



TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP.HCM
KHOA ĐIỆN TỬ - VIỄN THÔNG

PHÒNG THÍ NGHIỆM DESLAB



DỰ ÁN KẾT THÚC TRAINIG DESLAB 2021

TRẠM HÀN STM32
SỬ DỤNG BỘ ĐIỀU KHIỂN PID

Mã Khải Minh

TP.HCM – 4/2021

MỤC LỤC

1. MỞ ĐẦU.....	3
2. LÝ THUYẾT.....	4
2.1 Tay hàn và trạm hàn.....	4
2.2 STM32	6
2.2.1 Timer.....	7
2.2.2 PWM.....	8
2.2.3 I ² C	9
2.3 Bộ điều khiển PID	10
3. THIẾT KẾ VÀ QUY TRÌNH HOẠT ĐỘNG	13
3.1 Phần cứng.....	13
3.2 Phần mềm.....	17
4. KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN.....	20
5. THAM KHẢO.....	21

MỤC LỤC HÌNH

Hình 1 - Cấu tạo tay hàn Hakko 907.....	4
Hình 2 - Đầu kết nối tay hàn và giá trị trở kháng ở nhiệt độ phòng	5
Hình 3 - Đáp ứng nhiệt độ và trở kháng của cảm biến nhiệt trong tay hàn	5
Hình 4 - Thông số và chức năng của STM32F103XX	6
Hình 5 - Mối quan hệ giữa xung PWM và các thanh ghi trong timer của STM32.....	8
Hình 6 - Cấu trúc truyền của I ² C	9
Hình 7 - Sơ đồ khối bộ điều khiển PID.....	10
Hình 8 - Sơ đồ khối của trạm hàn	13
Hình 9 - Sơ đồ mạch khối đọc nhiệt độ của trạm hàn.....	15
Hình 10 - Sơ đồ mạch khối gia nhiệt của trạm hàn.....	15
Hình 11 - Lưu đồ giải thuật của trạm hàn	17
Hình 12 - Cấu hình chân của STM32F103C8T6	18
Hình 13 - Trạm hàn thực tế trên breadboard.....	20

1. MỞ ĐẦU

Nhằm vận dụng các kiến thức từ khóa training Vi điều khiển cho thành viên mới của DESLAB, dự án này nhằm đến việc tạo ra một trạm hàn sử dụng vi điều khiển STM32 và bộ điều khiển Vi tích phân tỉ lệ (PID) để kiểm soát nhiệt độ tay hàn.

Trạm hàn bao gồm một tay hàn phổ thông, tích hợp sẵn đầu gia nhiệt và cặp nhiệt điện, được điều khiển qua MOSFET bằng xung PWM. Vi điều khiển STM32 có nhiệm vụ đọc nhiệt độ từ tay hàn và điều chỉnh xung PWM để tay hàn gia nhiệt đến nhiệt độ người dùng thiết lập, sử dụng bộ điều khiển PID.

2. LÝ THUYẾT

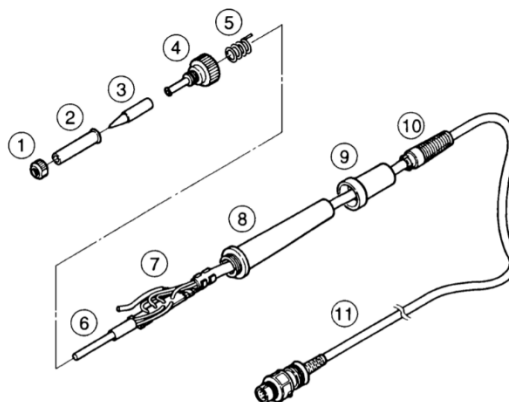
2.1 Tay hàn và trạm hàn

Trên thị trường hiện nay có rất nhiều loại tay hàn. Thông dụng và dễ dàng sử dụng nhất là loại sử dụng điện xoay chiều gia dụng. Loại tay hàn này rất phổ biến nhưng lại có nhược điểm là hiệu suất không được cao, không thể tùy chỉnh được nhiệt độ tay hàn để hàn các linh kiện điện tử nhạy cảm với nhiệt độ khi hàn. Một giải pháp cho vấn đề này là sử dụng trạm hàn bao gồm tay hàn và một bộ điều khiển nhiệt độ cho tay hàn.

Bên trong tay hàn của trạm hàn thường là một thành phần gia nhiệt làm bằng gốm. Khi được đặt vào một điện áp nhất định thì thành phần này sẽ nóng lên và gia nhiệt cho tay hàn. Các thành phần gia nhiệt hiện tại thường được tích hợp thêm một cảm biến nhiệt là cặp nhiệt điện. Cảm biến nhiệt này sẽ thay đổi giá trị điện trở của nó khi nhiệt độ được thay đổi, nhờ đó đọc được nhiệt độ từ tay hàn. Bộ điều khiển của trạm hàn có nhiệm vụ đọc nhiệt độ hiện tại của tay hàn, làm tay hàn nóng lên đến nhiệt độ mà người dùng cài đặt và duy trì nhiệt độ ở mức đó. Các loại trạm hàn trên thị trường đa số sử dụng mạch tích hợp số để thực hiện công việc này.

907,908

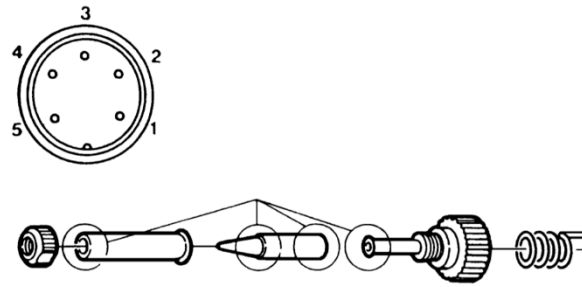
Item No.	Part No.	Part Name	Description	For
1	B1784	Nut		907
	B1794	Nut		908
2	B1786	Tip Enclosure		907
	B1787	Tip Enclosure		908
3		Soldering Tip	See. P. 6	907
		Soldering Tip	See. P. 6	908
4	B2022	Nipple		907
	B2033	Nipple		908
5	B2032	Grounding Spring		907,908
6	A1321	Heating Element	Old part No.900M-H.900L-H	907,908
7	B2028	Terminal Board	w/Cord Stopper	907,908
8	B2023	Handle	w/Handle Cover	907
	B2024	Handle	w/Handle Cover, E.S.D.	907
	B2025	Handle	w/Handle Cover	908
	B2026	Handle	w/Handle Cover, E.S.D.	908
9	B2027	Handle Cover		907,908
10	B2031	Cord Bushing		907,908
11	B2029	Cord Asse'y		907,908
	B2030	Cord Asse'y	E.S.D.	907,908



