



**White Paper**

# **The Need for Autonomous Peripheral Interoperation in Sensorless BLDC Applications**

**Dave Coulson, Senior Field Application Engineer**

WP002003-0111

## The Need for Autonomous Peripheral Interoperation in Sensorless BLDC Applications

In this white paper, we will discuss optimized control schemes for driving sensorless BLDC motors and demonstrate the importance and advantages of microcontroller peripheral interoperation in these applications. We will also introduce you to some of the latest microcontrollers available for sensorless BLDC motor control applications.

The use of brushless DC (BLDC) motors has steadily increased over the last several years as requirements for improved efficiency and better reliability have risen. Variable-speed motor applications in industries such as White Goods, Automotive, Cooling, Aerospace, Medical and Industrial Automation are now using the BLDC motor in favor of other types of motors, such as Brushed DC and AC Induction. Because the drive electronics for BLDC motors are more complex, the semiconductor industry has responded with more cost-efficient solutions for these applications.

Several control methods exist for BLDC motors; the decision as to which method to use is defined by the requirements of the application. In sensor-based BLDC control applications, Hall Effect elements are integrated into the motor and are used to detect the position of the rotor for drive synchronization. Typically, three Hall sensors are used; each sensor provides a High signal for 180° of the electrical rotation and a Low signal for the other 180°. The three sensors are placed 120° apart, thereby dividing the rotation into six phases (i.e., six-step commutation). The microcontroller reads this information from these three sensors to determine the position of the rotor. In this scheme, the microcontroller always knows the rotor position - even when the motor is stopped - and can easily determine the correct commutation sequence required for the control loop. However, this scheme comes with the cost penalty of the Hall-Effect sensors - not to mention the additional assembly and wiring steps required to manufacture these motors.

In contrast, a more cost-effective sensorless control scheme employs detection of back EMF signals, which are generated (induced) by the unenergized phase windings in the BLDC motor; this signal information is used to synchronize the timing of the control loop. In effect, the Hall sensors can be eliminated, but the inference is that the rotor must first be moving (to generate the back EMF) before any positional information can become available. Although this particular issue certainly limits the types of applications in which a sensorless control scheme can be used, there are still many valid end products that can take advantage of its lower cost. Fans and blowers, for example, are excellent candidates for this type of control scheme.

There are many microcontroller-based solutions available on the market today, ranging from simple low-cost 8-bit devices to higher-performance 16- and 32-bit devices that all feature the minimum peripherals required to drive the sensorless BLDC motor: a 3-phase pulse-width modulator (PWM) to control the motor phases, an ADC to detect back EMF, and a comparator for over-current protection.

However, inasmuch as the peripheral set integrated on the MCU is key to targeting it to an application, the interoperation of these peripherals can greatly affect the performance of the MCU in the application. Terms such as MIPS become even more meaningless as autonomous peripherals are able to provide the services necessary for the application without CPU intervention.

As a case in point, Zilog has recently announced the Z16FMC family of 16-bit microcontrollers that, in addition to 4 channels of Linked List DMA, provides automated interoperation between the ADC and Timer and between the Comparator and PWM outputs.

Sensorless control of BLDC motors requires a microcontroller with fast interrupt response to handle PWM updates in real time. For applications that require additional functions such as high-speed serial communications, PWM demodulation, complex user interfaces and display control, the ability for the core motor control peripherals to operate autonomously becomes vital.

