

**MICROCHIP**

**PICmicro®**  
***DC Motor Control***  
***Tips 'n Tricks***



---

---

## Table of Contents

---

---

### Tips 'N Tricks Introduction

TIP #1:	Brushed DC Motor Drive Circuits .....	2
TIP #2:	Brushless DC Motor Drive Circuits .....	5
TIP #3:	Stepper Motor Drive Circuits .....	9
TIP #4:	Drive Software .....	13
TIP #5:	Writing a PWM Value to the CCP Registers with a Mid-range PICmicro <sup>®</sup> MCU.....	17
TIP #6:	Current Sensing .....	19
TIP #7:	Position/Speed Sensing .....	23

# Tips ‘n Tricks

---

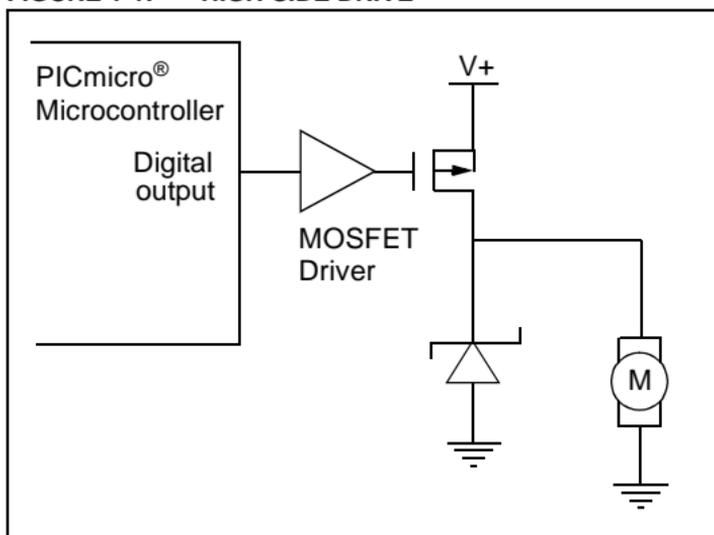
### TIPS 'N TRICKS INTRODUCTION

Every motor control circuit can be divided into the drive electronics and the controlling software. These two pieces can be fairly simple or extremely complicated depending upon the motor type, the system requirements and the hardware/software complexity trade-off. Generally, higher performance systems require more complicated hardware. This booklet describes many basic circuits and software building blocks commonly used to control motors. The booklet also provides references to Microchip application notes that describe many motor control concepts in more detail.

### TIP #1 Brushed DC Motor Drive Circuits

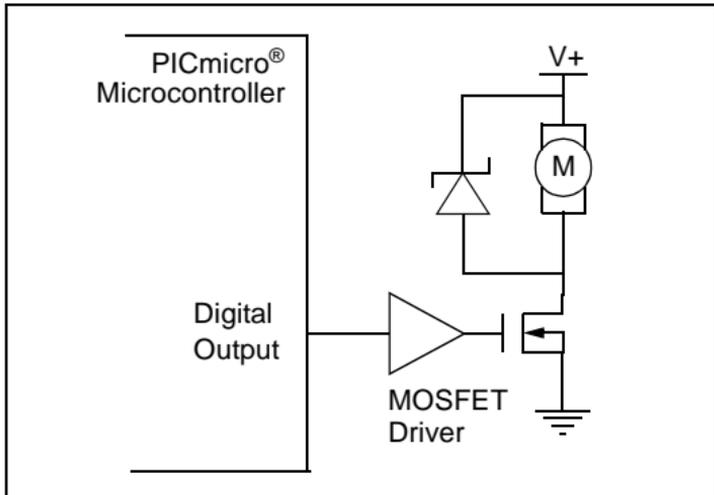
All motors require drive circuitry which controls the current flow through the motor windings. This includes the direction and magnitude of the current flow. The simplest type of motor, to drive, is the Brushed DC motor. Drive circuits for this type of motor are shown below.

FIGURE 1-1: HIGH SIDE DRIVE



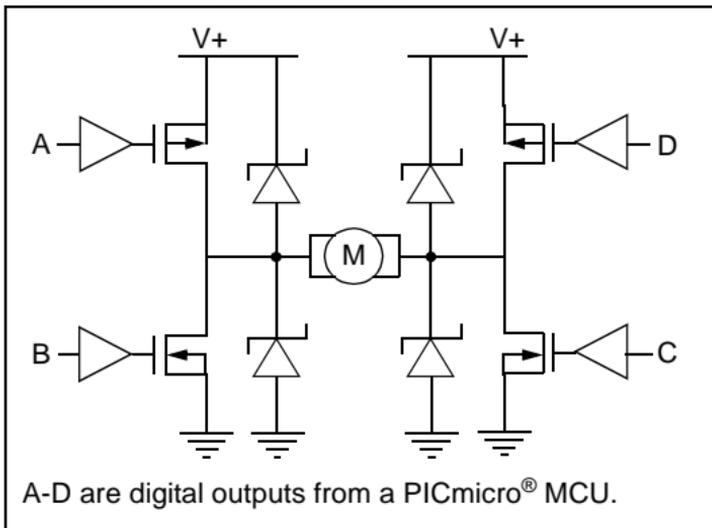
This drive can control a Brushed DC motor in one direction. This drive is often used in safety critical applications because a short circuit cannot turn the motor on.

**FIGURE 1-2: LOW SIDE DRIVE**



This is the lowest cost drive technique because of the MOSFET drive simplicity. Most applications can simply use an output pin from the PICmicro microcontroller to turn the MOSFET on.