Hình 1 - Cấu tạo tay hàn Hakko 907

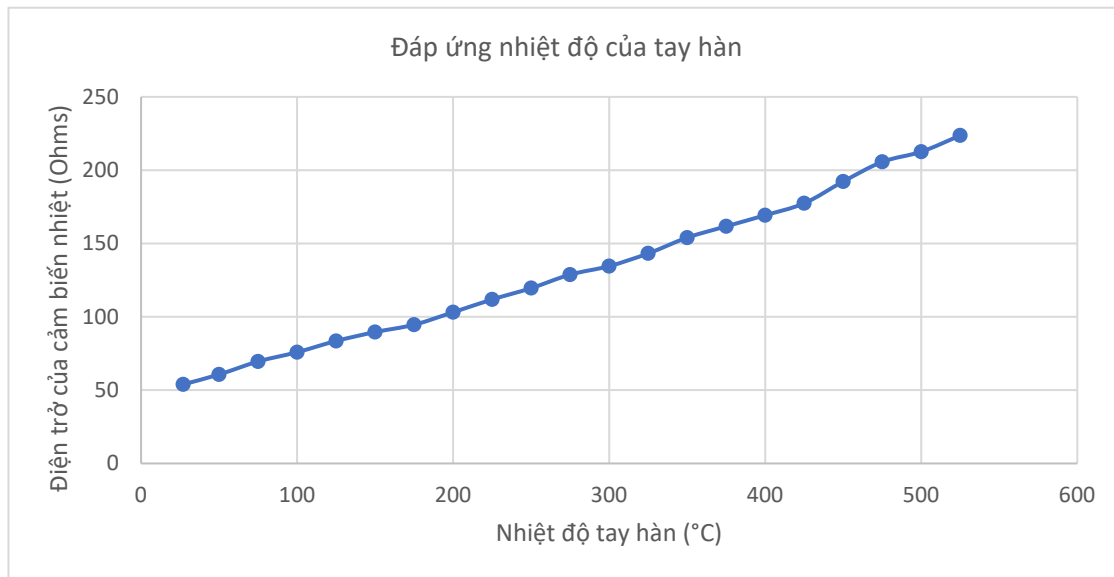
Một loại tay hàn dùng cho trạm hàn khá phổ biến trên thị trường là loại tay hàn sử dụng thành phần gia nhiệt A1321, được sử dụng nhiều trong các trạm hàn của hãng Hakko từ những năm 2000. Đặc tính kỹ thuật của thành phần gia nhiệt A1321 có trong tay hàn Hakko 907 ở nhiệt độ phòng được mô tả như sau:

a	Between pins 4&5 (Heating Element)	2.5 - 3.5Ω(Normal)
b	Between pins 1&2 (Sensor)	43 - 58Ω(Normal)
c	Between pin 3&Tip	Under 2Ω



Hình 2 - Đầu kết nối tay hàn và giá trị trở kháng ở nhiệt độ phòng

Dưới đây là hình 3 diễn tả mối quan hệ đáp ứng nhiệt độ của tay hàn với giá trị điện trở đọc được từ cảm biến nhiệt bên trong tay hàn.



Hình 3 - Đáp ứng nhiệt độ và trở kháng của cảm biến nhiệt trong tay hàn

Có thể thấy mối quan hệ của nhiệt độ tay hàn và giá trị của cảm biến nhiệt là tuyến tính và dễ dàng tính toán được nhiệt độ tay hàn khi đọc giá trị từ cảm biến nhiệt và ngược lại. Sử dụng kỹ thuật hồi quy tuyến tính, mối quan hệ của nhiệt độ tay hàn và giá trị cảm biến nhiệt được biểu diễn như sau:

$$T(^{\circ}C) = 2.96857 \times r(Ohm) - 115.83826$$

Phương trình trên thể hiện cách chuyển đổi từ giá trị cảm biến tức thời mà trạm hàn đọc được sang nhiệt độ tức thời của tay hàn, thuận tiện cho việc thiết kế trạm hàn sử dụng vi điều khiển.

2.2 STM32

STM32 là một dòng vi điều khiển 32-bit dựa trên lõi ARM Cortex-M 32-bit RISC được phát triển bởi STMicroelectronics. STM32 có hiệu năng cao, có khả năng xử lý thời gian thực, xử lý tín hiệu số, có điện năng tiêu thụ thấp và khả năng kết nối với nhiều ngoại vi khác nhau.

STM32F103C8T6 là vi điều khiển thuộc dòng F1 với lõi ARM Cortex-M3, có xung nhịp hoạt động (clock) tối đa là 72MHz. STM32F103C8T6 được đóng gói kiểu LQFP48, có giá thành rẻ so với các loại vi điều khiển có chức năng tương tự. Sau đây là bảng mô tả đặc tính kỹ thuật của dòng STM32F103, bao gồm cả F103C8T6.

Peripheral		STM32F103Tx		STM32F103Cx		STM32F103Rx		STM32F103Vx	
Flash - Kbytes		64	128	64	128	64	128	64	128
SRAM - Kbytes		20		20		20		20	
Timers	General-purpose	3		3		3		3	
	Advanced-control	1		1		1		1	
Communication	SPI	1		2		2		2	
	I ² C	1		2		2		2	
	USART	2		3		3		3	
	USB	1		1		1		1	
	CAN	1		1		1		1	
	GPIOs		26		37		51		80
12-bit synchronized ADC		2		2		2		2	
Number of channels		10 channels		10 channels		16 channels ⁽¹⁾		16 channels	
CPU frequency		72 MHz							
Operating voltage		2.0 to 3.6 V							
Operating temperatures		Ambient temperatures: -40 to +85 °C / -40 to +105 °C Junction temperature: -40 to + 125 °C							
Packages		VFQFPN36		LQFP48, UFQFPN48		LQFP64, TFBGA64		LQFP100, LFBGA100, UFBGA100	

Hình 4 - Thông số và chức năng của STM32F103XX

Bên trong hệ thống trạm hàn sẽ sử dụng vi điều khiển STM32F103C8T6 được tích hợp trên kit phát triển Blue Pill, một kit phát triển phổ biến và cũng có giá thành rẻ. Cùng với kit phát triển là mạch nạp và debug ST-Link V2 Mini có thể mua trên thị trường một cách dễ dàng.

