MỤC LỤC

[LỜI NÓI ĐẦU 1](#_Toc312714275)

[Phần 1: Nguyên tắc chung 3](#_Toc312714276)

[Phần 2. Cấu tạo 4](#_Toc312714277)

[2.1.Vật liệu – Linh kiện 4](#_Toc312714278)

[2.2 Mô tả kỹ thuật 4](#_Toc312714279)

[Phần 3: Thiết kế phần mềm 6](#_Toc312714280)

[3.1 Chi tiết thiết kế 6](#_Toc312714281)

[3.2 Phần mạch chính 13](#_Toc312714282)

[3.3 Lập trình cho 89S52 14](#_Toc312714283)

[KẾT LUẬN 29](#_Toc312714285)

[Tài liệu tham khảo 31](#_Toc312714286)

# 

# LỜI NÓI ĐẦU

Như chúng ta đã biết, vi xử lý có vai trò rất lớn trong đới sống của con người. Cùng với sự phát triển của khoa học kỹ thuật, nhiều thế hệ vi xử lý mới đã ra đời với nhiều chức năng và nhỏ gọn. Chính vì vậy, càng ngày vi xử lý càng thâm nhập sâu vào đời sống con người.

Nhằm giúp các sinh viên tự động hóa hiểu rõ hơn về môn vi xử lý cũng như nâng cao khả năng thưc hành môn học. Các sinh viên đã được tổ chức làm bài tập lớn với đề tài thiết kế robot dò đường. Một vấn đề rất được quan tâm khi thiết kế robot là phải lập trình làm sao cho robot có thể đi theo một quỹ đạo mong muốn. Điều này phụ thuộc rất nhiều vào người thiết kế. Bản báo cáo này được đưa ra để giải quyết các vấn đề trên với một robot dò đường đơn giản.

Đề tài này có sự tham gia của tất cả các thành viên trong nhóm.Tất cả đã cùng hợp tác với nhau để cho ra sản phẩm cuối cùng.

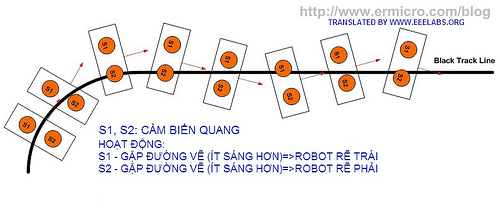
Mặc dù đã cố gắng rất nhiều, nhưng chắc chắn bản báo cáo này sẽ còn nhiều thiếu sót vì đây là lần đầu em tham gia làm mạch nên có rất ít kinh nghiệm. Rất mong được sự góp ý của thầy.

Em cũng xin chân thành cảm ơn thầy Vũ Văn Hà! Trong thời gian thiết kế robot, chúng em cũng nhận được nhiều sự giúp đỡ của thầy cũng như những kiến thức rất bổ ích trong học kỳ này.

# PHẦN 1. NGUYÊN TẮC CHUNG

Thật ra, robot dò đường là 1 biến thể đặc biệt của robot hướng sáng. Sở dĩ nói như vậy là do chúng có cùng nguyên tắc hoạt động là sử dụng cảm biến quang điện ( quang trở hoặc diode hồng ngoại) để so sánh cường độ sáng từ đó điều chỉnh hướng đi thích hợp.Tuy nhiên, ở robot dò đường, cảm biến được bố trí gần mặt đường và nguồn sáng để so sánh lúc này do chính robot tạo ra. Nhưng do đâu lại có sự sai lệch về cường độ sáng. Câu trả lời nằm ở đường vẽ, đường vẽ này có tính chất khác với xung quanh, thường thì nó có màu đen để hấp thụ ánh sáng. Khi robot đi lệch vào vùng có vạch vẽ, ánh sáng phát ra từ robot không phản xạ lại như bình thường mà bị đường kẻ hấp thụ 1 phần làm sai lệch độ sáng giữa 2 cảm biến. Việc còn lại là thiết kế sao cho robot có hành vi khắc phục sự sai lệch đó và ta có được loại robot đi theo đường vẽ.

Nhiệm vụ của các cảm biến quang được mô tả như hình sau:



Đường đi được vạch sẵn với chiều dài mặt cắt ngang xác định. Đường đi được đặt trên nền có vật liệu khác loại. Nếu đường đi là vật liệu phản quang thì nền không phản quang và ngược lại.

Ôtô sử dụng các LED hồng ngoại chiếu xuống mặt đường đi hoặc nền. Ánh sáng hồng ngoại sẽ phản xạ lại khi gặp vật liệu phản quang. Sử dụng các bộ thu các ánh sáng hồng ngoại phản quang này, ôtô xác định được trạng thái của mình trên mặt đường: lệch trái hoặc phải so với mặt đường, còn hoặc không còn trên mặt đường,…; từ đó đưa ra quyết định điều khiển tương ứng.