FIGURE 1-3: H-BRIDGE DRIVE



The H-bridge derived its name from the common way the circuit is drawn. This is the only solid state way to operate a motor in both directions.

Application notes that drive Brushed DC motors are listed below:

- **AN847** – “*RC Model Aircraft Motor Control*” (DS00847)
- **AN893** – “*Low-cost Bidirectional Brushed DC Motor Control Using the PIC16F684*” (DS00893)
- **AN905** – “*Brushed DC Motor Fundamentals*” (DS00905)

### **TIP #2    Brushless DC Motor Drive Circuits**

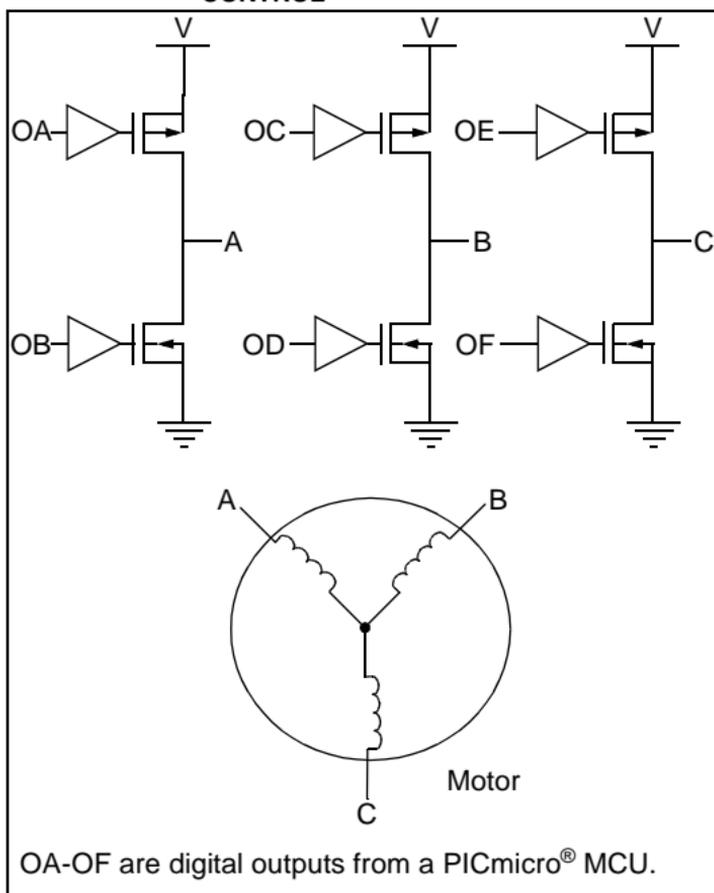
A Brushless DC motor is a good example of simplified hardware increasing the control complexity. The motor cannot commutate the windings (switch the current flow), so the control circuit and software must control the current flow correctly to keep the motor turning smoothly. The circuit is a simple half-bridge on each of the three motor windings.

There are two basic types of Brushless DC motors; sensor and sensorless. Because it is critical to know the position of the motor so the correct winding can be energized, some method of detecting the motor position is required. A sensor motor will directly report to the controller, the current position of the motor. Driving a sensor motor requires a look-up table. The current sensor position directly correlates to a commutation pattern for the bridge circuits. A sensorless motor requires that the induced voltage in the un-driven winding be sensed and used to determine the current speed of the motor. Then, the next commutation pattern can be determined by a time delay from the previous pattern.

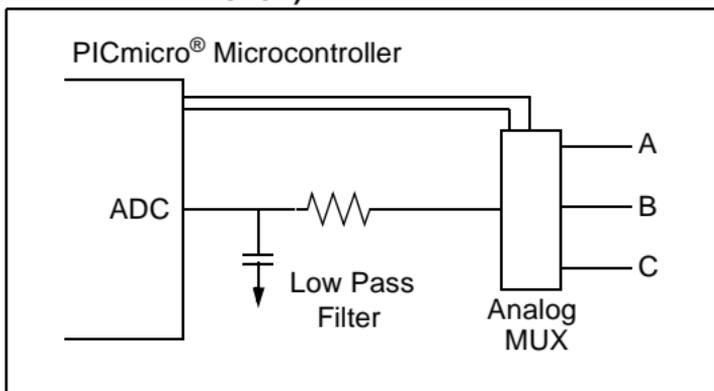
## Tips 'n Tricks

Sensorless motors are simpler to build due to the lack of the sensors, but they are more complicated to drive. A sensorless motor performs very well in applications that don't require the motor to start and stop. A sensor motor would be a better choice in applications that must periodically stop the motor.

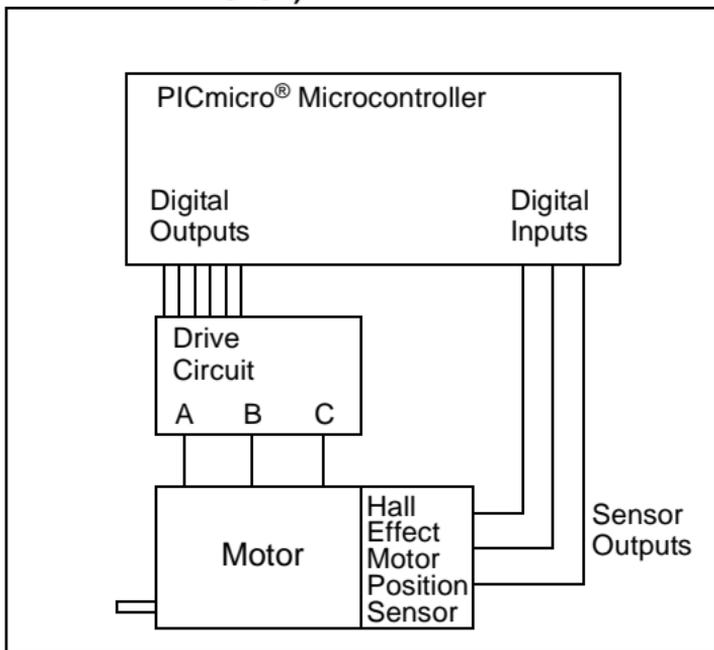
**FIGURE 2-1: 3 PHASE BRUSHLESS DC MOTOR CONTROL**



