

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KỸ THUẬT MÁY TÍNH



PHẠM QUỐC ĐĂNG
VIÊN MINH TÂN

ĐỒ ÁN MÔN HỌC
THIẾT KẾ MẠCH HIỆN CHỮ TRÊN MA TRẬN LED

TP. HỒ CHÍ MINH, 2021

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KỸ THUẬT MÁY TÍNH

PHẠM QUỐC ĐĂNG – 19520036

VIÊN MINH TÂN - 19520928

ĐỒ ÁN MÔN HỌC
THIẾT KẾ MẠCH CHẠY CHỮ TRÊN MA TRẬN LED

GIẢNG VIÊN HƯỚNG DẪN

PHAN ĐÌNH DUY

TP. HỒ CHÍ MINH, 2021

MỤC LỤC

Chương 1. TỔNG QUAN.....	2
1.1. Đặt vấn đề.....	2
1.2. Nhu cầu, ý tưởng	2
1.3. Mục tiêu.....	2
1.4. Kết luận.....	3
Chương 2. CƠ SỞ LÝ THUYẾT	4
2.1. IC AT89S51.....	4
2.1.1. Tổng quan.....	4
2.1.2. Cấu tạo, sơ đồ chân	5
2.1.3. Kiến trúc bên trong.....	6
2.1.4. Tổ chức bộ nhớ.....	7
2.1.4.1. Tổ chức bộ nhớ trong.....	7
2.1.4.2. Bộ nhớ chương trình.....	7
2.1.4.3. Bộ nhớ dữ liệu	7
2.1.5. Tổ chức bộ nhớ ngoài.....	9
2.1.6. Bộ định thời/Bộ đếm	9
2.1.6.1. Thanh ghi chức năng đặc biệt cho bộ định thời.....	9
2.1.6.2. Các chế độ định thời	10
2.1.7. Giao tiếp nối tiếp (UART).....	11
2.1.7.1. Thanh ghi cho giao tiếp nối tiếp	11
2.1.7.2. Các chế độ hoạt động của cổng nối tiếp	12
2.1.8. Hoạt động ngắt	12
2.1.8.1. Thanh ghi đặc biệt cho ngắt.....	13

2.1.8.2.	Chương trình phục vụ ngắt	14
2.1.8.3.	Ngắt reset	14
2.1.8.4.	Ngắt bộ định thời	14
2.1.8.5.	Ngắt ngoài.....	15
2.1.8.6.	Ngắt cổng nối tiếp.....	15
2.1.8.7.	Mức ưu tiên ngắt của vi điều khiển 8051	15
2.2.	IC 74HC595.....	16
2.3.	Module bluetooth HC-05.....	17
2.3.1.	Các chế độ hoạt động	18
2.3.2.	Tập lệnh AT.....	18
Chương 3.	PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	20
3.1.	Tổng thể.....	20
3.2.	LED MATRIX 8x8.....	20
3.3.	Khối mạch nguồn.....	22
3.4.	Khối điều khiển	23
3.5.	Khối Bluetooth	26
3.6.	Kết nối các khối.....	26
3.7.	Lập trình hệ thống.....	27
3.7.1.	Tổng quan	27
3.7.2.	Lưu đồ giải thuật	28
3.7.3.	Chương trình.....	28
3.8.	Thiết kế mạch in	35
Chương 4.	KẾT QUẢ THỰC NGHIỆM.....	38
4.1.	Hiệu ứng 1	38

4.2. Hiệu ứng 2	40
4.3. Hiệu ứng 3	41
Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	43
5.1. Kết luận.....	43
5.1.1. Những tiêu chí đã đạt được	43
5.1.2. Những hạn chế, khó khăn khi thực hiện.....	43
5.2. Hướng phát triển.....	43

DANH MỤC HÌNH VẼ

Hình 2.1 IC AT89S51	4
Hình 2.2 Sơ đồ chân IC AT89S51	5
Hình 2.3 Sơ đồ chân IC 74HC595	16
Hình 2.4 IC SN74HC595N	16
Hình 2.5 Module Bluetooth HC-05.....	17
Hình 3.1 Sơ đồ khối của mạch	20
Hình 3.2 LED matrix 8x8 LD-788BS	21
Hình 3.3 Sơ đồ chân LED ma trận trên lí thuyết và thực tế.....	21
Hình 3.4 Sơ đồ kết nối 4 LED ma trận	22
Hình 3.5 Sơ đồ khối mạch nguồn sử dụng IC ổn áp 7805	22
Hình 3.6 IC ổn áp 7805	23
Hình 3.7 Sơ đồ khối điều khiển.....	24
Hình 3.8 Sơ đồ kết nối các nút nhấn và thạch anh	25
Hình 3.9 Sơ đồ tổng thể toàn mạch.....	26
Hình 3.10 Lưu đồ giải thuật của chương trình	28
Hình 3.11 PCB layout của mạch trên phần mềm Proteus 8.8.....	36
Hình 3.12 Mặt trên của mạch khi xuất ra file pdf	36
Hình 3.13 Mặt dưới của mạch khi xuất file pdf.....	37
Hình 3.14 Mạch sau khi hoàn thành	37
Hình 4.1 Mạch với module HC-05 được kết nối	38
Hình 4.2 Truyền ký tự A trên app	39
Hình 4.3 Kết quả chạy hiệu ứng 1.....	39
Hình 4.4 Truyền ký tự B trên app	40
Hình 4.5 Kết quả chạy hiệu ứng 2.....	41
Hình 4.6 Truyền ký tự C trên app	42
Hình 4.7 Kết quả chạy hiệu ứng 3.....	42

DANH MỤC BẢNG

Bảng 2.1 Các thanh ghi sử dụng trong việc định thời.....	9
Bảng 2.2 Các bit chức năng trong thanh ghi điều khiển TCON	9
Bảng 2.3 Thanh ghi chế độ định thời TMOD	10
Bảng 2.4 Các chế độ định thời	10
Bảng 2.5 Thanh ghi điều khiển cổng nối tiếp (SCON).....	11
Bảng 2.6 Các chế độ hoạt động của cổng nối tiếp	12
Bảng 2.7 Các bit chức năng của thanh ghi IE	13
Bảng 2.8 Các bit chức năng của thanh ghi IP	13
Bảng 2.9 Các cờ ngắt	13
Bảng 2.10 Vector ngắt.....	14
Bảng 2.11 Các mức ưu tiên ngắt	15
Bảng 3.1 Chương trình và giải thích	28

DANH MỤC TỪ VIẾT TẮT

MỞ ĐẦU

Vi xử lý ra đời đã đánh dấu một bước ngoặt lớn trong lịch sử phát triển của khoa học và công nghệ. Nó đóng vai trò quan trọng trong các thiết bị phục vụ đời sống và sản xuất hiện nay; đặc biệt trong quá trình công nghiệp hoá và hiện đại hoá đất nước. Việc kết hợp vi xử lý và các thiết bị ngoại vi có thể giúp con người giải quyết rất nhiều công việc trong sản xuất và đời sống: nhận lệnh từ người điều khiển thông qua bàn phím, nút nhấn, cảm biến, ... xử lý, tính toán, xuất kết quả, hiển thị kết quả, xuất tín hiệu điều khiển thiết bị, ...

Nhờ những khả năng mạnh mẽ của vi điều khiển mà nó được sử dụng rộng rãi. Tìm hiểu về vi xử lý và ứng dụng nó vào đề tài “Thiết kế mạch chạy chữ trên ma trận LED” là bước đầu trong việc tiếp cận và làm chủ kỹ thuật, công nghệ, cùng với đó là ứng dụng các kiến thức đã được học và thực hành để rèn luyện và nâng cao kỹ năng của sinh viên. Sai sót trong quá trình thực hiện đề tài là điều không thể tránh khỏi nên nhóm rất mong nhận được sự nhiệt tình góp ý từ thầy và các bạn để nhóm hoàn thiện hơn vốn kiến thức hiện có cũng như kỹ năng hiện thực hoá ý tưởng thành sản phẩm sau này.

Nhóm xin chân thành cảm ơn!

Chương 1. TỔNG QUAN

1.1. Đặt vấn đề

Các bộ vi điều khiển đang có ứng dụng ngày càng rộng rãi trong rất nhiều lĩnh vực kĩ thuật và đời sống xã hội. Hầu hết các thiết bị từ đơn giản đến phức tạp như: các thiết bị điều khiển tự động, thiết bị văn phòng, thiết bị gia đình, ... đều dùng đến các vi điều khiển.

Các cửa hàng, siêu thị, cơ sở sản xuất, ... đều cần có các biển quảng cáo bằng LED để trang trí cũng như để thu hút khách hàng. Qua đó cho thấy được vai trò và tầm quan trọng của ứng dụng vi điều khiển và LED, cụ thể là LED ma trận trong đời sống. Nhóm xin được trình bày đề tài sau: Thiết kế mạch chạy chữ trên LED ma trận. Sử dụng vi điều khiển 8051, cụ thể là AT89S51 để điều khiển 4 ma trận LED 8x8 để hiển thị các ký tự A-Z, 0-9. Đây là một đề tài khá phổ biến và dễ tiếp cận với mọi người.

1.2. Nhu cầu, ý tưởng

LED ma trận có rất nhiều công dụng như để chiếu sáng, quảng cáo, trang trí, ... Với ưu điểm là giá thành rẻ, dễ điều khiển, độ bền cao, tiết kiệm điện năng. Từ vấn đề được đặt ra ở phần trên, cùng với nhu cầu áp dụng các kiến thức đã được học ở môn Vi xử lý – Vi điều khiển nhóm đã lên ý tưởng và thiết kế ra hệ thống LED ma trận 8x32 để hiển thị các ký tự A-Z, 0-9 và có thể kết nối Bluetooth. Ở báo cáo này, nhóm chỉ thực hiện những gì đã được đề cập ở mục này.

1.3. Mục tiêu

Bằng kiến thức đã được học trong môn học *Vi xử lý – Vi điều khiển*, thực hiện mô phỏng trên công cụ Proteus, thiết kế mạch in cho đề tài và vận dụng lập trình hợp ngữ ASM. Đề tài sẽ giới thiệu về 8051, cách ghép nối các vi xử lý và các thiết bị ngoại vi, hiểu thêm về nguyên lý hoạt động của LED ma trận, IC ghi dịch 74HC595.

1.4. Kết luận

Một sản phẩm hỗ trợ việc quảng bá thương hiệu là rất quan trọng và có tính khả thi cao. Hơn nữa, với các đề tài ứng dụng này các kiến thức khoa học kỹ thuật sẽ không còn chỉ nằm trên giấy thay vào đó sẽ giúp ích nhiều hơn với mọi người và kinh nghiệm thu lại được từ việc vận dụng các kiến thức đã học sẽ trở nên có ích hơn và thực tế hơn. Vì vậy, đề tài “Thiết kế mạch hiển chữ trên ma trận LED” là một đề tài thực tế và hữu ích.

Chương 2. CƠ SỞ LÝ THUYẾT

2.1. IC AT89S51

2.1.1. Tổng quan

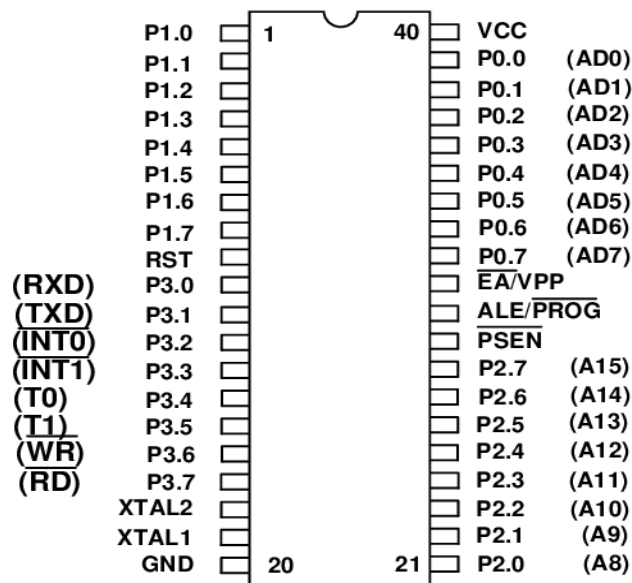


Hình 2.1 IC AT89S51

AT89S51 thuộc họ vi điều khiển 8051 và có một số đặc điểm cơ bản sau:

- Tương thích tập lệnh ASM của MCS – 51
- 4KB bộ nhớ trong
- 128 x 8bits RAM
- 32 cổng I/O có thể lập trình
- 2 bộ định thời/bộ đếm 16 bits
- 6 nguồn ngắt
- Có thể giao tiếp với bộ nhớ dữ liệu ngoài 64 KB
- Có thể giao tiếp với bộ nhớ lưu trữ ngoài 64KB
- Có thể xử lý theo bit, bộ nhớ có 210 bits được định địa chỉ theo bit
- Thực hiện phép nhân/chia trong 4 chu kỳ máy.

