Sensorless BLDC Motor Control Based on CY8C24x33

AN47936

Authors: Dino Gu
Jemmey Huang
Associated Project: CY3253-BLDC
Associated Part Family: CY8C24x33
Software Version: PSoC Designer™ 4.4
Associated Application Notes: AN2227, AN42102

Application Note Abstract

This application note describes the implementation of Sensorless Brushless DC (BLDC) Motor Control using the PSoC device CY8C24x33. The speed close loop and over current protection in BLDC control application are also explained.

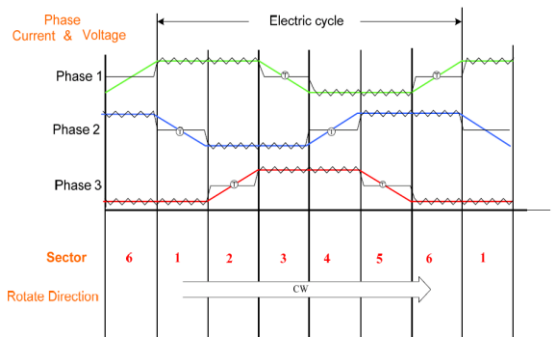
Introduction

Brushless DC Motor (BLDC) is constructed with a permanent magnet rotor and wire wound stator poles. Electrical energy is converted into mechanical energy by the magnetic attractive forces between the permanent magnet rotor and a rotating magnetic field that is induced in the wound stator poles.

The traditional brushed DC motor is being replaced by the BLDC motor that has higher reliability, efficiency, and lower noise. BLDC is popular in several fields such as consumer electronics, home appliances, and industrial controls.

Many BLDC motors have a three-phase winding topology with star connection. A motor with this topology is driven by energizing two phases simultaneously, while the other phase is kept afloat. The key to BLDC commutation is to sense the rotor position, then energize the phases that produce the maximum amount of torque. The rotor travels 60 electrical degrees at every commutation step. The appropriate stator current path is activated when the rotor is 120 degrees away from alignment with the corresponding stator magnetic field. It is then deactivated when the rotor is 60 degrees from alignment. Now, the next circuit is activated and the process repeats. This process is shown in Figure 1.

Figure 1. BLDC Motor Commutate Sequence



To implement this sequence, it is important to identify the rotor position. The simplest method is to use rotor position sensors such as Hall-effect sensors or an optical position encoder. However, sensors increase cost and cause reliability problems in motors operating in harsh environments. As a result, with the increasing power of embedded computing, the sensorless control method has become more popular.

Most BLDC sensorless controls are based on the Zero-crossing detection of Back Electromotive Force (BEMF). In Figure 1, every commutation sequence has one winding energized positive, the second negative, and the third winding is left open. The voltage polarity of Back EMF crosses from positive to negative, or from negative to positive (Zero-crossing), between two commutations. In ideal cases, the zero-crossing of BEMF occurs 30 electrical degrees after the last commutation, and 30 electrical degrees before the next commutation. By measuring the zero-crossing of BEMF and the 30 degrees time interval, the controller can perform the commutation without a position sensor.

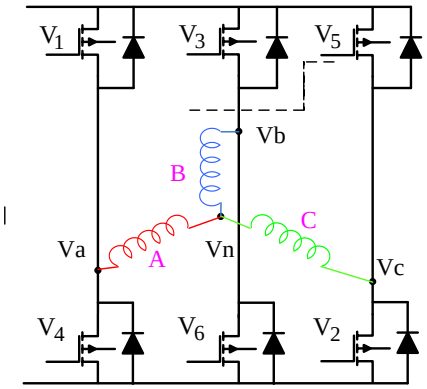
In this application note, a hybrid scheme is designed to achieve reliable sensorless rotor position detection. This is achieved by using BEMF waveform on the PSoC chip CY8C24x33. BLDC motor is in closed speed loop control and the experiment result is shown in the algorithm.

For more information of BLDC control based on Hall-effect sensors, refer to application note AN42102.

Back EMF Zero-Crossing Detection Technology

Figure 2 is a typical drive circuit of the three-phase BLDC motor. In this figure, Va, Vb, and Vc are terminal voltages, and Vn is neutral voltage.

Figure 2. BLDC Drive Circuit



Assuming that phase C is the nonfed phase, you can derive the following equations for the three terminal voltages.

$$V_a = RI_a + L\frac{dI_a}{dt} + E_a + V_n \quad \text{Equation 1}$$

$$V_b = RI_b + L\frac{dI_b}{dt} + E_b + V_n \quad \text{Equation 2}$$

$$V_c = E_c + V_n \quad \text{Equation 3}$$

In these equations, **E** is Back EMF, **I** is phase current, and **L** is inductance.