# PHẦN 2. CẤU TẠO

## 2.1. Vật liệu – Linh kiện

* 2x động cơ một chiều 12V(DC motor)
* 1x IC7805, 1x tản nhiệt
* 6x cặp thu phát hồng ngoại, 6x mắt thu hồng ngoại
* 4x IC LM393N, 1x IC L293D, 1x IC 89S52
* 2x Cáp 10 sợi, 3x juck 2 sợi
* 6x điện trở 1kΩ, 6x điện trở 4.7kΩ, 6x điện trở 20kΩ
* 3x Board mạch
* 2x tụ thường 33p, 2x tụ sứ 10µ, 1x tụ hóa 470µ
* 10x đèn LED đỏ
* 6x Diode 1N4007
* 1x thạch anh 12MHz
* 1xBăng đen và giấy A0 để làm đường đi
* 1xPin 11,1 V (3 cell)
* 2xNút nhấn

## 2.2. Mô tả kỹ thuật

Thiết kế xe tự động dò đường thỏa mãn các yêu cầu sau:

* Xe dừng sau khi chạy 3 vòng
* Đảm bảo dò đúng đường.
* Xe chạy đảm bảo tốc độ ổn định.
* Có đèn LED hiển thị đúng trạng thái hiện tại của xe trên đường.
* Xe có khả năng đảo chiều quay động cơ, hãm ngược.
* Thiết kế phần mềm dễ nhìn, dễ kiểm tra, sửa chữa đảm bảo thực hiện đúng yêu cầu đề bài.

Dựa trên yêu cầu của đề bài và bám sát với nội dung môn học, nhóm chọn phương án thiết kế:

* Xe vận hành có 3 bánh, 2 bánh sau chủ động, bánh trước xoay tròn định hướng (bánh lái).
* Xe dẫn động bằng 2 động cơ 1 chiều đặt ở hai bánh sau.
* Phần dò đường dùng 6 cặp cảm biến (1 led phát hồng ngoại và 1 quang trở.), tín hiệu đưa về đi qua khâu so sánh LM293D để đảm bảo mức điện áp đưa vào vi điều khiển.
* Phần xử lý chọn vi điều khiển 89S52 của Atmel.
* Xe có hiển thị trạng LED (báo nguồn, động cơ…) tương ứng với trạng thái xe.
* Phần điều khiển chuyển động chọn 2 mạch cầu H được tích hợp sẵn trong L298.

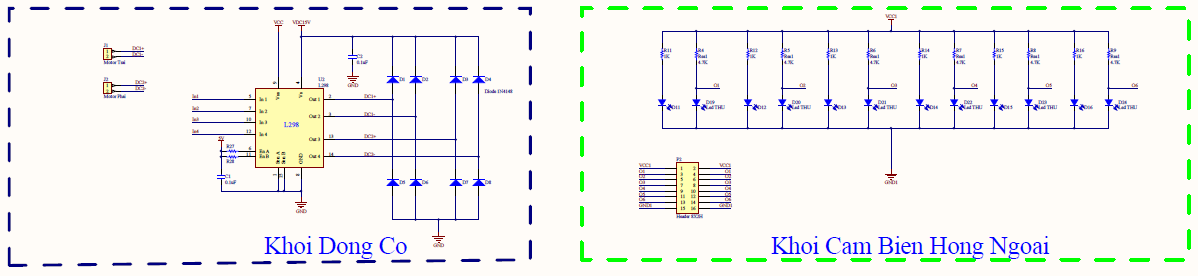
# PHẦN 3. THIẾT KẾ PHẦN MỀM

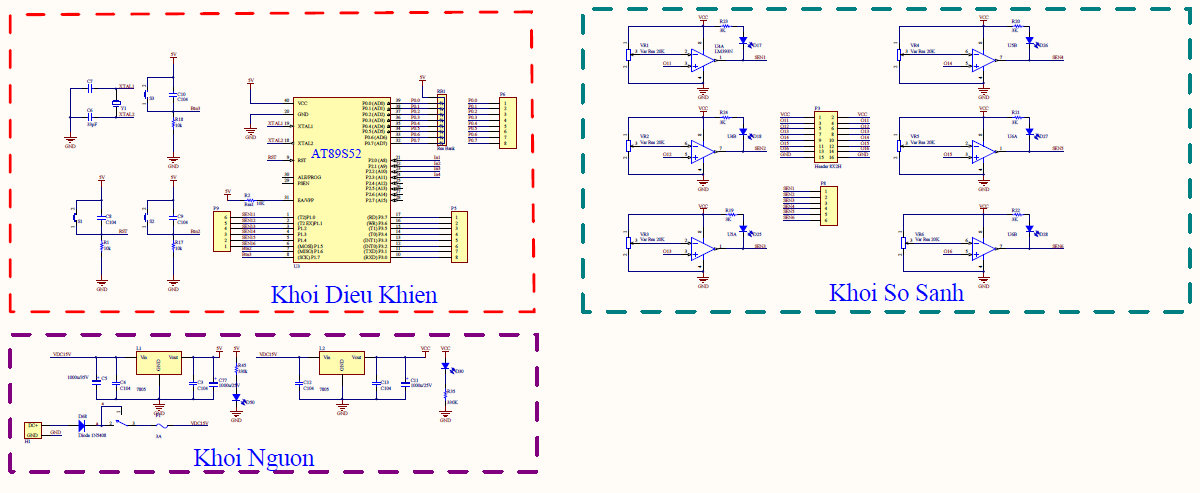
**Tổng quan thiết kế sẽ bao gồm 5 khối chính như sau:**

