

## ĐỒ ÁN VI XỬ LÝ

### “ ĐO VÀ ĐIỀU KHIỂN TỐC ĐỘ ĐỘNG CƠ DÙNG 8051”

#### Lời nói đầu

Trong những năm gần đây với sự phát triển mạnh mẽ của khoa học – kĩ thuật.Đặc biệt của ngành công nghiệp chế tạo các linh kiện bán dẫn,các các vi mạch tích hợp IC một hướng phát triển mới của Vi xử lý đã hình thành đó là Vi Điều Khiển.Với những ưu điểm to lớn về tốc độ,độ chính xác cao,khả năng xử lý các bài toán,tính linh hoạt nên các Vi Điều khiển đã được ứng dụng trên hầu hết các lĩnh vực trong cuộc sống. Bằng cách áp dụng Vi Điều Khiển trong sản xuất và xử lý,Vi Điều Khiển đã thực sự thể hiện được các ưu điểm của mình so với các thiết bị thông thường khác.

Vì những ứng dụng to lớn của Vi điều khiển,do đó mà ở các trường Đại Học,Cao Đẳng,TCCN.... Về khoa học – công nghệ .Môn vi xử lý đã trở thành một môn học không thể thiếu được trong trương trình đào tạo.vi điều khiển 8051 sẽ cung cấp cho sinh viên những khái niệm cơ bản cách thức hoạt động của Vi xử lý qua đó sinh viên có tư duy ,kiến thức nền tảng,để có thể giải quyết các bài toán ứng dụng thực tế trong cuộc sống,cũng như là cơ sở để học tập nghiên cứu các dòng Vi xử lý khác như :PIC,AVR....

Qua đồ án này chúng em có cái nhìn thực tế hơn,sâu sắc hơn về vi điều khiển.chúng em cũng đã hiểu thêm nhiều về cách thức xử lý một bài toán thực tế phức tạp.

Với sự hường dẫn tận tình của thầy:Đào Đức Thịnh chúng em hi vọng chúng em có thể hoàn thành tốt đồ án này.

Chúng em xin chân thành cảm ơn thầy!

## CHƯƠNG I. GIỚI THIỆU CHUNG

### I.1. khảo sát hệ thống.

Hiện nay trong rất nhiều lĩnh vực đời sống và sản xuất các loại động cơ điện ngày càng được ứng dụng rộng rãi hơn so với những loại động cơ sử dụng năng lượng như xăng, dầu, khí đốt..... Những loại động cơ điện này có những ưu điểm hơn hẳn về hiệu suất sử dụng, cấu tạo nhỏ gọn, giá thành hợp lý, dễ dàng điều chỉnh tốc độ, đảo chiều, cưỡng bức các quá trình khởi động, quá trình hãm dừng dễ dàng. Trong thực tế có rất nhiều loại động cơ điện được sử dụng bao gồm:

+ Động cơ điện 3 pha được chia làm các loại cơ bản là: Động cơ điện 3 pha dây quấn và 3 pha roto lồng sóc, động cơ điện 1 pha.

+ Động cơ điện 1 chiều bao gồm các loại như: kích từ song song và loại kích từ nối tiếp.

Thông thường với những động cơ thường xuyên đòi hỏi đảo chiều, tăng, giảm, hãm dừng thì thường sử dụng động cơ điện 1 chiều là chủ yếu, vì sẽ dễ dàng điều khiển hơn so với động cơ xoay chiều. Để tiến hành điều khiển động cơ 1 chiều, có rất nhiều biện pháp được ứng dụng như là: thay đổi điện áp phản ứng, thay đổi từ thông, hoặc sử dụng điện trở phụ mắc thêm vào phản ứng của động cơ. Để làm được điều đó, chúng ta cần đi xây dựng những hệ thống điều khiển, có rất nhiều hệ thống được ứng dụng như là: các hệ thống điều khiển PID, điều khiển động cơ bằng Vi điều khiển, hay có thể điều khiển bằng cách sử dụng các loại khí cụ điện. Trong đề tài này chúng em sẽ tiến hành nghiên cứu các quá trình điều khiển động cơ bằng phương pháp sử dụng Vi điều khiển họ 8051.

### I.2. Nhiệm vụ và yêu cầu kĩ thuật.

\* Thiết kế một mạch điện điều khiển động cơ DC 12 v. với các yêu cầu quay thuận, quay nghịch, tăng tốc, giảm tốc, và đảo chiều quay động cơ.

\* Thực hiện thao tác điều chỉnh các chức năng trên thông qua các nút ấn bao gồm: nút khởi động động cơ, nút điều khiển quay thuận, nút điều khiển quay ngược, nút tăng tốc, nút giảm tốc, và nút dừng động cơ, thông qua các nút ấn này cho phép ta có thể điều chỉnh động cơ làm việc ở 1 tốc độ như mong muốn.

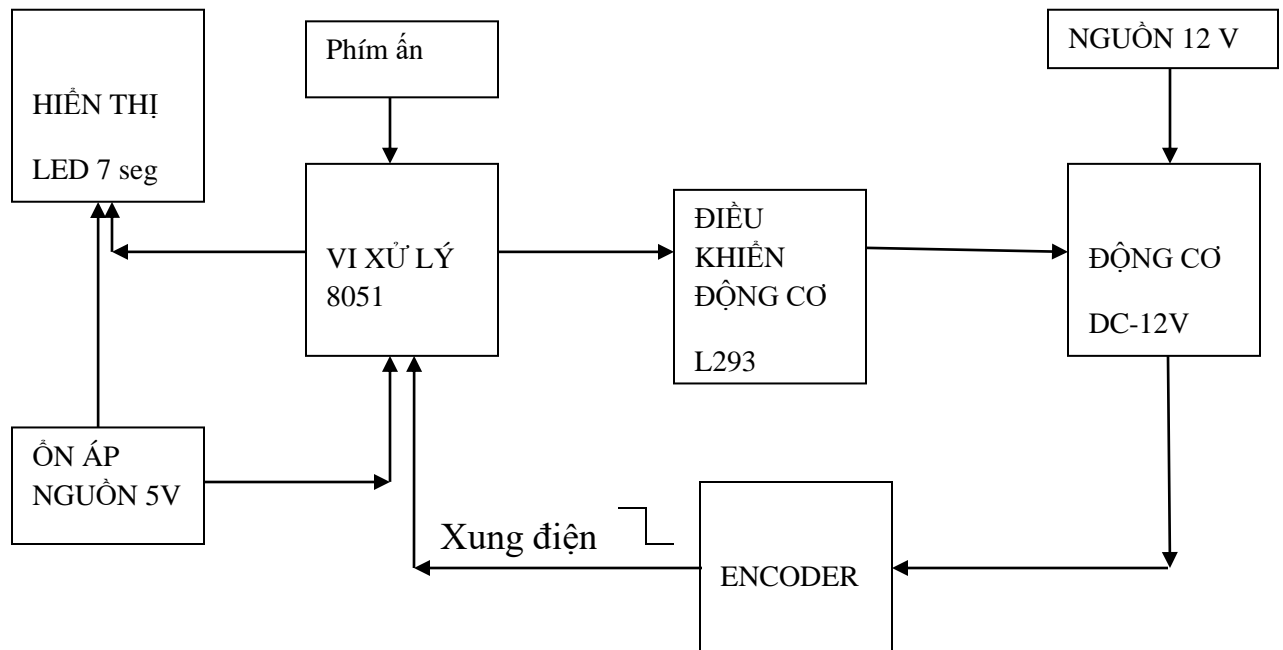
- \* Hệ thống cho phép hiển thị tốc độ làm việc có thể sử dụng LCD hoặc LED 7 thanh, đồng thời có các led đơn thông báo trạng thái quay thuận, quay ngược, dừng còn giá trị tốc độ động cơ được hiển thị chi tiết là bao nhiêu trên LED 7 đoạn.
- \* Thông qua đề tài, làm quen với cách thức, nguyên lý điều khiển đối tượng động cơ 1 chiều.
- \* Tìm hiểu thực tế các linh kiện, các loại IC, hoạt động của các loại cảm biến....
- \* Viết chương trình cho vi điều khiển thực hiện thành công theo yêu cầu đề ra.
- \* Tìm hiểu các hướng phát triển của đề tài, nâng cao chất lượng của hệ thống.
- \* chi phí cho hệ thống không quá cao, phù hợp với yêu cầu kinh tế.

Mạch sử dụng các linh kiện sau:

- Vi điều khiển :AT89c52.
- Bộ cảm biến quang học:encoder (đã được gắn trực tiếp trên động cơ DC)
- Động cơ điện một chiều: 12v
- Led 7seg.
- Nút nhấn.
- Led đơn.
- LM7805.
- Mạch cầu H: L298.
- Ngoài ra còn các phần tử điện khác như:tụ hóa,tụ phân cực,điện trở...

## CHƯƠNG II:SƠ ĐỒ KHỐI

### II.1.sơ đồ khối của mạch



Sơ đồ khối của mạch.

### II.2.nhiệm vụ - chức năng của từng khối.

- ❖ Động cơ: động cơ điện được sử dụng trong mạch là động cơ điện một chiều có điện áp đặt vào tối đa 24V. Trên trục động cơ có gắn một đĩa tròn có khoét các lỗ tròn để cho ánh sáng từ led phát quang có thể đi qua tới con mắt thu quang để có thể đo được tốc độ động cơ. ở đây chúng ta dùng động cơ DC 12V.
- ❖ Encoder : dùng để đo số vòng quay của động cơ và phát hiện chiều quay của động cơ. encoder nó sẽ đo tốc độ động cơ thông qua sự liên lạc, mắt liên lạc của led phát quang và bộ phận thu quang rồi chuyển thành các xung điện áp vuông gửi tới chân ngắt của Vi Xử Lý.

- ❖ Vi Xử Lý: nhận các tín hiệu từ encoder thông qua cơ chế ngắt từ đó căn cứ vào số xung do đó nó sẽ tính toán xử lý để:
  - Đưa ra tốc độ động cơ hiển thị lên led 7 thanh.
  - Điều chế độ rộng xung PWM để điều khiển tốc độ động cơ cho phù hợp với yêu cầu.

⇒ Khối vi xử lý là trái tim là khối óc của hệ thống là phần quan trọng nhất điều khiển mọi hoạt động của mạch.
- ❖ Khối điều khiển: điều khiển hướng của động cơ điện một chiều. trong bài toán này chúng em sử dụng mạch cầu H để điều khiển hướng của động cơ.
- ❖ Khối hiển thị: nhận số liệu về tốc độ động cơ từ vi xử lý rồi hiển thị lên các led 7 thanh theo sự điều khiển của Vi điều khiển.
- ❖ Khối nguồn ổn áp 5V: có chức năng cung cấp điện áp ổn định cho các khối trong mạch. Cụ thể trong mạch ta sử dụng hai nguồn riêng biệt:
  - Nguồn 5V DC dùng để nuôi các IC trong mạch hoạt động tạo ra các tín hiệu xuất ra chuẩn TTL, tránh các trường hợp nhiễu điện áp không đúng với điện áp cấp cho IC => tránh IC không hoạt động, hỏng hóc, chập cháy.
  - Nguồn 12V DC dùng để cung cấp cho động cơ một chiều DC (trong đồ án này sử dụng động cơ một chiều DC 12V.)

## CHƯƠNG III: CƠ SỞ LÝ THUYẾT GIẢI QUYẾT BÀI TOÁN

### III.1. Vi Điều Khiển 8051.

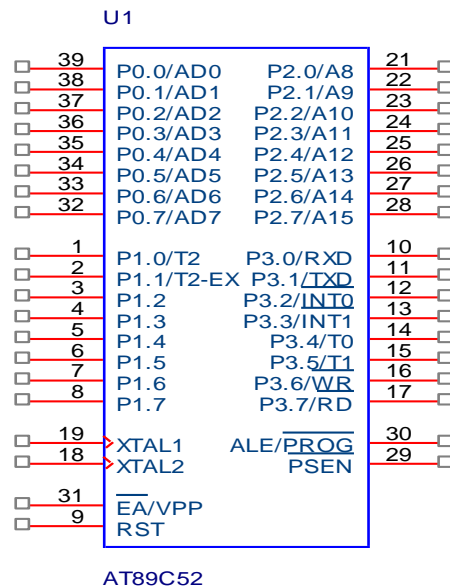
#### III.1.1. Kiến trúc phần cứng AT89C52

AT89C51 là phiên bản 8051 có ROM trên chip là Flash. Phiên bản này thích hợp cho ứng dụng nhanh vì bộ nhớ Flash có thể xóa trong vài giây. AT89C51 có thể được lập trình qua cổng COM của máy tính IBM PC.

Các thành phần bên trong nó gồm có:

- 128 byte RAM
- 8Kbyte ROM
- 32 đường xuất nhập
- 3 bộ định thời đếm 16 bit
- 8 nguyên nhân ngắt
- một port nối tiếp song công
- một mạch dao động và tạo xung clock trên chip

a. Cấu hình các chân của 89ATC52:



b. Mô tả các chân

- Vcc (40): chân cấp điện (5V)

- GND (20): chân đất (0V)

\* Cổng Port 1 được chỉ định là cổng I/O từ chân 1 đến 8. Chúng được sử dụng cho mục đích duy nhất là giao tiếp với thiết bị khi cần thiết. Ngoài ra các chân P1.0, P1.1 là 2 chân liên quan đến hoạt động ngắt của bộ định thời 2.

Trong những mô hình thiết kế không dùng bộ nhớ ngoài, Port 0 là cổng I/O. Còn đối với các hệ thống lớn hơn có yêu cầu một số lượng đáng kể bộ nhớ ngoài thì Port 0 trở thành các đường truyền dữ liệu và 8 bit thấp của bus địa chỉ. Ngoài ra chân P1.0(T2) là ngõ vào của bộ đếm thời gian 2. P1.1(T2EX) là chân capture/reload của bộ đếm thời gian 2.

\* Cổng Port 2 là cổng I/O hoặc là đường truyền 8 bit cao của bus địa chỉ cho những mô hình thiết kế có bộ nhớ chương trình ở năm ngoài học có hơn 256 byte bộ nhớ dữ liệu ngoài .

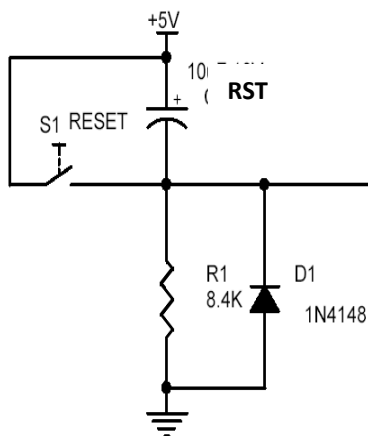
\* Cổng Port 3 ngoài mục đích chung là cổng I/O, những chân này còn kiêm luôn nhiều chức năng khác liên quan đến đặc tính đặc biệt của vi điều khiển.

Bit	Tên	Địa chỉ bit	Chức năng thứ hai
P3.0	RXD	B0H	Nhận dữ liệu cho cổng nối tiếp
P3.1	TXD	B1H	Truyền dữ liệu cho cổng nối tiếp
P3.2	'INT0	B2H	Ngắt 0 bên ngoài
P3.3	'INT1	B3H	Ngắt 1 bên ngoài
P3.4	T0	B4H	Ngõ vào bộ đếm thời gian 0
P3.5	T1	B5H	Ngõ vào bộ đếm thời gian 1
P3.6	'WR	B6H	Tín hiệu điều khiển ghi bộ nhớ dữ liệu ngoài
P3.7	'RD	B7H	Tín hiệu điều khiển đọc bộ nhớ dữ liệu

			ngoài
--	--	--	-------

### Những chức năng thứ hai của chân cổng Port 3

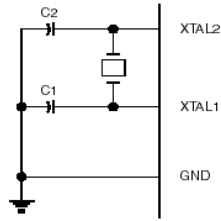
- /PSEN là một tín hiệu điều khiển cho phép bộ nhớ chương trình bên ngoài hoạt động. Nó thường được kết nối đến chân /OE (Output Enable) của /EPROM để đọc các byte chương trình. Xung tín hiệu /PSEN luôn ở mức thấp trong suốt phạm vi quá trình của một lệnh. Còn khi thi hành chương trình từ ROM ở ngay bên trong chip, chân /PSEN luôn ở mức cao.
- Tín hiệu ALE có chức năng đặc biệt tách byte địa chỉ thấp và bus dữ liệu khi cổng P0 được sử dụng ở chế độ tuần tự hay còn gọi là chế độ dồn kênh, nghĩa là sử dụng cùng một đường truyền cho các bit dữ liệu và byte thấp của bus địa chỉ
- Khi chân /EA ở mức cao, vi điều khiển được thực hiện các chương trình lưu trữ ở vùng nhớ thấp hơn 8Kbyte ROM bên trong chip. Còn /EA ở mức thấp chỉ có những chương trình lưu ở bộ nhớ ngoài mới được thực hiện
- AT89S52 có một bộ dao động nội bên trong chip hoạt động theo tần số của một dao động thạch anh nằm bên ngoài. Tần số thông dụng của thạch anh là 11,0592 MHz.
- RST (9): ngõ vào reset ở mức cao trên chân này trong 2 chu kỳ máy.



