

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ VIỄN THÔNG

THIẾT KẾ BỘ ĐIỀU KHIỂN TỪ XA DÙNG CẢM BIẾN HỒNG NGOẠI
SỬ DỤNG PIC18F4520

CBHD: TS. Trần Đình Thông

Sinh viên: Nguyễn Văn Bảy

Mã số sinh viên: 2017602582

NGUYỄN VĂN BẢY

CNKT ĐIỆN TỬ VIỄN THÔNG

Hà Nội – Năm 2021

MỤC LỤC

MỤC LỤC	i
LỜI CẢM ƠN	iv
DANH MỤC CÁC TỪ VIẾT TẮT	v
DANH MỤC HÌNH ẢNH	vi
DANH MỤC BẢNG BIỂU	viii
LỜI NÓI ĐẦU	ix
CHƯƠNG 1. TỔNG QUAN VỀ VI ĐIỀU KHIỂN PIC18F4520 VÀ HỆ THỐNG ĐIỀU KHIỂN TỪ XA	11
1.1. Giới thiệu chung về vi điều khiển [1]	11
1.1.1. Giới thiệu về vi điều khiển.....	11
1.1.2. Các loại vi điều khiển hiện nay trên thị trường Việt Nam.....	11
1.2. Vi điều khiển PIC18F4520 [11].....	16
1.2.1. Một vài đặc điểm của PIC18F4520	16
1.2.2. Sơ đồ chân của PIC18F4520.....	18
1.2.3. Sơ đồ khối của PIC18F4520	19
1.3. Tổng quan về hệ thống điều khiển từ xa.....	19
1.3.1. Sơ lược về hệ thống điều khiển từ xa [2].....	19
1.3.2. Các phương pháp điều khiển từ xa [3].....	20
1.3.3. Nhiệm vụ cơ bản của hệ thống điều khiển từ xa.....	22
1.3.4. Kết cấu tin tức của hệ thống điều khiển từ xa.....	23
1.3.5. Sơ đồ khối hệ thống điều khiển từ xa	23
1.4. Điều khiển từ xa dùng cảm biến hồng ngoại	24
1.4.1. Tìm hiểu về hồng ngoại [4].....	24
1.4.2. Led phát hồng ngoại [5]	26
1.4.3. Phương thức giao tiếp hồng ngoại [6].....	27
1.4.4. Truyền dữ liệu thông qua ánh sáng hồng ngoại [7]	27
1.5. Kết luận chương	28
CHƯƠNG 2. THIẾT KẾ MẠCH ĐIỀU KHIỂN TỪ XA DÙNG CẢM BIẾN HỒNG NGOẠI SỬ DỤNG PIC18F4520.....	29
2.1. Xác định bài toán cho hệ thống.....	29
2.1.1. Xác định bài toán	29

2.1.2. Yêu cầu của bài toán thiết kế hệ thống điều khiển từ xa	29
2.1.3. Giải pháp công nghệ	29
2.1.4. Giải pháp thiết kế	30
2.2. Sơ đồ khối hệ thống điều khiển từ xa bằng cảm biến hồng ngoại	30
2.2.1. Sơ đồ hệ thống phân phát.....	30
2.2.2. Sơ đồ hệ thống phân thu.....	31
2.2.3. Sơ đồ khối hệ thống tổng quát	31
2.3. Một số linh kiện sử dụng trong mạch	33
2.3.1. Vi điều khiển PIC18F4520 [12].....	33
2.3.2. Led thu hồng ngoại [9].....	38
2.3.3. Triac	39
2.3.4. Transistor.....	40
2.3.5. Diot.....	41
2.3.6. IC ổn áp 7805	41
2.3.7. Thạch anh 20MHz.....	42
2.3.8. MOC 3023.....	42
2.4. Kết luận chương	43
CHƯƠNG 3. THIẾT KẾ MÔ HÌNH ĐIỀU KHIỂN TỪ XA	44
3.1. Các mạch điện tử sử dụng trong thiết kế mạch điều khiển từ xa.....	44
3.1.1. Khối nguồn.....	44
3.1.2. Khối điều khiển	44
3.1.3. Khối LED thu hồng ngoại.....	45
3.1.4. Khối chấp hành	45
3.1.5. Khối Reset.....	46
3.1.6. Khối nút nhấn.....	46
3.1.7. Khối led status.....	47
3.1.8. Khối thạch anh (20 MHz)	47
3.2. Sơ đồ nguyên lí	48
3.3. Mạch PCB trên phần mềm Altium.....	49
3.4. Mạch hoàn thiện.....	50
3.5. Lưu đồ thuật toán	51
3.6. Các bước lập trình cho vi điều khiển	52

3.7. Mô hình sản phẩm hoàn thiện	53
3.8. Kết quả thực nghiệm	54
3.9. Kết luận và hướng phát triển.....	55
TÀI LIỆU THAM KHẢO.....	57
PHỤ LỤC	58

LỜI CẢM ƠN

Để hoàn thành đề tài nghiên cứu này, lời đầu tiên em xin được gửi lời cảm ơn chân thành đến toàn thể quý thầy cô **Trường Đại học Công Nghiệp Hà Nội** nói chung và các thầy cô trong **Khoa Điện tử - Viễn thông** nói riêng luôn tạo điều kiện tốt nhất và tận tình hướng dẫn, trang bị cho em những kiến thức nền tảng cũng như kiến thức chuyên ngành quan trọng, giúp em có được cơ sở lý thuyết vững vàng.

Đặc biệt, em xin chân thành cảm ơn thầy **Tiến Sĩ Trần Đình Thông** đã tận tình giúp đỡ, tạo điều kiện thuận lợi cũng như góp ý để em có thể giải quyết những vấn đề gặp phải trong suốt quá trình thực hiện đề tài. Em cũng xin cảm ơn tất cả các bạn bè, những người đã luôn ủng hộ, giúp đỡ em trong quá trình nghiên cứu đề tài. Do kiến thức và kinh nghiệm của bản thân em còn nhiều hạn chế nên việc thực hiện đề tài này không thể tránh khỏi những thiếu sót. Vì vậy, em rất mong được nhận những lời góp ý của thầy cô giáo và các bạn hơn nữa.

Em xin chân thành cảm ơn !

DANH MỤC CÁC TỪ VIẾT TẮT

Ký hiệu	Từ viết tắt	Ý nghĩa
PIC	Programmable Interface Controller	Máy tính thông minh có thể lập trình được
ALU	Arithmetic Logic Unit	Bộ logic và số học
ROM	Bộ nhớ đọc Read-Only Memory	Bộ nhớ chỉ đọc
RAM	Random Access Memory	Bộ nhớ truy cập ngẫu nhiên
EEPROM	Electrically Erasable Programmable	Chip nhớ chỉ đọc, có khả năng lập trình lại
PWM	Pulse Width Modulation	Điều chế độ rộng xung
SRAM	Static random-access memory	Bộ nhớ truy cập ngẫu nhiên tĩnh
I/O	Input/Output	Đầu vào/ Đầu ra
USART	Universal Synchronous Asynchronous Receiver / Transmitter	Bộ thu phát đồng bộ/ không đồng bộ
PDA	Personal Digital Assistant	Thiết bị kỹ thuật số hỗ trợ cá nhân
IR	Infrared	Tia hồng ngoại
RF	Radio frequency	Sóng radia
AC	Alternating Current	Dòng điện xoay chiều
DC	Direct Current	Dòng điện một chiều
BTN	Button	Nút nhấn
ADC	Analog-to-digital converter	Bộ chuyển đổi tương tự ra số

DANH MỤC HÌNH ẢNH

Hình 1. 1. Vi điều khiển 8051 [5]	11
Hình 1. 2. Vi điều khiển ARM STM32 [5].....	12
Hình 1. 3. Vi điều khiển AVR [5].....	13
Hình 1. 4. Vi điều khiển PIC18F4520 [5].....	16
Hình 1. 5. Sơ đồ chân vi điều khiển PIC 18F4520 [8].....	18
Hình 1. 6. Sơ đồ khối kiến trúc vi điều khiển PIC18F4520 [8].....	19
Hình 1. 7. Điều khiển từ xa bằng giọng nói [10]	21
Hình 1. 8. Điều khiển từ xa sử dụng sóng vô tuyến RF [3]	21
Hình 1. 9. Điều khiển từ xa dùng hồng ngoại (IR) [5].....	22
Hình 1. 10. Sơ đồ khối hệ thống điều khiển từ xa	23
Hình 1. 11. Ánh sáng hồng ngoại.....	24
Hình 1. 12. Hình ảnh led phát hồng ngoại	26
Hình 1. 13. Chuẩn giao tiếp NEC [6].....	27
Hình 2. 1. Sơ đồ phân phát hệ thống điều khiển từ xa dùng hồng ngoại.....	30
Hình 2. 2. Remote điều khiển từ xa [5].....	30
Hình 2. 3. Sơ đồ hệ thống phân thu điều khiển từ xa dùng hồng ngoại.....	31
Hình 2. 4. Sơ đồ khối hệ thống điều khiển từ xa	31
Hình 2. 6. Hình ảnh vi điều khiển PIC18F4520 [5].....	33
Hình 2. 7. Led thu hồng ngoại.....	38
Hình 2. 8. Hình ảnh Triac [5].....	39
Hình 2. 9. Hình ảnh và kí hiệu của transistor [5]	40
Hình 2. 10. Diode [5]	41
Hình 2. 11. IC ổn áp 7805 [5]	41
Hình 2. 12. Thạch anh 20MHz [5]	42
Hình 3. 1. Khối nguồn.....	44
Hình 3. 2. Khối điều khiển Pic18f4520	44
Hình 3. 3. Khối led thu hồng ngoại.....	45
Hình 3. 4. Khối chấp hành	45

Hình 3. 5. Khởi reset	46
Hình 3. 6. Khởi nút nhấn.....	46
Hình 3. 7. Khởi hiển thị led trạng thái	47
Hình 3. 8. Khởi tạo dao động thạch anh 20MHz	47
Hình 3. 9. Sơ đồ nguyên lí mạch điều khiển từ xa cảm biến hồng ngoại	48
Hình 3. 10. Mạch in 2D.....	49
Hình 3. 11. Mạch in 3D.....	49
Hình 3. 12. Mặt trước mạch thực tế	50
Hình 3. 13. Mặt sau mạch thực tế	50
Hình 3. 14. Lưu đồ thuật toán	51
Hình 3. 15. Cài đặt cấu hình, thông số cho vi điều khiển	52
Hình 3. 16. Hiển thị màn hình viết code	52
Hình 3. 17. Chạy và kiểm tra lỗi chương trình	53
Hình 3. 18. Hình ảnh mô hình hệ thống.....	53
Hình 3. 19. Hình ảnh bật thiết bị 1, 2, 3.....	54
Hình 3. 20. Hình ảnh đang học lệnh bật/tắt thiết bị 1	54

DANH MỤC BẢNG BIỂU

Bảng 2. 1. Các thanh ghi liên quan đến PORTA [11].....	33
Bảng 2. 2. Các thanh ghi liên quan đến PORTB [11].....	34
Bảng 2. 3. Các thanh ghi liên quan đến PORTC [11].....	34
Bảng 2. 4. Các thanh ghi liên quan đến PORTD [11].....	34
Bảng 2. 5. Các thanh ghi liên quan đến PORTE [11].....	34

LỜI NÓI ĐẦU

Ngày nay, với tốc độ phát triển nhanh chóng của các ngành công nghệ như công nghệ thông tin, công nghệ truyền thông, cơ khí, động lực. Ngành điện tử đóng vai trò rất quan trọng và ảnh hưởng tới cuộc sống con người khá sớm từ những thiết bị đơn giản đến những máy móc phức tạp. Do đó, với sự trợ giúp của máy móc, những công cụ thông minh con người không phải trực tiếp làm việc, hay những công việc mà con người không thể làm được với khả năng của mình mà chỉ việc điều khiển hoàn toàn tự động đã mang lại những lợi ích hết sức to lớn, giảm nhẹ và tối ưu hóa công việc.

Trong sinh hoạt hằng ngày của con người như những thiết bị (robot, xe điều khiển từ xa,...) cho đến những ứng dụng gần gũi với chúng ta cũng được cải tiến cho phù hợp với việc sử dụng đạt mức tiện lợi nhất. Điều khiển từ xa có chức năng làm việc điều khiển mô hình, thiết bị ở một khoảng cách nào đó mà người dùng không nhất thiết phải đến gần nơi đặt hệ thống. Trong đó, điều khiển từ xa sử dụng bộ thu phát bằng hồng ngoại cho độ chính xác và nhanh chóng đang ngày một được ứng dụng rộng rãi trong thực tế.

Xuất phát từ những ý tưởng trên, em đã lựa chọn nghiên cứu, tìm hiểu về mạch điều khiển thiết bị từ xa sử dụng cảm biến hồng ngoại. Để thực hiện được đề tài theo yêu cầu, trong đồ án tập trung nghiên cứu về các vấn đề chính sau đây:

- Tổng quan về thiết bị điều khiển từ xa.
- Phân tích và thiết kế hệ thống.
- Thiết kế hoàn thiện hệ thống.

Với mong muốn thiết kế những chức năng cần thiết cho cuộc sống của con người, thay vì phải đứng dậy bật quạt, tắt điện, bật điện, thì giờ đây chỉ với một chiếc điều khiển từ xa trong tay ta có thể ở nguyên vị trí trong nhà mà có thể điều khiển được tất cả thiết bị điện tắt mở theo ý muốn. Vì vậy, trong đề tài này em đi sâu vào nghiên cứu hệ thống điều khiển từ xa bằng hồng ngoại dùng để điều khiển các thiết bị điện tử trong gia đình của nước ta hiện nay. Ngoài ra,

việc thiết kế đề tài này còn giúp em có cơ hội học hỏi và tìm hiểu về các linh kiện điện tử, vi điều khiển PIC18F4520 với phần mềm lập trình đã học cũng như nâng cao khả năng, năng lực của bản thân để sau khi ra trường trở thành một kỹ sư giỏi đóng góp nhiều cho nền công nghiệp nước nhà và xã hội.

CHƯƠNG 1. TỔNG QUAN VỀ VI ĐIỀU KHIỂN PIC18F4520 VÀ HỆ THỐNG ĐIỀU KHIỂN TỪ XA

1.1. Giới thiệu chung về vi điều khiển [1]

1.1.1. Giới thiệu về vi điều khiển

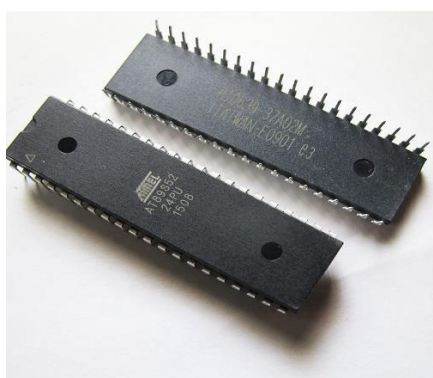
Vi điều khiển là một máy tính được tích hợp trên một chip, nó thường được sử dụng để điều khiển các thiết bị điện tử. Vi điều khiển thực chất gồm một vi xử lý có hiệu suất đủ cao và giá thành thấp (so với các vi xử lý đa năng dùng trong máy tính) kết hợp với các thiết bị ngoại vi như các bộ nhớ, các mô đun vào/ra, các mô đun biến đổi từ số sang tương tự và từ tương tự sang số, mô đun điều chế độ rộng xung (PWM),...

Vi điều khiển thường được dùng để xây dựng hệ thống nhúng. Nó xuất hiện nhiều trong các dụng cụ điện tử, thiết bị điện, máy giặt, lò vi sóng, điện thoại, dây truyền tự động,... Hầu hết các loại vi điều khiển hiện nay có cấu trúc Harvard là loại cấu trúc mà bộ nhớ chương trình và bộ nhớ dữ liệu được phân biệt riêng.

Cấu trúc của một vi điều khiển gồm CPU, bộ nhớ chương trình (thường là bộ nhớ ROM hoặc bộ nhớ Flash), bộ nhớ dữ liệu (RAM), các bộ định thời, các cổng vào/ra để giao tiếp với các thiết bị bên ngoài, tất cả các khối này được tích hợp trên một vi mạch.

1.1.2. Các loại vi điều khiển hiện nay trên thị trường Việt Nam

a. Vi điều khiển 8051.



Hình 1. 1. Vi điều khiển 8051 [5]

Intel 8051 - là vi điều khiển đơn tinh thể kiến trúc Harvard, lần đầu tiên được sản xuất bởi Intel năm 1980, để dùng trong các hệ thống nhúng. Trong những năm 1980 và đầu những năm 1990 đã rất nổi tiếng. Tuy nhiên hiện tại đã cũ và được thay thế bằng các thiết bị hiện đại hơn, với các lõi phối hợp 8051, được sản xuất bởi hơn 20 nhà sản xuất độc lập như Atmel, Maxim IC (công ty con của Dallas Semiconductor), NXP Semiconductors (Philips Semiconductor trước đây), Winbond, Silicon Laboratories, Texas Instruments và Cypress Semiconductor.

Tên gọi chính thức của họ vi điều khiển Intel 8051 - MCS 51. Những vi điều khiển Intel 8051 được sản xuất với việc dùng công nghệ MOSFET, những bản sau chứa kí hiệu “C” trong tên như 80C51, dùng công nghệ CMOS và yêu cầu công suất thấp, hơn những cái MOSFET trước (điều này cho phép trang bị cho các thiết bị với nguồn là pin).

Các thông số kỹ thuật: 8 bit ALU, 8 bit thanh ghi, 8 bit dữ liệu bus, 16 bit địa chỉ bus vì vậy, không gian bộ nhớ tối đa cho ROM và RAM lên tới 64 kb, bộ nhớ dữ liệu SRAM 128 bytes, bộ nhớ chương trình ROM 4 kb, 32 chân vào/ra đa hướng, giao tiếp nối tiếp UART, hai bộ timer/counter 16 bit, hai ngắt ngoài.

b. Vi điều khiển ARM.

