

## Lời nói đầu

Ngày nay việc ứng dụng vi điều khiển, vi xử lý đang ngày càng phát triển rộng rãi và thâm nhập ngày càng nhiều vào các lĩnh vực kỹ thuật và đời sống xã hội. Với xu hướng tất yếu này cùng với sự phát triển mạnh mẽ của công nghệ chế tạo, người ta đã tạo những vi điều khiển có cấu trúc mạnh hơn, đáp ứng thời gian thực tốt hơn, chuẩn hóa hơn so với các vi điều khiển 8 bit trước đây.

Với sự phát triển mạnh mẽ của khoa học, đặc biệt là ngành điện, điện tử, sự phát minh ra các linh kiện điện tử đã và đang ngày càng đáp ứng được yêu cầu của các hệ thống. Ưu điểm của việc sử dụng các linh kiện điện tử làm cho các hệ thống linh hoạt và đa dạng hơn, giá thành thấp hơn và độ chính xác cao hơn.

Sau thời gian học tập và tìm hiểu, chúng em đã được làm quen với môn học vi xử lý và đo lường hệ thống. Để áp dụng lý thuyết với thực tế của môn học này chúng em nhận bài tập lớn : " thiết kế mạch mô phỏng đo và hiển thị tốc độ động cơ ( có gắn Encoder 100 xung/vòng, khoảng đo [ 0-2500 vòng/phút]".

Tuy nhiên do kiến thức còn hạn chế, tài liệu tham khảo có giới hạn nên còn có những sai sót. Chúng em rất mong thầy, cô giáo thông cảm và giúp đỡ chúng em hoàn thiện bài tập lớn này.

Chúng em xin chân thành cảm ơn!

Mạch đo và hiển thị tốc độ động cơ ( có gắn Encoder 100 xung/vòng, khoảng đo [ 0 – 2500 vòng/phút] gồm:

## **Phần I : giới thiệu chung**

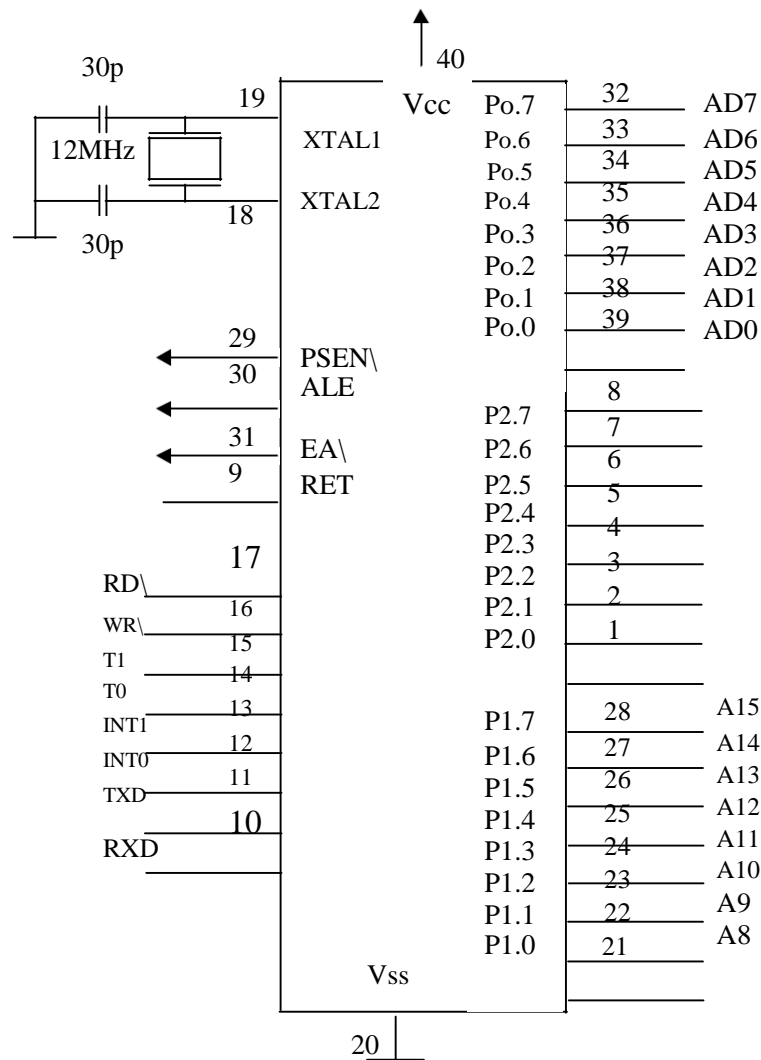
- 1.1. giới thiệu về vi điều khiển 8051
- 1.2. giới thiệu về encoder
- 1.3. giới thiệu chức năng của các linh kiện khác trong mạch.

