



Joe Ji and Janz Bai

## ABSTRACT

Low voltage BLDC motors are popular in appliances due to the safe high efficiency and excellent features. This application note details a low voltage motor control system built with MSPM0C1104 MCU and MCF8315D BLDC motor controller. With the two devices, the design can function with a code-free sensorless FOC design and high system-level flexibility. This application note aims to accelerate customer software development and help products launch to the market. The hardware setup with TI EVMs and software structures, system test that shows the test setup, and control result of the demo system are presented in this application note. Users can port this code easily to other MSPM0 devices using the system configuration tool (SYSCONFIG). Demo code can be accessed from [E2E](#). For hardware design, refer to [How to Optimize MCF8315 Hardware Design in Fan Application application note](#).

## Table of Contents

<b>1 Introduction</b> .....	2
1.1 System Introduction.....	2
1.2 MSPM0C110x Introduction.....	2
1.3 MCF8315D Introduction.....	4
<b>2 Hardware Setup</b> .....	5
2.1 LP-MSPM0C1104 Hardware Setup .....	5
2.2 MCF8315PWPEVM Hardware Setup.....	5
<b>3 Software Structure</b> .....	6
3.1 Software Function and Flow Diagram.....	6
3.2 Project File Structure.....	7
3.3 Configure and Control MCF8315.....	7
<b>4 System Test</b> .....	10
4.1 Test Setup .....	10
4.2 System Value Monitor.....	11
<b>5 Summary</b> .....	11
<b>6 References</b> .....	11

## List of Figures

Figure 1-1. System Block Diagram of LV BLDC Control.....	2
Figure 1-2. MSPM0C110x Functional Block Diagram.....	3
Figure 1-3. MCF8315D (PWP) Functional Block Diagram.....	4
Figure 2-1. LP-MSPM0C1104 and MCF8315PWPEVM Hardware Connection.....	5
Figure 3-1. MSPM0 Software Diagram.....	6
Figure 3-2. MCF8315D I2C Write Transaction Sequence.....	8
Figure 3-3. MCF8315D I2C Read Transaction Sequence.....	8
Figure 4-1. Test Environment Setup.....	10
Figure 4-2. Emulator Test Connection with MSPM0C1104.....	10
Figure 4-3. MCU Variables Monitor Window.....	11

## List of Tables

Table 2-1. Signal Connection between LP-MSPM0C1104 and MCF8315PWPEVM.....	5
Table 3-1. Project File Structure.....	7
Table 3-2. I2C Data Word Format.....	7
Table 3-3. 24-bit Control Word Format.....	7

## Trademarks

Arm® and Cortex® are registered trademarks of Arm Limited. All trademarks are the property of their respective owners.

## 1 Introduction

### 1.1 System Introduction

The low voltage DC power supply structure helps the appliance system design to reach higher efficiency and safety level. To make the system design easier and practical, TI summarized decades of BLDC control experiences into MCF8315D to help customers adapt to the BLDC control code-freely.

This application note presents a highly flexible LV BLDC control system with only two integrated circuits (ICs) as shown in [Figure 1-1](#), MCF8315D integrates all the necessary components to realize a code-free LV BLDC control; MSPM0C1104 is used to initialize, control, or monitor MCF8315D to fit a variety of system needs.

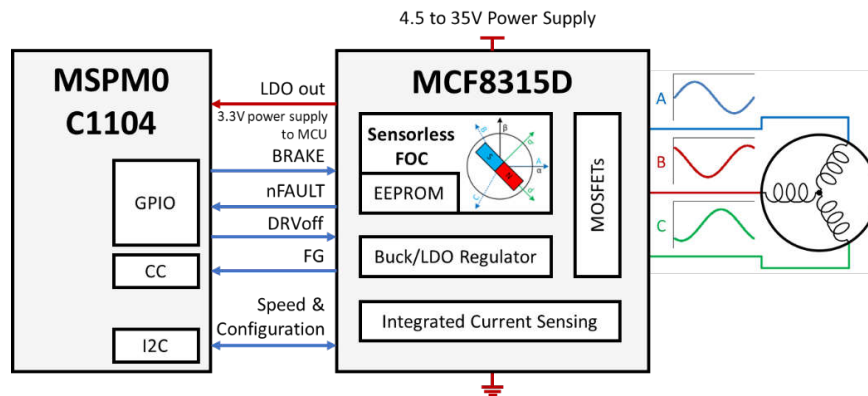


Figure 1-1. System Block Diagram of LV BLDC Control

### 1.2 MSPM0C110x Introduction

MSPM0C110x microcontrollers (MCUs) are part of the MSP highly-integrated ultra-low-power 32-bit MCU family based on the enhanced Arm® Cortex®-M0+ core platform operating at up to 24MHz frequency. These MCUs offer high-performance analog peripheral integration.

