

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY

UNIVERSITY OF INFORMATION TECHNOLOGY

FACULTY OF COMPUTER ENGINEERING

STUDENT FULL NAME

Phạm Trí Đức

Lương Văn Đại

CAPSTONE PROJECT

CONTROL FAN ON/OFF BASED ON TEMPERATURE USING 8051

MIDTERM CAPSTONE PROJECT REPORT

HO CHI MINH CITY, May 2023

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY
FACULTY OF COMPUTER ENGINEERING

FULLNAME - ID

Phạm Trí Đức

21521966

Lương Văn Đại

21521913

CAPSTONE PROJECT
CONTROL FAN ON/OFF BASED ON TEMPERATURE USING 8051
MIDTERM CAPSTONE PROJECT REPORT

MENTOR

PhD. TRI NHUT DO

HO CHI MINH CITY, May 2023

VIETNAM NATIONAL UNIVERSITY

SOCIALIST REPUBLIC OF VIETNAM

HO CHI MINH CITY

Independence – Freedom - Happiness

UNIVERSITY OF INFORMATION

TECHNOLOGY

DETAILED TOPICS

VIETNAMESE PROJECT NAME: Điều khiển quạt bật tắt dựa trên nhiệt độ sử dụng 8051
ENGLISH PROJECT NAME: CONTROL FAN ON/OFF BASED ON TEMPERATURE USING 8051
Instructor PhD. Tri Nhut Do, Faculty of Computer Engineering
Implementation time: From: 15/04/2023 To: 11/05/2023
Student Perform: Phạm Trí Đức – 21521966, Lương Văn Đại - 21521913
Overview of the topic: Quạt có thêm chức năng bật tắt tự động tùy thuộc vào nhiệt độ môi trường. Thông thường khi nhiệt độ cao, quạt sẽ bật và nhiệt độ thấp thì tắt. Điều này được thực hiện thủ công bằng cách sử dụng con người. Trong đồ án này đề xuất giải pháp điều khiển tự động để bật tắt quạt.

The goal of the subject: Thiết kế và phát triển một hệ thống quạt thông minh bật tắt tự động dựa trên vi điều khiển. từ đó phát triển và đưa vào thực tế

Methods of implementation: Phương pháp kế thừa: kế thừa từ các tài liệu, công trình nghiên cứu đó về mảng: Cấu trúc vi điều khiển 8051. Phương pháp kiểm chứng: sau khi đã xây dựng xong cơ sở lý thuyết của đề tài tiến hành thử nghiệm, mô phỏng hoạt động và đưa ra kết luận

Main contents of the topic: tự động hóa điều khiển quạt dựa trên vi điều khiển 8051.

Certification of Instructor

(Sign and clearly state full name)

HCM city, 2023 Month DayDay

Student

(Sign and clearly state full name)

TABLE OF CONTENTS

Chapter 1. INTRODUCTION	1
1.1 Đặt vấn đề	1
1.2 Mục tiêu	1
1.3 Nội dung đề tài	1
Chapter 2. THEORY BASIS.....	6
2.1 Vi điều khiển AT89C51	2
2.2 Cảm biến nhiệt độ sử dụng độ C LM35	8
2.3 ADC0804	11
2.4 Khối điều khiển động cơ L293D	14
Chapter 3. SYSTEM IMPLEMENTATION PROCESS	17
3.1 Sơ đồ khối	17
3.2 Lưu đồ giải thuật	18
3.3 Thiết kế mạch và nguyên lý hoạt động	20
3.4 Code và nguyên lý hoạt động	22
Chapter 4. TEST RESULTS AND EVALUATION:.....	27
4.1 Phương pháp thực nghiệm: Mô phỏng	27
4.2 Kết quả mô phỏng trên Proteus	28
Chapter 5. SUMMARY, RESTRICTIONS AND DEVELOPMENT ORIENTATIONS:	29
5.1 Tính ứng dụng	29
5.2 Kết quả và phát triển	30
REFERENCE:.....	31

LIST OF FIGURES

Hình 2.1: Sơ đồ khối vi điều khiển 8051	3
Hình 2.2 Sơ Đồ chân vi điều khiển AT89C51	4
Hình 2.3 Sơ đồ kết nối chân RST	6
Hình 2.4 Sơ đồ kết nối chân XTAL1, XTAL2	7
Hình 2.5 Hình ảnh thực tế các chân LM35	9
Hình 2.6 Sơ đồ nối chân LM35	10
Hình 2.7 Sơ đồ chân ADC0804.....	12
Hình 2.8 sơ đồ nối chân ADC 0804	13
Hình 2.9 Sơ đồ chân L293D	15
Hình 3.1 sơ đồ khối của mạch	18
Hình 3.2 Lưu đồ giải thuật của mạch	19
Hình 3.3 Thiết kế mạch Proteus	20

LIST OF TABLES

Bảng 2.1 Bảng chức năng Port 3	5
Bảng 2.2 Bảng chức năng chân của LM35	9
Bảng 2.3 Chức năng của chân L293D	15
Bảng 4.1 Kết quả mô phỏng	28

Chapter 1. INTRODUCTION

1.1 Đặt vấn đề

Ngày nay, kỹ thuật máy tính đã liên tục có những tiến bộ vượt bậc, đặc biệt là trong kỹ thuật thiết kế vi mạch và thiết kế các hệ thống thông minh, hệ thống nhúng. Sự phát triển này là bước ngoặt quan trọng góp phần hỗ trợ trong việc chế tạo sản phẩm máy móc có tính năng tự động và thông minh hơn.

Đời sống vật chất phát triển tạo ra nhiều tiện nghi cho cuộc sống của con người. Đối với một quốc gia có nhiều biến động khí hậu như Việt Nam, nhu cầu sử dụng quạt điện, thiết bị điều hòa chiếm phần lớn trong đời sống sinh hoạt. Tuy nhiên, hầu hết các thiết bị này vẫn còn hạn chế về công năng tắt mở. Ví dụ như khi đi ngủ, người ta có thói quen mở máy quạt và điều hòa để làm mát nhưng khi họ vào giấc ngủ sâu thì không khí trở nên lạnh hơn, họ phải thức dậy ngắt quãng để tắt thiết bị. Điều này gây ảnh hưởng đến chất lượng giấc ngủ và sức khỏe của con người. Đứng trước vấn đề đó, việc nghiên cứu sử dụng thiết bị điều hòa không khí có thể tự động tắt/mở phụ thuộc vào nhiệt độ là vô cùng cần thiết.

Nhằm cải thiện cuộc sống thuận tiện, dễ dàng hơn, chúng em bắt tay vào thực hiện đề tài “Thiết kế hệ thống quạt bật tắt dựa trên nhiệt độ”

1.2 Mục tiêu

Thiết kế và phát triển một hệ thống quạt thông minh bật tắt tự động dựa trên cảm biến nhiệt độ, giúp con người chủ động trong việc điều hòa nhiệt độ không gian và bên cạnh đó là tiết kiệm điện.

Nghiên cứu và phát triển điều khiển cơ chế bật tắt hoạt động của quạt một cách tự động, sử dụng các cảm biến nhiệt độ LM35, bộ chuyển đổi analog sang digital 8 bit ADC0804, bộ điều khiển động cơ L293D và vi điều khiển AT89C51 thuộc họ 8051.

Tối ưu hóa tính ổn định và độ chính xác của hệ thống thông qua nghiên cứu và phát triển các kỹ thuật điều khiển và giao tiếp giữa các thành phần trong hệ thống.

Đánh giá hiệu quả và tính khả thi của hệ thống thông qua các thử nghiệm.

1.3 Nội dung đề tài

- Tìm hiểu, thiết kế mạch và mô phỏng kết quả trên phần mềm
- Lập trình ngôn ngữ C trên vi điều khiển AT89C51 thuộc họ 8051
- Chuyển đổi giá trị Analog sang digital để có thể đọc giá trị và xuất lên màn hình LCD

Phần mềm sử dụng:

