Space Vector Modulation of a 3-Phase BLDC Motor with the Z8FMC16100 MCU

AN036903-1215

Abstract

Space vector modulation techniques can be applied for AC induction motors, permanent magnet synchronous motors, and BLDC motor types. PMSM motors can be more efficient at smaller motor frame sizes compared with the an ACIM machine of the same size.

A 3-phase BLDC motor can be controlled by creating a rotating voltage reference vector within a hexagon; the speed of rotation of this voltage reference vector determines the frequency of motor rotation. The space vector modulation application discussed in this application note uses three Hall sensors for angular position feedback.

Constant cost pressure and increased consumer expectations have driven design engineers to seek minimal hardware solutions that extract maximum performance from motors used in consumer goods. This application note demonstrates how Zilog's Z8FMC MCU can implement efficient, cost-conscious vector modulation of a BLDC motor.

> **Note:** The source code file associated with this application note, <u>AN0369-SC01</u>, is available free for download from the Zilog website. This source code has been tested with ZDS II – Z8 Encore! version 5.0.0. Subsequent releases of ZDS II may require you to modify the code supplied with this application note.

Z8FMC16100 Series Flash Microcontrollers

The Z8FMC16100 Series of Flash MCUs is based on Zilog's advanced 8-bit Z8 Encore! CPU core. The MCUs in this series are optimized for motor control applications and support control of single- and multiphase variable-speed motors. Target applications are consumer appliances, HVAC, factory automation, refrigeration, and automotive applications, among others.

To rotate a 3-phase motor, three AC voltage signals must be supplied and phase-shifted 120 degrees from each other. To control a 3-phase motor, the MCU must provide six Pulse Width Modulation (PWM) outputs. The Z8FMC16100 Series Flash MCU features a flexible PWM module with three complementary pairs supporting deadband operation and fault protection trip input. These features provide multiphase control capability for various motor types, and ensure safe operation of the motor by providing immediate shutdown of the PWM pins during a fault condition.

Discussion

An electric motor consists of a stator and a stationary frame in which a rotating component, or rotor, is mounted on a shaft and bearings. In a 3-phase BLDC motor, the stator is

laced with three sets of inductor windings energized by three AC voltage inputs that are phase-offset 120 degrees from each other to produce a rotating field of magnetic flux inside the motor. This field creates induced magnetic flux in short passive windings on the rotor. The stator field exerts a magnetic force on the rotor flux field, resulting in torque on the output shaft.

In a 3-phase motor control application, the input to the motor is produced by a 3-phase inverter bridge. A bridge contains three complementary source/drain transistor pairs which connect either ground or high-voltage DC to each of its three outputs in response to digital control signals from the microcontroller. The microcontroller uses PWM on the bridge control signals to generate three approximately-sinusoidal AC waveforms on the bridge outputs, with the required 120-degree phase offset.

The duty cycle of each microcontroller PWM output is varied to control the period and amplitude of the generated AC signal which, in turn, determines the speed and torque of the motor.

Theory of Operation

Similarly to third harmonic-injected sinusoidal PWM, the Space Vector Modulation method utilizes about fifteen percent more of the available bus voltage, therefore increasing the efficiency of motor operation.

Unlike a non-third harmonic-injected sinusoidal PWM, the neutral point of the phase voltages is constrained to one-half of the bus voltage, as illustrated in Figure 1.
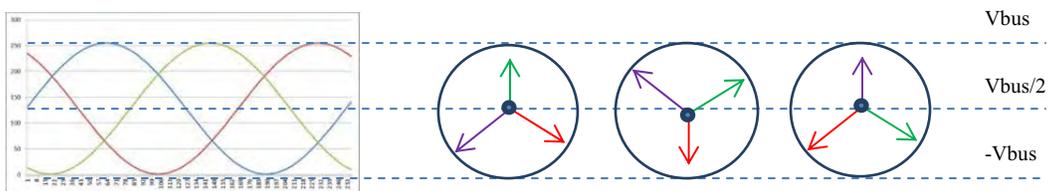

Figure 1. The Rotating Vectors are Constrained by ± V_{BUS} and the Center of the V_{BUS} Voltage

Space vector modulation is not confined to the limits of the V_{BUS} and the center voltages, and can float in space as illustrated in Figure 2.
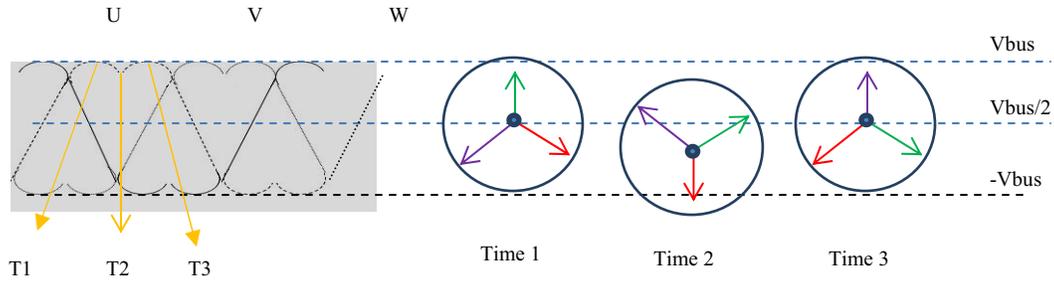
Figure 2. Center Voltage of the Space Vector Floats in Space

Unlike sinusoidal PWM which generates sinusoidal currents separately in each push/pull stage of the inverter, space vector modulation operates the entire inverter as a single unit to produce the sinusoidal currents. In doing so, the inverter is operated in eight different states within the hexagon, two of which are referred to as zero vectors because they produce no voltages, and six states which produce non-zero voltages.

The rotating reference voltage V_S within this hexagon, seen in Figure 3, is represented by a space vector using the following equation:
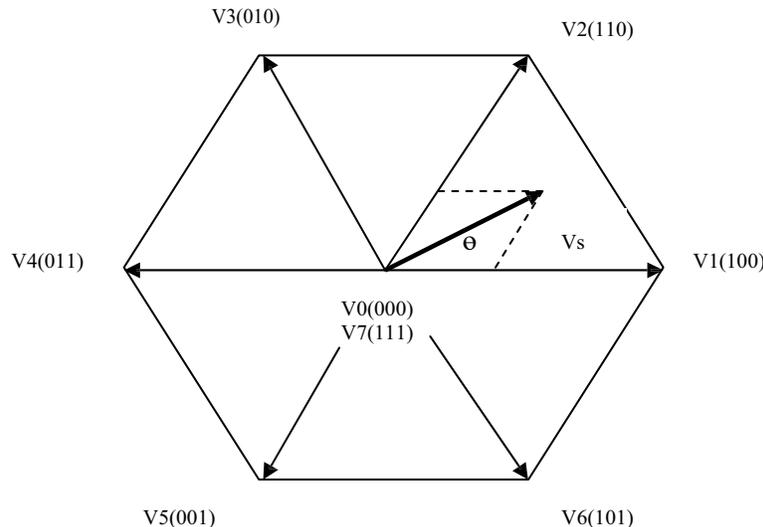
$$V_S = V_S * e^{j\Theta}$$

