CHAPTER 5
DC Motor Control
Tips 'n Tricks

Table Of Contents

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. The application notes can be found on the Microchip web site at www.microchip.com.

Additional motor control design information can be found at the Motor Control Design Center (www.microchip.com/motor).

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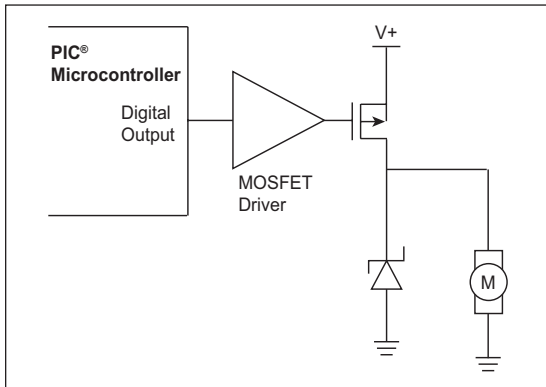
This drive can control a Brushed DC motor in one direction. This drive is often used in safety critical applications because a short circuit at the motor terminals cannot turn the motor on.

Figure 1-2: Low Side Drive
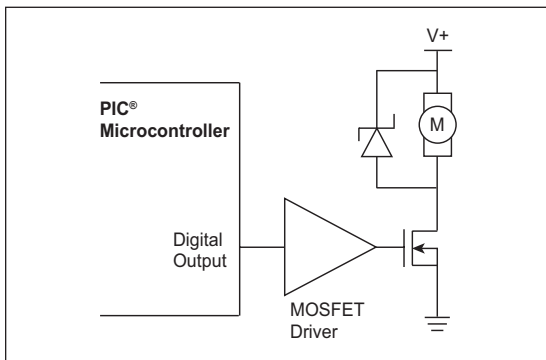
This is the lowest cost drive technique because of the MOSFET drive simplicity. Most applications can simply use an output pin from the PIC® microcontroller to turn the MOSFET on.

Figure 1-3: H-Bridge Drive

A-D are digital outputs from a PIC® MCU.
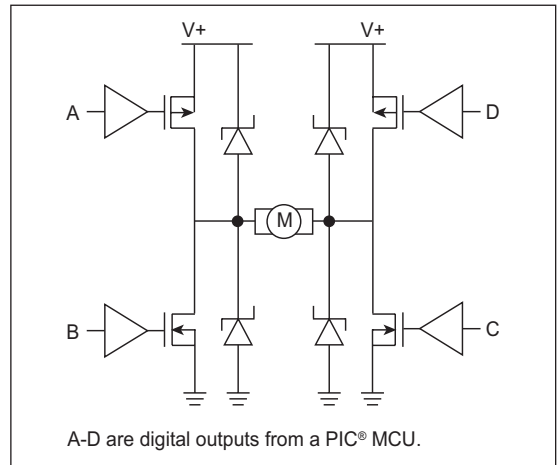
The H-Bridge derived its name from the common way the circuit is drawn. This is the only solid state way to operate a motor in both directions.

Application notes that drive Brushed DC motors are listed below and can be found on the Microchip web site at: www.microchip.com.

- 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 commutation methods for Brushless DC motors; sensored 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 rotor position is required. A motor with sensors will directly report the current position to the controller. Driving a sensored motor requires a look-up table. The current sensor position directly correlates to a commutation pattern for the bridge circuits.

Without sensors, another property of the motor must be sensed to find the position. A popular method for sensorless applications is to measure the back EMF voltage that is naturally generated by the motor magnets and windings. The induced voltage in the un-driven winding can 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.