Because only two currents can flow in the stator windings simultaneously, the two phase currents are equal and opposite. This is shown in Equation 4.

$$I_a = -I_b \quad \text{Equation 4}$$

By adding the three terminal voltage equations (Equation 1, 2, and 3), the following equation is obtained.

$$V_a + V_b + V_c = E_a + E_b + E_c + 3V_n \quad \text{Equation 5}$$

It is evident that at the BEMF zero-crossing points the sum of the three BEMFs equals zero. As a result, Equation 5 can be reduced to:

$$V_a + V_b + V_c = 3V_n \quad \text{Equation 6}$$

For the nonfed phase (zero current flowing), the stator terminal voltage can be rewritten as:

$$E_c = V_c - V_n \quad \text{Equation 7}$$

From Equations 6 and 7:

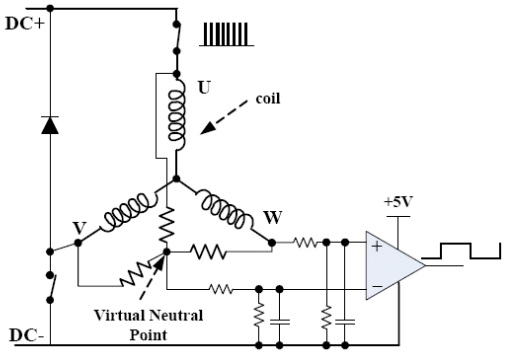
$$3E_c = 2V_c - (V_a + V_b) \quad \text{Equation 8}$$

Based on these equations, there are four existing BEMF zero-crossing detection methods. These methods are explained in the following sections.

1. Compared with Virtual Neutral Point

As shown in Equation 5, the zero-crossing of BEMF phase C can be detected by comparing Vc and Vn. However, Vn is usually not available in three-phase BLDC driven circuit. A resistor network is required to construct the virtual neutral point Vn, as shown in Figure 3. Voltage dividers and low pass filters are necessary to process the signals. However, the signal/noise ratio at low speed is less because, the BEMF amplitude is proportional to motor speed. The second problem is that the low pass filter causes phase delay at high speed.

Figure 3. Neutral Voltage Detection



2. Compared with 0V During PWM-OFF

Assume that PWM is only applied on the high side switch and the low side is ON during the entire step, ignoring the voltage drop on Vmos and Vd. The current flow during PWM-OFF is shown in Figure 4. In this period:

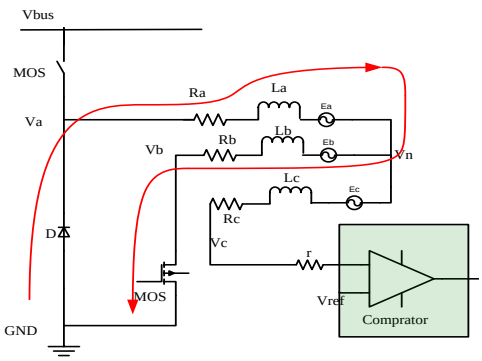
$$V_a = V_b = 0 \quad \text{Equation 9}$$

Combine Equations 8 and 9:

$$3E_c = 2V_c \quad \text{Equation 10}$$

From this equation, it is not necessary to detect the neutral voltage. The BEMF signal is directly proportional to the float phase voltage. As a result, to detect zero-crossing of BEMF Ec, just sample the terminal voltage Vc and compare it with a voltage a little more than 0V.

Figure 4. Detecting During PWM-Off



This method requires only a few external components, and the applicable speed range is large. This sensing technique can be used in a high voltage or low voltage system with no effort to scale the voltage. However, the low side switch must always be ON. In addition, a voltage reference that is a little higher than 0V for a comparator cannot be easily selected.

3. Compared with Half DC Bus During PWM-ON

Assume that PWM is only applied on the high side switch and the low side is ON during the entire step. During PWM-ON, ignore the voltage drop on Vmos and Vd. The following equation is obtained:

$$V_a = V_{bus} \quad \text{Equation 11}$$

$$V_b = 0 \quad \text{Equation 12}$$

As a result, from Equation 8 on page 2:

$$3E_c = 2V_c - V_{bus} \quad \text{Equation 13}$$

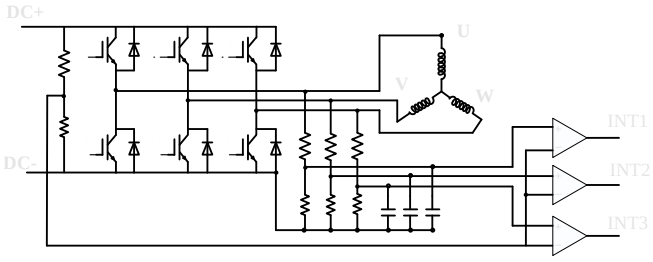
To judge the zero-crossing of BEMF Ec (Ec = 0) is exactly the same as judging the sign of (2Vc-Vbus). This means, you only need to sample the terminal voltage Vc and compare it with half Vbus.

