TN0516
Technical note

Overview of the STM32F0x/F100xx/F103xx and STM32F2xx/F30x/F4xx MCUs PMSM single/dual FOC SDK V4.0

Introduction

This technical note is an overview of the main features of the Motor Control Software Development Kit (generically called software library) designed for and to be used with STM32F0xx, STM32F100xx, STM32F103xx, STM32F2xx, STM32F302xB/C, STM32F303xB/C and STM32F4xx MCUs. The library exploits a new sensorless technique that, in conjunction with an I-PMSM motor, is able to extend the range of allowed speed to zero. This novel sensor-less algorithm takes benefit of the motor structure in order to detect the rotor angular position even when the motor is at low speed or still. In this note we will refer to this technique as “High Frequency Injection”, or HFI. This new algorithm takes benefit of the floating point unit of STM32F30x and STM32F4xx MCUs.The software library implements the Field Oriented Control (FOC) drive of 3-phase Permanent Magnet Synchronous Motors (PMSM), both Surface Mounted (SM-PMSM) and Internal (I-PMSM).

The PMSM FOC library can be used to quickly evaluate ST microcontrollers and complete ST application platforms, and to save time when developing Motor Control algorithms to be run on ST microcontrollers. It is written in C language, and implements the core Motor Control algorithms (reference frame transformations, currents regulation, speed regulation, space-vector modulation, energy efficiency optimizations) as well as sensor reading/decoding algorithms (three shunts, ST-patented single DC link shunt, isolated current sensors, incremental encoder, hall sensors) and a sensorless algorithm for rotor position reconstruction.The library can be easily configured to make use of STM32F30x's embedded advanced analog peripheral set (fast comparators and Programmable Gain Amplifiers, PGA) for current sensing and protection, thus simplifying application board.

When deployed with STM32F103xx (Flash memory from 256KBytes to 1MByte), STM32F2xx, STM32F303xB/C or STM32F4xx devices, the library allows simultaneous dual FOC of two different motors. The library can be customized to suit user application parameters (motor, sensors, power stage, control stage, pin-out assignment) and provides a ready-to-use Application Programming Interface (API).

A user project has been implemented to demonstrate how to interact with the Motor Control API. This project provides LCD and UART User Interface, thus representing a convenient real-time fine-tuning and remote control tool. A PC Graphical User Interface (GUI), the ST MC Workbench, allows a complete and easy customization of the PMSM FOC library. In conjunction with the ST motor control starter kits, a PMSM motor can be made to run in a very short time. Moreover a set of ready-to-use examples are provided together with the library to explain the usage of the motor control API and it's most common features.

Table 1. Applicable products

Type	Applicable products
Microcontrollers	STM32F030C6/C8/K6/R8, STM32F050C6/G6/K6, STM32F051C6/C8/K6/K8/R6/R8, STM32F100 Value line, STM32F103, STM32F2 Series, STM32F302xB/xC, STM32F303xB/xC, STM32F4 Series

Contents

1 Motor control library features

- Single or simultaneous Dual PMSM FOC sensorless / sensored (Dual PMSM FOC only when running on STM32F103xx High-Density, STM32F103xx XL-Density or STM32F2xx or STM32F303xB/C or STM32F4xx)
- Speed feedback:
  - Sensorless (High Frequency Injection HFI plus B-EMF State Observer, PLL rotor speed/angle computation from B-EMF, only for STM32F30x or STM32F4xx)
  - Sensorless (B-EMF State Observer, PLL rotor speed/angle computation from B-EMF)
  - Sensorless (B-EMF State Observer, CORDIC rotor angle computation from B-EMF)
  - 60° or 120° displaced Hall sensors decoding, rising/falling edge responsiveness
  - Quadrature incremental encoder
  - For each motor, dual simultaneous speed feedback processing
  - On-the-fly speed sensor switching capability
- Current sampling methods:
  - Two ICS (only when running on STM32F103xx or STM32F2xx or STM32F4xx)
  - Single, common DC-link shunt resistor (ST patented)
  - Three-shunt resistors placed on the bottom of the three inverter legs (only when running on STM32F103xx or STM32F2xx or STM32F302xB/C or STM32F30xB/C or STM32F4xx)
- Embedded analog features (STM32F30x only)
  - PGA (Programmable Gain Amplifiers) for current sensing: support for three-shunt and single-shunt, internal and external gain
  - Comparators for over-current protection: support for three-shunt and single-shunt, internal and external threshold
  - Comparators for over-voltage protection: support for motor phases short-circuiting mode and free-wheeling mode, internal and external threshold
- FOC hardware acceleration(STM32F30x only)
  - ADC queue of context (ST patented architecture) support
  - CCM (Core Coupled Memory) RAM support
  - Advanced Timer structures for single shunt (ST patented) support
- Flux weakening algorithm to attain higher than rated motor speed (optional)
- Feed-Forward, high performance current regulation algorithm (optional)
- SVPWM generation:
  - Centered PWM pattern type
  - Adjustable PWM frequency
- Torque control mode, speed control mode; on-the-fly switching capability
- Brake strategies (optional):
  - Dissipative DC link brake resistor handling
  - Motor phases short-circuiting (with optional hardware over-current protection disabling)

