Freescale Semiconductor
Application Note

Document Number: AN4718
Rev. 0, 04/2013

3-Phase BLDC Hall Sensor Application Using S12ZVM

Based on the MTRCKTSBNZVM128 Development Kit

by: Petr Cholasta

Contents

1 Introduction

This application note introduces MC9S12ZVML128 device in a 3-phase Hall sensor-based BLDC motor control application.

The MC9S12ZVML128 and MC9S12ZVMC128 devices offer right blend of digital programmability and high precision analog to streamline the final design. The MC9S12ZVML128 and MC9S12ZVMC128 integrate a 16-bit microcontroller build on proven S12 technology, an automotive voltage regulator, a LIN or CAN interface, and a gate driver unit in order to drive six external MOSFETs.

The 3-phase BLDC Hall sensor application is based on the MTRCKTSBNZVM128 development kit Section 3., “MTRCKTSBNZVM128 Development Kit, www.freescale.com/AutoMCDevKits”. To get the application running, modification of the board jumper settings and a download of the AN4718SW code to the S12ZVM microcontroller are required. For further details, see the Quick Start Guide Section 4., “MTRCKTSBNZVM128HQSG Quick Start Guide, www.freescale.com/AutoMCDevKits”.

Part of this document is the software AN4718SW.

2 Configuration of MC9S12ZVML128 modules

The BLDC Hall sensor-based motor control application uses the following set of peripheral modules:

1. Clock, Reset, and Power Management Unit (CMPU)
2. Pulse Width Modulator with Fault Protection (PMF)
3. Programmable Trigger Unit (PTU)
4. Analog-to-Digital Converter (ADC)
5. Timer module (TIM)
6. Gate Drive Unit (GDU)
7. Port Integration Module (PIM)
8. Serial Communication Interface (SCI)
9. Interrupt (INT)

The module’s usage is described in subsequent sections. Detailed information can be found in the MC9S12ZVML128 reference manual Section 1., “MC9S12ZVM Family Reference Manual, www.freescale.com”.

An application software peripherals code is developed to meet the following specification:

1. Targeted at the MC9S12ZVML128 Evaluation board
2. BLDC motor control using Hall sensors
3. 20 kHz PWM switching frequency
4. 1 ms speed and current loop calculation period
5. DC-bus voltage measurement using the S12ZVM HD pin
6. DC-bus current measurement using an internal operational amplifier AMP0
7. FreeMASTER control enabled

For detailed information see Section 3, “Hall sensor-based BLDC motor control application”.

2.1 Clock, Reset, and Power Management Unit

The Clock, Reset, and Power Management Unit (CMPU) is configured to set the CPU clock to 25 MHz and bus clock to 12.5 MHz using the Internal 1 MHz clock signal.

The Internal 1 MHz reference clock is selected as the source clock for PLL (CPMUREFDIV_REFDIV = 0, CPMUREFDIV_REFFRQ = 0).

The PLL VCOCLK frequency is set to 50 MHz (CPMUSYNR_SYNDIV = 24):

$$VCOCLK = (2 \times (SYNDIV + 1)) \times 1MHz \quad \textbf{\textit{Eqn. 1}}$$

The VCOCLK signal frequency is then divided by 2 (CPMUPOSTDIV_POSTDIV = 1) and this is used as the 25 MHz core clock signal ECLKX2 and the 12.5 MHz bus clock signal ECLK.

2.2 Pulse Width Modulator with Fault Protection

The Pulse Width Modulator with Fault Protection (PMF) module is configured to generate an edge-aligned (PMFCFG0_EDGEx = 1) PWM with a frequency of 20 kHz (PMFMODA = 1250). In order to protect the MOSFET devices in the same leg of the inverter, deadtime is set to approximately 0.5 µs (PMFDTMA = 13). PWM generator A runs as the master and generates the Reload Signal as a synchronization signal for the other submodules (PMFCFG2_REV[0:1] = 1). The reload signal is generated at every PWM opportunity (PMFFQCA = 0). Pair A, Pair B, and Pair C PWMs are synchronized to PWM generator A (PMFCFG0_MTG = 0). All three pairs use value register zero (PMFCFG3_VLMODE = 1) to set the duty cycle. The PWM generator is restarted with every commutation event (PMFENCA_RSTRTA = 1).

To achieve a unipolar PWM pattern, the first phase is in complementary PWM mode while the second phase is grounded by the software control mode (PMFOUTC) and the third phase stays unpowered, by masking the output (PMFCFG2[MSK5:MSK0]) as shown in Figure 1. The registers (PMFOUTC, PMFCFG2[MSK5:MSK0]) are used to control the unipolar PWM pattern. The double buffered registers are swapped at a commutation event. The PWM pulse width PMFVAL registers are also double buffered and are swapped when GLDOK is set and the PWM reload signal occurs. The GLDOK is an external signal generated by the PTU module. The GLDOK is enabled at the PWM module (PMFENCA_GLDOKA = 1)

