



Industrial
Summit 2021
Shenzhen, China
POWERING YOUR INNOVATION



X-CUBE-MCSDK Update

RENDONG WANG

NA ZHANG

MDG Group

STMicroelectronics, AP Region

Agenda

1 STM32 for Motor Control

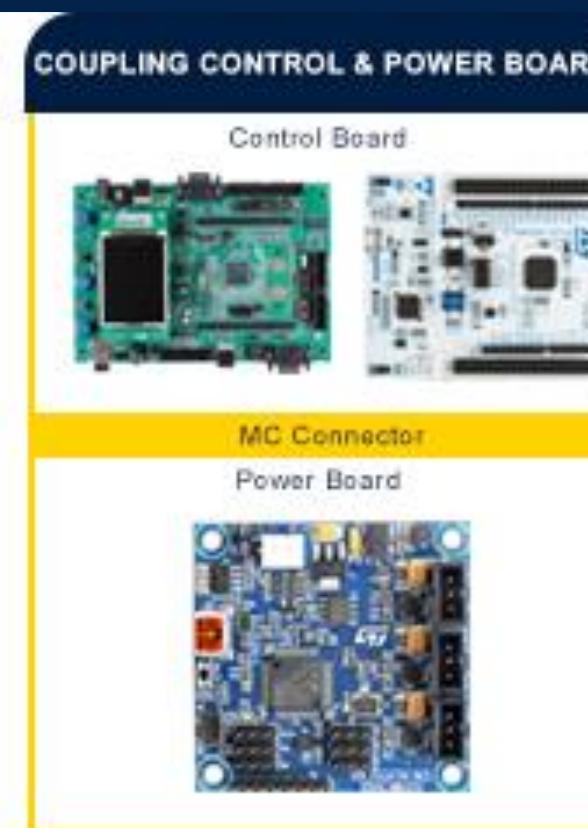
5 Q&A

2 X-CUBE-MCSDK Update

3 X-CUBE-MCSDK 5.Y.3 Demo

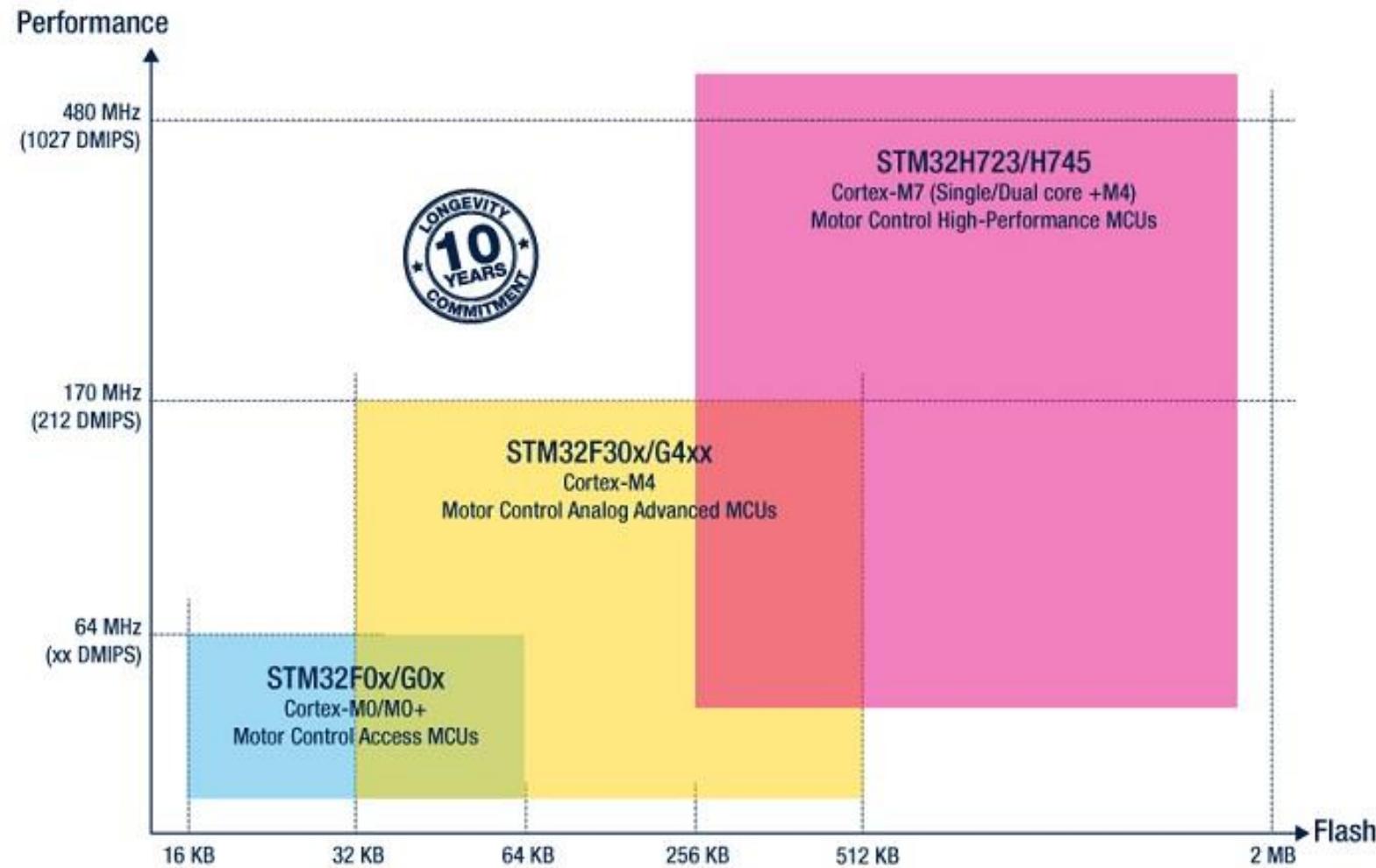
4 Highlights on X-CUBE-MCSDK

STM32 for Motor Control



All our boards are listed in the [ST-MC-SUITE](#) tool

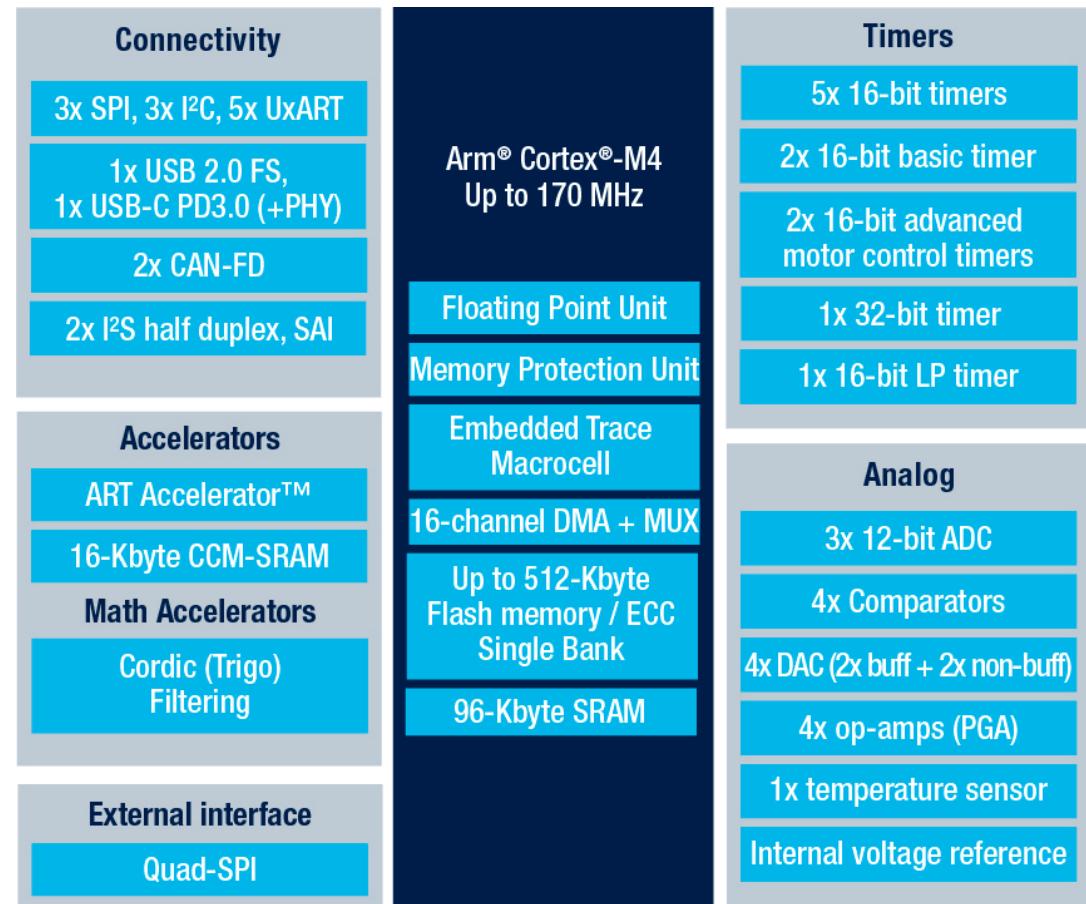
STM32 for Motor Control



STM32G491 Block Diagram

Access line [32KB .. 512KB]