**FIGURE 2-2: BACK EMF SENSING (SENSORLESS MOTOR)**



**FIGURE 2-3: QUADRATURE DECODER (SENSOR MOTOR)**



## Tips 'n Tricks

---

Application notes describing Brushless DC Motor Control:

- AN857 – “*Brushless DC Motor Control Made Easy*” (DS00857)
- AN885 – “*Brushless DC Motor Fundamentals*” (DS00885)
- AN899 – “*Brushless DC Motor Control Using PIC18FXX31*” (DS00899)
- AN901 – “*Using the dsPIC30F for Sensorless BLDC Control*” (DS00901)

### **TIP #3 Stepper Motor Drive Circuits**

Stepper motors are similar to Brushless DC motors in that the control system must commutate the motor through the entire rotation cycle. Unlike the brushless motor, the position and speed of a stepping motor is predictable and does not require the use of sensors.

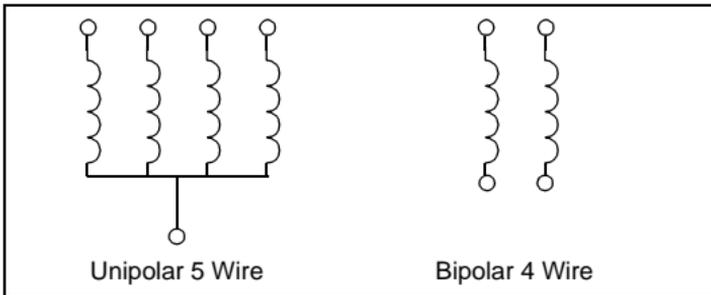
There are two basic types of stepper motors, although some motors are built to be used in either mode. The simplest stepper motor is the unipolar motor. This motor has four drive connections and one or two center tap wires that are tied to ground or  $V_{SUPPLY}$ , depending on the implementation. Other motor types are the bipolar stepper and various combinations of unipolar and bipolar, as shown in Figure 3-1 and Figure 3-2. When each drive connection is energized, one coil is driven and the motor rotates one step. The process is repeated until all the windings have been energized. To increase the step rate, often the voltage is increased beyond the motors rated voltage. If the voltage is increased, some method of preventing an over current situation is required.

## Tips 'n Tricks

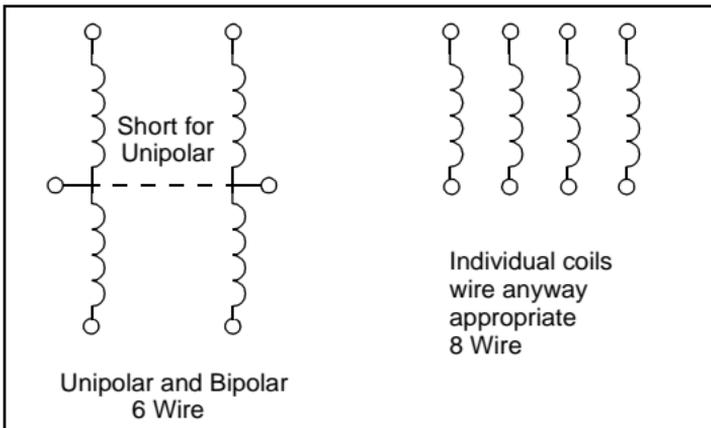
---

There are many ways to control the winding current, but the most popular is a chopper system that turns off current when it reaches an upper limit and enables the current flow a short time later. Current sensor systems are discussed in TIP #6. Some systems are built with a current chopper, but they do not detect the current, rather the system is designed to begin a fixed period chopping cycle after the motor has stepped to the next position. These are simpler systems to build, as they only require a change in the software.

