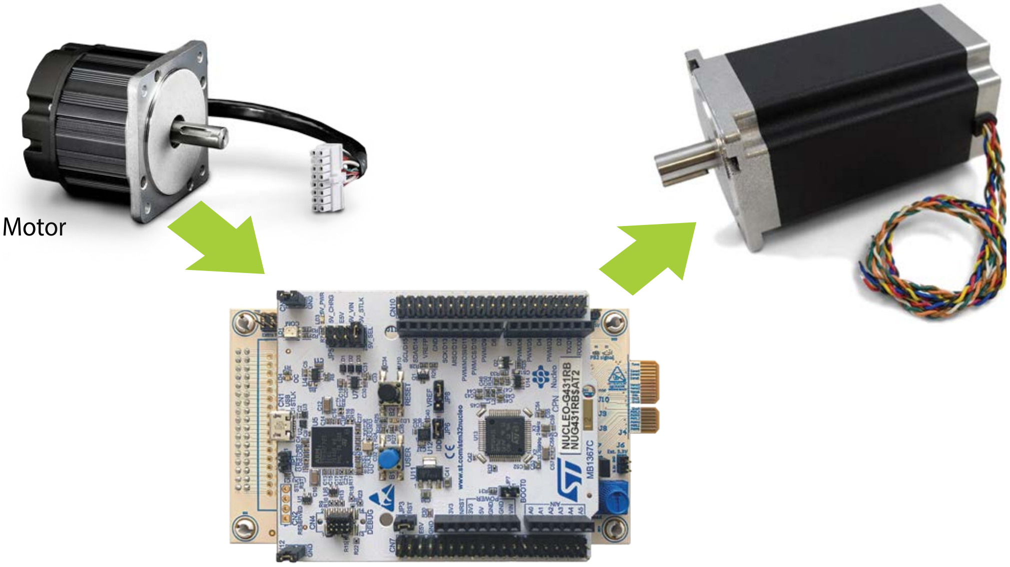
Commissioning a Motor for use with EPC motor drives that operate using ST Motor Control Workbench® Development Suite and EPC9147C – Rev.2.2

Revision 1.0

EFFICIENT POWER CONVERSION

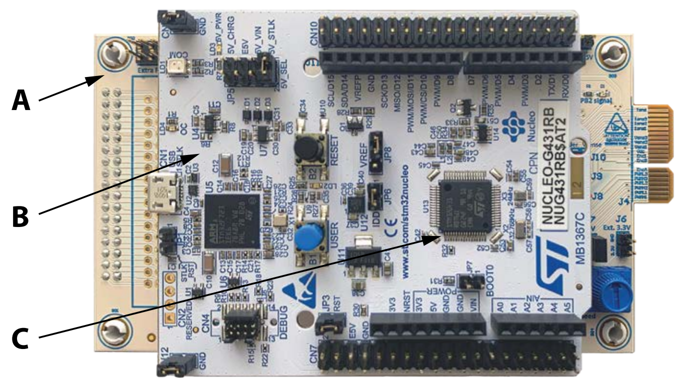
OVERVIEW OF THE PROCESS

- Background
- Equipment needed
- Measuring the motor parameters
- Inputting the motor parameters into ST Motor Control Development Suite
- Generating the control firmware:
  - Compiling
  - Build
  - Flash
- Operating the motor drive system



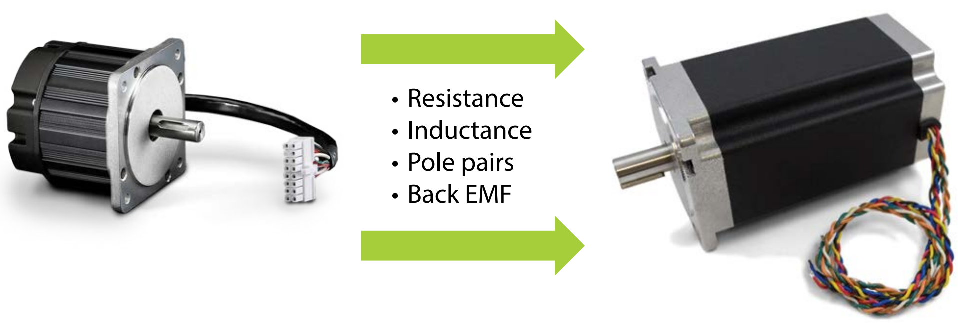
CONTROLLER BOARD BACKGROUND

- Process is for **EPC9147C Only** **(A)**, equipped with ST Nucleo G431RB **(B)** (that uses STM32G431RB **(C)**)
- ST Motor Control Workbench®
- EPC9147C (Provided with motor drive KIT’s)
  - Pre-programmed with a **sensor-less motor control algorithm** for a **specific motor** (Teknic_M-3411P-LN-08D) Note: When unboxing, ***it works only with the Teknic motor*** **(D)**



MOTOR CONTROL BACKGROUND

- For sensor-less motor control algorithms:
  - Only the three motor terminals connect to the inverter board
  - Depends on specific motor parameters (a model of the motor is used for control)
- New motor parameters **must be programmed before** operating a different motor



EQUIPMENT NEEDS, MOTOR ACCESS

Motor Access

- Direct access to the motor terminals
  – Motor terminal must be disconnected from inverter board
- Direct access to the motor shaft
  – Need to turn it by hand

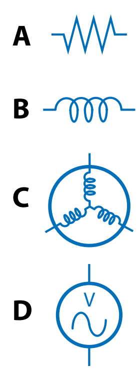
Equipment

- LCR meter
  – To measure line-to-line resistance and inductance
- Oscilloscope
  – To measure line-to-line Back EMF (BEMF)

MEASURING THE MOTOR PARAMETERS

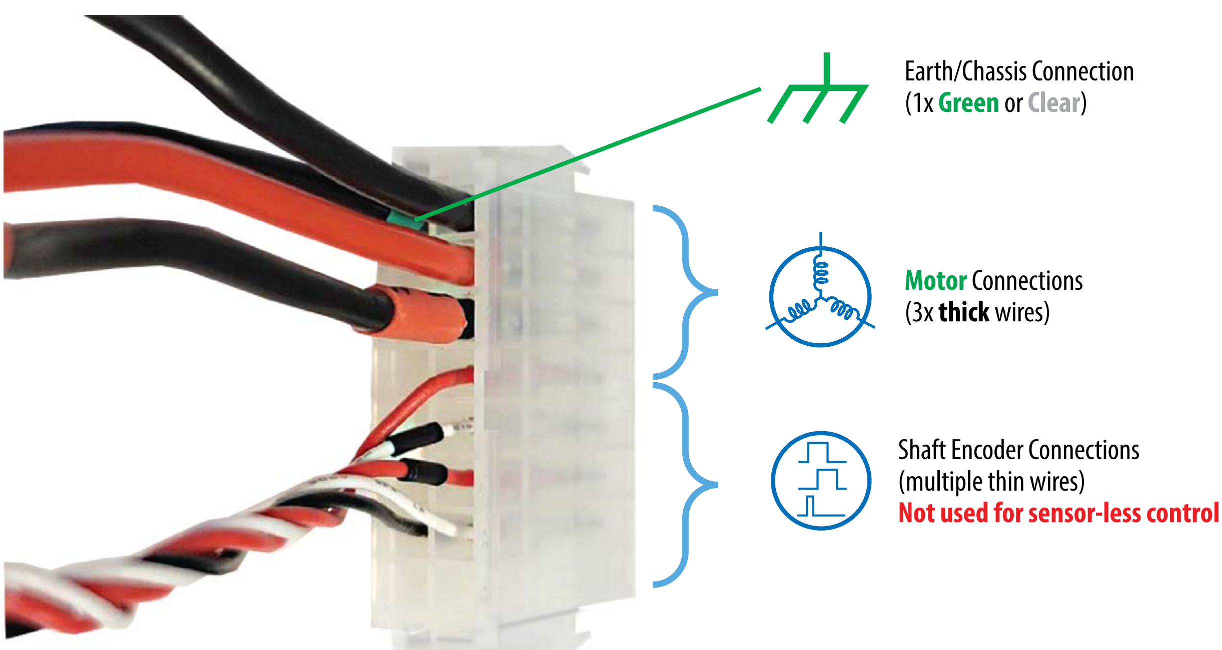
Motor Parameters Needed

- Terminal resistance **(A)**
  – Line-to-line
- Terminal inductance **(B)**
- – Line-to-line
- Pole pairs **(C)**
- Back EMF constant **(D)**



Identification of Motor Terminals

Example for Teknic Model M-3411P-LN-08D



Line-to-Line Resistance Measurement

1. Disconnect all three motor terminals from inverter
2. Connect **only two motor** terminals to an ohm-meter, third terminal is left floating
3. Measure the **line-to-line resistance**
4. **4-wire** resistance measurement is more accurate (if available
5. Use line resistance by dividing by two (**R = 400 mΩ**) in ST software

This motor has **$R_{L\text{-}L}$ = 800 mΩ** line to line resistance
(100 mΩ due to LCR meter leads)

Line-to-Line Inductance Measurement

1. Disconnect all three motor terminals from inverter
2. Connect **only two motor** terminals to the LCR-meter, third terminal is left floating
3. Measure the **line-to-line inductance**
4. **Note –** long leads will add inductance. Twisting the leads will help reduce inductance. More important for low inductance motors.
5. For motors with **varying inductance with shaft angle**, find the minimum and the maximum inductance values, by measuring at different angles.
6. Determine the average inductance:

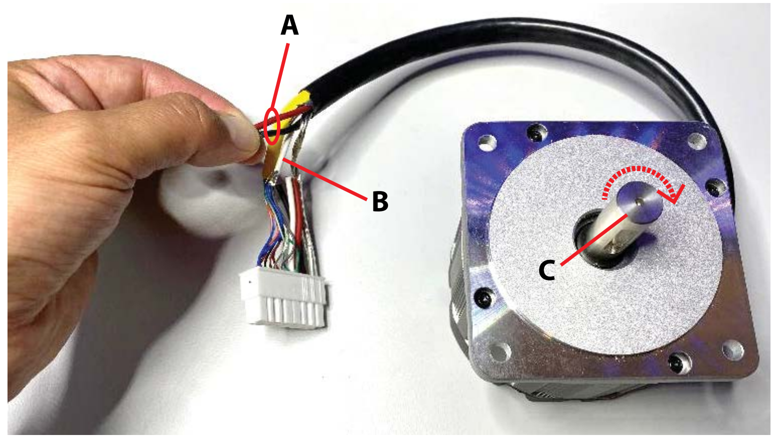
$$L_{avg} = \frac{L_{min} + L_{max}}{2}$$

7. For the example: Rounded 932/2 µH = **466 µH**.
8. Use the same value for L_d and L_q

This motor has **$L_{L\text{-}L}$ = 932 µH** line to line inductance
(LCR meter leads may also have inductance, **use autozero function if available**)

Determination of the Pole Pairs Number

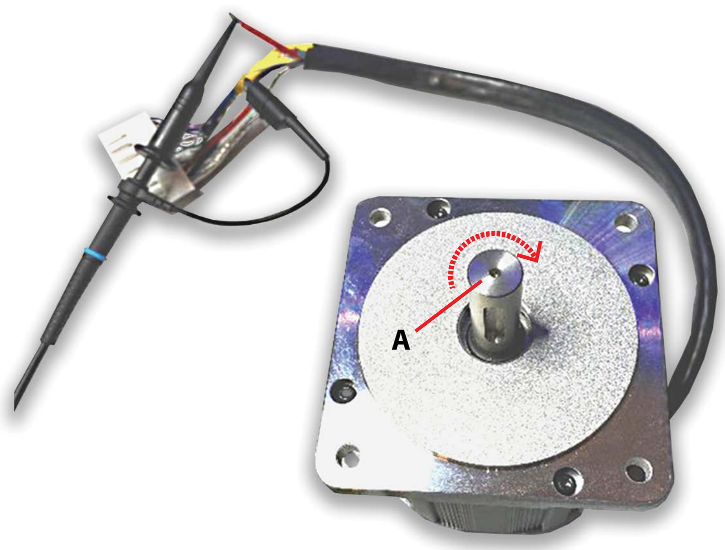
1. Disconnect all three motor terminals from inverter
2. Short **any two (A)** **motor** terminals, third terminal is left floating **(B)**
3. ***Gently*** and ***slowly*** hand spin the motor shaft **(C)** and make **one mechanical turn only**
   - Count the notches/steps/jumps that you feel with as the motor axle is rotated = **motor poles number**
4. Divide the **motor poles number** by 2 = **Pole Pairs number (pp)**

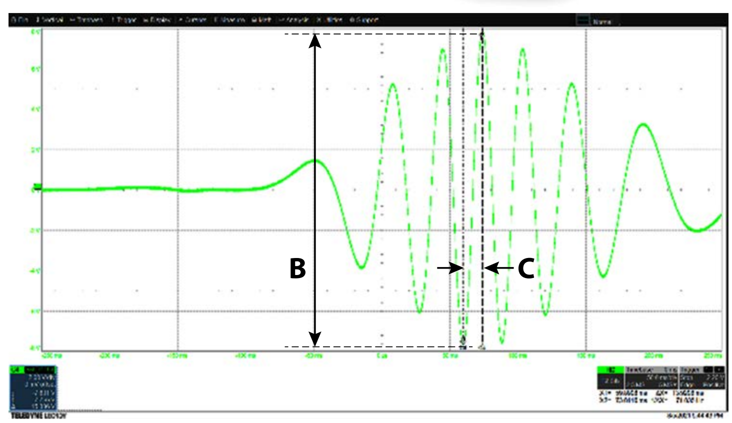


This motor has **pp= 4 pole pairs**

Line-to-line BEMF constant Measurement

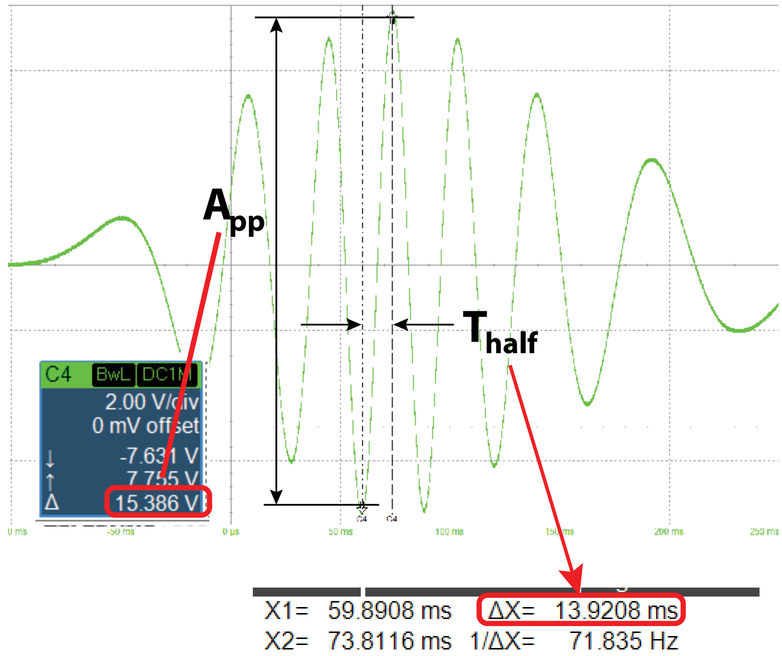
1. Disconnect all three motor terminals from inverter
2. Connect **one** of the **motor** terminals to an oscilloscope probe **ground** lead and the **other** **motor** terminal to the **tip**. The third motor terminal is left floating
3. Hand **spin** the motor shaft **(A)** and record the voltage signal on the oscilloscope.
4. **(B)** Measure the **peak-to-peak** voltage of **one-half sinusoid** (details on next slide)
5. **(C)** Measure the time period between the **same two peaks** (details next slide)





Line-to-line BEMF Constant Calculation

- **A_{pp}** = Half-sinusoid peak-to-peak voltage amplitude ($A_{pp} = 15.836\ V_{pp}$)
- **T_{half}** = Half sinusoid peak-to-peak period ($T_{half} = 13.92$ ms)
- **pp** = Pole Pairs (pp = 4)
- Calculate BEMF (for 1 krpm):
  - Units: A_{pp} [V], T_{half} [s]
  - $K_e = \frac{A_{pp}}{2 \cdot \sqrt{2}} \cdot \frac{1000 \cdot pp}{60} \cdot (2 \cdot T_{half})$
  - $K_e = 11.785\ pp \cdot A_{pp} \cdot T_{half}$ **Use this**
  - **K_e = 10.096** Vrms/krpm for example motor (will use 10.2 in motorBench)



INSTALLING ST MOTOR CONTROL DEVELOPMENT SUITE AND INPUTTING THE MOTOR DRIVE AND MOTOR PARAMETERS INTO A PROJECT

Install STM32 Motor Control Software Development Kit (MCSDK)®

Refer to ST website to install following software, follow exactly the steps indicated in ST website

1. Register/Login at the st.com website **(A)**.
2. Download the X-Cube-MCSDK **(B)**

   Note: ST Motor Profiler does **NOT** work with EPC power boards. If you try, you may damage the EPC power board.

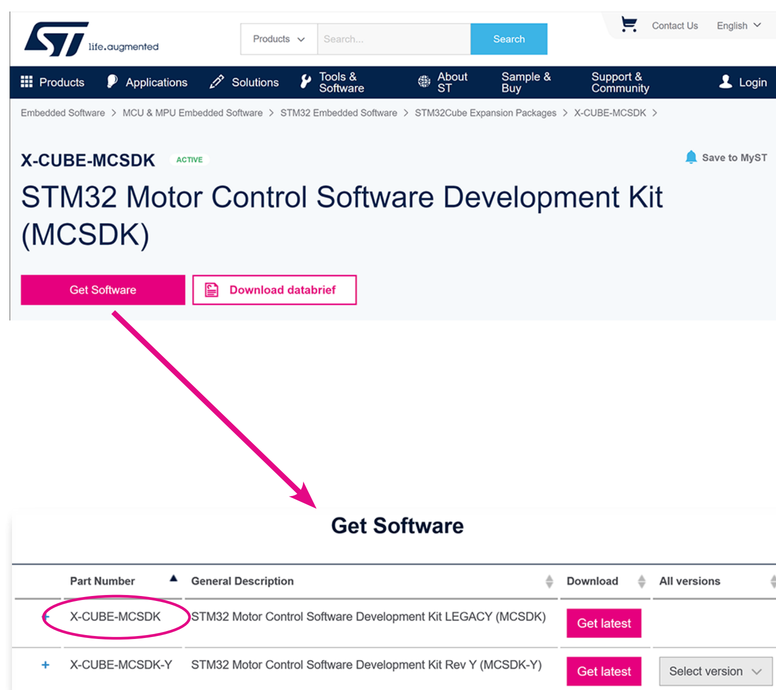
3. Install the **X-Cube-MCSDK** downloaded executable file, following the default installation settings.

