

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA ĐIỆN – ĐIỆN TỬ

ĐỒ ÁN TỐT NGHIỆP
ĐẠI HỌC



NGÀNH : CÔNG NGHỆ KỸ THUẬT ĐIỆN – ĐIỆN TỬ
CHUYÊN NGÀNH: KỸ THUẬT ĐIỆN TỬ

ĐỀ TÀI:

THIẾT KẾ VÀ THI CÔNG HỆ THỐNG QUẢN LÝ MƯỢN TRẢ SÁCH THƯ VIỆN ỨNG DỤNG QUÉT QR CODE

Người hướng dẫn: Ths. Võ Thị Hương

Sinh viên thực hiện: Nguyễn Văn Bình

Msv:1811505120303 Lớp: 18D4

Trần Thế Mỹ

Msv:1811504210429 Lớp: 18D3

Đà Nẵng, 20/05/2022

**ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA ĐIỆN – ĐIỆN TỬ**

**ĐỒ ÁN TỐT NGHIỆP
ĐẠI HỌC**



**NGÀNH: CÔNG NGHỆ KỸ THUẬT ĐIỆN – ĐIỆN TỬ
CHUYÊN NGÀNH: KỸ THUẬT ĐIỆN TỬ**

ĐỀ TÀI:

**THIẾT KẾ VÀ THI CÔNG HỆ THỐNG QUẢN
LÝ MƯỢN TRẢ SÁCH THƯ VIỆN ỨNG DỤNG
QUÉT QR CODE**

Người hướng dẫn: Ths. Võ Thị Hương

Sinh viên thực hiện: Nguyễn Văn Bình

Msv: 1811505120303 Lớp: 18D4

Trần Thế Mỹ

Msv: 1811504210429 Lớp: 18D3

Đà Nẵng, 20/05/2022

TÓM TẮT

Tên đề tài: **THIẾT KẾ VÀ THI CÔNG HỆ THỐNG QUẢN LÝ MƯỢN TRẢ SÁCH THƯ VIỆN ỨNG DỤNG QUÉT QR CODE**

Giảng viên hướng dẫn: **Ths. Võ Thị Hương**

Sinh viên thực hiện: **Nguyễn Văn Bình** Mã SV: **1811505120303** Lớp: **18D4**

Trần Thế Mỹ Mã SV: **1811504210429** Lớp: **18D3**

Từ quá trình thăm quan và quan sát các thư viện hiện nay trên địa bàn thành Phố Đà Nẵng, nhóm em nhận thấy hầu hết tất cả các nhà sách đều sử dụng con người để quản lý và phục vụ người dùng, trăn trở về vấn đề chưa được tối ưu này tụi em đã quyết định tìm hiểu và tiến hành thực hiện đề tài “**THIẾT KẾ VÀ THI CÔNG HỆ THỐNG QUẢN LÝ MƯỢN TRẢ SÁCH THƯ VIỆN ỨNG DỤNG QUÉT QR CODE**”.

Nhóm em đã nghiên cứu và sử dụng ngôn ngữ lập trình C# để lập trình giao diện Winform C# tạo một giao diện trên Window có các tính năng: lấy và trả sách một cách tự động bằng cách tạo QR Code cho từng loại sách qua đó, người dùng chỉ cần quét QR code đã lưu trước đó trong hệ thống thông qua camera. QR code được nhận diện truyền tín hiệu điều khiển xuống mạch STM32F103C8T6 dùng ‘USB TTL CH340’ thông qua mạch điều khiển động cơ bước Tb6560 điều khiển các trục động cơ bước đến các tọa độ đã cài đặt sẵn trên giá sách để lấy và trả sách và luôn hướng về vị trí home sau khi hoàn thành lệnh và xuất quá trình hoạt động lên LCD, bên cạnh đó ta có thể điều khiển bằng các nút nhấn thể hiện vị trí giá để từng loại sách. Và dữ liệu người dùng sẽ được lưu trữ trong từng tài khoản người dùng gồm thời gian lấy và trả sách....

Việc sử dụng STM32 làm vi điều khiển trung tâm cho đề tài này có mục đích làm tăng tính sáng tạo và mới mẻ hơn so với việc sử dụng Arduino để lập trình mà đa số hiện nay sinh viên thường hay sử dụng ngoài ra STM32 không hỗ trợ thư viện sẵn bắt buộc người lập trình cần phải tìm kiếm tài liệu và nỗ lực rèn luyện thì mới có thể đạt được mục tiêu.

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Giảng viên hướng dẫn: Ths. Võ Thị Hương

Sinh viên thực hiện: NGUYỄN VĂN BÌNH

Mã SV: 1811505120303

TRẦN THẾ MỸ

Mã SV: 1811504210429

1. Tên đề tài: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG QUẢN LÝ MƯỢN TRẢ SÁCH THƯ VIỆN ỨNG DỤNG QUÉT QR CODE

2. Các số liệu, tài liệu ban đầu:

- Giáo trình C# - Phạm Văn Việt, Trương Lập Vĩ
- Tìm hiểu về QR code – blog Nguyễn Phùng Hưng
- Lập trình căn bản ARM CORTEX M3 STM32F103C8T6-Nguyễn Ngọc Hà

3. Những nội dung chính:

- Sử dụng giao diện Winform C# build ra app.exe cho window
- Sử dụng STM32 làm vi điều khiển trung tâm
- Sử dụng mạch điều khiển Tb6560 điều khiển động cơ step
- Sử camera để quét QR Code
- Dùng giao diện Winform C# để nhận dạng QR Code, lưu thông tin, gửi tín hiệu điều khiển xuống vi điều khiển

4. Các sản phẩm dự kiến

- Báo cáo tổng kết đồ án tốt nghiệp
- Mô hình thực tiễn hệ thống quản lý mượn trả sách thư viện ứng dụng quét QR Code

5. Ngày giao đồ án: 22/2/2022

6. Ngày nộp đồ án: 29/5/2022

Đà Nẵng, ngày 25 tháng 02 năm 2022

Trưởng Bộ môn

Người hướng dẫn

Ths.Võ Thị Hương

LỜI NÓI ĐẦU

Sau 4 năm học tập và rèn luyện tại trường Đại học sư phạm kỹ thuật Đà Nẵng, đồ án tốt nghiệp này là một dấu ấn quan trọng đánh dấu việc nhóm em đã hoàn thành nhiệm vụ của mình trên ghế giảng đường Đại học. Trong phạm vi đồ án tốt nghiệp của mình.

Em xin chân thành cảm ơn nhà trường ban lãnh đạo khoa Điện –Điện đã trang bị cho nhóm những kiến thức nền tảng và chuyên ngành quan trọng, qua đó giúp nhóm chúng em có được cơ sở lý thuyết vững vàng và tạo điều kiện tốt nhất cho chúng em để thực hiện tốt đề tài.

Trong suốt quá trình thực hiện đề tài nhóm em luôn được sự quan tâm và giúp đỡ từ thầy cô và bạn bè trong khoa Điện – Điện Tử cùng với sự động viên từ cô và bạn bè

Nhóm em xin tỏ lòng kính trọng và biết ơn sâu sắc đến cô Võ Thị Hương là người trực tiếp hướng dẫn đồ án. Cô đã hướng dẫn và chỉ ra những lỗi để đồ án của em được hoàn thiện bên cạnh đó còn củng cố kiến thức đã thiếu sót trong quá trình học tập và cung cấp thêm những kiến thức mới thông qua đề tài này.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

Nguyễn Văn Bình

Trần Thế Mỹ

CAM ĐOAN

Đề tài này là do chính nhóm em thực hiện dưới sự hướng dẫn của Cô Võ Thị Hương dựa vào tài liệu đã có và không sao chép công trình nghiên cứu của bất kỳ ai.

Đồng thời những số liệu được cung cấp từ báo cáo đều là của nhóm em tính toán đây là kết quả nghiên cứu hoàn toàn trung thực, những tài liệu trích dẫn đều đã được ghi rõ nguồn gốc.

Nhóm em xin chịu hoàn toàn trách nhiệm trước nhà trường nếu trường hợp phát hiện ra bất cứ sai phạm hay vấn đề sao chép nào trong đề tài này.

Đã bổ sung, cập nhật theo yêu cầu của Giảng viên phản biện và Hội đồng chấm Đồ án tốt nghiệp họp ngày ngày 17, 18/6/2022

Giảng viên hướng dẫn

Sinh viên thực hiện

{Chữ ký, họ và tên sinh viên}

DANH SÁCH HÌNH VẼ

Hình 1. 1 Các phương tiện lưu trữ của thư viện [1].....	1
Hình 1. 2 Vai trò của thư viện hiện nay [1].....	1
Hình 1. 3 Xu hướng thiết kế thư viện hiện nay [2].....	2
Hình 1. 4 Hình ảnh mô hình thư viện ở Việt Nam [2].....	2
Hình 1. 5 Hình ảnh mô hình thư viện Learning commons [3].....	3
Hình 1.6 Hình ảnh cấu trúc QR Code [5].....	5
Hình 1. 7 Hình ảnh bản nâng cấp của mã vạch truyền thống [6].....	7
Hình 2.1 Hình ảnh ngôn ngữ lập trình C# [7].....	9
Hình 2.2 Ứng dụng của SQL Sever [9].....	11
Hình 2.3 Biểu tượng phần mềm CubeMX [10].....	12
Hình 2.4 Tạo mới một Project trên CubeMX.....	12
Hình 2.5 Chọn dòng chip cần sử dụng trên CubeMX.....	13
Hình 2.6 Cấu hình chân chip trên STM32 CubeMX.....	13
Hình 2.7 Cấu hình ngoại vi trên STM32 CubeMX.....	14
Hình 2.8 Chọn nguồn dao động và tần số trên STM32 CubeMX.....	14
Hình 2.9 Lưu file và sinh code trên STM32 CubeMX.....	15
Hình 2.10 Hình ảnh phần mềm Keli C V5 [10].....	15
Hình 2.11 Giao diện phần mềm Keli C V5.....	16
Hình 2.12 Biểu tượng phần mềm Visual Studio [11].....	17
Hình 2.13 Cấu tạo của STM32 [12].....	19
Hình 2.14 Cấu tạo cơ bản của động cơ bước [13].....	19
Hình 2.15 Động cơ bước Hybrid [13].....	20
Hình 2.16 Cấu tạo động cơ bước Hybrid [14].....	20
Hình 2.17 Cấp nguồn cuộn dây A cho động cơ [14].....	21
Hình 2.18 Cấp nguồn cuộn dây B cho động cơ [14].....	21
Hình 2.19 Mạch điều khiển động cơ bước Tb6560 [15].....	22
Hình 2.20 Sơ đồ nối chân LCD 16x2 [16].....	23
Hình 2.21 Mạch hạ áp LM2596 [17].....	24
Hình 3.1 Sơ đồ khối hệ thống.....	27
Hình 3.2 Nguồn tổ ong 12V-10A [15].....	28
Hình 3.3 Công tắc hành trình [16].....	28
Hình 3.4 Giao diện làm việc của phần mềm Visual Studio.....	29
Hình 3.5 Cửa sổ làm việc của phần mềm STM32 CubeMX.....	31
Hình 3.6 Lưu đồ thuật toán khối điều khiển trung tâm.....	32
Hình 3.7 Lưu đồ thuật toán khối giao diện.....	33
Hình 3.8 Lưu đồ thuật toán giao diện Winform C# bảo mật.....	34
Hình 3.9 Sơ đồ nguyên lý STM32 nhận tín hiệu và điều khiển.....	35
Hình 3.10 Hình ảnh thực tế UART USB to TTL CH340G [17].....	35
Hình 3.11 Mạch 3D khối xử lý.....	36
Hình 3.12 Mạch 1 sau khi thi công.....	37

<i>Hình 3.13 Hình ảnh mô hình sản phẩm.....</i>	<i>37</i>
<i>Hình 3.14 Giao diện Home.....</i>	<i>38</i>
<i>Hình 3.15 Giao diện vận hành.....</i>	<i>38</i>
<i>Hình 3.16 Giao diện tạo QR Code.....</i>	<i>39</i>
<i>Hình 3.17 Giao diện quản lý.....</i>	<i>39</i>

DANH SÁCH BẢNG

<i>Bảng 1.1 So sánh thư viện sử dụng RFID và QR Code [4].....</i>	<i>5</i>
<i>Bảng 1.2 Khả năng lưu trữ dữ liệu của QR Code [6].....</i>	<i>8</i>
<i>Bảng 2.1 Tính năng kỹ thuật của SQL Sever [8].....</i>	<i>11</i>
<i>Bảng 2.2 Chế độ hoạt động của động cơ bước [14].....</i>	<i>23</i>
<i>Bảng 2.3 Các thông số kỹ thuật của Tb6560 [15].....</i>	<i>24</i>
<i>Bảng 2.4 Các thông số kỹ thuật của LCD 16x2 [16].....</i>	<i>25</i>
<i>Bảng 2.5 Thông số kỹ mạch hạ áp LM2596 [18].....</i>	<i>26</i>
<i>Bảng 2.6 Mạch nguồn tổ ong 12V-10A [18].....</i>	<i>26</i>
<i>Bảng 3.1 Mức tiêu thụ điện năng của 1 số linh kiện.....</i>	<i>30</i>

MỤC LỤC

TÓM TẮT.....	i
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP.....	ii
LỜI NÓI ĐẦU.....	iii
CAM ĐOAN.....	iv
DANH SÁCH HÌNH VẼ.....	v
DANH SÁCH BẢNG.....	vi
MỤC LỤC.....	vii
CÁC CHỮ VIẾT TẮT VÀ CÁC KÝ HIỆU.....	viii
MỞ ĐẦU.....	1
CHƯƠNG 1: TỔNG QUAN.....	2
1.1. Thuật ngữ - khái niệm thư viện.....	2
1.2. Vai trò của thư viện hiện nay.....	2
1.3. Tính thực tế của thư viện trong thời đại hiện nay.....	3
1.4. Mô hình thư viện trong và ngoài nước.....	3
1.5. Giống và khác nhau giữa thư viện truyền thống và thư viện hiện đại.....	4
1.6. Công cụ quản lý trong thư viện hiện đại.....	5
1.7. So sánh thư viện sử dụng RFID và QR Code.....	5
1.8. Giới thiệu về QR Code.....	6
1.8.1. Đặt vấn đề	6
1.8.2. Định nghĩa QR Code	6
1.8.3. Cách hoạt động của QR Code	7
1.8.4. Ứng dụng của QR Code	7
1.8.5. Ưu điểm và nhược điểm	7
1.9. Đặc trưng của thư viện tự động sử dụng QR Code	8
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	10
2.1. Ngôn ngữ lập trình C#.....	10
2.1.1. Khái niệm ngôn ngữ lập trình C#	10
2.1.2. Các đặc trưng cơ bản của ngôn ngữ C#	10
2.1.3. Ứng dụng của C#	11
2.2. Tìm hiểu về SQL Sever.....	11
2.2.1. Khái niệm về SQL Sever	11
2.2.2. Ưu và nhược điểm SQL Sever	12

2.3. Phần mềm STM32 CubeMX.....	12
2.3.1. Tổng quan về STM32 CubeMX	12
2.3.2. Phần mềm Keil C V5	16
2.3.3. Phần mềm Visual Studio	17
2.4. Giới thiệu phần cứng.....	18
2.4.1. Giới thiệu về STM32F103C8T6	18
2.4.2. Giới thiệu về động cơ bước	20
2.4.3. Mạch điều khiển động cơ bước Tb6560	23
2.4.4. LCD 16x2	24
2.4.5. Mạch hạ áp LM259625	
2.4.6. Nguồn tổ ong 12V 10A	26
CHƯƠNG 3: THIẾT KẾ VÀ THI CÔNG.....	27
3.1. Sơ đồ khối hệ thống.....	27
3.1.1. Yêu cầu đề tài	27
3.1.3. Chức năng từng khối	28
3.2. Thiết kế đề tài.....	29
3.2.1. Thiết kế các động cơ bước	29
3.2.2. Thiết kế vi điều khiển trung tâm	30
3.2.3. Thiết kế mạch điều khiển động cơ bước	30
3.2.2. Thiết kế khối xử lý	31
3.2.3. Khối giao tiếp chuyển Đổi USB - TTL	36
3.3. Thi công sản phẩm.....	37
3.3.1. Mạch 3D	37
3.3.2. Mạch sau khi thi công	37
3.4. Mô hình sản phẩm và quá trình hoạt động.....	38
3.4.1. Mô hình sản phẩm	38
3.4.2. Quá trình hoạt động của hệ thống	41
3.5. Hướng dẫn sử dụng phần mềm.....	42
KẾT LUẬN.....	43
TÀI LIỆU THAM KHẢO.....	45
PHỤ LỤC.....	48

CÁC CHỮ VIẾT TẮT VÀ CÁC KÝ HIỆU

< :	Dấu bé hơn
%	Phần trăm
...	Còn nữa
STM32	STM32F103C8T6
SQL Sever	Quản trị cơ sở dữ liệu
QR Code	Mã vạch ma trận
LM2596	Mạch hạ áp
TB6560	Mạch điều khiển động cơ bước
C#	Ngôn ngữ lập trình C#

MỞ ĐẦU

Hiện nay, đất nước ta đang trong giai đoạn phát triển, thực hiện công cuộc công nghiệp hóa, hiện đại hóa. Cùng với sự phát triển của nền kinh tế thị trường, việc nâng cao tri thức để đáp ứng nhu cầu xã hội luôn được ưu tiên hàng đầu vì vậy sách luôn là nguồn tri thức dồi dào để đáp lại vấn đề đó tuy nhiên ở 1 số thư viện hoặc nhà sách hiện nay vẫn chủ yếu dùng con người để thực hiện việc trao đổi và mua bán sách chính vì vậy mà nhóm em đã quyết định chọn đề tài “ Hệ thống thư viện tự động sử dụng QR Code ”. Với Mục đích:

Tối ưu được quá trình mượn và trả sách để người dùng không còn thấy bất tiện khi đến thư viện để mượn sách, và có thể tương lai sẽ phát triển những mô hình thư viện sách tự động đặt mọi nơi để khơi dậy cảm hứng đọc sách đang dần mai một như hiện nay.

Đây là đề tài bước đầu tạo cơ sở nền tảng kiến thức cơ bản để từ đó tìm tòi phát triển ứng dụng khác. Sau khi hoàn thành đồ án nhóm cần đạt các mục tiêu:

- Sử dụng camera để quét QR Code tăng tính chính xác cao
- Thiết kế trực tuyến và trả sách một cách tự động
- Thiết kế giao diện Winform C# với các tính năng tiện ích dễ quản lý
- Biết cách lập trình ngôn ngữ lập trình C#, sử dụng phần mềm Visual Studio, Keil C V5 và biết được phương pháp để điều khiển thiết bị
- Thiết kế được một giao diện để giám sát trên máy tính
- Xây dựng được mô hình mô phỏng

Đối tượng nghiên cứu mà đề tài hướng đến đó là:

phần mềm Keil C V5, STM 32 CubeMX để setup và lập trình cho STM32, thiết kế giao diện Winform C#, mạch điều khiển động cơ bước Tb6560 và động cơ bước, mạch nạp ST-Link V2, USB TO TTL CH340..

Và phạm vi nghiên cứu: ở đề tài này thì ngoài những kiến thức đã học thì nhóm phải tìm hiểu thêm STM32 và ngôn ngữ lập trình C#, tối ưu thiết kế mô hình.

Đề tài nhóm em được hoàn thành một phần nhận được sự giúp đỡ tận tình của cô Võ Thị Hương, cũng như kiến thức tích lũy được từ bộ môn khoa điện – điện tử và qua tìm hiểu trên mạng để thực hiện. Rất mong nhận được sự góp ý của hội đồng bảo vệ, thầy cô, và các bạn sinh viên để có thể nâng cao chất lượng của đồ án.

Nội dung đề tài bao gồm: tổng quan đề tài, cơ sở lý thuyết, xây dựng phần cứng

Cấu trúc Đề tài nhóm em gồm có ba chương:

Chương 1: Tổng quan

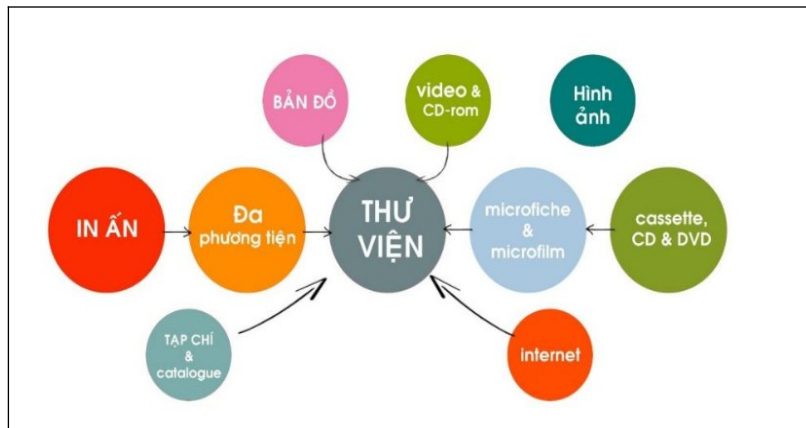
Chương 2: Cơ sở lý thuyết

Chương 3: Thiết kế và thi công

CHƯƠNG 1: TỔNG QUAN

1.1. Thuật ngữ - khái niệm thư viện

Theo ý nghĩa truyền thống, thuật ngữ thư viện là sưu tập sách và tạp chí. Tuy nó có thể đến từ kho sưu tập cá nhân của riêng một người, nhưng nó chỉ đến từ những nhà sưu tập sách báo xuất bản được bảo quản bởi thành phố hay học viện. Tuy nhiên ngày nay giấy không còn là phương tiện duy nhất để lưu trữ thông tin. Vì thế khái niệm thư viện mới nhất của UNESCO không phụ thuộc vào tên gọi: là bất kì bộ sưu tập có tổ chức của sách, báo, tài liệu, ấn phẩm định kì...

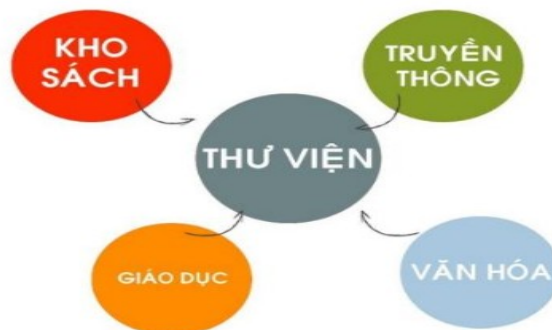


Hình 1. 1 Các phương tiện lưu trữ của thư viện [1]

1.2. Vai trò của thư viện hiện nay

Thư viện trên thế giới ngày nay đã thoát khỏi khía cạnh của những kho chứa sách để trở nên năng động hơn với 3 vai trò chủ yếu sau đây:

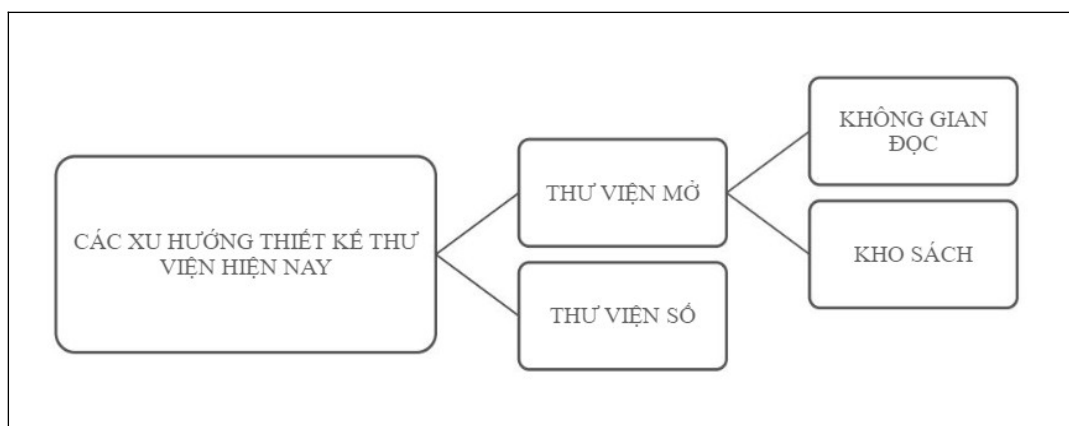
- Thư viện đóng góp vào việc cải tiến giáo dục
- Thư viện là một phần trung tâm của việc phát triển văn hóa
- Thư viện là một trong những cơ quan truyền thông đại chúng



Hình 1. 2 Vai trò của thư viện hiện nay [1]

1.3. Tính thực tế của thư viện trong thời đại hiện nay

Thư viện đem lại hiệu quả cao về tính giáo dục và tính nhân văn. Góp phần khuyến khích môi trường tự học và tự nghiên cứu, kích thích sự chủ động của người học. Là nơi giao lưu, sinh hoạt cộng đồng, là biểu tượng của bộ mặt văn hóa xã hội.



