AVR440: Sensorless Control of Two-Phase Brushless DC Motor

8-bit AVR® Microcontrollers

Application Note

Features

- Sensorless Control of Two-phase Motor typically used in Fans
- Adjustable speed with according to external speed reference
- PWM-based speed control with closed loop speed control
- Motor stall detection and automatic restart after short delay
- Configurable start-up speed ramp to match motor characteristics
- Low cost implementation with few and inexpensive components
- Firmware written in C
- In Circuit Debugging through debugWIRE interface
- Firmware modification and adaptation to specific motor easy
- Optional tacho output signal

1 Introduction

Two-phase brushless DC motors (BLDC) are widely used in fans for ventilating and cooling CPUs, graphics processors, power supplies and many other applications. The advantage of the BLDC motors is that they, compared to brushed DC motors are lighter, accelerate faster, produce little electrical and acoustic noise, and that they require no maintenance (no brush wearing). The requirements for cooling fans are continuously increasing as the use of powerful heat-producing electronics is increasing. As more fans are installed in more products, the need for low cost fan solutions is evident to keep the overall cost of end-products low. Further, as the number of fans in homes and offices are increased, the need to keep these fans as quiet and efficient as possible is also getting more pronounced. The characteristics of BLDC motors match the requirements of fans very well.

This application note describes how to implement the electronics and microcontroller firmware to control a two-phase BLDC motor using an 8-bit AVR microcontroller. The implementation is based on the small and low cost ATtiny13. The built-in ADC and PWM are used in a way so that Hall-sensors are not required (Patent Filed); it uses the EMF voltage over the passive coil to determine when to perform the commutation of the current flow in the stator coils. This solution to eliminate the Hall-sensor (position sensor) is a state-of-the-art low cost method for controlling two-phase BLDC motors.

Rev. 8007A-AVR-09/05

2 Theory of operation

To understand the implementation a brief introduction to the basic two-phase BLDC motor and the EMF is provided.

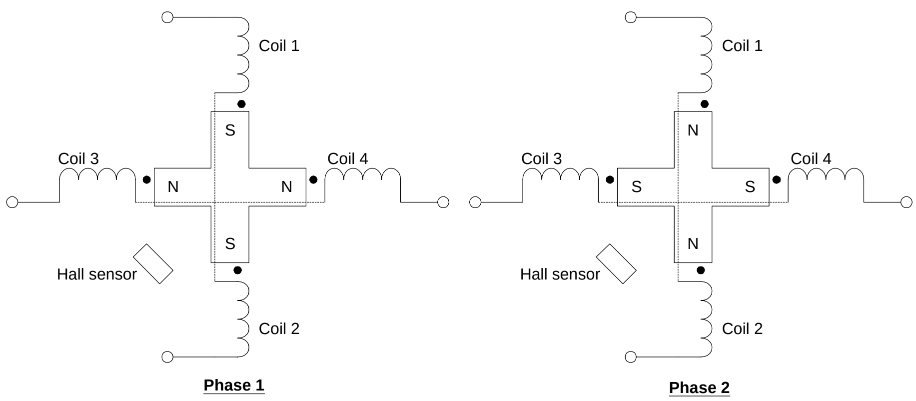
2.1 A Two-phase BLDC Motor

Brushless DC motors consist of a rotor with permanent magnets and a stator with a number of coils. The number of poles on the permanent magnets and the number of coils vary depending on the desired characteristics of the motor. A basic two-phase motor is illustrated in Figure 2-1. The figure shows a motor with 4 coils and 2 pole set. The upper and lower coils in the figure are electrically connected, as are the left and right coils.

The terminology of the motor commutation needs to be clear: The passive, or un-energized coil is the coil, which does not contribute to the magnetic field that turns the rotor (as no current flows through it). This is opposed to the active, or energized, coil in which a current flows causing a magnetic field to emerge. In normal operation, one coil should be active and the other passive at any given time. The orientation of the rotor magnets defines which coil that should be energized to drive the rotor in the desired direction. During operation, a change in orientation of the rotor must be responded to by alternating the current flow through the coils – referred to as “commutation” – so that the rotation is sustained by changing the magnetic field generated by the stator coils.

Energizing coils 1 and 2 in Figure 2-1 will push/pull the poles of the rotor magnets towards an alignment with the coils (Phase 1). Once rotating, the inertia of the rotor will ensure that the rotor is not only attracted to, but also passes by the active coils. As soon as the magnet passes by the active coils, the other coils must be activated (phase 2) and the rotation thereby is continued. If the commutation is made at the right moment, the magnetic fields of the coils are changed so that the produced torque is always in the same direction. Or rather, if the commutation is made too early or too late a torque in the opposite direction of the rotation is produced for a short while, slowing down the rotor speed. It is therefore necessary to have information about the orientation of the rotor magnets relative to the coils to get maximum performance.

Figure 2-1. The fundamental design of the two-phase BLDC motor.



The commutation of two-phase BLDC motors is usually controlled by the use of a Hall sensor to detect the orientation of the rotor. A hall sensor is a magnetic switch, which is used to determine the orientation of a magnetic field, in this case generated by the rotor magnets. The hall sensor sets or clears its output dependent on the polarity of the magnetic field generated by the rotor's magnets. If the Hall sensor is placed between the coils 2 and 3 (see Figure 2-1) the Hall sensor changes its output when the rotor magnets are aligned with the coils, as this is when the magnetic field of the two poles cancels out each other and the polarity of the magnetic changes. The Hall sensors allow the control system to know when to commutate the stator coils to maintain the rotation. The down side of using this rather simple trigger for commutation control is that the hall sensor is a cost adder, and that it needs to be positioned accurately to provide reliable information. Even when the Hall sensor is accurately positioned, position information is only provided at each commutation instant.

