

基于 PIC16 的无传感器无刷直流电机控制实现

作者: Joseph Julicher
Dieter Peter
Microchip Technology Inc.

简介

无刷直流电机 (BLDC) 引起众多关注。与有刷 DC 电机相比较, BLDC 电机具有以下诸多优点:

- 省却了机械换向器可实现更高速运行
- 电刷性能限制了 DC 电机的瞬态响应能力
- 直流电机电刷上的压降增加了电机损耗
- 由于电枢电抗电压导致的电刷缺陷限制了铁心长度, 从而降低了速度响应且增加了特定转矩下的电机惯量
- BLDC 电机中的发热源位于定子上, 而 DC 电机的发热源位于转子上, 因此 BLDC 的散热更为容易
- 降低了音频噪声和电磁噪声

无刷直流电机具有许多不同的种类, 它们的区别在于:

- 定子中的相数不同
- 转子上的极数不同
- 转子和定子之间的相对位置不同 (即内转子和外转子结构的区别)

本应用笔记讨论了三相电机。AN1178, 《智能风机控制》(DS01178A_CN) 则对两相电机进行了讨论, 而单相电机则是从两相电机简化、衍生而来。

背景知识

对三相无刷直流电机具有的全面介绍, 请参阅应用笔记“*Brushless DC Motor Control Made Easy*”(DS00857)。AN857 对无刷直流电机及如何对其实现位置传感器反馈换相驱动进行了极佳的描述。通过采用高性能比较器模式和一些新的软件技术, 此应用笔记展示了具有更高性能的无位置传感器换相控制策略。

电机控制

BLDC 电机控制包括两部分。第一部分是使电机以最高效的方式进行换相。第二部分是使电机速度控制在设定的参数范围内。本应用笔记的目的在于展示如何在低成本单片机上实现高性能的无位置传感器控制技术。所有演示软件都在开环速度模式下运行。

硬件

BLDC 系统的硬件可分解为以下部分:

- 电机功率驱动器
- 使用反电势检测来检测转子位置
- 电流监测
- 单片机
- 单片机电源
- 速度设定点输入

电机功率驱动器

所有 BLDC 电机都需要三个半桥型驱动器。每个半桥驱动器控制电机的一相, 如图 1 所示:

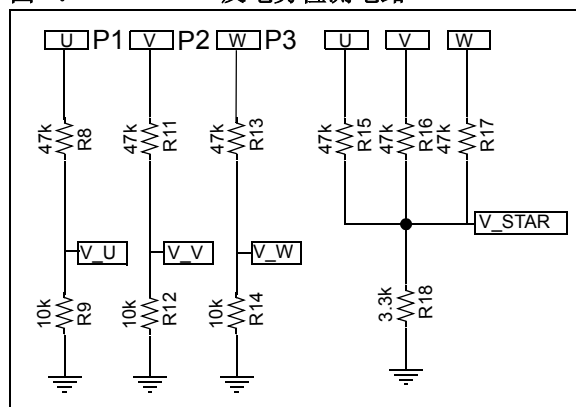


在该示例原理图中，采用三个P沟道MOSFET对从+Vcc端流进电机每一相的电流进行控制。另外，还有三个N沟道MOSFET对从电机每一相流入地的电流进行控制。在N沟道MOSFET和地之间设置有一个小电阻(R7)，用来对流经电机的电流进行检测，其两端的电压与流经该电阻的电流成正比。三个BJT晶体管用来驱动P沟道MOSFET。N沟道MOSFET则由PIC® MCU的I/O引脚进行驱动。对于较小功率的MOSFET和双极型晶体管输出级，通常不需要MOSFET驱动器。

反电势检测

为获得转子当前位置，系统需要采用某种转子位置检测环节。在有位置传感器的系统中，转子位置的检测是通过一系列霍尔效应传感器来实现的，霍尔效应传感器能够感知转子永磁磁极的位置。对于无位置传感器的设计，则是通过检测磁极何时从非导通相绕组下通过来获取转子位置信息。在每一个换相周期中，将有一相绕组处于不导通状态，因此可检测到转子磁极从在该绕组经过的时刻。以下电路采用自偏置形式，且使用一个比较器来实现反电势位置检测。