Hình 1. 3 Xu hướng thiết kế thư viện hiện nay [2]

1.4. Mô hình thư viện trong và ngoài nước

Trong nước: một số mô hình thư viện đã được đi vào hoạt động và đạt được những hiệu quả tích cực cao như: thư viện ngoài trời, thư viện xanh, thư viện thân thiện... với việc giúp các bạn đọc gần gũi với một thế giới sách mà còn xây dựng một văn hóa đọc. Nhằm nâng cao chất lượng phục vụ cũng như quản lý thư viện những năm gần đây Việt Nam đang đẩy mạnh các mục tiêu và giải pháp hướng đến việc hiện đại hóa thư viện, áp dụng các trang thiết bị hiện đại như: máy vi tính, phần mềm chuyên dụng, hệ thống an ninh tài liệu như: camera, RIFD, QR Code... tùy thuộc vào nhu cầu người dùng và công nghệ từ đó thư viện hiện đại được thiết kế với các cấp độ khác nhau và tương ứng với chúng là những tên gọi khác nhau như: thư viện đa phương tiện, thư viện điện tử, thư viện số... việc ứng dụng công nghệ thông tin đang góp phần thay đổi về chất lượng thư viện trong nước.



Hình 1. 4 Hình ảnh mô hình thư viện ở Việt Nam [2]

Ngoài nước: với nền công nghệ và khoa học phát triển các mô hình thư viện ở nước ngoài đã đạt được những thành công nhất định. Tuy nhiên với xu thế phát triển của thế giới hiện nay, mô hình được xem là hữu ích nhất: Learning commons - không gian học tập chung là mô hình kết hợp giữa thư viện với công nghệ thông tin và các dịch vụ hỗ trợ học tập nhằm cung cấp những dịch vụ tốt nhất đáp ứng các nhu cầu của người dùng. Đây là một không gian học tập với nội thất, trang thiết bị công nghệ hiện đại. Mô hình này nhằm đến không gian cho phép tự tra cứu, đọc tài liệu hoặc nghiên cứu, hoặc đơn giản là nghỉ ngơi, thư giãn. Mô hình cho phép người dùng chủ động sử dụng các trang thiết bị hiện đại để tiếp cận các tài liệu hoặc không cần sự tác động của con người. Như vậy thư viện sẽ phát huy được tối đa tính chủ động trong việc tiếp cận tài liệu từ đó đem lại cho họ tính tự học và nghiên cứu tốt nhất.



Hình 1. 5 Hình ảnh mô hình thư viện Learning commons [3]

1.5. Giống và khác nhau giữa thư viện truyền thống và thư viện hiện đại

Giống nhau:

- Người quản lý vẫn là con người. Các hoạt động chuyên môn và hoạt động khác đều do con và chính con người thực hiện
- Quản lý thư viện đều sử dụng các nguyên tắc, phương pháp và công cụ quản lý
- Nội dung quản lý: quản lý cơ sở vật chất và chi phí, quản lý nguồn nhân lực và các hoạt động khác

Khác nhau:

- Về công cụ quản lý: thư viện truyền thống sử dụng các công cụ quản lý như: ghi chép, đo, đếm. Chẳng hạn như việc ghi chép người dùng mượn và trả sách qua sổ sách. Trong thư viện hiện đại người dùng quản lý sử dụng các

công nghệ tiên tiến và hiện đại như máy móc, phần mềm, camera... để quản lý

- Về trình độ quản lý: thư viện truyền thống yêu cầu người quản lý phải có chuyên môn, trình độ và kỹ năng quản lý. Bên cạnh những kiến thức trên thư viện hiện đại đòi hỏi người quản lý phải có các kiến thức về ứng dụng công nghệ thông tin vào thư viện và đặc biệt biết sử dụng thành thạo các phần mềm.

Với những mặt vượt trội thì thư viện hiện đại đang ngày càng phát triển và có tính ứng dụng cao trong cuộc sống ngày nay.

1.6. Công cụ quản lý trong thư viện hiện đại

RFID: được sử dụng rộng rãi trong việc quản lý thư viện vào những năm 2000 hướng đến sự tiện nghi và chủ động cho người dùng. Ngay thời điểm áp dụng, RFID đã cho thấy được sự tiện ích và ưu thế so với các công nghệ quản lý thư viện trước đây. Tuy nhiên với giá thành cũng như chi phí vận hành khá cao vượt ngoài tầm với của các thư viện.

QR Code: bên cạnh RFID thì QR Code cũng là một phương thức để áp dụng trong mô hình thư viện, người dùng không cần phải luôn mang thẻ bên mình vẫn có thể mượn sách thông qua chiếc điện thoại smart phone để quét QR Code và từng loại sách sẽ có 1 mã khác nhau để phân biệt tránh sự nhầm lẫn trong quá trình mượn và trả sách. Tuy nhiên những mã này có thể bị ảnh hưởng từ môi trường nên cần được bảo quản kỹ tránh bị bay mực trên mã thì camera mới có thể quét được dữ liệu.

1.7. So sánh thư viện sử dụng RFID và QR Code

RFID	QR Code
-Có thể đọc thẻ ở khoảng cách xa (100m)	-Nhỏ gọn và dễ sử dụng
-Đọc được nhiều thẻ cùng một lúc	-Ít tốn kém
-Bảo mật cao	-Mã vạch hoạt động với độ chính xác cao
-Độ bền lớn và có thể tái sử dụng	-Phù hợp để phân biệt nhiều loại sản phẩm
-Giá thành cao	-Thiết bị hiện nay vừa có thể đọc hoặc tạo mã vạch cho sản phẩm
-Tín hiệu truyền kém khi đi qua kim loại, chất lỏng	-Khoảng cách đọc khá gần
-Có hiện tượng xung đột tín hiệu	-Có thể bị ảnh hưởng bởi môi trường

Bảng 1.1 So sánh thư viện sử dụng RFID và QR Code [4]

Những ưu thế cũng như việc tiếp cận người dùng dễ dàng thì QR Code đang được hướng đến trong các mô hình quản lý thư viện.

1.8. Giới thiệu về QR Code

1.8.1. Đặt vấn đề

Hiện nay QR Code đang dần đi sâu vào cuộc sống của con người, QR Code đang thay thế những phương thức lưu trữ dữ liệu truyền thống như hiện nay bởi tính ưu việt của nó. QR Code được coi là công cụ giúp ích cho việc quản lý thư viện, đặc biệt đối với các trường đại học với số lượng lớn sinh viên sử dụng điện thoại thông minh thông qua đó nâng cao chất lượng phục vụ trong việc quản lý sách khắc phục được các nhược điểm chật hẹp của thư viện bằng một giao diện quản lý người dùng. Vì vậy trong đề tài này chúng ta sẽ cùng tìm hiểu và làm rõ về QR Code và những lợi ích thực tế mà mã này đã thể hiện qua đề tài này.

1.8.2. Định nghĩa QR Code

QR Code là viết tắt của “ Quick response code ”, hay còn được gọi là mã vạch ma trận hoặc mã vạch 2D, QR Code được xuất hiện vào năm 1994 do Denso Wave (công ty con của toyota). QR Code có hình dạng là những chấm đen và các ô vuông mẫu trên nền trắng, có thể chứa dữ liệu đa dạng như: ảnh, URL, giới thiệu sản phẩm, sự kiện QR Code là 1 dạng mã hóa thay thế gọn gọn cho chuỗi ký tự nào đó như: SMS, Số điện thoại, URL ... hoặc 1 nội dung bất kỳ.



Hình 1.6 Hình ảnh cấu trúc QR Code [5]

Với nền công nghệ 4.0 phát triển mạnh việc tạo QR Code hiện nay rất đơn giản bởi hiện có rất nhiều cách nhưng có hai cách được sử dụng phổ biến:

Tạo trực tiếp trên website: một vài trang web cho phép bạn tạo QR Code ngay trên trang của họ chỉ cần bạn nhập thông tin và trang sẽ xuất ra QR Code cho bạn dùng.

Tạo bằng các phần mềm trên điện thoại: với hai hệ điều hành là Android và IOS.

Cách quét QR Code:

Hiện nay công cụ để đọc QR Code có rất nhiều như phần mềm máy tính, máy quét mã vạch (được sử dụng rộng rãi trong siêu thị, winmart ...). Nhưng thông dụng hơn chính là điện thoại di động bằng các phần mềm hỗ trợ quét mã trên máy (zalo, scan ,)

1.8.3. Cách hoạt động của QR Code

Mọi dữ liệu trong QR Code đều được mã hóa theo mã nhị phân.

Máy quét QR Code sẽ dựa vào hình dáng, nhận diện 3 ô vuông trong mã từ đó định hình được hình dáng và vị trí nội dung có thể quét.

Sau đó máy sẽ bắt đầu phân tích. Lúc đầu máy quét sẽ biến hình ảnh thành những ô vuông nhỏ khác nhau. Mỗi ô vuông riêng lẻ này đều chứa một vùng dữ liệu riêng dựa trên việc màu trắng hoặc đen. Và sau đó máy sẽ ghép những mảnh hình vuông nhỏ đó lại và tạo thành một vùng dữ liệu lớn hơn.

1.8.4. Ứng dụng của QR Code

Với ưu điểm tiện lợi nên đa số mọi lĩnh vực đều áp dụng ví dụ: y tế (dùng QR Code thay cho giấy đi đường hoặc chứng nhận tiêm vacxin), mua sắm và thanh toán (app momo, Vnpay sử dụng QR Code để thanh toán mọi chi phí, kiểm tra nguồn gốc sản phẩm), lưu trữ thông tin bảo mật ...

1.8.5. Ưu điểm và nhược điểm

Ưu điểm:

- Khối lượng dữ liệu lớn
- Dễ dàng tiếp cận với người dùng
- Thông tin vô cùng đa dạng, từ văn bản, hình ảnh, video, cho tới việc truy cập WiFi,...
- Tiết kiệm thời gian
- Không yêu cầu thiết bị phức tạp, chỉ cần một chiếc điện thoại thông minh có camera là đủ

Nhược Điểm:

- Không thể quét nếu không có mạng. Mặc dù ai cũng biết đâu đâu cũng có WiFi nhưng chỉ có ở trong các tòa nhà, hàng quán, khu dân cư mới có, mà đâu có được dùng miễn phí. Mạng miễn phí thì còn lâu mới phủ sóng hết được. Do đó đây là một rào cản lớn
- Nếu bạn là người lần đầu tiếp xúc với QR Code thì có lẽ sẽ gặp không ít khó khăn. Đặc biệt là đối với người lớn tuổi hay người chưa được tiếp xúc nhiều với Internet

1.8.6. So sánh QR Code với mã vạch truyền thống

QR Code có rất nhiều ưu điểm tiện lợi so với mã vạch truyền thống như là đối với mã vạch truyền thống chỉ có thể lưu tối đa 20 ký tự số trong khi đó QR Code có thể lưu tối đa 7.089 ký tự số và 4.296 ký tự chữ số chính vì vậy mà lượng thông tin lưu trữ sẽ nhiều hơn, bên cạnh đó tính thẩm mỹ của QR Code nhìn gọn gàng và đỡ rối mắt so với mã vạch truyền thống.



Hình 1. 7 Hình ảnh bản nâng cấp của mã vạch truyền thống [6]

Số đơn thuần	Tối đa 7.089 kí tự
Số và chữ cái	Tối đa 4.296 kí tự
Số nhị phân	Tối đa 2.953 Byte

Bảng 1.2 Khả năng lưu trữ dữ liệu của QR Code [6]

1.9. Đặc trưng của thư viện tự động sử dụng QR Code

Sau khi tìm hiểu khái niệm về QR Code là gì, nhóm chúng em đã có cái nhìn khái quát về QR Code. Để tìm hiểu đặc trưng của QR Code và hoạt động của mô hình này gồm:

➤ Tự động hoá

Là sự vận hành của máy móc được con người lập trình sẵn, cùng với đó là khả năng tự động hoá trong khâu vận hành, ở đó chúng ta sẽ tham gia rất ít vào quá trình sản xuất hay có thể là không cần tham gia vào. Do đó mà hệ thống thư viện sử dụng QR Code sẽ vận hành một cách tự động.

➤ Tính thông minh

Là một trong những đặc trưng quan trọng và không thể thiếu được của đề tài đó là việc tận dụng tối ưu việc quét QR Code. Thông qua QR Code chúng ta có thể quản lý được số lượng lớn sách ở trên kệ mang lại tính hiệu quả, chính xác và nhanh chóng hơn.

➤ Thời gian thực

Thư viện tự động là sự kết hợp từ nhiều phương tiện giữa con người và máy móc. Vậy nên việc giám sát hay vận hành đều được diễn ra trong thời gian thực. Từ đó đem lại khá nhiều lợi ích đáng kể cho người dùng. Cụ thể như người dùng có thể tối ưu hoá sử dụng nguồn lực 1 cách hiệu quả, tăng hiệu suất ,...

➤ **Chủ động**

Như chúng ta đã biết thì thư viện tự động là mô hình hoạt động cơ bản dựa trên thiết bị, máy móc. Do đó, sự chủ động là một trong những khả năng ưu việt của thư viện tự động.

➤ **Linh hoạt**

Không những tự động mà thư viện tự động còn giúp người dùng có tính linh hoạt hơn. Bằng việc sử dụng QR Code người dùng có thể dễ dàng cập nhật sách có trên kệ đảm bảo tính linh hoạt của hệ thống.

➤ **Tối ưu hoá**

Với sự liên kết chặt chẽ giữa các yếu tố con người, công nghệ đã tạo ra một hệ thống đạt chất lượng cũng như đạt năng suất tốt hơn. Quan trọng, thư viện tự động giúp ích cho việc tối ưu thời gian và nhân công cho người dùng.

Trong thời buổi công nghệ 4.0 thì việc áp dụng QR Code vào đời sống giúp ích rất nhiều cho chúng ta. Để khai thác, áp dụng hệ thống này một cách hiệu quả thì đầu tiên người dùng phải hiểu về nó.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Ngôn ngữ lập trình C#

2.1.1. Khái niệm ngôn ngữ lập trình C#

C sharp (được viết tắt C#) ngôn ngữ lập trình được Microsoft đầu tư ra đời ngày 12/2/2002 cùng với. Net Framework 1.0. C# có sự kết hợp chặt chẽ cùng với. NET Framework nhằm hỗ trợ tạo ra các ứng dụng Windows Forms hoặc WPF một cách dễ dàng và nhanh chóng hơn.



Hình 2.1 Hình ảnh ngôn ngữ lập trình C# [7]

2.1.2. Các đặc trưng cơ bản của ngôn ngữ C#

- Ngôn ngữ lập trình hướng đối tượng
- Tuân theo chuẩn ECMA
- Bộ thu gom rác tự động
- An toàn kiểu
- Hỗ trợ Generic
- Partial Classes
- Anonymous Methods
- Nullable Type
- Lập trình song song

2.1.3. Ứng dụng của C#

- Xây dựng ứng dụng trên nền tảng web, web service
- Game
- Ứng dụng Windows Store
- Moblie
- Các ứng dụng Destop
- Điện toán đám mây

2.2. Tìm hiểu về SQL Sever

2.2.1. Khái niệm về SQL Sever

SQL Sever là viết tắt của cụm từ “Structured Query Language” là công cụ sử dụng để tổ chức, quản lý và truy xuất dữ liệu được lưu trữ trong các cơ sở dữ liệu. SQL Sever là một hệ thống ngôn ngữ bao gồm tập các câu lệnh sử dụng để tương tác với cơ sở dữ liệu quan hệ.

SQL Sever được coi là một ngôn ngữ hoàn thiện vì được sử dụng trong hệ thống cơ sở dữ liệu và là một thành phần rất quan trọng không thể thiếu trong các hệ quản trị cơ sở dữ liệu tuy khác ngôn ngữ lập trình như C, C++, Java... nhưng các câu lệnh mà SQL Sever cung cấp có thể được nhúng vào trong các ngôn ngữ lập trình nhằm xây dựng các ứng dụng tương tác với cơ sở dữ liệu.

Định nghĩa dữ liệu	SQL Sever cung cấp khả năng định nghĩa các cơ sở dữ liệu, các cấu trúc lưu trữ và tổ chức dữ liệu cũng như mối quan hệ giữa các thành phần dữ liệu
Truy xuất và thao tác dữ liệu	Với SQL Sever, người dùng có thể dễ dàng thực hiện các thao tác truy xuất, bổ sung, cập nhật và loại bỏ dữ liệu trong các cơ sở dữ liệu
Điều khiển truy cập	SQL Sever có thể được sử dụng để cấp phát và kiểm soát các thao tác của người sử dụng trên dữ liệu, đảm bảo sự an toàn cho cơ sở dữ liệu
Đảm bảo toàn vẹn dữ liệu	SQL Sever định nghĩa các ràng buộc toàn vẹn trong cơ sở dữ liệu nhờ đó đảm bảo tính hợp lệ và chính xác của dữ liệu trước các thao tác cập nhật cũng như các lỗi của hệ thống

Bảng 2.1 Tính năng kỹ thuật của SQL Sever [8]

2.2.2. Ưu và nhược điểm SQL Sever

Ưu điểm:

- Hỗ trợ cài nhiều phiên bản SQL Sever khác nhau trên cùng 1 máy tính
- Duy trì riêng biệt các môi trường sản xuất, phát triển, thử nghiệm
- Giảm thiểu các vấn đề tạm thời trên cơ sở dữ liệu
- Tách biệt các đặc quyền bảo mật
- Duy trì máy chủ dự phòng

Nhược điểm:

- SQL Sever chỉ chạy trên window
- Cần thanh toán phí để chạy nhiều CSDL



Hình 2.2 Ứng dụng của SQL Sever [9]

2.3. Phần mềm STM32 CubeMX

2.3.1. Tổng quan về STM32 CubeMX

STM32 CubeMX là một phần mềm giúp ích cho việc cấu hình ngoại vi, clock, tính toán dòng tiêu thụ, tạo project với nhiều dòng chip ARM STM32.... , cấp clock tùy chỉnh mà không cần liên quan đến code.

Việc tạo project với thư viện chuẩn (standard library) là khá khó khăn vì cần nhiều bước tạo project mới. STM32 CubeMX ra đời để làm điều đó, với giao diện đơn giản để chúng ta dễ dàng lập trình.

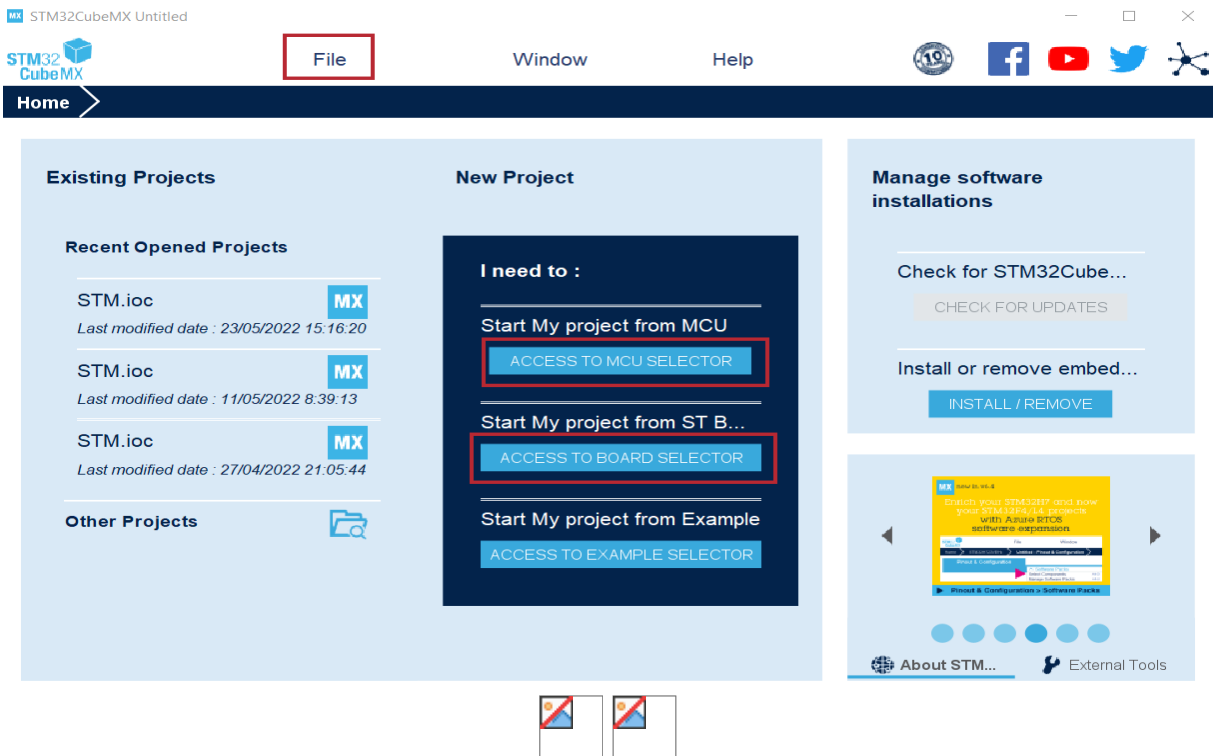


Hình 2.3 Biểu tượng phần mềm CubeMX [10]

Hướng dẫn sử dụng phần mềm CubeMX cơ bản:

Bước 1: Để tạo một Project mới ta nhấn vào:

File -> New Project hoặc nhấn vào ACCESS TO MCU SELECTOR nếu lập trình 1 MCU STM32 bất kỳ, hoặc nhấn vào ACCESS TO BOARD SELECTOR nếu bạn lập trình trên 1 board của hãng STM32.

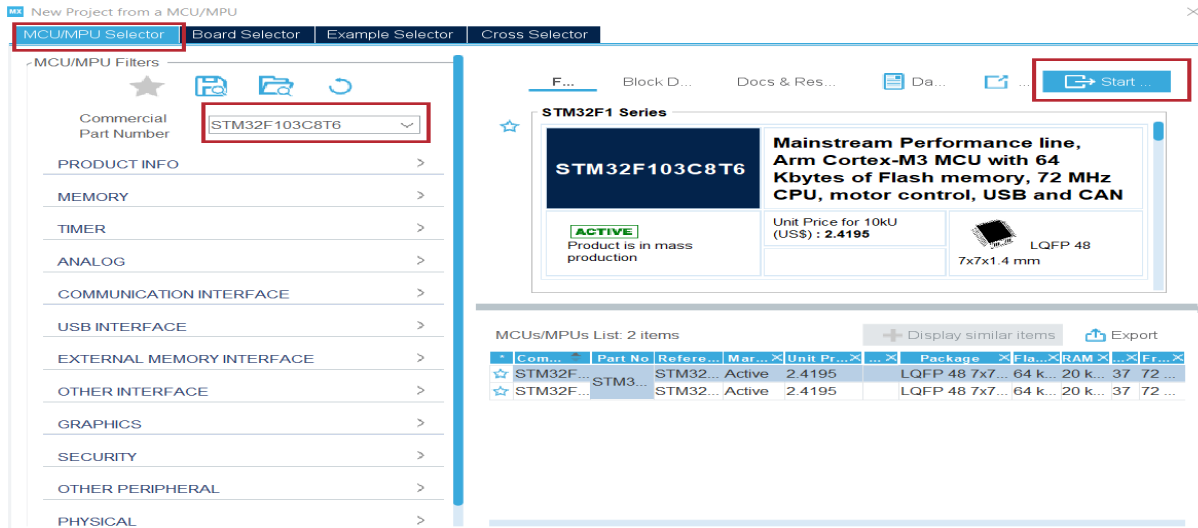


Hình 2.4 Tạo mới một Project trên CubeMX

Bước 2: Sau khi tạo mới một Project giao diện chọn vi điều khiển STM32 sẽ xuất hiện.

Chọn vi điều khiển: Tại mục Part Number Search nhập tên vi điều khiển muốn cấu hình.