* Khối nguồn: cung cấp điện áp 12V cho động cơ và 5V cho khối vi điều khiển.
* Khối vi điều khiển.
* Khối sensor: nhận tín hiệu đường đi.
* Khối so sánh.
* Khối điều khiển 2 động cơ DC.

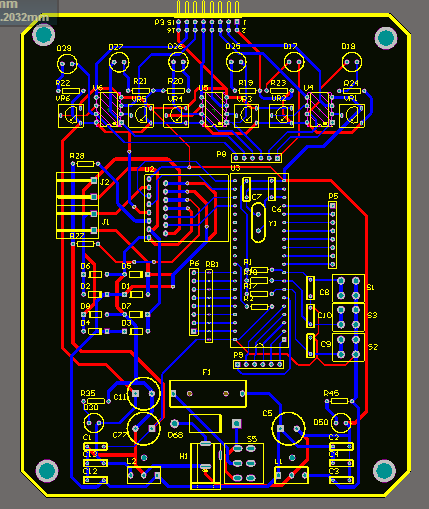
## 3.1 Chi tiết thiết kế

* Sơ đồ nguyên lý các khối.



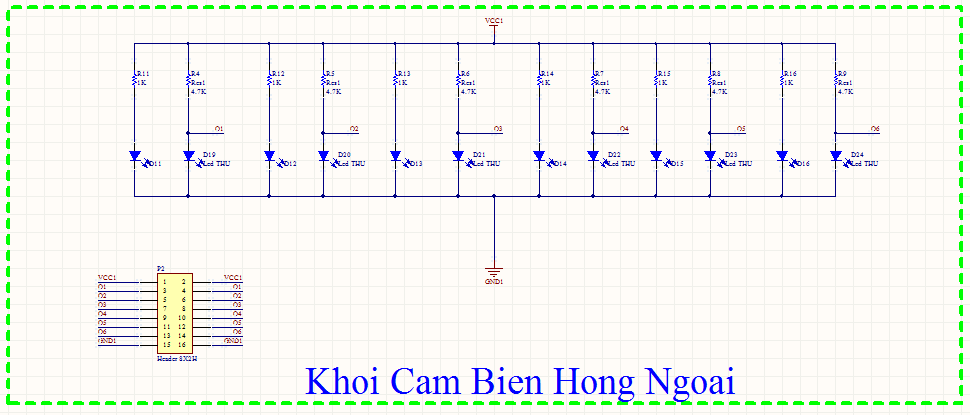


* Sơ đồ mạch đi dây.



a.Khối senser

Dùng 6 LED phát hồng ngoại và 6 quang trở:



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  |  | | | | | Khi Led phát tín hiệu khi có tín phản lại thì quang trở sẽ thay đổi điện trở nó phát tín hiệu vào LM393 để so sánh rồi đưa vào vi điều khiển để vi điều khiển xử lý rồi đưa ra tín hiệu để phát lệnh điều khiển động cơ. |

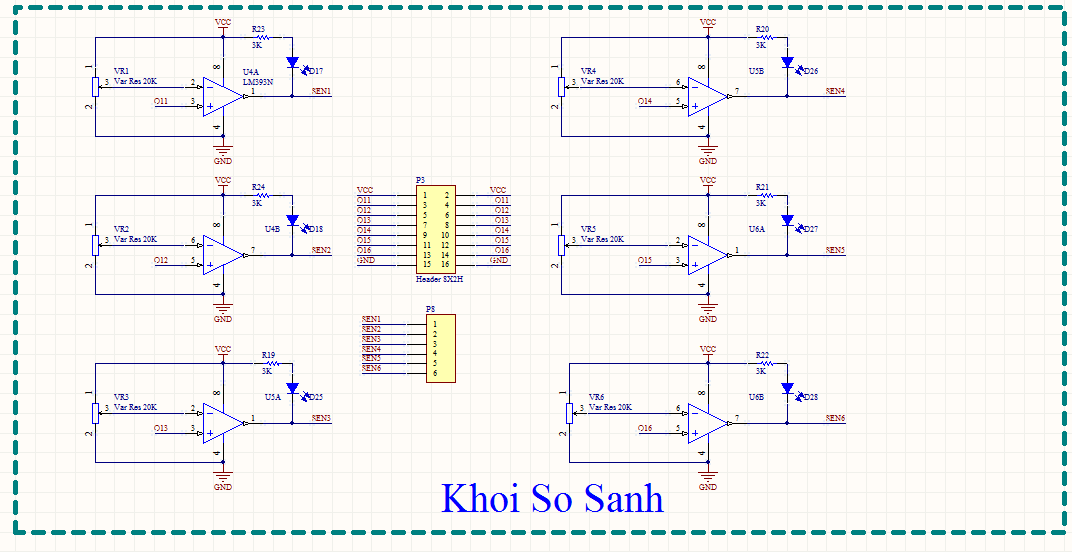
|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  |  | |

Chức năng của khối:

+ Tạo tín hiệu phản hồi về khối vi xử lý trung tâm để từ đó điều khiển động cơ.