Sensorless motors are lower cost 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.
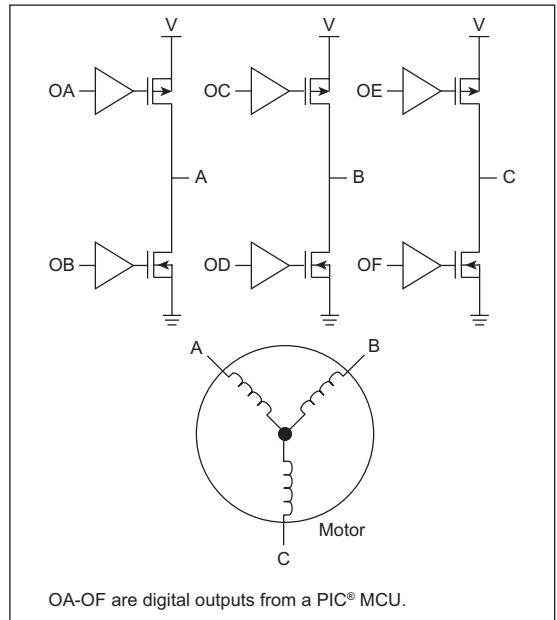
Figure 2-1: 3 Phase Brushless DC Motor Control

OA-OF are digital outputs from a PIC® MCU.

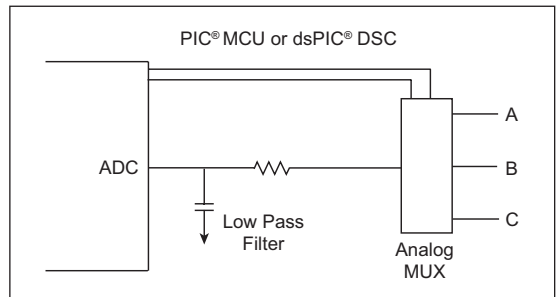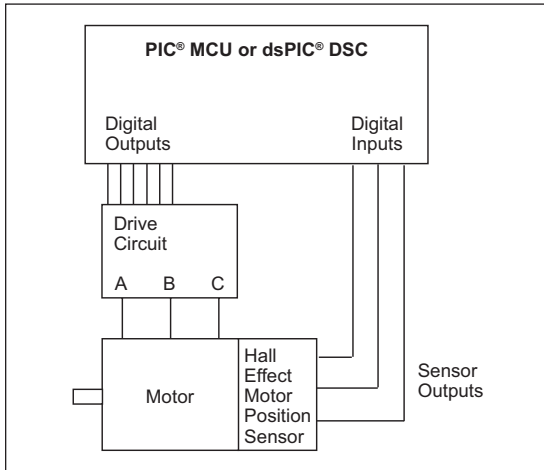**Figure 2-2: Back EMF Sensing (Sensorless Motor)**

Figure 2-3: Quadrature Decoder (Sensor Motor)

Application notes describing Brushless DC Motor Control are listed below and can be found on the Microchip web site at: www.microchip.com.

- 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)
- AN957, "*Sensored BLDC Motor Control Using dsPIC30F201*0" (DS00957)
- AN992, "*Sensorless BLDC Motor Control Using dsPIC30F2010*" (DS00992)
- AN1017, "*Sinusoidal Control of PMSM with dsPIC30F DSC*" (DS01017)
- GS005, "*Using the dsPIC30F Sensorless Motor Tuning Interface*" (DS93005)

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 Vsupply, 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.

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.
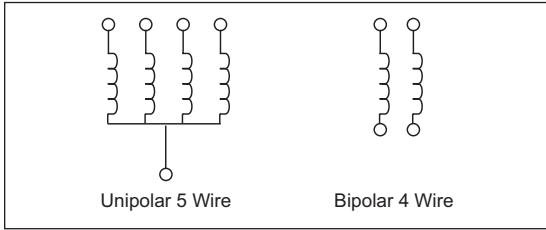
Figure 3-1: 4 and 5 Wire Stepper Motors

Unipolar 5 Wire   Bipolar 4 Wire

Figure 3-2: 6 and 8 Wire Stepper Motors

Short for
Unipolar

Individual coils
wire anyway
appropriate
8 Wire

Unipolar and Bipolar
6 Wire
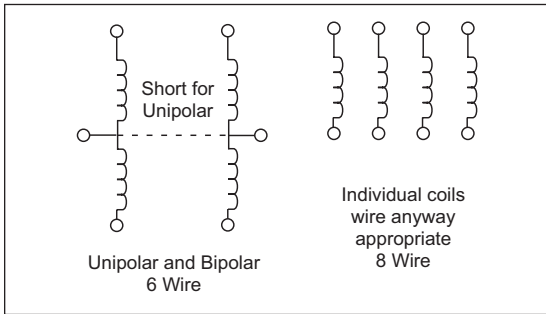
Figure 3-3: Unipolar Motor (4 Low Side Switches)