2.1.2. Cấu tạo, sơ đồ chân



Hình 2.2 Sơ đồ chân IC AT89S51

IC AT89S51 có 40 chân tín hiệu bao gồm 32 chân I/O được chia làm 4 cổng (P0, P1, P2, P3) mỗi cổng có 8 chân. Ngoài chức năng I/O, các cổng P0, P2, P3 còn chứa thêm một số chức năng khác tùy vào các chân của cổng.

- Port 0: các chân từ 32 đến 39 trên vi điều khiển, không có điện trở kéo lên bên trong. Được sử dụng làm chân I/O khi không kết nối bộ nhớ ngoài; làm bus dữ liệu và bus địa chỉ đa hợp khi kết nối bộ nhớ ngoài.
- Port 1: các chân từ 1 đến 9 trên vi điều khiển. Chỉ có chức năng I/O. Bên trong có điện trở kéo lên. Dùng làm 8 bits địa chỉ thấp trong quá trình nạp và kiểm tra chip.
- Port 2: các chân từ 21 đến 28. Có chức năng nhập xuất khi không kết nối bộ nhớ ngoài, dùng làm 8 bits địa chỉ cao trong 16 bits địa chỉ khi kết nối bộ nhớ ngoài. Dùng làm 8 bits địa chỉ cao hoặc tín hiệu điều khiển trong quá trình nạp hoặc kiểm tra chip.
- Port 3: các chân từ 10 đến 17. Ngoài chức năng nhập xuất, Port 3 còn có nhiều chức năng riêng biệt.

AT89S51 sử dụng nguồn DC +5V, cực dương nối vào chân 40, cực âm nối vào chân 20.

Chân PSEN (chân số 29) cho phép đọc bộ nhớ chương trình mở rộng đối với các chương trình sử dụng bộ nhớ ngoài.

Chân ALE (chân số 30), cho phép xuất tín hiệu chốt địa chỉ để giải đa hợp bus dữ liệu và bus địa chỉ ở port 1 trong quá trình giao tiếp với bộ nhớ ngoài.

Chân EA (Chân số 31) cho phép thực thi chương trình từ bộ nhớ chương trình ngoài. Khi sử dụng ROM nội thì chân EA nối với VCC, khi sử dụng ROM ngoài thì chân EA nối với GND.

Chân RESET (Chân số 9) dùng để thiết lập lại trạng thái ban đầu cho hệ thống. Khi chân này được nối với VCC trong ít nhất 2 chu kỳ máy thì chip được nạp các giá trị như khi khởi động hệ thống (ngoại trừ thanh ghi SBUF).

Chân XTAL1 và XTAL2 (chân 18 và chân 19) dùng để kết nối với bộ tạo xung thạch anh ngoài.

2.1.3. Kiến trúc bên trong

Kiến trúc bên trong của họ vi điều khiển 8051 bao gồm các thành phần sau:

- CPU để thực thi các lệnh
- OSC để tạo xung nhịp hoạt động cho vi điều khiển, bao gồm bộ tạo dao động nội và bộ tạo dao động ngoại.
- Cổng vào nối tiếp (UART)
- 2 timer
- RAM và các thanh ghi có chức năng đặc biệt
- ROM

2.1.4. Tổ chức bộ nhớ

2.1.4.1. Tổ chức bộ nhớ trong

Được thiết kế theo cấu trúc Harvard bao gồm bộ nhớ chương trình (ROM) và bộ nhớ dữ liệu (RAM).

ROM lưu trữ chương trình thực thi và không bị mất khi không cấp nguồn.

RAM lưu các kết quả tạm thời trong quá trình vi điều khiển hoạt động, dữ liệu trong RAM bị mất trong khi reset hệ thống hoặc không cấp nguồn.

2.1.4.2. Bộ nhớ chương trình

Chương trình khi được biên dịch thành các file HEX để nạp vào ROM tại các địa chỉ theo byte. Địa chỉ bắt đầu là 0000H, địa chỉ kết thúc là 0FFFFH. Vùng nhớ chương trình từ 0000H đến 002FH được dùng để chứa bản vector ngắt.

2.1.4.3. Bộ nhớ dữ liệu

Được chia thành nhiều vùng với các mục đích khác nhau: vùng RAM nội (00H đến 7FH) và vùng các thanh ghi có chức năng đặc biệt (80H đến FHH).

a. Vùng RAM nội

- Các bank thanh ghi (00H đến 1FH) được chia thành 4 bank thanh ghi: bank 0 từ 00H đến 07H, bank 1 từ 08H đến 0FH, bank 2 từ 10H đến 17H, bank 3 từ 18H đến 1FH.
- Vùng RAM định địa chỉ theo bit: địa chỉ từ 20H đến 2FH gồm 16 bytes có thể truy xuất theo byte hoặc bit bằng các lệnh xử lý bit. Các bit của vùng này có địa chỉ bit bắt đầu tại 00H và kết thúc tại 7FH. Địa chỉ byte bắt đầu 20H gồm 8 bits (từ 00H đến 70H), địa chỉ byte kết thúc có địa chỉ bit từ 78H đến 7FH.
- Vùng RAM đa dụng: có 80 bytes từ địa chỉ 30H đến 7FH có thể truy xuất theo byte bằng các chế độ địa chỉ trực tiếp hay gián tiếp. Dùng để lưu các dữ liệu trong quá trình hoạt động của vi điều khiển.

b. Vùng các thanh ghi có chức năng đặc biệt

- Thanh ghi tích lũy (A): có thể truy xuất trực tiếp thông qua địa chỉ 0E0H (theo byte) hay truy xuất theo từng bit thông qua địa chỉ từ E0H đến E7H.
- Thanh ghi phụ (B): dùng chung với thanh ghi A cho các phép nhân, chia và có thể dùng như một thanh ghi tạm để chứa các kết quả trung gian. Thanh ghi B có địa chỉ byte là F0H và cũng có thể truy cập theo từng bit với địa chỉ bit từ F0H đến F7H.
- Thanh ghi trạng thái chương trình (PSW): chứa các bit phản ánh trạng thái hiện tại của CPU.
- Thanh ghi con trỏ stack (SP): là thanh ghi 8 bits có địa chỉ 81H, chứa địa chỉ của dữ liệu nằm ở đỉnh stack.
- Thanh ghi con trỏ dữ liệu (DPTR) dùng để truy xuất bộ nhớ chương trình ngoài hoặc bộ nhớ dữ liệu ngoài. Là một thanh ghi 16 bits bao gồm 2 thanh ghi 8 bits có địa chỉ ở hai byte liên tiếp trong bộ nhớ là 82H và 83H.
- Các thanh ghi cổng vào ra: bao gồm thanh ghi cổng P0 (80H), cổng P1 (90H), cổng 2 (A0H) và cổng P3 (B0H).
- Thanh ghi bộ đệm dữ liệu nối tiếp (SBUF) địa chỉ 99H, giữ cả hai dữ liệu truyền và nhận.
- Thanh ghi định thời (Timer): Bộ định thời 0 gồm hai thanh ghi TH0 (8CH) chứa 8 bits cao và TL0 (8AH) chứa 8 bits thấp của timer 0; thanh ghi TH1 (8DH) chứa 8 bits cao và TL1 (8BH) chứa 8 bits thấp của timer 1.
- Các thanh ghi điều khiển: thanh ghi IP và IE điều khiển ngắt, thanh ghi TMOD và TCON điều khiển timer, thanh ghi SCON điều khiển cổng nối tiếp, thanh ghi PCON điều khiển nguồn.

2.1.5. Tổ chức bộ nhớ ngoài

- Bộ nhớ chương trình ngoài là một bộ nhớ chỉ đọc (ROM) được cho phép bởi tín hiệu PSEN.
- Bộ nhớ dữ liệu ngoài là bộ nhớ đọc/ghi (RAM) được cho phép hoạt động bởi các tín hiệu RD và WR của các chân P3.7 và P3.6
- Bộ nhớ dùng chung: Để giải quyết việc mỗi lần cần sửa đổi chương trình lại cần phải nạp lại chương trình từ đầu, chúng ta có thể sử dụng chung một vùng nhớ RAM ngoài cho cả chương trình và dữ liệu.

2.1.6. Bộ định thời/Bộ đếm

Trong vi điều khiển 8051 có hai timer 16 bits: timer 0 và timer 1; mỗi timer có 4 chế độ hoạt động.

2.1.6.1. Thanh ghi chức năng đặc biệt cho bộ định thời

Bảng 2.1 Các thanh ghi sử dụng trong việc định thời

SFR	Chức năng	Địa chỉ	Địa chỉ theo bit
TCON	Điều khiển	88H	Có
TMOD	Chọn chế độ	89H	Không
TL0	Byte thấp của timer 0	8AH	Không
TL1	Byte thấp của timer 1	8BH	Không
TH0	Byte cao của timer 0	8CH	Không
TH1	Byte cao của timer 1	8DH	Không

a. Thanh ghi điều khiển Timer (TCON)

Bảng 2.2 Các bit chức năng trong thanh ghi điều khiển TCON

BIT	Tên	Địa chỉ	Chức năng
7	TF1	8FH	Cờ báo tràn Timer 1, được lập bởi phần cứng khi có tràn timer 1, được xoá bởi phần mềm hoặc phần cứng khi phục vụ chương trình ngắt
6	TR1	8EH	Bit điều khiển hoạt động của Timer 1, được lập hoặc xoá bởi phần mềm để điều khiển Timer 1 hoạt động hay ngưng hoạt động

5	TF0	8DH	Cờ báo tràn của Timer 0
4	TR0	8CH	Bit điều khiển hoạt động của Timer 1
3	IE1	8BH	Được sử dụng để phục vụ ngắt
2	IT1	8AH	
1	IE0	89H	
0	IT0	88H	

b. Thanh ghi chế độ định thời (TMOD)

Bảng 2.3 Thanh ghi chế độ định thời TMOD

BIT	Tên	Timer	Chức năng
7	GATE	1	Bit điều khiển cổng. Khi GATE = 1, timer chỉ hoạt động khi INT1 ở mức cao
6	C/T	1	Bit chọn chức năng đếm hoặc định thời: 1-đếm sự kiện, 0-định thời
5	M1	1	Bit chọn chế độ timer 1
4	M0	1	Bit chọn chế độ timer 1
3	GATE	0	Bit điều khiển cổng. Khi GATE = 1, timer chỉ hoạt động khi INT0 ở mức cao
2	C//T	0	Bit chọn chức năng đếm hoặc định thời: 1-đếm sự kiện, 0-định thời
1	M1	0	Bit chọn chế độ timer 0
0	M0	0	Bit chọn chế độ timer 0

Bảng 2.4 Các chế độ định thời

M1	M0	Chế độ	Mô tả
0	0	0	Chế độ định thời 13 bit
0	1	1	Chế độ định thời 16 bit
1	0	2	Chế độ định thời tự động nạp lại 8 bit
1	1	3	Chế độ định thời chia tách: Timer 0: TLO là bộ định thời 8 bit điều khiển bởi chế độ của Timer 0; TH0 là bộ định thời 8 bit điều khiển bởi chế độ của Timer 1 Timer 1 ngưng hoạt động

2.1.6.2. Các chế độ định thời

- Chế độ 0: chế độ định thời 13 bit. Chế độ này hoạt động bằng cách ghép 5 bit thấp của thanh ghi TL1/TL0 và thanh ghi TH1/TH0 để tạo thành bộ định thời 13 bits.