## **Phần II : xây dựng phần cứng, lưu đồ thuật toán và chương trình điều khiển.**

- 2.1 Thiết kế phần cứng và sơ đồ mạch mô phỏng
- 2.2 lưu đồ thuật toán
- 2.3 chương trình điều khiển

## Phần I : giới thiệu chung

### 1.1 Giới thiệu về vi điều khiển AT89C51



Bộ vi điều khiển 8 bit AT89C51 hoạt động ở tần số 12 MHz, với bộ nhớ ROM 4kbyte, bộ nhớ RAM 128 byte cư trú bên trong và có thể mở rộng bộ nhớ ra ngoài. Ở bộ vi điều khiển này còn có 4 cổng 8 bit (P0,P1,P2,P3) vào/ra 2 chiều để giao tiếp với thiết bị ngoại vi. Ngoài ra, nó còn có:

2 bộ định thời 16 bit (Time 0 và Time 1)

Mạch giao tiếp nối tiếp

Bộ xử lý bit

Hệ thống điều khiển và xử lý ngắt

Các kênh điều khiển/ dữ liệu/ địa chỉ.

CPU

Các thanh ghi chức năng đặc biệt (SFR)

sơ đồ chân tín hiệu của AT89C51:

Chức năng của các chân tín hiệu như sau:

-P0.0 đến P0.7 là các chân của cổng 0 -

P1.0 đến P1.7 là các chân của cổng 1 -

P2.0 đến P2.7 là các chân của cổng 2 -

P3.0 đến P3.7 là các chân của cổng 3 -

RxD : nhận tín hiệu kiểu nối tiếp -TxD :

truyền tín hiệu kiểu nối tiếp -INT0: ngắt ngoài 0

-INT1: ngắt ngoài 1

-T0: chân vào 0 của bộ Timer/Counter 0

-T1: chân vào 1 của bộ Timer/Counter 1

-Wr: ghi dữ liệu vào bộ nhớ ngoài

-RST: chân vào Reset, tích cực ở mức logic cao trong khoảng 2 chu kỳ máy

-XTAL1: chân vào mạch khuếch đại dao động

-XTAL2: chân ra từ mạch khuếch đại dao động

-PSEN: chân cho phép đọc bộ nhớ chương trình ngoài

-ALE: chân tín hiệu cho phép chốt địa chỉ để truy cập bộ nhớ ngoài, khi On – chip xuất ra byte thấp của địa chỉ. Nó có thể được dùng cho các bộ Timer ngoài hoặc cho mục đích tạo xung Clock.

-EA/Vpp: cho phép On – chip truy cập bộ nhớ chương trình ngoài khi EA=0, nếu EA=1 thì On-chip sẽ làm việc với bộ nhớ chương trình nội trú.

-VCC : cung cấp nguồn cho On-chip

-GND: nối mát

+, Các thanh ghi chức năng đặc biệt khác:

Các thanh ghi chức năng đặc biệt là các thanh ghi đảm nhiệm các chức năng khác nhau trong chip. Chúng nằm ở RAM bên trong chip chiếm vùng không gian bộ nhớ

128bytes được định địa chỉ từ 80h đến Ffh.

Thanh ghi tích lũy (ACC):

đây là thanh ghi quan trọng trong chip, dùng để lưu trữ các toán hạng và kết quả của phép tính. Thanh ghi ACC dài 8 bit, có địa chỉ là E0h trong SFR.

Thanh ghi B:

thanh ghi thường sử dụng khi thực hiện các phép toán nhân, chia. Đối với các lệnh khác, thanh ghi B có thể xem như là thanh ghi đệm tạm thời. Trong SFR thanh ghi B dài 8 bits và có địa chỉ là F0h.