*Mạch reset tác động bằng tay và sẽ tự động reset lại máy.*

- XTAL1 và XTAL2: là hai ngõ vào và ra của bộ khuếch đại đảo của mạch giao động, được cấu hình dùng để dùng như một bộ giao động trên chip.





Note: C1, C2 = 30 pF  $\pm$  10 pF for Crystals  
= 40 pF  $\pm$  10 pF for Ceramic Resonators

Không có yêu cầu nào về chu kỳ nghiệm vụ của tín hiệu xung Clock bên ngoài do tín hiệu này phải qua mạch flip-flop chia hai trước khi tới mạch tạo xung bên trong. Tuy nhiên các chi tiết kĩ thuật về thời gian mức thấp và thời gian mức cao, điện áp cực đại, điện áp cực tiểu cần được xem xét.

### c. tổ chức bộ nhớ

Không gian bộ nhớ của bộ vi điều khiển được phân chia thành 2 phần: bộ nhớ dữ liệu và bộ nhớ chương trình. Hầu hết các IC MCS đều có bộ nhớ chương trình nằm bên trong chip, tuy nhiên cũng có thể mở rộng dung lượng lên đến 64K bộ nhớ chương trình và 64K dữ liệu bằng cách sử dụng một số bộ nhớ ngoài.

Bên trong chip vi điều khiển AT89C51 có 128 byte bộ nhớ dữ liệu. Không gian bộ nhớ bên trong được chia thành các bank thanh ghi, RAM địa chỉ theo bit, RAM dùng chung và các thanh ghi chức năng đặc biệt.

7F	<b>RAM dùng chung</b>	FF								
		F0	F	F	F	F	F	F	F	B
			7	6	5	4	3	2	1	0
		E0	E	E	E	E	E	E	E	ACC
			7	6	5	4	3	2	1	0
		D0	D	D	D	D	D	D	D	PSW
			7	6	5	4	3	2	1	0
		CD								TH2
		CC								TL2

30									CB									RCA P2L															
									CA									RCA P2H															
									C8																	T2C ON							
									B8																								
2F	7 <i>F</i>	7 <i>E</i>	7 <i>D</i>	7 <i>C</i>	7 <i>B</i>	7 <i>A</i>	7 9	7 8	B0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0	P3															
2E	7 7	7 6	7 5	7 4	7 3	7 2	7 1	7 0	A8	A F	-	-	A C	A B	A A	A 9	A 8	IE															
2 D	6 <i>F</i>	6 <i>E</i>	6 <i>D</i>	6 <i>C</i>	6 <i>B</i>	6 <i>A</i>	6 9	6 8	A0	A 7	A 6	A 5	A 4	A 3	A 2	A 1	A 0	P2															
2C	6 7	6 6	6 5	6 4	6 3	6 2	6 1	6 0	99	SBU F								SCO N															
2B	5 <i>F</i>	5 <i>E</i>	5 <i>D</i>	5 <i>C</i>	5 <i>B</i>	5 <i>A</i>	5 9	5 8	98										9 F	9 E	9 D	9 C	9 B	9 A	9 9	9 8							
2 A	5 7	5 6	5 5	5 4	5 3	5 2	5 1	5 0	90	9 7	9 6	9 5	9 4	9 3	9 2	9 1	9 0	P1															
29	4 <i>F</i>	4 <i>E</i>	4 <i>D</i>	4 <i>C</i>	4 <i>B</i>	4 <i>A</i>	4 9	4 8	8D	TH1								TH0															
28	4 7	4 6	4 5	4 4	4 3	4 2	4 1	4 0	8C										TL1														
27	3 <i>F</i>	3 <i>E</i>	3 <i>D</i>	3 <i>C</i>	3 <i>B</i>	3 <i>A</i>	3 9	3 8	8B	TL0																							
26	3 7	3 6	3 5	3 4	3 3	3 2	3 1	3 0	8A																								

### Bank Register

d.các thanh ghi đặc biệt.

11

- **Thanh ghi trạng thái PSW** (program status word):

Bit	Ký hiệu	Địa chỉ	Mô tả bit
PSW.7	CY	D7H	Cờ nhớ
PSW.6	AC	D6H	Cờ nhớ phụ
PSW.5	FO	D5H	Cờ 0
PSW.4	RS1	D4H	Chọn dãy thanh ghi(bit 1)
PSW.3	RS0	D3H	Chọn dãy thanh ghi(bit 0) 00=bank1: địa chỉ từ 00h đến 07h 01=bank2: địa chỉ từ 08h đến 0Fh 10=bank3: địa chỉ từ 10h đến 17h 01=bank2: địa chỉ từ 18h đến 1Fh
PSW.2	OV	D2H	Cờ tràn
PSW.1	-	D1H	Dự trữ
PSW.0	P	D0H	Cờ kiểm tra chẵn lẻ

- **Thanh ghi B**: thanh ghi luôn được sử dụng kèm theo thanh ghi A để thực hiện các phép toán nhân và chia. Thanh ghi B xem như là thanh ghi đệm dùng chung. Nó có địa chỉ từ F0 đến F7.

- **Con trỏ ngăn xếp**: là một thanh ghi 8 bit, nó chứa địa chỉ của phần dữ liệu đang hiện diện tại đỉnh ngăn xếp. Ngăn xếp hoạt động theo phương thức LIFO. Hoạt động đẩy vào ngăn xếp làm tăng SP lên trước khi ghi dữ liệu vào. Hoạt động lấy ra khỏi ngăn xếp sẽ đọc dữ liệu ra rồi giảm SP.

- **Con trỏ dữ liệu DPTR**(Data Pointer): DPTR được sử dụng để truy cập vào bộ nhớ chương trình và bộ nhớ dữ liệu ngoài, đó là thanh ghi 16 bit có 8 bit thấp ở địa chỉ 82H (DPL) và 8 bit cao ở địa chỉ 83h (DPL)

- **Các thanh ghi cổng:** Các cổng I/O của VDL bao gồm P0 tại địa chỉ 80H, P1 ở địa chỉ 90H, P2 tại địa chỉ A0H, P3 tại địa chỉ B0H. Tất cả các cổng đều có địa chỉ bit nên cung cấp khả năng giao tiếp với bên ngoài rất mạnh

- **Các thanh ghi bộ đếm thời gian:** AT89C52 có 3 bộ đếm thời gian 16 bit để định các khoảng thời gian hay đếm các sự kiện. Timer0 có địa chỉ 8AH (TL0: bit thấp) và 8CH (TH0: byte cao). Timer1 có địa chỉ 8BH (TL1: bit thấp) và 8DH (TH1: byte cao). Timer2 có địa chỉ CCH (TL2: bit thấp) và 8CD (TH2: byte cao). Hoạt động của các bộ đếm thời gian được thiết lập bởi các thanh ghi TMOD, TCON, T2CON. Ngoài ra các thanh ghi RCAP2L, RCAP2H được sử dụng trong chế độ tự nạp của 16 bit bộ định thời 2.

- **Các thanh ghi cổng tuần tự:** IC AT89C52 chứa một cổng nối tiếp để kết nối với các thiết bị nối tiếp như modem hoặc để giao tiếp với các IC khác sử dụng giao tiếp nối tiếp. Bộ đệm dữ liệu nối tiếp SBUF lưu giữ cả dữ liệu truyền đi và dữ liệu nhận được.

- **Các thanh ghi ngắt:** AT89C52 có 6 nguyên nhân ngắt và 2 ngắt ưu tiên. Các ngắt bị cấm sau khi hệ thống khởi động lại và để được bật bằng cách ghi vào thanh ghi cho phép ngắt IE. Mức ưu tiên được thiết lập thông qua thanh ghi ưu tiên IP.

- **Thanh ghi điều khiển năng lượng PCON** (Power Control Register): chứa nhiều bit điều khiển đảm bảo các chức năng khác nhau.

### III.1.2. hoạt động của timer.

#### a. các thanh ghi của bộ định thời.

Để truy cập bộ định thời ta sử dụng 11 thanh ghi FSR:

SFR	Mục đích	Địa chỉ	Định địa chỉ bit
TCON	Điều khiển	88H	Có
TMOD	Chọn chế độ	89H	Không
TL0	Byte thấp của bộ định thời 0	8AH	Không
TL1	Byte thấp của bộ định thời 1	8BH	Không
TH0	Byte cao của bộ định thời 0	8CH	Không

TH1	Byte cao của bộ định thời 1	8DH	Không
T2CON	Điều khiển bộ định thời 2	C8H	Có
RCAP2L	Nhận byte thấp của bộ định thời 2	CAH	Không
RCAP2H	Nhận byte cao của bộ định thời 2	CBH	Không
TL2	Byte thấp của bộ định thời 2	CCH	Không
TH2	Byte cao của bộ định thời 2	CDH	Không

- Thanh ghi TMOD (Timer Moder Register):

Bit	Tên	Timer	Chức năng
7	GATE	1	Khi bit Gate=1 và 'INT1 cao thì Timer 1 mới hoạt động
6	C/'T	1	Bit chọn counter/timer (1/0)
5	M1	1	Bit mode 1
4	M0	1	Bit mode 0
3	GATE	0	Bit GATE của timer 0
2	C/'T	0	Bit chọn counter/timer (1/0) Timer 0
1	M1	0	Bit mode 1 của Timer 0
0	M0	0	Bit mode 0 của Timer 0

Các bit địa chỉ của thanh ghi TMOD:

Thanh ghi TMOD được chia thành 2 nhóm 4 bit dùng để truy cập các chế độ hoạt động của Timer0 và Timer1.

Các chế độ hoạt động của bộ định thời:

- + M1=0, M0=0: Mode 0 (Chế độ định thời 13-bit)
- + M1=0, M0=1: Mode 1 (Chế độ định thời 16 bit)
- + M1=1, M0=0: Mode 2 (Chế độ tự động nạp 8 bit)
- + M1=1, M0=1: Mode 3 (Chế độ định thời chia xẻ).

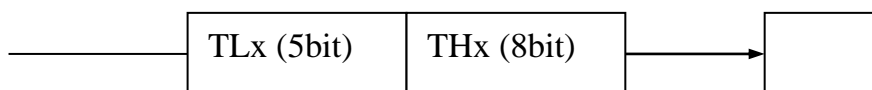
- Thanh ghi điều khiển bộ định thời TCON (Timer control register):

Bit	Ký hiệu	Địa chỉ bit	Mô tả
TCON.7	TF1	8FH	Cờ tràn bộ định thời
TCON.6	TR1	8EH	Bit điều khiển hoạt động của bộ định thời 1
TCON.5	TF0	8DH	Cờ tràn bộ định thời 0
TCON.4	TR0	8CH	Bit điều khiển hoạt động của bộ định thời 1
TCON.3	IE1	8BH	Cờ ngắt bên ngoài 1
TCON.2	IT1	8AH	Cờ ngắt bên ngoài 1
TCON.1	IE0	89H	Cờ ngắt bên ngoài 0
TCON.0	IT0	88H	Cờ ngắt bên ngoài 0

c. các chế độ của timer và cờ tràn

❖ Chế độ định thời 13 bit (mode 0):

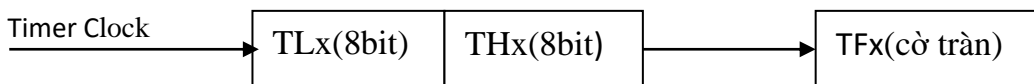
Mode 0 ít được sử dụng trong các hệ thống mới. Byte cao của bộ định thời THx được kết hợp với 5 bit có trọng số nhỏ nhất của byte thấp của bộ định thời TLx để tạo nên bộ định thời 13 bit. #bit còn lại của TLx không được sử dụng.



❖ Chế độ định thời 16 bit (mode1):

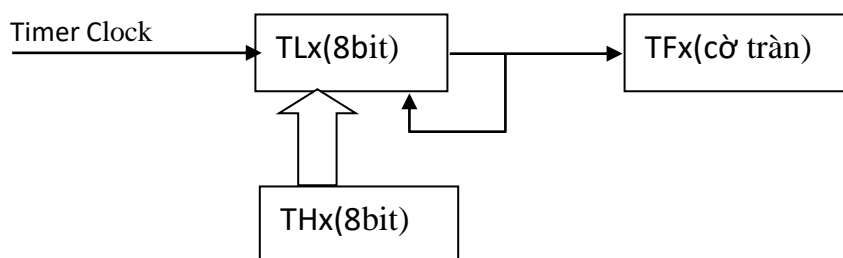
Trong Mode 1, tín hiệu đồng hồ được đưa vào cả 2 byte cao và thấp của bộ định thời (TLx, THx). Khi nhận xung đồng hồ, bộ định thời bắt đầu đếm lên từ

0000H. Hiện tượng tràn xảy ra khi có chuyển tiếp từ FFFFH về 0000H và làm bật cờ tràn.



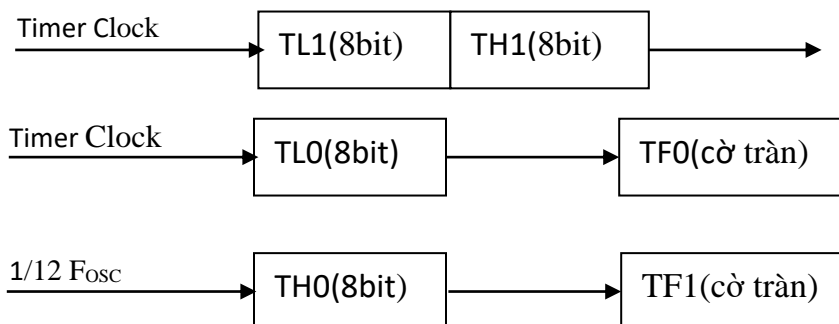
❖ Chế độ định thời 8 bit tự động nạp lại (mode 2):

Trong Mode 2, thanh ghi định thời TLx hoạt động như là bộ định thời 8 bit trong khi byte cao của bộ định thời lưu giá trị nạp lại. Khi quá trình đếm ở TLx bị tràn từ FFH về 00H thì không những cờ tràn bật lên mà giá trị tổng THx được nạp vào TLx, và tiếp tục quá trình đếm từ giá trị này tới khi xảy ra sự chuyển đổi tiếp theo từ FFH về 00H.



❖ Chế độ định thời phân chia (mode 3):

Timer 0 trong mode 3 được chia thành 2 bộ định thời 8b bit. TL0 và TH0 hoạt động như 2 bộ định thời riêng rẽ và sử dụng các cờ tràn tương ứng là TF0, TF1. Timer 1 trong mode 3 ngừng làm việc nhưng có thể hoạt động bằng cách chuyển nó sang một trong mode khác. Điều hạn chế duy nhất là cờ tràn của Timer mode không bị ảnh hưởng khi xảy ra tràn Timer1, bởi vì nó được nối đến TH0



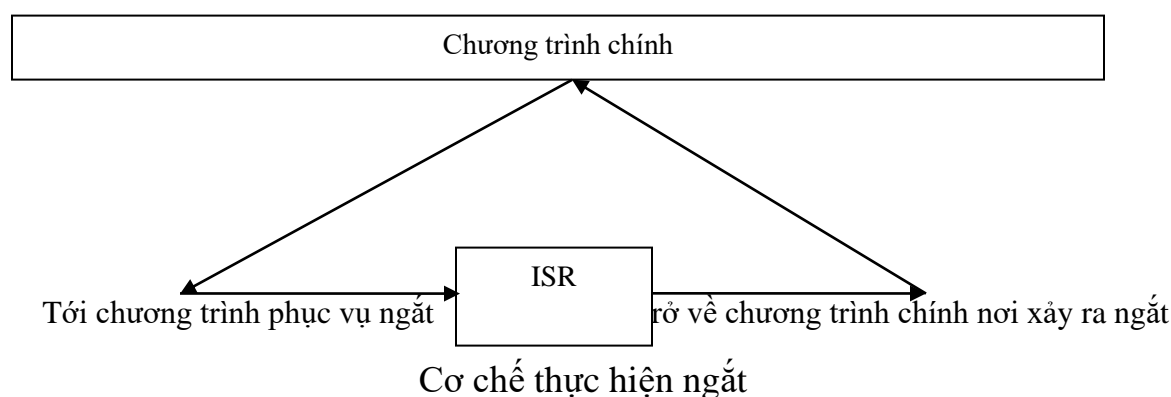


### III.1.4.hệ thống ngắt.

#### a.giới thiệu chung.

Ngắt đóng vai trò trong việc thiết kế và thực hiện các ứng dụng của vi điều khiển. Chúng cho phép hệ thống đáp ứng một cách không đồng bộ đến một sự kiện và giải quyết sự kiện đó khi chương trình khác đó khi chương trình khác đang chạy.