Về phần mềm, việc phát triển và lập trình đa số được thực hiện trên TrueSTUDIO for STM32. Đây một công cụ phát triển hệ thống nhúng của Atollic dành riêng cho STM32, với nhiều tính năng trải dài từ mức độ cơ bản đến chuyên nghiệp, bao gồm cả thư viện HAL, giúp người lập trình phát triển phần mềm một cách dễ dàng.

Bên cạnh đó là STM32CubeMX, một công cụ với giao diện đồ hoạ, giúp người lập trình cấu hình cho tất cả dòng vi điều khiển STM32 và khởi tạo mã nguồn với các cấu hình đó. Công cụ này giúp người lập trình tiết kiệm thời gian trong việc cấu hình cho vi điều khiển trước khi bắt đầu phát triển phần mềm.

2.2.1 Timer

Timer/counter là ngoại vi cơ bản của tất cả các vi điều khiển với chức năng chủ yếu là để đếm thời gian, xung nhịp. Timer/counter được sử dụng nhiều trong các hệ thống có công việc cần kích hoạt dựa trên thời gian hoặc tính toán liên quan đến thời gian. Độ chính xác của timer có liên quan đến độ chính xác của xung clock ngõ vào (do thạch anh cung cấp). Đối với STM32, xung clock được cấp từ hai nguồn: nguồn ngoài (External source) và nguồn nội (Internal source). STM32F103C8T6 có tất cả 4 bộ Timer từ TIM1 đến TIM4, trong đó TIM1 là “Advanced-control timers” và TIM2 đến TIM4 là “General-purpose timers”.

Mỗi bộ Timer đều có các thông số như sau:

- Prescaler (16 bit): bộ chia trước dùng cho việc từ một xung clock gốc, người lập trình có thể giảm tần số xung clock xuống thông qua prescaler
- Counter Mode: chế độ đếm lên hoặc xuống, mỗi lần xung clock thì thanh ghi của timer tăng lên hoặc giảm xuống một đơn vị.
- Counter Period (AutoReload register 16 bit): Thanh ghi chứa giá trị đếm cao nhất của bộ Timer trước khi được nạp lại giá trị để đếm lại, tương ứng với khả năng và giới hạn đếm mà người lập trình yêu cầu. Khả năng đếm tối đa là $2^{16} = 65536$ đơn vị.
- Mỗi bộ Timer sẽ có 4 kênh (channel) độc lập với các chức năng như Input Capture, Output Compare và PWM generation.

2.2.2 PWM

Điều chế độ rộng xung (Pulse-Width Modulation) là một kỹ thuật có trên bộ Timer của STM32F103C8T6, được sử dụng nhiều để điều khiển giá trị trung bình của điện áp cung cấp cho các thiết bị điện bằng cách đóng ngắt điện áp trong một chu kỳ nhất định. Các khái niệm liên quan đến PWM bao gồm:

Duty cycle: Tỷ lệ thời gian ngõ ra ở mức cao trong một chu kỳ xung PWM. Tỷ lệ này càng cao thì điện áp trung bình càng cao, đặc trưng cho độ mạnh yếu của ngõ ra càng cao.

$$Duty\ Cycle = \frac{t_{on}}{t_{on} + t_{off}} = \frac{t_{on}}{T}$$

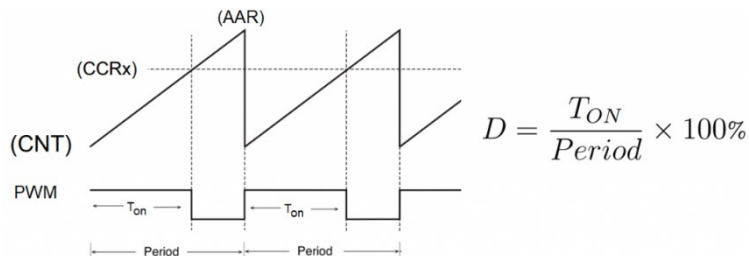
Chu kỳ: Khoảng thời gian lặp lại việc bật tắt của xung PWM

$$T = t_{on} + t_{off}$$

Tần số: Số lần lặp lại việc bật tắt của xung PWM trong 1 giây. Việc chọn tần số liên quan đến đặc trưng phần cứng mà xung PWM áp vào.

$$f = \frac{1}{T}$$

Trong bộ Timer của STM32 có một tính năng là PWM Generator. Tính năng này cho phép tạo ra xung PWM với tần số được xác định bởi tần số clock của Timer sau khi được chia bởi Prescaler và Counter Period (thanh ghi ARR), Duty cycle được xác định bởi giá trị thanh ghi CCR, có khoảng giá trị từ 0 đến giá trị của thanh ghi ARR.



Hình 5 - Mối quan hệ giữa xung PWM và các thanh ghi trong timer của STM32

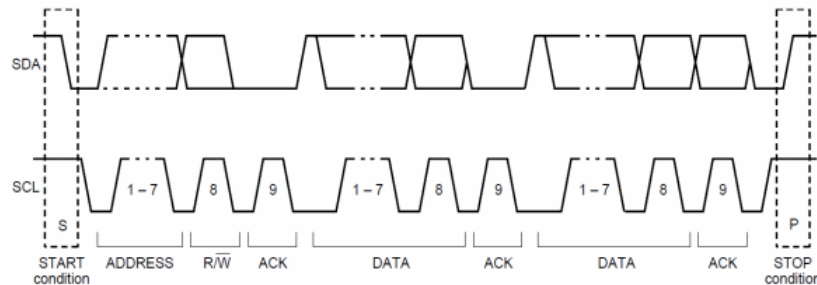
2.2.3 I²C

I²C (Inter – Integrated Circuit) là một giao thức giao tiếp nối tiếp đồng bộ được phát triển bởi Philips Semiconductors. I²C thường được dùng để giao tiếp giữa vi điều khiển với các ngoại vi, cảm biến, thiết bị hiển thị, ADC, DAC, EEPROM...