- Chế độ 1: chế độ định thời 16 bit, hoạt động tương tự chế độ 0. Khi nhận được xung nhịp bộ định thời sẽ đếm lên từ giá trị chứa bên trong thanh ghi định thời, nếu thanh ghi định thời chưa được nạp thì bộ định thời sẽ đếm lên từ 0000H. Cờ tràn sẽ được bật lên 1 khi bộ định thời chuyển trạng thái từ FFFFH về 0000H, và bộ định thời sẽ tiếp tục đếm. Cờ tràn TF1/TF0 trong thanh ghi TCON được đọc hoặc ghi bằng phần mềm. Giá trị bắt đầu của bộ định thời được nạp vào các thanh ghi TH0/TH1 và TL0/TL1.
- Chế độ 2: chế độ định thời 8 bit tự nạp lại. Bộ định thời sẽ tự hoạt động sau khi cài đặt các thanh ghi TMOD và TH0/TH1 mà không cần phải nạp lại giá trị vào các thanh ghi định thời mỗi khi cờ tràn bật lên.
- Chế độ 3: chế độ định thời chia sẻ, trong chế độ này hai bộ định thời có các hoạt động khác nhau. Bộ định thời 0 ở chế độ 3 được chia thành hai bộ định thời 8 bits hoạt động độc lập: TL0 có cờ báo tràn là TF0, TH0 có cờ báo tràn TF1.

2.1.7. Giao tiếp nối tiếp (UART)

2.1.7.1. Thanh ghi cho giao tiếp nối tiếp

a. Thanh ghi bộ đệm cổng nối tiếp (SBUF)

Có địa chỉ là 99H bao gồm 2 bộ đệm. Khi cần truyền dữ liệu thì dữ liệu sẽ được ghi vào thanh ghi SBUF, khi đó dữ liệu từ thanh ghi SBUF sẽ được truyền ra thiết bị bên ngoài qua chân TxD, khi muốn nhận dữ liệu vào vi điều khiển thì chỉ cần đọc giá trị tại thanh ghi SBUF vì dữ liệu đã được nhận từ chân RxD.

b. Thanh ghi điều khiển cổng nối tiếp (SCON)

Bảng 2.5 Thanh ghi điều khiển cổng nối tiếp (SCON)

BIT	Ký hiệu	Địa chỉ	Chức năng
7	SM0	9FH	Bit chọn chế độ
6	SM1	9EH	Bit chọn chế độ
5	SM2	9DH	Bit chọn chế độ, cho phép truyền thông tin trong chế độ đa xử lý trong chế độ 2 và 3; RI sẽ không tác động nếu nhận bit thứ 9

4	REN	9CH	Cần được thiết lập để cho phép nhận dữ liệu
3	TB8	9BH	Khi TB8 = 1 thì bit thứ 9 sẽ được truyền trong chế độ 2 và 3
2	RB8	9AH	Khi RB8 = 1 thì bit thứ 9 sẽ được nhận tổng chế độ 2 và 3
1	TI	99H	Được lập khi kết thúc truyền 1 byte dữ liệu, có thể xoá bằng phần mềm
0	RI	98H	Được lập khi kết thúc nhận 1 byte dữ liệu, có thể xoá bằng phần mềm

2.1.7.2. Các chế độ hoạt động của cổng nối tiếp

Bảng 2.6 Các chế độ hoạt động của cổng nối tiếp

SM0	SM1	Chế độ	Chức năng	Tốc độ Baud
0	0	0	Thanh ghi dịch	Cố định (tần số /12)
0	1	1	8 bit UART	Thay đổi (theo giá trị timer)
1	0	2	9 bit UART	Cố định (Tần số /64)
1	1	3	9 bit UART	Thay đổi (theo giá trị timer)

- Chế độ 0: Việc truyền dữ liệu được bắt đầu bằng một lệnh ghi dữ liệu cần truyền vào thanh ghi SBUF. Dữ liệu sẽ được dịch ra ngoài qua chân RxD với các xung nhịp dịch bit được đưa ra trên chân TxD. Mỗi bit được truyền đi trong một chu kỳ máy.
- Chế độ 1: Thực hiện việc truyền nhận dữ liệu nối tiếp với mỗi ký tự dữ liệu, bắt đầu bằng bit start ở mức thấp, tiếp theo là 8 bit dữ liệu và cuối cùng là bit stop ở mức cao.
- Chế độ 2: Truyền/nhận dữ liệu bất đồng bộ 9 bit với tốc độ baud cố định bằng tần số chia cho 12 hoặc 64. 11 bit sẽ được truyền và nhận bao gồm 1 bit start, 8 bit dữ liệu, bit dữ liệu thứ 9 có thể lập trình được và 1 bit stop.
- Chế độ 3: Truyền/nhận bất đồng bộ 9 bit với tốc độ baud có thể thay đổi được bằng cách lập trình timer 1, tốc độ baud có thể lập trình được.

2.1.8. Hoạt động ngắt

Vi điều khiển có 6 nguồn ngắt: 1 ngắt reset có mức ưu tiên cao nhất, 2 nguồn ngắt ngoài, 2 nguồn ngắt từ 2 timer và 1 nguồn ngắt từ cổng nối tiếp.

2.1.8.1. Thanh ghi đặc biệt cho ngắt

a. Thanh ghi cho phép ngắt (IE – Interrupt Enable)

Bảng 2.7 Các bit chức năng của thanh ghi IE

BIT	Ký hiệu	Địa chỉ	Chức năng (1: cho phép, 0: cấm)
IE7	EA	AFH	Cho phép/cấm toàn bộ ngắt
IE6		AEH	Không sử dụng
IE5	ET2	ADH	Cho phép ngắt timer 2 (8052)
IE4	ES	ACH	Cho phép ngắt cổng nối tiếp
IE3	ET1	ABH	Cho phép tắt timer 1
IE2	EX1	AAH	Cho phép ngắt ngoài INT1
IE1	ET0	A9H	Cho phép ngắt timer 0
IE0	EX0	A8H	Cho phép ngắt ngoài INT0

b. Thanh ghi ưu tiên ngắt (IP – Interrupt Priority)

Bảng 2.8 Các bit chức năng của thanh ghi IP

BIT	Ký hiệu	Địa chỉ	Chức năng (1: cho phép, 0: cấm)
IP7			Không sử dụng
IP6			Không sử dụng
IP5	PT2	BDH	Ưu tiên ngắt Timer 2 (8052)
IP4	PS	BCH	Cho phép ngắt cổng nối tiếp
IP3	PT1	BBH	Ưu tiên ngắt Timer 1
IP2	PX1	BAH	Ưu tiên cho ngắt ngoài từ INT1
IP1	PT0	B9H	Ưu tiên cho ngắt Timer 0
IP0	PX0	B8H	Ưu tiên cho ngắt ngoài từ INT0

c. Thanh ghi chứa cờ ngắt

Bảng 2.9 Các cờ ngắt

Ngắt	Cờ	Bit cụ thể
INT0	IE0	TCON.1
INT1	IE1	TCON.3
Timer 0	TF0	TCON.5
Timer 1	TF1	TCON.7
Cổng nối tiếp	TI	SCON.1
Cổng nối tiếp	RI	SCON.3
Timer 2 (8052)	TF2	T2CON.7
Timer 2 (8052)	EXF2	T2CON.6

2.1.8.2. Chương trình phục vụ ngắt

Khi một ngắt xảy ra và được vi điều khiển chấp nhận, chương trình chính sẽ bị ngắt quãng và những hoạt động sau sẽ xảy ra:

- Hoàn tất việc thực thi lệnh hiện hành
- Cắt thanh ghi PC vào ngăn xếp
- Trạng thái ngắt hiện hành được lưu lại
- Các ngắt được chặn lại ở mức ngắt
- Bộ đếm chương trình PC được nạp địa chỉ vector của mình phục vụ ngắt ISR
- ISR được thực thi

Bảng 2.10 Vector ngắt

Ngắt	Cờ	Địa chỉ ngắt
Reset	RST	0000H
INT0	IE0	0003H
Timer0	TF0	000BH
INT1	IE1	0013H
Timer1	TF1	001BH
Cổng nối tiếp	RI hoặc TI	0023H
Timer2 (8052)		

2.1.8.3. Ngắt reset

Khi ngắt reset xảy ra, chương trình sẽ chạy lại ở vị trí 0000H và đồng thời reset toàn bộ các thanh ghi và nội dung RAM

2.1.8.4. Ngắt bộ định thời

Các ngắt này xảy ra khi các thanh ghi Timer tràn và lập cờ báo tràn lên 1. Khi chương trình phục vụ ngắt được gọi thì cờ báo tràn sẽ tự động xóa bằng phần cứng.

2.1.8.5. Ngắt ngoài

Xảy ra khi có mức thấp hoặc có cạnh xuống kích hoạt lên chân INT0 (P3.2) hoặc INT1 (P3.3). Việc chọn ngắt tích cực mức thấp hoặc tích cực cạnh xuống được lập trình thông qua các bit IT0 và IT1 trong thanh ghi TCON.

a. Ngắt theo mức

Để cài đặt sử dụng ngắt ngoài theo mức, thì phải nạp IT0/IT1 = 0 và khi chân INT1/INT0 đang ở mức cao ít nhất một chu kỳ máy được chuyển xuống mức thấp liên tục 4 chu kỳ máy thì một ngắt theo mức sẽ được xác lập.

b. Ngắt ngoài theo cạnh

Ngắt theo mức cần nạp IT0/IT1 = 1 và khi chân INT0/INT1 đang ở mức cao một chu kỳ máy xuống mức thấp 1 chu kỳ máy thì ngắt được xác lập trong khoảng thời gian xuống đó và bật bit IE1/IE0 lên 1 sau đó bỏ qua các sườn xuống khác.

2.1.8.6. Ngắt cổng nối tiếp

Ngắt cổng nối tiếp xảy ra khi cờ ngắt truyền TI hoặc cờ ngắt nhận RI bằng 1. Một ngắt truyền được gọi khi hoàn tất việc truyền một ký tự trong thanh ghi SBUF. Một ngắt nhận được gọi khi một ký tự nhận được nhận đầy đủ và đang ở trong SBUF để chờ được vi điều khiển đọc. Việc ngắt truyền xảy ra khi bộ đệm truyền SBUF rỗng và ngắt nhận xảy ra khi bộ đệm nhận SBUF đầy.

2.1.8.7. Mức ưu tiên ngắt của vi điều khiển 8051

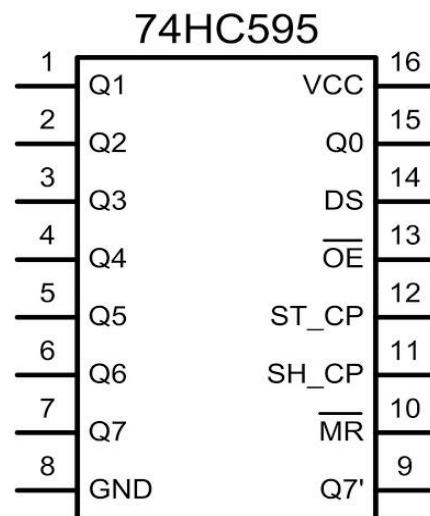
Bảng 2.11 Các mức ưu tiên ngắt

Mức ưu tiên ngắt	Ngắt	Cờ
1	Reset	
2	Ngắt ngoài 0	INT0
3	Ngắt bộ định thời 0	TF0

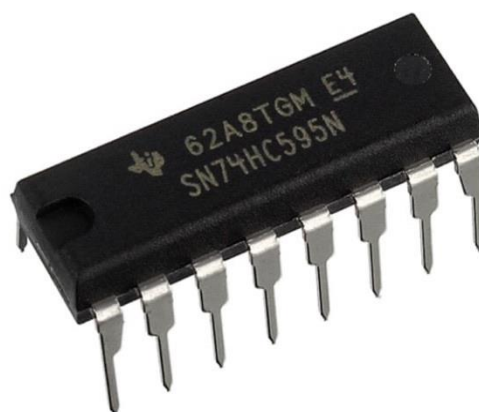
4	Ngắt ngoài 1	INT1
5	Ngắt bộ định thời 1	TF1
6	Ngắt truyền nối tiếp	RI hoặc TI

Tuy nhiên có thể thay đổi độ ưu tiên các ngắt trên (trừ reset) bằng cách gán mức ưu tiên cao hơn cho bất kỳ ngắt nào bằng cách lập trình bật bit tương ứng trong thanh ghi IP.

2.2. IC 74HC595



Hình 2.3 Sơ đồ chân IC 74HC595



Hình 2.4 IC SN74HC595N

74HC595 là IC ghi dịch 8 bit kết hợp chốt dữ liệu. Đầu vào nối tiếp, đầu ra song song.

