

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
ĐẠI HỌC BÁCH KHOA THÀNH PHỐ HỒ CHÍ MINH
KHOA ĐIỆN – ĐIỆN TỬ

BÁO CÁO BÀI TẬP LỚN

ĐO LƯỜNG CÔNG NGHIỆP

ĐỀ TÀI:

**THIẾT KẾ MẠCH ĐO VÀ HIỂN THỊ VỊ TRÍ ĐỘNG CƠ
SỬ DỤNG CẢM BIẾN ENCODER**

NGUYỄN GIA KHIÊM - 1810236

GIẢNG VIÊN HƯỚNG DẪN:

NGUYỄN ĐỨC HOÀNG

TP HỒ CHÍ MINH, THÁNG 12 NĂM 2020

MỤC LỤC

1.	CƠ SỞ LÝ THUYẾT	2
1.1.	Cảm biến encoder xoay (Rotary Encoder)	2
1.1.1.	Absolute Encoder	2
1.1.2.	Incremental Encoder	3
1.2.	Nguyên lý đo và hiệu chỉnh (calibration) encoder	4
2.	NỘI DUNG THỰC HIỆN	5
2.1.	Tổng quan kết nối.....	5
2.2.	Thiết kế và thi công mạch ra chân	6
2.3.	Lập trình vi điều khiển.....	7
2.3.1.	Lập trình giao tiếp GUI máy tính	7
2.3.2.	Lưu đồ lập trình	8
2.4.	Lập trình giao diện	9
3.	KẾT QUẢ	10

1. CƠ SỞ LÝ THUYẾT

1.1. Cảm biến encoder xoay (Rotary Encoder)

Cảm biến encoder xoay (Rotary Encoder hay còn được gọi là Shaft Encoder) là một thiết bị cơ điện có khả năng chuyển đổi vị trí hoặc chuyển động góc xoay của một trục sang tín hiệu điện dạng số hay tương tự. Có hai loại cảm biến xoay chính là loại tuyệt đối (Absolute encoder) và loại tương đối (Incremental encoder). Tín hiệu ra của absolute encoder cho biết vị trí trục hiện tại, được xem như một bộ chuyển đổi góc. Tín hiệu ra của incremental encoder cung cấp thông tin về chuyển động của trục, thông tin này thường được xử lý ở vi điều khiển hoặc máy tính thành thông tin như vị trí, tốc độ và khoảng cách.

1.1.1. Absolute Encoder

Encoder tuyệt đối duy trì thông tin vị trí cả khi ngắt điện. Vị trí của encoder có sẵn ngay lập tức khi cấp nguồn. Mỗi quan hệ giữa giá trị encoder và vị trí vật lý của trục được điều khiển được thiết lập khi lắp ráp; hệ thống không cần quay lại điểm hiệu chuẩn để duy trì độ chính xác của vị trí. Một encoder tuyệt đối có nhiều vòng mã với các trọng số nhị phân khác nhau (thường dùng mã Gray – mỗi giá trị chỉ lệch nhau 1 bit) để cung cấp một từ mã đại diện cho vị trí tuyệt đối của trục trong một vòng quay. Loại encoder này thường được gọi là encoder tuyệt đối song song. Encoder tuyệt đối có nhiều vòng quay bao gồm vòng xoay có đục lỗ từ trong ra ngoài với tần số gấp đôi. Vòng mã có độ phân giải cao đo vòng quay phân đoạn và vòng mã số có độ phân giải thấp hơn ghi lại số vòng quay toàn bộ của trục.