图 2: 反电势检测电路



注意到反电势检测电路包括四个单元，其中三个具有相同的电路结构。这些单元的目的在于检测过零事件，即使VDD电压处于变化过程中。可通过两种简单的方法来检测正弦波反电势的中点（过零点）。第一种方法是对其取反并将其与原波形进行比较。此时两波形的相交的地方即为反电势波形的中点。第二种方法是将反电势波形幅值缩小并将其与三相中点幅值缩小后的波形进行比较。同样两波形的交点则是反电势波形的中点。第二种方法最为简单，因为它只需单个比较器和一些电阻。由于电机是一个三相系统，每一个电周期存在六个过零点，即三个上升沿过零点和三个下降沿过零点。当换相发生时，通过写入单片机中的CMxCON0 SFR可选择三

相输入中的一相。为降低成本，通常比较器输入端没有配置硬件滤波器，因此电机运行时造成的噪声可能导致虚假的过零点。解决方法是采用基于软件实现的择多检测器。为使择多检测器的实现更为简化，CMxCON0寄存器中的极性位在每一次换相时进行翻转，这将使得所有比较器输出上的过零看似一个下降沿。

电流监测

在任何电机控制系统中，电流监测都是一个比较好的功能，而对于BLDC电机则显得更为必要。电流监测具有以下优点：

- 大电流且没有出现过零点，表示转子堵转
- 过电流限制
- 转矩控制

电流监测功能可通过在半桥开关单元对地回路中加入一个小的检流电阻来实现。如果检流电阻很小，则可能需要采用运算放大器。

最简单的过电流监测器只需对单片机进行复位并重新启动换相。图1中显示了该方法。电流检测电阻用来驱动Q7的基极。如果外部MCLR被使能，该晶体管将导致单片机复位。如果外部MCLR未被使能，则软件可进一步对该输入进行查询，并在检测到过电流条件时采取相应的保护措施。

软件

软件将实现以下任务：

- 起动电机
- 检测过零点
- 定子绕组换相
- 调节换相频率使之与电机转速匹配

电机起动

电机起动是无传感器控制最需技巧的部分。最简单的方法是通过低速率、低占空比的方式对电机进行换相从而起动电机。换相应与转子位置保持一定关系，这样在某一点过零检测器将能发现过零点。一旦检测到过零点，转子将开始与换相保持同步，从而进入正常运行模式。该方法非常简便，但存在一些问题：

- 电机可能进行无规律旋转直到获得同步
- 电机可能同步于实际转速的谐波转速上
- 电机起动过程可能耗时较长

为克服这些缺陷，可采用其他一些方法确定转子堵转位置并立即从该点启动换相。

对于许多电机，一种简单的方法是在过零点之后延迟相应的时间进行强迫换相，这样可以获得令人满意的性能；本应用笔记中也采用了这一方法。

过零检测器

过零检测电路应能按照不同换相序列进行同步切换以选择不同的比较器输入，并对比较器的输出进行监控。比较器输出使用择多检测器进行滤波。该滤波器由换相序列中的事件进行驱动，且在高电平到低电平跳变时进行触发。一旦检测到跳变事件，即进行换相。

过零择多检测器

在无噪声系统中，过零事件可通过观测反电势检测比较器的输出何时从 1 跳变到 0 来确定。高压条件下对大电流进行开关控制将会在系统中引入大量噪声（见图3）。在这样的环境下准确确定过零事件需采用某种形式的滤波方法来抑制噪声影响。采用分立元件构成滤波器的方案会带来太大延迟，使其难以应用于高速电机系统。分立的滤波器还存在温漂问题，从而导致对延迟问题的处理复杂化。一个较好的滤波器方案是延时具有可预测性且不受环境影响。

择多滤波器可通过软件实现。软件滤波器具有固定且可预测的延时，且不会受到环境的影响。滤波器使用一系列比较器输出采样来检测过零事件。如果前半部分采样值中的大多数为 1 而后半部分采样值中的大多数为 0，则认为出现了过零事件。对于 6 采样窗口，当前三次采样值中有 2 或 3 次为 1 而后三次采样值中有 2 或 3 次为 0，这表明检测到过零事件。表 1 显示了所有满足这些条件的可能组合。