This method requires only a few external components and has a high detection range for different speeds. The PWM duty cycle can be up to 100%. This sensing technique can be used in a high voltage or low voltage system with no effort to scale the voltage.

There are two typical solutions to implement this control algorithm. One solution is based on a high speed ADC with the capability of synchronous sampling aligning with the PWM. This means, the firmware samples the float terminal voltage and the DC bus voltage in every PWM-ON period. The comparison is then performed in the firmware to do the zero-crossing judgment.

The second solution is to use comparators to do the voltage comparison with half DC-bus voltage. The judgment is then done in software. In this case, the comparators' output always contains the PWM carrier waveform. As result, it is necessary to have a complex firmware to achieve that, or use a special PWM scheme or a RC filter to simplify the firmware.

Figure 5. Detection During PWM-ON



These methods require a high MIPS controller to handle the detection algorithm.

4. Simultaneous Sampling of 3 Phase Terminal Voltages by ADC

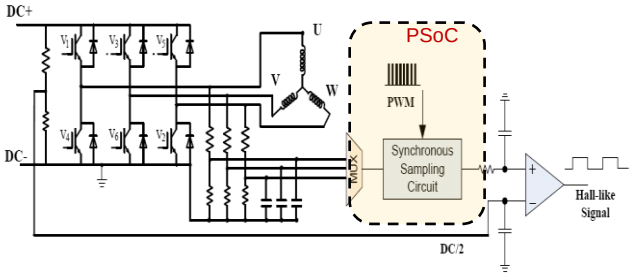
According to Equation 8 on page 2, this scheme to detect the zero crossing of BEMF Ec (Ec=0) simultaneously samples the three-phase voltages Va, Vb, and Vc. The float terminal voltage Vc is then compared with half of the fed phase voltage (Va+Vb) by firmware.

The algorithm is simple and does not require a filter and comparator. It does not rely on PWM mode and can sample voltage signals at any moment of the PWM period. However, it requires a high speed and multi-channel ADC. A high MIPS controller is also required.

Implementing BEMF Zero-Crossing Detection in PSoC

In this application, the BEMF Zero-crossing detection scheme is based on the Compared with Half DC Bus During PWM-ON method. Nevertheless, because of the versatility of the PSoC chip, this BEMF Zero-crossing detection scheme does not require a high speed ADC or a high MIPS controller. The internal multiplexer and other analog resources also reduce external components. Figure 6 shows the block diagram of PSoC implementation.

Figure 6. PSoC Implementation



A brief introduction of circuit implementation in PSoC follows.

- The multiplexer can switch between 3-channel BEMF signals in the sampling circuit according to the commutation status.
- The synchronous sample circuit, which consists of SC blocks and digital blocks, can sample the BEMF signal to RC filter during PWM-ON period. During PMW-OFF, the sampling circuit turns off the sampling, and the SC block keeps the same output voltage as the PWM-ON period. The input voltage on the comparator is also the same as PWM-ON period.
- A comparator compares the analog signal after the synchronous sampling circuit with the DC bus scales down the voltage. Then the signal is output without PWM carrier waveform to indicate the commutation.

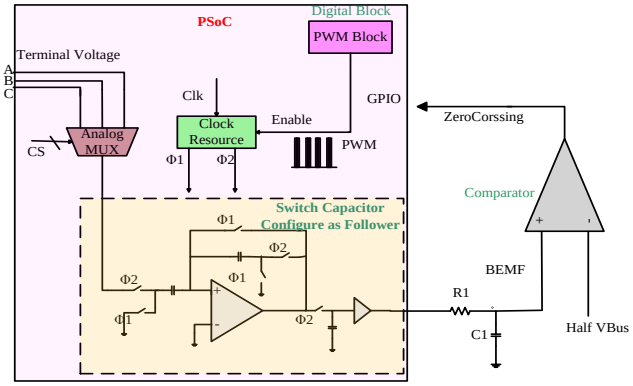
With the sample and hold circuit, the input of the comparator keeps the same at PWM-ON and PWM-OFF period. This removes the PWM carrier waveform on the comparator output and simplifies the firmware implementation.