Motor

V+

01

02

03

04

01-04 are outputs from a PIC® MCU or dsPIC® DSC.
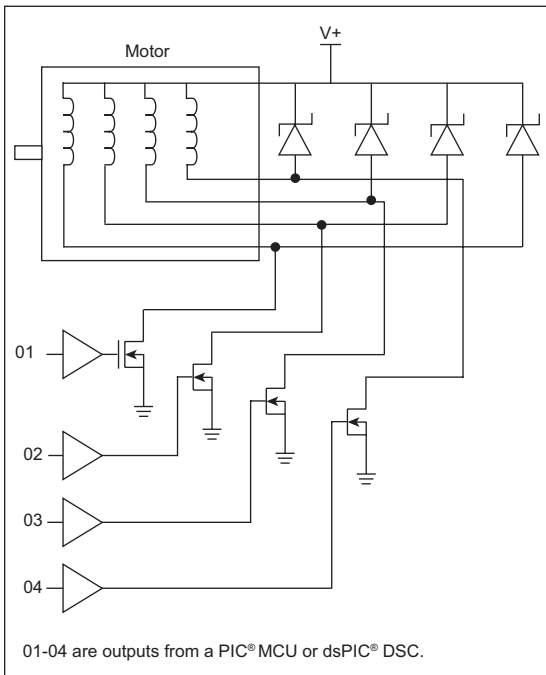
Figure 3-4: Bipolar Motor (4 Half-Bridges)

V   V

A   C

B   D

Motor

V   V

E   G

F   H
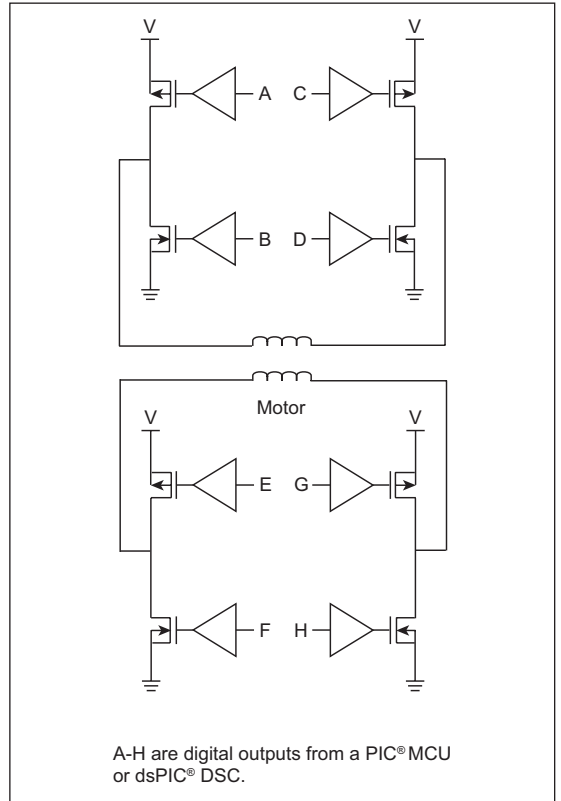
A-H are digital outputs from a PIC® MCU or dsPIC® DSC.

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 PIC devices and all dsPIC DSCs 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 PIC 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 PIC 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.

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 signals
                        ;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
 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 the CCP Registers With a Mid-Range PIC® 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     CCPR1L
```

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     CCPR1L
```