Figure 3. The Rotating Reference Vector V_{REF} within the Hexagon

With vector addition, the resulting reference space vector is:

$$V_S = (r_1 * V_1) + (r_2 * V_2)$$

To produce this rotating vector, the angular position within any two base vectors must be known, as shown in Figure 4.
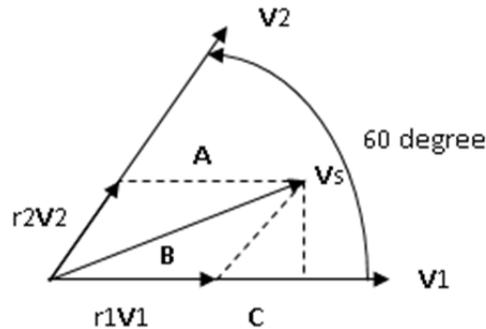

Figure 4. The Location of the Reference Vector in between any of Two Base Vectors

Knowing the angle of rotation and the adjacent base vectors within the hexagon, the scalar coefficients for the adjacent base vector r_1 and r_2 must be calculated to time-modulate the base vectors V_1 and V_2 toward generating the resulting voltage reference vector V_S.

After the angular information is obtained, the scalar quantities r_1 and r_2 can be calculated using the following equations:

$$r_1 = m \cdot \sqrt{3} \cdot \sin(60 - \Delta\theta)$$

$$r_2 = m \cdot \sqrt{3} \cdot \sin(\Delta\theta)$$

In the above equations, *m* is the magnitude of the rotating space vector V_S.

The time periods for which the adjacent base vectors are modulated to obtain the reference vector can be calculated using equations 1, 2, and 3:

Equation 1

$$t_0 = (1 - t_1 - t_2) * T$$

Equation 2

$$t_1 = r_1 * T$$

Equation 3

$$t_2 = r_2 * T$$

In these three equations, T is the sum of t_0, t_1, and t_2, and cannot be greater than the time period of the PWM. Next, t_0 becomes the time period for which either or both zero vectors are applied in combination with t_1 and t_2, as illustrated in Figure 5.
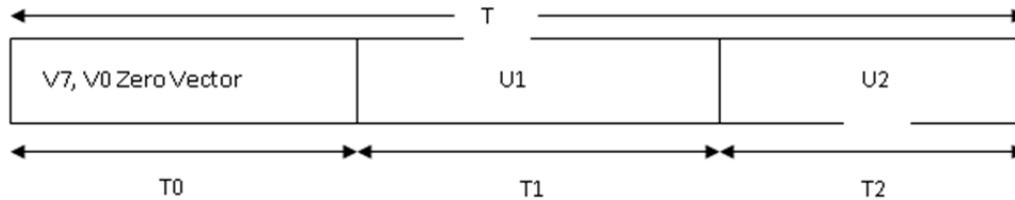
Figure 5. The Switching Times for Base Vectors V_1, V_2, and Zero Vectors

To find the time periods t_0, t_1, and t_2, the angles are determined by using V_1 as the reference axis in a counterclockwise direction to determine the base vector angle to be subtracted from the angle of the reference vector.

Reference Angle and Reference Voltage Calculations

Using equations 1, 2, and 3 above, if the bus voltage is 24 V and the desired reference vector magnitude is 12 V, then the following equation can be calculated:

$$12 * e^{j*190}$$

In this equation, the angle is 190 degrees and the adjacent base vector is V_4, which is 180 degrees. Therefore, r_1 and r_2 can be calculated using the following equations:

$$r_1 = \sqrt{3} * \frac{12V}{24V} * \sin[60deg - (190deg - 180deg)] = 0.663$$

$$r_2 = \sqrt{3} * \frac{12V}{24V} * \sin(190deg - 180deg) = 0.15$$

If the PWM period is $T = 50 \mu s$, then the time duration for either zero vector V_0, V_7 is:

$$t_0 = (1 - r_1 - r_2) * T = 9.3 \mu s$$

The time duration for t_1 is:

$$t_1 = r_1 * T = 33.2 \mu s$$

And the time duration for t_2 is:

$$t_2 = r_2 * T = 7.5 \mu s$$

Space vector control allows for different switching combinations using t_1 and t_2 based on the choice of the null vectors which are applied for duration of time t_0. Applying the zero vectors V_0, V_7, or both V_0 and V_7 results in different switching patterns to generate either less total harmonic distortion or to reduce linear switching power losses in the switching

devices. Using these V_0, V_7 zero vectors can serve to obtain a regenerative braking effect, especially when using ACIM. This application note discusses the alternate reverse switching mode pattern.

To reduce linear switching losses with either V_0 or V_7, zero vectors can be applied in sequence; i.e, using V_0 as the zero vector in the sequence $t_1 \rightarrow t_2 \rightarrow t_0$ or using the V_7 zero vector in the same sequence. In both cases, each of the three phases in the inverter does not switch for one-third of the time in a cycle.

However, different combinations of switching sequences have different effects on the inverter circuit, depending on the size of the bootstrap capacitors used for the high- and low-side drivers. If the V_0 zero vector is used, the bootstrap will still work because the capacitors can discharge; however, such may not be the case when using the V_7 vector as a zero vector.

Table 1 presents PWM timing examples for using the null vector V0 only.

Table 1. PWM Timing for Each of Six Sectors Using V0 as Zero Vector

Sector 1	Sector 2	Sector 3	Sector 4	Sector 5	Sector 6
PhsA = $t_1 + t_2$	PhsA = $t_0 + t_1$	PhsA = 0	PhsA = t_0	PhsA = t_2	PhsA = 100%
PhsB = t_2	PhsB = 100%	PhsB = $t_1 + t_2$	PhsB = $t_0 + t_1$	PhsB = 0	PhsB = t_0
PhsC = 0	PhsC = t_0	PhsC = t_2	PhsC = 100%	PhsC = $t_1 + t_2$	PhsC = $t_0 + t_1$

Application

To apply space vector theory, the Z8FMC16100 microcontroller's PWM module is configured as three complementary output pairs. Each output pair controls one complementary source/drain transistor pair in the inverter bridge. The PWM module is configured to insert a deadband between ON states to prevent leakage that might occur if one transistor in a pair turns on before the other is fully off.

Each PWM output pair produces a stream of complementary on/off pulses to activate the corresponding source or drain transistor in the inverter bridge. The voltage of each bridge output varies with the source/drain pulse duty cycle.

The period of each PWM cycle is configured to be 50µs; the PWM module generates an interrupt request at the end of each cycle to calculate the PWM timings for of the space vector modulation signals. These signals are loaded into the three PWM registers for Phase A, Phase B, and Phase C. Therefore, the primary goal of the ISR is to update the duty cycle value for each PWM channel, as required, to produce the appropriate AC waveforms at the inverter bridge outputs.

The frequency of the rotating vector is calculated as:

$$\text{Freq} = \frac{\text{LUToffset}}{\text{PWMperiod * LUTsize}}$$

In the above equation, the *LUToffset* value is a 16-bit integer index, of which only the upper byte is used to select the Look-Up Table (LUT) entries.