The PSoC CY8C24x33 chip contains the following:

- One AMUX8 user module multiplexes 3-channel terminal voltage signals into the sampling circuit.
- One PWM8 user module works as a clock resource. The input clock frequency is much higher than the PWM frequency and is enabled by the PWM signal. This means, during the PWM-ON period, the clock resource provides the clock to the SC module and enables the SC block for sample and hold.
- One SC (Switch Capacitor) user module works as a 1:1 amplifier. It is driven by high frequency clocks Φ1 and Φ2 from the clock resource mentioned earlier. From the working principle of the SC block, it is evident that at clock Φ1 period, the SC samples the signal. At Φ2, the signal is output.

Figure 7 shows a detailed diagram of PSoC internal configuration.

Figure 7. PSoC Internal Configuration Diagram



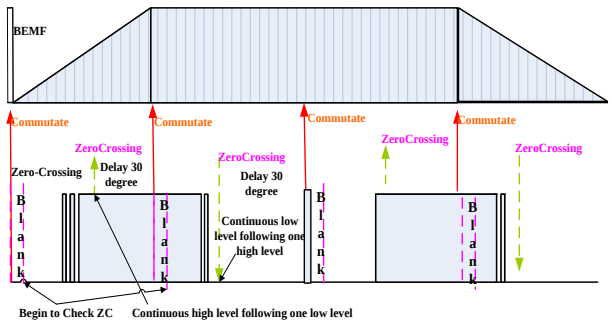
These user modules construct a synchronous sampling circuit. During PWM-ON, the clock source that provides to the SC block is enabled, and the BEMF signal can charge to C1. During PWM-OFF, the voltage on C1 remains the same.

An external RC circuit filters the glitches of BEMF. Because of the resource limitation on CY8C24x33, the BEMF signal is compared with the half bus voltage by an external comparator. The output of the comparator is the BEMF Zero-crossing signal, which is routed back to the PSoC chip through a GPIO pin. Firmware probes the voltage level to verify the BEMF Zero-crossing event.

Firmware Verification of BEMF Zero-crossing Event and Commutation

Figure 8 illustrates BEMF zero-crossing and commutation procedure in firmware. In this figure, the upper picture is one phase terminal voltage signal, and the second picture is the algorithm implementation for the three-phase BEMF zero-crossing signal.

Figure 8. Firmware Process Sequence



- When commutation begins, there is an incorrect pulse impact on BEMF zero-crossing signal because of the flywheel of the diode. To cancel this impact, F/W discards the first few PWM period samples of BEMF after commutation. This period is called “Blank” period.
- After “Blank” period, F/W probes the zero-crossing signal in every PWM period. However, near the zero-crossing point of BEMF, the output of comparator usually has several flip-flops. This is because of the voltage ripple. To validate the zero-crossing event, F/W judges a continuous high/low level, following a low/high level as the valid zero-crossing event.
- Zero-crossing validation causes some delay during the real zero-crossing moment. F/W compensates for this delay based on computing and experiments. The compensating value is different when running in different speeds.
- In ideal cases, according to Figure 1 on page 1, the zero-crossing of BEMF occurs at 30 electrical degrees after the previous commutation and 30 electrical degrees before the next commutation. As result, when a zero-crossing is checked, a 30 electrical degrees delay time must be recorded to evoke the next commutation. A Timer16 user module does this work. This Timer16 user module is also used as a free-running timer to stamp the commutation moment and zero-crossing moment for period calculation.

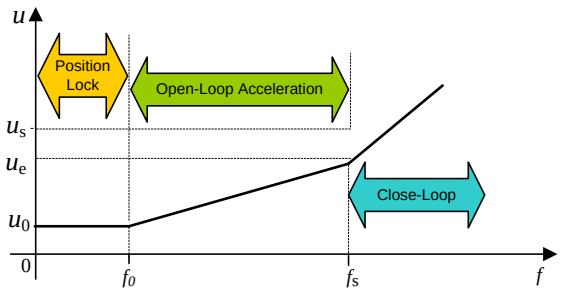
Free-Run Stages of BLDC Motor

The BEMF amplitude of BLDC is proportional to the motor speed. If the motor is still, there is no BEMF at all. When the motor runs at low speed, the BEMF is also too weak to be detected. As a result, before the BEMF can be measured by the firmware, there is a stage called the free-run stage. This stage drives BLDC motor step by step to acquire the initial BEFM zero-crossing signal before running sensorless control.

When run at a constant speed with open-loop stepping stimulation, the rotor position is approximately 90 electrical degrees ahead of that when run correctly under sensorless control. As a result, BEMF zero crossings cannot be sensed. To observe the zero-crossings, it is necessary to accelerate the motor at a certain rate.