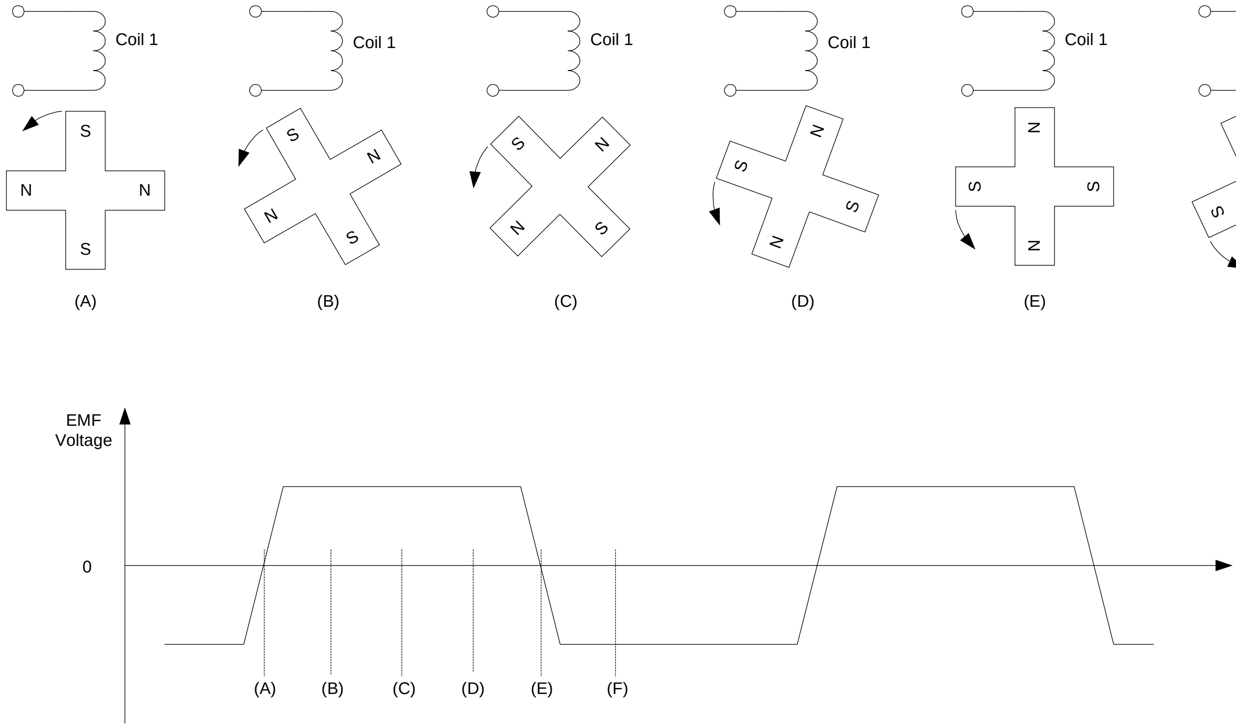
2.2 Electromotive Force

The Electromotive Force (EMF) is a voltage over an inductor (coil) generated by an alternating magnetic field (caused by the magnets on moving rotor). The EMF shape in a BLDC motor is approximately trapezoidal, as shown in Figure 2-2.

A BLDC motor is highly influenced by the EMF: EMF is generated over the stator coils when the rotor is turning. The amplitude of the EMF is directly proportional to the rate at which the magnetic field (seen by a coil) is changing – in other words the EMF amplitude is proportional to the speed of the motor. The EMF will, when the motor reaches a certain speed, be of the same amplitude as the voltage used to energize the coils. This limits the current flow through the coil and therefore the maximum speed of the motor since a current is required to generate a magnetic field and a magnetic field is required to produce torque. The EMF induced in an energized coil is called the Back EMF (B-EMF).

The EMF is not only generated in the energized coil, but also in the passive coil. The EMF of the passive coil can be measured and used to determine the position of the rotor. Figure 2-2 illustrates the trapezoidal EMF voltage over a passive coil relative to the position (angle) of the rotor magnet.

Figure 2-2. EMF generated by rotating rotor magnets.



The down side to using the EMF to sense determine the orientation of the rotor is that the EMF amplitude depends on the speed of the motor and there is a practical lower limit to how small voltage levels that can be measured – also considering the noise that will be present in a given system.

As seen in Figure 2-2 the EMF changes polarity when the rotor magnet is aligned with the coil, the same instant the output of a Hall sensor would change. This is utilized when using sensorless commutation methods to control a BLDC motor.

2.3 Sensorless control of Two-phase BLDC motor

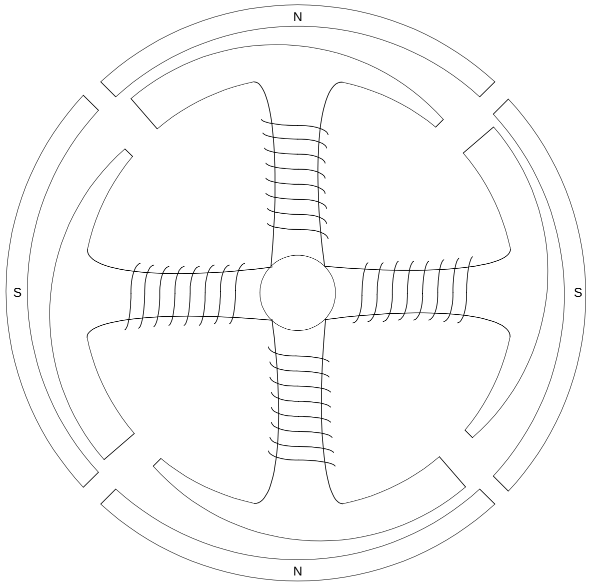
For three-phase BLDC motors sensorless commutation methods based on EMF feedback are commonly used. In these implementations the EMF is used to determine the orientation of the rotor and thus the timing for the commutation.