Con trỏ ngăn xếp:

thanh ghi này dài 8 bits, có địa chỉ trong SFR là 81h, giá trị của nó được tăng tự động trước khi thực hiện các lệnh CALL, PUSH. Ngăn xếp có thể đặt bất cứ nơi nào trong RAM của chip, nhưng sau khi khởi động lại ngăn xếp thì con trỏ ngăn xếp mặc định sẽ trỏ tới địa chỉ khởi đầu là 07h, vậy ngăn xếp sẽ được tạo ra bắt đầu từ 08h.

Con trỏ dữ liệu :

là thanh ghi dài 16 bits, gồm hai thanh dài 8 bits hợp lại là thanh ghi byte cao DPH và thanh ghi byte thấp DPL. Con trỏ dữ liệu có thể sử dụng như là thanh ghi 16 bits hoặc hai thanh ghi 8 bits độc lập. Trong SFR thanh ghi DPH có địa chỉ là 83h, còn thanh ghi DPL có địa chỉ là 82h.

Thanh ghi PSW:

là thanh ghi dài 8 bits, có địa chỉ trong SFR là D0h. Thanh ghi PSW dùng để chứa thông tin về trạng thái chương trình. Mỗi bit của PSW đảm nhiệm một chức năng cụ thể. Thanh ghi này được phép truy cập ở dạng mức bit.

Thanh ghi PCON : thanh ghi điều khiển nguồn.

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 xóa bit Enable của nó.

ET2 : bit cho phép hoặc không cho phép ngắt bộ Timer 2

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

EX1 : bit cho phép hoặc không cho phép ngắt ngoài 0

Thanh ghi IP : thanh ghi ưu tiên ngắt

Thanh ghi TCON: thanh ghi điều khiển bộ Timer/Counter

TF1: cờ tràn Timer1

TR1 : bit điều khiển bộ Timer 1 hoạt động

TF0 : cờ tràn Timer0

TR0 : bit điều khiển bộ Timer 0 hoạt động

IE1 : cờ ngắt ngoài 1

IE0 : cờ ngắt ngoài 0

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

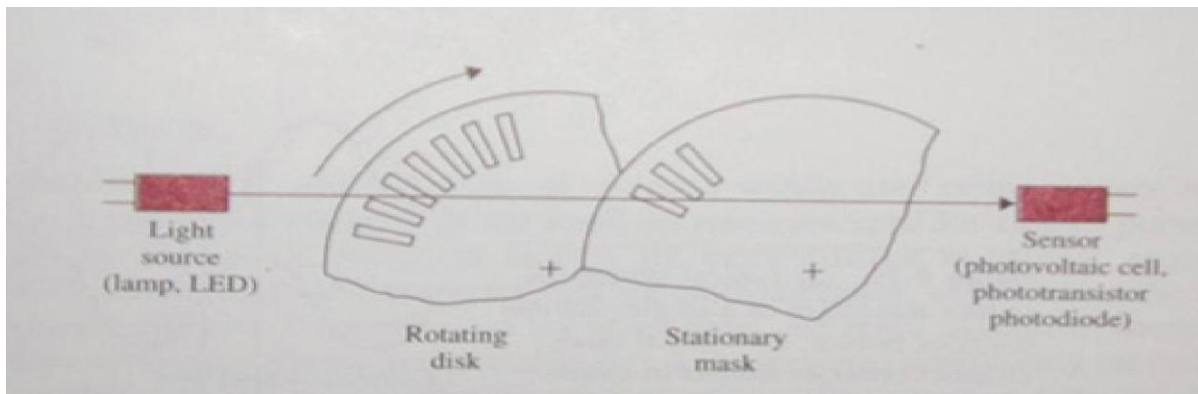
C/T : bit này cho phép chọn chức năng là Timer hay Counter.

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

Thanh ghi SCON : là thanh ghi trạng thái và điều khiển cổng nối tiếp.

## **1.2 Giới thiệu về Encoder :**

### Cấu tạo của encoder :



Nhìn trên hình ta thấy encoder gồm: 1 tấm tròn có khắc lỗ, 1 Hệ thống LED phát và thu.