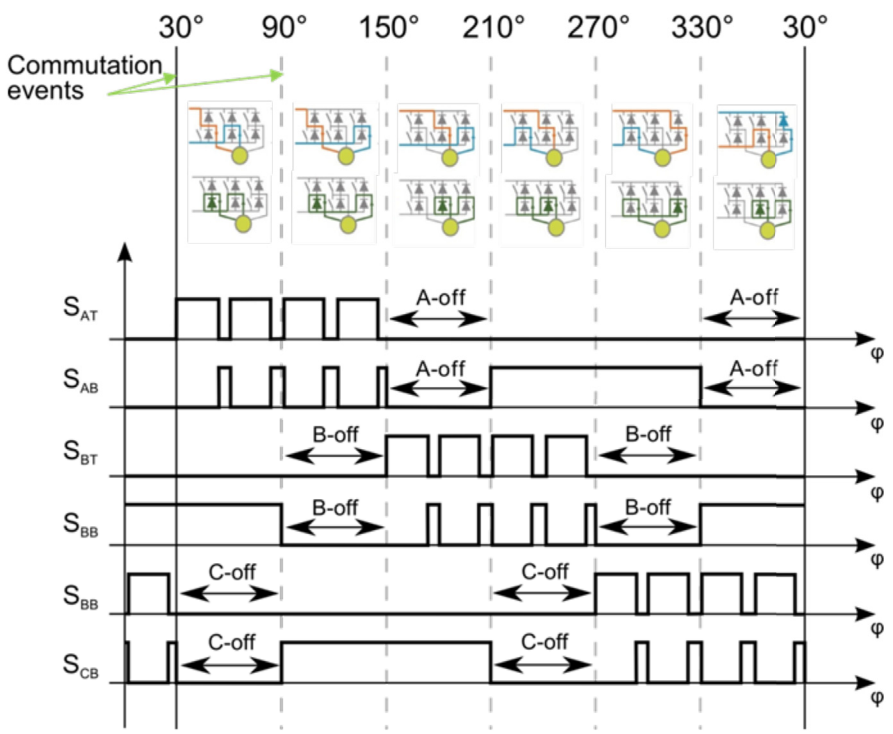


Figure 1. BLDC motor unipolar PWM switching

For example, in the first cycle, Phase A is powered by the complementary PWM signal, while the bottom transistor of Phase B is grounded (PMFOUTC = 0x0C) and Phase C is unpowered (PMFCFG2[MSK5:MSK0] = 0x34). After the commutation event at 90¬× electrical degrees, Phase A is still powered by the complementary PWM signal, Phase B is now unpowered (PMFCFG2[MSK5:MSK0] = 0x1C) and Phase C becomes grounded instead (PMFOUTC = 0x30).

2.3 Programmable Trigger Unit

The Programmable Trigger Unit (PTU) is intended to completely avoid CPU involvement in the time acquisitions of state variables during the control cycle.

The PTU module consists of two trigger generators (TG). For each TG, a separate enable bit is available, so that both TGs can be enabled independently. Trigger generator 0 is connected to ADC0, and trigger generator 1 is connected to ADC1. The trigger generation of the PTU module is synchronized to the incoming reload event. This reload event resets and restarts the internal time base counter and makes sure that the first trigger value from the actual trigger list is loaded. Furthermore, the corresponding ADC is informed that a new control cycle has started.

If the counter value matches the current trigger value, then a trigger event is generated. In this way, the reload event is delayed by the number of bus clock cycles defined by the current trigger value. All acquisition time values are stored inside the global memory map, basically, inside the system memory as a three dimensional array of integers (PTUTriggerEventList[][][]). The exact location of the acquisition time values (PTUTriggerEventList[][][]) in the system memory is given by the user.

The application is using PTU TG0 only. The generator uses only one list to load the trigger values from the memory.

The PTU module generates the LDOK signal used to inform other modules that the double buffered registers were updated by software.

2.4 Analog-to-Digital Converter

The MC9S12ZVML128 includes two independent Analog-to-Digital Converters (ADCs). Both ADCs are n-channel multiplexed input successive approximation analog to digital converters. The List Based Architecture (LBA) provides a flexible conversion sequence definition, as well as flexible oversampling. Both ADC conversion command lists are stored inside the global memory map, basically, inside the system memory as two-dimensional arrays of bytes (ADC0CommandList[][], ADC1CommandList[][]). The exact location of the ADC conversion commands in the system memory is given by user. The same strategy is used for the ADC Results. The Conversion results are stored in an array of shorts (ADC0ResultList[], ADC1ResultList[]) located in system memory. The application is using ADC0 only.

The ADC conversion clocks are set to be 6.25 MHz (ADC0TIM = 0). The results are stored in memory as 12-bit (ADC0FMT_SRES = 4) left-justified data (ADC0FMT_DJM = 0).

Conversion flow of ADC0 is controlled by internal signals (generated by the PTU) and by the DataBus (ADC0CTL_0_ACC_CFG = 3). The results are stored in system memory even if commutation occurs when conversion is ongoing (ADC0CTL_0_STR_SEQA = 1).

The ADC0 schedules the end of list interrupt (ADC0CONIE_1_EOL_IE = 1) to calculate the motor phase current and DC-bus voltage.

2.5 Timer Module

The Timer Module (TIM) is a basic scalable timer that consists of a 16-bit, software-programmable counter driven by a flexible programmable prescaler. The BLDC Hall sensor based algorithm employs two timer channels in output compare mode (TIM0TIOS_IOS0 = 1; TIM0TIOS_IOS3 = 1) and one timer channel in input capture mode (TIM0TIOS_IOS1 = 0).

Timer Channel 0 serves to identify the commutation event. The output compare signal is internally routed to the PMF module as an async_event in order to perform commutation of the PWM pairs. The timer is configured to no action on channel output (TIM0TCTL2_OL0 = 0; TIM0TCTL2_OM0 = 0).

Timer Channel 1 operates as a commutation interrupt source. The Hall interface signals (PT1, PT2, PT3) are XORed in the PIM module and the signal is routed (MODRR2_T0IC1RR = 1) in Timer Channel 1. During Timer Channel 1 interrupt execution, the unipolar PWM pattern is changed and the commutation is forced (TIM0CFORC_FOC0 = 1) by a Timer Channel 0 async_event.

Timer Channel 3 is employed to control the torque of the motor by software tasks. Timer Channel 3 schedules periodic interrupts (TIM0TIE_C3I = 1) with a period of 1 ms (TIM0TC3 = 98).

The prescaler of timers is equal to 128 (TIM0TSCR2_PR = 7) and all output compare pins of the timers are disconnected (TIM0OCPD = 0xFF).

2.6 Gate Drive Unit

The Gate Drive Unit (GDU) is a Field Effect Transistor (FET) predriver designed for 3-phase motor control applications.

The boost option is enabled (GDUE_GBOE = 1) to maintain the gate to source voltage in case, the power supply voltage is below 10 V. The boost circuit is running at a 1 MHz switching frequency.

The charge pump is used to maintain the high-side driver gate source voltage VGS when PWM is running at a 100% duty cycle. The clock for the charge pump is set to be f_bus/32 (GDUCLK2_GCPCD = 2).

The GDU integrates three desaturation comparators for the low-side FET pre-drivers and three desaturation comparators for the high-side FET pre-drivers. The desaturation level is set to be 1.35 V (GDUDSLVL = 0x77) for both low-side and high-side FETs. The blanking time is set to be approximately 7 μs (GDUCTR = 0x13).

Internal current sense amplifier 0 (GDUE_GCSE0 = 1) is used to measure motor phase current. The output of the current sense amplifier 0 is routed internally to ADC0 AN0.

2.7 Port Integration Module

The Port Integration Module (PIM) is used to configure the Hall sensor port, SCI signals, and the user LEDs.

The Hall sensor port pins PT1, PT2, PT3 pull-ups are enabled (PERT = 0x0E). As the Hall interface power supply voltage source, PP0 is selected and enabled (PTP_PTP0 = 1, DDRP_DDRP0 = 1). The Module routing register 2 (MODRR2_T0IC1RR = 1) routes the Hall interface XORed signals in TIM channel 1 to proceed period measurement and BLDC motor commutation.