The moving rotor magnets of a two-phase motor do, like in the three-phase motor, generate an EMF over the passive stator coil. The concept of using the EMF over the passive stator coil to determine the orientation of the rotor has been utilized in the solution proposed in this application note. One advantage of using sensorless commutation is obvious, that the Hall-sensor can be eliminated, but more advantages will become clear below.

2.4 Direction of rotation

In many motor applications, it is desirable to operate the motor in one direction rather than another: if the motor is used to e.g. cool a PC power supply it should blow cold air from outside the cabinet into the power supply rather than sucking hot air from inside the cabinet through the supply. Further, most fan blades are optimized for rotation in one direction.

Figure 2-3. Tapered air-gap in two-phase motor.



The fundamental two-phase motor of Figure 2-1 does not guarantee the direction of rotation when starting the motor. The direction is determined by the starting position of the rotor and the coil activation order. If the starting position of the rotor is known, the coil activation order that will result in a particular direction of rotation can be determined.

In sensorless operation, no information about the starting position of the rotor is available. It is therefore necessary to bring the rotor to a known position before starting the motor.

When one set of coils are activated, the rotor magnets will align with an axis passing through the coils, such that rotor north poles face stator south poles or vice versa. Activating the coils in this way will force the rotor into a known orientation. However, the knowledge about the orientation of the rotor alone is not sufficient is not sufficient to ensure that the motor rotates in the desired direction: The physical shape of the ferro cores of the coils is used to produce a preferred rotation direction.

A tapered air gap, shown in Figure 2-3 is used in many two-phase BLDC motors to ensure correct direction of rotation. The air gap between the rotor magnet and stator is uneven. When the stator coils are un-energized, the rotor magnet will align in a position where the air gap between the magnet poles and the Ferro core is smallest (where the reluctance is smallest). The motor in Figure 2-3 has four such resting positions. These resting positions are located between the alignment positions when a coil ser is energized:. If the rotor is aligned with coils set (by activating the coil), and then released, it will rotate forward towards the closest resting position. The position of the rotor is hereby known and further, the rotation direction is controllable by selecting the appropriate order of the subsequent commutations.

2.5 Commutation waveforms

In order to understand how to drive a motor efficiently, it is useful to have an understanding of how torque is produced. The equation for electrical torque generated by one phase is shown in Equation 2-1.

Equation 2-1. Electrical torque generation

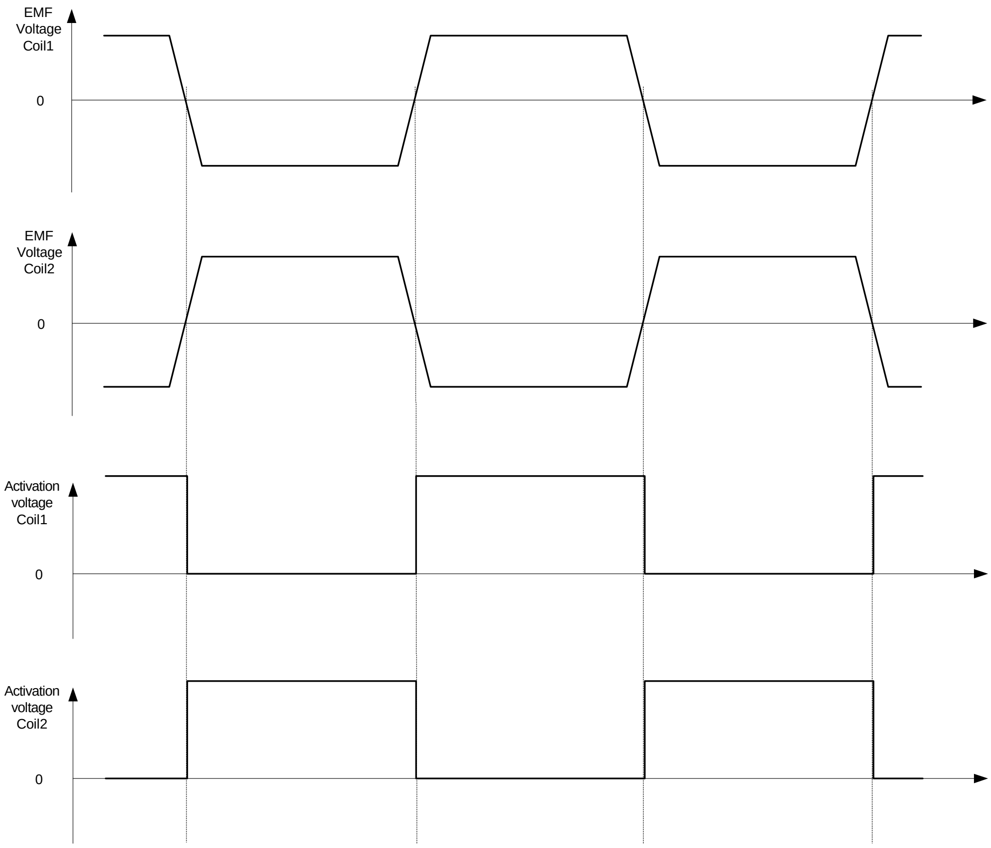
$$T_e = \frac{ei}{\omega}$$

T_e is the electrical torque, e is the EMF, i is the current through the motor coils and ω is the angular velocity of the rotor.

Equation 2-1. shows that the generated torque is in the same direction as the rotation as long as e and i has the same sign. Conversely, this means that if e and i has opposite signs; the electrical energy is wasted decelerating the motor.