图 3: PWM 噪声环境下的典型过零事件

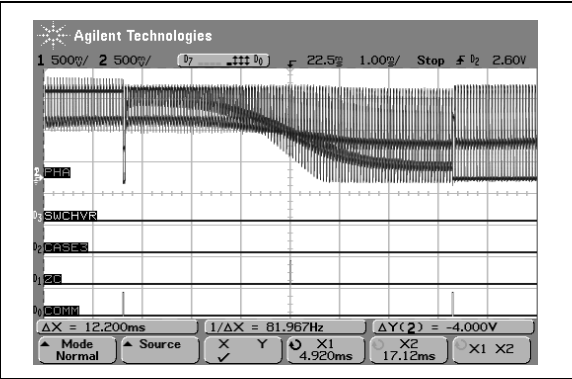


表 1: 过零事件

位模式	对应的数值
011000	24
011001	25
011010	26
011100	28
101000	40
101001	41
101010	42
101100	44
110000	48
110001	49
110010	50
110100	52
111000	56
111001	57
111010	58
111100	60

每一个位模式中的最高位是这一系列中的第一个采样值。当进行每一次新的采样时，它将占据最低位而其他所有数据位左移以腾出空间。移位后的结果是最高位被丢失。实际上，位模式向左移到通过 6 采样窗口。

当采样值通过采样窗口，择多滤波器在软件中是基于以下原理进行实现的。考虑到采样窗口的初始位为全零。当进行了一次逻辑高电平的采样，采样值将被左移进入滤波器采样窗口，这样导致窗口中的全部值为 1。当进行新的采样时，其值将移位进入窗口，并将现有的值左移。如果采样值为 1，而其他所有后续采样为零，在它移出和窗口值返回零之前，则窗口中的值开始为 1，随后变为 2、4、8、16 直到最后为 32。窗口值将保持为零直至进行另一次逻辑高电平采样。对于每一次采样，窗口值首先加倍，随后将新采样值的逻辑电平加入窗口值。例如，窗口值为 4 时进行的采样为逻辑高电平，则窗口值将变为 8 加 1 为 9。在下次采样时，9 将通过左移而加倍并加入新的采样值，这样结果将是 18 或 19，取决于新的采样是否为逻辑高电平。

初看起来，也许有人会认为择多滤波器可通过采用采样窗口寻址查询表的方法来构建。符合择多标准的地址将从表中返回过零事件的指示标志。这一点是可行的，但一些位模式在移过窗口时可能返回多个过零事件。这一问题也可通过在检测到过零事件后清除采样窗口的方法来解决。但存在两个问题：第一，某些模式可能永远不会出现；第二，清除采样窗口需要时间。

对于第一种情形，考虑到模式 60 在下次采样发生时可能变为模式 56 或模式 57，这些模式都将返回事件标志。这表明择多标准表存在问题，模式 56 确实是一个无噪声的过零事件而模式 57 则是紧邻的第二个。如果表中存在模式 60，则实际的事件模式 56 将不可能出现。一个简单的方法是将模式 60 从表中移除。但这不是唯一存在问题的模式。模式 28 也可能在下次采样发生时变成模式 56 或模式 57。模式 28 同样会导致模式 56 不会出现。实际上，还存在许多其他相类似的情况。

表 2: 事件值

位模式	对应的数值	后续值	先前值
011000	(24)	49*, 48*	44*, 12
011001	(25)	51, 50*	44*, 12
011010	(26)	53, 52*	45, 13
011100	(28)	57*, 56*	46, 14
101000	(40)	17, 16	52*, 20
101001	(41)	19, 18	52*, 20
101010	42	21, 20	53, 21
101100	44	25*, 24*	54, 22
110000	(48)	33, 32	56*, 24*
110001	(49)	35, 34	56*, 24*
110010	(50)	37, 36	57*, 25*
110100	52	41*, 40*	58*, 26*
111000	56	49, 48	60*, 28*
111001	57	51, 50*	60*, 28*
111010	58	53, 52*	61, 29
111100	(60)	57*, 56*	62, 30