Therefore, in this stage, the PWM duty cycle increases gradually, and the commutating period is longer than the usual case. A table *baStartUpTimeTbl[ ]* in F/W defines the timeout of every commutating period. The value of the table content results from experiments based on different motor types. When sufficient valid zero-crossing events are detected, F/W enters into normal synchronous running stage. If F/W cannot detect sufficient valid zero-crossing events during a period of time, an error occurs and BLDC must stop and wait for the next round of the free-running stage.

Figure 9. BLDC Starting Stages



PI Speed Close Loop

The PI control algorithm is useful in a continuous control system. There are two basic PI control algorithms: absolute mode and increment mode PI control algorithm. Equation 14 is a discrete expression of the position mode of the PI algorithm.

$$u_k = K_P * e_k + K_I * \sum_{i=1}^{k-1} e_i + u_0 \qquad \text{Equation 14}$$

In Equation 14:

e_k is the speed error.

K_I is the integration coefficient.

K_p is the proportional coefficient.

The disadvantages of the absolute mode PI algorithm are:

- Switching between closed loop and open loop has system impulsive force, which results in the unstable running of motor.
- Output of integration action is related to all past status. This increases the workload of MCU, and also accumulates systematic error.

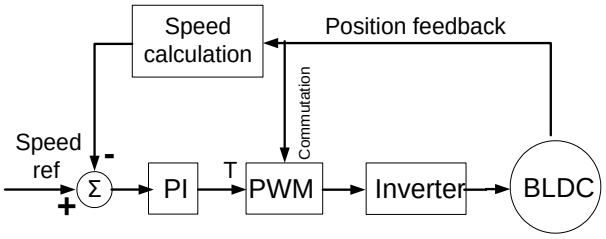
These disadvantages can be solved by using the increment mode PI algorithm, shown in Equation 15.

$$\Delta u_k = u_k - u_{k-1} = K_P * (e_k - e_{k-1}) + K_I e_k \qquad \text{Equation 15}$$

The control increment is the output, which is added to the current control input. The implementation on an 8-bit controller becomes easier.

Figure 10 shows the block diagram of PSoC PI closed loop speed control.

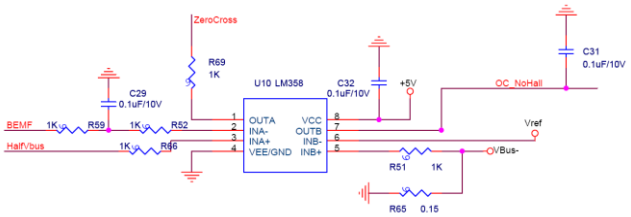
Figure 10. Diagram of Speed PI Close Loop



Over Current Protection

In this application, over current protection is done by H/W and F/W. As shown in Figure 11, R44 is the current shunt resistor. The bus current signal is compared with Vref through an external OPAMP, which is also used as a comparator for BEMF zero-crossing detection. OC_NoHall signal is the output signal.

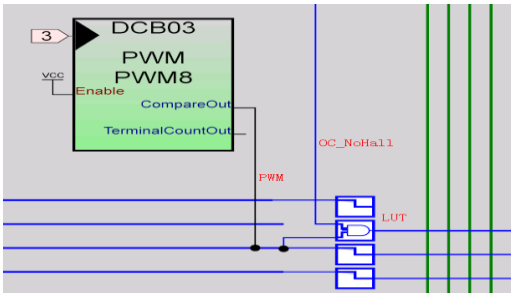
Figure 11. Current Sample Circuit



In normal condition, the output signal (*OC_Nohall*) of the comparator is low level voltage. However, when the bus current is more than the reference voltage, the output of the comparator is overturned into high level voltage.

OC_Nohall signal is routed to the GlobleInput type pin of PSoC. This signal is inverted and routed into the Look-up Table (LUT) unit inside the PSoC. In LUT, PWM signal and the inverted *OC_Nohall* are the two inputs of an AND gate. As a result, when *OC_Nohall* is high, PWM output is switched off. The motor current is limited. Figure 12 shows PSoC device configuration for current limitation.

Figure 12. Configuration for Over Current Protection



Meanwhile, the firmware also probes the level of the OC-Nohall IO pin. If a rising edge on this pin is detected, F/W prompts an interrupt and counts once. F/W judges the over current when it counts enough rising edges. As a result, all output PWMs are turned OFF and the motor is stopped.

Experiment Result

The proposed BLDC sensorless control technique is applied in the CY3253-BLDC kit. The DC voltage is 24V. BLDC speed is ranged from 500 rpm to 2250 rpm.

Figure 13 and Figure 14 show some key waveforms of the system.

In Figure 13, the yellow waveform is the one phase terminal voltage. The BEMF signal is combined with PWM noise. By sampling the float phase BEMF during PWM-ON, the BEMF signal can be derived from PWM noise. The green waveform is the multiplexed BEMF of three floating phases. Due to freewheel of diode, there is a glitch at the beginning of every commutation event. The special treatment in F/W can avoid this influence.