- Chân 15 và 1 đến 7: các chân xuất dữ liệu
- Chân 14: chân vào dữ liệu nối tiếp
- Chân 13 chân cho phép tích cực mức thấp. Khi ở mức cao, tất cả đầu ra của IC trở về trạng thái cao.
- Chân 9: chân dữ liệu nối tiếp
- Chân 11: chân vào xung clock
- Chân 12: chân chốt dữ liệu
- Chân 10: Chân reset tích cực mức thấp

Hoạt động:

- Tại một thời điểm xung clock ở chân 11 ở cạnh lên thì 1 bit ở chân 14 sẽ được đưa vào IC dịch. Chiều dịch dữ liệu là từ Q0 – Q7.
- Khi muốn chốt dữ liệu đầu ra ta chỉ cần kéo chân 12 từ 0 thành 1 thì dữ liệu sẽ được chốt ở các chân dữ liệu ra.

2.3. Module bluetooth HC-05



Hình 2.5 Module Bluetooth HC-05

Được thiết kế dựa trên chip BC417

HC-05 gồm có:

- Key (hoặc EN): dùng để chọn chế độ AT mode hoặc Data mode.
- Vcc: Chân cấp nguồn DC từ 3.6 đến 6 V.
- Gnd: Chân nối đất
- State: không dùng đến trong trường hợp này

2.3.1. Các chế độ hoạt động

HC – 05 có 2 chế độ hoạt động là Command Mode và Data Mode.

- Chế độ Command Mode: Ta có thể giao tiếp với HC-05 thông qua cổng serial trên module bằng tập lệnh AT.
- Chế độ Data Mode: ở chế độ này module có thể truyền và nhận dữ liệu tới module khác.

Nếu đưa chân EN lên mức cao trước khi cấp nguồn sẽ đưa vào chế độ Command Mode với baud rate mặc định là 38400.

Nếu module chưa được thiết lập thì thiết lập mặc định của nó như sau:

- Baud rate: 9800, data 8bits, 1 bit stop, parity: none, handshake: none
- Password: 1234 hoặc 0000
- Tên thiết bị: HC-05

2.3.2. Tập lệnh AT

Một số lệnh AT cơ bản:

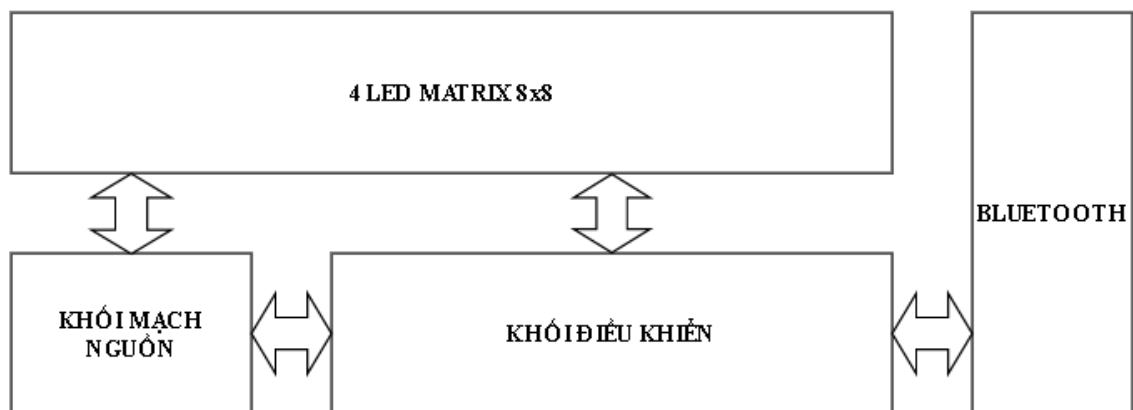
- AT: kiểm tra kết nối
- AT+ORGL: reset lại thiết lập mặc định
- AT+UART?: kiểm tra baudrate hiện tại
- AT+UART=9600,0,0: thiết lập baudrate mới là 9600 cho module
- AT+NAME?: kiểm tra tên hiện tại của module
- AT+NAME=<tên>: đặt tên cho module

- AT+PSWD?: Kiểm tra password hiện tại
- A+PSWD=<mật khẩu>: Đặt lại mật khẩu cho module

Chương 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1. Tổng thể

Dựa theo vấn đề được đặt ra là: “Thiết kế mạch chạy chữ trên ma trận LED”, cùng với việc thực hiện kết nối bluetooth để điều khiển thiết bị từ xa, ta có sơ đồ các khối chức năng như hình sau:

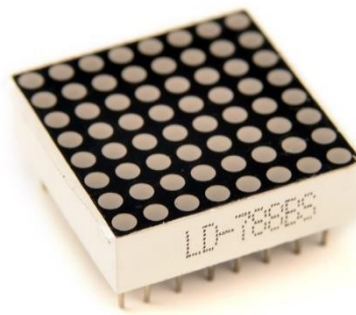


Hình 3.1 Sơ đồ khối của mạch

3.2. LED MATRIX 8x8

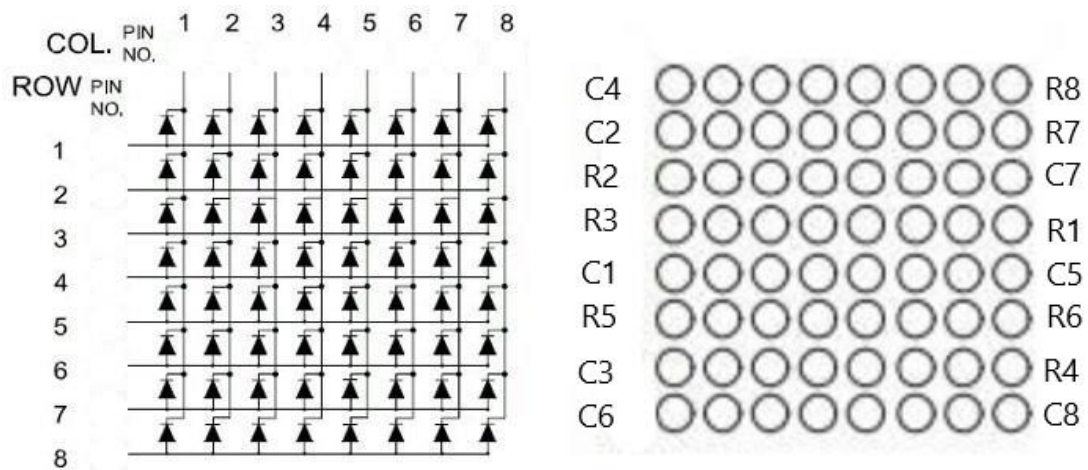
LED ma trận là ma trận diode hai chiều (mỗi điểm là một LED) có cực dương sắp theo hàng và cực âm sắp theo cột. Có thể điều khiển từng điểm của LED bằng cách điều khiển dòng điện đi qua mỗi cặp diode theo cột hoặc hàng.

Trong ma trận LED 8x8 có 64 đèn LED. Những LED này được hàn trên bảng mạch một mặt. Cực dương của LED này được nối với cực dương của LED kia thành một hàng, 8 hàng tương tự như vậy. Cực âm của LED này được nối với cực âm của LED kia thành 8 cột.



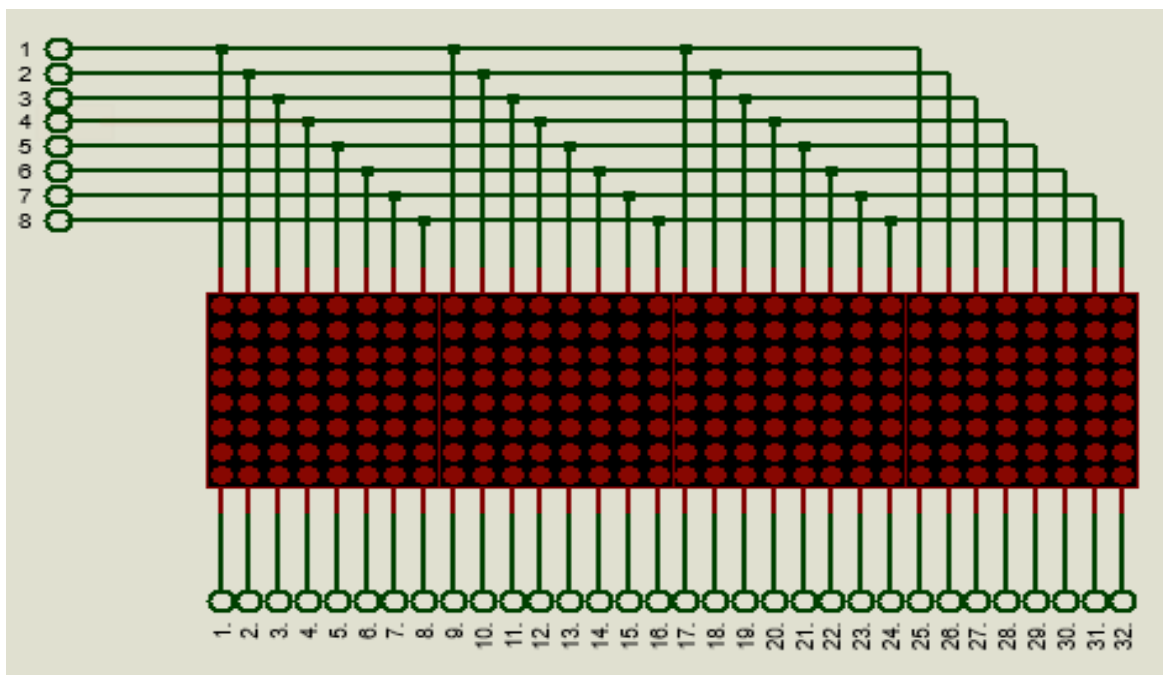
Hình 3.2 LED matrix 8x8 LD-788BS

Hình 3.3 cho ta sơ đồ chân trên lý thuyết và thực tế của LED ma trận 8x8.



Hình 3.3 Sơ đồ chân LED ma trận trên lý thuyết và thực tế

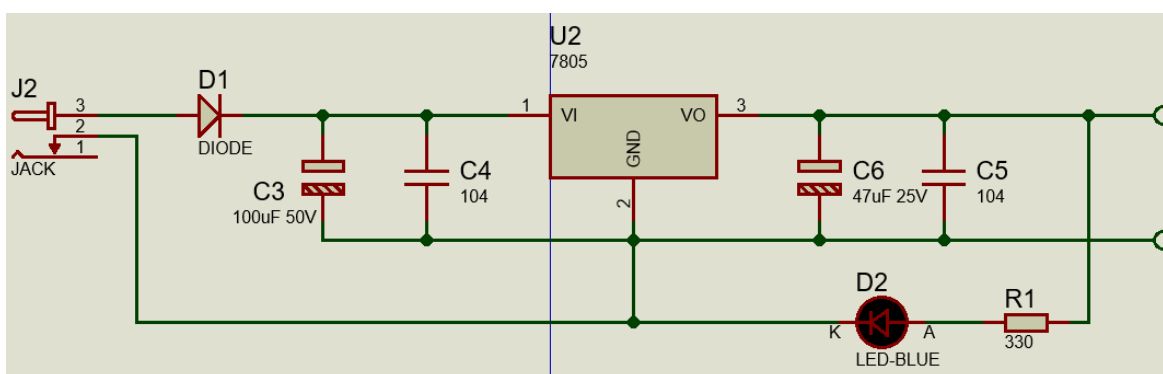
Trong mạch này ta kết hợp 4 ma trận LED 8x8 với nhau được một ma trận LED 8x64 và được kết nối với khối điều khiển như sau:



Hình 3.4 Sơ đồ kết nối 4 LED ma trận

- Các chân 1 đến 8 lần lượt nối với các chân từ P2.0 đến P2.1 của AT89S51
- Các chân 1. đến 32. lần lượt nối với các chân đầu ra dữ liệu của 4 IC 74HC595

3.3. Khối mạch nguồn



Hình 3.5 Sơ đồ khối mạch nguồn sử dụng IC ổn áp 7805

Khối mạch nguồn sử dụng IC ổn áp 7805 để tạo nguồn +5V cho toàn bộ hệ thống.