表 2 列举了所有事件值及其先前和后续的值。如果某一个事件值的前面或后面也是另一个事件值，则应考虑将其从表中移除。进行移除选择时，应衡量那一个事件值最能表征实际过零事件。移除多余的事件值还能避免由于先前误检测从而导致过零事件误判的情况。表中由括号标明的值将被星号指示的先前或后续事件值所覆盖，因此应从事件表中移除。

对于某些事件模式，如果该事件值的某一先前值也已被移除，此时移除该事件值的必要性可能并不明显。例如，事件值 50 被移除，这是因为它可能被先前值 57 所覆盖。然而，事件值 50 将不会被先前的事件值 25 所覆盖，因此为该值也已被移除。移除事件值 25 是因为它可能被先前的事件值 44 和非事件值 12 所覆盖。如果事件值 25 仍保留在表中，它将在先前值 12 出现后触发一个虚假的事件值，因此必须将其移除。同样可知，如果事件值 50 仍保留在表中，非事件值 12 将通过事件值 25 的传递最终触发事件 50。基于这个原因，必须移除事件值 50。同样道理，还应移除事件值 49、48、41 和 40。

通过在每一个地址单元放置一个与过零事件对应的事件标志来构成查询表。这一标志是一个特殊表值，我们将在后面对其进行讨论。对该表地址进行 6 位截取处理以获得相对地址，并将该地址乘以 2。将处理后的地址值装入所有其他表单元中，这样在对每一个位进行采样时可获得一个简便的算法对事件表进行操作。算法在前一个表地址单元内容中加入了采样位以形成新的表地址。如果该新的地址单元包含特殊标志，这表明已检测到过零事件且将执行换相操作。

表中包含 64 个地址（地址 0 至 63），但只是用了 6 个位。过零事件标志是一个为 1 的值。这样，表单元中为 1 的值表明一个过零事件，且将下一个查询表中单元中的值临时设为 1。该临时设置地址值将被换相子程序清零，以便采样窗口能够开始刷新以寻找下一个过零事件。表 3 列举了最终的择多滤波器表。

表 3: 最终的择多滤波器表

表地址	表内容	表地址	表内容
0	0	32	0
1	2	33	2
2	4	34	4
3	6	35	6
4	8	36	8
5	10	37	10
6	12	38	12
7	14	39	14
8	16	40	16
9	18	41	18
10	20	42	1
11	22	43	22
12	24	44	1
13	26	45	26
14	28	46	28
15	30	47	30
16	32	48	32
17	34	49	34
18	36	50	36
19	38	51	38
20	40	52	1
21	42	53	42
22	44	54	44
23	46	55	46
24	48	56	1
25	50	57	1
26	52	58	1
27	54	59	54
28	56	60	56
29	58	61	58
30	60	62	60
31	62	63	62

换相角

理想的换相时间是当转子磁极距离上一次过零点处 30 度电角度后（参见图 4）。由于线圈激励需要一定时间，因此采用略微超前的换相角具有更佳的效果。为使得系统较为简化，本应用笔记采用两邻过零点之间时间间隔的 50% 作为换相点。这一时间对应于 30 电角度。它非常适用于许多小的电机。

换相角按以下原则进行计算：

- 对换相时间进行 16 单元移动平均值计算。
 - 将移动平均值除以 2。
- 平均值处理可作为低通滤波器以减少换相时序中出现的抖动。过度的抖动将增加电流损耗并降低最大速度。

