

MỤC LỤC

LỜI MỞ ĐẦU	1
CHƯƠNG 1: GIỚI THIỆU CHUNG VỀ VI ĐIỀU KHIỂN 8051 ..	2
1.Khái quát chung	2
2.Cấu trúc bên trong của 8051:	2
2.1.Chức năng các chân điều khiển	4
2.2. Các thanh ghi đặc biệt	7
2.2.1. Thanh ghi ACC:.....	8
2.2.2. Thanh ghi B :.....	9
2.2.3. Thanh ghi SP:.....	9
2.2.4. Thanh ghi DPTR:.....	9
2.2.5. Ports 0 to 3:	9
2.2.6 Thanh ghi SBUF:	9
2.2.7. Các Thanh ghi Timer:.....	10
2.2.8. Các thanh ghi điều khiển:	10
2.2.9. Thanh ghi PSW:	10
2.2.10. Thanh ghi PCON:.....	11
2.2.11. Thanh ghi IE:.....	11
2.2.12. Thanh ghi IP:.....	12
2.2.13. Thanh ghi TCON :	12
2.2.14. Thanh ghi TMOD:.....	12
2.2.15. Thanh ghi SCON:.....	13
2.3. Khôi tạo thời gian và bộ đếm (Timer/Counter)	14
2.4. Cơ chế ngắt trong On-chip AT89C51:	18
2.4.1. Phân loại ngắt trong On-chip:	18
2.4.2. Các bước thực hiện ngắt.	19
2.4.3. Mức ngắt ưu tiên trong on-chip:	20
2.4.4. Nguyên lý điều khiển ngắt của AT89:	20
2.5. Bảo vệ chương trình	23
2.6. Tra cứu nhanh tập lệnh của 8051	24

CHƯƠNG 2. THIẾT KẾ VÀ THI CÔNG PHẦN CỨNG	28
A.Giới thiệu phần tử	28
2.1.Chỉnh lưu cầu một pha	28
2.2. IC tạo ổn áp 7805:(IC ổn áp 5v).....	31
2.3.Ghép nối mạch cầu H và mạch khuếch đại	32
2.4.Khởi Reset:.....	36
2.5. Khởi tạo xung dao động:.....	37
2.6. THIẾT KẾ MODULE LCD.....	37
2.6.1. Giới thiệu.....	37
2.6.2. Mô tả chân của LCD	38
2.6.3. Tập lệnh của LCD.	39
2.6.4 SƠ ĐỒ GHÉP NỐI LCD	42
2.7. ĐO TỐC ĐỘ ĐỘNG CƠ MỘT CHIỀU	43
2.7.1 Giới thiệu.....	43
2.7.2 Nguyên lý hoạt động.	43
2.8. IC 74HC245	43
2.8.1. Mô tả chung:.....	43
2.8.2. Sơ đồ chân:	44
2.8.3. Bảng hoạt động của IC 74HC245:	44
B.Phần cứng.....	46
2.9.Thiết kế mạch hoạt động cho AT89C51.....	46
2.10. Thiết kế bộ nguồn.....	47
2.11. MẠCH IN.....	49
CHƯƠNG 3: THIẾT KẾ PHẦN MỀM.....	50
3.1 THUẬT TOÁN	50
3.2 PHÂN LẬP TRÌNH	51
3.2.1 ĐIỀU KHIỂN ĐỘNG CƠ 12VDC(MOTO_12VDC)	51
3.2.2 TẠO THỜI GIAN TRỄ	54
3.2.3 ĐIỀU KHIỂN LCD.....	55
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	59
TÀI LIỆU THAM KHẢO	60

LỜI MỞ ĐẦU

Ứng dụng vi xử lý trong truyền động điện – điều khiển tốc độ động cơ điện là lĩnh vực quan trọng và ngày càng phát triển. Các nhà sản xuất không ngừng cho ra đời các sản phẩm và công nghệ mới về các phần tử bán dẫn công suất và các thiết bị điều khiển dung vi xử lý đi kèm. Do đó khi thực hiện đồ án chúng em đã cố gắng cập nhật những kiến thức mới nhất, những công nghệ mới trong lĩnh vực điều khiển động cơ dùng vi xử lý. Với yêu cầu thiết kế bộ điều khiển động cơ một chiều theo phương pháp thay đổi độ rộng xung, chúng em đã cố gắng tìm hiểu kỹ về các phương án công nghệ sao cho bản thiết kế vừa đảm bảo yêu cầu kỹ thuật, yêu cầu kinh tế. Với hy vọng đồ án vi xử lý này là một bản thiết kế kỹ thuật có thể áp dụng được trong thực tế nên chúng em đã cố gắng mô tả cụ thể, tính toán cụ thể các thông số của các sơ đồ mạch và viết các thông số ngay trên mạch.

Đồ án tốt nghiệp “Thiết kế xây dựng bộ đếm xung, ứng dụng đo tốc độ động cơ trong hệ thống truyền động điện” là kết quả trong quá trình học tập, tìm kiếm và tổng hợp tài liệu, lựa chọn, tính toán và trình bày một cách ngắn gọn súc tích, dễ hiểu nhất. Tuy nhiên trong quá trình tìm hiểu, tính toán không tránh được những nhận thức sai sót mong thầy cô và các bạn góp ý.

Trong quá trình làm đồ án em được sự giúp đỡ tận tình của thầy NGUYỄN TRỌNG THẮNG về tài liệu, cách trình bày, cách tìm kiếm tài liệu, tính toán thiết kế để có được đồ án môn học thành công như mong muốn.

Đề tài của em gồm 3 chương:

- Chương 1: Giới thiệu về họ vi điều khiển 8051
- Chương 2: Thiết kế và thi công phần cứng
- Chương 3: Thiết kế phần mềm

Em xin chân thành cảm ơn!

Hải Phòng, ngày tháng năm 2010
Sinh viên

Tạ Văn Luận

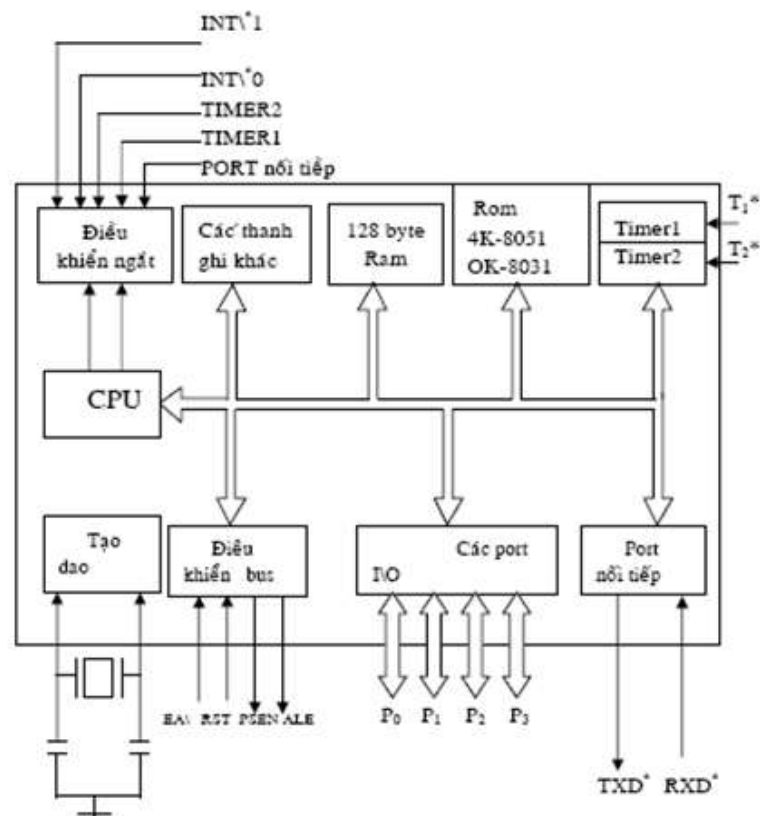
CHƯƠNG 1: GIỚI THIỆU CHUNG VỀ VI ĐIỀU KHIỂN 8051

1. Khái quát chung

IC vi điều khiển 8051 thuộc họ MCS51 có đặc điểm sau:

- 4kb ROM(được lập trình bởi nhà sản xuất chỉ có ở 8051)
- 128 byte Ram
- 4 port I/O 8 bit
- 2 bộ định thời 16 bit
- Giao tiếp nối tiếp
- 64 kb không gian bộ nhớ chương trình mở rộng
- 64 kb không gian bộ nhớ dữ liệu mở rộng
- 1 bộ xử lý luận lý (thao tác trên các bit đơn)
- 210 bit nhận địa chỉ hóa
- bộ nhân / chia $4\mu s$

2. Cấu trúc bên trong của 8051:



Hình 2.1 : Sơ đồ khối 8051

Phần chính của vi điều khiển 8051 là bộ xử lý trung tâm(CPU:centeral processing unit) bao gồm :

- Thanh ghi tích lũy A
- Thanh ghi tích lũy phụ B,dùng cho phép nhân và chia
- Đơn vị logic học (ALU : Arithmetic Logical Unit)
- Từ trạng thái chương trình (PSW :Program Status Word)
- Bốn băng thanh ghi
- Con trỏ ngăn xếp
- Ngoài ra còn có bộ nhớ chương trình ,bộ giải mã lệnh, bộ điều khiển thời gian và logic

Đơn vị xử lý trung tâm nhận trực tiếp xung từ bộ giao động ,ngoài ra còn có khả năng đưa một tín hiệu giữ nhịp từ bên ngoài.

Chương trình đang chạy có thể cho dừng lại nhờ một khối điều khiển ngắt ở bên trong .Các nguồn ngắt có thể là : các biến cố ở bên ngoài ,sự tràn bộ đếm định thời hoặc cũng có thể là giao diện nối tiếp.

Hai bộ định thời 16 bit hoạt động như 1 bộ đếm.

Các cổng (port0,port1,port2,port3).Sử dụng vào mục đích điều khiển.Ở cổng 3 có thêm các đường dẫn điều khiển dùng để tra đổi với một bộ nhớ bên ngoài,hoặc để đấu nối giao diện nối tiếp,cũng như các đường ngắt dẫn ở bên ngoài

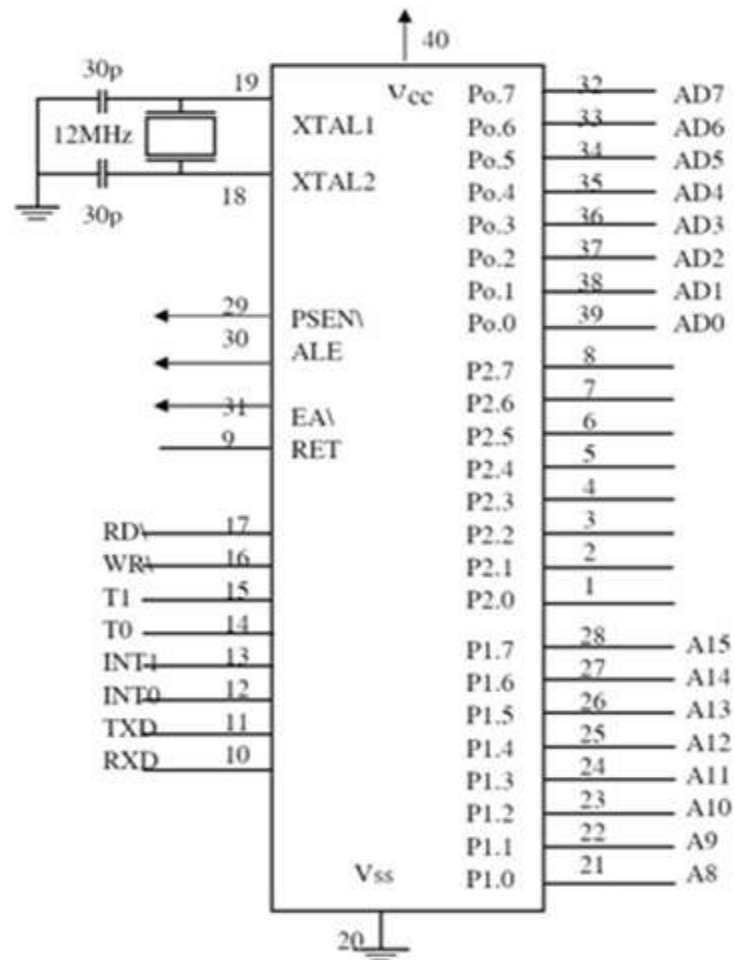
Giao diện nối tiếp có chứa một bộ truyền và một bộ nhận không đồng bộ,làm việc độc lập với nhau.Tốc độ truyền qua cổng nối tiếp có thể đặt trong dải rộng và được ấn định bằng một bộ định thời.

Trong vi điều khiển 8051 có 2 thành phần quan trọng khác đó là bộ nhớ và các thanh ghi:

Bộ nhớ gồm có bộ nhớ ram và bộ nhớ rom(chỉ có ở 8031) dùng để lưu trữ dữ liệu và mã lệnh.

Các thanh ghi sử dụng để lưu trữ thông tin trong quá trình xử lý .Khi CPU làm việc nó làm thay đổi nội dung của các thanh ghi.

2.1.Chức năng các chân điều khiển



Hình 2.1 sơ đồ chân 8051

a.port0 :là port có chức năng ở trên chân từ 32 đến 39 trong các thiết kế cỡ nhỏ (không dùng bộ nhớ mở rộng) có 2 chức năng như các đường I/O.Đối với các thiết kế cỡ lớn (với bộ nhớ mở rộng) nó được kết hợp kênh giữa các bus.

b.port1 : port1 là một port I/O trên các chân 1 – 8.các chân được kí hiệu p1.0,p1.1,p1.2...có thể dùng cho các thiết bị ngoài nếu cần.Port1 không có chức năng khác, vì vậy chúng ta chỉ được dùng trong giao tiếp với các thiết bị ngoài .

c.port2 : port2 là một port công cụ kép trên các chân 21 – 28 được dùng như các đường xuất nhập hoặc byte cao của bus địa chỉ đối với các thiết kế dùng bộ nhớ mở rộng.

d.port3 : port3 là một port công cụ kép trên các chân 10 -17.các chân của port này có nhiều chức năng , các công cụ chuyển đổi có liên hệ với các tính đặc biệt của 8051 như ở bảng sau:

Bit	Tên	Chức năng chuyển đổi
P3.0	RXD	Dữ liệu nhận cho port nối tiếp
P3.1	TXD	Dữ liệu phát cho port nối tiếp
P3.2	INT0	Ngắt 0 bên ngoài
P3.3	INT1	Ngắt 1 bên ngoài
P3.4	TO	Ngõ vào của timer/counter 0
P3.5	T1	Ngõ vào của timer/counter 1
P3.6	WR	Xung ghi bộ nhớ dữ liệu ngoài
P3.7	RD	Xung đọc bộ nhớ dữ liệu ngoài

e.Psen (program store enable) : 8052 có 4 tín hiệu điều khiển PSEN là tín hiệu ra trên chân 29.Nó là tín hiệu điều khiển để cho phép bộ nhớ chương trình mở rộng và thường được nối đến chân OE (output enable) của 1 EPROM để cho phép đọc các byte mã lệnh.

PSEN sẽ ở mức thấp trong thời gian lấy lệnh.Các mã nhị phân của chương trình được đọc từ EPROM qua bus và được chốt vào thanh ghi lệnh của 8051 để giải mã lệnh.Khi thi hành chương trình trong ROM nội (8051) PSEN sẽ ở mức thụ động (mức cao).

f.Ale (Address latch Enable):

Tín hiệu ra ALE trên chân 30 tương tự với các thiết bị làm việc với các xử lý 8585, 8088 ,8086 , 8051 dùng ALE một cách tương tự cho làm việc giải các kênh bus địa chỉ và dữ liệu khi port0 được dùng trong chế độ chuyển đổi của nó : Vừa là bus dữ liệu vừa là bus thấp của địa chỉ, ALE là tín hiệu để chốt địa chỉ vào một thanh ghi bên ngoài trong nửa đầu của chu kỳ bộ nhớ.Sau đó các đường port 0 dùng để xuất nhập dữ liệu trong nửa sau chu kỳ của bộ nhớ.

Các xung tín hiệu ALE có tốc độ bằng 1/6 lần tần số dao động trên chip và có thể được dùng là nguồn xung nhịp cho các hệ thống.nếu xung trên 8051 là 12MHz thì

ALE có tần số 2MHz. Chỉ ngoại trừ khi thi hành lệnh MOVX, 1 xung ALE sẽ bị mất. Chân này cũng được làm ngõ vào cho xung lập trình cho EPROM trong 8051.

g.Ea (External Access) :

Tín hiệu vào EA trên chân 31 thường được mắc lên mức cao (+5v) hoặc mức thấp (GND) .Nếu ở mức cao ,8051 thi hành chương trình từ ROM nội trong khoảng địa chỉ thấp (4k) .Nếu ở mức thấp, chương trình chỉ được thi hành từ bộ nhớ mở rộng.Nếu EA được nối mức thấp bộ nhớ bên trong, chương trình 8051 sẽ bị cấm và chương trình thi hành từ EPROM mở rộng. Người ta còn dùng chân EA làm chân cấp điện áp 12V khi lập trình cho EPROM trong 8051.

h.Rst (Reset):

Ngõ vào RST trên chân 9 là ngõ reset của 8051. Khi tín hiệu này được đưa lên mức cao (trong ít nhất 2 chu kỳ máy) ,các thanh ghi trong 8051 được tải những giá trị thích hợp để khởi động hệ thống.

Figure 14. Using the On-Chip Oscillator

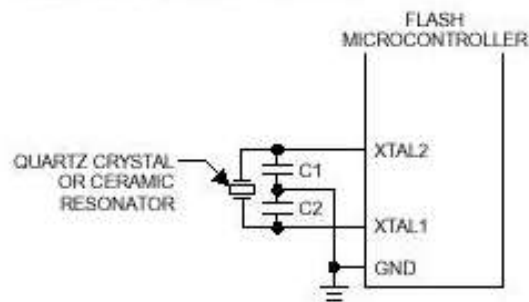
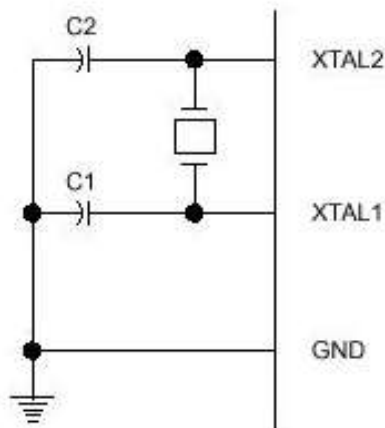


Figure 15. A: Oscillator Connections



i. Các ngõ vào bộ dao động trên chip :

Như đã thấy trong các hình trên , 8051 có 1 bộ dao động trên chip.Nó thường được nối với thạch anh giữa 2 chân 18 và 19.Các tụ giữa cũng cần thiết kể như đã vẽ.Tần số thạch anh thông thường là 12MHz.

j. Các chân nguồn :

Vcc: Cung cấp dương nguồn cho On-chip (+ 5V).