Hình 3.6 IC ổn áp 7805

Ta cấp điện áp đầu vào qua J2 và sẽ nhận được điện áp đầu ra +5V ở chân đầu ra Vo.

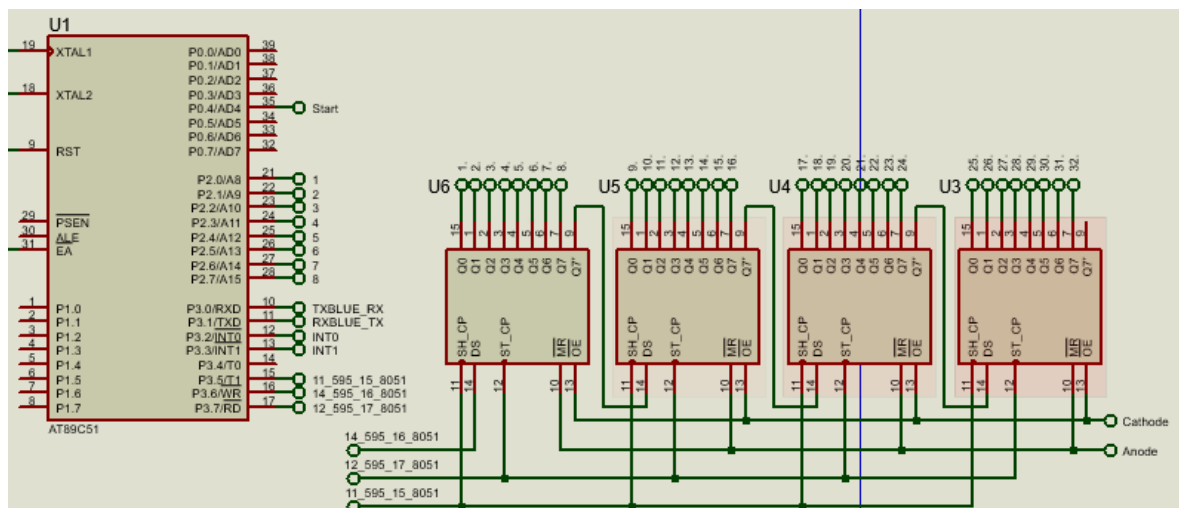
Tụ C3 và C4 có tác dụng lọc điện áp đầu vào cấp cho chân Vi của IC 7805. Tụ C3 có tác dụng cung cấp điện áp tạm thời cho chân Vi khi nguồn đột ngột bị sụt áp. Tụ C4 là tụ gốm có trở kháng lớn có tác dụng ngăn nguồn đầu vào tăng áp đột ngột.

Tụ C5 và C6 có tác dụng lọc điện áp đầu ra cho tải tiêu thụ, điện áp đầu ra được lấy từ chân Vo của IC 7805. Tụ C6 có tác dụng cung cấp điện áp tạm thời cho tải khi điện áp tải đột ngột bị sụt áp. Tụ C6 có trở kháng cao có tác dụng lọc nhiễu điện áp đầu ra.

Diode D1 có tác dụng chống dòng điện ngược. LED báo nguồn D2 có điện trở 330 Ohm bảo vệ, báo hệ thống đã được cấp nguồn.

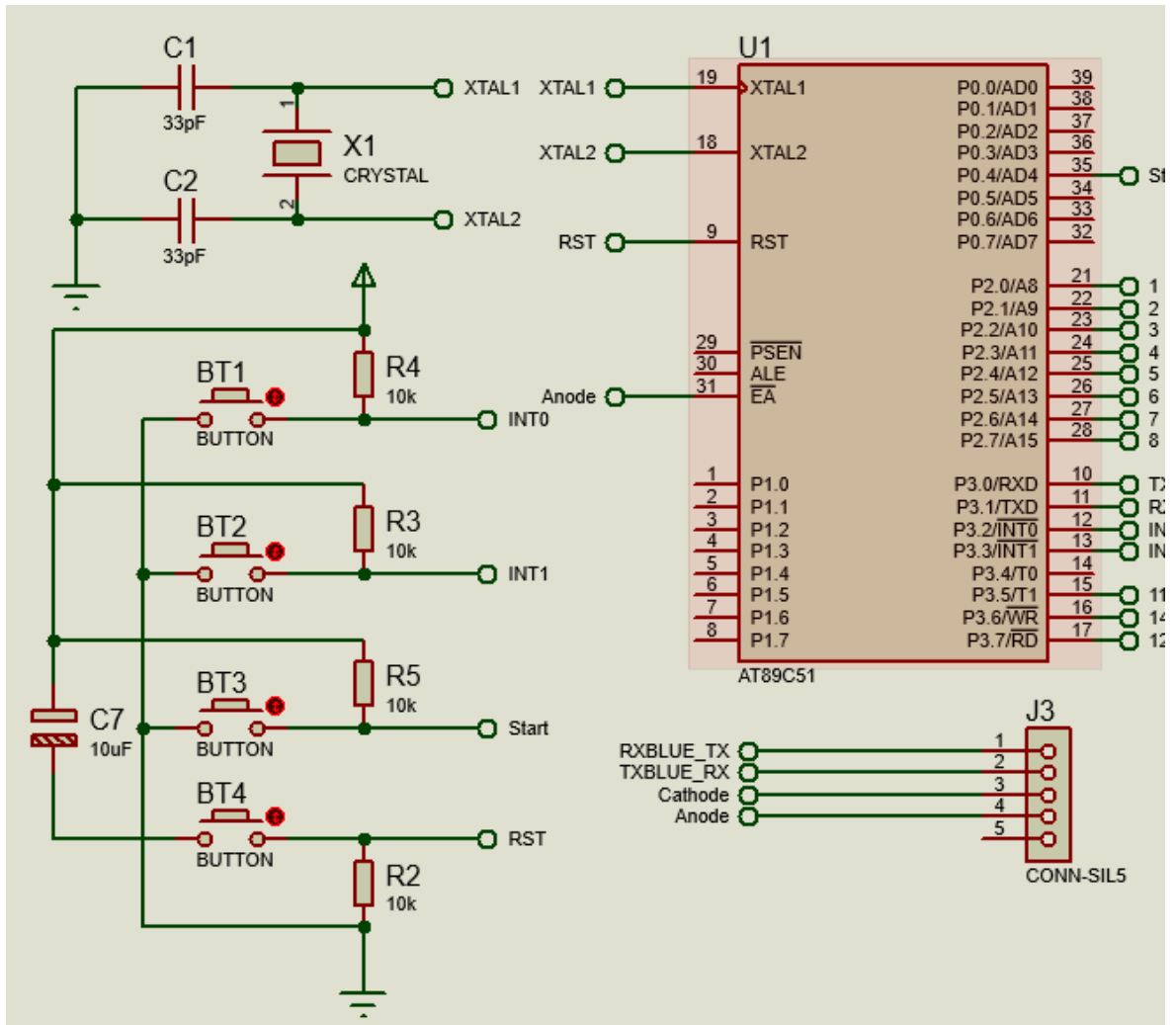
3.4. Khối điều khiển

Khối điều khiển gồm 1 IC AT89S51 và 4 IC 74HC595 làm nhiệm vụ điều khiển toàn bộ hoạt động của mạch được kết nối theo hình sau:



Hình 3.7 Sơ đồ khối điều khiển

- Chân 10 của 4 IC 74HC595 nối chung với nhau và nối về GND
- Chân 13 của 4 IC 74HC595 nối chung với nhau và nối lên VCC +5V
- Chân 14 của U6 nối với chân 16 của AT89S51 làm nhiệm vụ truyền dữ liệu từ AT89S51 đến 4 IC ghi dịch.
- Chân 11 của 4 IC 74HC595 nối chung với nhau và nối về chân 15 của AT89S51 làm chân vào xung clock cho IC 74HC595 hoạt động.
- Chân 12 của 4 IC 74HC595 nối chung với nhau và nối về chân 17 của AT89S51 làm chân chốt dữ liệu.
- Các chân ở Port 2 của AT89S51 nối với hàng của LED ma trận.
- Các chân từ Q0 đến Q7 (chân 15, 1, 2, 3, 4, 5, 6, 7) của IC 74HC595 nối với các cột của LED ma trận.
- Chân 35 nhận tín hiệu bắt đầu.



Hình 3.8 Sơ đồ kết nối các nút nhấn và thạch anh

- Các chân XTAL1 và XTAL2 nối với thạch anh X1 = 11.0592MHz và 2 tụ gốm 33pF.
- Có 4 nút nhấn BT1 nối với INT0, BT2 nối với INT1, BT3 nối với chân Start (chân 35), BT4 nối với chân Reset (chân 9) của AT89S51
- 3 trở kéo lên R3, R4, R5 lần lượt sử dụng cho các chân INT0, INT1 và chân Start; trở kéo xuống giúp chân Reset (Chân 9) luôn giữ ở mức thấp khi không nhấn nút và tụ hoá C7 giúp tín hiệu reset không bị nhiễu.
- Chân EA của vi điều khiển nối với VCC
- J3 giúp kết nối các chân của module bluetooth HC-05 với vi điều khiển

3.7. Lập trình hệ thống

3.7.1. Tổng quan

Dựa theo *Chương 1*, mạch sẽ hiển thị các chuỗi ký tự như sau:

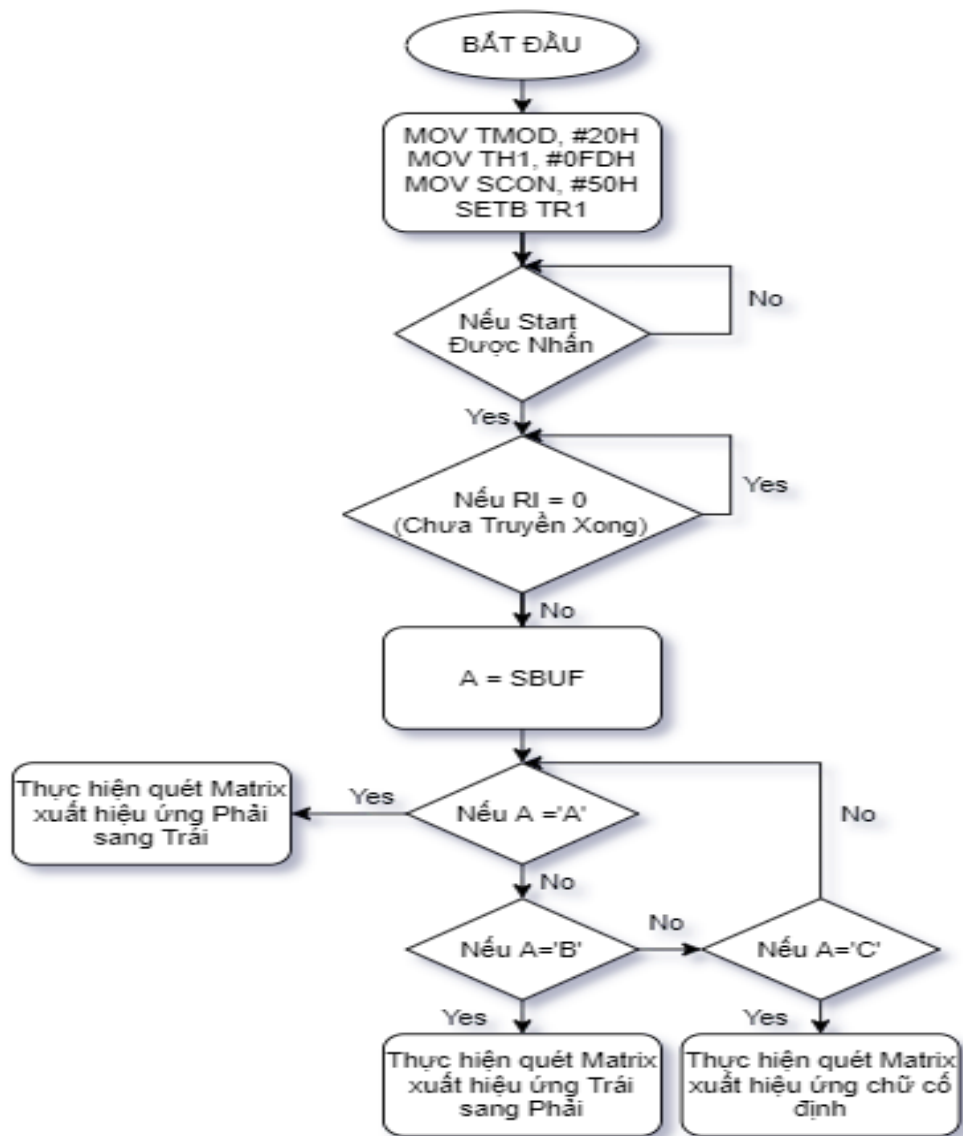
- Hiệu ứng 1: ĐH CNTT – KTMT VXL XDK (Đại học Công nghệ thông tin – Kỹ thuật máy tính Vi xử lý Vi điều khiển)
- Hiệu ứng 2: MINH TAN – QUOC DANG (Minh Tân – Quốc Đăng)
- Hiệu ứng 3: -GG- (Tên nhóm: GG)

Hệ thống thực hiện kết nối bluetooth với smartphone và được điều khiển bằng phần mềm *Bluetooth Terminal HC-05*

- Khi nhập A: hiệu ứng 1
- Khi nhập B: hiệu ứng 2
- Khi nhập C: hiệu ứng 3

Trước mỗi hiệu ứng cần nhấn nút Start.