**FIGURE 3-1: 4 AND 5 WIRE STEPPER MOTORS**



**FIGURE 3-2: 6 AND 8 WIRE STEPPER MOTORS**



**FIGURE 3-3: UNIPOLAR MOTOR (4 LOW SIDE SWITCHES)**

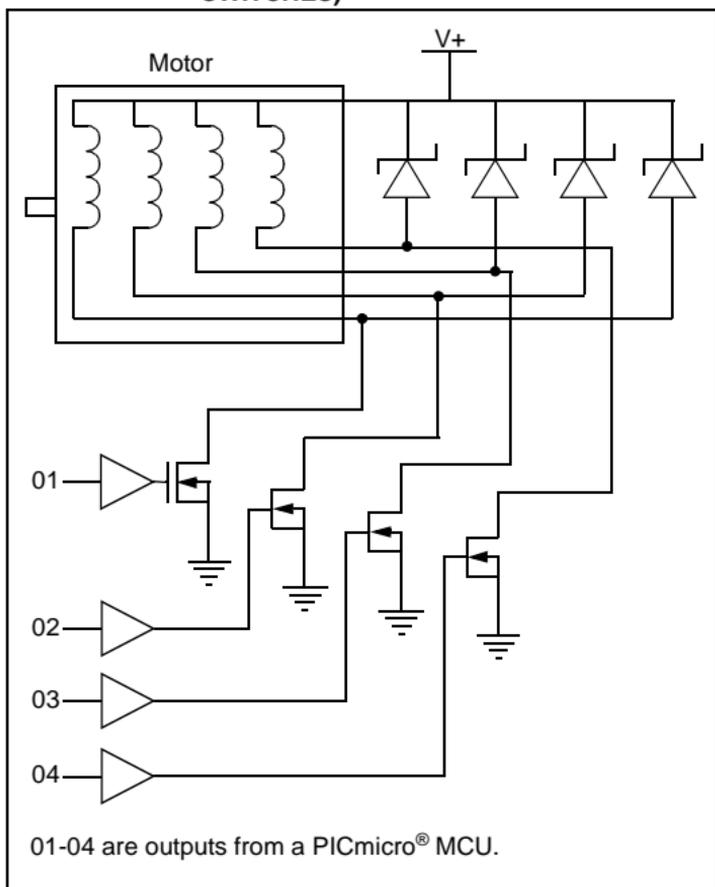
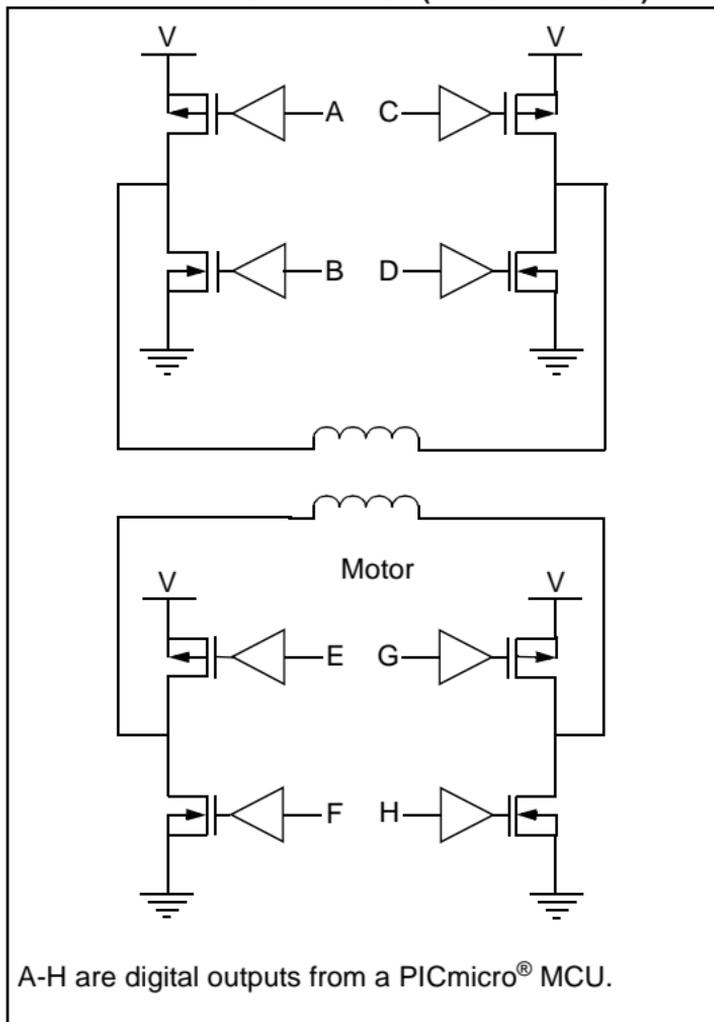


FIGURE 3-4: BIPOLAR MOTOR (4 HALF-BRIDGES)



### **TIP #4 Drive Software**

#### **Pulse-Width Modulation (PWM) Algorithms**

Pulse-Width Modulation is critical to modern digital motor controls. By adjusting the pulse width, the speed of a motor can be efficiently controlled without larger linear power stages. Some PICmicro devices have hardware PWM modules on them. These modules are built into the Capture/Compare/PWM (CCP) peripheral. CCP peripherals are intended for a single PWM output, while the Enhanced CCP (ECCP) is designed to produce the complete H-Bridge output for bidirectional Brushed DC motor control. If cost is a critical design point, a PICmicro device with a CCP module may not be available, so software generated PWM is a good alternative.

The following algorithms are designed to efficiently produce an 8-bit PWM output on the mid-range family of PICmicro microcontrollers. These algorithms are implemented as macros. If you want these macros to be a subroutine in your program, simply remove the macro statements and replace them with a label and a return statement.