GND: nối mát.

2.2. Các thanh ghi đặc biệt

SFR đảm nhiệm các chức năng khác nhau trong On-chip. Chúng nằm ở RAM bên trong On-chip, chiếm vùng không gian nhớ 128 Byte được định địa chỉ từ 80h đến FFh. Cấu trúc của SFR bao gồm các chức năng thể hiện ở bảng 2.3 và bảng 2.4.

Thanh ghi	MSB	Nội dung						LSB
<i>IE</i>	EA	-	ET2	ES	ET1	EX1	ET0	EX0
<i>IP</i>	-	-	PT2	PS	PT1	PX1	PT0	PX0
<i>PSW</i>	CY	AC	FO	RS1	RS0	OV	-	P
<i>TMOD</i>	GATE	C/(/T)	M1	M0	GATE	C/(/T)	M1	M0
<i>TCON</i>	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
<i>SCON</i>	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
<i>PCON</i>	SMOD	-	-	-	GF1	GF0	PD	IDL
<i>P1</i>	T2	T2EX			/SS	MOS I	MISO	SCK
<i>P3</i>	RXD	TXD	/INT 0	/INT1	T0	T1	/WR	/RD

Bảng 2.3. Chức năng riêng của từng thanh ghi trong SFR

Symbol	Name	Address	Reset Values
* ACC	Thanh ghi tích lũy	0E0h	00000000b
* B	Thanh ghi B	0F0h	00000000b
* PSW	Từ trạng thái chương trình	0D0h	00000000b
SP	Con trỏ ngăn xếp	81h	00000111b
DP0L	Byte cao của con trỏ dữ liệu 0	82h	00000000b
DP0H	Byte thấp của con trỏ dữ liệu 0	83h	00000000b
* P0	Cổng 0	80h	11111111b
* P1	Cổng 1	90h	11111111b
* P2	Cổng 2	0A0h	11111111b
* P3	Cổng 3	0B0h	11111111b
* IP	TG điều khiển ngắt ưu tiên	0B8h	xxx00000b
* IE	TG điều khiển cho phép ngắt	0A8h	0xx00000b
TMOD	Điều khiển kiểu Timer/Counter	89h	00000000b
* TCON	TG điều khiển Timer/Counter	88h	00000000b
TH0	Byte cao của Timer/Counter 0	8Ch	00000000b
TL0	Byte thấp của Timer/Counter 0	8Ah	00000000b
TH1	Byte cao của Timer/Counter 1	8Dh	00000000b
TL1	Byte thấp của Timer/Counter 1	8Bh	00000000b
* SCON	Serial Control	98h	00000000b
SBUF	Serial Data Buffer	99h	indeterminate
PCON	Power Control	87h	0xxx0000b

* : có thể định địa chỉ bit, x: không định nghĩa

Bảng 2.4. Địa chỉ, ý nghĩa và giá trị của các SFR sau khi Reset

2.2.1. Thanh ghi ACC: là thanh ghi tích lũy, dùng để lưu trữ các toán hạng và kết quả của phép tính. Thanh ghi ACC dài 8 bits. Trong các tập lệnh của On-chip, nó thường được quy ước đơn giản là A.

2.2.2. Thanh ghi B : Thanh ghi này được dùng khi thực hiện các phép toán nhân và chia. Đối với các lệnh khác, nó có thể xem như là thanh ghi đệm tạm thời. Thanh ghi B dài 8 bits. Nó thường được dùng chung với thanh ghi A trong các phép toán nhân hoặc chia.

2.2.3. Thanh ghi SP: Thanh ghi con trỏ ngăn xếp dài 8 bit. SP chứa địa chỉ của dữ liệu hiện đang ở đỉnh của ngăn xếp. Giá trị của nó được tự động tăng lên khi thực hiện lệnh PUSH trước khi dữ liệu được lưu trữ trong ngăn xếp. SP sẽ tự động giảm xuống khi thực hiện lệnh POP. Ngăn xếp có thể đặt ở bất cứ nơi nào trong RAM on-chip, nhưng sau khi khởi động lại hệ thống thì con trỏ ngăn xếp mặc định sẽ trỏ tới địa chỉ khởi đầu là 07h, vì vậy ngăn xếp sẽ bắt đầu từ địa chỉ 08h. Ta cũng có thể định con trỏ ngăn xếp tại địa chỉ mong muốn bằng các lệnh di chuyển dữ liệu thông qua định địa chỉ tức thời.

2.2.4. Thanh ghi DPTR: Thanh ghi con trỏ dữ liệu (16 bit) bao gồm 1 thanh ghi byte cao (DPH-8bit) và 1 thanh ghi byte thấp (DPL-8bit). DPTR có thể được dùng như thanh ghi 16 bit hoặc 2 thanh ghi 8 bit độc lập. Thanh ghi này được dùng để truy cập RAM ngoài.

2.2.5. Ports 0 to 3: P0, P1, P2, P3 là các chốt của các cổng 0, 1, 2, 3 tương ứng. Mỗi chốt gồm 8 bit. Khi ghi mức logic 1 vào một bit của chốt, thì chân ra tương ứng của cổng ở mức logic cao. Còn khi ghi mức logic 0 vào mỗi bit của chốt thì chân ra tương ứng của cổng ở mức logic thấp. Khi các cổng đảm nhiệm chức năng như các đầu vào thì trạng thái bên ngoài của các chân cổng sẽ được giữ ở bit chốt tương ứng. Tất cả 4 cổng của on-chip đều là cổng I/O hai chiều, mỗi cổng đều có 8 chân ra, bên trong mỗi chốt bit có bộ “Pullup-tăng cường” do đó nâng cao khả năng nối ghép của cổng với tải (có thể giao tiếp với 4 đến 8 tải loại TTL).

2.2.6 Thanh ghi SBUF: Đệm dữ liệu nối tiếp gồm 2 thanh ghi riêng biệt, một thanh ghi đệm phát và một thanh ghi đệm thu. Khi dữ liệu được chuyển tới SBUF, nó sẽ đi vào bộ đệm phát, và được giữ ở đây để chế biến thành dạng truyền tin nối tiếp. Khi dữ liệu được truyền đi từ SBUF, nó sẽ đi ra từ bộ đệm thu.

2.2.7. Các Thanh ghi Timer: Các đôi thanh ghi (TH0, TL0), (TH1, TL1) là các thanh ghi đếm 16 bit tương ứng với các bộ Timer/Counter 0 và 1.

2.2.8. Các thanh ghi điều khiển: Các thanh ghi chức năng đặc biệt: IP, IE, TMOD, TCON, SCON, và PCON bao gồm các bit trạng thái và điều khiển đối với hệ thống ngắt, các bộ Timer/Counter và cổng nối tiếp. Chúng sẽ được mô tả ở phần sau.

2.2.9. Thanh ghi PSW: Từ trạng thái chương trình dùng để chứa thông tin về trạng thái chương trình. PSW có độ dài 8 bit, mỗi bit đảm nhiệm một chức năng cụ thể. Thanh ghi này cho phép truy cập ở dạng mức bit.

* CY: Cờ nhớ. Trong các phép toán số học, nếu có nhớ từ phép cộng bit 7 hoặc có số mượn mang đến bit 7 thì CY được đặt bằng 1.

* AC: Cờ nhớ phụ (Đối với mã BCD). Khi cộng các giá trị BCD, nếu có một số nhớ được tạo ra từ bit 3 chuyển sang bit 4 thì AC được đặt bằng 1. Khi giá trị được cộng là BCD, lệnh cộng phải được thực hiện tiếp theo bởi lệnh **DA A** (hiệu chỉnh thập phân thanh chứa A) để đưa các kết quả lớn hơn 9 về giá trị đúng.

* F0: Cờ 0 (Có hiệu lực với các mục đích chung của người sử dụng)

* RS1: Bit 1 điều khiển chọn băng thanh ghi.

* RS0: Bit 0 điều khiển chọn băng thanh ghi.

Lưu ý: RS0, RS1 được đặt/xoá bằng phần mềm để xác định băng thanh ghi đang hoạt động (Chọn băng thanh ghi bằng cách đặt trạng thái cho 2 bit này)

	RS1	RS0
Bank 0	0	0
Bank 1	0	1
Bank 2	1	0
Bank 3	1	1

Bảng 2.5. Chọn băng thanh ghi

* OV: Cờ tràn. Khi thực hiện các phép toán cộng hoặc trừ mà xuất hiện một tràn số học, thì OV được đặt bằng 1. Khi các số có dấu được cộng hoặc được trừ, phần mềm có thể kiểm tra OV để xác định xem kết quả có nằm trong tầm hay không. Với phép cộng các số không dấu, OV được bỏ qua. Kết quả lớn hơn +128 hoặc nhỏ hơn -127 sẽ đặt OV=1.

- * -: Bit dành cho người sử dụng tự định nghĩa(Nếu cần).
- * P: Cờ chặn lẻ. Được tự động đặt/ xoá bằng phần cứng trong mỗi chu trình lệnh để chỉ thị số chẵn hay lẻ của bit 1 trong thanh ghi tích lũy. Số các bit 1 trong A cộng với bit P luôn luôn là số chẵn.

2.2.10. Thanh ghi PCON: Thanh ghi điều khiển nguồn.

- * SMOD: Bit tạo tốc độ Baud gấp đôi. Nếu Timer 1 được sử dụng để tạo tốc độ baud và SMOD=1, thì tốc độ Baud được tăng lên gấp đôi khi công truyền tin nối tiếp được dùng bởi các kiểu 1, 2 hoặc 3.
- * -: Không sử dụng, các bit này có thể được dùng ở các bộ VXL trong tương lai. Người sử dụng không được phép tự định nghĩa cho các bit này.
- * GF0, GF1: Cờ dùng cho các mục đích chung (đa mục đích).
- * PD: bit nguồn giảm. Đặt bit này ở mức tích cực để vận hành chế độ nguồn giảm trong AT89C51. Chỉ có thể ra khỏi chế độ bằng Reset.
- * IDL: bit chọn chế độ nghỉ. Đặt bit này ở mức tích cực để vận hành kiểu Idle (Chế độ không làm việc) trong AT89C51.

Lưu ý: Nếu PD và IDL cùng được kích hoạt cùng 1 lúc ở mức tích cực, thì PD được ưu tiên thực hiện trước. Chỉ ra khỏi chế độ bằng 1 ngắt hoặc Reset lại hệ thống.

2.2.11. Thanh ghi IE: Thanh ghi cho phép ngắt

- * EA: Nếu EA=0, không cho phép bất cứ ngắt nào hoạt động. Nếu EA=1, mỗi nguồn ngắt riêng biệt được phép hoặc không được phép hoạt động bằng cách đặt hoặc xoá bit Enable của nó.
- * -: Không dùng, người sử dụng không nên định nghĩa cho Bit này, bởi vì nó có thể được dùng ở các bộ AT89 trong tương lai.
- * ET2: Bit cho phép hoặc không cho phép ngắt bộ Timer 2.
- * ES: Bit cho phép hoặc không cho phép ngắt công nối tiếp (SPI và UART).
- * ET1: Bit cho phép hoặc không cho phép ngắt tràn bộ Timer 1
- * EX1: Bit cho phép hoặc không cho phép ngắt ngoài 1.
- * ET0: Bit cho phép hoặc không cho phép ngắt tràn bộ Timer 0
- * EX0: Bit cho phép hoặc không cho phép ngắt ngoài 0.

2.2.12. Thanh ghi IP: Thanh ghi ưu tiên ngắt.

* - : Không dùng, người sử dụng không nên ghi “1” vào các Bit này.

* PT2: Xác định mức ưu tiên của ngắt Timer 2.

* PS: Định nghĩa mức ưu tiên của ngắt cổng nối tiếp.

* PT1: Định nghĩa mức ưu tiên của ngắt Timer 1.

* PX1: Định nghĩa mức ưu tiên của ngắt ngoài 1.

* PT0: Định nghĩa mức ưu tiên của ngắt Timer 0.

* PX0: Định nghĩa mức ưu tiên của ngắt ngoài 0.

2.2.13. Thanh ghi TCON : Thanh ghi điều khiển bộ Timer/Counter

* TF1: Cờ tràn Timer 1. Được đặt bởi phần cứng khi bộ Timer 1 tràn. Được xoá bởi phần cứng khi bộ vi xử lý hướng tới chương trình con phục vụ ngắt.

* TR1: Bit điều khiển bộ Timer 1 hoạt động. Được đặt/xoá bởi phần mềm để điều khiển bộ Timer 1 ON/OFF

* TF0: Cờ tràn Timer 0. Được đặt bởi phần cứng khi bộ Timer 0 tràn. Được xoá bởi phần cứng khi bộ vi xử lý hướng tới chương trình con phục vụ ngắt.

* TR0: Bit điều khiển bộ Timer 0 hoạt động. Được đặt/xoá bởi phần mềm để điều khiển bộ Timer 0 ON/OFF.

* IE1: Cờ ngắt ngoài 1. Được đặt bởi phần cứng khi sườn xung của ngắt ngoài 1 được phát hiện. Được xoá bởi phần cứng khi ngắt được xử lý.

* IT1: Bit điều khiển ngắt 1 để tạo ra ngắt ngoài. Được đặt/xoá bởi phần mềm.

* IE0: Cờ ngắt ngoài 0. Được đặt bởi phần cứng khi sườn xung của ngắt ngoài 0 được phát hiện. Được xoá bởi phần cứng khi ngắt được xử lý.

* IT0: Bit điều khiển ngắt 0 để tạo ra ngắt ngoài. Được đặt/xoá bởi phần mềm.

2.2.14. Thanh ghi TMOD: Thanh ghi điều khiển kiểu Timer/Counter

* GATE: Khi TRx được thiết lập và GATE=1, bộ TIMER/COUNTERx hoạt động chỉ khi chân INTx ở mức cao. Khi GATE=0, TIMER/COUNTERx sẽ hoạt động chỉ khi TRx=1.

* C/(/T): Bit này cho phép chọn chức năng là Timer hay Counter.

- Bit này được xoá để thực hiện chức năng Timer

- Bit này được đặt để thực hiện chức năng Counter

* M0, M1: Bit chọn Mode, để xác định trạng thái và kiểu Timer/Counter:

- M1=0, M0=0: Chọn kiểu bộ Timer 13 bit. Trong đó THx dài 8 bit, còn TLx dài 5 bit.

- M1=0, M0=1: Chọn kiểu bộ Timer 16 bit. THx và TLx dài 16 bit được ghép tầng.

- M1=1, M0=0: 8 bit Auto reload. Các thanh ghi tự động nạp lại mỗi khi bị tràn. Khi bộ Timer bị tràn, THx dài 8 bit được giữ nguyên giá trị, còn giá trị nạp lại được đưa vào TLx.

- M1=1, M0=1: Kiểu phân chia bộ Timer. TL0 là 1 bộ Timer/Counter 8 bit, được điều khiển bằng các bit điều khiển bộ Timer 0, Còn TH0 chỉ là bộ Timer 8 bit, được điều khiển bằng các bit điều khiển Timer 1.

- M1=1, M0=1: Timer/Counter 1 Stopped

2.2.15. Thanh ghi SCON:

SCON là thanh ghi trạng thái và điều khiển cổng nối tiếp. Nó không những chứa các bit chọn chế độ, mà còn chứa bit dữ liệu thứ 9 dành cho việc truyền và nhận tin (TB8 và RB8) và chứa các bit ngắt cổng nối tiếp.

* SM0, SM1: Là các bit cho phép chọn chế độ cho cổng truyền nối tiếp.

SM0	SM1	Mode	Đặc điểm	Tốc độ Baud
0	0	0	Thanh ghi dịch	$F_{osc}/12$
0	1	1	8 bit UART	Có thể thay đổi (được đặt bởi bộ Timer)
1	0	2	9 bit UART	$F_{osc}/64$ hoặc $F_{osc}/32$
1	1	3	9 bit UART	Có thể thay đổi (được đặt bởi bộ Timer)

Bảng 2.6. Chọn Mode trong SCON

* SM2: Cho phép truyền tin đa xử lý, thể hiện ở Mode 2 và 3. ở chế độ 2 hoặc 3, nếu đặt SM2 = 1 thì RI sẽ không được kích hoạt nếu bit dữ liệu thứ 9 (RB8) nhận được giá trị bằng 0. ở Mode 1, nếu SM2=1 thì RI sẽ không được kích hoạt nếu bit dừng có hiệu lực đã không được nhận. ở chế độ 0, SM2 nên bằng 0

* REN: Cho phép nhận nối tiếp. Được đặt hoặc xoá bởi phần mềm để cho phép hoặc không cho phép nhận.

* TB8: Là bit dữ liệu thứ 9 mà sẽ được truyền ở Mode 2 và 3. Được đặt hoặc xoá bởi phần mềm.

* RB8: Là bit dữ liệu thứ 9 đã được nhận ở Mode 2 và 3. Ở Mode 1, nếu SM2=0 thì RB8 là bit dừng đã được nhận. Ở Mode 0, RB8 không được sử dụng.

* TI: Cờ ngắt truyền. Được đặt bởi phần cứng tại cuối thời điểm của bit thứ 8 trong Mode 0, hoặc đầu thời điểm của bit dừng trong các Mode khác. Ở bất kỳ quá trình truyền nối tiếp nào, nó cũng phải được xoá bằng phần mềm.

* RI: Cờ ngắt nhận. Được đặt bởi phần cứng tại cuối thời điểm của bit thứ 8 trong Mode 0, hoặc ở giữa thời điểm của bit dừng trong các Mode khác. Ở bất kỳ quá trình nhận nối tiếp nào (trừ trường hợp ngoại lệ, xem SM2), nó cũng phải được xoá bằng phần mềm.

2.3. Khởi tạo thời gian và bộ đếm (Timer/Counter).

On-chip AT89C51 có 2 thanh ghi Timer/Counter dài 16 bit, đó là: Timer 0 và Timer 1. Trong On-chip AT89C52, ngoài Timer 0 và Timer 1 nó còn có thêm bộ Timer 2. Cả 3 bộ Timer này đều có thể được điều khiển để thực hiện chức năng thời gian hay bộ đếm, thông qua thanh ghi TMOD.