3.7.2. Lưu đồ giải thuật



Hình 3.10 Lưu đồ giải thuật của chương trình

3.7.3. Chương trình

Bảng 3.1 Chương trình và giải thích

Code	Giải thích
ORG 000H	
CLK BIT P3.5	Khởi tạo nhãn CLK cho chân P3.5
DL_LOAD BIT P3.6	Khởi tạo nhãn DL_LOAD cho chân P3.6

<p>CHOT_DL BIT P3.7 DTOUT EQU P2</p> <p>MOV TMOD, #20H MOV TH1, #0FDH MOV SCON, #50H SETB TR1</p> <p>WAIT: JB P0.4, WAIT</p> <p>AGAIN: CLR RI</p> <p>REPEAT: JNB RI, REPEAT MOV A, SBUF CJNE A, #'A', CHECKNEXT LJMP PHAISANGTRAI</p> <p>CHECKNEXT: CJNE A, #'B', CHECKNEXT1 LJMP TRAISANGPHAI</p> <p>CHECKNEXT1: CJNE A, #'C', AGAIN LJMP SHOW</p> <p>PHAISANGTRAI: MOV A, #0 MOV DPTR, #ARR COUNT:</p>	<p>Khởi tạo nhãn CHOT_DL cho chân P3.7 Khởi tạo nhãn DTOUT cho port 2</p> <p>0010 0000: Timer 1 hoạt động Tốc độ baud 9600 Khởi tạo MOD 1 cho UART Timer 1 Bắt đầu</p> <p>Vòng lặp cho đến khi nút Start được nhấn (P0.4 = 0)</p> <p>Nhãn AGAIN: RI = 0</p> <p>Vòng lặp while (RI = 0) A = SBUF Nếu A = 'A' thì nhảy đến PHAISANGTRAI, ngược lại nhảy đến CHECKNEXT</p> <p>Nhãn CHECKNEXT Nếu A = 'B' thì nhảy đến TRAISANGPHAI, ngược lại nhảy đến CHECKNEXT1</p> <p>Nhãn CHECKNEXT1 Nếu A = 'C' thì nhảy đến SHOW, ngược lại nhảy đến AGAIN</p> <p>Nhãn PHAISANGTRAI A = 0 DPTR trở đến Vị trí ARR Nhãn COUNT:</p>
--	--

<p>MOV R0,#3</p> <p>LOOP:</p> <p>LCALL DISPLAY</p> <p>DJNZ R0,LOOP</p> <p>INC DPTR</p> <p>INC A</p> <p>CJNE A,#175, COUNT</p> <p>LJMP PHAISANGTRAI</p> <p>DISPLAY:</p> <p>PUSH ACC</p> <p>PUSH 00H</p> <p>MOV R0,#00H</p> <p>SETB DL_LOAD</p> <p>SCAN:</p> <p>MOV A,R0</p> <p>MOVC A,@A+DPTR</p> <p>MOV DTOUT,A</p> <p>CLR CLK</p> <p>SETB CLK</p> <p>CLR CHOT_DL</p> <p>SETB CHOT_DL</p> <p>LCALL DELAY</p> <p>MOV DTOUT,#0FFH</p> <p>CLR DL_LOAD</p> <p>INC R0</p> <p>CJNE R0,#32,SCAN</p> <p>POP 00H</p> <p>POP ACC</p> <p>RET</p>	<p>R0 = 3</p> <p>Nhãn LOOP:</p> <p>Gọi hàm DISPLAY</p> <p>Nếu R0 != 0 nhảy đến LOOP và giảm R0</p> <p>Tăng DPTR</p> <p>Tăng A</p> <p>Nhảy đến COUNT nếu A không bằng 175</p> <p>Nhảy đến PHAISANGTRAI</p> <p>Nhãn DISPLAY</p> <p>Push A vào Stack</p> <p>Push R0 vào Stack</p> <p>R0 = 0</p> <p>DL_LOAD = 1</p> <p>Nhãn SCAN:</p> <p>A = R0</p> <p>A = *(A +DPTR)</p> <p>DTOUT = A</p> <p>CLK = 0</p> <p>CLK = 1</p> <p>CHOT_DL = 0</p> <p>CHOT_DL = 1</p> <p>Gọi hàm DELAY</p> <p>DTOUT = FFH</p> <p>DTOUT = 0</p> <p>Tăng R0</p> <p>Nếu R0 khác 32 nhảy đến SCAN</p> <p>POP từ stack ra R0</p> <p>POP từ stack ra ACC</p> <p>Quay về vị trí gọi hàm</p>
---	---

<pre> TRAISANGPHAI: MOV A, #0 MOV DPTR,#ARR1 COUNT_1: MOV R0,#3 LOOP_1: LCALL DISPLAY_1 DJNZ R0,LOOP_1 INC DPTR INC A CJNE A,#160,COUNT_1 LJMP TRAISANGPHAI DISPLAY_1: PUSH ACC PUSH 00H MOV R0,#32H SETB DL_LOAD SCAN_1: MOV A,R0 MOVC A,@A+DPTR MOV DTOUT,A CLR CLK SETB CLK CLR CHOT_DL SETB CHOT_DL LCALL DELAY MOV DTOUT,#0FFH CLR DL_LOAD DJNZ R0,SCAN_1 POP 00H </pre>	<p>Tương tự PHAISANGTRAI</p>
--	------------------------------

<pre> POP ACC RET SHOW: MOV A, #0 MOV DPTR, #ARR2 COUNT_2: MOV R0, #3 LOOP_2: LCALL DISPLAY_2 DJNZ R0, LOOP_2 INC A CJNE A, #175, COUNT_2 LJMP SHOW DISPLAY_2: PUSH ACC PUSH 00H MOV R0, #00H SETB DL_LOAD SCAN_2: MOV A, R0 MOVC A, @A+DPTR MOV DTOUT, A CLR CLK SETB CLK CLR CHOT_DL SETB CHOT_DL LCALL DELAY MOV DTOUT, #0FFH CLR DL_LOAD INC R0 </pre>	
---	--

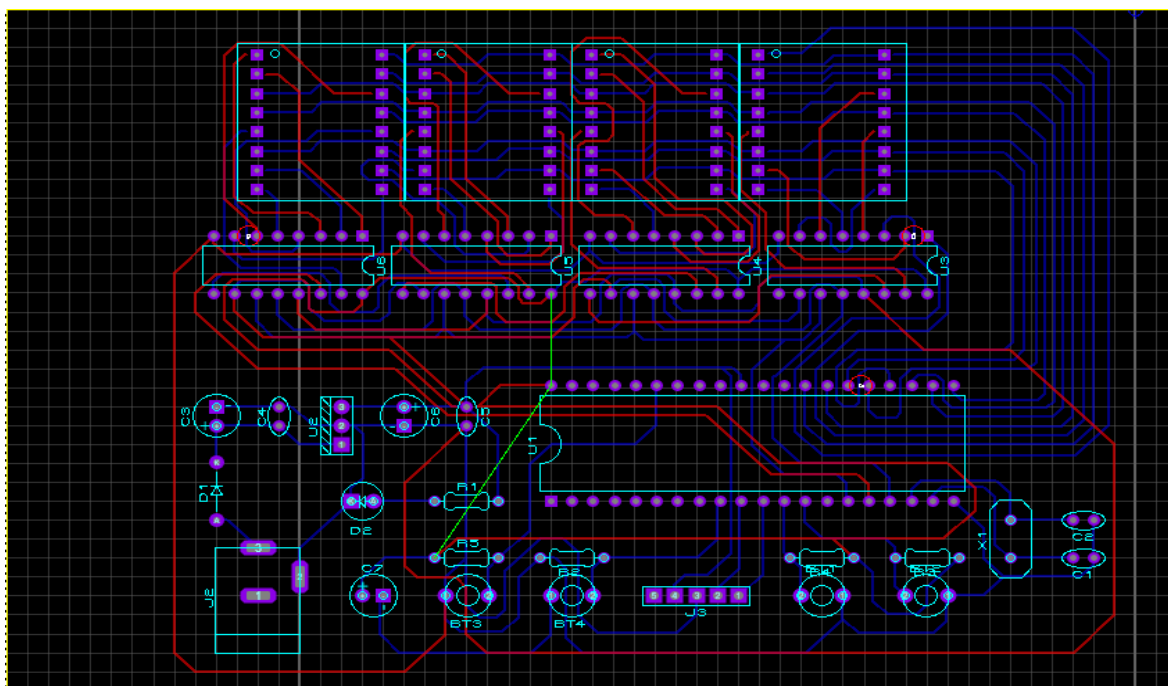
<p>CJNE R0,#32,SCAN_2</p> <p>POP 00H</p> <p>POP ACC</p> <p>RET</p> <p>DELAY:</p> <p>MOV R7,#2</p> <p>KT1: MOV R6,#247</p> <p>KT2: DJNZ R6,KT2</p> <p>DJNZ R7,KT1</p> <p>RET</p> <p>ARR:</p> <p>DB 0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH</p> <p>DB 0FFH,01H,01H,7DH,7DH,01H,83H, 0FFH, 0FFH,01H,01H,0EFH,0EFH,01H, 01H, 0FFH,0FFH,0FFH ;DH</p> <p>DB 0FFH,83H,01H,7DH,39H,0BBH,0FFH, 0FFH,01H,01H,0F3H,0E7H,0CFH,01H, 01H,0FFH,0F9H,0FDH,01H,01H,0FDH, 0F9H,0FFH,0FFH,0F9H,0FDH,01H,01H, 0FDH,0F9H,0FFH ;CNTT</p> <p>DB 0FFH,0E7H,0E7H,0E7H,0E7H,0FFH</p> <p>DB 0FFH,01H,01H,0C7H,93H,39H,7DH, 0FFH,0FFH,0F9H,0FDH,01H,01H,0FDH, 0F9H,0FFH,0FFH,01H,01H,0F3H,0E7H,</p>	<p>Hàm delay 494 chu kỳ</p> <p>Khởi tạo dữ liệu ARR</p>
---	---

<p>0F3H,01H,01H,0FFH,0F9H,0FDH,01H, 01H,0FDH,0F9H,0FFH</p> <p>DB 0FFH,0E7H,0E7H,0E7H,0E7H,0FFH</p> <p>DB 0FFH,0C1H,81H,3FH,3FH,81H,0C1H, 0FFH,0FFH,39H,11H,0C7H,0EFH,0C7H, 11H,39H,0FFH,01H,01H,7FH,7FH,7FH, 7FH,0FFH,0FFH,0FFH,0FFH,0FFH,0C1H, 81H,3FH,3FH,81H,0C1H,0FFH,0FFH,01H, 01H,7DH,7DH,01H,83H,0FFH,0FFH,01H, 01H,0C7H,93H,39H,7DH,0FFH</p> <p>DB 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH, 0FFH,0FFH</p> <p>ARR1:</p> <p>DB 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH, 0FFH,0FFH</p> <p>DB 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH,9BH,19H, 5DH,7DH,01H,83H,0FFH,01H,01H,0CFH, 0E7H,0F3H,01H,01H,0FFH, 0FFH,03H,01H,0EDH,01H,03H,0FFH, 0FFH,083H,01H,7DH,7DH,01H,01H,0FFH, 0FFH,0FFH,0FFH</p> <p>DB 0FFH,0BBH,39H,7DH,7DH,01H,83H, 0FFH, 0FFH,83H,01H,7DH,7DH,01H,83H, 0FFH, 0FFH,01H,01H,7FH,7FH,01H,81H,</p>	<p>Khởi tạo dữ liệu ARR1</p>
---	------------------------------