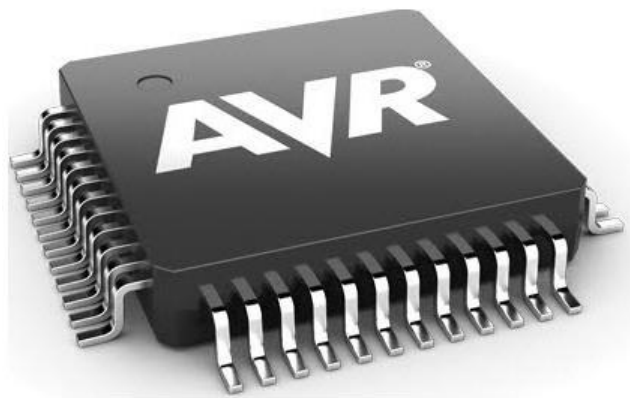


Hình 1. 2. Vi điều khiển ARM STM32 [5]

Cấu trúc ARM (viết tắt từ tên gốc là Acorn RISC Machine) là một loại cấu trúc vi xử lý 32-bit kiểu RISC được sử dụng rộng rãi trong các thiết kế nhúng. Được phát triển lần đầu trong một dự án của công ty máy tính Acorn. Do có đặc điểm tiết kiệm năng lượng, các bộ CPU ARM chiếm ưu thế trong các sản phẩm điện tử di động, mà với các sản phẩm này việc tiêu tán công suất thấp là một mục tiêu thiết kế quan trọng hàng đầu.

Ngày nay, hơn 75% CPU nhúng 32-bit là thuộc họ ARM, điều này khiến ARM trở thành cấu trúc 32-bit được sản xuất nhiều nhất trên thế giới. CPU ARM được tìm thấy khắp nơi trong các sản phẩm thương mại điện tử, từ thiết bị cầm tay (PDA, điện thoại di động, máy đa phương tiện, máy trò chơi cầm tay và máy tính cầm tay) cho đến các thiết bị ngoại vi máy tính (ổ đĩa cứng, bộ định tuyến để bàn.).

c. Vi điều khiển AVR.



Hình 1. 3. Vi điều khiển AVR [5]

Là dòng vi điều khiển do hãng Atmel sản xuất.

Là vi điều khiển 8 bit với tiêu thụ điện năng thấp dựa trên kiến trúc RISC (Reduced Instruction Set Computer). Vào ra Analog – digital và ngược lại. Với công nghệ này cho phép các lệnh thực thi chỉ trong một chu kỳ xung nhịp, vì thế tốc độ xử lý dữ liệu có thể đạt đến 1 triệu lệnh trên giây ở tần số 1Mhz. Vi điều khiển này cho phép người thiết kế có thể tối ưu hoá chế độ tiêu thụ năng lượng mà vẫn đảm bảo tốc độ xử lý. Lỗi AVR có tập lệnh phong phú với số lượng với 32 thanh ghi làm việc chung với nhau.

Phần mềm này hỗ trợ nhiều ứng dụng và có nhiều hàm có sẵn nên việc lập trình tốt hơn:

- Bộ nhớ: Flash 16KB EEPROM 512 Byte SRAM 1KB.
- Ngoại vi: Hai timer 8 bit Một timer 16 bit Bộ counter với tần số riêng.
- 4 bộ điều chế độ rộng xung PWM.
- Tám kênh ADC 10 bit USART, giao tiếp SPI, giao diện I2C, Watchdog timer, bộ so sánh tương tự trên chip.

d. Vi điều khiển PIC.

PIC là một họ vi điều khiển RISC được sản xuất bởi công ty Microchip Technology. Dòng PIC đầu tiên là PIC1650 được phát triển bởi Microelectronics Division thuộc General Instrument. PIC bắt nguồn là chữ viết tắt của "Programmable Intelligent Computer" (Máy tính khả trình thông minh). Là vi điều khiển với kiến trúc RISC thực thi một lệnh với một chu kỳ máy (bằng bốn chu kỳ của bộ dao động).

Ngày nay có nhiều dòng PIC được sản xuất với hàng loạt các mô đun ngoại vi tích hợp sẵn như ADC, PWM, USART, SPI... với bộ nhớ chương trình từ 512 word đến 32 Kword. Các họ vi điều khiển PIC:

- Họ 8 bit: PIC 10/ PIC 12/ PIC 16/ PIC 18.
- Họ 16 bit: PIC 24F/ PIC 24H/ dsPIC 30/ dsPIC 33.
- Họ 32 bit: PIC 32.

➤ **Một vài đặc tính của PIC:**

- Chân vào/ra I/O có thể lập trình được.
- Flash và ROM có thể tùy chọn từ 256 byte đến 512 Kbyte.
- Bộ dao động bên trong.
- 8/16/32 bit Timers.
- Bộ nhớ EEPROM nội.
- Chuẩn giao tiếp nối tiếp đồng bộ và không đồng bộ USART.
- MSSP Peripheral cho giao tiếp I2C và SPI.

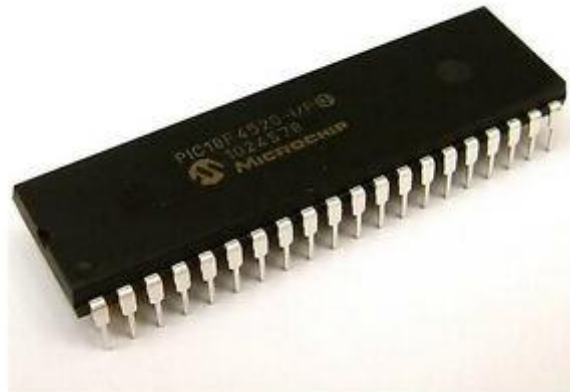
- Các chế độ so sánh, bắt giữ và điều chế độ rộng xung PWM.
- Bộ so sánh điện áp.
- Bộ chuyển đổi ADC (tần số có thể lên tới 1 MHz).
- Hỗ trợ các giao thức USB, CAN, Ethernet.
- Mô đun điều khiển động cơ, mô đun đọc encoder.
- Hỗ trợ bộ nhớ ngoài.
- DSP những tính năng xử lý tín hiệu số (dsPIC).

➤ **Một số lí do vi điều khiển PIC được sử dụng nhiều hiện nay trên thị trường:**

- PIC phát triển và được sử dụng phổ biến ở nước ta nên việc tìm mua và trao đổi kinh nghiệm hết sức thuận lợi.
- Có sự hỗ trợ cao của nhà sản xuất về các công cụ lập trình, trình biên dịch, mạch nạp Pic từ đơn giản tới phức tạp.
- Các tính năng đa dạng của các dòng Pic không ngừng được phát triển. Có nhiều bộ phận ngoại vi ngay trên chip, bao gồm: Cổng và/ra số, bộ biến đổi ADC, bộ nhớ EEPROM, bộ định thời, bộ điều chế độ rộng xung (PWM)...
- Bộ nhớ chương trình và bộ nhớ dữ liệu được tích hợp ngay trên chip.
- Giá thành các dòng PIC phù hợp với người sử dụng.
- PIC có đầy đủ tính năng để hoạt động độc lập.

Các dòng vi điều khiển rất đa dạng từ đơn giản đến phức tạp tùy vào yêu cầu kỹ thuật và giá thành của sản phẩm. Vận dụng những kiến thức đã học và những đặc điểm đã nêu ở trên em lựa chọn sử dụng dòng vi điều khiển PIC18F4520 để thực hiện đồ án này.

1.2. Vi điều khiển PIC18F4520 [11]



Hình 1. 4. Vi điều khiển PIC18F4520 [5]

Pic18F4520 nằm trong dòng sản phẩm PIC18F2420/2520/4420/4520 của nhà sản xuất Microchip với đặc điểm 28/40/44 -Pin Enhanced Flash Microcontrollers with 10- Bit A/D and nanoWatt Technology.

Dòng sản phẩm này có nhiều cải tiến đáng kể về tính năng so các dòng Pic trước đó như:

- Bộ nhớ chương trình được tăng cường (16Kbytes for PIC18F2420/4420 devices and 32Kbytes for PIC18F2520/4520 devices).
- I/O ports (3 bidirectional ports on 28-pin devices, 5 bidirectional ports on 40/44-pin devices).
- Tăng cường modul CCP.
- Sử dụng công nghệ nanoWatt.

Dòng sản phẩm này nói chung có nhiều sự tương đồng về tính năng nhưng có thể chia làm hai nhóm Pic18F2420/2520 với 28 chân và Pic 18F4420/4520 với 40 hoặc 44 chân ghép nối. Trong đồ án chúng ta quan tâm chủ yếu Pic18F4520 với 40 chân sử dụng trong mạch thiết kế.

1.2.1. Một vài đặc điểm của PIC18F4520

Sử dụng công nghệ nano watt: hiệu năng cao, tiêu thụ năng lượng ít.

Kiến trúc RISC:

- 75 lệnh mạnh, hầu hết các lệnh thực hiện trong bốn chu kì xung.

- Tốc độ thực hiện lên tới 10 triệu lệnh trong 1s với tần số 40Mhz.
- Có bộ nhân cứng.

Các bộ nhớ chương trình và dữ liệu cố định:

• 32 Kbytes bộ nhớ flash có khả năng tự lập trình trong hệ thống có thể thực hiện được 100.000 lần ghi/xóa.

- 256 bytes EEPROM có thể thực hiện được 1.000.000 lần ghi/xóa.
- 256 bytes SRAM.

Những bộ ngoại vi tiêu biểu của PIC18F4520:

• 4 bộ định thời/bộ đếm 8 bit với các chế độ tỉ lệ đặt trước và chế độ so sánh.

- Bộ đếm thời gian thực với bộ tạo dao động riêng biệt.
- 2 kênh PWM.
- 13 kênh ADC 10 bit.
- Bộ truyền tin nối tiếp USART khả trình.
- Watchdog Timer khả trình với bộ tạo dao động bên trong riêng biệt.
- Bộ so sánh tương tự.

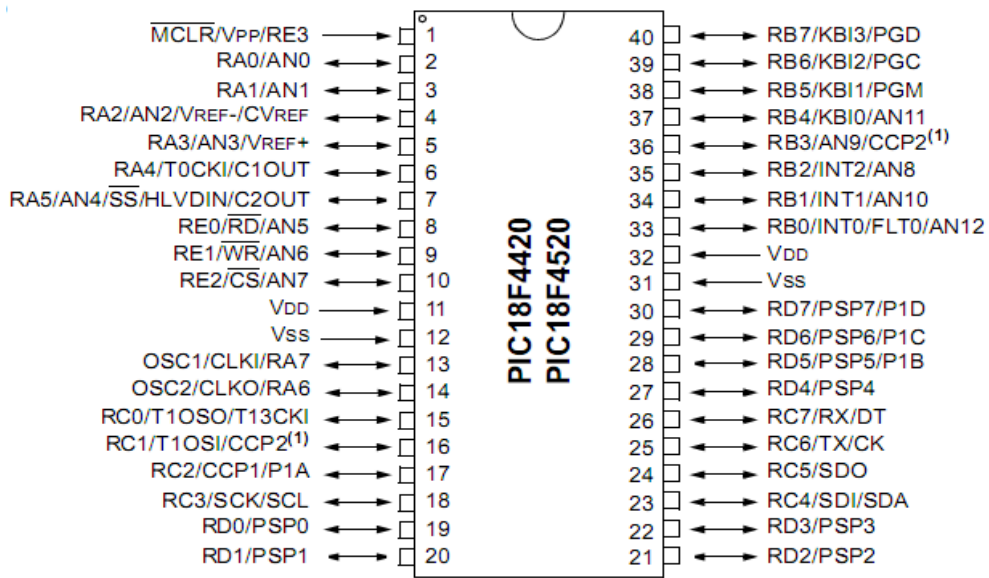
Các đặc điểm đặc biệt khác:

- Power on Reset và dò Brown out khả trình.
- Bộ tạo dao động RC được định cỡ bên trong.
- Các nguồn ngắt bên trong và bên ngoài.

I/O và các kiểu đóng gói:

- 32 đường I/O khả trình.
- Đóng gói 40-pin PDIP, 44-lead TQFP và 44-pad MLF.

1.2.2. Sơ đồ chân của PIC18F4520



Hình 1. 5. Sơ đồ chân vi điều khiển PIC 18F4520 [8]

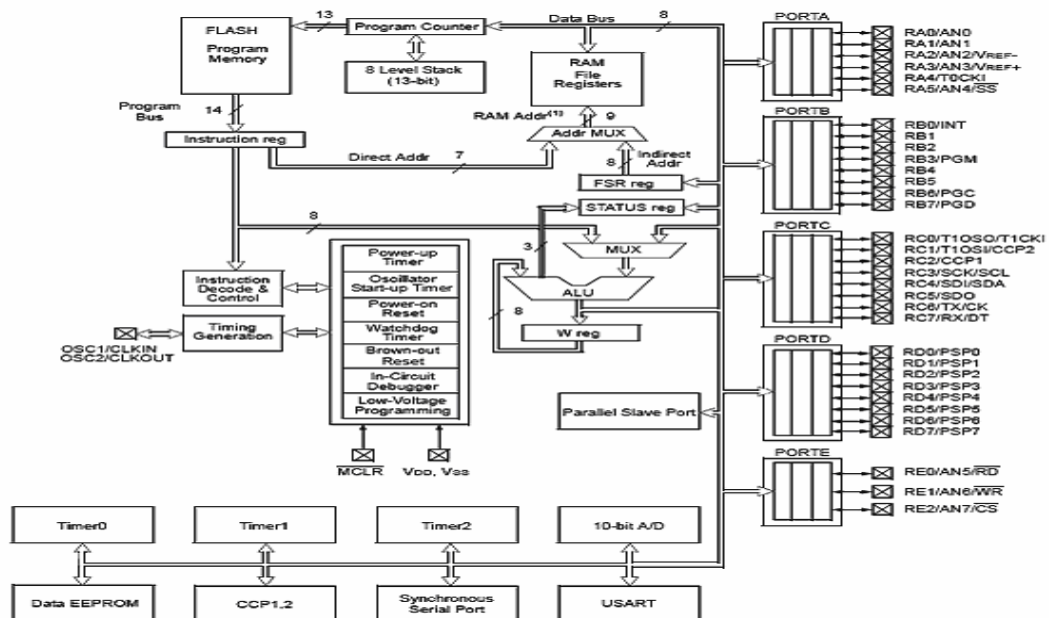
Chức năng các chân của PIC18F4520 loại 40 chân (40 Pin PDIP):

- **Chân 1 (MCLR /VPP/RE3):** Là các cổng đầu vào và hoạt động ở mức thấp để có thể khởi động lại thiết bị và điều chỉnh điện áp đầu vào.
- **Các chân thuộc cổng vào ra Port A:** Gồm các chân từ 1-7 và 13, 14: Với chức năng chủ yếu của các chân là đầu vào, ra số, đầu vào tương tự và đầu vào ra của bộ dao động thạch anh cũng như cung cấp nguồn xung từ bên ngoài vào cho các chân.
- **Các chân cổng vào ra hai chiều Port B:** Cũng như các chân thuộc port A thì các chân từ 33 – 44 thuộc port B cũng có nhiệm vụ hoạt động như là các đầu vào ra số, đầu vào tương tự, cộng thêm đó là vai trò như những chiếc chân ngắt ngoài và có thể thay đổi mở ngắt tùy vào mỗi chân và yêu cầu lập trình.
- **Các chân cổng Port C:** Đó là các đầu vào ra số, cùng với đó mỗi chân lại như là các chân đầu vào ra của timer, đầu vào ra của chuỗi dữ liệu.
- **Các chân cổng Port D:** Gồm các chân từ 19 – 22 và 27 - 30 có thể vào ra hai hướng hoặc cổng song song phụ thuộc(PSP). Với

các chân RD thì vai trò là các chân đầu vào ra số, PSP là các cổng dữ liệu song song phụ thuộc, cộng thêm vào đó mỗi chân từ 28 – 30 có thêm nhiệm vụ là các đầu ra tăng cường cho cổng CCP1.

- **Các chân cổng Port E:** Hoạt động gồm các chân từ 8 – 10 với RE là các đầu vào ra số, RD đầu vào điều khiển để có thể đọc được các thông tin cho PSP, còn các chân AN là các chân đầu vào tương tự. Đầu RE3 nằm ở chân 1.
- **Các chân khác:** Chân 12, 31 (VSS) nối đất chuẩn cho I/O và logic. Chân 11, 32 (VDD) cung cấp nguồn dương cho I/O và logic.

1.2.3. Sơ đồ khối của PIC18F4520



Hình 1. 6. Sơ đồ khối kiến trúc vi điều khiển PIC18F4520 [8]

1.3. Tổng quan về hệ thống điều khiển từ xa

1.3.1. Sơ lược về hệ thống điều khiển từ xa [2]

Hệ thống điều khiển từ xa là một hệ thống cho phép ta điều khiển các thiết bị từ một khoảng cách xa mà không cần phải điều khiển trực tiếp trên thiết bị hoặc hệ thống đó. Những chiếc điều khiển từ xa đầu tiên trên thế giới được ra đời nhằm mục đích phục vụ cho chiến tranh. Các loại điều khiển từ xa bằng tần số vô tuyến được dùng vào chiến tranh thế giới thứ nhất.

Vào cuối những năm 1930, một số nhà sản xuất radio đã cung cấp điều khiển từ xa cho một số kiểu máy cao cấp hơn của họ. Hầu hết trong số này được kết nối với các thiết bị được điều khiển bởi dây, nhưng Philco Bí ẩn Control (1939) là một đài phát thanh tần số thấp phát hoạt động bằng pin, do đó nó trở thành điều khiển từ xa không dây đầu tiên cho một thiết bị điện tử tiêu dùng. Thiết bị sử dụng điều chế số xung, đây cũng là điều khiển từ xa không dây kỹ thuật số đầu tiên.