Note: This guide is showing the process with the MCSDK kit. In principle same steps apply also for the MCSDK- Y kit. The only difference is that the .stmcx file needs to be created for MCSDK-Y kit. The .stmcx file for MCSDK-Y kit can be created using same parameters as they are shown in this guide.

It may be necessary to disable Antivirus and firewall programs during installation

A

B



Install STM32CubeMX

Refer to ST website to install following software, follow exactly the steps indicated in ST website

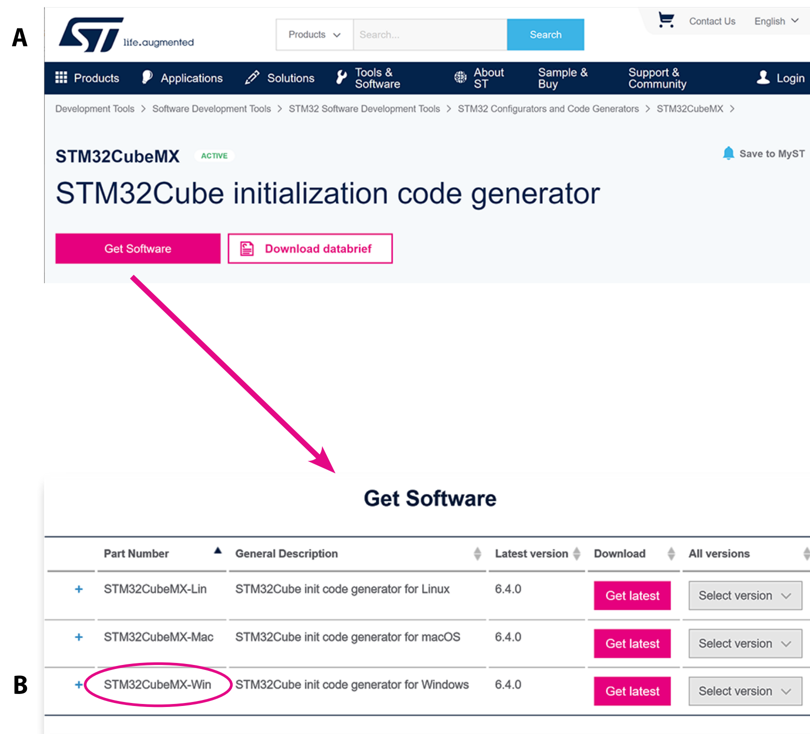
1. Register/Login at the st.com website **(A)**
2. Download the STM32CubeMX-Win **(B)**

   Note: ST Motor Profiler does **NOT** work with EPC power boards. If you try, you may damage the EPC power board.

3. Install the **STM32CubeMX-Win** downloaded executable file, following the default installation settings.

Note: This procedure has been tested with version 6.2.0 and version 6.3.0

It may be better to disable Antivirus and firewall programs during installation



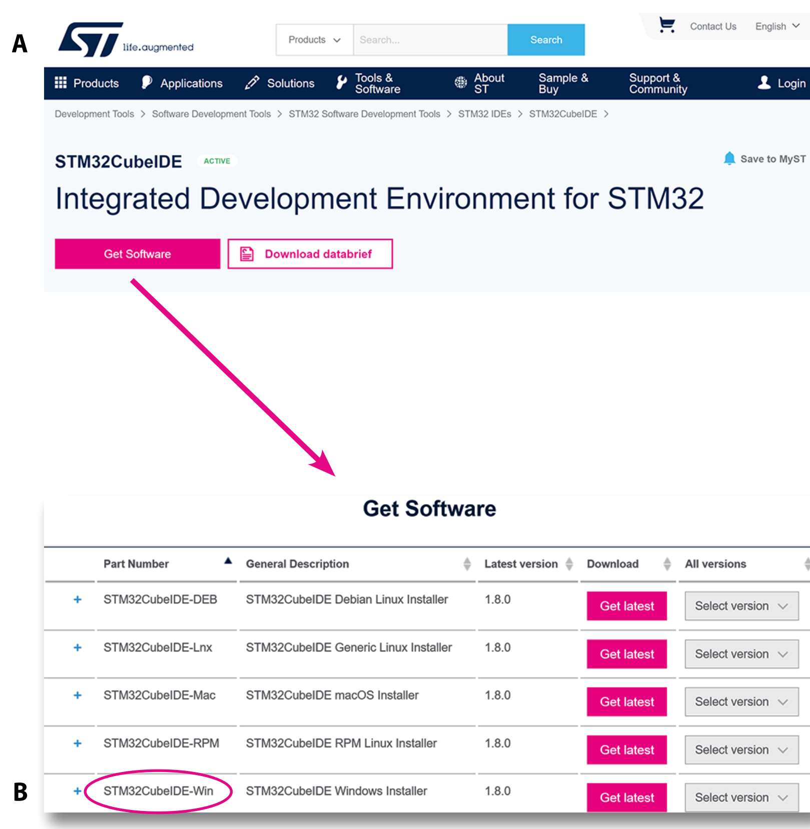
Install STM32Cube-IDE

Refer to ST website to install following software, follow exactly the steps indicated in ST website

1. Register/Login at the st.com website **(A)**
2. Download the STM32CubeIDE-Win **(B)**
3. Install the **STM32CubeMX-Win** downloaded executable file, following the default installation settings.

Note: This procedure has been tested with version 1.6.0 and version 1.7.0

It may be better to disable Antivirus and firewall programs during installation



Install STM32CubeProg

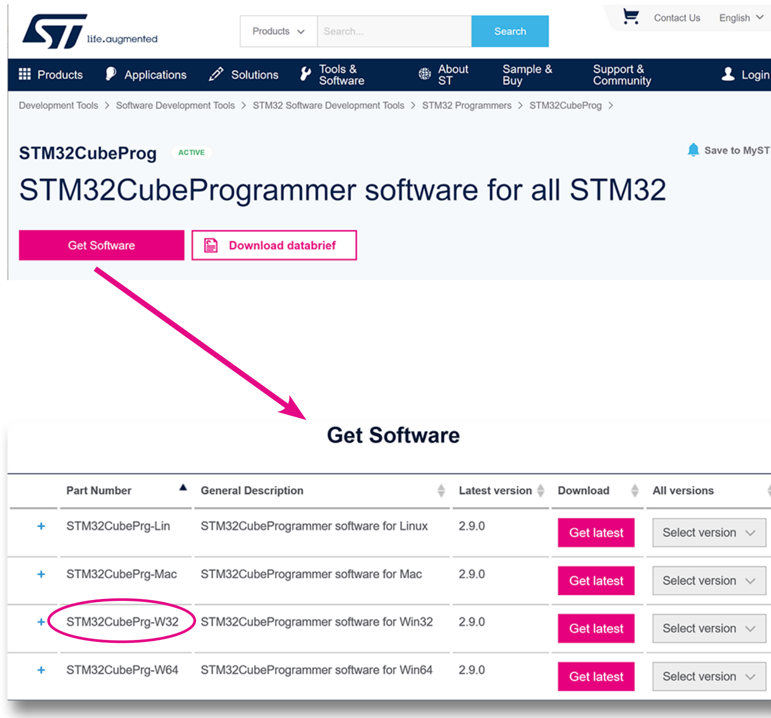
Refer to ST website to install following software, follow exactly the steps indicated in ST website

1. Register/Login at the st.com website **(A)**
2. Download the STM32CubeProg **(B)**
3. Install the **STM32CubeProg** downloaded executable file, following the default installation settings.

Note: The STM32CubeProg can be used to flash the device with a .elf file, if the .elf file is already available. The steps are explained in ECP9147C quick start guide. The aim of this guide is to show the process to generate the .elf file

It may be better to disable Antivirus and firewall programs during installation

A



B

Download the specific .stmcx file for the EPC power board

Refer to EPC website to download the proper ST Motor Control Workbench **.stmcx** definition file.

As an example, EPC9145's product page, the file link can be found at :

https://epc-co.com/epc/Products/DemoBoards/EPC9145.aspx

The filename is:

G431-EPC9145-DummyNema34_50k_100n.stmcx

and needs to be loaded in **ST Motor Control Workbench** program.

Note: This file was made for MCSDK Kit 5.4.x and it will not be loaded by the MCSDK-Y kit software.

Folder naming directions: The working directory should have short path and should not contain spaces or symbols.