### Cấu tạo chính của encoder :

Gồm 1 bộ phát ánh sáng ( led phát ), một bộ thu ánh sáng nhạy từ ánh sáng nhạy từ ánh sáng của bộ phát (bộ thu thường là photodiode hoặc phototransistor) 1 hay 2 đĩa quang gắn trên trục quay đặt giữa bộ phát và thu, thông thường trục quay này sẽ được gắn với trục quay của đối tượng cần đo tốc độ.

### Nguyên tắc hoạt động :

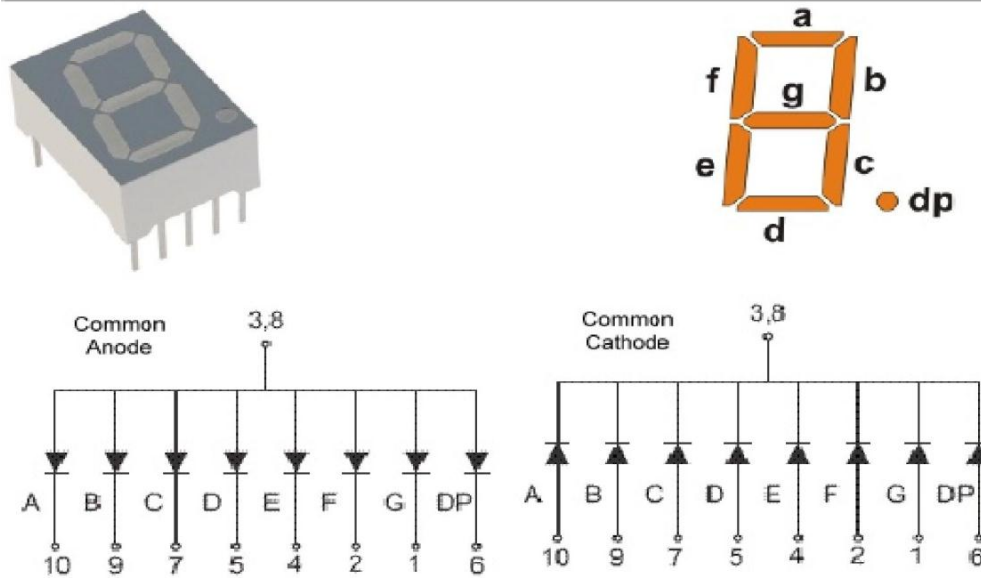
Nguyên lý cơ bản của encoder : đó là một đĩa tròn xoay, quay quanh trục. Trên đĩa có các lỗ (rãnh). Người ta dùng một đèn led để chiếu lên mặt đĩa. Khi đĩa quay, chỗ không có lỗ (rãnh), đèn led không chiếu xuyên qua được, chỗ có lỗ (rãnh), đèn led sẽ chiếu xuyên qua. Khi đó, phía mặt bên kia của đĩa, người ta đặt một con mắt thu. Với các tín hiệu có, hoặc không có ánh sáng chiếu qua, người ta ghi nhận được đèn led có chiếu qua lỗ hay không. Số xung đếm được và tăng lên nó tính bằng số lần ánh sáng bị cắt.


Như vậy là encoder sẽ tạo ra các tín hiệu xung vuông và các tín hiệu xung vuông này được cắt từ ánh sáng xuyên qua lỗ. Nên tần số của xung đầu ra sẽ phụ thuộc vào tốc độ quay của tấm tròn đó. Ứng dụng của encoder : trong các bài toán đo tốc độ động cơ, trong các máy CNC dùng để xác định khoảng dịch chuyển của 1 đối tượng thông qua đếm số vòng quay của trục.