+ Nguyên lý hoạt động: Khi có ánh sáng thì điện trở của quang trở sẽ giảm và ngược lại, điện trở của nó thay đổi cỡ từ 0.5k (khi có ánh sáng) đến 250k (không có ánh sáng), ta sử dụng led phát làm nguồn sáng cho nó. Khi gặp nền trắng, ánh sáng sẽ phản xạ lên quang trở làm điện trở nó giảm xuống và khi gặp vạch đen (băng keo đen) thì ánh sáng sẽ khó phản xạ nên quang trở nhận ít ánh sáng => trở nó tăng. Từ đó dựa vào 2 quang trở, robot có thể phân biệt được vạch đen trên nền trắng (dưới sự hỗ trợ của opamp so sánh).

b.Khối so sánh



Chức năng của mạch so sánh:

|  |  |
| --- | --- |
|  |  |

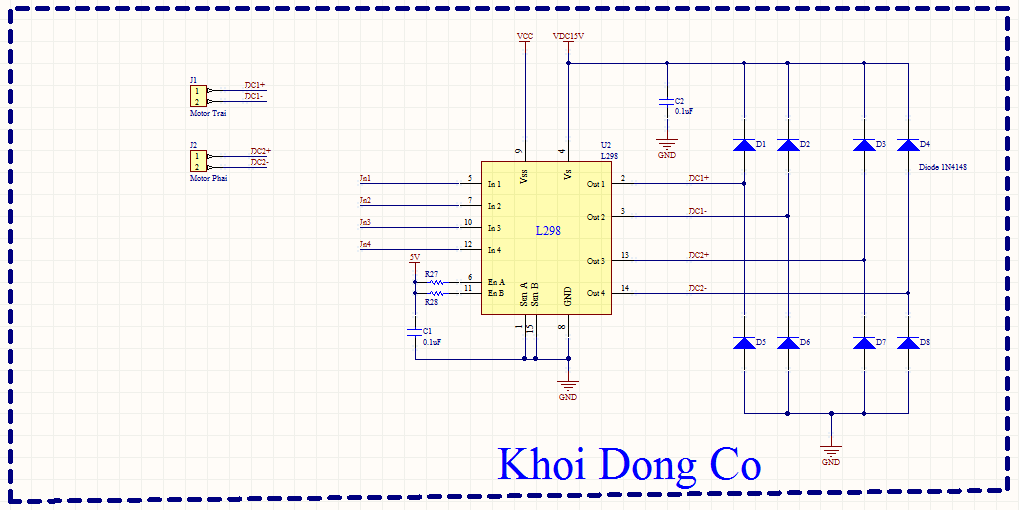
|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | Do tín hiệu ra từ sensor như nằm ở 1 mạch phân áp nên điện áp này sẽ dao động phụ thuộc vào độ sáng phản xạ lại, do đo ta nên dùng 1 khối so sánh LM393 để tạo mức tín hiệu phù hợp cho VĐK xử lý hoạt động.  Mức điện áp đầu ra so sánh được nhận biết bằng sáu led ở sáu đầu ra so sánh, nếu gặp vạch trắng, mức điện áp ra là 0V, đèn sáng và nếu gặp vạch đen, mức điện áp ra là 5V, đèn tắt. | |

c.Khối động cơ

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | Loại động cơ ta sử dụng là 2 động cơ DC-12V, phục vụ cho việc điều khiển xe tiến, lùi, tăng tốc, quay phải, quay trái, với các cấp tốc độ khác nhau | |

Ta lựa chọn L298 để điều khiển động cơ với 2 mạch cầu H đã tích hợp sẵn ở bên trong.

Sơ đồ khối điều khiển:



d.Khối nguồn

Dùng nguồn 12V để cấp điện áp vào, sau đó thông qua khối lọc, hạ áp dùng 7805 ta lấy ra điện áp 5V ổn định để cung cấp cho các khối còn lại.

Ngoài ra còn có một số biện pháp bảo vệ như dùng cầu chi 2A, dùng diode chống ngược nguồn.

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  |  | |

e.Khối điều khiển

Sử dụng vi xử lý 8 bit AT89S2 có cấu trúc tương tự như 89C51 (8051).