Hình 1.1. Mô tả một encoder tuyệt đối

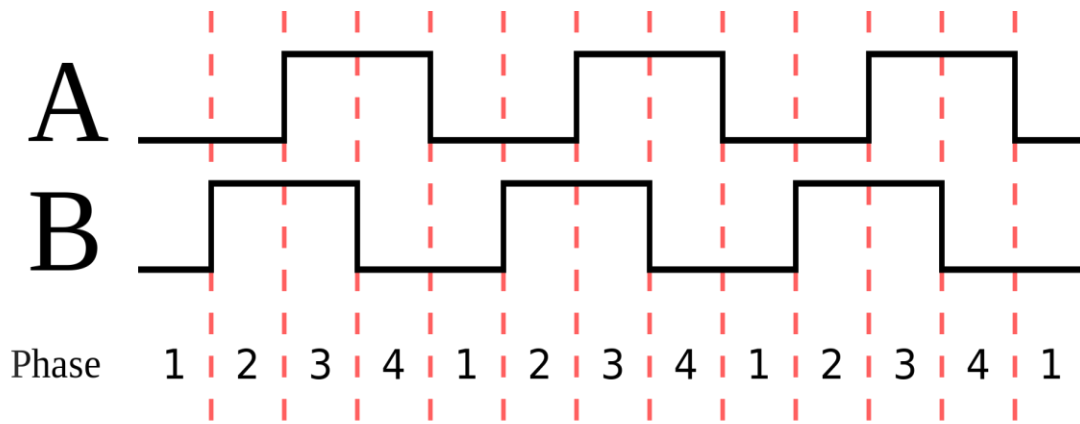
1.1.2. Incremental Encoder

Encoder tương đối được sử dụng rộng rãi nhất trong tất cả các encoder xoay do khả năng cung cấp thông tin vị trí theo thời gian thực. Độ phân giải đo lường của encoder tương đối thường không bị giới hạn. Người ta có thể tìm thấy trong thị trường các encoder tương đối với tối đa 10.000 xung mỗi vòng quay hoặc hơn. Các encoder tương đối cho ra tốc độ dữ liệu nhanh hơn so với tốc độ của hầu hết các loại encoder tuyệt đối. Do đó, encoder tương đối thường được sử dụng trong các ứng dụng yêu cầu đo chính xác vị trí và vận tốc. Encoder tương đối có thể sử dụng cảm biến cơ học, quang học hoặc từ tính để phát hiện các thay đổi vị trí quay. Loại cơ học thường được sử dụng như một "chiết áp" vận hành bằng tay trên thiết bị điện tử. Ví dụ, các đàn âm thanh gia đình và xe hơi hiện đại thường sử dụng encoder xoay cơ học làm bộ điều khiển âm lượng. Các bộ mã hóa có cảm biến cơ học yêu cầu về chuyển mạch và do đó bị hạn chế về tốc độ quay mà chúng có thể xử lý. Loại quang học được sử dụng khi có tốc độ cao hơn hoặc yêu cầu độ chính xác cao hơn.



Hình 1.2. Mô tả một encoder tương đối

Một encoder tương đối có hai tín hiệu đầu ra, A và B, tạo ra dạng sóng vuông tuần hoàn khi trục encoder xoay. Tần số dạng sóng cho biết tốc độ quay của trục và số xung cho biết quãng đường đã di chuyển, trong khi mối quan hệ pha A-B cho biết chiều quay. Một số encoder tương đối quay có đầu ra "chỉ zero" bổ sung (thường được gắn nhãn Z), phát ra một xung khi trục đi qua một góc cụ thể. Sau mỗi lần quay, tín hiệu Z được xác nhận, thường luôn ở cùng một góc, cho đến khi trạng thái AB tiếp theo thay đổi.



Hình 1.3. Hai xung A, B của encoder tương đối

Không giống như bộ mã hóa tuyệt đối, encoder tương đối không chỉ ra vị trí tuyệt đối của hệ thống cơ học mà nó được gắn vào. Do đó, để xác định vị trí tuyệt đối tại bất kỳ thời điểm cụ thể nào, cần phải "đọc xung" để vị trí tuyệt đối bằng các giá trị tương đối.

1.2. Nguyên lý đo và hiệu chỉnh (calibration) encoder

Trong bài tập lớn này, em sử dụng động cơ giảm tốc có gắn cảm biến encoder tương đối. Vì thế nguyên lý đo và hiệu chỉnh encoder sẽ được trình bày theo loại encoder tương đối.