The synchronous speed of the rotor can then be calculated as:

$$\text{Speed} \; = \; 120 * \frac{\text{Freq}}{\text{Poles}}$$

The timer0 ticks captured for the PI loop are calculated as:

$$\text{TimerTicks} \; = \; \frac{\dfrac{\text{MCUclockFreq}}{\text{TimerPrescaler}}}{\dfrac{\text{LUToffset}}{\text{PWMperiod} * \text{LUTsize}}}$$

Equipment Used

The following equipment is used to demonstrate the space vector modulation technique. The first four items are contained in the MultiMotor Development Kit (ZMULTIMC100ZCOG).

- MultiMotor Development Board (99C1358-0001G)

- 24 V AC/DC power supply

- LINIX 3-phase 24 VDC, 30 W, 3200 RPM BLDC motor (45ZWN24-30)

- Opto-Isolated UART-to-USB adapter (99C1359-001G)

- Z8FMC MultiMotor MCU Module (99C1395-001G) – Order separately

- Opto-Isolated USB SmartCable (99C0968) – Order separately

- Digital Oscilloscope

- PC with Internet access and at least two open USB ports

- Tektronix DPO 2014B

Hardware Setup

Figure 6 illustrates the application hardware connections required to operate the motor with space vector modulation.
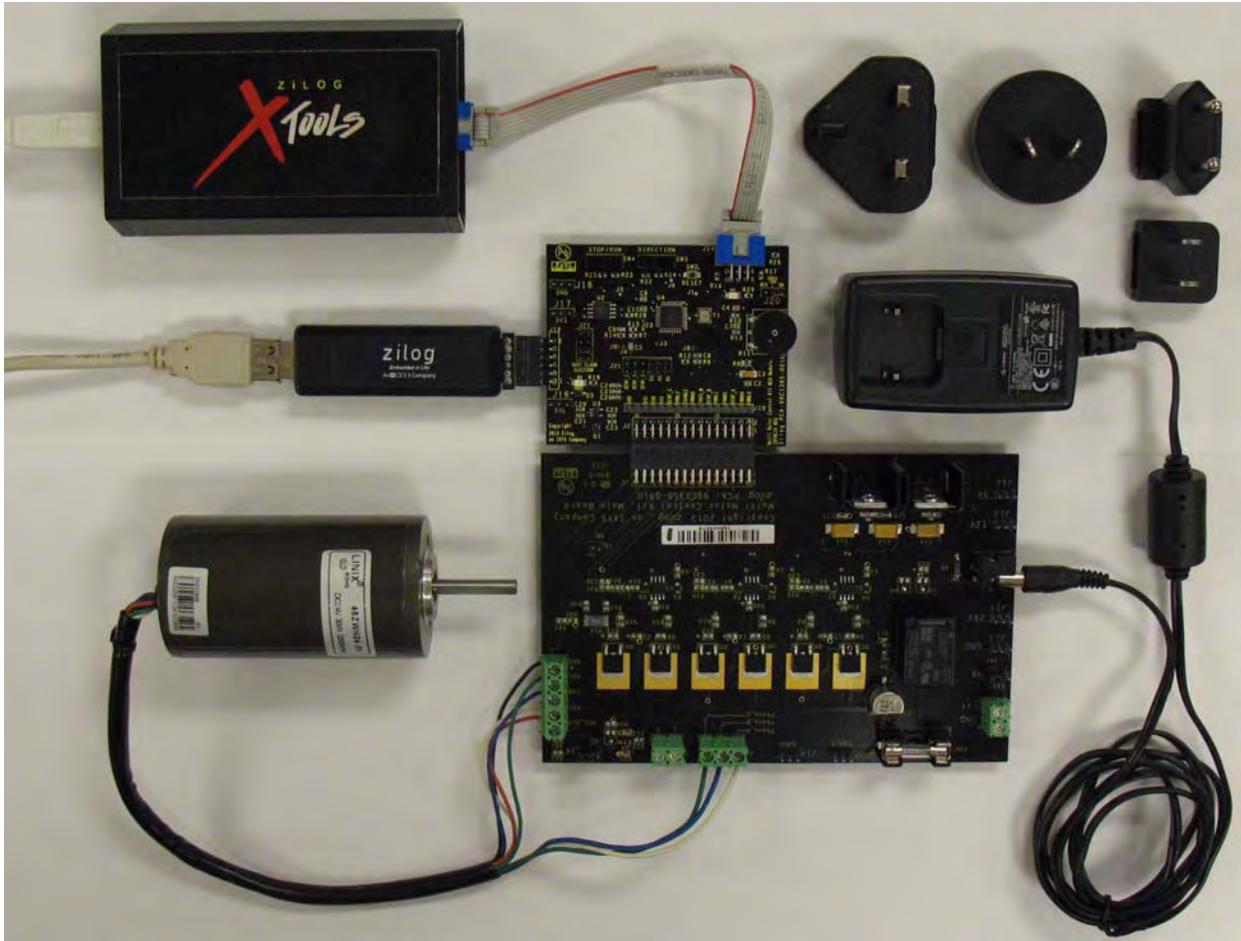
Figure 6. The MultiMotor Development Kit with the Z8FMC MCU Module and Smart Cable

Testing Procedure

Observe the following procedure to test space vector modulation on the Z8FMC16100 MCU Module.

1. Download ZDS II – Z8 Encore! 5.0.0 (or newer) from the Zilog Store and install it onto your PC.

2. Download the AN0369-SC01.zip source code file from the Zilog website and unzip it to an appropriate location on your PC.

3. Connect the hardware as shown in Figure 6.

   a. The cables from the opto-isolated USB SmartCable and the UART-to-USB adapter must be connected to two of the PC's USB ports.

   b. Download and install the drivers for the SmartCable and the UART-to-USB adapter, if required. For assistance, refer to the MultiMotor Series Development Kit Quick Start Guide (QS0091).

4. Power up the MultiMotor Series Development Board using the 24 V DC adapter that is included in the Kit.

5. Using a serial terminal emulation program such as HyperTerminal, TeraTerm, or Real-Term, configure the serial port to 57600-8-N-1-N. A console screen should appear on the PC which will show the status of the motor and allow changes to the motor's operation.

6. Launch ZDS II – Z8 Encore!, and select **Open Project** from the **File** menu. Browse to the directory on your PC into which you downloaded the AN0369-SC01 source code. Locate the `AN0369_SC01.zdsproj` file, click to highlight it, and select **Open**.

7. Ensure that the RUN/STOP switch on the Z8FMC16100 MCU Module is in the STOP position.

8. In ZDS II, compile and flash the firmware to the Z8FMC16100 MCU Module by selecting **Rebuild All** from the **Build** menu. Next, select **Debug → Download code**, followed by **Debug → Go**.

