

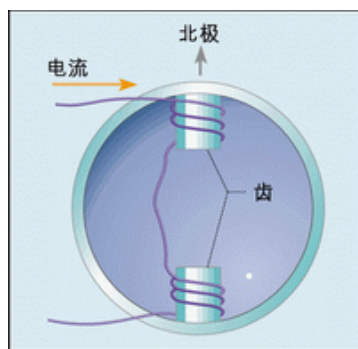
前言：电机控制属于比较繁杂的控制系统，鄙人想通过一系列的引用、事例，从原理、以及代码，转繁为简、深入浅出的，和大家一起进入电机控制领域。欢迎大家拍砖。

步进电机原理

作者：Dan Simon，电子与计算机工程系，克里夫兰州立大学

步进电机也叫步进器，它利用电磁学原理，将电能转换为机械能，人们早在20世纪20年代就开始使用这种电机。随着嵌入式系统(例如打印机、磁盘驱动器、玩具、雨刷、震动寻呼机、机械手臂和录像机等)的日益流行，步进电机的使用也开始暴增。不论在工业、军事、医疗、汽车还是娱乐业中，只要需要把某件物体从一个位置移动到另一个位置，步进电机就一定能派上用场。步进电机有许多种形状和尺寸，但不论形状和尺寸如何，它们都可以归为两类：可变磁阻步进电机和永磁步进电机。本文重点讨论更为简单也更常用的永磁步进电机。

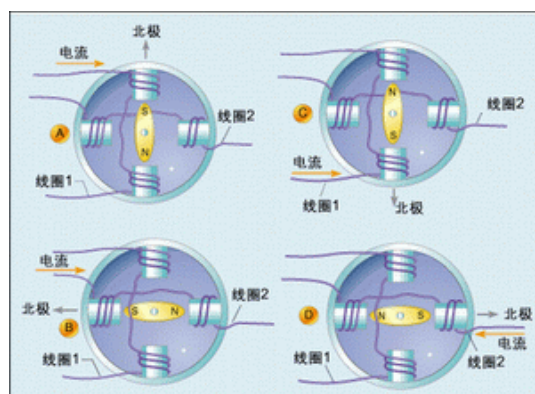
步进电机的构造



(图一，具有双齿槽和单绕组的定子)

如图 1 所示，步进电机是由一组缠绕在电机固定部件--定子齿槽上的线圈驱动的。通常情况下，一根绕成圈状的金属丝叫做螺线管，而在电机中，绕在齿上的金属丝则叫做绕组、线圈、或相。如果线圈中电流的流向如图 1 所示，并且我们从电机顶部向下看齿槽的顶部，那么电流在绕两个齿槽按逆时针流向流动。根据安培定律和右手准则，这样的电流会产生一个北极向上的磁场。

现在假设我们构造一个定子上缠绕有两个绕组的电机，内置一个能够绕中心任意转动的永久磁铁，这个可旋转部分叫做转子。图 2 给出了一种简单的电机，叫做双相双极电机，因为其定子上有两个绕组，而且其转子有两个磁极。如果我们按图 2a 所示方向给绕组 1 输送电流，而绕组 2 中没有电流流过，那么电机转子的南极就会自然地按图中所示，指向定子磁场的北极



(图 2：双相双极电机)

接着，我们再将绕组 2 的电流切断，按照图 2c 的方向给绕组 1 输送电流，注意：这时绕组 1 中的电流流向与图 2a

所示方向相反。于是定子的磁场北极就会指向下，从而导致转子旋转，其南极也指向下方。

然后我们又切断绕组 1 中的电流，按照图 2d 所示方向给绕组 2 输送电流，于是定子磁场又会指向右侧，从而使得转子旋转，其南极也指向右侧。

最后，我们再一次切断绕组 2 中的电流，并给绕组 1 输送如图 2a 所示的电流，这样，转子又会回到原来的位置。

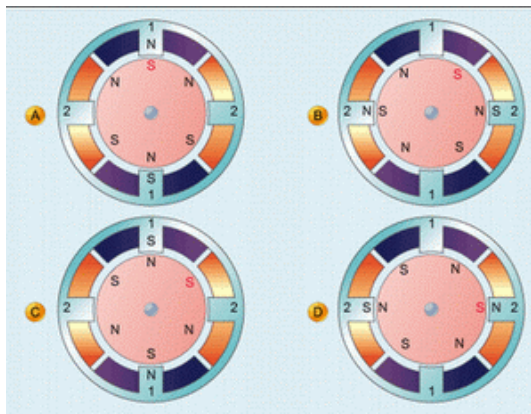
至此，我们对电机绕组完成了一个周期的电激励，电机转子旋转了一整圈。也就是说，电机的电频率等于它转动的机械频率。