Khi gắn encoder tương đối vào một trục của động cơ giảm tốc (có hộp số), ta có thể biết được số xung đơn A (hoặc B) khi trục đầu ra động cơ quay được một vòng theo công thức:

$$\text{số xung A khi trục đầu ra xoay 1 vòng} = \text{số lỗ đục trên 1 vòng} \times \text{tỷ số truyền}$$

Để có thể tính được vị trí và vận tốc của trục động cơ, ta phải đếm được số xung A và B. Có hai cách đo cơ bản:

- Đo tần số: Đếm số xung n trong một khoảng thời gian T .
- Đo chu kỳ: Đo khoảng thời gian T để có được n xung.

Từ đây ta có thể tính được vị trí của trục động cơ theo công thức:

$$\theta = a \times \text{số xung} + b$$

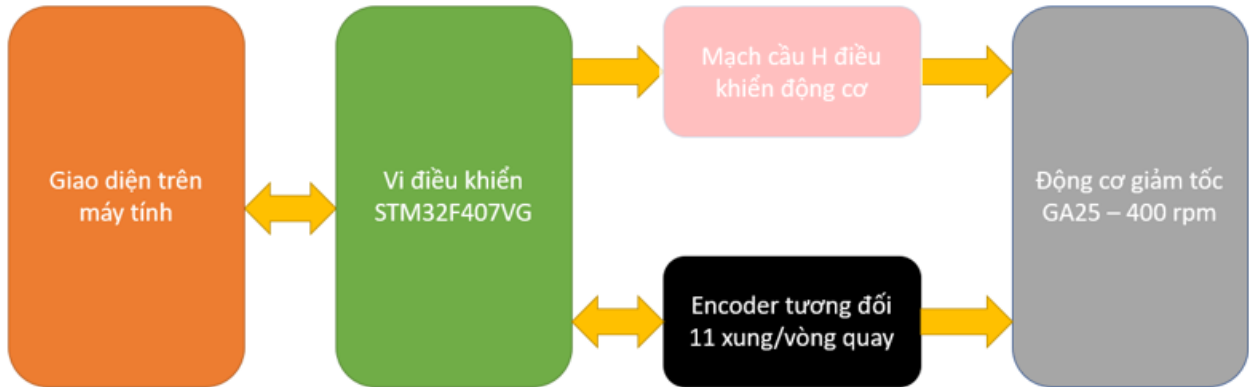
Do là encoder tương đối nên khoảng cách các lỗ đục bằng nhau và ta có thể đặt một vị trí làm zero cho nên ta có thể đặt $b = 0$. Hệ số a tỷ lệ với số xung encoder trên một vòng. Nhưng trong thực tế, ta sẽ làm một giao diện để có thể hiệu chỉnh cho nhiều encoder và động cơ mà không cần điều chỉnh trong firmware với một độ chính xác tương đối.

$$a = \frac{\theta}{\text{số xung}}$$

Với θ là góc lệch so với zero mà ta đã đặt. Khi có được a , ta sử dụng công thức phía trên để tính toán góc và tiếp tục calib cho đến khi có được hệ số a chính xác.

2. NỘI DUNG THỰC HIỆN

2.1. Tổng quan kết nối



Hình 2.1. Tổng quan kết nối phần cứng

Bài tập lớn này sử dụng encoder tương đối có thông số là 11 xung/vòng quay gắn vào trục động cơ giảm tốc GA25 – 400 rpm. Mạch cầu H sử dụng ở đây dùng để điều khiển động cơ nhằm mục đích đánh giá việc đọc xung từ encoder.