换相

通过将以下表中的值写入比较器 CCP 和 PORT 寄存器可方便地实现电机换相。每一个表中的 8 个入口可避免系统进入错误的表指针。

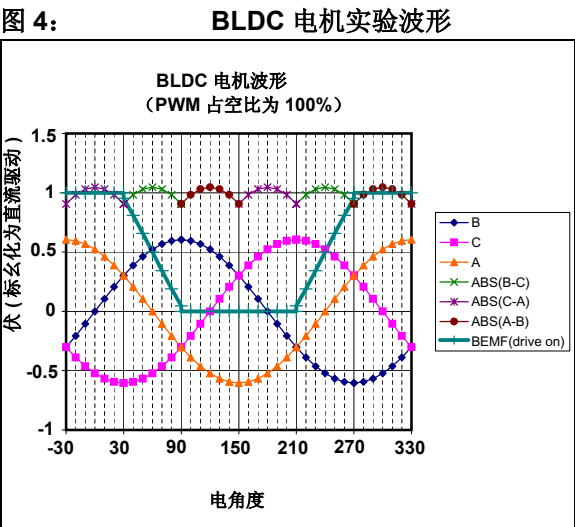


表 4 至表 6 显示了换相顺序：

表 4: 换相顺序（三个表之一）

CM2CON	C2ON	C2OUT	C2OE	C2POL	-	C2R	C2CH1	C2CH0		SAMPLE
00	假	假	假	假	假	假	假	假		
93	真	假	假	真	假	假	真	真		C12IN3
81	真	假	假	假	假	假	假	真	反极性	C12IN1
90	真	假	假	真	假	假	假	假		C12IN0
83	真	假	假	假	假	假	真	真	反极性	C12IN3
91	真	假	假	真	假	假	假	真		C12IN1
80	真	假	假	假	假	假	假	假	反极性	C12IN0
00	假	假	假	假	假	假	假	假		

表 5: 换相顺序（三个表之二）

CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0		
00	假	假	假	假	假	假	假	假		
cc	真	真	假	假	真	真	假	假	全桥	高电平有效
4e	假	真	假	假	真	真	真	假	半桥	P1A 和 P1C 低电平有效
4e	假	真	假	假	真	真	真	假	半桥	P1A 和 P1C 低电平有效
0c	假	假	假	假	真	真	假	假	单输出	高电平有效
0c	假	假	假	假	真	真	假	假	单输出	高电平有效
cc	真	真	假	假	真	真	假	假	全桥	高电平有效
00	假	假	假	假	假	假	假	假		

表 6: 换相顺序（三个表之三）

PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	
00	假	假	假	假	假	假	假	假	
04	假	假	假	假	假	真	假	假	RA2 为高电平
04	假	假	假	假	假	真	假	假	RA2 为高电平
10	假	假	假	真	假	假	假	假	RA4 为高电平
10	假	假	假	真	假	假	假	假	RA4 为高电平
20	假	假	真	假	假	假	假	假	RA5 为高电平
20	假	假	真	假	假	假	假	假	RA5 为高电平
00	假	假	假	假	假	假	假	假	

使用换相表可大大简化换相任务。应用于不同硬件时需对这些表进行更新以反映硬件的实际情况。

可灵活配置的 PWM 和比较器是实现低成本 BLDC 控制的关键单元。

结论

单片机具有的灵活特性和通过软件实现的择多滤波功能使得三相 BLDC 的无位置传感器控制系统可在低成本单片机上进行实现。这对于成本敏感的应用是理想的解决方案。

注:

附录 1 软件

软件许可协议

Microchip Technology Incorporated（以下简称“本公司”）在此提供的软件旨在向本公司客户提供专门用于 Microchip 生产的产品的软件。

本软件为本公司和 / 或其供应商所有，并受到适用的版权法保护。版权所有。使用时违反前述约束的用户可能会依法受到刑事制裁，并可能由于违背本许可的条款和条件而承担民事责任。

本软件是按“现状”提供的。任何形式的保证，无论是明示的、暗示的或法定的，包括但不限于有关适销性和特定用途的暗示保证，均不适用于本软件。对于在任何情况下、因任何原因造成的特殊的、附带的或间接的损害，本公司概不负责。