Đến chiến tranh thế giới thứ 2, điều khiển từ xa dùng để kích nổ những quả bom. Sau chiến tranh, những công nghệ tuyệt vời của chúng tiếp tục được cải tiến để phục vụ đặc lực trong cuộc sống con người. Và đến nay gần như ai cũng đã từng sử dụng điều khiển từ xa để điều khiển một thiết bị nào đó.

Như vậy, điều khiển từ xa đã thực sự có từ lâu trên thế giới cũng như Việt Nam. Điều khiển được lập trình dựa trên công nghệ tần số vô tuyến điện (radio frequency), sau đó được cải tiến trên nền tảng ứng dụng công nghệ hồng ngoại (Infrared Remote).

Ngày nay, điều khiển từ xa đã liên tục được phát triển và nâng cấp, sử dụng kết nối internet, kết nối Bluetooth, cảm biến chuyển động và điều khiển bằng giọng nói tiện lợi. Hiện nay, điều khiển từ xa đều được sử dụng công nghệ hồng ngoại, số ít sử dụng công nghệ tần số vô tuyến do công nghệ hồng ngoại đem lại tiện ích và hiện đại hơn rất nhiều. Điều khiển từ xa không chỉ đơn thuần được hiểu theo nghĩa của một hành động nhất định, nó không chỉ được phát triển dựa trên các thiết bị điện tử mà còn được phát triển trên những nền tảng phần mềm thông minh - phần mềm điều khiển từ xa.

1.3.2. Các phương pháp điều khiển từ xa [3]

Có ba phương pháp cơ bản được sử dụng nhiều trong lĩnh vực điện tử của chúng ta hiện nay là dùng điều khiển bằng giọng nói, sóng vô tuyến tần số radio (RF) và điều khiển dùng hồng ngoại (IR).

a. Điều khiển bằng giọng nói



Hình 1. 7. Điều khiển từ xa bằng giọng nói [10]

- Được gọi là điều khiển từ xa siêu âm.
- Kết hợp dễ dàng với các app hoặc phần mềm trên thiết bị thông minh để điều khiển.
- Giúp giảm tải tối đa thao tác của người dùng nên phương pháp này có thể được dùng nhiều hơn.

b. Sử dụng tần số vô tuyến RF

Khi sử dụng phương pháp này có những thuận lợi sau:

- Truyền đạt tín hiệu với khoảng cách xa.
- Không bị ảnh hưởng nhiều bởi vật cản.
- Tầm phát rộng, nhiều hướng khác nhau nên có thể điều khiển cùng một lúc với các thiết bị nhận kênh đồng thời.



Hình 1. 8. Điều khiển từ xa sử dụng sóng vô tuyến RF [3]

Tuy nhiên khi phát hay thu đều cần có anten, mã hóa phức tạp hơn, gây nhiễu vô tuyến sai méo tín hiệu do số lượng lớn sóng vô tuyến xung quanh chúng ta gần như mọi lúc làm không điều khiển được.

c. Sử dụng công nghệ hồng ngoại:



Hình 1. 9. Điều khiển từ xa dùng hồng ngoại (IR) [5]

Phương pháp này có những ưu điểm sau:

- Không dây dẫn.
- Thiết bị thu dẫn nhỏ gọn, không cần anten thu phát, led có kích thước nhỏ nên dễ lắp đặt và có độ tin cậy cao.
- Điện áp cung cấp thấp, công suất nhỏ.
- Điều khiển được nhiều thiết bị.
- Tính khả thi cao, linh kiện dễ tìm thấy, giá thành rẻ, dễ thiết kế và chế tạo.

Trong những cách đã kể trên, em chọn sử dụng điều khiển từ xa bằng cảm biến hồng ngoại vì nó có nhiều ứng dụng trong thực tế, dễ chế tạo và phù hợp với trình độ hiện tại của bản thân.

1.3.3. Nhiệm vụ cơ bản của hệ thống điều khiển từ xa

Hệ thống điều khiển từ xa có những nhiệm vụ cơ bản sau:

- Phát tín hiệu điều khiển.
- Sản sinh ra xung hoặc hình thành các xung cần thiết.
- Tổ hợp xung thành mã.

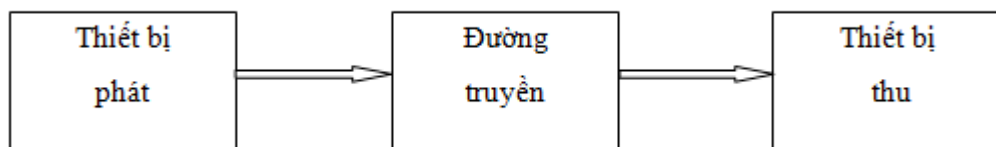
- Phát các tổ hợp mã đến điểm chấp hành.
- Ở điểm chấp hành (thiết bị thu) sau khi nhận được mã phải biến đổi các mã nhận được thành các lệnh điều khiển và đưa đến các thiết bị, đồng thời kiểm tra sự chính xác của mã mới nhận.

1.3.4. Kết cấu tin tức của hệ thống điều khiển từ xa

Trong hệ thống điều khiển từ xa độ tin cậy truyền dẫn tin tức có quan hệ rất nhiều đến kết cấu tin tức. Nội dung về kết cấu tin tức có hai phần: về lượng và về chất. Về lượng có các biến lượng điều khiển và lượng điều khiển thành từng loại xung gì cho phù hợp, và những xung đó cần áp dụng phương pháp nào để hợp thành tin tức, để có dung lượng lớn nhất và có tốc độ truyền dẫn nhanh nhất.

1.3.5. Sơ đồ khối hệ thống điều khiển từ xa

Do hệ thống điều khiển từ xa có những đường truyền dẫn xa nên chúng ta cần phải nghiên cứu về sơ đồ khối hệ thống để đảm bảo tín hiệu được truyền đi chính xác và nhanh chóng. Đây là sơ đồ khối hệ thống:

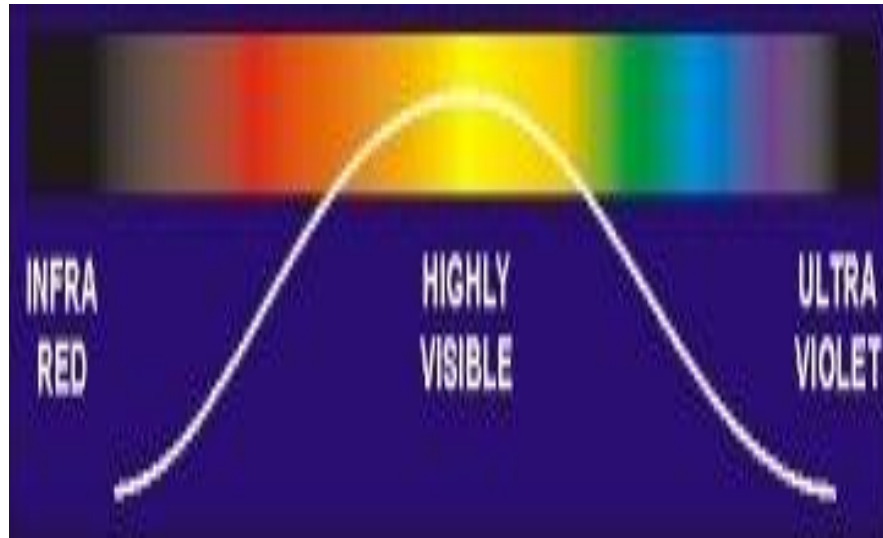


Hình 1. 10. Sơ đồ khối hệ thống điều khiển từ xa

- Thiết bị phát: biến đổi lệnh điều khiển thành tin tức tín hiệu và phát đi.
- Đường truyền: truyền tín hiệu điều khiển từ thiết bị phát đến thiết bị thu.
- Thiết bị thu: nhận tín hiệu điều khiển từ đường truyền, qua quá trình biến đổi, biến dịch để tái hiện lại lệnh điều khiển rồi đưa đến các thiết bị thi hành.

1.4. Điều khiển từ xa dùng cảm biến hồng ngoại

1.4.1. Tìm hiểu về hồng ngoại [4]



Hình 1. 11. Ánh sáng hồng ngoại

Tia hồng ngoại là bức xạ điện từ có bước sóng dài hơn ánh sáng mà mắt thường có thể nhìn thấy.

Thông thường, mắt chúng ta có thể nhìn thấy 7 màu của ánh sáng từ tím đến đỏ, trong đó ánh sáng đỏ có bước sóng lớn nhất là 700nm. Do đó, tia hồng ngoại sẽ có bước sóng vào khoảng 700 nm đến 1 mm và được chia làm 3 loại theo chiều dài bước sóng.

Trong đó bao gồm tia hồng ngoại gần, tia hồng ngoại trung và tia hồng ngoại xa. Tia hồng ngoại xa có bước sóng dài nhất và có năng lượng bức xạ thấp nhất. Khái niệm “hồng ngoại” có nghĩa là “dưới mức đỏ”, màu đỏ là màu sắc có bước sóng dài nhất trong ánh sáng thường. Mọi vật có nhiệt độ lớn hơn 0°K đều phát ra tia hồng ngoại.

Tính chất của hồng ngoại:

- Tác dụng nhiệt.
- Có thể gây ra hiện tượng quang điện trong ở chất bán dẫn.
- Có thể tác dụng lên một số kính ảnh đặc biệt.
- Có thể biến điệu như sóng điện từ cao tần.
- Tia hồng ngoại tuân theo các định luật: truyền thẳng, phản xạ, và cũng gây được hiện tượng nhiễu xạ, giao thoa như ánh sáng thông thường.

Sóng hồng ngoại có những đặc tính quan trọng giống như ánh sáng (sự hội tụ qua thấu kính, tiêu cự...). Ánh sáng thường và ánh sáng hồng ngoại khác nhau rất rõ trong sự xuyên suốt qua vật chất. Có những vật chất ta thấy nó một màu xám đục nhưng với ánh sáng hồng ngoại nó trở nên xuyên suốt. Vì vật liệu bán dẫn “trong suốt” đối với ánh sáng hồng ngoại và tia hồng ngoại không bị yếu đi khi vượt qua các lớp bán dẫn để đi ra ngoài.

Tia hồng ngoại dễ bị hấp thụ, khả năng xuyên thấu kém. Trong điều khiển từ xa bằng tia hồng ngoại, chùm tia hồng ngoại phát đi hẹp, có hướng, do đó khi thu phải đúng hướng.

Tia hồng ngoại có thể truyền đi được nhiều kênh tín hiệu. Nó ứng dụng rộng rãi trong công nghiệp, lượng thông tin có thể đạt được 3Mbit/s. Lượng thông tin được truyền đi với ánh sáng hồng ngoại lớn gấp nhiều lần so với sóng điện từ mà ta vẫn dùng.

Hồng ngoại thật thú vị bởi vì nó tạo ra một cách dễ dàng và không bị ảnh hưởng bởi nhiễu điện từ. Do đó nó được sử dụng rộng rãi và tiện lợi trong thông tin và điều khiển. Tuy nhiên nó không hoàn hảo, nhiều nguồn sáng khác nhau có thể phát ra hồng ngoại và có thể gây nhiễu đến thông tin này. Một cách để tránh những nguồn hồng ngoại khác là tạo ra một khóa. Do đó điều khiển từ xa dùng để điều biến hồng ngoại của nó tại một tần số nào đó. Đầu thu hồng ngoại ở TV/VCR sẽ đi theo tần số này mà bỏ đi các hồng ngoại khác nhận được. Khoảng tần số hay sử dụng là $30 \rightarrow 60$ kHz, tốt nhất là khoảng $36 \rightarrow 38$ kHz. Hồng ngoại phát ra từ các diode hồng ngoại theo các xung nhịp với tần số 36000 lần một giây phát ra các mức logic “0” và “1”.

Ứng dụng của tia hồng ngoại:

- Đo nhiệt độ.
- Phát nhiệt.
- Kỹ thuật hồng ngoại trong quân sự.
- Điều khiển điện tử.
- Truyền thông.

- Thiết bị nhìn đêm.
- Nghiên cứu thiên văn.
- Bảo mật tiền và tài liệu quý.

1.4.2. Led phát hồng ngoại [5]



Hình 1. 12. Hình ảnh led phát hồng ngoại

Led phát hồng ngoại được viết tắt là IR Red (Infra Red Led), hay còn gọi là nguồn phát tia hồng ngoại (Infra Red Emitters), giống như led bình thường (led phát quang- light emitting diode) và phát ra ánh sáng hồng ngoại. Nó được chế tạo bằng chất Arsenic-Galium (GaAs). Led hồng ngoại có tuổi thọ khoảng 100.000 giờ (khoảng 11 năm) với các tính năng như:

- Sử dụng trong điều khiển, remote hồng ngoại.
- Sử dụng vào các mạch ứng dụng đếm sản phẩm, đọc ecoder tốc độ động cơ,...
- Sử dụng mạch mô phỏng remote, mạch chống trộm bằng hồng ngoại, mạch dò đường,...

Thông số kĩ thuật

Điện Áp: 1.2 - 1.6v DC.

Dòng: 10 - 20mA.

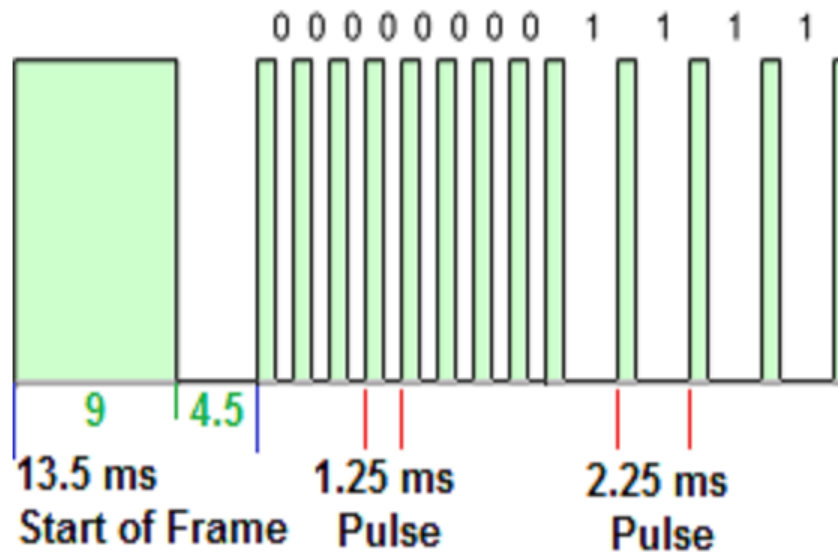
Bước Sóng: 940nm.

Kích Thước: LED Phi 5 (5mm), LED phi 3 (3mm).

Ánh sáng phát ra từ LED hồng ngoại không nhìn thấy được.

1.4.3. Phương thức giao tiếp hồng ngoại [6]

Bộ điều khiển và bộ thu từ xa IR tuân theo các giao thức chuẩn để gửi và nhận dữ liệu. Một số giao thức tiêu chuẩn là NEC, JVC, SIRC,... Giao thức NEC cũng là loại phổ biến và thường được dùng nhất trong các dự án vi điều khiển. Do đó, trong đồ án này em sử dụng remote bất kỳ có sẵn trên thị trường theo chuẩn NEC để làm mạch phân phát.



Hình 1. 13. Chuẩn giao tiếp NEC [6]

Phần Start: là 1 xung mở đầu gói tin kéo dài 9ms và theo sau là khoảng nghỉ 4.5ms, tổng thời gian truyền là 13.5ms.

Phần data: được truyền theo sau phần Start với bit có trọng số thấp nhất được truyền trước.

Bit 1: được thể hiện bằng 1 xung kéo dài khoảng 0.56ms và theo sau là khoảng nghỉ 1.69ms, tổng thời gian truyền 2.25ms.

Bit 0: được thể hiện bằng 1 xung kéo dài khoảng 0.625ms và theo sau là khoảng nghỉ 0.625ms, tổng thời gian truyền 1.25ms.

1.4.4. Truyền dữ liệu thông qua ánh sáng hồng ngoại [7]

Ánh sáng hồng ngoại được phát ra từ mọi vật liệu có nhiệt độ, từ mặt trời, bóng đèn cho đến các vật dụng trong phòng. Điều đó có nghĩa là có rất nhiều ánh sáng hồng ngoại gây nhiễu xung quanh chúng ta. Để ngăn tín hiệu nhiễu này ảnh hưởng đến tín hiệu IR (hồng ngoại), kỹ thuật điều chế tín hiệu được sử dụng.

Trong điều chế tín hiệu hồng ngoại, bộ mã hóa trên bộ phát hồng ngoại chuyển đổi tín hiệu nhị phân thành tín hiệu điện. Tín hiệu điện này được gửi đến đèn LED phát. Đèn LED phát sẽ chuyển đổi tín hiệu điện được điều chế thành ánh sáng hồng ngoại. Sau đó, bộ thu hồng ngoại giải mã tín hiệu ánh sáng hồng ngoại và chuyển đổi trở lại thành nhị phân trước khi truyền thông tin đến vi điều khiển.

Tín hiệu hồng ngoại được điều chế thành một chuỗi các xung ánh sáng được bật và tắt ở tần số cao được gọi là tần số sóng mang.

Tần số sóng mang được sử dụng bởi hầu hết các remote phát hồng ngoại là 38 kHz, vì nó rất hiếm trong tự nhiên và do đó có thể loại bỏ được nhiễu từ môi trường xung quanh. Bằng cách này, bộ thu hồng ngoại sẽ biết rằng tín hiệu 38 kHz được gửi từ máy phát mà không phải là từ môi trường xung quanh.

Led thu nhận được tất cả các tần số của ánh sáng hồng ngoại, nhưng thông qua bộ lọc thông dải nó chỉ cho phép bước sóng có tần số 38 kHz đi qua. Sau đó, tín hiệu này được khuếch đại bởi bộ tiền khuếch đại và chuyển đổi nó thành tín hiệu nhị phân trước khi gửi nó đến vi điều khiển.