Đây là VĐK thuộc họ MCS-51, với 8K bytes bộ nhớ, có thể đọc/ghi dữ liệu khoảng 1000 lần.

Điện áp hoạt động đầu vào từ 4.0 đến 5.5V

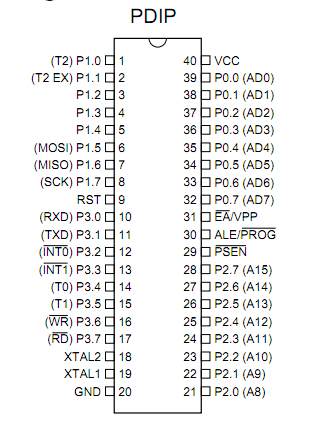
Tần số cung cấp tĩnh từ 0 đến 33 kHz.

32 đường I/O có thể lập trình được.

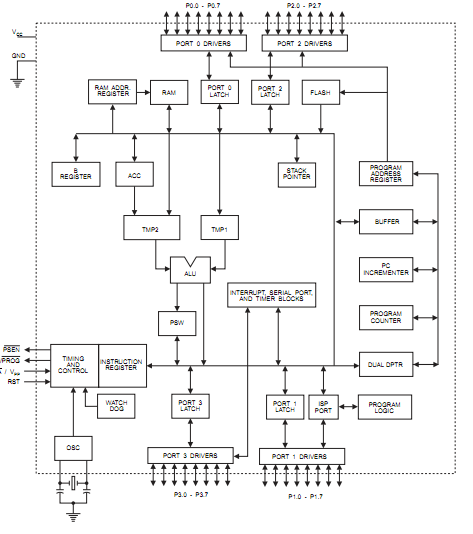
3 Timer/Counter 16 bit

8 nguồn ngắt ngoài

Cấu trúc chân và sơ đồ khối bên trong:

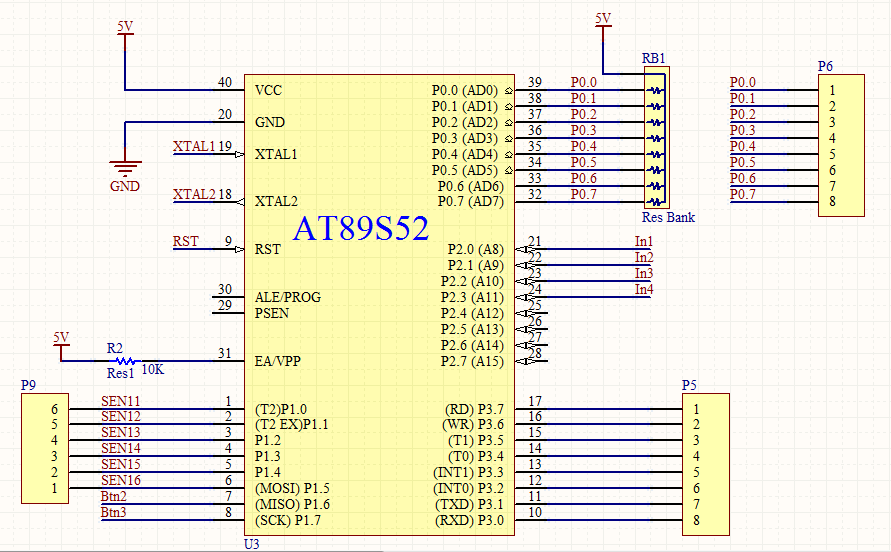


Sơ đồ khối bên trong:



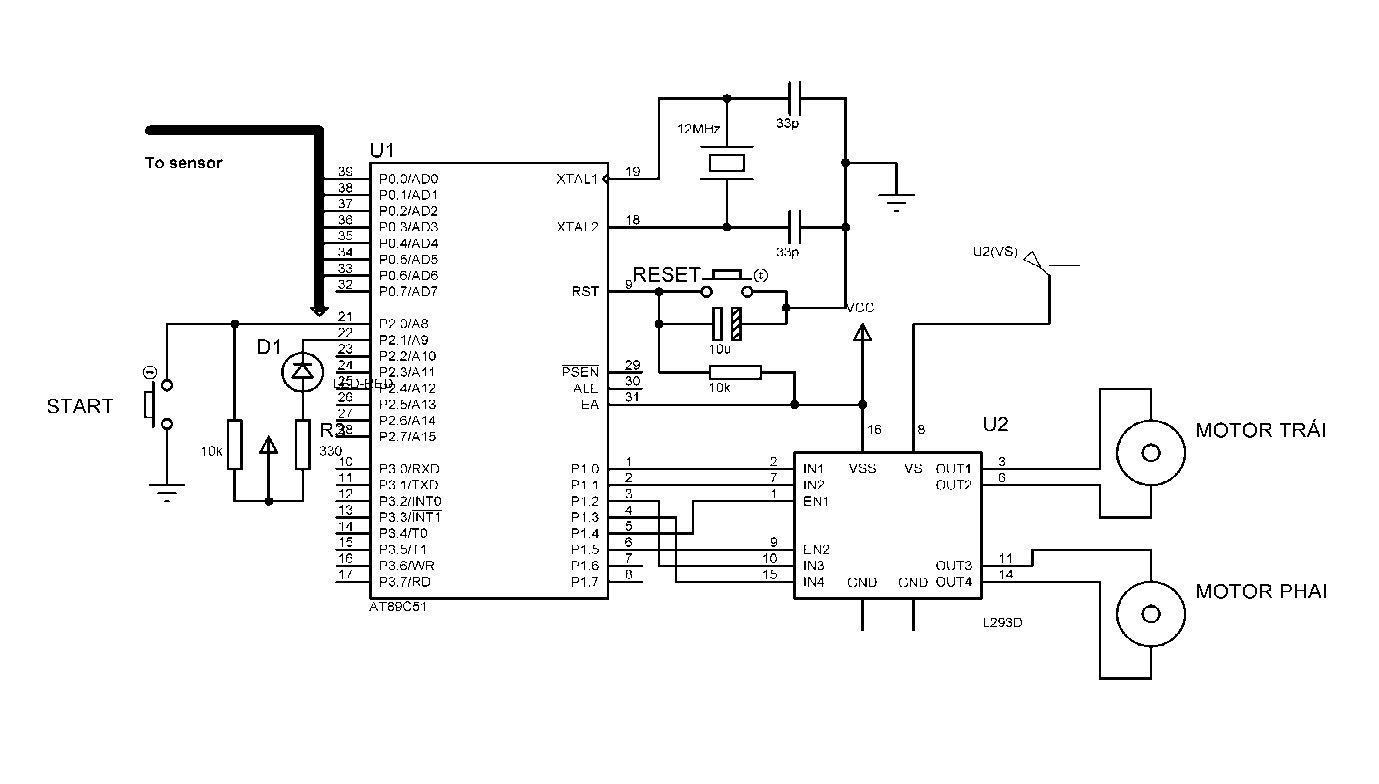
Trong mạch ứng dụng em dùng thạch anh 12MHz. Các chân tín hiệu sensor được đưa vào Port1, tín hiệu điều khiển động cơ ra từ Port2.

Sơ đồ chi tiết thiết kế:



## 3.2 Phần mạch chính

*Sơ đồ mạch chính:*

****

## 3.3 Lập trình cho 89S52

a. Sơ đồ khối điều khiển được tóm tắt như sau:

**MẠCH**

**SENSORS**

**MẠCH**

**VI ĐIỀU KHIỂN**

**ĐIỀU KHIỂN ĐỘNG CƠ**

**XE CHUYỂN ĐỘNG**

Lưu đồ thuật toán điều khiển:

**KHỞI TẠO**

**GIỚI THIỆU**

**DÒ ĐƯỜNG**

**LỆCH**

YES

NO

**Cua trái**

**Or Cua phải**

**Đi thẳng**

YES

NO

**Chương trình con khi bắt vạch đích**

**VẠCH**

Đơn giản ở đây ta chỉ sử dụng phương pháp điều khiển động cơ theo kiểu ON/OFF.

b.Code chương trình bằng ngôn ngữ C:

Chương trình viết cho vi điều khiển sử dụng ngôn ngữ Assembler với trình dịch là phần mềm Keil C. Ưu điểm của ngôn ngữ Assembler là mã nguồn chạy nhanh và gọn, giúp sinh viên nắm chắc cấu trúc phần cứng của vi điều khiển cũng như các bước lập trình phần mềm. Tuy nhiên nhược điểm của ngôn ngữ này là khó hiểu và rắc rối.

Dưới đây là chương trình viết bằng ngôn ngữ C cho IC AT89S52. Chương trình được viết và dịch trên phần mềm keil C:

#include<REGX51.H>

//==========Dinh nghia chieu quay banh========

#define go 1

#define back 0

//--------------------------------------------

#define out 0x0FF //0B11111111

//==========Dinh nghia mat do=================

#define R5 0x0BF // 0B11011111

#define R4 0x09F // OB11001111

#define R3 0x0DF // OB11101111

#define R2 0x0AF // OB11100111

#define R1 0x0EF // OB11110111

#define TT 0x0E7 // OB11110011

#define L1 0x0F7 // OB11111011

#define L2 0x0F3 // OB11111001

#define L3 0x0FB // OB11111101

#define L4 0x0F9 // OB11111100

#define L5 0x0FD // OB11111110

//---------------------------------------------

//===Dinh nghia Pin xung PWM, Pin dieu khien===

sbit dk1a = P1^0;

sbit dk1b = P1^1;

sbit dk2a = P1^2;

sbit dk2b = P1^3;

sbit xung1 = P1^4;

sbit xung2 = P1^5;

//----------------------------------------------

//===============Khai bao=======================

unsigned char toc\_do1,toc\_do2;

unsigned char x=0, y=0, m, n;

unsigned long int counter0 = 0;

unsigned char INPUT;