- Thiết kế và mô phỏng mạch : Proteus. - IDE code bằng ngôn ngữ C : Keil C

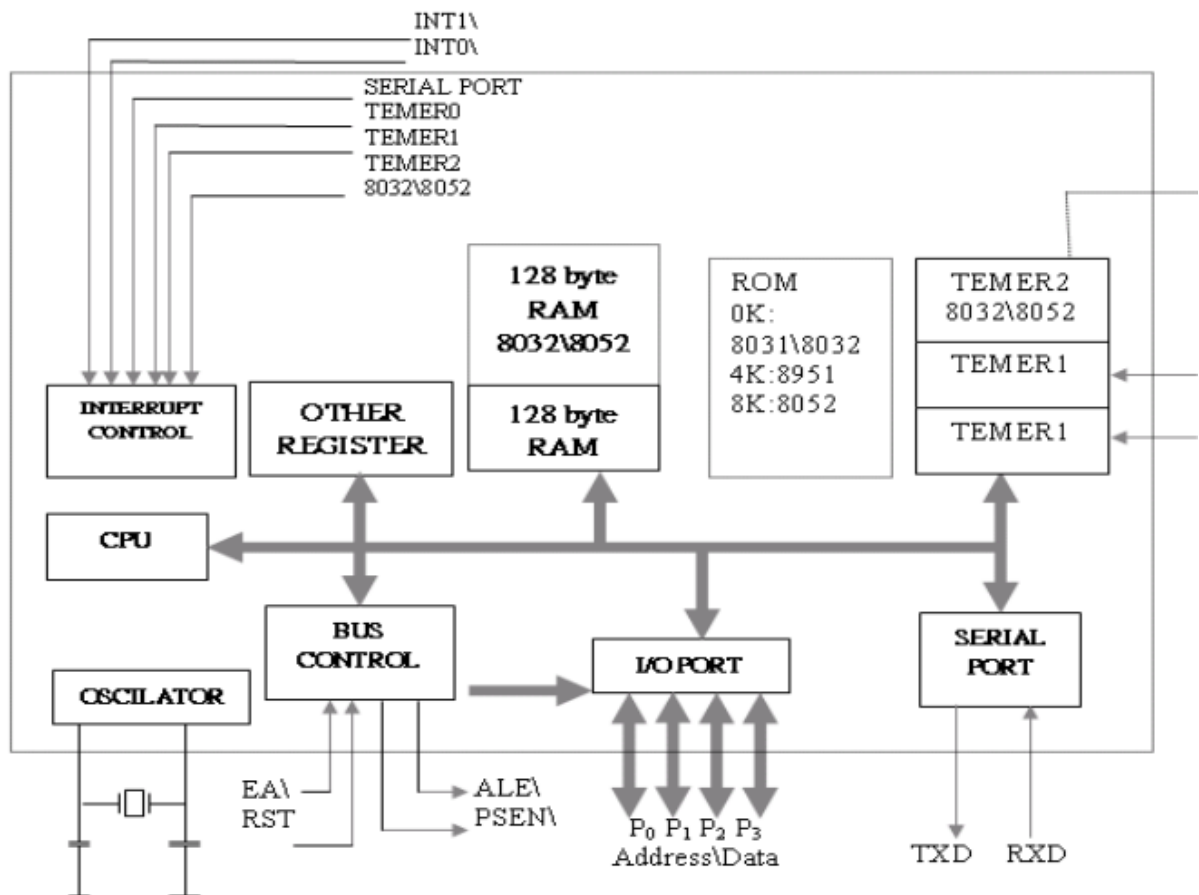
Chapter 2. THEORY BASIS

2.1 Vi điều khiển AT89C51

Đặc điểm và chức năng hoạt động của các IC họ MSC-51 hoàn toàn tương tự như nhau. Ở đây giới thiệu IC8951 là một họ IC vi điều khiển do hãng Intel của Mỹ sản xuất. Chúng có các đặc điểm chung như sau:

- + 4 KB bộ nhớ có thể lập trình lại nhanh, có khả năng tới 1000 chu kỳ ghi xóa
- + Tần số hoạt động từ: 0Hz đến 24 MHz
- + 3 mức khóa bộ nhớ lập trình
- + 2 bộ Timer/counter 16 Bit
- + 128 Byte RAM nội.
- + 4 Port xuất /nhập I/O 8 bit.
- + Giao tiếp nối tiếp.
- + 64 KB vùng nhớ mã ngoài
- + 64 KB vùng nhớ dữ liệu ngoài.
- + Xử lý Boolean (hoạt động trên bit đơn).
- + 210 vị trí nhớ có thể định vị bit.
- + 4 chu kỳ máy (4 μ s đối với thạch anh 12MHz) cho hoạt động nhân hoặc chia.

Sơ đồ khối chức năng



Hình 2.1: Sơ đồ khối vi điều khiển 8051

Cấu trúc kết nối phần cứng của các bộ vi điều khiển trong họ gần tương tự như nhau, một số khác biệt giữa chúng cũng được biểu diễn trên sơ đồ khối hình 2.1

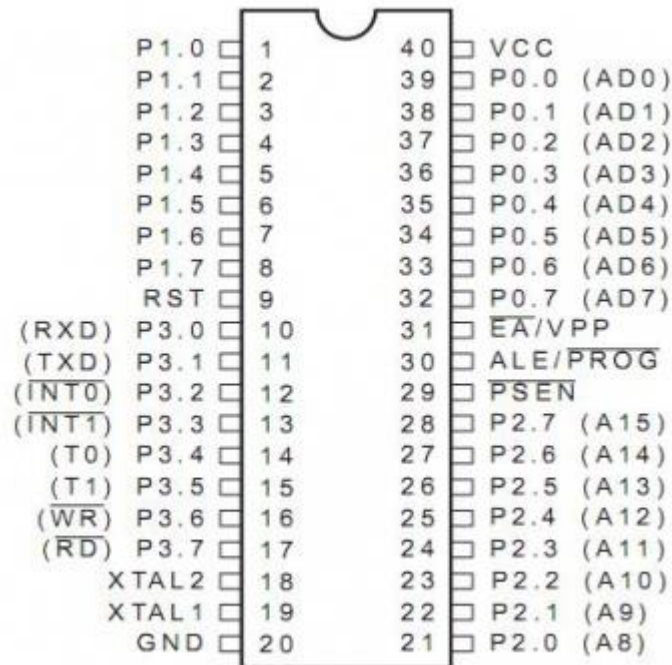
Khối xử lý trung tâm (CPU) nhận tín hiệu xung nhịp từ bộ dao động, tần số ra của bộ tạo dao động sẽ tùy thuộc vào tần số thạch anh bên ngoài. Hầu hết các bộ vi điều khiển trong họ đều có ít nhất 128 byte RAM bên trong. Các thanh ghi thông thường nằm trong phần RAM. Ngoài 8031/32 các vi điều khiển còn lại đều có bộ nhớ ROM lưu trữ chương trình điều khiển. Bộ nhớ ROM này có thể là Mask-ROM chỉ lập trình được bởi nhà sản xuất, có thể là EPROM hoặc EEPROM có thể lập trình lại nhiều lần bởi người sử dụng.

Các bộ định thời lập trình được có thể đếm theo xung cung cấp từ bên ngoài hoặc xung chuẩn từ bộ tạo dao động, có bộ đếm này có ứng dụng rất phổ biến trong điều khiển tự động.

Bộ điều khiển Bus cung cấp các tín hiệu điều khiển giao tiếp với bên ngoài, và kiểm soát hoạt động của các cổng vào ra dữ liệu song song. Hai trong bốn cổng vào ra song song (P0 và P2) có thể sử dụng làm các Bus địa chỉ và dữ liệu trong chế độ giao tiếp bộ nhớ ngoài. Cổng vào ra nối tiếp có hai đường truyền và nhận dữ liệu nối tiếp với các thiết bị khác.

Bộ điều khiển ngắt tích hợp trong chip cho phép nhận hai yêu cầu ngắt cung cấp thẳng từ bên ngoài, hoặc từ cổng nối tiếp và các bộ định thời bên trong.

Sơ đồ chân vi điều khiển AT89C51



Hình 2.2 Sơ Đồ chân vi điều khiển AT89C51

Port 0(P0.0-P0.7): Các chân từ 32-39

- Chức năng xuất/nhập :các chân này được dùng để nhận tín hiệu từ bên ngoài vào để xử lý, hoặc dùng để xuất tín hiệu ra bên ngoài, chẳng hạn xuất tín hiệu để điều khiển led đơn sáng tắt.

-Chức năng là bus dữ liệu và bus địa chỉ (AD7-AD0) : 8 chân này (hoặc Port 0) còn làm nhiệm vụ lấy dữ liệu từ ROM hoặc RAM ngoại (nếu có kết nối với bộ nhớ ngoại), đồng thời Port 0 còn được dùng để định địa chỉ của bộ nhớ ngoại.

Port 1 (P1.0 – P1.7): Từ chân số 1 – 8.

-Port 1 có chức năng :Port xuất nhập dữ liệu (P1.0 – P1.7) . Port1 không có chức năng khác, vì vậy chúng chỉ được dùng cho giao tiếp với các thiết bị ngoại

Port 2 (P2.0-P2.7): Các chân từ 21-28

-Chức năng xuất/nhập

-Chức năng là bus địa chỉ cao (A8-A15): khi kết nối với bộ nhớ ngoại có dung lượng lớn,cần 2 byte để định địa chỉ của bộ nhớ, byte thấp do P0 đảm nhận, byte cao do P2 này đảm nhận.

Port 3 (P3.0-P3.7): Port 3 gồm 8 chân (từ chân 10 đến 17):

-Chức năng xuất/nhập

Với mỗi chân có một chức năng riêng thứ hai như trong bảng sau

Bit	Tên	Chức năng chuyển đổi
P3.0	RxD	Nhận dữ liệu trong tuyến dữ liệu nối tiếp
P3.1	TxD	Xuất dữ liệu trong truyền dữ liệu nối tiếp
P3.2	INT0	Ngõ vào ngắt ngoại 0
P3.3	INT1	Ngõ vào ngắt ngoại 1
P3.4	T0	Ngõ vào nhận xung đếm của timer/counter thứ 0
P3.5	T1	Ngõ vào nhận xung đếm của timer/counter thứ 1
P3.6	WR	Tín hiệu điều khiển ghi dữ liệu lên bộ nhớ ngoại

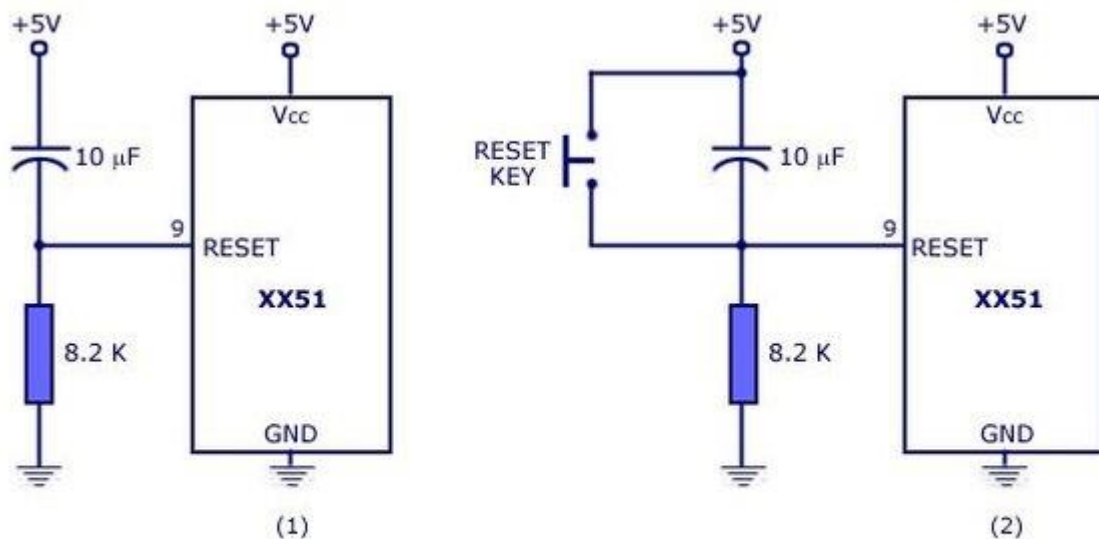
Bit	Tên	Chức năng chuyển đổi
P3.7	RD	Tín hiệu điều khiển đọc dữ liệu lên bộ nhớ ngoài