The commutation voltage waveform used in most fans is a square wave (also referred to as block commutation). Figure 2-4 illustrates the block commutation, where the commutation is triggered directly by the zero-crossing (polarity change) of the EMF voltage to keep the sign of the phase current equal to the sign of the EMF.

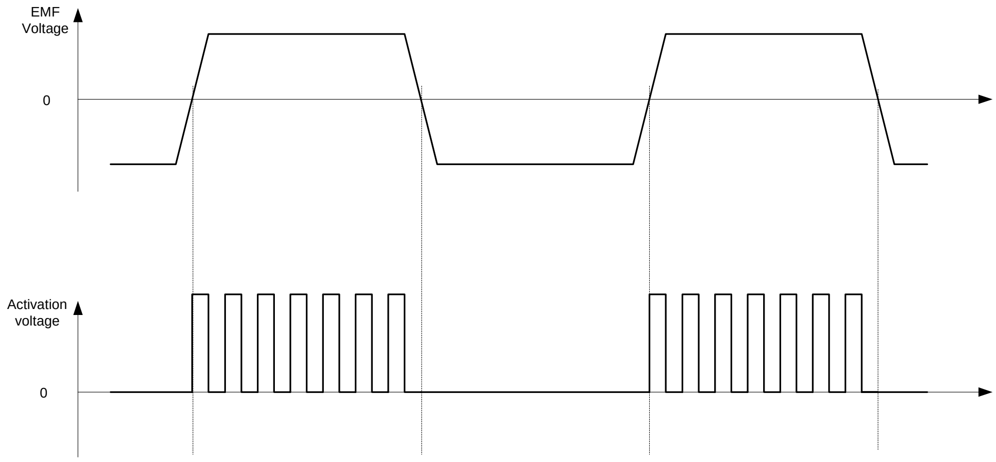
Figure 2-4. Fundamental square wave commutation (block commutation).



2.6 Speed control using Pulse Width Modulation

As stated previously it can be desirable to control the speed of a motor. In a fan application it can also be desirable to reduce acoustic noise and to reduce the power consumption of the motor. One way to control the speed of the motor is to control the operating voltage of the motor; however, very few systems e.g. PCs can provide an adjustable operating voltage for the motor (fan) directly, without a hardware power stage that is. The consequence of this is that many systems have a hardware that controls the supply voltage for the fan – which does not come for free. When considering that the fan control electronics include driving transistors, which can be used to control the voltage across the motor, the need for the system to adjust the supply voltage to the fan should be seen as redundant functionality. The alternative is to build the voltage/speed control into the control electronics of the motor. If a microcontroller is used to control the commutation the motor, Pulse Width Modulation (PWM) can be used to control the average activation voltage of the motor. By applying a PWM output to the motor coils, the average activation voltage and thus the current through the coils can be controlled (see Figure 2-5). The duty cycle of the PWM output defines the average activation voltage for the coil; in Figure 2-5 a 50% duty cycle is shown – the average activation voltage is thereby 50% of the operating voltage. Increasing the duty cycle of the PWM out increases the speed/torque of the fan.

Figure 2-5. Activation voltage modulated with PWM.



To be able to implement speed control using a PWM it is an advantage to have a hardware PWM in the microcontroller. This ensures correct and glitch-free timing and that it is possible to use the full voltage range (duty cycle) from 0% to 100% (as opposed to a software implemented PWM). The operation of a hardware PWM is also operating very much independent of the rest of the firmware, leaving more clock cycles to the important tasks of commutation control, safety features and closed-loop speed control.

One of the basic requirements when using PWM to control the motor speed is that the base frequency of the PWM is above the audible frequency range (20 Hz - 20 kHz). This means that it is in general desirable to have a PWM base frequency of more than 20 kHz. The mechanical response characteristics of the fan should however be considered as well: The acoustic noise from small fans will most often be un-hearable though the PWM base frequency is slightly below 20 kHz. Still, one should be aware that PWM frequencies below 20 kHz might, depending on the fan, be possible to hear as a high-pitched tone. Higher PWM frequencies than 20 kHz can be used, one should however be aware that power dissipation in the driving transistors is related to the switching frequency of the PWM.

2.7 Speed reference

The speed of a motor are in most applications adjusted according to an external reference. The reference can be e.g. an analog signal, like a temperature sensor or potentiometer, or a PWM signal generated by a system host controller. A microcontroller offers great flexibility to use the speed reference signal as input to control the speed of the motor, especially when an internal ADC is available. Any type of speed reference can then be measured, and by varying the PWM duty cycle that controls the activation voltage, the motor speed can be controlled accordingly. This makes it possible to use closed loop speed control, where the PWM is constantly changed to minimize the difference between the speed reference and the motor speed.

2.8 Failure conditions to consider

For all electrical motors the problem of the motor stalling (being locked) must be considered – if not, the motor can be permanently damaged or even take fire. Therefore any implementation of motor control must be able to respond to a stall situation. In some cases the stall is temporary, and it can thus be desired to restart the motor. A common approach is to stop the commutation of the motor when stalling is detected, wait for a while and then try to restart it. A microcontroller based motor control can handle this easily by monitoring the rotation speed.

A potential problem for an electrical motor and a driver stage is that it overheats. This occurs if the motor draws high currents, e.g. due to driving a large load or accelerates fast. If a temperature sensor is embedded in the motor or the driving stage, the ADC of a microcontroller can be used to monitor overheating problems. Most common is it to monitor the current flow through the driver and the coils, using a shunt resistor, as this is a good indicator of potential heat development and overload of the motor.

3 Proposed implementation overview

This section describes the implementation of the hardware and software (firmware) for the proposed motor control using an ATtiny13 microcontroller.

