STM32 PMSM single/dual FOC SDK Release 4.2

Introduction

This release note provides important information about the STM32 PMSM FOC SDK motor control software design kit composed by PMSM FOC FW library and ST MC Workbench (reference: STSW-STM32100).

This release note is updated periodically in order to keep you abreast of evolutions of the package and any problems or limitations found. Check the ST microcontroller support website at *www.st.com* to ensure that this is the latest version of this release note.

Customer support

For more information or help concerning STM32 PMSM FOC SDK, please contact the nearest sales office. For a complete list of ST offices and distributors, please refer to *www.st.com*.

Software updates

You can download software updates and all the latest documentation from the ST microcontroller support site at *www.st.com.*

Microcontrollers supported

- STM32F030C6/C8/K6/R8
- STM32F051C6/C8/K6/K8/R6/R8
- STM32F100, STM32F103
- STM32F2 Series
- STM32F302xB/C, STM32F303xB/C, STM32F302x6/x8
- STM32F405xx, STM32F407xx, STM32F415xx, STM32F417xx.

Contents

1 Description

ST's STM32 offers the performance of the industry-standard ARM$^®$ Cortex$^®$-M core at the service of vector (or field-oriented) control (FOC) algorithms, widely used in high-performance drives. The STM32 PMSM FOC SDK (STSW-STM32100), which includes the PMSM FOC FW library and ST MC Workbench, allows the user to evaluate the STM32 performance in applications driving single or dual Field Oriented Control of 3-phase Permanent Magnet motors (PMSM, BLDC).

ST MC Workbench is a PC software that reduces the design effort and time in the STM32 PMSM FOC firmware library configuration. The user, through a graphical user interface (GUI), generates all parameter header files which configure the library according to the application needs and can in real time monitor and change some variables of the algorithm.
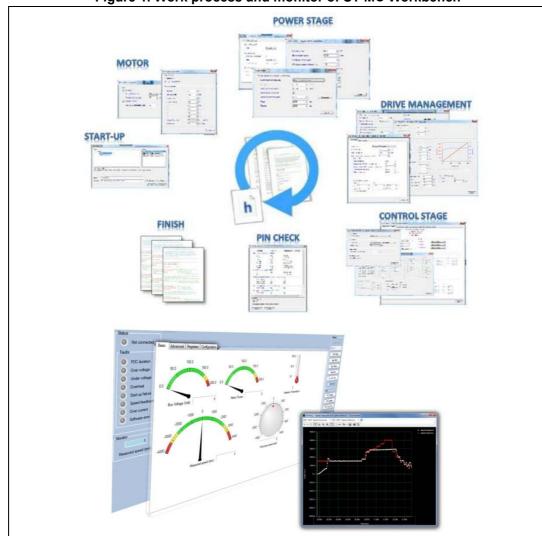
Figure 1. Work process and monitor of ST MC Workbench

2 ST Motor Control Workbench

2.1 What's new

- Support for STM32F302x6/x8

2.2 Program features

- Configuration of all the parameters required by 'STM32 PMSM FOC firmware library' supported
- Generation of all the header files (.h) required by 'STM32 PMSM FOC firmware library' supported
- Support for single and dual motor control
- Online communication with motor control application
- High frequency injection sensor less (HFI)
- Motor Profiler support
- New project creation starting from the board
- On-the-fly sensorless startup

2.3 Supported ST MC Platform

- STM32 PMSM FOC SDK v4.2 (support for STM32F302x6/x8 microcontrollers)
- STM32 PMSM FOC SDK v4.1
- STM32 PMSM FOC SDK v4.0
- STM32 PMSM FOC SDK v3.4 (support for STM32F03x, STM32F050x, STM32F051x, STM32F3xx and previous microcontrollers)
- STM32 PMSM FOC SDK v3.3
- STM32 PMSM FOC SDK v3.2 (support for STM32F2xx and STM32F4xx)
- STM32 PMSM FOC SDK v3.0.1 (for online communication)
- STM32 PMSM FOC SDK v3.0 .