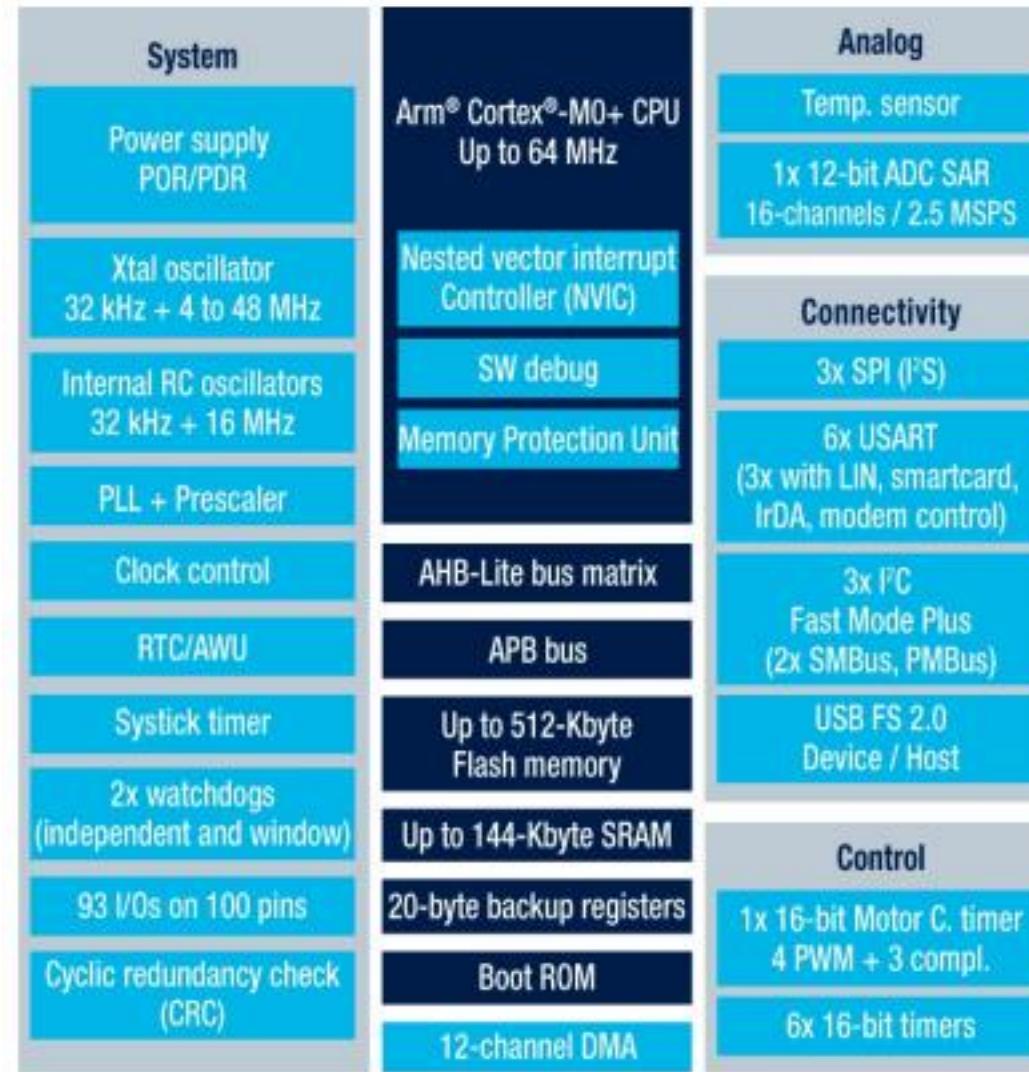
- 32-bit Arm Cortex-M4 core with FPU
- ART + CCM-SRAM + Mathematic Accelerators
- Single Bank Flash with ECC
- SRAM with Parity bit
- +/- 1% internal clock
- 1.72 to 3.6V power supply
- Up to 125°C



- Advanced Motor Control timers
- Rich Advanced Analog
- CAN Flexible Data rate
- USB-C Power Delivery3.0
- Advanced Security and Safety features
- Robustness: highest level 5 / FTB/ESD - IEC 61000-4-4

STM32 G0 Value Line-STM32G030

- 32-bit Arm Cortex-M0+ core
- 2.0 to 3.6V power supply
- RAM maximization
- 1% internal clock
- Direct Memory Access (DMA)
- Communication peripherals

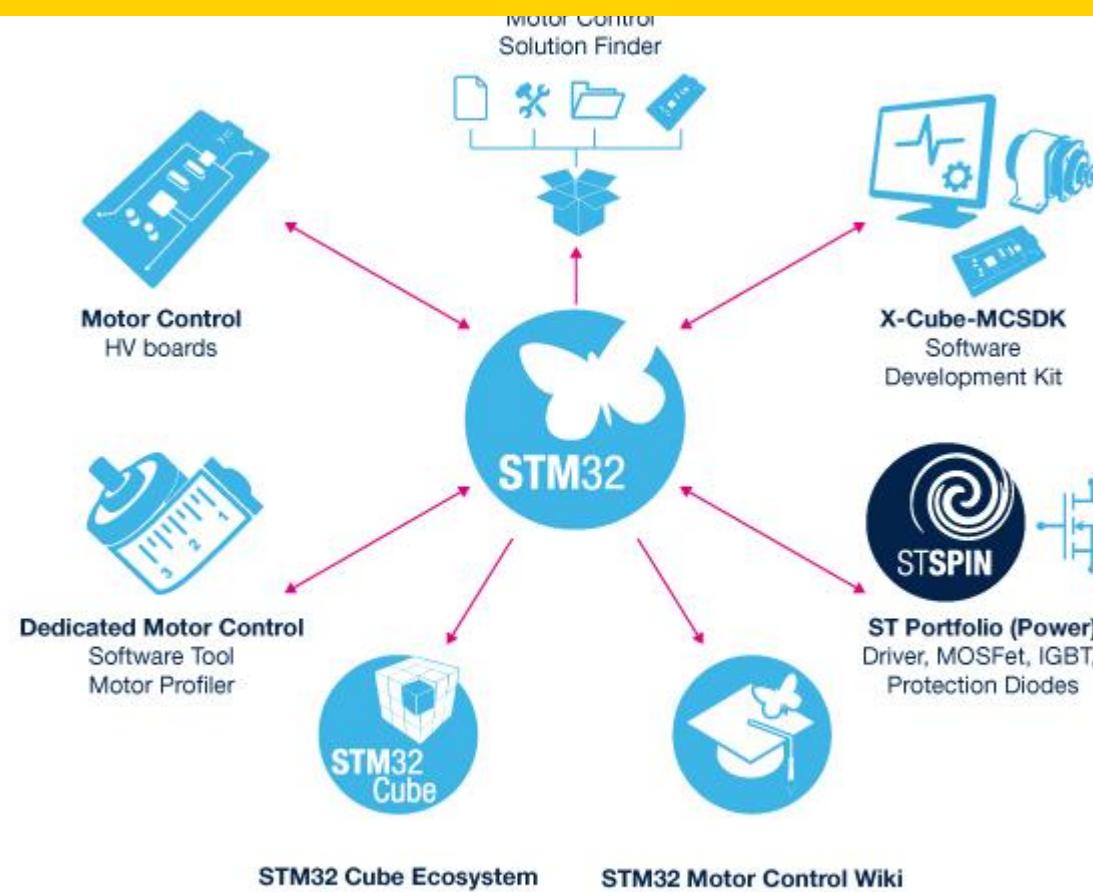


- Timers
- Real-time Clock
- I/O ports maximization
- 12-bit Ultra-fast ADC
- Safety features

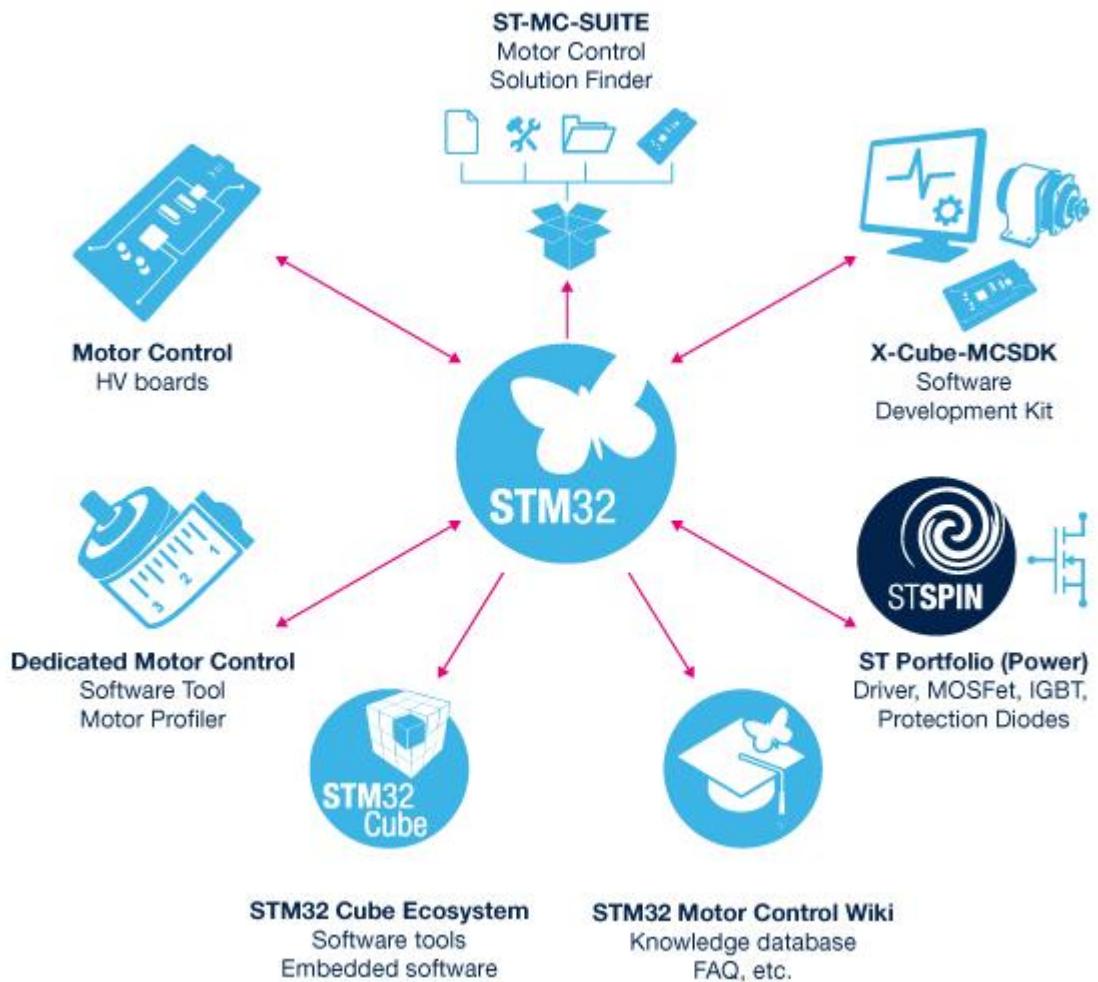
STM32H7 Features for Motor Control

Features	STM32H723/733/725/735/730	Benefit
Core	Cortex M7	Performance and efficiency
FPU	yes	Performance and efficiency
MPU	yes	Safety
Freq CPU max	550MHz	Performance and efficiency
DMIPS	1177	Performance and efficiency
Flash / SRAM data size	128kB to 1MB/564kB	Performance and integration/cost
Including: ICTM/DCTM RAM	Up to 256kB(configurable) / 128kB	Performance and efficiency
Error Code Correction	SECDED on full memory map	Safety
ADC SAR	2x16bit 3.6Msps, 1x12bit 5Msps	Efficiency
Other Analog	2xcomp, 2xPGA, 2xDAC, 1xDFSDM	Integration/cost
Advanced Motor Control Timer	2x(275MHz)	Performance and efficiency
Cache and Accelerator	32kB+32kB L1 cache Graphic, CORDIC, FMAC, Cypro*	Performance and efficiency
Security Services (SFI and SB-SFU)	Yes*	System Integrity
Package	VFQFPN68 LQFP100/144/176 BGA100/144/169/176 WLCSP11	Cost/Integration/flexibility
Max Temperature range °C	[-40 .. +125] T _j max 140°C	Integration and cost