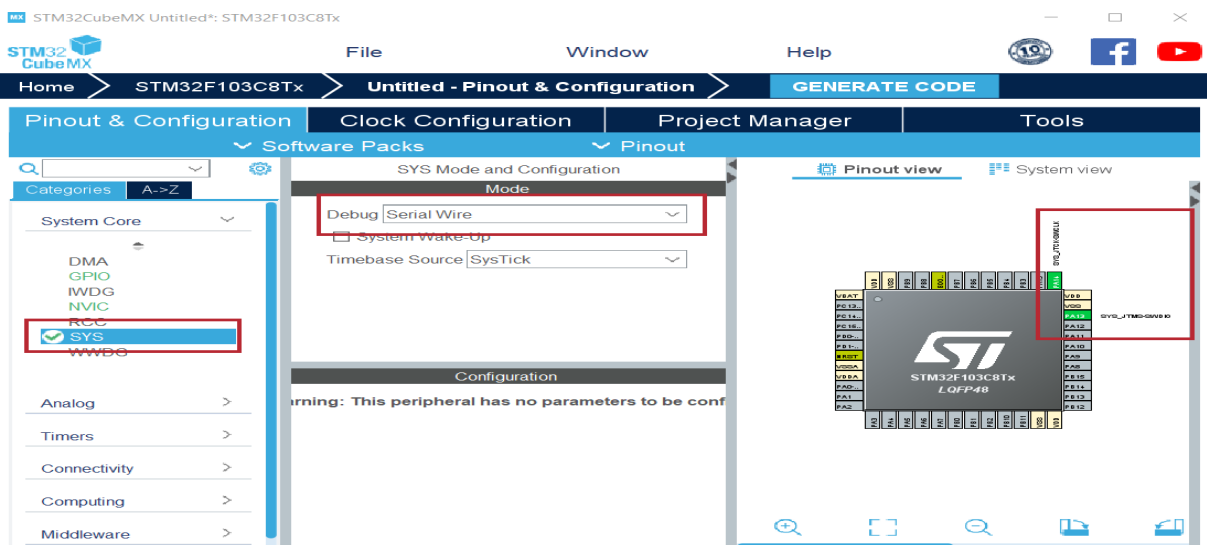
Bắt đầu Project: Nhấn vào Start Project.



Hình 2.5 Chọn dòng chip cần sử dụng trên CubeMX

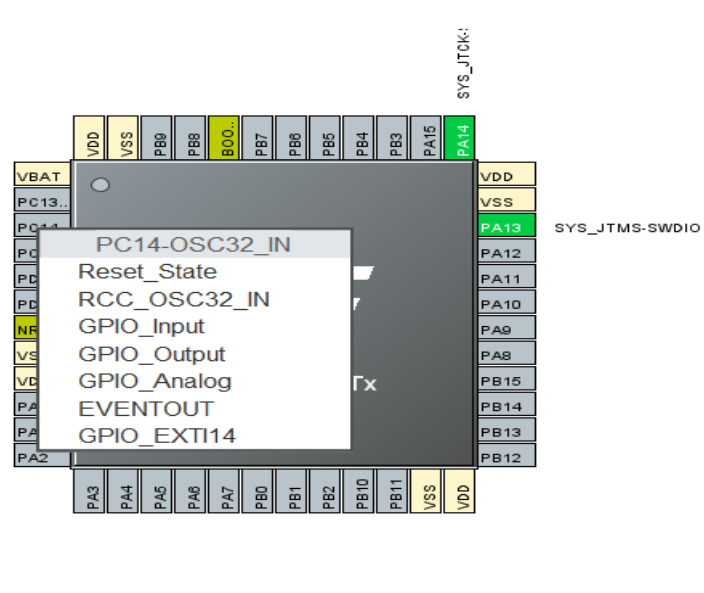
Bước 3:

Tại mục Pinout & Configuration: Chọn system Core -> SYS -> Debug: Serial Wire để vi điều khiển được cấu hình nạp code thông qua chân SWDIO và SWCLK (Sử dụng mạch nạp St – Link để kết nối STM32F103C8T6 qua các chân này.



Hình 2.6 Cấu hình chân chip trên STM32 CubeMX

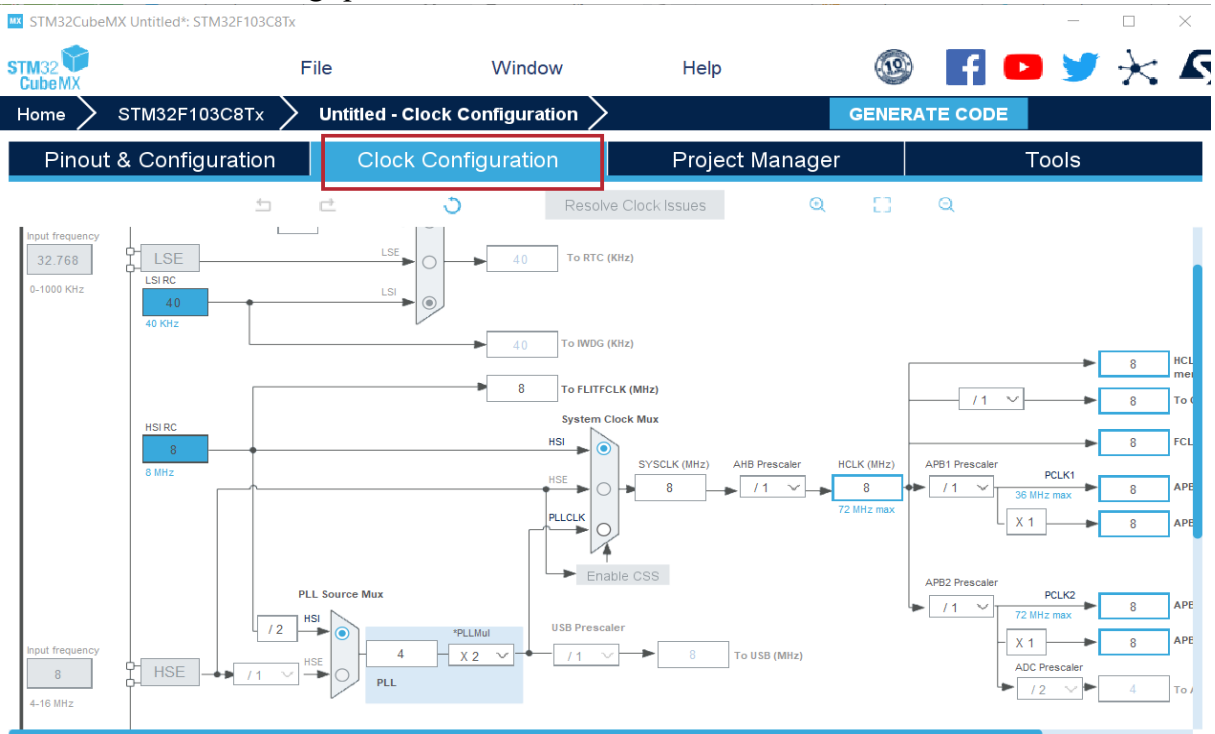
- Bước 4: Cấu hình ngoại vi bằng cách click chuột phải vào chân mình muốn đặt
Vd: Cài đặt chân PC14 hoạt động với chức năng OUTPUT



Hình 2.7 Cấu hình ngoại vi trên STM32 CubeMX

Ngoài ra cấu hình các ngoại vi khác như: Analog , Timers , Connetivity...

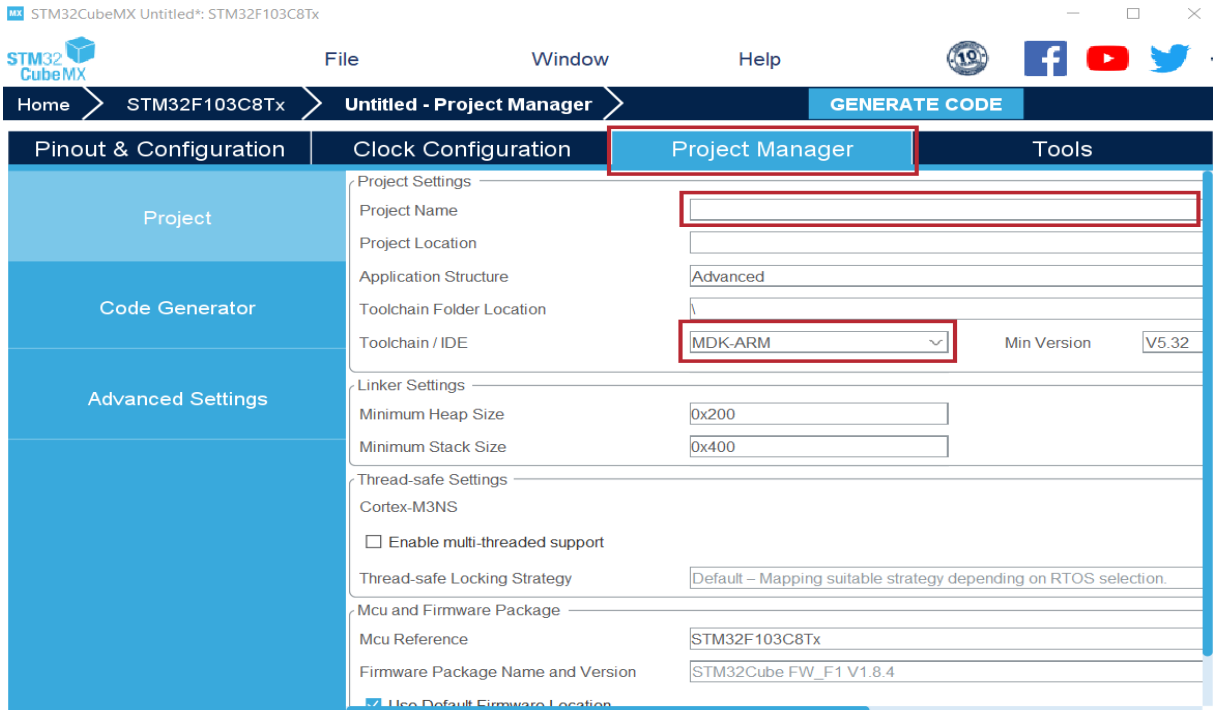
- Bước 5: Tại mục Clock Configuration cấu hình nguồn dao động và tần số hoạt động của vi điều khiển thông qua Clock tree.



Hình 2.8 Chọn nguồn dao động và tần số trên STM32 CubeMX

Bước 6:

Chúng ta vào mục Project manager, đặt tên cho project, nơi lưu trữ và chọn Toolchain /IDE là MDK-ARM V5 nếu sử dụng Keil C IDE để code và debug. Cấu hình xong chúng ta chọn vào GENERATE CODE để sinh code.



Hình 2.9 Lưu file và sinh code trên STM32 CubeMX

2.3.2. Phần mềm Keil C V5

Keil C là một phần mềm hỗ trợ người dùng lập trình các vi điều khiển nhiều loại khác nhau (Atmel,VRM,...) phần mềm giúp chúng ta soạn thảo và biên dịch chương trình C hay ASM thành ngôn ngữ máy tính để nạp cho vi điều khiển có thể nói Keil C là người thông dịch viên giữa người lập trình và máy tính.

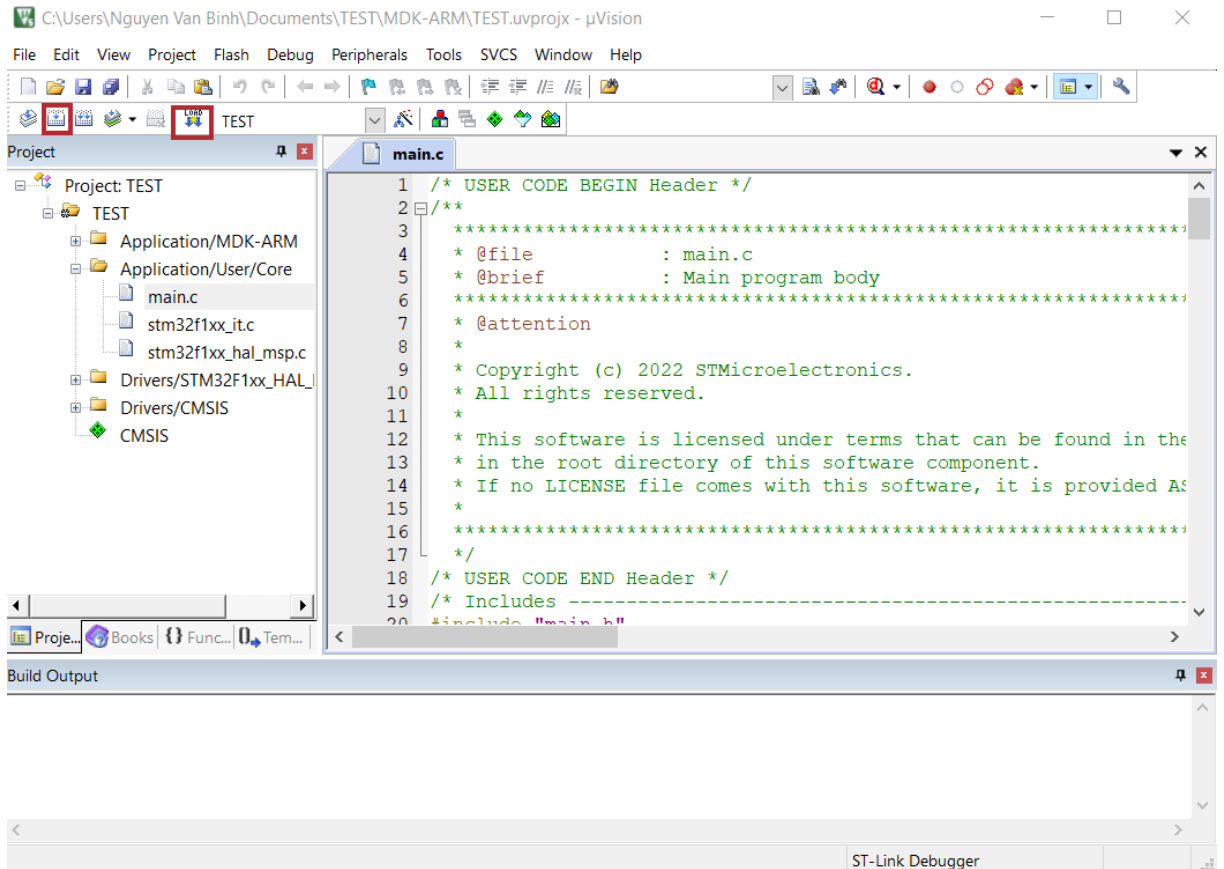


Hình 2.10 Hình ảnh phần mềm Keli C V5 [10]

Thao tác trên Keli C V5:

Sau khi sinh code từ phần mềm STM32 CubeMX và mở Project, ta mở file main.c tại mục Application/User

Chúng ta Buid (F7) để compile code và Load (F8) để nạp code



Hình 2.11 Giao diện phần mềm Keli C V5

2.3.3. Phần mềm Visual Studio

Phần mềm Visual Studio là một trong những phần mềm hỗ trợ và giúp ích cho việc lập trình. Microsoft là nơi sáng tạo và tạo nên phần mềm này. Lúc trước, Visual Studio có tên mã là Project Boston. Nhưng sau này, Microsoft đã hợp tác và đóng gói thành 1 sản phẩm duy nhất.

Visual Studio là 1 hệ thống mà ở đó tổng hợp tất cả những gì liên quan tới phát triển ứng dụng , trong đó có trình thiết kế, trình chỉnh mã, gỡ lỗi. Có nghĩa là chúng ta có thể sử dụng, viết code tiện ích và dễ dàng hơn với phần mềm Visual Studio. Không những thế, chúng ta có thể thiết kế cũng như tạo giao diện theo ý của mình, trải nghiệm trong Visual Studio như lúc phát triển ứng dụng UWP , Xamarin , ...

Visual Studio gồm có 2 phiên bản là Visual Studio Professional và Visual Studio Enterprise, những phiên bản này có tính phí và thông dụng với những công ty chuyên về lập trình

Gồm nhiều tính năng:

- + Sử dụng nhiều ngôn ngữ lập trình
- + Nhiều nền tảng
- + Tính năng comment
- + Có hỗ trợ Git
- + Có hỗ trợ debug
- + Màn hình đa nhiệm
- + Lưu trữ an toàn
- + Phân cấp trong lưu trữ
- + Kho tiện ích phong phú
- + Dễ dàng viết Code[13]



Hình 2.12 Biểu tượng phần mềm Visual Studio [11]

2.4. Giới thiệu phần cứng

2.4.1. Giới thiệu về STM32F103C8T6

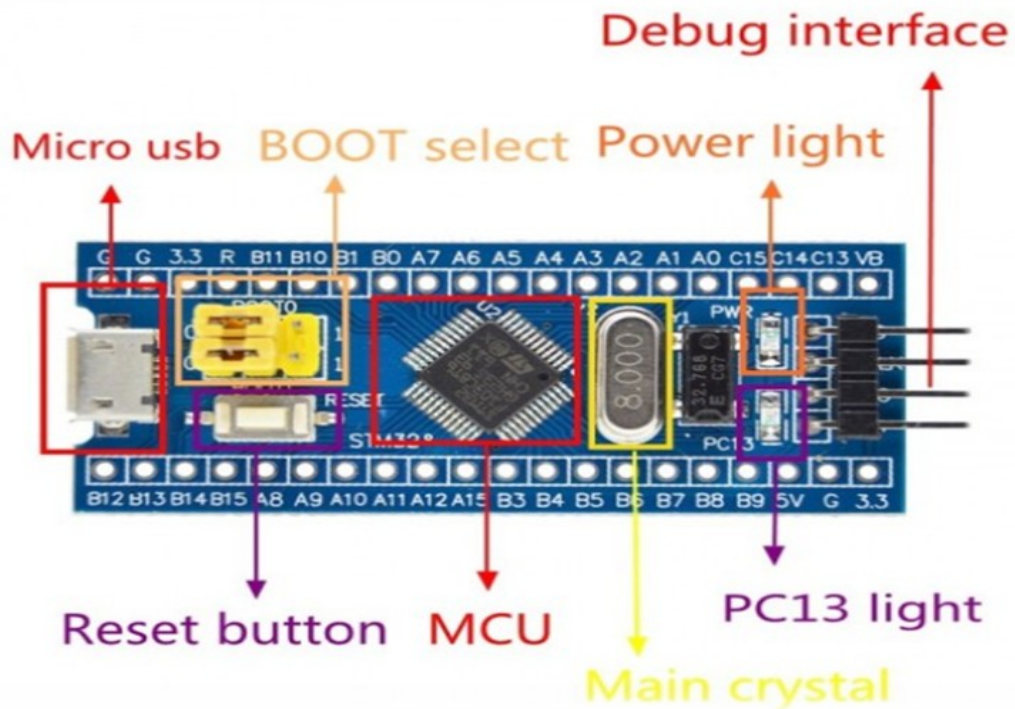
STM32 là chip khá phổ biến của dòng ST gồm có rất nhiều họ phổ biến là F0, F1, F2, F3, F4 Ở đó thì STM32f103 là họ F1 gồm lõi ARM COTEX M3. Các thông số cơ bản của STM32 là nó có thể có tốc độ tối đa là 72 Mhz và nó là vi điều khiển 32 bit. Bên cạnh đó thì giá thành nó lại phù hợp và rẻ so với những vi điều khiển có chức năng tương tự. STM32 có mạch nạp dễ sử dụng.

Gồm những ứng dụng cơ bản: Điều khiển ứng dụng thông thường, thiết bị cầm tay và thuốc, dùng cho driver để điều khiển ứng dụng, máy tính và thiết bị ngoại vi chơi

game, thiết bị lập trình PLC, GPS cơ bản, hệ thống cảnh báo, các ứng dụng trong công nghiệp, ...

Cấu hình chi tiết của STM32F103C8T6:

- ARM 32 bit Cortex M3, Clock max: 72 Mhz
- Reset, quản lý nguồn và Clock
 - Điện áp hoạt động từ 2 đến 3.6 V
 - Power on reset(POR), Power down reset(PDR) và programmable voltage detector (PVD)
 - Thạch anh ngoài từ 4 đến 20 Mhz
 - Thạch anh nội dùng dao động RC ở mode 8Mhz hoặc 40khz.
 - Sử dụng thạch anh ngoài 32.768khz được sử dụng cho RTC
- Về bộ nhớ:
 - 20KB SRAM
 - 64 KB bộ nhớ Flash
- Trong trường hợp điện áp thấp:
 - Những module như ngủ, dừng đều có thể hoạt động ở chế độ chờ
 - Cấp nguồn ở chân Vbat bằng pin để hoạt động bộ RTC và sử dụng lưu trữ data khi mất nguồn cấp chính
- 2 bộ ADC 12 bit với 9 kênh cho mỗi bộ
 - Giá trị chuyển đổi từ 0 đến 3.6 V
 - Lấy mẫu 1 kênh hay nhiều kênh
 - Hỗ trợ cảm biến nội
- DMA: bộ chuyển đổi này giúp tăng tốc độ xử lý do không có sự can thiệp quá sâu của CPU
 - 7 kênh DMA
 - Hỗ trợ DMA cho UART, I2C, SPI, ADC
- Gồm 7 timer:
 - 3 timer 16 bit hỗ trợ các mode IC/OC/PWM
 - 1 timer 16 bit hỗ trợ để điều khiển động cơ với các mode bảo vệ như ngắt input, dead-time..
 - 2 watchdog timer dùng để bảo vệ và kiểm tra lỗi
 - 1 sysTick timer 24 bit đếm xuống dùng cho các ứng dụng như hàm Delay....
- 9 kênh giao tiếp gồm có:
 - 2 bộ I2C(SMBus/PMBus)
 - 3 bộ USART
 - 2 SPIs
 - 1 bộ CAN interface
 - USB 2.0 full-speed interface
- Kiểm tra lỗi CRC và 96-bit ID



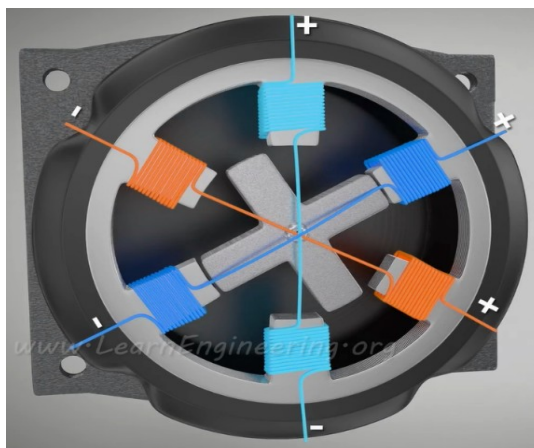
Hình 2.13 Cấu tạo của STM32 [12]

2.4.2. Giới thiệu về động cơ bước

Động cơ bước là loại động cơ DC không chổi than, số bước sẽ được phân chia như nhau trong mỗi vòng quay, từ đó ta có thể kiểm soát góc quay với độ chính xác cao. Ngày nay, động cơ bước được ứng dụng rộng rãi với nhiều ứng dụng như máy CNC, máy in 3D, các thiết bị y tế, v.v..