Chương trình giải quyết yêu cầu của một ngắt gọi là thủ tục phục vụ ngắt ISR. ISR dùng để đáp ứng lại một ngắt và thường là thực hiện các hoạt động vào ra đối với một thiết bị vào ra nối với vi điều khiển. Khi xảy ra một ngắt chương trình chính tạm dừng công việc đang thi hành và rẽ nhánh sang ISR, tiếp theo ISR hoạt động để đáp ứng yêu cầu của ngắt và nó sẽ kết thúc bằng lệnh quay trở về, chương trình chính sẽ hoạt động tiếp tục ngay sau điểm rẽ nhánh. Chương trình chính thực hiện ở mức cơ bản còn ISR thực hiện ở mức ngắt.



#### b.tổ chức ngắt.

AT89C51 có tất cả 6 nguyên nhân ngắt: hai ngắt do bên ngoài, ba ngắt do bộ định thời, một ngắt do port nối tiếp. Tất cả các ngắt đều bị cấm sau khi hệ thống khởi động (reset) sau đó chúng được cho phép bằng phần mềm.

#### c.độ ưu tiên ngắt.

Mỗi một nguồn ngắt có thể được lập trình để đạt được một trong 2 mức ưu tiên thông qua thanh ghi chức năng đặc biệt có địa chỉ bit IP tại 0B8H. Thanh ghi

IP bị xoá sau khi hệ thống khởi động để đặt các ngắt ở mức ưu tiên thấp hơn so với mặc định. Trong AT89C51 tồn tại 2 mức ưu tiên. Khi một ưu ngắt có mức ưu tiên cao xuất hiện trong một ISR có mức ưu tiên thấp đang thi hành thì ISR đó sẽ bị ngừng lại, ISR có mức ưu tiên cao hơn sẽ được thực hiện. Nếu 2 ngắt có mức ưu tiên khác nhau xảy ra cùng một lúc thì ngắt có mức ưu tiên cao hơn sẽ được phục trước:

EA	ET2	ES	ET1	EX1	EX0	ET0
Điều khiển các nguồn ngắt						
IE		(0: không cho phép; 1: cho phép)				
IE.7	EA	Cho phép/ không cho phép toàn cục				
IE.6	---	Không sử dụng				
IE.5	ET2	Cho phép ngắt do bộ định thời 2				
IE.4	ES	Cho phép ngắt do port nối tiếp				
IE.3	ET1	Cho phép ngắt cho bộ định thời 1				
IE.2	EX1	Cho phép ngắt từ bên ngoài (ngắt ngoài 1)				
IE.1	EX0	Cho phép ngắt từ bên ngoài (ngắt ngoài 0)				
IE.0	ET0	Cho phép ngắt do bộ định thời 0				

Thanh ghi IE.

Các bit trong thanh ghi IP (thanh ghi điều khiển ưu tiên ngắt):

Bit	Ký hiệu	Địa chỉ bit	Mô tả
IP.7	-	-	Không định nghĩa
IP.6	-	-	Không định nghĩa
IP.5	PY2	BDH	Ưu tiên cho ngắt Timer 2
IP.4	PS	BCH	Ưu tiên cho ngắt cổng nối tiếp
IP.3	PT1	BBH	Ưu tiên cho ngắt Timer 1
IP.2	PX1	BAH	Ưu tiên cho ngắt ngoài 1
IP.1	PT0	B9H	Ưu tiên cho ngắt Timer 0
IP.0	PX0	B8H	Ưu tiên cho ngắt ngoài 0

d.cơ chế lựa chọn tuần tự.

Nếu có 2 ngắt cùng mức ưu tiên xảy ra đồng thời, một cơ chế chọn lựa theo thứ tự có sẵn sẽ xác định ngắt nào được đáp ứng trước. Việc chọn lựa theo thứ tự là: External 0, Timer 0, External 1, Timer 1, Serial Port, Timer 2.

Quá trình xử lý ngắt:

Khi một ngắt xuất hiện và nó được CPU chấp nhận, chương trình chính bị ngừng, các hoạt động tiếp theo xảy ra như sau:

- + Thực hiện xong lệnh hiện hành đó
- + Bộ đếm chương trình PC được lưu vào trong Stack
- + Lưu giữ tình trạng của ngắt hiện tại
- + Các nguồn ngắt được giữ tại mức của ngắt hiện tại
- + Nạp vào PC địa chỉ Vector của ISR
- + ISR thực hiện

ISR hoạt động để đáp ứng lại yêu cầu ngắt. ISR kết thúc bằng lệnh RETI có tác dụng quay trở về chương trình chính, lệnh này sẽ nạp lại giá trị cũ của PC trong ngăn xếp và khôi phục tình trạng của ngắt cũ. Việc thực hiện chương trình chính tiếp tục diễn ra tại nơi nó tạm dừng.

e.vector ngắt.

Khi một ngắt được chấp nhận thì giá trị nạp vào PC gọi là vector ngắt. Nó chính là địa chỉ bắt đầu của ISR tương ứng với ngắt được chấp nhận.

\* Hàm ngắt:

```
Void tenhamngat(void) interrupt nguồn ngắt
{
    // Chương trình ngắt ở đây
}
```

\* Chú ý về hàm ngắt.

- Hàm ngắt không được trả lại hay truyền biến vào hàm
- Tên hàm bất kỳ
- Interrupt là hàm ngắt phải phân biệt với hàm khác
- Nguồn ngắt từ 0-5 theo bảng vectơ ngắt
- Bảng thanh ghi Ram chọn từ 0-3
- 

Sau đây là bảng các vectơ ngắt

Ngắt	Cờ	Địa chỉ Vector
System reset	RST	0000H
External 0	IE0	0003H
Timer 0	TF0	000BH
External 1	IE1	0013H
Timer 1	TF1	001BH
Serial Port	RI or TI	0023H
Timer 2	TF2 or EXF2	002BH

*Bảng: Các vectơ ngắt.*

## III.2.cảm biến tốc độ (encoder).

### III.2.1.cấu tạo và phân loại encoder.

#### III.2.1.1.phân loại.

Gồm :

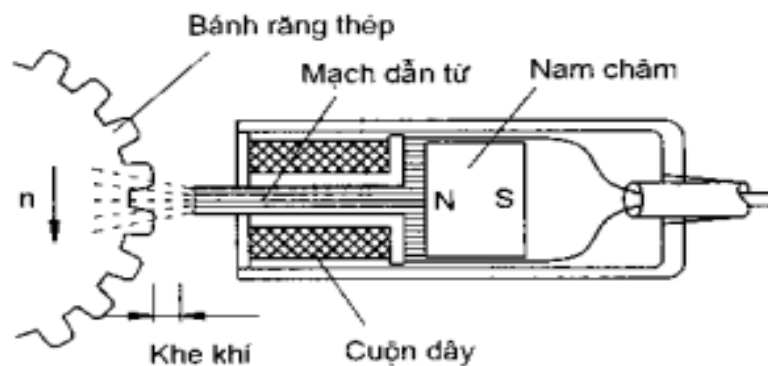
- ❖ encoder kiểu cảm ứng.
- ❖ encoder tương đối(encoder đếm xung).

### III.2.1.2. cấu tạo và nguyên lý hoạt động của một loại encoder.

#### a. encoder kiểu cảm ứng. ( Absolute encoder)

##### ❖ cấu tạo.

- Gồm một nam châm vĩnh cửu N-S trên có một cuộn dây.
- Một bánh răng trên đó có p răng. bánh răng được gắn trên trục quay của động cơ.



##### ❖ nguyên lý hoạt động.

Khi động cơ quay với tốc độ  $n$  (vòng/phút) thì bánh răng cũng quay với tốc độ  $n$  (vòng/phút) → thì từ thông của nam châm vĩnh cửu gửi qua cuộn dây sẽ biến thiên. làm xuất hiện trên hai cuộn dây một sức điện động  $E$  có tần số phụ thuộc vào tốc độ quay của bánh răng.

$$n = 60f/p \text{ hoặc } n=60f/N$$

Trong đó :

$p$ : số răng trên bánh răng hay số lỗ.

$n$ : tốc độ của động cơ (vòng/phút).

$N$ : số xung/vòng của encoder.

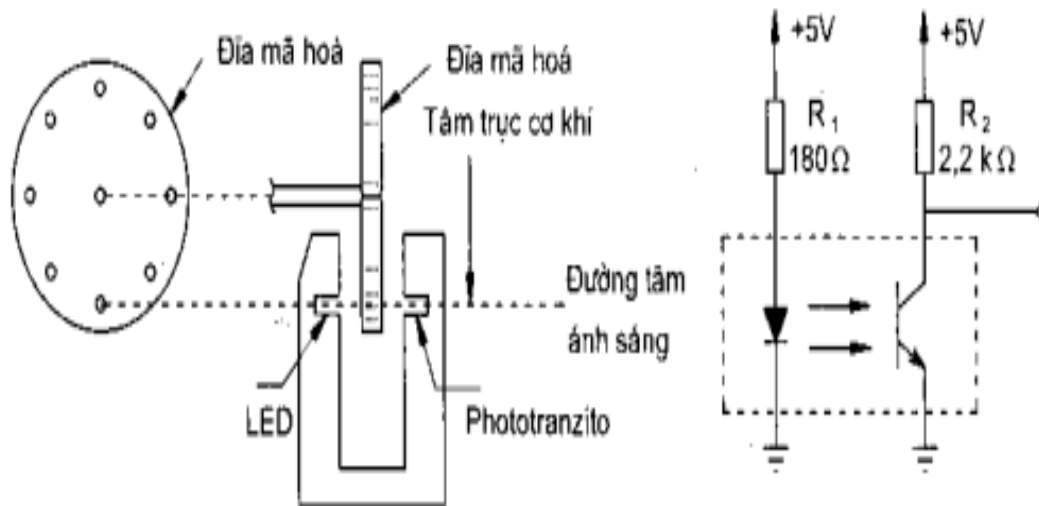
$f$ : tần số của sức từ động tạo ra trên hai đầu cuộn dây.

⇒ Chỉ cần đo được tần số xung  $f$  này ta có thể xác định được tốc độ của động cơ.

b.encoder tương đối (encoder đếm xung).

❖ Cấu tạo.

- Một led hồng ngoại (bộ phát).
- Một transistor quang P.TZT (bộ thu).
- Đĩa mã hóa: nằm giữa led hồng ngoại (bộ phát) và transistor P.TZT (bộ thu).



Cấu tạo trong của encoder tương đối.

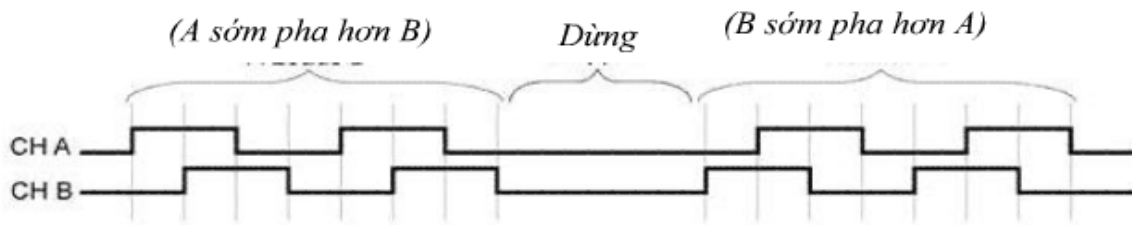
❖ Nguyên tắc hoạt động.

- Để tạo mã xung thì mỗi bộ ENCODER sẽ sử dụng hai led phát và tương ứng là hai bộ tách sóng (hai con mắt thu). hai led được đặt sao cho hai tín hiệu ra có pha vuông góc nhau để xác định chiều quay của đĩa (tương ứng với chiều quay của động cơ).



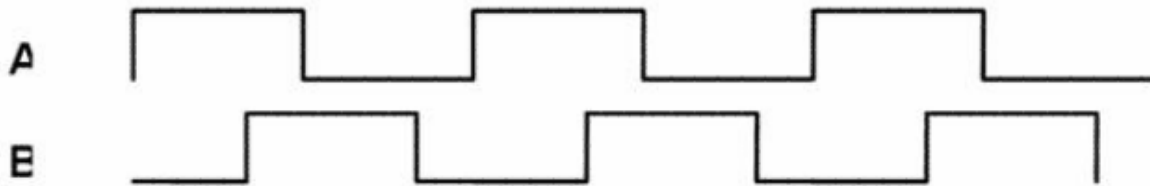
**Hình 3.3.** Vị trí đặt của hai LED phát nguồn sáng tạo tín hiệu cầu phương

- Tốc độ quay của đĩa được xác định nhờ vào tần số của tín hiệu. chiều được xét bằng cách xem tín hiệu nào sớm pha hay muộn pha hơn.

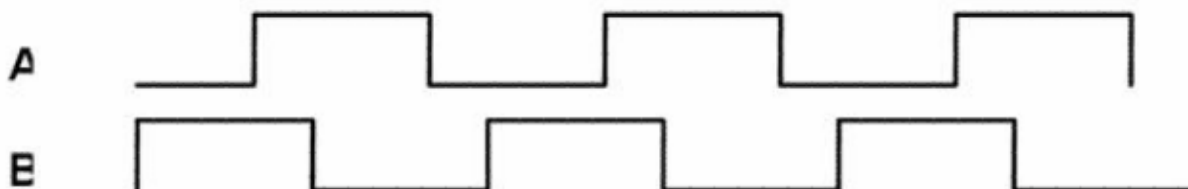


**Hình 3.4.** Dạng tín hiệu cầu phương thu được ở bộ tách sóng quang (photodetector) khi động cơ quay

Cụ thể như trên hình 3.5 và 3.6 sau:



**Hình 3.5.** Dạng tín hiệu lập mã quang khi động cơ có chiều quay phải (cùng chiều kim đồng hồ)



**Hình 3.6.** Dạng tín hiệu lập mã quang khi động cơ có chiều quay trái (ngược chiều kim đồng hồ)

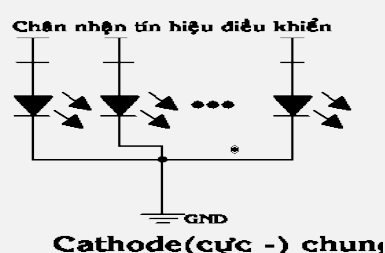
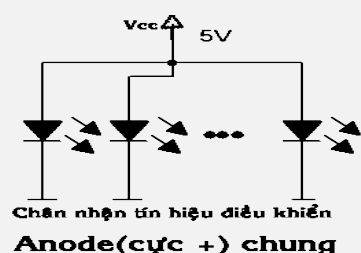
Bộ encoder trong thực tế gắn trên động cơ thường có 4 dây trong đó:

- Dây đỏ: là dây nguồn 5V.
- Dây xanh là :GND
- Dây vàng và dây trắng là hai tín hiệu lệch pha nhau 90 độ dùng để đo tốc độ động cơ.