X-CUBE-MCSDK Update

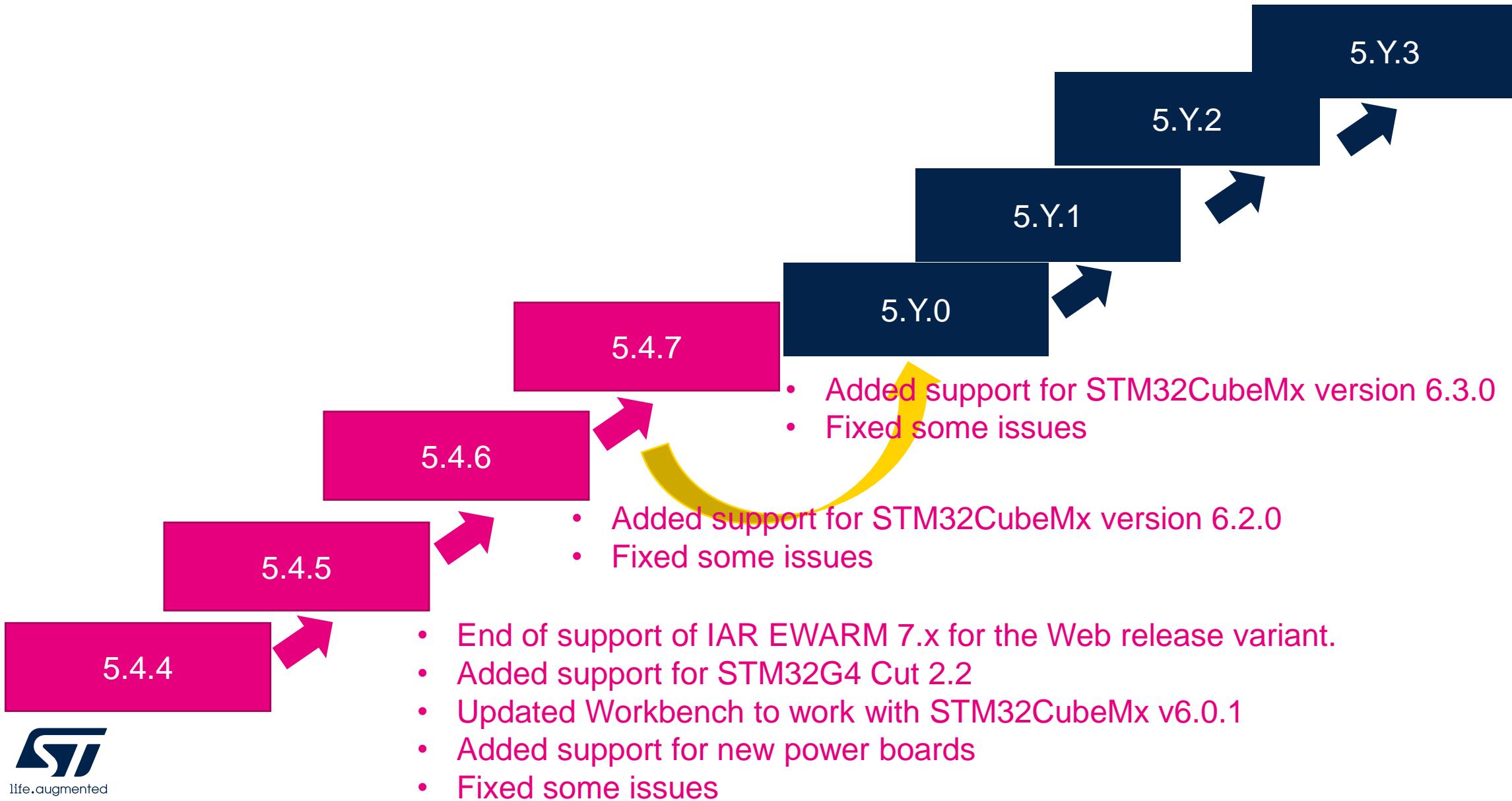


STM32 Ecosystem for Motor Control



- [X-CUBE-MCSDK](#)
- [ST-MC-SUITE](#)
- **Motor Profiler**
- [STM32 Cube Ecosystem](#)
- [STM32 Motor Control Wiki](#)

Update information since X-CUBE-MCSDK5.4.4



X-CUBE-MCSDK 5.Y(1/2)



- Introduced a version of ST Motor Pilot, the new monitoring tool of the X-CUBE_MCSDK.
- Removed the legacy Monitor from the Workbench
- A brand new communication protocol has been implemented and replace the legacy version.

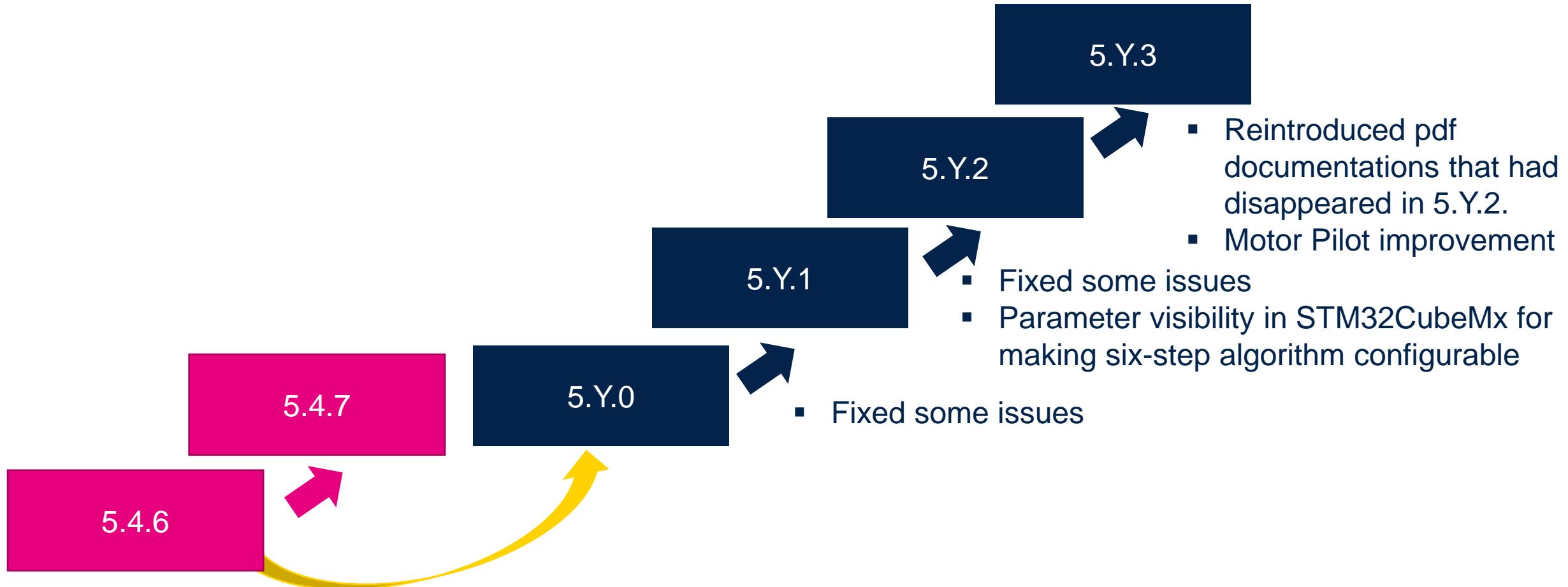
For PMSM/BLDC Motors:

- Discontinuous PWM (aka. Two-phase modulation)
- Over Modulation can be activated from the Workbench, in the Firmware Drive Management / Additional Features tab
- Single Shunt with Phase Shift.
- Circle Limitation VD, a better variant of the circle limitation algorithm
- Added support for the STSPIN32G4 devices
- Added support for new boards
- Added support for the EVALSTDRIEVE101 power board
- Added support of the HSI clock source for STM32L452 and STM32L476 devices.
- **Removed support for the STM32F1 devices series.(PFC using F1 is not supported anymore accordingly)**
- Six Step examples improvements

For Asynchronous Motors:

- ACIM motors are supported as two examples featuring FOC sensorless and V/f (scalar) modes. A graphical PC tool is delivered with the SDK to help configuring this example: ACIM GUI. The two examples are designed for NUCLEO-G431RB + STEVAL-IHM023V3 configurations.

X-CUBE-MCSDK 5.Y(2/2)





Motor Control – SDK Roadmap at a glance

SDK Product support

SDK V5.4	SDK V5.Y	SDK V6.0	SDK 6.1	SDK 6.x
STM32H7	STM32G4	STSPIN32G4		

FW solutions (in SDK)

Position Control	Discontinuous PWM	New MC Workbench + New FW lib architecture: FOC & 6-STEP	New FOC Sensorless algorithms	ACIM
Dual Drive G4	Single shunt – Phase shift	Over modulation		SwR Motor

FW examples (outside SDK)

6-Step V1 (STM32G4, STSPIN32F0)	6-Step V2		Class B (G4)	Other SW IPs
	ACIM (G4)		3x MC-FOC (G4)	

SW Tools

	New Motor Pilot (variable monitoring)	New Workbench v2.0	New MC Profiler v2.0 (inside MC FW)	