Bảng 2.1 Bảng chức năng Port 3

Chân RST(Reset) : chân số 9

Chức năng:

Khi tín hiệu này được đưa lên mức cao (trong ít nhất 2 chu kỳ máy), các thanh ghi bên trong 8051 được nạp những giá trị thích hợp để khởi động hệ thống



(1) Power-on Reset Circuit and (2) With Manual Reset Option

Hình 2.3 Sơ đồ kết nối chân RST

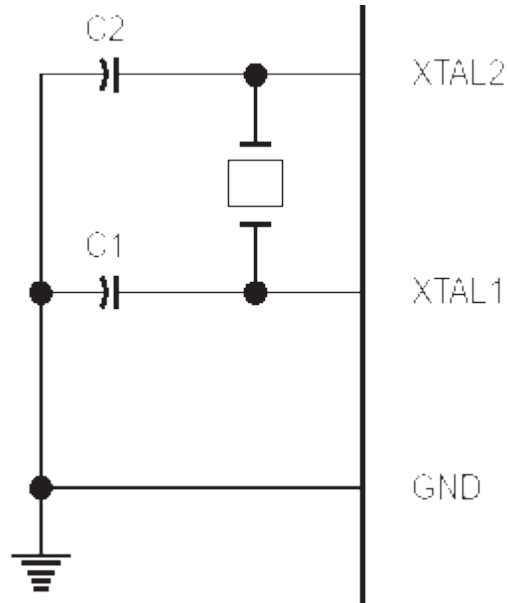
Chân XTAL1, XTAL2: Chân số 18-19

Chức năng:

Được nối với thạch anh hoặc mạch dao động cung cấp tín hiệu xung clock cho mạch

XTAL1 _ ngõ vào mạch tạo xung clock trong chip.

XTAL2 _ ngõ ra mạch xung clock trong chip.



Hình 2.4 Sơ đồ kết nối chân XTAL1, XTAL2

Chân cho phép bộ nhớ chương trình PSEN

-PSEN (program store enable) tín hiệu được xuất ra ở chân 29 dùng để truy xuất bộ nhớ chương trình ngoài. Chân này thường được nối với chân OE (output enable) của ROM ngoài.

-Khi vi điều khiển làm việc với bộ nhớ chương trình ngoài, chân này phát ra tín hiệu kích hoạt ở mức thấp và được kích hoạt 2 lần trong một chu kỳ máy. Khi thực thi một chương trình ở ROM nội, chân này được duy trì ở mức logic không tích cực (logic 1)(Không cần kết nối chân này khi không sử dụng đến)

Chân ALE (chân cho phép chốt địa chỉ-chân 30)

-Khi vi điều khiển truy xuất bộ nhớ từ bên ngoài, port 0 vừa có chức năng là bus địa chỉ, vừa có chức năng là bus dữ liệu do đó phải tách các đường dữ liệu và địa chỉ.

Tín hiệu ở chân ALE dùng làm tín hiệu điều khiển để giải đa hợp các đường địa chỉ và các đường dữ liệu khi kết nối chúng với IC chốt.

-Các xung tín hiệu ALE có tốc độ bằng 1/6 lần tần số dao động đưa vào Vi điều khiển, như vậy có thể dùng tín hiệu ở ngõ ra ALE làm xung clock cung cấp cho các phần khác của hệ thống. Ghi chú: khi không sử dụng có thể bỏ trống chân này

Chân EA

Chân EA dùng để xác định chương trình thực hiện được lấy từ ROM nội hay ROM ngoại.

Khi EA nối với logic 1(+5V) thì Vi điều khiển thực hiện chương trình lấy từ bộ nhớ nội

Khi EA nối với logic 0(0V) thì Vi điều khiển thực hiện chương trình lấy từ bộ nhớ ngoại

2.2 Cảm biến nhiệt độ sử dụng độ C LM35

LM35 là một cảm biến nhiệt độ tương tự, điện áp ở đầu ra của cảm biến tỷ lệ với nhiệt độ tức thời và có thể dễ dàng được xử lý để có được giá trị nhiệt độ bằng °C và có các thông số kỹ thuật sau:

- Hiệu chuẩn trực tiếp theo °C
- Điện áp hoạt động: 4-30VDC
- Dòng điện tiêu thụ: khoảng 60uA
- Nhiệt độ thay đổi tuyến tính: 10mV/°C
- Khoảng nhiệt độ đo được: -55°C đến 150°C
- Điện áp thay đổi tuyến tính theo nhiệt độ: 10mV/°C
- Độ tự gia nhiệt thấp, 0,08°C trong không khí tĩnh
- Sai số: 0,25°C

- Trở kháng ngõ ra nhỏ, $0,2\Omega$ với dòng tải 1mA
- Kiểu chân: TO92
- Kích thước: $4.3 \times 4.3\text{mm}$

Ưu điểm của LM35 so với cặp nhiệt điện là nó không yêu cầu bất kỳ hiệu chuẩn bên ngoài nào. Lớp vỏ cũng bảo vệ nó khỏi bị quá nhiệt. Chi phí thấp và độ chính xác cao đã khiến cho loại cảm biến này trở thành một lựa chọn đối với những người yêu thích chế tạo mạch điện tử, người làm mạch tự chế và các bạn sinh viên.

Sơ đồ chân của cảm biến nhiệt độ LM35



Hình 2.5 Hình ảnh thực tế các chân LM35

Số chân	Tên chân	Chức năng
1	V_{CC} hay $+V_S$	Chân cấp nguồn với điện áp từ 4V đến 30V
2	V_{OUT}	Chân lấy điện áp ra, điện áp ở chân này thay đổi $10\text{mV}/^\circ\text{C}$
3	GND	Chân nối đất

Bảng 2.2 Bảng chức năng chân của LM35

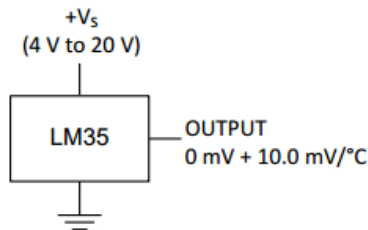
Nguyên lý hoạt động của cảm biến nhiệt độ LM35

Cảm biến LM35 hoạt động bằng cách cho ra một giá trị điện áp nhất định tại chân V_{OUT} (chân giữa) ứng với mỗi mức nhiệt độ. Như vậy, bằng cách đưa vào chân bên trái của cảm biến LM35 điện áp 5V, chân phải nối đất, đo hiệu điện thế ở chân giữa, bạn sẽ có được nhiệt độ (0-100°C) tương ứng với điện áp đo được.

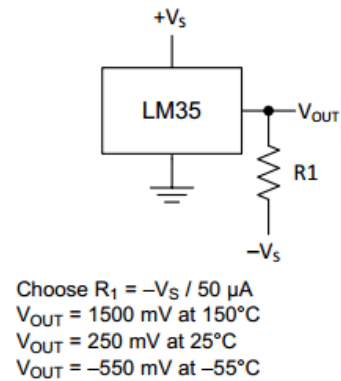
Mạch đo nhiệt độ

LM35 có thể được sử dụng một trong hai cấu hình mạch như hình bên dưới. Cả hai đều mang lại kết quả khác nhau.

**Basic Centigrade Temperature Sensor
(2°C to 150°C)**



Full-Range Centigrade Temperature Sensor



Hình 2.6 Sơ đồ nối chân LM35

Trong cấu hình mạch phía bên trái, cảm biến chỉ có thể đo nhiệt độ dương từ 2 °C đến 150 °C. Theo cấu hình mạch này, chúng ta chỉ cần cấp nguồn cho LM35 và kết nối đầu ra trực tiếp với bộ chuyển đổi tương tự sang số.

Trong cấu hình mạch thứ hai, chúng ta có thể đo nhiệt độ toàn dải từ -55 °C đến 150 °C. Cấu hình mạch này hơi phức tạp nhưng mang lại kết quả cao. Trong trường hợp này, chúng ta phải kết nối một điện trở bên ngoài (R_1) để chuyển mức điện áp âm lên dương. Giá trị điện trở bên ngoài có thể được tính toán theo công thức ghi bên dưới cấu hình mạch.

2.3 ADC0804

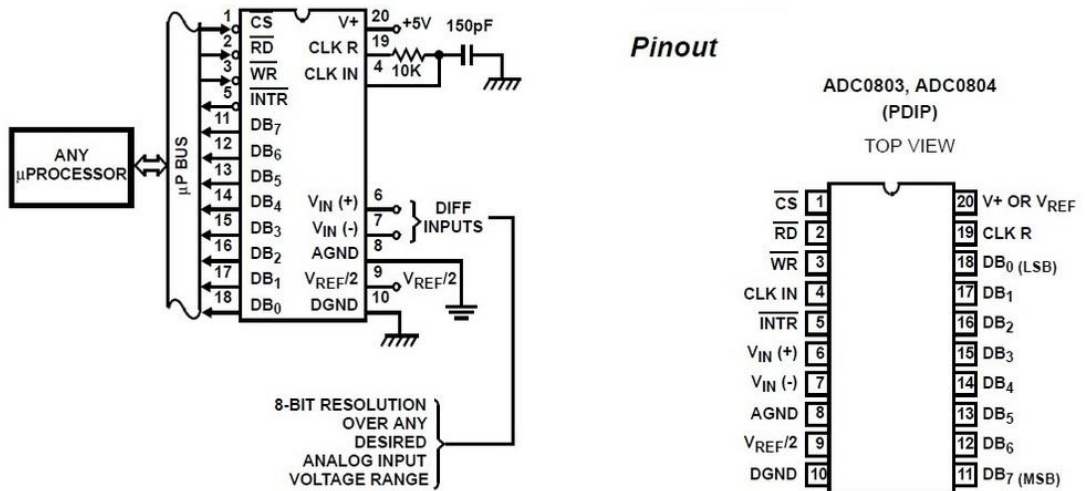
ADC0804 là IC chuyển đổi tín hiệu analog sang digital 8bit có thể chuyển đồng thời 8 đầu vào analog. Giá trị đầu ra digital có thể thay đổi trong khoảng từ 0 đến 255. Nó sử dụng bộ chuyển đổi xấp xỉ (Successive approximation converter) dựa vào thang đo điện áp vi sai (Differential potentiometric ladder).