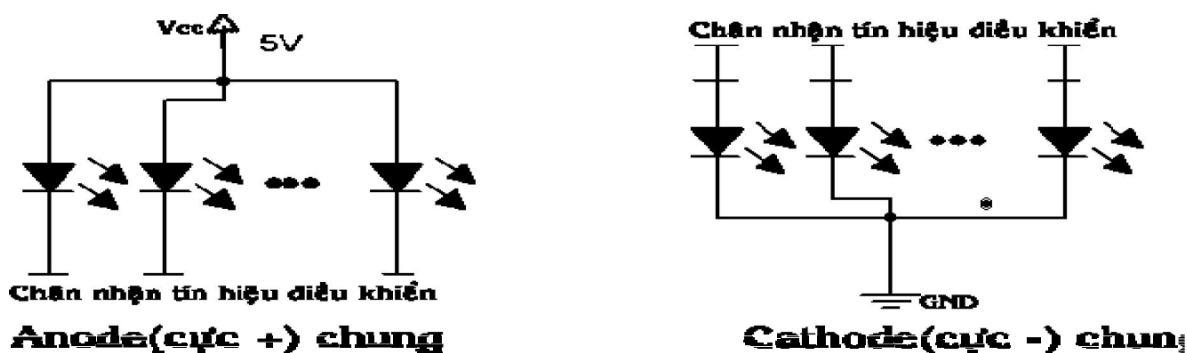
### ***1.3 Giới thiệu về các thiết bị khác trong mạch :***

**Giới thiệu về led 7 thanh :**



Led 7 đoạn có cấu tạo bao gồm 7 led đơn có dạng thanh xếp theo hình  và có thêm một led đơn hình tròn nhỏ thể hiện dấu chấm tròn ở góc dưới, bên phải của led 7 đoạn.

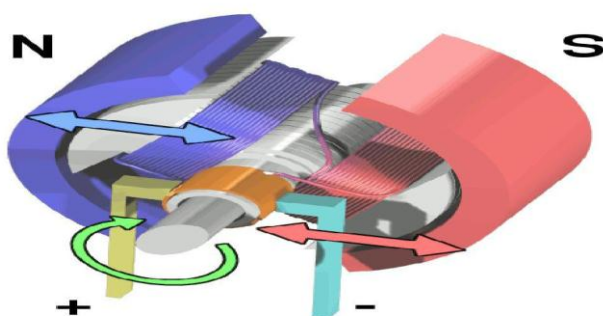
8 led đơn trên led 7 đoạn có Anode(cực +) hoặc Cathode(cực -) được nối chung với nhau vào một điểm, được đưa chân ra ngoài để kết nối với mạch điện. 8 cực còn lại trên mỗi led đơn được đưa thành 8 chân riêng, cũng được đưa ra ngoài để kết nối với mạch điện. Nếu led 7 đoạn có Anode(cực +) chung, đầu chung 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

**Giới thiệu động cơ điện 1 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 phần 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 phần 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).  
 Hình ảnh động cơ điện 1 chiều :





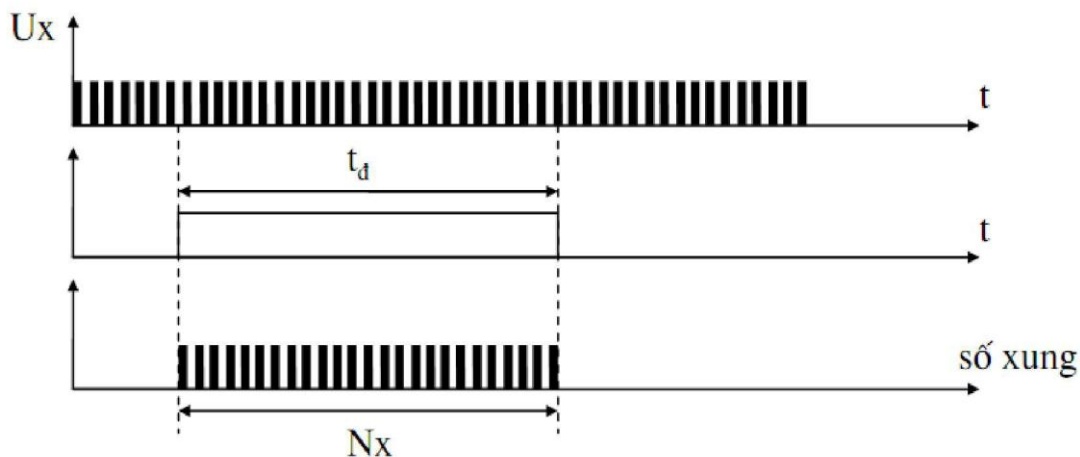
**Phần II : xây dựng phần cứng, lưu đồ thuật toán và chương trình điều khiển.**