Khi thanh ghi Timer/Counter làm việc ở kiểu Timer, thì sau mỗi chu kỳ máy nội dung trong thanh ghi được gia tăng thêm 1 đơn vị. Vì vậy thanh ghi này đếm số chu kỳ máy. Một chu kỳ máy có 12 chu kỳ dao động, do đó tốc độ đếm của thanh ghi là 1/12 tần số dao động.

Khi thanh ghi Timer/Counter làm việc ở kiểu Counter, xung nhịp bên ngoài được đưa vào để đếm ở T0 hoặc T1. Nội dung thanh ghi được tăng lên khi có sự chuyển trạng thái từ 1 về 0 tại chân đầu vào ngoài T0 hoặc T1. Xung nhịp ở các đầu vào ngoài được lấy mẫu tại thời điểm S5P2 của mỗi chu kỳ máy. Khi quá trình lấy mẫu phát hiện ra mức cao ở 1 chu kỳ và mức thấp ở chu kỳ tiếp theo, thì bộ đếm được tăng lên. Giá trị mới của bộ đếm xuất hiện trong thanh ghi tại thời điểm S3P1 của chu kỳ máy sau khi sự chuyển trạng thái đã được phát hiện. Vì vậy để nội dung của thanh ghi tăng lên 1 đơn vị phải mất 2 chu kỳ máy, nên tốc độ đếm tối đa là 1/24 tần số bộ dao động. Không có sự giới hạn số vòng thực hiện của tín hiệu ở đầu vào ngoài, nhưng nó sẽ giữ ít nhất 1 chu kỳ máy đầy đủ

để đảm bảo chắc chắn rằng một mức đã cho được lấy mẫu ít nhất 1 lần nữa trước khi nó thay đổi.

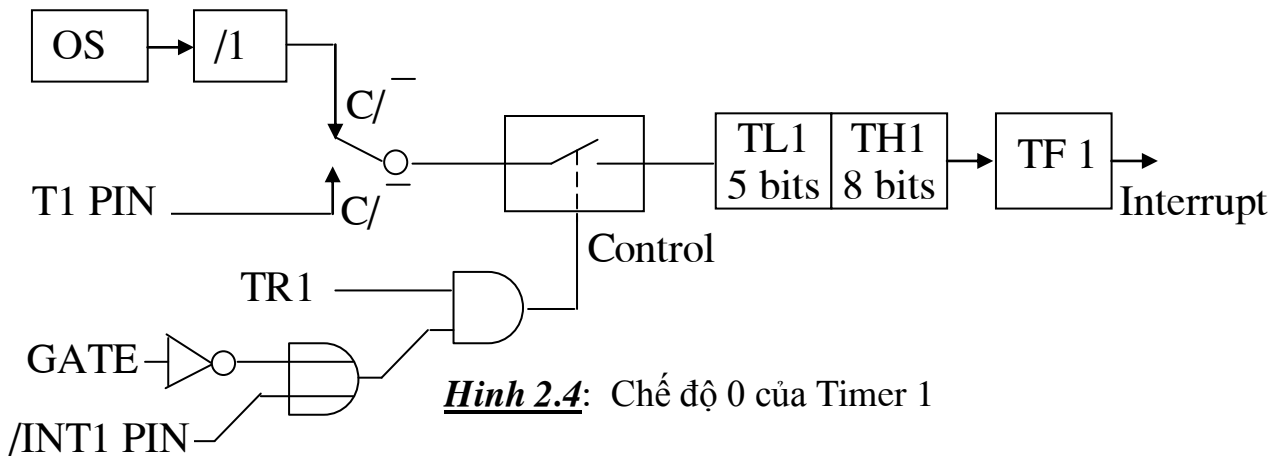
Do xung nhịp bên ngoài có tần số bất kỳ nên các bộ Timer (0 và 1) có 4 chế độ làm việc khác nhau để lựa chọn: (13 bit Timer, 16 bit Timer, 8 bit auto-reload, split Timer).

Timer 0 và Timer 1:

Trong AT89C51 và AT89C52 đều có các bộ Timer 0 và 1. Chức năng Timer hay Counter được chọn lựa bởi các bit điều khiển C/(/T) trong thanh ghi TMOD. Hai bộ Timer/Counter này có 4 chế độ hoạt động, được lựa chọn bởi cặp bit (M0, M1) trong TMOD. Chế độ 0, 1 và 2 giống nhau cho các chức năng Timer/Counter, nhưng chế độ 3 thì khác. Bốn chế độ hoạt động được mô tả như sau:

+ **Chế độ 0:** Cả 2 bộ Timer 0 và 1 ở chế độ 0 có cấu hình như một thanh ghi 13 bit, bao gồm 8 bit của thanh ghi THx và 5 bit thấp của TLx. 3 bit cao của TLx không xác định chắc chắn, nên được làm ngơ. Khi thanh ghi được xoá về 0, thì cờ ngắt thời gian TFX được thiết lập. Bộ Timer/Counter hoạt động khi bit điều khiển TRx được thiết lập (TRx=1) và, hoặc Gate trong TMOD bằng 0, hoặc /INTx=1. Nếu đặt GATE=1 thì cho phép điều khiển Timer/ Counter bằng đường vào ngoài /INTx, để dễ dàng xác định độ rộng xung.

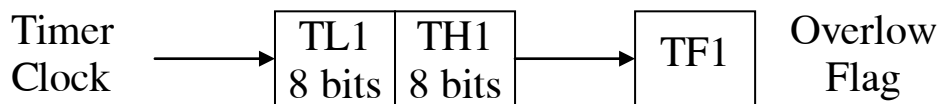
Khi hoạt động ở chức năng thời gian thì bit C/(/T)=0, do vậy xung nhịp từ bộ dao động nội, qua bộ chia tần cho ra tần số $f=f_{osc}/12$ được đưa vào để đếm trong



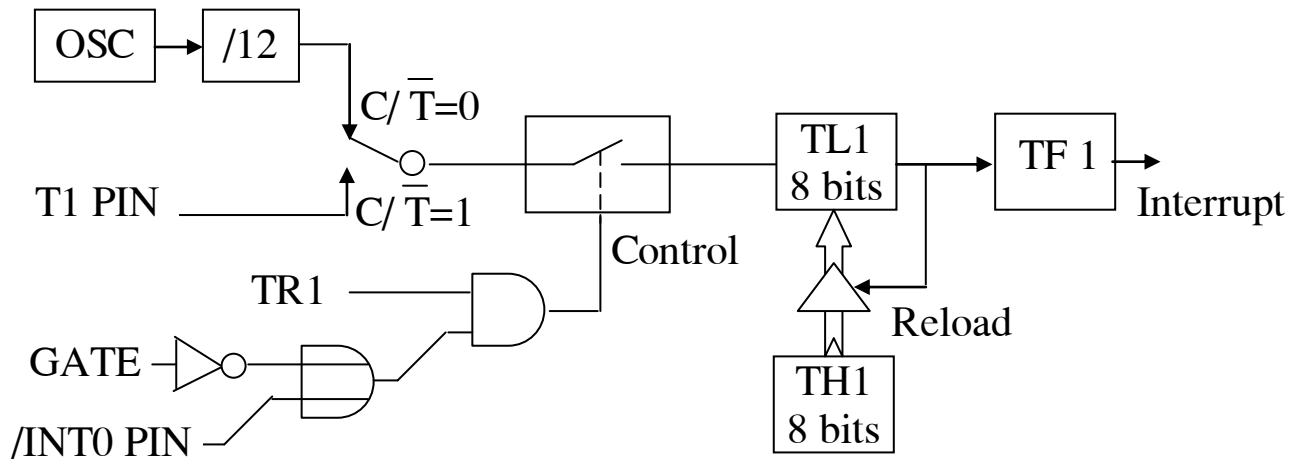
Hình 2.4: Chế độ 0 của Timer 1

thanh ghi Timer/Counter. Khi hoạt động ở chức năng bộ đếm thì bit C/(/T)=1, lúc đó xung nhịp ngoài đưa vào sẽ được đếm.

+ **Chế độ 1:** hoạt động tương tự như chế độ 0, chỉ khác là thanh ghi Timer/Counter được sử dụng cả 16 bit. Xung nhịp được dùng kết hợp với các thanh ghi thời gian byte thấp và byte cao (TH1 và TL1). Khi xung Clock được nhận, bộ Timer sẽ đếm tăng lên: 0000h, 0001h, 0002,... Khi hiện tượng tràn xảy ra, cờ tràn sẽ chuyển FFFFh về 0000h, và bộ Timer tiếp tục đếm. Cờ tràn của Timer 1 là bit TF1 ở trong TCON, nó được đọc hoặc ghi bởi phần mềm, xem hình 2.5 (Timer/Counter 1 Mode 1: 16 bit Counter).



Hình 2.5: Chế độ 1 của Timer 1

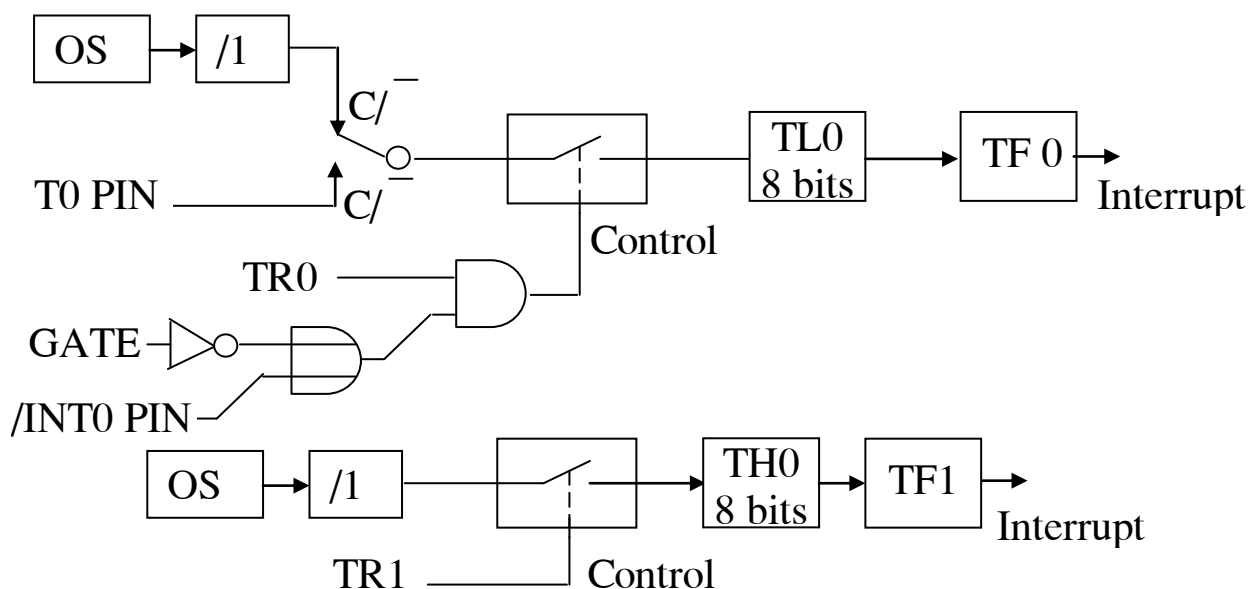


Hình 2.6: Chế độ 2 của Timer 1

+ **Chế độ 2:** Chế độ này của thanh ghi Timer cũng hoạt động tương tự như 2 chế độ trên, nhưng nó được tổ chức như bộ đếm 8 bit (TL1) với chế độ tự động nạp lại, như hình 2.6. Khi xảy ra hiện tượng tràn ở TL1, không chỉ thiết lập bit TF1 mà còn tự động nạp lại cho TL1 bằng nội dung của TH1, đã được thiết lập bởi phần mềm. Quá trình nạp lại cho phép nội dung của TH1 không bị thay đổi. Chế độ 2 của Timer/Counter 0 cũng tương tự như Timer/Counter 1.

+ **Chế độ 3:** Ở chế độ này, chức năng Timer/Counter 0 và chức năng Timer/Counter 1 khác nhau. Bộ Timer 1 ở chế độ 3 chỉ chứa chức năng đếm của nó, kết quả giống khi đặt TR1=0. Bộ Timer 0 ở chế độ 3 thiết lập TH0, TL0 như là 2 bộ đếm riêng biệt. Mạch Logic đối với chế độ 3 của Timer 0 thể hiện ở hình 2.7. Bộ đếm TL0 được điều khiển bởi các bit: C/(/T), GATE, TR0, /INT0 và khi đếm tràn nó thiết lập cờ ngắt TF0. Bộ đếm TH0 chỉ được điều khiển bởi bit TR1, và khi đếm tràn nó thiết lập cờ ngắt TF1. Vậy, TH0 điều khiển ngắt Timer/Counter 1.

Chế độ 3 thường được dùng khi yêu cầu cần có bộ thời gian hoặc bộ đếm ngoài 8 bit. Đối với Timer 0 ở chế độ 3, AT89C51 có thể có 3 bộ Timer/Counter, còn AT89C52 có thể có 4 bộ. Khi Timer 0 hoạt động ở chế độ 3, thì Timer 1 có thể được bật hoặc tắt bằng chuyển mạch ngoài. Ở chế độ này, Timer 1 có thể được sử dụng bởi cổng nối tiếp như một bộ tạo tốc độ Baud, hoặc trong bất kỳ ứng dụng nào mà không yêu cầu một ngắt.



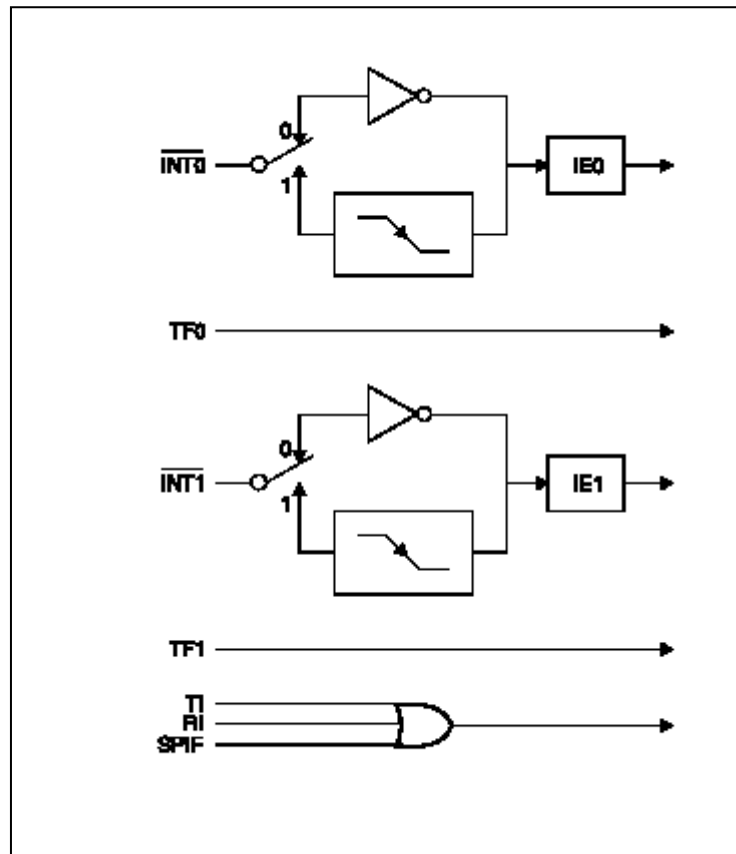
Hình 2.7: Chế độ 3 của Timer 0

2.4. Cơ chế ngắt trong On-chip AT89C51:

2.4.1. Phân loại ngắt trong On-chip:

Bộ AT89C51 có tất cả 5 Vectors ngắt bao gồm: 2 ngắt ngoài ($\overline{\text{INT0}}$ và $\overline{\text{INT1}}$), 2 ngắt của khối thời gian (Timer 0, 1), và ngắt cổng truyền tin nối tiếp.

Mỗi nguồn ngắt có thể được kích hoạt hoặc không kích hoạt bằng cách đặt hoặc xoá Bit ở trong IE. IE cũng chứa bit có thể không cho tất cả các ngắt hoạt động EA (Nếu $\text{EA}=0$). Các ngắt ngoài có thể được kích hoạt theo mức hoặc theo sườn xung, tùy thuộc vào giá trị của các bit IT0 , IT1 trong TCON. Ngắt ngoài có 2 cờ ngắt tương ứng là IE0 , IE1 cũng nằm trong TCON. Khi một ngắt được thực hiện thì cờ ngắt tương ứng của nó bị xoá bằng phần cứng. Chương trình con phục vụ ngắt hoạt động chỉ khi ngắt được kích hoạt theo sườn xung. Nếu ngắt được kích hoạt theo mức thì nguồn yêu cầu ngắt từ bên ngoài điều khiển cờ ngắt.



Hình 2.17. Các nguồn ngắt của AT89C51

Các ngắt trong, với ngắt Timer/Counter 0, 1 được phát sinh bởi cờ ngắt TF0, TF1. Hai cờ ngắt này được thiết lập khi thanh ghi Timer/Counter thực hiện quay vòng, tại thời điểm S5P2 của chu trình máy. Khi một ngắt được thực hiện thì cờ ngắt tương ứng phát sinh ra ngắt sẽ bị xoá bằng phần cứng trong On-chip. Ngắt cổng nối tiếp được phát sinh bởi các ngắt RI, TI, SPIF thông qua phân tử Logic OR, khi chương trình con phục vụ ngắt được kích hoạt thì các cờ ngắt phát sinh tương ứng được xoá bằng phần mềm. Các ngắt trong có thể được phép hoặc không được phép kích hoạt bằng cách đặt hoặc xoá một bit trong IE.

2.4.2. Các bước thực hiện ngắt.

Theo đúng trình tự, để sử dụng các ngắt trong Flash Microcontroller, cần thực hiện các bước như sau:

- Đặt bit EA ở trong IE mức logic 1.
- Đặt bit cho phép ngắt tương ứng ở trong IE mức logic 1.
- Bắt đầu chương trình con phục vụ ngắt tại địa chỉ của ngắt tương ứng đó.

(Xem bảng địa chỉ Vector của các nguồn ngắt)

Ngoài ra, đối với các ngắt ngoài, các chân /INT0, /INT1 phải được đặt mức 1. Và tùy thuộc vào ngắt được kích hoạt bằng mức hay sườn xung, mà các bit IT0, IT1 ở trong TCON có thể cần phải đặt mức 1.

ITx=0: Kích hoạt bằng mức

ITx=1: Kích hoạt bằng sườn xung.

2.4.3. Mức ngắt ưu tiên trong on-chip:

Mỗi nguồn ngắt có thể được lập trình riêng cho 1 hoặc 2 mức ưu tiên bằng cách đặt hoặc xoá 1 bit trong IP của SFR. Mỗi ngắt ưu tiên ở mức thấp có thể được ngắt bằng ngắt ưu tiên ở mức cao hơn nhưng không thể ngắt bằng ngắt có mức ưu tiên ở mức thấp hơn được. Một ngắt ưu tiên ở mức cao có thể được ngắt bởi bất kỳ nguồn ngắt nào khác.