2.4 Supported STM32 microcontrollers

- STM32F030C6/C8/K6/R8
- STM32F051C6/C8/K6/K8/R6/R8
- STM32F100 Value line
- STM32F103
- STM32F2 Series
- STM32F302xB/xC, STM32F302x6/x8, STM32F303xB/xC
- STM32F405xx, STM32F407xx, STM32F415xx, STM32F417xx

2.5 Release information

2.5.1 About release 4.2

- Support for STM32F032x6/x8
- Minor bug fixes

2.5.2 About release 4.1

- Motor
  - Added Motor Profiler support
- Power stage
  - Changed "Bus voltage sensing" management
- Drive Management
  - Added support for On-the-fly sensorless startup
- Control Stage
  - Changed "Bus voltage feedback" management
- Program Feature
  - New project creation starting from the board
  - Changed Output Folder option
  - The ".stmc" saved extension format has been changed in ".stmcx"
  - Updated documentation
- Minor bug fixes

2.5.3 About release 4.0.0

- Drive Management
  - Added HFI sensor less management
  - Improved Start-up parameters (start-up rump plotting).
  - Added AC Input selection
- Power Stage
  - Improved Driving Signals Polarity: Added the possibility to force the same values for all U, V, W drivers simultaneously.
- Program features:
  - Adding Compact/Extend mode: showing/hiding advanced feature.
- Minor bug fixes

2.5.4 About major previous releases

Release 3.0.4

- Added unit measure for Torque&Flux - Cut-off frequency
- Changed behavioral of bus voltage sensing inverting input
- Improved conversion from STM32F05xx of WB 2.1 to STM32F051x of WB 3.0
- Removed read-only for OPAMP inverting pins if shared resource enable and OPAMP Gain is External in dual motor configuration
- Removed read-only for COMP output pins
- Improved start-up time
- Minor bug fixes

Release 3.0.2

- Minor bug fixes

Release 3.0.1

- Added Alternate Function information for comparators
- Pin management in change target library
- External Protection and No protection selection available also for no STM32F3x
- Added managing of the ADC sampling for phase current feedback (clock frequency, minimum, divider)
- Allowing internal/external or vice-versa gain type in shared resource configuration
- Minor bug fixes

Release 3.0.0

- Support for STM32F3x for single and dual
- Extended support for STM32F0xx (STM32F030x, STM32F050x, STM32F051x)
- Example projects list
- Recent project list
- Export for logs
- Support for embedded OPAMP
- Support for embedded COMP
- PFC support
- Extended DAC functionality
- Added Amplification Network Gain form and export in HTML format
- Minor bug fixes

2.6 Host PC system requirements

- PC running on Microsoft Windows? operating systems.
- Required Space: ~ 30 MB
- Minimum screen resolution: 1024x768
- RS-232 serial communication port or, equivalently, a USB to RS-232 converter (required for online communication)

Note: *The software requires Microsoft .NET Framework 3.5 SP1 or higher*

Note: *System Administration Rights are required for Setup.*

3 STM32 PMSM FOC FW library

3.1 New features

Version v4.2

- Support of STM32F302x6/x8 microcontrollers
- Three shunt algorithm using only one ADC for STM32F302x6/x8 MCUs
- Native support (project configuration) for P-NUCLEO-IHM001 kit
- Motor profiler improvement (hysteretic control of current to speed up the procedure)
- One touch tuning (hysteretic control to speed up the procedure, start-up ramp set according to the load set in WB)

Version v4.1

- Motor Profiler (only for STM32F30x and STM32F4xx)
- One Touch Tuning (only for STM32F30x and STM32F4xx)
- On-the-fly sensorless startup

Version v4.0

- Sensorless (High Frequency Injection HFI plus B-EMF State Observer, PLL rotor speed/angle computation from B-EMF, only for STM32F30x or STM32F4xx)
- Support to ARM® Keil® µVision® and IAR Embedded Workbench® IDEs
- Simplify the MC SDK with a self-explaining approach
- Ready to use application examples
- Fast unidirectional serial communication
- Simplified user LCD interface

Version v3.4

- Support of the STM32F30x microcontroller families has been added.
- Support of STM32F3's enhanced set of peripherals including comparators, PGAs, DACs, high-speed ADCs and CCMRAM.
- Support of the STM32F0x family enlarged, now comprising STM32F030x, STM32F050x, STM32F051x.
- HardFault handler now used for error signaling and application safety.
- The STM3210B-EVAL new LCD MB895/S (HX8347D) is now supported by LCD FW.
- Support of L6230's enable pins.
- Default variables can be configured and displayed using the DAC functionality.
- In dual motor control mode, the same bus voltage measurement can be used for both motor controls.

Version v3.3

- Support of the STM32F05xx microcontroller families has been added.
- Support of inrush current limiter has been added.
- General purpose ramps generation class has been added.