The SCI0 signals are routed to PS2 and PS3 pins (MODRR0_SCI1RR = 1). The SCI is used to control the application using FreeMASTER tool Section 5., “MTRCKTSBNZVM128HFS Fact Sheet, www.freescale.com/AutoMCDevKits”.

2.8 Serial Communication Interface

The Serial Communication Interface (SCI) is used as the FreeMASTER application communication channel. The communication speed is set to 9600 Bd (SCI1BD = 1302).

2.9 Interrupt

The Interrupt module is used to set the interrupts priority as follows, starting with the highest priority:

1. The ADC0 conversion done - motor current and DC-bus voltage samples load
2. The TIM channel 1 - commutation interrupt initiated by a Hall sensor interface signal change
3. The TIM channel 3 - 1 ms speed and current loop calculation period
4. The ADC0 error interrupt routine

3 Hall sensor-based BLDC motor control application

This section describes the BLDC Hall sensor-based motor control application flow. The 3-phase Sensorless BLDC Motor Control Development Kit MTRCKTSBNZVM128 Section 3., "MTRCKTSBNZVM128 Development Kit, www.freescale.com/AutoMCDevKits" is used as the application hardware.

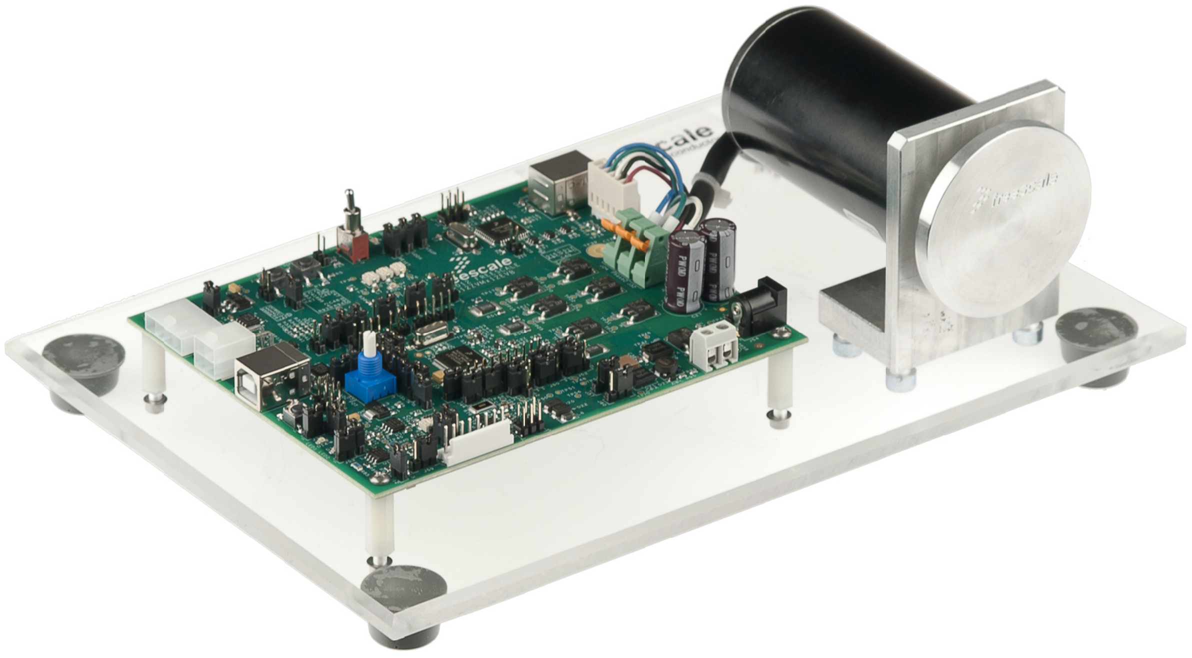


Figure 2. 3-phase BLDC Motor Control Development Kit MTRCKTSBNZVM128

To get the application running, the MTRCKTSBNZVM128 Development Kit jumper setting needs to be changed and the new software needs to be downloaded to the S12ZVM MCU. Detailed info can be found in Quick Start Guide Section 4., "MTRCKTSBNZVM128HQSG Quick Start Guide, www.freescale.com/AutoMCDevKits".

The application can be manually controlled by the onboard switches and monitored by the onboard LEDs:

- SW1 - push to start the motor and to increase the motor rotation speed.
- SW2 - push to decrease the motor rotation speed. When the minimal speed is reached, a following SW2 push stops the motor.
- LED1 - while the motor is running, the LED is ON
- LED2 - when a fault is detected, the LED is ON

Advanced application control and visualization is achieved by the FreeMASTER tool. For more info on tool installation and application control, see the Quick Start Guide Section 4., "MTRCKTSBNZVM128HQSG Quick Start Guide, www.freescale.com/AutoMCDevKits".

3.1 Building blocks

An application block diagram is depicted in Figure 3. Based on the Hall sensor's signal value, the actual PWM pattern is generated to drive the BLDC motor. The PWM duty cycle is driven by the speed and current PI controllers. The actual motor speed is measured using a Hall sensor signal interface and its value is compared with the required speed. The motor phase current is compared with the current limit. Based on the PI controllers calculation results, the PWM duty cycle is controlled.