Nếu có yêu cầu ngắt của 2 mức ưu tiên cùng nhau (cùng 1 lúc), yêu cầu của mức ưu tiên cao hơn sẽ được phục vụ (Ngắt nào có mức ưu tiên cao hơn sẽ được phục vụ). Nếu các yêu cầu ngắt có cùng mức ưu tiên, thì thứ tự quay vòng bên trong sẽ quyết định ngắt nào được phục vụ.

Thứ tự ưu tiên ngắt từ cao xuống thấp của AT89C51 như sau:

IE0, TF0, IE1, TF1, RI hoặc TI.

2.4.4. Nguyên lý điều khiển ngắt của AT89:

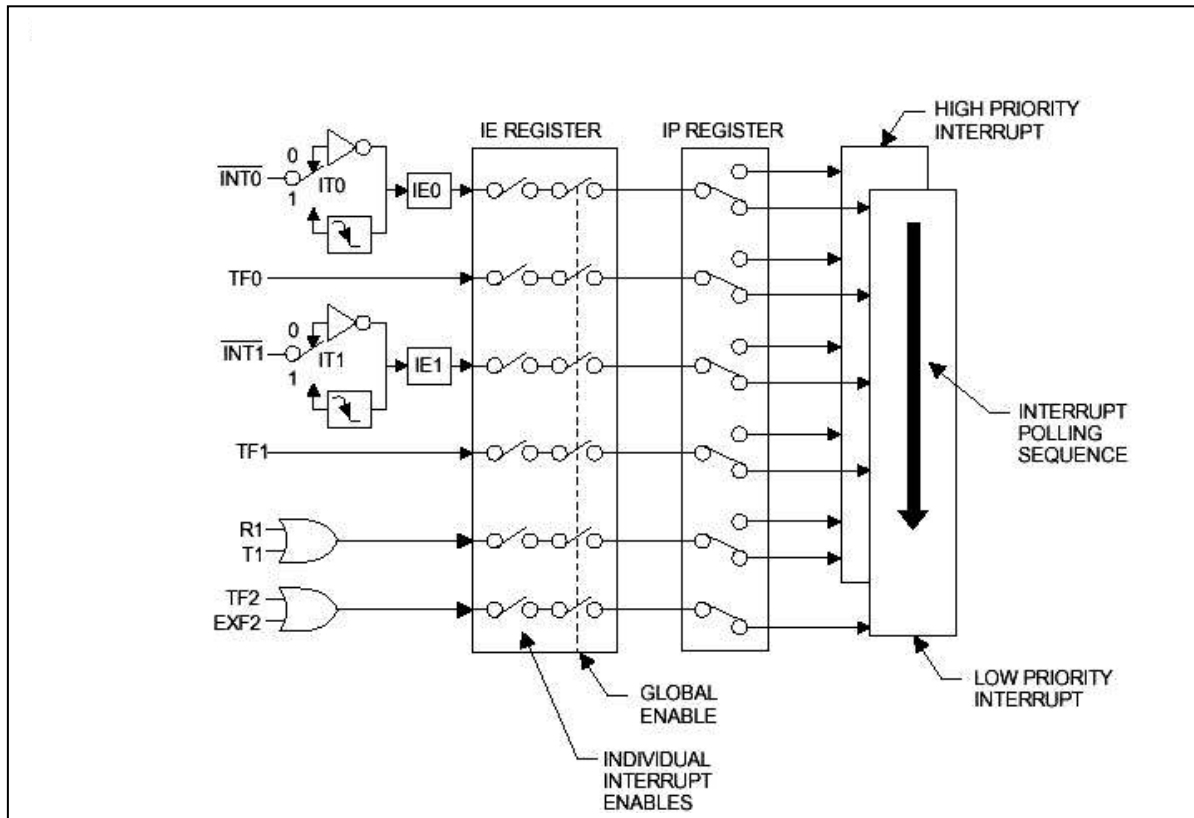
Các cờ ngắt được thiết lập tại thời điểm S5P2 của mỗi chu kỳ máy. Chu kỳ máy tiếp theo sau chu kỳ máy có cờ ngắt được thiết lập, thì chương trình con được thiết lập khi có lệnh gọi LCALL. Lệnh LCALL phát sinh nhưng lại bị cấm hoạt động khi gặp các tình huống sau:

a- Đồng thời có ngắt với mức ưu tiên cao hơn hoặc bằng ngắt đang phục vụ.

(Một ngắt có mức ưu tiên bằng hoặc cao hơn đang sẵn sàng để được phục vụ)

b- Chu kỳ máy hiện hành không phải là chu kỳ máy cuối cùng của lệnh đang thực hiện.

c- Lệnh đang thực hiện là RETI hoặc bất kỳ lệnh nào ghi vào thanh ghi IE hoặc IP.



Hình 2.18. Hệ thống ngắt của AT89

Bất kỳ một trong 3 điều kiện này xuất hiện sẽ cản trở việc tạo ra LCALL đối với chương trình phục vụ ngắt. Điều kiện 2 đảm bảo rằng, lệnh đang thực hiện sẽ được hoàn thành trước khi trở tới bất kỳ chương trình phục vụ nào. Điều kiện 3 đảm bảo rằng, nếu lệnh đang thực hiện là RETI hoặc bất kỳ sự truy cập nào vào IE hoặc IP, thì ít nhất một lệnh nữa sẽ được thực hiện trước khi bất kỳ ngắt nào được trở tới. Chu trình kiểm tra vòng được lặp lại với mỗi chu trình máy, và các giá trị được kiểm tra là các giá trị mà đã xuất hiện ở thời điểm S5P2 của chu trình máy trước đó. Nếu một chỉ thị ngắt có hiệu lực nhưng không được đáp ứng vì các điều kiện trên và nếu chỉ thị này vẫn chưa có hiệu lực khi điều kiện cản trở được loại bỏ, thì ngắt bị từ chối này sẽ không được phục vụ nữa.

LCALL do phần cứng tạo ra sẽ chuyển nội dung của bộ đếm chương trình vào ngăn xếp (Nhưng không ghi vào PSW) và nạp lại cho PC một địa chỉ phụ thuộc vào nguồn gây ngắt đang được phục vụ, như bảng dưới đây

Ngắt	Nguồn ngắt	Địa chỉ Véc tơ
External 0	IE0	0003h
Timer 0	TF0	000Bh
External 1	IE1	0013h
Timer 1	TF1	001Bh
Serial Port	RI hoặc TI	0023h
Timer 2(AT89C52)	TF2 hoặc EXF2	002Bh
System Reset	RST	0000h

Bảng 2.7. Địa chỉ véc tơ ngắt

Lệnh RETI thông báo cho bộ VXL rằng thủ tục ngắt này đã kết thúc, sau đó lấy ra 2 Byte từ ngăn xếp và nạp lại cho PC để trả lại quyền điều khiển cho chương trình chính.

* *.Các ngắt ngoài:*

Vì các chốt ngắt ngoài được tạo mẫu mỗi lần trong mỗi chu trình máy, nên một giá trị cao hoặc thấp của đầu vào sẽ duy trì trong ít nhất là 12 chu kỳ xung nhịp của bộ dao động để đảm bảo tạo mẫu. Nếu ngắt ngoài được kích hoạt bằng sườn xung, thì nguồn ngắt ngoài phải duy trì ở chốt yêu cầu giá trị cao ít nhất 1 chu kỳ máy và sau đó duy trì giá trị thấp ít nhất 1 chu kỳ máy nữa. Việc này được thực hiện để đảm bảo rằng quá trình chuyển tiếp cho thấy chỉ thị yêu cầu ngắt IEx sẽ được xác lập. IEx sẽ tự động được xoá bởi CPU khi thủ tục ngắt đáp ứng được gọi.

Nếu ngắt ngoài được kích hoạt theo mức, thì nguồn ngắt bên ngoài phải duy trì cho yêu cầu này có hiệu lực cho đến khi ngắt đã được yêu cầu thực sự được tạo ra. Sau đó nguồn ngắt ngoài phải huỷ yêu cầu đó trước khi thủ tục phục vụ ngắt hoàn thành, nếu không ngắt khác sẽ được tạo ra.

*. Vận hành Single-Step:

Cấu trúc ngắt AT89C51 cho phép thực hiện các bước đơn với sự tham gia của rất ít phần mềm. Như đã lưu ý trước đây, một yêu cầu ngắt sẽ không được đáp ứng khi một ngắt khác có cùng mức ưu tiên vẫn đang hoạt động, nó cũng không được đáp ứng sau khi có lệnh RETI cho đến khi có ít nhất một lệnh khác đã được thực hiện. Do đó mỗi khi một thủ tục ngắt được đưa vào, thì nó không thể được đưa vào lần nữa cho đến khi ít nhất một lệnh của chương trình ngắt được thực hiện. Một cách để sử dụng đặc điểm này đối với hoạt động theo bước đơn là lập trình cho 1 trong những ngắt ngoài(chẳng hạn /INT0) được kích hoạt theo mức.

Nếu chân /INT0 được duy trì ở mức thấp, thì CPU sẽ chuyển ngay đến thủ tục ngắt ngoài 0 và dừng ở đó cho tới khi INT0 được nhận xung từ thấp lên cao rồi xuống thấp. Sau đó nó sẽ thực hiện lệnh RETI, trở lại nhiệm vụ chương trình, thực hiện một lệnh, và ngay sau đó nhập lại thủ tục ngắt ngoài 0 để đợi xung nhịp tiếp theo của P3.2. Mỗi bước của nhiệm vụ chương trình được thực hiện vào mỗi thời điểm chân P3.2 được nhận xung.

2.5. Bảo vệ chương trình.

Họ VDK	Các bit khoá
AT89C51	LB1, LB2, LB3
AT89C52	LB1, LB2, LB3
AT89C2051	LB1, LB2
AT89C1051	LB1, LB2

Khoá bộ nhớ chương trình cho họ VDK AT89C51:

Chế độ	LB1	LB2	LB3	Loại bảo vệ
1	U	U	U	Không có đặc trưng khoá chương trình.
2	P	U	U	Các lệnh MOVC được thực thi từ bộ nhớ chương trình ngoài, không được phép tìm nạp lệnh từ bộ nhớ nội. EA được lấy mẫu và chốt khi reset. Việc lập trình trên Flash bị cấm.
3	P	P	U	Như chế độ 2, ngoài ra còn cấm việc kiểm tra chương trình.
4	P	P	P	Như chế độ 3, ngoài ra còn cấm việc thực thi chương trình ngoài.

Lưu ý: P=Programmed, U = Unprogrammed

2.6. Tra cứu nhanh tập lệnh của 8051

Bảng 1 : Các lệnh toán học của bộ VDK họ ATMEN:

Câu lệnh	Chức năng	Các kiểu định địa chỉ				Thời gian Thực hiện(us)
		Trực tiếp	Gián tiếp	Thanh ghi	Tức thời	
ADD A,<byte>	$A = A + \langle \text{byte} \rangle$	x	x	x	x	1
ADDC A,<byte>	$A = A + \langle \text{byte} \rangle + C$	x	x	x	x	1
SUBB A,<byte>	$A = A - \langle \text{byte} \rangle$	x	x	x	x	1
INC A	$A = A + 1$	Chỉ với A				1
INC <byte>	$\langle \text{byte} \rangle = \langle \text{byte} \rangle + 1$	x	x	x		1
INC DPTR	$DPTR = DPTR + 1$	Chỉ với DPTR				2
DEC A	$A = A - 1$	Chỉ với A				1
DEC <byte>	$\langle \text{byte} \rangle = \langle \text{byte} \rangle - 1$	x	x	x		1
MUL AB	$BA = A * B$	Chỉ với A&B				4
DIV AB	$A = \text{Int}(A/B);$ $B = \text{Mode}(A/B)$	Chỉ với A&B				4
DA A	Hiệu chỉnh số thập phân	Chỉ với A				1

Bảng 2. Các lệnh chuyển đổi dữ liệu để truy cập vùng nhớ dữ liệu trong:

Câu lệnh	Chức năng	Các kiểu định địa chỉ				Thời gian Thực hiện(us)
		Trực tiếp	Gián tiếp	Thanh ghi	Tức thời	
MOV A,<scr>	$A = \langle \text{scr} \rangle$	x	x	x	x	1
MOV <dest>,A	$\langle \text{dest} \rangle = A$	x	x	x		1
MOV <dest>,<scr>	$\langle \text{dest} \rangle = \langle \text{scr} \rangle$	x	x	x	x	2
MOV <DPTR>,#data16	$DPTR = \text{h/số tức thời } 16 \text{ bit}$				x	2
PUSH <scr>	INC SP; Mov “@SP“, <scr>	x				2
POP <dest>	Mov <dest>,”@SP“ ;DEC SP	x				2
XCH a,<byte>	Đổi dữ liệu giữa A&byte	x	x	x		1
XCHD A,@Ri	Đổi nửa bit thấp giữa A&@Ri		x			1

Bảng 3. Các lệnh số học:

Câu lệnh	Chức năng	Các kiểu định địa chỉ				Thời gian Thực hiện(us)
		Trực tiếp	Gián tiếp	Thanh ghi	Tức thời	
ANL A,<byte>	A=A AND <byte>	x	x	x	x	1
ANL <byte>,A	<byte>= <byte> AND A	x				1
ANL <byte>,#data	<byte>= <byte> AND #data	x				2
ORL A, <byte>	A=A OR <byte>	x	x	x	x	1
ORL <byte>,A	<byte>= <byte> OR A	x				1
ORL <byte>,#data	<byte>= <byte> OR #data	x				2
XRL A, <byte>	A=A XOR <byte>	x	x	x	x	1
XRL <byte>,A	<byte>= <byte> XOR A	x				1
XRL <byte>,#data	<byte>= <byte> XOR #data	x				2
CLR A	A = 00h	Chỉ với A				1
CPL A	A = NOT A	Chỉ với A				1
RL A	Dịch A sang trái 1 bit	Chỉ với A				1
RLC A	Dịch A sang trái thông qua C	Chỉ với A				1
RR A	Dịch A sang phải 1 bit	Chỉ với A				1
RRC A	Dịch A sang phải thông qua C	Chỉ với A				1
SWAP A	Đổi nửa bit trong A	Chỉ với A				1

Bảng 4. Các lệnh đại số:

Câu lệnh	Chức năng	Thời gian Thực hiện(us)
ANL C,bit	$C = C \text{ AND bit}$	2
ANL C,/bit	$C = C \text{ AND NOT bit}$	2
ORL C,bit	$C = C \text{ ORL bit}$	2
ORL C,/bit	$C = C \text{ ORL NOT bit}$	2
MOV C,bit	$C = \text{bit}$	1
MOV bit,C	$\text{Bit} = C$	2
CLR C	$C = 0$	1
CLR bit	$\text{Bit} = 0$	1
SETB C	$C = 1$	1
SETB bit	$\text{Bit} = 1$	1
CPL C	$C = \text{NOT } C$	1
CPL bit	$\text{Bit} = \text{NOT bit}$	1
JC rel	Nhảy nếu $C = 1$	2
JNC rel	Nhảy nếu $C = 0$	2
JB bit,rel	Nhảy nếu bit = 1	2
JNB bit,rel	Nhảy nếu bit = 0	2
JBC bit,rel	Nhảy nếu bit=1, sau đó xoá bit	2

Bảng 5. Các lệnh chuyển đổi dữ liệu để truy cập RAM ngoài:

Độ lớn địa chỉ	Câu lệnh	Chức năng	Th.gian thực hiện (us)
8 bits	MOVX A,@Ri	Đọc RAM ngoài tại @Ri	2
8 bits	MOVX @Ri,A	Ghi vào RAM ngoài tại @Ri	2
16 bits	MOVX A,@DPTR	Đọc RAM ngoài tại @DPTR	2
16 bits	MOVX @DPTR,A	Ghi vào RAM ngoài tại @DPTR	2

Bảng 6. Các lệnh chuyển Byte mã lệnh:

Câu lệnh	Chức năng	Thời gian thực hiện
MOVC A,@A+DPTR	Đọc ROM tại (A+DPTR)	2 us
MOVC A,@A+PC	Đọc ROM tại (A+PC)	2 us

Bảng 7. Các lệnh nhảy không điều kiện trong Flash Microcontrollers:

Câu lệnh	Chức năng	Thời gian thực hiện(us)
JMP addr	Nhảy tới addr.	2
JMP @A+DPTR	Nhảy tới A+DPTR.	2
CALL addr	Gọi C.trình con tại addr.	2
RET	Quay trở về từ C.trình con.	2
RETI	Quay trở về từ ngắt.	2

Bảng 8. Các lệnh nhảy có điều kiện:

Câu lệnh	Chức năng	Các kiểu định địa chỉ				Thời gian Thực hiện(us)
		Trực tiếp	Gián tiếp	Th/ghi	Tức thời	
JZ rel	Nhảy nếu A=0	Chỉ với A				2
JNZ rel	Nhảy nếu A≠0	Chỉ với A				2
DJNZ <byte>,rel	Giảm & nhảy nếu ≠ 0	x		x		2
CJNE A,<byte>,rel	Nhảy nếu A ≠ <byte>	x			x	2
CJNE <byte>,#data,rel	Nhảy nếu <byte> ≠ #data		x	x		2

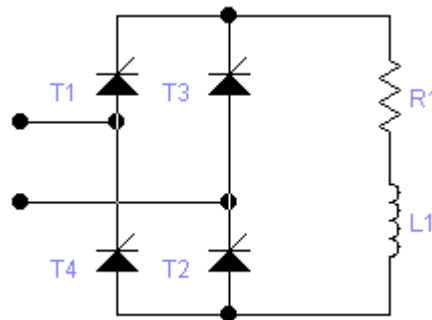
CHƯƠNG 2. THIẾT KẾ VÀ THI CÔNG PHẦN CỨNG

A. Giới thiệu phần tử

2.1. Chỉnh lưu cầu một pha

Mục đích: Chỉnh lưu dòng xoay chiều thành dòng 1 chiều. Lấy dòng 1 chiều hoạt động cho IC nguồn 7805

Sơ đồ mạch:



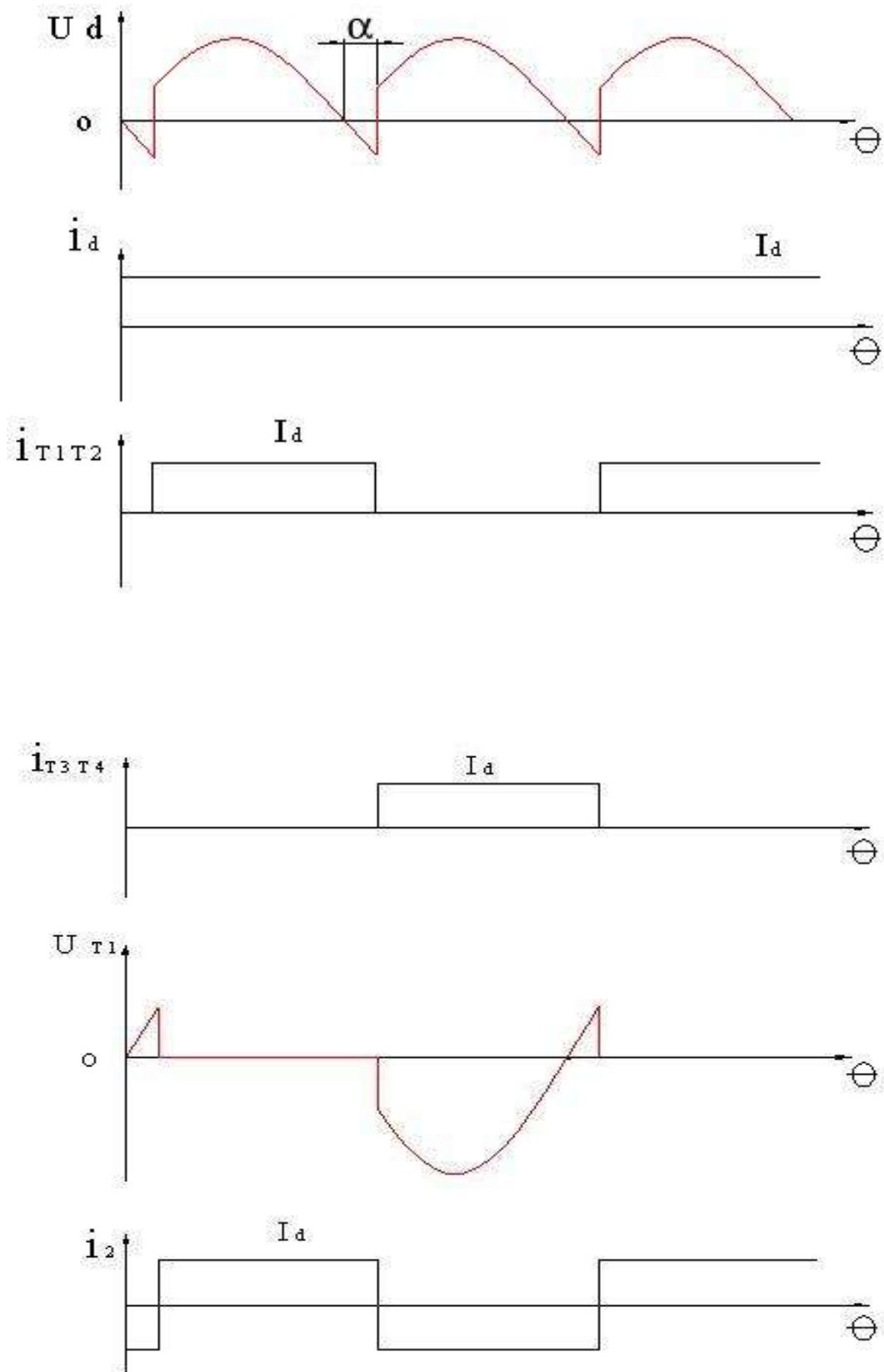
Xét với tải RL, điện cảm L đủ lớn để lọc dòng, áp, dòng là liên tục.

- Mạch có T₁, T₃ chung Katot

T₂, T₄ chung Anot

Nửa chu kỳ đầu $U_2 > 0$, Anot của T₁ dương, Katot của T₂ âm. Nếu có xung điều khiển mở đồng thời T₁ và T₂ thì cả hai van này được mở thông và đặt điện áp lưới lên tải $U_d = U_2$. Điện áp tải một chiều bằng điện áp xoay chiều ($U_d = U_2$) cho đến khi nào T₁, T₂ còn dẫn. (Khoảng dẫn của các van phụ thuộc vào tải).

Nửa chu kỳ sau, điện áp đổi dấu, anot của T₃ dương và katot T₄ âm, nếu có xung điều khiển mở đồng thời T₃, T₄ thì các van này được mở thông và $U_d = -U_2$, với điện áp một chiều có cùng chiều với nửa chu kỳ trước



Hình 3.1. Các đặc tính điện áp tải, dòng qua tải, dòng qua van và điện áp ngược trên van

+ Giá trị trung bình của tải

$$U_d = \frac{2}{2\pi} \int_{\alpha}^{\pi+\alpha} \sqrt{2} U_2 \sin\theta d\theta = \frac{2\sqrt{2}}{2} U_2 \cos\alpha = U_{do} \cos\alpha.$$

U_{do} : điện áp trung bình tải trong chỉnh lưu cầu không điều khiển .

+ Dòng qua máy biến áp cũng bằng dòng qua van (khi van mở) .

+ Giá trị hiệu dụng dòng thứ cấp biến áp

$$I_2 = \sqrt{\frac{2}{2\pi} \int_{\alpha}^{\pi+\alpha} I_d^2 \theta} = I_d$$

+ Điện áp ngược lớn nhất van phải chịu

$$U_{nv} = \sqrt{2} U_2$$

+ Dòng làm việc của van tính theo giá trị trung bình

$$I_T = \sqrt{\frac{1}{2\pi} \int_{\alpha}^{\pi+\alpha} I_d^2 \theta} = \frac{I_d}{\sqrt{2}} = 0.71 I_d$$

+ $K_{dm} = 0,67$

+ Công suất biến áp $S_{ba} = 1,23P_d$

* Nhận xét :

Chất lượng điện áp của chỉnh lưu cầu một pha giống như chỉnh lưu hình tia .

Hình dạng đường cong điện áp , dòng tải , dòng qua van bán dẫn , hệ số đập mạch như trong chỉnh lưu hình tia .

❖ Ưu điểm :

- Điện áp ngược trên van nhỏ hơn một nửa so với chỉnh lưu hình tia

$$U_{nv} = \sqrt{2} U_2.$$

- Máy biến áp chế tạo đơn giản hơn , và có hiệu suất cao hơn so với chỉnh lưu hình tia .

❖ Nhược điểm :

- Số van nhiều hơn

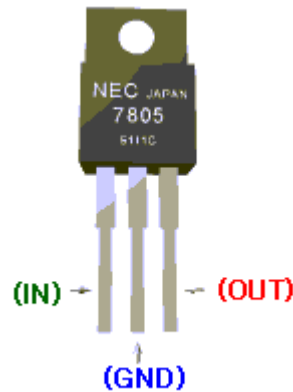
- Điều khiển van T_1, T_2 và nhóm T_3, T_4 phải đồng thời nên khó khăn hơn

❖ Mục đích:

Đổi nguồn điện từ xoay chiều thành một chiều cung cấp cho mạch hoạt động

2.2. IC tạo ổn áp 7805:(IC ổn áp 5v).

Với những mạch điện không đòi hỏi độ ổn định của điện áp quá cao, sử dụng IC ổn áp thường được người thiết kế sử dụng vì mạch điện khá đơn giản. Các loại ổn áp thường được sử dụng là IC 78xx, với xx là điện áp cần ổn áp. Ví dụ 7805 ổn áp 5V, 7812 ổn áp 12V. Việc dùng các loại IC ổn áp 78xx tương tự nhau, dưới đây là minh họa cho IC ổn áp 7805:



Sơ đồ phía dưới IC 7805 có 3 chân:

- * Chân số 1 là chân IN.
- * Chân số 2 là chân GND.
- * Chân số 3 là chân OUT.

Ngõ ra OUT luôn ổn định ở 5V dù điện áp từ nguồn cung cấp thay đổi. Mạch này dùng để bảo vệ những mạch điện chỉ hoạt động ở điện áp 5V (các loại IC thường hoạt động ở điện áp này). Nếu nguồn điện có sự cố đột ngột: điện áp tăng cao thì mạch điện vẫn hoạt động ổn định nhờ có IC 7805 vẫn giữ được điện áp ở ngõ ra OUT 5V không đổi.

Mạch trên lấy nguồn một chiều từ một máy biến áp với điện áp từ 7V đến 9V để đưa vào ngõ IN. Khi kết nối mạch điện, do nhiều nguyên nhân, người dùng dễ nhầm lẫn cực tính của nguồn cung cấp khi đấu nối vào mạch, trong trường hợp này rất dễ ảnh hưởng đến các linh kiện trên board mạch. Vì lí do đó một diode cầu được lắp thêm vào mạch, diode cầu đảm bảo cực tính của nguồn cấp cho mạch theo một chiều duy nhất, và người dùng cũng không cần quan tâm đến cực tính của nguồn khi nối vào ngõ IN nữa.

2.3. Ghép nối mạch cầu H và mạch khuếch đại

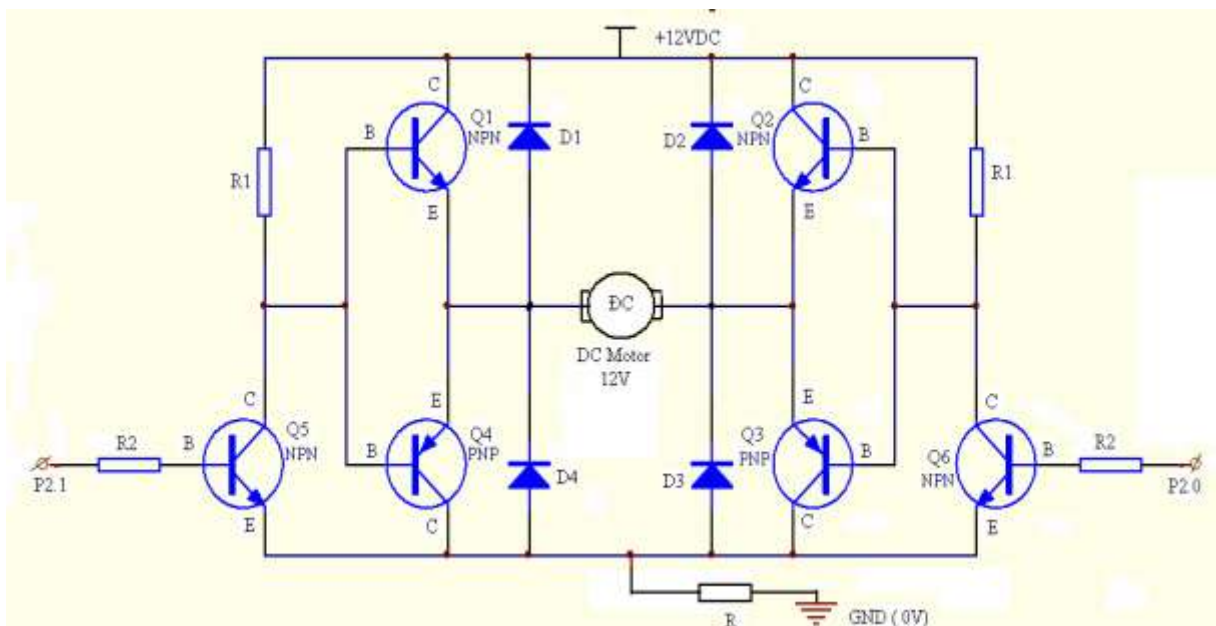
Thiết kế mạch cầu H điều khiển, đảo chiều động cơ DC.

Để điều khiển truyền động cho cửa tự động ta dùng động cơ điện một chiều có điện áp định mức thường là loại 12VDC – 24VDC. Ở đồ án này em chọn động cơ có điện áp định mức là 12VDC, với công suất định mức $P = 24W$.

Từ các yêu cầu khi thiết kế cửa tự động đã trình bày ở mục 2.1.2. Khái quát chung về cửa tự động nên thiết kế mạch động lực thì phải đảm bảo điều khiển được động cơ một chiều quay theo cả hai chiều: chiều thuận, chiều ngược và dừng chính xác, có thể điều khiển và kiểm soát được tốc độ động cơ.

Có nhiều phương án thiết kế để điều khiển, khởi động và đảo chiều động cơ điện một chiều. Đó là dùng Relay – Contactor có đảo chiều, dùng bộ chỉnh lưu cầu Tiristo mắc song song ngược thường được ứng dụng ở các tải có công suất lớn hoặc dùng mạch cầu H để điều khiển động cơ một chiều... Ở đây vì tải có công suất nhỏ nên em chọn sử dụng động cơ DC điều khiển bởi mạch cầu H.

❖ Mạch cầu H điều khiển đảo chiều động cơ một chiều:



Hình 3.1. Mạch cầu H điều khiển động cơ một chiều.

- **Mạch gồm:**

- + 4 Transistor công suất Q1, Q2, Q3, Q4. Q1, Q2 là loại Transistor NPN TIP41, Q3, Q4 là loại Transistor PNP TIP42, chịu được áp 40V và dòng lớn lên tới 10A.

- + 2 Transistor điều khiển Q5, Q6 là loại NPN C8283.

- + 2 trở R_1 .

- + 2 trở R_2 .

- + 4 Điốt (D1 ÷ D4) 2N4007.

- + Động cơ một chiều 12VDC, 24W.

- + Nguồn điện cấp là 12VDC.

- **Tính toán công suất:**

Dòng và áp của động cơ điện một chiều làm việc ở các chế độ tải khác nhau: chế độ khởi động, làm việc định mức, chế độ quá tải... để chọn các linh kiện và Transistor công suất trong mạch cầu H như sau:

- + Tính dòng điện làm việc định mức của động cơ điện :

$$I = \frac{P}{U} = \frac{24}{12} = 2A.$$

- + Tính dòng điện khi khởi động động cơ một chiều, giả sử $I_{kđđc} = 3I_{đmđc} = 3 \cdot 2 = 6A$

- + Chọn chế độ động cơ có thể làm việc quá áp bằng $200\%U_{đm} = 2 \cdot 12 = 24V$.

- + Chọn chế độ quá dòng cho Transistor = (1,6 ÷ 2,5) lúc làm việc với tải nặng nhất là chế độ khởi động hoặc lúc quá tải = $1,6 \cdot 6 = 9,6A$.

Do đó ta chọn loại Transistor công suất chịu dòng trên 10A, điện áp làm việc max lớn hơn 24VDC. Tra bảng datasheet, chọn được loại Transistor TIP41 và TIP42 có thông số: $V_{CB0} = 40V$, $V_{CE0} = 40V$, $V_{EB0} = 5V$, $I_C = 10A$, $I_B = 2A$.

- **Từ mạch cầu H, ta kết nối tới vi điều khiển AT89C51 như sau:**

- + Chân P2.1 nối tới chân B của Q5.

- + Chân P2.0 nối tới chân B của Q6.

Khi đó tín hiệu điều khiển hoạt động của động cơ sẽ được cấp từ vi điều khiển AT89C51. Động cơ được đặt trong hộp kỹ thuật nối trực tiếp với hệ trục vít me, đai ốc và bánh răng nối với cánh cửa để truyền động cho cửa. Ở đây là biến chuyển động quay của động cơ thành chuyển động tịnh tiến để đóng mở cửa. Cửa được trượt trên 2 đường ray trên và dưới với bốn bánh xe (ổ vòng bi hoặc cũng có thể là các bạc đồng) gắn ở phía trên và phía dưới của cửa.

❖ **Nguyên lý điều khiển:**

Cấp nguồn điện 12VDC, 5 VDC như hình vẽ. Giả sử nếu bộ điều khiển xuất các tín hiệu điều khiển đặt vào Transistor Q5 và Q6 theo bảng chân lý thì sẽ điều khiển được trạng thái làm việc của động cơ: quay thuận, quay ngược, dừng. Điều khiển trạng thái làm việc của động cơ tuân theo bảng chân lý dưới đây:

Bảng 3.1. Bảng chân lý trạng thái hoạt động của động cơ DC.

Q5 (P2.1)	Q6 (P2.0)	Trạng thái làm việc của động cơ
0	0	Động cơ dừng
0	1	Động cơ quay thuận
1	0	Động cơ quay ngược
1	1	Động cơ dừng

Trong đó: + Trạng thái logic 1 = VCC, điều khiển Transistor dẫn.

+ Trạng thái logic 0 = 0 V, điều khiển Transistor khoá.

• **Điều khiển động cơ quay thuận:**

Giả sử nếu cấp tín hiệu điều khiển vào chân B của Transistor Q6, Q6 thông dẫn dòng, làm mất áp trên cực B của Q3 từ 12V xuống 0V nên Q3 thông, cực B của Q5 không được cấp tín hiệu điều khiển nên Q5 bị khoá, khi đó có áp đặt lên cực B của Transistor Q1, Q1 thông dẫn dòng chảy qua động cơ qua Q3 về đất 0V. Lúc đó động cơ sẽ quay theo chiều thuận.

• **Điều khiển động cơ quay ngược:**

Giả sử nếu cấp tín hiệu điều khiển mở Q5 và không cấp tín hiệu điều khiển mở Q6. Lúc đó Q5 thông dẫn dòng về nguồn 0V nên Q1 bị khoá và Q4 thông, Q6 khoá làm Q2 thông. Dòng điện từ nguồn chảy qua Q2 đến động cơ qua Q4 về nguồn 0V. Lúc đó động cơ sẽ quay ngược.