ADC0804 là IC điện áp thấp sử dụng để chuyển đổi tín hiệu analog sang tín hiệu digital 8bit điện áp thấp. Hoạt động với nguồn 0-5 V, có 1 đầu vào Analog và 8 chân digital đầu ra.

Các tính năng của ADC0804

- Tương thích với tất cả các bộ vi điều khiển và bộ vi xử lý hoạt động ở 5 V.
- Có thể tính điện áp khác nhau từ 0 đến 5V bằng cách chỉ sử dụng một nguồn cấp 5V duy nhất.
- Hoạt động với mức điện áp tham chiếu khác nhau. Mức tối thiểu là 2,5 V.
- Hỗ trợ các linh kiện điện tử CMOS và TTL.
- Nó có một xung nhịp bên trong với tần số 640KHz.
- Vận hành không cần hiệu chỉnh về 0
- Thời gian chuyển đổi tín hiệu tối thiểu 110us
- Kích thước rộng 0,3 inch với package DIP 20 chân
- Có đầu vào điện áp analog khác nhau.
- Dải giá trị đầu ra digital từ 0 – 255
- Dải điện áp đầu vào 2,5V – 6,5V
- Hoạt động độc lập với bộ vi xử lý 8 bit bên trong.
- Khi $V_{ref} = 5V$, cứ mỗi lần tăng 19,53mV giá trị analog thì sẽ tăng một bit ở đầu ra digital.

Sơ đồ chân ADC0804

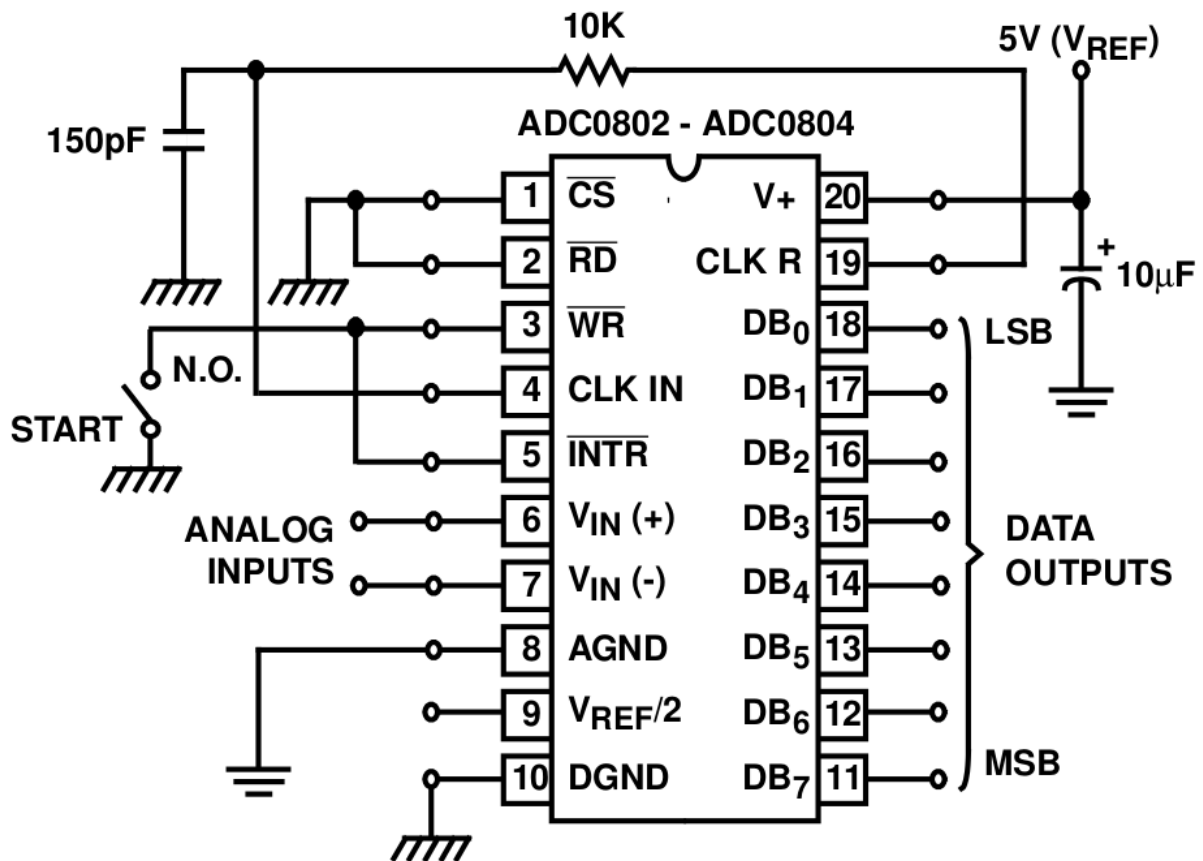


Hình 2.7 Sơ đồ chân ADC0804

1. CS: Là chân Chip select được sử dụng để chọn thiết bị. Khi sử dụng nhiều ADC thì chân này được sử dụng để chọn thiết bị thực hiện. Kích hoạt ở mức logic thấp.
2. RD: Là chân đọc tín hiệu. Chân RD được sử dụng khi muốn nhận giá trị đầu ra từ thanh ghi bên trong. Xung thay đổi logic từ cao xuống thấp sẽ kích hoạt chức năng của chân này
3. WR: Là chân đầu vào Write được sử dụng để bắt đầu chuyển đổi tín hiệu khi được cấp xung thay đổi từ cao xuống thấp
4. CLK IN: Là chân đầu vào sử dụng để nhận xung clock bên ngoài.
5. INTR: Là chân ngắt báo quá trình chuyển đổi tín hiệu đã hoàn tất. Nó sẽ ở mức logic thấp khi quá trình chuyển đổi hoàn tất.
6. Vin (+): Là chân đầu vào analog cho tín hiệu đảo. Hầu hết các thiết bị cấp tín hiệu analog ở dạng không đảo nên sử dụng chân này làm đầu vào analog.
7. Vin (-): Là chân đầu vào analog cho tín hiệu đảo. Do bản chất tín hiệu hầu hết ở dạng không đảo, nên mắc chân vào mass.
8. AGND : Chân mass cho đầu vào analog

9. View : Được sử dụng để cấp điện áp tham chiếu cho phép đọc đầy đủ thang đo giá trị dòng điện.
10. DGND : Chân này mắc vào mass của linh kiện nhận đầu ra digital.
11. DB0 – DB7: Là đầu ra digital ở dạng 8-bit.
12. CLK R: Chân này dùng để định thời RC sử dụng xung nhịp bên trong.
13. VCC: Chân cấp nguồn. Không được lớn hơn +6,5 V. Chủ yếu sử dụng nguồn +5.0 V.

Sơ đồ nối chân



Hình 2.8 sơ đồ nối chân ADC 0804

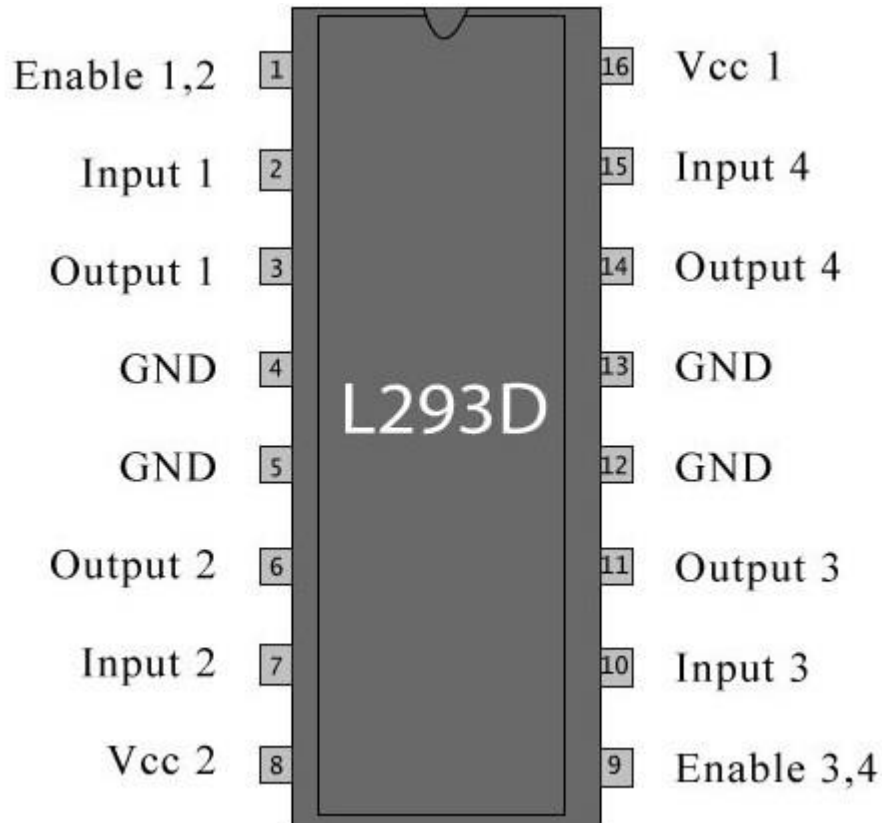
2.4 Khối điều khiển động cơ L293D

L293D về cơ bản là một IC trình điều khiển hay bộ điều khiển động cơ. Nó có hai mạch cầu H tích hợp có thể điều khiển đồng thời hai động cơ DC theo cả chiều kim đồng hồ và ngược chiều kim đồng hồ. Nó hoạt động như một bộ khuếch đại dòng cao vì nó lấy tín hiệu dòng điện thấp ở đầu vào và cung cấp tín hiệu dòng điện cao hơn ở đầu ra để điều khiển các tải khác nhau, ví dụ động cơ bước và động cơ DC. Các tính năng của nó bao gồm phạm vi điện áp nguồn đầu vào lớn, tín hiệu đầu vào chống nhiễu cao dòng điện đầu ra lớn,... Các ứng dụng thực tế phổ biến của nó bao gồm trình điều khiển động cơ bước, trình điều khiển relay, trình điều khiển động cơ DC,...