3.1 Hardware

Figure 3-1 shows the main hardware building blocks that need to be present to perform sensorless control of a two-phase BLDC with a microcontroller. The exact hardware requirements will vary between different motors, but the principle is the same.

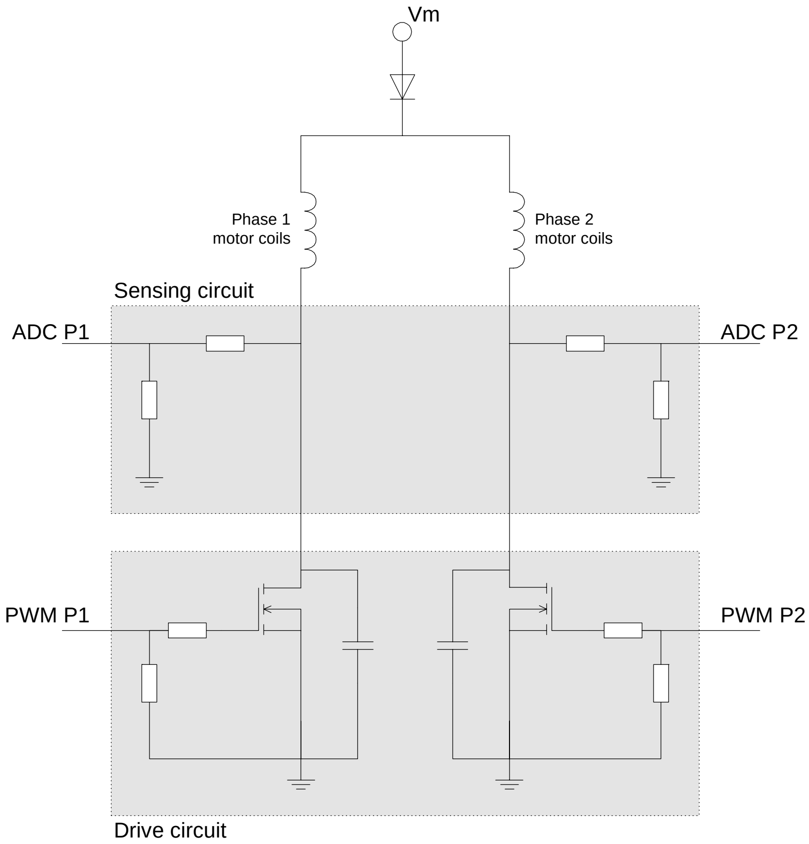
The block labeled “Drive circuit” is basically two on/off switches that, when turned on, completes the coil circuit, allowing current to flow. Transistors are used to switch the coils on and off. The capacitors across the transistors help reduce the transients experienced when opening a circuit with an inductive load and also reduce noise.

The block labeled “Sensing circuit” performs signal conditioning. In Figure 3-1, this circuit is a simple voltage divider, scaling voltages to the microcontroller’s ADC range. Note that the EMF voltages are affected by the resistor network: Without any voltage division, the measured voltage equals $V_m - V_d - V_c$, where V_d is the voltage drop over the diode and V_c is the total voltage drop over the coil. The voltage drop across the coil is equal to the EMF when the coil is not activated. The due to the location where the EMF is measured and due to the resistor network the (EMF) voltage measured is the negative EMF superimposed on a constant bias.

The diode between motor supply voltage and the motor, acts as polarity protection as well as ensuring that kick-back from the motor does not enter the supply network.

The terminals labeled “ADC P1/2” and “PWM P1/2” are the connections to the microcontroller.

Figure 3-1. Drivers and sensing circuitry for sensorless control of two-phase BLDC



3.2 Microcontroller

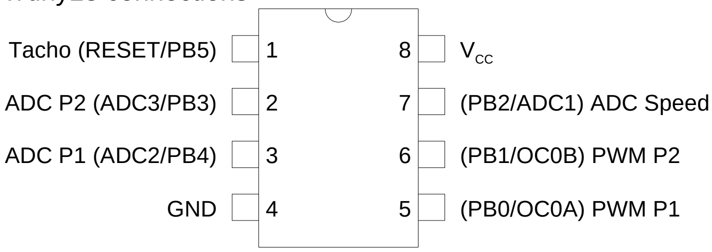
Sensorless control of a two-phase BLDC motor as described in this document requires two ADC channels for EMF sensing and two PWM outputs for commutation/speed control. In addition, one ADC channel is required if an external speed reference is needed. A tacho signal, if required, takes up an additional output. 6 IO pins are thus required.

The ATtiny13 is an 8-pin, microcontroller capable of running the sensorless control of a two-phase BLDC motor with external speed reference and tacho output signal. The built-in ADC, internal analog reference, two channel PWM, and calibrated internal 9.6MHz RC oscillator reduces the need for external components to a minimum.

The ATtiny13 pin utilization used in the included code example is shown in Figure 3-2. Note that the tacho signal resides on the reset pin. The Reset Disable (RSTDISBL) fuse must be programmed in order to use the reset pin as an I/O pin. Also note that a voltage higher than 10.5V on the reset pin is by the microcontroller perceived as a reset. If there is a possibility that the voltage on reset pin will exceed 10.5 volts it should be ensured, by hardware, to limit the signal level to Vcc level, e.g. by use of a zener or clamp diode. If desired, the tacho signal and external speed reference pins can be interchanged.

ADC inputs use the internal 1.1V volt reference for conversions. All ADC signals should therefore be in the range 0-1.1V.