### III.3.Led 7seg.

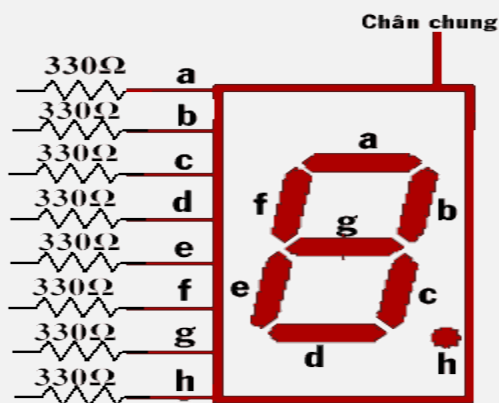
Led 7seg có thể coi là một tập hợp của 7 led đơn ghép lại dưới dạng các thanh tạo ra .có hai loại led 7 thanh là loại catot chung và loại anot chung.

8 led đơn trên led 7 đoạn có Anode(cực +) hoặc Cathode(cực -) được nối chung với nhau vào một điểm, được đưa chân ra ngoài để kết nối với mạch điện. 8 cực còn lại trên mỗi led đơn được đưa thành 8 chân riêng, cũng được đưa ra ngoài để kết nối với mạch điện. Nếu led 7 đoạn có Anode(cực +) chung, đầu chung này được nối với +Vcc, các chân còn lại dùng để điều khiển trạng thái sáng tắt của các led đơn, led chỉ sáng khi tín hiệu đặt vào các chân này ở mức 0. Nếu led 7 đoạn có Cathode(cực -) chung, đầu chung này được nối xuống Ground (hay Mass), các chân còn lại dùng để điều khiển trạng thái sáng tắt của các led đơn, led chỉ sáng khi tín hiệu đặt vào các chân này ở mức 1.



Vì led 7 đoạn chứa bên trong nó các led đơn, do đó khi kết nối cần đảm bảo dòng qua mỗi led đơn trong khoảng 10mA-20mA để bảo vệ led. Nếu kết nối với nguồn 5V có thể hạn dòng bằng điện trở 330Ω trước các chân nhận tín hiệu điều khiển.

Sơ đồ vị trí các chân được bố trí như sau:





Trong đó :các điện trở 330 ôm mắc ở ngoài dùng để hạn dòng qua led 7 thanh khi nó được nối với nguồn điện áp 5V.

Mã led 7 thanh:

- Mã cho led 7 thanh anot chung (các led đơn sáng ở mức logic 0):

<i>Số hiển thị trên led 7 đoạn</i>	<i>Mã hiển thị led 7 đoạn dạng nhị phân</i>	<i>Mã hiển thị led 7 đoạn dạng thập lục phân</i>
	<b>h g f e d c b a</b>	
<b>0</b>	<b>1 1 0 0 0 0 0 0</b>	<b>C0</b>
<b>1</b>	<b>1 1 1 1 1 0 0 1</b>	<b>F9</b>
<b>2</b>	<b>1 0 1 0 0 1 0 0</b>	<b>A4</b>
<b>3</b>	<b>1 0 1 1 0 0 0 0</b>	<b>B0</b>
<b>4</b>	<b>1 0 0 1 1 0 0 1</b>	<b>99</b>
<b>5</b>	<b>1 0 0 1 0 0 1 0</b>	<b>92</b>
<b>6</b>	<b>1 1 0 0 0 0 1 0</b>	<b>82</b>
<b>7</b>	<b>1 1 1 1 1 0 0 0</b>	<b>F8</b>
<b>8</b>	<b>1 0 0 0 0 0 0 0</b>	<b>80</b>
<b>9</b>	<b>1 0 0 1 0 0 0 0</b>	<b>90</b>
<b>A</b>	<b>1 0 0 0 1 0 0 0</b>	<b>88</b>
<b>B</b>	<b>1 0 0 0 0 0 1 1</b>	<b>83</b>
<b>C</b>	<b>1 1 0 0 0 1 1 0</b>	<b>C6</b>
<b>D</b>	<b>1 0 1 0 0 0 0 1</b>	<b>A1</b>
<b>E</b>	<b>1 0 0 0 0 1 1 0</b>	<b>86</b>
<b>F</b>	<b>1 0 0 0 1 1 1 0</b>	<b>8E</b>
<b>-</b>	<b>1 0 1 1 1 1 1 1</b>	<b>BF</b>

- Mã led 7 thanh cato chung (các led sáng ở mức logic 1):

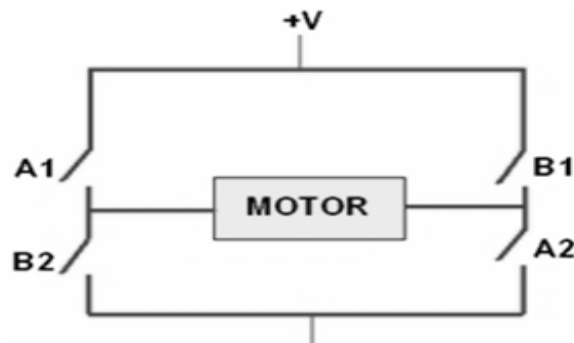
<i>Số hiển thị trên led 7 đoạn</i>	<i>Mã hiển thị led 7 đoạn dạng nhị phân</i>	<i>Mã hiển thị led 7 đoạn dạng thập lục phân</i>
<b>0</b>	<b>0 0 1 1 1 1 1 1</b>	<b>3F</b>
<b>1</b>	<b>0 0 0 0 0 1 1 0</b>	<b>06</b>
<b>2</b>	<b>0 1 0 1 1 0 1 1</b>	<b>5B</b>
<b>3</b>	<b>0 1 0 0 1 1 1 1</b>	<b>4F</b>
<b>4</b>	<b>0 1 1 0 0 1 1 0</b>	<b>66</b>

5	0 1 1 0 1 1 0 1	6D
6	0 1 1 1 1 1 0 1	7D
7	0 0 0 0 0 1 1 1	07
8	0 1 1 1 1 1 1 1	7F
9	0 1 1 0 1 1 1 1	6F
A	0 1 1 1 0 1 1 1	77
B	0 1 1 1 1 1 0 0	7C
C	0 0 1 1 1 0 0 1	39
D	0 1 0 1 1 1 1 0	5E
E	0 1 1 1 1 0 0 1	79
F	0 1 1 1 0 0 0 1	71
-	0 1 0 0 0 0 0 0	40

### III.4.mạch cầu H.

#### III.4.1.nguyên lý hoạt động của mạch cầu H

Một mạch cầu H đơn giản có dạng như sơ đồ sau:



Theo sơ đồ trên ta mạch cầu H gồm có:

- Dây cấp nguồn +V.
- Dây cấp mass GND.
- 4 khóa đóng mở.
  - Khi khóa A1 và A2 đóng thì dòng chạy từ nguồn +V qua A1,MOTOR,A2 => động cơ chạy thuận.

- Khi khóa B1 và B2 đóng thì dòng chạy từ nguồn +V qua B1, MOTOR, B2 => động cơ quay ngược.

Các khóa này có thể đóng mở được bất cứ lúc nào. Chúng ta có 4 khóa vậy sẽ có 16 trạng thái. Tuy nhiên chỉ có 4 trạng thái là được sử dụng. Những trạng thái còn lại sẽ không hoạt động và nếu hoạt động sẽ gây cháy nổ. Trong quá trình điều khiển chúng ta phải luôn tránh các trạng thái không mong muốn. Cách thức hoạt động được tóm tắt như bảng sau.

A1	B1	A2	B2	Trạng thái của động cơ
1	0	1	0	Quay thuận
0	1	0	1	Quay nghịch
1	1	0	0	Hãm động cơ
0	0	1	1	Hãm động cơ
1	0	0	1	Chập điện
0	1	1	0	Chập điện

Ở đây ta xét:  $A, B_x = 1$  là đóng công tắc.  $A, B_x = 0$  là mở công tắc.

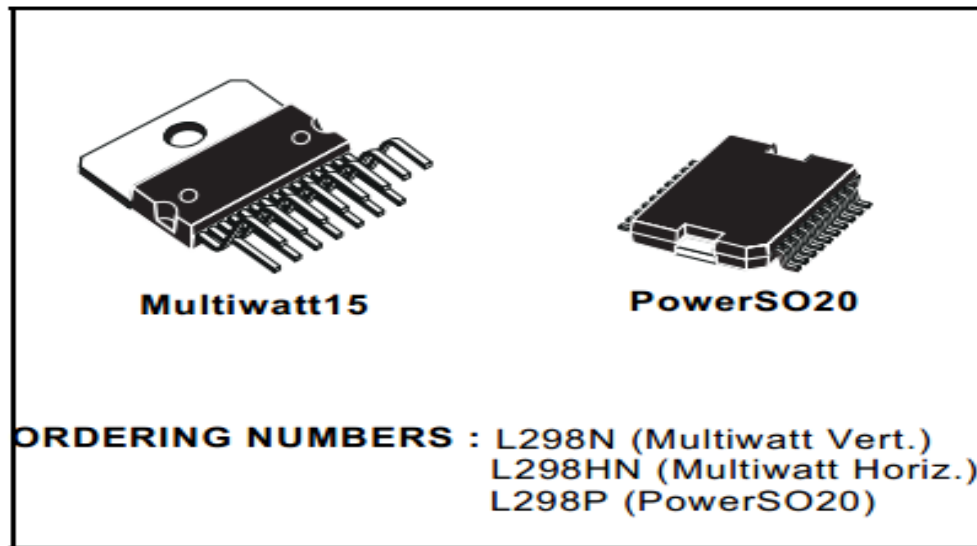
Từ bảng trên ta nhận thấy, cầu H chỉ dùng với 4 trạng thái đầu tiên. Vì vậy khi sử dụng cần thiết phải tránh các trạng thái không mong muốn.

Mạch cầu H chúng ta dùng để điều khiển chiều quay của động cơ. Có rất nhiều loại mạch cầu H khác nhau như: mạch cầu H bán dẫn, mạch cầu H tích hợp, mạch cầu H dùng relay ...

### III.4.2. giới thiệu về L298 (IC mạch cầu H).


IC L298 là mạch tích hợp đơn chip có kiểu vỏ công suất 15 chân (multiwatt 15) và POWERSO20 (linh kiện dán công suất). IC L298 là một mạch cầu đôi (dual full- bridge) có khả năng hoạt động ở điện áp cao, dòng cao.

- Điện áp cấp lên tới 46V
- Tổng dòng một chiều chịu được tới 4A
- Điện áp bảo hòa
- Chức năng bảo vệ quá nhiệt
- Logic 0 từ 1.5V trở xuống

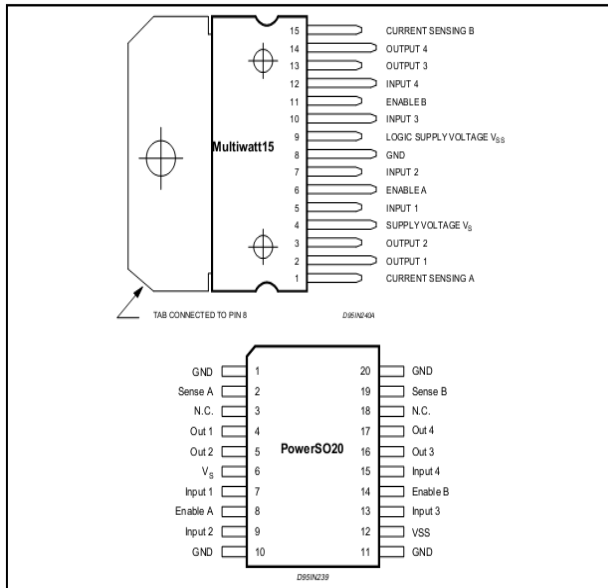


Dạng đóng vỏ của L298.

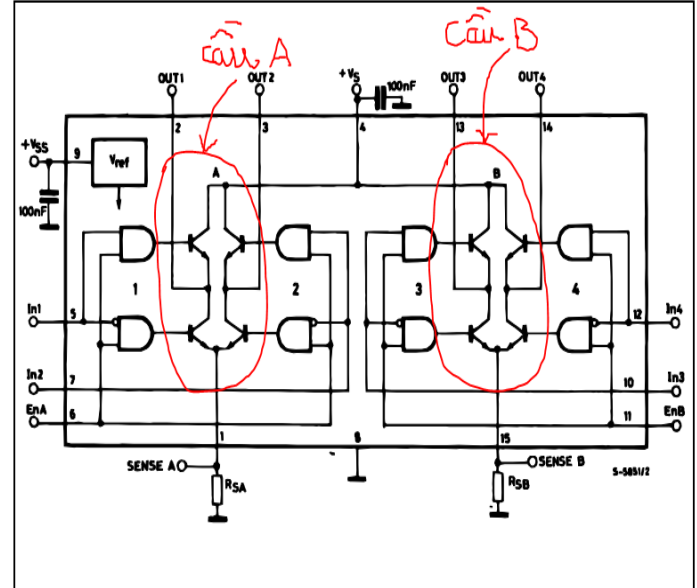
❖ Các giá trị đặc trưng:

Ký hiệu	Đặc trưng 	Giá trị	Đơn vị
$V_s$	Nguồn cung cấp	50	V
$V_{ss}$	Điện thế cấp cho mạch Logic	7	V
$V_I, V_{en}$	Điện thế vào cho phép	-0.3 đến 7	V
$I_0$	Dòng ra cực đại (mỗi kênh)		
	- Không lặp lại ( $t = 100 \mu s$ )	3	A
	- Lặp lại (80% mở - 20% ngắt, $t_{mở}=10ms$ )	2.5	A
	- Dòng hoạt động một chiều	2	A
$V_{sens}$	Điện thế cảm biến	-1 đến 2.3	V
$P_{tot}$	Tổng năng lượng suy hao ( $T_{vỏ} = 75^{\circ}C$ )	25	W
$T_{op}$	Nhiệt độ mặt tiếp giáp hoạt động	-25 đến 130	$^{\circ}C$
$T_{stg}, T_j$	Nhiệt độ bộ tích trữ và mặt tiếp giáp	-40 đến 150	$^{\circ}C$

❖ Sơ đồ chân và cấu trúc bên trong:



sơ đồ chân



cấu trúc bên trong

❖ Chức năng các chân.

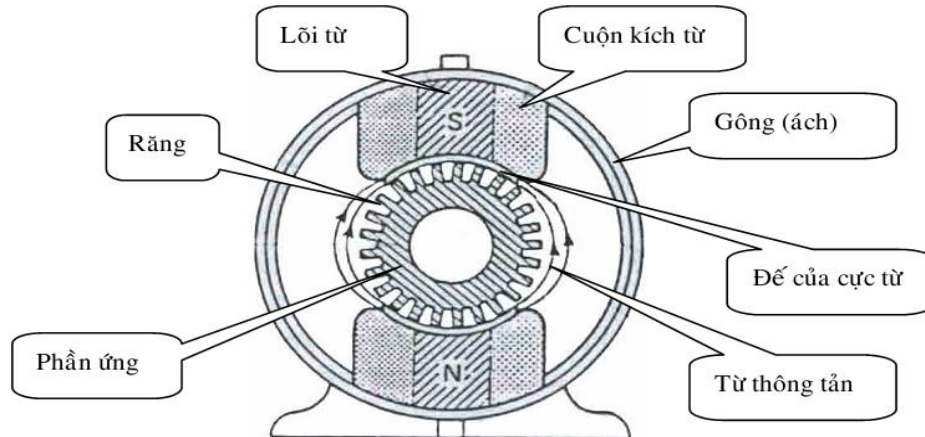
MW.15	powerSO	Tên	Chức năng
1,15	2,19	senseA ,senseB	Chân này qua điện trở cảm ứng dòng xuống GND để điều khiển dòng tải
2,3 4	4,5 6	Out 1,out 2 VSS	Ngõ ra của cầu A,dòng của tải mắc giữa hai chân này được quy định bởi chân 1. Chân cấp nguồn cho tầng công suất.cần có một tụ điện 100nF nối giữa chân này với chân GND
5,7	7,9	Input1, Input 2	Chân ngõ vào của cầu A,tương thích chuẩn TTL.
6,11	8,14	enableA enableB	Chân ngõ vào enable(cho phép) tương thích chuẩn TTL.mức thấp ở chân này sẽ cấm ngõ ra cầu A (đối với chân enableA)hoặc cầu B (với enableB).
8	1,10,11,20	GND	Chân đất (ground)
9	12	VSS	Cấp nguồn cho khối logic,cần 1 tụ 100nF nối giữa chân này với GND.
10,12	13,15	Input 3	Các chân ngõ vào cầu B

		,input 4	
13,14	15,17	Out 3, Out 4	Ngõ ra của cầu B, dòng của tải mắc giữa hai chân này được quy định bởi chân 15.
-	3,18	N.C	Không kết nối (bỏ ngõ)

### III.5. động cơ điện một chiều DC.



Hình ảnh thực tế của động cơ DC và encoder



Cấu tạo bên trong của động cơ điện một chiều.