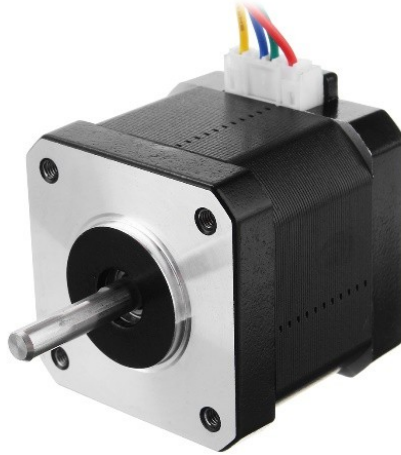
Động Cơ bước cấu tạo gồm 2 phần:

- + Rôto (phần động): các lá thép dát mỏng
- + Stato (phần tĩnh): có 6 cuộn dây được cấp với 3 nguồn DC riêng biệt

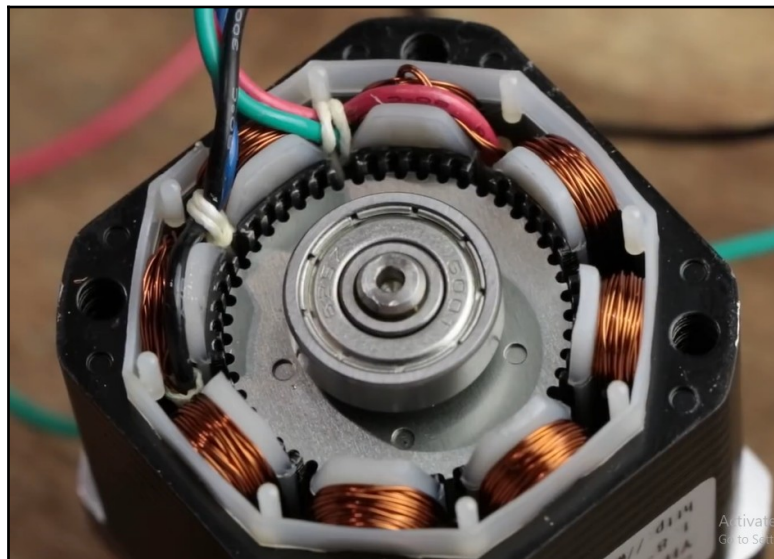


Hình 2.14 Cấu tạo cơ bản của động cơ bước [13]

Trong thực tế: loại động cơ bước linh hoạt và được sử dụng phổ biến trong các ứng dụng là động cơ Hybrid



Hình 2.15 Động cơ bước Hybrid [13]



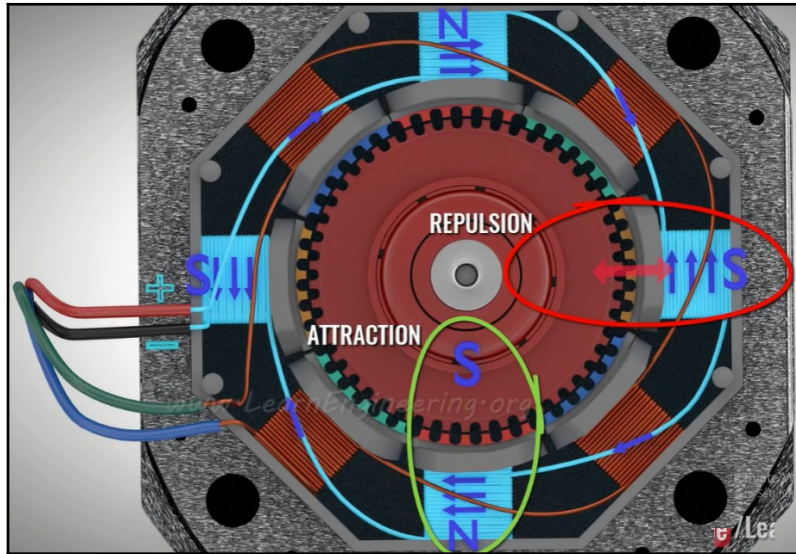
Hình 2.16 Cấu tạo động cơ bước Hybrid [14]

+ **Rôto:** được từ hóa bởi một lõi nam châm và bọc bên ngoài với các nắp thép có răng. Vì vậy, một đầu của Rôto trở thành cực bắc và đầu kia trở thành cực nam. Thông thường các nắp thép có 50 răng.

+ **Stato:** gồm 8 cuộn dây (thực chất là hai cuộn dây được cấp 2 nguồn riêng biệt), có 48 răng (ít hơn số răng của Rôto), được chia thành 4 cặp nhóm như hình minh họa.

Tính hoạt động chính xác của động cơ nằm ở chỗ sắp xếp các răng của các cặp cuộn dây Stato so với Rôto. Bộ màu xanh lá và xanh lam được sắp xếp thẳng hàng một nửa so với Rôto, bộ màu vàng hoàn toàn không thẳng hàng và còn bộ màu đỏ thì thẳng hàng với Rôto.

Khi cấp nguồn cho cuộn A thì tạo ra hai cặp cực từ tính ở bốn cuộn dây tác động lên Rôto. Các răng hút nhau thì thẳng hàng còn đẩy nhau thì không thẳng hàng.



Hình 2.17 Cấp nguồn cuộn dây A cho động cơ [14]

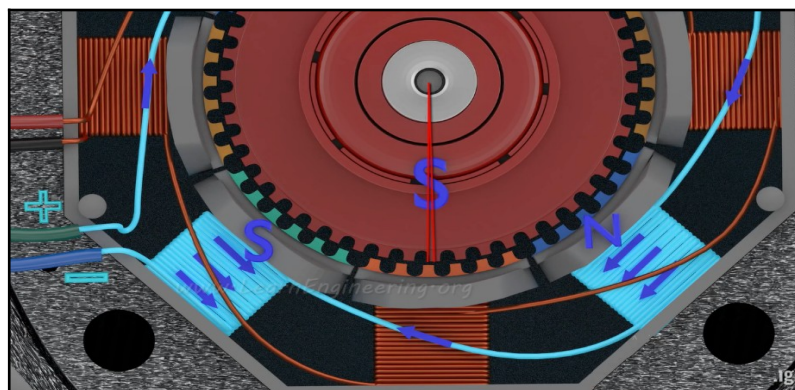
Tiếp theo, ngắt nguồn ở cuộn A và cấp nguồn cho cuộn B. Roto sẽ dịch chuyển một góc $\alpha = 1^\circ$ (tương ứng với một bước) và được tính toán như sau:

$$\alpha = \frac{1}{4} A_p \text{ (độ)}$$

Trong đó:

- + là góc quay (Step angle) của một bước.
- + là góc cao độ (Angular pitch) được tính như sau:

$$A_p = \frac{360}{50} \text{ (độ)}$$



Hình 2.18 Cấp nguồn cuộn dây B cho động cơ [14]

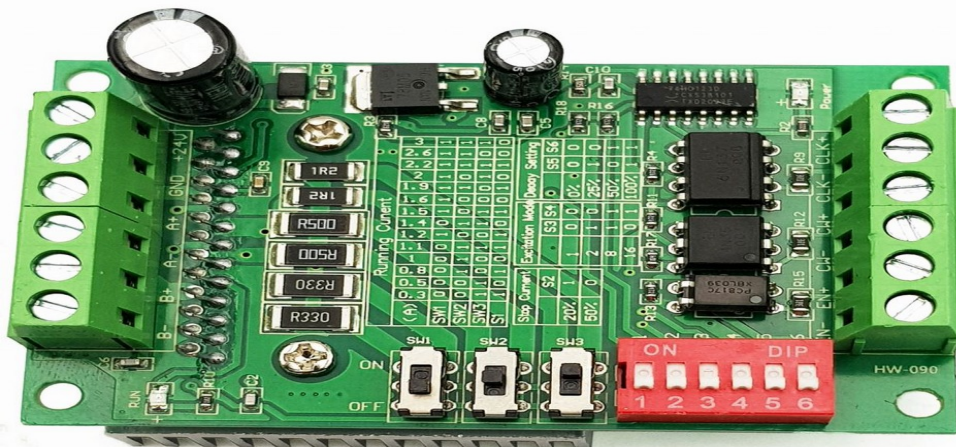
Để vi chỉnh góc quay của Rôto thì ta có các kỹ thuật điều chế xung để cho động cơ hoạt động theo những chế độ theo bảng sau:

Chế độ	Số bước/ vòng quay
Full step	200
Half step	400
1/4 step	800
1/8 step	1600
1/16 step	3200

Bảng 2.2 Chế độ hoạt động của động cơ bước [14]

2.4.3. Mạch điều khiển động cơ bước Tb6560

Mạch dùng để điều khiển các động cơ bước, điều khiển được các động cơ bước 2 pha với công suất tối đa 3A, có tích hợp đo quá dòng áp và ngõ vào có cách ly quang tốc độ cao. Được ứng dụng nhiều trong các máy CNC và các cơ khí.



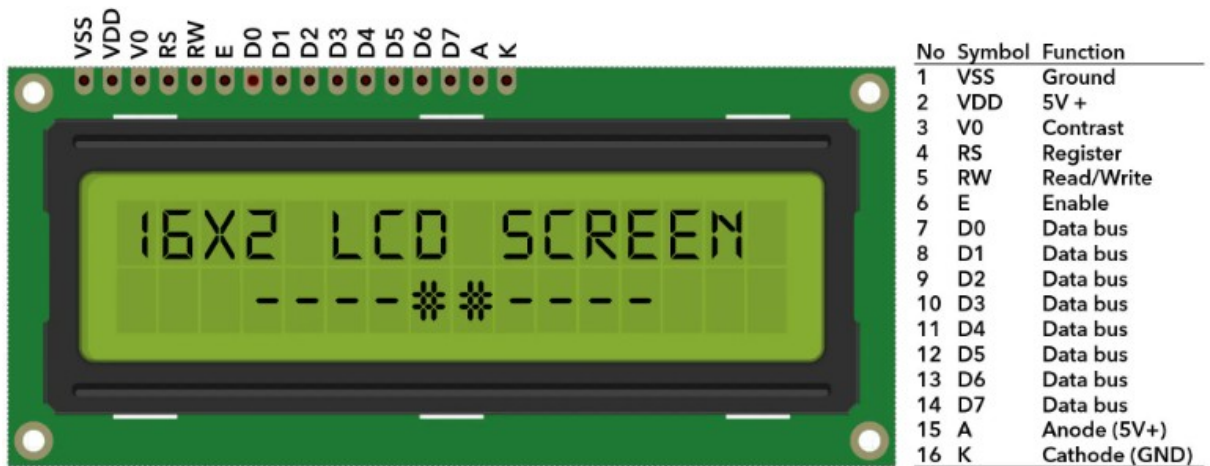
Hình 2.19 Mạch điều khiển động cơ bước Tb6560 [15]

Thông số	Giá trị
Điện áp vào	10~ 35V
Dòng cấp	Tối đa 3A
Kích thước	50*75*35mm

Bảng 2.3 Các thông số kỹ thuật của Tb6560 [15]

2.4.4. LCD 16x2

Màn hình LCD 16x2 là loại LCD có 2 hàng hiển thị, hiển thị tổng cộng 32 mã ASCII lên màn hình, nghĩa là hiển thị một hàng tối đa 16 ký tự.



Hình 2.20 Sơ đồ nối chân LCD 16x2 [16]

Thông số	Giá trị
Điện áp MAX	7V
Điện áp MIN	-0,3V
Điện áp ra mức thấp	<0.4V

Điện áp ra mức cao	>2.4V
Hoạt động ổn định	2.7-5.5V
Dòng điện cấp nguồn	350uA-600uA
Nhiệt độ hoạt động	-30°C đến +75°C

Bảng 2.4 Các thông số kỹ thuật của LCD 16x2 [16]

2.4.5. Mạch hạ áp LM2596

Mạch hạ áp LM2596 3A có khả năng hạ áp từ 30VDC xuống 1.5VDC mà vẫn đạt hiệu quả cao, được ứng dụng cho các việc như chia nguồn, hạ áp... cấp cho các thiết bị.



Hình 2.21 Mạch hạ áp LM2596 [17]

Thông số	Giá trị
----------	---------

Điện áp đầu vào	Từ 3V -30V
Điện áp đầu ra	Điều chỉnh trong khoảng 1.5V- 30V
Dòng đáp ứng tối đa	3A
Hiệu suất chuyển đổi	94%
Công suất	15W
Kích thước sản phẩm	45mm ×21mm ×15mm

Bảng 2.5 Thông số kỹ mạch hạ áp LM2596 [18]

2.4.6. Nguồn tổ ong 12V 10A

Nguồn tổ ong 12V 10A hay còn được gọi là bộ nguồn một chiều để chuyển đổi điện áp từ nguồn xoay chiều 220VAC thành nguồn một chiều 12VDC cung cấp cho các thiết bị hoạt động. Thường được sử dụng trong các thiết bị như tủ điện...



Bảng 2.6 Mạch nguồn tổ ong 12V-10A [18]

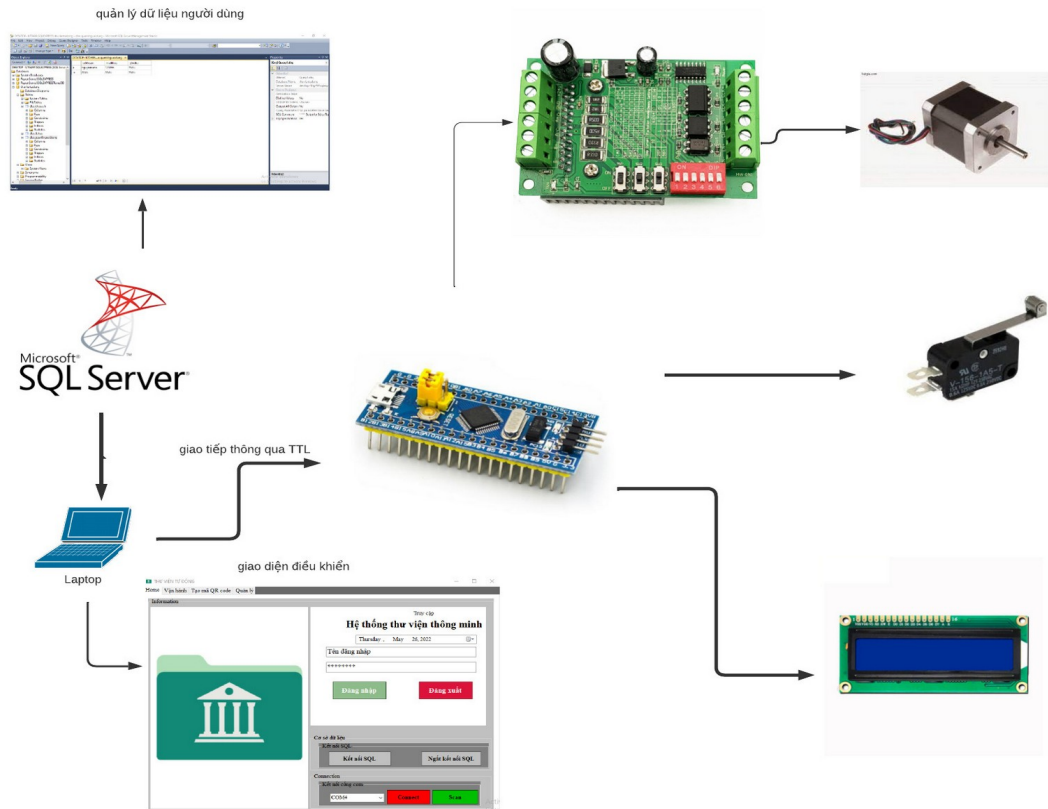
CHƯƠNG 3: THIẾT KẾ VÀ THI CÔNG

3.1. Sơ đồ khối hệ thống

3.1.1. Yêu cầu đề tài

Yêu cầu: Hệ thống sử dụng camera quét QR Code với giao diện trên nền tảng Window kết hợp với SQL Sever giúp lưu trữ dữ liệu thuận tiện và tự động hoặc điều khiển lấy- trả sách thông qua các động cơ step.

3.1.2. Thiết kế sơ đồ khối đề tài



Hình 3.1 Sơ đồ khối hệ thống

Mô hình có mục đích lưu trữ và quản lý dữ liệu người dùng trên SQL Sever thông qua giao diện Winform C#. Ta có thể tự động lấy và trả sách bằng việc quét QR Code từ giao diện Winform C# điều khiển các nút nhấn (vị trí, lấy sách, trả sách) tín hiệu sẽ được truyền thông qua USB TO TTL CH340 sang STM32. Từ đó vi điều khiển sẽ gửi tín hiệu xuống mạch điều khiển động cơ Tb6560 để các động cơ chạy đến vị trí mà mình cần lấy và trả sách, hiển thị các bước qua màn hình LCD.

Hệ thống gồm các khối cơ bản:

- + Khối nguồn
- + Khối điều khiển trung tâm
- + Khối giao diện

3.1.3. Chức năng từng khối

- Khối nguồn: Khối này sẽ cung cấp nguồn cho hệ thống.

Nguồn chính sử dụng cho mạch là nguồn tổ ong 12V – 10A, Nguồn này sẽ qua module giảm áp để cho về 5V để cấp cho STM32, LCD và công tắc hành trình. Nguồn 12V sẽ cung cấp cho mạch điều khiển động cơ Tb6560 để điều khiển động cơ bước.

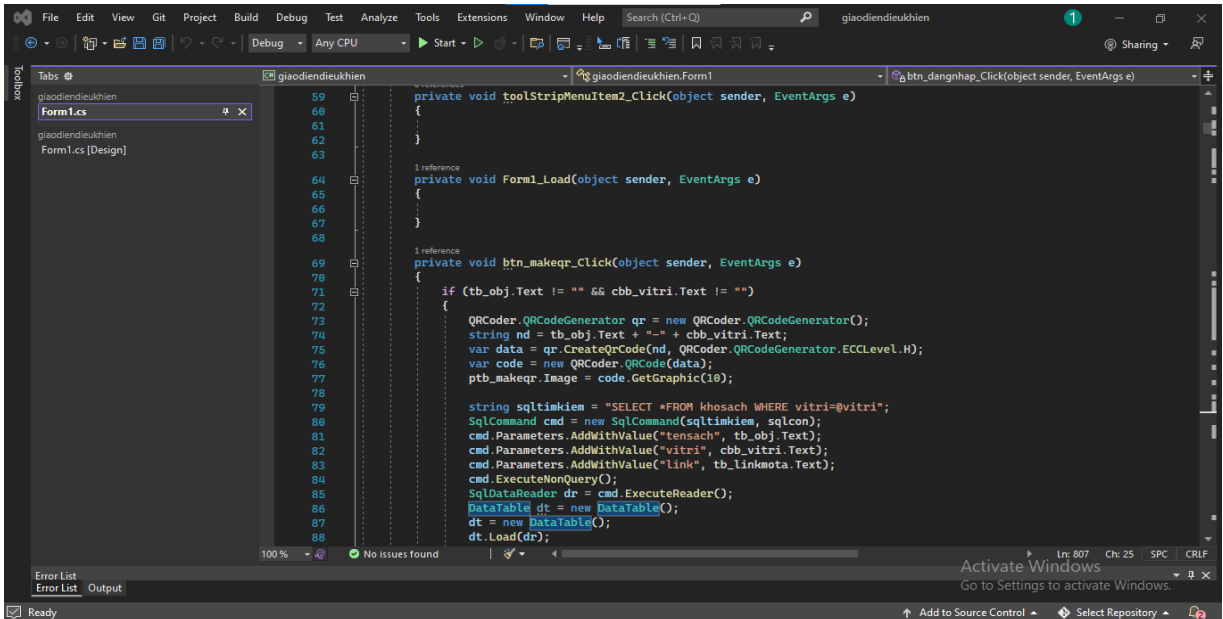


Hình 3.2 Nguồn tổ ong 12V-10A [15]

- Khối xử lý trung tâm: Khối này dùng để xử lý tín hiệu, truyền, nhận tín hiệu và giao tiếp để điều khiển thiết bị, sử dụng vi điều khiển STM32f103c8t6 có nhiệm vụ truyền tín hiệu điều khiển qua mạch điều khiển động cơ Tb6560 để điều khiển động cơ, nhận tín hiệu ngắt động cơ thông qua công tắc hành trình, hiển thị quá trình lên LCD.
- Khối ngoại vi: Khối này có chức năng hiển thị thông tin, ngắt động cơ. Khi động cơ hoạt động thì LCD sẽ hiển thị vị trí mà động cơ hướng đến và khi đến được vị trí yêu cầu thì công tắc hành trình sẽ ngắt động cơ.



Hình 3.3 Công tắc hành trình [16]



Hình 3.4 Giao diện làm việc của phần mềm Visual Studio

- Khối công suất: khối này có chức năng nhận tín hiệu điều khiển từ khối xử lý trung tâm sau đó cấp xung cho các động cơ bước.
- Khối giao diện: khối này có chức năng quản lý, hiển thị thông tin và điều khiển dữ liệu.

Khối giao diện có chức năng kết nối người dùng với khối điều khiển để truyền nhận giữ liệu và điều khiển thiết bị thông qua USB TO TTL CH340.

Nhóm sử dụng phần mềm Visual Studio để tạo giao diện Web.

3.2. Thiết kế đề tài

3.2.1. Thiết kế các động cơ bước

Các động cơ này có nhiệm vụ là nhận xung từ mạch điều khiển động cơ Tb6560 để chuyển động góc quay với mỗi động cơ bước sẽ tương ứng với các driver : X, Y, Z.

Ở đề tài nhóm em sử dụng động cơ bước 2 pha với 200 xung thì động cơ step quay được với mỗi xung động cơ quay 1.8 độ.

Sau khi đo đạc và tính toán các chiều dài của thanh ray chúng em thiết kế số vi bước cho đề tài như sau:

Vi bước X,Y: 8

Vi bước Z: 2

Công thức tính số xung:

$$\text{Số xung} = 200 \times \text{Vi bước}$$

Suy ra số xung X, Y = 8 x 200 = 1600 xung và Z = 2 x 200 = 400 xung sẽ quay được 1 vòng

3.2.2 Thiết kế vi điều khiển trung tâm

Ở trên thị trường có 2 dòng chip khá phổ biến đó là Ardiuno và STM32 .Tuy nhiên với các ứng dụng yêu cầu cao và phức tạp hơn với nền tảng dựa trên ARM như STM32 được sử dụng phổ biến hơn. STM32 phức tạp và giá cao hơn Arduino do đó linh hoạt hơn. Với hướng đề tài mới mẻ nên chúng em chọn STM32 làm vi điều khiển trung tâm.

3.2.3 Thiết kế mạch điều khiển động cơ bước

Có 2 mạch điều khiển động cơ bước thường được sử dụng : Tb6560-Tb6600, 2 mạch đều có các chức năng cơ bản tương đối giống nhau.

Tuy nhiên đối với Tb6600 có những ưu điểm vượt trội hơn :

- Dòng ra tối đa lên đến 4A/42VDC
- Có tích hợp đo dòng quá áp

Nhưng đối với đề tài này thì việc sử dụng Tb6560 lại thích hợp hơn vì:

- Nguồn ra 3A cung cấp đủ công suất cho động cơ giảm hao phí điện năng
- Nhỏ gọn dễ thiết kế mô hình
- Giá thành rẻ, dễ sử dụng

Từ đó ta chọn lựa linh kiện như sau:

- + Động cơ bước: Stepper motor
- + Công tắc thành trình
- + Mạch điều khiển động cơ bước: Tb6560
- + Mạch nguồn giảm áp: LM2596
- + Màn hình LCD: LCD16x2
- + Vi điều khiển trung tâm: STM32

Linh kiện	Mức tiêu thụ
STM32	36mA
Stepper Motor	2A
Tb6560	3A
LM2596	MAX 3A
LCD16x2	350uA-600uA

Bảng 3.1 Mức tiêu thụ điện năng của 1 số linh kiện

3.2.2. Thiết kế khối xử lý

a. Khối xử lý trung tâm

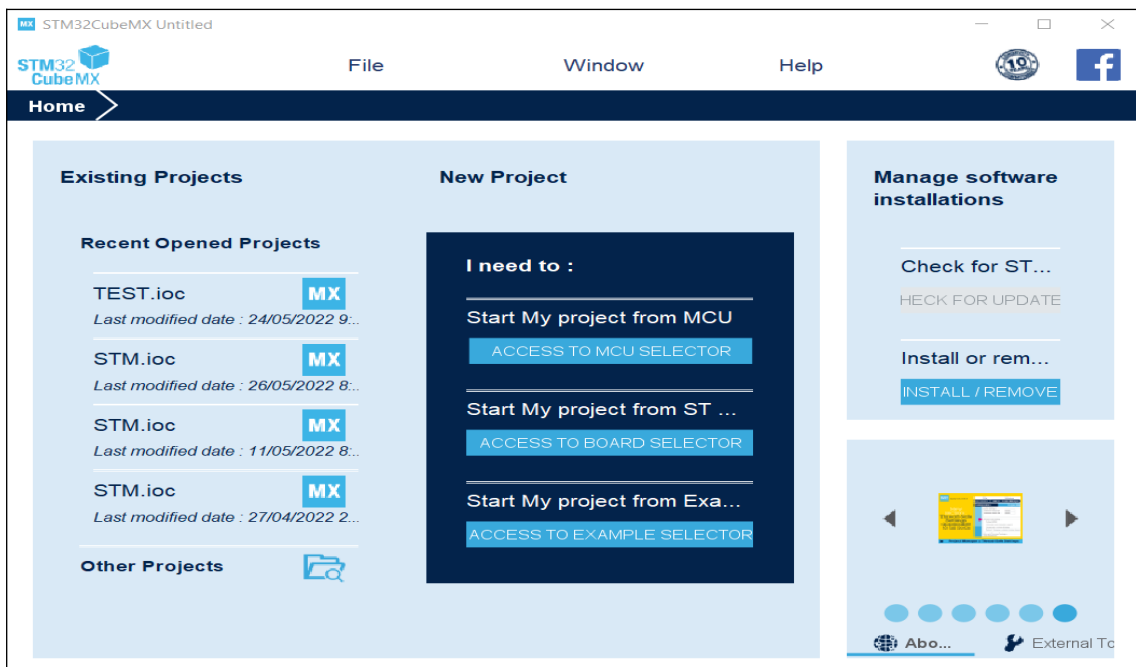
Nhóm em chọn vi điều khiển STM32 với cấu hình mạnh mẽ cho dù với phân khúc thấp nhất STM32F0x cũng có thể hoạt động lên tới 48Mhz, 64kB Flash, 16kB RAM, 8 bộ Timer 16 bit, 1 bộ Timer 32 bit, 10 bộ ADC 12 bit, 8 bộ USART, 2 bộ SPI, 2 bộ I2C.