Figure 13. Terminal Voltage vs. Filtered BEMF

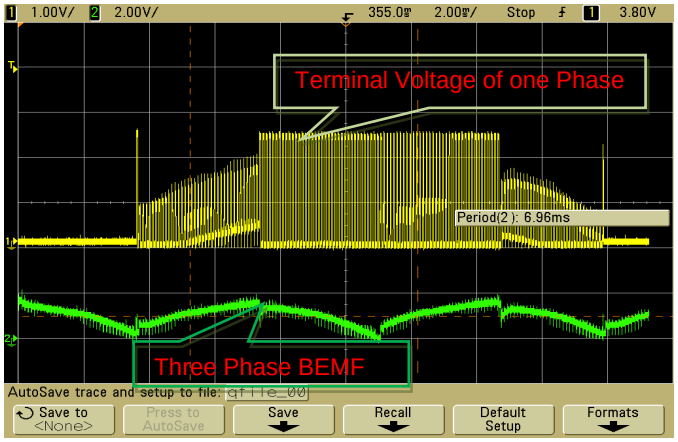
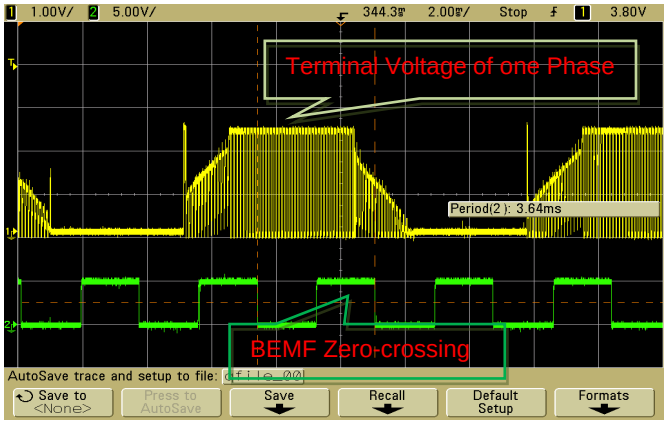


Figure 14 shows the terminal voltage and BEMF zero-crossing signal. The yellow waveform is the one-phase terminal voltage. The green waveform is the three-phase BEMF zero-crossing signal. From the figure, it is evident that the zero-crossing is evenly distributed. It is very similar to the input signal of the Hall-effect sensor.

Figure 14. Terminal Voltage vs. Zero-crossing



Conclusion

This application note describes BLDC motor sensorless speed control based on the PSoC chip. With the assistance of the PSoC device, the BLDC motor driver detects BEMF zero-crossing with a few external components, and completes speed closed loop control. Figure 16 on page 8 showcases the feasibility.

References

1. BLDC Closed Loop Control Based On CY8C24x33, Bill Jiang and Jemmey Huang. AN42102, Cypress
2. Brushless DC Motor Control, Andrey Magarita. AN2227, Cypress