## Tips 'n Tricks

---

### EXAMPLE 4-1: 1 OUTPUT 8-BIT PWM

```
pwm_counter equ xxx ;variable
pwm         equ xxx ;variable

set_pwm macro A      ;sets the pwm
                ;setpoint to the
                ;value A

    MOVLW A
    MOVWF pwm
    endm

update_PWM macro      ;performs one update
                    ;of the PWM signal
                    ;place the PWM output
                    ;pin at bit 0 or 7 of
                    ;the port

    MOVF pwm_counter,w
    SUBWF pwm, w      ;if the output
                    ;is on bit 0

    RLF         PORTC,f ;replace PORTC with
                    ;the correct port if
                    ;the output is on bit
                    ;7 of the port
                    ;replace the rlf with
                    ;rrf incf
                    ;pwm_counter,f
```

## EXAMPLE 4-2: 8 OUTPUT 8-BIT PWM

```

pwm_counter equ xxx      ;variable
pwm0         equ xxx      ;
pwm1         equ pwm0+1
pwm2         equ pwm1+1
pwm3         equ pwm2+1
pwm4         equ pwm3+1
pwm5         equ pwm4+1
pwm6         equ pwm5+1
pwm7         equ pwm6+1
output       equ pwm7+1

set_pwm macro A,b;sets pwm b with
                ;the value A

    MOVLW  pwm0
    ADDLW  b
    MOVWF  fsr
    MOVLW  a
    MOVWF  indf
    endm

update_PWM macro                ;peforms one
                                ;update of all 8
                                ;PWM siganls
                                ;all PWM signals
                                ;must be on the
                                ;same port

    MOVF   pwm_counter,w
    SUBWF  pwm0,w
    RLF    output,f
    MOVF   pwm_counter,w
    SUBWF  pwm1,w
    RLF    output,f
    MOVF   pwm_counter,w
    SUBWF  pwm2,w
    RLF    output,f

```

## Tips 'n Tricks

---

### EXAMPLE 4-3: 8 OUTPUT 8-BIT PWM (CONTINUED)

```
MOVF    pwm_counter,w
SUBWF   pwm3,w
RLF     output,f
MOVF    pwm_counter,w
SUBWF   pwm4,w
RLF     output,f
MOVF    pwm_counter,w
SUBWF   pwm5,w
RLF     output,f
MOVF    pwm_counter,w
SUBWF   pwm6,w
RLF     output,f
MOVF    pwm_counter,w
SUBWF   pwm7,w
RLF     output,w
MOVWF   PORTC
INCF    pwm_counter,f
endm
```

### TIP #5 Writing a PWM Value to the CCP Registers With a Mid-range PICmicro® Microcontroller

The two PWM LSB's are located in the CCPCON register of the CCP. This can make changing the PWM period frustrating for a developer.

Example 5-1 through Example 5-3 show three macros written for the mid-range product family that can be used to set the PWM period. The first macro takes a 16-bit value and uses the 10 MSb's to set the PWM period. The second macro takes a 16-bit value and uses the 10 LSB's to set the PWM period. The last macro takes 8 bits and sets the PWM period. This assumes that the CCP is configured for no more than 8 bits.

#### EXAMPLE 5-1: LEFT JUSTIFIED 16-BIT MACRO

```

pwm_tmp    equ xxx ;this variable must be
                ;allocated someplace
setPeriod macro a ;a is 2 SFR's in
                ;Low:High arrangement
                ;the 10 MSb's are the
                ;desired PWM value
    RRF      a,w ;This macro will
                ;change w
    MOVWF   pwm_tmp
    RRF     pwm_tmp,w
    ANDLW   0x30
    IORLW   0x0F
    MOVWF   CCP1CON
    MOVF    a+1,w
    MOVWF   CCP1L
    
```

## Tips 'n Tricks

---

### EXAMPLE 5-2: RIGHT JUSTIFIED 16-BIT MACRO

```
pwm_tmp    equ xxx ;this variable must be
                ;allocated someplace
setPeriod macro a ;a is 2 bytes in
                ;Low:High arrangement
                ;the 10 LSb's are the
                ;desired PWM value

    SWAPF    a,w    ;This macro will
                ;change w

    ANDLW    0x30
    IORLW    0x0F
    MOVWF    CCP1CON
    RLF      a,w
    IORLW    0x0F
    MOVWF    pwm_tmp
    RRF      pwm_tmp,f
    RRF      pwm_tmp,w
    MOVWF    CCP1L
```

### EXAMPLE 5-3: 8-BIT MACRO

```
pwm_tmp    equ xxx ;this variable must be
                ;allocated someplace
setPeriod macro a ;a is 1 SFR

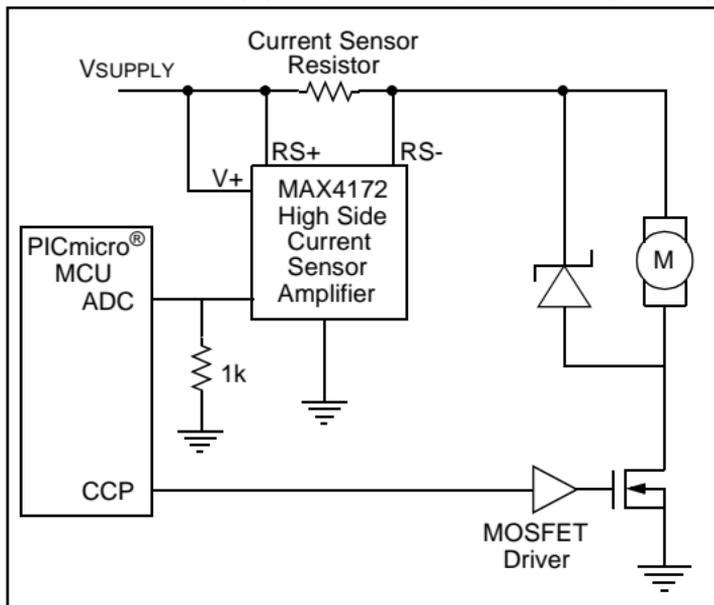
    SWAPF    a,w    ;This macro will
                ;change w

    ANDLW    0x30
    IORLW    0x0F
    MOVWF    CCP1CON
    RRF      a,w
    MOVWF    pwm_tmp
    RRF      pwm_tmp,w
    MOVWF    CCP1L
```