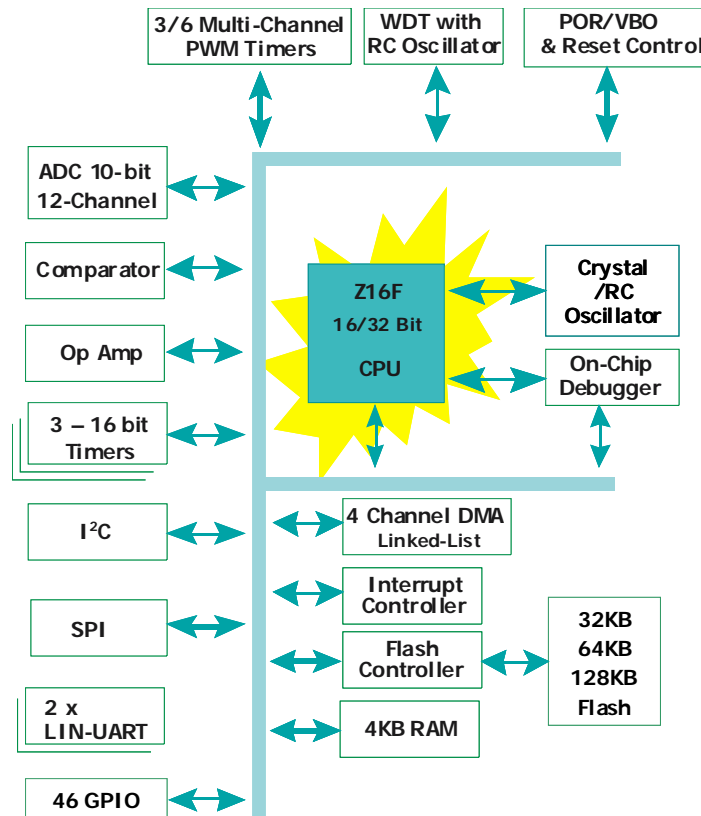


Figure 1. Z16FMC Motor Control MCU Block Diagram

## Z16FMC MCU Features

The Z16FMC MCU includes the following features:

- 20 MHz ZNEO CPU
- 128 KB internal Flash memory with 16-bit access and In-Circuit Programming (ICP)
- 4 KB internal RAM with 16-bit access
- An external interface that allows for seamless connections to external data memory and peripherals, including:
  - Six chip selects with programmable Wait states
  - A 24-bit address bus which supports 16 MB
  - Selectable 8-bit or 16-bit data bus widths
  - Programmable chip select signal polarity
  - ISA-compatible mode
- 12-channel, 10-bit Analog-to-Digital Converter (ADC)
- Operational Amplifier
- Analog Comparator
- 4-channel Direct Memory Access (DMA) controller which supports internal or external DMA requests
- One full-duplex 9-bit Universal Asynchronous Receiver/Transmitter (UART) with support for Local Interconnect Network (LIN) and Infrared Data Association (IrDA)
- Internal Precision Oscillator (IPO)
- Inter-Integrated Circuit (I<sup>2</sup>C) master/slave controller
- Enhanced Serial Peripheral Interface (ESPI)
- 12-bit Pulse Width Modulation (PWM) module with three complementary pairs or six independent PWM outputs with deadband generation and fault trip input
- Three standard 16-bit timers with Capture, Compare and PWM capability
- Watchdog Timer (WDT) with internal RC oscillator
- 76 General-Purpose Input/Output (GPIO) pins
- 24 interrupts with programmable priority
- On-Chip Debugger (OCD)
- Voltage Brownout (VBO) protection
- Power-On Reset (POR)
- 2.7V to 3.6V operating voltage with 5 V-tolerant inputs

- 0°C to +70°C standard temperature and –40°C to +105°C extended temperature operating ranges

The 12-bit PWM module has three complementary pairs or six independent PWM outputs supporting programmable deadband and fault protection trip input to provide the output control; while the ADC has a 2 μs conversion time and can be triggered automatically by the PWM module. A special Time Stamp feature automatically captures a 16-bit timer value based on the ADC sample/hold.

## Peripheral Interoperation

The peripheral set integrated on the MCU is essential for optimizing it to the application, but the interoperation of these peripherals can greatly affect the performance of the MCU in the application. The Z16FMC series provides Linked List DMA and automated interoperation between the ADC and Timer (Time Stamp feature) and between the Comparator and PWM outputs (fault response).