9. Set the RUN/STOP switch on the Z8FMC16100 MCU Module to RUN. The motor should begin turning.

10. In the GUI terminal console, enter the letter U to switch to UART control; a menu similar to the example shown in Figure 7 should appear. As a result, commands can now be entered using the console to change the motor's operation.
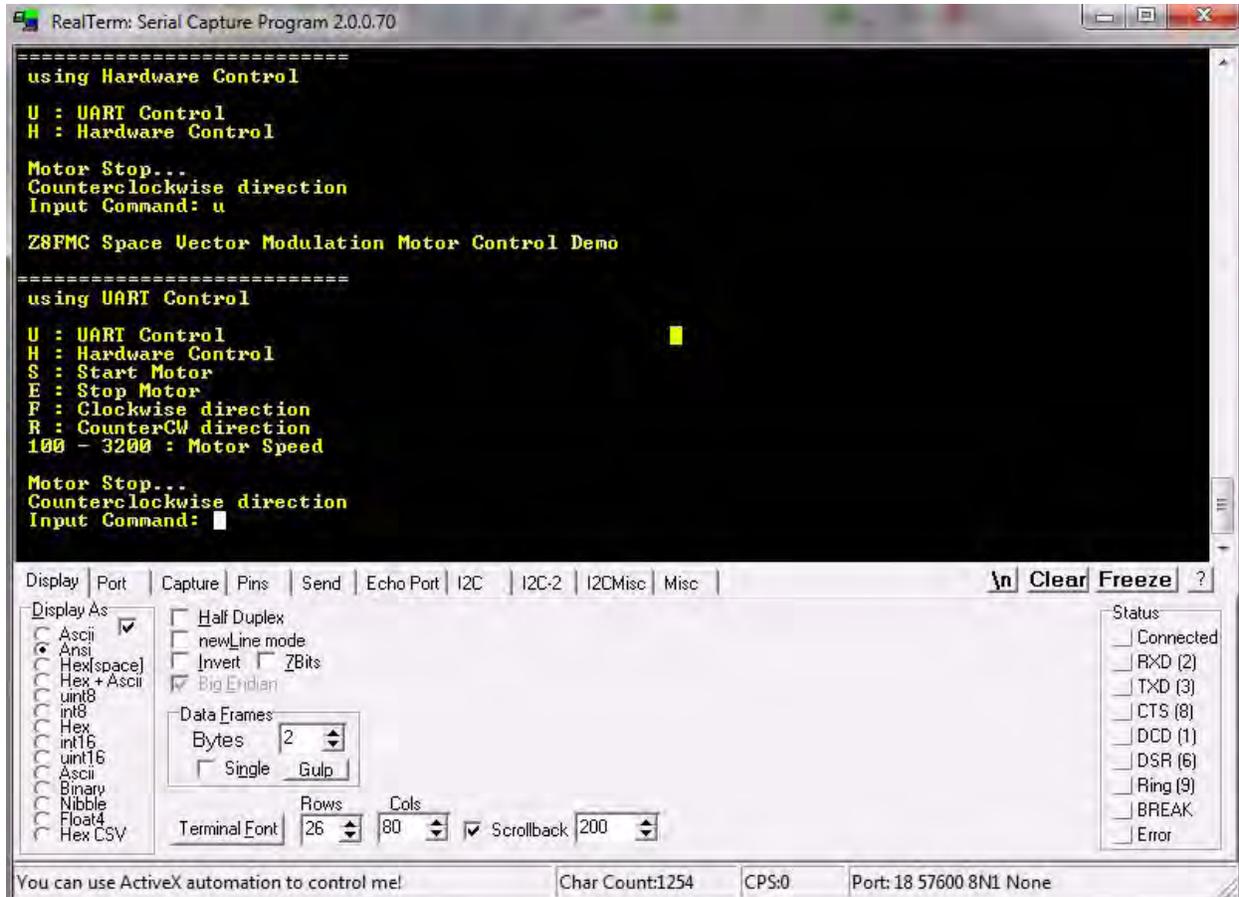
Figure 7. GUI Terminal Showing the UART Control

11. At the `Input Command:` prompt, enter the letter `H` to establish hardware control; see Figure 8..
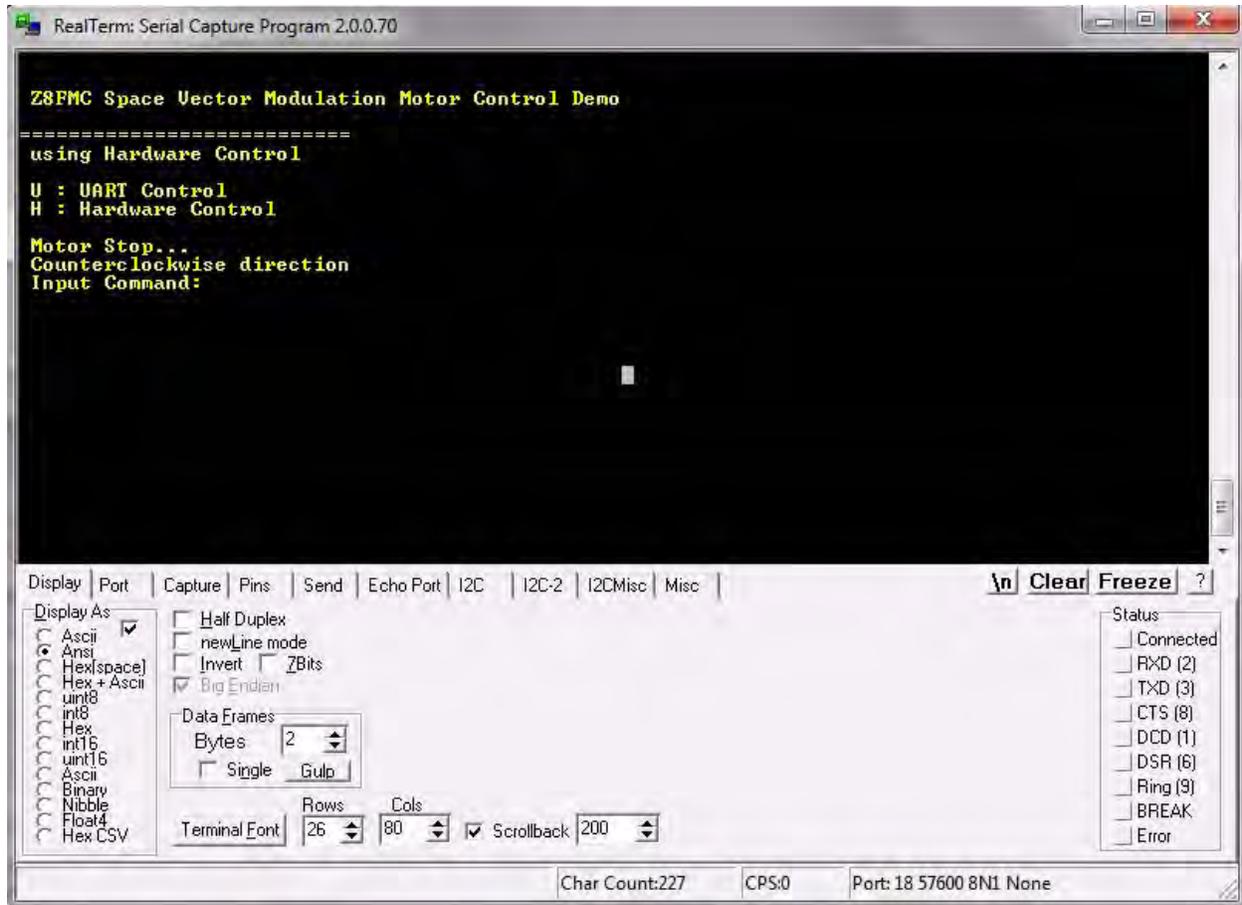
Figure 8. GUI Terminal Showing Hardware Control

Results

In this application, three oscilloscope probes are connected to the Phase A, Phase B, and Phase C offsets of the MultiMotor Series Development Board to show the space vector alternate reverse switching pattern modulation. These probes were also connected to BEMF voltage dividers to monitor the generated BEMF voltages and, ultimately, to view the associated voltage-switching pattern waveforms.