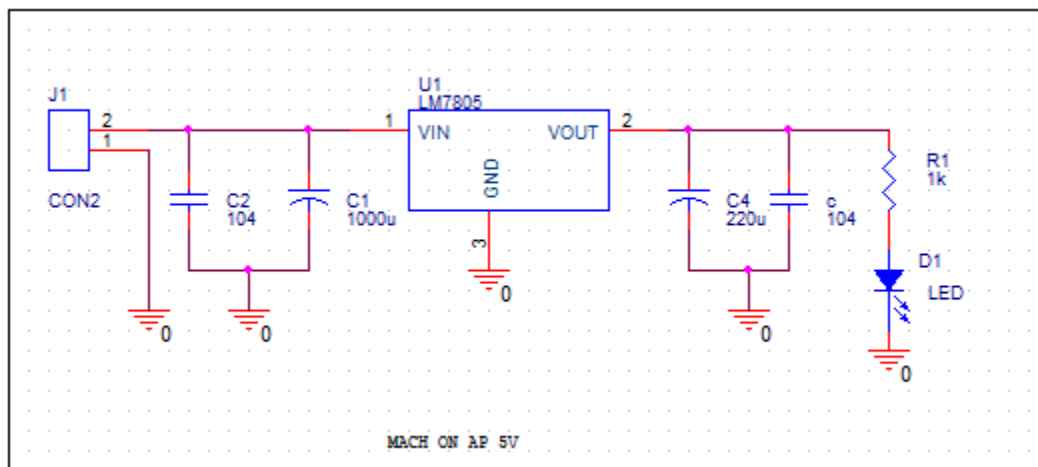
Động cơ điện một chiều là động cơ điện hoạt động với dòng điện một chiều. Cấu tạo của động cơ gồm có 2 phần: stato đứng yên và rôto quay so với stato. Phần cảm (phần kích từ-thường đặt trên stato) tạo ra từ trường đi trong mạch từ, xuyên qua các vòng dây quấn của phần ứng (thường đặt trên rôto). Khi có dòng điện chạy trong mạch phần ứng, các thanh dẫn phần ứng sẽ chịu tác động bởi các lực điện từ theo phương tiếp tuyến với mặt trụ rôto, làm cho rôto quay. Chính xác hơn, lực

điện từ trên một đơn vị chiều dài thanh dẫn là tích có hướng của vectơ mật độ từ thông  $B$  và vectơ cường độ dòng điện  $I$ . Dòng điện phản ứng được đưa vào rôto thông qua hệ thống chổi than và cổ góp. Cổ góp sẽ giúp cho dòng điện trong mỗi thanh dẫn phản ứng được đổi chiều khi thanh dẫn đi đến một cực từ khác tên với cực từ mà nó vừa đi qua (điều này làm cho lực điện từ được sinh ra luôn luôn tạo ra mômen theo một chiều nhất định).

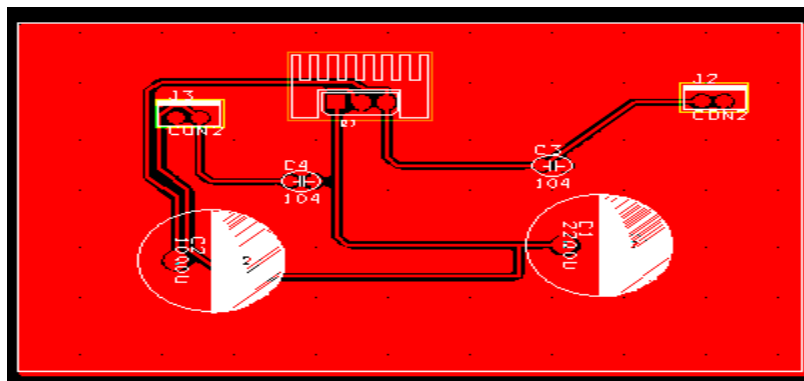
## CHƯƠNG IV: THIẾT KẾ VÀ GIA CÔNG PHẦN CỨNG

### IV.1. mạch nguồn nuôi.

Trong thực tế ngày nay đối với các vi điều khiển, IC... người ta đa số dùng mức tương thích TTL hơn là CMOS. mức TTL thì IC TTL dùng được và IC CMOS cũng dùng được (các IC có dải điện áp 3:8v). Mức TTL có chuẩn dương 5V và chuẩn âm 0 V. các chip vi điều khiển, IC TTL... đòi hỏi phải có nguồn cung cấp ổn định 5V (giao động từ 4.75 tới 5.25) nếu điện áp không nằm trong dải đó mà xuống thấp hơn thì IC không hoạt động, cao hơn thì IC cháy hỏng. Do đó trong các thiết kế mạch sử dụng IC phải cung cấp cho chúng một nguồn nuôi ổn định không lên xuống theo điện áp nguồn cấp vào.



Mạch nguồn ổn áp 5v.



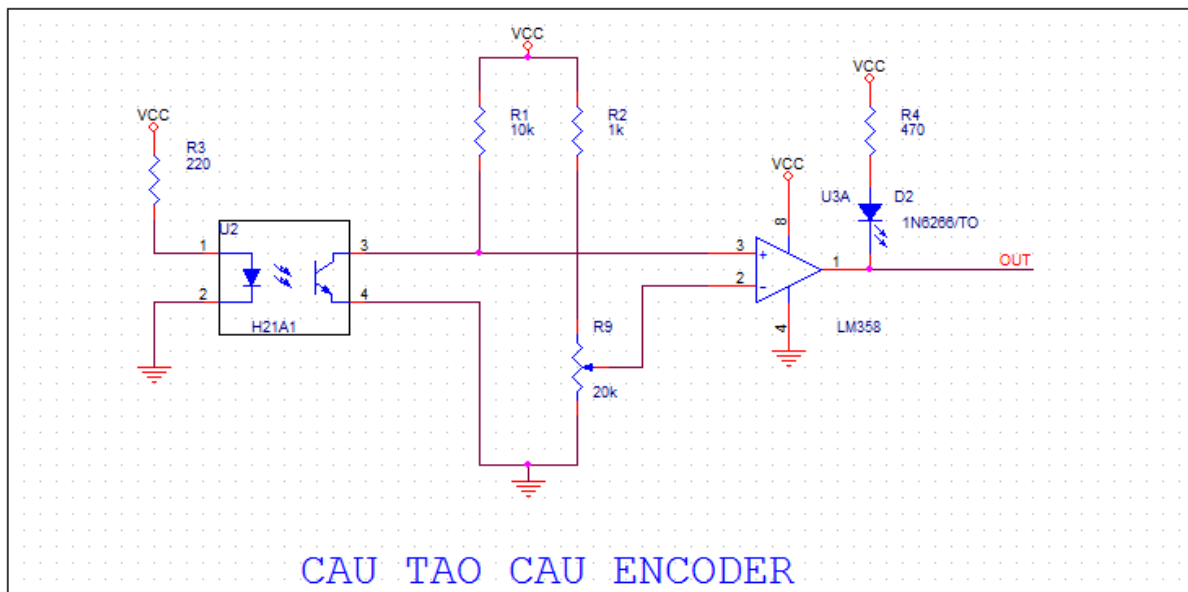


## Mạch in của khối nguồn.

Mạch này sử dụng:

- Một IC ổn áp LM 7805.
- Một led đơn : báo trạng thái có nguồn.
- 4 tụ: chống nhiễu và san phẳng điện áp

## IV.2.encoder

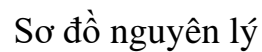


Khi tia sáng từ led hồng ngoại chiếu vào chân B của transistor quang làm nội trở Rce của transistor giảm → mức 0 được đưa vào chân dương (+) của OPAM lúc này đầu ra của bộ khuếch đại thuật toán là 0.

Đĩa mã hóa được gắn trên trục động cơ dẫn tới khi động cơ quay có lúc đĩa sẽ chắn tia sáng chiếu từ led quang tới chân B của transistor làm cho nội trở Rce của transistor tăng lên đưa điện áp dương (5V) vào chân dương (+) của OPAM làm cho đầu ra của bộ khuếch đại là mức dương.

Biến trở: R9 tạo điện áp so sánh.

### IV.3.mạch VĐK 89c51.



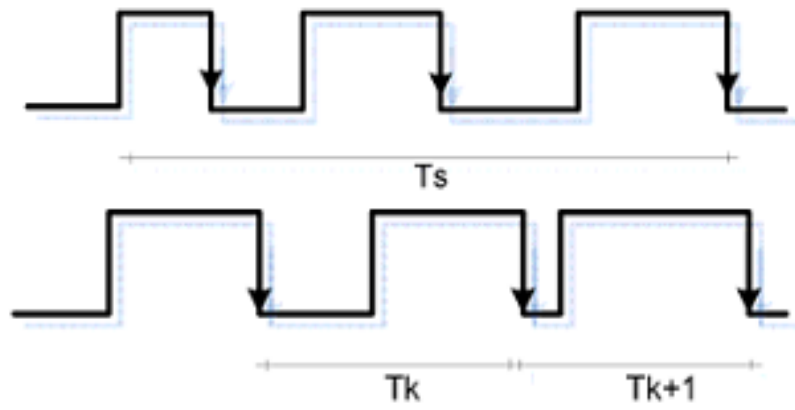
## CHƯƠNG V: TÍNH TOÁN THAM SỐ VÀ THIẾT KẾ PHẦN MỀM

### V.1. tính toán tham số.

#### V.1.1. tính toán giá trị nạp vào cho timer0 và timer1.

##### a. Phân tích lựa chọn phương án đo tốc độ bằng encoder:

- ❖ phương án 1: Tính số cạnh xuống trong khoảng thời gian  $T_s$  (sampling time) để suy ra vận tốc trung bình của động cơ (Pulse/ $T_s$ ).
- ❖ phương án 2: Tìm thời gian suất hiện hai cạnh xuống liên tiếp của encoder từ đó có thể suy ra vận tốc trung bình của động cơ.



##### b. Phân tích lựa chọn phương án:

- Phương án 1:
  - Sử dụng phương án này ta cần một timer để định thời gian lấy mẫu và một counter để đếm số xung.
  - Nếu sử dụng phương án 1 sẽ bị giới hạn về mặt đáp ứng do encoder gây ra (số xung quá lớn, tràn counter/hoặc encoder có độ phân giải thấp thì thời gian lấy mẫu  $T_s$  phải lớn để giảm thiểu sai số đo đạc)
- Phương án 2: Nếu dùng phương án này

- ta cũng cần 1 counter để định thời gian và 1 chân (có thể là counter/ngắt ngoài/1 chân in/out bình thường).
- thời gian thực thi chiếm hầu hết thời gian hoạt động của MCU nên sẽ dùng một thiết bị (device) riêng biệt làm nhiệm vụ này.
- Trong thuật giải của MCU MASTER, dùng giao tiếp song song để giảm thiểu thời gian lấy mẫu, tất nhiên cần giao thức bắt tay để đảm bảo nhận đúng dữ liệu, chiều quay của động cơ cũng được ENCODER READER CARD nhận và gửi lên.

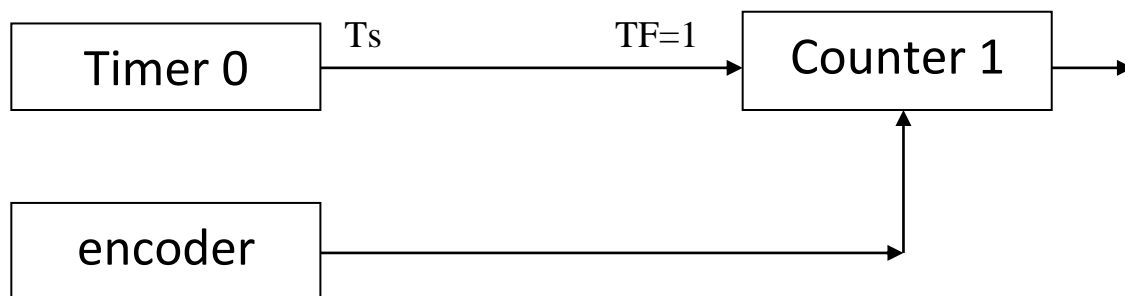
### c.lựa chọn phương án

Trong đề tài này chúng em chọn phương án 1 vì :

- Đơn giản
- Tốn ít thiết bị hơn phương án 2
- Giải thuật đơn giản.

### d.tính toán với phương án đã chọn

Để đo tốc độ động cơ người ta sẽ làm như sau:

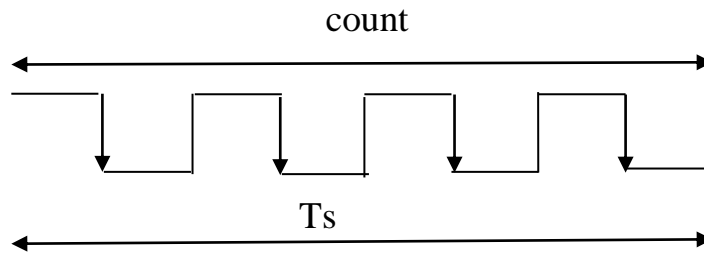


Hình 1.sơ đồ đọc tốc độ từ encoder.

- Sử dụng Timer 1 ở chế độ 1 dùng làm counter đếm số xung cạnh của encoder.
- Sử dụng timer 0 để định thời gian ngắt ( $T_s$ ) thời gian này được lập trình để có thể thay đổi tùy vào mục đích, ý đồ của người sử dụng. nhưng chúng ta nên chọn  $T_s$  sao cho việc tính toán tốc độ động cơ là đơn giản nhất:

Ta có:

Giả sử trong  $T_s$  (được xác định bởi bộ định thời timer0) thì timer1 ở chế độ count1 đếm được count xung được minh họa như hình vẽ dưới:



Thì ta có chu kỳ của xung là:  $T = \frac{T_s}{count}$  (s) nên tần số xung là  $f = 1/T$

Suy ra: có tốc độ động cơ là:  $n = \frac{60f}{N} = \frac{60 \cdot count}{N \cdot T_s}$  (vong/giay)

Để cho việc tính toán tốc độ động cơ được dễ dàng chúng ta nên chọn:  $\frac{60}{N \cdot T_s} = 1$  khi đó tốc độ động cơ chỉ đơn giản là:

$$n = count \text{ (vong/giay).}$$

Vậy là bây giờ chúng ta chỉ việc hiển thị con số count lên thiết bị hiển thị (trong đồ án này thì thiết bị hiển thị là led 7seg)

Trong bài này chúng em dùng encoder  $N = 100 \text{ xung/vong}$ . nên từ biểu thức trên ta có:  $\frac{60}{100 \cdot T_s} = 1 \rightarrow T_s = 0.6 \text{ (s)}$ .

Tần số thạch anh của 8051 là 12MHz  $\Rightarrow$  mỗi nhịp xung đồng hồ có:

$$T_0 = 12/f = 1 \mu s$$

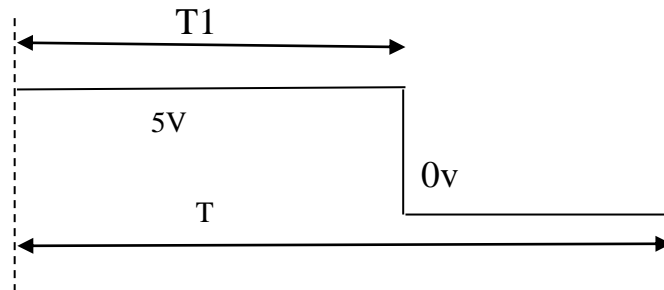
Suy ra số nhịp cần cho VĐK (có tần số thạch anh  $f = 12 \text{ MHz}$ ) trong thời gian 0.6(s) là:  $\frac{0.6}{10^{-6}} = 600000 = 12.50000 > 65536$  do đó ta nạp cho các thanh ghi TH0 và

TL0 của vi điều khiển giá trị:  $65536 - 50000 = 15536 \text{ D} = 3C0B \text{ H}$

$\Rightarrow$  TH0=0x3C và TL0=0xB0. Và cần cho VĐK chạy lặp lại 12 lần để đảm bảo  $T = 12.500000 \cdot 10^{-6} = 0.6 \text{ (s)}$ .