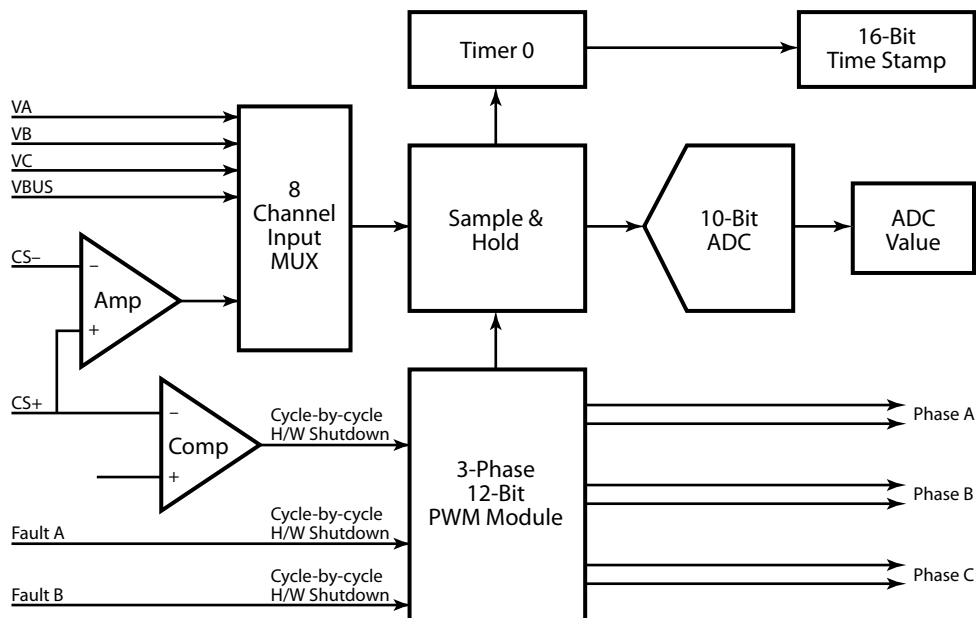


Figure 2. PWM Timing

## Linked List DMA

Direct Memory Access is a feature that allows data to be transferred on the bus independent of the CPU. Linked List DMA takes this simple concept to another level by using descriptors that provide the source and destination information, along with a link to the next descriptor to further reduce the overhead and real-time response requirements of the CPU.

Additionally, the multi-bus structure of the Z16F core allows data to be transferred across the data bus while instructions are being fetched from the Program bus further reducing the overhead of the transaction.

There are two basic ways to take advantage of this feature in Sensorless BLDC and more complex Permanent Magnet Synchronous Motors and Vector Control applications:

- The DMA can be used to automatically update the motor control PWM timers while the CPU handles all other system level tasks. Wave tables stored in RAM are updated by the CPU as required, but the act of sending the data to the PWM module is handled by the DMA. As a result, the CPU is released from the task of handling interrupts at the rate of the PWM frequency.
- The DMA can be used to offload the CPU and handle serial communications in order to optimize the real-time response to the PWM controller.

## Time Stamp Feature for Speed Control

Most microcontrollers use at least one dedicated comparator to detect the zero crossing of the input back EMF voltage signal so that the output-driving pulses can be synchronized and adjusted to properly regulate motor speed. An alternative approach based on Zilog's motor control MCU eliminates the need for this comparator by interconnecting the ADC with a timer.

In this case, the timer is used to generate an interrupt at a point where the back EMF voltage crosses the bus voltage. During this interrupt, the timer continues running in the background and the back EMF and bus voltages are sampled. The timer value is updated based on the difference between the voltages (and direction of the motor) to lock it onto the actual speed of the motor. This event is used to determine the commutation point and update the PWM to efficiently regulate the speed of the motor. This Time Stamp approach results in a very simple and cost-effective solution for smooth operation of the motor in steady state.