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     CCPR1L
```

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.

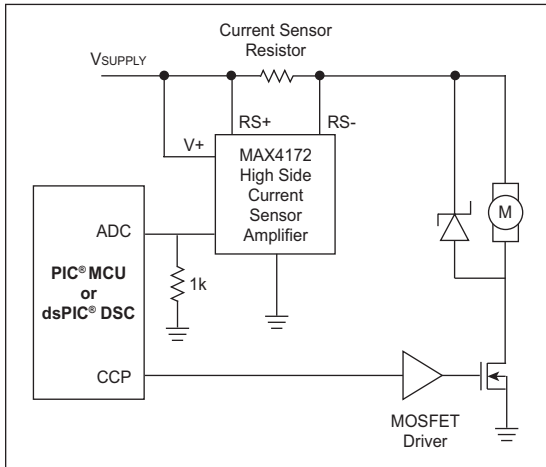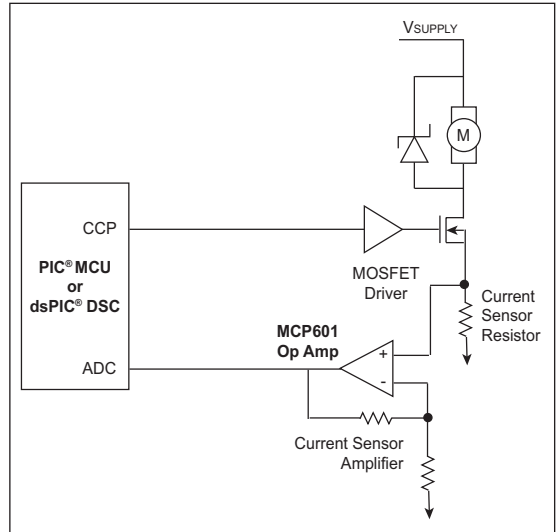### Figure 6-1: Resistive High Side Current Sensing

Figure 6-2: Resistive Low Side Current Sensing

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. This is a non-intrusive method that can be used to measure motor current.
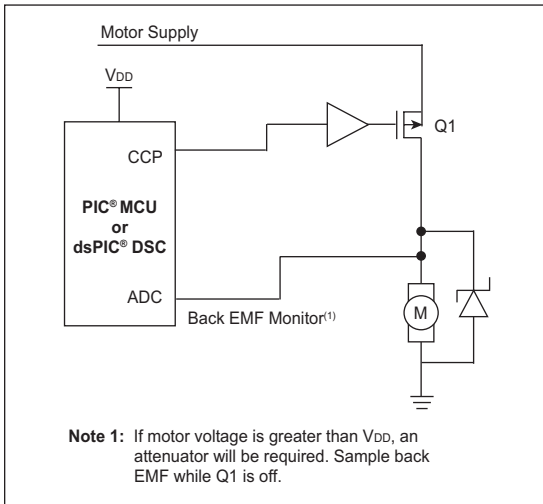
Figure 6-3: Magnetic Current Sensing
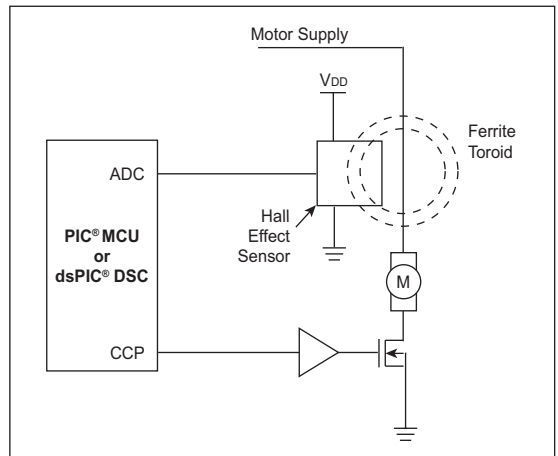
TIP #7 Position/Speed Sensing

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

Note 1: If motor voltage is greater than V_{DD}, an attenuator will be required. Sample back EMF while Q1 is off.
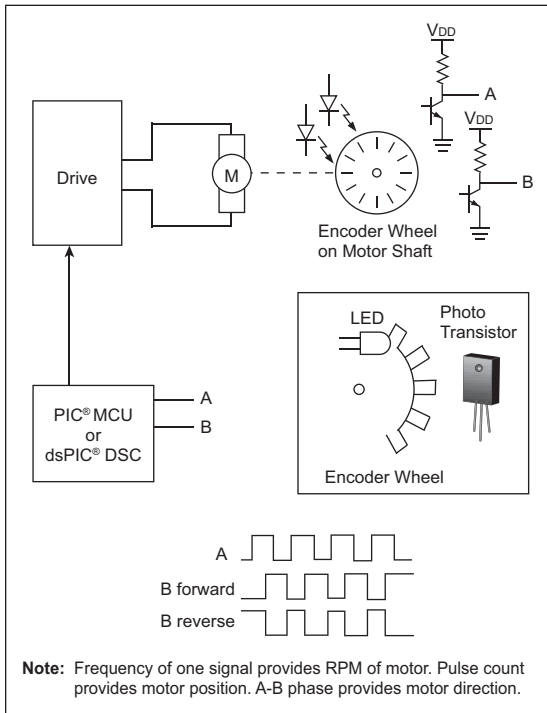
Rotary Encoder Sensing

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.

Sensing motor position requires a second LED and photo detector. The second sensor pair is mounted so the output pulses are 90° 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 rotor can be known.

An encoder with quadrature outputs can be used to track relative position from a known reference point. Another type of encoder uses a binary encoded disk so that the exact rotor position is always known. This type of encoder is called an absolute encoder.

Figure 7-2: Optical Speed/Direction/Position Sensing

Note: Frequency of one signal provides RPM of motor. Pulse count provides motor position. A-B phase provides motor direction.

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 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 Servomotor*" (DS00696)
- AN718, "*Brush-DC Servomotor Implementation using PIC17C756A*" (DS00718)
- AN822, "*Stepper Motor Microstepping 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)
- AN905, "*Brushed DC Motor Fundamentals*" (DS00905)
- AN906, "*Stepper Motor Control Using the PIC16F684*" (DS00906)
- AN907, "*Stepper Motor Fundamentals*" (DS00907)
- AN1017, "*Sinusoidal Control of PMSM Motors with dsPIC30F DSC*" (DS01017)