V.1.2.tính toán giá trị nạp các thanh ghi của cho timer2 để thực hiện PWM.

Tạo một xung có tần số  $f=1\text{kHz} \rightarrow T=1/1000=0.001\text{s}=1000\mu\text{s}=1000$  chu kỳ máy.chúng ta tạo PWM với 10 cấp điều chỉnh tức là phải tạo được xung 10%,20%,30%....100%.một xung như sau:



Trong đó :T1 là khoảng thời gian xung ở mức cao 5V

T là chu kỳ của xung.

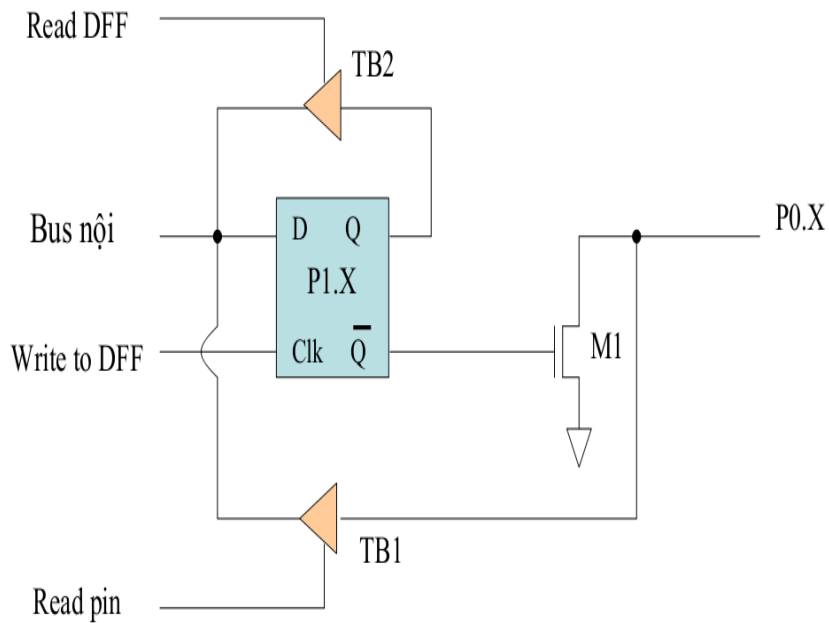
Xung  $\alpha\%$  tức là  $T1/T=\alpha\%=\alpha/100$ .

- Xung PWM sẽ được đưa ra điều khiển L298 thông qua độ rộng xung.khi không có xung động cơ không quay,khi 100% xung thì động cơ quay max.tuy xung phải lớn hơn một mức nào đó mới đủ để khởi động động cơ.
- Để có thể thay đổi 10 cấp với chu kỳ  $T=1000\mu\text{s}$  thì ta phải khởi tạo cho ngắt timer2:100us ngắt 1 lần .
- Sử dụng timer2 với chế độ tự nạp 16bit thì ta phải nạp cho RCAP2H;RCAP2L các con số :65536-100=65436=FF9C H.

$\Rightarrow$  RCAP2H=0xFF và RCAP2L=0x9C.

V.1.3.một số tính toán khác.

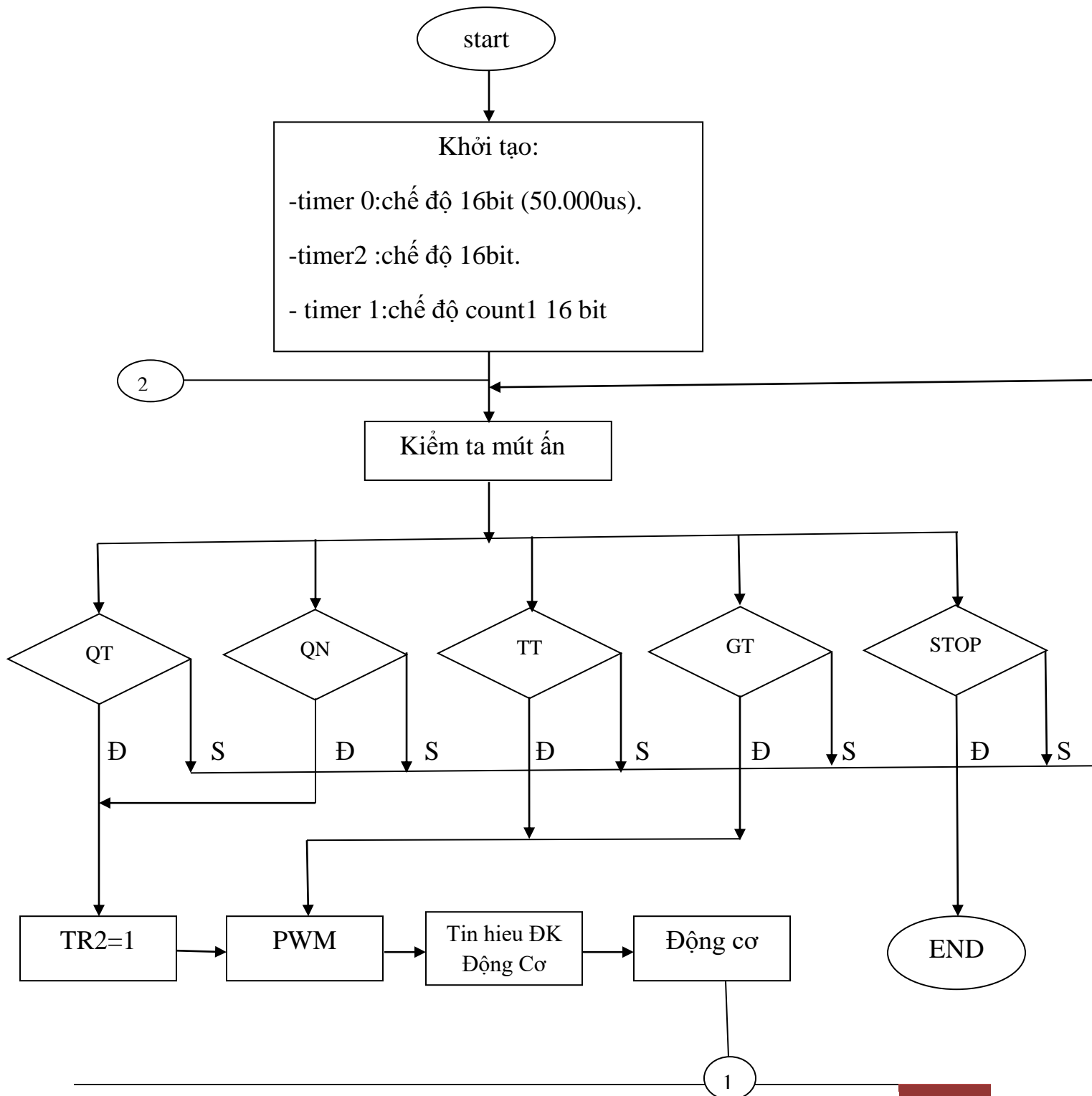
- Do cổng P0 của vi điều khiển 89C52 có dạng



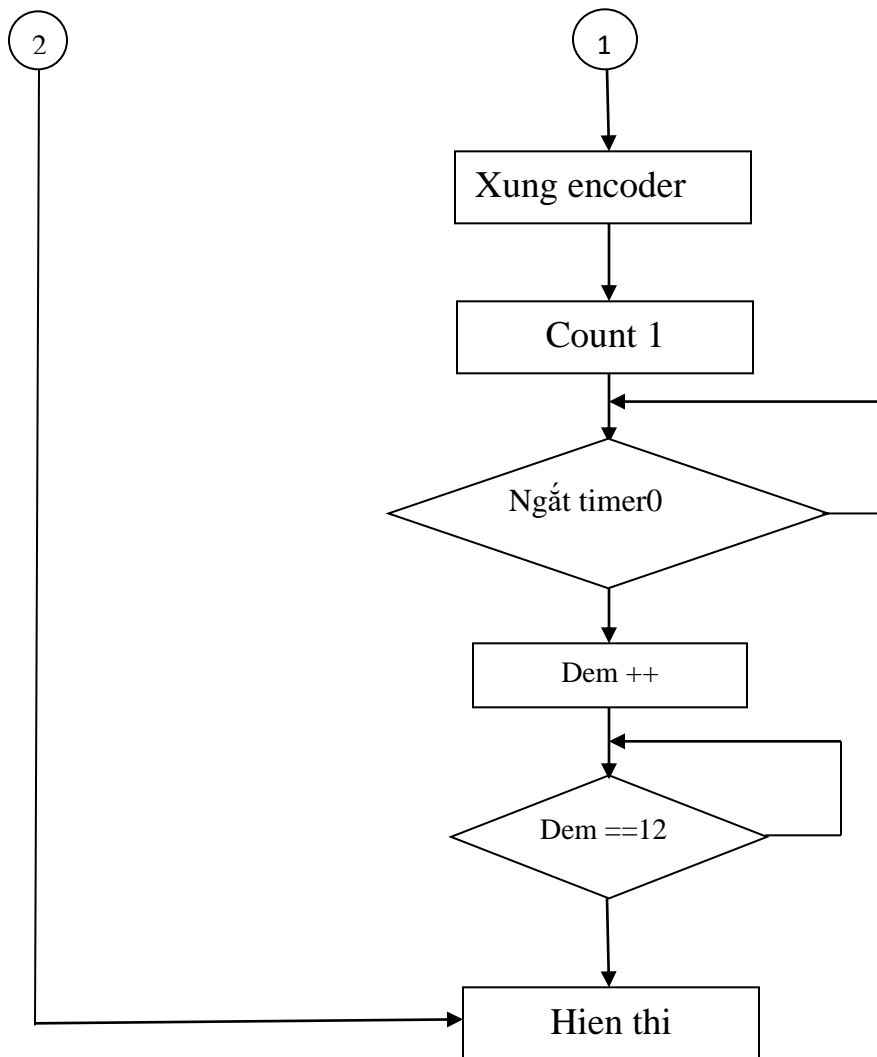
Như vậy cần có điện trở treo khoảng 10K (hoặc 4.7k)

## V.2.lưu đồ thuật toán

### a.lưu đồ chương trình của mạch.

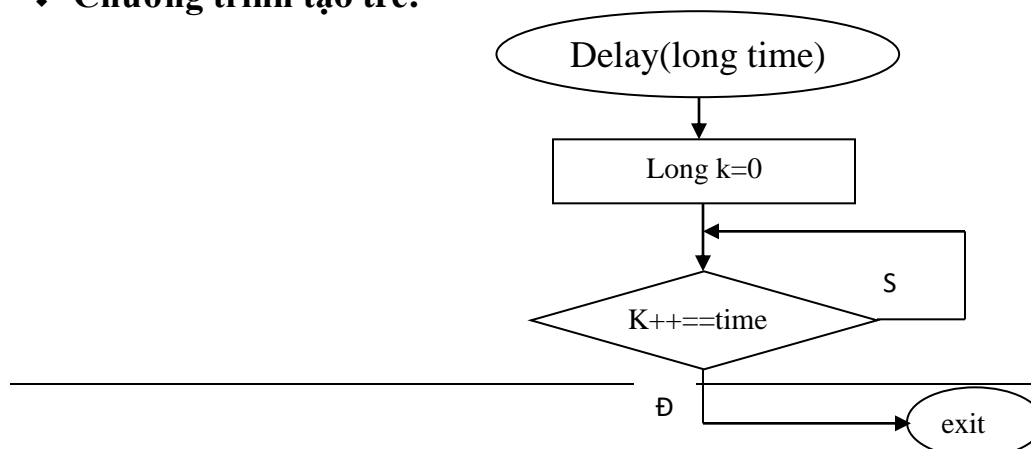




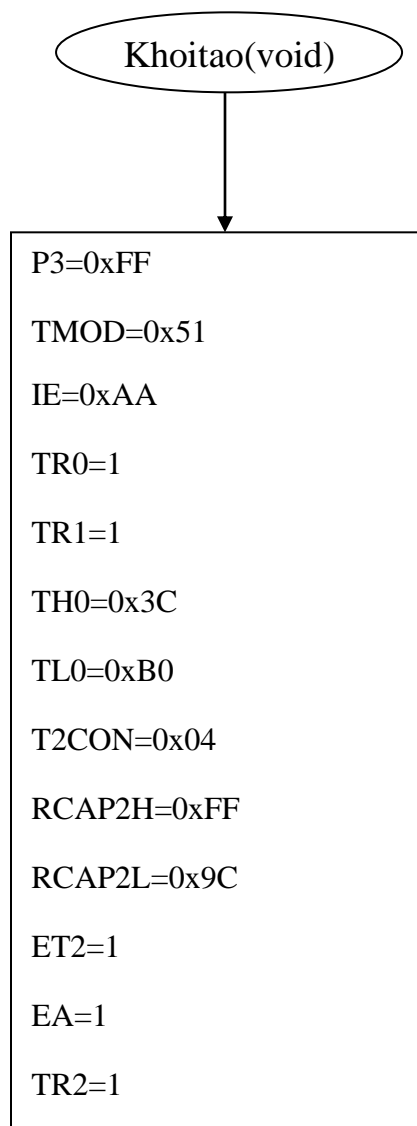


**b. các lưu đồ chương trình con và các hàm chức năng.**

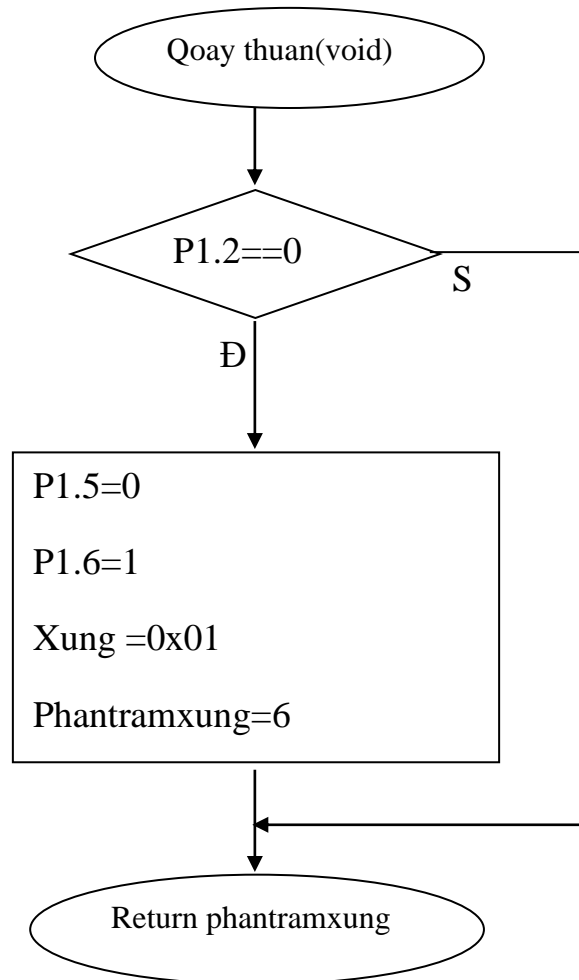
❖ **Chương trình tạo trễ:**



❖ Chương trình khởi tạo:

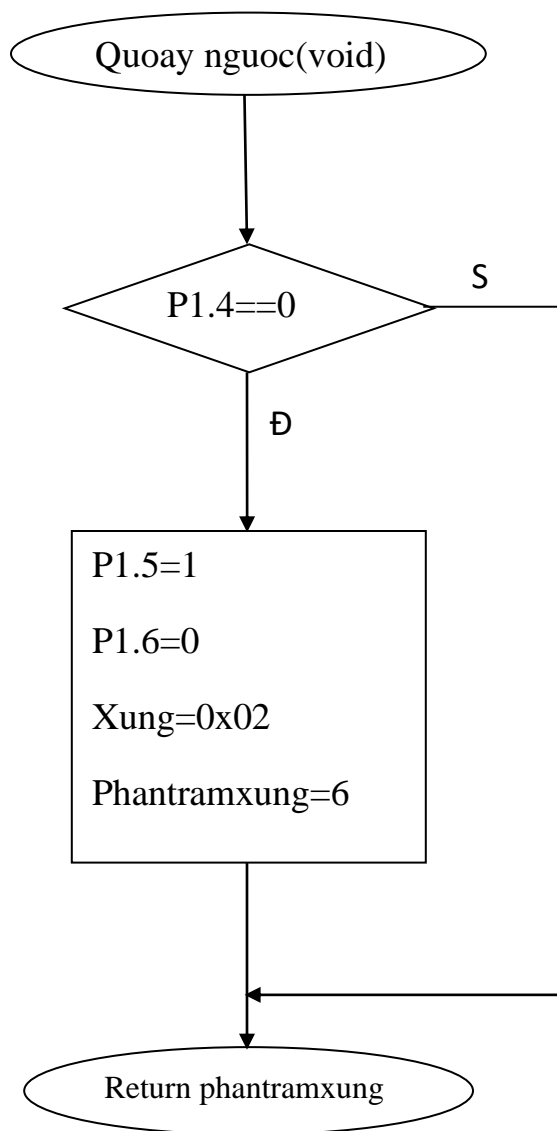


❖ Hàm khởi động động cơ quay thuận.