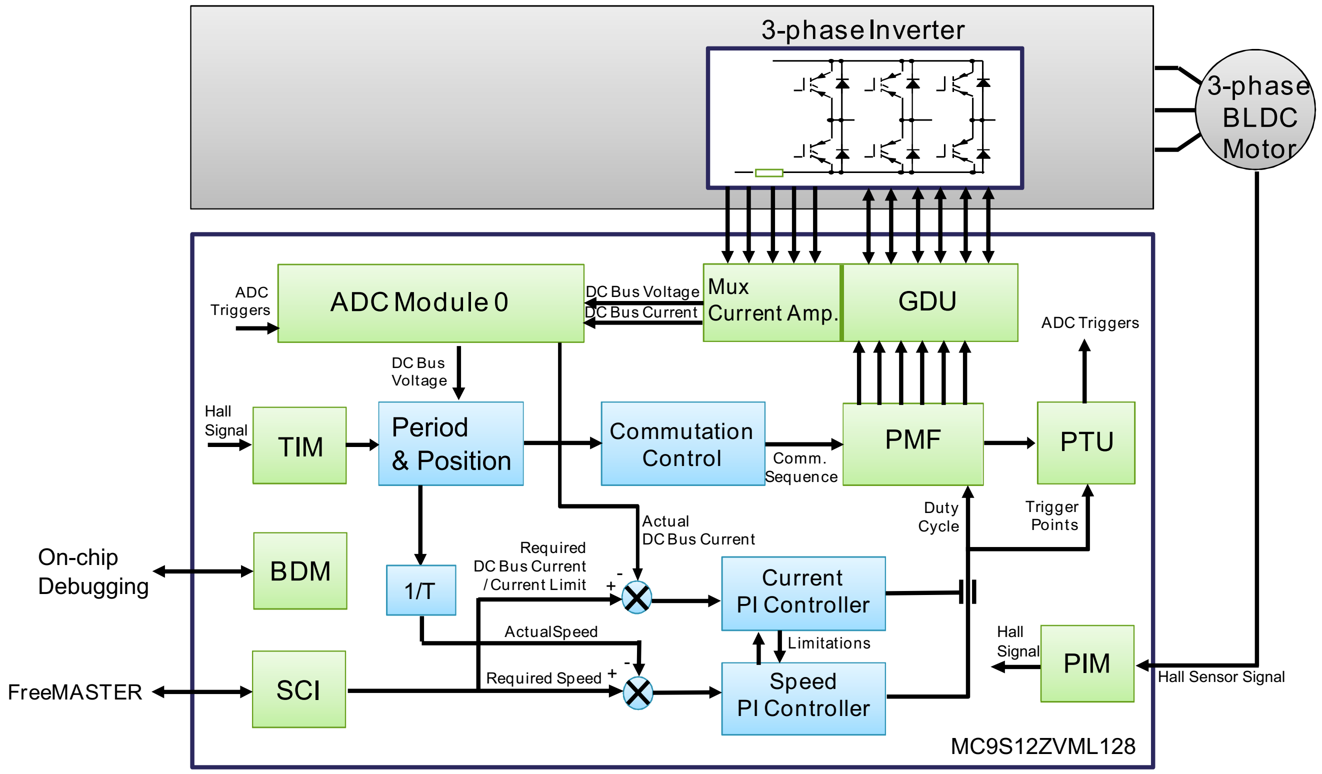


Figure 3. Application block diagram

3.2 State machine

Once the S12ZVM module's initialization is done, see Section 2, "Configuration of MC9S12ZVML128 modules", the application state machine is initiated for the application flow control.

An application state machine is implemented using a function pointer array and can reach four different states described in the sections below. Figure 4 shows the state diagram.

3.2.1 Init

All the PWM MOSFETs are switched OFF and the PWM duty cycle is set to 0%. Onboard LED1 is turned OFF. The state machine transits to the STOP state.

3.2.2 Stop

In the STOP state, the DC-bus current offset calibration proceeds and the motor is ready to start. Transition to the RUN state is performed once the motor run is required.

3.2.3 Run

The motor is started-up with an initial duty cycle. The onboard LED1 is turned ON. The motor commutation is performed on a Hall interface signal value change, see Section 3.4, “BLDC motor commutation”. The motor speed control and motor phase current limitation are performed as described in Section 3.5, “BLDC motor speed calculation and control and Section 3.6, “BLDC motor phase current control”. If the motor stop is required, the state machine is switched to the INIT state. The motor rotation speed and phase current calculation proceeds until the motor does finally stop.

3.2.4 Fault

An application detects:

- DC-bus undervoltage (VSUP < 6 V)
- DC-bus overvoltage (VSUP > 16 V)
- Motor stall

Once any of the faults are detected, the state machine enters the FAULT state. The PWM MOSFETs are switched OFF, onboard LED1 is turned OFF and onboard LED2 is turned ON. To exit the FAULT state, all fault sources must be cleared. Then, the onboard LED2 is turned OFF and the state machine enters the INIT state.

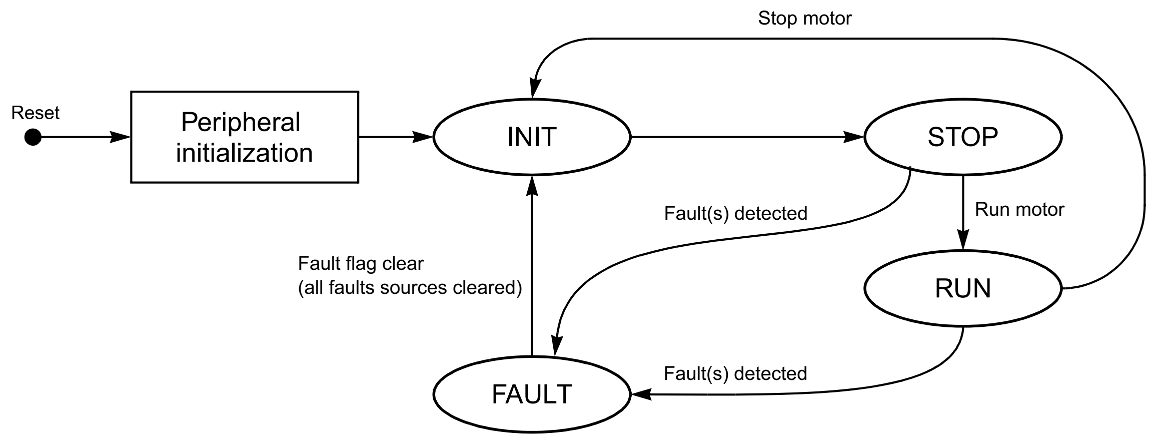


Figure 4. Application state diagram

3.3 Interrupts

An application flow is controlled by interrupts as follows.

3.3.1 ADC0 conversion interrupt

When the PMF modulo counter PMFMODA overflows, the reload event resets and restarts the PTU internal time base counter, see Figure 5. Once PTU counter value matches the trigger value, the ADC0 conversion is initiated. On the first PTU trigger event, the motor current is sampled. The second PTU trigger initiates the DC-bus voltage conversion. When the ADC0 loads the end-of-sequence command, the ADC0 conversion interrupt is called and the sampled values are stored in memory for further processing.

In the case of a 20 kHz PWM, the ADC0 conversion interrupt arises every 50 µs.

3.3.2 TIM channel 1 interrupt

The TIM channel 1 commutation interrupt arises on a Hall interface signal change. The module's configuration is described in Section 2.7, "Port Integration Module".

During an interrupt processing, the period between recent Hall interface edges is measured and stored in memory for further processing during the motor control speed loop processing.

Based on the actual Hall interface pattern, the registers (PMFOUTC, PMFCFG2[MSK5:MSK0]) are updated and a TIM channel 0 async_event is generated to force the new PWM pattern to the MOSFET outputs, see Figure 5.