**2.1 Thiết kế phần cứng và sơ đồ mạch mô phỏng :**



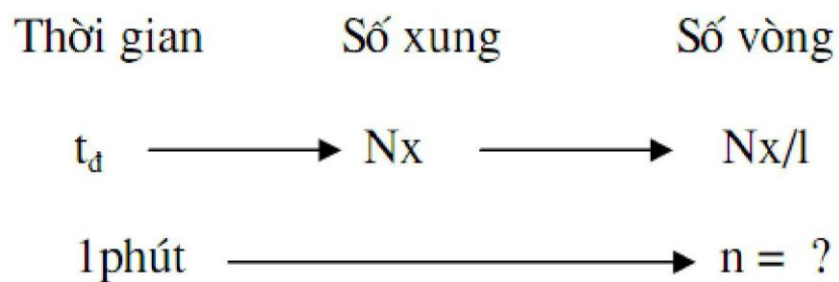
để đo tốc độ ta dùng phương pháp đếm số xung trong một khoảng thời gian đo ( $t_d$ ). Như vậy với phương pháp này thì ta lựa chọn encoder để biến tốc độ thành một dãy xung có tần số tỷ lệ với tốc độ quay của động cơ.

Tính toán kết quả đo :



Phương pháp đo là đếm số xung trong một khoảng thời gian đo ( $t_d$ ), số xung đếm được trong thời gian đo  $t_d$  là  $N_x$ . Ta đo tốc độ động cơ 1 chiều có gắn

encoder 100 xung/vòng, vậy ta chọn thời gian đo là  $t_d = 0,6s$  để đảm bảo thông tin cập nhật một cách tối ưu nhất.



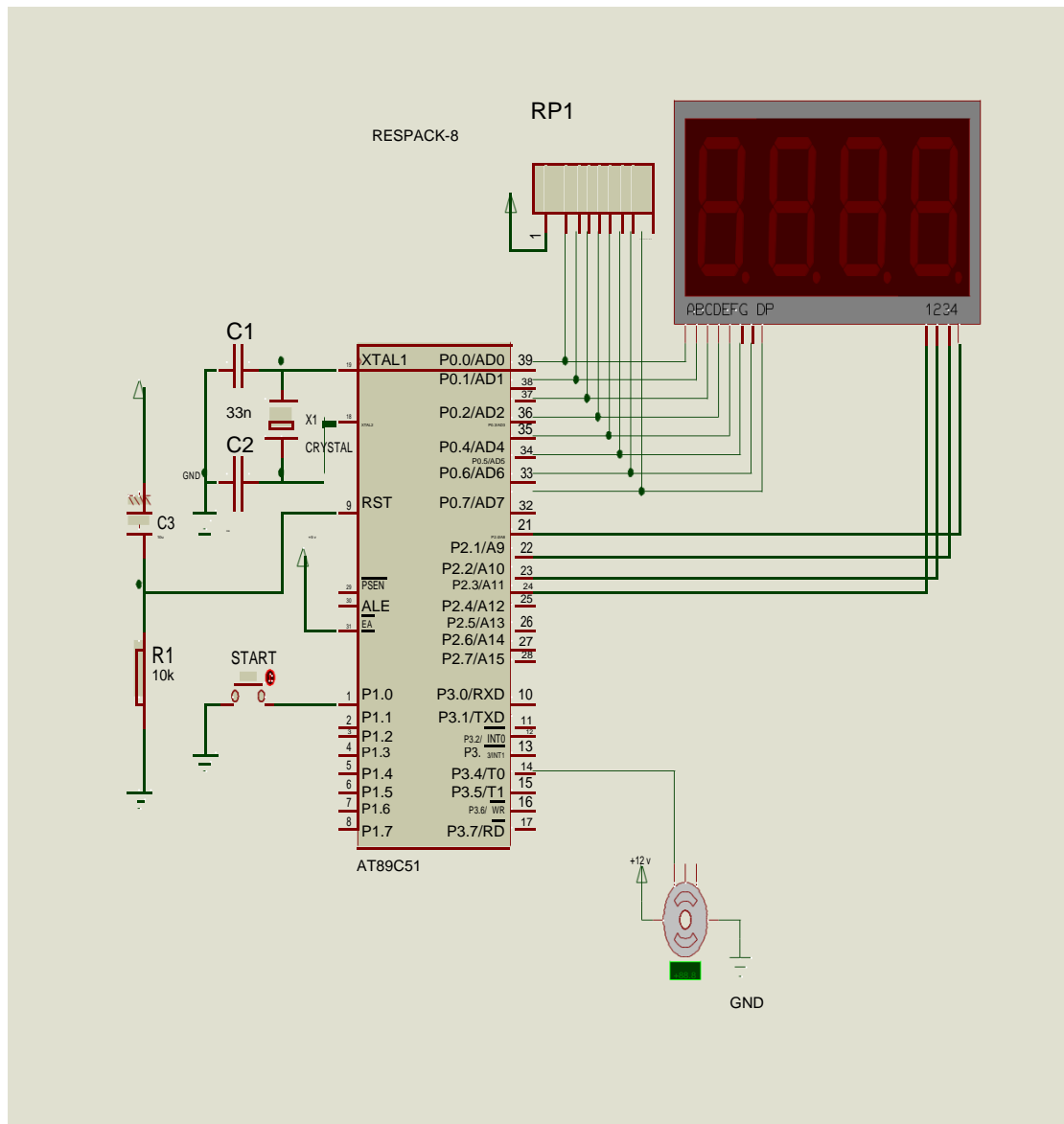
n : là tốc độ động cơ.