• Điều khiển động cơ dừng:

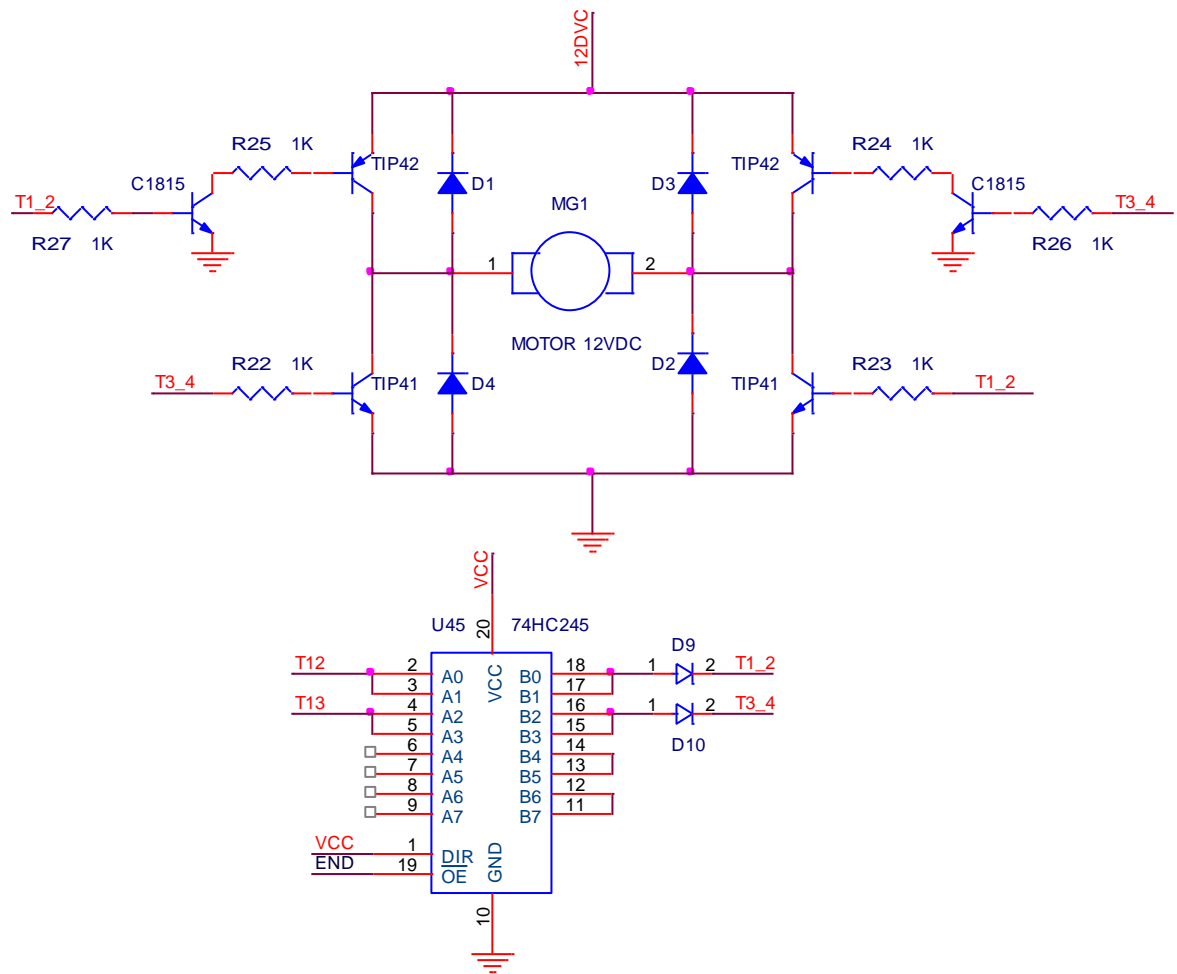
Muốn dừng động cơ thì ta cấp đồng thời tín hiệu điều khiển mở Q5 và Q6 hoặc không cấp tín hiệu điều khiển mở Q5 và Q6. Lúc đó tất cả các Transistor Q1 đến Q4 đều khoá dẫn đến không có dòng cấp cho động cơ. Động cơ sẽ không làm việc (không quay) .

Bốn Diốt 2N4007 làm nhiệm vụ ngăn dòng cảm ứng ngược tạo ra khi tắt cuộn dây động cơ (mất điện) gọi là Diốt hoàn năng lượng, khép kín năng lượng trong mạch khi động cơ một chiều dừng quay hoặc đang làm việc bị mất nguồn.

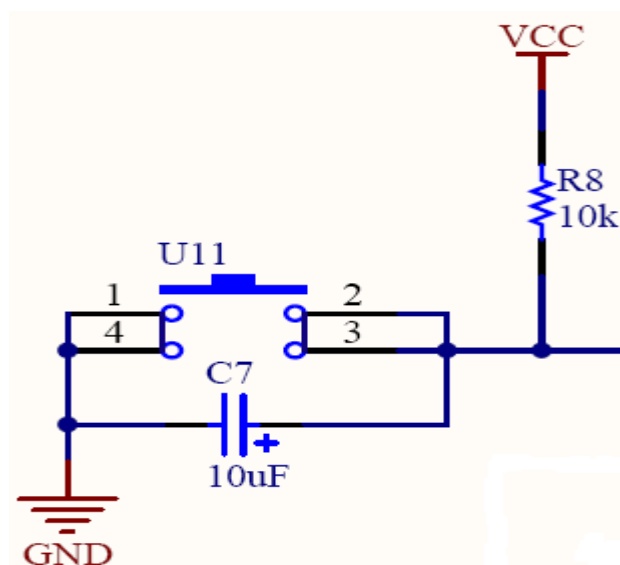
Các điện trở từ R_1 và R_2 lắp trong mạch dùng để hạn dòng đặt lên cực B của Transistor, bảo vệ Transistor. $R = 10K\Omega$ là điện trở treo dùng để kéo, tăng dòng điều khiển mở Transistor ở cổng Port2.

❖ Ưu điểm:

Điều khiển động cơ một chiều có đảo chiều dùng mạch cầu H có ưu điểm là mạch đóng mở không tiếp điểm bằng Transistor công suất nên có độ bền ổn định, độ tin cậy và tuổi thọ cao, dễ điều khiển trạng thái làm việc của động cơ. Tuy nhiên mạch cầu H chỉ dùng để điều khiển động cơ một chiều với tải tương đối nhỏ. Đối với các tải tương đối lớn và lớn thì không đáp ứng được nên không sử dụng. Khi đó ta phải dùng mạch cầu Tiristor.

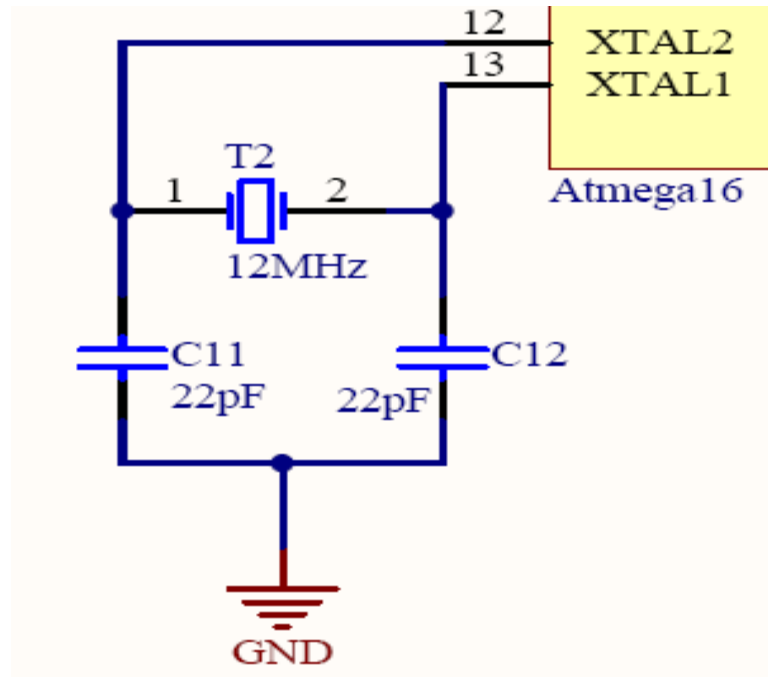


2.4. Khỏi Reset:



Khởi RESET có tác dụng đưa vi điều khiển về trạng thái ban đầu. Khi nút Reset được ấn điện áp +5V từ nguồn được nối vào chân Reset của vi điều khiển được chạy thẳng xuống đất lúc này điện áp tại chân vi điều khiển thay đổi đột ngột về 0, VĐK nhận biết được sự thay đổi này và khởi động lại trạng thái ban đầu cho hệ thống.

2.5. Khởi tạo xung dao động:



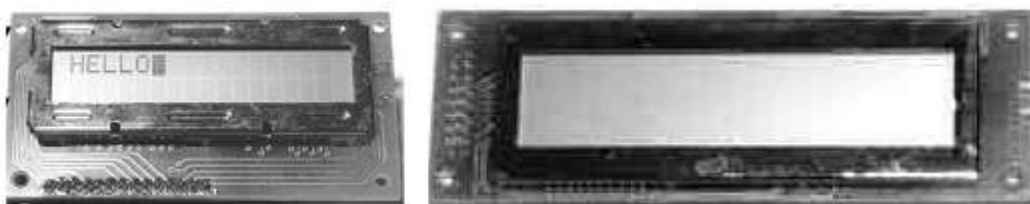
Đây là bộ dao động thạch anh có tác dụng tạo xung nhịp với tần số 12MHz cho VĐK hoạt động. Hai đầu này được nối vào 2chân XTAL1 và XTAL2 của VĐK.

2.6. THIẾT KẾ MODULE LCD.

2.6.1. Giới thiệu

Màn hình tinh thể lỏng LCD (Liquid Crystal Display) ngày càng được sử dụng rộng rãi và đang dần thay thế các đèn Led (7 đoạn và nhiều đoạn).

- Màn hình LCD có giá thành hạ.
- Khả năng hiển thị số, ký tự và đồ họa tốt hơn nhiều so với đèn Led.
- Công suất tiêu thụ nhỏ



2.6.2. Mô tả chân của LCD

1. V_{DD} , V_{SS} , và V_{EE} :

Là chân nguồn + 5VDC và chân nối đất 0V tương ứng, còn V_{EE} dùng để điều khiển độ tương phản của LCD.

Độ tương phản của LCD phụ thuộc vào nguồn cung cấp và các thông tin được hiển thị trên một hoặc hai dòng. Để điều chỉnh độ tương phản ta thay đổi giá trị điện áp (từ 0V đến V_{DD}) cung cấp cho chân có ký hiệu là V_{EE} . Để làm điều này ta có thể để chân V_{EE} nối với biến trở 10K Ω để điều chỉnh mức điện áp. Trong một số loại LCD, để tiện cho việc quan sát các thông tin hiển thị trên LCD được rõ ràng, ánh sáng môi trường bên ngoài không đủ độ chiếu sáng, người ta thường tích hợp trên modul LCD một đèn nền (Backlight) có màu xanh dương hoặc xanh lá cây. Trong quá trình sử dụng, để kéo dài tuổi thọ cho đèn nền của LCD ta mắc nối tiếp với đèn nền một điện trở để giới hạn dòng điện qua đèn (tương tự như giới hạn dòng qua các Led thông thường).

2. Chân chọn thanh ghi RS (Register Select):

Có hai thanh ghi rất quan trọng bên trong LCD. Chân RS được dùng để chọn các thanh ghi này. Nếu RS = 0 thì thanh ghi mã lệnh được chọn, cho phép người dùng gửi một lệnh chẳng hạn như lệnh xoá màn hình, đưa con trỏ về đầu dòng... Nếu RS = 1 thì thanh ghi dữ liệu được chọn cho phép người dùng gửi dữ liệu cần hiển thị lên LCD.

3. Chân đọc\ ghi R\ W (Read\ Write):

Chân đầu vào đọc\ ghi cho phép đọc\ ghi thông tin lên LCD. Khi R\W = 0 thì ghi thông tin lên LCD, khi R\W = 1 thì đọc.

Ở chế độ “ đọc”, nghĩa là MCU sẽ đọc thông tin từ LCD thông qua các chân DBx. Còn khi ở chế độ “ ghi”, nghĩa là MCU xuất thông tin điều khiển cho LCD thông qua các chân DBx.

4. Chân cho phép E (Enable):

Chân cho phép E được LCD sử dụng để chốt thông tin hiện đang có trên chân dữ liệu của nó. Khi dữ liệu được cấp đến chân dữ liệu thì một xung mức cao xuống thấp phải được áp đến chân E để LCD chốt dữ liệu trên chân dữ liệu. Xung này phải có độ rộng tối thiểu là 450ns.

5. Chân dữ liệu DBO- DB7:

Đây là 8 chân dữ liệu 8 bit, được dùng để gửi thông tin lên LCD hoặc đọc nội dung của các thanh ghi trong LCD.

Để hiển thị các chữ cái và các con số, mã ASCII của các chữ cái từ A- Z, a- z và các con số từ 0 – 9 được gửi đến các chân này khi bật RS = 1.

2.6.3. Tập lệnh của LCD.

❖ Gồm 4 nhóm lệnh sau:

+ Các lệnh về kiểu hiển thị. VD: Kiểu hiển thị (1 hàng / 2 hàng), chiều dài dữ liệu (8 bit / 4 bit) ...

+ Chỉ định địa chỉ RAM nội.

+ Nhóm lệnh truyền dữ liệu trong RAM nội.

+ Các lệnh còn lại.

Bảng dưới đây liệt kê mã lệnh của LCD.

Chân	Ký hiệu	I/O	Mô tả
1	VSS	-	Đất
2	VCC	-	Dương nguồn 5V
3	VEE	-	Nguồn điều khiển tương phản
4	RS		RS=0 chọn thanh ghi lệnh,RS=1 chọn thanh ghi dữ liệu
5	R/W		R/W=1 đọc dữ liệu,R/W=0 ghi dữ liệu
6	EN	I/O	Chân cho phép
7	DB0	I/O	Bus dữ liệu 8 bit
8	DB1	I/O	Bus dữ liệu 8 bit
9	DB2	I/O	Bus dữ liệu 8 bit
10	DB3	I/O	Bus dữ liệu 8 bit
11	DB4	I/O	Bus dữ liệu 8 bit
12	DB5	I/O	Bus dữ liệu 8 bit
13	DB6	I/O	Bus dữ liệu 8 bit
14	DB7	I/O	Bus dữ liệu 8 bit

Bảng 3.7 Mô tả chân của LCD.

❖ Khởi tạo LCD:

Là việc thiết lập các thông số làm việc ban đầu. Đối với LCD, khởi tạo giúp thiết lập các giao thức làm việc giữa LCD và vi điều khiển. Việc khởi tạo chỉ được thực hiện 1 lần duy nhất ở đầu chương trình điều khiển LCD và bao gồm các thiết lập sau:

+ **Display clear:** Xóa/ không xóa toàn bộ nội dung hiển thị trước đó.

+ **Function set:** Kiểu giao tiếp 8bit/ 4bit, số hàng hiển thị 1hàng/ 2hàng, kiểu kí tự 5x8/ 5x10.

+ **Display on/ off control:** Hiển thị/ tắt màn hình, hiển thị/ tắt con trỏ, nhấp nháy/ không nhấp nháy.

+ **Entry mode set:** Các thiết lập kiểu nhập kí tự như: Dịch/ không dịch, tự tăng/ giảm (Increment).

Màn hình LCD này dùng để giao diện giữa người dùng và bộ điều khiển cửa, để hiển thị các trạng thái làm việc của bộ điều khiển, các dòng thông báo tới người dùng và dòng hiển thị nhập mã Pin code vào từ bàn phím

Bảng 12.4: Danh sách liệt kê các lệnh và địa chỉ lệnh của LCD.

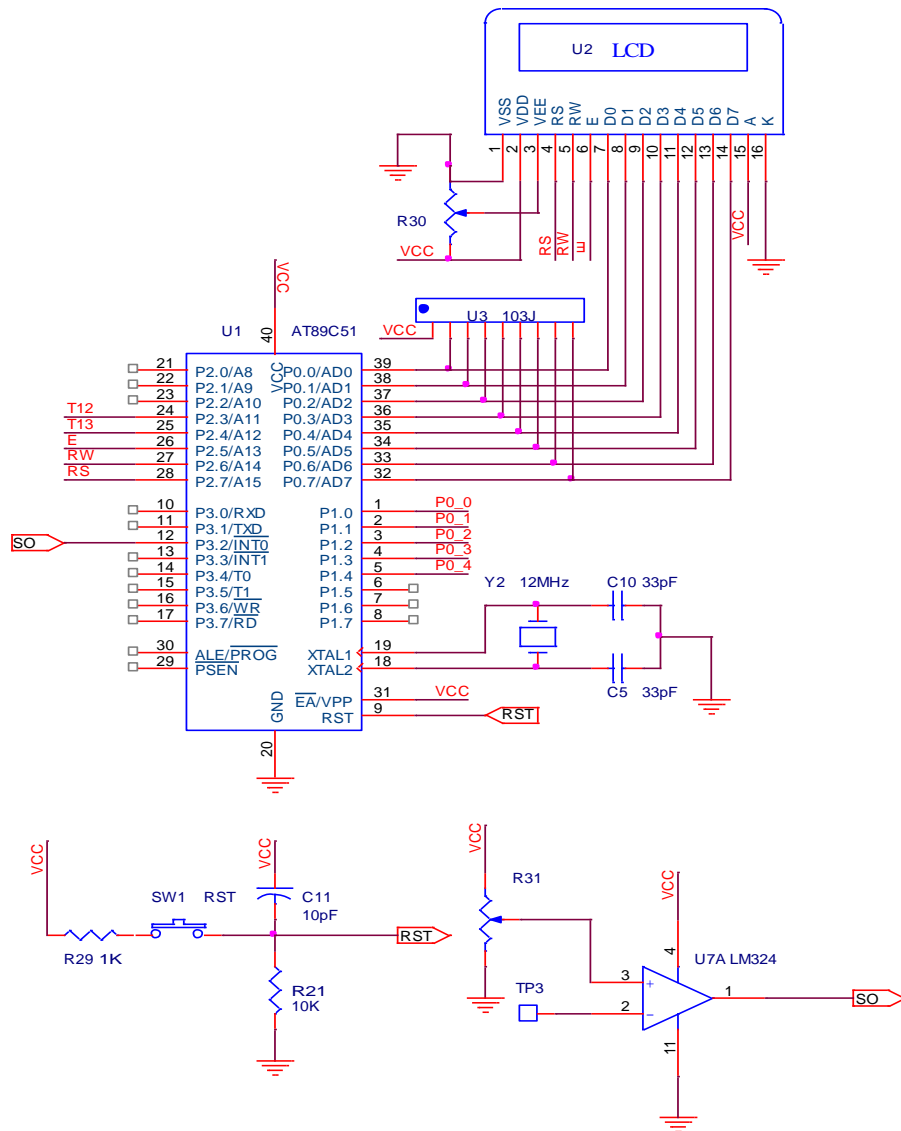
Lệnh	Mô tả											Thời gian thực hiện		
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0				
Xoá màn hình	0	0	0	0	0	0	0	0	0	1		Xoá toàn bộ màn hình và đặt địa chỉ 0 của DD RAM vào bộ đếm địa chỉ	1.64 μ s	
Trở về đầu dòng	0	0	0	0	0	0	0	0	0	1	-	Đặt địa chỉ 0 của DD RAM như bộ đếm địa chỉ. Trả hiển thị dịch về vị trí gốc DD RAM không thay đổi	1.64 μ s	
Đặt chế độ truy nhập	0	0	0	0	0	0	0	1		1/D	S	Đặt hướng chuyển dịch con trỏ và xác định dịch hiển thị các thao tác này được thực hiện khi đọc và ghi dữ liệu	40 μ s	
Điều khiển Bật/tắt hiển thị	0	0	0	0	0	0	1		D	C	B	Đặt Bật/ tắt màn hình (D) Bật/ tắt con trỏ (C) và nhấp nháy ký tự ở vị trí con trỏ (B)	40 μ s	
Dịch hiển thị và con trỏ	0	0	0	0	0	1		S / C	R / L	-	-	Dịch con trỏ và dịch hiển thị mà không thay đổi DD RAM	40 μ s	
Đặt chức năng	0	0	0	0	1		D / L	N	F	-	-	Thiết lập độ dài dữ liệu (DL) số dòng hiển thị (L) và phòng ký tự (F)	40 μ s	
Đặt địa chỉ CGRAM	0	0	0	1							AGC	Thiết lập địa chỉ C6 RAM dữ liệu CG RAM được gửi đi và nhận sau thiết lập này	40 μ s	
Thiết lập địa chỉ DD RAM	0	0	1								ADD	Thiết lập địa chỉ DD RAM dữ liệu DD RAM được gửi và nhận sau thiết lập này	40 μ s	
Cờ bận đọc và địa chỉ	0	1									ADD	Cờ bận đọc (BF) báo hoạt động bên trong đang được thực hiện và đọc nội dung bộ đếm địa chỉ	40 μ s	
Ghi dữ liệu CG hoặc DD RAM	1	0										Ghi dữ liệu	Ghi dữ liệu vào DD RAM hoặc CG RAM	40 μ s
Đọc dữ liệu CG hoặc DD RAM	1	1										Đọc dữ liệu	Đọc dữ liệu từ DD RAM hoặc CG RAM	40 μ s

Ghi chú:

1. Thời gian thực là thời gian cực đại khi tần số f_{CP} hoặc f_{osc} là 250KHz
2. Thời gian thực thay đổi khi tần số thay đổi. Khi tần số f_{EP} hay f_{osc} Là 270kHz thì thời gian thực hiện được tính $250/270 \times 40 = 35\mu s$ v.v...
3. Các ký hiệu viết tắt trong bảng là:
- 4.