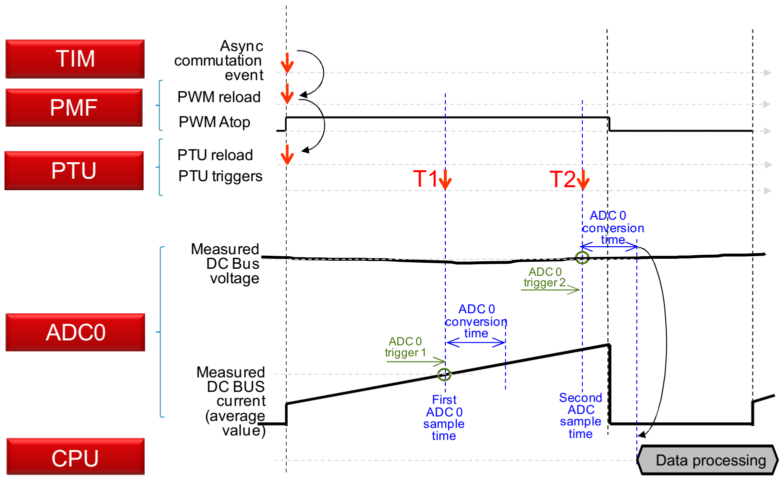


Figure 5. Commutation event flow

3.3.3 TIM channel 3 interrupt

The TIM channel 3 interrupt arises every 1 ms. During an interrupt execution, the actual motor speed and motor phase current are calculated. The actual motor speed is compared with the required speed. Based on the result from the speed PI controller, the actual PWM duty cycle and PTU trigger values are updated.

In case the actual motor phase current reaches the maximal defined value, the current PI controller takes control over the PWM duty cycle. The value of the motor phase current is maintained at the maximal defined level. Out of the application reset, the maximal level is set to 4A.

3.3.4 ADC0 error interrupt

The ADC0 error interrupt routine serves as the ADC0 user feedback.

3.4 BLDC motor commutation

The motor Hall sensors are used to sense the actual rotor position. The Hall sensor interface signal pattern changes six times per one motor electrical revolution. The number of electrical revolutions per one mechanical revolution is equal to the number of motor pole-pairs.

The commutation process is an interrupt driven process, see Section 3.3.2, “TIM channel 1 interrupt”. The Hall sensor outputs are connected to S12ZVM inputs PT1 (HA), PT2 (HB), PT3 (HC). The Hall interface signals are internally XORed and routed to TIM channel 1. The TIM channel 1 interrupt is generated on each edge of a Hall interface signal. Based on the actual Hall interface signal value and the required rotation direction, the proper commutation vector is applied (see Table 2 or Table 3).

3.4.1 Hall sensor pattern

The Hall sensor pattern is different for each BLDC motor commutation sector, see Figure 6.

The following steps describe how to define the Hall sensor pattern:

1. Mark motor phases A, B, C and the corresponding Hall sensor outputs as HA, HB, HC.
2. Set the power supply current limit to 20% of the nominal motor current.
3. Set the power supply voltage to 0 V.
4. Connect the motor phases to the power supply following Table 1.
5. Increase power supply voltage to reach the current limit.
6. Read the Hall pattern.
7. Repeat from point 3 for each sector.

The Table 1 displays the Hall sensor interface pattern measured for the BLDC motor (Linix 45ZWN24-90) used in the MTRCKTSBNZVM128 development kit. Corresponding sectors are shown in Figure 6.

Table 1. Linix 45ZWN24-90 Hall sensor pattern

Powered phase			Sector	Hall sensor output		
Phase A	Phase B	Phase C		Hall C	Hall B	Hall A
$+V_{DBC}$	$-V_{DBC}$	$-V_{DBC}$	I	0	1	0
$+V_{DBC}$	$+V_{DBC}$	$-V_{DBC}$	II	1	1	0
$-V_{DBC}$	$+V_{DBC}$	$-V_{DBC}$	III	1	0	0
$-V_{DBC}$	$+V_{DBC}$	$+V_{DBC}$	IV	1	0	1
$-V_{DBC}$	$-V_{DBC}$	$+V_{DBC}$	V	0	0	1
$+V_{DBC}$	$-V_{DBC}$	$+V_{DBC}$	VI	0	1	1