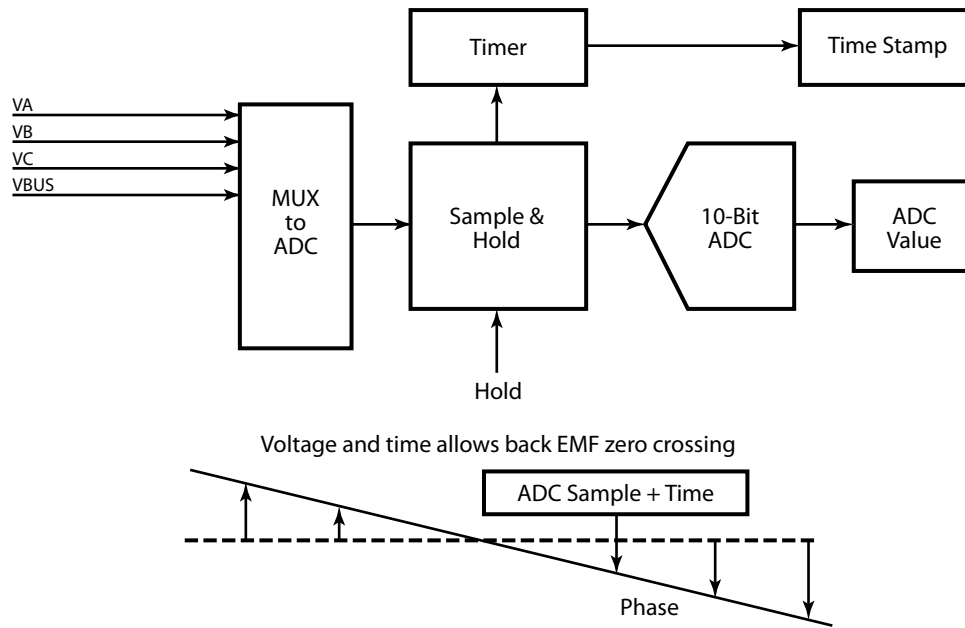


Figure 3. Time Stamp for Steady-State Speed Control

## Fault Response

Over-current faults can result from many different causes and are sometimes destructive. Shorted motor windings, shorted motor leads, problems in mechanical drives and linkages, a stuck rotor or changes in the load, breakdowns or misfiring of power devices and many other problems can arise - some of them permanent, some merely temporary. The Z16FMC Series features an over-current comparator that is directly coupled to the PWM module, thereby guaranteeing that a shutdown can truly occur in a cycle-by-cycle mode. This approach ensures minimum lag in responding to the fault condition.

In Figure 4, the current across the sense resistor generates the voltage that is presented to the negative input on the comparator. You can see that as the voltage generated across the sense resistor increases to the comparator threshold value, the PWM output is immediately driven Low until the following cycle. The next PWM cycle is generated normally once the voltage falls below the threshold.

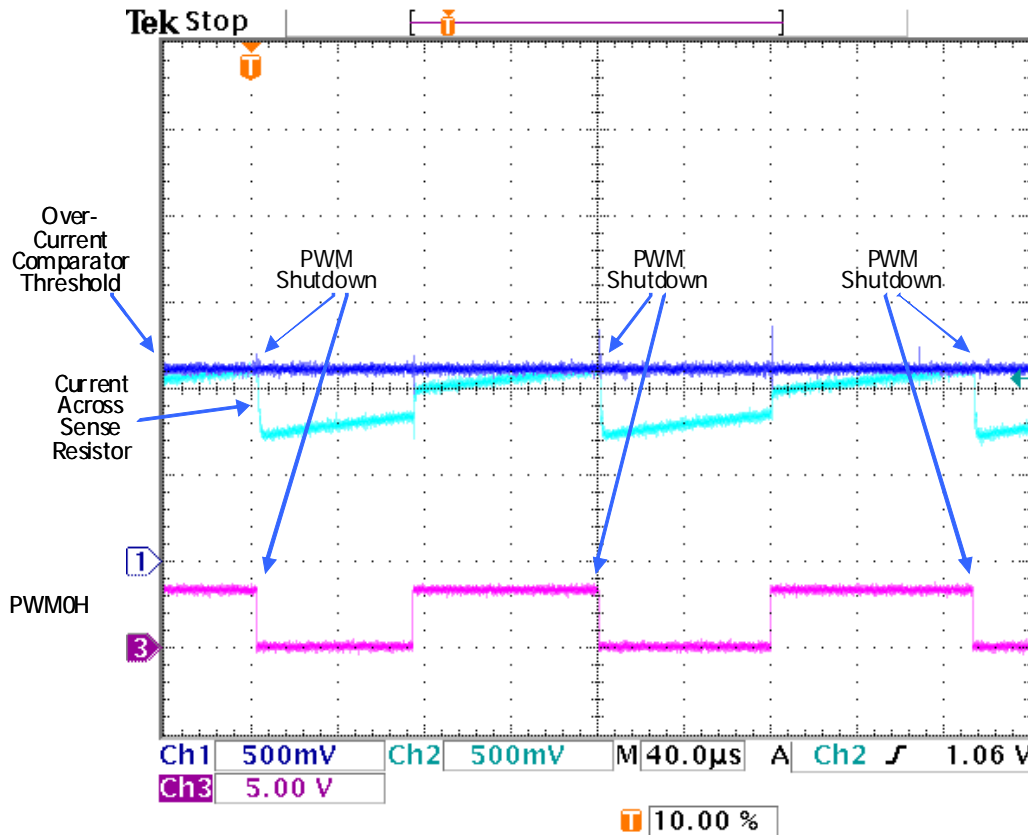


Figure 4. PWM Response to Over-Current Faults

## Upgrade Path

The Z16FMC Motor Control Series provides an excellent upward migration path from the Z8FMC16100 series and is also suitable for Permanent Magnet Synchronous Motors (PMSM) and AC vector control applications.