  - Motor phases free-wheeling
- When running Dual FOC, any combination of the above-mentioned speed feedback, current sampling, control mode, optional algorithm
- Optimized I-PMSM and SM-PMSM drive
- Programmable speed ramps (parameters duration and final target)
- Programmable torque ramps (parameters duration and final target)
- Real-time fine tuning of:
  - PID regulators
  - Sensorless algorithm
  - Flux weakening algorithm
  - Start-up procedure (in case of sensorless)
- Fault conditions management:
  - Over-current
  - Over-voltage
  - Over-temperature
  - Speed feedback reliability error
  - FOC algorithm execution overrun
- Easy customization of options, pin-out assignments, CPU clock frequency through ST MC Workbench GUI
- C language code:
  - Compliant with MISRA-C 2004 rules
  - Conforms strictly with ISO/ANSI
  - Object-oriented programming architecture

1.1 User project and interface features

Two options are available:

- FreeRTOS-based user project (for STM32F103xx and STM32F2xx only)
- SysTick-timer-easy-scheduler-based user project

Available User Interface options (and combinations of them):

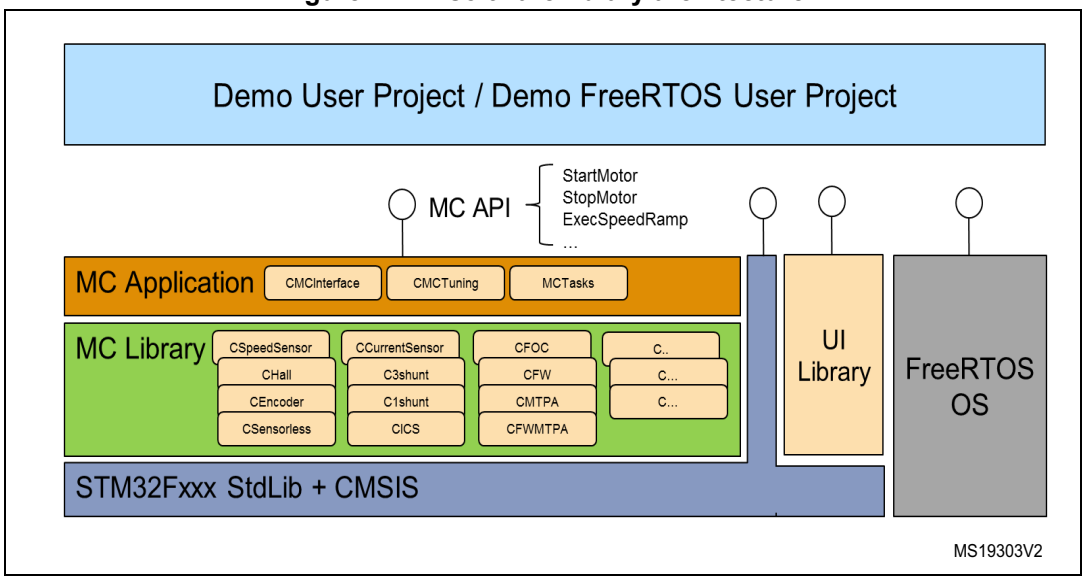
- Full LCD plus joystick
- Light LCD plus joystick
- Serial communication protocol bidirectional (compatible with ST MC Workbench GUI)
- Serial communication protocol fast unidirectional
- Drive system variables logging/displaying via:
  - SPI
  - DAC (DAC peripheral is not present in the STM32F103xx low or medium density; in this case, RC-filtered PWM signal option is available).