//==============Ham tao tre====================

void delay (unsigned long int a)

{

unsigned long int i;

for (i = 0; i <= a; i++);

}

//----------------------------------------------

//===============Dieu khien banh trai============

void banh\_trai (unsigned char toc\_do0,unsigned char chieu)

{

if (chieu == go)

{

TR0 = 1;

tocdo1 = m + (toc\_do0 - 10) \* n;

dk1a = 1;

dk1b = 0;

}

if (chieu == back)

{

TR0=1;

toc\_do1 = m + (toc\_do0 - 10) \*n;

dk1a = 0;

dk1b = 1;

}

}

//-----------------------------------------------------

//===============Dieu khien banh phai============

void banh\_phai (unsigned char toc\_do0,unsigned char chieu)

{

if (chieu == go)

{

TR1 = 1;

toc\_do2 = m + (toc\_do0 - 10) \*n;

dk2a = 1;

dk2b = 0;

}

if (chieu == back)

{

TR1 = 1;

toc\_do2 = m + (toc\_do0 - 10) \*n;

dk2a = 0;

dk2b = 1;

}

}

//-----------------------------------------------------

//==============Tao xung PWM banh trai=================

void ngatt0 (void) interrupt 1

{

x++;

if (x <= 25)

{

TR0 = 0;

TL0 = 256 – toc\_do1;

xung1 = 1;

TR0 = 1;

}

else if (x < 50)

{

TR0 = 0;

TL0 = 256 - (200 – toc\_do1);

xung1 = 0;

TR0 = 1;

}

else if (x == 50) {x = 0;}

}

//-----------------------------------------------------

//==============Tao xung PWM banh phai=================

void ngatt1 (void) interrupt 3

{

y++;

if (y <= 25)

{

TR1 = 0;

TL1 = 256 – toc\_do2;

xung2 = 1;

TR1 = 1;

}

else if (y < 50)

{

TR1 = 0;

TL1 = 256 - (200 – toc\_do2);

xung2 = 0;

TR1 = 1;

}

else if (y == 50) {y=0;}

}

//-----------------------------------------------------

//================Ham do duong=========================

void doduong(unsigned char so\_vong)

{

unsigned char nho\_trai, nho\_phai, nhoT, nhoP, start = 0, startdk=0;

unsigned char a;

unsigned char INPUT;

while (start <= so\_vong)

{

INPUT = P0; //nhan thong tin mat do

if ((P0\_0 == 0) && (P0\_7 == 0))

{ if (startdk == 0) {start++;}

startdk = 1;

a=1; m=30; n=3;

while (a) {if((P0\_0 == 1) && (P0\_7 == 1)) {a=0;}}

}

if ((startdk == 1) && (P0\_0 == 0) && (P0\_1 == 0) && (P0\_2 == 0))

{

banh\_phai (11, go);

banh\_trai (9, go);

nho\_trai = 1; nhoT = 0;nhoP = 0;

startdk = 0;

a=1; m=40; n=3;

while (a) {if (P0 == out) {a=0;};}

delay (1000);

}

switch (INPUT)

{

case TT:

{

banh\_phai (10, go);

banh\_trai (10, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case L1:

{

banh\_phai (11, go);

banh\_trai (9, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case L2:

{

banh\_phai (12, go);

banh\_trai (8, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case L3:

{

banh\_phai (13, go);

banh\_trai (7, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case L4:

{

banh\_phai (14, go);

banh\_trai (6, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case L5:

{

banh\_phai(15, go);

banh\_trai (5, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case R1:

{

banh\_phai (9, go);

banh\_trai (11, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case R2:

{

banh\_phai (8, go);

banh\_trai (12, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case R3:

{

banh\_phai (7, go);

banh\_trai (13, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case R4:

{

banh\_phai (6, go);

banh\_trai (14, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case R5:

{

banh\_phai (5, go);

banh\_trai (15, go);

nho\_trai = 0;

nho\_phai = 0;

break;

}

case out:

{

if (nhoP == 1)

{nho\_phai = 1; nhoP = 0;}

if (nhoT == 1)

{nho\_trai = 1; nhoT = 0;}

else if (nho\_trai == 1)

{

banh\_phai (18, go);

banh\_trai (4, back);

}

else if (nho\_phai == 1)

{

banh\_phai (4, back);

banh\_trai (18, go);

}

break;

}

default:

{

If ((INPUT != R1) || (INPUT != R2) || (INPUT != R3) ||

(INPUT != R4) || (INPUT != R5) ||(INPUT != L1) || (INPUT != L2) ||

(INPUT != L3) || (INPUT != L4) || (INPUT != L5) ||(INPUT != out))

{

Banhphai (10, go);

banhtrai (10, go);

}

break;

}

}

}

}

//=================START===========================

void start ()

{

char exit, key;

exit = 1;

while (exit)

{

Key = P2\_0;

if (key == 0)

{exit = 0;}

}

}

//---------------------------------------------------

//==================stop==============================

void stop ()

{

TR0 = 0;

TR1 = 0;

xung1 = 0;

xung2 = 0;

}

//===================Ham chinh=======================

void main()

{

TMOD = 0x22;

IE = 0x08B;

IT0 = 1;

IP = 0;

M = 40; n = 3;

start ();

while (1) {P2\_1 = 0; do\_duong (2); stop (); while (1);}

}

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  |  |  KẾT LUẬN |  | | | |

Trong quá trình thực hiện bài tập lớn, chúng em gặp một số khó khăn sau và đã từng bước khắc phục.