Figure 9 illustrates the alternate-reverse space vector modulation pattern on Phase A, Phase B, and Phase C.

Figure 9. Alternate-Reverse Space Vector Modulation Pattern

Figure 10 illustrates the alternate-reverse voltage waveform on Phase A, Phase B, and Phase C.
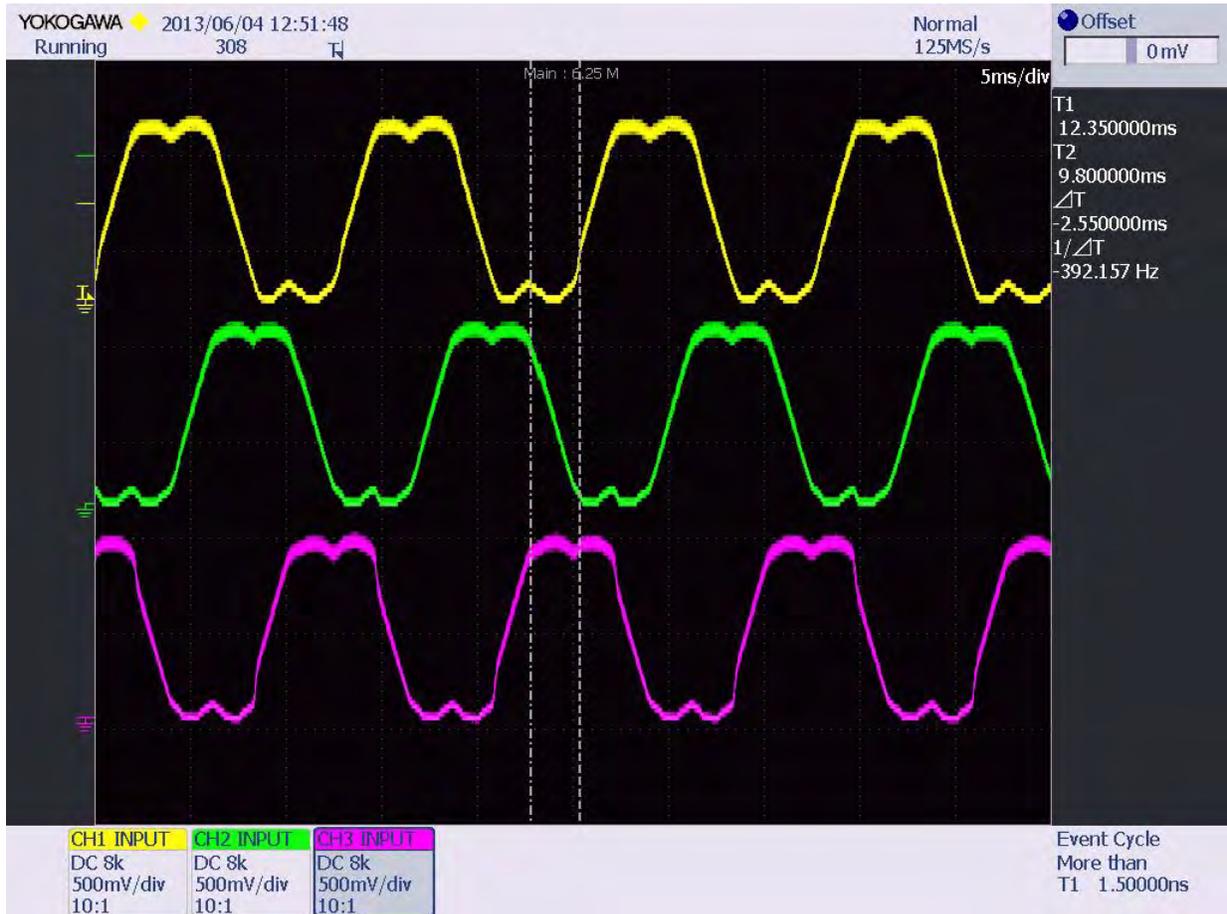
Figure 10. Alternate-Reverse Voltage Waveforms

Closed-Loop Control Performance

To monitor closed-loop speed control performance, the motor speed was set to 2000 RPM at a nominal operating voltage of 24 V. As this operating voltage was increased and decreased by ±4 V, motor speed was observed to remain constant. To test the PI loop under load, the motor load was increased, which caused the PI loop to quickly ramp up the current to maintain the set speed. The PI loop stability was verified by observing the voltage sine wave while loading the running motor, a condition for which the sine wave period time must be maintained constant in both amplitude and frequency with no jitter. The closed-loop speed control essentially provides power operation of the motor which remains constant.

Open-Loop Control Performance

To monitor the performance of the speed control function while operating in an open loop, the motor speed was set to 2000 RPM at a nominal operating voltage of 24 V. As this operating voltage was increased or decreased by ±4 V, motor speed was observed to vary accordingly. The motor load was then increased, which caused the motor current to be increased while its speed dropped.

Summary

The purpose of this application is to demonstrate the operation of a BLDC- or PMSM-type machine using the Space Vector Modulation technique. To generate sinusoidal voltages and currents, a voltage reference vector is rotated 360 degrees within a hexagon. Each of the six sectors within this hexagon creates unique switching patterns for the space vector modulation.

Space vector modulation has the advantage of utilizing about fifteen percent more of the available bus voltage. Formulas discussed in this document have been shown to calculate the space vector modulation timings and motor frequency. Because the frequency calculations include the PWM period, all space vector sinusoidal wave constructions are executed in the PWM interrupt service routine. The execution time for the sine wave reconstruction in the PWM service interrupt routine is approximately 18 µs. The execution time of the Hall interrupt service routine is approximately 16 µs. Both execution times are based on a 20 MHz external clock.

To maintain synchronization and commutation angle between the reference vector frequency and rotor frequency, the Hall interrupt service routine captures the binary Hall state upon each interrupt and fetches the corresponding reference angle from a Look-Up Table (LUT). The High byte of the PWM sine Look-Up Table index is then used to fetch the next value from the Sine Look-Up Table. Any positive or negative offset value to this high byte of the PWM sine look-up table will increment or decrement the frequency of rotation of the reference vector.

Space vector modulation has the advantage of commutating a BLDC or PMSM motor with less acoustical and electrical noise, because the sine current through the windings has no steep current transitions. The effects of total harmonic distortions and linear switching power losses can be further manipulated by applying different space vector modulation switching schemes. Such manipulations allow for higher life expectancy of ripple current capacitors and ball bearings because the sinusoidal commutation approach causes virtually no torque or current ripple in an ACIM, PMSM, or BLDC machine. In addition to electrical and acoustical noise reduction, the PWM sine approach also increases the efficiency of a BLDC-/PMSM-type motor due to its fifteen percent higher bus voltage utilization.