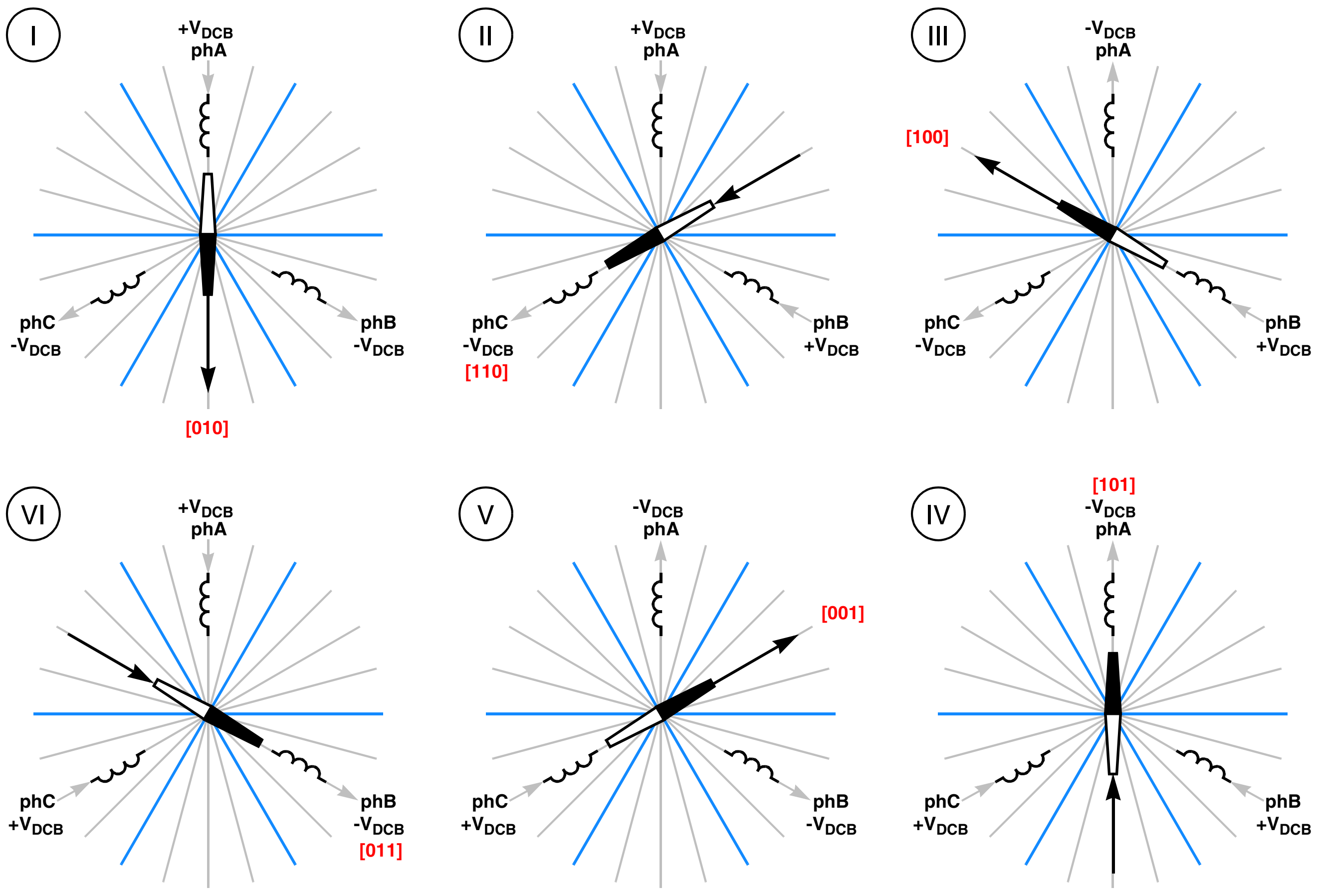


Figure 6. Commutation sector definition

3.4.2 Commutation vectors

The commutation vectors are defined in order to reach the BLDC motor maximal torque. The BLDC motor maximal torque is achieved when the angle between the stator and rotor flux is 90 degrees. The six commutation sector motor flux angles vary from 120° to 60° for each defined Hall pattern, see Figure 7. The Table 2 and Table 3 are defined based on Figure 7, which displays the motor phase voltage polarity to achieve the maximal torque for each commutation sector.

The commutation vectors are assigned based on the rotation direction. The Hall sensor patterns 0 [000] and 7 [111] indicate a Hall sensor interface fault.

Table 2. Linix 45ZWN24-90 clockwise direction commutation sequence

Commutation vector			Vector	Hall sensor pattern definition			Decimal result
Phase A	Phase B	Phase C		Hall C	Hall B	Hall A	
$+V_{DBC}$	$-V_{DBC}$	NC	0	0	0	1	1
$+V_{DBC}$	NC	$-V_{DBC}$	1	0	1	1	3
NC	$+V_{DBC}$	$-V_{DBC}$	2	0	1	0	2
$-V_{DBC}$	$+V_{DBC}$	NC	3	1	1	0	6
$-V_{DBC}$	NC	$+V_{DBC}$	4	1	0	0	4
NC	$-V_{DBC}$	$+V_{DBC}$	5	1	0	1	5

Table 3. Linix 45ZWN24-90 counterclockwise direction commutation sequence

Commutation vector			Vector	Hall sensor pattern definition			Decimal result
Phase A	Phase B	Phase C		Hall C	Hall B	Hall A	
$-V_{DBC}$	$+V_{DBC}$	NC	3	0	0	1	1
NC	$+V_{DBC}$	$-V_{DBC}$	2	1	0	1	5
$+V_{DBC}$	NC	$-V_{DBC}$	1	1	0	0	4
$+V_{DBC}$	$-V_{DBC}$	NC	0	1	1	0	6
NC	$-V_{DBC}$	$+V_{DBC}$	5	0	1	0	2
$-V_{DBC}$	NC	$+V_{DBC}$	4	0	1	1	3