Recent key features

5.Y.3
16-Sep-2021

Q1 22

Q2 22

Beyond

Product support

STM32F3	STM32G4
---------	---------

STSPIN32

Core products

STM32F1	STM32G0
--------------------	---------

STM32F0

Cost efficient

STM32F4/F7	STM32H7
------------	---------

High performance

STM32L4

Low Power

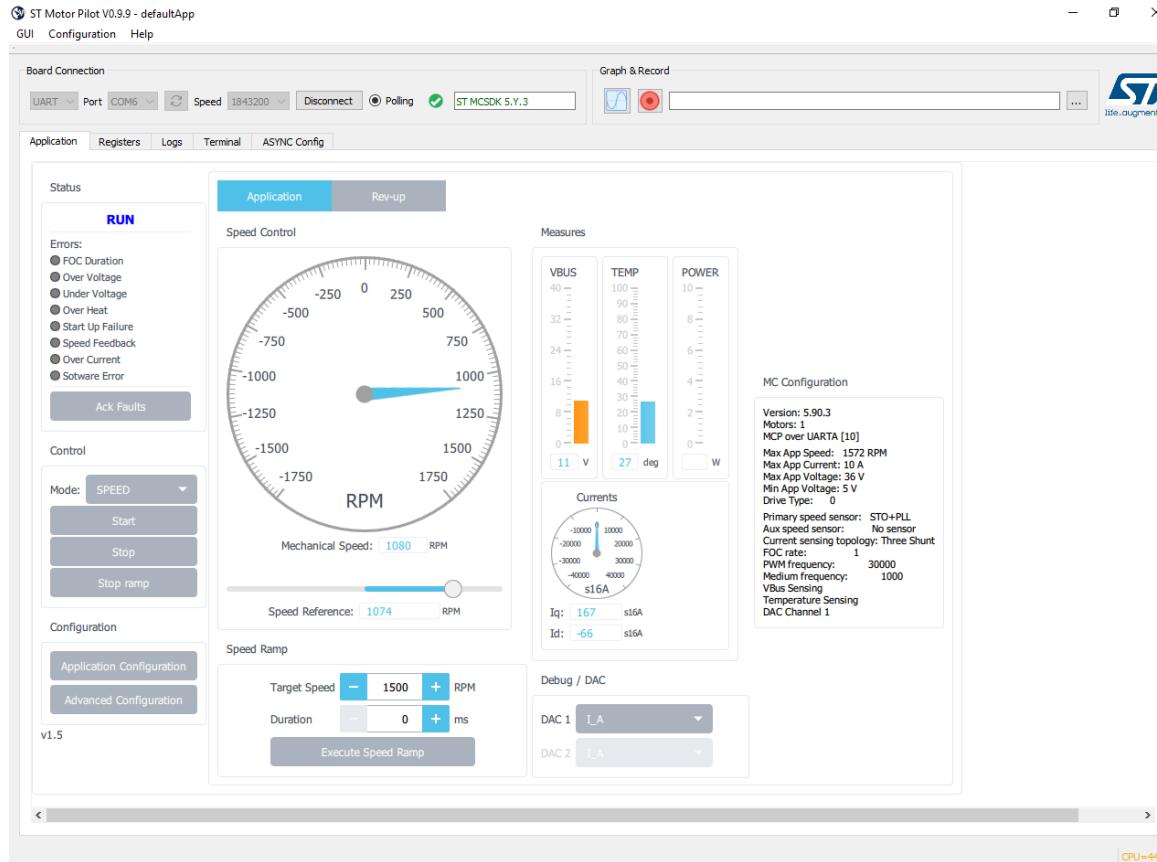
Reference documents

Doc Number	Title
<u>UM2374</u>	Getting started with STM32 motor control SDK v5.0
<u>UM2392</u>	STM32 motor control SDK5.x - Firmware
<u>UM2380</u>	STM32 motor control SDK v5.2 tools
<u>AN5143</u>	How to migrate motor control application software from SDK v4.3 to SDK v5.x
<u>AN5166</u>	Guidelines for control and customization of power boards with STM32 MC SDK v5.0

X-CUBE-MCSDK 5.Y.3 Demo

STM32 MC Motor Pilot

Monitoring tool for STM32 motor-control applications

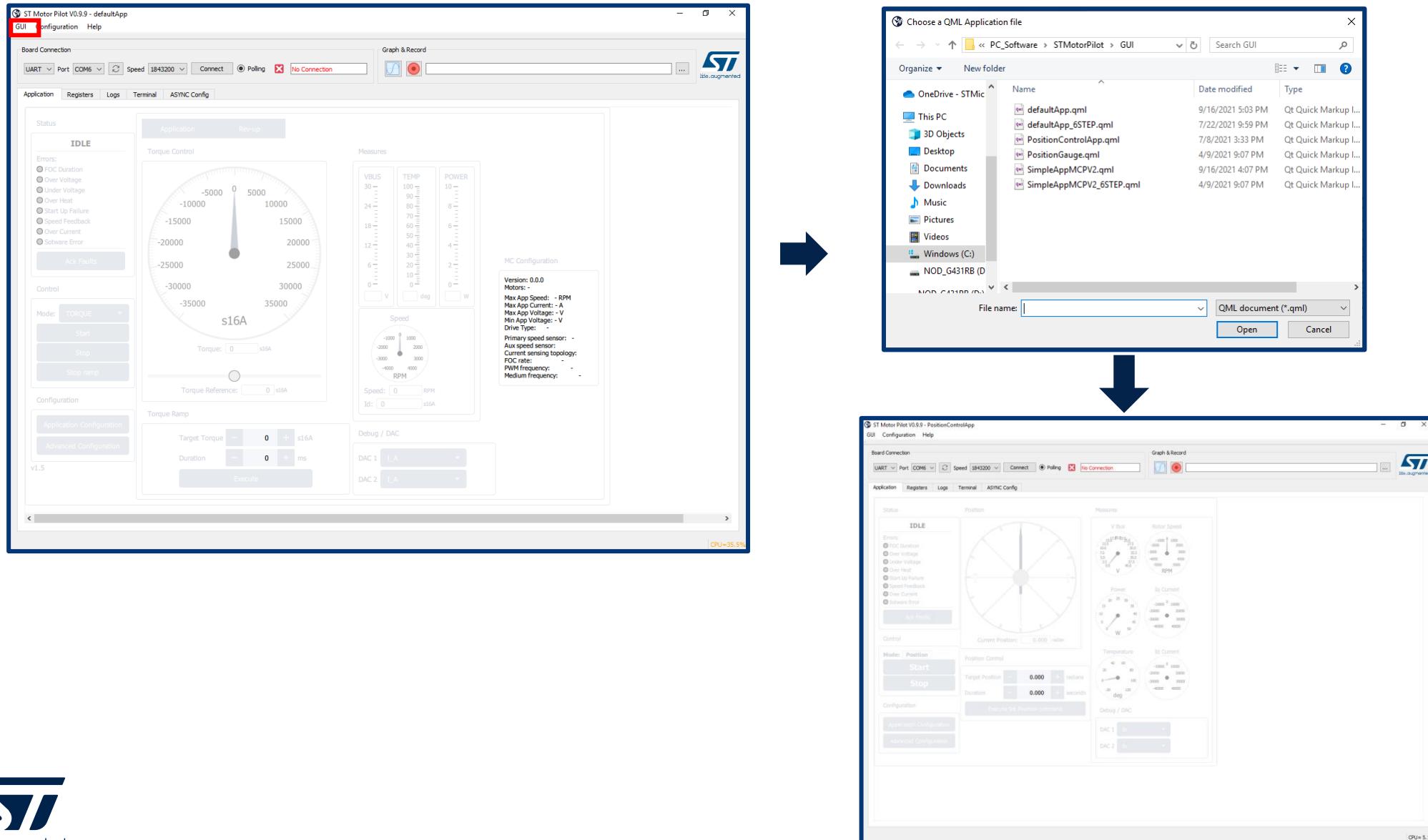


- Connects to MC applications built with the UI module through the serial port.
- Allows for controlling, monitoring, and tuning MC applications.
- Will replace the monitor part of STM32 MC workbench.

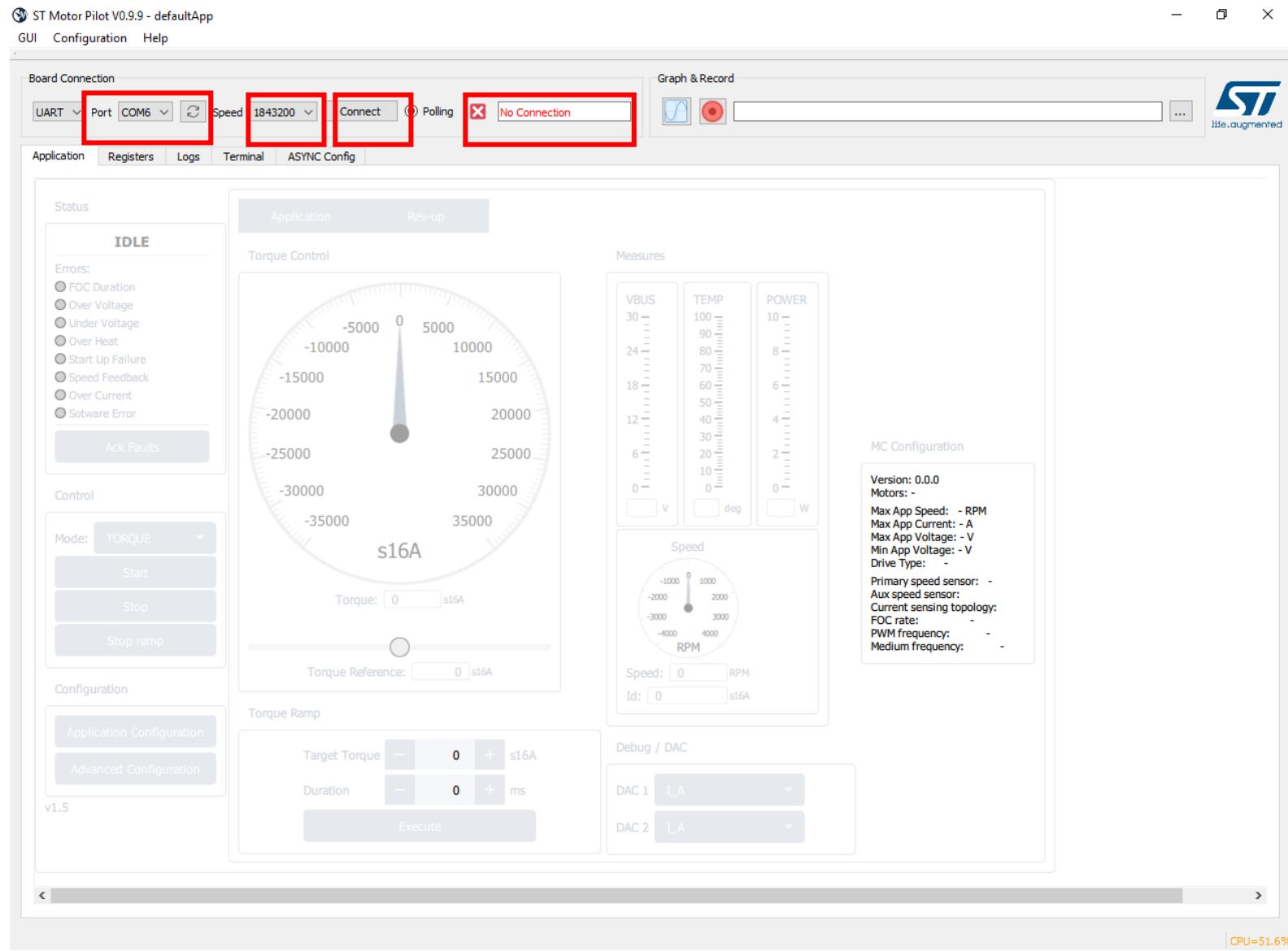
Runs on all three MCD target platforms: Windows, Mac, and Linux

