serial!")

# Gửi dữ liệu

ser.write(b'Hello Arduino\n')

# Đọc dữ liệu từ vi điều khiển

data = ser.readline().decode('utf-8').strip()

if data:

    print("Nhận được:", data)

# Đóng cổng khi xong

ser.close()

```

4.3.3. Liên kết GUI với logic:

Python GUI (Tkinter) + Logic Serial

Cài thư viện PySerial

Mở terminal gõ: pip install pyserial

```
import serial

import tkinter as tk

from tkinter import messagebox

# ===== LOGIC KẾT NỐI SERIAL =====

ser = None

def connect_serial():

    global ser

    try:

        ser = serial.Serial('COM3', 9600, timeout=1) # Thay COM3 bằng cổng của bạn

        messagebox.showinfo("Thông báo", f"Đã kết nối {ser.port}")

    except Exception as e:

        messagebox.showerror("Lỗi", f"Không thể kết nối: {e}")

def disconnect_serial():

    global ser

    if ser and ser.is_open:

        ser.close()
```

```

        messagebox.showinfo("Thông báo", "Đã ngắt kết nối")

def send_led_on():

    if ser and ser.is_open:

        ser.write(b'1')

    else:

        messagebox.showwarning("Cảnh báo", "Chưa kết nối Serial")

def send_led_off():

    if ser and ser.is_open:

        ser.write(b'0')

    else:

        messagebox.showwarning("Cảnh báo", "Chưa kết nối Serial")

# ===== TẠO GUI =====

root = tk.Tk()

root.title("Điều khiển LED qua Serial")

btn_connect = tk.Button(root, text="Kết nối", command=connect_serial)

btn_connect.pack(pady=5)

btn_disconnect = tk.Button(root, text="Ngắt kết nối", command=disconnect_serial)

btn_disconnect.pack(pady=5)

btn_on = tk.Button(root, text="Bật LED", command=send_led_on)

btn_on.pack(pady=5)

```

```
btn_off = tk.Button(root, text="Tắt LED", command=send_led_off)
```

```
btn_off.pack(pady=5)
```

```
root.mainloop()
```

cách hoạt động

GUI (Tkinter) tạo các nút bấm.

Khi bấm nút → gọi hàm **logic** tương ứng (connect_serial(), send_led_on(), ...).

Logic dùng pyserial để gửi lệnh (1 hoặc 0) qua cổng COM.

Arduino/vi điều khiển nhận lệnh và xử lý.

4.4. Tích hợp Visual Studio:

- Tạo Project Python.
- Thêm các file code .py và giao diện .ui.
- Debug để kiểm tra dữ liệu gửi/nhận.

CHƯƠNG 5: KẾT QUẢ, ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết quả đạt được:

- Giao diện Qt thân thiện, dễ điều khiển.
- Robot phản hồi nhanh, chính xác.
- Có thể lưu và chạy lại các chuyển động.

5.2. Đánh giá:

- **Ưu điểm:** Mã nguồn mở, dễ mở rộng, GUI đẹp, lập trình nhanh.

- **Nhược điểm:** Phụ thuộc vào cổng kết nối, chưa có thị giác máy tính.

5.3. Hướng phát triển:

- Thêm **camera AI** để điều khiển theo hình ảnh.
- Kết nối điều khiển **qua Wi-Fi hoặc Internet**.
- Tích hợp **tự động lập kế hoạch quỹ đạo**.

KẾT LUẬN

Báo cáo đã mô tả đầy đủ quá trình xây dựng hệ thống điều khiển cánh tay robot 6DOF bằng Python, Qt Designer và Visual Studio. Hệ thống này phù hợp cả cho **ngiên cứu học thuật** và **ứng dụng thực tế**, với ưu điểm là chi phí thấp, dễ mở rộng và bảo trì.

TÀI LIỆU THAM KHẢO

1. Mark Summerfield, *Rapid GUI Programming with Python and Qt*, Prentice Hall, 2008.
2. Qt Documentation: <https://doc.qt.io>
3. PySide6 Reference: <https://doc.qt.io/qtforpython/>
4. Arduino Serial Communication:
<https://www.arduino.cc/reference/en/language/functions/communication/serial/>
5. Craig, J. J., *Introduction to Robotics: Mechanics and Control*, Pearson, 2017.