Giao thức I²C truyền dữ liệu giữa một bộ xử lý trung tâm với nhiều thiết bị thông qua chỉ hai đường truyền tín hiệu Serial Clock Line (SCL), Serial Data Line (SDA). Các bit dữ liệu được truyền từng bit một theo các khoảng thời gian đều đặn được thiết lập bởi xung clock tham chiếu. Hệ thống địa chỉ 7 bit được sử dụng để xác định một thiết bị cụ thể trên bus I²C.

I²C có nhiều chế độ hoạt động khác nhau. Về tốc độ I²C có các chế độ như chế độ tiêu chuẩn (Standard Mode), chế độ nhanh (Fast Mode), chế độ cao tốc (High – Speed Mode). Về mối quan hệ giữa các thiết bị, các chế độ của I²C bao gồm một chủ một tớ (One Master – One Slave), một chủ nhiều tớ (One Master – Multi Slave), nhiều chủ nhiều tớ (Multi Master – Multi Slave).

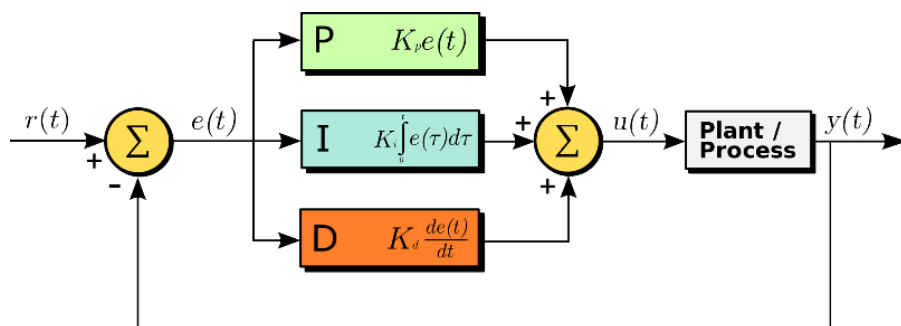
Quy trình truyền dữ liệu của I²C giữa một thiết bị truyền và một thiết bị nhận được thực hiện như sau.



Hình 6 - Cấu trúc truyền của I²C

2.3 Bộ điều khiển PID

Bộ điều khiển PID (Proportional Integral Derivative) là một cơ chế điều khiển tự động có vòng phản hồi, được sử dụng và chấp nhận rộng rãi trong công nghiệp nhờ tính chất đơn giản nhưng lại có hiệu suất mạnh mẽ và ổn định. Bộ điều khiển PID sẽ tính toán giá trị sai số là hiệu số giữa giá trị đặt mong muốn (Set Point) và giá trị đo thông số biến đổi (Process Variable). Bộ điều khiển sẽ thực hiện giảm tối đa sai số bằng cách điều chỉnh giá trị điều khiển của hệ thống cần điều khiển.



Hình 7 - Sơ đồ khối bộ điều khiển PID

Giải thuật của bộ điều khiển PID bao gồm 3 phần: tỉ lệ (P), tích phân (I) và đạo hàm (D). Sơ đồ khối trên cho thấy quá trình tính toán của bộ điều khiển và các phần trong bộ điều khiển PID. Bộ điều khiển sẽ tính giá trị sai số $e(t)$ dựa trên khoảng chênh lệch giữa giá trị mong muốn $r(t)$ và giá trị đo được $y(t)$, sau đó áp dụng các biến đổi dựa trên 3 phần của giải thuật PID và cho ra giá trị điều khiển tổng $u(t)$. Kết quả cho ra là giá trị chính xác và tối ưu nhất để điều khiển hệ thống đạt được giá trị mong muốn theo thời gian.

Ở trong sơ đồ khối trên:

- Phần tính tỉ lệ (Proportional) cho ra kết quả là tỉ lệ của sai số $e(t)$ với hệ số độ lợi K_p . Khi sai số $e(t)$ có giá trị lớn và dương, giá trị đo được đang dưới mức giá trị cần đạt, kết quả cho ra của phần tỉ lệ cũng sẽ có giá trị lớn và dương làm cho hệ thống hoạt động theo hướng giảm sai số $e(t)$. Khi sai số $e(t)$ là 0, phần tính tỉ lệ sẽ còn điều khiển hệ thống. Nếu chỉ sử dụng phần tính tỉ lệ để điều khiển hệ thống thì sai số $e(t)$ của hệ thống sẽ giảm nhưng sẽ không đạt được đến giá trị mong muốn do hệ thống sẽ còn số dư sai ổn định (steady state error).

- Phần tính tích phân (Integral) xem xét đến các giá trị sai số $e(t)$ tức thời tại thời điểm đo từ quá khứ cho đến hiện tại và thực hiện tích phân theo thời gian để cho ra kết quả. Sau đó biên độ/độ lớn của kết quả sẽ được hiệu chỉnh bởi hệ số độ lợi K_i . Phần tính tích phân sẽ gia tốc quá trình hệ thống giảm sai số về 0 và loại trừ số dư sai ổn định. Tuy nhiên, vì phần tính tích phân này là tổng hợp các sai số trong quá khứ nên có thể dẫn đến việc hệ thống bị vọt lố và động trước khi ổn định tại giá trị mong muốn.
- Phần tính đạo hàm (Derivative) xem xét biên độ thay đổi của sai số $e(t)$ theo thời gian để kiểm soát và làm chậm tốc độ thay đổi của đầu ra hệ thống, giảm biên độ vọt lố do phần tính tỉ lệ và tính phân ra gây ra. Tuy nhiên, tín hiệu nhiễu từ giá trị đo trong hệ thống $y(t)$ sẽ gây ra nhiễu ở sai số $e(t)$. Các giá trị nhiễu sẽ được khuếch đại lên nhiều lần, có thể gây ra sự mất ổn định trong hệ thống. Phần tính đạo hàm cũng sẽ được hiệu chỉnh bởi hệ số độ lợi riêng, K_d .

Ở dạng toán học, bộ điều khiển PID được biểu diễn như sau:

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d e(t) \frac{d}{dt}$$

Mặc dù bộ điều khiển PID có 3 phần tính tỉ lệ, tích phân và đạo hàm, tùy theo nhu cầu và đặc trưng của hệ thống đang điều khiển mà bộ điều khiển đôi khi chỉ cần 1 hoặc 2 phần tính toán. Điều này đồng nghĩa với việc hiệu chỉnh hệ số độ lợi của các phần không mong muốn có mặt về 0. Các bộ điều khiển được gọi tắt bằng cách bỏ đi phần tính toán không mong muốn như bộ điều khiển P (Proportional – Tỉ lệ), bộ điều khiển PI (Proportional Integral – Tỉ lệ và Tích phân) và bộ điều khiển PD (Proportional Derivative – Tỉ lệ và Đạo hàm).