- Enhanced plotting feature: the user can now plot most registers
- The user can easily customize GUI
 - To fit specific needs or to experiment with new firmware features
- Solid foundation to support future firmware features
 - ACIM, 6-Step, Sensorless Zero speed, enhanced debug features

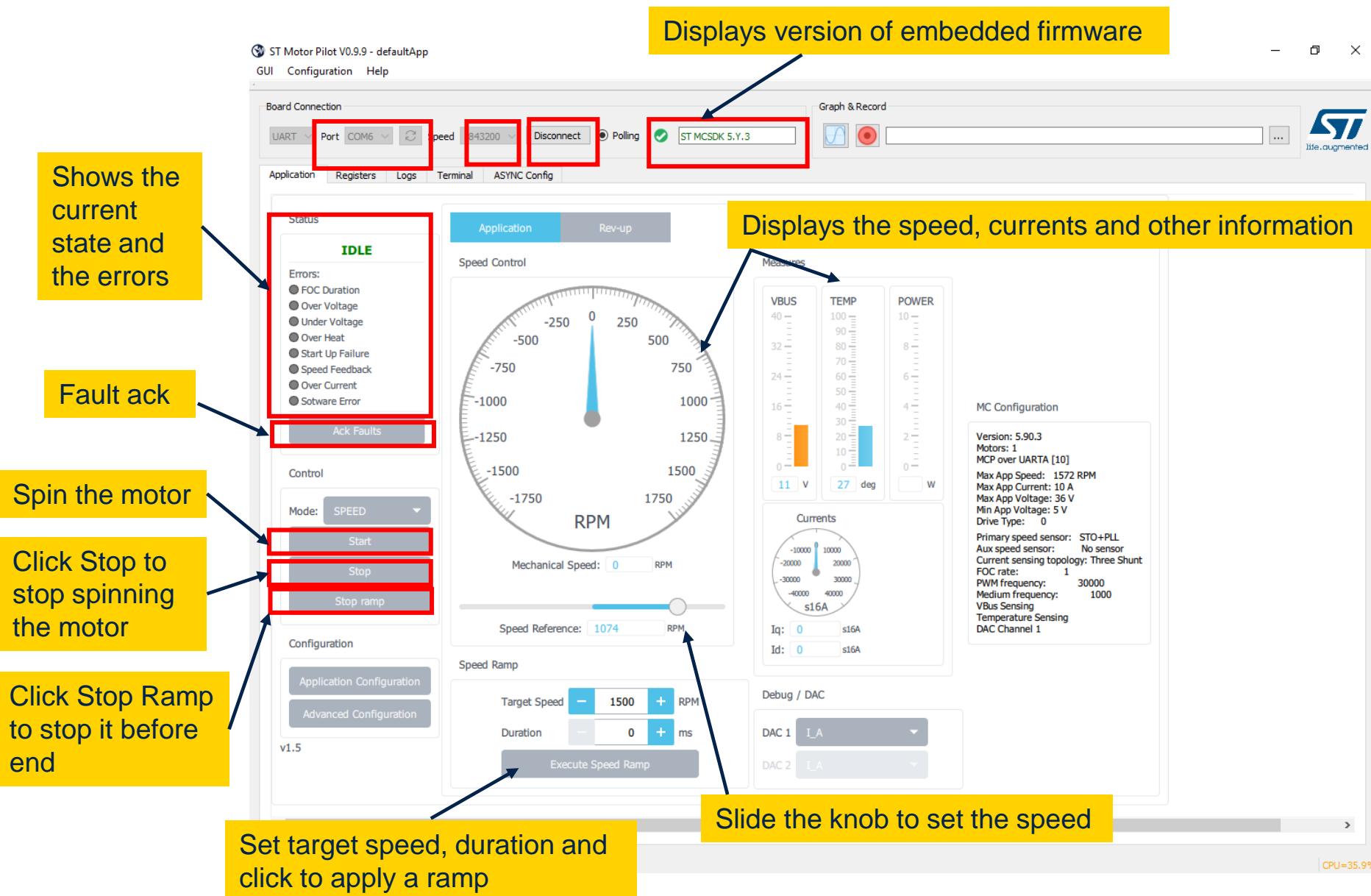
Several Motor Pilot applications



Quick start-up

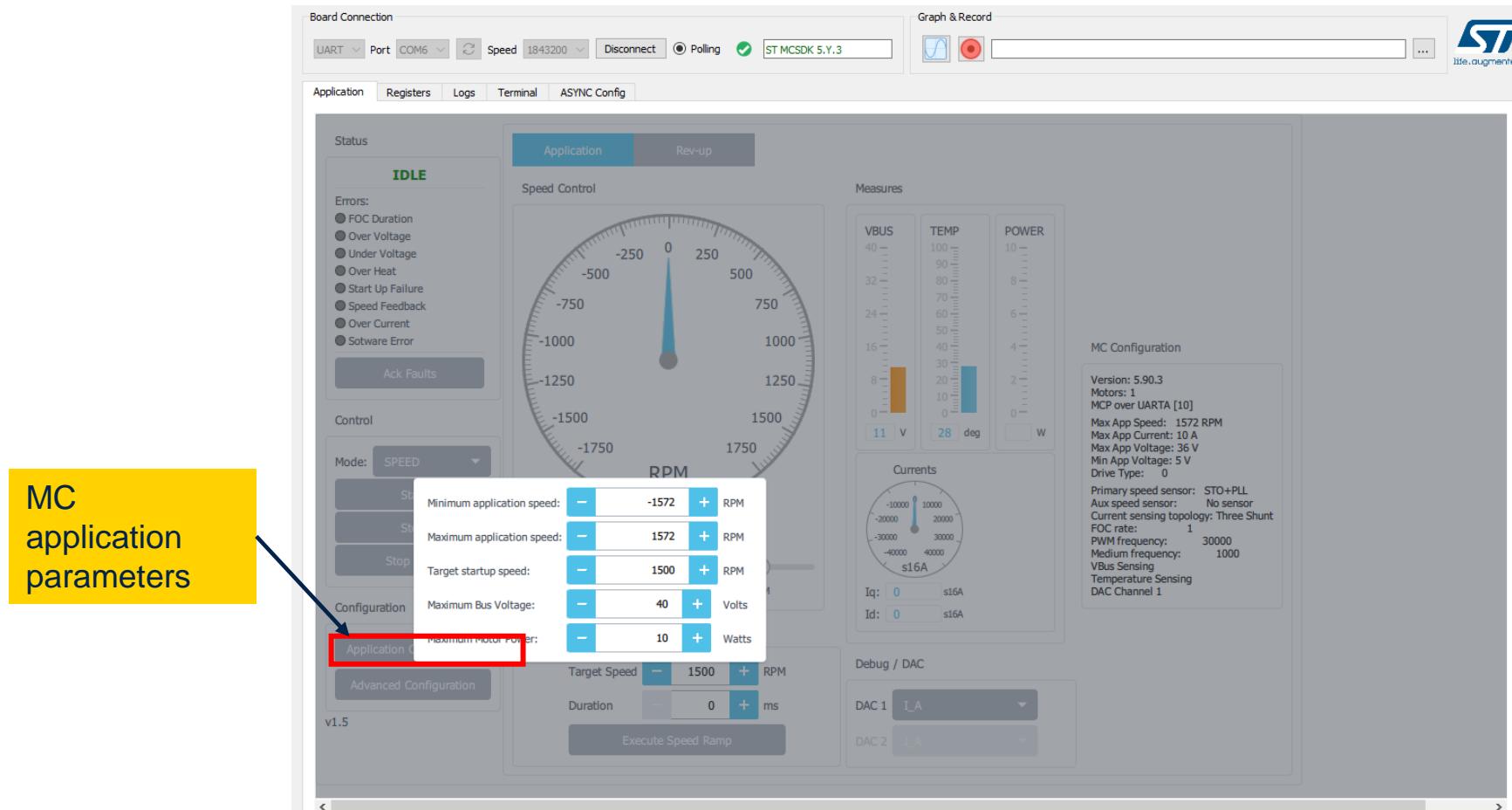


Control the motor



MC application parameters

STM32 MC Motor Pilot gets MC application parameters from the board and adapts widgets to them



MC application advanced settings(1)

Speed PID, Toque PID, Flux PID and Flux reference

Board Connection

UART Port COM6 Speed 1843200 Disconnect Polling ST MCSDK 5.Y.3

Graph & Record

Application Registers Logs Terminal ASYNC Config

Status: IDLE

Errors: FOC Duration, Over Voltage, Under Voltage, Over Heat, Start Up Failure, Speed Feedback, Over Current, Software Error

Ack Faults

Control: Mode: SPEED, Start, Stop, Stop ramp

Configuration: Application Configuration, Advanced Configuration

Application Rev-up

Speed Control: RPM (Mechanical Speed: 0 RPM, Speed Reference: 1058 RPM)

Measures: VBUS (11 V), TEMP (28 deg), POWER (0 W)

MC Configuration: Version: 5.90.3, Motors: 1, MCP over UARTA [10], Max App Speed: 1572 RPM, Max App Current: 10 A, Max App Voltage: 36 V, Min App Voltage: 5 V, Drive Type: 0, Primary speed sensor: STO+PLL, Aux speed sensor: No sensor, Current sensing topology: Three Shunt, FOC rate: 1, PWM frequency: 30000, Medium frequency: 1000, VBus Sensing, Temperature Sensing, DAC Channel 1

Speed Ramp: Target Speed 1500 RPM, Duration 0 ms, Execute Speed Ramp

Debug / DAC: DAC 1 1_A, DAC 2 1_A

v1.5

Click Advanced Configuration

MC application advanced settings(2)

Speed PID, Toque PID, Flux PID and Flux reference

The screenshot shows the ST MC SDK 5.Y.3 application interface with the following details:

- Board Connection:** UART Port COM6, Speed 1843200, Disconnect, Polling, ST MCSDK 5.Y.3.
- Graph & Record:** Buttons for Graph and Record.
- Advanced Configuration:** A red box highlights the following settings:
 - Speed PI regulator:**
 - Speed Kp: 2714
 - Speed Ki: 559
 - Kp divisor: 256
 - Ki divisor: 16384
 - Torque (Iq) PI regulator:**
 - Torque Kp: 3540
 - Torque Ki: 2360
 - Flux (Id) PI:**
 - Flux Kp: 3540
 - Flux Ki: 2360
- Status:** IDLE. Errors: FOC Duration, Over Voltage, Under Voltage, Over Heat, Start Up Failure, Speed Feedback, Over Current, Software Error.
- Control:** Mode: SPEED, Start, Stop, Stop ramp.
- Configuration:** Application Configuration, Advanced Configuration.
- Application:** Rev-up. Speed Control: RPM gauge (0-1750), Speed Reference: 1058 RPM. Measures: VBUS (11V), TEMP (28 deg), POWER (0W). Currents: Iq (0 s16A), Id (0 s16A).
- MC Configuration:** Version: 5.90.3, Motors: 1, MCP over UARTA [10]. Parameters: Max App Speed: 1572 RPM, Max App Current: 10 A, Max App Voltage: 36 V, Min App Voltage: 5 V, Drive Type: 0. Sensors: Primary speed sensor: STO+PLL, Aux speed sensor: No sensor, Current sensing topology: Three Shunt. FOC rate: 1, PWM frequency: 30000, Medium frequency: 1000, VBus Sensing, Temperature Sensing, DAC Channel 1.
- Speed Ramp:** Target Speed: 1500 RPM, Duration: 0 ms. Execute Speed Ramp button.
- Debug / DAC:** DAC 1: I_A, DAC 2: I_A.