thông số kỹ thuật L293D

- Có thể được sử dụng để chạy hai động cơ DC với cùng một IC.
- Có thể kiểm soát tốc độ và hướng
- Điện áp động cơ V_{cc2} (V_s): 4,5V đến 36V
- Dòng động cơ cao nhất tối đa: 1.2A
- Dòng động cơ liên tục tối đa: 600mA
- Điện áp cung cấp cho V_{cc1} (v_{ss}): 4,5V đến 7V
- Thời gian chuyển tiếp: 300ns (ở 5V và 24V)
- Tự động tắt nhiệt
- Có các gói DIP, TSSOP, SOIC 16 chân

Sơ đồ chân



Hình 2.9 Sơ đồ chân L293D

Số chân	Tên chân	Mô tả
1	Enable 1,2	Chân này bật chân đầu vào Input 1 (2) và Input 2 (7)
2	Input 1	Trực tiếp điều khiển chân Output 1. Điều khiển bằng mạch kỹ thuật số
3	Output 1	Được kết nối với một đầu của động cơ 1
4	Ground	Chân ground được nối với mass của mạch (0V)
5	Ground	Chân ground được nối với mass của mạch (0V)
6	Output 2	Được kết nối với một đầu khác của động cơ 1
7	Input 2	Điều khiển trực tiếp chân Output 2. Điều khiển bằng mạch kỹ thuật số
8	Vcc2 (Vs)	Kết nối với chân điện áp để chạy động cơ (4,5V đến 36V)
9	Enable 3,4	Chân này bật chân đầu vào Input 3 (10) và Input 4 (15)

10	Input 3	Điều khiển trực tiếp chân Output 3. Điều khiển bằng mạch kỹ thuật số
11	Output 3	Được kết nối với một đầu của động cơ 2
12	Ground	Chân ground được nối với mass của mạch (0V)
13	Ground	Chân ground được nối với mass của mạch (0V)
14	Output 4	Được kết nối với một đầu khác của động cơ 2
15	Input 4	Trực tiếp điều khiển chân Output 4. Điều khiển bằng mạch kỹ thuật số
16	Vcc2 (Vss)	Kết nối với + 5V để bật chức năng IC

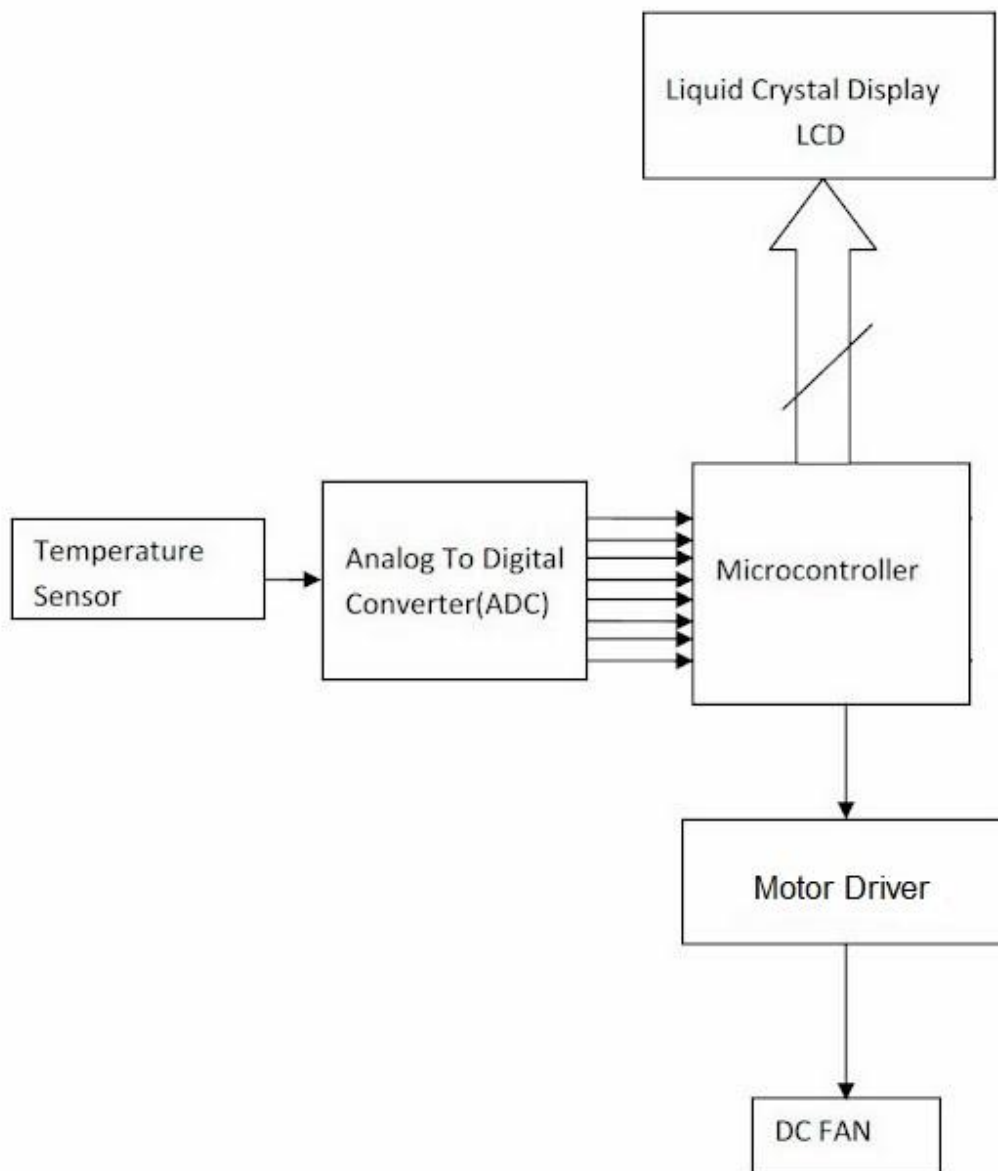
Bảng 2.3 Chức năng của chân L293D

Chapter 3: SYSTEM IMPLEMENTATION PROCESS

3.1 Sơ đồ khối

Hình trên là sơ đồ khối của hệ thống mà em đã thiết kế. Về cơ bản, nó bao gồm một bộ vi điều khiển là bộ xử lý trung tâm của toàn bộ hoạt động điều khiển. Cảm biến nhiệt độ LM35 sẽ đo nhiệt độ từ môi trường cung cấp điện áp đầu ra tương tự. Điện áp tương tự này được cấp cho bộ chuyển đổi ADC0804. Bộ chuyển đổi ADC0804 sau đó chuyển đổi điện áp đầu vào tương tự từ cảm biến nhiệt độ thành các bit nhị phân tương đương. Bộ chuyển đổi được chuyển đổi dữ liệu nhị phân từ bộ chuyển đổi ADC0804 được áp dụng cho vi điều khiển. Bộ vi điều khiển đọc dữ liệu nhị phân từ bộ chuyển đổi ADC0804, chuyển đổi nó sang dạng phù hợp và thực hiện các hoạt động khác nhau dựa trên giá trị nhiệt độ đọc được từ bộ chuyển đổi ADC0804 thông qua bộ điều khiển động cơ L293D

Màn hình LCD được sử dụng để hiển thị dữ liệu do vi điều khiển cung cấp. Trong hệ thống này, nếu nhiệt độ đọc được < 25 độ C, nó được coi là trạng thái bình thường. Trong điều kiện này, quạt tắt nhưng nhiệt độ và trạng thái được hiển thị trên màn hình LCD. Nếu nhiệt độ của phòng lớn hơn 25 độ C đó được coi là tình huống bật quạt.