Figure 3-2. ATtiny13 connections



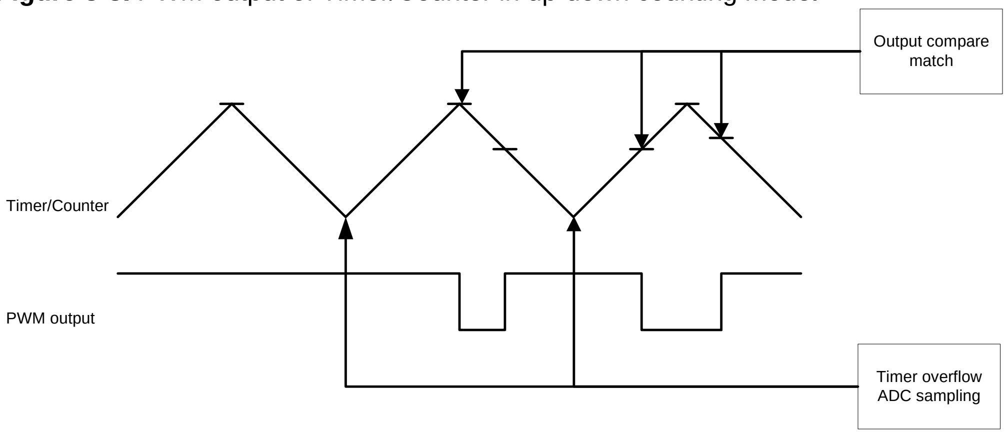
3.3 Firmware

The different parts of the firmware needed for sensorless control of two-phase motors are presented in this section.

3.3.1 Voltage control using PWM

To control the voltage over the active coil, and thereby the speed of the motor, two PWM channels are used. The two PWM channels are connected to the transistors of the driving stage. An up-down timer counter is used to realize the PWMs. The timer generates three events: two compare events that set or clear corresponding output pins (the PWM channels) and one overflow interrupt. The up-down counting and the PWM output can be seen from Figure 3-3.

Figure 3-3. PWM output of Timer/Counter in up-down counting mode.



The primary reason to use the counter in up-down mode is that the timer overflow interrupt can trigger an ADC conversion while no switching noise is contaminating the EMF signal

The timer overflow interrupt also triggers the execution of an interrupt service routine, in which the position of the rotor is evaluated and the commutation is changed – if the conditions are right (that is, if the EMF voltage level measured by the ADC is passing a given threshold).

The PWM base frequency is preferred close to 20 kHz to be outside the audible frequency range. Running the timer from the internal 9.6MHz RC oscillator and using it in up-down counting mode, the desired top value to obtain a base frequency of 20kHz is 240 (9.6MHz/20kHz/2). However, since it is not possible to change the top value of the timer in ATtiny13, while using both compare channels, the resulting base frequency is defined by the default top value of the timer, which is 255. The base frequency is therefore 18.85 kHz, which is in acceptable in most fan implementations. If the fan can cause acoustic noise at frequencies up to 20 kHz tuning of the internal RC oscillator is an option. By tuning the internal RC oscillator frequency upwards the

PWM based frequency can be increased. The frequency of the internal RC can be tuned by up to 10% about the nominal frequency without affecting the operation of the device. Refer to the AVR datasheets for more information about the tuning of the internal RC oscillator. Note that the CKDIV8 fuse, which divides the clock by 8 is programmed by default. This fuse must be cleared to achieve a clock frequency of 9.6MHz.

3.3.2 Use of ADC for measuring the EMF voltage

The sampling with the ADC in the ATtiny13 can as mentioned be triggered by e.g. a timer overflow event (refer to Figure 3-3). When using the timer in an up-down counting mode, the overflow event occurs when the timer reaches the bottom (zero), which is the moment in time where the PWM is not switching unless the duty cycle is very low. Most fans require between 10% and 40% duty cycle to operate and switching will therefore not occur while sampling. If timer overflow triggered sampling had not been available, significant analog filtering or computationally complex firmware would have been required to perform sensorless control of a two-phase BLDC motor as the switching noise would otherwise contaminate the sampling of the EMF.

The selection of the appropriate ADC channel is controlled by the same part of the code that evaluates the measured ADC value and controls the commutation of the motor. This is described in more details in the section related to commutation control.

The ADC is capable of measuring with a resolution up to 10-bit at 15ksps. At higher sampling rates the resolution starts to decrease. In the proposed implementation the processing of an EMF measurement as well as a reading of the external speed reference is done in the timer overflow service routine, which triggered the measurement. To ensure that the interrupt gets sufficient time to process the interrupt, the ADC clock is selected to be 1/8 of the system clock, which is 1.2MHz. Using this fast ADC clock degenerates the resolution of the ADC to approximately eight bits, but this is sufficient for detecting the EMF readings.

3.3.3 Prepositioning and startup

Prepositioning of the rotor is done by increasing the activation voltage over a coil set and wait a given period – until the rotor has settled in the desired position. The activation voltage is then decreased to let the rotor rotate to a resting position. This ensures that the position is known. The motor is then started in open loop, with inter-commutation delays according to a look-up table. This allows the control to be adapted to the motor that is controlled – to the mechanical response time of the motor.

The inter-commutation delays during startup can be calculated if sufficient information about the motor acceleration characteristics is available. Alternatively, the response can be determined by measuring the voltage across the coils with a scope.

3.3.4 Commutation

Once the starting sequence is completed, sensorless commutation is enabled by enabling the timer/counter overflow interrupt. A block commutation scheme is used. The internal ADC is used to measure the EMF. The measured EMF is compared to a threshold. If the EMF has passed the threshold a commutation is performed. The first few samples after commutation are discarded, to avoid false readings due to switching transients.