MAIN.C

```
/*
*****
*/
/*
*****
*/
/*
*****
*/
/* Microchip Technology Incorporated（以下简称“本公司”）在此提供的软件旨在向本公司客户
提供专门用于 Microchip 生产的产品的软件。
*/
/*
*****
*/
/*
*****
*/
/* 本软件为本公司和 / 或其供应商所有，并受到适用的版权法保护。版权所有。使用时违反前述约束的用户
可能会依法受到刑事制裁，并可能由于违背本许可的条款和条件而承担民事责任。
*/
/*
*****
*/
/*
*****
*/
/* 本软件是按“现状”提供的。任何形式的保证，无论是明示的、暗示的或法定的，包括但不限于有关适
销性和特定用途的暗示保证，均不适用于本软件。对于在任何情况下、因任何原因造成的特殊的、附带
的或间接的损害，本公司概不负责。
*/
/*
*****
*/
/*
*****
*/
/*
*****
*/
/* 16F690 BLDC 速度控制
*/
/*
*****
*/
/* 作者: Dieter Peter
*/
/* 公司: Microchip Technology Inc.
*/
/* 版本: 1.0
*/
/* 日期: 11/08/2007
*/
/*
*****
*/

#include <htc.h>
#include "main.h"
#define _690

#ifdef _690
    _CONFIG (FCMDIS & IESODIS & BORDIS & UNPROTECT & MCLREN & PWRTEN & WDTDIS & INTIO);
#endif

#ifdef _616
    _CONFIG (OSC_8MHZ & BORDIS & UNPROTECT & MCLREN & PWRTEN & WDTDIS & INTIO);
#endif
```

附录 A

```
const unsigned char cCM2CON0[8]={0x00,0x93,0x81,0x90,0x83,0x91,0x80,0x00};
//      !V_W   V_V   !V_U   V_W   !V_V   V_U

const unsigned char cPORTA[8]={0x00,0x04,0x04,0x10,0x10,0x20,0x20,0x00};
//  U_H U_H   V_H   V_H   W_H   W_H

// 通过 PSTRCON 可将低端 PWM 信号切换至不同的可用 I/O,
// 或者在单输出、全桥正向和全桥反向模式中进行选择

const unsigned char cCCP1CON[8]={0x00,0xCC,0x4E,0x4E,0x0C,0x0C,0xCC,0x00};
//  V_L W_L   W_L   U_L   U_L   V_L
const unsigned char cPSTRCON[8]={0x00,0x02,0x08,0x08,0x01,0x01,0x02,0x00};
//      V_L   W_L   W_L   U_L   U_L   V_L

// 该简单的择多滤波器用于反电势检测
const unsigned char cBEMF_FILTER[64]={ 00,02,04,06,08,10,12,14,16,18,20,22,24,26,28,30,
                                         32,34,36,38,40,42,44,46,48,50,52,54,56,58,60,62,
                                         00,02,04,06,08,10,12,14,16,18,01,22,01,26,28,30,
                                         32,34,36,38,01,42,44,46,01,01,01,54,56,58,60,62};

// 通用变量

unsigned int adc_result;

// 换相参数

signed char      comm_state;
unsigned int      comm_time;
bit              comm_done,comm_dir;

unsigned int      pwm_demand;
unsigned char      bemf_filter;

// 换相变量

unsigned int      comm_time,comm_time_max,comm_timer;
signed int        phase_delay_counter;
unsigned int      phase_delay;
unsigned int      phase_delay_filter=COMM_TIME_MAX<<3;
bit              zc_detected;
bit              rotor_locked;

void commutate(void);
static void interrupt interrupt_handler(void);
char read_adc(void);

void main(void)
{
    OSCCON=INT8MHZ;                // 配置为最高速度。
                                   // 运行于 1/2 速度 (4MHZ)

    OPTION=OPTION_INIT;

    TRISA=TRISA_INIT;             // PORTA 用作速度输入
    PORTA=PORTA_INIT;             // 和高端晶体管换相
```

```

TRISC=TRISC_INIT;           // PORTC 用作低端晶体管换相
PORTC=PORTC_INIT;

comm_dir=0;                  // 旋转方向控制
CM2CON0=0x80;                // 初始的比较器设定

ANSEL=ANSEL_INIT;           // 设置 RA0 作为模拟输入
ANSELH=ANSELH_INIT;         // 将 PORTB 配置为模拟输入（关断）

ADCON1 = ADCON1_INIT;        // 使能 ADC
ADCON0 = ADCON0_INIT;        // 选择 AN0 (RA0)，右对齐，Clock / 16

comm_time_max=COMM_TIME_MAX; // 初始化最大换相时间。
PR2=PR2_INIT;                // 配置定时器 Timer2
T2CON= T2CON_INIT;

pwm_demand=100;              // 将 CCP 模块设置为 PWM 模式
CCP1CON=CCP1CON_INIT;

comm_state=1;                 // 准备在状态 1 启动换相
TMR2IE=1;                     // 使能 Timer2 中断
PEIE=1;
GIE=1;

while(1)
{
    if (comm_done&read_adc()) // 如果已经换相且 ADC 操作结束。
    {
        pwm_demand = adc_result; // 更新 pwm 给定值
        comm_done=0;             // 清除 comm_done 标志以等待换相事件。
    }
}

} // 主程序结束

static void interrupt

interrupt_handler(void)
{
    TMR2IF=0;                  // 只有一个中断源。如果多于一个，加入处理代码
    T2CON=0x04;                // 清除 timer2 中断标志
                                // 初始化 T2 控制以关断后分频器（由换相函数使
                                // 能）
    ++comm_time;               // 更新换相定时器
    if(C2OUT) bemf_filter|=0b00000001; // 将 C2 输出拷贝至 bemf_filter 指针低位字节
                                // LSB
    bemf_filter=cBEMF_FILTER[bemf_filter]; // 执行滤波表查询
    if (bemf_filter&0b00000001) zc_detected=1; // 检查 ODD 结果
    if (zc_detected)           // 已检测到过零事件，因此……
    {
        rotor_locked = 0; // 表明转子未处于堵转状态，且……
        if (!(phase_delay_counter--)) // 按计数值进行换相。
        {
            commutate(); // 换相
        }
    }
}