Appendix

Figure 15. Schematic Drawing of BLDC Sensorless Control Board

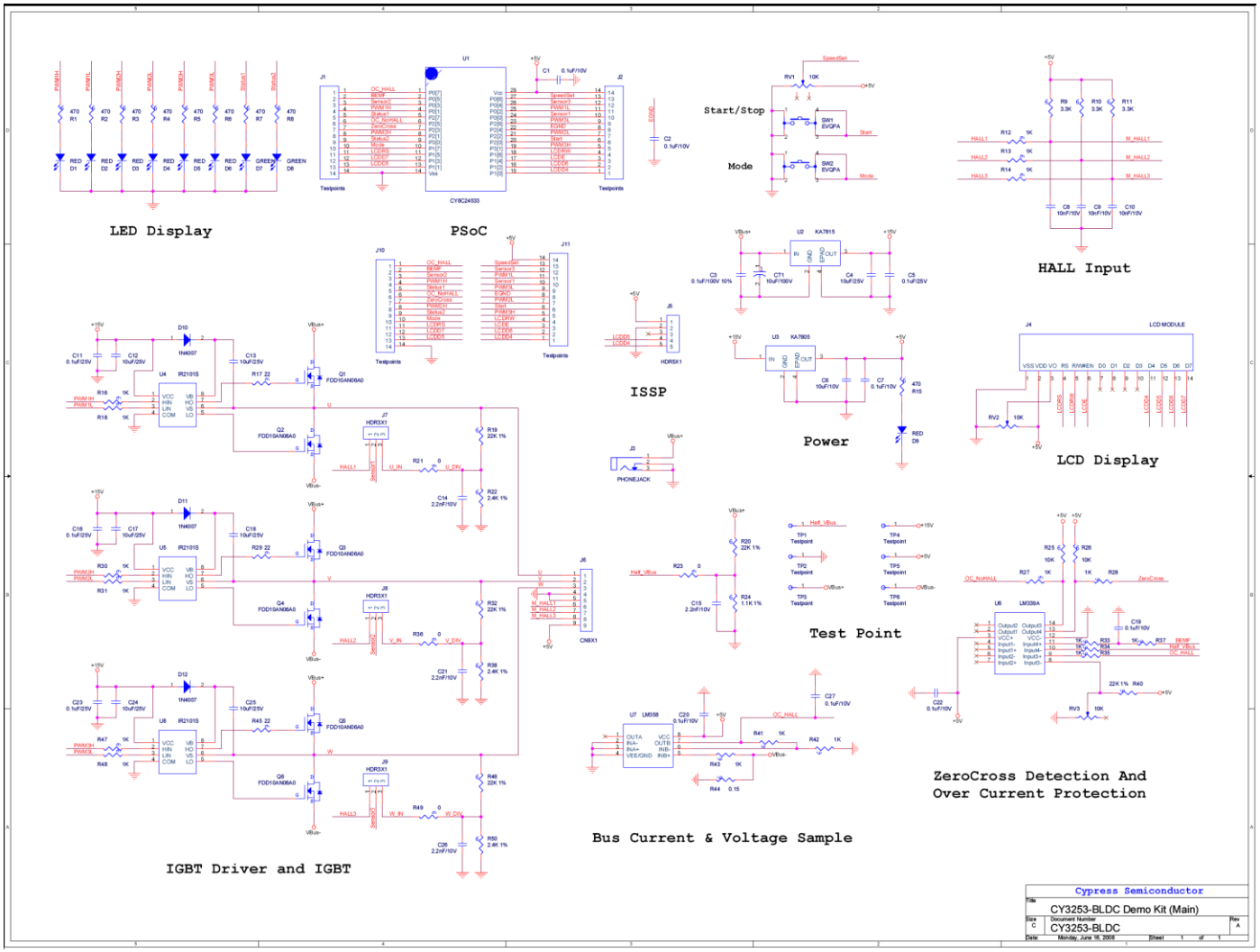


Figure 16. Photograph of BLDC Sensorless Control Board

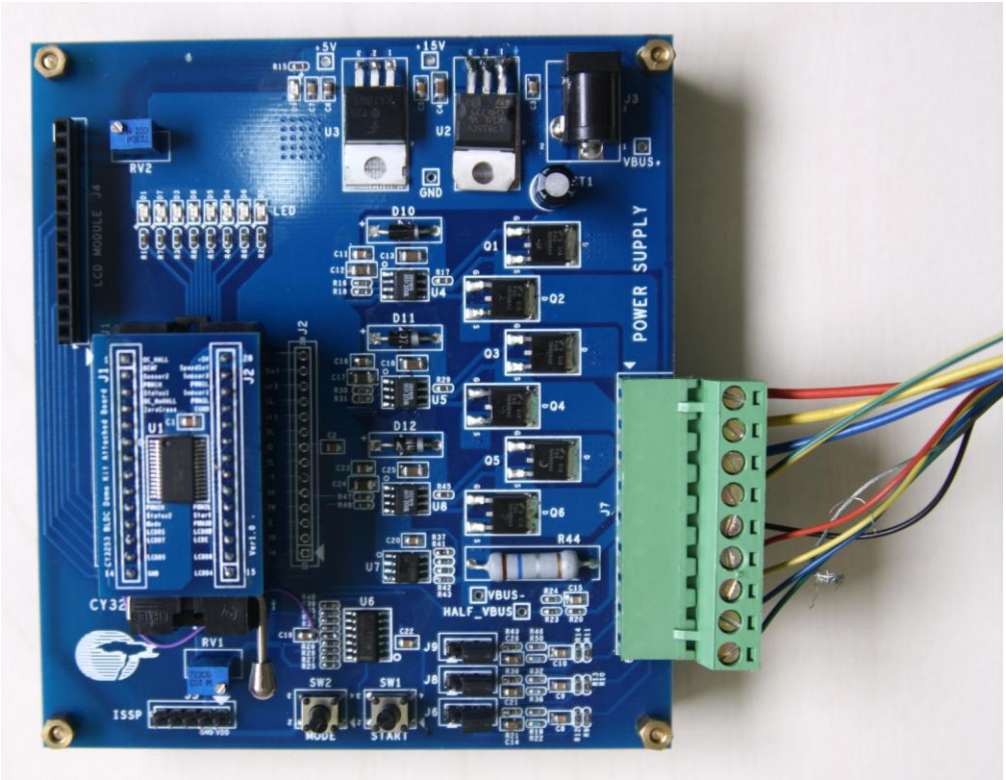
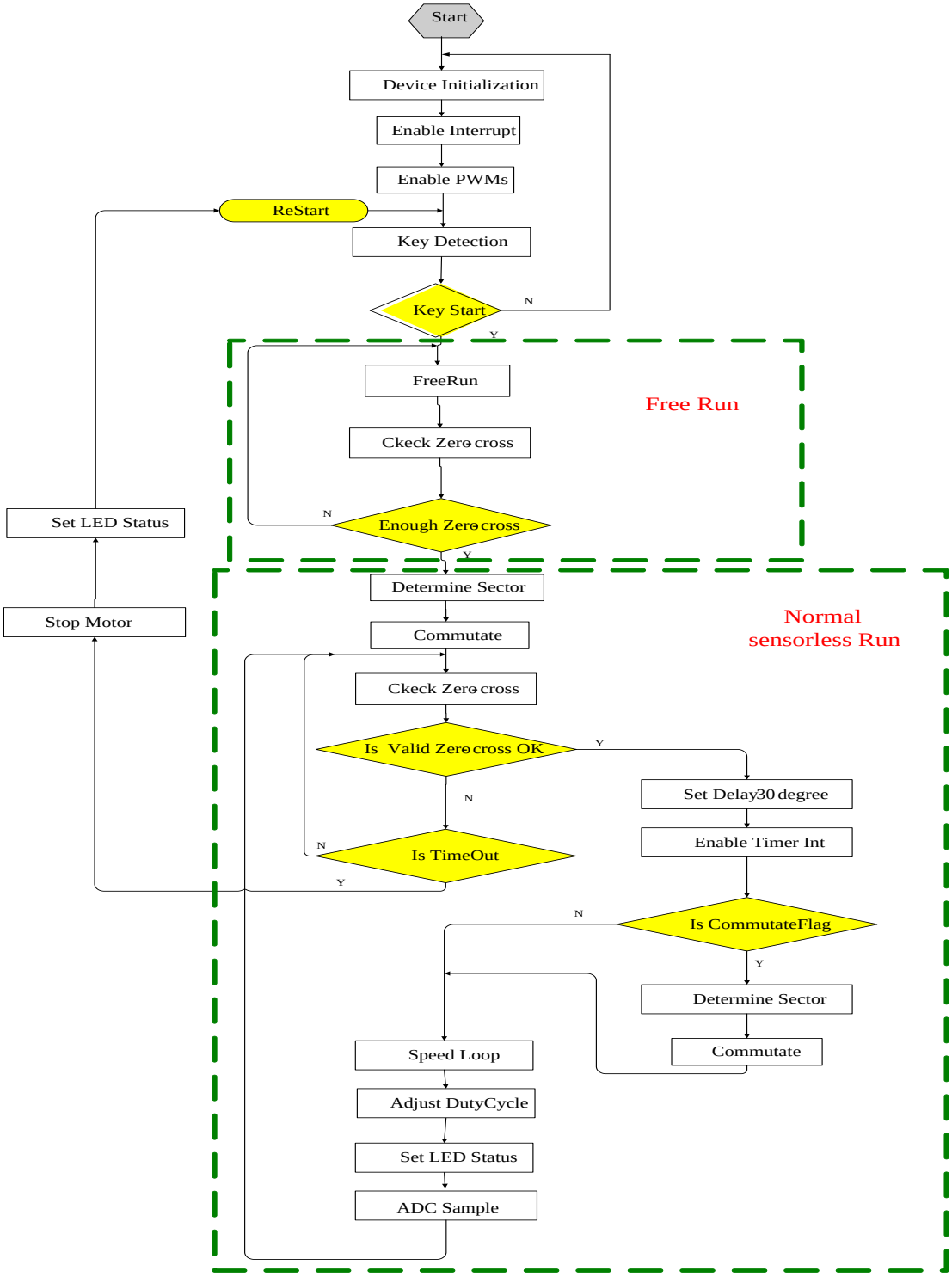


Figure 17. Firmware Flowchart of BLDC Sensorless Control Board



About the Authors

Name:	Dino Gu	Jemmey Huang
Title:	Application Engineer Sr.	Product Apps Manager Sr.
Contact:	qdgu@cypress.com	jhu@cypress.com

Document History

Document Title: Sensorless BLDC Motor Control Based on CY8C24x33

Document Number: 001-47936

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	2549704	QDGU/JHU/AESA	08/08/2008	New application note.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
http://www.cypress.com/