The MSPM0C110x devices provide up to 16KB embedded flash program memory with 1KB SRAM. This device integrates a variety of high-performance analog peripherals such as one 12-bit 1.5Msps ADC with VDD as the voltage reference, and an on-chip temperature sensor. The device also offers intelligent digital peripherals such as one 16-bit advanced timer, two 16-bit general purpose timer, one windowed watchdog timer, and a variety of communication peripherals including one UART, one SPI, and one I2C. These communication peripherals offer protocol support for LIN, IrDA, DALI, Manchester, smart card, SMBus, and PMBus. Refer to the [MSPM0C110x, MSPS003 Mixed-Signal Microcontrollers data sheet](#) for more information.

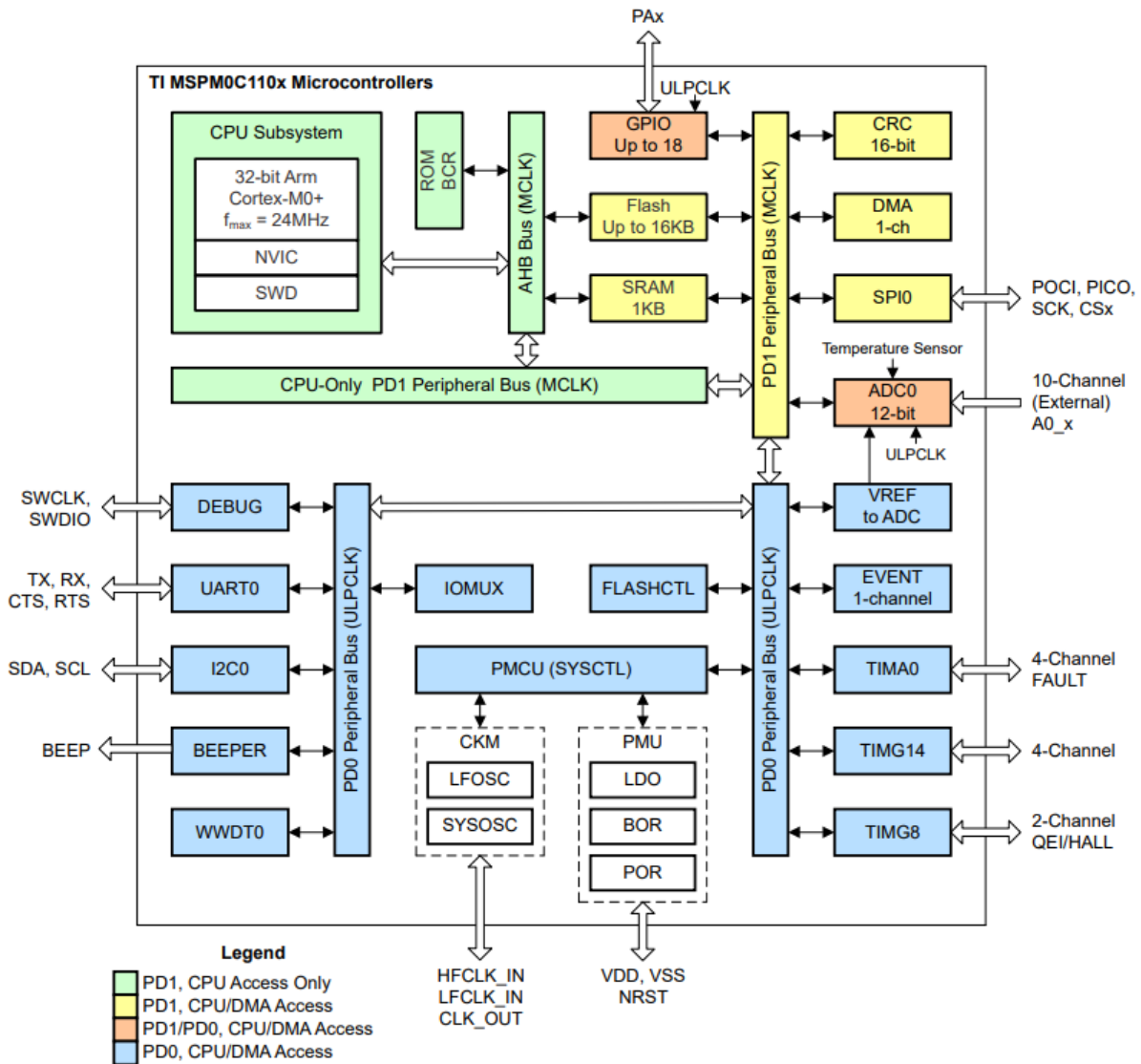


Figure 1-2. MSPM0C110x Functional Block Diagram

### 1.3 MCF8315D Introduction

The MCF8315D provides a single-chip, code-free sensorless FOC device for customers driving speed control on 12V to 24V brushless-DC motors (BLDC) or Permanent Magnet Synchronous motor (PMSM) up to 4A peak current. The MCF8315D integrates three half bridges with 40V absolute maximum capability and a very low RDS(ON). The integrated power management circuits including a voltage-adjustable buck regulator (3.3V, 5V, 170mA) and LDO (3.3V, 20mA) can be used to power external circuits. A large number of protection features are integrated into the MCF8315D, such as protection of the device, motor, and system against fault events. Refer to the [MCF8315D Sensorless Field Oriented Control \(FOC\) Integrated FET BLDC Driver](#) for more information.