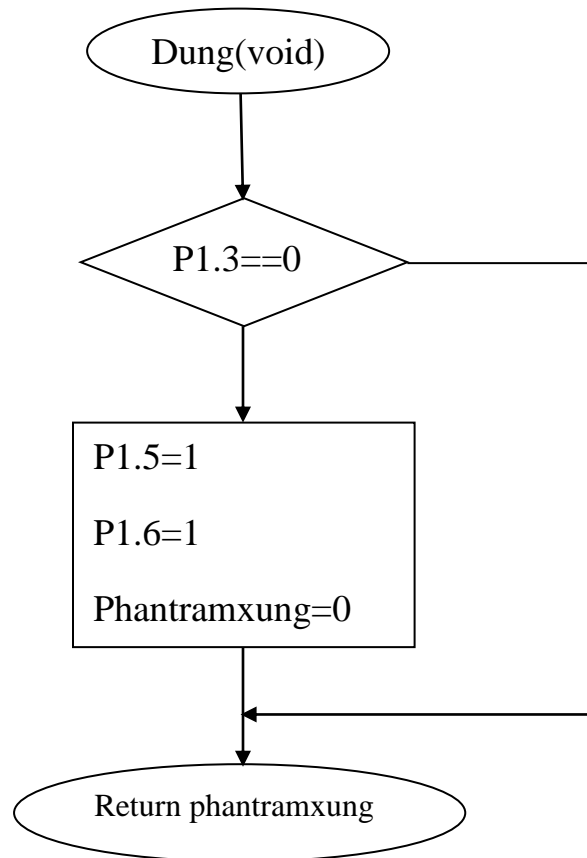


Note: P1.5 và P1.6 là các chân điều khiển sáng led thông báo là đang chạy thuận hay chạy nghịch.

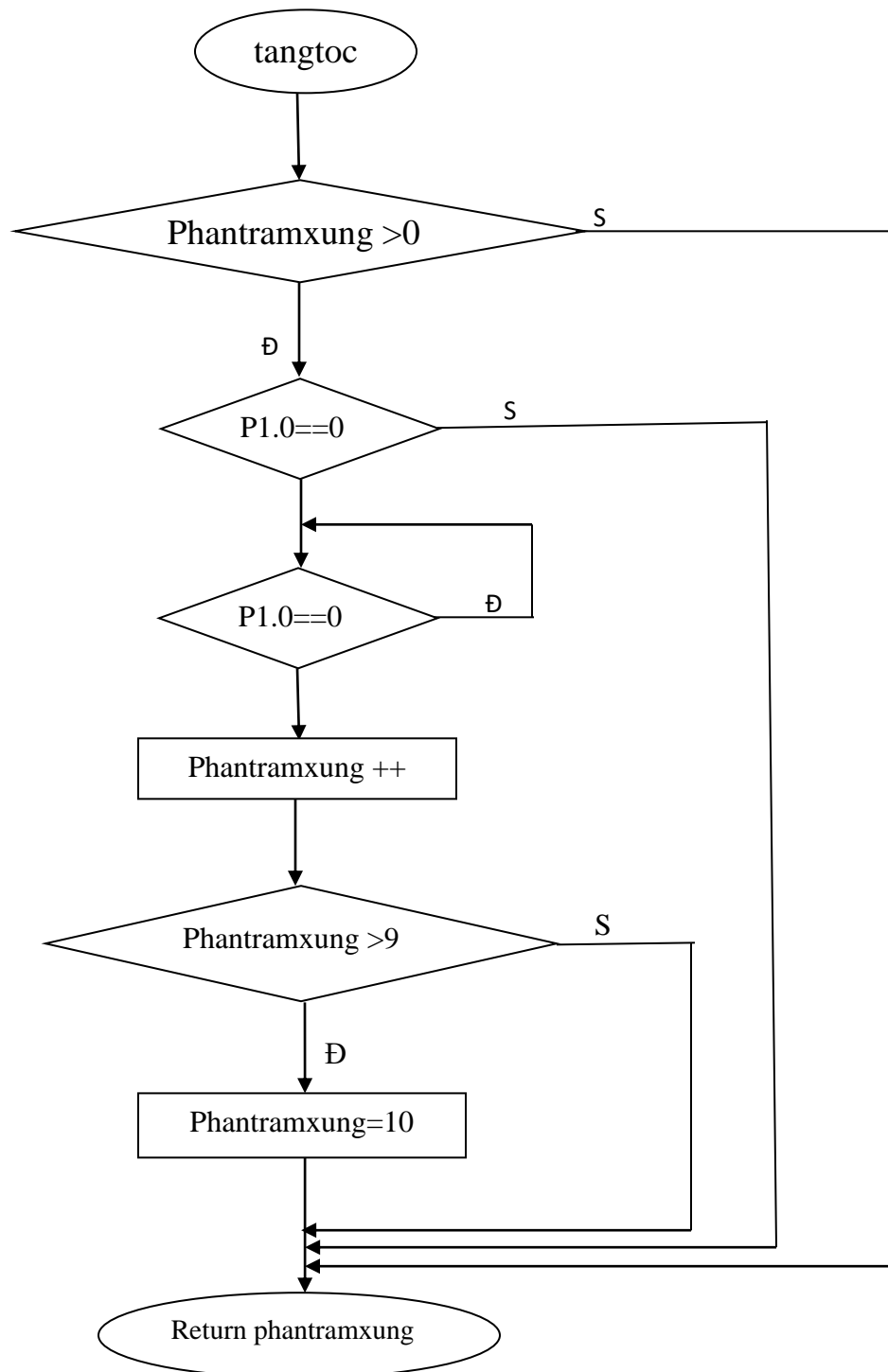
❖ Hàm khởi động động cơ quay ngược.



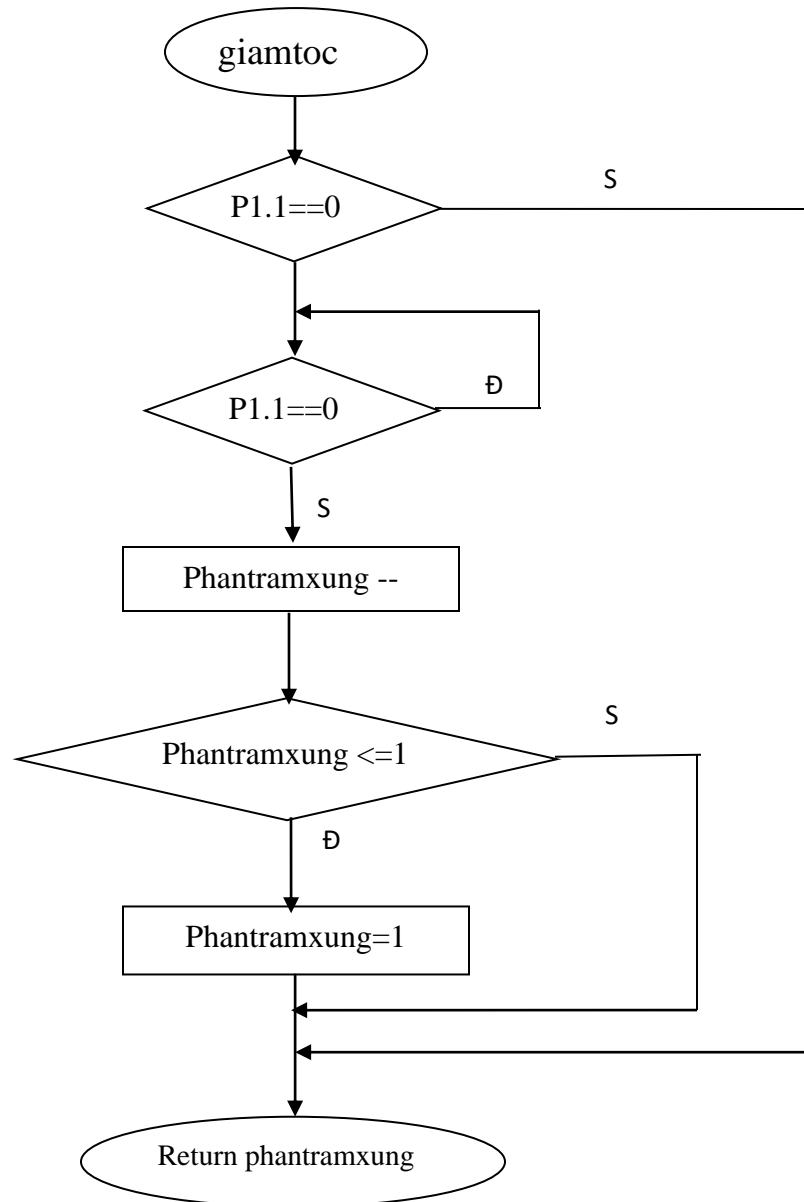
❖ Hàm khởi tạo dừng động.



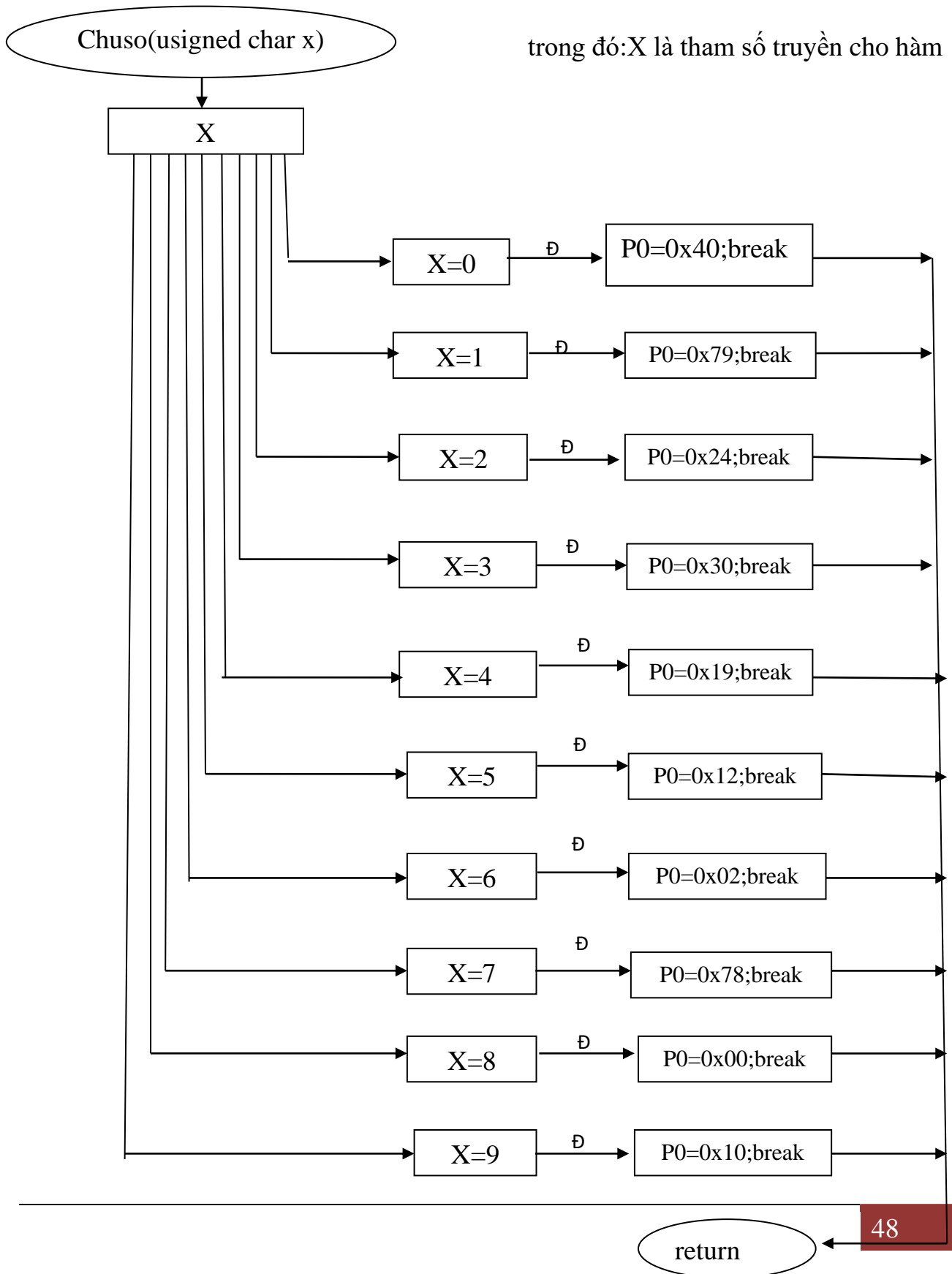
❖ hàm tăng tốc :



❖ Hàm giảm tốc độ:



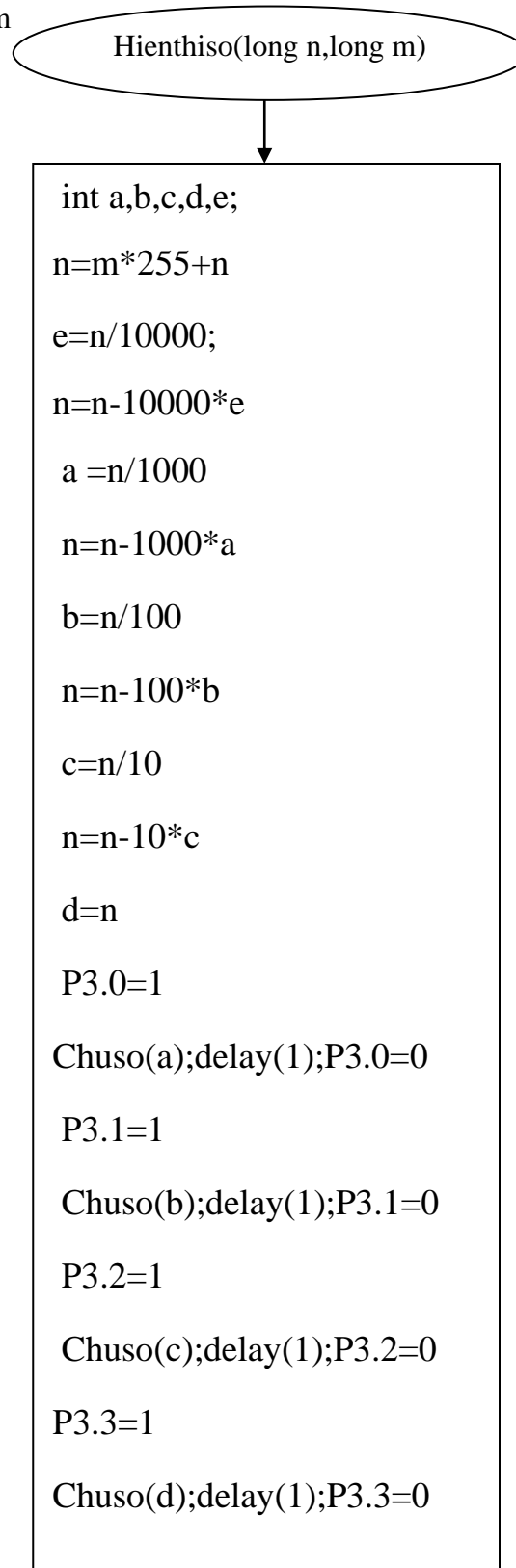
❖ Hàm hiển thị chữ số ra các PORT.