The example application and techniques described in this document should prove helpful for anyone who intends to develop motor control applications based on the Z8FMC16100 Series family of microcontrollers.

Appendix C. References

Documents associated with the Z8FMC16100 Series of products are listed below. Each of these documents can be obtained from the Zilog website by clicking the link associated with its document number where indicated.

- Z8 Encore! CPU User Manual (UM0128)

- Z8FMC16100 Series Product Specification (PS0246)

- MultiMotor Series Development Kit Quick Start Guide (QS0091)

- MultiMotor Series Development Kit User Manual (UM0262)

- BLDC Motor Control Using Sensored Sinusoidal PWM Modulation with the Z8FMC16100 MCU Application Note (AN0367)

- Three-Phase Hall Sensor BLDC Driver Using the Z8FMC16100 MCU Application Note (AN0368)

- Sensorless Brushless DC Motor Control with the Z8FMC1600 MCU Application Note (AN0370)

The following external documents offer sound fundamentals for understanding motor control concepts.

- *Motor Control Electronics Handbook*, Richard Valentine; McGraw Hill.

- *Lehrstuhl fuer Electrische Antriebssysteme und Leistungselektronik*, Prof. Dr. Ing. Ralph Kennel, Technische Universitaet Muenchen.

- *Electric Machinery*, Peter F. Ryff.

Appendix D. Schematic Diagrams

Figures 11 and 12 show the schematics for the Z8FMC MCU Module.
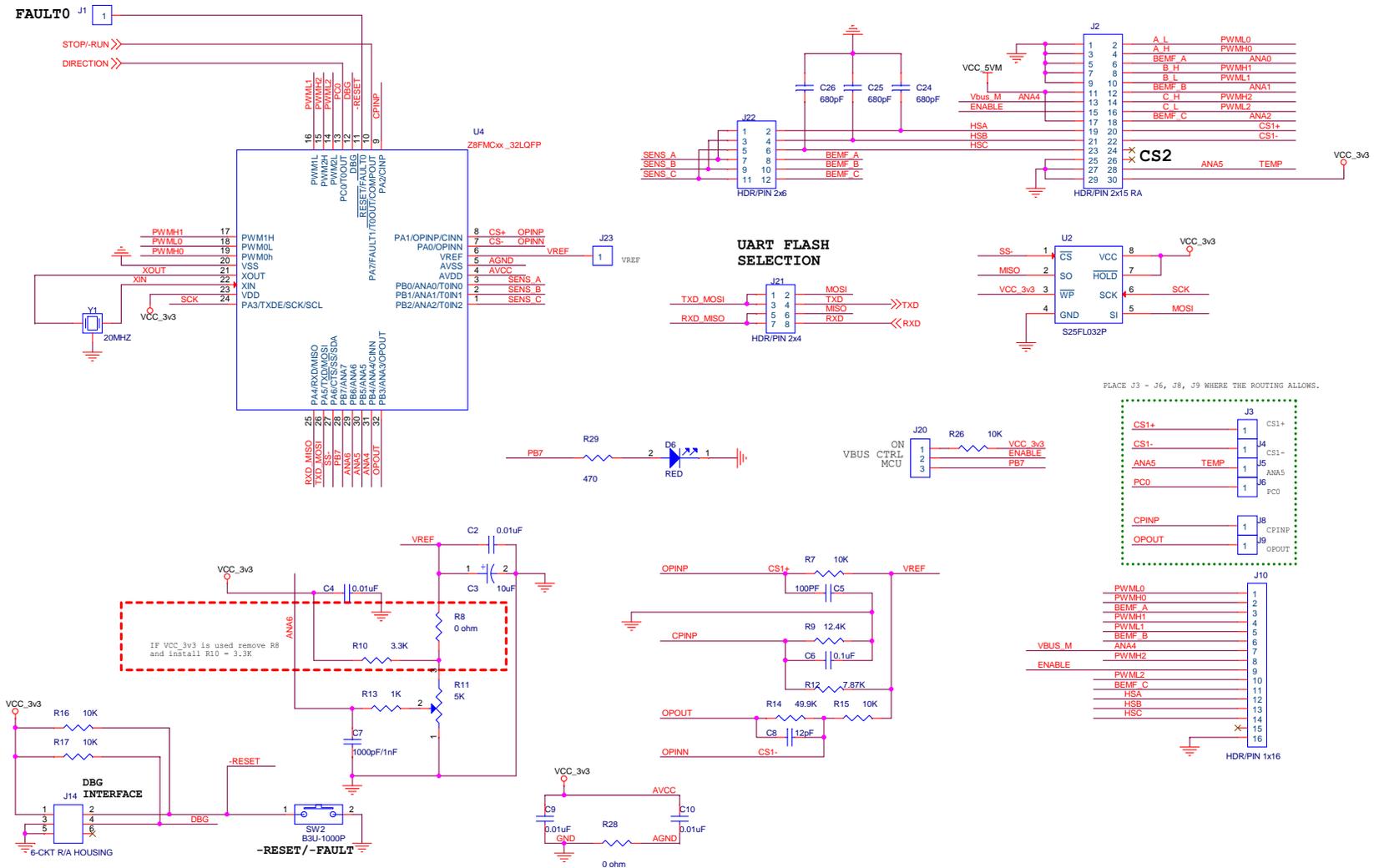
Figure 11. Z8FMC MultiMotor MCU Module, #1 of 2
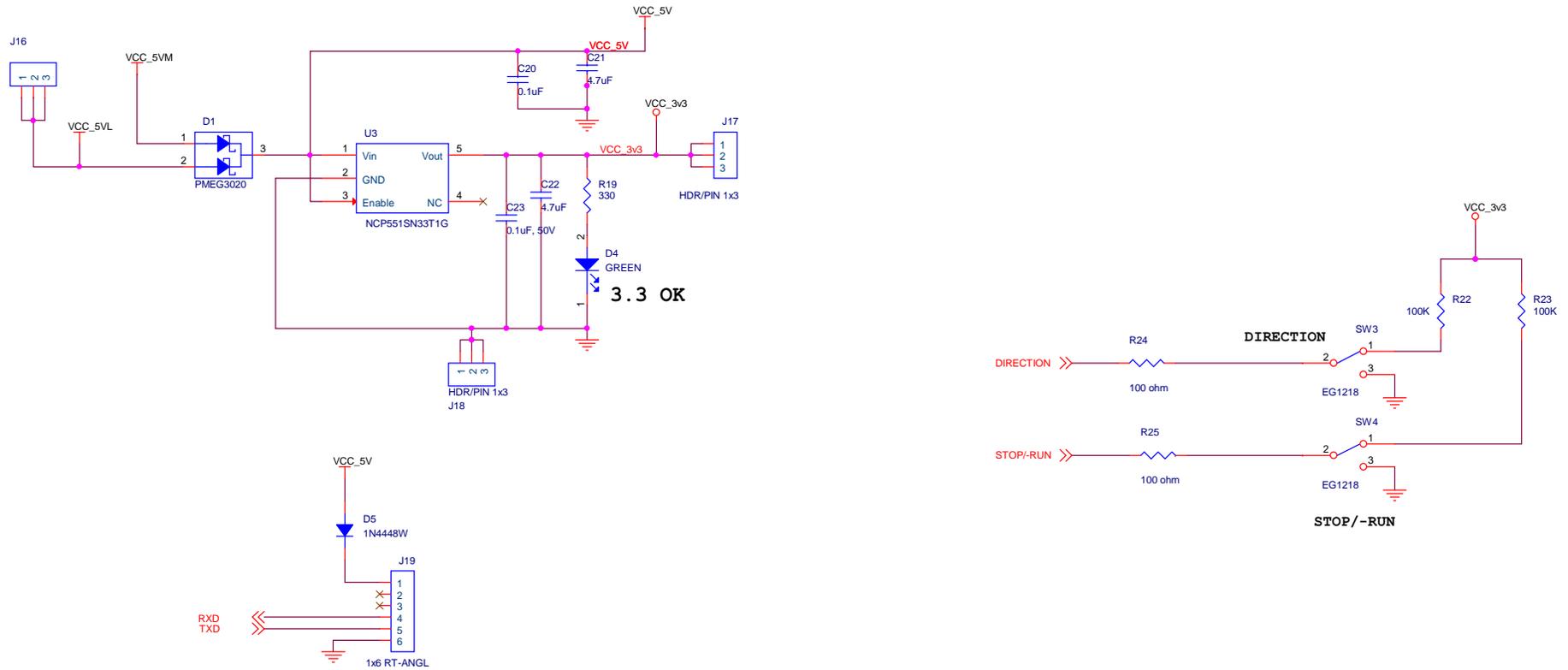
Figure 12. Z8FMC MultiMotor MCU Module, #2 of 2