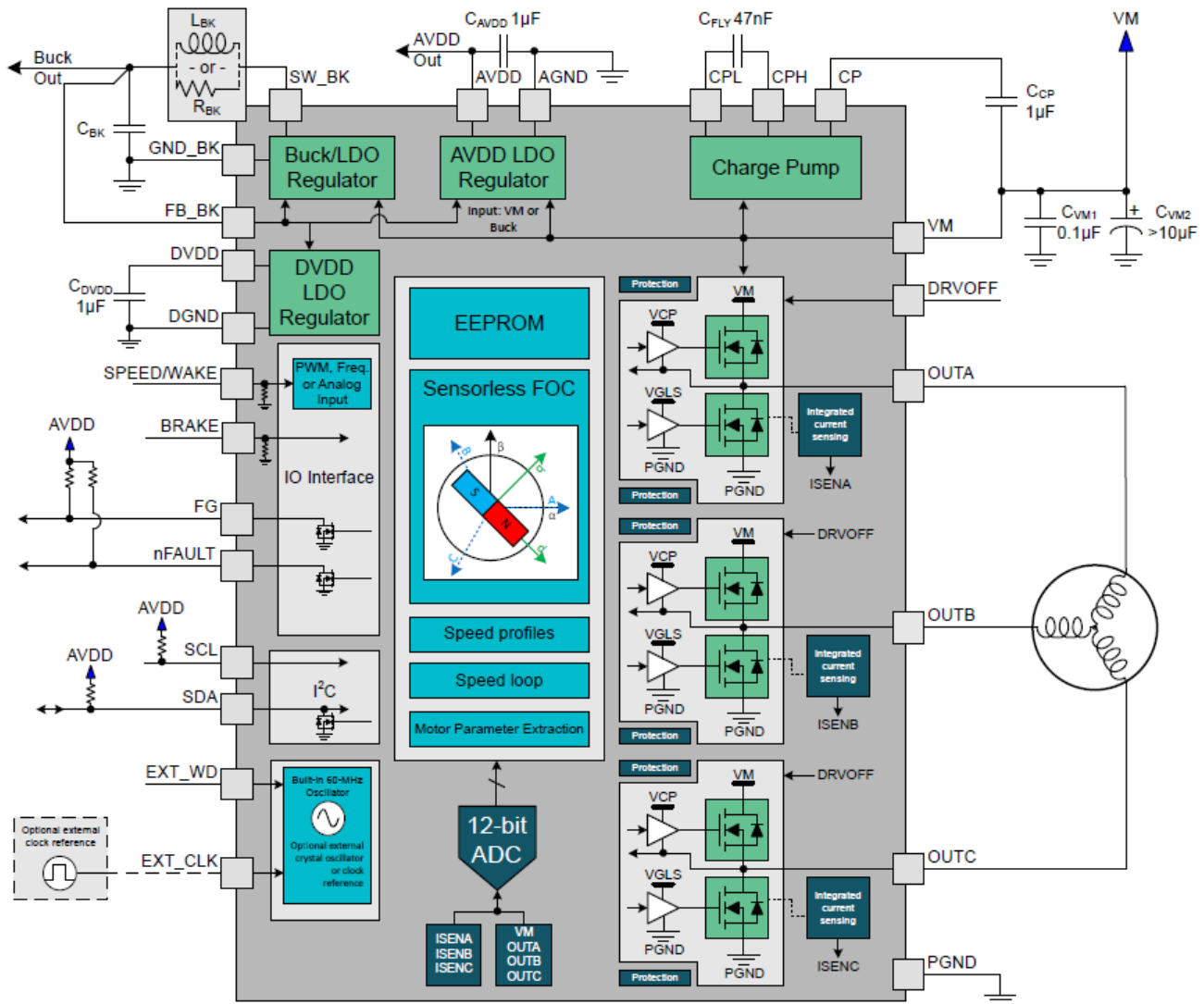


Figure 1-3. MCF8315D (PWP) Functional Block Diagram

## 2 Hardware Setup

Figure 2-1 shows the evaluation system setup with LP-MSPM0C1104 and MCF8315PWPEVM. The signal connections are represented by blue and power connections are represented by red. The actual hardware connection is shown in Section 4.

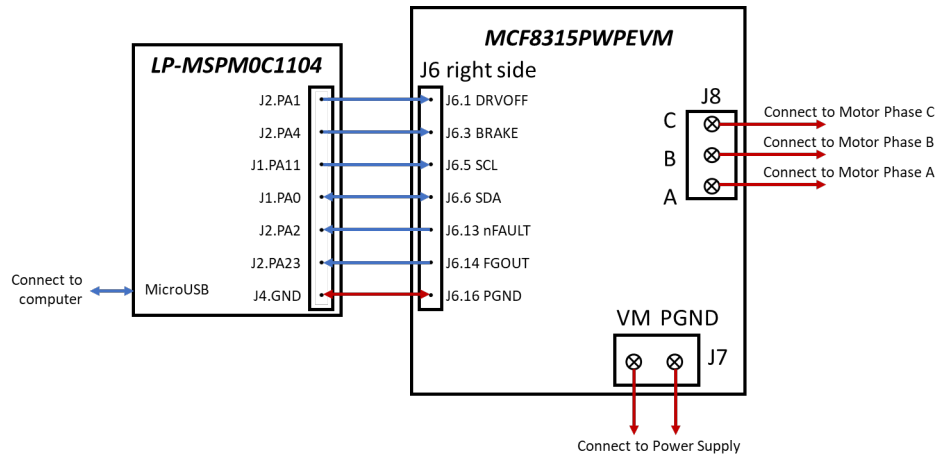


Figure 2-1. LP-MSPM0C1104 and MCF8315PWPEVM Hardware Connection

### 2.1 LP-MSPM0C1104 Hardware Setup

For detailed information and schematics of LP-MSPM0C1104, please refer to [LP-MSPM0C1104 Evaluation Module User's Guide](#), the table below shows the connection between LP-MSPM0C1104 and MCF8315PWPEVM and the corresponding function.

Table 2-1. Signal Connection between LP-MSPM0C1104 and MCF8315PWPEVM

MCF8315PWPEVM Pin Name	LP-MSPM0C1104 Pin Name	Function
DRVOFF	PA1	Disables MCF gate drivers
BRAKE	PA4	Turns on all MCF low-side MOSFETs
SCL	PA11	I2C communication clock
SDA	PA0	I2C communication data
nFAULT	PA2	Report MCF8315 Fault to MCU
FGOUT	PA23	Feedback motor speed to MCU
PGND	GND	GND