Khi áp dụng vào hệ thống trạm hàn và lập trình cho vi điều khiển, bộ điều khiển PID được thay đổi một số chi tiết nhỏ như sau:

- Giá trị điều khiển ở đầu ra bộ điều khiển $u(t)$ sẽ được giới hạn trong khoảng đáp ứng của hệ thống trên thực tế, điều này tránh việc bộ điều khiển cho ra giá trị cao hơn khả năng đáp ứng đầu vào của hệ thống.
- Kết quả của phần tính tích phân cũng sẽ được giới hạn ở một mức độ để tránh tình trạng giá trị tích phân cộng dồn quá lớn, gây vọt lố trong hệ thống.

- Cuối cùng việc thay đổi dạng biểu diễn toán học của bộ điều khiển PID, cụ thể ở hệ số độ lợi phân tích tích phân. Điều này khiến cho hệ thống trở nên ổn định và linh hoạt hơn khi cần thay đổi hệ số độ lợi của cả hệ thống.

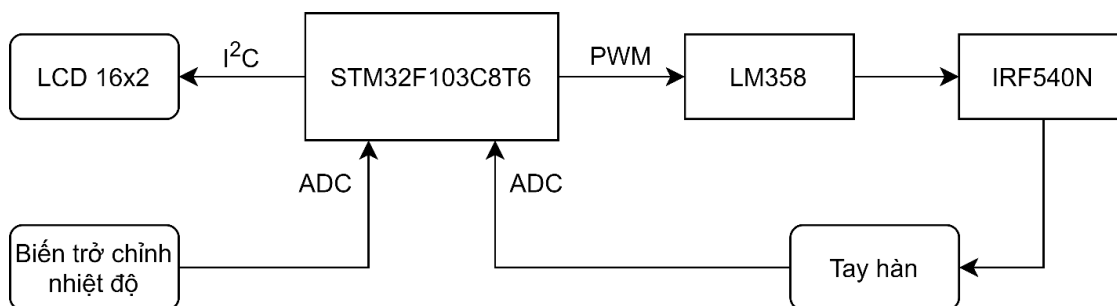
$$u(t) = K_p e(t) + \int K_i e(t) dt + K_d e(t) \frac{d}{dt}$$

3. THIẾT KẾ VÀ QUY TRÌNH HOẠT ĐỘNG

3.1 Phần cứng

Các trạm hàn có trên thị trường thường có các thành phần giống nhau: tay hàn, mạch điều khiển gia nhiệt cho tay hàn và nguồn. Trong các mạch điều khiển thì tay hàn được bật tắt để tăng giảm và duy trì nhiệt thông qua MOSFET được điều khiển bởi các mạch tích hợp số. Ở trong hệ thống trạm hàn của bài báo cáo này cũng sẽ có cấu tạo gần giống với các mạch trạm hàn trên. Trạm hàn sẽ sử dụng một vi điều khiển STM32F103C8T6, có nhiệm vụ đọc giá trị nhiệt độ từ tay hàn sau đó xử lý và điều khiển bật tắt một N – MOSFET bằng xung PWM để điều khiển điện áp cấp cho tay hàn, làm cho tay hàn nóng lên đến nhiệt độ yêu cầu và duy trì ở mức đó. Người dùng có thể chỉnh nhiệt độ của tay hàn thông qua một biến trở. Các thông tin như nhiệt độ yêu cầu và nhiệt độ hiện tại sẽ được hiển thị trên màn hình LCD 16x2, cũng được điều khiển bởi vi điều khiển STM32.

Sơ đồ khối của trạm hàn được thể hiện như sau:

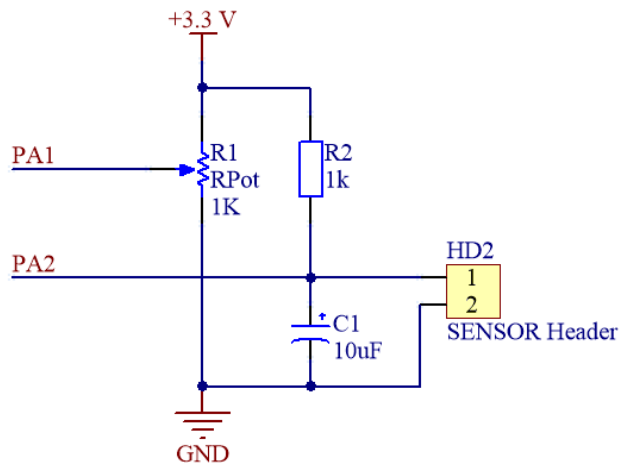


Hình 8 - Sơ đồ khối của trạm hàn

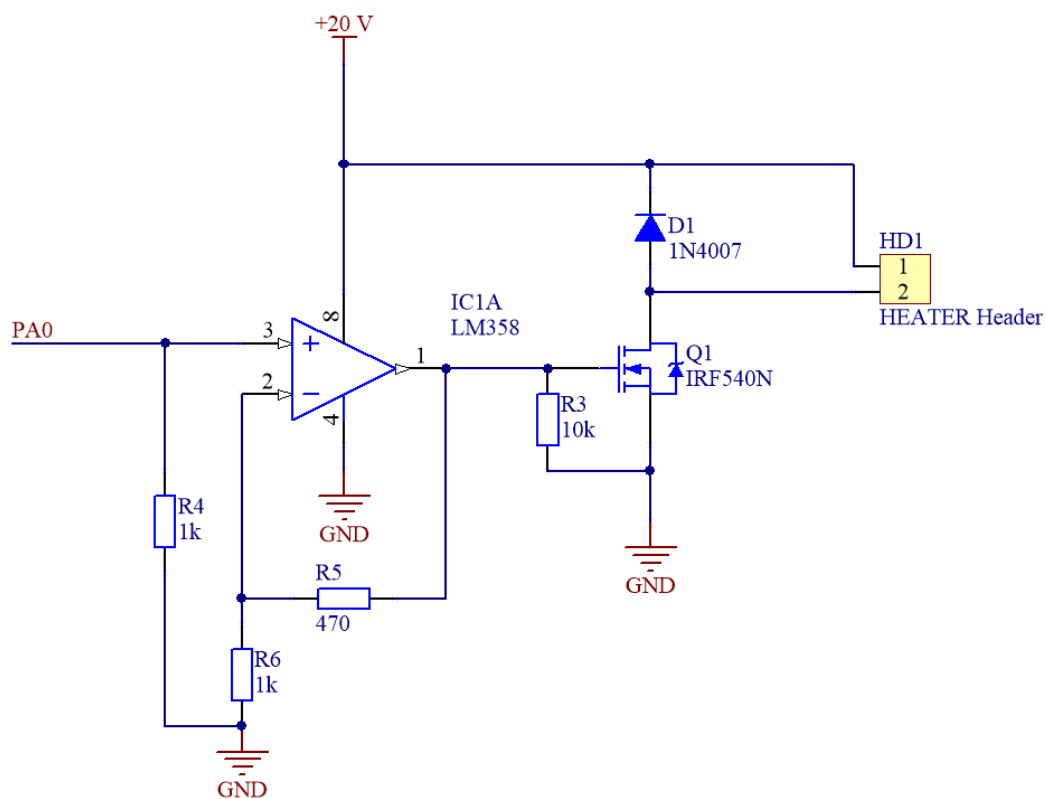
Trạm hàn có cấu hình được liệt kê chi tiết như sau:

- Tay hàn: tay hàn được sử dụng là Hakko 907, bên trong là thành phần gia nhiệt có cảm biến nhiệt A1321. Tay hàn hoạt động với điện áp 24V DC, có công suất là 50W. Tay hàn kết nối với trạm hàn với một đầu kết nối 5 chân gồm 2 chân cho thành phần gia nhiệt, 2 chân cho cảm biến nhiệt và 1 chân đất nối với thân tay hàn dùng để loại bỏ tĩnh điện khi hàn các linh kiện điện tử.

- STM32: STM32F103C8T6 được sử dụng trong trạm hàn dưới dạng một bo mạch phát triển được ra chân sẵn có tên là Blue Pill. Bo mạch có kích thước nhỏ gọn và giá thành rẻ, phù hợp với nhu cầu sử dụng hiệu năng không quá cao. Vi điều khiển STM32 bên trong trạm hàn đảm nhận nhiệm vụ đọc giá trị từ biến trở cài đặt nhiệt độ, cảm biến nhiệt từ tay hàn, thực hiện tính toán và điều khiển xung PWM để làm nóng tay hàn.
- MOSFET: Để bật tắt nguồn điện cấp cho tay hàn để gia nhiệt thì cần đến một transistor hiệu ứng trường (MOSFET). Trong trạm hàn sử dụng MOSFET kênh N IRF540N có điện thế cực máng – cực nguồn tối đa là 100V, dòng tối đa là 33A trở kháng cực máng – cực thu tối đa là 44mΩ.
- OpAmp LM358: do điện thế I/O của STM32 chỉ hoạt động ở mức 3.3V, không đủ đạt đến mức điện áp tại cực cổng của MOSFET để MOSFET bắt đầu dẫn, LM358 được sử dụng để khuếch đại mức điện áp của xung PWM để vi điều khiển có thể điều khiển được việc bật tắt của MOSFET. Opamp được ở cấu hình khuếch đại không đảo.
- Nguồn: Nguồn cấp cho trạm hàn được khuyến khích chung yêu cầu với tay hàn (24V 50W). Tuy nhiên có thể linh động và sử dụng các điện áp từ 15V đến 20V để cấp cho trạm hàn.
- Linh kiện khác: Các linh kiện khác bao gồm breadboard, dây cắm, điện trở, tụ điện dùng để kết nối các linh kiện với nhau, tạo thành mạch điện.