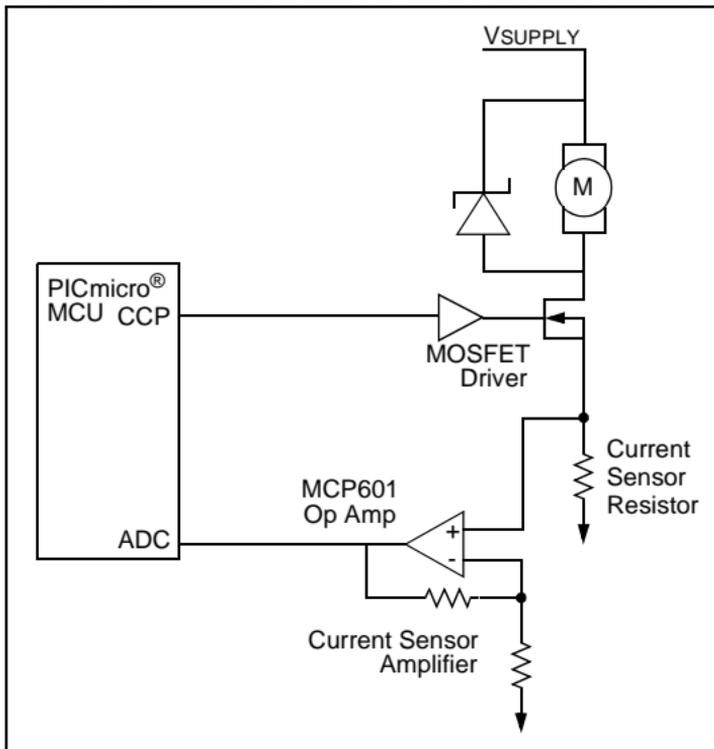
### **TIP #6    Current Sensing**

The torque of an electric motor can be monitored and controlled by keeping track of the current flowing through the motor. Torque is directly proportional to the current. Current can be sensed by measuring the voltage drop through a known value resistor or by measuring the magnetic field strength of a known value inductor. Current is generally sensed at one of two places, the supply side of the drive circuit (high side current sense) or the sink side of the drive circuit (low side current sense). Low side sensing is much simpler but the motor will no longer be grounded, causing a safety issue in some applications. High side current sensing generally requires a differential amplifier with a common mode voltage range within the voltage of the supply.

**FIGURE 6-1: RESISTIVE HIGH SIDE CURRENT SENSING**



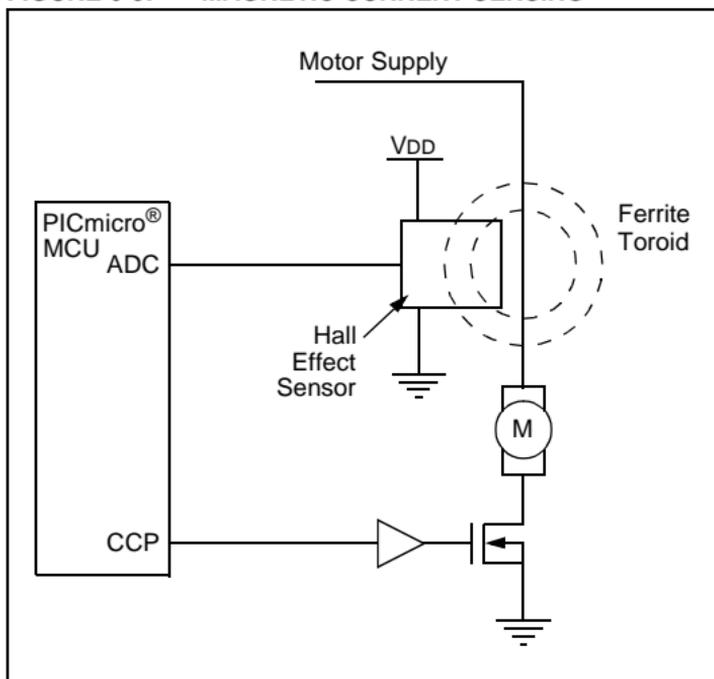
**FIGURE 6-2: RESISTIVE LOW SIDE CURRENT SENSING**



## Tips 'n Tricks

Current measurement can also be accomplished using a Hall Effect Sensor to measure the magnetic field surrounding a current carrying wire. Naturally, this Hall Effect Sensor can be located on the high side or the low side of the load. The actual location of the sensor does not matter because the sensor does not rely upon the voltage on the wire.

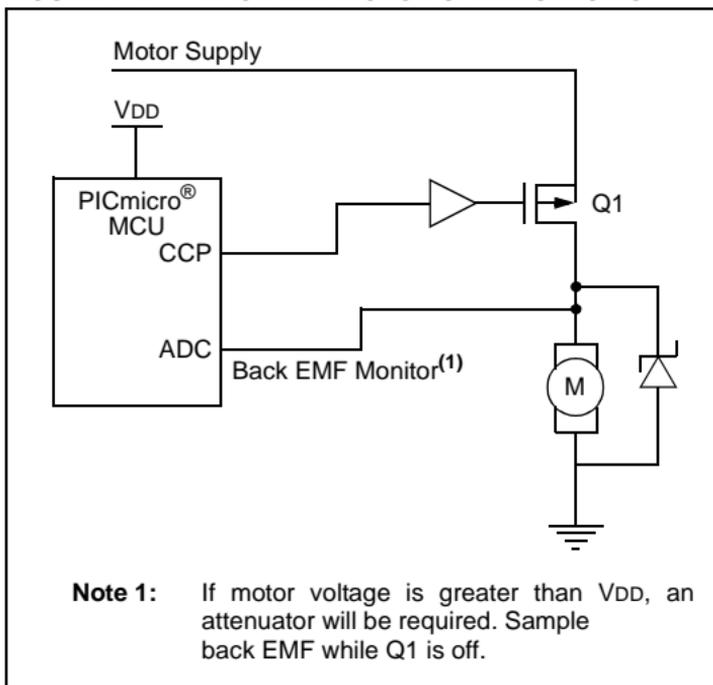
**FIGURE 6-3: MAGNETIC CURRENT SENSING**