1.5. Kết luận chương

Chương 1 đã giới thiệu qua về hệ thống điều khiển từ xa, tìm hiểu tổng quan và nghiên cứu về vi điều khiển PIC18F4520, các đặc tính của tia hồng ngoại, các thanh ghi trong sơ đồ các chân linh kiện và các cổng xuất nhập, các lệnh. Đây là những kiến thức cơ sở để thiết kế và lập trình cho thiết bị điều khiển từ xa dùng hồng ngoại sử dụng PIC18F4520 sẽ được trình bày trong chương 2 và chương 3 của đề án.

CHƯƠNG 2. THIẾT KẾ MẠCH ĐIỀU KHIỂN TỪ XA DÙNG CẢM BIẾN HỒNG NGOẠI SỬ DỤNG PIC18F4520

2.1. Xác định bài toán cho hệ thống

2.1.1. Xác định bài toán

Đề tài tập trung nghiên cứu, thiết kế mạch với các chức năng bật/tắt thiết bị, cụ thể là bật/tắt hệ thống đèn thông qua cảm biến hồng ngoại, là công nghệ thu phát tín hiệu từ xa.

Thiết kế hệ thống điều khiển từ xa bằng cảm biến hồng ngoại dùng vi điều khiển gồm các khối chính sau: khối nguồn, khối điều khiển, khối nhận và led thu hồng ngoại, khối chấp hành, khối reset, khối nút nhấn và khối hiển thị led.

2.1.2. Yêu cầu của bài toán thiết kế hệ thống điều khiển từ xa

Để đảm bảo được các yêu cầu và lợi ích của người sử dụng, hệ thống điều khiển từ xa có các yêu cầu sau:

- Tốc độ làm việc phải nhanh và chính xác, thời gian hoạt động được tính trên giây.
- Thiết bị phải cung cấp đủ nguồn tránh xảy ra trường hợp hỏng, cháy các linh kiện trong mạch để đảm bảo an toàn cho người dùng.
- Kết cấu phải đơn giản.
- Hiệu quả và năng suất làm việc cao với độ chính xác đúng nhiệm vụ đề ra.

Hệ thống điều khiển từ xa có hiệu quả cao là hệ thống đạt tốc độ điều khiển cực đại đồng thời đảm bảo độ chính xác trong phạm vi cho phép.

2.1.3. Giải pháp công nghệ

- Mạch phát dùng remote theo chuẩn giao tiếp NEC.
- Mạch thu dùng vi điều khiển.
- Dùng led thu hồng ngoại nhận tín hiệu phát.
- Các đèn báo trạng thái dùng led.

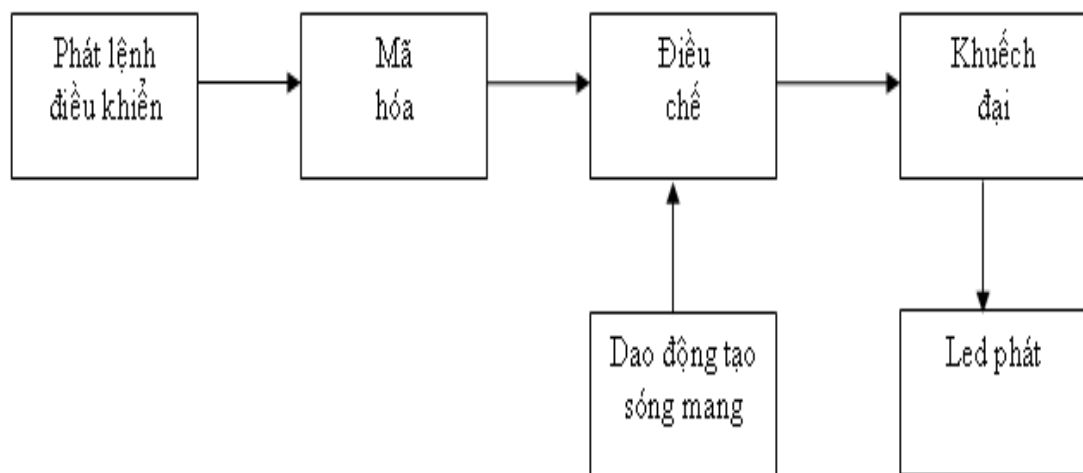
2.1.4. Giải pháp thiết kế

- Công cụ lập trình: phần mềm MikroC Pro for PIC.
- Thiết kế mạch in bằng phần mềm Altium.

2.2. Sơ đồ khối hệ thống điều khiển từ xa bằng cảm biến hồng ngoại

Một hệ thống điều khiển từ xa bao gồm phần phát và phần thu. Phần phát phát đi sóng điện từ hoặc ánh sáng hồng ngoại đến phần thu. Phần thu nhận được các tín hiệu này sẽ biến đổi thành tín hiệu điện điều khiển thiết bị.

2.2.1. Sơ đồ hệ thống phần phát



Hình 2. 1. Sơ đồ phần phát hệ thống điều khiển từ xa dùng hồng ngoại

Trong đồ án này, em sử dụng remote theo chuẩn giao tiếp NEC để làm mạch phần phát, tạo ra lệnh điều khiển, mã hóa và phát tín hiệu đến máy thu:



Hình 2. 2. Remote điều khiển từ xa [5]

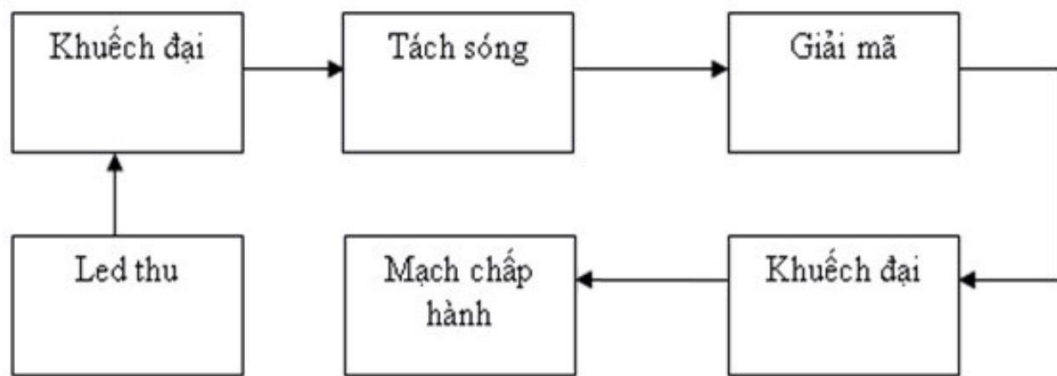
Nhấn nút nhấn trên remote để tạo ra lệnh điều khiển, mỗi phím chức năng tương ứng với một số thập phân.

Mã hóa, chuyển đổi các lệnh điều khiển thành mã nhị phân tương ứng với dạng mã lệnh tín hiệu số gồm các bit 0 và 1. Mã hóa bằng phương pháp điều chế mã xung đơn giản và dễ thực hiện.

Tạo sóng mang tần số ổn định mang tín hiệu điều khiển truyền ra môi trường, khuếch đại đủ lớn để led phát hồng ngoại phát tín hiệu đến máy thu.

2.2.2. Sơ đồ hệ thống phần thu

Để thực hiện được thiết kế và chế tạo ra hệ thống điều khiển từ xa dùng cảm biến hồng ngoại dựa trên sơ đồ khối em đưa ra sơ đồ thiết kế phần thu như sau:

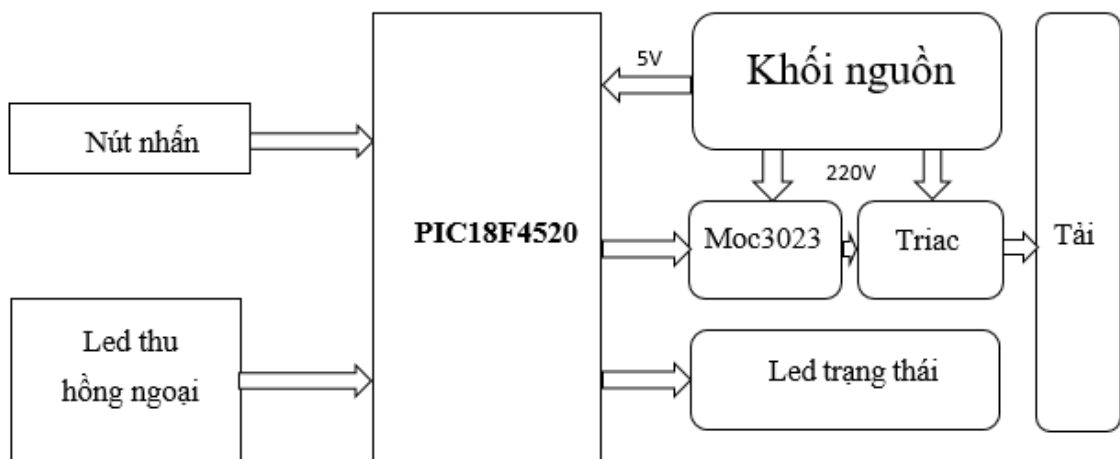


Chức năng phần mạch thu: Khi thu được tín hiệu từ máy phát, loại bỏ

Hình 2. 3. Sơ đồ hệ thống phần thu điều khiển từ xa dùng hồng ngoại

sóng mang, giải mã tín hiệu điều khiển thành các lệnh riêng biệt, sau đó mỗi lệnh sẽ điều khiển được các thiết bị cụ thể.

2.2.3. Sơ đồ khối hệ thống tổng quát



Hình 2. 4. Sơ đồ khối hệ thống điều khiển từ xa

➤ Chức năng của các khối:

- PIC18F4520: Giống như bộ não của bảng mạch, đưa ra các chỉ thị điều khiển hoạt động của hệ thống.
- Khối nguồn: cung cấp nguồn cho các khối hoạt động.
- Moc3023: Dùng để cách ly điện áp cao với mạch điều khiển và cho phép Triac dẫn hoặc không.
- Triac: Dùng để đóng cắt điện cho tải cao áp xoay chiều.
- Led thu hồng ngoại: thu tín hiệu hồng ngoại do máy phát truyền tới và biến đổi thành tín hiệu điều khiển.
- Nút nhấn: bật/tắt thiết bị và chọn mode học lệnh.
- Led status: led tương ứng với từng thiết bị khi bật/tắt.
- Tải: các thiết bị điện muốn điều khiển.

➤ Nguyên lý hoạt động:

Cấp nguồn 12V cho mạch qua IC ổn áp LM7805 tạo ra điện áp 5V cấp vào các khối trong hệ thống. Khi vào trạng thái học lệnh rồi nhấn bất kì phím nào trên remote, led phát trên remote có nhiệm vụ biến dòng điện thành quang phát xạ ra môi trường. Led thu nhận tín hiệu đó biến đổi từ quang thành điện. Sau đó khuếch đại, lọc tín hiệu, tách sóng và loại bỏ sóng mang chỉ giữ lại tín hiệu như tín hiệu đã gửi đi từ máy phát. Giải mã tín hiệu điều khiển dưới dạng các bit nhị phân hay các dạng khác để đưa đến khối chấp hành cụ thể. Cuối cùng nhờ nguyên lý kích mở của Triac thông qua sự kích dẫn của Moc để điều khiển thiết bị điện 220V tắt/mở theo ý muốn.

2.3. Một số linh kiện sử dụng trong mạch

2.3.1. Vi điều khiển PIC18F4520 [12]



Hình 2. 5. Hình ảnh vi điều khiển PIC18F4520 [5]

a. Đặc điểm của vi điều khiển Pic18F4520

- Hiệu năng cao, tiêu thụ năng lượng ít.
- Kiến trúc RISC: tốc độ của CPU khá cao, có 75 lệnh mạnh. Tần số làm việc của xung clock là 40Mhz.
- Các bộ nhớ chương trình và dữ liệu cố định: 256 bytes EEPROM, 1536 bytes SRAM.
- Ngoài ra còn có những ngoại vi tiêu biểu như: các kênh PWM, ADC 10 bit, các bộ đếm thời gian thực, USART có chức năng truyền tin nối tiếp và các nguồn ngắt trong, ngắt ngoài.

b. Hoạt động vào/ra

Tên	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
LATA	LATA7	LATA6	Thanh ghi chốt dữ liệu PORTA					
TRISA	TRISA7	TRISA6	Thanh ghi hướng dữ liệu PORTA					
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
CVRCON	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0

Bảng 2. 1. Các thanh ghi liên quan đến PORTA

Tên	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
LATB	Thanh ghi chốt dữ liệu PORTB							
TRISB	Thanh ghi hướng dữ liệu PORTB							
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
INTCON2	RBPUR	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP
INTCON3	INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0

Bảng 2. 2. Các thanh ghi liên quan đến PORTB

Tên	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
LATC	Thanh ghi chốt dữ liệu của PORTC (Chốt dữ liệu đọc ghi)							
TRISC	Thanh ghi chọn hướng dữ liệu của PORTC							

Bảng 2. 3. Các thanh ghi liên quan đến PORTC

Tên	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
LATD	Thanh ghi chốt dữ liệu của PORTD (Chốt dữ liệu đọc ghi)							
TRISD	Thanh ghi chọn hướng dữ liệu của PORTD							
TRISE(1)	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0

Bảng 2. 4. Các thanh ghi liên quan đến PORTD

Tên	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTE	—	—	—	—	RE3	RE2	RE1	RE0
LATE(2)	—	—	—	—	—	Thanh ghi xuất dữ liệu LATE		
TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0

Bảng 2. 5. Các thanh ghi liên quan đến PORTE

c. Hoạt động ngắt

❖ Tổ chức ngắt của PIC 18F4520

Vi điều khiển PIC 18F4520 có 10 thanh ghi điều khiển hoạt động ngắt: Gồm có các thanh ghi điều khiển ngắt (INTCON, INTCON2, INTCON3), các thanh ghi đề nghị ngắt ngoại vi (PIR1,2), cho phép ngắt ngoại vi (PIE1,2) và ưu tiên ngắt ngoại vi (IPR1,2), kèm theo đó là thanh ghi RCON giúp điều khiển chức năng reset. Thanh ghi RCON: bit cho phép ưu tiên ngắt là bit thứ 7, việc cho phép hay không cho phép ngắt được biểu diễn bằng số “1” và “0”.

❖ Các thanh ghi điều khiển ngắt INTCON

Các thanh ghi INTCON (Interrupt Control) có thể ghi/đọc theo cả byte hoặc từng bit. Các thanh ghi này gồm các bit cho phép/cấm các nguồn ngắt, các bit đặt mức ưu tiên, các bit cờ ngắt.

Thanh ghi điều khiển ngắt INTCON gồm 7 bit với các chức năng và nhiệm vụ khác nhau:

bit 7 **GIE/GIEH**: Bit cho phép ngắt toàn cục.

Sẽ được chia thành 2 trường hợp: khi IPEN=0 sẽ ngắt tất cả và không có ưu tiên ngắt và IPEN = 1 sẽ có ưu tiên ngắt cao.

bit 6 **PEIE/GIEL**: Bit cho phép ngắt ngoại vi.

Cũng giống như bit 7. Khi IPEN=0 thì sẽ cho phép ngắt tất cả, còn khi IPEN = 1 thì sẽ ưu tiên ngắt thấp.

bit 5 **TMR0IE**: Bit cho phép ngắt tràn Timer0 (TMR0)

bit 4 **INT0IE**: Bit cho phép ngắt ngoài INT0.

bit 3 **RBIE**: Bit cho phép ngắt do thay đổi mức trên PortB.

bit 2 **TMR0IF**: Bit cờ báo ngắt tràn Timer0 (TMR0).

1 = Tràn thanh ghi TMR0 của Time0 (phải được xóa bằng phần mềm).

0 = Không xảy ra tràn thanh ghi TMR0.

bit 1 **INT0IF**: Bit cờ báo ngắt ngoài INT0.

1 = Có ngắt ngoài INT0 (phải được xóa bằng phần mềm).

0 = Chưa phát hiện ngắt ở chân INT0.

bit 0 **RBIF**: Bit cờ ngắt báo đã thay đổi mức trên PortB.

1 = Ít nhất một bit từ RB7, RB4 có sự thay đổi trạng thái (bit này phải được xóa bằng phần mềm)

0 = Không có sự thay đổi trạng thái trên các chân RB7, RB4.

- Thanh ghi điều khiển ngắt 2: **INTCON2**

bit 7 **RBPU**: Bit cho phép treo các chân PORTB (“0” cho phép, ”1” cấm treo).

bit 6 **INTEDG0**: Bit lựa chọn sườn xung cho ngắt ngoài INT0.

bit 5 **INTEDG1**: Bit lựa chọn sườn xung cho ngắt ngoài INT1.

bit 4 **INTEDG2**: Bit lựa chọn sườn xung cho ngắt ngoài INT2.

(Bit 4, 5, 6 khi ngắt bằng sườn dương và âm sẽ được biểu thị lần lượt bằng “1” và “0”).

bit 3: Bit không được định nghĩa.

bit 2 **TMR0IP**: Bit lựa chọn mức ưu tiên ngắt cho ngắt tràn Timer0 (TMR0).

bit 1: Bit không được định nghĩa.

bit 0 **RBIP**: Bit lựa chọn mức ưu tiên ngắt do thay đổi PortB.

(Hai bit 0 và bit 2 khi ưu tiên cao sẽ được biểu thị bằng số “1” và thấp bằng số “0”).

d. Các thanh ghi của EEPROM

- **Thanh ghi điều khiển EEPROM 1: EECON1**

Gồm 7 bit với các chức năng và nhiệm vụ khác nhau:

bit 7 **EEPGD**: Bit lựa chọn bộ nhớ dữ liệu EEPROM hay bộ nhớ chương trình Flash:

1 = Truy cập bộ nhớ chương trình Flash.

0 = Truy cập bộ nhớ dữ liệu EEPROM.

bit 6 **CFGS**: Bit lựa chọn cấu hình hoặc bộ nhớ chương trình Flash/dữ liệu EEPROM.

1 = Truy cập thanh ghi cấu hình.