## Development Kit

The Sensorless BLDC Development Kit, available from Zilog, is based on Z16FMC28 device and uses high-efficiency IXYS MOSFETs to provide a very economical solution for control of a 30W three-phase BLDC motor. It ships complete with:

- Z16FMC28 Series MDS Development Board
- BLDC Motor Control Application Board



- Brushless DC (BLDC) motor
- Opto-isolated USB Smart Cable
- 5V DC universal power supply
- Zilog Developer Studio II IDE with a full ANSI C-Compiler
- Sample code

All source code is provided with the Development Kit (Z16FMC28200KITG), which can be purchased from a Zilog distributor. For details, visit the Zilog website ([www.zilog.com](http://www.zilog.com)).

## Reference Design Operation

Two motor phases are energized at any given time and the back EMF voltage is generated in the unenergized phase winding. The zero crossing of this induced voltage is detected for synchronization of the subsequent closed-loop control events. The Time Stamp feature of the Z8FMC16100 MCU provides for robust, efficient implementation of this critical sensing function without the requirement for an additional comparator. The algorithm for back EMF sensing is based on implementation of a Phase Locked Loop (PLL). This implementation is especially advantageous during startup, resulting in a very smooth increase in the motor speed, as well as nearly instantaneous reversal of direction of rotation on command. With the conventional approach, during the start-up sequence, power is applied to the windings in order to place the rotor in a known starting position, followed by commutation and start of back EMF sensing and control. In contrast, the PLL-based approach makes it possible to lock to the back EMF signal from the onset of the start-up phase, without the need for initial placement of the rotor in a specific position. Moreover, this approach significantly reduces the jerky movement of the motor during startup or reversal of direction.

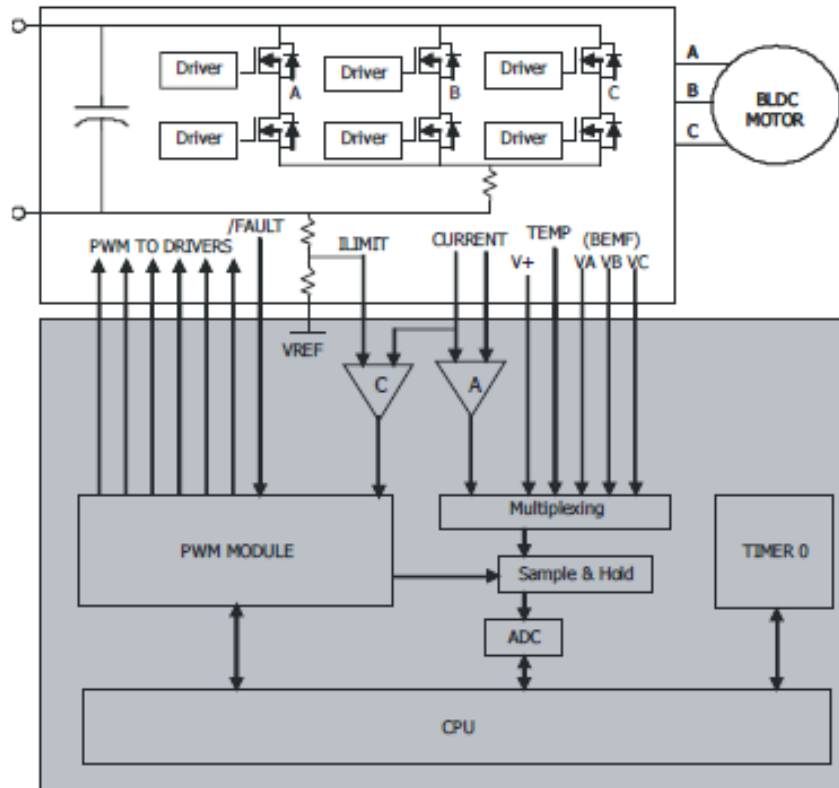


Figure 5. A Z8FMC16100 Reference Design



**Warning:** DO NOT USE IN LIFE SUPPORT

### **LIFE SUPPORT POLICY**

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

### **As used herein**

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

### **Document Disclaimer**

©2011 Zilog Inc. All rights reserved. Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

Z8, Z8 Encore!, Z8 Encore! XP, Z8 Encore! Motor Control Series, eZ80, ZNEO and Z16FMC are trademarks or registered trademarks of Zilog, Inc. All other product or service names are the property of their respective owners.