```

```

        }
    }
    if (comm_time>comm_time_max) // 如果 comm_timer 达到最大值,则表明已等待太
                                // 长时间,必须进行换相.....
    {
        //
        commutate();
        rotor_locked = 1; // 转子可能被堵转 (或至少被丢失)
    }
}

void commutate(void)
{
    T2CON=0x34; // 换相时禁止滤波
    CCP1CON=CCP1CON[comm_state]; // 查询 CCP1CON 状态
    PORTA=cPORTA[comm_state]; // 查询 PORTA 状态
    CM2CON0=cCM2CON0[comm_state]; // 查询 CM2CON0 状态
    phase_delay_filter+=comm_time; // 对 comm_time 执行 32 点的移动平均计算
    phase_delay=phase_delay_filter>>5; // 并将 phase_delay 设置为平均计算值
    phase_delay_filter-=phase_delay; // phase_delay 计数器为 phase_delay/2
    phase_delay_counter=phase_delay>1; // 这将换相时间设置为两个相邻过零点之间的中间
    // 点
    // 清零状态变量
    zc_detected=0;
    bemf_filter=0;
    comm_time=0;
    comm_done=1;
    CCP1L=pwm_demand>>1; // 更新 PWM 占空比周期
    if (comm_dir) // 更新 comm_state 变量
    { // 使用 comm_dir 来指定换相的方向。
        if (++comm_state>6)
        {
            comm_state=1;
        }
    }
    else
    {
        CM2CON0^=0x10; // 如果是反向,比较器极性取反
        if (--comm_state==0)
        {
            comm_state=6;
        }
    }
} // 换相子程序结束

char read_adc(void)
{
    char result = 0;
    if(GODONE == 0)
    {
        result = 1;
        adc_result = ADRESH * 256 + ADRESL;
        GODONE = 1;
    }
    return result;
}