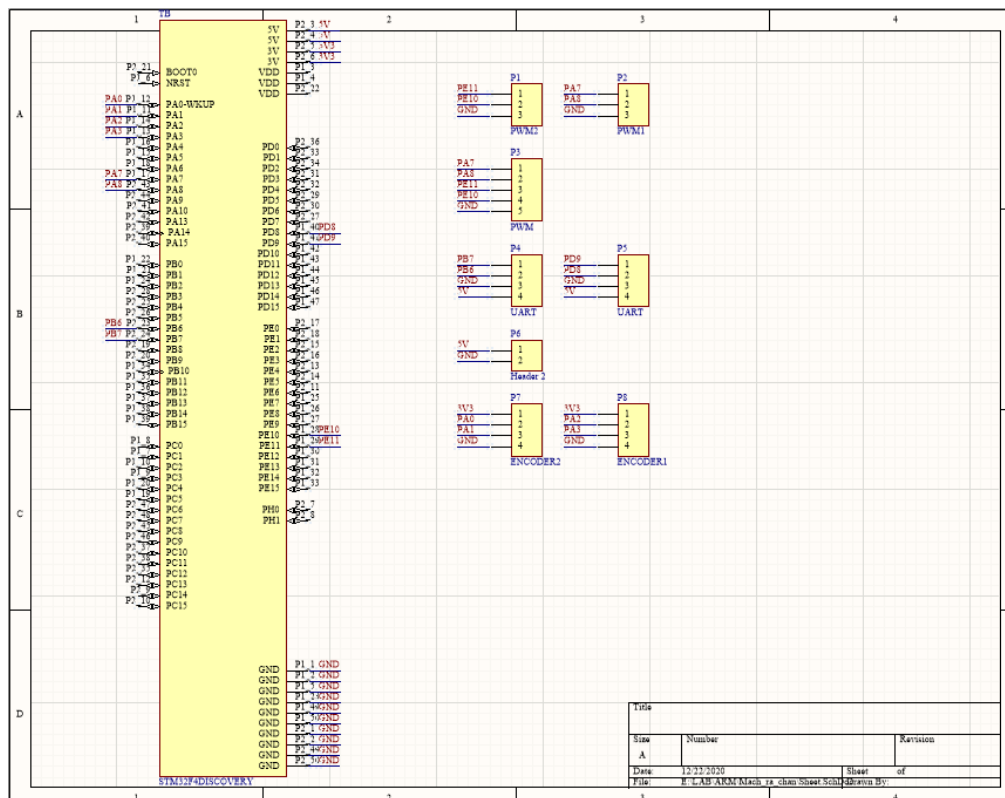
Vi điều khiển STM32F407VG là một trong những vi điều khiển mạnh mẽ ở thời điểm hiện tại, có nhiều chức năng, hỗ trợ từ cộng đồng rất lớn. Vi điều khiển được sử dụng với các chức năng: đọc xung từ encoder sử dụng chức năng Input Capture với Encoder Interface (để x4 lần xung đọc được) được hỗ trợ sẵn, giao tiếp với GUI trên máy tính bằng cổng COM với chức năng USART.

Chức năng	Chân	Ứng dụng
USART3, DMA1	PD8, PD9	Giao tiếp với GUI trên máy tính
PWM1, TIMER1	PA7, PA8	Xuất PWM cho động cơ
TIM6, TIM3		Timer thời gian dùng để ngắt
TIM5	PA0, PA1	Đọc xung từ Encoder

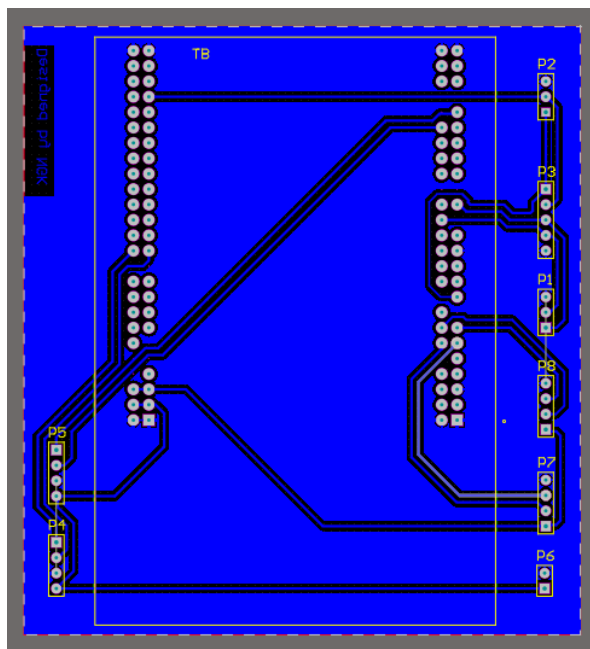
Bảng 2.1. Chức năng và các chân sử dụng trên vi điều khiển STM32F407VG

2.2. Thiết kế và thi công mạch ra chân

Sử dụng phần mềm Altium ta vẽ được sơ đồ nguyên lý mạch và PCB ra chân:



Hình 2.2. Sơ đồ nguyên lý mạch ra chân



Hình 2.3. PCB của mạch ra chân

2.3. Lập trình vi điều khiển

2.3.1. Lập trình giao tiếp GUI máy tính

Ta quy định các frame truyền đi từ vi điều khiển như sau:



Hình 2.4. Frame truyền đi từ vi điều khiển

Frame truyền có chiều dài 18 byte, cụ thể là:

- 7 byte Start: “\$\$DLCN,” dùng để nhận diện frame
- 4 byte Speed: theo kiểu float
- 4 byte Position: theo kiểu float
- 1 byte CheckSum: dùng để kiểm tra lỗi
- 2 byte Stop: “\r\n”

Ta quy định các frame nhận vào vi điều khiển như sau:



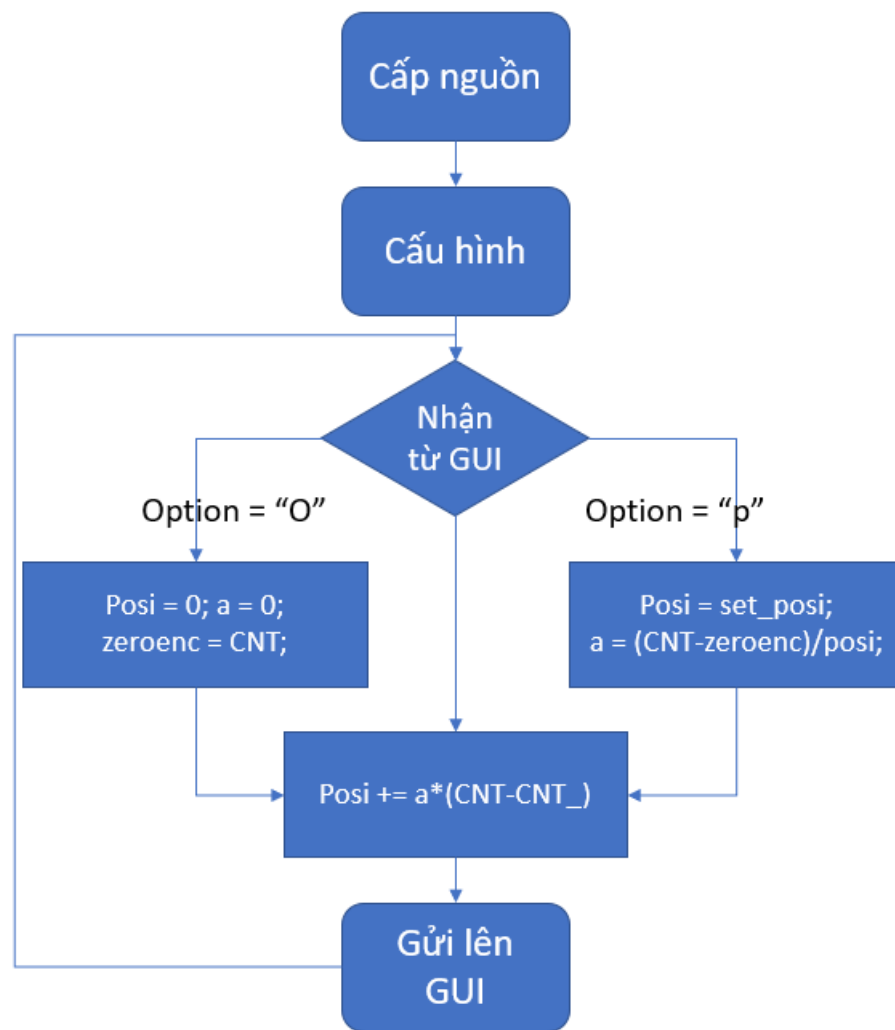
Hình 2.5. Frame nhận vào vi điều khiển

Frame nhận có chiều dài 14 byte, cụ thể là:

- 6 byte Start: “\$Set_,” dùng để nhận diện frame
- 1 byte Option: “O” hoặc “p” để vi điều khiển biết là đặt zero hay đặt góc
- 4 byte Position: theo kiểu float (chỉ dùng khi đặt góc)
- 1 byte CheckSum: dùng để kiểm tra lỗi
- 2 byte Stop: “\r\n”