DD RAM	RAM dữ liệu hiển thị (Display Data RAM)		
CG RAM	RAM máy phát ký tự (character Generator)		
ACC	Địa chỉ của RAM máy phát ký tự		
ADD	Địa chỉ của RAM dữ liệu hiển thị phù hợp với địa chỉ con trỏ.		
AC	Bộ đếm địa chỉ (Address Counter) được dùng cho các địa chỉ DD RAM và CG		
RAM.			
1/D = 1	Tăng	1/D = 0	Giảm
S = 1	Kèm dịch hiển thị		
S/C = 1	Dịch hiển thị	S/C = 0	Dịch con trỏ
R/L = 1	Dịch sang phải	R/L = 0	Dịch trái
DL = 1	8 bit	DL = 0	4 bit
N = 1	2 dòng	N = 1	1 dòng
F = 1	Ma trận điểm 5×10	F = 0	Ma trận điểm 5×7
BF = 1	Bật	BF = 0	Có thể nhận lệnh

2.6.4 SƠ ĐỒ GHEP NOI LCD



2.7. ĐO TỐC ĐỘ ĐỘNG CƠ MỘT CHIỀU

2.7.1 Giới thiệu.

Trên module điều khiển động cơ một chiều người ta thường gắn một thiết bị để đo tốc độ động cơ một chiều, từ đó để người điều khiển có thể đưa ra các thuật toán điều khiển động cơ theo ý muốn. Để đáp ứng nhu cầu thực tiễn đó trong đồ án này em có thiết kế lên module đo tốc độ động cơ một chiều sử dụng cảm biến từ.

2.7.2 Nguyên lý hoạt động.

Ta sử dụng module này để đo tốc độ của động cơ một chiều, ta sử dụng cảm biến từ để đo tốc độ của động cơ.

Nguyên lý hoạt động của cảm biến từ này như sau: Khi ta cho trục động cơ gắn với một cái đĩa, trên đĩa có gắn 1 mẫu giấy bạc. Cho đĩa này quay qua đầu phát của cảm biến từ, cảm biến từ sẽ phát hiện được thời điểm có mẫu giấy bạc chạy qua, mỗi lần phát hiện được thì cảm biến từ sẽ chuyển ra trạng thái ngõ ra của cảm biến một xung vuông. Nếu ta dùng bộ đếm xung vuông để đếm sự chuyển trạng thái ngõ ra của cảm biến từ trong một đơn vị thời gian thì ta có thể đo tính được tốc độ của động cơ.



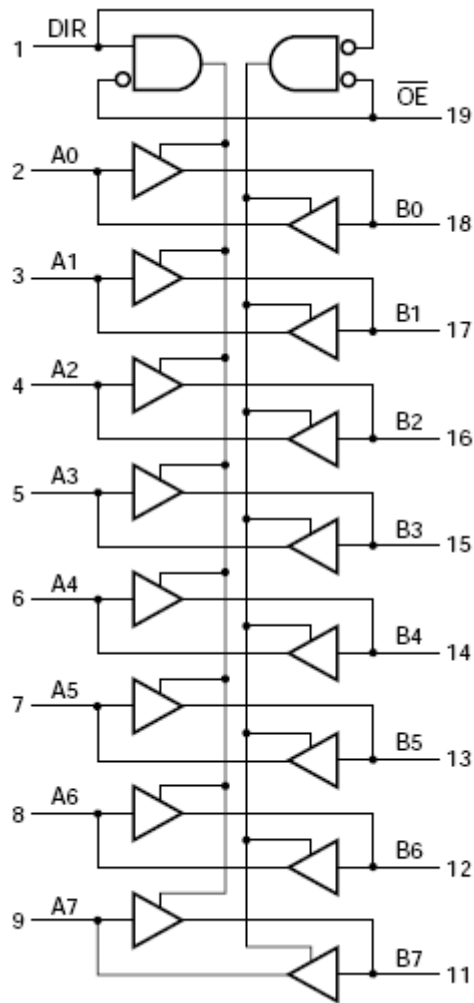
2.8. IC 74HC245

2.8.1. Mô tả chung:

IC 74HC245 là IC đệm có khả năng chốt đầu ra

- Thời gian chậm trễ trên đường truyền: 18 ns
- Phạm vi điện áp vận hành rộng: 2 -> 6 V
- Thích hợp với hệ thống Bus
- Đầu ra điều khiển: 15 LS-TTL tải

2.8.2. Sơ đồ chân:



- A0 ->A7: Dữ liệu đầu vào
- OE: Chân cho phép
- DIR: Quy định đầu ra
- B0 -> B7: 8 đầu ra
- GND: mass (0V)
- Vcc: Điện áp dương (5V)

2.8.3. Bảng hoạt động của IC 74HC245:

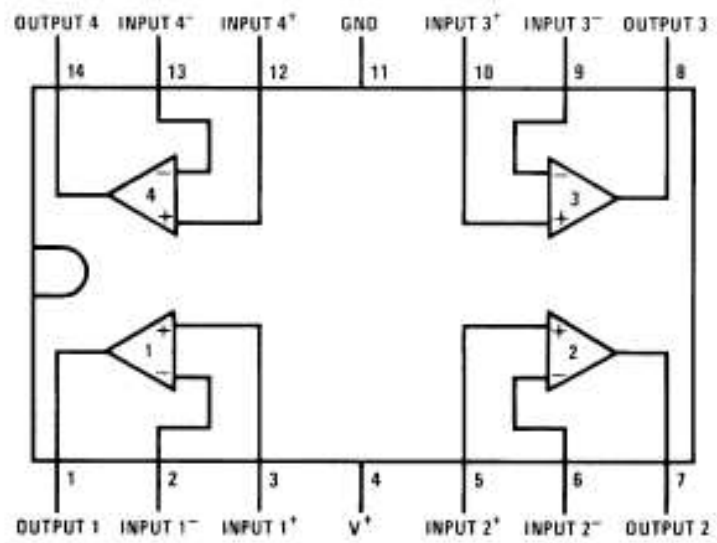
INPUTS		INPUTS/OUTPUTS	
\overline{OE}	DIR	A_N	B_N
L	L	A = B	\overline{NPUE}
L	H	\overline{NPUE}	B = A
H	X	Z	Z

- H: Mức cao
- L: Mức thấp

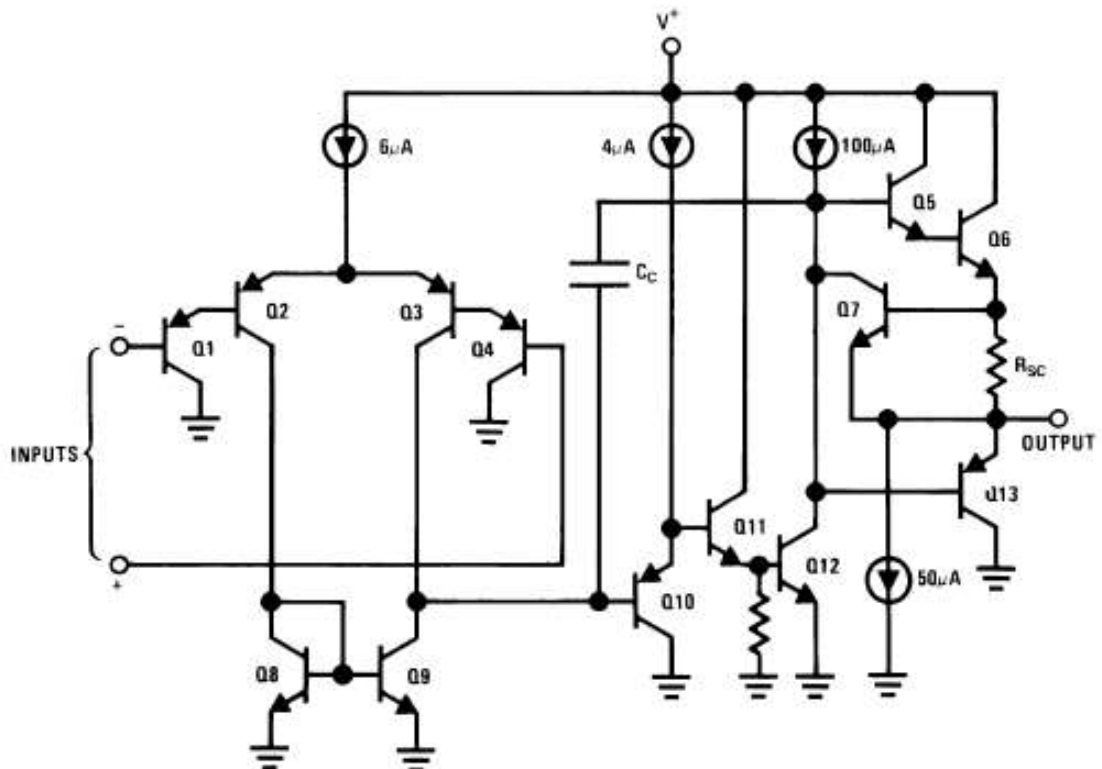
- Z: Trở kháng cao

2.1.4. IC LM 324:

Sơ đồ chân:



Sơ đồ nguyên lý IC LM324:



0022902

B. Phần cứng

2.9. Thiết kế mạch hoạt động cho AT89C51.

Để bộ điều khiển của tự động có thể hoạt động được thì phải thiết lập mạch hoạt động cho chip AT89C51 như sau:

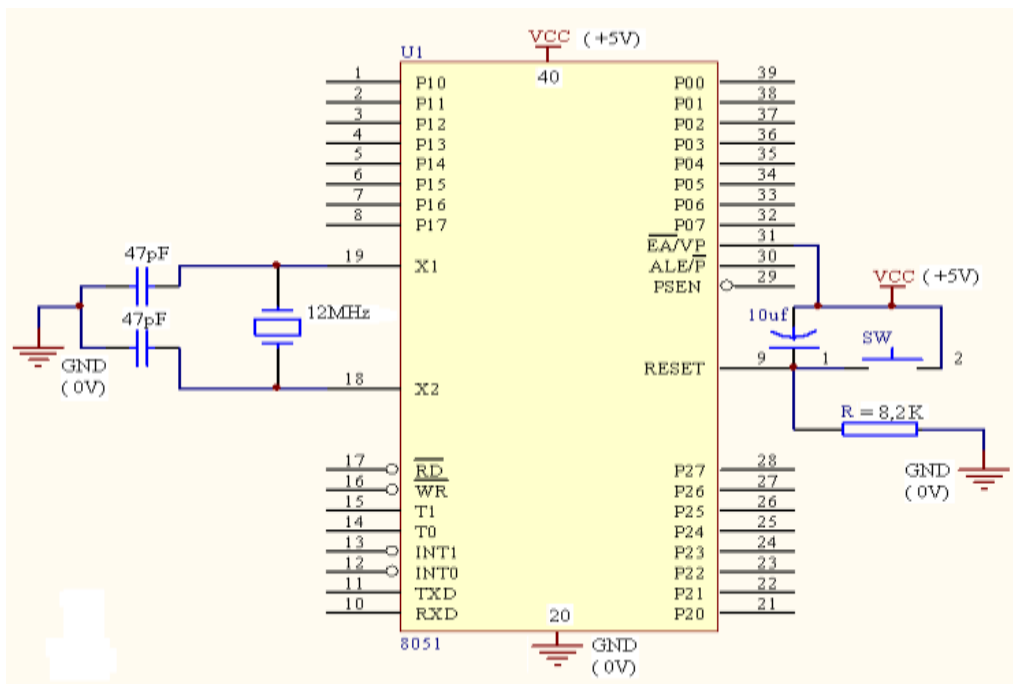
+ Cấp nguồn nuôi cho AT89C51: Chân 40 nối với nguồn +5VDC, chân 20 nối với mass GND (0V).

+ Cấp tín hiệu Reset cho AT89C51: bằng hai chế độ, chế độ tự động và chế độ bằng tay. Chế độ tự động dùng mạch tự dao động RC ($R = 8,2K\Omega$, $C = 10\mu F$) nối với nguồn +5VDC, khi cấp nguồn cho bộ điều khiển thì vi điều khiển sẽ tự động Reset trạng thái ban đầu. Mạch Reset bằng tay gồm nút bấm nối tiếp với điện trở 100Ω cấp vào chân số 9, khi muốn Reset lại bộ điều khiển chỉ cần nhấn nút này ở bất cứ thời điểm nào.

+ Cấp tín hiệu dao động cho bộ vi điều khiển từ bộ dao động thạch anh bên ngoài có tần số 12MHz, hai tụ $C = 47pF$ nối vào chân 18 (XTALL2), 19 (XTALL1) như hình vẽ bên dưới.

+ Ở đây do sử dụng bộ nhớ chương trình Rom bên trong AT89C51 nên ta nối chân EA (chân số 31) với nguồn +5VDC.

Mạch hoạt động cho AT89C51 như hình dưới đây:

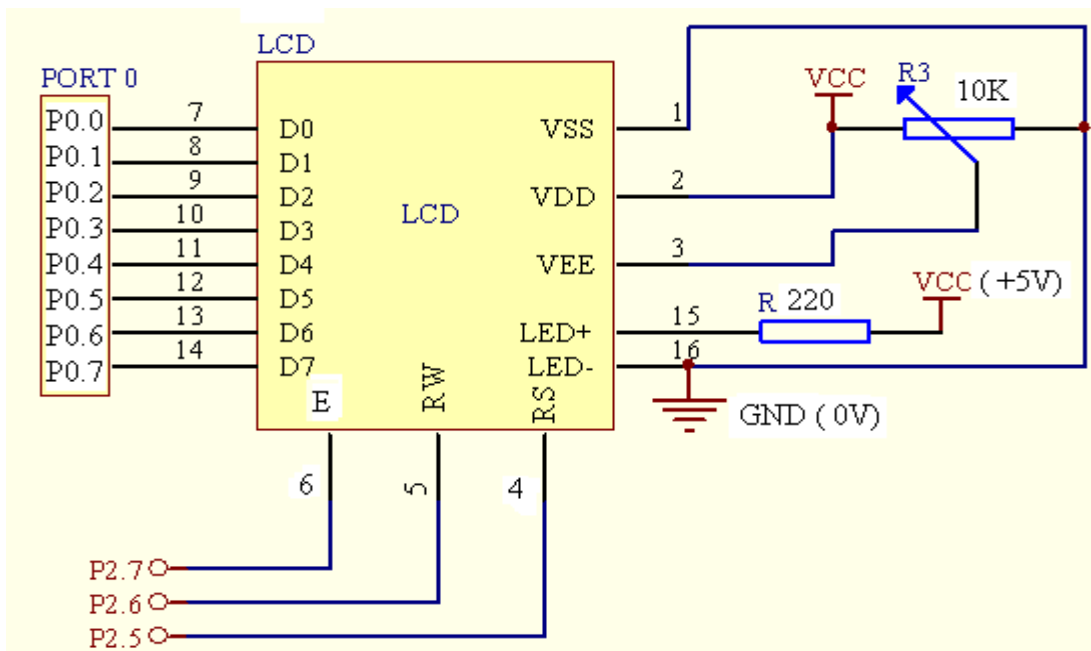


Hình 3.3. Thiết kế mạch hoạt động cho AT89C52.

Thiết kế mạch giao tiếp AT89C51 với màn hình LCD.

Sơ đồ kết nối chân như sau:

- + Chân số 1: V_{SS} của LCD nối với 0V (GND).
- + Chân số 2: V_{DD} của LCD nối với +5VDC.
- + Chân số 3: V_{BE} (V_0) của LCD nối với +5VDC qua 1 biến trở tinh chỉnh 10K Ω .
- + Chân số 4: RS của LCD nối với P2.5 của AT89C51.
- + Chân số 5: R\W của LCD nối với P2.6 của AT89C51.
- + Chân số 6: E của LCD nối với P2.7 của AT89C51.
- + Chân số 7 - 14: D0 – D7 của LCD nối với Port 0 (P0.0 – P0.7).
- + Chân số 15: A (Back light) của LCD nối với nguồn 5VDC qua một điện trở $R = 220 \Omega$ để hiện ánh sáng đèn nền Back light.
- + Chân số 16: K (Back light ground) của LCD nối với 0V (GND).



Hình 3.4. Sơ đồ mạch ghép nối màn hình LCD với AT89C51.