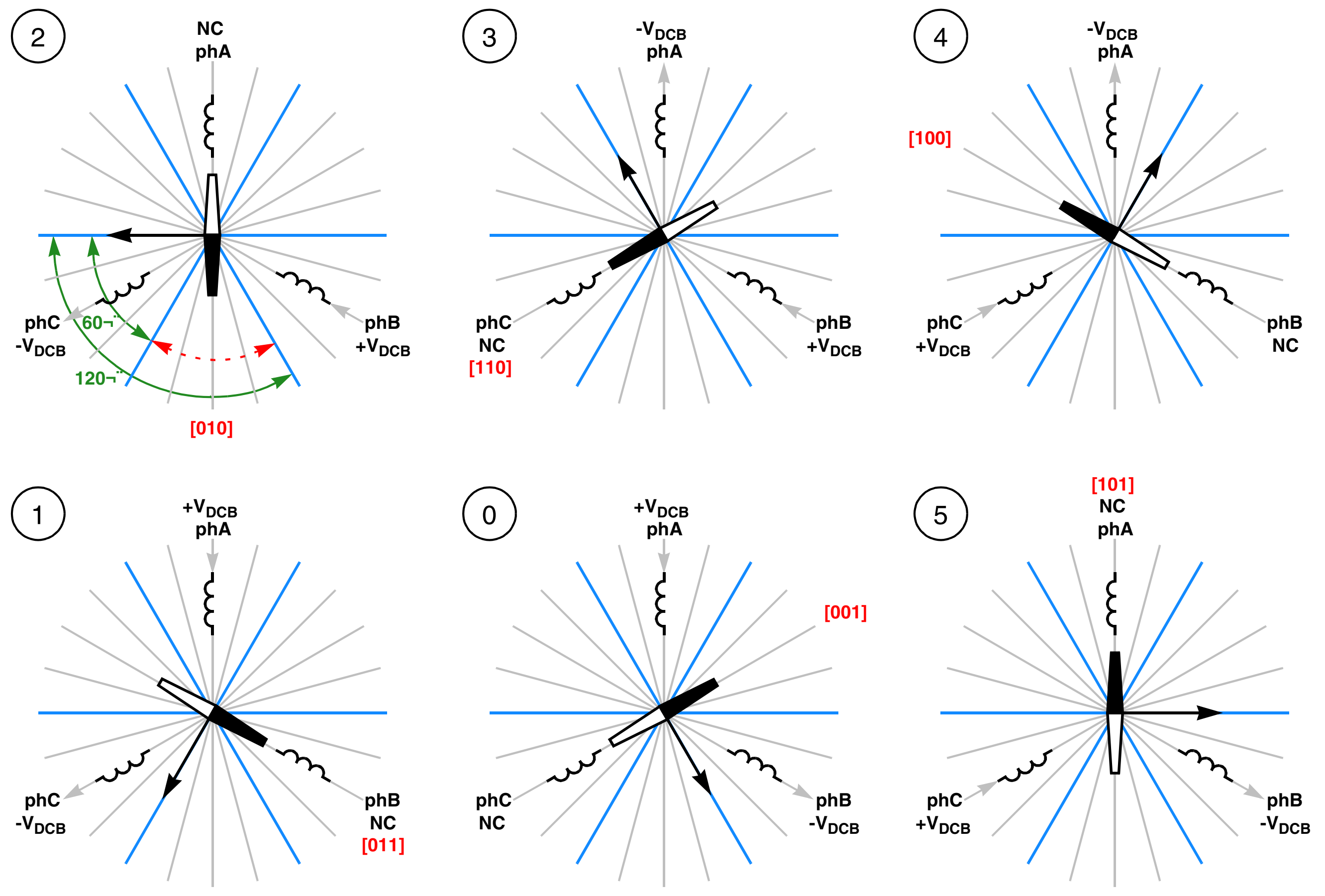


Figure 7. Commutation vector definition for clockwise rotation direction

3.5 BLDC motor speed calculation and control

The motor speed is controlled every 1 ms, see Section 3.3.3, “TIM channel 3 interrupt”. Where the motor phase current is below the maximal limit, the PWM duty cycle is controlled by the speed PI controller. The speed PI controller input parameter is the difference between the required and actual speeds. The calculated PWM duty cycle is limited in the range of 0.04 to 0.96 of PMF modulo counter (PMFMODA = 1250).

The actual motor speed is calculated before the speed PI controller is called. The Hall interface interrupt, see Section 3.3.2, “TIM channel 1 interrupt”, loads the commutation period in TIM counter ticks. The period of one electrical revolution in seconds is given by Equation 2.

$$T_{elrev} = \frac{T \times N}{f_{TIM}} \qquad \textit{Eqn. 2}$$

where:

- T—Commutation period in TIM counter ticks

- N—Number of commutations in one electrical revolution
- f_{TIM}—TIM counter clock frequency
- T_{elrev}—Calculated period of one electrical revolution in seconds

To calculate the period of one mechanical revolution in seconds, the electrical revolution period needs to be multiplied by the number of motor pole-pairs:

$$T_{mechrev} = T_{elrev} \times p = \frac{T \times N \times p}{f_{TIM}} \qquad \textbf{\textit{Eqn. 3}}$$

where:

- p—Number of pole-pairs

Mechanical speed in revolutions per minute can be calculated by Equation 4.

$$RPM = \frac{60}{T_{mechrev}} = \frac{60 \times f_{TIM}}{T \times N \times p} \qquad \textbf{\textit{Eqn. 4}}$$

The application software calculates actual motor speed from the period of one electrical revolution which equals the period of six consecutive motor commutations. The mechanical speed constant Equation 5 can be obtained by grouping the constant parameters from the Equation 4. In the case of a TIM single tick period 10.24 µs

$$c = \frac{60}{T_{TIM} \times p} = \frac{60}{10,24us \times 4} = 1464844 \qquad \textbf{\textit{Eqn. 5}}$$

The speed of the 4 pole-pair Linix 45ZWN24-90 motor can be calculated by Equation 6.

$$RPM = \frac{c}{6 \times T} = \frac{1464844}{6 \times T} \qquad \textbf{\textit{Eqn. 6}}$$

3.6 BLDC motor phase current control

In order to protect the motor and power electronics, the motor phase current control is implemented. The motor current is controlled every 1 ms, see Section 3.3.3, “TIM channel 3 interrupt” by the current PI controller. The current PI controller input parameter is the difference between the maximal motor phase current and the average motor phase current sampled during six consecutive commutations. The calculated PWM duty cycle is limited in the range of 0.04 to 0.96 of the PMF modulo counter (PMFMODA = 1250).

If the current limiting PI controller output value is lower than the speed PI controller output value, the PWM duty cycle is controlled by the current limiting PI controller, see Figure 8.

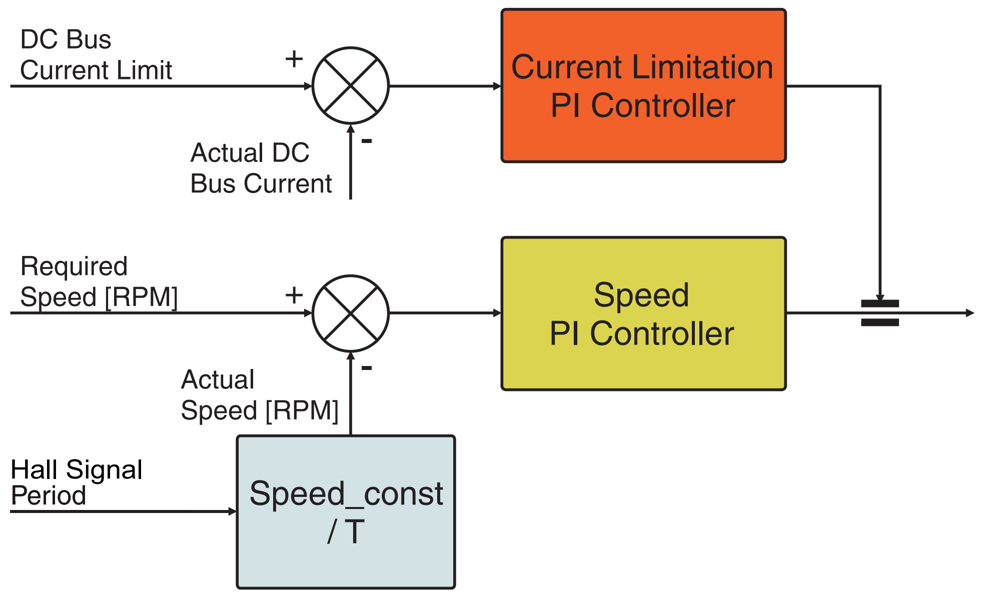


Figure 8. Speed control with current limitation

3.7 Math and Motor Control Library

The Freescale Math and Motor Control Library for the MC9S12ZVML128 microcontroller Section 7., "Math and Motor Control Library, www.freescale.com/autoMClib" is used to properly calculate basic and advanced mathematical operations. The Automotive Math and Motor Control Library Set is a precompiled software library containing the building blocks for a wide range of motor control and general mathematical applications. It consists of three groups of functions:

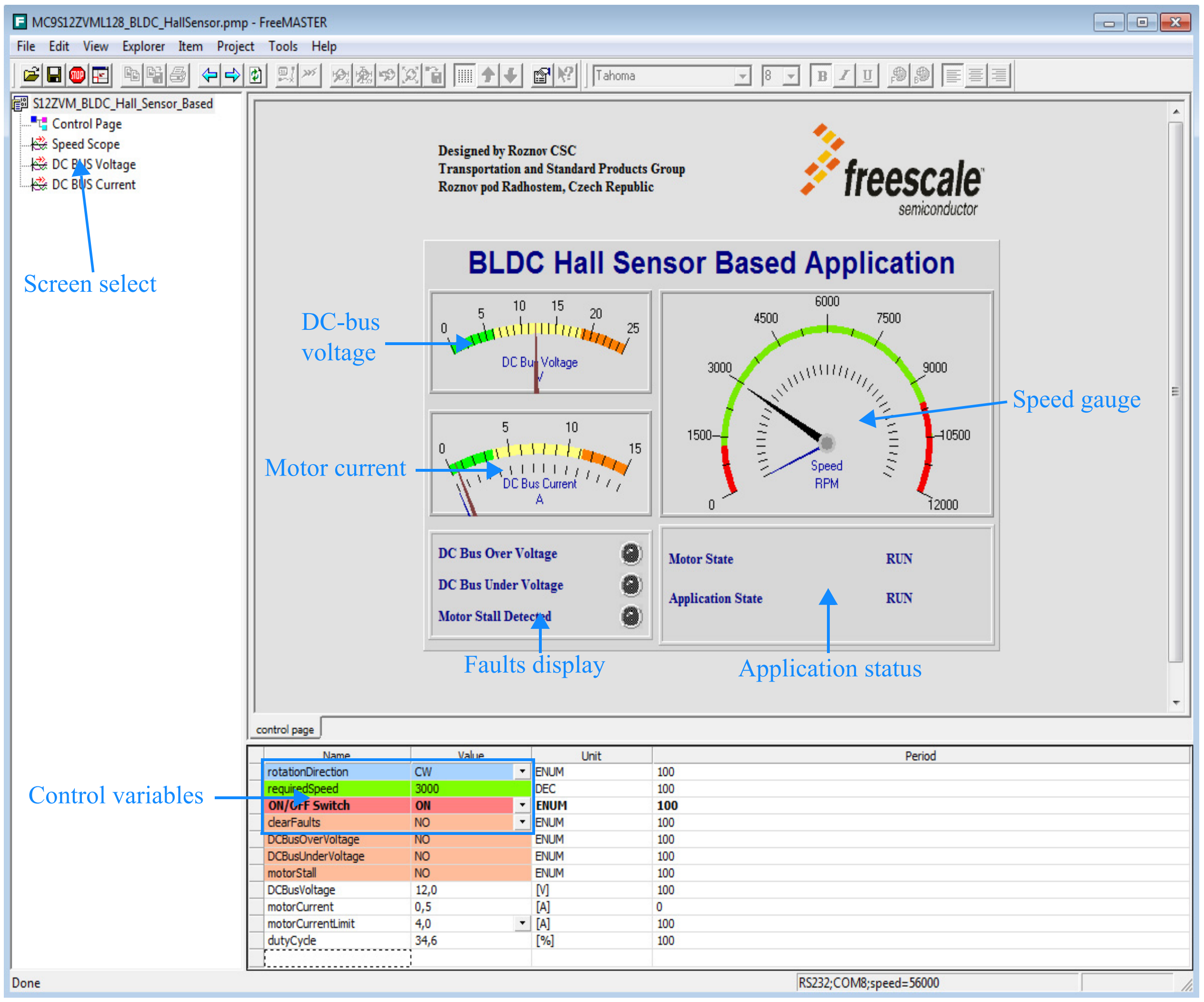
- **General trigonometric and basic functions** (GFLIB) with basic math, trigonometric, look-up table, and controller functions.
- **General motor control functions** (GMCLIB) with space vector modulation, transformation, and motor control specific functions.
- **General digital filters functions** (GDFLIB) with digital filter functions for signal controlling.

3.8 FreeMASTER tool

An advanced application control and visualization is achieved by the FreeMASTER tool Section 6., "FreeMASTER Run-Time Debugging Tool, www.freescale.com/freemaster". The FreeMASTER is a real-time debug monitor and data visualization tool. It can be used for an application development as well as application data management. The FreeMASTER tool supports non-intrusive monitoring of variables on a running system. The multiple variables changing over time can be displayed oscilloscope-like, or viewed in text form.

The FreeMASTER control page runs on a PC. An application FreeMASTER control page is depicted in Figure 9. The PC and the MTRCKTSBNZVM128 development kit Section 2., "MC9S12ZVML128

Evaluation Board User Manual, www.freescale.com/AutoMCDevKits" are connected via USB. For more information, see the Quick Start Guide Section 4., "MTRCKTSBNZVM128HQSG Quick Start Guide, www.freescale.com/AutoMCDevKits".



Name	Value	Unit	Period
rotationDirection	CW	ENUM	100
requiredSpeed	3000	DEC	100
ON/OFF Switch	**ON**	**ENUM**	**100**
clearFaults	NO	ENUM	100
DCBusOverVoltage	NO	ENUM	100
DCBusUnderVoltage	NO	ENUM	100
motorStall	NO	ENUM	100
DCBusVoltage	12,0	[V]	100
motorCurrent	0,5	[A]	0
motorCurrentLimit	4,0	[A]	100
dutyCycle	34,6	[%]	100

Figure 9. FreeMASTER control page

4 References

1. MC9S12ZVM Family Reference Manual, www.freescale.com
2. MC9S12ZVML128 Evaluation Board User Manual, www.freescale.com/AutoMCDevKits
3. MTRCKTSBNZVM128 Development Kit, www.freescale.com/AutoMCDevKits
4. MTRCKTSBNZVM128HQSG Quick Start Guide, www.freescale.com/AutoMCDevKits
5. MTRCKTSBNZVM128HFS Fact Sheet, www.freescale.com/AutoMCDevKits
6. FreeMASTER Run-Time Debugging Tool, www.freescale.com/freemaster
7. Math and Motor Control Library, www.freescale.com/autoMClib

5 Acronyms

ADC	Analog to Digital Converter
BDM	Background Debug Module
BLDC	Brushless DC
DC	Direct Current
ISR	Interrupt Service Routine
LIN	Local Interconnect Network
MCU	Microcontroller Unit
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
PI	Proportional Integral
PC	Personal Computer
RPM	Revolutions per Minute
PWM	Pulse Width Modulation
USB	Universal Serial Bus

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Number: AN4718
Rev. 0
04/2013