Example:

C:\motor_software (acceptable: no spaces and no special characters - Note: an underscore symbol ("_") is not considered a special character)

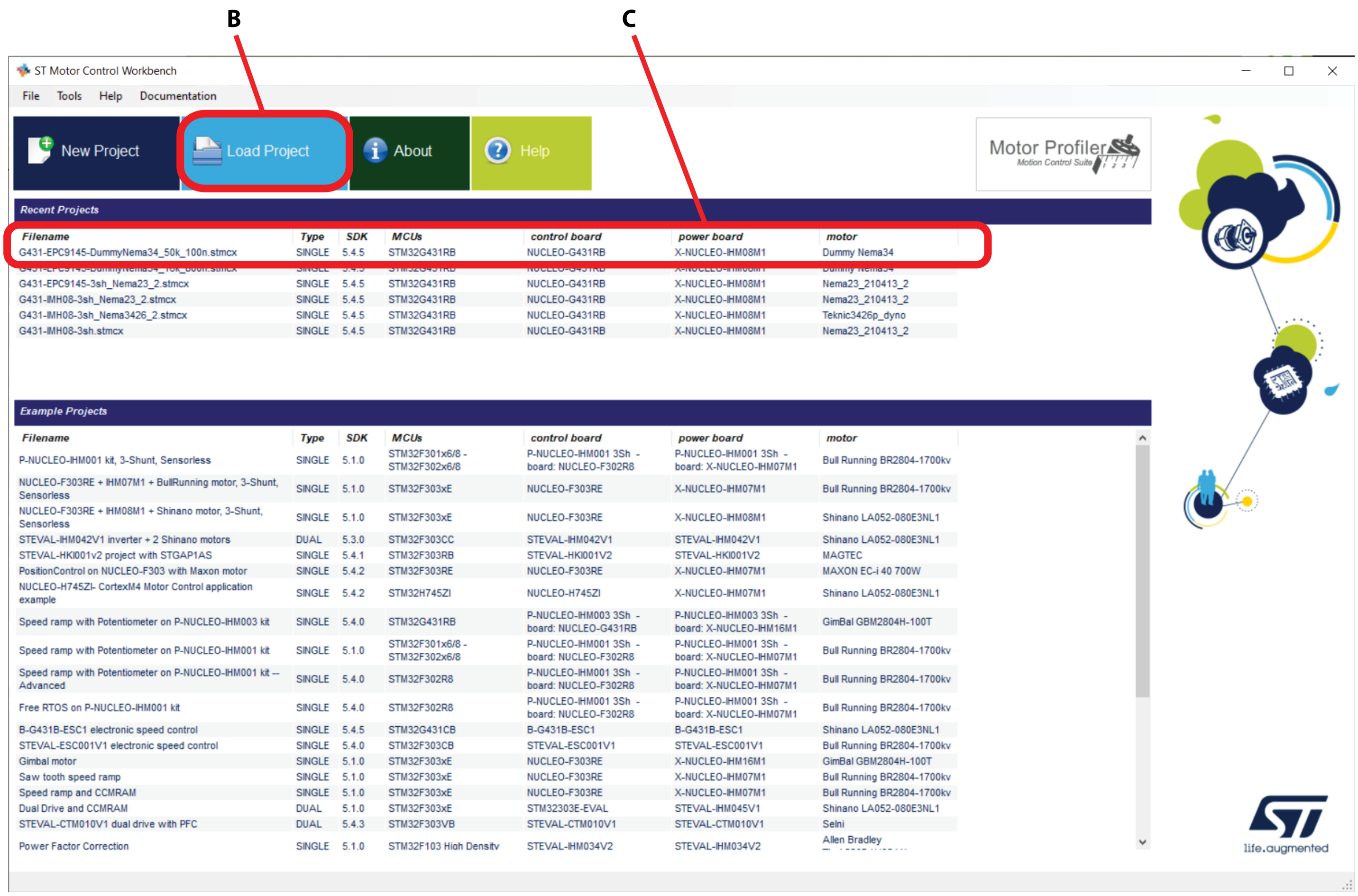
C:\motor & software (unacceptable: uses spaces and a special character)

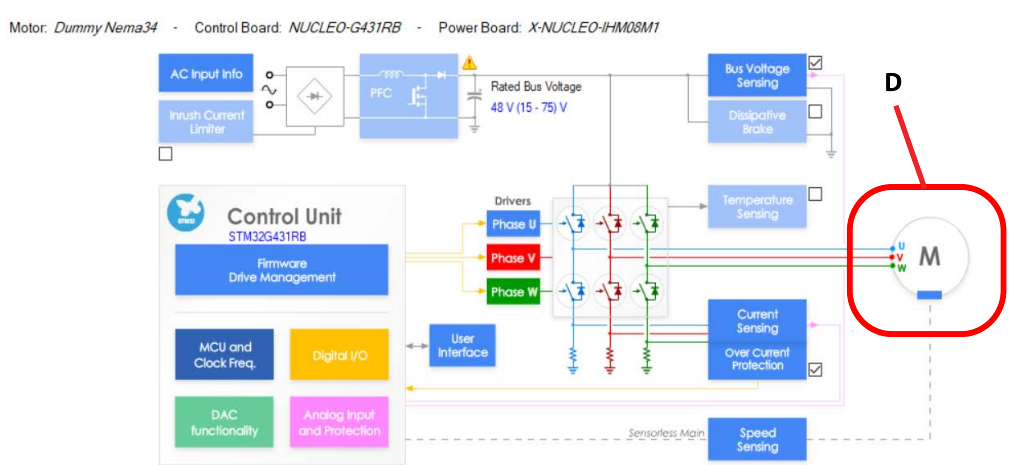
Launch Motor Control Workbench®

1. Start Motor Control Workbench 5.4.x **(A)**
2. Load specific sample project **(B)**

   or click on Recent Project's filename to open it **(C)**
3. Once the project is loaded, it is already set with proper parameters for **specific** power board and for **specific** motor (**Teknic_M-3411P-LN-08D**)

(A)



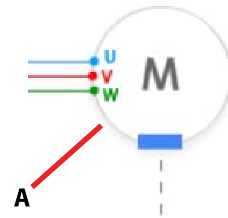


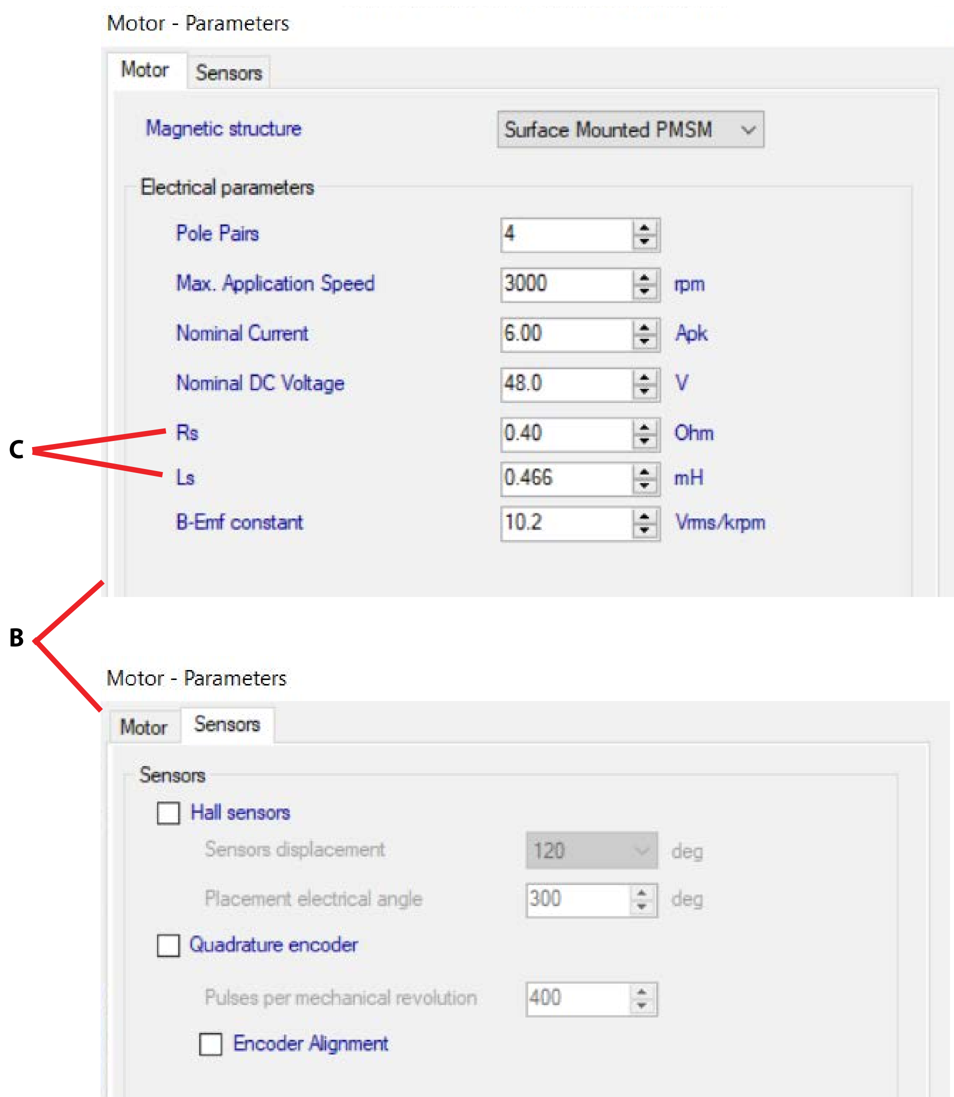
Note:

1. If you downloaded the project for the specific power board, it is already set properly to work for the **Teknic_M-3411P-LN-08D** motor.
2. To adapt to a new motor you have to change only the motor parameters by clicking on the motor icon circled in red **(D)**

Motor parameters dialog box

- Click on motor icon **(A)** to visualize the motor parameters
- Parameters shown here **(B)** are for the **Teknic_M-3411P-LN-08D** motor.
- To adapt to a new motor, change only the motor parameters.
- **Note**: the Rs and Ls parameters **(C)** are referred to phase to neutral point (i.e. the resistance and the inductance measured phase to phase must be divided by 2).





Motor Control Workbench parameters

As an example, all parameters related to the EPC9145 are shown in the next figures.

For other EPC power boards, download the corresponding **.stmcx** file from EPC website.