### 2.2 MCF8315PWPEVM Hardware Setup

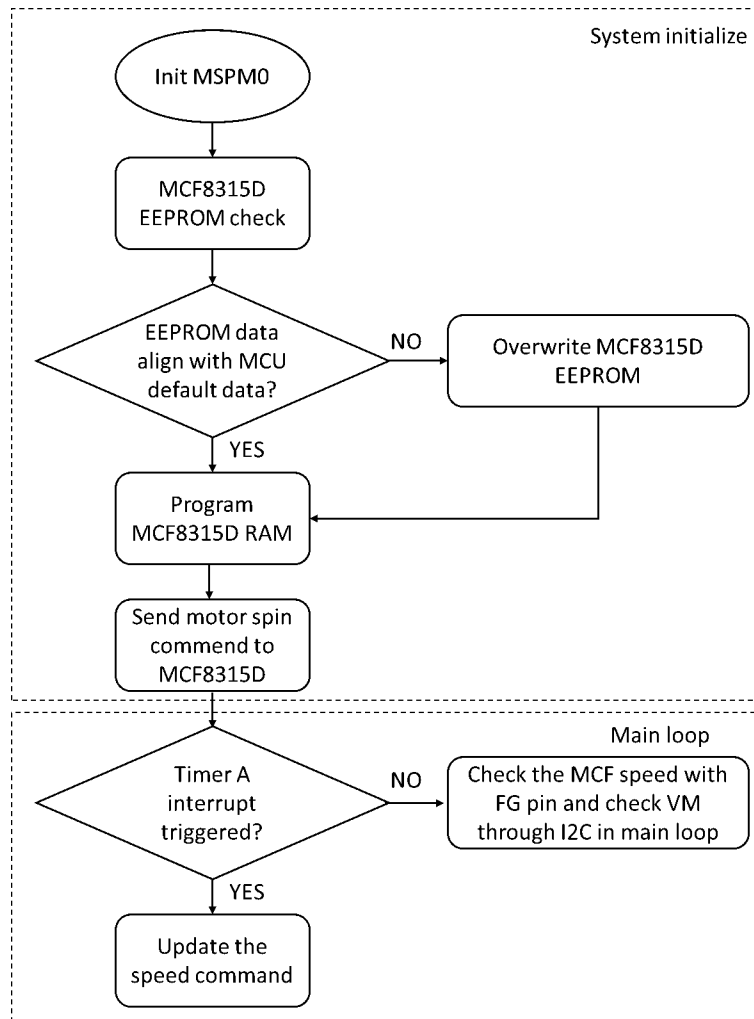
For the overall hardware description and setup of MCF8315PWPEVM, refer to the [MCF8315PWPEVM Evaluation Module](#) user's guide. Since the EVM is controlled by MSPM0C1104, all jumpers in J6 must be disconnected and reconnected as shown in Table 2-1. Leave the pins open if those pins are not mentioned in the table. DC voltage source needs to be connected to VM and PGND. Motor phase A, B, or C needs to be connected to J8 the Motor Studio GUI, and can be used for tuning the motor and export the configuration data, which is needed for MCF8315 initialization. The MCF8315D hardware design is shown in the [How to Optimize MCF8315 Hardware Design in Fan Application](#) application note.

## 3 Software Structure

### 3.1 Software Function and Flow Diagram

MSPM0C1104 configures and controls MCF8315D through I2C interface by changing MCF8315D registers. The APIs of MCF8315D EEPROM data checking and updating, MCF8315D RAM programming, motor speed control, speed feedback through FG pin, VM checking are included in the [demo code](#).

The code starts with MSPM0C1104 initialization, then MCF initialization is done through EEPROM checking and RAM programming, and the motor spin command is sent to MCF8315D after all the initiation is done. The motor speed command is updated in the timer A interrupt. Speed feedback and VM voltage monitor is done in main loop. Software flow diagram is shown in [Figure 3-1](#).



**Figure 3-1. MSPM0 Software Diagram**

## 3.2 Project File Structure

Users can easily use the System Configuration Tool to migrate this project from MSPM0C1104 to another MSPM0 devices. Key files in the project are list in [Table 3-1](#)

**Table 3-1. Project File Structure**

File	Description
C1104LP_MCF8315D.c	Program entrance
C1104_Config.syscfg	MSPM0C1104 system and peripherals configuration
mcf_configuration.h	Predefine of MCF8315D EEPROM register offset and default setting
mcf_i2c.c	MCF registers read and write functions to perform MCF register read and write with the required protocol
mcf_i2c.h	Header file to define MCF registers read and write functions
crc8.c	8 bits CRC calculation with CRC table
crc8.h	Header file to define CRC verification
parity_even.c	Data parity check align with MCF8315D EEPROM parity check rules
parity_even.h	Header file to define parity_even function

## 3.3 Configure and Control MCF8315

### 3.3.1 MCF8315D I2C Protocol Description

MSPM0C1104 communicate with MCF8315D over an I2C serial interface. MSPM0C1104 works as I2C controller, MCF8315D works as I2C target. The I2C communication follows the I2C data word format shown in [Table 3-2](#).

**Table 3-2. I2C Data Word Format**

TARGET_ID	R/W	CONTROL WORD	DATA	CRC-8
A6 - A0	W0	CW23 - CW0	D15, D31, D63 - D0	C7 - C0

**Target ID and R/W Bit:** The first byte includes the 7-bit I2C target ID, followed by the read or write command bit. Every packet in MCF8315D the communication protocol starts with writing a 24-bit control word, hence, the R/W bit is always 0.

**24-bit Control Word:** The control word format is shown in [Table 3-2](#) and the detailed descriptions can be found in the [MCF8315D Sensorless Field Oriented Control \(FOC\) Integrated FET BLDC Driver](#) data sheet. In the demo code, CRC\_EN is set to 1 to enable the CRC verification, DLEN set to 01b to use 32-bit data length, and MEM\_SEC and MEM\_PAGE are set to 0 since all other memory sectors and pages are not for external use.