Hình 9 - Sơ đồ mạch khối đọc nhiệt độ của trạm hàn

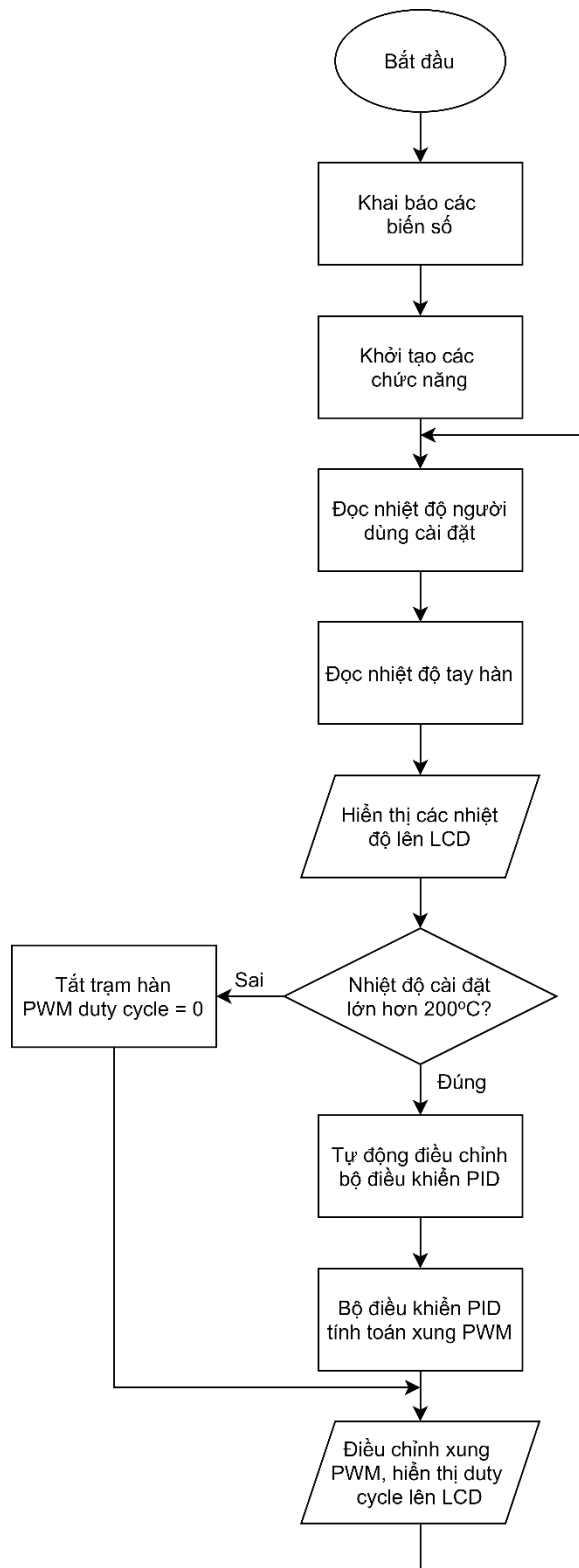


Hình 10 - Sơ đồ mạch khối gia nhiệt của trạm hàn

Hình 9 mô tả sơ đồ mạch điện khối đọc cảm biến nhiệt từ tay hàn và biến trở để điều chỉnh nhiệt độ tay hàn theo ý người dùng. VDD 3.3V được cấp từ kit phát triển Blue Pill, tương đương với việc V_{ref} của ADC cũng sẽ là 3.3V. Biến trở sử dụng có giá trị 1k Ohm được nối với chân PA1 của vi điều khiển STM32. Cảm biến nhiệt từ tay hàn là một cặp nhiệt điện nên có thể dùng nhiều phương pháp khác nhau để lấy giá trị trở kháng. Cách được sử dụng trong trạm hàn này là thiết lập cầu phân áp. Với R2 là 1k Ohm và khoảng giá trị trở kháng của cảm biến nhiệt từ 50 Ohm đến 200 Ohm, điện áp trả về để ADC đọc được là từ khoảng 0.15V đến 0.55V qua chân PA2 của vi điều khiển. Tụ C1 có nhiệm vụ là lọc nhiễu các tín hiệu để quá trình đọc nhiệt độ ổn định hơn.

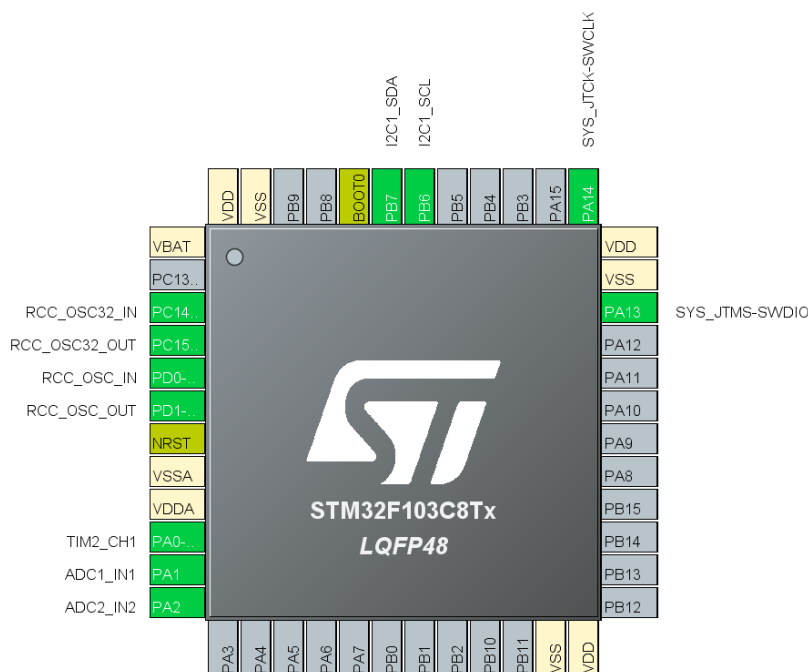
Hình 10 mô tả sơ đồ mạch điện khối điều khiển PWM và gia nhiệt tay hàn bằng MOSFET kênh N IRF540N. Nguồn được cấp cho khối này là +20V cho OpAmp LM358 và cũng là VDS của MOSFET. Diode D1 được dùng bảo vệ ngược dòng từ tay hàn và MOSFET. Chân PA0 là chân tín hiệu điều khiển PWM từ vi điều khiển có mức điện áp là 3.3V (mức cao) và 0V (mức thấp), được khuếch đại lên 3 lần khi qua một OpAmp LM358 để tăng mức áp lên ~10V (mức cao) và 0V (mức thấp), khiến cho MOSFET dẫn và không dẫn hoàn toàn theo xung PWM từ vi điều khiển. Điện trở R3 10k Ohm kéo VGS xuống mức thấp được khuyến khích khi dùng MOSFET kênh N. Điện trở R4 1k Ohm kéo xuống đất làm cho tín hiệu vào OpAmp trở nên ổn định hơn.

3.2 Phần mềm



Hình 11 - Lưu đồ giải thuật của trạm hàn

Cấu hình chân của STM32F103C8T6 trên STM32CubeMX như sau:



Hình 12 - Cấu hình chân của STM32F103C8T6

Trong đó một số chân kết nối với các vai trò cụ thể sau:

- PA0 – TIM2_CH1: Chân tạo xung PWM điều khiển mở tay hàn. Chu kì PWM cố định tại 500 Hz.
- PA1 – ADC1_IN1: Cổng đọc ADC từ biến trở, đọc nhiệt độ tay hàn người dùng yêu cầu.
- PA2 – ADC2_IN2: Cổng đọc ADC từ một cầu phân áp gồm cặp nhiệt điện từ tay hàn, đọc nhiệt độ hiện tại của tay hàn.
- PB6, PB7 – I2C SCL/SDA: chân clock và data dùng để giao tiếp với LCD 16x02.

Khi trạm hàn khởi động, việc đầu tiên vi điều khiển thực hiện là khai báo các biến số được sử dụng trong chương trình, khởi tạo các hàm chức năng ADC, I2C, TIMER, PWM. Quá trình đọc nhiệt độ của người dùng cài đặt thông qua biến trở được thực hiện và trả về kết quả khi hàm `int get_user_temp()` được gọi. Nhiệt độ từ tay hàn hiện tại được trả về khi hàm `int get_heater_temp()` được gọi. Sau đó, thông tin của 2 nhiệt độ trên được hiển thị lên màn hình LCD 16x2, với “TG” là nhiệt độ mà người dùng cài đặt (target) và nhiệt độ hiện tại của tay hàn là “CR” (current).

Tiếp theo, vi điều khiển sẽ quyết định bật tắt trạm hàn dựa trên nhiệt độ cài đặt từ người dùng. Nếu nhiệt độ người dùng cài đặt dưới 200°C , duty cycle của xung PWM sẽ được gán giá trị 0%, tương đương với việc MOSFET sẽ không dẫn, tay hàn sẽ không được gia nhiệt. Nếu người dùng cài đặt nhiệt độ trên 200°C , dựa trên nhiệt độ cài đặt và nhiệt độ hiện tại, bộ điều khiển PID sẽ tự điều chỉnh lại các cài đặt cho phù hợp thông qua hàm `semi_auto_config_PID()`, sau đó thực hiện việc tính toán để cho ra duty cycle mới làm MOSFET dẫn và tay hàn được gia nhiệt. Quá trình này được lặp lại liên tục. Nhiệt độ tối đa của trạm hàn hiện tại là 390°C .