2 MC software development kit architecture

Figure 1 shows the system architecture. The Motor Control SDK has a four-layers structure:

- STM32Fxxx standard peripherals library and CMSIS library
- Motor Control Library
- Motor Control Application
- Demonstration user project

Figure 1. MC software library architecture



From the bottom layer upwards:

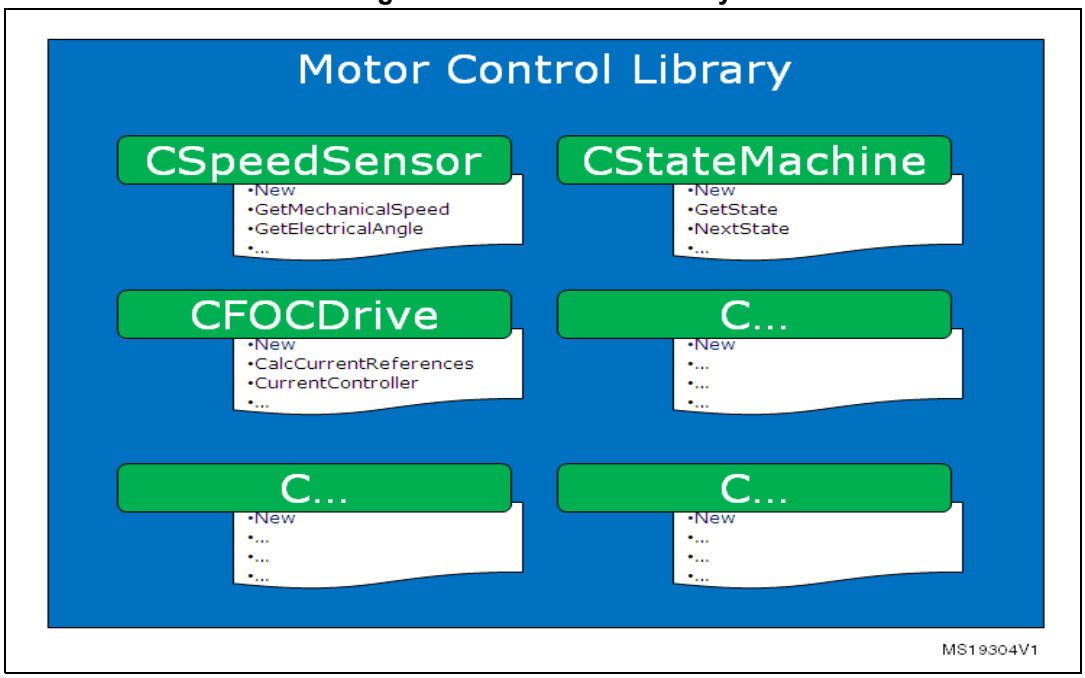
The STM32Fxxx standard peripherals library is an independent firmware package that contains a collection of routines, data structures and macros that cover the features of the STM32 peripherals. Version 3.5.0 of STM32F10x standard peripheral library is included in the MC SDK, version 1.0.0 is available for STM32F05xx, STM32F2xx and STM32F4xx, version 1.0.1 is available for STM32F30x. The STM32Fxxx standard peripherals library is CMSIS and MISRA-C compliant. Visit www.st.com/stm32 for complete documentation.

The Motor Control Library is a wide collection of classes that describe the functionality of elements involved in motor control (such as speed sensors, current sensors, algorithms). Each class has an interface, which is a list of methods applicable to objects of that class. *Figure 2* is a conceptual representation of the library.

Two distributions of the Motor Control Library are available:

- Web distribution, available free of charge at www.st.com, where the Motor Control Library is provided as a compiled .lib file.
- Confidential distribution, available free of charge on demand by contacting your nearest ST sales office or support team. Source files of classes are provided, with the exception of ST protected IPs, which are provided as compiled object files. Source files of protected IPs can also be provided free of charge to ST partners upon request. Contact your nearest ST office or support team for further information.