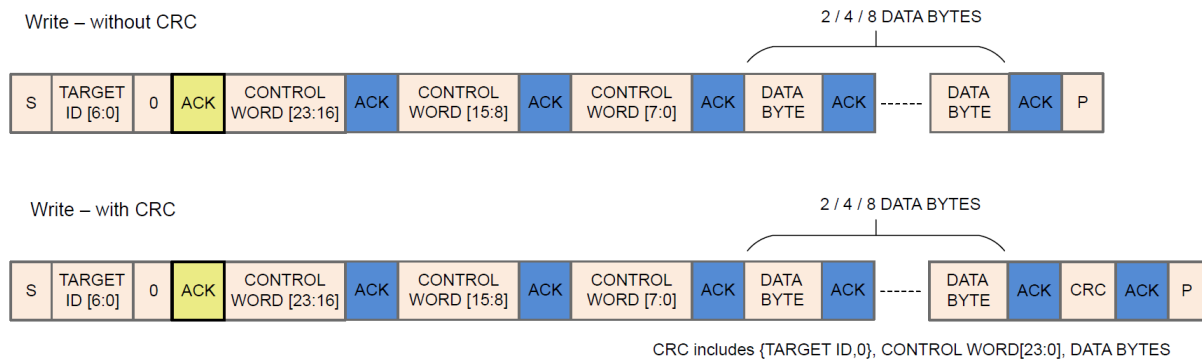
**Table 3-3. 24-bit Control Word Format**

OP_R/W	CRC_EN	DLEN	MEM_SEC	MEM_PAGE	MEM_ADDR
CW23	CW22	CW21- CW20	CW19 - CW16	CW15 - CW12	CW11 - CW0

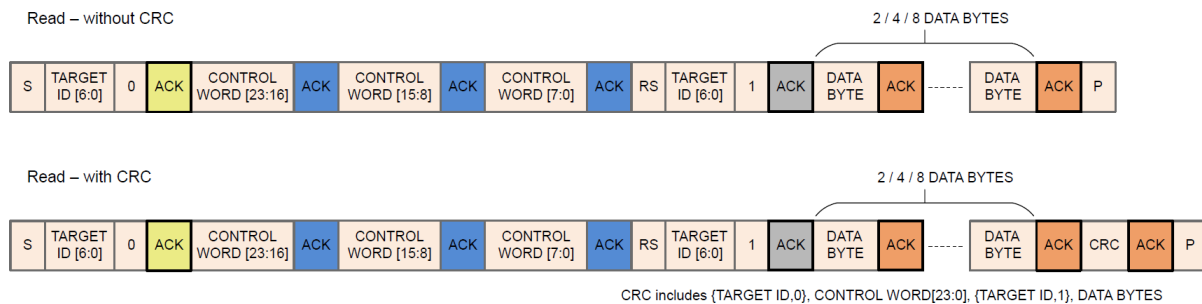
**Data Bytes:** the data needs to be input in corresponding MEM\_ADDR and the DLEN field in the control word corresponds with the number of bytes sent in this section. In case of mismatch between number of data bytes and DLEN, the write operation is discarded.

**CRC Byte:** If the CRC feature is enabled in the control word, CRC byte has to be sent at the end of a write transaction. The details of CRC Byte calculation is described in 3.3.2 CRC verification and parity verification

[Figure 3-2](#) and [Figure 3-3](#) show the I2C read or write sequence according to MCF8315D I2C protocol. To communicate with MCF8315D properly, MCU needs to support I2C clock stretch. For more information, refer to the [MSPM0C110x, MSPS003 Mixed-Signal Microcontrollers](#) data sheet and the [MCF8315D Sensorless Field Oriented Control \(FOC\) Integrated FET BLDC Driver](#) data sheet.



**Figure 3-2. MCF8315D I2C Write Transaction Sequence**



**Figure 3-3. MCF8315D I2C Read Transaction Sequence**

### 3.3.2 CRC Verification and Parity Check

An 8-bit CCIT polynomial ( $x^8 + x^2 + x + 1$ ) and CRC initial value 0xFF is used for CRC computation.