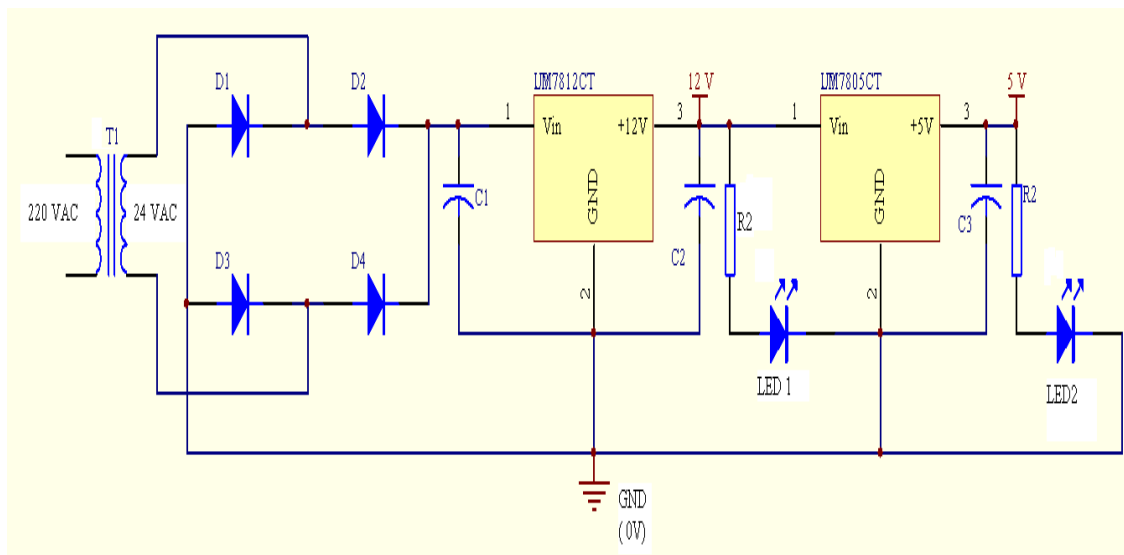
2.10. Thiết kế bộ nguồn.

Vi điều khiển AT89C51 và các thiết bị trên Mainboard dùng nguồn điện một chiều 5VDC chuẩn và mạch động cơ dùng nguồn điện 12VDC. Vì vậy em thiết kế bộ nguồn 5 VDC cung cấp cho bộ điều khiển cửa hoạt động. Nguồn 12VDC cung cấp cho tải động cơ.

Để có ổn áp ra tốt và ổn định, em chọn biến áp có $U_R = (1,2 \div 1,8) U_V$. Do đó em chọn biến áp nguồn là loại có đầu ra hạ áp 24VAC. Từ nguồn điện 220VAC, 50Hz, qua biến áp hạ áp xuống 24VAC, qua bộ chỉnh lưu cầu gồm bốn Điốt chỉnh lưu thành nguồn một chiều 24VDC qua bộ IC ổn áp 78LM12 ra thành nguồn một chiều 12VDC, qua bộ lọc gồm tụ điện $C1 = 1000\mu F$, $C2 = 1000\mu F$ có tác dụng san phẳng điện áp, và lọc những xung nhiễu từ nguồn vào. Nguồn 12VDC này được cung cấp cho mạch cầu H điều khiển động cơ một chiều điều khiển đóng mở cửa, cấp nguồn nuôi cho 2 bộ cảm biến an toàn và là nguồn đầu vào cho IC ổn áp 78LM05 và tụ $C3 = 1000\mu F$ để đưa ra điện áp đầu ra 5VDC cung cấp nguồn cho bộ điều khiển cửa tự động và các thiết bị khác như màn hình LCD, Loa...

Tác dụng của tụ lọc là đặc biệt quan trọng đối với nguồn điện áp +5VDC nó nhằm tránh các xung, gai nhiễu từ nguồn, động cơ hoặc tạo ra bởi các hiệu ứng cảm khi các linh kiện số chuyển trạng thái. Tránh làm treo vi điều khiển.

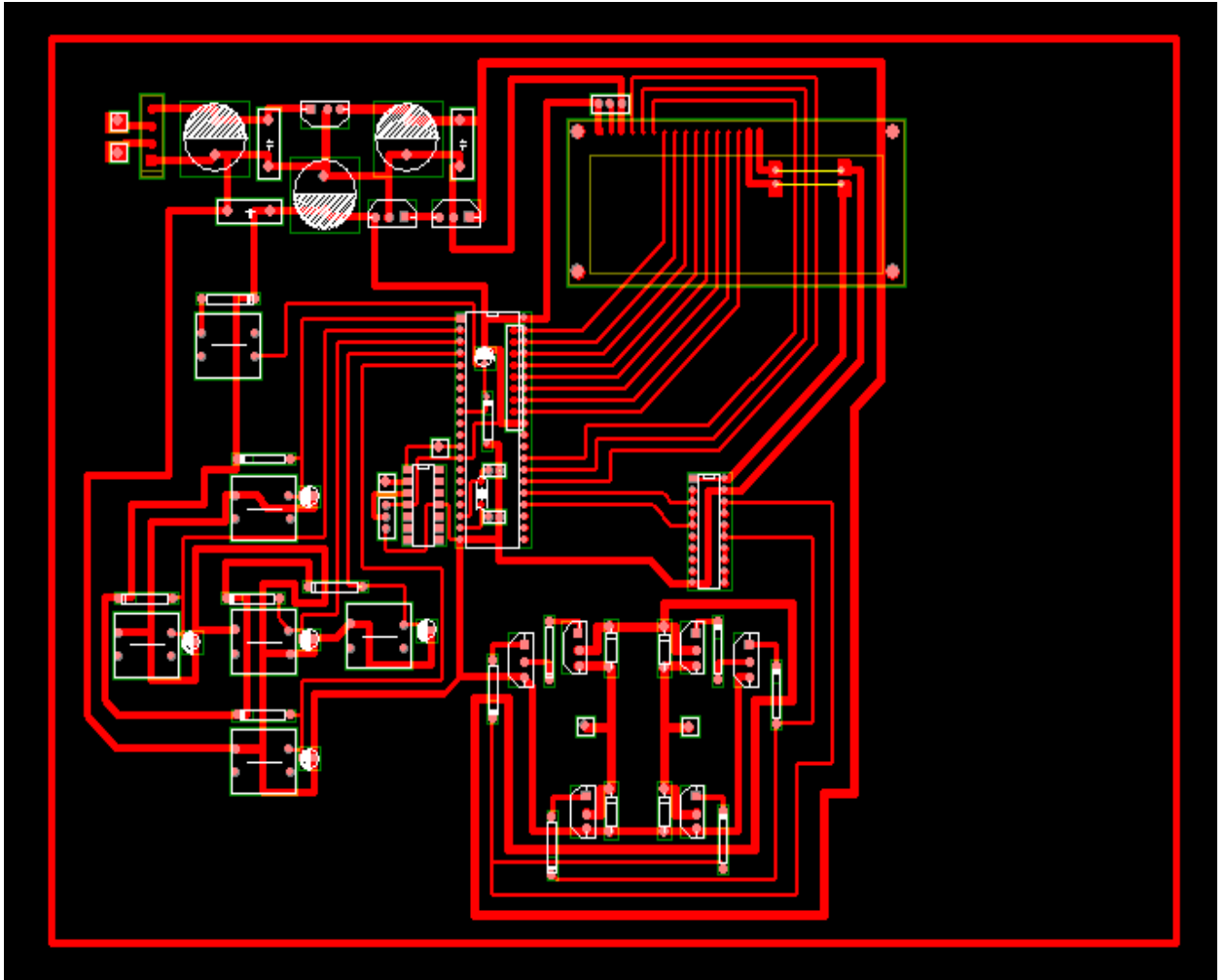
Hình vẽ dưới đây mô tả sơ đồ bộ nguồn cấp:



Hình 3. Sơ đồ mạch bộ nguồn.

Hoặc có thể dùng riêng biến áp 220VAC, 50Hz, đầu ra là 12VAC, 50Hz, và dòng đầu ra khoảng 10A qua bộ chỉnh lưu cầu 4 Điốt và các tụ lọc để cung cấp trực tiếp nguồn nuôi cho mạch cầu H điều khiển hoạt động của động cơ đảm bảo được dòng và áp cung cấp ổn định cho động cơ khoảng xấp xỉ 12VDC, 10A

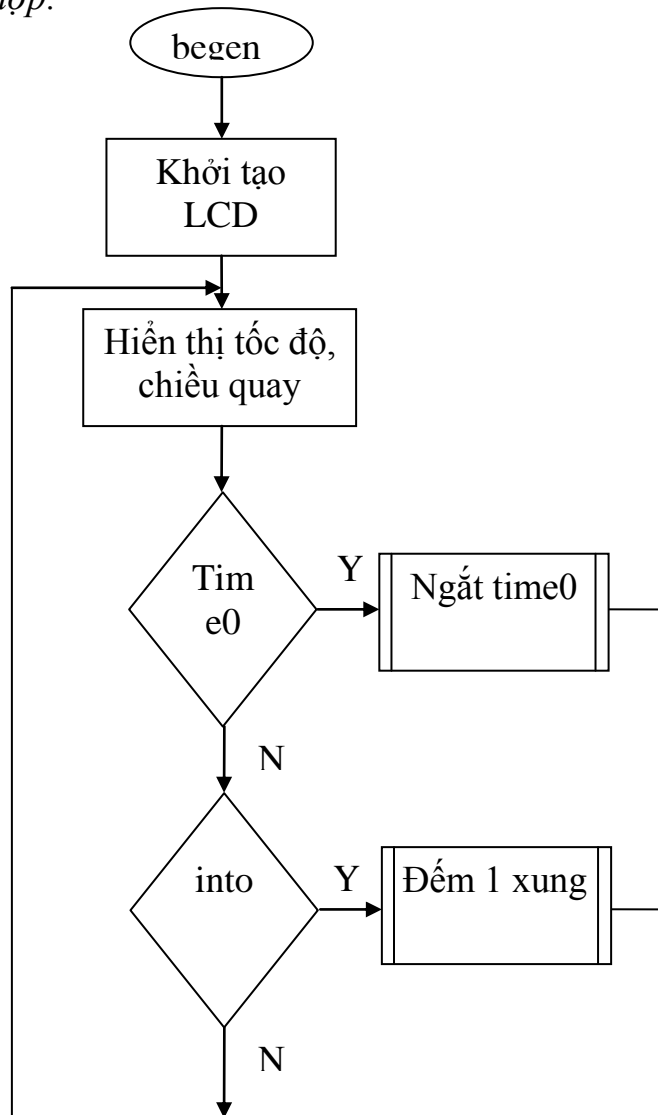
2.11. MẠCH IN



CHƯƠNG 3: THIẾT KẾ PHẦN MỀM

3.1 THUẬT TOÁN

Thuật toán tổng hợp:



3.2 PHẦN LẬP TRÌNH

3.2.1 ĐIỀU KHIỂN ĐỘNG CƠ 12VDC(MOTO_12VDC)

```
#include <AT89X52.H>
#include <delay.c>
#include <lcd.c>
/*=====Định nghĩa chân diều
khien=====*/
#define      PWM1          P2_4
#define      PWM2          P2_3
#define      start_stop    P1_2
#define      left          P1_1
#define      right         P1_3
#define      up            P1_0
#define      down          P1_4
/*=====Khai báo biến toàn cục=====*/
unsigned char
memorize_start_top=0,memorize_rotate_left=0,memorize_rotate_right=0;
unsigned char size=0,x,startup,add;
unsigned long count,SPEED;
unsigned int memorize;
/*=====Khai báo các hàm su
dung=====*/
void delay(int n);
void main(void);
void lead_key(void);
void init_motor(unsigned char p);
void scan_port(void);
/*===== ham con =====*/
void lead_key(void){
    x=P1;
    x=~x;
```

```

        if(x!=0){
            x=P1;
            x=~x;
            if(x!=0){
                x=P1;
                x=~x;
                if(x!=0){
                    x=P1;
                    x=~x;
                    startup=x;
                }
            }
            loop: x=P1;
                x=~x;
                while(x!=0){
                    goto loop;
                }
        }
    }

/*===== ham con =====*/
void init_motor(unsigned char u){
    switch(u){
        case 4 : memorize_start_top=~memorize_start_top;size=0;
                memorize_rotate_left=0;memorize_rotate_right=0; break;
        case 2 :memorize_rotate_left=1;memorize_rotate_right=0; break;
        case 8 :memorize_rotate_left=0;memorize_rotate_right=1; break;
        case 1 :size++;if(size>10)size=10; break;
        case 16 :size--;if(size>=10)size=0; break;
    }
    startup=0;
}

```

```

}
/*===== ham con =====*/
void scan_port(void){lead_key();init_motor(startup);}
/*===== Ham khoi tao timer =====*/

void init_interrupt(void)
{
TMOD=0x02; // Dat timer 0 che do 8 bit
TH0=0x9B; // Nap gia tri 155

EA=1; // Cho phep ngat toan cuc
ET0=1; // Cho phep ngat timer 0
EX0=1; // Cho phep ngat ngoai 0
IT0=1; // Cap phat ngat theo xuon
TR0=1; // Chay timer 0
}
/*===== phuc vu ngat time0 =====*/
void timer_0_overflow(void) interrupt TF0_VECTOR
{
TR0=0;TF0=0;
if(memorize_start_top!=0){add++;
if(add<=size){PWM1=memorize_rotate_left;
PWM2=memorize_rotate_right;}

else{PWM1=0;PWM2=0;}
if(add==10)add=0;
memorize++;
if(memorize==100){memorize=0;SPEED=count; count=0;}
}
else{PWM1=0;PWM2=0;}
scan_port();
TR0=1;

```

```

}
/*===== ngat ngoai 0 =====*/
void external_0_interrupt(void) interrupt 0 {count++;}
/*===== MAIN PROGRAM =====*/
void main (void)
{
init_lcd();
init_interrupt();
while(1){
    if(memorize_start_top!=0){
        print_str(0x80,"STAR  PWM= /10");
        print_val(0x8B,size,2);
        if(memorize_rotate_left==1) {print_str(0xC0,"LEFT SD:  v/ms"
);print_val(0xC9,SPEED,3);}
        if(memorize_rotate_right==1){print_str(0xC0,"RIGHT SD:  v/ms"
);print_val(0xC9,SPEED,3);}
    }

    else{write_cmd(0x01);print_str(0x00,"STOP" );}
    delay(1000);
}
}

```

3.2.2 TẠO THỜI GIAN TRỄ

```

void delay( unsigned int time )
{
    while(time -- )
    {
        TMOD = 0x10 ;
        TH1  = 0xFC ;//D8
        TL1  = 0x17 ;//F0
        TF1  = 0 ;
    }
}

```



```

    TR1 = 1 ;
    while(!TF1);
    TF1 = 0 ;
    TR1 = 0 ;
}

```

```

}

```

3.2.3 ĐIỀU KHIỂN LCD

```

    /*=====Đinh          nghĩa          hang          toan
cuc=====*/
    #define    return_home    0x02 // ve dau dong
    #define    funset        0x38 //matrix 5x7
    #define    entrymode    0x06 //dich con tro sang phai
    #define    offdsp        0x0C //tat hien thi nhap nhay con tro

    #define    cleardsp    0x01 //xoa man hinh
    /*=====Khai    bao    chan    dieu    khien    lcd
=====*/
    #define    LCD    P0
    #define    rs    P2_5
    #define    rw    P2_6
    #define    en    P2_7
    #define    write_cm {rs=0;rw=0;en=1;en=0;}
    #define    write_da {rs=1;rw=0;en=1;en=0;}
    /*=====Khai          bao          bien          toan
cuc=====*/
    char a;
    /*=====viet          cac
ham=====
    void busy_flag(void)
    {
    LCD=0xff;

```

```

rs=0;
rw=1;
do {
en=1;
delay(2);
en=0;
a=LCD;
a=a&0x80;
}
while(a==0x80);
}
/*=====ham ghi lenh len lcd
=====*/

void write_cmd(unsigned char lcd_cmd)
{
//busy_flag();
LCD=lcd_cmd;
write_cm;
delay(5);
}
/*=====ham ghi du lieu len lcd
=====*/

void write_data(unsigned char lcd_data)
{
//busy_flag();
LCD=lcd_data;
write_da;
delay(5);
}
/*=====Ham ghi lenh 1 chuo ky tu ra LCD
=====*/

```

```

void print_str(unsigned char place, char *input)
{
write_cmd(place|0x80);
delay(10);
while(*input!=0)
{
write_data(*input);
input++;

}
}
/*=====Ham   ghi   con   so   ra   LCD
=====*/

void print_val(unsigned char place,unsigned int val, unsigned int num)
{
int i;
unsigned char j=0;
unsigned char temp[10];
i=num;
write_cmd(place|0x80);
delay(10);
if(val==0)
{
if(num==1)
write_data('0');
else
{
for(j=0;j<(num-1);j++)
write_data(' ');
write_data('0');
}
}
}

```

```

else
{
while(i--)
    {
    if(val)
temp[i]=(val%10)|0x30;
    else
temp[i]=' ';
    val/=10;
    }
for(i=0;i<num;i++)
write_data(temp[i]);
}
}
/*=====
=====*/
/*-----*/
void init_lcd(void)

{
write_cmd(return_home);
write_cmd(funset);
write_cmd(entrymode);
write_cmd(offdsp);
write_cmd(cleardsp);
write_cmd(0x80);
}

```

Khoi tao cau hinh LCD

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau một thời gian tìm hiểu và tổng hợp tài liệu, tính toán một cách nghiêm túc, chúng em đã hoàn thành đề tài về “***Thiết kế xây dựng bộ đếm xung, ứng dụng đo tốc độ trong hệ thống truyền động điện***”. Đề tài được lựa chọn dựa trên tính thông dụng, hiệu quả và ứng dụng rộng rãi trong thực tế. Cách trình bày đơn giản ngắn gọn và dễ hiểu nhất.

Trong thời gian làm việc em đã học hỏi được nhiều kinh nghiệm quý báu như: hoàn thiện kỹ năng tìm kiếm và xử lý thông tin, tính toán, kỹ năng trình bày một bài đề tài, công dụng của bộ điều khiển động cơ một chiều theo phương pháp thay đổi độ rộng xung, các khâu trong mạch điều khiển độ rộng xung, có cơ hội được tiếp xúc và tìm hiểu nhiều hơn về các phần mềm mô phỏng như: AutoCAD 2007, Orcad10.5, Proteus... cũng như kỹ năng làm việc theo nhóm và độc lập hiệu quả.

TÀI LIỆU THAM KHẢO

1. Nguyễn Tăng Cường – Phan Quốc Thắng(2004)
Cấu trúc và lập trình họ vi điều khiển 8051, NXB Khoa học và kỹ thuật Hà Nội.
2. Ngô Diên Tập (2006) **Vi Điều Khiển Với Lập Trình C**
NXB Khoa học và kỹ thuật.
3. Tống Văn On (2006), **Thiết Kế Hệ Thống Với Họ 8051**
NXB Phương Đông
4. Tống Văn On – Hoàng Đức Hải (2008), **Họ Vi Điều Khiển 8051**
NXB Khoa học Xã Hội.