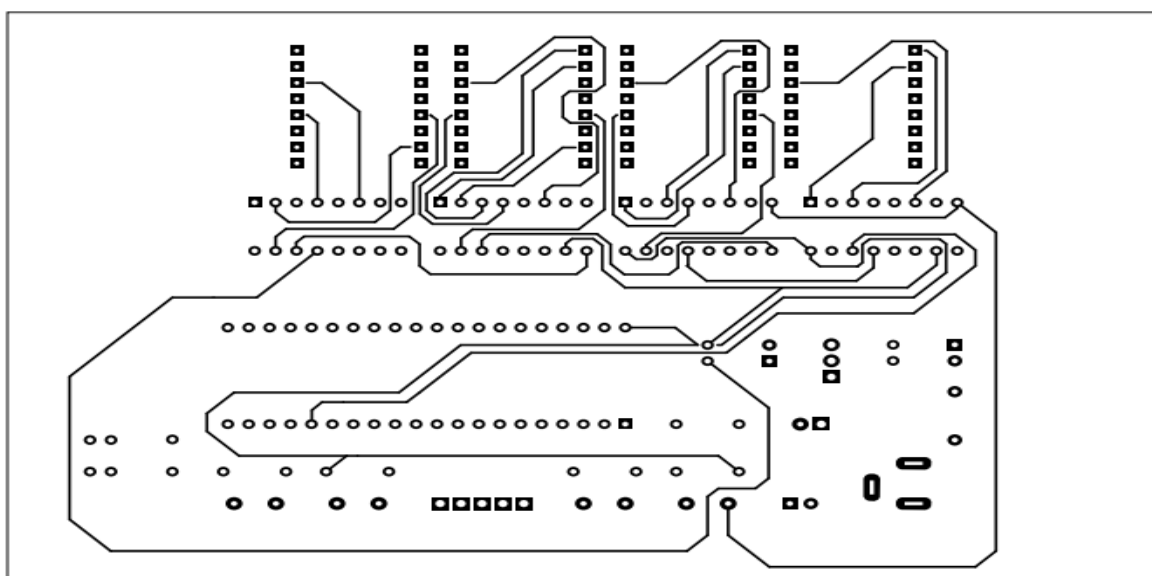
0FFH, 0FFH,43H,01H,9DH,0BDH,81H, 0C3H,0FFH DB 0FFH,0E7H,0E7H,0E7H,0E7H,0FFH DB 01H,01H,0CFH,0E7H,0F3H,01H,01H, 0FFH, 0FFH,03H,01H,0EDH,01H,03H, 0FFH, 0FFH,0F9H,0FDH,01H,01H,0FDH, 0F9H,0FFH, 0FFH,0FFH,0FFH DB 0FFH,01H,01H,0EFH,0EFH,01H,01H, 0FFH, 01H,01H,0CFH,0E7H,0F3H,01H,01H, 0FFH, 0FFH,0FFH,7DH,01H,01H,7DH,0FFH, 0FFH, 01H,01H,0F3H,0EFH,0F3H,01H,01H, 0FFH DB 0FFH,0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH, 0FFH, 0FFH,0FFH,0FFH, 0FFH, 0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH ARR2: DB 0FFH,0FFH,0E7H,0E7H,0E7H,0E7H, 0FFH,0FFH, 0FFH,83H,01H,7DH,5DH,19H, 9BH,0FFH, 0FFH,83H,01H,7DH,5DH,19H, 9BH,0FFH, 0FFH,0FFH,0E7H,0E7H,0E7H, 0E7H,0FFH, 0FFH,0FFH,0FFH,0FFH, 0FFH,0FFH, 0FFH,0FFH END	Khởi tạo dữ liệu ARR2 Kết thúc chương trình
--	--

3.8. Thiết kế mạch in

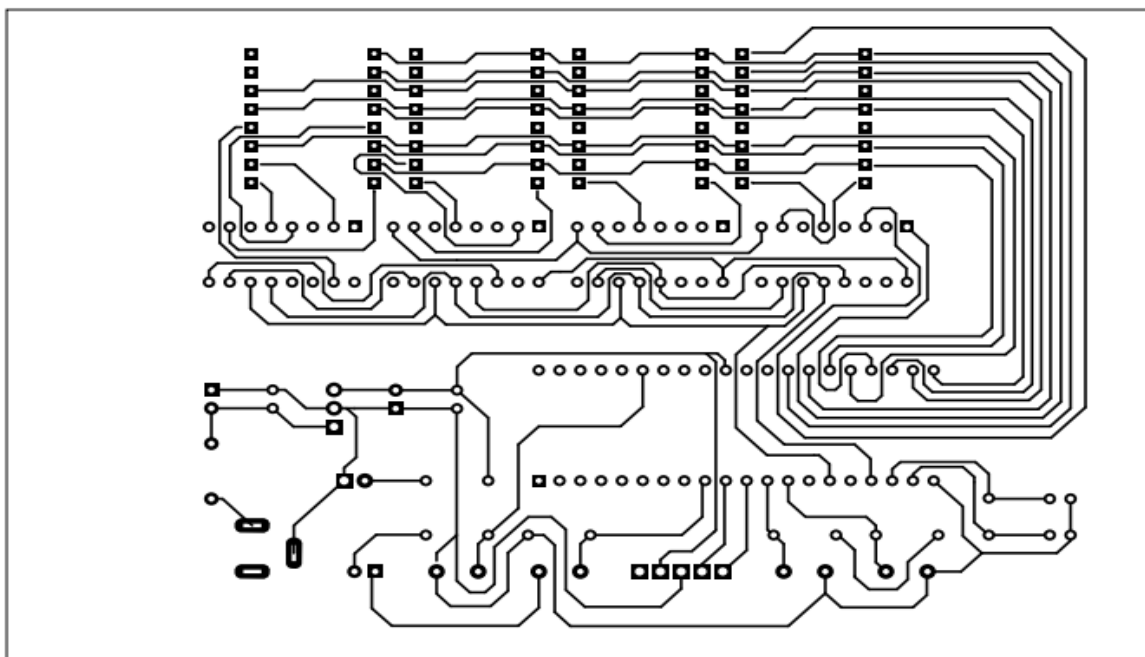
Sử dụng phần mềm Proteus 8.8 để vẽ mạch, in mạch và thực hiện ủ mạch thủ công.



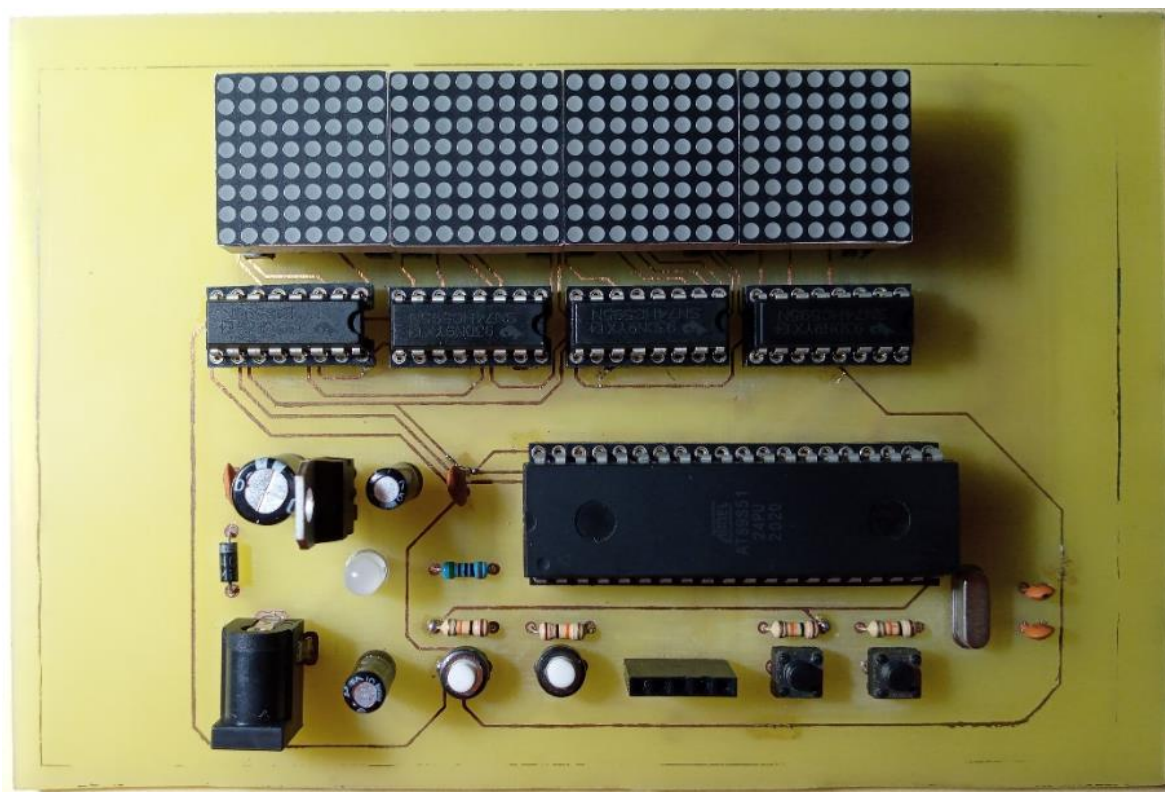
Hình 3.11 PCB layout của mạch trên phần mềm Proteus 8.8



Hình 3.12 Mặt trên của mạch khi xuất ra file pdf



Hình 3.13 Mặt dưới của mạch khi xuất file pdf
Thực hiện hàn linh kiện và hoàn thành mạch.