如果我们用 1 秒钟顺序完成了图 2 所示的这 4 个步骤，那么电机的电频率就是 1Hz。其转子旋转了一周，因而其机械频率也是 1Hz。总之，一个双相步进电机的电频率和机械频率之间的关系可以用下式表示： $f_e = f_m \cdot P/2$ (1)

其中， f_e 代表电机的电频率， f_m 代表其机械频率，而P则代表电机转子的等距磁极数。

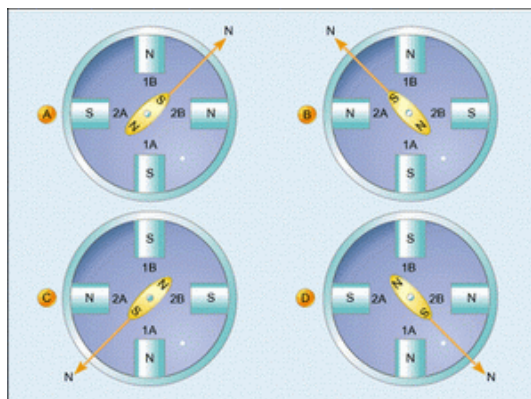
从图2中我们还可以看出，每一步操作都会使转子旋转90°；也就是说，一个双相步进电机每一步操作造成的旋转度数可由下式表示： $1 \text{ step} = 180^\circ / P$ (2)

由等式(2)可知，一个双极电机每动作一次可以旋转 $180^\circ / 2 = 90^\circ$ ；这与我们在图 2 中看到的情形正好相符。此外，该等式还表明，电机的磁极数越多，步进精度就越高。常见的是磁极数在 12 和 200 个之间的双相步进电机，这些电机的步进精度在 15° 和 0.9° 之间。



(图 3：双相六极电机)

图 3 给出的例子是一个双相、6 极步进电机，其中包含 3 个永久磁铁，因而有 6 个磁极。第一步，如图 3a 所示，我们给绕组 1 施加电压，在定子中产生一个北极指向其顶部的磁场，于是，转子的南极(图 3a 中红色的“S”一端)转向了该图的上方。接着，在图 3b 中，我们给绕组 2 施加电压，定子中产生一个北极指向其左侧的磁场。于是，转子的一个距离最近的南极转向了图的左方，即转子顺时针转动了 30°。第三步，在图 3c 中，我们又向绕组 1 施加一个电压，在定子中产生一个北极指向图下方的磁场，从而又使转子顺时针旋转 30°到达图 3c 所示的位置。而在图 3d 中，我们给绕组 2 施加电压，在定子中产生一个北极指向定子右侧的磁场，再一次使转子顺时针旋转 30°，到达图 3d 所示的位置。最后，我们再向绕组 1 施加电压，产生一个如图 3a 所示的北极指向定子上方的磁场，使得转子顺时针旋转 30°，结束一个电周期。如此可以看出，4 步电激励造成了 120°的机械旋转。也就是说，该电机的电频率是机械频率的 3 倍，这一结果符合等式 (1)。此外，我们从图 3 和等式(2)也能看出，该电机的转子每一步旋转 30°。



(图 4：同时激励电机的两个绕组)

如果同时向两个绕组输送电流，还能增大电机的扭矩，如图 4 所示。这时，电机定子的磁场是两个绕组各自产生的磁场的矢量和，虽然这一磁场每一次动作仍然只使电机旋转 90°，就象图 2 和图 3 中一样，但因为我们同时激励两个电机绕组，所以此时的磁场比单独激励一个绕组时更强。由于该磁场是两个垂直场的矢量和，因此它等于单独每个场的 2×1.414 倍，从而电机对其负载施加的扭矩也成正比增大。