3.2 Known problems and limitations

3.2.1 Know problems/limitations in v4.1 fixed in v4.2

- Hardfault occurring when enabling MTPA in confidential or web distribution
- Motor speed not reset in Idle state (sensorless configuration)

3.2.2 Know problems/limitations in v4.0 fixed in v4.1

- Error: "TIM8_BRK_IRQHandler already defined" compiling a single drive project for STM32F3 using TIM8
- Ramp manager class and SpeednTorqCtrl class internal overflow for long ramp duration
- When enabling Fast Unidirectional Serial communication the firmware gets stuck in the USART interrupt
- If BKIN input is active while MCboot not completed MemFault exception is raised
- WB example for STEVAL-IFN003V1 - Enabling signals not checked
- Unidirectional serial generate only one channel even if two are selected in WB
- Keil® project - Inrush Current Limiter not included in the MC Lib (ICL)
- LCD FW for STEVAL-IHM039V1 for dual drive wrong
- Wrong clock configuration for USART in STM32F3 projects (affects STM32F303x8, STM32F302x8, STM32F301x8)

3.2.3 Know limitations in v4.2

- Start/stop button management not implemented in the FW
- HW dead time not used even if selected, SW used instead
- Sampling time not taken in consideration to determine if sampling in the middle of period is possible (affects STM32F3).
- Dual sensor-ed configuration is not working (is not possible to enable both HALL as primary speed sensor and Encoder as auxiliary sensor, or vice versa)
- STEVAL-IHM022V1 using Free RTOS project not working in dual drive
- ENCODER speed sensor function: one Input Capture pin of the selected timer must be grounded, according to the remapping in use and this rationale:
  - IC4 in case of TIM2
  - IC4 in case of TIM3
  - IC3 in case of TIM4
  - IC4 in case of TIM5.
- AF function for TIM3, TIM4, etc used by sensor feedback is not set properly
- Hall / Encoder modules: GPIO configuration is not locked
- False spike in the DAC variable related to measured current during calibration phase
- LCD User Interface: when the "start both motors" button is pushed, a new value of target speed set before is not taken into account
- The linker file (IAR) for MC project (STM32F2 and STM32F4 microcontrollers) doesn't take in account the Flash and RAM memory reserved for the LCD Project
- DAC functionality: TIM3 remap not configurable when working with STM32F103 with Flash memory density lower than or equal to 128kBytes.
- Serial Com User Interface: MCI_StartMotor and MCI_StopMotor, return value not checked.

3.2.4 Know limitations in v4.1

- Start/stop button management not implemented in the FW
- HW dead time not used even if selected, SW used instead
- AF function for TIM3, TIM4, etc used by sensor feedback is not set properly
- Sampling time not taken in consideration to determine if sampling in the middle of period is possible (affects STM32F3).
- Dual sensor-ed configuration is not working (is not possible to enable both HALL as primary speed sensor and Encoder as auxiliary sensor, or vice versa)
- STEVAL-IHM022V1 using Free RTOS project not working in dual drive
- ENCODER speed sensor function: one Input Capture pin of the selected timer must be grounded, according to the remapping in use and this rationale:
  - IC4 in case of TIM2
  - IC4 in case of TIM3
  - IC3 in case of TIM4
  - IC4 in case of TIM5.
- Hall / Encoder modules: GPIO configuration is not locked.
- False spike in the DAC variable related to measured current during calibration phase.

- LCD User Interface: when the "start both motors" button is pushed, a new value of target speed set before is not taken into account.
- The linker file (IAR) for MC project (STM32F2 and STM32F4 microcontrollers) doesn't take in account the Flash and RAM memory reserved for the LCD Project.
- DAC functionality: TIM3 remap not configurable when working with STM32F103 with Flash memory density lower than or equal to 128kBytes.
- Serial Com User Interface: MCI_StartMotor and MCI_StopMotor, return value not checked.

3.2.5 Know problems/limitations in v3.4 fixed in v4.0

- Current reading error for STM32F3x due wrong computation of sampling time in NS done in the parameter conversion.
- Wrong encoder alignment is performed
  - when set alignment angle is different from 90 degrees or
  - when motor poles pairs are different from 2.
- - If HALL sensor is used then an error (division by 0) occurs when the timer pre-scaler becomes 0.

3.2.6 Known limitations in v4.0

- STEVAL-IHM022V1 Dual Drive - Free RTOS - Hard fault driving both motors.
- Sampling time is not taken in consideration to determine if sampling in the middle of period is possible (affects STM32F3).
- ENCODER speed sensor function: one Input Capture pin of the selected timer must be grounded, according to the remapping in use and this rationale: IC4 in case of TIM2, IC4 in case of TIM3, IC3 in case of TIM4, IC4 in case of TIM5.
- Hall / Encoder modules: GPIO configuration is not locked.
- False spike in the DAC variable related to measured current during calibration phase.
- LCD User Interface: when the "start both motor" button is pushed, a new value of target speed settled before is not taken into account.
- The linker file (IAR) for MC project (STM32F2 and STM32F4 microcontrollers) doesn't take in account the Flash and RAM memory reserved for the LCD Project.
- DAC functionality: TIM3 remap not configurable when working with STM32F103 with Flash memory density lower than or equal to 128kBytes.
- Serial Com User Interface: MCI_StartMotor and MCI_StopMotor, return value not checked.