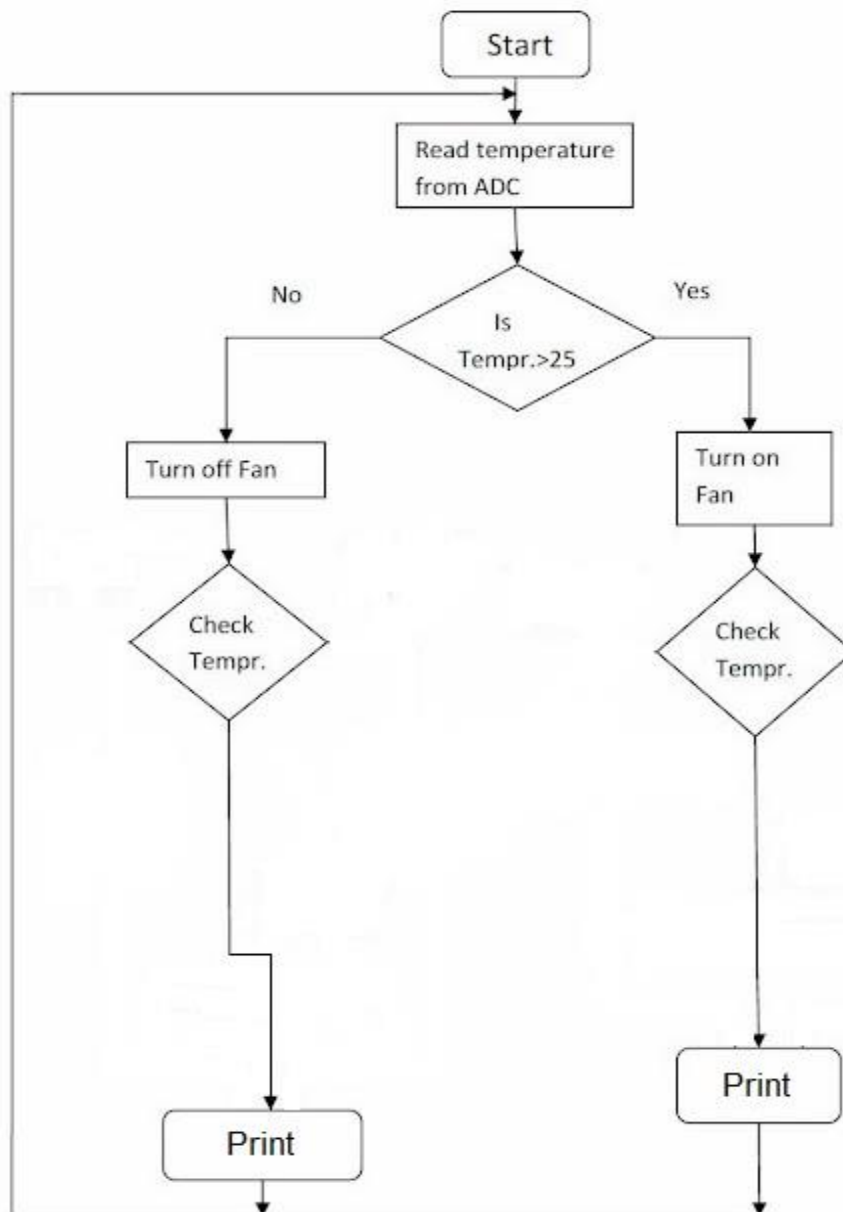


Hình 3.1 sơ đồ khối của mạch

3.2 Lưu Đồ giải thuật

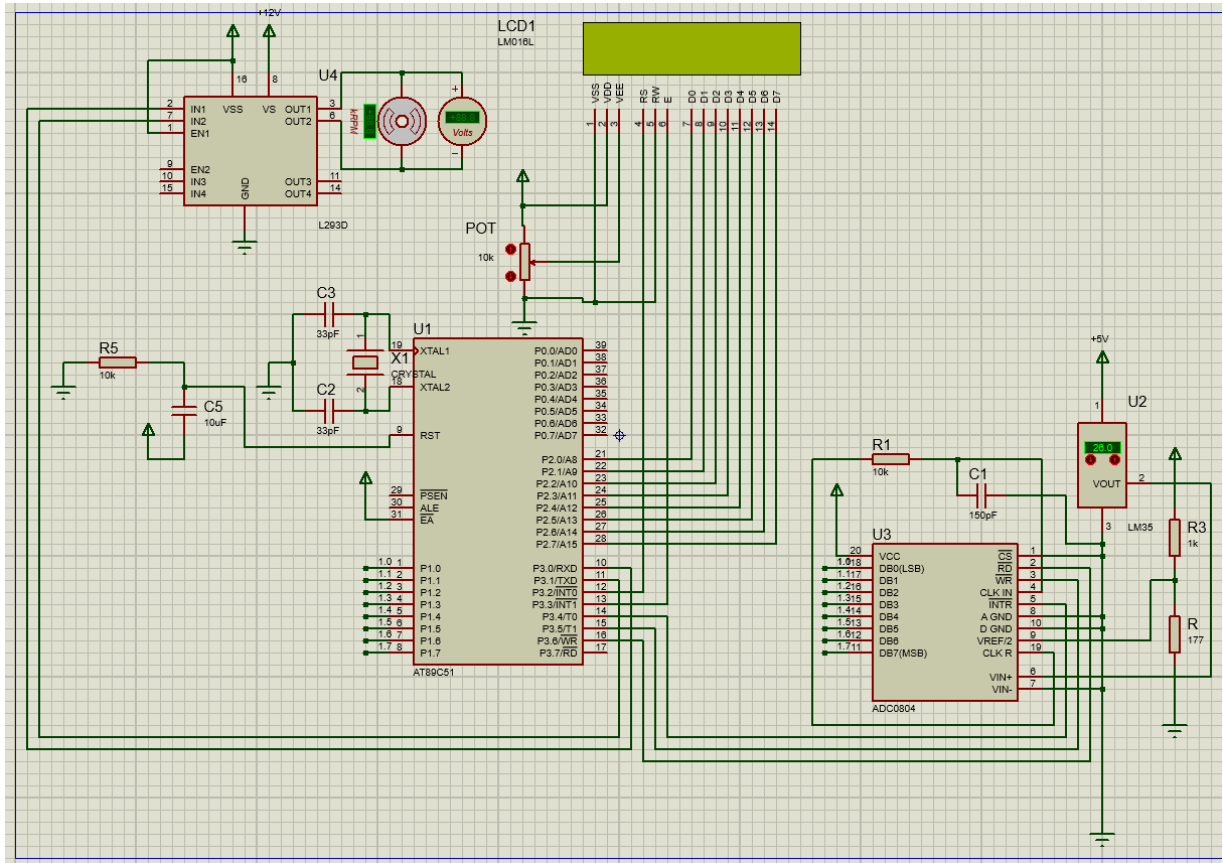
Đầu tiên là cảm biến nhiệt độ LM35 sẽ đo nhiệt độ từ môi trường, sau đó chuyển thành điện áp ra chân Vout đưa vào ADC0804 để xử lý. Tín hiệu analog sau khi được ADC chuyển đổi thành tín hiệu số được đưa vào lưu trữ trong thanh ghi của vi điều khiển 8051 và hiển thị lên màn hình LCD. So sánh giá trị nhiệt độ với ngưỡng

bật/tắt quạt: Giá trị nhiệt độ đọc được từ LM35 sẽ được so sánh với giá trị ngưỡng bật/tắt quạt. Nếu nhiệt độ cao hơn hoặc bằng ngưỡng bật quạt là 25 độ, quạt sẽ được bật và ngược lại nếu nhiệt độ thấp hơn ngưỡng, quạt sẽ được tắt qua. Quạt sẽ được điều khiển bằng IC điều khiển động cơ L293D. Nó được kết nối với vi điều khiển 8051 và quạt. Khi vi điều khiển 8051 gửi tín hiệu điều khiển cho L293D, L293D sẽ điều khiển quạt bật hoặc tắt tùy thuộc vào giá trị ngưỡng nhiệt độ.



Hình 3.2 Lưu đồ giải thuật của mạch

3.3 Thiết kế mạch và nguyên lý hoạt động



Hình 3.3 Thiết kế mạch Proteus

Nguyên lý hoạt động

Để đảm bảo mạch hoạt động ta cần cấp nguồn vào các khối mạch: Chân EA của AT89C51 (Khi EA nối với logic 1(+5V) thì Vi điều khiển thực hiện chương trình lấy từ bộ nhớ nội), Chân VCC của ADC0804, Chân 16 VCC là chân cấp nguồn cho L293D để kích hoạt hoặc để bật nó lên. Điện áp hoạt động từ 4,5 – 7V. Điện áp quá 7V sẽ làm cháy IC. Chân 8 là chân điện áp cho Motor. Không được lớn hơn 36V và không được nhỏ hơn 4,5V. Ở đây sử dụng 12V vì 12V là ngưỡng hoạt động tối đa của DC Fan

Đầu tiên vi điều khiển dùng chân XTAL1, XTAL2 tiến hành cung cấp tín hiệu xung clock cho mạch hoạt động với tần số thạch anh 11.0592Mhz

Tiếp theo là cảm biến nhiệt độ LM35 sẽ đo nhiệt độ từ môi trường, sau đó chuyển thành điện áp ra chân Vout đưa vào chân VIN+ ADC0804 để xử lý. Chân V Là chân đầu vào analog cho tín hiệu đảo. Hầu hết các thiết bị cấp tín hiệu analog ở dạng không đảo nên sử dụng chân này làm đầu vào analog.

Sau khi thực hiện quá trình chuyển đổi, bộ ADC tích cực chân đọc tín hiệu. Chân RD được sử dụng khi muốn nhận giá trị đầu ra từ thanh ghi bên trong được nối với Chân P3.6 của AT89C51 cho phép tín hiệu điều khiển ghi dữ liệu lên bộ nhớ ngoài. Các chân DB của ADC là đầu ra digital ở dạng 8-bit được nối với Port 1 của Vi điều khiển vì Port 1 chỉ có tác dụng nhập xuất dữ liệu từ các thiết bị bên ngoài.

Tiếp theo là khởi tạo LCD, Tín hiệu digital ở dạng 8 bit được đưa vào LCD 16x2 tiến hành giải mã và xuất ra màn hình giá trị đo được từ cảm biến LM35. Vi điều khiển có nhiệm vụ so sánh giá trị nhiệt độ với ngưỡng bật/tắt quạt: Giá trị nhiệt độ đọc được từ LM35 sẽ được so sánh với giá trị ngưỡng bật/tắt quạt. Nếu nhiệt độ cao hơn hoặc bằng ngưỡng bật quạt là 25 độ, quạt sẽ được bật và ngược lại nếu nhiệt độ thấp hơn ngưỡng, quạt sẽ được tắt qua. Quạt sẽ được điều khiển bằng IC điều khiển động cơ L293D. Nó được kết nối với vi điều khiển 8051 và quạt. Vi điều khiển nối trực tiếp với chân 2,7 của L293D trực tiếp điều khiển chân Output. L293D sẽ điều khiển quạt bật hoặc tắt tùy thuộc vào giá trị ngưỡng nhiệt độ.