电机的激励顺序

既然我们知道了一系列激励会使步进电机旋转，接下来就要设计硬件来实现所需的步进序列。一块能让电机动起来的硬件(或结合了硬件和软件的一套设备)就叫做电机驱动器。

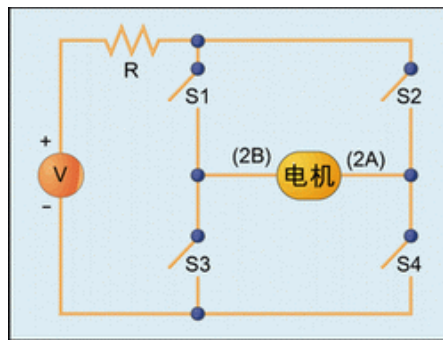
从图4中可以看出我们怎样激励双相电机的绕组才能使电机转子旋转，图中，电机内的绕组抽头分别被标为1A、1B、2A和2B。其中，1A和1B是绕组1的两个抽头，2A和2B则是绕组2的两个抽头。

首先，要给脚1B和2B施加一个正电压，并将1A和2A接地。然后，给脚1B和2A施加一个正电压，而将1A和2B接地，这一过程其实取决于导线绕齿槽缠绕的方向，假设导线缠绕的方向与上一节所述相符。依次进行下去，我们就得到了表1中总结的激励顺序，其中，“1”表示正电压，“0”表示接地。

Step #	Winding 1		Winding 2	
	Lead A	Lead B	Lead A	Lead B
1	0	1	0	1
2	0	1	1	0
3	1	0	1	0
4	1	0	0	1

(表1: 双相电机动作过程中的绕组抽头极性)

电流在电机绕组中有两种可能的流向，这样的电机就叫做双极电机和双极驱动序列。双极电机通常由一种叫做H桥的电路驱动,图5给出了连接H桥和步进电机两根抽头的电路。H桥通过一个电阻连接到一个电压固定的直流电源(其幅度可根据电机的要求选取),然后,该电路再经过4个开关(分别标为S1、S2、S3和S4)连接到绕组的两根抽头。这一电路的分布看起来有点象一个大写字母H,因此叫做H桥。



(图5: 可用于驱动电机每个绕组的H桥电路)

从表1中可以看出,要激励该电机,第一步应将抽头2A设为逻辑0,2B设为逻辑1,于是,我们可以闭合开关S1和S4,并断开开关S2和S3。接着,需要将抽头2A设为逻辑1,2B设为逻辑0,于是,我们可以闭合S2、S3,并断开S1和S4。与此类似,第三步我们可以闭合S2、S3并断开S1和S4,第四步则可以闭合S1、S4并断开S2、S3。

对绕组1的激励方法也不外乎如此,使用一对H桥就能产生需要的激励信号序列。表2所示就是激励过程中每一步开关所在的位置。

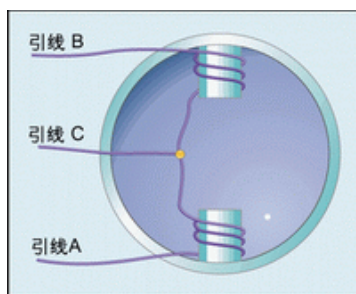
Step #	Winding 1				Winding 2			
	S1	S2	S3	S4	S1	S2	S3	S4
1	1	0	0	1	1	0	0	1
2	1	0	0	1	0	1	1	0
3	0	1	1	0	0	1	1	0
4	0	1	1	0	1	0	0	1

(表2: 双相电机动作过程中开关的位置)

注意,如果 $R=0$,而开关S1和S3又不小心同时闭合,那么流经开关的电流将达到无穷大。这时,不但开关会被烧坏,电源也可能损坏,因此电路中使用了一个非零阻值的电阻。尽管这个电阻会带来一定的功耗,也会降低电机驱动器的效率,但它可以提供短路保护。