3.2.7 Known problems/limitations in v3.3 fixed in v3.4

- An issue in the state observer (speed and position sensor) has been solved; the malfunctioning was introduced in v3.3, when optimizing execution speed on STM32F0.
- In Timebase.c, counters declared as 8bit now corrected as 16bit variables.
- RampMngr Class compliancy with MISRA rule.
- InrushCurrentLimiter Class compliancy with MISRA rules fixed.
- In 3shunt and ICS current reading class, function SwitchOnPWM now waits the timer update before activating driving signals.
- In 3shunt current reading class, phase C calibration now done correctly using 1 ADC peripheral only.
- The linker file (IAR) for MC project (F2 and F4) didn't take into account the Flash and RAM memory reserved for LCD Project.
- Added the support for the LCD marked MB895/S C-03 of STM3210B-EVAL and STEVAL-IHM022v1.
- Added the support for the LCD marked MB895/P C-03 of STM3210E-EVAL.

3.2.8 Known limitations in v3.4

- ENCODER speed sensor function: one Input Capture pin of the selected timer must be grounded, according to the remapping in use and this rationale: IC4 in case of TIM2, IC4 in case of TIM3, IC3 in case of TIM4, IC4 in case of TIM5.
- Hall / Encoder modules: GPIO configuration is not locked.
- False spike in the DAC variable related to measured current during calibration phase.
- LCD User Interface: when the "start both motor" button is pushed, a new value of target speed settled before is not taken into account.
- The linker file (IAR) for MC project (STM32F2 and STM32F4 microcontrollers) doesn't take in account the Flash and RAM memory reserved for the LCD Project.
- DAC functionality: TIM3 remap not configurable when working with STM32F103MD.
- Serial Com User Interface: MCI_StartMotor and MCI_StopMotor, return value not checked.

3.2.9 Known problems/limitations in v3.2 fixed in v3.3

- The issue that stuck the LCD firmware when the DAC function was not enabled has been fixed.
- The generation of wrong PWM frequencies in the STM32F1xx configuration for TIM_CLOCK_DIVIDER not equal to one has been fixed.
- The generation of wrong dead times in the STM32F1xx configuration for TIM_CLOCK_DIVIDER not equal to one has been fixed.
- The integral term of PID objects with MAX integral term instead of zero has been fixed.
- The "Initial Electrical Angle" settled via the STMC workbench did not have any effect in the previous version of the library. The issue has been fixed.
- Cross checks and backward compatibilities with the workbench generated files has been fixed in the LCD project.
- The current measurement for ICS sensors (STM32F103HD, STM32F2xx or STM32F4xx) using the wrong offset has been fixed.
- The switch context algorithm of all dual motor current sensing classes (STM32F103HD, STM32F2xx or STM32F4xx) has been fixed.

3.2.10 Known limitations in v3.3

- ENCODER speed sensor function: one Input Capture pin of the selected timer must be grounded, according to the remapping in use and this rationale: IC4 in case of TIM2, IC4 in case of TIM3, IC3 in case of TIM4, IC4 in case of TIM5.

4 Revision history

Table 1. Document revision history

Date	Revision	Changes
14-Nov-2012	1	Initial release.
18-Apr-2014	2	Added Version 3.4.
4-Jun-2014	3	Updated *Introduction*, *Section 1: Description* and *Section 3.2.5*. Added ST MC Workbench release notes in *Section 2*. Added Version 4.0 of PMSM FOC FW library in *Section 3.2.5* and *Section 3.2.6*.
26-May-2015	4	Added Version 4.1. Updated *Microcontrollers supported*, *Section 2.1: What's new*, *Section 2.3: Supported ST MC Platform*, *Section 2.4: Supported STM32 microcontrollers* and *Section 2.6: Host PC system requirements*. Added *Section 2.5.1: About release 4.2*, *Version v4.2* in *Section 3.1*, *Section 3.2.2: Know problems/limitations in v4.0 fixed in v4.1* and *Section 3.2.3: Know limitations in v4.2*. Updated Disclaimer.
03-Sep-2015	5	Added Version 4.2. Updated *Microcontrollers supported*, *Section 2.1: What's new*, *Section 2.2: Program features*, *Section 2.3: Supported ST MC Platform*, *Section 2.4: Supported STM32 microcontrollers* and *Section 2.6: Host PC system requirements*. Added *Section 2.5.1: About release 4.2*, *Version v4.2* in *Section 3.1*, *Section 3.2.1: Know problems/limitations in v4.1 fixed in v4.2* and *Section 3.2.3: Know limitations in v4.2*.

IMPORTANT NOTICE – PLEASE READ CAREFULLY