MC application advanced settings(3)

State Observer, PLL parameters, CORDIC parameters and DAC settings

The screenshot shows the ST MC SDK 5.0.3 application interface with the following sections highlighted:

- Advanced Configuration (Red Box):** Contains tables for Currents_Speed and Observers, and a detailed description of the State Observer with PLL settings. The table data is as follows:

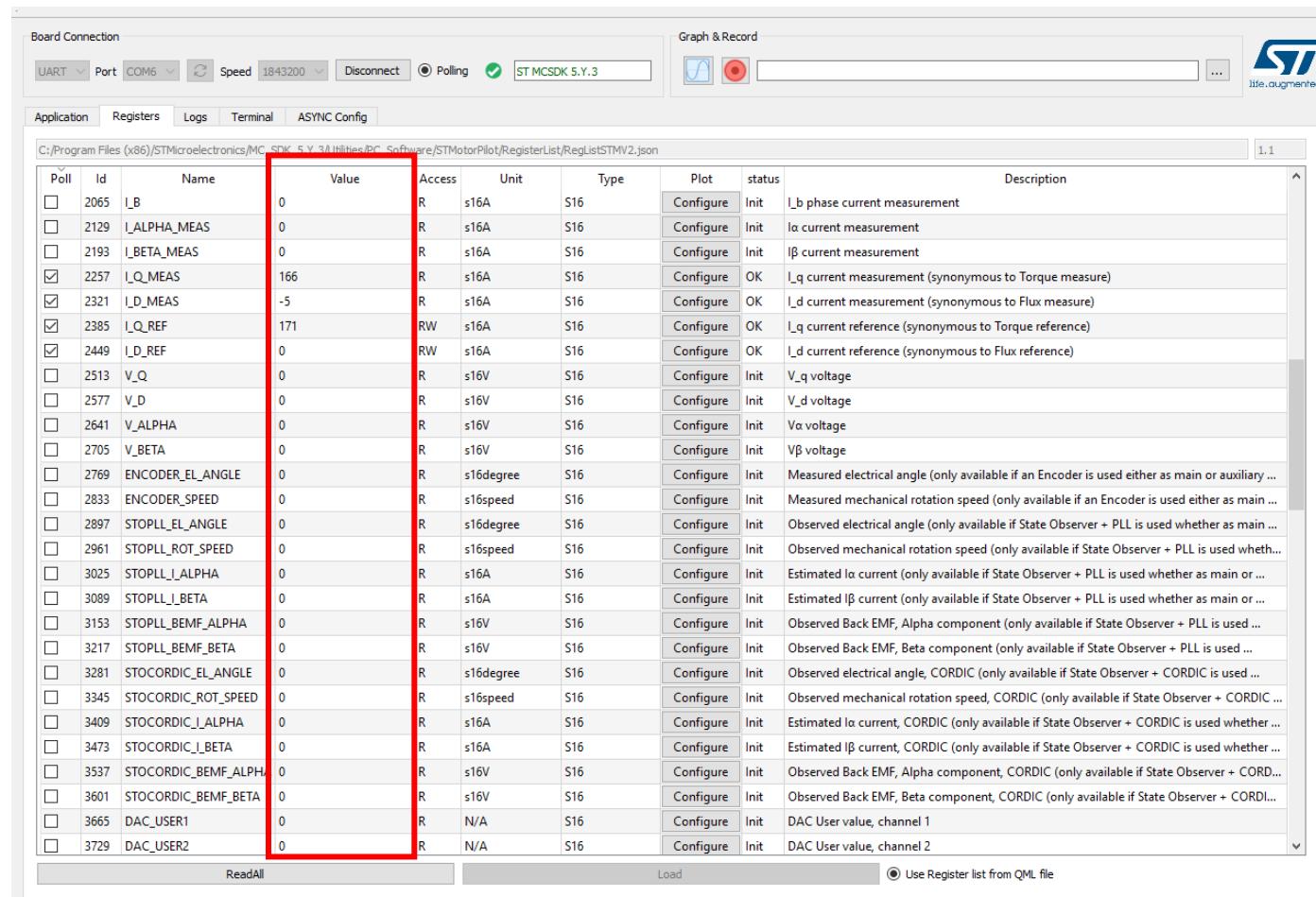
	Currents_Speed	Observers
C1:	-22528	+ N/A
C2:	31533	+ N/A
PLL Kp:	195	dpp/s16V
PLL Ki:	4	dpp/s16V

- Debug / DAC (Red Box):** Shows DAC 1 set to I_A and DAC 2 set to I_A. A yellow box labeled "DAC settings" is positioned below this section.

State Observer + PLL parameters
State Observer + CORDIC parameters

Viewing registers by pooling (low-speed plotting)

Clicking the Registers tab displays the list of all the registers, which can be streamed to the Pilot

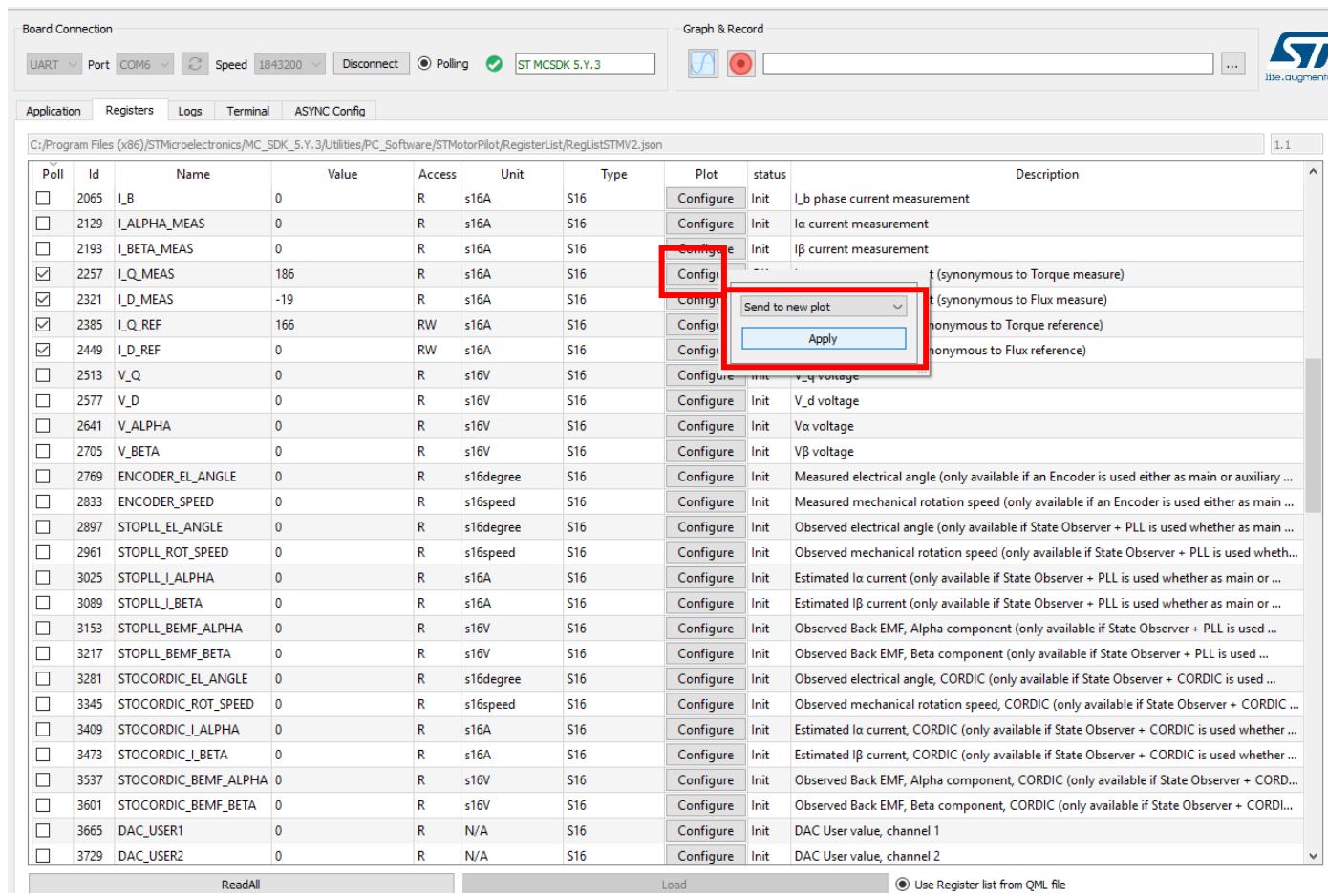


The screenshot shows the ST MotorPilot software interface with the 'Registers' tab selected. The table lists various registers with their details. The 'Value' column is highlighted with a red box.