单极电机及其驱动器

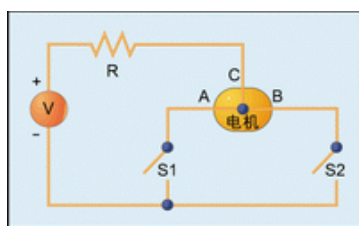
前面我们已经讨论了双极步进电机和驱动器。单极电机与双极电机类似,不同的是在单极电机中外部能够接触到的只有每个绕组的中心抽头,如图6所示。我们将从绕组顶部抽出的抽头标为抽头B,底部抽出的标为抽头A,中间的为抽头C。



(图 6)

有时我们会遇到一些抽头没有标注的电机，如果我们清楚步进电机的构造，就很容易通过测量抽头之间的阻值，识别出哪些抽头属于哪根绕组。不同绕组的抽头之间阻抗通常为无穷大。如果经测量，抽头 A 和 C 之间的阻抗为 100 欧姆，那么抽头 B 和 C 之间的阻抗也应是 100 欧姆，而 A 和 B 之间的阻抗为 200 欧姆。200 欧姆这一阻抗值就叫做绕组阻抗。

图 7 给出一个单极电机的单相驱动电路。从中可以看出，当 S1 闭合而 S2 断开时，电流将由右至左流经电机绕组；而当 S1 断开，S2 闭合时，电流流向变为由左至右。因此，我们仅用两个开关就能改变电流的流向(而在双极电机中需要 4 个开关才能做到)。表 3 所示为单极电机驱动电路中，每一步激励时开关所处的位置。



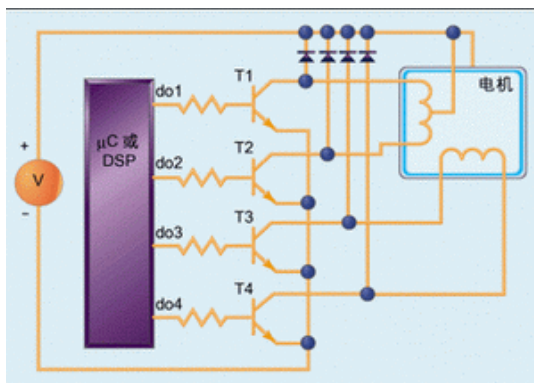
(图 7: 单极电机的控制电路)

虽然单极电机的驱动器控制起来相对简单，但由于在电机中使用了中心抽头，因此它比双极电机更复杂，而且其价格通常比双极电机贵。此外，由于电流只流经一半的电机绕组，所以单极电机只能产生一半的磁场。

在知道了单极电机和双极电机的构造原理之后，当我们遇到一个没有标示抽头也没有数据手册的电机时，我们就能自己推导出抽头和绕组的关系。带 4 个抽头的电机就是一个双相双极电机，我们可以通过测量导线之间的阻抗来分辨哪两个抽头属于同一个绕组。带 6 个抽头的电机可能是一个双相单极电机，也可能是一个三相双极电机，具体情况可以通过测量导线之间的阻抗来确定。

电机控制

本文前面讨论的电机控制理论可以采用全硬件方案实现，也可以用微控制器或 DSP 实现。图 8 说明了如何用晶体管作为开关来控制双相单极电机。每个晶体管的基极都要通过一个电阻连接到微控制器的一个数字输出上，阻值可以从 1 到 10M 欧姆，用于限制流入晶体管基极的电流。每个晶体管的发射极均接地，集电极连到电机绕组的 4 个抽头。电机的中心抽头均连接到电源电压的正端。



(图 8: 步进电机的控制电路图)

每个晶体管的集电极均通过一个二极管连接到电压源，以保护晶体管不被旋转时电机绕组上的感应电流烧坏。转子旋转时，电机绕组上会出现一个感应电压，如果晶体管集电极没有通过二极管连接到电压源，感应电压造成的电流就会涌入晶体管的集电极。

举个例子，假设数字输出 do1 为高而 do2 为低，于是 do1 会使晶体管 T1 导通，电流从 +V 流经中心抽头和 T1 的基极，然后由 T1 的发射极输出。但此时 do2 处于断开状态，因此电流无法流经 T2。这样推理下去，我们就能将表 3 改