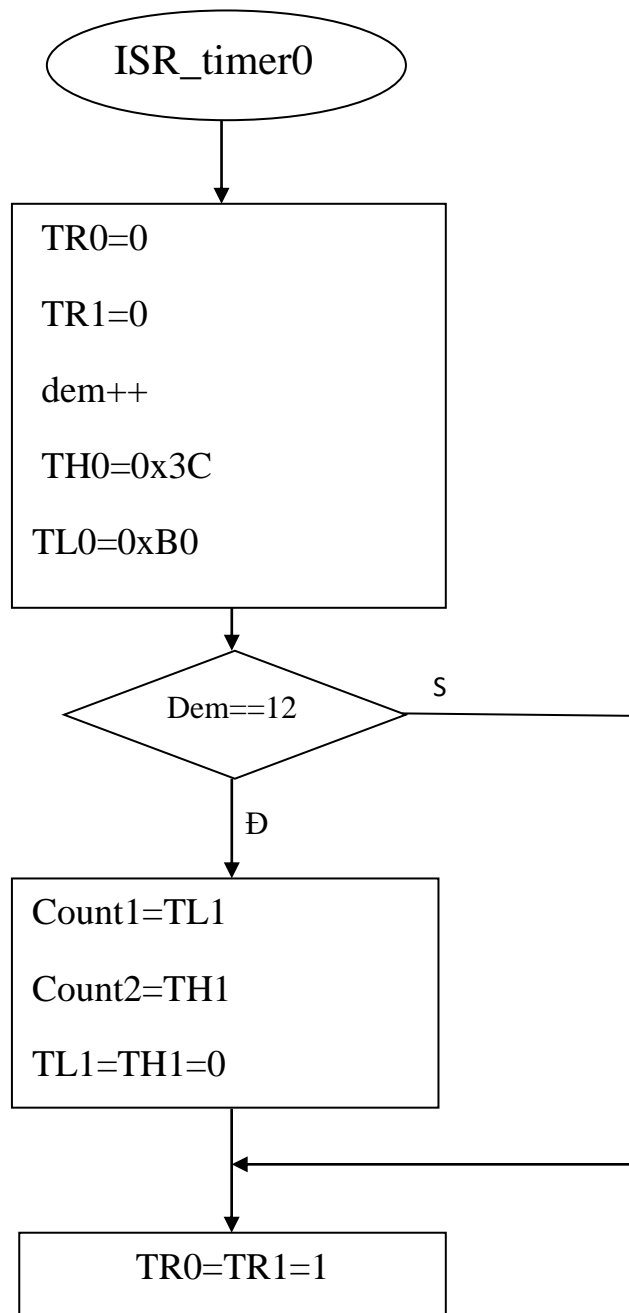


❖ **Hàm hiển thị số .**

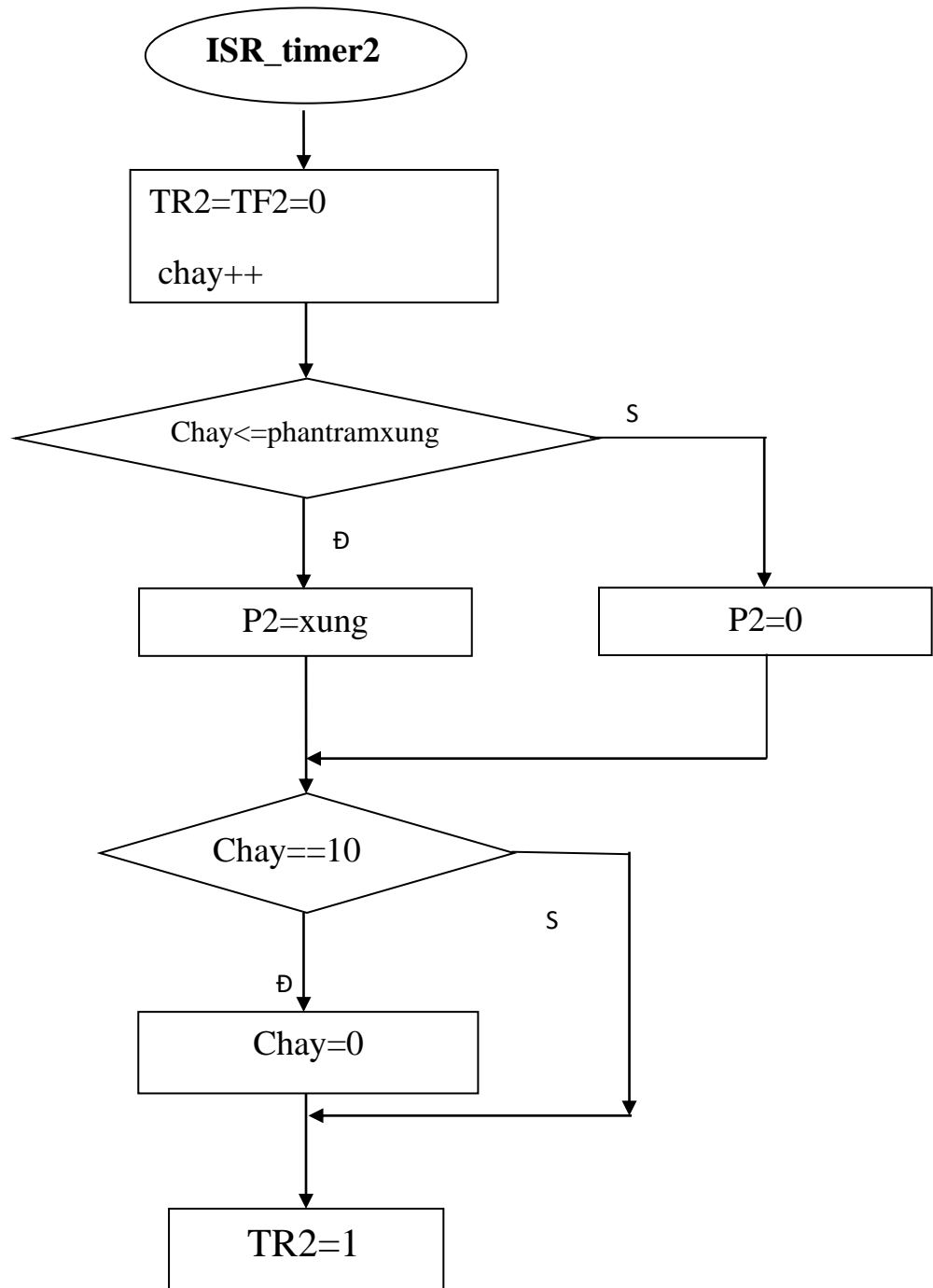
tham số truyền cho hàm là n,m



❖ Chương trình ngắt timer0.



❖ Hàm tạo xung PWM sử dụng timer2.



Note: khi chạy thuận thì `P2.0=1`  $\Leftrightarrow$  `xung=0x01` còn khi chạy ngược thì `P2.1=1`  $\Leftrightarrow$  `xung=0x02`.

## V.2.code được viết bằng C.

```

/*****Mo ta chuong trinh*****/

/*****Chuong trinh do luong va dieu khien dong co DC *****/

/***** sinh vien thuc hien *****/

TRƯỜNG VĂN TRỌNG      | PHAN CÔNG TOÁN   20092784
SHSV:20092846           | SHSV :20092784
LỚP :DK&TĐH 4-K54      | LỚP :ĐK&TĐH 4-K54

*****/

#include <REGX52.H>

int count1=0,count2=0;

int dem=0;

long k;

void khoitao(void){// Ham khoi tao

    P3=0xFF;

    TMOD=0x51;

    IE=0xAA;

    TR0=1;

    TR1=1;

    TH0=0x3C;

    TL0=0xB0;

    //T2MOD=0xc9;//Timer 2 che do 2 8 bit auto reload,tine1 16 bit

    T2CON=0x04; //timer 2 o che do tu nap 16 bit va nap boi RCAPL2

```

```

RCAP2H=0xFF; //Cho timer2 o che do dem 50.000 micro giay
RCAP2L=0x9C;
ET2=1; // Cho phep ngat timer 0
EA=1; // Cho phep ngat toan cuc
TR2=1; // Chay timer 0 bat dau dem so chu ki may
}

/*****Chương trình tạo tre *****/

void delay(long time){
    for (k=0;k<time;k++)    {}
}

/*****Ngat timer 0 *****/

void timer() interrupt 1 {
    TR0=0;
    TR1=0;
    dem++;
    TH0=0x3C;
    TL0=0xB0;

    if(dem==12){
        count1=TL1;
        count2=TH1;
        TL1=TH1=0;
    }
}

```

```

        dem=0;

    }

    TR0=1;

    TR1=1;

}

/*****ngat tiner2 tao PWM*****/

unsigned char xung;

unsigned char phantramxung;

unsigned char chay=0;// Khai bao bien dem de dem tu 1 den 10

void timer2(void) interrupt 5{ //Ngat timer 2

    TR2=0;

    TF2=0;          // Dung chay timer 2

    chay++;

    if(chay<=phantramxung) P2=xung; // Neu bien dem < phan tram xung thi
    dua gia tri 1 ra chan

    else P2=0x0F;      // Neu dem = phan tram xung

    if(chay==10) chay=0; // Neu dem du 10 thi gan lai bang 0 de bat dau chu ki
    moi

    TR2=1; // Cho chay timer

}

/*****Ham tang toc*****/

char tangtoc(void){//ham tang toc dong co

    if(phantramxung>0){

```

```

if (P1_0==0){ //neu nut tang duoc bam
    while(P1_0==0){ //nut dang giu;
        ; //khong lam j ca;
    }
    phantramxung++;
    if (phantramxung>9) phantramxung=10;
}
}

return phantramxung;
}

/*****Ham giam toc*****/
char giam(void){ //ham giam toc dong co
    if (P1_1==0){ //neu nut giam duoc bam
        while(P1_1==0){ //nut dang giu;
            ; //khong lam j ca;
        }
        phantramxung--;
        if (phantramxung<=1) phantramxung=1;
    }
    return phantramxung;
}

/*****ham khoi dong*****/

```

```
char quoythuan(void){//ham khoi dong dong co
```

```
    if(P1_2==0){
```

```
        P1_5=0;
```

```
        P1_6=1;
```

```
        xung=0x01;
```

```
        phantramxung=6;
```

```
    }
```

```
    return phantramxung;
```

```
}
```

```
/******ham quay nguoc*****/
```

```
char quaynguoc(void){//ham khoi dong dong co
```

```
    if(P1_4==0){
```

```
        P1_5=1;
```

```
        P1_6=0;
```

```
        xung=0x02;
```

```
        phantramxung=6;
```

```
    }
```

```
    return phantramxung;
```

```
}
```

```
/******ham dung*****/
```

```
char dung(void){//ham khoi dong dong co
```

```
    if(P1_3==0){
```



```

    P1_5=1;

    P1_6=1;

    phantramxung=0;

}

return phantramxung;

}

/***** cac ham dung de hien thi len leg *****/

void chuso(unsigned char x){ // Ham dua du lieu ra PORT

    switch(x){ // Tuy thuoc vao bien dau vao ma dua du lieu ra tu 0...9

        //logic 1 tat led, logic 0 bat led

        case 0: { P0=0x40; break; } // So 0
        case 1: { P0=0x79; break; } // So 1
        case 2: { P0=0x24; break; } // So 2
        case 3: { P0=0x30; break; } // So 3
        case 4: { P0=0x19; break; } // So 4
        case 5: { P0=0x12; break; } // So 5
        case 6: { P0=0x02; break; } // So 6
        case 7: { P0=0x78; break; } // So 7
        case 8: { P0=0x00; break; } // So 8
        case 9: { P0=0x10; break; } // So 9

    }
}

```

```

}

void hienthiso(long n,long m){
    int a,b,c,d,e;
    n=m*255+n;
    e=n/10000;
    n=n-10000*e;
    a=n/1000;n=n-1000*a;
    b=n/100;n=n-b*100;
    c=n/10; n=n-c*10;
    d=n;
    P3_0=1;chuso(a);delay(1);P3_0=0;
    P3_1=1;chuso(b);delay(1);P3_1=0;
    P3_2=1;chuso(c);delay(1);P3_2=0;
    P3_3=1;chuso(d);delay(1);P3_3=0;
}

/***** chương trình chính *****/

void main(void){
    khoitao();
    while(1){
        hienthiso(count1,count2);
        quoaythuan();
        tangtoc();
    }
}

```

```
giam();  
dung();  
quaynguoc();  
}
```

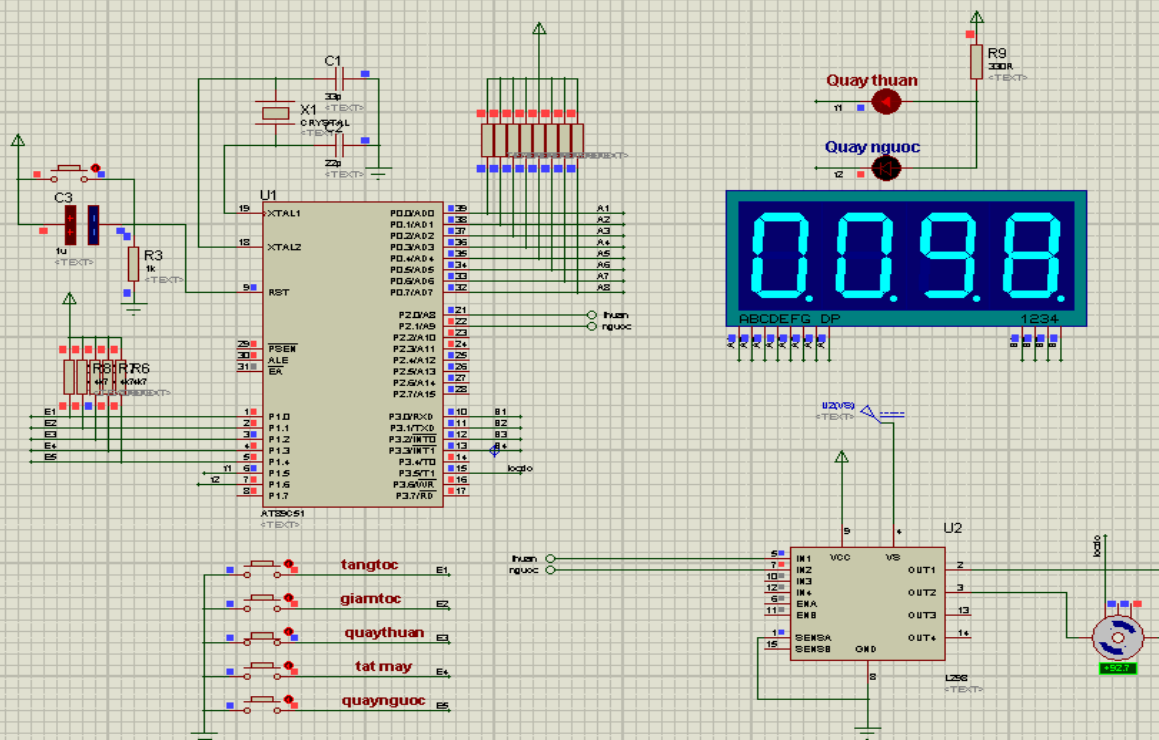
## CHƯƠNG VI: KẾT LUẬN

### VI.1 kết quả của đồ án.

Sau một thời gian tìm hiểu nghiên cứu chúng em đã hoàn toàn bài toán :

### “ĐO LƯỜNG VÀ ĐIỀU KHIỂN TỐC ĐỘ ĐỘNG CƠ “

Thỏa mãn các chỉ tiêu kĩ thuật đã dự định, chương trình được viết trên keilC và được mô phỏng trong proteus cho kết quả như mong đợi. đây là hình ảnh mô phỏng qua proteus:



### VI.2. ứng dụng và phương hướng phát triển.

#### 1. ứng dụng.

Mạch đo và điều khiển tốc độ động cơ dùng để xác định tốc độ động cơ một chiều từ đó Có thể:

- Biết tình trạng hoạt động của động cơ. phát hiện động cơ có chạy theo đúng thiết kế ban đầu không → biết được động cơ có bị hỏng hóc hay không.

- Điều chỉnh tốc độ động cơ theo yêu cầu tốc độ đã định sẵn.
- Giúp người vận hành thiết bị có thêm một công cụ quan sát trực quan hơn để có thể nắm tình hình, hạn chế phần nào sự có kỹ thuật xảy ra → vận hành thiết bị được tốt hơn.
- Trong các thiết bị sản xuất, dân dụng cần hoạt động với tốc độ ổn định.

## **2. phương hướng phát triển.**

Từ những ứng dụng thực tế đó mà các thiết bị đo tốc độ ngày càng được quan tâm phát triển. có mặt trong hàng loạt các ngành khoa học, trong sản xuất công nghiệp và cuộc sống hàng ngày.