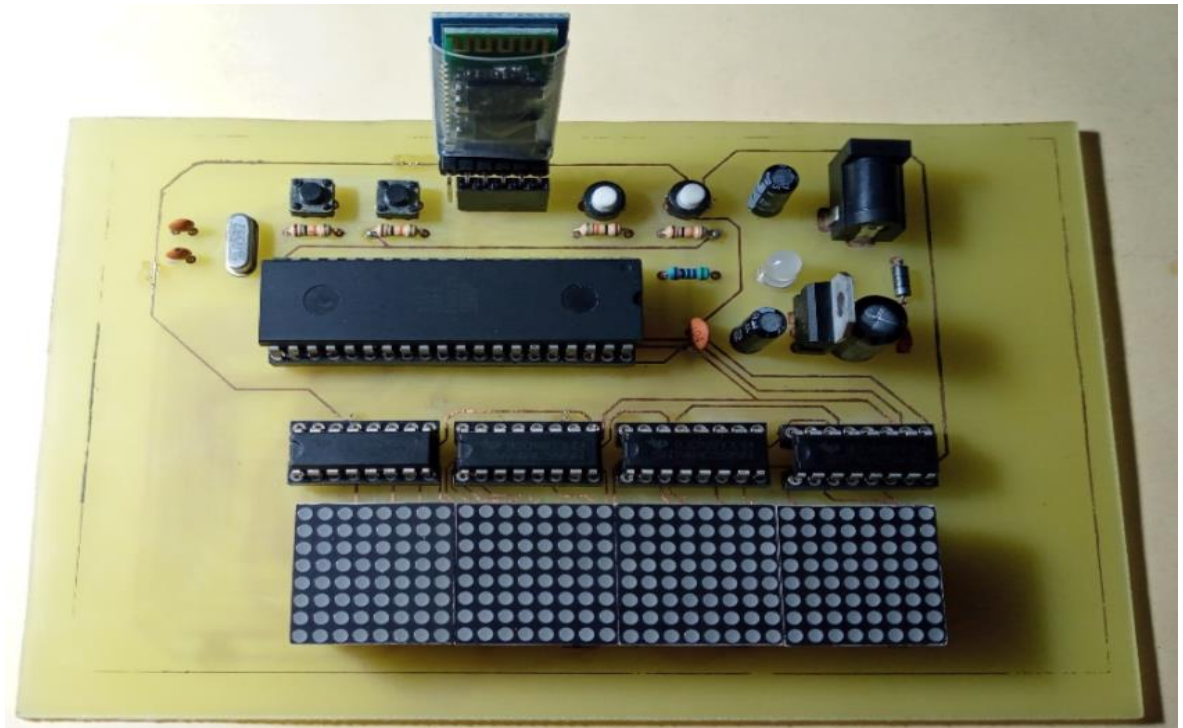


Hình 3.14 Mạch sau khi hoàn thành

Chương 4. KẾT QUẢ THỰC NGHIỆM

4.1. Hiệu ứng 1

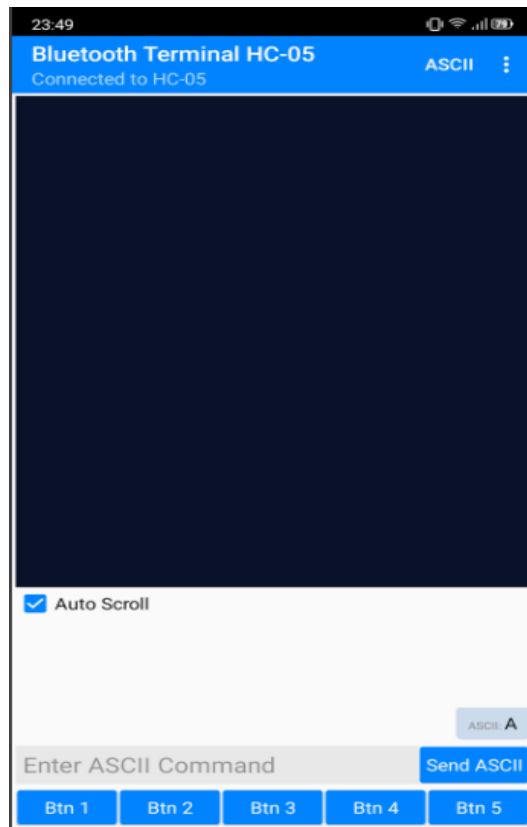
Sử dụng Adapter 12V 1A để cấp nguồn cho mạch. Gắn module bluetooth HC-05 và header J3.



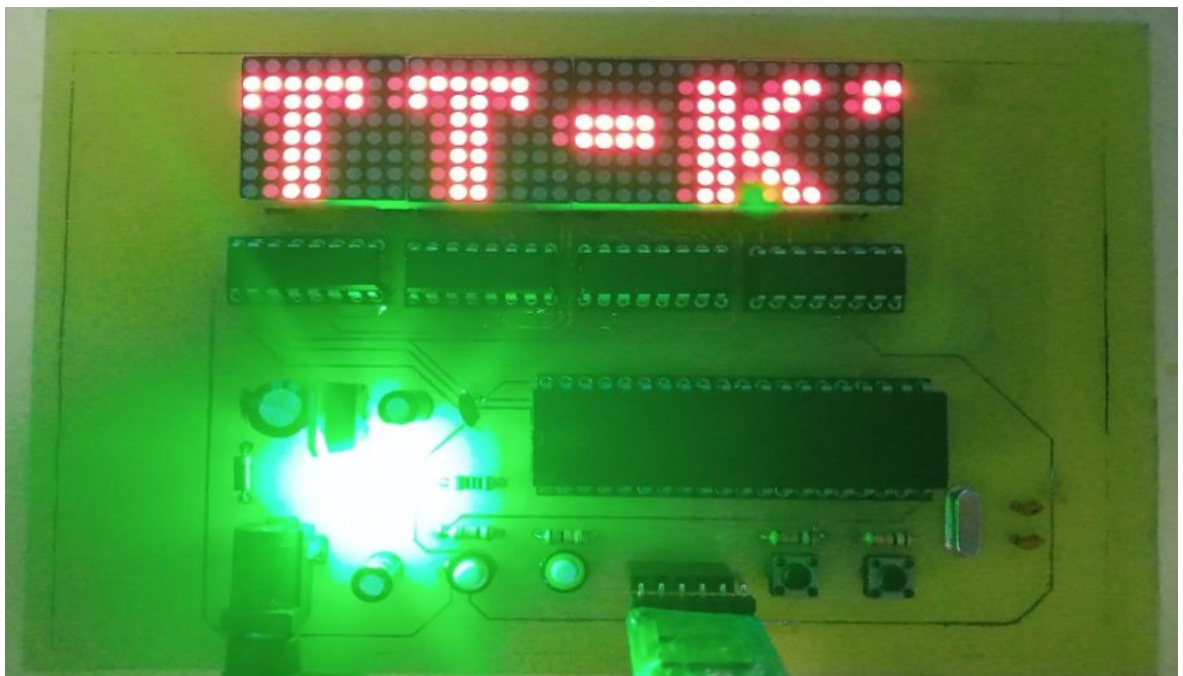
Hình 4.1 Mạch với module HC-05 được kết nối

Các bước thực hiện:

- Cấp nguồn
- Nhấn Reset, kết nối bluetooth với smartphone, điều khiển bằng app *Bluetooth Terminal HC-05*.
- Nhấn Start
- Gửi dữ liệu (Ký tự A)



Hình 4.2 Truyền ký tự A trên app



Hình 4.3 Kết quả chạy hiệu ứng 1

4.2. Hiệu ứng 2

Thực hiện:

- Nhấn reset
- Nhấn Start
- Truyền dữ liệu (ký tự B)



Hình 4.4 Truyền ký tự B trên app

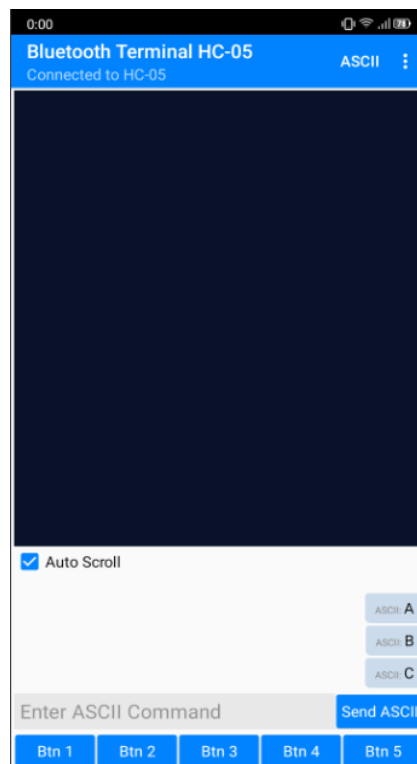


Hình 4.5 Kết quả chạy hiệu ứng 2

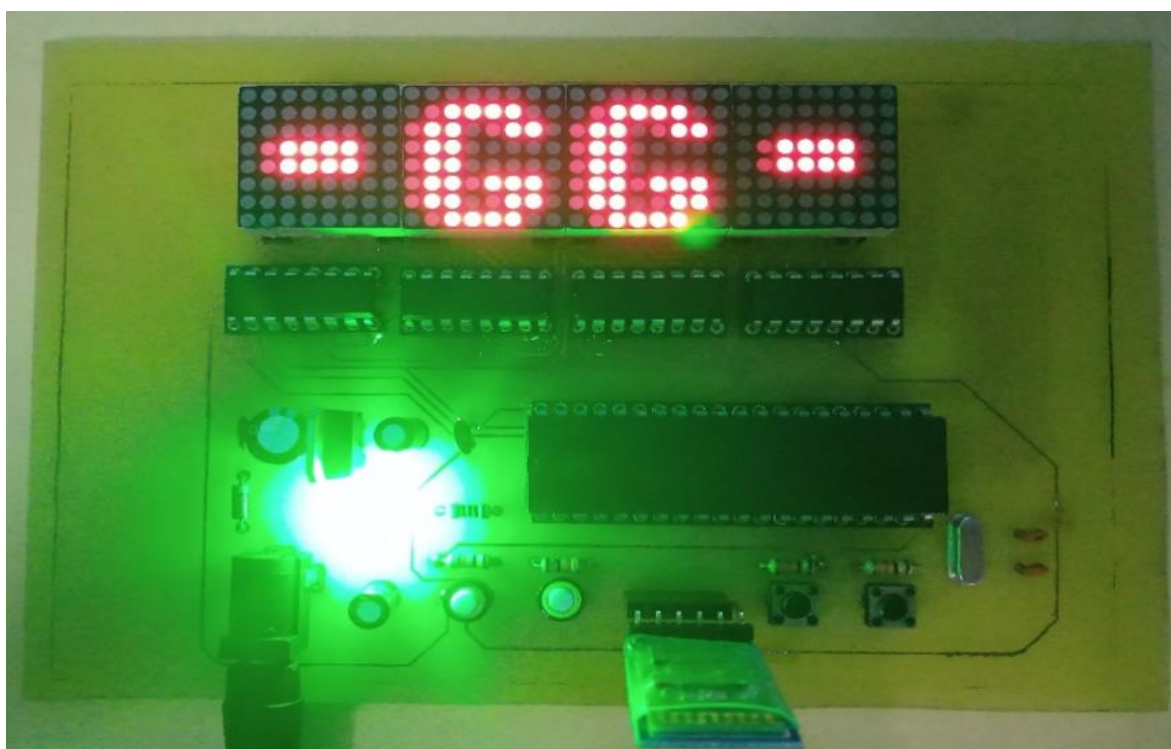
4.3. Hiệu ứng 3

Thực hiện:

- Nhấn reset
- Nhấn Start
- Truyền dữ liệu (ký tự C)



Hình 4.6 Truyền ký tự C trên app



Hình 4.7 Kết quả chạy hiệu ứng 3

Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

5.1.1. Những tiêu chí đã đạt được

- Thực hiện áp dụng kiến thức đã được học về vi điều khiển 8051 để áp dụng vào thực hiện đồ án.
- Áp dụng được kỹ thuật quét LED vào thực hành cụ thể.
- Đáp ứng được yêu cầu đề ra từ đầu: “Thiết kế mạch chạy chữ trên ma trận LED”.

5.1.2. Những hạn chế, khó khăn khi thực hiện

5.1.2.1. Hạn chế

- Mạch in không đẹp, đường dây đồng bị bong, tróc.
- Hiệu ứng không mượt mà.

5.1.2.2. Khó khăn

- Dịch bệnh, không đặt mạch được nên phải thực hiện ủi mạch thủ công.
- Cấu hình HC-05 gặp lỗi
- L7805 toả nhiều nhiệt.

5.2. Hướng phát triển

- Nâng cấp mạch: thực hiện nhập dữ liệu (chuỗi ký tự bất kỳ) từ app và hiển thị lên ma trận LED.
- Mở rộng ma trận LED từ 8x32 lên 8x64 hoặc nhiều hơn (nếu có thể).

TÀI LIỆU THAM KHẢO

- [1] Vũ Đức Lung, Lê Quang Minh, Phan Đình Duy, Giáo trình Vi điều khiển, Nhà xuất bản Đại học Quốc gia Thành phố Hồ Chí Minh, 2016.
- [2] Phạm Quang Huy, Vi điều khiển và ứng dụng Arduino dành cho người tự học, Nhà xuất bản Thanh Niên, 2018.
- [3] Atmel (Atmel Corporation), “Alldatasheet,” [Online]. Available: <https://www.alldatasheet.com/datasheet-pdf/pdf/77367/ATMEL/AT89S51.html>. [Accessed 1 July 2021].
- [4] TI (Texas Instrument), “Alldatasheet,” [Online]. Available: <https://html.alldatasheet.com/html-pdf/27939/TI/SN74HC595N/24/1/SN74HC595N.html>. [Accessed 1 July 2021].
- [5] “Mạch ổn áp sử dụng LM7805,” [Online]. Available: <https://dientutuonglai.com/mach-on-ap-su-dung-lm7805.html>. [Accessed 2 July 2021].
- [6] “Tìm hiểu về LED ma trận,” [Online]. Available: <https://dientutuonglai.com/tim-hieu-led-ma-tran.html>. [Accessed 2 July 2021].
- [7] “Module Bluetooth HC-05 (Phần 1),” 20 June 2016. [Online]. Available: <https://linhkienst.com/blogs/huong-dan-su-dung-cac-module/1000125965-module-bluetooth-hc-05-phan-1>. [Accessed 29 June 2021].
- [8] Green Technology, “Arduino | Vào lệnh Command cấu hình Bluetooth HC-05,” 22 May 2021. [Online]. Available: https://www.youtube.com/watch?v=HhulNYNHAsM&ab_channel=GreenTechnology. [Accessed 17 June 2021].