2.3.2. Lưu đồ lập trình

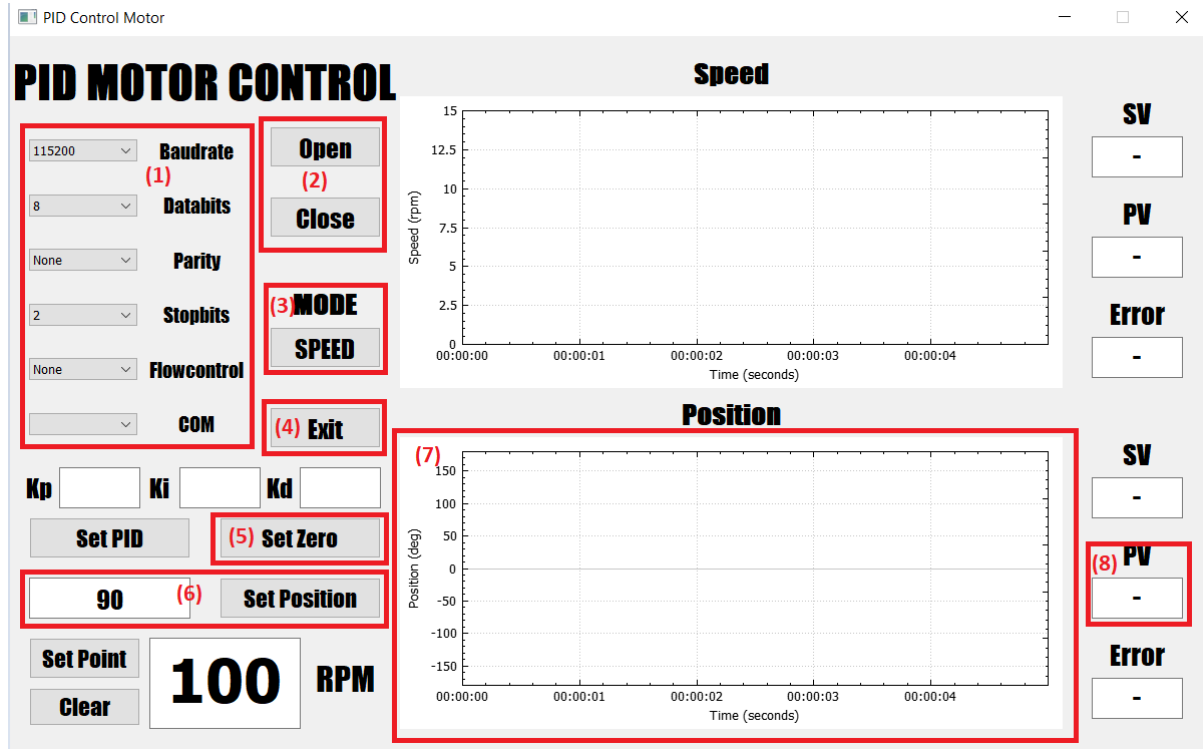
Ta lập trình vi điều khiển theo lưu đồ giải thuật sau



Hình 2.6. Lưu đồ giải thuật lập trình vi điều khiển

2.4. Lập trình giao diện

GUI được viết bằng ngôn ngữ C++ sử dụng framework Qt – một framework mạnh mẽ, hỗ trợ nhiều chức năng khác nhau và có một cộng đồng lớn. Chức năng của giao diện được minh họa bằng hình bên dưới:

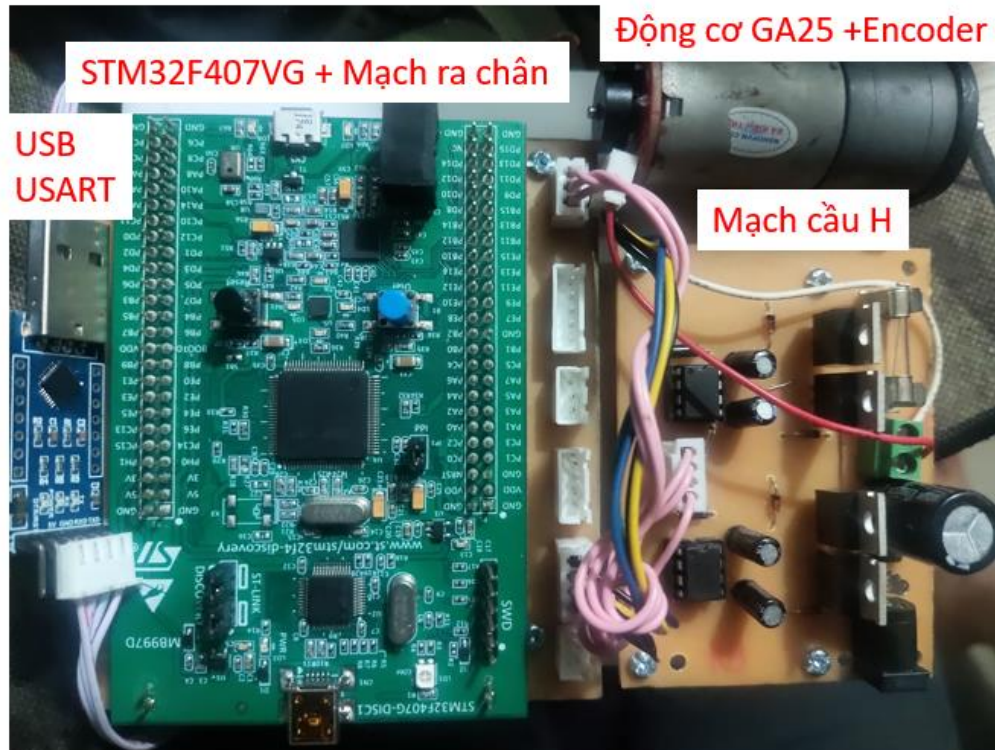


Hình 2.7. Giao diện trên PC

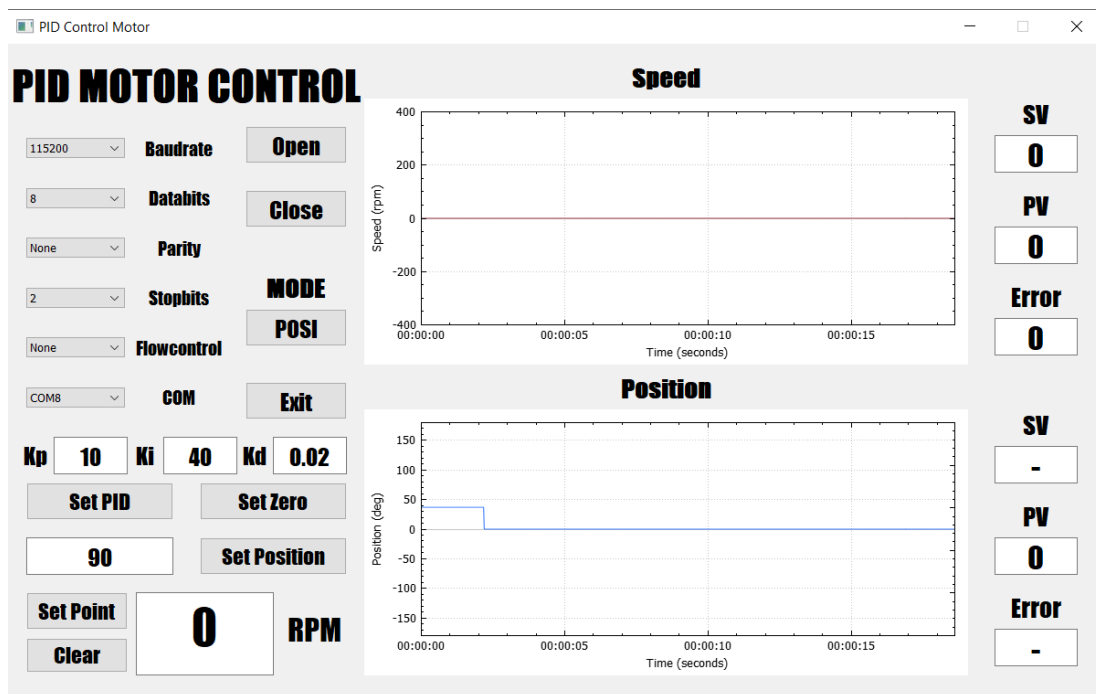
- (1): Cấu cho cổng USART
- (2): Nút Open để mở và kết nối USART, nút Close dùng để ngắt kết nối
- (3): Nút chọn mode, ở đây ta chọn sang mode Position
- (4): Nút Exit để đóng ứng dụng
- (5): Nút Set Zero dùng để gửi lệnh set vị trí ban đầu
- (6): Ô nhập vị trí góc hiện tại và nút Set Position để gửi cho vi điều khiển tính toán a.
- (7): Đồ thị Position (độ) theo thời gian t
- (8): Ô hiển thị vị trí hiện tại cập nhật liên tục