3.4 Code và nguyên lý hoạt động

```
#include<reg51.h>
#define LCD_DATA P2
//LCD
sbit RS=P3^2;
sbit EN=P3^3;
//ADC0804
sbit RD_ADC=P3^6; //read adc
sbit WR_ADC=P3^5; //wr adc
sbit I_ADC=P3^4; // Interrupt adc
//Fan
sbit FAN1 = P3^0;
sbit FAN2 = P3^1;

void timer(void);
void Delay_ms(unsigned int);
void LCD_init(void);
void LCD_CMD(unsigned char);
void LCD_Char_Cp(char);
void LCD_String (char *);

void timer()//tao ngat tran timer voi ck 1ms
{
    TMOD = 0x01;
    TH0=0xFC;
    TL0=0x17;
    TR0 = 1;           // bat timer
    while (TF0 == 0);
    TR0 = 0;           // dung timer
    TF0 = 0;           // xóa flag ve 0
}
void Delay_ms(unsigned int count)// delay_ms
{
    unsigned int i;
    for(i=0;i<count;i++)
    {
        timer();
    }
}
//ghi lenh ra LCD
void LCD_CMD(unsigned char cmd)
{
    RS=0;           //Ghi lenh
    LCD_DATA=cmd;   //gui CMD ra LCD
    EN=1;           //Cho phep
    EN=0;
    if(cmd<=0x02) // cho nay nhìn lại tap lenh cua LCD là hiệu
    {
        Delay_ms(2);
    }
    else Delay_ms(1);
}
```

```

//hien thi ky tu tai vi tri con tro (Cp=current position)
void LCD_Char_Cp(char char_dat)
{
    RS=1; // ghi du lieu
    LCD_DATA=char_dat;
    EN=1;
    EN=0;
    Delay_ms(1);
}
// hien thi chuoi ky tu tai vi tri con tro
void LCD_String(char *str)
{
    while(*str)
    {
        LCD_Char_Cp(*str++);
    }
}
void LCD_init (void) // ham khoi tao LCD
{
    LCD_CMD(0x38); // giao tiep 8bit, 2 dong, font 5x7
    LCD_CMD(0x0C); // bat hien thi va tat con tro
    LCD_CMD(0x01); // xoa hien thi
    LCD_CMD(0x80); // dua tro ve dau dong 1
    LCD_CMD(0x06); // tu dong tang con tro
}
void convert_display(unsigned char value) //chuyen doi du lieu va xuat ra LCD
{
    unsigned char x0,x1,x2,x3;

    LCD_CMD(0xc9);
    x0 = (value/100);
    x0=x0+(0x30);
    x1=((value % 100)/10); //chia lay phan nguyen roi gan vào x1
    x1=x1+(0x30);
    x2=value%10; //chia lay phan du roi gan vào x2
    x2=x2+(0x30); //chuyen kieu du lieu tu bin sang ascii de hien thi ra lcd bang
    cách add 0x30
    x3=0xDF; //ma của dấu (°)

    if(x0 != 0 + 0x30)
        LCD_Char_Cp(x0);
    if(x1 != 0 + 0x30)
        LCD_Char_Cp(x1); // dua nhiet do ra LCD//chuc
    LCD_Char_Cp(x2); // don vi
    LCD_Char_Cp(x3); // dau (°)
    LCD_Char_Cp('C');
}
void main()
{
    unsigned char value;
    LCD_init();
    LCD_String("MC Project");
    LCD_CMD(0xc0);
}

```

```

LCD_String("Nhiet do: ");
P1=0xFF; //Port 1 là o mode input
I_ADC=1;
RD_ADC=1; //dua chân rd lên mức cao
WR_ADC=1; //dua chân wr lên mức cao
FAN1 = 1;
FAN2 = 1;
while(1)
{
    WR_ADC=0; //tao mot xung wr tu cao xuống thấp
    de cho phép adc chuyển đổi dữ liệu vào
    Delay_ms(1);
    WR_ADC=1;

    while(I_ADC==1); //đợi cho đến khi kết thúc chuyển
    doi
    RD_ADC=0; //RD =0 đọc dữ liệu từ ADC0804
    value=P1; //lay data trên port 1 lưu vào biến
    value
    value = value * 0.58822352941f + 0.5f;
    if(value >= 25)
    {
        FAN1 = 1;
        FAN2 = 0;
    }
    else
    {
        FAN1 = 1;
        FAN2 = 1;
    }
    convert_display(value); //gọi lệnh chuyển đổi và xuất ra LCD
    Delay_ms(1000); //tạm nghỉ để đèn chu kỳ chuyển đổi
    ke tiếp
    RD_ADC=1; //RD=1 bắt đầu chu kỳ chuyển đổi ke
    tiếp
}
}

```

Giải thích:

Khởi tạo LCD: bằng hàm LCD_Init sử dụng lệnh 0x38 để đặt chức năng giao tiếp 8 bit, 2 dòng, 5X8 dots; sử dụng lệnh 0x0C để hiện màn hình, tắt con trỏ; sử dụng lệnh 0x01 để xóa hiển thị; sử dụng lệnh 0x80 và 0x06 để đưa con trỏ về đầu dòng 1 sau đó tự động tăng con trỏ,

Gửi lệnh đến LCD(LCD_CMD): chuyển chân RS xuống 0 để truy xuất lệnh gán LCD_DATA bằng lệnh muốn gửi lên LCD, kích EN lên 1 để cho phép sau đó đưa EN trở về 0. sau đó Delay, Nếu lệnh là 0x01 và 0x02 thì delay 2ms còn các lệnh khác delay

1ms bởi vì lệnh 0x01 và 0x02 trong giao tiếp LCD đều liên quan đến việc xóa màn hình LCD hoặc di chuyển con trỏ về vị trí đầu dòng đầu tiên, đó là những thao tác mà yêu cầu LCD phải thực hiện nhiều tác vụ phức tạp để hoàn thành.

Hàm hiển thị một ký tự tại vị trí con trỏ(LCD_Char_Cp). Đặt RS bằng 1 để chọn chế độ ghi dữ liệu, gán LCD_DATA bằng ký tự cần hiển thị, kích EN lên 1 để cho phép sau đó đưa EN trở về 0, Delay 1ms

Hàm hiển thị một chuỗi tại vị trí con trỏ(LCD_String): đưa vào hàm một con trỏ char của ký tự đầu tiên trong chuỗi, chừng nào con trỏ còn khác NULL thì sử dụng hàm LCD_Char_Cp để hiển thị ký tự mà con trỏ đó trỏ đến và tăng con trỏ lên một đơn vị

Hàm chuyển đổi giá trị và hiển thị ra LCD (convert_display): sử dụng hàm LCD_CMD để truyền lệnh 0xc9 để thiết lập vị trí bắt đầu hiển thị trên màn hình LCD, đặt x0 là số hàng trăm, x1 là số hàng chục, x2 là số hàng đơn vị và x3 = 0xDF để lưu mã ascii của dấu độ (°), nhiệt độ truyền vào hàm là value, $x0 = \text{value} / 100$, $x1 = (\text{value} \% 100) / 10$, $x2 = \text{value} \% 10$, các giá trị sẽ được cộng thêm 0x30 sau đó để chuyển thành mã ascii, nếu x0 khác 0 + 0x30 thì dùng hàm LCD_Char_Cp để hiển thị x0. nếu x1 khác 0 + 0x30 thì dùng hàm LCD_Char_Cp để hiển thị x1. x2 dùng hàm LCD_Char_Cp không cần điều kiện. in ra x3 và sau đó là chữ C để hiển thị °C

Ở hàm main, đưa các chân của ADC lên mức cao. RD: Chân số 2, là chân nhận tín hiệu vào tích cực ở mức thấp. WR: Chân số 3, đây là chân vào tích cực mức thấp được dùng báo cho ADC biết để bắt đầu quá trình chuyển đổi. Ngắt INTR : Chân số 5, là chân ra tích cực mức thấp.

gọi hàm LCD, LCD_Init để khởi tạo LCD

Đưa hai chân FAN1 và FAN2 chính là Input1 và Input 2 của L293D, lên mức 1 để chắc chắn rằng quạt đứng yên trước khi nhiệt độ tăng lên ngưỡng, cho WR_ADC về 0 để tạo một xung từ cao xuống thấp cho phép ADC chuyển đổi dữ liệu vào. Delay 1ms, sau đó đưa trở lại WR_ADC lên lại mức cao, Đợi cho đến khi quá trình chuyển đổi kết thúc. Tức là đợi cho đến khi I_ADC bằng 0 vì khi I_ADC chuyển từ 1 về 0 thì vi điều khiển sẽ nhận được tín hiệu rằng quá trình chuyển đổi của ADC đã kết thúc. sau đó kích hoạt chân RD_ADC bằng cách đưa nó từ trạng thái cao xuống thấp(RD_ADC = 0) để có thể đọc dữ liệu đã chuyển đổi từ ADC. án biến value(lưu đầu ra của ADC) = P1(các chân của vi điều khiển nối với các chân DB của ADC nhận dữ liệu sau khi chuyển đổi từ ADC)

Nhiệt độ sẽ được tính bằng công thức $t = V_{out}(mV) / 10mV$. với V_{out} được tính bằng $\text{value} * V_{ref}(V) / (2^n - 1)$. trong đó n là số bit của ADC, ở đây sử dụng ADC 8 bit nên $n = 8$ và V_{ref} là điện áp tham chiếu trong đồ án được sử dụng là 1.5V.

Kết quả của nhiệt độ sẽ được gán trở lại vào biến value bằng công thức $value = value * 0.58822352941f + 0.5f$; trong đó 0.58822352941 là kết quả gần đúng của $V_{ref}(V) * 1000 / ((2^8 - 1) * 10)$

Kết quả này được cộng thêm 0.5 để tránh việc nhiệt độ bị làm tròn xuống. sau đó nếu nhiệt độ lớn hơn hoặc bằng ngưỡng là 25 độ thì FAN1 = 1 và FAN2 sẽ bằng 0 để quạt quay theo cùng chiều kim đồng hồ. ngược lại FAN1 = 1 và FAN2 = 1 để quạt ngừng quay.