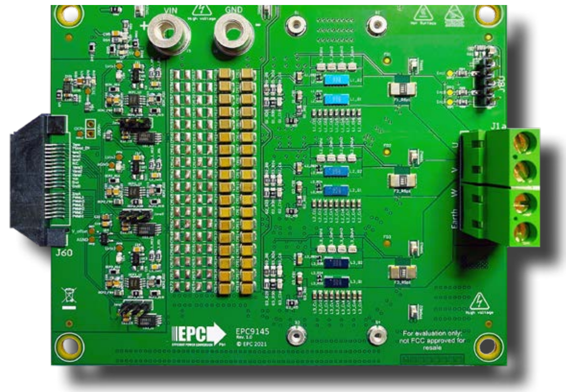
Warning: ST Motor Profiler does NOT work with EPC power boards. If you try, you may damage the EPC power board.

The next figures can be also used to modify an existing project originally made for the **ST MCSDK-Y** kit.

Note: This guide is based on ST MCSDK kit only.

The following figures are optional and are not needed if only motor parameters need to be changed. In that case you can skip and proceed to the **Code Generation** steps.

Refer to **ST Motor Control Workbench Guide** for further information.



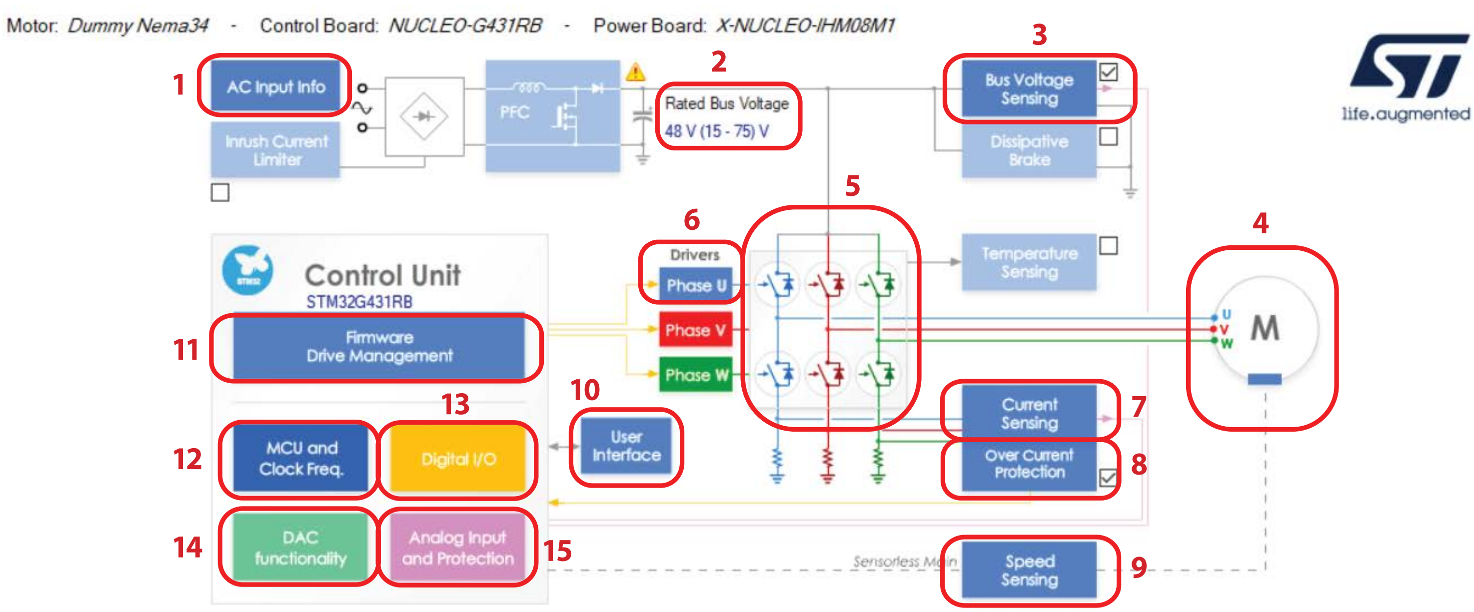
EPC9145 board

Example MCSDK project for EPC9145

Overview

Red circled boxes/areas below indicate which categories have their parameters set in the following example project.

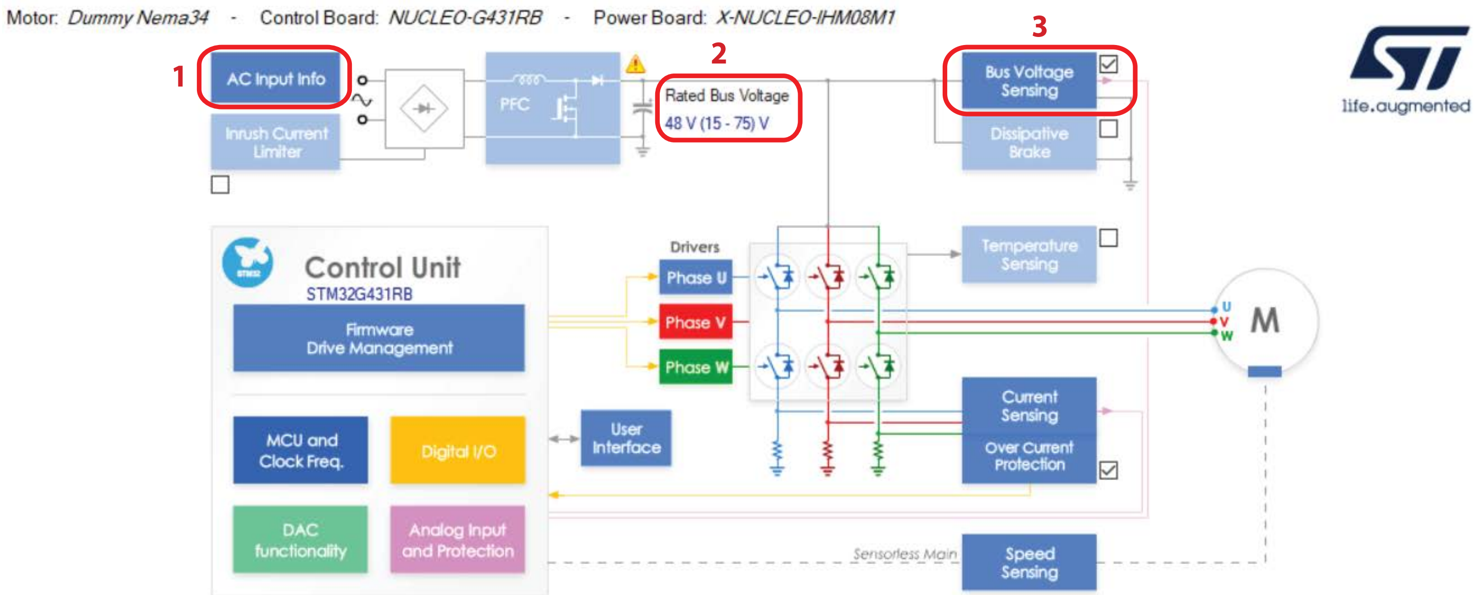
Note: Make sure that all parameters set match exactly as shown.



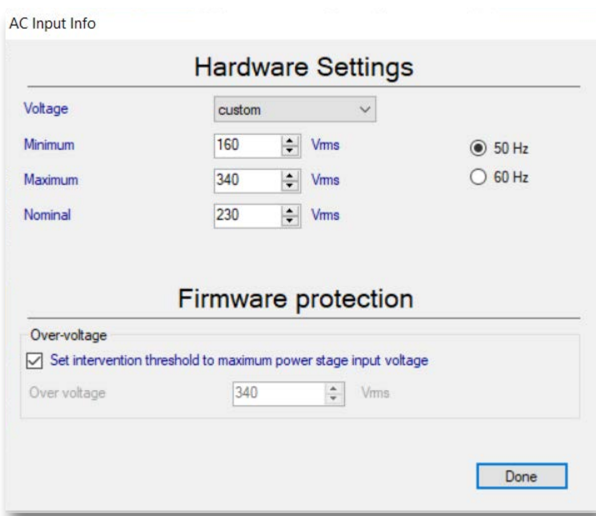
Legend:

1. AC Input info
2. Power Stage - Rated Bus Voltage Info
3. Power Stage - Bus Voltage Sensing
4. Motor Parameters
5. Power Stage - Power Switches
6. Power Stage - Driving Signals Polarity - U Driver
7. Power Stage - Current Sensing
8. Power Stage - Overcurrent Protection
9. Drive Management - Speed Position Feedback Management
10. User Interface
11. Drive Management
12. MCU and Clock Frequency Selection
13. Control Stage - Digital I/O
14. Control Stage - DAC Functionality
15. Control Stage - Analog Input and Protection

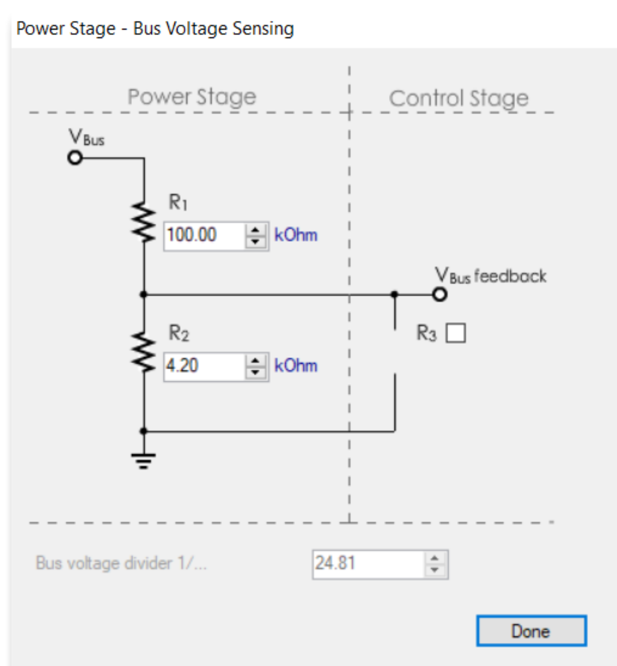
Specific settings for Example MCSDK project for EPC9145