Figures 13 and 14 show the schematics for the MultiMotor Main Development Board.
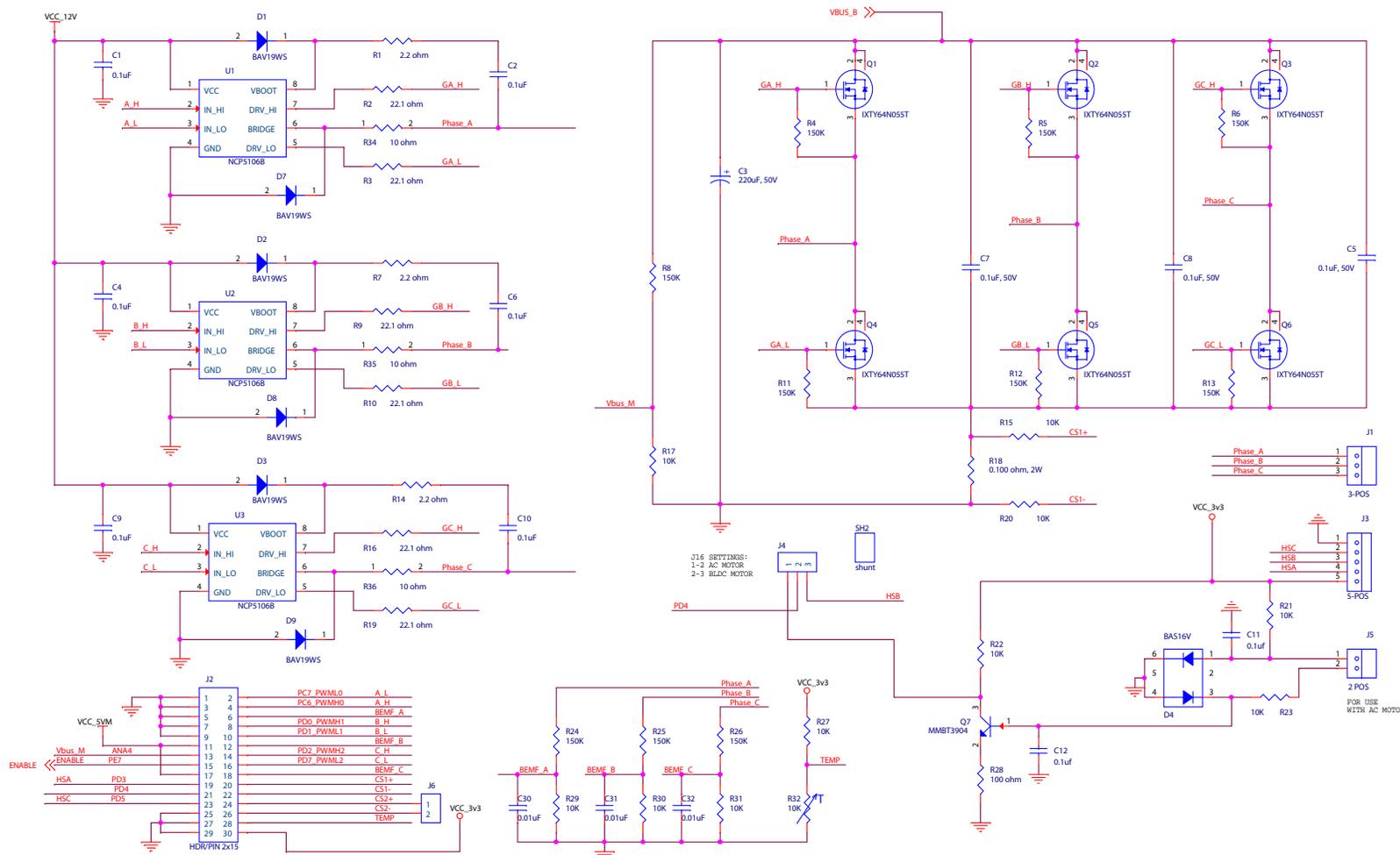
Figure 13. MultiMotor Development Board, #1 of 2
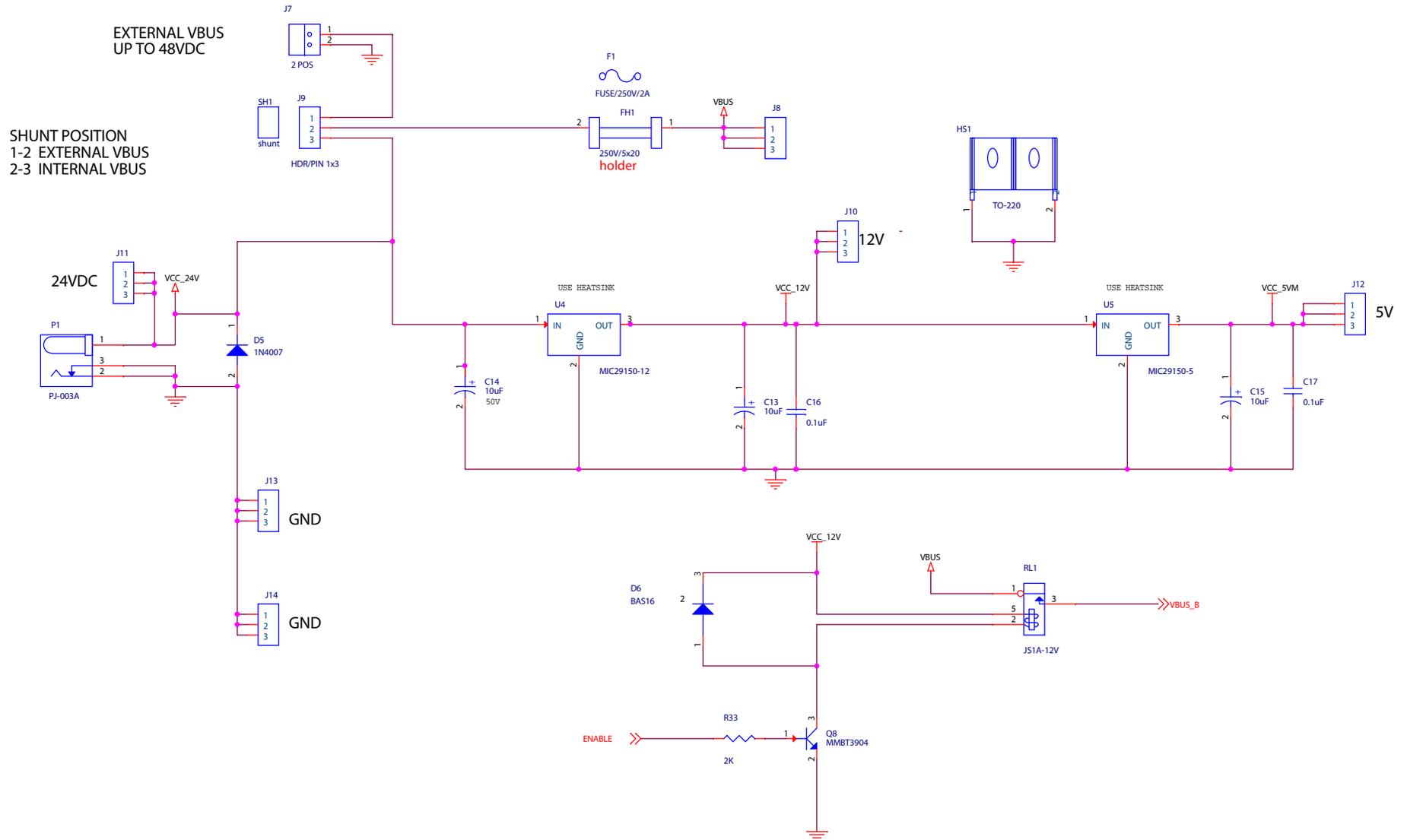
Figure 14. MultiMotor Development Board, #2 of 2