- GS001, "*Getting Started with BLDC Motors and dsPIC30F Devices*" (DS93001)

Application notes can be found on the Microchip web site at www.microchip.com.

Motor Control Development Tools

- PICDEM™ MC Development Board (DM183011)

  Used to evaluate the PIC18FXX31 8-bit microcontroller family.
- PICDEM™ MCLV Development Board (DM183021)
- 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 an can be downloaded at www.microchip.com

  Visit the Motor Control Design Center at: www.microchip.com/motor for additional design resources.

NOTES:

NOTES:

Sales Office Listing

AMERICAS

Atlanta
Tel: 678-957-9614

Boston
Tel: 774-760-0087

Chicago
Tel: 630-285-0071

Cleveland
Tel: 216-447-0464

Dallas
Tel: 972-818-7423

Detroit
Tel: 248-538-2250

Kokomo
Tel: 765-864-8360

Los Angeles
Tel: 949-462-9523

Santa Clara
Tel: 408-961-6444

Toronto
Mississauga, Ontario
Tel: 905-673-0699

EUROPE

Austria - Wels
Tel: 43-7242-2244-39

Denmark - Copenhagen
Tel: 45-4450-2828

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

Germany - Munich
Tel: 49-89-627-144-0

Italy - Milan
Tel: 39-0331-742611

Netherlands - Drunen
Tel: 31-416-690399

Spain - Madrid
Tel: 34-91-708-08-90

UK - Wokingham
Tel: 44-118-921-5869

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8528-2100

China - Chengdu
Tel: 86-28-8665-5511

China - Hong Kong SAR
Tel: 852-2401-1200

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-5407-5533

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8203-2660

China - Wuhan
Tel: 86-27-5980-5300

China - Xiamen
Tel: 86-592-2388138

China - Xian
Tel: 86-29-8833-7252

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-4182-8400

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-2566-1512

Japan - Yokohama
Tel: 81-45-471- 6166

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-572-9526

Taiwan - Kaohsiung
Tel: 886-7-536-4818

Taiwan - Taipei
Tel: 886-2-2500-6610

Thailand - Bangkok
Tel: 66-2-694-1351

1/30/07

MICROCHIP

www.microchip.com

Microchip Technology Inc. · 2355 W. Chandler Blvd. · Chandler, AZ 85224-6199

Microcontrollers · Digital Signal Controllers · Analog · Serial EEPROMs