3.3.5 External speed reference

The external speed reference is implemented as an analog input that the microcontroller evaluates to determine the desired speed of the motor. The speed reference itself is read in the timer/counter interrupt routine, after the EMF measurement is completed. The speed closed loop control is handled by the foreground main loop. The implementation of the closed loop control is a simple step up-down regulation, but it can be modified to a PI control if desired. .

3.3.6 Fault handling

Fault handling is limited to stall detection. In a stall situation, commutations will not trigger. Since speed control is implemented the current speed (or rather the number of timer/counter overflows between commutations) is monitored. This information is used to reset the device in a stall situation. The watchdog timer is used for this purpose. The watchdog timer is reset on every commutation. If the motor has stalled, commutations will not occur and the watchdog timer is not reset. This results in the watchdog timer overflowing, leading to a reset of the microcontroller. It is possible to check the Watchdog Reset flag at startup to detect if the last reset was a watchdog reset. This can be used to keep track of the number of failures, e.g. by incrementing a counter in EEPROM.

3.3.7 Tacho output

Fans often have an open collector output to generate a tacho signal out to let a PC (or other applications) know the speed of the fan. This can be accomplished using the Reset pin on the ATtiny13, which can be configured as an IO pin. Doing this in software eliminates the need for any external components to produce this signal, reducing overall cost.

3.3.8 Debugging options

It is possible to enable the debugWIRE debugging feature to use in system debugging using the JTAGICE mkII. This only requires this one pin and offers a efficient debugging method at an “affordable” expense. The adaptation of the code example to a given two-phase motor is made much easier in this way. In the final application the Reset pin can, if desired, be used for other purposes, such as a tacho signal. Refer to the documentation of the JTAGICE mkII for more details about the debugging options when using debugWIRE.

4 Description of the included source code

The included source code implements sensorless control of two-phase BLDC motors as proposed in this document. All configuration parameters are easily configurable so that the source code can easily be adapted to any two-phase BLDC motor. The source code is documented in flowcharts in the sections below and as html. The html documentation is found together with the source files – the file readme.html is the root file of the html documentation.

4.1 Software flowcharts

Software flowcharts found below covers the following: The top-level flowchart for software implementation is seen in Figure 4-1. Figure 4-2 shows the flowchart for the prepositioning and startup routine. Figure 4-3 shows the flowchart for timer/counter0 interrupt service routine, where the sensorless commutation occurs.

Figure 4-1. Implementation overview.