## TIP #7 Position/Speed Sensing

The motor RPM can be measured by understanding that a motor is a generator. As long as the motor is spinning, it will produce a voltage that is proportional to the motors RPM. This is called back EMF. If the PWM signal to the motor is turned off and the voltage across the windings is measured, the back EMF voltage can be sensed from there and the RPM's can be known.

**FIGURE 7-1: BACK EMF MOTOR SPEED SENSING**

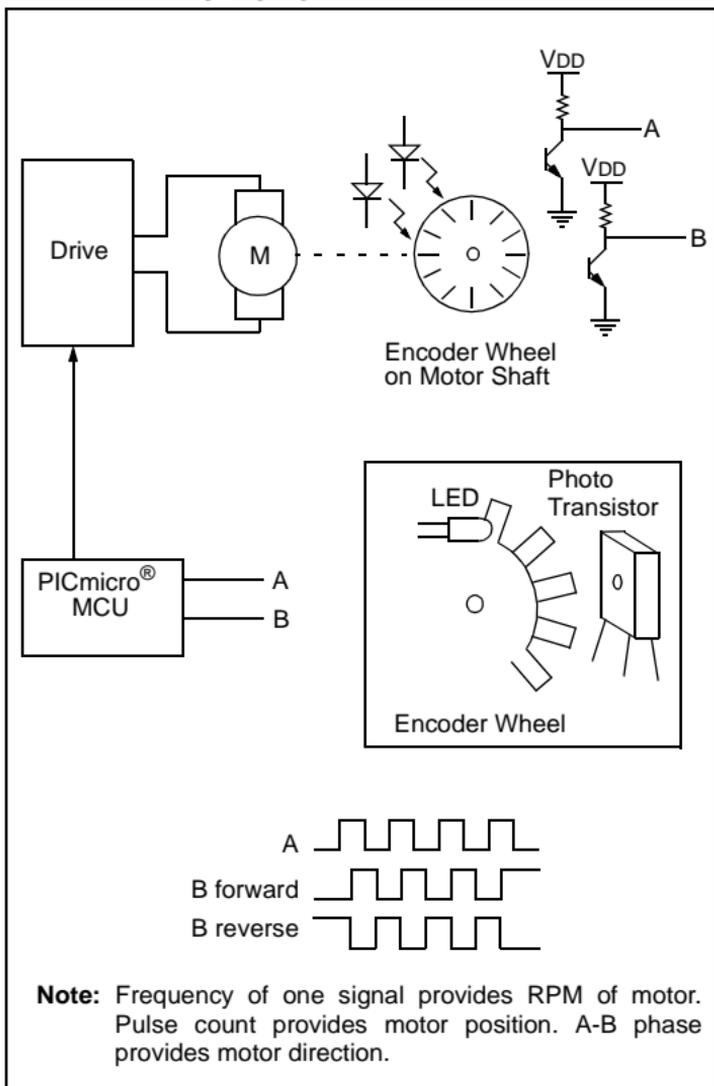


### Rotary Encoder Sensing

Rotary encoders are typically used to provide direct physical feedback of motor position, and/or speed. A rotary encoder consists of a rotary element attached to the motor that has a physical feature, measured by a stationary component. The measurements can yield motor speed and sometimes they can provide a motor position. Rotary encoders are built using many different technologies. The most common type is an optical rotary encoder. The optical rotary encoder is used in the computer mice that have a ball. It is built with an encoder disc that is attached to the motor. The encoder disc has many radial slots cut into the disc at a specific interval. An LED and a photo detector are used to count the slots as they go by. By timing the rate that the slots go by, the speed of rotation can be determined.

Motor position requires a second LED and photo detector. The second sensor pair is mounted so the output pulses are 90° degrees out of phase from the first pair. The two outputs represent the motion of the encoder disc as a quadrature modulated pulse train. By adding a third index signal, that pulses once for each revolution, the exact position of the motor can be known at all times.

**FIGURE 7-2: OPTICAL SPEED/DIRECTION/POSITION SENSING**



## Tips 'n Tricks

---

Quadrature sensing can easily be accomplished in software, but there is generally an upper limit to the RPM. By using a few gates, the sensing can be done partially in hardware and partially in software. The new PIC18FXX31 and dsPIC<sup>®</sup> 16-bit Digital Signal Controller families include an encoder interface that allows MUCH higher RPM motors to be measured with an excellent degree of accuracy.

### **Older Methods of Motor Sensing**

Resolvers and analog tachometers are two older technologies for motor position/velocity sensing. An analog tachometer is simply an electric generator with a linear output over a specified range of RPM's. By knowing the output characteristics, the RPM can be known by simply measuring the voltage across the tachometer terminals.