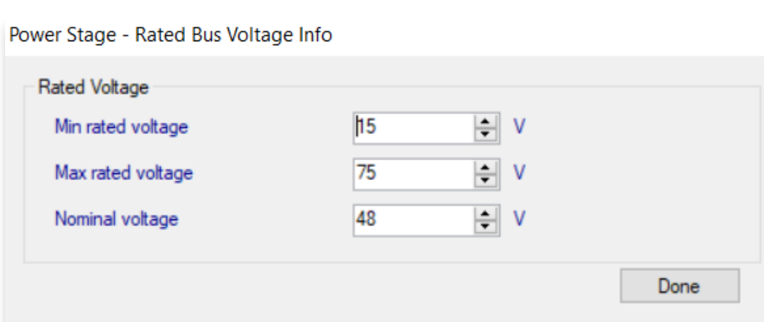
1



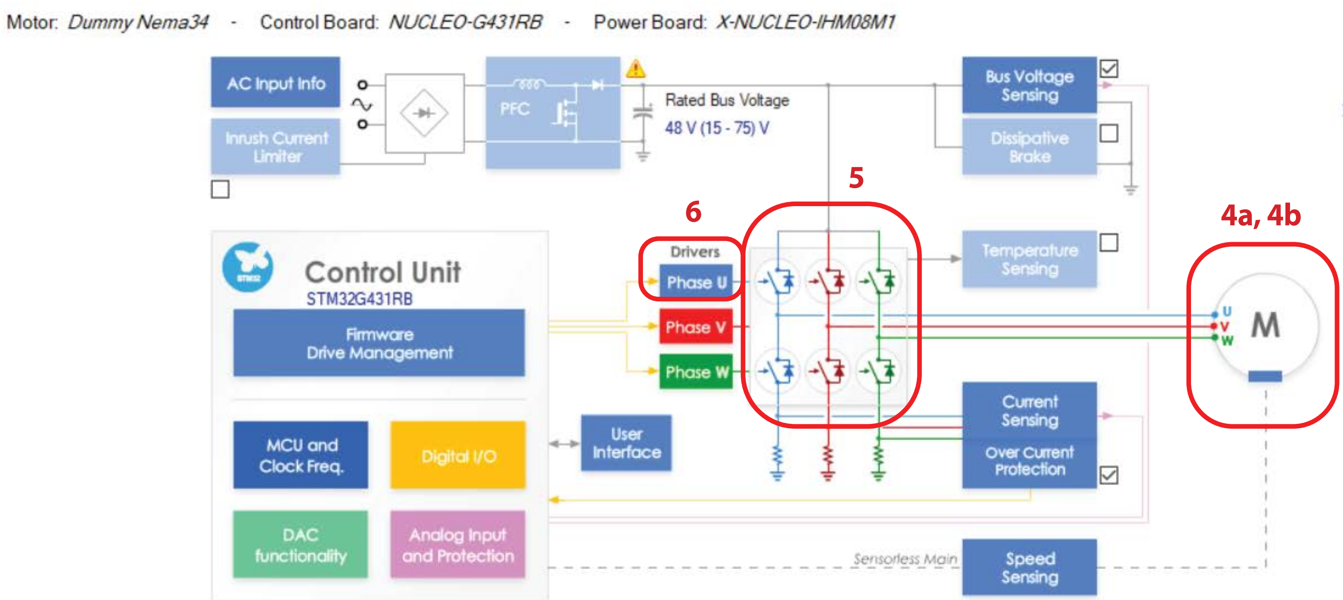
3



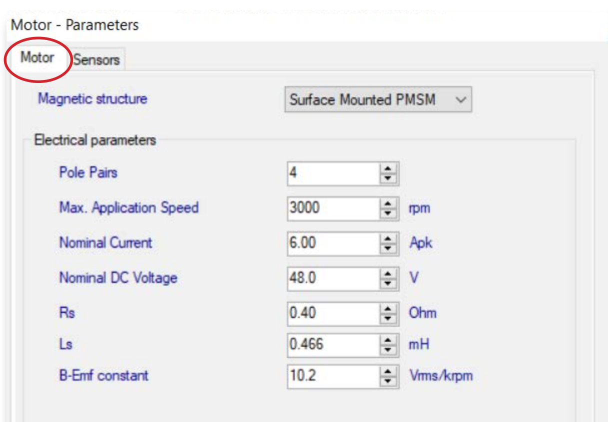
2



Specific settings for Example MCSDK project for EPC9145 *(continued)*

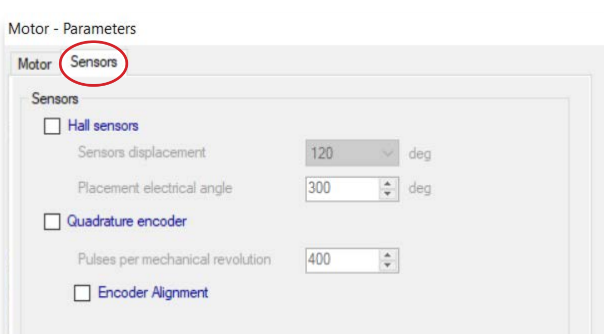


4a

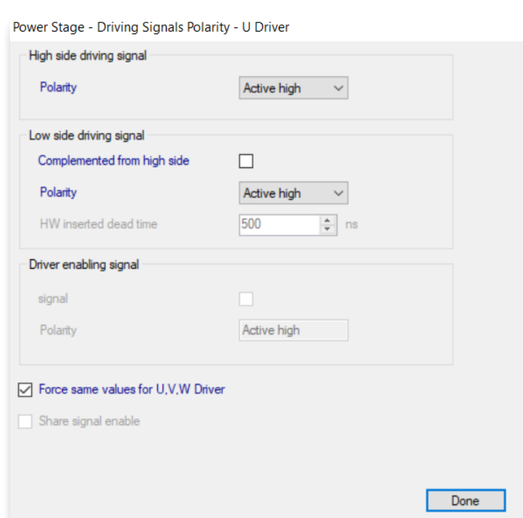


5

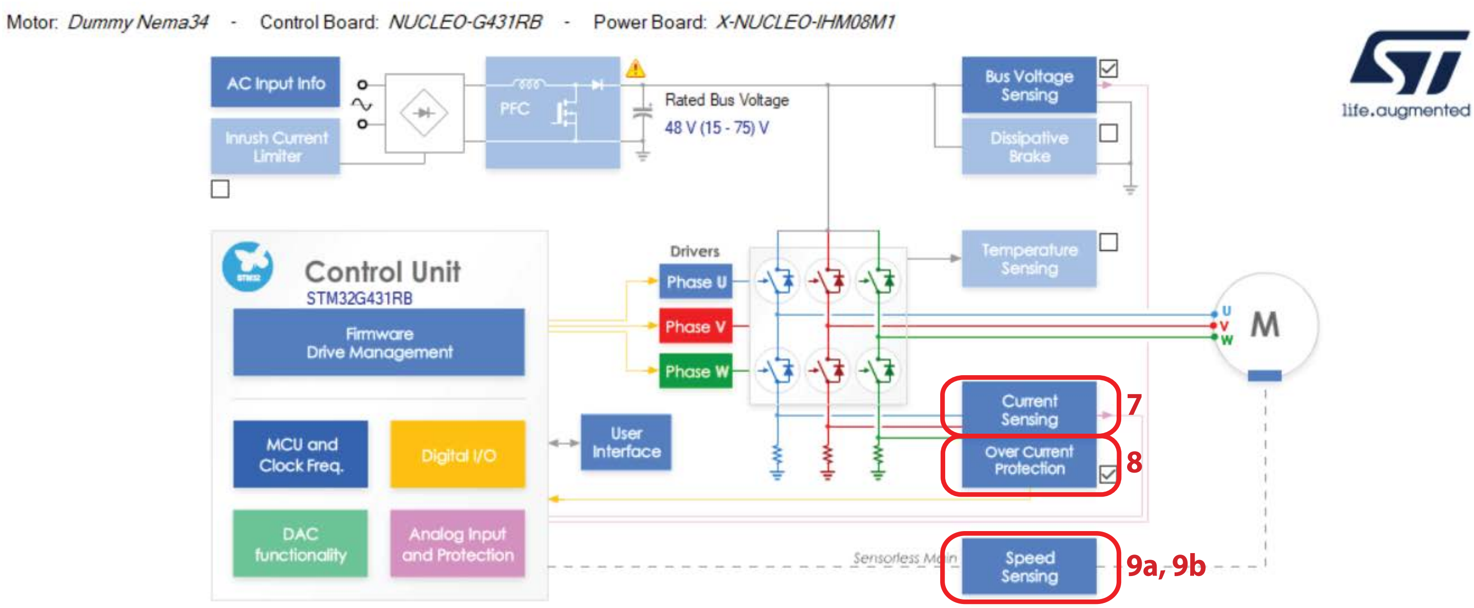
4b



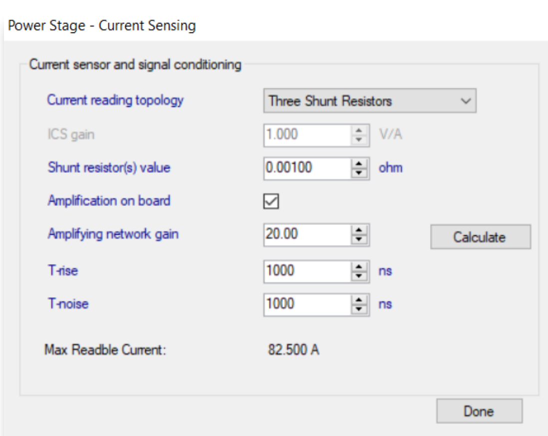
6



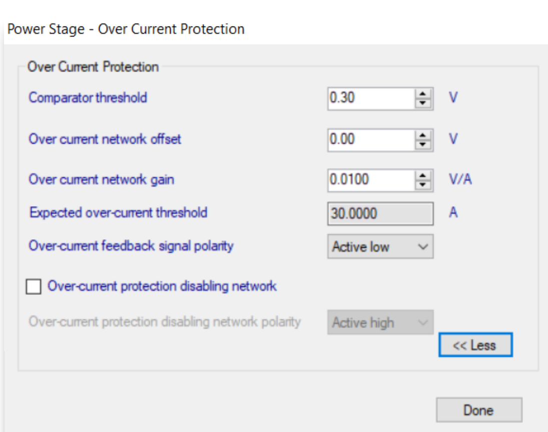
Specific settings for Example MCSDK project for EPC9145 *(continued)*