Phần mềm lập trình cho vi điều khiển là phần mềm STM32CubeMX.

Chip này hoạt động lên đến 48MHz, có 32kB Flash, 4kB RAM, 5 bộ Timer 16 bit, 10 bộ ADC 12 bit, 1 bộ USART, 1 bộ I2C, 1 bộ SPI. Đây là mức giá rẻ nhất so với các dòng vi điều khiển có cấu hình tương đương.

STM32F103C8T6 có chi phí thấp cho sinh viên và người yêu thích nó để khám phá thiết bị ARM Cortex-M3. Thiết bị này có sẵn và cực kỳ tiềm năng. Cuối cùng, thiết kế của PCB nhỏ cho phép hàn chân vào các cạnh và cắm vào breadboard. Breadboard là board thuận tiện nhất để thực hiện các thí nghiệm.

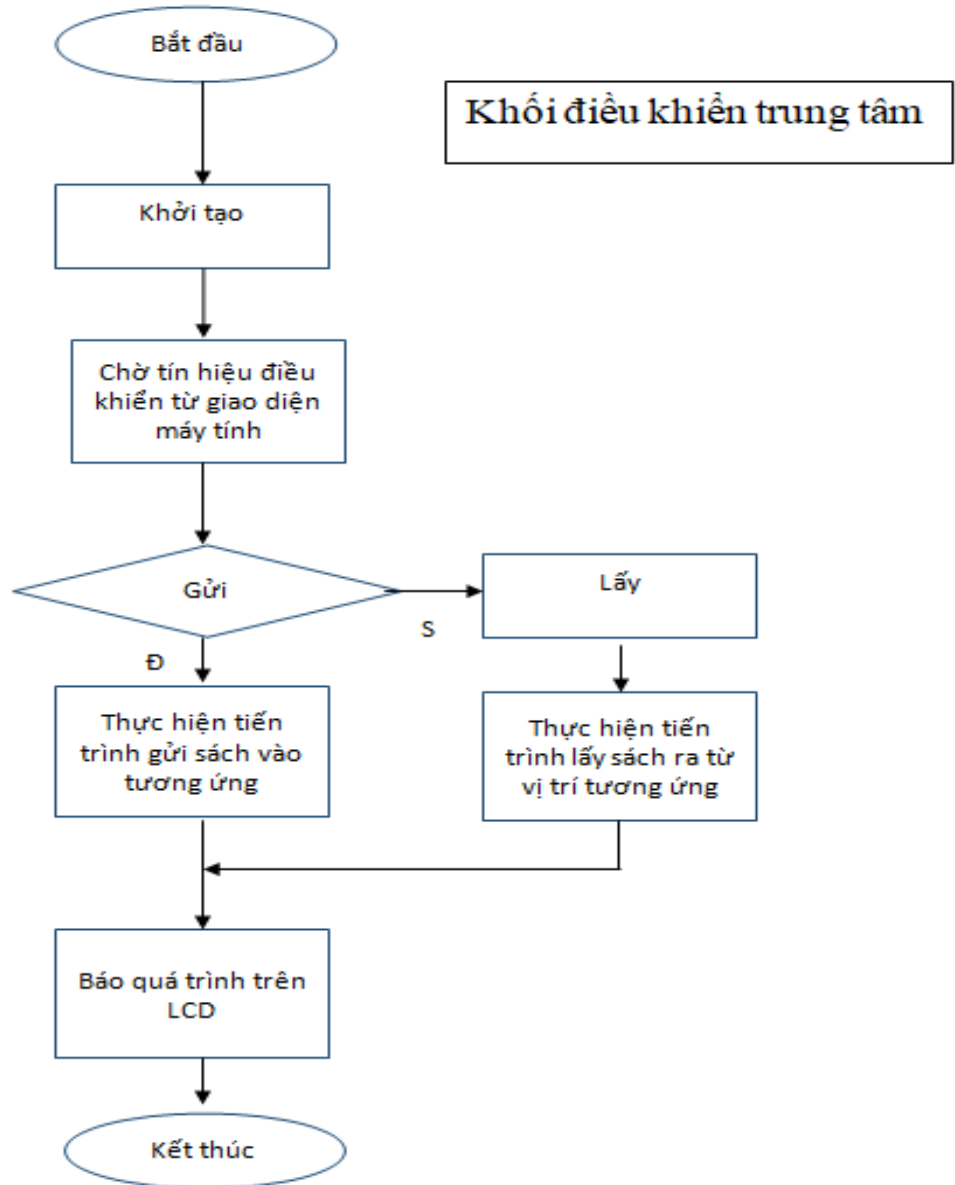
Tuy vậy, vẫn còn có một số hạn chế. Ví dụ như, bộ điều khiển USB và CAN không thể hoạt động cùng một lúc. Các thiết bị ngoại vi khác có thể khác với các chân I / O được sử dụng. Hầu hết sự khác nhau giữa các chân được quản lý thông qua cấu hình AFIO (Đầu ra chức năng thay thế), cho phép các chân khác nhau được sử dụng cho chức năng của thiết bị ngoại vi.



Hình 3.5 Cửa sổ làm việc của phần mềm STM32 CubeMX

Trong cấu hình ngoại vi, một số xung riêng biệt có thể được bật riêng để điều chỉnh mức sử dụng năng lượng

Thẻ nhớ flash được sử dụng ở 64K byte, nhưng có thể thấy rằng nó hỗ trợ 128K. Điều này cho phép các ứng dụng tốt được tận dụng vào thiết bị Mạch được thiết kế gồm 4 khối chính:



Hình 3.6 Lưu đồ thuật toán khối điều khiển trung tâm

Khi nhận tín hiệu lấy và trả sách từ khối giao diện, STM32 sẽ truyền tín hiệu điều khiển cho mạch điều khiển động cơ Tb6560, mạch điều khiển sẽ biến tín hiệu thành các xung cấp cho động cơ bước để thực hiện lấy và trả sách đồng thời thông báo quá trình qua LCD.

- **Khối nguồn:**

Đề tài nhóm dùng nguồn tổ ong 12V-10A cấp cho động cơ mạch điều khiển động cơ TB6560 và module LM2596 hạ áp từ 12V xuống 5V cung cấp cho vi điều khiển và các cổng vào ra.

- Khối vi điều khiển:

Sử dụng vi điều khiển chính là STM32F103C8T6.

Dùng để đọc các cách tín hiệu điều khiển từ giao diện máy tính.

STM cũng có nhiều ưu điểm đó là tương thích với nhiều hệ điều hành, chương trình lập trình đơn giản rõ ràng, dễ sử dụng, sử dụng mã nguồn mở và có thể kết hợp với nhiều module khác nhau.

- Khối công suất:

Dùng mạch điều khiển động cơ Tb6560 điều khiển 3 động cơ step.

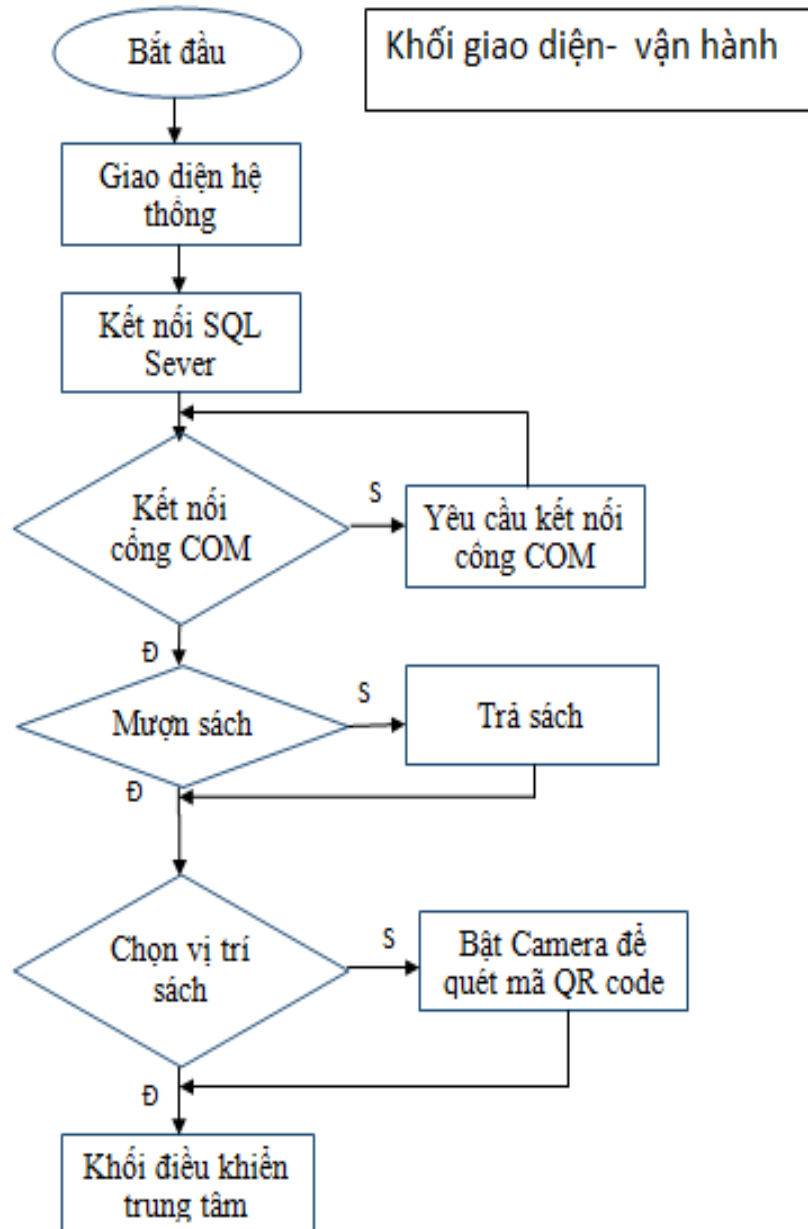
-Khối ngoại vi:

Gồm: LCD, Công tắc hành trình để hiển thị và ngắt động cơ.

- Khối giao Diện:

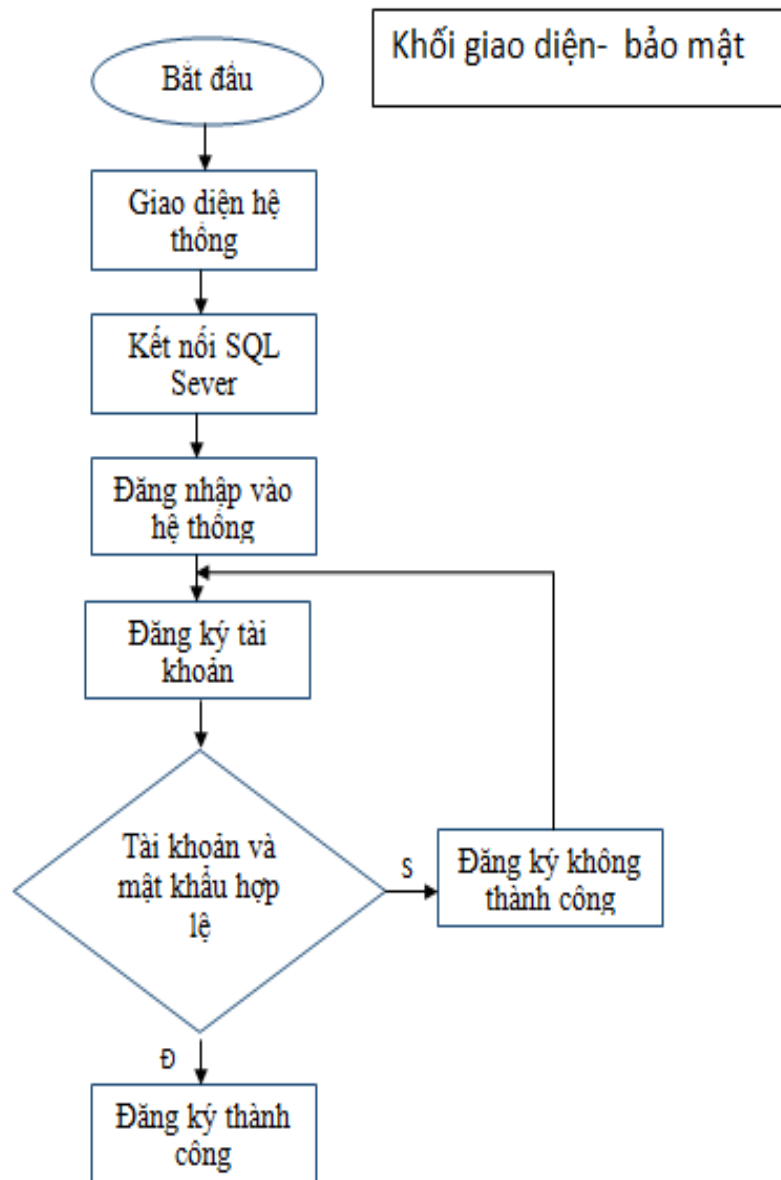
Khối giao diện có chức năng nhận diện QR code được quét từ camera, và quản lý thông tin người dùng, bên cạnh đó còn có chức năng truyền tín hiệu lấy-trả sách xuống mạch điều khiển STM32 thông qua USB TTL.

Chúng ta chỉ cần kết nối với SQL Sever và đăng nhập hệ thống người dùng thì dữ liệu sẽ được lưu trên SQL Sever và thực hiện việc lấy và trả sách.



Hình 3.7 Lưu đồ thuật toán khởi giao diện vận hành

Khi chúng ta khởi động chương trình cho phép vào giao diện hệ thống. Ta cần kết nối với phần mềm SQL Sever để quản lý cơ sở dữ liệu kế tiếp chúng ta cần kết nối cổng COM để có thể truyền tín hiệu xuống khối điều khiển tại giao diện hệ thống.

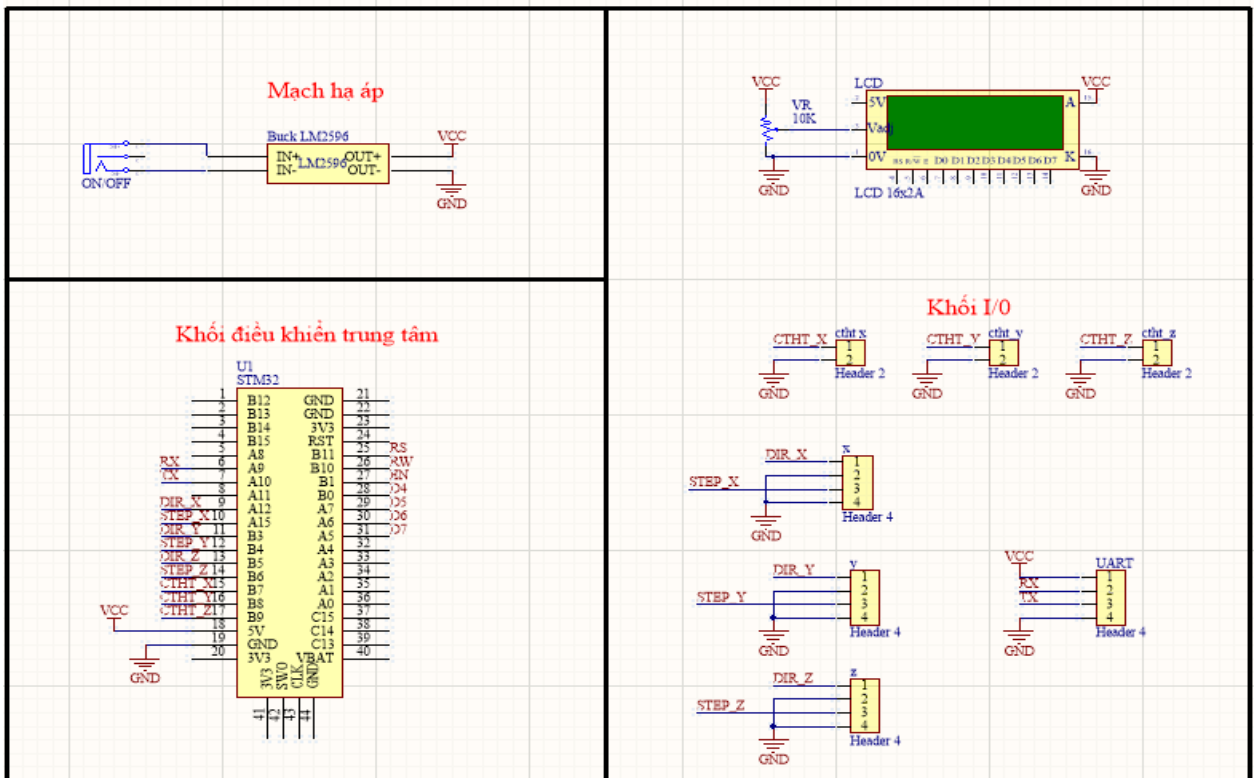


Hình 3.8 Lưu đồ thuật toán giao diện Winform C# bảo mật

Sau khi kết nối SQL Sever và cổng COM người quản lý sẽ đăng nhập vào tài khoản hoặc đăng ký tài khoản cho người dùng mới và yêu cầu người dùng không được đặt mật khẩu (trùng, có dấu), nếu tài khoản và mật khẩu không hợp lệ sẽ thông báo và quay về lại chỗ đăng ký.

- Khối i/o:

Gồm các chân I/O vi điều khiển



Hình 3.9 Sơ đồ nguyên lý STM32 nhận tín hiệu và điều khiển

3.2.3. Khối giao tiếp chuyển Đổi USB - TTL

Khối này có chức năng truyền/nhận dữ liệu giữa STM32 với máy tính thông qua UART bằng 2 chân TX và RX trên chip STM32

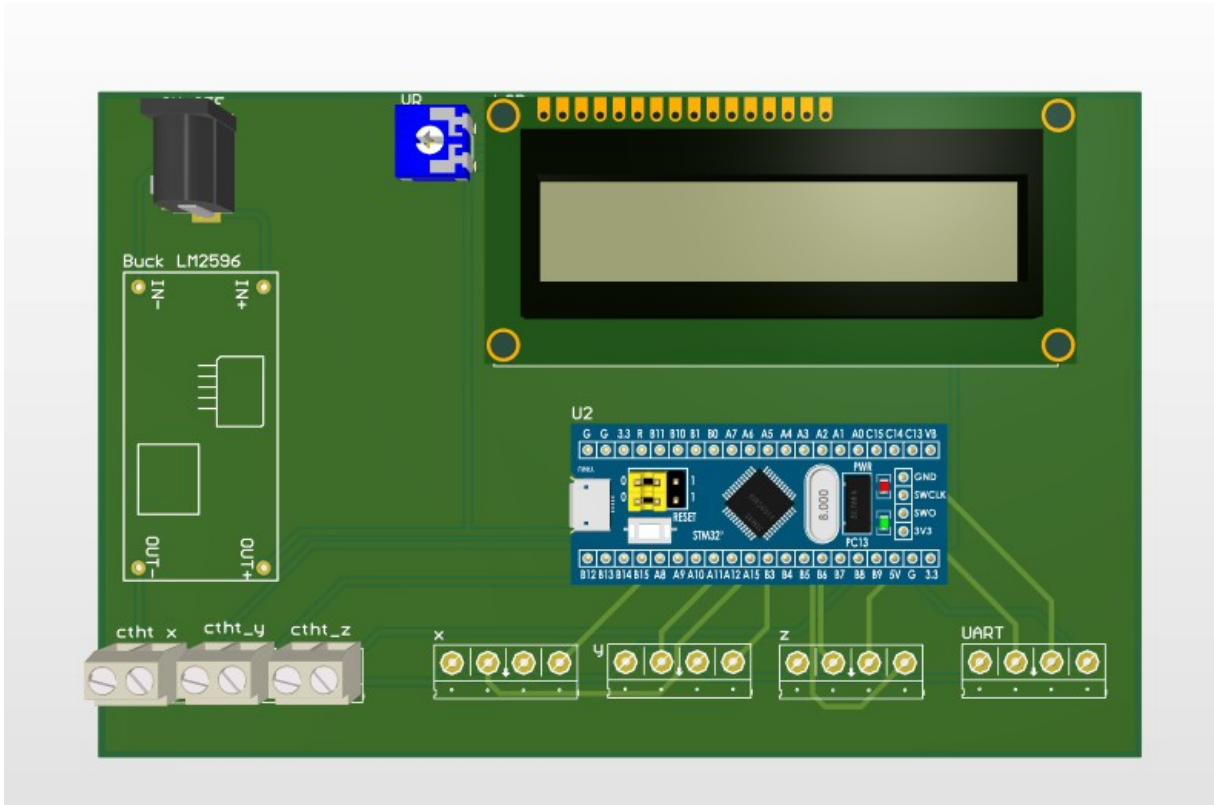


Hình 3.10 Hình ảnh thực tế UART USB to TTL CH340G [17]

3.3. Thi công sản phẩm

Ở phần thi công sản phẩm: nhóm em tự thiết kế các khối mạch in cho các khối của hệ thống và giới thiệu các quá trình làm việc thiết kế cũng như là thi công sản phẩm. Bên cạnh đó cung có phần demo sản phẩm.

3.3.1. Mạch 3D

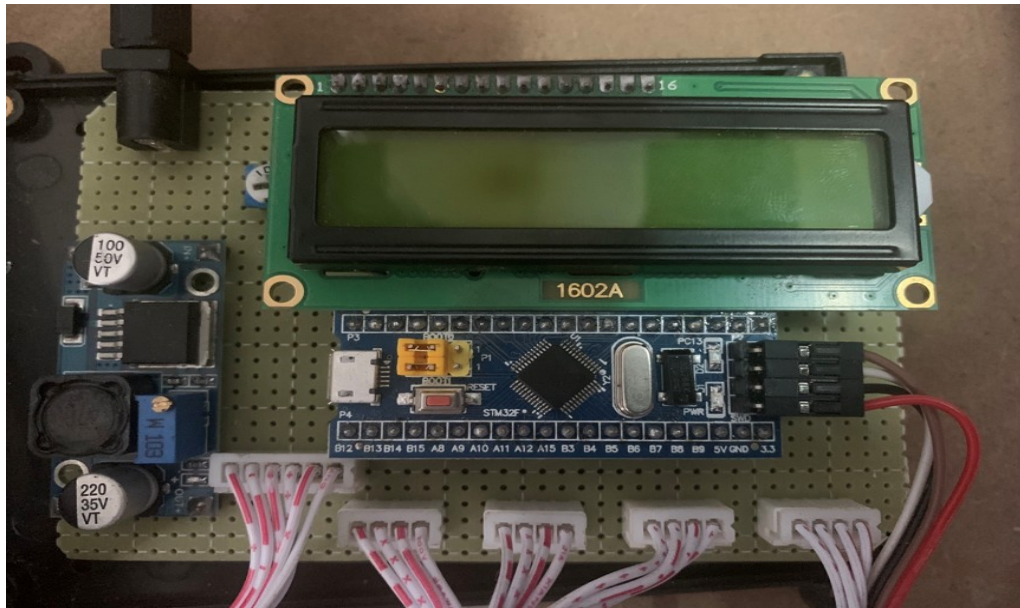


Hình 3.11 Mạch 3D khối xử lý

Trong hình 3.11 là sơ đồ mô phỏng của mạch điều khiển chính được vẽ bằng phần mềm Altium Designer dựa trên sơ đồ nguyên lý mà nhóm đưa ra. Các linh kiện và vị trí sẽ được nhóm bố trí và sắp xếp như trong hình.

3.3.2. Mạch sau khi thi công

Nhóm hàn linh kiện và lắp ráp mạch của khối giao tiếp. Tiến hành đo và kiểm tra nguồn vào và ra rồi chỉnh lại sao cho phù hợp.