**CRC Calculation in Write Operation:** MCF8315D checks whether the MCU CRC byte is aligned with the CCIT polynomial when the external MCU writes to MCF8315D and the CRC is enabled. If there is a mismatch, then the write request is discarded. Input data for CRC calculation by external MCUs for write operations are listed below:

- Target ID + write bit
- Control word – 3 bytes
- Data bytes – 2/4/8 bytes

**CRC Calculation in Read Operation:** When the external MCU reads from MCF8315D, if the CRC is enabled, MCF8315D sends the CRC byte at the end of the data. Input data for CRC calculation by external MCU to verify the data sent by MCF8315D are listed below:

- Target ID + write bit
- Control word – 3 bytes
- Target ID + read bit
- Data bytes – 2/4/8 bytes

**Parity Calculation in Read/Write Operation:** MCF8315D calculates the parity internally following the rules below once the EEPROM is programmed and write data to EEPROM parity bit is ignored.

- If the sum of the bits is even, then parity is 0.
- If the sum of the bits is odd, then add 1 to parity to make be even.



### 3.3.3 MCF EEPROM or RAM Program

MCF8315D integrates EEPROM, which can program the default configurations to support a fast system power-up. The corresponding RAM address is 0x000080-0x0000AE. The fast power-up is practical in battery power supply applications, such as portable fan. This can also support the MCF8315D to operate as a stand-alone once the device has been configured. Due to the write cycle limitations of EEPROM, TI recommends to program the EEPROM only at first power-up. Once the EEPROM is programmed, MCUs need to check whether the EEPROM data is correct when the system is powered up again.

If the configuration needs to be changed during MCF8315D run, then change the data in RAM (0x000080-0x0000AE) to make the change effect in real time, without changing the EEPROM data.

EEPROM write procedure:

- Write all the default configuration data into shadow or RAM register (0x000080-0x0000AE)
- Write 0x8A500000 into register 0x0000EA to write the shadow or RAM register (0x000080-0x0000AE) values into the EEPROM
- Wait for 750ms for the EEPROM write operation to complete
- After 750ms, read 0x0000EA register to make sure this has been reset to 0x0

EEPROM Write procedure:

- Write 0x40000000 into register 0x0000EA to copy the EEPROM data into the shadow/RAM registers (0x000080-0x0000AE)
- Wait for 100ms for the EEPROM read operation to complete
- Read the shadow/RAM register values

---

#### Note

MCF8315D allows EEPROM write and read operations only when the motor is not spinning. EEPROM can be written and read using the I2C serial interface but erase cannot be performed using I2C serial interface.

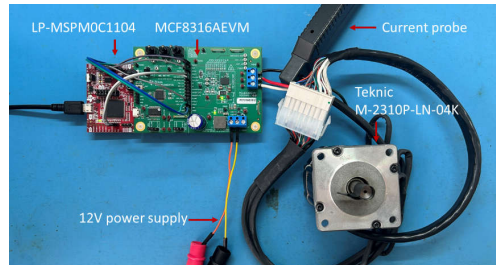
---

## 4 System Test

This section shows how to set up the test environment with LP-MSPM0C1104 and MCF8316AEVM. The device can be changed to MCF8315D with MCF8316AEVM since the devices are pin-to-pin capable. VM feedback and speed feedback test results are shown in the MSPM0 debug window.

### 4.1 Test Setup

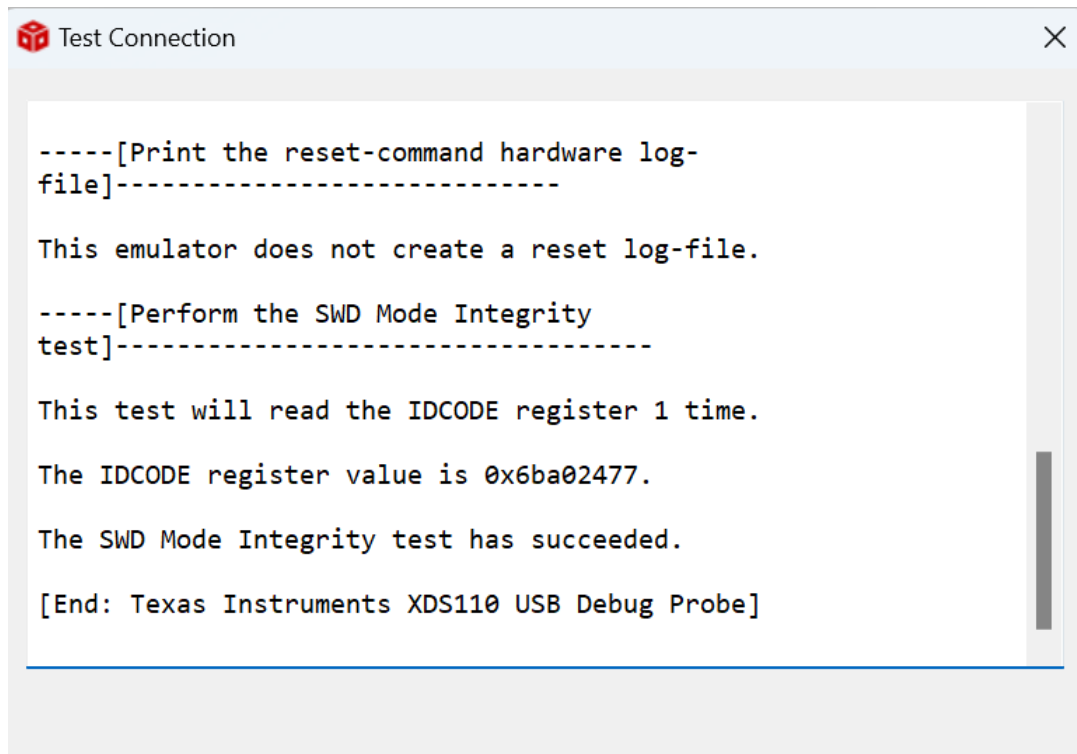
The steps to set up the test environment are detailed in this section. [Figure 4-1](#) shows the actual test environment photo for the system connection block diagram.