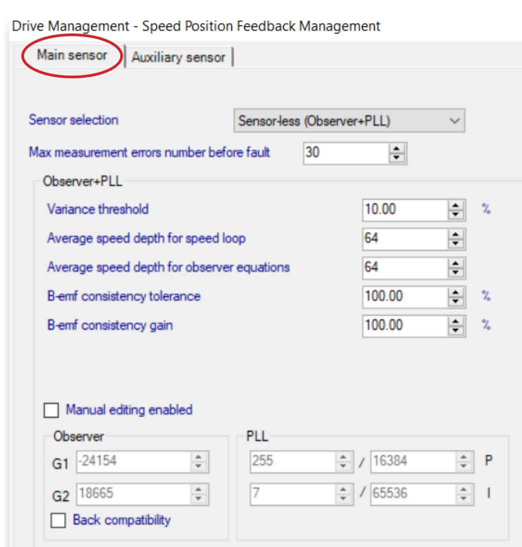
7



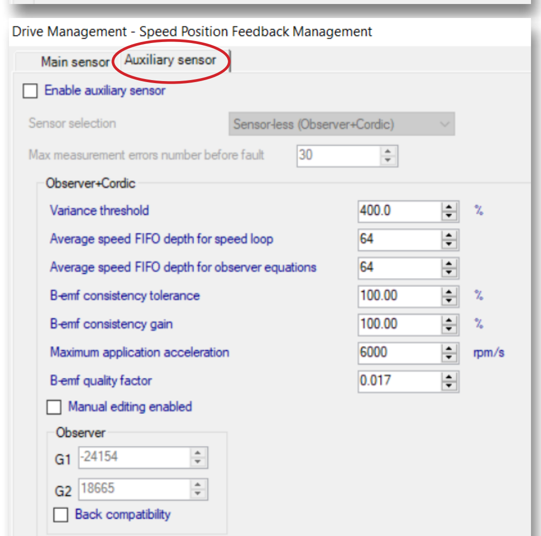
8



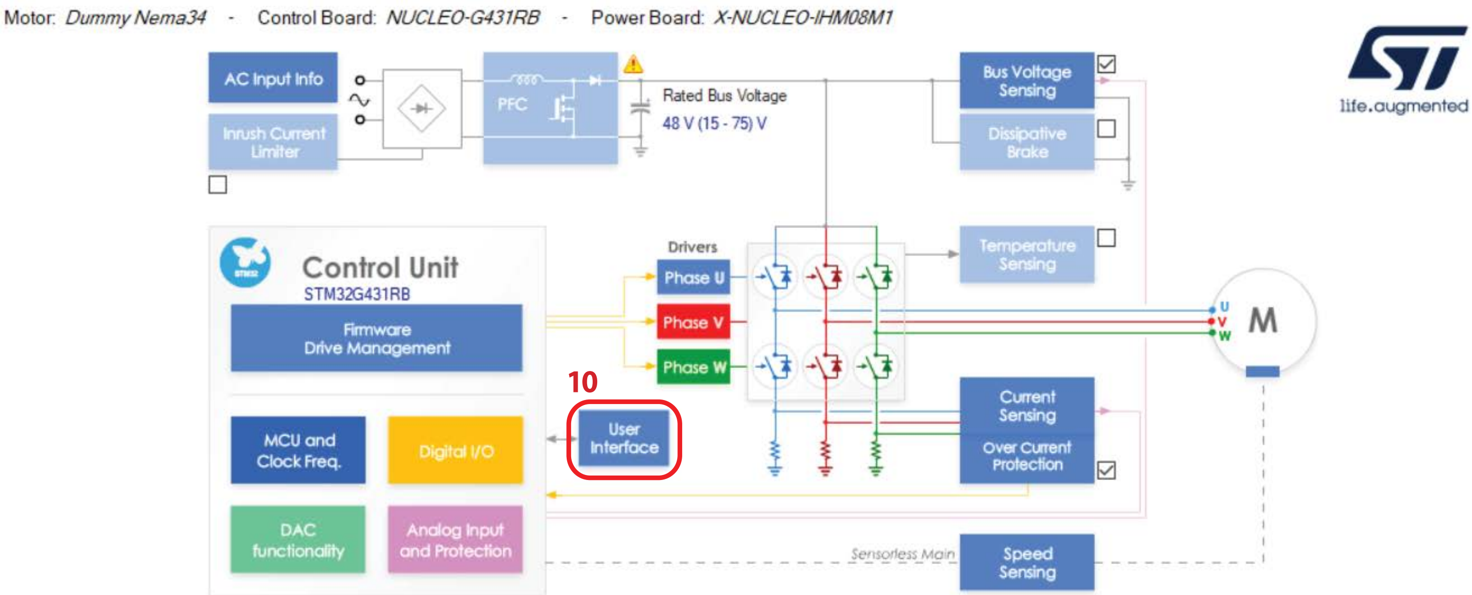
9a



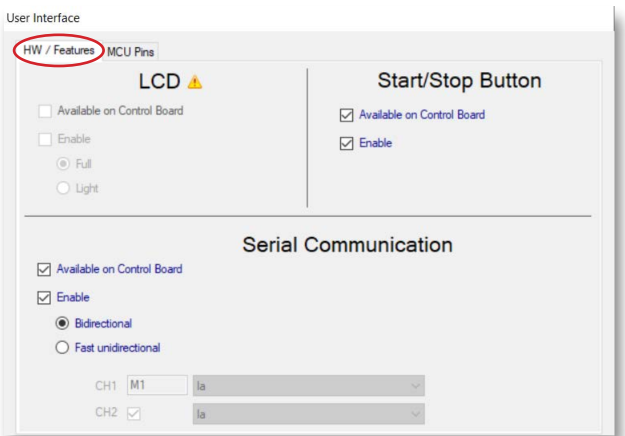
9b



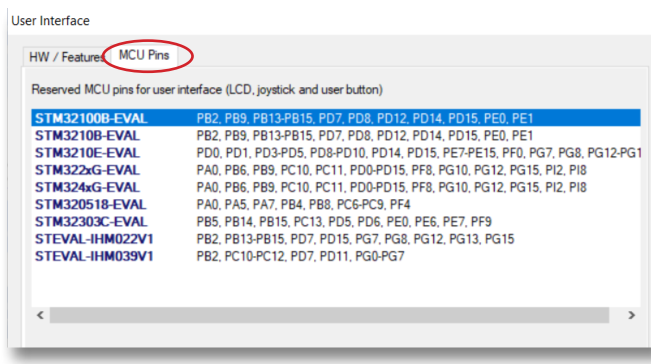
Specific settings for Example MCSDK project for EPC9145 *(continued)*