Các hàm trong PID được giải thích cụ thể như sau:

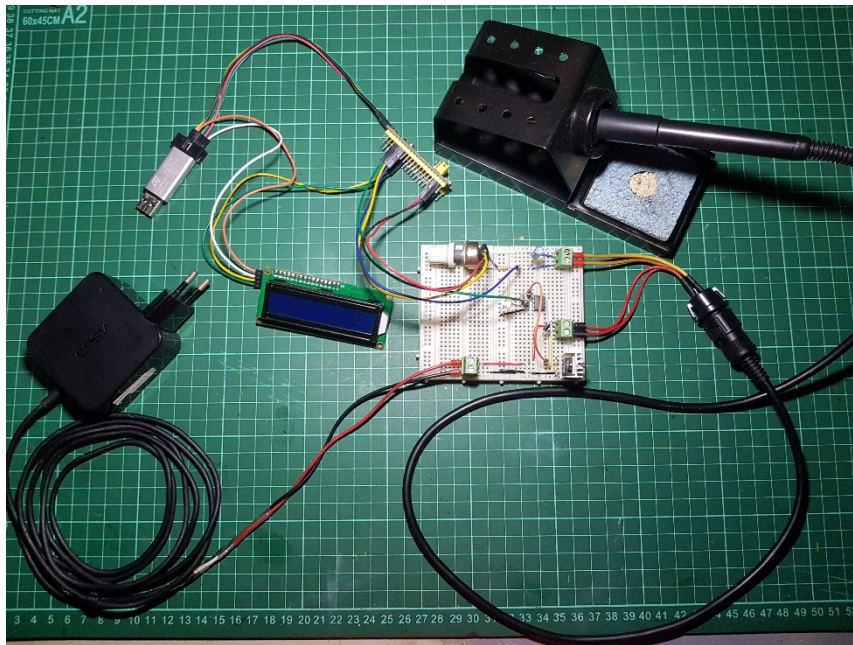
- `void PID_init()` : hàm khởi động PID, đặt tất cả các biến số mà bộ điều khiển sử dụng về 0.
- `void PID_set_gain(float kP_set, float kI_set, float kD_set)` : cài đặt các giá trị độ lợi của từng phần tỉ lệ, tích phân và đạo hàm.
- `void PID_set_clamp(double kI_clamp_MIN_set, double kI_clamp_MAX_set, double output_MIN_set, double output_MAX_set)` : cài đặt các giới hạn của thuật toán và đầu ra của bộ điều khiển, tránh tình trạng vọt lố đáp ứng của hệ thống mà bộ điều khiển PID đang điều khiển.
- `double PID_compute(double set_point, double input, double time_interval_sec)` : thực hiện việc tính toán đầu ra của bộ điều khiển PID.

Duty cycle của xung PWM cũng sẽ được hiển thị lên LCD dưới dạng phần trăm. Sau đó hệ thống sẽ được lặp lại từ quá trình đọc nhiệt độ người dùng cài đặt và đọc nhiệt độ tay hàn hiện tại.

Đáp ứng của bộ điều khiển PID phụ thuộc vào các hệ số độ lợi K_p , K_i và K_d . Đối với K_p , hệ số này sẽ quyết định độ nhạy của hệ thống với sai số $e(t)$, khiến cho hệ thống (ở đây là trạm hàn) phản ứng nhanh với thay đổi nhiệt độ. Giá trị hiện tại của hệ số K_p là 5, tương đương với việc thanh ghi xác định duty cycle xung PWM sẽ tăng 5 đơn vị mỗi 1°C . Đối với K_i , giá trị của hệ số này sẽ giúp hệ thống loại bỏ số dư sai ổn định, giúp tay hàn đạt đến nhiệt độ ổn định. Giá trị hiện tại của K_i là 0.38, là giá trị có được sau khi thử nghiệm hệ thống nhiều lần. Cuối cùng là K_d , giá trị của hệ số này có ý nghĩa là độ hãm dựa trên sự thay đổi của $e(t)$, khiến cho $e(t)$ tăng hoặc giảm một cách ổn định. Giá trị hiện tại của K_d là 0.2, là giá trị có được sau khi thử nghiệm hệ thống nhiều lần.

4. KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN

Trạm hàn dùng STM32 trong dự án này hoạt động ổn định với chức năng cơ bản là chỉnh nhiệt độ tay hàn bằng biến trở do người dùng cài đặt. Nhiệt độ từ nhiệt độ phòng gia nhiệt lên đến nhiệt độ cài đặt nhanh chóng, có vọt lố 1-2⁰C. Trong quá trình sử dụng khi chỉnh nhiệt độ lên hoặc xuống, trạm hàn đáp ứng tăng và giảm nhiệt độ nhanh nhưng có vọt lố hoặc thiếu hụt $\pm 2^0\text{C}$. Khi hoạt động, do các linh kiện hiện tại được thử trên breadboard nên có hiện tượng nóng ở đáy breadboard và MOSFET.



Hình 13 - Trạm hàn thực tế trên breadboard

Trong tương lai, dự án này vẫn sẽ được tiếp tục cải thiện. Một số công việc dự kiến là nghiên cứu đáp ứng nhiệt độ khi gia nhiệt của tay hàn, tinh chỉnh tần số PWM, thiết kế PCB cho hệ thống mạch điện, sử dụng được cho nhiều loại tay hàn khác nhau.

5. THAM KHẢO

- [1] Brett Beauregard, **Improving the Beginner's PID – Introduction**, 2011.
Đường dẫn: <http://brettbeauregard.com/blog/>

- [2] International Rectifier, **IRF540NPbF Datasheet**, 2003.

- [3] STMicroelectronics, STM32F103 **Reference manual RM0008**, Rev 20, 2018.

- [4] Tapit.vn, **Giao tiếp STM32F103C8T6 với LCD 16x02 thông qua module I2C**, 2018.
Đường dẫn: <https://tapit.vn/giao-tiep-stm32f103c8t6-voi-lcd-16x2-thong-qua-module-i2c/>