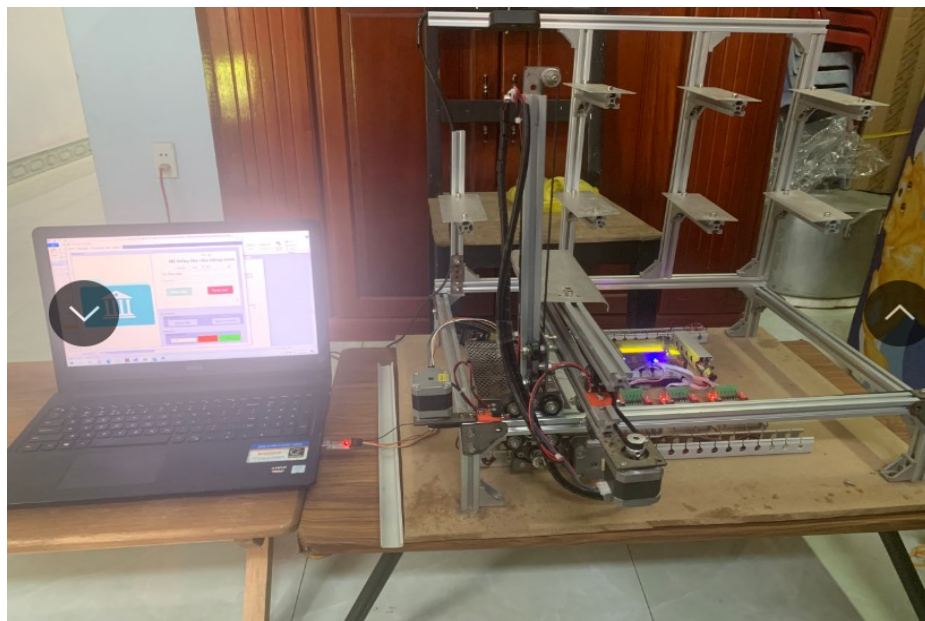
Hình 3.12 Mạch 1 sau khi thi công

Nhóm hàn linh kiện và lắp ráp mạch của khối giao tiếp. Tiến hành đo và kiểm tra nguồn vào và ra rồi chỉnh lại sao cho phù hợp với công suất của mạch

3.4. Mô hình sản phẩm và quá trình hoạt động

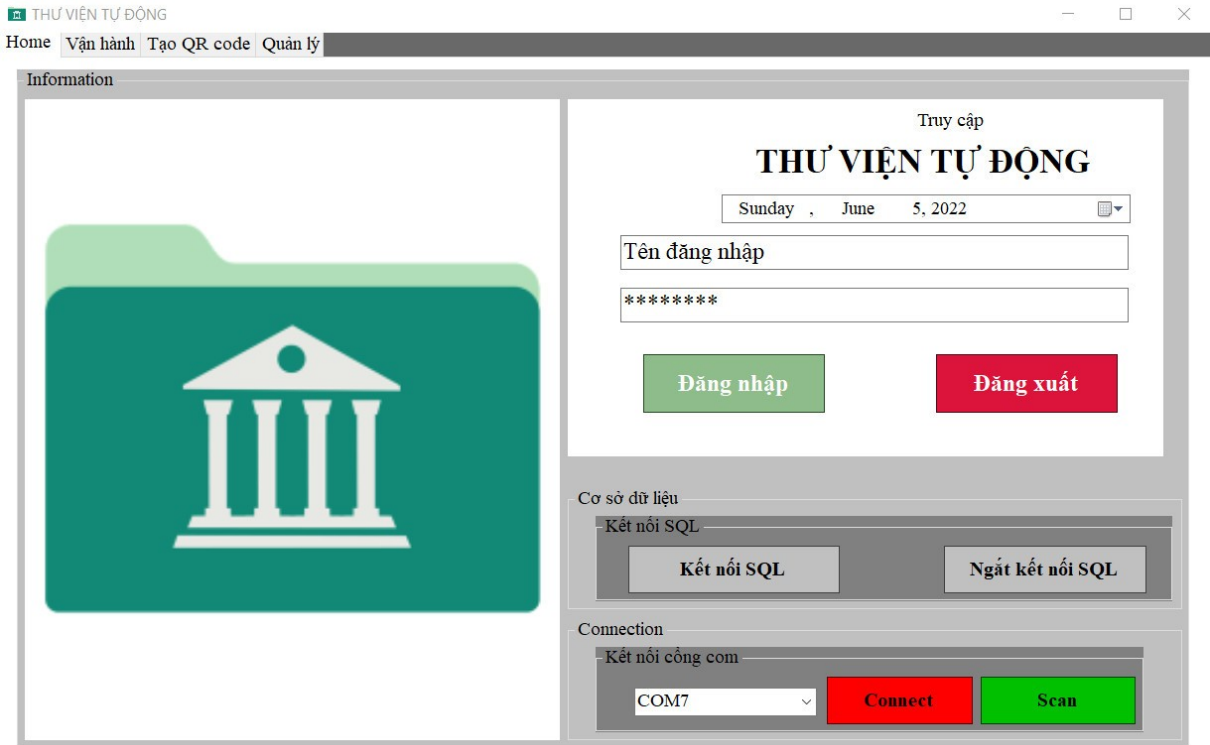
3.4.1. Mô hình sản phẩm

Ở đề tài này, nhóm chúng em đã tìm hiểu và sử dụng phần mềm SQL Sever để quản lý dữ liệu và lập trình bằng phần mềm Visual studio và phần mềm STM32 cubeMX để mô phỏng và set up cho STM32.

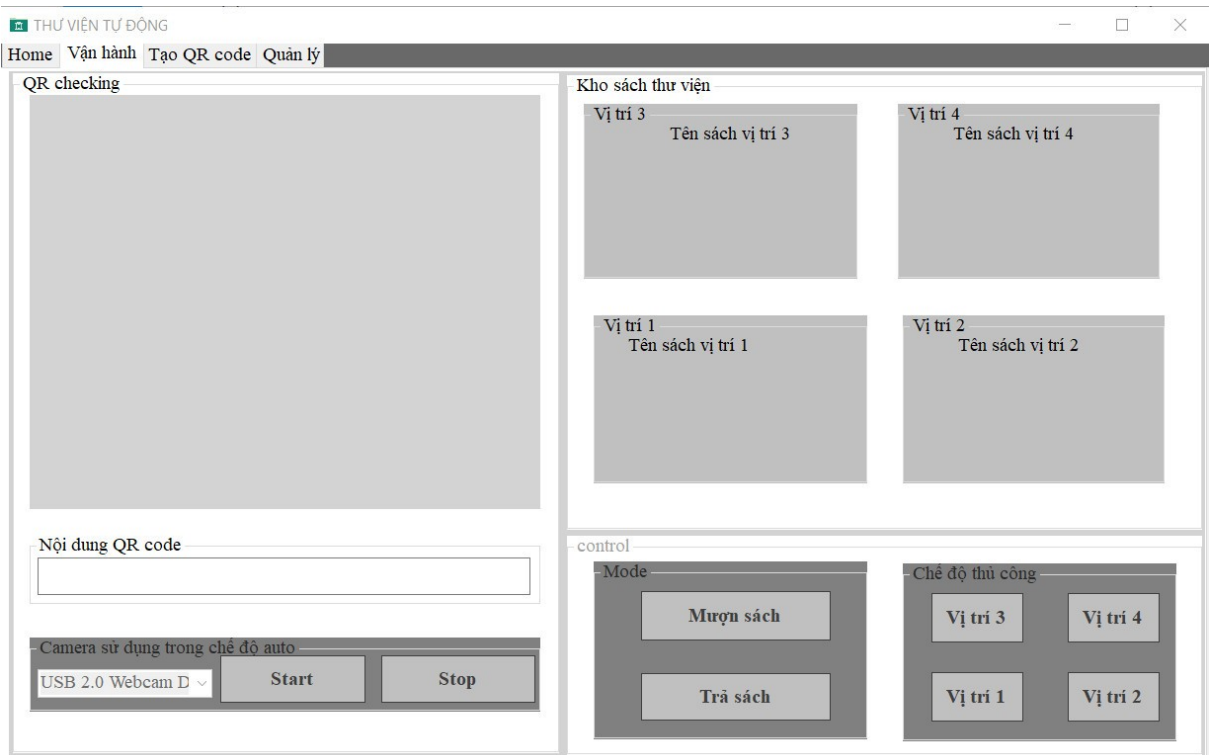


Hình 3.13 Hình ảnh mô hình sản phẩm

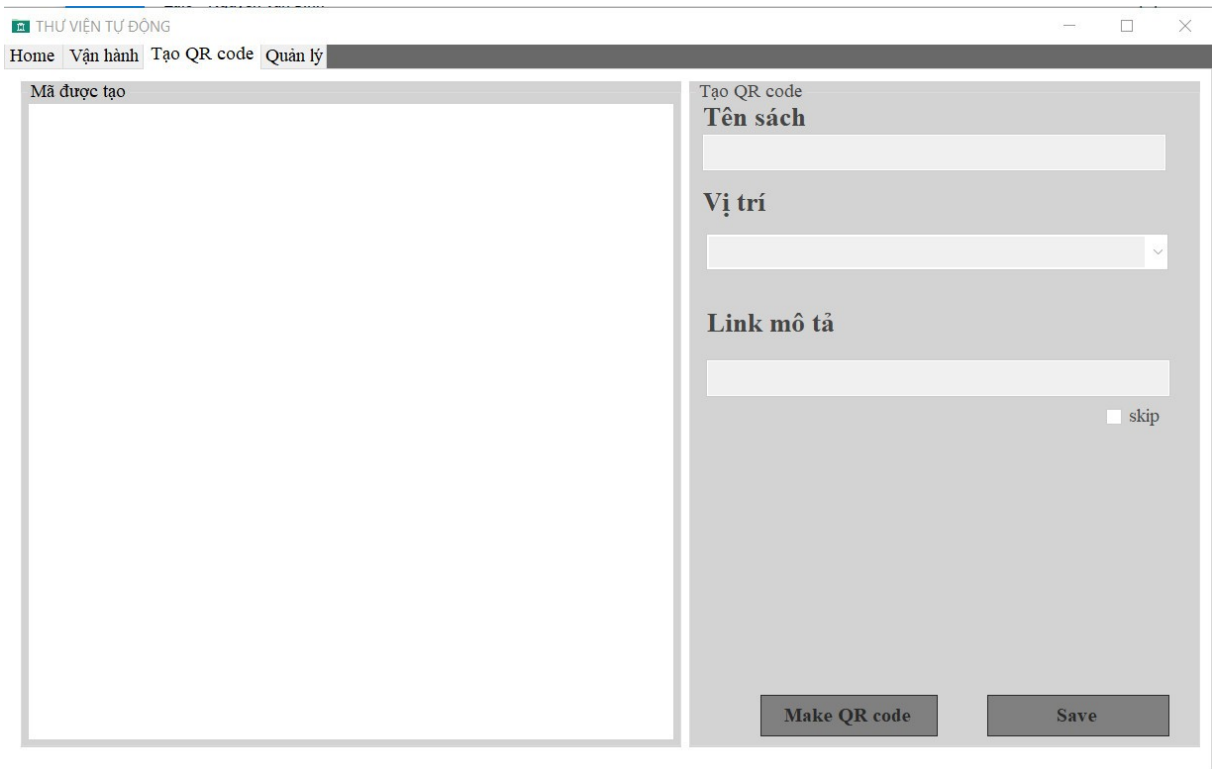
Thiết kế và thi công hệ thống quản lý mượn trả sách thư viện ứng dụng quét QR Code



Hình 3.14 Giao diện Home



Hình 3.15 Giao diện vận hành



Hình 3.16 Giao diện tạo QR Code



Hình 3.17 Giao diện quản lý

3.4.2. Quá trình hoạt động của hệ thống

❖ Xử lý trung tâm

Sau khi cấp nguồn, STM32 sẽ nhận tín hiệu điều khiển từ giao diện thông qua USB TO TTL CH340 truyền tín hiệu xuống các mạch điều khiển xuất xung cho các động cơ bước (step)

❖ Giao diện

Sau khởi động nguồn, giao diện sẽ kết nối với cổng COM trên máy tính thông qua USB TO TTL CH340

❖ SQL Sever

Quản lý các dữ liệu của giao diện hệ thống

Kết quả đạt được:

- Tiếp cận, tìm hiểu, hoàn thiện sản phẩm
- Truyền/nhận dữ liệu từ máy tính sang STM32 qua cổng COM ảo
- Sử dụng được công nghệ, xây dựng được hệ thống điều khiển và giám sát
- Dữ liệu gửi lên hệ thống là thời gian thực
- Ứng dụng sản phẩm dễ dàng, dễ tiếp cận, dễ sử dụng
- Nâng cao khả năng nghiên cứu, học hỏi vào phát triển

3.5. Hướng dẫn sử dụng phần mềm

Sau khi mở giao diện, chúng ta đăng nhập vào tài khoản user đã tạo trên cơ sở dữ liệu để đăng nhập và đăng ký thêm tài khoản user.

➤ Tab Home

- Khi đăng nhập hợp lệ sẽ được đăng ký mới tài khoản user mới
- Ở đây khi ta bấm đăng kí thì sẽ tạo được một tài khoản user mới và đã liên kết với cơ sở dữ liệu.
- Tài khoản user được đưa cho người dùng
- Sau khi đăng nhập thành công người dùng sẽ cần kết nối với cơ sở dữ liệu và cổng COM

➤ Tab tạo QR Code

- Người dùng cần điền tên sách, vị trí và link mô tả sau đó nhấn Make QR Code để tạo QR Code sử dụng ở sách đó và tiến hành lưu lại.

➤ Tab vận hành

- Hiện thị vị trí cũng như tên sách mà người dùng đã tạo trước đó
- Người dùng nhấn vào phần control để tiến hành mượn sách hoặc trả sách với 2 chế độ là thủ công và sử dụng QR Code:
 - + Chế độ thủ công người dùng cần nhấn vào vị trí muốn lấy sách
 - + Chế độ quét QR code ta tiến hành Start để bật camera phục vụ việc quét QR code đã lưu trước đó.

➤ Tab quản lý

- Ở đây ta thấy được một giao diện với 2 thông số là kho sách và lịch sử vận hành

KẾT LUẬN

Kết luận:

Qua thời gian thực hiện đề tài từ ngày 22/02/2022 đến ngày 29/05/2022 nhóm đã hoàn thành đồ án đúng thời gian dự kiến để báo cáo trước hội đồng. Nhóm thực hiện đề tài đã tạo ra một sản phẩm cụ thể, đó là:

- Tạo nên một giao diện điều khiển hoặc quét QR Code thực hiện việc lấy và trả sách một cách tự động
- Sử dụng STM32 tạo tính mới mẻ cho đề tài thay vì sử dụng Arduino

Sau thời gian nghiên cứu và thi công, nhóm đã thực hiện được công việc như xây dựng phần cứng, viết giao diện quản lý, xây dựng cơ sở dữ liệu cho hệ thống. Hiểu được cách lập trình STM32, hiểu biết cơ bản về cơ chế và cách thức lập trình C# và SQL Sever. Từ những yêu cầu đặt ra khi nghiên cứu, nhóm đã đạt được một số yêu cầu sau:

- Hoàn thiện được mô hình, tốc độ đọc QR code nhanh chóng và chính xác cao
- Phần mềm hoàn thiện và hoạt động ổn định
- Hiểu biết cơ bản về lập trình C# tạo giao diện hệ thống và các xây dựng cơ sở dữ liệu trên SQL Sever
- Hiểu biết về mạch điều khiển động cơ TB6560, động cơ Step và các ngoại vi khác
- Với STM32 là vi điều khiển đòi hỏi có kiến thức lập trình chuyên sâu hơn so với Arduino và cũng còn mới mẻ đối với sinh viên, đây là điểm đặt biệt của đề tài và cũng là vấn đề khá quan trọng

Ưu điểm của sản phẩm:

- Hệ thống hoạt động ổn định và chính xác với sai số không đáng kể
- Thiết kế mô hình đơn giản và thẩm mỹ
- Tiết kiệm được chi phí
- Cần thiết và có tính ứng dụng cao
- Điều khiển dễ dàng

Nhược điểm của sản phẩm:

- Do thời gian tiếp cận và hoàn thành còn chưa nhiều nên hệ thống còn chưa được tối ưu nhất
- Cần nâng cấp thêm kết nối internet

Nhận xét và đánh giá:

Nhận xét:

- Đề tài nhằm mục đích giúp nâng cao đời sống từ việc công nghệ hoá – hiện đại hoá, với những chức năng này ta có thể kết hợp với những tính năng khác như thuê xe đạp, bán hàng tự động ...

Đánh giá:

- Tạo ra được sản phẩm có tính ứng dụng
- Hệ thống có tốc độ phản hồi ổn định
- Tối ưu được vấn đề tiện lợi và bảo mật
- Khó tiếp cận đối với người chưa có smartphone

Hướng phát triển sản phẩm:

- Nhóm sẽ tối ưu sản phẩm để nó có thể hoạt động chính xác hơn lúc nào động cơ gặp các vấn đề khác nhau
- Nhóm cũng sẽ hoàn thiện và nâng cấp sản phẩm hơn để sản phẩm có nhiều chức năng mới hơn. Tối ưu sản phẩm để nó càng ngày càng nhỏ gọn, tiện ích và hiệu quả hơn

TÀI LIỆU THAM KHẢO

- [1]. Tiêu chuẩn TCVN 10274:2013 Thuật ngữ, định nghĩa về hoạt động thư viện
<https://luatvietnam.vn/van-hoa/tieu-chuan-tcvn-10274-2013-thuat-ngu-dinh-nghia-ve-hoat-dong-thu-vien-159872-d3.html>
- [2]. Đẩy mạnh ứng dụng khoa học công nghệ trong hoạt động thư viện ở Việt Nam
<https://www.hcmcpv.org.vn/tin-tuc/day-manh-ung-dung-khoa-hoc-cong-nghe-trong-hoat-dong-thu-vien-o-viet-nam-1491864890>
- [3]. York University Learning Commons / Levitt Goodman Architects
<https://www.archdaily.com/271485/york-university-learning-commons-levitt-goodman-architects>
- [4]. So sánh RFID-BAR Code-QR Code – NFC
<https://itgtechnology.vn/so-sanh-rfid-bar-code-qr-code-nfc/>
- [5]. Mã QR - Wikipedia tiếng việt
https://vi.wikipedia.org/wiki/M%C3%A3_QR
- [6]. Những kiến thức cần biết về QR Code
<https://brandinfo.biz/marketing/nhung-kien-thuc-can-biet-ve-qr-code/144-238-281.html>
- [7]. Ngôn ngữ lập trình C# và những lý do “quyến rũ chết người” khiến bạn phải học chúng ngay lập tức!
<https://mindx.edu.vn/blog/ngon-ngu-lap-trinh-csharp>
- [8]. Ebook Giáo trình SQL – Trần Nguyên Phong
<https://cuongquach.com/ebook-giao-trinh-sql-tran-nguyen-phong-pdf.html>
- [9]. SQL Server là gì? Mục đích của việc sử dụng SQL Server
<https://itnavi.com.vn/blog/sql-server/>
- [10]. Hướng dẫn cài đặt STM32 CubeMx và Keil C
<https://deviot.vn/tutorials/stm32f1.23165131/huong-dan-cai-dat-stm32-cubemx-va-keil-c.04048071>
- [11]. Visual studio là gì? Giới thiệu các tính năng phần mềm Visual Studio

<https://visualcpp.net/visual-studio-la-gi-gioi-thieu-phan-mem-visual-studio/>

[12]. Lập trình STM32 từ A tới Z

<https://khuenguyencreator.com/lap-trinh-stm32-tu-a-toi-z/>

[13]. ĐỘNG CƠ BƯỚC LÀ GÌ ? CẤU TẠO, PHÂN LOẠI VÀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN

<https://www.ngananhphat.com/vn/dong-co-buoc-la-gi-cau-tao-phan-loai-va-cac-phuong-phap-dieu-khien.html>

[14]. Cấu tạo và nguyên lý hoạt động của động cơ bước

<https://www.youtube.com/watch?v=opkEJ-n7tIU>

[15]. Mạch driver động cơ bước TB6560 3A

<https://icdayroi.com/mach-driver-dong-co-buoc-tb6560-3a>

[16]. Giải thích hoạt động LCD và giới thiệu module LCD 16x2

<https://www.youtube.com/watch?v=OL6uUfmXvw0>

[17]. Mạch Giảm Áp DC LM2596 3A

<https://nshopvn.com/product/mach-giam-ap-dc-lm2596-3a/>

[18]. Nguồn tổ ong 12V 10A Power Supply

<https://nshopvn.com/product/nguon-to-ong-12v-10a-power-supply/>

[19]. Công tắc hành trình là gì ? cấu tạo & nguyên lý chi tiết

<https://thietbidientt.com/cong-tac-hanh-trinh-la-gi/>

[20]. USB TO COM/TTL Chip CH340G

<https://lkcg.vn/usb-to-com-ttl-chip-ch340g>

[21].Đề tài: Nghiên cứu QR Code và Ứng dụng- Nguyễn Thị Quỳnh Chi

https://tailieuxanh.com/vn/dlID1448415_de-tai-nghien-cuu-qr-code-va-ung-dung.html

[22]. Lập trình căn bản ARM CORTEX M3 STM32F103C8T6-Nguyễn Ngọc Hà

<https://www.dientuhello.com/tai-lieu-lap-trinh-can-ban-arm-cortex-m3-stm32f103c8t6>

[23]. Giáo trình C# PDF bản update mới nhất (Cập nhật 2021)

<https://codegym.vn/blog/2021/01/21/giao-trinh-c-pdf/>

[24]. Thiết kế giao diện ứng dụng Lập lịch C# Winform

<https://howkteam.vn/course/lap-trinh-ung-dung-lap-lich-voi-c-winform/thiet-ke-giao-dien-ung-dung-lap-lich-c-winform-1137>

[25]. Kết nối cơ sở dữ liệu SQL Server trong C#

<https://openplanning.net/10511/ket-noi-co-so-du-lieu-sql-server-trong-csharp>

[26]. Stm32 PWM điều khiển tốc độ động cơ DC

<https://www.youtube.com/watch?v=HRBSNB3YofA&list=PLsrx4GB70yy7Y5frssMfEW3hHhrljUdm3&index=26>