Poll	Id	Name	Value	Access	Unit	Type	Plot	status	Description
	2065	I_B	0	R	s16A	S16	Configure	Init	I_b phase current measurement
	2129	I_ALPHA_MEAS	0	R	s16A	S16	Configure	Init	I α current measurement
	2193	I_BETA_MEAS	0	R	s16A	S16	Configure	Init	I β current measurement
<input checked="" type="checkbox"/>	2257	I_Q_MEAS	166	R	s16A	S16	Configure	OK	I q current measurement (synonymous to Torque measure)
<input checked="" type="checkbox"/>	2321	I_D_MEAS	-5	R	s16A	S16	Configure	OK	I d current measurement (synonymous to Flux measure)
<input checked="" type="checkbox"/>	2385	I_Q_REF	171	RW	s16A	S16	Configure	OK	I q current reference (synonymous to Torque reference)
<input checked="" type="checkbox"/>	2449	I_D_REF	0	RW	s16A	S16	Configure	OK	I d current reference (synonymous to Flux reference)
	2513	V_Q	0	R	s16V	S16	Configure	Init	V q voltage
	2577	V_D	0	R	s16V	S16	Configure	Init	V d voltage
	2641	V_ALPHA	0	R	s16V	S16	Configure	Init	V α voltage
	2705	V_BETA	0	R	s16V	S16	Configure	Init	V β voltage
	2769	ENCODER_EL_ANGLE	0	R	s16degree	S16	Configure	Init	Measured electrical angle (only available if an Encoder is used either as main or auxiliary ...)
	2833	ENCODER_SPEED	0	R	s16speed	S16	Configure	Init	Measured mechanical rotation speed (only available if an Encoder is used either as main ...)
	2897	STOPLL_EL_ANGLE	0	R	s16degree	S16	Configure	Init	Observed electrical angle (only available if State Observer + PLL is used whether as main ...)
	2961	STOPLL_ROT_SPEED	0	R	s16speed	S16	Configure	Init	Observed mechanical rotation speed (only available if State Observer + PLL is used whether as main ...)
	3025	STOPLL_I_ALPHA	0	R	s16A	S16	Configure	Init	Estimated I α current (only available if State Observer + PLL is used whether as main or ...)
	3089	STOPLL_I_BETA	0	R	s16A	S16	Configure	Init	Estimated I β current (only available if State Observer + PLL is used whether as main or ...)
	3153	STOPLL_BEMF_ALPHA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Alpha component (only available if State Observer + PLL is used ...)
	3217	STOPLL_BEMF_BETA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Beta component (only available if State Observer + PLL is used ...)
	3281	STOCORDIC_EL_ANGLE	0	R	s16degree	S16	Configure	Init	Observed electrical angle, CORDIC (only available if State Observer + CORDIC is used ...)
	3345	STOCORDIC_ROT_SPEED	0	R	s16speed	S16	Configure	Init	Observed mechanical rotation speed, CORDIC (only available if State Observer + CORDIC ...)
	3409	STOCORDIC_I_ALPHA	0	R	s16A	S16	Configure	Init	Estimated I α current, CORDIC (only available if State Observer + CORDIC is used whether ...)
	3473	STOCORDIC_I_BETA	0	R	s16A	S16	Configure	Init	Estimated I β current, CORDIC (only available if State Observer + CORDIC is used whether ...)
	3537	STOCORDIC_BEMF_ALPHA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Alpha component, CORDIC (only available if State Observer + CORDIC ...)
	3601	STOCORDIC_BEMF_BETA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Beta component, CORDIC (only available if State Observer + CORDIC ...)
	3665	DAC_USER1	0	R	N/A	S16	Configure	Init	DAC User value, channel 1
	3729	DAC_USER2	0	R	N/A	S16	Configure	Init	DAC User value, channel 2

Plotting registers by pooling (low-speed plotting)

Register selection for ASYNC plot available

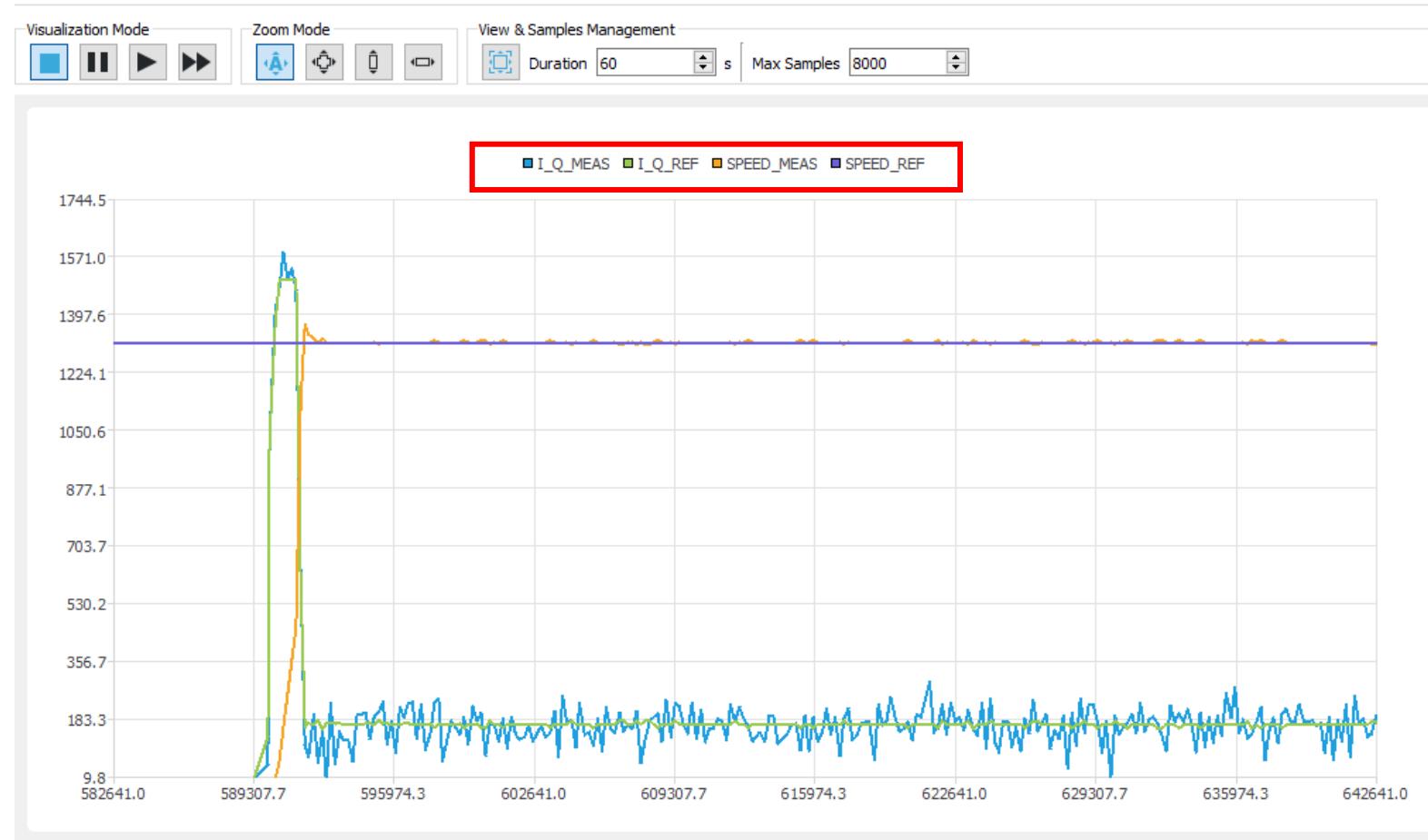


The screenshot shows the ST MotorPilot software interface. The top bar includes 'Board Connection' (UART, Port COM6, Speed 1843200, Disconnect, Polling, ST MCSDK 5.Y.3) and 'Graph & Record' (Plot, status, ...). Below is a navigation bar with tabs: Application, Registers (selected), Logs, Terminal, ASYNC Config. The main area displays a table of registers from 'C:\Program Files (x86)\STMicroelectronics\MC_SDK_5.Y.3\Utilities\PC_Software\STMotorPilot\RegisterList\RegListSTMV2.json'. The table columns are: Poll, Id, Name, Value, Access, Unit, Type, Plot, status, and Description. A red box highlights the 'Plot' column for register 2257 (I_Q_MEAS) and the dropdown menu 'Send to new plot' with the 'Apply' button. The table contains numerous entries, including various current and voltage measurements, encoder angles, and Back EMF components.

Poll	Id	Name	Value	Access	Unit	Type	Plot	status	Description
	2065	I_B	0	R	s16A	S16	Configure	Init	I_b phase current measurement
	2129	I_ALPHA_MEAS	0	R	s16A	S16	Configure	Init	I α current measurement
	2193	I_BETA_MEAS	0	R	s16A	S16	Configure	Init	I β current measurement
<input checked="" type="checkbox"/>	2257	I_Q_MEAS	186	R	s16A	S16	Configure	Init	I q current measurement (synonymous to Torque measure)
<input checked="" type="checkbox"/>	2321	I_D_MEAS	-19	R	s16A	S16	Configure	Init	I d current measurement (synonymous to Flux measure)
<input checked="" type="checkbox"/>	2385	I_Q_REF	166	RW	s16A	S16	Configure	Init	I q reference (synonymous to Torque reference)
<input checked="" type="checkbox"/>	2449	I_D_REF	0	RW	s16A	S16	Configure	Init	I d reference (synonymous to Flux reference)
	2513	V_Q	0	R	s16V	S16	Configure	Init	V q voltage
	2577	V_D	0	R	s16V	S16	Configure	Init	V d voltage
	2641	V_ALPHA	0	R	s16V	S16	Configure	Init	V α voltage
	2705	V_BETA	0	R	s16V	S16	Configure	Init	V β voltage
	2769	ENCODER_EL_ANGLE	0	R	s16degree	S16	Configure	Init	Measured electrical angle (only available if an Encoder is used either as main or auxiliary ...)
	2833	ENCODER_SPEED	0	R	s16speed	S16	Configure	Init	Measured mechanical rotation speed (only available if an Encoder is used either as main ...)
	2897	STOPPLL_EL_ANGLE	0	R	s16degree	S16	Configure	Init	Observed electrical angle (only available if State Observer + PLL is used whether as main ...)
	2961	STOPPLL_ROT_SPEED	0	R	s16speed	S16	Configure	Init	Observed mechanical rotation speed (only available if State Observer + PLL is used whether as main ...)
	3025	STOPPLL_I_ALPHA	0	R	s16A	S16	Configure	Init	Estimated I α current (only available if State Observer + PLL is used whether as main or ...)
	3089	STOPPLL_I_BETA	0	R	s16A	S16	Configure	Init	Estimated I β current (only available if State Observer + PLL is used whether as main or ...)
	3153	STOPPLL_BEMF_ALPHA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Alpha component (only available if State Observer + PLL is used ...)
	3217	STOPPLL_BEMF_BETA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Beta component (only available if State Observer + PLL is used ...)
	3281	STOCORDIC_EL_ANGLE	0	R	s16degree	S16	Configure	Init	Observed electrical angle, CORDIC (only available if State Observer + CORDIC is used ...)
	3345	STOCORDIC_ROT_SPEED	0	R	s16speed	S16	Configure	Init	Observed mechanical rotation speed, CORDIC (only available if State Observer + CORDIC ...)
	3409	STOCORDIC_I_ALPHA	0	R	s16A	S16	Configure	Init	Estimated I α current, CORDIC (only available if State Observer + CORDIC is used whether ...)
	3473	STOCORDIC_I_BETA	0	R	s16A	S16	Configure	Init	Estimated I β current, CORDIC (only available if State Observer + CORDIC is used whether ...)
	3537	STOCORDIC_BEMF_ALPHA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Alpha component, CORDIC (only available if State Observer + CORDIC ...)
	3601	STOCORDIC_BEMF_BETA	0	R	s16V	S16	Configure	Init	Observed Back EMF, Beta component, CORDIC (only available if State Observer + CORDIC ...)
	3665	DAC_USER1	0	R	N/A	S16	Configure	Init	DAC User value, channel 1
	3729	DAC_USER2	0	R	N/A	S16	Configure	Init	DAC User value, channel 2