**Figure 4-1. Test Environment Setup**

According to [Figure 2-1](#), connect LP- MSPM0C1104 with MCF8316AEVM power supply and motor. Power up the MCF8316AEVM with 12V DC power supply, then the green LED D1 and D4 is lit up. If the nFault LED is lit up, then power cycle the EVM.

Connect LP- MSPM0C1104 to the PC with a USB cable for debug purposes, and test the emulator connection with MC. If the connection is correct, then the information shown below can be viewed.



```

Test Connection
-----[Print the reset-command hardware log-
file]-----

This emulator does not create a reset log-file.

-----[Perform the SWD Mode Integrity
test]-----

This test will read the IDCODE register 1 time.

The IDCODE register value is 0x6ba02477.

The SWD Mode Integrity test has succeeded.

[End: Texas Instruments XDS110 USB Debug Probe]

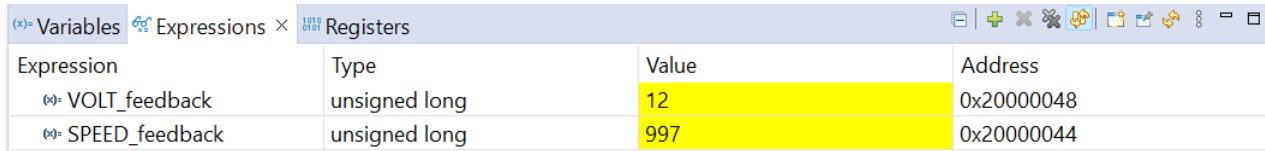
```

Import the demo code to CCS. Click *debug* and run the program.

**Figure 4-2. Emulator Test Connection with MSPM0C1104**

## 4.2 System Value Monitor

Add the VOLT\_feedback (V) and SPEED\_feedback (Hz) to the Expressions window to monitor the data. The motor speed is changed by setting the SetMotorspeed value, and the motor current waveform can be monitored in this scope.



Expression	Type	Value	Address
VOLT_feedback	unsigned long	12	0x20000048
SPEED_feedback	unsigned long	997	0x20000044

Figure 4-3. MCU Variables Monitor Window

## 5 Summary

This application note presents an easy-to-use LV BLDC system with MCF8315D and MSPM0C1104. The document can help designers accelerate evaluation and development of LV BLDC systems. To build the specific LV BLDC system, users can port this code easily to other MSPM0 devices using the system configuration tool (SYSCONFIG). Demo code can be accessed from the E2E. For hardware design, refer to [How to Optimize MCF8315 Hardware Design in Fan Application](#) application note.

## 6 References

- Texas Instruments, [MCF8315D Sensorless Field Oriented Control \(FOC\) Integrated FET BLDC Driver](#), data sheet
- Texas Instruments, [MCF8315PWPEVM Evaluation Module](#), user's guide
- Texas Instruments, [MSPM0C110x, MSPS003 Mixed-Signal Microcontrollers](#), data sheet
- Texas Instruments, [MSPM0 C-Series Microcontrollers](#), technical reference manual
- Texas Instruments, [How to Optimize MCF8315 Hardware Design in Fan Application](#), application note
- Texas Instruments, [MSPM0C1104: \[Not question\] MCF8315 configuration with MSPM0C](#)

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](https://www.ti.com) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
Copyright © 2025, Texas Instruments Incorporated