```

Main.h

```
/*
*****
/*
/*          软件许可协议
/*
/*
/*
/* Microchip Technology Incorporated （以下简称 “本公司”）在此提供的软件旨在向本公司客户
/* 提供专门用于 Microchip 生产的产品的软件。
/*
/*
/*
/* 本软件为本公司和 / 或其供应商所有，并受到适用的版权法保护。版权所有。使用时违反前述约束的用户
/* 可能会依法受到刑事制裁，并可能由于违背本许可的条款和条件而承担民事责任。
/*
/*
/*
/* 本软件是按 “现状” 提供的。任何形式的保证，无论是明示的、暗示的或法定的，包括但不限于有关适
/* 销性和特定用途的暗示保证，均不适用于本软件。对于在任何情况下、因任何原因造成的特殊的、附带
/* 的或间接的损害，本公司概不负责。
/*
/*
/*
/*
*****
/* 16F690 BLDC 速度控制
/*
/*
/* 作者 : Dieter Peter
/* 公司 : Microchip Technology Inc.
/* 版本 : 1.0
/* 日期 : 11/08/2007
/*
/*
*****

// 振荡器
#define INT8MHz          0b01110000
#define OPTION_INIT      0b10001000

// PORTA (PORT)
#define TRISA_INIT       0b00000011
#define PORTA_INIT       0b00000000
#define TRISA_ERROR      0b11111111
#define PORTA_ERROR      0b11111111

// PORTB (PORT)
#define TRISB_INIT       0b00000000
#define PORTB_INIT       0b00000000
#define TRISB_ERROR      0b11111111
#define PORTB_ERROR      0b11111111

// PORTC (PORT)
#define TRISC_INIT       0b00001011
#define PORTC_INIT       0b00000000
#define TRISC_ERROR      0b11111111
#define PORTC_ERROR      0b11111111

// A/D 和比较器
#define CM1CON0_INIT     0b10000000
#define ANSEL_INIT       0b10110011
#define ANSELH_INIT      0b00000010
#define ADCON0_INIT      0b10000001
#define ADCON1_INIT      0b01010000
```

附录 A

```
// PWM
#define CCP1CON_INIT    0b000001100
#define T2CON_INIT      0b000000100
#define PR2_INIT        0x7F           // 64  $\mu$ s PWM 周期
// 用于起动的最大换相时间
#define COMM_TIME_MAX 0x400
```

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、Microchip 徽标、Accuron、dsPIC、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、rfPIC、SmartShun 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL、SmartSensor 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、In-Circuit Serial Programming、ICSP、ICEPIC、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、PICkit、PICDEM、PICDEM.net、Pictail、PIC³² 徽标、PowerCal、PowerInfo、PowerMate、PowerTool、REAL ICE、rfLAB、Select Mode、Total Endurance、WiperLock 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2008, Microchip Technology Inc. 版权所有。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC[®] MCU 与 dsPIC[®] DSC、KEELOQ[®] 跳码器件、串行 EEPROM、单片机外设、非易失性存储器及模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外，Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

全球销售及服务中心

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:
<http://support.microchip.com>
网址: www.microchip.com

亚特兰大 **Atlanta**

Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

波士顿 **Boston**

Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 **Chicago**

Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 **Dallas**

Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 **Detroit**

Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 **Kokomo**

Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 **Los Angeles**

Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 **Santa Clara**

Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

加拿大多伦多 **Toronto**

Mississauga, Ontario,
Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 **Asia Pacific Office**

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京

Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 - 成都

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 香港特别行政区

Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 南京

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳

Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 - 武汉

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 厦门

Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 珠海

Tel: 86-756-321-0040
Fax: 86-756-321-0049

台湾地区 - 高雄

Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 - 新竹

Tel: 886-3-572-9526
Fax: 886-3-572-6459

亚太地区

澳大利亚 **Australia - Sydney**

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 **India - Bangalore**

Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

印度 **India - New Delhi**

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 **India - Pune**

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 **Japan - Yokohama**

Tel: 81-45-471-6166
Fax: 81-45-471-6122

韩国 **Korea - Daegu**

Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 **Korea - Seoul**

Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 **Malaysia - Kuala Lumpur**

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 **Malaysia - Penang**

Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 **Philippines - Manila**

Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 **Singapore**

Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 **Thailand - Bangkok**

Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 **Austria - Wels**

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 **Denmark - Copenhagen**

Tel: 45-4450-2828
Fax: 45-4485-2829

法国 **France - Paris**

Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 **Germany - Munich**

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 **Italy - Milan**

Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 **Netherlands - Drunen**

Tel: 31-416-690399
Fax: 31-416-690340

西班牙 **Spain - Madrid**

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 **UK - Wokingham**

Tel: 44-118-921-5869
Fax: 44-118-921-5820