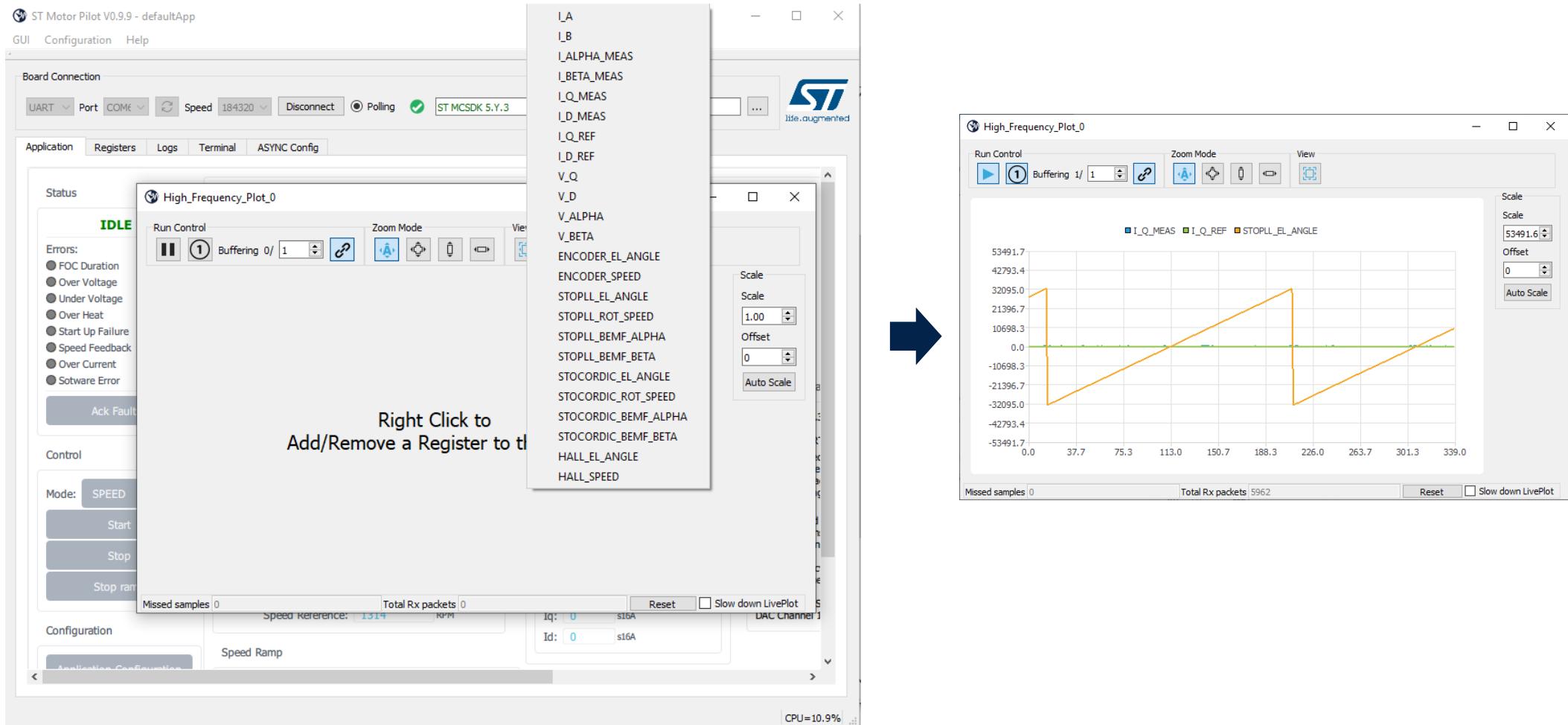
Buttons at the bottom include 'ReadAll', 'Load', and 'Use Register list from QML file'.

Plotting registers by pooling (low-speed plotting)



Datalog Service

Provides an oscilloscope-like display to plot high frequency tasks



Highlights on X-CUBE-MCSDK

Driving one or two different motors simultaneously (one for 6-Step)

- A PC software application to automatically measure electromechanical parameters of PMSM motors (**STM32F30x and STM32F4xx only**);
- LCD screen on EVAL boards **not supported**;
- Development Toolchains:
 - IAR Embedded Workbench for ARM (IAR Systems AB) v8.20.2. (**no support for v7.x.x**)
 - µVision® IDE for Arm® (Keil® MDK) v5.24.2
 - STM32CubeIDE v1.6.1
 - STM32CubeProgrammer 2.6.0

X-CUBE-MCSDK version 5.Y is released to provide insight on the directions that the development of the Motor Control SDK is following

- STM32F1 Family is not supported by X-CUBE-MCSDK version 5.Y. If you are using a STM32F1 MCU, please keep on using X-CUBE-MCSDK 5.4.x versions.
- Dual Drive is not supported by current X-CUBE-MCSDK version 5.Y. If you are using dual drive, please keep on using X-CUBE-MCSDK 5.4.x versions.
- Projects generated with previous X-CUBE-MCSDK versions will not load with the version 5.Y. Please stay with versions 5.4.x if you want to keep the compatibility.
- X-CUBE-MCSDK version 5.Y is not intended to reach the same level of maturity as legacy X-CUBE-MCSDK version 5.4.6

SPEED_UNIT (from release v5.4.0)

previous speed unit used by API functions that expect or return a speed (e.g., [MC_ProgramSpeedRampMotor1\(\)](#) or [MC_GetMecSpeedAverageMotor1\(\)](#))

- 01Hz (a tenth of a Hertz)

It is now possible to use other units for these functions. With release 5.4.0, the two new speed units are made available:

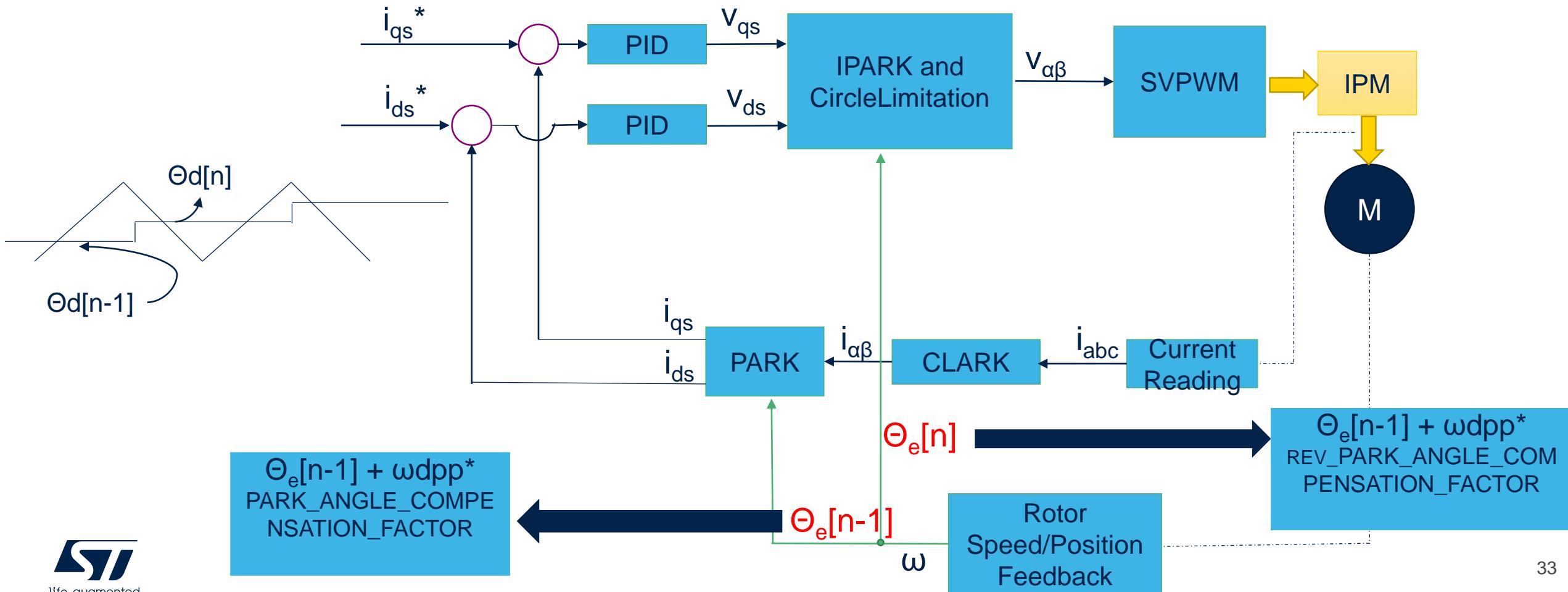
- RPM (Revolution Per Minute)
- 001Hz (a hundredth of a Hertz)

The choice of the speed unit is made at compile time by setting the SPEED_UNIT symbol to the proper value in file [mc_stm_types.h](#):

- [_RPM](#)
- [_001HZ](#)
- [_01HZ](#)

The SPEED_UNIT define is placed in a User Section and retained after project regeneration.

Electric Angle Estimation Compensation for Sensorless (from release v5.4.0)



Our technology starts with You



Find out more at www.st.com

© STMicroelectronics - All rights reserved.

ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.

For additional information about ST trademarks, please refer to www.st.com/trademarks.

All other product or service names are the property of their respective owners.