Figure 2. Motor control library



The Motor Control Library uses the lower STM32Fxxx Standard Peripheral Library layer extensively for initializations and settings on peripherals. Direct access to STM32 peripheral registers is preferred when optimizations (in terms of execution speed or code size) are required. More information about the Motor Control Library, its classes and object oriented programming can be found in the *Advanced developers guide for STM32F0x/F100xx/F103xx and STM32F2xx/F30x/F4xx MCUs PMSM single/dual FOC library (*UM1053).

The Motor Control Application (MCA) is an application that uses the Motor Control Library in order to accomplish commands received from the user level. This set of commands is specified in its Application Programming Interface (API).

During its boot stage, the MCA creates the required controls in accordance with actual system parameters, defined in specific .h files that are generated by the ST MC Workbench GUI (or manually edited). It coordinates them continuously for the purpose of accomplishing received commands, by means of tasks of proper priority and periodicity. More information about the MCA can be found in User manual *STM32F0x/F100xx/F103xx/F2xx/F30x/F4xx PMSM single/dual FOC SDK v4.0 (*UM1052), and details on tasks and implemented algorithms in the *Advanced developers guide for STM32F0x/F100xx/F103xx and STM32F2xx/F30x/F4xx MCUs PMSM single/dual FOC library (*UM1053).

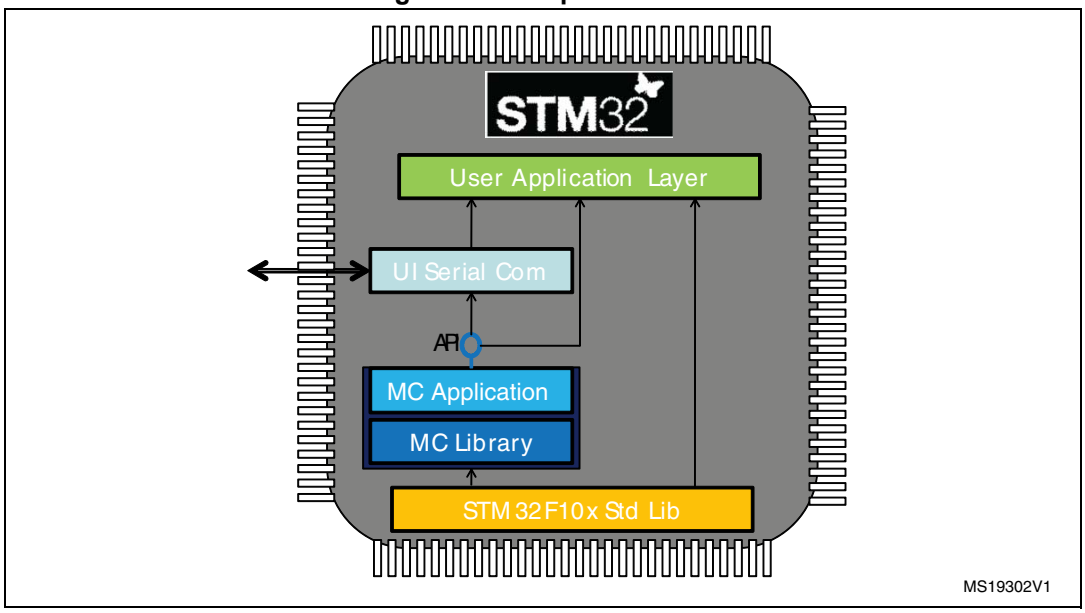
At the user level, a user project has been implemented to demonstrate how to interact with the MC API to successfully achieve the execution of commands. Depending on definable options, the user project can act as a Human Interface Device (using a joystick, buttons and LCD screens), as a command launcher through a serial communication protocol, as a data logging/displaying utility, or as a tuning tool.

Two versions of this user project are available (STM32F103xx and STM32F2xx only). One is based on FreeRTOS, the other is not. The demonstration user project can be dismantled and replaced by the user application layer, or quite easily integrated, as shown in *Figure 3*: the user application layer uses the STM32F10x Standard Library for its own purposes and sends commands directly to the MC API, while the serial communication interface provided

in the demonstration user project dispatches commands received from the outer world to the MC API.

More information about the modules integrated with the demonstration user project, such as serial communication protocol, drive variables monitoring through DAC / SPI, HID (generically called 'UI library') and a description of LCD screens can be found in *STM32F05xx/STM32F100xx/STM32F103xx/STM32F2xx/STM32F30x or STM32F4xx permanent-magnet synchronous motor single/dual FOC software library V4.0 (*UM1052).