0 = Truy cập bộ nhớ chương trình Flash hoặc dữ liệu EEPROM.

bit 5: Không sử dụng, đọc trả về giá trị '0'.

bit 4 **FREE**: Bit cho phép xóa hàng bộ nhớ Flash.

1 = Xóa hàng bộ nhớ chương trình được thiết lập, địa chỉ chứa trong thanh ghi TBLPTR, được xóa từ lệnh WR kế tiếp.

0 = Chỉ thực hiện ghi.

bit 3 **WREERR**: Bit cờ lỗi bộ nhớ chương trình Flash/ bộ nhớ dữ liệu EEPROM.

1 = Lỗi hoạt động ghi (hoạt động ghi bị kết thúc trước).

0 = Ghi hoàn thành.

bit 2 **WREN**: Bit cho phép ghi bộ nhớ chương trình Flash/Dữ liệu EEPROM.

1 = Cho phép ghi vào bộ nhớ chương trình Flash /dữ liệu EEPROM.

0 = Không cho phép ghi.

bit 1 **WR**: Bit điều khiển ghi.

1 = Khởi tạo quá trình xóa/ghi bộ nhớ dữ liệu EEPROM hoặc xóa bộ nhớ chương trình hoặc ghi bộ nhớ chương trình. (Được xóa bằng phần cứng khi việc ghi hoàn thành. Thiết lập được bằng phần mềm nhưng không được xóa).

0 = Quá trình ghi hoàn thành.

bit 0 RD: Bit điều khiển đọc.

1 = Khởi tạo quá trình đọc bộ nhớ EEPROM (Đọc mất một chu kỳ máy. Bit RD được xóa bằng phần cứng. Thiết lập được bằng phần mềm nhưng không được xóa. Bit RD không được thiết lập khi EEPGD = 1 hoặc CFGS = 1).

0 = Không khởi tạo quá trình đọc EEPROM.

- Thanh ghi điều khiển EEPROM 2: EECON2

Thanh ghi EECON2 không phải là thanh vật lý, nó được dành riêng cho việc ghi và xóa bộ nhớ. Đọc EECON2 sẽ được '0'.

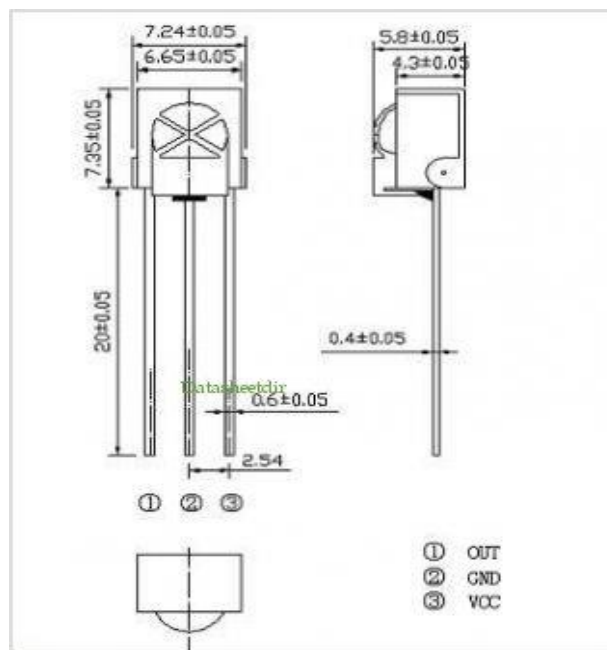
- Thanh ghi dữ liệu EEPROM: EEDATA

Thanh ghi EEDATA có 8 bit, là thanh ghi đệm dữ liệu cho bộ nhớ dữ liệu EEPROM, được sử dụng để truy cập vào dữ liệu của bộ nhớ (Cho phép đọc ghi bằng phần mềm, mỗi ô nhớ của EEPROM có 8 bit).

- Thanh ghi địa chỉ EEPROM: EEADR

Thanh ghi EEADR có 8 bit, là thanh ghi địa chỉ của bộ nhớ dữ liệu EEPROM. 8 bit của thanh ghi EEADR được sử dụng để địa chỉ hóa 256 ô nhớ của EEPROM từ 00h đến FFh.

2.3.2. Led thu hồng ngoại [9]



Hình 2. 6. Led thu hồng ngoại

- Đối với modul mắt thu hồng ngoại có 2 loại: vỏ bằng sắt và vỏ bằng nhựa.

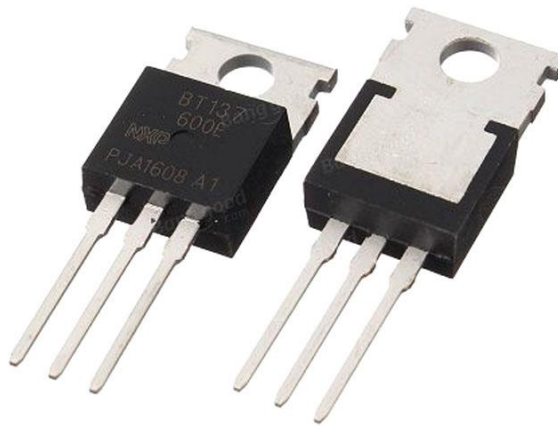
- Trong mạch thu tín hiệu hồng ngoại sử dụng mắt nhận tia hồng ngoại vỏ sắt có tên 1838B. Dùng loại modul này chống được nhiễu bên ngoài và thu được tín hiệu xung quanh nó.

Thông số kỹ thuật:

- Khoảng cách truyền: tối đa 10m (tùy thuộc vào môi trường xung quanh và các nhân tố khác).
- Góc truyền hiệu quả: trong 60 độ.
- Vật liệu cấu tạo: 0.125mmPET, tuổi thọ sử dụng 20000 giờ.

- Điện áp hoạt động: 3-5VDC.
- Dòng thụ động: 3-5 μ A.
- Dòng chủ động: 3-5mA.
- Giao tiếp ngõ ra: TTL hoặc CMOS.
- Gồm 3 chân như hình trên:
 - Chân VCC.
 - Chân GND.
 - Chân tín hiệu.

2.3.3. Triac



Hình 2. 7. Hình ảnh Triac [5]

Triac là từ được viết tắt của từ: Triode for Alternating Current. Triac còn là một linh kiện điện tử bán dẫn chuyên dụng được sử dụng rất nhiều trong các bo mạch điện tử để đóng cắt điện xoay chiều cho các phụ tải.

Triac có thể điều khiển cho mở dẫn dòng bằng cả xung dương (dòng đi vào cực điều khiển) lẫn xung âm (dòng đi ra khỏi cực điều khiển). Tuy nhiên xung dòng điều khiển âm có độ nhạy kém hơn, nghĩa là để mở được TRIAC sẽ cần một dòng điều khiển âm lớn hơn so với dòng điều khiển dương. Vì vậy trong thực tế để đảm bảo tính đối xứng của dòng điện qua TRIAC thì sử dụng dòng điện dương là tốt hơn cả.

Thông số kĩ thuật:

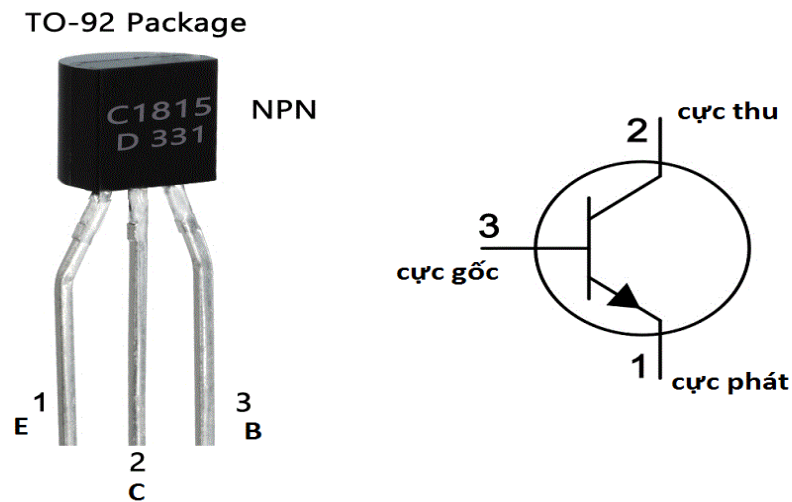
Điện áp cực đại: 600V

Dòng điện thuận cực đại: 8A

Điện áp điều khiển mở van: 1.5V

Dòng điều khiển mở van: 100mA

Nhiệt độ làm việc: $-40^{\circ}\text{C} \sim 125^{\circ}\text{C}$

2.3.4. Transistor

Hình 2. 8. Hình ảnh và kí hiệu của transistor [5]

Transistor là một linh kiện bán dẫn thường được sử dụng như một thiết bị khuếch đại hoặc một khóa điện tử.

Transistor là khối đơn vị cơ bản xây dựng nên cấu trúc mạch ở máy tính điện tử và tất cả các thiết bị điện tử hiện đại khác.

Các thông số kĩ thuật:

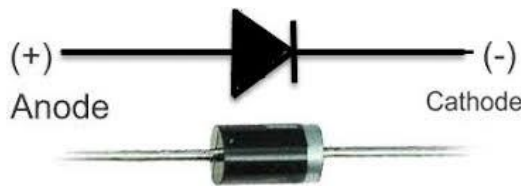
- Loại gói: TO-92
- Loại bóng bán dẫn: NPN
- Bộ IC dòng điện tối đa (I_C): 150mA
- Điện áp cực đại Collector-Emitter (V_{CE}): 50V
- Điện áp cực đại Collector-Base (V_{CB}): 60V
- Điện áp cực đại cực phát (V_{EBO}): 5V
- Max Collector Dissipation (P_c): 400 miliWatt

- Tần số chuyển đổi tối đa (fT): 80 MHz
- Mức tăng dòng DC tối thiểu và tối đa (h FE): 70 – 700
- Lưu trữ tối đa và nhiệt độ hoạt động phải là: -55 đến +150 C.

Mỗi transistor đều có ba cực:

- Cực gốc (*base*).
- Cực thu (*collector*).
- Cực phát (*emitter*).

2.3.5. Diot



Hình 2. 9. Diode [5]

Điot là 1 loại linh kiện điện tử được cấu tạo từ 2 lớp bán dẫn tiếp xúc nhau.

Điot có 2 cực là Anode (A) và Katot (K). Nó chỉ cho dòng điện một chiều đi từ A sang K, coi như van một chiều trong dòng điện và được ứng dụng rộng rãi.

2.3.6. IC ổn áp 7805



Hình 2. 10. IC ổn áp 7805 [5]

Dùng để ổn định điện áp đầu ra là dương 5V.

Thông số kĩ thuật:

- Điện áp vào lớn nhất: 20V

- Điện áp vào nhỏ nhất: 7V
- Dòng đầu ra: 1.5A
- Điện áp ổn định: 5V

L7805C2T gồm 3 chân:

1-Vin: chân nguồn đầu vào.

2-GND: chân nối đất.

3 -Vout: chân nguồn đầu ra.

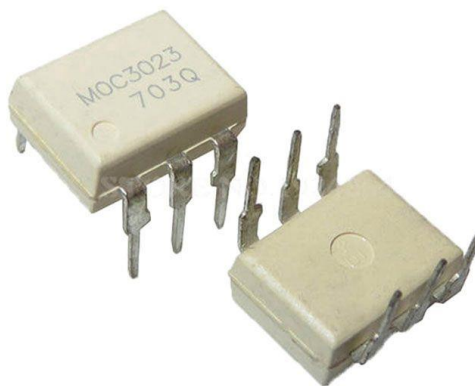
2.3.7. Thạch anh 20MHz



Hình 2. 11. Thạch anh 20MHz [5]

- Thạch Anh giúp cho các dụng cụ đo lường điện tử có được những dao động ổn định hơn.
- Là một phần không thể thiếu của mạch khuếch đại tần số cho các hệ thống liên lạc: điện thoại, radio, anten,

2.3.8. MOC 3023



Hình 2. 12. Opto Khuếch Đại Thuật Toán MOC3023 DIP6 [5]

MOC3023 DIP6 là cách ly quang (hay còn gọi là OPTO) là một linh kiện bán dẫn cấu tạo gồm 1 bộ phát quang và một cảm biến quang tích hợp trong 1 khối bán dẫn.

Bộ phát quang là 1 diode phát quang dùng để phát ra ánh sáng kích cho các cảm biến quang dẫn, còn cảm biến quang là phototran.

Thông số kỹ thuật:

- Số chân: 6
- Loại đầu ra: PhotoTriac
- Số lượng kênh: 1
- Điện áp cách ly: 5000 Vrms
- If - Dòng thuận: 50 mA
- Ff - Điện áp thuận: 1.15V
- Pd - Tiêu tán nguồn: 330 mW
- Điện áp đầu ra trạng thái tắt - VDRM: 400V
- Dòng đầu ra liên tục tối đa: 100 mA
- Nhiệt độ làm việc tối thiểu: - 40°C
- Nhiệt độ làm việc tối đa: + 100°C
- Sê-ri: MOC3023
- Chiều cao: 3.53 mm
- Chiều dài: 8.89 mm
- Chiều rộng: 6.6 mm
- Kiểu gắn: Through Hole
- Dòng kích hoạt tối đa: 5 mA
- Thời gian bật tối đa: 20 us

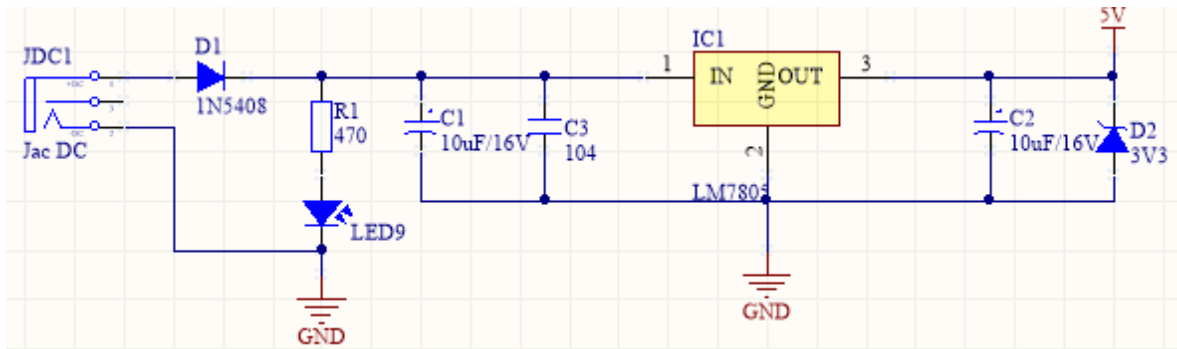
2.4. Kết luận chương

Ở chương 2 đã nghiên cứu về cấu trúc của hệ thống cũng như các linh kiện liên quan sử dụng để thiết kế mạch điều khiển từ xa. Việc lựa chọn các linh kiện và bộ vi điều khiển có ý nghĩa và cơ sở khoa học trong việc thiết kế mô hình sẽ được trình bày chi tiết trong chương 3.

CHƯƠNG 3. THIẾT KẾ MÔ HÌNH ĐIỀU KHIỂN TỪ XA

3.1. Các mạch điện tử sử dụng trong thiết kế mạch điều khiển từ xa

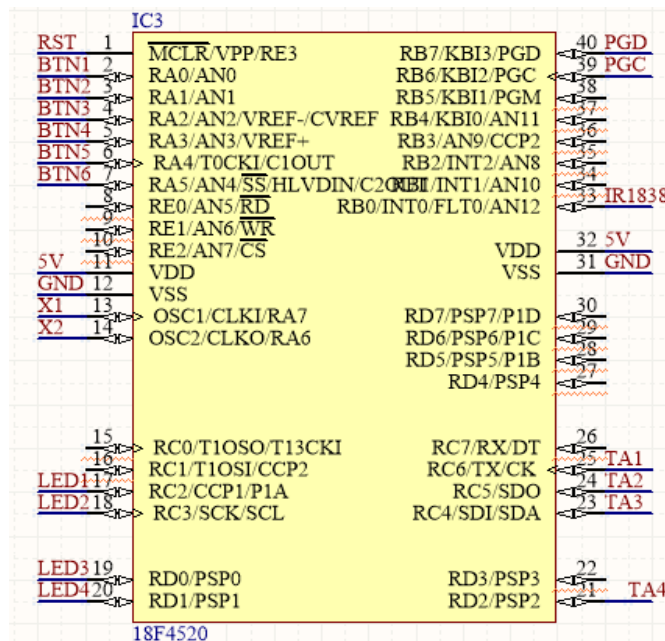
3.1.1. Khối nguồn



Hình 3. 1. Khối nguồn

Khối nguồn sử dụng nguồn JDC1 12V để cấp nguồn. Diode và led1 dùng để bảo vệ chân cấp nguồn và báo hiệu có nguồn cấp vào. Sử dụng IC ổn áp L7805C2T để ổn định điện áp dương đầu ra 5V cung cấp điện năng cho các khối trong mạch làm việc. Ở đầu vào và đầu ra của IC ổn áp đều mắc thêm tụ, gồm cả tụ hóa (có phân cực) và tụ keo (không phân cực) để ổn định điện áp.

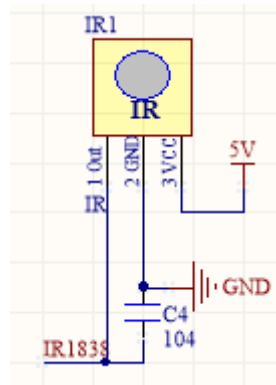
3.1.2. Khối điều khiển



Hình 3. 2. Khối điều khiển Pic18f4520

Vi điều khiển PIC18F4520 đóng vai trò như bộ não điều khiển hoạt động các khối của hệ thống.