3. KẾT QUẢ

Phần cứng sau khi thiết kế và thi công

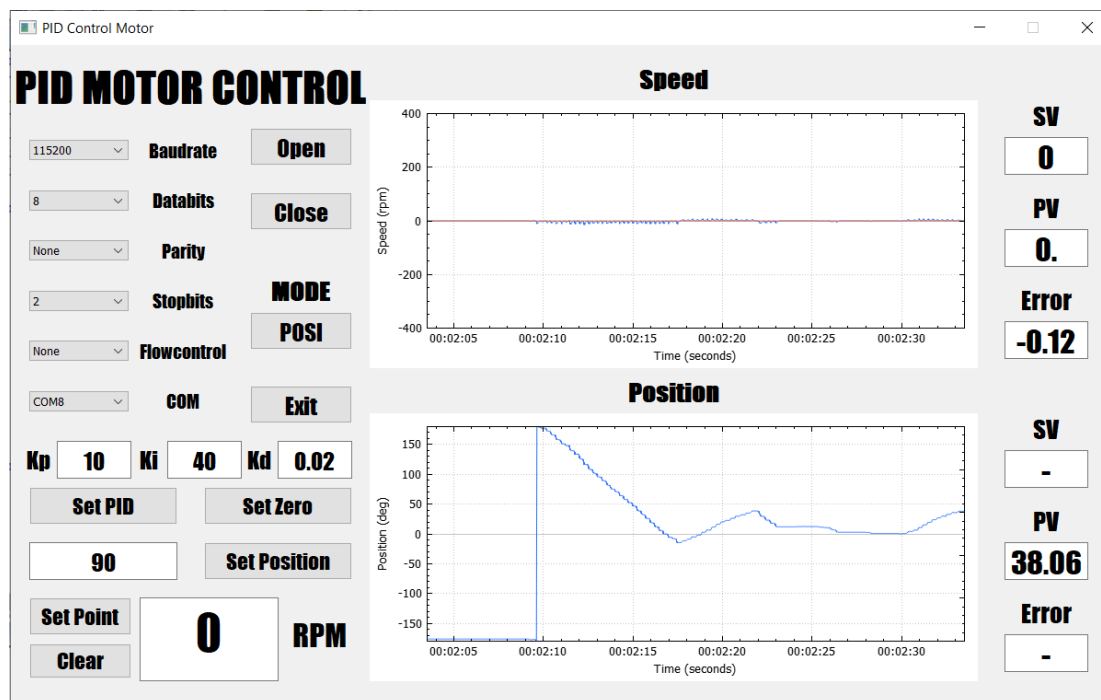


Hình 3.1. Sản phẩm hoàn thiện



Hình 3.2. Giao diện sau khi nhất Set Zero

Sau đó ta quay một góc trên trục động cơ đến vị trí mà ta biết chính xác, nhập vào ô và bấm Set Position.



Hình 3.2. Giao diện sau khi nhấn Set Position và quay

Bảng giá trị các lần chạy:

Lần	Góc đặt	Hệ số a	Góc kiểm tra	Góc tính được	Sai số
1	90	0.396	180	-173.2	6.8
2	90	0.381	180	-177.7	2.3
3	90	0.385	180	178	2.0
4	90	0.402	180	173	7
5	90	0.384	180	-178.2	1.8

Bảng 2. Giá trị các lần chạy