为驱动电机所需的微控制器数字输出的改变顺序。

一旦清楚了驱动电机所需的硬件和数字输出的顺序，我们就可以对最顺手的微控制器或 DSP 编写软件，实现这些序列。

固件控制

我本人在一块 Microchip PIC16F877 上，利用 1N4003 二极管和 2SD1276A 达灵顿晶体管实现了以上谈到的电机控制器。PIC 的 PortA 第 0 位到第 3 位用来做数字输出。电机采用在 Jameco 购买的 5V 双相单极电机(Airpax [Thomson] 生产，型号为 M82101-P1)，并且用同一个 5V 电源为 PIC 和电机供电。但在真正应用时，为避免给微控制器的电源信号引入噪声，建议大家还是分别用不同的电源为电机和微控制器供电。

```
; Declare variables
cblock 0x20
    Count1      ; Delay counter
    Count2      ; Delay counter
endc

; Declare constants
WaitTime equ  d 50      ; Wait 50 msec between steps
Position1 equ  b 1010    ; digital output values for 1st step
Position2 equ  b 1001    ; digital output values for 2nd step
Position3 equ  b 0101    ; digital output values for 3rd step
Position4 equ  b 0110    ; digital output values for 4th step

;***** Program Start *****
;
org      0              ; Reset Vector
nop      ; nop for Microchip s In-Circuit Debugger (ICD)

;***** Initialization *****
banksel  TRISA
movlw   0xC0
movwf   TRISA          ; Set Port A bits 0-3 to be all outputs
movlw   b 00001110     ; Configure the Port A bits as digital
movwf   ADCON1
banksel  PORTA         ; Back to bank 0
clrf    Count1        ; Clear counter
clrf    Count2        ; Clear counter
movlw   Position1     ; Start the motor in Position 1
movwf   PORTA

;***** Motor drive *****
Drive
movlw   WaitTime      ; Initialize WaitTime to Count1
movwf   Count1
Loop
call    Timer         ; Wait 1 msec
decfsz  Count1, f     ; Decrement Count1. Count1 = 0 ?
goto    Loop          ; No - keep waiting
movf    PORTA, w      ; Read the present position
sublw   Position1
bnz     Drive1

movwf   PORTA
goto    Drive         ; Step again

Drive1
movf    PORTA, w      ; Read the present position
sublw   Position4
bnz     Drive2
movlw   Position1     ; Presently in Position4 - step to Position1
movwf   PORTA
goto    Drive         ; Step again

Drive2
movf    PORTA, w      ; Read the present position
sublw   Position3
bnz     Drive3
movlw   Position4     ; Presently in Position3 - step to Position4
movwf   PORTA
goto    Drive         ; Step again

Drive3
movf    PORTA, w      ; Read the present position
sublw   Position2
bnz     Drive4
movlw   Position3     ; Presently in Position2 - step to Position3
movwf   PORTA
goto    Drive         ; Step again

Drive4
movlw   Position1     ; Presently in unknown position - step to Position1
movwf   PORTA         ; Write PORTA
goto    Drive         ; Step again

;***** 1 msec Delay Subroutine *****
; This routine takes 1 msec to execute (at clock speed is 4 MHz)
Timer
movlw   d 200         ; Set loop count
movwf   Count2        ; Save loop count
Time1
nop
nop
decfsz  Count2, f     ; Count2 - 1 = 0 ?
goto    Time1        ; No - Continue
return   ; Yes - Return
```

列表 1 给出了控制程序的汇编源代码，该程序每 50 毫秒旋转电机一次。首先，程序会将微控制器的数字输出初始化为表 4 中第一步的值，然后每隔 50 毫秒(此时间常数由程序中的常量 waitTime 定义)按照正确的顺序循环输出数字信号。若需使电机反向旋转，只需按与表 4 所示相反的顺序输出数字信号即可。