3.1.3. Khối LED thu hồng ngoại

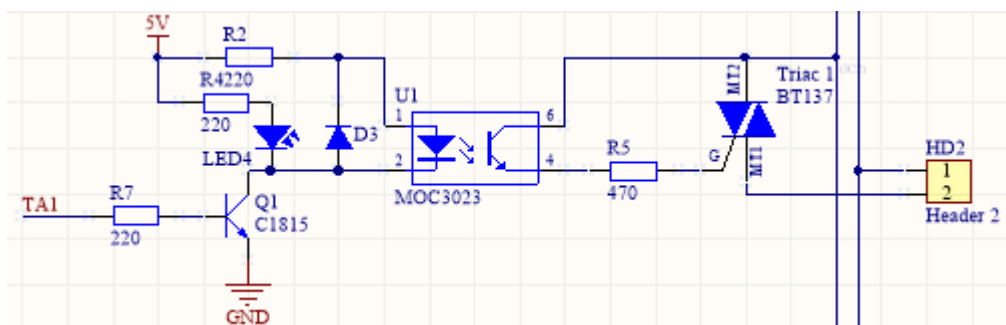


Hình 3. 3. Khối led thu hồng ngoại

Khi nhận được chuỗi ánh sáng hồng ngoại từ mạch phát đến, xác nhận được mã địa chỉ này xuất phát từ đúng chiếc điều khiển đã được cài đặt trước, led thu hồng ngoại nhận tín hiệu đó biến đổi từ quang thành điện, sau đó đưa qua bộ khuếch đại, tách sóng loại bỏ sóng mang và giải mã tín hiệu điều khiển đưa ra chân 3 (Out).

Mắt nhận rất dễ bị nhiễu do nguồn nuôi nên ở chân data đưa về PIC mắc thêm tụ C4 để lọc tín hiệu nhiễu.

3.1.4. Khối chấp hành



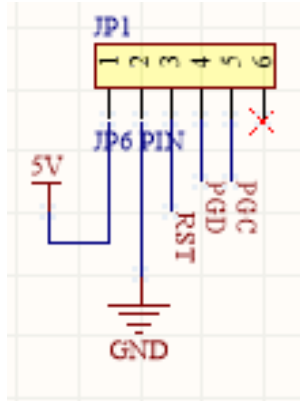
Hình 3. 4. Khối chấp hành

Sử dụng Triac để đóng/cắt thiết bị. Ở đề tài này em sử dụng Triac BT137 điều khiển bằng cách ly quang sử dụng MOC 3023 đóng/ngắt được 4 thiết bị trong nhà. Khi cấp nguồn 5V, Q1 dẫn, led 4 sáng báo hiệu có nguồn, MOC 3023 đóng kích Triac 1 dẫn cho dòng 220V đi qua, thiết bị mở. Ngược lại khi

Q1 không dẫn, MOC 3023 ngắt, Triac ngắt không cho dòng điện chạy qua, thiết bị tắt. Diode D3 dùng để tránh dòng ngược trong trường hợp hỏng transistor.

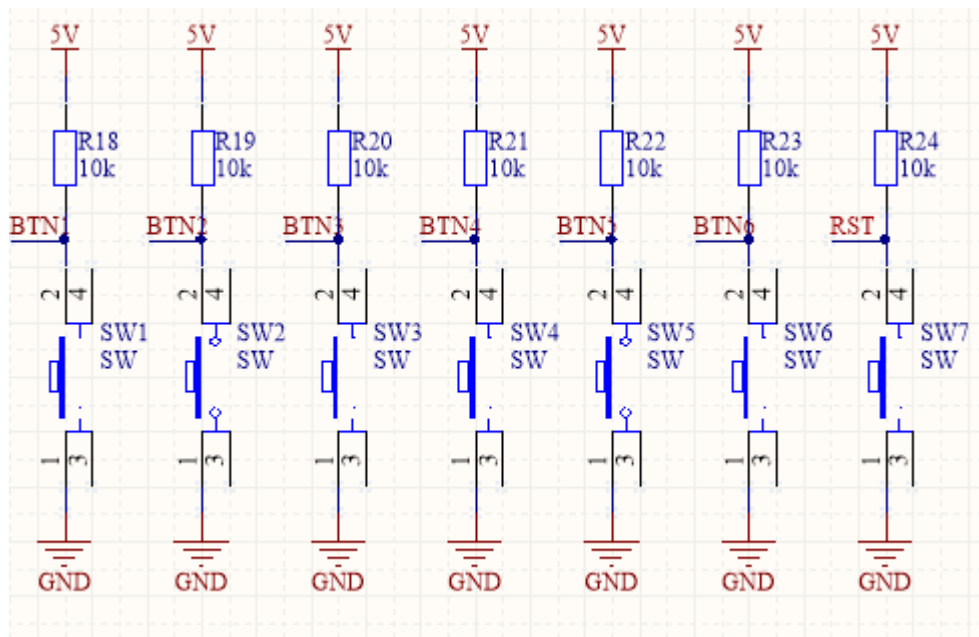
3.1.5. Khởi Reset

Reset được nối với chân 1 (RE3/MCLR) của vi điều khiển để tránh trường hợp mạch bị đơ hoặc nhiễu tín hiệu thì mạch sẽ hoạt động lại từ đầu.



Hình 3. 5. Khởi reset

3.1.6. Khởi nút nhấn

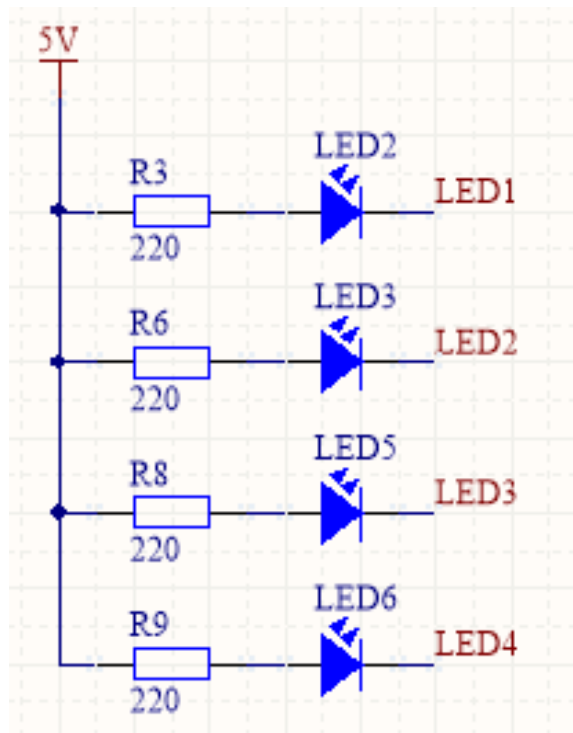


Hình 3. 6. Khởi nút nhấn

Do điều khiển 4 thiết bị nên mạch có 4 nút nhấn điều khiển riêng cho từng thiết bị, BTN5 điều khiển cho cả 4 thiết bị và BTN6 là nút nhấn để học lệnh.

Các nút nhấn được nối với chân VCC và GND qua con trở 10k để điều khiển trực tiếp các thiết bị ngay trên mạch khi không có remote.

3.1.7. Khối led status

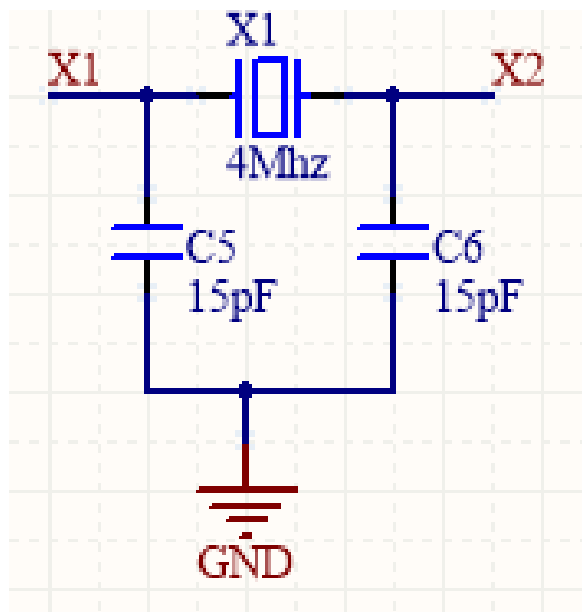


Hình 3. 7. Khối hiển thị led trạng thái

Khối này để hiển thị trạng thái của các led tương ứng với các thiết bị.

3.1.8. Khối thạch anh (20 MHz)

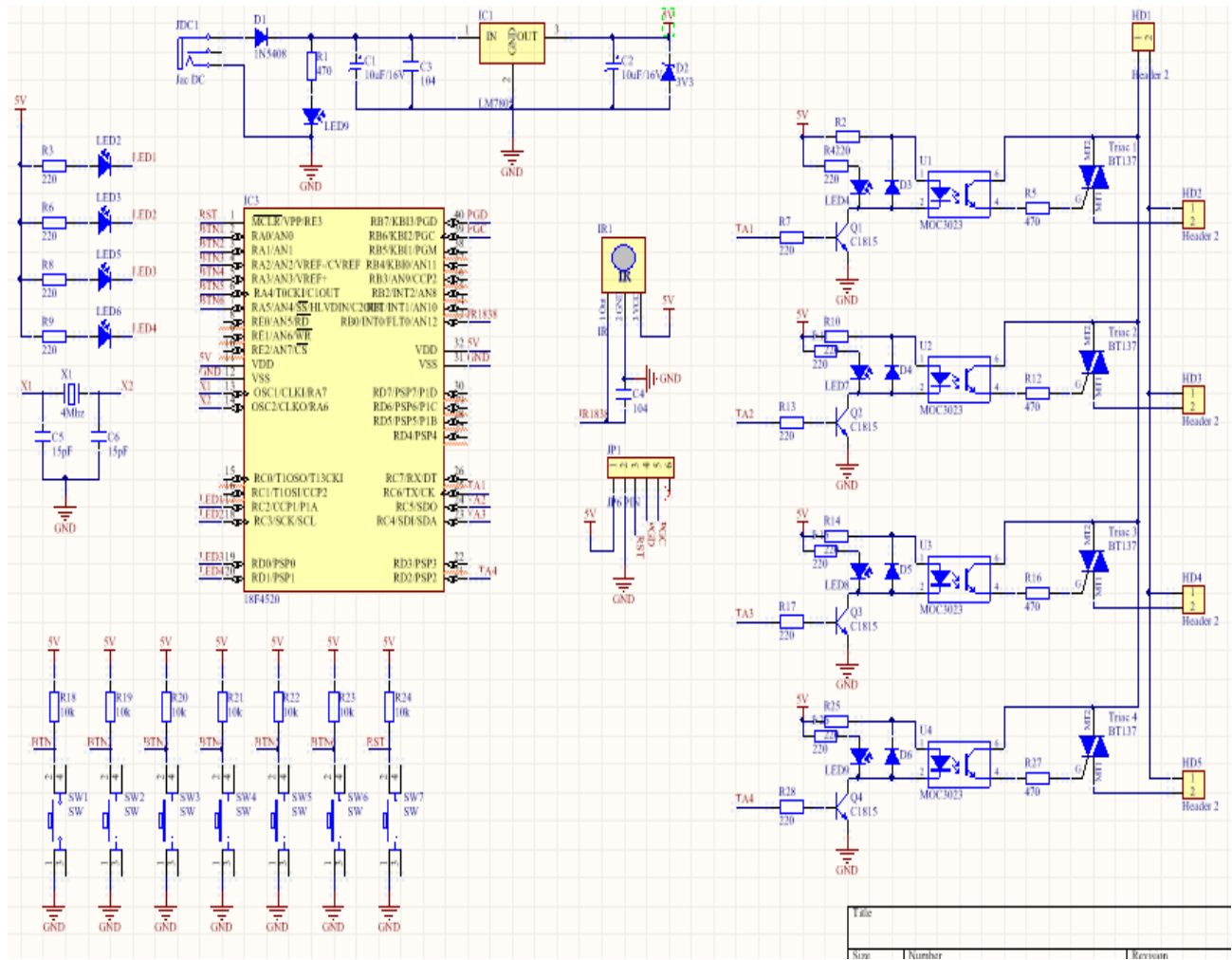
Dùng 2 tụ C5 và C6 để lọc nhiễu.



Hình 3. 8. Khối tạo dao động thạch anh 20MHz

3.2. Sơ đồ nguyên lí

Sau khi thiết kế các khối cần thiết, em có mạch nguyên lí sau:

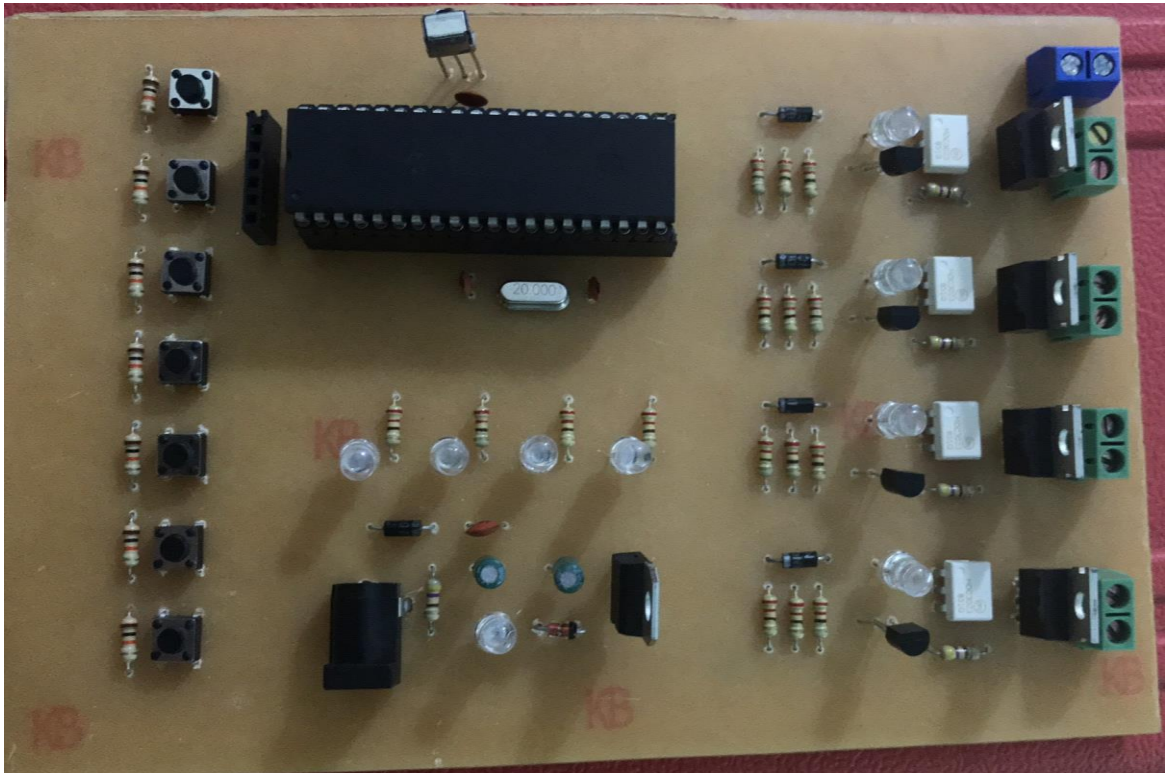


Hình 3. 9. Sơ đồ nguyên lí mạch điều khiển từ xa cảm biến hồng ngoại

3.3. Mạch PCB trên phần mềm Altium

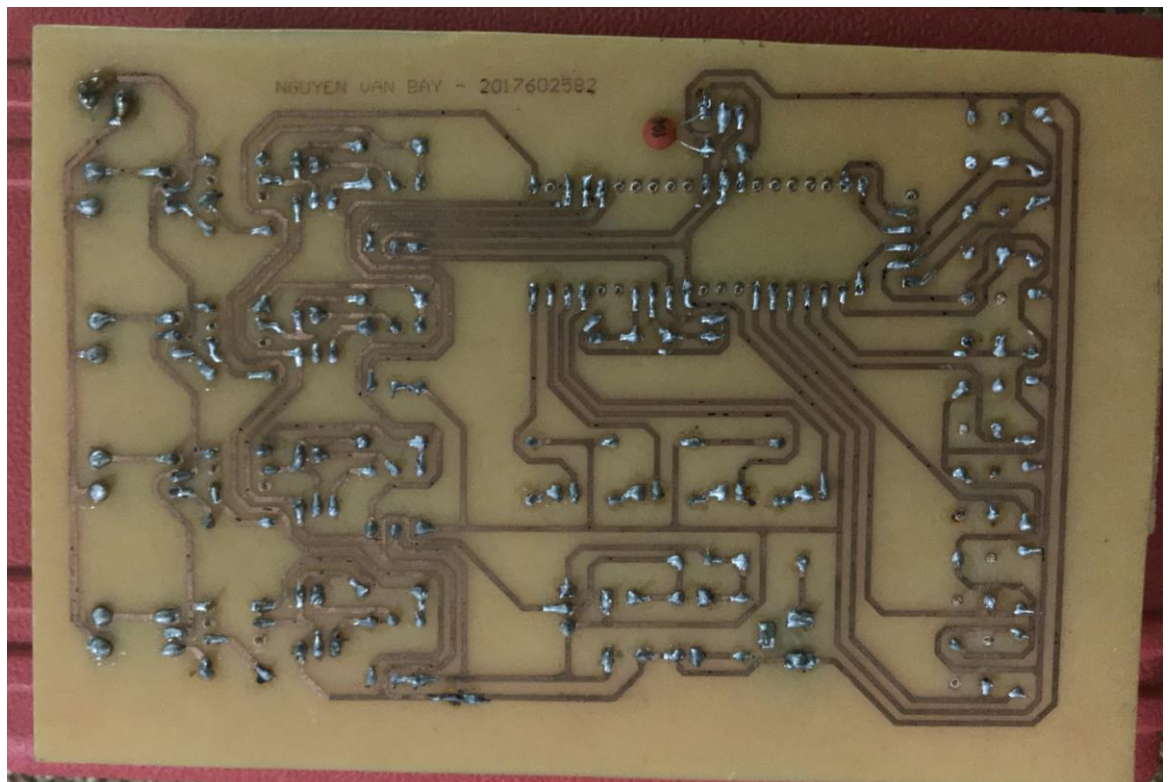
3.4. Mạch hoàn thiện

Mặt trước:



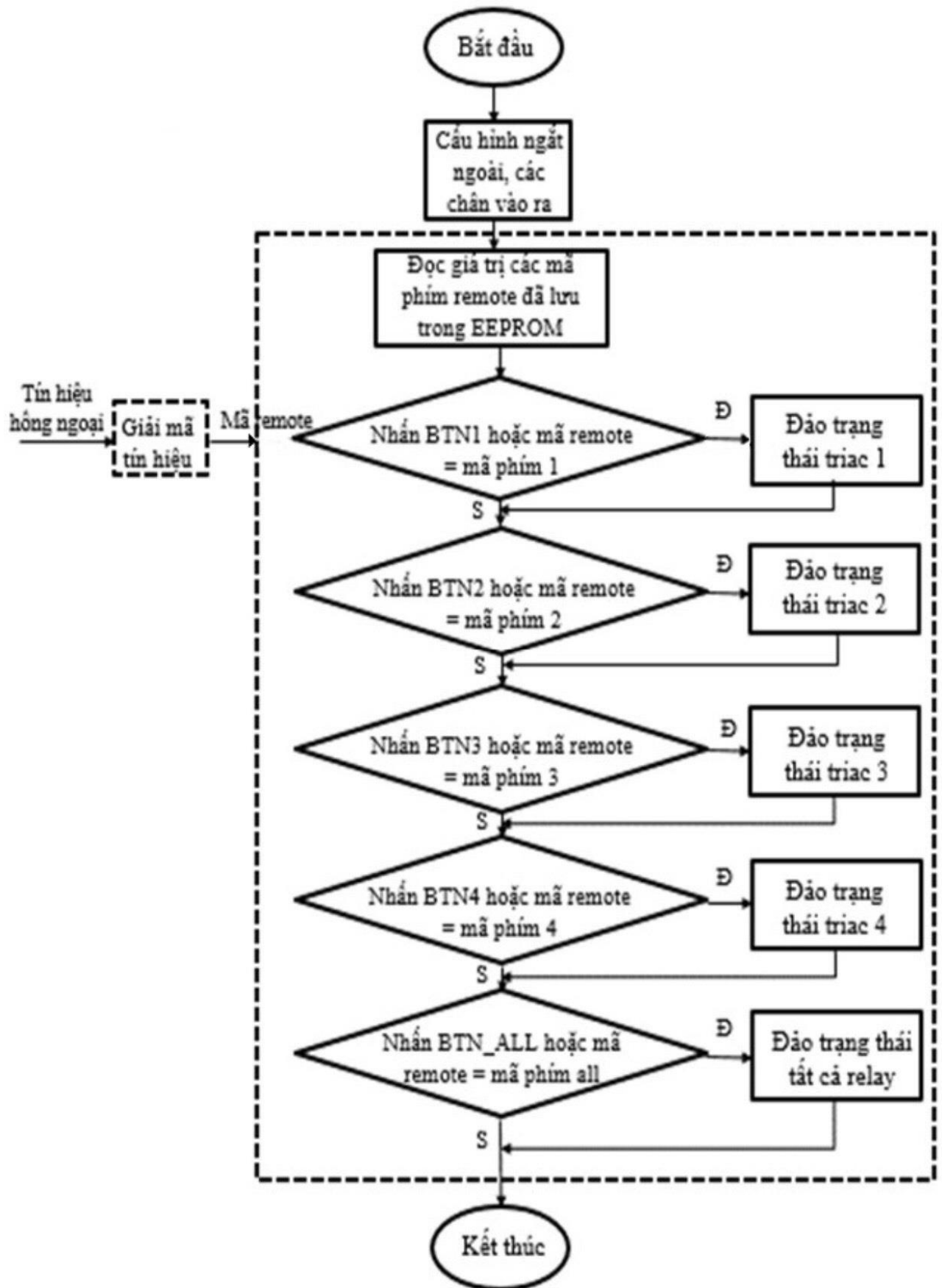
Hình 3. 12. Mặt trước mạch thực tế

Mặt sau:



Hình 3. 13. Mặt sau mạch thực tế

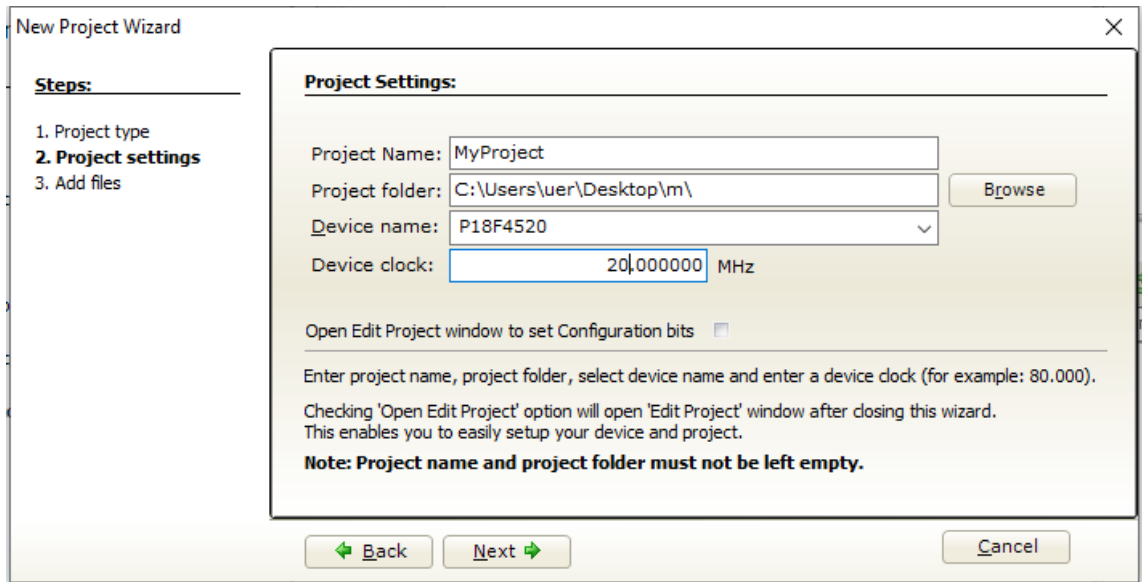
3.5. Lưu đồ thuật toán



Hình 3. 14. Lưu đồ thuật toán

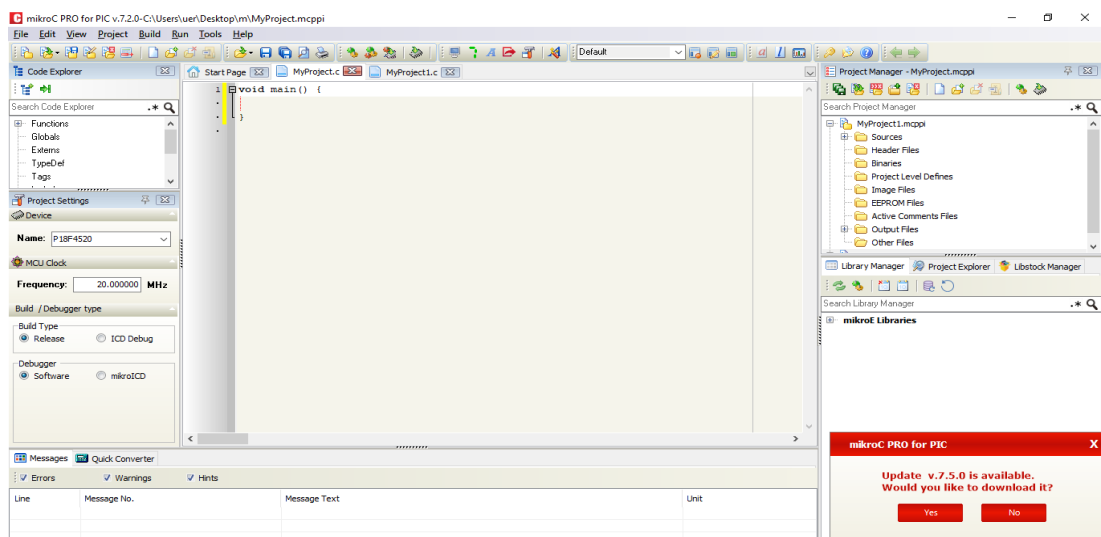
3.6. Các bước lập trình cho vi điều khiển

1. Mở phần mềm MikroC pro for PIC.
2. Vào New Project, chọn tên chương trình, đường dẫn, thiết bị và xung clock



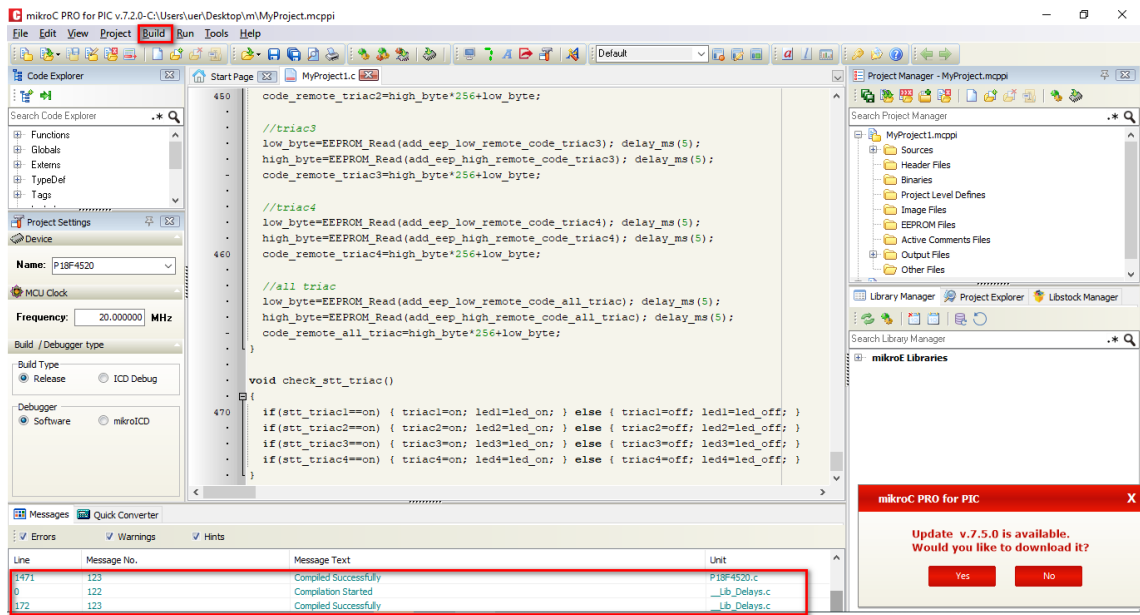
Hình 3. 15. Cài đặt cấu hình, thông số cho vi điều khiển

3. Nhấn Next, tích chọn “Include all” để cài đặt cấu hình mặc định bao gồm tất cả các thư viện cần thiết.
4. Hiện thị màn hình viết code:



Hình 3. 16. Hiện thị màn hình viết code

5. Chạy và kiểm tra lỗi chương trình:
6. Kích vào biểu tượng Build trên màn hình để chạy chương trình.



Hình 3. 17. Chạy và kiểm tra lỗi chương trình

7. Cuối cùng nạp chương trình cho vi điều khiển.

3.7. Mô hình sản phẩm hoàn thiện



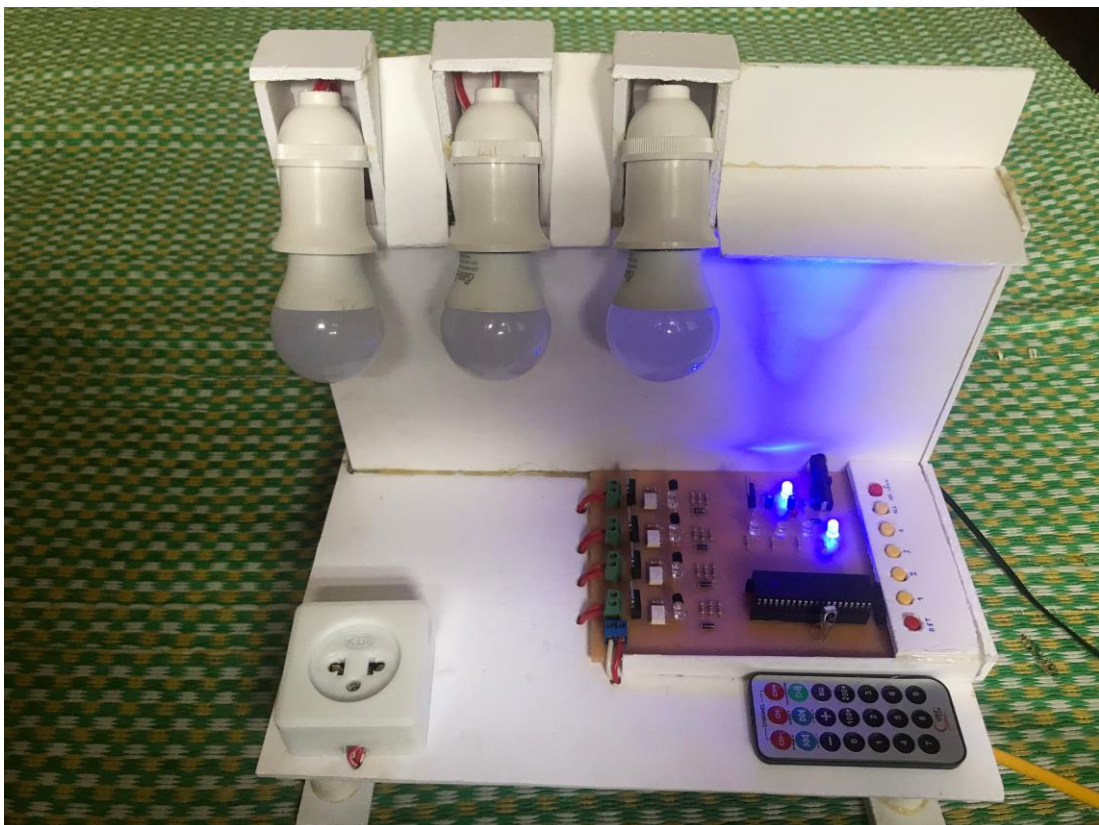
Hình 3. 18. Hình ảnh mô hình hệ thống

3.8. Kết quả thực nghiệm

- Hình ảnh thực tế khi mạch hoạt động:



Hình 3. 19. Hình ảnh bật thiết bị 1, 2, 3



Hình 3. 20. Hình ảnh đang học lệnh bật/tắt thiết bị 1

3.9. Kết luận và hướng phát triển

Sau một thời gian nghiên cứu và thực hiện đồ án, em đã đạt được những kết quả như sau:

- Thiết kế thành công mạch điều khiển từ xa, có thể tắt mở thiết bị theo ý muốn.
- Mạch thiết kế đúng chuẩn yêu cầu kỹ thuật đã đưa ra.
- Mạch thiết kế có độ chính xác điều khiển nhanh với thời gian tính trên đơn vị giây, tính ổn định cao.
- Sử dụng bộ nhớ ngoài EEPROM để lưu trữ các lệnh điều khiển thiết bị trước khi mất điện, khi có điện trở lại sẽ đọc bộ nhớ đó để điều khiển ngõ ra.
- Mạch đáp ứng được các yêu cầu của người dùng, sử dụng thuận tiện và hữu ích.

Tuy nhiên, trong quá trình thực hiện đồ án, do thời gian và kiến thức còn hạn hẹp nên còn những mặt hạn chế như:

- Tín hiệu chỉ truyền được theo đường thẳng của ánh sáng, nên loại điều khiển này có tầm hoạt động trong phạm vi khoảng 10 mét và cũng khá hạn chế khi không thể truyền qua các bức tường hay việc phải đi qua các góc khuất. Nó chỉ hoạt động tốt nhất khi ta chỉ thẳng vào led thu của mạch.
- Do ảnh hưởng của nhiều nguồn nhiễu hồng ngoại xung quanh phát ra như ánh sáng mặt trời, đèn huỳnh quang và bức xạ của con người nên gây ra ảnh hưởng và hạn chế tầm phát tín hiệu. Do đó chỉ dùng trong phòng, kho, hoặc những nơi có nhiệt độ môi trường thấp.

Đối với những sản phẩm trên thị trường:

Thiết kế chắc chắn, mạch hoạt động ổn định với tuổi thọ cao khoảng từ 1 đến 1,5 năm.

Tuy nhiên do lần đầu làm mạch nên chi phí linh kiện còn đắt hơn các sản phẩm trên thị trường, chưa có phần vỏ sản phẩm nên nhìn khá thô sơ.

Phương pháp thiết kế còn thủ công.

Ứng dụng:

Hiện nay ta sử dụng thiết bị điều khiển bằng hồng ngoại cho hầu hết các vật dụng trong nhà như ti vi, máy stereo, điều hòa nhiệt độ,...

Hướng phát triển đồ án:

- Mạch có thể kết hợp với bộ định thời gian để hẹn giờ cho các thiết bị tắt mở tự động khi cần thiết.
- Mở rộng mạch điều khiển thêm nhiều thiết bị.
- Nghiên cứu thêm để có thể ứng dụng mạch điều khiển từ xa bằng hồng ngoại vào nhiều mô hình khác lớn hơn, ưu việt hơn như điều khiển từ xa đóng mở cửa tự động, điều khiển TV, cassette...
- Mở rộng tầm thu, mở rộng khoảng cách bằng cách tăng số lượng led phát hoặc phân cực cho các led chạy mạnh hơn để có thể điều khiển nhay các hệ thống.
- Kết hợp với wifi hoặc bluetooth để điều khiển các thiết bị thuận lợi hơn.