* *Tài liệu*

Đây là lần đầu tiên thực hiện một sản phẩm thực dựa trên nền kiến thức học tập được nên nhiều vấn đề chưa được thông suốt. Trước hết, là cách làm xe về cơ khí, thiết kế mạch, dò đường ra sao, cua xe trái phải thuận ngược.

*Khắc phục:* học hỏi từ hướng dẫn của thầy giáo, các bạn trong lớp đặc biệt là các bạn đã từng tham dự cuộc thi Robocon, học hỏi từ các anh các khóa trước, tham khảo nguồn tài liệu trên mạng Internet….

*Kết quả*: Học hỏi được nhiều kinh nghiệm làm robot đơn giản từ các nguồn trên, các tutorial về thiết kế mạch, thiết kế cơ khí, băm xung, ngắt…

* *Thiết kế mạch*

*Vấn đề*: Chưa có kinh nghiệm về thiết kế mạch thực tế, cần giải quyết các vấn đề: vẽ mạch nguyên lý và mạch in sử dụng phần mềm Altium Designer 9.0, các kĩ thuật sắp xếp linh kiện, đi dây…

Do thời gian cũng khá ngắn nên không thể làm mạch bằng tay nên em đã tiến hành đặt mạch cho kịp thời gian nên chi phí đắt.

*Khắc phục*: Học cách sử dụng các phần mềm nói trên với nhiều tutorial trên các diễn đàn như [www.dientuvietnam.net](http://www.dientuvietnam.net), [www.picvietnam.com](http://www.picvietnam.com) , trong mạch cũng sử dụng thư viện chân linh kiện download trên hai diễn đàn trên đã được chúng em chỉnh sửa lại.

* *Lập trình*

Đây có thể coi là vấn đề phức tạp nhất vì là linh hồn của xe tự hành, điều khiển toàn bộ hoạt động của xe và cũng là vấn đề tiêu tốn nhiều thời gian nhất.

*Vấn đề 1:* Đọc và xử lý tín hiệu SENSORS

Tính toán, thu thập các trường hợp xe lệch ra khỏi đường, xe gặp khúc cua, xe không bắt được vạch đen, xe bắt vạch đích, xe cua ở góc cua gấp…

Xử lý các trường hợp trên như thế nào?

Trong quá trình test xe, nhiều khi Sensors bắt không chuẩn hoặc không ổn định.

*Khắc phục*: Để thu thập các trường hợp, dùng tay đẩy xe đi theo đường đặc biệt là các khúc cua và ghi tín hiệu hiển thị trên các led báo về, tính toán các trường hợp lệch trái, lệch phải, lệch ít, lệch nhiều, không lệch, bắt vạch đích…

Hàn chắc các mối hàn mạch SENSORS, kiểm tra cáp nối với mạch vi điều khiển, điều chỉnh biến trở để lấy điện áp so sánh hoạt động ổn định đưa vào IC L298.

*Kết quả:*Mạch hoạt động ổn định, SENSORS bắt rất tốt, tín hiệu báo về trên mạch vi điều khiển chuẩn xác.

* *Vấn đề 2:* Không sử dụng phương pháp điều chế độ rộng xung PWM để đưa vào chân IN1, IN3 điều khiển chiều quay cũng như tốc độ quay của hai động cơ truyền động do đó xe chuyển động chưa được trơn mà bị giật một chút, tốc độ xe chưa cao, mômen động cơ còn nhỏ khó vượt qua chướng ngại vật nhỏ để đi tiếp.

Trên đây là những báo cáo sơ lược về robot dò đường. Mặc dù đã cố gắng nhiều nhưng không thể tránh khỏi những sai sót.

# TÀI LIỆU THAM KHẢO

1. Tống Văn On. *Họ Vi điều khiển 8051.*
2. Nguyễn Tăng Cường, Phan Quốc Thắng. *Cấu trúc và lập trình họ vi điều khiển 8051.*
3. *Bài giảng trên lớp*
4. Website:

*Datasheetarchive.com*

[www.dientuvietnam.net](http://www.dientuvietnam.net)

[www.picvietnam.com](http://www.picvietnam.com)

[www.8051projects.com](http://www.8051projects.com)

[www.dieukhien.net](http://www.dieukhien.net)