Sau đó gọi hàm chuyển đổi và xuất nhiệt độ ra LCD (hàm `convert_display`) delay 1000ms, kích RD_ADC lên trở lại 1 và tiếp tục chu kì mới.

Chapter 4: TEST RESULTS AND EVALUATION

4.1 Phương pháp thực nghiệm: Mô phỏng

Một số ưu điểm của việc thử nghiệm kết quả mô phỏng bao gồm:

- Chi phí thấp hơn: Thay vì phải xây dựng một mô hình mạch hoặc thực nghiệm để kiểm tra kết quả, mô phỏng có thể được thực hiện với chi phí thấp hơn, đặc biệt là khi các mô hình mạch hoặc thực nghiệm phải mất nhiều thời gian và tài nguyên.
- Tiết kiệm thời gian: Mô phỏng có thể được thực hiện trong một thời gian ngắn hơn so với thực nghiệm hoặc mô hình mạch, cho phép nhanh chóng kiểm tra và đánh giá các ý tưởng mới.
- Tính linh hoạt: Mô phỏng cho phép thử nghiệm nhiều kịch bản khác nhau và điều chỉnh các tham số để tìm kiếm các giải pháp tối ưu.
- Độ chính xác cao: Mô phỏng có thể cung cấp kết quả với độ chính xác cao, cho phép em đánh giá các tính năng hoặc hiệu suất của các sản phẩm hoặc quá trình.
- Hiệu quả trong việc giảm thiểu rủi ro: Mô phỏng có thể giúp giảm thiểu rủi ro bằng cách cho phép em kiểm tra các kịch bản khác nhau và đánh giá các rủi ro tiềm năng trước khi triển khai một sản phẩm hoặc quá trình.

Tóm lại, mô phỏng là một công cụ hữu ích để kiểm tra và đánh giá các giải pháp mới một cách hiệu quả và tiết kiệm chi phí.

Mặc dù việc sử dụng mô phỏng có nhiều ưu điểm, nhưng nó cũng có một số nhược điểm sau:

- Mô hình chưa chính xác: Nếu mô hình mô phỏng không được xây dựng hoặc thiết lập đúng cách, kết quả của nó có thể không chính xác hoặc không phản ánh được hiện thực.
- Thiếu tính toàn vẹn của dữ liệu: Mô phỏng cần phải dựa trên các giả định về tính chất của các vật liệu, điều kiện môi trường, và các yếu tố khác. Nếu các giả định này không chính xác hoặc thiếu tính toàn vẹn của dữ liệu, kết quả mô phỏng có thể không phù hợp với thực tế.
- Cần kỹ năng cao để thiết lập và chạy mô hình: Việc thiết lập và chạy mô hình mô phỏng cần phải được thực hiện bởi các chuyên gia có kiến thức và kỹ năng cao, do đó, nó có thể yêu cầu thời gian và chi phí đào tạo để đạt được sự thành công.

- Cần phải được xác thực: Kết quả của mô phỏng cần phải được xác thực bằng các thực nghiệm hoặc kiểm tra thực tế để đảm bảo tính chính xác của nó.
- Có thể bị giới hạn về phạm vi và độ phức tạp: Mô phỏng có thể bị giới hạn về phạm vi và độ phức tạp, đặc biệt là trong trường hợp các quá trình phức tạp hoặc các ứng dụng đòi hỏi tính toán cao.

Tóm lại, việc sử dụng mô phỏng cần phải được thực hiện cẩn thận để đảm bảo tính chính xác và phù hợp với mục đích sử dụng của nó.

4.2 Kết quả mô phỏng trên Proteus

Mạch hoạt động và chạy tốt

Số lần đo	Nhiệt độ mô phỏng	Nhiệt độ đo được	Quạt hoạt động
1	23	21	Không
2	24	23	Không
3	25	24	Không
4	26	25	Có
5	27	26	Có
6	28	26	Có
7	29	28	Có

Bảng 4.1 Kết quả mô phỏng

Kết luận:

- Có sự sai số nhẹ trong khoảng 1-2 độ trong quá trình đo. Bởi vì lẽ đó mà khi nhiệt độ mô phỏng 25 độ bằng với ngưỡng nhiệt độ lập trình sẵn mà quạt vẫn không chạy là do nhiệt độ đo được chỉ bằng 24 nên quạt không quay
- Hạn chế chưa tính ra đúng giá trị để có thể chạy ra kết quả đúng với giá trị nhiệt độ thực tế
- Nhìn chung, có sự khác biệt giữa nhiệt độ mô phỏng và nhiệt độ đo được ở mỗi lần đo, và sự khác biệt này tăng dần khi nhiệt độ mô phỏng tăng. Điều này cho thấy rằng mô hình mô phỏng có thể không chính xác hoặc cần được điều chỉnh để đảm bảo tính chính xác của kết quả.
- Tuy nhiên, ta có thể thấy rằng quạt bắt đầu hoạt động khi nhiệt độ mô phỏng đạt từ 26 đến 29 độ C, điều này có thể được sử dụng để đánh giá tính hiệu quả của quạt trong điều kiện mô phỏng. Tuy nhiên, để đánh giá tính hiệu quả của quạt trong điều kiện thực tế, ta cần phải thực hiện thêm các thử nghiệm hoặc kiểm tra thực tế khác để đảm bảo tính chính xác của kết quả.

Chapter 5: SUMMARY, RESTRICTIONS AND DEVELOPMENT ORIENTATIONS

5.1 Tính ứng dụng

Quạt hoạt động dựa trên cảm biến nhiệt độ có thể áp dụng vào thực tế, phục vụ vào đời sống con người. Ứng dụng được trong Y tế, công nghiệp, ô tô, Transit, ... Điển hình nhất là mô hình này được lồng vào trong máy lạnh, quạt sẽ thổi liên tục đến một ngưỡng nhiệt độ nhất định và sẽ ngừng thổi để tiết kiệm điện năng.

Ưu điểm:

- Tự động điều khiển quạt: Mô hình này cho phép quạt hoạt động tự động, không cần sự can thiệp của người dùng. Quạt sẽ được bật và tắt tự động dựa trên nhiệt độ môi trường.
- Tiết kiệm năng lượng: Quạt sẽ chỉ hoạt động khi nhiệt độ môi trường vượt qua giới hạn được cài đặt, giúp tiết kiệm năng lượng.
- Độ tin cậy cao: Mô hình sử dụng vi điều khiển 8051, được biết đến với độ tin cậy cao và khả năng xử lý tốt.
- Dễ dàng cài đặt và sử dụng: Mô hình này có thiết kế đơn giản và dễ dàng cài đặt, người dùng có thể dễ dàng sử dụng mô hình mà không cần có kiến thức chuyên môn sâu về vi điều khiển.
- Chi phí thấp: Vi điều khiển 8051 là một trong những loại vi điều khiển phổ biến và có giá thành rẻ, giúp giảm chi phí cho mô hình.

Tóm lại, mô hình quạt bật tắt tự động sử dụng vi điều khiển 8051 là một giải pháp đơn giản, hiệu quả và tiết kiệm năng lượng để điều khiển quạt.

Nhược điểm:

- Giới hạn chức năng: Mô hình này chỉ cho phép bật và tắt quạt tự động dựa trên nhiệt độ môi trường. Nó không có khả năng thay đổi tốc độ quạt hoặc hướng gió như một số mô hình quạt thông minh khác.
- Không thể điều chỉnh ngưỡng nhiệt độ: Mô hình này có ngưỡng nhiệt độ mặc định để kích hoạt hoặc tắt quạt. Người dùng không thể thay đổi ngưỡng này theo nhu cầu của mình.
- Không thể đọc được nhiệt độ chính xác: Mô hình này sử dụng cảm biến nhiệt độ đơn giản để đo nhiệt độ môi trường. Cảm biến này có thể không đọc được nhiệt độ chính xác trong một số trường hợp.
- Không có tính năng đảo chiều gió: Mô hình này không cho phép người dùng thay đổi hướng gió của quạt.

- Không thể điều khiển từ xa: Mô hình này không có tính năng điều khiển từ xa, người dùng chỉ có thể điều khiển quạt từ bên trong phòng.

Tóm lại, mặc dù mô hình quạt bật tắt tự động sử dụng vi điều khiển 8051 có nhiều ưu điểm nhưng cũng có một số nhược điểm nhất định. Người dùng nên cân nhắc các yêu cầu của mình trước khi quyết định sử dụng mô hình này để đảm bảo phù hợp với nhu cầu của mình.

5.2 Kết quả và phát triển

Trong đề tài lần này bọn em đánh giá đã hoàn thành khá tốt được đề tài, ứng dụng tốt những kiến thức truyền đạt từ mentor bao gồm interface LCD 16x2, timer ... Cùng với đó là học và biết thêm về những khối mới như ADC0804, LM35, L293D. Hiểu sâu và rõ hơn về các hoạt động của vi điều khiển 8051 nói chung và vi điều khiển AT89C51 nói riêng.

Qua đề tài này chúng em được tự chủ động tiếp thu kiến thức và tính ứng dụng thực tiễn. Bên cạnh đó cũng có khó khăn trong quá trình thực hiện vì kiến thức còn chưa đủ nên cả 2 rất cố gắng để tìm hiểu và học hỏi thêm. Đề này là nền tảng và cơ sở để em phát triển và ứng dụng được thêm nhiều thứ vào những môn sau.

REFERENCES.

- [1] https://github.com/AdityaKshetri/Temperature_Controlled_Fan
- [2] <https://www.elprocus.com/temperature-controlled-dc-fan-using-microcontroller/>
- [3] <https://luanvan.co/luan-van/do-an-vi-xu-li-trong-dieu-khien-62295/>
- [4] <https://dientuviet.com/cam-bien-nhiet-do-lm35/#ftoc-heading-1>
- [5] <https://blog.mecsu.vn/gioi-thieu-ic-adc0804/>
- [6] <https://bkladh.blogspot.com/2013/11/8051-based-temperature-controlled.html>
- [7] <https://dientutuonglai.com/tim-hieu-l293d.html>