本人所用的电机为 24 极电机，即每一步输出可以控制电机旋转 $180 \div 24 = 7.5^\circ$ 。电机每 50 毫秒旋转 7.5° ，也就是每 2.4 秒转一周。如果将常量 waitTime 减小一半，电机转速会加快一倍。但因为转子受惯性、摩擦力和其他机械限制，所以电机转速有一个上限，当定子磁场旋转过快时，转子的转速无法跟上，导致电机的旋转也无法跟上，开始跳动 (skipping)。如果这时再降低欧姆 aitTime，电机很可能干脆就停止旋转。

除了本文重点讨论的双相电机以外，步进电机还有其他类型，如三相步进电机或四相步进电机。另外还有一些双相步进电机，它们只有一个中心抽头，同时连接到两个绕组的中心点，这类步进电机外部有 5 个抽头引出。

同样，步进电机也不是电机家族中的唯一成员，最古老也最简单的电机是直流(DC)电机。早期的直流电机使用电刷，现在已经不再流行。如今常见的无刷直流电机，就是利用电子线路代替电刷进行换向的直流电机，这类电机中不存在电刷老化问题，因此其寿命比有刷直流电机长很多。

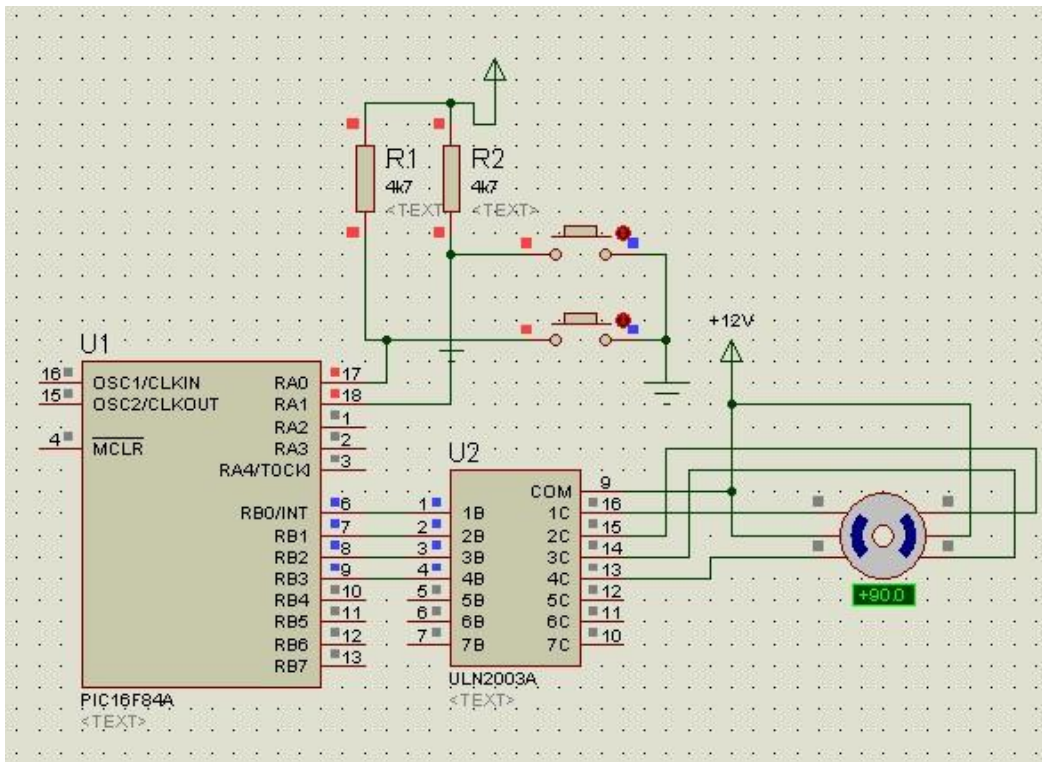
还有一种感应电机，其工作原理与步进电机或直流电机完全不同。直流电机采用的是直流电压源，而感应电机则采用交流(AC)电压源，并且步进电机和直流电机中转子与定子磁场的旋转是同步的，而感应电机中转子的转速滞后于定子磁场的转速。

本文小结

本文对步进电机进行了概括性的介绍，更多的细节等待着您的发现。但只要您理解了本文介绍的电机工作原理，那么您就已经完全可以开始设计、维护和调试步进电机的驱动软、硬件了。

