MC2110RN

Motor Control Release Notes v2.11.0

Rev. 0 — 14 January 2022 Release Notes

by: NXP Semiconductors

Contents

1 Introduction

These release notes are for the motor-control middleware group of applications released together with the MCUXpresso SDK v2.11.0. This document provides a list of application examples, their notable features, supported hardware platforms, changes since the last MCUXpresso release, known issues, and links to further documentation.

2 Description

This motor-control middleware release contains application examples for the following three-phase electrical machine topologies:

- **AC Induction Motor (ACIM)**

  The Field-Oriented Control (FOC) sensorless (machine model-based estimator algorithms are used to replace the position and speed sensor) application examples **mc_acim** were developed for the high-voltage development platform and Kinetis MCUs with a floating-point unit (see Supported platforms for exact platform support details). All examples integrate the Motor Identification (MID) software module and feature the Motor Control Application Tool (MCAT) to enable quick development.

  See the user's guide in the *\docs\MC* folder in your *SDK Documentation* package (see Examples) or the www.nxp.com/motorcontrol_acim web page for more details.

- **Permanent Magnet Synchronous Motor (PMSM)**

  These FOC applications support both high- and low-voltage hardware platforms and various MCU types (see Supported platforms for exact platform support details). The following application types are available in the *mc_pmsm* folder of your SDK archive (see Examples):

  — **pmsm_snsless** - Sensorless FOC examples utilizing both fractional and floating-point arithmetics. The Motor Identification (MID) software module in combination with the Motor Control Application Tool (MCAT) allow for rapid application development.

  — **pmsm_snsless_reg_init** - Sensorless FOC examples with same features as **pmsm_snsless**. MCU peripherals are completely initialized using MCUXpresso Config Tools.

  — **pmsm_enc** - This PMSM FOC application is identical to the **pmsm_snsless** example, except for the added option of acquiring the rotor position and speed from the encoder sensor.

  See the user's guide in the *\docs\MC* folder in your SDK Documentation package (see Examples) or the www.nxp.com/motorcontrol_pmsm web page.

All examples support the FreeMASTER interface for quick and simple application debugging, tuning, control, and monitoring. See www.nxp.com/freemaster and the application user's guide for more information.

3 Examples

The example projects are distributed only in the form of the MCUXpresso SDK Archive and the release documentation is available in the SDK Documentation package. To acquire both packages (specific to your development platform), use the online MCUXpresso SDK Builder tool and perform the following steps:

- Go to www.mcuxpresso.nxp.com.
- Click the **Select Development Board** button.
- Sign in or create the NXP account (if requested).
- Choose one of the supported platforms (see Supported platforms for the list of boards supported by this release).
- Click the **Build MCUXpresso SDK** button.
- Make sure that the **Motor Control** middleware is selected and click the **Download SDK** button.
- When the SDK Documentation and SDK Archive package build is done (you receive a notification email), it can be downloaded freely.

4 Supported platforms

The motor-control application examples were developed and tested with the following development tools:

- IAR Embedded Workbench IDE version 9.20.2
- Arm®-MDK - Keil® µVision® version 5.36
- MCUxpresso IDE version 11.5.0

FreeMASTER tool version 3.1.3 was used for application monitoring. See www.nxp.com/freemaster for the latest version.

The hardware platforms supported by this release are listed in the following table.

Table 1. Supported platforms

Board	mc_acim	pmsm_snsless	pmsm_enc
EVK-MIMXRT1170			✓fp, mid
FRDM-KE17Z		fix, mid	
FRDM-KV11Z		✓fix, mid, reg_init	
FRDM-KV31F		✓fp, mid, reg_init	
HVP-KV11Z75M		✓fix, mid	
HVP-KV31F120M	✓fp, mid	✓fp, mid	

fix Fixed-point arithmetics.

fp Floating-point arithmetics.

mid Motor Identification (MID) software module is available.

reg_init MCU peripherals initialized using MCUXpresso Config Tools.

5 What is new

This section describes all notable changes since the last motor-control middleware MCUXpresso SDK release v2.10.0.

1. **New Motor Identification (MID)**

   Floating-point version of the **mc_pmsm** and **mc_pmsm_enc** examples features enhanced electrical parameters estimation algorithm which doesn't require characterized power stage and achieves higher estimation accuracy at motors with lower electrical parameters.

2. **New Motor Control Application Tuning (MCAT) tool**

   Floating-point version of the **mc_pmsm** and **mc_pmsm_enc** examples features new MCAT which is build on the latest technologies and solves some imperfections from previous version.

3. **Updated documentation**

   The documentation for the **mc_acim**, **pmsm_enc**, **pmsm_snsless**, and **pmsm_snsless_reg_init** examples was updated and divided into separate documents according to selected MCU. The documents are available as a part of the SDK Documentation package (see Examples).

6 Known issues

This chapter contains the description of known issues or non-standard behavior of the released example.

1. **Using a MAP file instead of TSA**

   Applies to all **mc_pmsm**, **mc_pmsm_enc** and **mc_pmsm_reg_init** examples. When user disables FreeMASTER TSA functionality and switches to MAP file specification approach, the Board ID variable might not be read properly. Consequently path to m1_pmsm_appconfig.h is not deternied and data are not loaded to MCAT. This can be workarounded by refreshing the MCAT page manully (F5) after MAP file is selected and FreeMASTER communication started.

2. **MID Charaterization routine removed**

   Applies to **mc_pmsm**, **mc_pmsm_enc** and **mc_pmsm_reg_init** floating point examples. Power stage characterization routine which was available at MID MCAT tab was removed because not it is not necessary for stator resistance measurement anymore. Dead time compensation LUT still uses data obtained by characterization routine. Characterization data can be obtained using SDK package 2.10

3. **Constants not updated by MCAT**

   Applies to **mc_pmsm**, **mc_pmsm_enc** and **mc_pmsm_reg_init** floating point examples. Following constants are not updated using MCAT's update button. This issue can be workarounded by regenerating m1_pmsm_appcofig.h file and re-compiling the project.

   Angular scale (FMSTR_MID_speedAngularScale)

   DCbus voltage and Tracking Observer IIR filters

   Startup Speed integrator and Scalar Frequency interator gains

   V/Hz factor (M1 V/Hz factor)

7 Feedback

Your feedback is very important to us. Please feel free to leave a comment here.

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer’s applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 14 January 2022

Document identifier: MC2110RN

arm