PHỤ LỤC

Code khối xử lý trung tâm:

```
#include "main.h"
#include "stdio.h"
#include "stdlib.h"

UART_HandleTypeDef huart1;
char temp[20];
char Rx_data[2];
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_USART1_UART_Init(void);
int vitri_x=0,vitri_y=0,vitri_z=0;
int n1,n2,n3;
/*****Ctr giao tiep LCD 16x2 4bit*****/
void delay_us(unsigned int t){
    unsigned int i;
    for(i=0;i<t;i++);
}
void LCD_Enable(void){
    HAL_GPIO_WritePin(LCD_EN_GPIO_Port,LCD_EN_Pin,1);
    HAL_Delay(1);
    HAL_GPIO_WritePin(LCD_EN_GPIO_Port,LCD_EN_Pin,0);
    HAL_Delay(1);
}
void LCD_Send4Bit(unsigned char Data){
    HAL_GPIO_WritePin(LCD_D4_GPIO_Port,LCD_D4_Pin,Data & 0x01);
    HAL_GPIO_WritePin(LCD_D5_GPIO_Port,LCD_D5_Pin,
(Data>>1)&0x01);
    HAL_GPIO_WritePin(LCD_D6_GPIO_Port,LCD_D6_Pin,
(Data>>2)&0x01);
    HAL_GPIO_WritePin(LCD_D7_GPIO_Port,LCD_D7_Pin,
(Data>>3)&0x01);
}
void LCD_SendCommand(unsigned char command){
```

```
LCD_Send4Bit(command >>4);/* Gui 4 bit cao */
LCD_Enable();
LCD_Send4Bit(command); /* Gui 4 bit thap*/
LCD_Enable();
}
void LCD_Clear(){
    LCD_SendCommand(0x01);
    HAL_Delay(1);
}
void LCD_Init(){
    HAL_GPIO_WritePin(LCD_RS_GPIO_Port,LCD_RS_Pin,0);

    HAL_GPIO_WritePin(LCD_RW_GPIO_Port,LCD_RW_Pin,0);
    LCD_SendCommand( 0x02 );// chuyen cn tro ve dau mang
    LCD_SendCommand( 0x28 );// giao thuc 4 bit, hien thi 2 hang, ki tu 5x8
    LCD_SendCommand( 0x0C); // cho phép hien thi man hinh
    LCD_SendCommand( 0x06 );// tang ID, khong dich khung hinh
    LCD_SendCommand( 0x01); // xoa toan bo khung hinh
}
void LCD_Gotoxy(unsigned char x, unsigned char y){
    unsigned char address;
    if(!y)address=(0x80+x);
    else address=(0xc0+x);
    HAL_Delay(1);
    LCD_SendCommand(address);
    HAL_Delay(1);
}
void LCD_PutChar(unsigned char Data){//Ham Gui 1 Ki Tu
    HAL_GPIO_WritePin(LCD_RS_GPIO_Port,LCD_RS_Pin,1);
    LCD_SendCommand(Data);
    HAL_GPIO_WritePin(LCD_RS_GPIO_Port,LCD_RS_Pin,0);
}
void LCD_Puts (char *s){//Ham gui 1 chuoai ky tu
    while (*s){
        LCD_PutChar(*s);
        s++;
    }
}
```

```
void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart){
    if(huart-> Instance == USART1){
        HAL_UART_Receive_IT(&huart1,(uint8_t*)Rx_data,1);
    }
}
void go_home()
{
    HAL_GPIO_WritePin(DIR_X_GPIO_Port,DIR_X_Pin,0);
    HAL_GPIO_WritePin(DIR_Y_GPIO_Port,DIR_Y_Pin,0);
    HAL_GPIO_WritePin(DIR_Z_GPIO_Port,DIR_Z_Pin,1);
    LCD_Gotoxy(0,0);
    sprintf(temp," Go Y home ");
    LCD_Puts(temp);
    while(1)
    {
        if(HAL_GPIO_ReadPin(ctht_y_GPIO_Port,ctht_y_Pin)==0){
            if(n1<100){
                n1++;
            }
            if(n1==10){
                break;
            }
        }
        else{
            n1=0;
        }
        HAL_GPIO_WritePin(STEP_Y_GPIO_Port,STEP_Y_Pin,1);
        delay_us(1000);
        HAL_GPIO_WritePin(STEP_Y_GPIO_Port,STEP_Y_Pin,0);
        delay_us(1000);
    }
    LCD_Gotoxy(0,0);
    sprintf(temp," Go Z home ");
    LCD_Puts(temp);
    while(1)
    {
        if(HAL_GPIO_ReadPin(ctht_z_GPIO_Port,ctht_z_Pin)==0){
            if(n2<100){
                n2++;
            }
        }
    }
}
```



```
        }
        if(n2==10){
            break;
        }
    }
    else{
        n2=0;
    }
    HAL_GPIO_WritePin(STEP_Z_GPIO_Port,STEP_Z_Pin,1);
    delay_us(1000);
    HAL_GPIO_WritePin(STEP_Z_GPIO_Port,STEP_Z_Pin,0);
    delay_us(1000);
}
LCD_Gotoxy(0,0);
sprintf(temp," Go X home ");
LCD_Puts(temp);
while(1)
{
    if(HAL_GPIO_ReadPin(ctht_x_GPIO_Port,ctht_x_Pin)==0){
        if(n3<100){
            n3++;
        }
        if(n3==10){
            break;
        }
    }
    else{
        n3=0;
    }
    HAL_GPIO_WritePin(STEP_X_GPIO_Port,STEP_X_Pin,1);
    delay_us(1000);
    HAL_GPIO_WritePin(STEP_X_GPIO_Port,STEP_X_Pin,0);
    delay_us(1000);
}
LCD_Gotoxy(0,0);
sprintf(temp," RUN OK ");
LCD_Puts(temp);
vitri_x=0;
vitri_y=0;
```

```
    vitri_z=0;
}
void run_step(int x,int y,int z){
    while(1){
        // x chieu ra xa vitri home là tang
        // y chieu đi vào la chieu tang
        // z chieu đi lên là chieu tang
        if(vitri_y>y){
            HAL_GPIO_WritePin(DIR_Y_GPIO_Port,DIR_Y_Pin,0);
            HAL_GPIO_WritePin(STEP_Y_GPIO_Port,STEP_Y_Pin,1);
            delay_us(1000);
            HAL_GPIO_WritePin(STEP_Y_GPIO_Port,STEP_Y_Pin,0);
            delay_us(1000);
            vitri_y--;
        }
        else{
            HAL_GPIO_WritePin(DIR_Y_GPIO_Port,DIR_Y_Pin,1);
            HAL_GPIO_WritePin(STEP_Y_GPIO_Port,STEP_Y_Pin,1);
            delay_us(1000);
            HAL_GPIO_WritePin(STEP_Y_GPIO_Port,STEP_Y_Pin,0);
            delay_us(1000);
            vitri_y++;
        }
        // sau khi chieu y chay xong thi moi chay chieu x,z
        if(vitri_y==y){
            if(vitri_x>x){
                HAL_GPIO_WritePin(DIR_X_GPIO_Port,DIR_X_Pin,0);
                HAL_GPIO_WritePin(STEP_X_GPIO_Port,STEP_X_Pin,1);
                delay_us(300);
                HAL_GPIO_WritePin(STEP_X_GPIO_Port,STEP_X_Pin,0);
                delay_us(300);
                vitri_x--;
            }
            else{
                HAL_GPIO_WritePin(DIR_X_GPIO_Port,DIR_X_Pin,1);
                HAL_GPIO_WritePin(STEP_X_GPIO_Port,STEP_X_Pin,1);
                delay_us(300);
                HAL_GPIO_WritePin(STEP_X_GPIO_Port,STEP_X_Pin,0);
                delay_us(300);
            }
        }
    }
}
```

```
        vitri_x++;
    }
    if(vitri_x==x){
        if(vitri_z>z){
            HAL_GPIO_WritePin(DIR_Z_GPIO_Port,DIR_Z_Pin,1);

            HAL_GPIO_WritePin(STEP_Z_GPIO_Port,STEP_Z_Pin,1);
                delay_us(100);

            HAL_GPIO_WritePin(STEP_Z_GPIO_Port,STEP_Z_Pin,0);
                delay_us(100);
                vitri_z--;
        }
        else{
            HAL_GPIO_WritePin(DIR_Z_GPIO_Port,DIR_Z_Pin,0);

            HAL_GPIO_WritePin(STEP_Z_GPIO_Port,STEP_Z_Pin,1);
                delay_us(100);

            HAL_GPIO_WritePin(STEP_Z_GPIO_Port,STEP_Z_Pin,0);
                delay_us(100);
                vitri_z++;
        }
        if(vitri_x==x && vitri_z==z){
            HAL_Delay(500);
            break;
        }
    }
}

/*map vi buoc xy vb8, z vb2
x 0 6500 13500
y 0 7500

z 6750
5000
```

```
        2000
        0

*/
// chương trình gui
void run_g1(){
    LCD_Gotoxy(0,0);
    sprintf(temp," Run G1 ");
    LCD_Puts(temp);
    run_step(0,0,0); // ve home
    run_step(0,7500,0);// dua tay vào
    run_step(0,7500,2000);// up
    run_step(0,0,2000);// lay ra
    run_step(6500,0,2000);// di chuyen sang o 1
    run_step(6500,7500,2000);// dua vào
    run_step(6500,7500,0);// ha xuống
    run_step(6500,0,0);// thu về
    LCD_Gotoxy(0,0);
    sprintf(temp," RUN OK ");
    LCD_Puts(temp);
}
void run_g2(){
    LCD_Gotoxy(0,0);
    sprintf(temp," Run G2 ");
    LCD_Puts(temp);
    run_step(0,0,0); // ve home
    run_step(0,7500,0);// dua tay vào
    run_step(0,7500,2000);// up
    run_step(0,0,2000);// lay ra
    run_step(13500,0,2000);// di chuyen sang o 2
    run_step(13500,7500,2000);// dua vào
    run_step(13500,7500,0);// ha xuống
    run_step(13500,0,0);// thu về
    LCD_Gotoxy(0,0);
    sprintf(temp," RUN OK ");
    LCD_Puts(temp);
}
void run_g3(){
    LCD_Gotoxy(0,0);
```

```
sprintf(temp," Run G3 ");
LCD_Puts(temp);
run_step(0,0,0); // ve home
run_step(0,7500,0);// dua tay vao
run_step(0,7500,2000);// up
run_step(0,0,2000);// lay ra
run_step(6500,0,2000);// di chuyen sang o 3x
run_step(6500,0,6750);// di chuyen sang o 3z
run_step(6500,7500,6750);// dua vao
run_step(6500,7500,5000);// ha xuong
run_step(6500,0,5000);// thu ve
LCD_Gotoxy(0,0);
sprintf(temp," RUN OK ");
LCD_Puts(temp);
}
void run_g4(){
    LCD_Gotoxy(0,0);
    sprintf(temp," Run G4 ");
    LCD_Puts(temp);
    run_step(0,0,0); // ve home
    run_step(0,7500,0);// dua tay vao
    run_step(0,7500,2000);// up
    run_step(0,0,2000);// lay ra
    run_step(13500,0,2000);// di chuyen sang o 4
    run_step(13500,0,6750);// di chuyen sang o 4
    run_step(13500,7500,6750);// dua vao
    run_step(13500,7500,5000);// ha xuong
    run_step(13500,0,5000);// thu ve
    LCD_Gotoxy(0,0);
    sprintf(temp," RUN OK ");
    LCD_Puts(temp);
}
// chuong trinh lay
void run_l1(){
    LCD_Gotoxy(0,0);
    sprintf(temp," Run L1 ");
    LCD_Puts(temp);
    run_step(6500,0,0); // di chuyen sang o 1
    run_step(6500,7500,0);// dua tay vao
```

```
run_step(6500,7500,2000);// up
run_step(6500,0,2000);// lay ra
run_step(0,0,2000);// di chuyen ve home
run_step(0,7500,2000);// dua vao
run_step(0,7500,0);// ha xuong
run_step(0,0,0);// thu ve
LCD_Gotoxy(0,0);
sprintf(temp,"  RUN OK  ");
LCD_Puts(temp);
}
void run_l2(){
LCD_Gotoxy(0,0);
sprintf(temp,"  Run L2  ");
LCD_Puts(temp);
run_step(13500,0,0); // di chuyen sang o 2
run_step(13500,7500,0);// dua tay vao
run_step(13500,7500,2000);// up
run_step(13500,0,2000);// lay ra
run_step(0,0,2000);// di chuyen ve home
run_step(0,7500,2000);// dua vao
run_step(0,7500,0);// ha xuong
run_step(0,0,0);// thu ve
LCD_Gotoxy(0,0);
sprintf(temp,"  RUN OK  ");
LCD_Puts(temp);
}
void run_l3(){
LCD_Gotoxy(0,0);
sprintf(temp,"  Run L3  ");
LCD_Puts(temp);
run_step(6500,0,5000); // di chuyen sang o 3
run_step(6500,7500,5000);// dua tay vao
run_step(6500,7500,6750);// up
run_step(6500,0,6750);// lay ra
run_step(0,0,2000);// di chuyen ve home
run_step(0,7500,2000);// dua vao
run_step(0,7500,0);// ha xuong
run_step(0,0,0);// thu ve
LCD_Gotoxy(0,0);
```

```
    sprintf(temp,"  RUN OK  ");
    LCD_Puts(temp);
}
void run_l4(){
    LCD_Gotoxy(0,0);
    sprintf(temp,"  Run L4  ");
    LCD_Puts(temp);
    run_step(13500,0,5000); // di chuyen sang o 3
    run_step(13500,7500,5000);// dua tay vao
    run_step(13500,7500,6750);// up
    run_step(13500,0,6750);// lay ra
    run_step(0,0,2000);// di chuyen ve home
    run_step(0,7500,2000);// dua vao
    run_step(0,7500,0);// ha xuong
    run_step(0,0,0);// thu ve
    LCD_Gotoxy(0,0);
    sprintf(temp,"  RUN OK  ");
    LCD_Puts(temp);
}
int main(void)
{
    HAL_Init();
    SystemClock_Config();
    MX_GPIO_Init();
    MX_USART1_UART_Init();
    HAL_UART_Receive_IT(&huart1,(uint8_t*)Rx_data,1);// cho phép nhan 1 byte
    LCD_Init();
    go_home();
    while (1)
    {
        // gui
        if(Rx_data[0]=='1')
        {
            run_g1();
            Rx_data[0]='0';
        }
        if(Rx_data[0]=='2')
        {
            run_g2();
        }
    }
}
```

```
        Rx_data[0]='0';
    }
    if(Rx_data[0]=='3')
    {
        run_g3();
        Rx_data[0]='0';
    }
    if(Rx_data[0]=='4')
    {
        run_g4();
        Rx_data[0]='0';
    }
    // lay
    if(Rx_data[0]=='5')
    {
        run_11();
        Rx_data[0]='0';
    }
    if(Rx_data[0]=='6')
    {
        run_12();
        Rx_data[0]='0';
    }
    if(Rx_data[0]=='7')
    {
        run_13();
        Rx_data[0]='0';
    }
    if(Rx_data[0]=='8')
    {
        run_14();
        Rx_data[0]='0';
    }
}
}

/**
 * @brief System Clock Configuration
 * @retval None
```



```
*/  
void SystemClock_Config(void)  
{  
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};  
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};  
  
    /** Initializes the CPU, AHB and APB busses clocks  
    */  
    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;  
    RCC_OscInitStruct.HSEState = RCC_HSE_ON;  
    RCC_OscInitStruct.HSEPredivValue = RCC_HSE_PREDIV_DIV1;  
    RCC_OscInitStruct.HSIState = RCC_HSI_ON;  
    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;  
    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;  
    RCC_OscInitStruct.PLL.PLLMUL = RCC_PLL_MUL9;  
    if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)  
    {  
        Error_Handler();  
    }  
    /** Initializes the CPU, AHB and APB busses clocks  
    */  
    RCC_ClkInitStruct.ClockType      =      RCC_CLOCKTYPE_HCLK|  
RCC_CLOCKTYPE_SYSCLK  
        |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;  
    RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;  
    RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;  
    RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV2;  
    RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV1;  
  
    if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_2) != HAL_OK)  
    {  
        Error_Handler();  
    }  
}  
  
/**  
 * @brief USART1 Initialization Function  
 * @param None  
 * @retval None
```

```
*/
static void MX_USART1_UART_Init(void)
{

/* USER CODE BEGIN USART1_Init 0 */

/* USER CODE END USART1_Init 0 */

/* USER CODE BEGIN USART1_Init 1 */

/* USER CODE END USART1_Init 1 */
huart1.Instance = USART1;
huart1.Init.BaudRate = 115200;
huart1.Init.WordLength = UART_WORDLENGTH_8B;
huart1.Init.StopBits = UART_STOPBITS_1;
huart1.Init.Parity = UART_PARITY_NONE;
huart1.Init.Mode = UART_MODE_TX_RX;
huart1.Init.HwFlowCtl = UART_HWCONTROL_NONE;
huart1.Init.OverSampling = UART_OVERSAMPLING_16;
if (HAL_UART_Init(&huart1) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN USART1_Init 2 */

/* USER CODE END USART1_Init 2 */

}

/**
 * @brief GPIO Initialization Function
 * @param None
 * @retval None
 */
static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStruct = {0};

/* GPIO Ports Clock Enable */
```

```
__HAL_RCC_GPIOC_CLK_ENABLE();
__HAL_RCC_GPIOD_CLK_ENABLE();
__HAL_RCC_GPIOA_CLK_ENABLE();
__HAL_RCC_GPIOB_CLK_ENABLE();

/*Configure GPIO pin Output Level */
HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, GPIO_PIN_RESET);

/*Configure GPIO pin Output Level */
HAL_GPIO_WritePin(GPIOA, LCD_D7_Pin|LCD_D6_Pin|LCD_D5_Pin|DIR_X_Pin
                 |STEP_X_Pin, GPIO_PIN_RESET);

/*Configure GPIO pin Output Level */
HAL_GPIO_WritePin(GPIOB, LCD_D4_Pin|LCD_EN_Pin|LCD_RW_Pin|LCD_RS_Pin
                 |DIR_Y_Pin|STEP_Y_Pin|DIR_Z_Pin|STEP_Z_Pin, GPIO_PIN_RESET);

/*Configure GPIO pin : LED_Pin */
GPIO_InitStruct.Pin = LED_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(LED_GPIO_Port, &GPIO_InitStruct);

/*Configure GPIO pins : LCD_D7_Pin LCD_D6_Pin LCD_D5_Pin */
GPIO_InitStruct.Pin = LCD_D7_Pin|LCD_D6_Pin|LCD_D5_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);

/*Configure GPIO pins : LCD_D4_Pin LCD_EN_Pin LCD_RW_Pin LCD_RS_Pin */
GPIO_InitStruct.Pin = LCD_D4_Pin|LCD_EN_Pin|LCD_RW_Pin|LCD_RS_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);

/*Configure GPIO pins : DIR_X_Pin STEP_X_Pin */
GPIO_InitStruct.Pin = DIR_X_Pin|STEP_X_Pin;
```

```
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_HIGH;
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);

/*Configure GPIO pins : DIR_Y_Pin STEP_Y_Pin DIR_Z_Pin STEP_Z_Pin */
GPIO_InitStruct.Pin = DIR_Y_Pin|STEP_Y_Pin|DIR_Z_Pin|STEP_Z_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_HIGH;
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);

/*Configure GPIO pins : ctht_x_Pin ctht_y_Pin ctht_z_Pin */
GPIO_InitStruct.Pin = ctht_x_Pin|ctht_y_Pin|ctht_z_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
GPIO_InitStruct.Pull = GPIO_PULLUP;
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);

}

/* USER CODE BEGIN 4 */

/* USER CODE END 4 */

/**
 * @brief This function is executed in case of error occurrence.
 * @retval None
 */
void Error_Handler(void)
{
/* USER CODE BEGIN Error_Handler_Debug */
/* User can add his own implementation to report the HAL error return state */

/* USER CODE END Error_Handler_Debug */
}

#ifdef USE_FULL_ASSERT
/**
 * @brief Reports the name of the source file and the source line number

```

```
*      where the assert_param error has occurred.
* @param file: pointer to the source file name
* @param line: assert_param error line source number
* @retval None
*/
void assert_failed(uint8_t *file, uint32_t line)
{
    /* USER CODE BEGIN 6 */
    /* User can add his own implementation to report the file name and line number,
    tex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
    /* USER CODE END 6 */
}
#endif /* USE_FULL_ASSERT */

/***** (C) COPYRIGHT STMicroelectronics *****END OF FILE*****/
```

Code khối giao diện:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using AForge.Video;
using AForge.Video.DirectShow;
using MessagingToolkit.QRCode.Codec.Data;
using System.Threading;
using ZXing;
using System.IO;
using System.IO.Ports;
using System.Data.SqlClient;

namespace giaodiendieukhien
{
```

```
public partial class Form1 : Form
{
    string ngdung = "";
        string strcon = @"Data Source=DESKTOP-L0HIT9B\MY_SQL;Initial
Catalog=thuvientudong;Integrated Security=True";
    SqlConnection sqlcon = null;

    int mode=0; // mặc định là mượn sách
    int choose_com = 0;
    Image img;
    Image img_qr_auto;
    private FilterInfoCollection cameras;
    private VideoCaptureDevice cam;

    public Form1()
    {
        InitializeComponent();
        cbb_com.DataSource = SerialPort.GetPortNames();
        cameras = new FilterInfoCollection(FilterCategory.VideoInputDevice);
        foreach(FilterInfo info in cameras )
        {
            cbb_cam.Items.Add(info.Name);
        }
        cbb_cam.SelectedIndex = 0;
        try
        {
            sqlcon = new SqlConnection(strcon);
            if (sqlcon.State == ConnectionState.Closed)
            {
                sqlcon.Open();
                //MessageBox.Show("Đã kết nối đến cơ sở dữ liệu", "Thông báo");
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
    }
}
```

```
private void toolStripMenuItem2_Click(object sender, EventArgs e)
{
}

private void Form1_Load(object sender, EventArgs e)
{
}

private void btn_makeqr_Click(object sender, EventArgs e)
{
    if (tb_obj.Text != "" && cbb_vitri.Text != "")
    {
        QRCodeGenerator qr = new QRCodeGenerator();
        string nd = tb_obj.Text + "-" + cbb_vitri.Text;
        var data = qr.CreateQrCode(nd, QRCodeGenerator.ECCLevel.H);
        var code = new QRCode(data);
        ptb_makeqr.Image = code.GetGraphic(10);

        string sqltimkiem = "SELECT *FROM khosach WHERE vitri=@vitri";
        SqlCommand cmd = new SqlCommand(sqltimkiem, sqlcon);
        cmd.Parameters.AddWithValue("tensach", tb_obj.Text);
        cmd.Parameters.AddWithValue("vitri", cbb_vitri.Text);
        cmd.Parameters.AddWithValue("link", tb_linkmota.Text);
        cmd.ExecuteNonQuery();
        SqlDataReader dr = cmd.ExecuteReader();
        DataTable dt = new DataTable();
        dt = new DataTable();
        dt.Load(dr);
        dtgv_khosach.DataSource = dt;
        dtgv_khosach.AutoSizeColumnHeadersHeight();

        if (dtgv_khosach.Rows[0].Cells[0].Value != null) // nếu mã tồn tại thì hỏi sửa, nếu
        ko thì thực hiện thêm // cần thực hiện tương tác trên cả hai bảng
        {
            MessageBoxButtons btn = MessageBoxButtons.YesNo;
            DialogResult result = MessageBox.Show("Mã sản phẩm đã tồn tại, bạn có muốn
            update mã sản phẩm", "Thông báo", btn);
        }
    }
}
```

```
if (result == DialogResult.Yes)// sửa
{
    string sqledit = "UPDATE khosach SET
tensach=@tensach,vitri=@vitri,link=@link WHERE vitri=@vitri";
    cmd = new SqlCommand(sqledit, sqlcon);
    cmd.Parameters.AddWithValue("tensach", tb_obj.Text);
    cmd.Parameters.AddWithValue("vitri", cbb_vitri.Text);
    cmd.Parameters.AddWithValue("link", tb_linkmota.Text);
    cmd.ExecuteNonQuery();
    MessageBox.Show("Update thành công", "Thông báo");
    Hienthi();
}
}
else // thêm
{
    string sqladd = "INSERT INTO khosach VALUES (@tensach, @vitri, @link)";
    cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("tensach", tb_obj.Text);
    cmd.Parameters.AddWithValue("vitri", cbb_vitri.Text);
    cmd.Parameters.AddWithValue("link", tb_linkmota.Text);
    cmd.ExecuteNonQuery();
    MessageBox.Show("Thêm thành công", "Thông báo");
    Hienthi();
}
}
else
{
    MessageBox.Show("Vui lòng nhập đủ thông tin !!!","Thông báo");
}
if (!cb_mota.Checked)
{
    if (tb_linkmota.Text == "")
    {
        MessageBox.Show("Bạn cần thêm link ảnh mô tả sách", "Thông báo");
    }
}
}
}

private void btn_save_Click(object sender, EventArgs e)
```



```
{
    img = ptb_makeqr.Image;
    SaveFileDialog sf = new SaveFileDialog();
    sf.Filter = "JPG(*.JPG)|*.jpg";
    if(sf.ShowDialog()==DialogResult.OK)
    {
        img.Save(sf.FileName);
    }
}

private void btn_start_auto_Click(object sender, EventArgs e)
{
    timer_checkqr.Enabled = true;
    grb_dk_thuong.Enabled = false;
    grb_dk_thuong.BackColor = Color.Gray;
    if(cam!=null && cam.IsRunning)
    {
        cam.Stop();
    }
    cam = new VideoCaptureDevice(cameras[cbb_cam.SelectedIndex].MonikerString);
    cam.NewFrame += cam_NewFrame;
    cam.Start();
}

void cam_NewFrame(object sender, NewFrameEventArgs eventArgs)
{
    Bitmap bm = (Bitmap)eventArgs.Frame.Clone();
    ptb_qr.Image = bm;
}

private void btn_stop_auto_Click(object sender, EventArgs e)
{
    if (cam != null && cam.IsRunning)
    {
        cam.Stop();
        PictureBox ptb = new PictureBox();
        ptb.BackColor = Color.LightGray;
        ptb_qr.Image = ptb.Image;
    }
}
```