////////////////////////////////////

另附件用 c 写的个简单步进控制:



```

LIST    p=16F84 ; PIC16F844 is the target processor

#include "P16F84.INC" ; Include header file

CBLOCK 0x10 ; Temporary storage

    pos
    dc1
    dc2
ENDC

ORG    0
    
```

```

entrypoint    goto start

                ORG 4

intvector     goto intvector

start         clrw                ; Zero.

                movwf PORTB        ; Ensure PORTB is zero before we enable it.

                bsf  STATUS,RP0    ; Select Bank 1
                movlw 0xF0         ; Set port B bits 0-3 as outputs
                movwf TRISB        ; Set TRISB register
                bcf  STATUS,RP0    ; Select Bank 0

                movlw 3            ; b"00000011" Initialize the motor position
                movwf pos
                movwf PORTB
                call delay
                clrf PORTB        ; Motor drive off

;Main loop
loop          btfss PORTA,0        ; Test clockwise button
                call stepcw
                btfss PORTA,1      ; Test anti-clockwise button
                call stepccw
                goto loop

;Rotate one step clockwise
stepcw        bcf  STATUS,C        ; Clear the carry flag
                btfsc pos,3        ; Set carry if this bit set
                bsf  STATUS,C
                rlf  pos,W          ; Pick up and rotate the motor's current position
                andlw 0x0F         ; Mask to lower nibble
                movwf pos
                movwf PORTB        ; Drive the outputs
                call delay         ; Wait
                clrf PORTB        ; Clear the output
                return

;Rotate one step counter clockwise
stepccw       bcf  STATUS,C        ; Clear the carry flag
                btfsc pos,0
                bsf  pos,4
                rrf  pos,W          ; Pick up and rotate the motor's current position
                andlw 0x0F         ; Mask to lower nibble
                movwf pos
                movwf PORTB        ; Drive the outputs
                call delay         ; Wait
                clrf PORTB        ; Clear the output
                return

```

; This routine implements the delay between steps,

; and thus controls the motor speed.

```

delay      movlw 18      ; Outer loop iteration count
           movwf dc1
dl1        clrf  dc2      ; Initialize inner loop
dl2        nop
           nop
           decfsz dc2,F
           goto  dl2
           decfsz dc1,F
           goto  dl1
           return
END

```

对照的c程序:

```

#include <pic.h>

void delay(unsigned int data);

void stepcw();//顺时针
void stepccw();//逆时针

unsigned char pos;

void main()
{
    PORTB=0;
    TRISB=0xF0;
    pos=0x03;
    PORTB=0x03;
    delay(3000);
PORTB=0;
    while(1)
    {
        if(RA0==0)
        {
            stepcw();
        }
        if(RA1==0)
        {
            stepccw();
        }
    }
}

```

函数: void delay(int data)

描述: data=1000;延迟40ms (4MHz)

*****/

```

void delay(unsigned int data)
{
    unsigned int d=data;
    while(--d){;}
}

```



```

}
void stepcw()
{
    if(pos>8)
    {
        pos=(pos<<1)|0x01;
        pos=pos&0x0f;
        PORTB=pos;
        delay(3000);
        PORTB=0;

    }
    Else
    {
        pos=pos<<1;
        pos=pos&0x0f;
        PORTB=pos;
        delay(3000);
        PORTB=0;
    }
}
void stepccw()
{
    if((pos&0x01)==0)
    {
        pos=(pos>>1)&0x0f;
        PORTB=pos;
        delay(3000);
        PORTB=0;
    }
    else if((pos&0x10)==0)
    {
        pos=pos|0x10;
        pos=(pos>>1)&0x0f;
        PORTB=pos;
        delay(3000);
        PORTB=0;
    }
    else{
        pos=(pos>>1)&0x0f;
        PORTB=pos;
        delay(3000);
        PORTB=0;
    }
}

```

(From:<http://www.goldenchip.com.cn/gdbbs/dispbbs.asp?ID=200742816282518422&prefile=list.asp&boardid=7&sortfield=lastposttime&sortorder=desc&timelimit=&operator=&page=2>)