A resolver is a pair of coils that are excited by an external AC signal. The two coils are at 90° to each other so they pick up the AC signal at different strengths, depending on their orientation. The result is a sine or cosine output related to the angle of the resolver in reference to the AC signal. Inverse cosine/sine will produce the angle of the sensor. This type of sensor can be very accurate and is still used where absolute position must be known.

### Application Note References

- **AN532** – “*Servo Control of a DC Brush Motor*” (DS00532)
- **AN696** – “*PIC18Cxxx/PIC16Cxxx DC Servo Motor*” (DS00696)
- **AN718** – “*Brush-DC Servomotor Implementation using PIC17C56A*” (DS00718)
- **AN822** – “*Stepper Motor Micro-stepping with the PIC18C452*” (DS00822)
- **AN843** – “*Speed Control of 3-Phase Induction Motor Using PIC18 Microcontrollers*” (DS00843)
- **AN847** – “*RC Model Aircraft Motor Control*” (DS00847)
- **AN857** – “*Brushless DC Motor Control Made Easy*” (DS00857)
- **AN885** – “*Brushless DC (BLDC) Motor Fundamentals*” (DS00885)
- **AN899** – “*Brushless DC Motor Control Using the PIC18FXX31*” (DS00899)
- **AN893** – “*Low-cost Bidirectional Brushed DC Motor Control Using the PIC16F684*” (DS00893)
- **AN894** – “*Motor Control Sensor Feedback Circuits*” (DS00894)
- **AN898** – “*Determining MOSFET Driver Needs for Motor Drive Applications*” (DS00898)
- **AN901** – “*Using the dsPIC30F for Sensorless BLDC Control*” (DS00901)

## Tips 'n Tricks

---

- **AN905** – “*Brushed DC Motor Fundamentals*” (DS00905)
- **AN906** – “*Stepper Motor Control Using the PIC16F684*” (DS00906)
- **AN907** – “*Stepper Motor Fundamentals*” (DS00907)

### Motor Control Development Tools

- “*PICDEM™ MC Development Board*” (DM183011)  
Used to evaluate the PIC18FXX3 8-bit Microcontroller family.
- “*dsPIC30F Motor Control Development System*” (DM300020)  
Used to evaluate the dsPIC30F 16-bit Digital Signal Controller family.
- *Motor Control (MC) Graphical User Interface (GUI)*  
The MC-GUI allows user to configure the motor and a wide range of system parameters for a selected motor type.  
The MC-GUI is free and can be downloaded at [www.microchip.com](http://www.microchip.com)

**Visit the Motor Control Design Center at [www.microchip.com/motor](http://www.microchip.com/motor) for additional design resources.**

---

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

The graphics in this document are for illustration only. Microchip reserves the right to modify the contents of its development systems.

### **Trademarks**

The Microchip name and logo, the Microchip logo, Accuron, dsPIC, KEELoQ, MPLAB, PIC, PICmicro, PICSTART, PRO MATE, PowerSmart and rPIC are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AmpLab, FilterLab, microID, MXDEV, MXLAB, PICMASTER, SEEVAL, SmartShunt and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Application Maestro, dsPICDEM, dsPICDEM.net, dsPICworks, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, PICkit, PICDEM, PICDEM.net, PICtail, PowerCal, PowerInfo, PowerMate, PowerTool, rLAB, Select Mode, SmartSensor, SmartTel and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2004, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.



Printed on recycled paper.

---

---

## Worldwide Sales and Service

---

---

### AMERICAS

#### Corporate Office

Tel: 480-792-7200  
Technical Support:  
480-792-7627

#### Atlanta

Tel: 770-640-0034

#### Boston

Tel: 978-692-3848

#### Chicago

Tel: 630-285-0071

#### Dallas

Tel: 972-818-7423

#### Detroit

Tel: 248-538-2250

#### Kokomo

Tel: 765-864-8360

#### Los Angeles

Tel: 949-263-1888

#### San Jose

Tel: 650-215-1444

#### Toronto

Tel: 905-673-0699

### ASIA/PACIFIC

#### Australia

Tel: 61-2-9868-6733

#### China-Beijing

Tel: 86-10-85282100

#### China-Chengdu

Tel: 86-28-86766200

#### China-Fuzhou

Tel: 86-591-7503506

#### China-Hong

#### Kong SAR

Tel: 852-2401-1200

#### China-Shanghai

Tel: 86-21-6275-5700

#### China-Shenzhen

Tel: 86-755-82901380

#### China-Shunde

Tel: 86-765-28395507

#### China-Qingdao

Tel: 86-532-5027355

#### India

Tel: 91-80-22290061

#### Japan

Tel: 81-45-471- 6166

#### Korea

Tel: 82-2-554-7200

#### Singapore

Tel: 65-6334-8870

#### Taiwan

Kaohsiung  
Tel: 886-7-536-4818

#### Taiwan

Tel: 886-2-2717-7175

### EUROPE

#### Austria

Tel: 43-7242-2244-399

#### Denmark

Tel: 45-4420-9895

#### France

Tel: 33-1-69-53-63-20

#### Germany

Tel: 49-89-627-144-0

#### Italy

Tel: 39-0331-742611

#### Netherlands

Tel: 31-416-690399

#### United Kingdom

Tel: 44-118-921-5869

02/17/04







**MICROCHIP**

Microchip Technology Inc.

2355 W. Chandler Blvd. • Chandler, AZ 85224 U.S.A.

Phone: 480-792-7200 • Fax: 480-792-9210

[www.microchip.com](http://www.microchip.com)

© 2004, Microchip Technology Inc., 04/04 DS41233A