```
        grb_dk_thucong.Enabled = true;
        grb_dk_thucong.BackColor = Color.Silver;
    }
}

protected override void OnClosed(EventArgs e)
{
    base.OnClosed(e);
    if (cam != null && cam.IsRunning)
    {
        cam.Stop();
    }
}

private void check_qr()
{
    while (true)
    {
        try
        {
            if (ptb_qr.Image != null)
            {
                MessageBox.Show("run");
                img_qr_auto = ptb_qr.Image;
                MessagingToolkit.QRCode.Codec.QRCodeDecoder decoder = new
MessagingToolkit.QRCode.Codec.QRCodeDecoder();
                tb_ndqr.Text = decoder.Decode(new QRCodeBitmapImage(ptb_qr.Image as
Bitmap));
            }
        }
        catch
        {
        }
    }
}

private void timer_checkqr_Tick(object sender, EventArgs e)
{
}
```

```
try
{
    if (ptb_qr.Image != null)
    {
        //MessageBox.Show("run");
        img_qr_auto = ptb_qr.Image;
        BarcodeReader reader = new BarcodeReader();
        var result = reader.Decode(img_qr_auto as Bitmap);
        if (result != null)
        {
            tb_ndqr.Text = result.ToString();
            string nd = tb_ndqr.Text;
            if (nd.IndexOf("Vị trí 1") != -1)
            {
                if (Serial.IsOpen)
                {
                    if (mode == 0)
                    {
                        Serial.WriteLine("5");
                    }
                    else
                    {
                        if (mode == 1)
                        {
                            Serial.WriteLine("1");
                        }
                    }
                }

                string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach,
@thaotac,@gio)";
                SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
                cmd.Parameters.AddWithValue("ten", ngdung);
                cmd.Parameters.AddWithValue("sach", lb_vitri1.Text);
                if (mode == 0)
                {
                    cmd.Parameters.AddWithValue("thaotac", "Mượn");
                }
                else
                {
                    cmd.Parameters.AddWithValue("thaotac", "Trả");
                }
            }
        }
    }
}
```

```
    }
    string thoigian = DateTime.Now.ToString("dddd , MMM dd
yyyy,hh:mm:ss");
    cmd.Parameters.AddWithValue("gio", thoigian);
    cmd.ExecuteNonQuery();
    Hienthi();
}
else
{
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");
}
}
if (nd.IndexOf("Vị trí 2") != -1)
{
    if (Serial.IsOpen)
    {
        if (mode == 0)
        {
            Serial.WriteLine("6");
        }
        else
        {
            if (mode == 1)
            {
                Serial.WriteLine("2");
            }
        }
    }
    string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach,
@thaotac,@gio)";
    SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("ten", ngdung);
    cmd.Parameters.AddWithValue("sach", lb_vitri2.Text);
    if (mode == 0)
    {
        cmd.Parameters.AddWithValue("thaotac", "Mượn");
    }
    else
    {
        cmd.Parameters.AddWithValue("thaotac", "Trả");
    }
}
```

```
    }
    string thoigian = DateTime.Now.ToString("dddd , MMM dd
yyyy,hh:mm:ss");
    cmd.Parameters.AddWithValue("gio", thoigian);
    cmd.ExecuteNonQuery();
    Hienthi();
}
else
{
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");
}
}
if (nd.IndexOf("Vị trí 3") != -1)
{
    if (Serial.IsOpen)
    {
        if (mode == 0)
        {
            Serial.WriteLine("7");
        }
        else
        {
            if (mode == 1)
            {
                Serial.WriteLine("3");
            }
        }
    }
    string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach,
@thaotac,@gio)";
    SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("ten", ngdung);
    cmd.Parameters.AddWithValue("sach", lb_vitri3.Text);
    if (mode == 0)
    {
        cmd.Parameters.AddWithValue("thaotac", "Mượn");
    }
    else
    {
        cmd.Parameters.AddWithValue("thaotac", "Trả");
    }
}
```

```
    }
    string thoigian = DateTime.Now.ToString("dddd , MMM dd
yyyy,hh:mm:ss");
    cmd.Parameters.AddWithValue("gio", thoigian);
    cmd.ExecuteNonQuery();
    Hienthi();
}
else
{
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");
}
}
if (nd.IndexOf("Vị trí 4") != -1)
{
    if (Serial.IsOpen)
    {
        if (mode == 0)
        {
            Serial.WriteLine("8");
        }
        else
        {
            if (mode == 1)
            {
                Serial.WriteLine("4");
            }
        }
    }
    string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach,
@thaotac,@gio)";
    SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("ten", ngdung);
    cmd.Parameters.AddWithValue("sach", lb_vitri4.Text);
    if (mode == 0)
    {
        cmd.Parameters.AddWithValue("thaotac", "Mượn");
    }
    else
    {
        cmd.Parameters.AddWithValue("thaotac", "Trả");
    }
}
```

```
    }
    string thoigian = DateTime.Now.ToString("dddd , MMM dd
yyyy,hh:mm:ss");
    cmd.Parameters.AddWithValue("gio", thoigian);
    cmd.ExecuteNonQuery();
    Hienthi();
}
else
{
    MessageBox.Show("Kiểm tra kết nối cổng com", "Thông báo");
}
}
if (nd.IndexOf("Vị trí 5") != -1)
{
    if (Serial.IsOpen)
    {
        if (mode == 0)
        {
            Serial.WriteLine("lay_5");
        }
        else
        {
            if (mode == 1)
            {
                Serial.WriteLine("gui_5");
            }
        }
    }
    string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach,
@thaotac,@gio)";
    SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("ten", ngdung);
    if (mode == 0)
    {
        cmd.Parameters.AddWithValue("thaotac", "Mượn");
    }
    else
    {
        cmd.Parameters.AddWithValue("thaotac", "Trả");
    }
}
```

```
string thoigian = DateTime.Now.ToString("dddd , MMM dd
yyyy,hh:mm:ss");
    cmd.Parameters.AddWithValue("gio", thoigian);
    cmd.ExecuteNonQuery();
    Hienthi();
}
else
{
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");
}
}
if (nd.IndexOf("Vị trí 6") != -1)
{
    if (Serial.IsOpen)
    {
        if (mode == 0)
        {
            Serial.WriteLine("lay_6");
        }
        else
        {
            if (mode == 1)
            {
                Serial.WriteLine("gui_6");
            }
        }
    }
    string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach,
@thaotac,@gio)";
    SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("ten", ngdung);
    if (mode == 0)
    {
        cmd.Parameters.AddWithValue("thaotac", "Mượn");
    }
    else
    {
        cmd.Parameters.AddWithValue("thaotac", "Trả");
    }
}
```



```
string thoigian = DateTime.Now.ToString("dddd , MMM dd  
yyyy, hh:mm:ss");  
    cmd.Parameters.AddWithValue("gio", thoigian);  
    cmd.ExecuteNonQuery();  
    Hienthi();  
}  
else  
{  
    MessageBox.Show("Kiểm tra kết nối cổng com", "Thông báo");  
}  
}  
}  
}  
}  
catch  
{  
}  
}  
private void btn_kn_Click(object sender, EventArgs e)  
{  
    if (choose_com == 1)  
    {  
        if (!Serial.IsOpen)  
        {  
            try  
            {  
                Serial.PortName = cbb_com.SelectedItem.ToString();  
                Serial.Open();  
                btn_kn.BackColor = Color.Green;  
            }  
            catch  
            {  
                MessageBox.Show("Vui lòng kết nối thiết bị !!!");  
            }  
        }  
    }  
    else  
    {  
        MessageBox.Show("Vui lòng chọn cổng com");  
    }  
}
```

```
    }  
  }  
  
  private void timer_ktra_kn_Tick(object sender, EventArgs e)  
  {  
    if (Serial.IsOpen)  
    {  
      btn_kn.BackColor = Color.Green;  
    }  
    else  
    {  
      btn_kn.BackColor = Color.Red;  
    }  
  
    if (cbb_com.Text != "")  
    {  
      choose_com = 1;  
    }  
  }  
  
  private void btn_scan_Click(object sender, EventArgs e)  
  {  
    cbb_com.DataSource = SerialPort.GetPortNames();  
    choose_com = 0;  
  }  
  
  private void btn_muon_Click(object sender, EventArgs e)  
  {  
    mode = 0;  
    btn_muon.BackColor = Color.Green;  
    btn_tra.BackColor = Color.Red;  
    grb_dk_thucong.Enabled = true;  
    grb_camera.Enabled = true;  
    grb_dk_thucong.BackColor = Color.Silver;  
    grb_camera.BackColor = Color.Silver;  
    grb_mode.BackColor = Color.Silver;  
  }  
  
  private void btn_tra_Click(object sender, EventArgs e)
```

```
{
    mode = 1;
    btn_muon.BackColor = Color.Red;
    btn_tra.BackColor = Color.Green;
    grb_dk_thucong.Enabled = true;
    grb_camera.Enabled = true;
    grb_dk_thucong.BackColor = Color.Silver;
    grb_camera.BackColor = Color.Silver;
    grb_mode.BackColor = Color.Silver;
}
// hiệu ứng nhấn nút
private void btn_start_auto_MouseDown(object sender, MouseEventArgs e)
{
    btn_start_auto.BackColor = Color.Gray;
}

private void btn_start_auto_MouseUp(object sender, MouseEventArgs e)
{
    btn_start_auto.BackColor = Color.Silver;
}

private void btn_stop_auto_MouseDown(object sender, MouseEventArgs e)
{
    btn_stop_auto.BackColor = Color.Gray;
}

private void btn_stop_auto_MouseUp(object sender, MouseEventArgs e)
{
    btn_stop_auto.BackColor = Color.Silver;
}

private void btn_makeqr_MouseDown(object sender, MouseEventArgs e)
{
    btn_makeqr.BackColor = Color.Silver;
}

private void btn_save_MouseDown(object sender, MouseEventArgs e)
{
```

```
        btn_save.BackColor = Color.Silver;
    }
    private void btn_makeqr_MouseUp(object sender, MouseEventArgs e)
    {
        btn_makeqr.BackColor = Color.Gray;
    }

    private void btn_save_MouseUp(object sender, MouseEventArgs e)
    {
        btn_save.BackColor = Color.Gray;
    }
    private void btn_vt1_MouseDown(object sender, MouseEventArgs e)
    {
        btn_vt1.BackColor = Color.Gray;
    }

    private void btn_vt2_MouseDown(object sender, MouseEventArgs e)
    {
        btn_vt2.BackColor = Color.Gray;
    }

    private void btn_vt3_MouseDown(object sender, MouseEventArgs e)
    {
        btn_vt3.BackColor = Color.Gray;
    }

    private void btn_vt4_MouseDown(object sender, MouseEventArgs e)
    {
        btn_vt4.BackColor = Color.Gray;
    }
    private void btn_vt1_MouseUp(object sender, MouseEventArgs e)
    {
        btn_vt1.BackColor = Color.Silver;
    }

    private void btn_vt2_MouseUp(object sender, MouseEventArgs e)
    {
        btn_vt2.BackColor = Color.Silver;
    }
}
```

```
private void btn_vt3_MouseUp(object sender, MouseEventArgs e)
{
    btn_vt3.BackColor = Color.Silver;
}

private void btn_vt4_MouseUp(object sender, MouseEventArgs e)
{
    btn_vt4.BackColor = Color.Silver;
}

private void btn_scan_MouseDown(object sender, MouseEventArgs e)
{
    btn_scan.BackColor = Color.Gray;
}

private void btn_scan_MouseUp(object sender, MouseEventArgs e)
{
    btn_scan.BackColor = Color.Green;
}

private void timer_serial_Tick(object sender, EventArgs e)
{
}

// serial event

private void btn_vt1_Click(object sender, EventArgs e)
{
    if (Serial.IsOpen)
    {
        if (mode == 0) // lay
        {
            Serial.WriteLine("5");
        }
        else
        {
            if (mode == 1) // tra
            {
                Serial.WriteLine("1");
            }
        }
    }
}
```

```
    }
  }
}
else
{
  MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");
}

string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach, @thaotac,@gio)";
SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
cmd.Parameters.AddWithValue("ten", ngdung);
cmd.Parameters.AddWithValue("sach", lb_vitri1.Text);
if (mode == 0)
{
  cmd.Parameters.AddWithValue("thaotac", "Mượn");
}
else
{
  cmd.Parameters.AddWithValue("thaotac", "Trả");
}
string thoigian = DateTime.Now.ToString("dddd , MMM dd yyyy,hh:mm:ss");
cmd.Parameters.AddWithValue("gio", thoigian);
cmd.ExecuteNonQuery();
Hienthi();
}

private void btn_vt2_Click(object sender, EventArgs e)
{
  if (Serial.IsOpen)
  {
    if (mode == 0)
    {
      Serial.WriteLine("6");
    }
    else
    {
      if (mode == 1)
      {
        Serial.WriteLine("2");
      }
    }
  }
}
```

```
    }  
  }  
}  
else  
{  
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");  
}  
string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach, @thaotac,@gio)";  
SqlCommand cmd = new SqlCommand(sqladd, sqlcon);  
cmd.Parameters.AddWithValue("ten", ngdung);  
cmd.Parameters.AddWithValue("sach", lb_vitri2.Text);  
if (mode == 0)  
{  
    cmd.Parameters.AddWithValue("thaotac", "Mượn");  
}  
else  
{  
    cmd.Parameters.AddWithValue("thaotac", "Trả");  
}  
string thoigian = DateTime.Now.ToString("dddd , MMM dd yyyy,hh:mm:ss");  
cmd.Parameters.AddWithValue("gio", thoigian);  
cmd.ExecuteNonQuery();  
Hienthi();  
}  
  
private void btn_vt3_Click(object sender, EventArgs e)  
{  
    if (Serial.IsOpen)  
    {  
        if (mode == 0)  
        {  
            Serial.WriteLine("7");  
        }  
        else  
        {  
            if (mode == 1)  
            {  
                Serial.WriteLine("3");  
            }  
        }  
    }  
}
```

```
    }  
  }  
  else  
  {  
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");  
  }  
  string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach, @thaotac,@gio)";  
  SqlCommand cmd = new SqlCommand(sqladd, sqlcon);  
  cmd.Parameters.AddWithValue("ten", ngdung);  
  cmd.Parameters.AddWithValue("sach", lb_vitri3.Text);  
  if (mode == 0)  
  {  
    cmd.Parameters.AddWithValue("thaotac", "Mượn");  
  }  
  else  
  {  
    cmd.Parameters.AddWithValue("thaotac", "Trả");  
  }  
  string thoigian = DateTime.Now.ToString("dddd , MMM dd yyyy,hh:mm:ss");  
  cmd.Parameters.AddWithValue("gio", thoigian);  
  cmd.ExecuteNonQuery();  
  Hienthi();  
}  
  
private void btn_vt4_Click(object sender, EventArgs e)  
{  
  if (Serial.IsOpen)  
  {  
    if (mode == 0)  
    {  
      Serial.WriteLine("8");  
    }  
    else  
    {  
      if (mode == 1)  
      {  
        Serial.WriteLine("4");  
      }  
    }  
  }  
}
```



```
}
else
{
    MessageBox.Show("Kiểm tra kết nối công com", "Thông báo");
}
string sqladd = "INSERT INTO lichsu VALUES (@ten, @sach, @thaotac,@gio)";
SqlCommand cmd = new SqlCommand(sqladd, sqlcon);
cmd.Parameters.AddWithValue("ten", ngdung);
cmd.Parameters.AddWithValue("sach", lb_vitri4.Text);
if (mode == 0)
{
    cmd.Parameters.AddWithValue("thaotac", "Mượn");
}
else
{
    cmd.Parameters.AddWithValue("thaotac", "Trả");
}
string thoigian = DateTime.Now.ToString("dddd , MMM dd yyyy,hh:mm:ss");
cmd.Parameters.AddWithValue("gio", thoigian);
cmd.ExecuteNonQuery();
Hienthi();
}
// -----kết nối sql
private void btn_knsql_Click(object sender, EventArgs e)
{
    try
    {
        sqlcon = new SqlConnection(strcon);
        if (sqlcon.State == ConnectionState.Closed)
        {
            sqlcon.Open();
            MessageBox.Show("Đã kết nối đến cơ sở dữ liệu", "Thông báo");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```

```
private void btn_nknsql_Click(object sender, EventArgs e)
{
    if(sqlcon!=null && sqlcon.State == ConnectionState.Open)
    {
        sqlcon.Close();
        MessageBox.Show("Đã ngắt kết nối cơ sở dữ liệu","Thông báo");
    }
}

private void dateTimePicker2_ValueChanged(object sender, EventArgs e)
{
}

private void btn_dangnhap_Click(object sender, EventArgs e)
{
    try
    {
        // tra tk, makhau
        string sqledit = "SELECT taikhoan FROM quanlinguoidung WHERE
matkhau=@matkhau";
        SqlCommand cmd = new SqlCommand(sqledit, sqlcon);
        cmd.Parameters.AddWithValue("matkhau", tb_pass.Text);
        cmd.ExecuteNonQuery();
        SqlDataReader dr = cmd.ExecuteReader();
        DataTable dt = new DataTable();
        dt = new DataTable();
        dt.Load(dr);
        dtgv_tk.DataSource = dt;
        if (dtgv_tk.Rows[0].Cells[0].Value != null)
        {
            if (dtgv_tk.Rows[0].Cells[0].Value.ToString() == tb_dangnhap.Text)
            {
                ngdung = tb_dangnhap.Text;
                grb_control.Enabled = true;
                grb_taoqr.Enabled = true;
                MessageBox.Show("Đăng nhập thành công", "Thông báo đăng nhập");
                string truycap = "Truy cập: " + tb_dangnhap.Text;
            }
        }
    }
}
```

```
        lb_truycap.Text = truycap;
        tb_dangnhap.Text = "";
        tb_pass.Text = "";
        btn_dangki.Visible = true;
        Hienthi();
    }
    else
    {
        MessageBox.Show("Đăng nhập không thành công", "Thông báo đăng nhập");
        lb_truycap.Text = "Chưa truy cập";
    }
}
else
{
    MessageBox.Show("Đăng nhập không thành công", "Thông báo đăng nhập");
    lb_truycap.Text = "Chưa truy cập";
}
}
catch
{
    MessageBox.Show("Chưa kết nối cơ sở dữ liệu","Thông báo");
}
}

private void btn_dangxuat_Click(object sender, EventArgs e)
{
    ngdung = "";
    grb_control.Enabled = false;
    grb_taoqr.Enabled = false;
    lb_truycap.Text = "Chưa truy cập";
    tb_dangnhap.Text = "";
    tb_pass.Text = "";
    btn_dangki.Visible = false;
    Serial.Close();
    Hienthi();
}

private void btn_dangki_Click(object sender, EventArgs e)
{
```

```
string sqltimkiem = "SELECT *FROM quanlinguoidung WHERE
taikhoan=@taikhoan";
SqlCommand cmd = new SqlCommand(sqltimkiem, sqlcon);
cmd.Parameters.AddWithValue("taikhoan", tb_dangnhap.Text);
cmd.Parameters.AddWithValue("matkhau", tb_pass.Text);
cmd.ExecuteNonQuery();
SqlDataReader dr = cmd.ExecuteReader();
DataTable dt = new DataTable();
dt = new DataTable();
dt.Load(dr);
dtgv_tk.DataSource = dt;
dtgv_tk.AutoSizeColumnHeadersHeight();
if (dtgv_tk.Rows[0].Cells[0].Value != null) // nếu mã tồn tại thì hỏi sửa, nếu ko thì
thực hiện thêm
{
    MessageBoxButtons btn = MessageBoxButtons.YesNo;
    DialogResult result = MessageBox.Show("Tài khoản đã tồn tại, bạn có muốn
update thông tin đăng nhập?", "Thông báo", btn);
    if (result == DialogResult.Yes)// sửa
    {
        string sqledit = "UPDATE quanlinguoidung SET
taikhoan=@taikhoan,matkhau=@matkhau,ghichu=@ghichu WHERE taikhoan=@taikhoan";
        cmd = new SqlCommand(sqledit, sqlcon);
        cmd.Parameters.AddWithValue("taikhoan", tb_dangnhap.Text);
        cmd.Parameters.AddWithValue("matkhau", tb_pass.Text);
        cmd.Parameters.AddWithValue("ghichu", " ");
        cmd.ExecuteNonQuery();
        Hienthi();
    }
}
else // thêm
{
    string sqladd = "INSERT INTO quanlinguoidung VALUES (@taikhoan,
@matkhau, @ghichu)";
    cmd = new SqlCommand(sqladd, sqlcon);
    cmd.Parameters.AddWithValue("taikhoan", tb_dangnhap.Text);
    cmd.Parameters.AddWithValue("matkhau", tb_pass.Text);
    cmd.Parameters.AddWithValue("ghichu", " ");
    cmd.ExecuteNonQuery();
}
```

```
Hienthi();
MessageBox.Show("Đã tạo tài khoản thành công", "Thông báo");
}
}
public void Hienthi()
{
    string sql_select = "SELECT *FROM khosach";
    SqlCommand cmd = new SqlCommand(sql_select, sqlcon);
    SqlDataReader dr = cmd.ExecuteReader();
    DataTable dt = new DataTable();
    dt.Load(dr);
    dtgv_khosach.DataSource = dt;
    dtgv_khosach.AutoSizeColumnHeadersHeight();

    sql_select = "SELECT *FROM lichsu WHERE ten=@ten";
    cmd = new SqlCommand(sql_select, sqlcon);
    cmd.Parameters.AddWithValue("ten", ngdung);
    dr = cmd.ExecuteReader();
    dt = new DataTable();
    dt.Load(dr);
    dtgv_lichsu.DataSource = dt;
    dtgv_lichsu.AutoSizeColumnHeadersHeight();

    sql_select = "SELECT *FROM quanlinguoidung";
    cmd = new SqlCommand(sql_select, sqlcon);
    dr = cmd.ExecuteReader();
    dt = new DataTable();
    dt.Load(dr);
    dtgv_tk.DataSource = dt;
    dtgv_tk.AutoSizeColumnHeadersHeight();

    // lấy nội dung hình ảnh để mô tả kho sách
    // vị trí 1
    string sqledit = "SELECT tensach FROM khosach WHERE vitri=@vitri";
    cmd = new SqlCommand(sqledit, sqlcon);
    cmd.Parameters.AddWithValue("vitri", "Vị trí 1");
    cmd.ExecuteNonQuery();
    dr = cmd.ExecuteReader();
    dt = new DataTable();
```

```
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    lb_vitri1.Text=(dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
}
sqledit = "SELECT link FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 1");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    string img = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
    try
    {
        ptb_vt1.Image = Image.FromFile(img);
    }
    catch
    {
    }
}
else
{
    MessageBox.Show("Kiểm tra mã sản phẩm");
}
// vị trí 2
sqledit = "SELECT tensach FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 2");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
```

```
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    lb_vitri2.Text = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
}
sqledit = "SELECT link FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 2");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    string img = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
    try
    {
        ptb_vt2.Image = Image.FromFile(img);
    }
    catch
    {
    }
}
else
{
    MessageBox.Show("Kiểm tra mã sản phẩm");
}
// vị trí 3
sqledit = "SELECT tensach FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 3");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
```

```
lb_vitri3.Text = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
}
sqledit = "SELECT link FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 3");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    string img = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
    try
    {
        ptb_vt3.Image = Image.FromFile(img);
    }
    catch
    {
    }
}
else
{
    MessageBox.Show("Kiểm tra mã sản phẩm");
}
// vị trí 4
sqledit = "SELECT tensach FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 4");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    lb_vitri4.Text = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
}
}
```



```
sqledit = "SELECT link FROM khosach WHERE vitri=@vitri";
cmd = new SqlCommand(sqledit, sqlcon);
cmd.Parameters.AddWithValue("vitri", "Vị trí 4");
cmd.ExecuteNonQuery();
dr = cmd.ExecuteReader();
dt = new DataTable();
dt.Load(dr);
dtgv_timkiem.DataSource = dt;
if (dtgv_timkiem.Rows[0].Cells[0].Value != null)
{
    string img = (dtgv_timkiem.Rows[0].Cells[0].Value.ToString());
    try
    {
        ptb_vt4.Image = Image.FromFile(img);
    }
    catch
    {
    }
}
else
{
    MessageBox.Show("Kiểm tra mã sản phẩm");
}
}

private void cbb_com_SelectedIndexChanged(object sender, EventArgs e)
{
}
}
}
```