TÀI LIỆU THAM KHẢO

1. Trang web tham khảo:

- [1].<http://arduino.vn/bai-viet/1073-lich-su-phat-trien-cua-vi-dieu-khien-va-vi-xu-li>
- [2].<https://ultraviewer.net/vi/72-tim-hieu-ve-lich-su-phat-trien-cua-cai-dieu-khien-tu-xa.html>
- [3].<https://dienmaygiare.net/dieu-khien-tu-xa-la-gi/>
- [4].https://vi.wikipedia.org/wiki/Tia_h%E1%BB%93ng_ngo%E1%BA%A1i
- [5].<https://banlinhkien.com/>
- [6].https://exploreembedded.com/wiki/NEC_IR_Remote_Control_Interface_with_8051
- [7].https://vi.wikipedia.org/wiki/Tia_h%E1%BB%93ng_ngo%E1%BA%A1i
- [8].<https://www.alldatasheet.com/>
- [9].<https://chotroi.vn/led-thu-hong-ngoai-ir1838-vo-sat>
- [10].<http://www.suativisaubaohanh.com/2020/05/sua-loi-khien-tivi-sony-khong-nhan-giong-noi.html>

2. Giáo trình tham khảo:

- [11]. Học lập trình PIC18F4520:
Tài liệu tra cứu PIC.
- [12]. Kiều Xuân Thực, Vũ Thị Thu Hương, Vũ Trung Kiên; *Vi điều khiển cấu trúc lập trình và ứng dụng*, Nhà xuất bản Giáo dục Việt Nam – 2008.

PHỤ LỤC

```

#define led_on 0
#define led_off 1
#define add_eep_stt_triac1      1
#define add_eep_stt_triac2      2
#define add_eep_stt_triac3      3
#define add_eep_stt_triac4      4
#define add_eep_high_remote_code_triac1    11
#define add_eep_low_remote_code_triac1     12
#define add_eep_high_remote_code_triac2    13
#define add_eep_low_remote_code_triac2     14
#define add_eep_high_remote_code_triac3    15
#define add_eep_low_remote_code_triac3     16
#define add_eep_high_remote_code_triac4    17
#define add_eep_low_remote_code_triac4     18
#define add_eep_high_remote_code_all_triac  19
#define add_eep_low_remote_code_all_triac   20
#define btn1 PORTA.B0// ON/OFF triac1
#define btn2 PORTA.B1// ON/OFF triac2
#define btn3 PORTA.B2// ON/OFF triac3
#define btn4 PORTA.B3// ON/OFF triac4
#define btn5 PORTA.B4// ON/OFF all triac
#define btn6 PORTA.B5// learn code remote

#define led1 PORTC.B2//show status triac1
#define led2 PORTC.B3//show status triac2
#define led3 PORTD.B0//show status triac3

```

```

#define led4 PORTD.B1//show status triac4

#define triac1 PORTC.B6 //control triac 1
#define triac2 PORTC.B5 //control triac 2
#define triac3 PORTC.B4 //control triac 3
#define triac4 PORTD.B2 //control triac 4

void init_btn_remote();
char read_remote();
void load_eeprom();
void check_stt_triac();
unsigned char receiver_data_arr[20],dem_arr,data_ok,data_temp;
void clear_receiver_data_arr();

//-----cac bien phuc vu tinh ma remote-----
-----

#define _1838B PORTB.B0 //dinh nghĩa chân kết nối với vs1838b
#define size_arr 20 //dinh nghĩa kích thước mảng lưu giá trị để phục vụ
việc tính ma remote
unsigned long ma_remote=0;//lưu giá trị mã remote đọc được
unsigned int count_logic0 = 0, count_logic1 = 0;
unsigned char arr_value[size_arr];//mảng lưu giá trị đọc được
char enable_write_arr = 0; //cho phép hoặc không cho phép lưu data vào
mảng
//-----
-----

char txt[15];
typedef enum stt_var {off,on}stt_var;

```

```

stt_var stt_triac1,stt_triac2,stt_triac3,stt_triac4, stt_all_triac;
unsigned int code_remote_triac1=0;
unsigned int code_remote_triac2=0;
unsigned int code_remote_triac3=0;
unsigned int code_remote_triac4=0;
unsigned int code_remote_all_triac=0;

unsigned int t=0;
//ham xu li bat su kien data truyen toi tu tay phat hong ngoai(remote)
void interrupt(void)
{
    //ngat hong ngoai
    if(INT0IF_bit)
    {
        INT0IF_bit=0;
        read_remote();
    }
}
//void interrupt(void)

void main()
{
    INT0IE_bit=1;//cho phep ngat int0
    INT0IF_bit=0;//reset co bao ngat int0
    INTEDG0_bit=0;//ngat canh xuong
    GIE_bit = 1; //cho phep ngat toan cuc
    TRISB.B0 = 1;//_1838B
    TRISA=0x0f;//port A is input
    ADCON1=0x0f; //A0,A1,A2 is mode analog

```

```

TRISA.B0=1;// ON/OFF triac1
TRISA.B1=1;// ON/OFF triac2
TRISA.B2=1;// ON/OFF triac3
TRISA.B3=1;// ON/OFF triac4
TRISA.B4=1;// ON/OFF all triac
TRISA.B5=1;// learn code remote

```

```

TRISC.B2=0;//show status triac1
TRISC.B3=0;//show status triac2
TRISD.B0=0;//show status triac3
TRISD.B1=0;//show status triac4

```

```

TRISC.B6=0; //control triac 1
TRISC.B5=0; //control triac 2
TRISC.B4=0; //control triac 3
TRISD.B2=0; //control triac 4

```

```

triac1=triac2=triac3=triac4=off;

```

```

dem_arr=data_ok=0;
ma_remote=1;
for(t=0;t<4;t++)
{
    led1=led2=led3=led4=led_on;
    delay_ms(200);
    led1=led2=led3=led4=led_off;
    delay_ms(200);
}

```

```

load_eeprom();
//check_stt_triac();

//=====
=====

while(1)
{
    if(!btn1 || ma_remote==code_remote_triac1)
    {
        delay_ms(5);
        if(!btn1 || ma_remote==code_remote_triac1)
        {
            stt_triac1=(stt_triac1)?off:on;
            triac1=(stt_triac1)?led_on:led_off;
            led1=(stt_triac1)?on:off;
            EEPROM_Write(add_eep_stt_triac1,stt_triac1);
delay_ms(5);
        }
        while(!btn1);

                                ma_remote=0;
    }

    if(!btn2 || ma_remote==code_remote_triac2)
    {
        delay_ms(5);
        if(!btn2 || ma_remote==code_remote_triac2)
        {

```

```

        stt_triac2=(stt_triac2)?off:on;
        triac2=(stt_triac2)?on:off;
        led2=(stt_triac2)?led_on:led_off;
        EEPROM_Write(add_eep_stt_triac2,stt_triac2);
delay_ms(5);
    }
    while(!btn2);

                                ma_remote=0;
}

if(!btn3 || ma_remote==code_remote_triac3)
{
    delay_ms(5);
    if(!btn3 || ma_remote==code_remote_triac3)
    {
        stt_triac3=(stt_triac3)?off:on;
        triac3=(stt_triac3)?on:off;
        led3=(stt_triac3)?led_on:led_off;
        EEPROM_Write(add_eep_stt_triac3,stt_triac3);
delay_ms(5);
    }
    while(!btn3);

                                ma_remote=0;
}

if(!btn4 || ma_remote==code_remote_triac4)
{
    delay_ms(5);
    if(!btn4 || ma_remote==code_remote_triac4)

```



```

    {
        stt_triac4=(stt_triac4)?off:on;
        triac4=(stt_triac4)?on:off;
        led4=(stt_triac4)?led_on:led_off;
        EEPROM_Write(add_eep_stt_triac4,stt_triac4);
delay_ms(5);
    }
    while(!btn4);

                                ma_remote=0;
}
if(!btn5 || ma_remote==code_remote_all_triac)
{
    delay_ms(5);
    if(!btn5 || ma_remote==code_remote_all_triac)
    {
        stt_all_triac=(stt_all_triac)?off:on;
        if(stt_all_triac==on)
        {
            stt_triac1=on;
            stt_triac2=on;
            stt_triac3=on;
            stt_triac4=on;
        }
        else
        {
            stt_triac1=off;
            stt_triac2=off;
            stt_triac3=off;
            stt_triac4=off;

```

```

    }
    triac1=(stt_triac1)?on:off;
    triac2=(stt_triac2)?on:off;
    triac3=(stt_triac3)?on:off;
    triac4=(stt_triac4)?on:off;
    led1=(stt_triac1==on)?led_on:led_off;
    led2=(stt_triac2==on)?led_on:led_off;
    led3=(stt_triac3==on)?led_on:led_off;
    led4=(stt_triac4==on)?led_on:led_off;
    EEPROM_Write(add_eep_stt_triac1,stt_triac1);
delay_ms(5);

    EEPROM_Write(add_eep_stt_triac2,stt_triac2);
delay_ms(5);

    EEPROM_Write(add_eep_stt_triac3,stt_triac3);
delay_ms(5);

    EEPROM_Write(add_eep_stt_triac4,stt_triac4);
delay_ms(5);

    }//if(btn5)
    while(!btn5);

                                ma_remote=0;

    }

//*****che do hoc lenh *****
    if(!btn6) delay_ms(30);//nhan giu ca 2 nut UP va d?m de vao che do
hoc lenh

    if(!btn6)//nhan giu ca 2 nut UP va d?m de vao che do hoc lenh
    {
        ma_remote=0;
        while(!btn6);

```

```

//learn code remote triac 1
    led1=led_on;
    led2=led_off;
    led3=led_off;
    led4=led_off;
while(1)
{
    if(ma_remote!=0)
    {
        code_remote_triac1=ma_remote;

EEPROM_Write(add_eep_high_remote_code_triac1,code_remote_triac1/256
); delay_ms(5);

EEPROM_Write(add_eep_low_remote_code_triac1,code_remote_triac1%25
6); delay_ms(5);

        ma_remote=0;
        break;
    }

        led1=led_on;
    delay_ms(50);
        led1=led_off;
    delay_ms(50);
        if(!btn6) break;
    }

    while(!btn6);
//learn code remote triac 2
    led1=led_off;
    led2=led_on;

```

```

        led3=led_off;
        led4=led_off;
while(2)
{
    if(ma_remote!=0)
    {
        code_remote_triac2=ma_remote;

EEPROM_Write(add_eep_high_remote_code_triac2,code_remote_triac2/256
); delay_ms(5);

EEPROM_Write(add_eep_low_remote_code_triac2,code_remote_triac2%25
6); delay_ms(5);

        ma_remote=0;
        break;
    }

        led2=led_on;
        delay_ms(50);
        led2=led_off;
        delay_ms(50);
        if(!btn6) break;
    }
    while(!btn6);
//learn code remote triac 3
    led1=led_off;
    led2=led_off;
    led3=led_on;
    led4=led_off;
while(3)

```

```

{
    if(ma_remote!=0)
    {
        code_remote_triac3=ma_remote;

EEPROM_Write(add_eep_high_remote_code_triac3,code_remote_triac3/256
); delay_ms(5);

EEPROM_Write(add_eep_low_remote_code_triac3,code_remote_triac3%25
6); delay_ms(5);

        ma_remote=0;
        break;
    }

        led3=led_on;
    delay_ms(50);
        led3=led_off;
    delay_ms(50);
    if(!btn6) break;
}

    while(!btn6);

//learn code remote triac 4
    led1=led_off;
    led2=led_off;
    led3=led_off;
    led4=led_on;
while(4)
{
    if(ma_remote!=0)

```

```

    {
        code_remote_triac4=ma_remote;

EEPROM_Write(add_eep_high_remote_code_triac4,code_remote_triac4/256
); delay_ms(5);

EEPROM_Write(add_eep_low_remote_code_triac4,code_remote_triac4%25
6); delay_ms(5);
        ma_remote=0;
        break;
    }

        led4=led_on;
delay_ms(50);
        led4=led_off;
delay_ms(50);
        if(!btn6) break;
    }

    while(!btn6);
//learn code remote all triac
    led1=led_on;
    led2=led_on;
    led3=led_on;
    led4=led_on;
while(5)
{
    if(ma_remote!=0)
    {
        code_remote_all_triac=ma_remote;

```

```
EEPROM_Write(add_eep_high_remote_code_all_triac,code_remote_all_tria
c/256); delay_ms(5);
```

```
EEPROM_Write(add_eep_low_remote_code_all_triac,code_remote_all_triac
%256); delay_ms(5);
```

```
    ma_remote=0;
```

```
    break;
```

```
}
```

```
    delay_ms(50);
```

```
    led1=led_on;
```

```
    led2=led_on;
```

```
    led3=led_on;
```

```
    led4=led_on;
```

```
    delay_ms(50);
```

```
    led1=led_off;
```

```
    led2=led_off;
```

```
    led3=led_off;
```

```
    led4=led_off;
```

```
if(!btn6) break;
```

```
}
```

```
while(!btn6);
```

```
for(t=0;t<4;t++)
```

```
{
```

```
    led1=led2=led3=led4=led_on;
```

```
    delay_ms(200);
```

```
    led1=led2=led3=led4=led_off;
```

```

        delay_ms(200);
    }

    led1=(stt_triac1==on)?led_on:led_off;
    led2=(stt_triac2==on)?led_on:led_off;
    led3=(stt_triac3==on)?led_on:led_off;
    led4=(stt_triac4==on)?led_on:led_off;
} //if(!btn6)
//*****

    delay_ms(5);
} //while(1)

//=====
=====

} //void main
//ham doc ma remote
//xu li tin hieu hong ngoai va giai ma
char read_remote()
{
    int i;
    count_logic0=count_logic1=0;
    if(enable_write_arr==0)
    {
        //kiem tra muc logic0
        while(!_1838B)
        {
            delay_us(100);
            if(count_logic0<70) count_logic0++; else break;
        }
        while(!_1838B); //doi het muc logic0 10ms
    }
}

```



```

//kiem tra muc logic1 3.3ms
while(!_1838B)
{
    delay_us(100);
    if(count_logic1<30) count_logic1++; else break;
}
if(count_logic0>=70 && count_logic1>=30) //nhan ra xung start
{
    enable_write_arr=1;//cho phep ghi gia tri vao mang
    while(!_1838B);//doi het xung logic1 3.3ms
}
} //if(enable_write_arr==0)
else
{
    for(i=0;i<size_arr;i++)//lay gia tri cua 20 bit
    {
        count_logic0=0;
        while(!_1838B)
        {
            delay_us(100);
            count_logic0++;
        }
        arr_value[i]=count_logic0;//gan gia tri do duoc vao mang
        count_logic1=0;
        while(!_1838B)
        {
            delay_us(10);
            if(count_logic1<500) count_logic1++; else break;
        }
    }
}

```

```

        }//for(int i=0;i<20;i++)
    ma_remote=0;
    for(i=0;i<size_arr;i++)
    {
        if(arr_value[i]>22) ma_remote|=0;
        else ma_remote|=1;
        ma_remote<<=1;
    }
    ma_remote=ma_remote&0xff00;//lay 16bit
    ma_remote=ma_remote>>8;
    enable_write_arr=0;
    for(i=0;i<size_arr;i++) arr_value[i]=0;//xoa mang
} //else
return 0;
} //char read_remote()
/*
void clear_receiver_data_arr()
{
    int i=0;
    for(i=0;i<20;i++)
        receiver_data_arr[i]=0;
}
*/
//doc trang thai cac triac duoc luu trong eeprom
void load_eeprom()
{
    unsigned char eep_temp=0;
    unsigned int low_byte=0;
    unsigned int high_byte=0;

```

```

//load status triac
//triac1
eep_temp=EEPROM_Read(add_eep_stt_triac1); delay_ms(5);
if(eep_temp>1) stt_triac1=off; else stt_triac1=eep_temp;
//triac2
eep_temp=EEPROM_Read(add_eep_stt_triac2); delay_ms(5);
if(eep_temp>1) stt_triac2=off; else stt_triac2=eep_temp;
//triac3
eep_temp=EEPROM_Read(add_eep_stt_triac3); delay_ms(5);
if(eep_temp>1) stt_triac3=off; else stt_triac3=eep_temp;
//triac4
eep_temp=EEPROM_Read(add_eep_stt_triac4); delay_ms(5);
if(eep_temp>1) stt_triac4=off; else stt_triac4=eep_temp;

//load remote code triac
//triac1
low_byte=EEPROM_Read(add_eep_low_remote_code_triac1);
delay_ms(5);
high_byte=EEPROM_Read(add_eep_high_remote_code_triac1);
delay_ms(5);
code_remote_triac1=high_byte*256+low_byte;

//triac2
low_byte=EEPROM_Read(add_eep_low_remote_code_triac2);
delay_ms(5);
high_byte=EEPROM_Read(add_eep_high_remote_code_triac2);
delay_ms(5);
code_remote_triac2=high_byte*256+low_byte;

```

```

        //triac3
        low_byte=EEPROM_Read(add_eep_low_remote_code_triac3);
delay_ms(5);
        high_byte=EEPROM_Read(add_eep_high_remote_code_triac3);
delay_ms(5);
        code_remote_triac3=high_byte*256+low_byte;

        //triac4
        low_byte=EEPROM_Read(add_eep_low_remote_code_triac4);
delay_ms(5);
        high_byte=EEPROM_Read(add_eep_high_remote_code_triac4);
delay_ms(5);
        code_remote_triac4=high_byte*256+low_byte;

        //all triac
        low_byte=EEPROM_Read(add_eep_low_remote_code_all_triac);
delay_ms(5);
        high_byte=EEPROM_Read(add_eep_high_remote_code_all_triac);
delay_ms(5);
        code_remote_all_triac=high_byte*256+low_byte;
    }

    void check_stt_triac()
    {
        if(stt_triac1==on) { triac1=on; led1=led_on; } else { triac1=off;
led1=led_off; }
        if(stt_triac2==on) { triac2=on; led2=led_on; } else { triac2=off;
led2=led_off; }

```

```
        if(stt_triac3==on) { triac3=on; led3=led_on; } else { triac3=off;
led3=led_off; }
        if(stt_triac4==on) { triac4=on; led4=led_on; } else { triac4=off;
led4=led_off; }
    }
```