$$n = \frac{Nx \cdot 60}{t_d \cdot l}$$

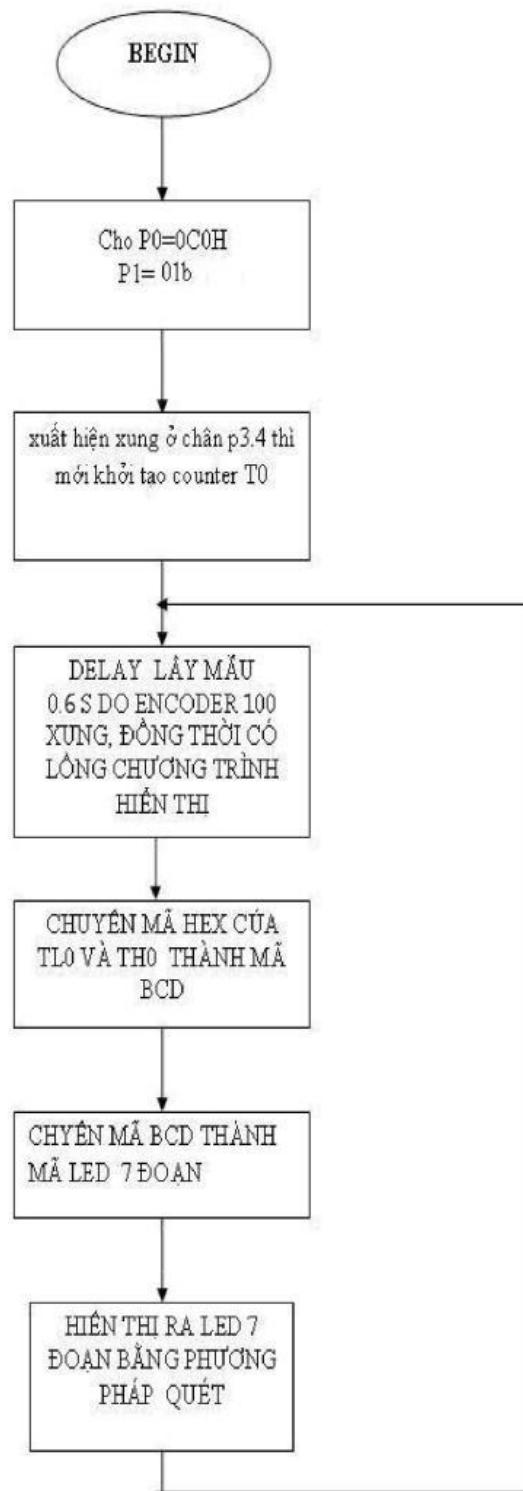
$$n = \frac{60 \cdot Nx}{t_d \cdot 100} = \frac{60 \cdot Nx}{0,6 \cdot 100} = Nx$$