Figure 3. Example scenario



Please contact your nearest ST sales office or support team to obtain the STM32F103xx/STM32F100xx PMSM single/dual FOC SDK v3.0 installation file. For a complete list of ST offices and distributors, refer to the ST website http://www.st.com.

3 Documentation architecture

3.1 Where to find the information you need

Technical information about the MC SDK is distinguished and organized by topic. The following is a list of the available documents and the subjects they cover:

- *STM32F05xx/STM32F100xx/STM32F103xx/STM32F2xx/STM32F30x or STM32F4xx permanent-magnet synchronous motor single/dual FOC software library V4.0* (UM1052). This provides the following:
  - Features
  - Architecture
  - Workspace
  - Customization processes
  - Overview of algorithms implemented (FOC, current sensors, speed sensors, embedded analog topologies supported)
  - MC API
  - Demonstrative user project
  - Demonstrative LCD user interface
  - Demonstrative serial communication protocol
- *Advanced developers guide for STM32F05xx/STM32F100xx/103xx and STM32F2xx/STM32F30x/STM32F4xx MCUs PMSM single/dual FOC library* (UM1053). This provides the following:
  - Object oriented programming style used for developing the MC library
  - Description of classes that belong to the MC library
  - Interactions between classes
  - Description of tasks of the MCA
- MC library source documentation (Doxygen-compiled HTML file). This provides a full description of the public interface of each class of the MC library (methods, parameters required for object creation).
- MC Application source documentation (Doxygen-compiled HTML file). This provides a full description of the classes that make up the MC API.
- User Interface source documentation (Doxygen-compiled HTML file). This provides a full description of the classes that make up the UI Library.
- STM32F0xx, STM32F10x, STM32F2xx, STM32F30x or STM32F4xx Standard Peripherals Library source documentation (Doxygen-compiled HTML file).
- ST MC Workbench GUI documentation. This is a field guide that describes the steps and parameters required to customize the library, as shown in the GUI.
- In-depth documentation about particular algorithms (sensorless position/speed detection, flux weakening, MTPA, feed-forward current regulation).

Please contact your nearest ST sales office or support team to obtain the documentation you are interested in if it was not already included in the software package you received or available on the ST web site (*www.st.com*).

3.2 Related documents

Available from www.arm.com

- Cortex™-M0 Technical Reference Manual, available from: *http://infocenter.arm.com/help/topic/com.arm.doc.ddi0432c/DDI0432C_cortex_m0_r0p0_trm.pdf*
- Cortex™-M3 Technical Reference Manual, available from: *http://infocenter.arm.com/help/topic/com.arm.doc.ddi0337e/DDI0337E_cortex_m3_r1p1_trm.pdf*
- Cortex™-M4 Technical Reference Manual, available from: *http://infocenter.arm.com/help/topic/com.arm.doc.ddi0439c/DDI0439C_cortex_m4_r0p1_trm.pdf*

Available from www.st.com or your STMicroelectronics sales office

- STM32F050x and STM32F051x datasheets
- STM32F100xx datasheet
- STM32F103xx datasheet
- STM32F20x and STM32F21x datasheets
- STM32F40x and STM32F41x datasheets
- STM32F050x and STM32F051x reference manual (RM0091)
- STM32F100xx reference manual (RM0041)
- STM32F103xx reference manual (RM0008)
- STM32F20x and STM32F21x reference manual (RM0033)
- STM32F40x and STM32F41x reference manual (RM0090)
- STM32F103xx AC induction motor IFOC software library V2.0 user manual (UM0483)
- STM32 and STM8 Flash Loader demonstrator user manual (UM0462)
- STM32F302xB/C datasheet
- STM32F303xB/C datasheet
- STM32F30x reference manual (RM316)

4 Revision history

Table 2. Document revision history

Date	Revision	Changes
19-Apr-2011	1	Initial release.
24-May-2011	2	Added references for web and confidential distributions of STM32 FOC PMSM SDK v3.0
05-Dec-2013	3	Updated *Table 1: Applicable products*. Updated Motor control library features. Added references to new UM1052 and UM1053.
22-May-2014	4	Updated *Introduction* and *Section 1: Motor control library features*. Updated references to new UM1052 and UM1053.