Appendix E. Flow Charts
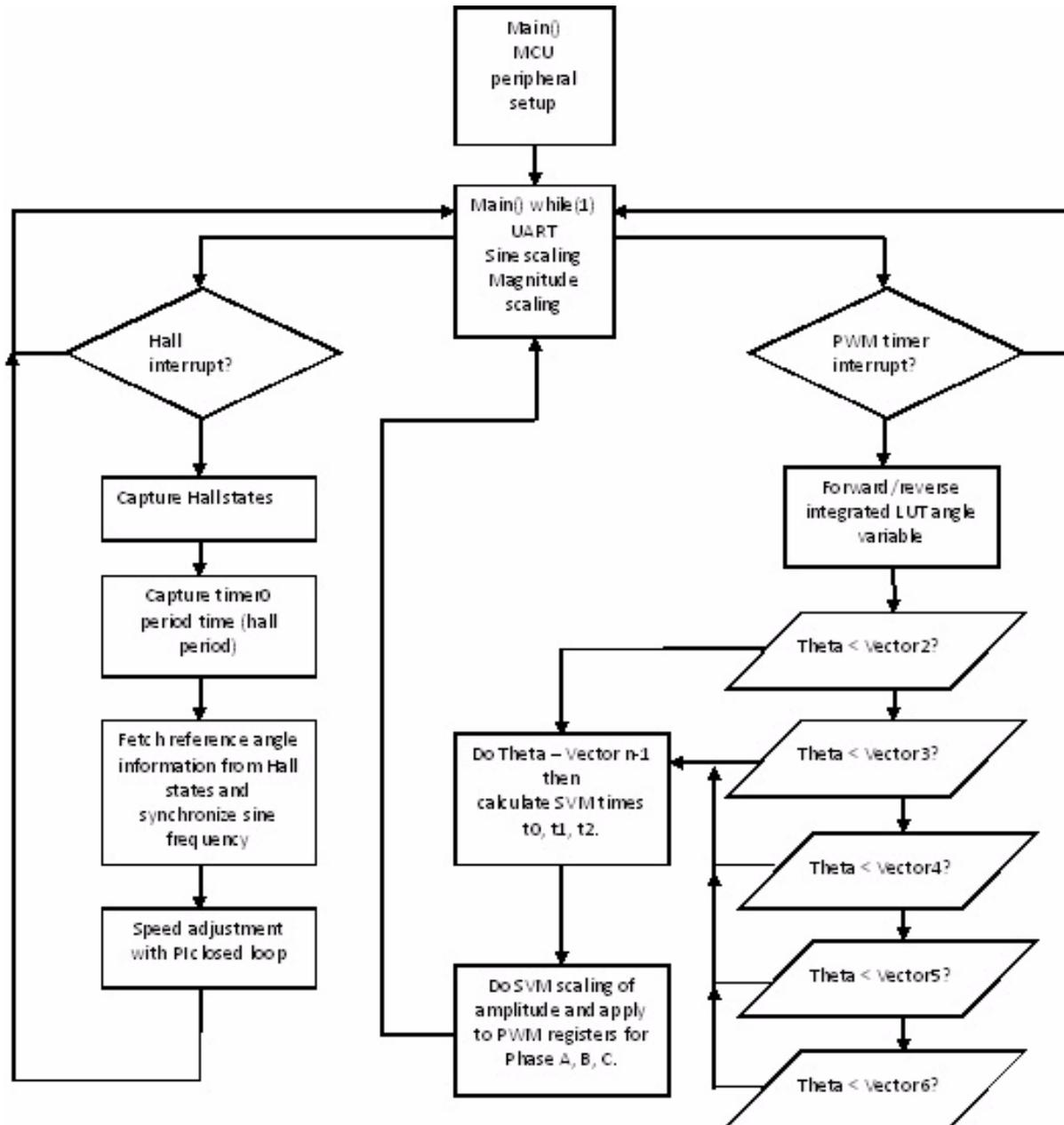
Figure 15 presents the typical flow of the space vector control routine.

Figure 15. Space Vector Modulation Flow

Customer Support

To share comments, get your technical questions answered, or report issues you may be experiencing with our products, please visit Zilog's Technical Support page at http://support.zilog.com.

To learn more about this product, find additional documentation, or to discover other facets about Zilog product offerings, please visit the Zilog Knowledge Base at http://zilog.com/kb or consider participating in the Zilog Forum at http://zilog.com/forum.

This publication is subject to replacement by a later edition. To determine whether a later edition exists, please visit the Zilog website at http://www.zilog.com.

> ⚠ **Warning:** DO NOT USE THIS PRODUCT IN LIFE SUPPORT SYSTEMS.