Trong đó :  $t_d$  : thời gian lấy mẫu đo kết quả là 0,6s

Sơ đồ mạch mô phỏng :





**2.2 Lưu đồ thuật toán:**

### 2.3 Chương trình điều khiển :

; chương trình đo và hiển thị tốc độ động cơ

ORG 0000h

dem equ 55h

chuky equ 57h

bl equ 61h

bh equ 69h

layxung equ p3.4

hangchucdonvi equ 13h

hangngantram equ 14h

setb P1.0

doi\_an\_phim:

JB P1.0, doi\_an\_phim ; Thoat khi P3.4 = 1

mov p0,#0c0h

mov p2,#11111110b

quet equ p2

led7 equ p0

setb layxung

jb layxung,\$

jnb layxung,lucdau

xoaydata16:

clr c

mov a,bl

rlc a

mov bl,a

mov a,bh

rlc a

mov bh,a

mov a,hangchucdonvi

rlc a

mov hangchucdonvi,a

mov a,hangngantram

rlc a

mov hangngantram,a

ret

main:

clr tf0

clr tr0

call hienthiled

lucdau:

```
mov tmod,#00010101b
mov th0,#00h
mov tl0,#00h
setb tr0
setb p3.4
mov dptr,#ma7doan
lcall delay03s
clr tf1
clr tr1
clr tr0
clr tf0
call tuhextobcd
call giaiama7doan

sjmp main
ret
```

; hex 16 bit to

bcd tuhextobcd:

```
mov hangchucdonvi,#00h
mov hangngantram,#00h
mov bh,th0
mov bl,tl0
mov dem,#15
```

tuhextobcda:

```
call xoaydata16
mov a,hangchucdonvi
anl a,#0fh
cjne a,#4,$+3 ;nhay neu nho hon
jc tuhextobcdb
mov a,hangchucdonvi
add a,#3
mov hangchucdonvi,a
```

tuhextobcdb:

```
mov a,hangchucdonvi
anl a,#0f0h
cjne a,#50h,$+3
jc tuhextobcdc
mov a,hangchucdonvi
add a,#30h
mov hangchucdonvi,a
```

tuhextobcdc:

```
mov a,hangngantram
anl a,#0fh
cjne a,#5,$+3
jc tuhextobcdd
mov a,hangngantram
add a,#3
```

```
mov hangngantram,a
```

```
tuhextobcdd:
```

```
mov a,hangngantram
anl a,#0f0h
cjne a,#50h,$+3
jc tuhextobcde
mov a,hangngantram
add a,#30h
mov hangngantram,a
```

```
tuhextobcde:
```

```
djnz dem,tuhextobcda
call xoaydata16
ret
```

```
;;;BCD to 7seg
```

```
giaima7doan:
```

```
mov a,hangchucdonvi
anl a,#0fh
movc a,@a+dptr
mov 27h,a
mov a,hangchucdonvi
anl a,#0f0h
swap a
movc a,@a+dptr
mov 26h,a
```

```
mov a,hangngantram
anl a,#0fh
movc a,@a+dptr
mov 25h,a
```

```
mov a,hangngantram
anl a,#0f0h
swap a
movc a,@a+dptr
mov 24h,a
```

```
ret
```

```
hienthiled:
```

```
mov a,#01b
mov r0,#27h
```

```
ht1:
```

```
mov led7,@r0
mov quet,a
mov quet,#00h
dec r0
rl a
cjne r0,#22h,ht1
ret
```

delay03s: ; chương trình delay lấy mau (0.6s)

mov r6,#12

lai1:mov tmod,#00010101b

mov th1,#03ch

mov tl1,#0b0h

setb tr1

here: lcall hienthiled

jnb tf1,here

clr tf1

djnz r6,lai1

ret

ma7doan:

db 0c0h,0f9h,0a4h,0b0h,99h,92h,82h,0f8h,80h,90h

ret

**end**