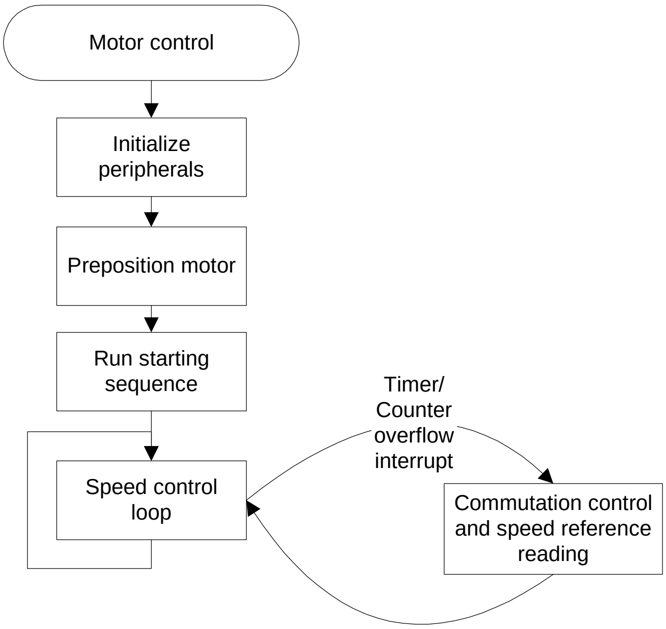


Figure 4-2. The prepositioning and startup routine

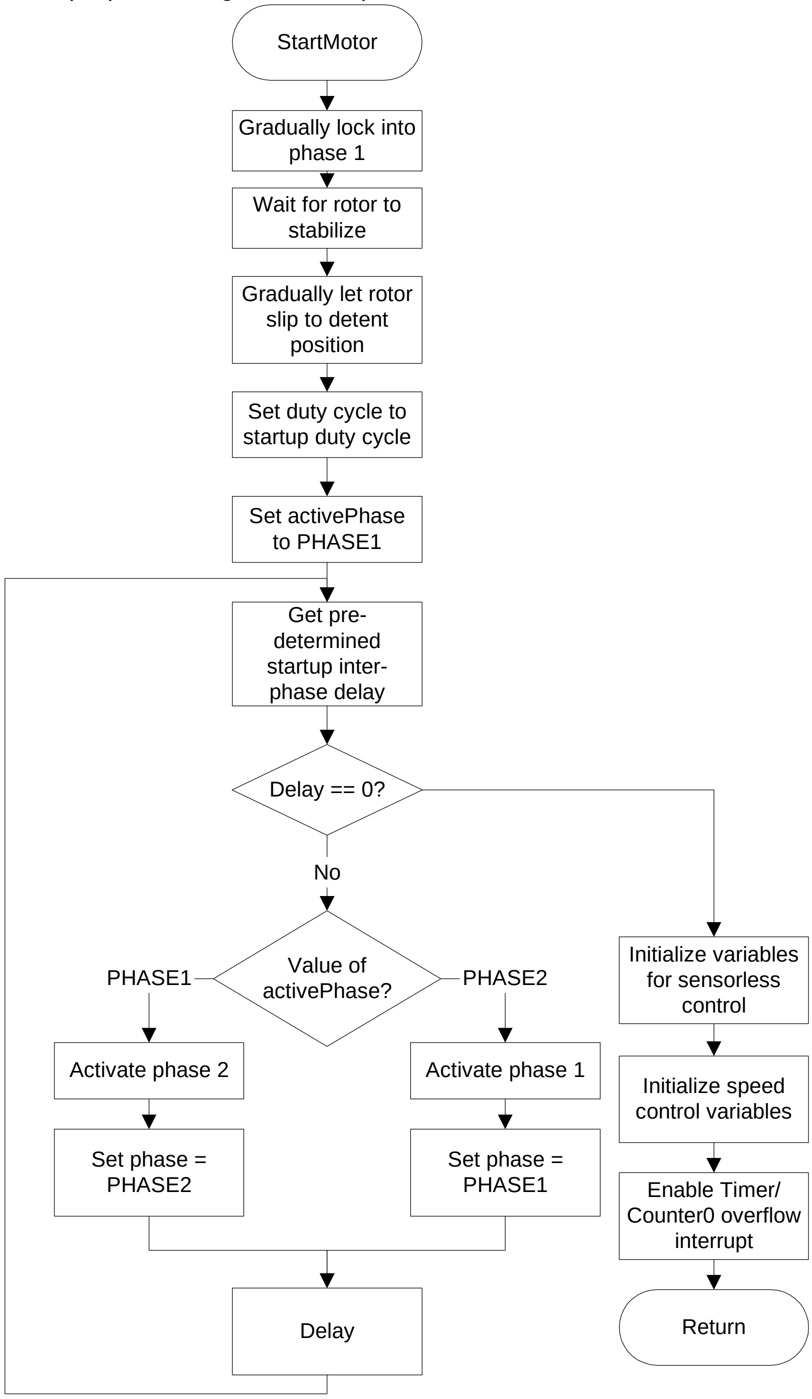
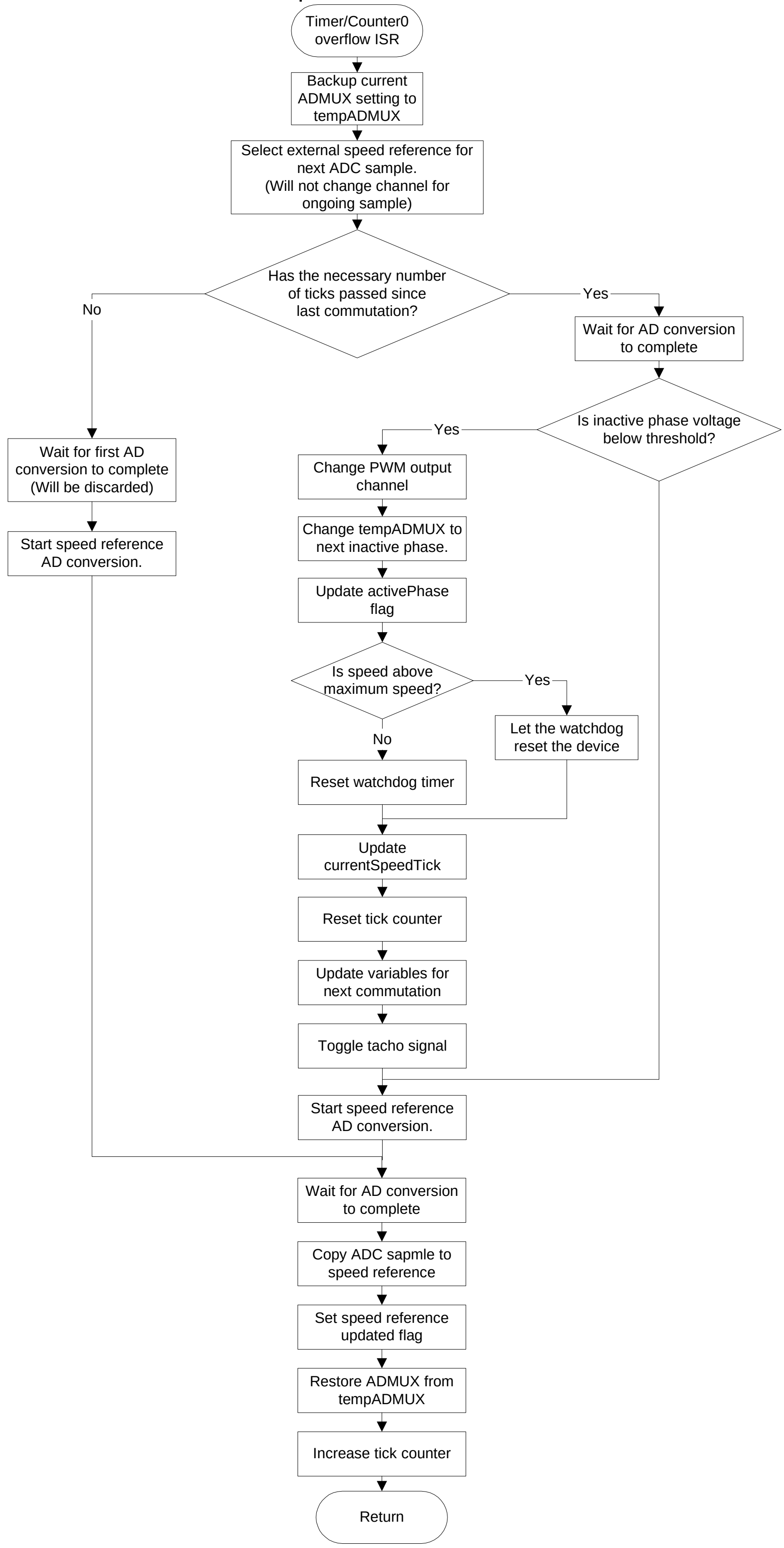


Figure 4-3. Timer/counter0 interrupt service routine.



Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

8007A-AVR-09/05