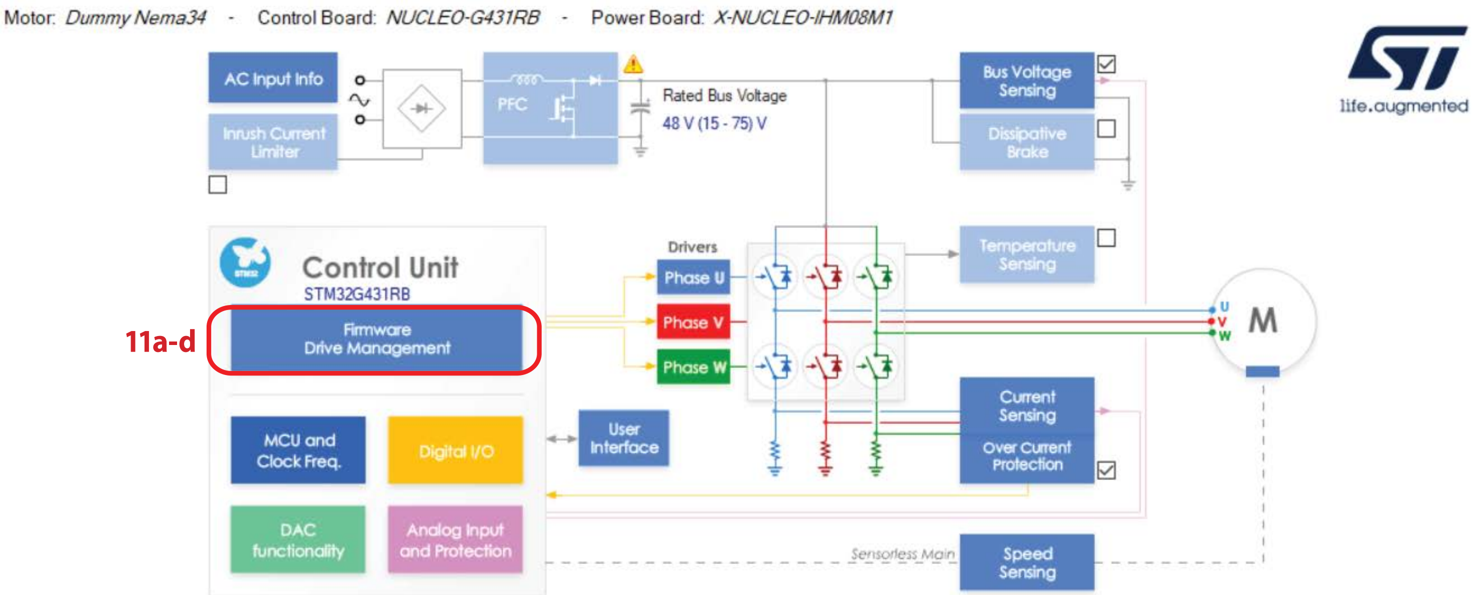
10a



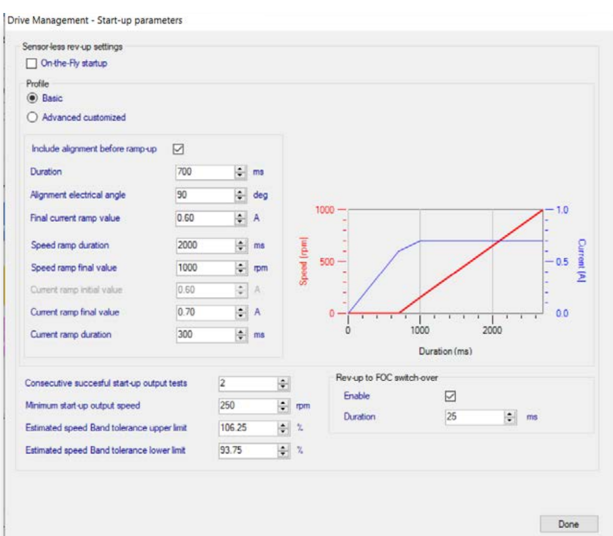
10b



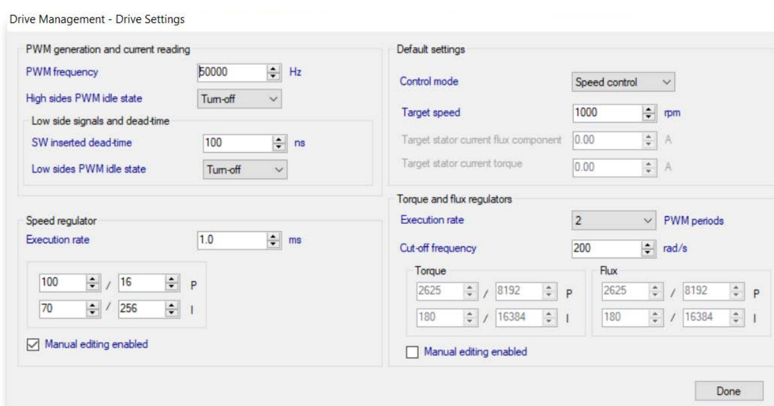
Specific settings for Example MCSDK project for EPC9145 *(continued)*



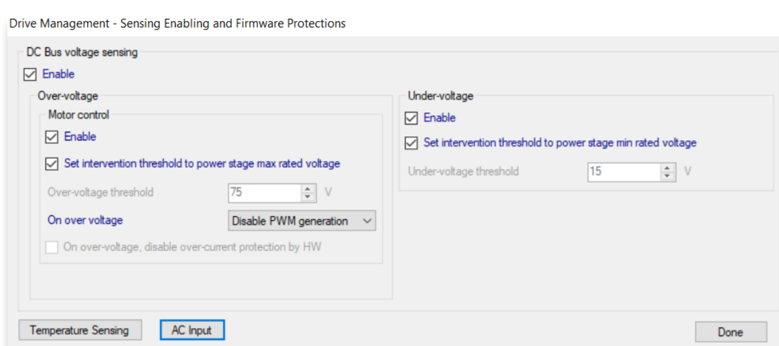
11a



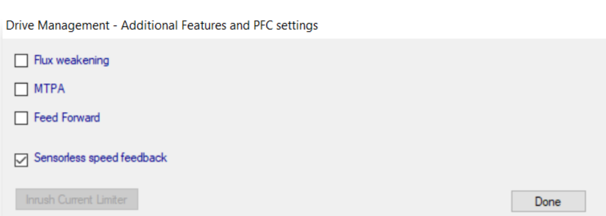
11b



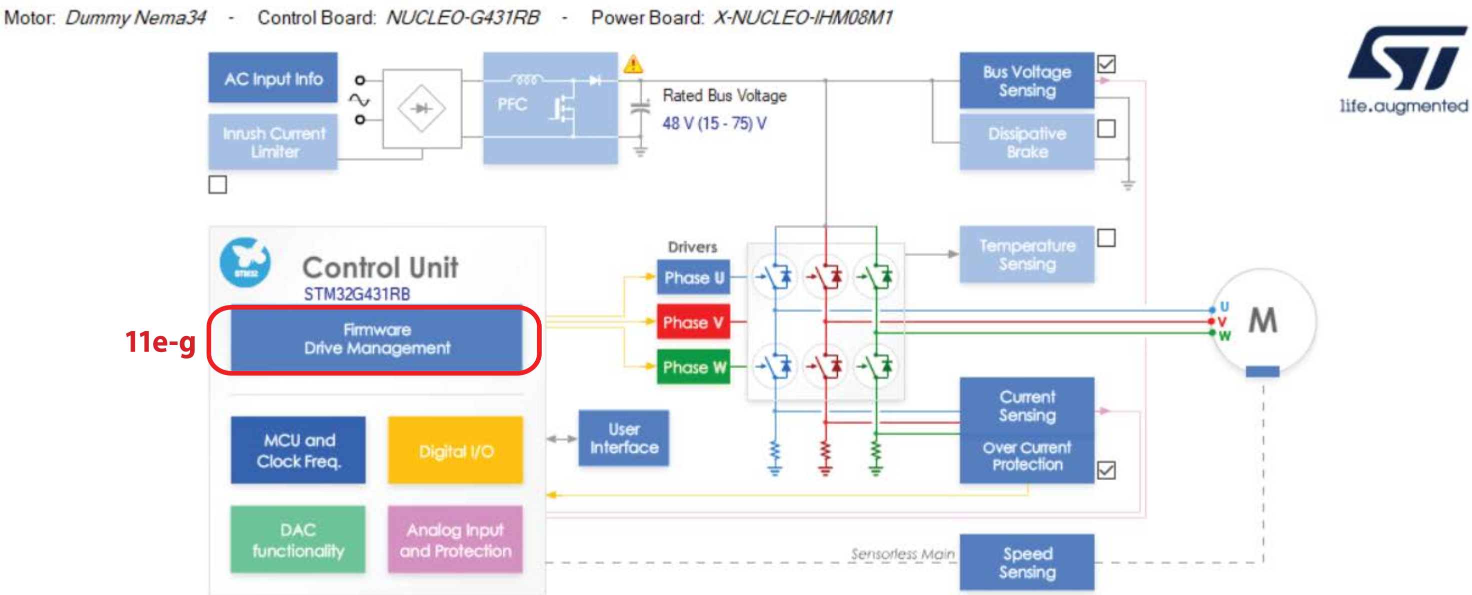
11d



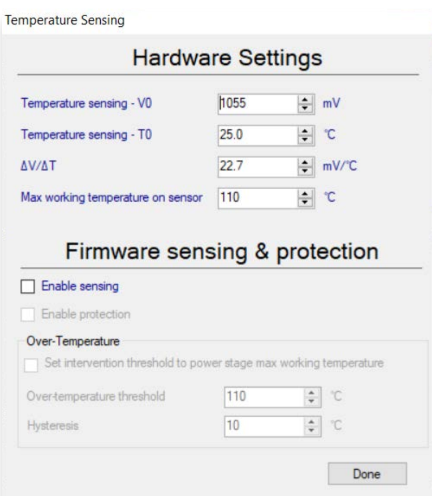
11c



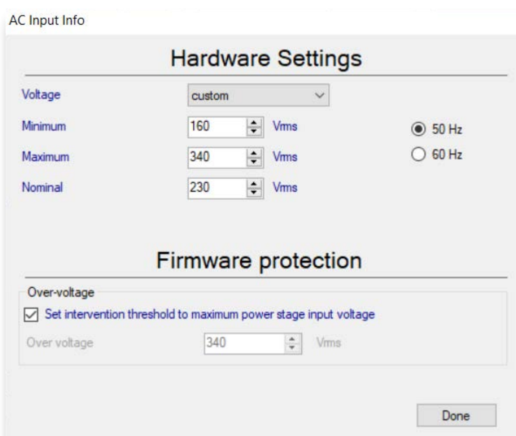
Specific settings for Example MCSDK project for EPC9145 *(continued)*



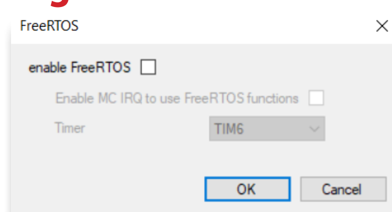
11e



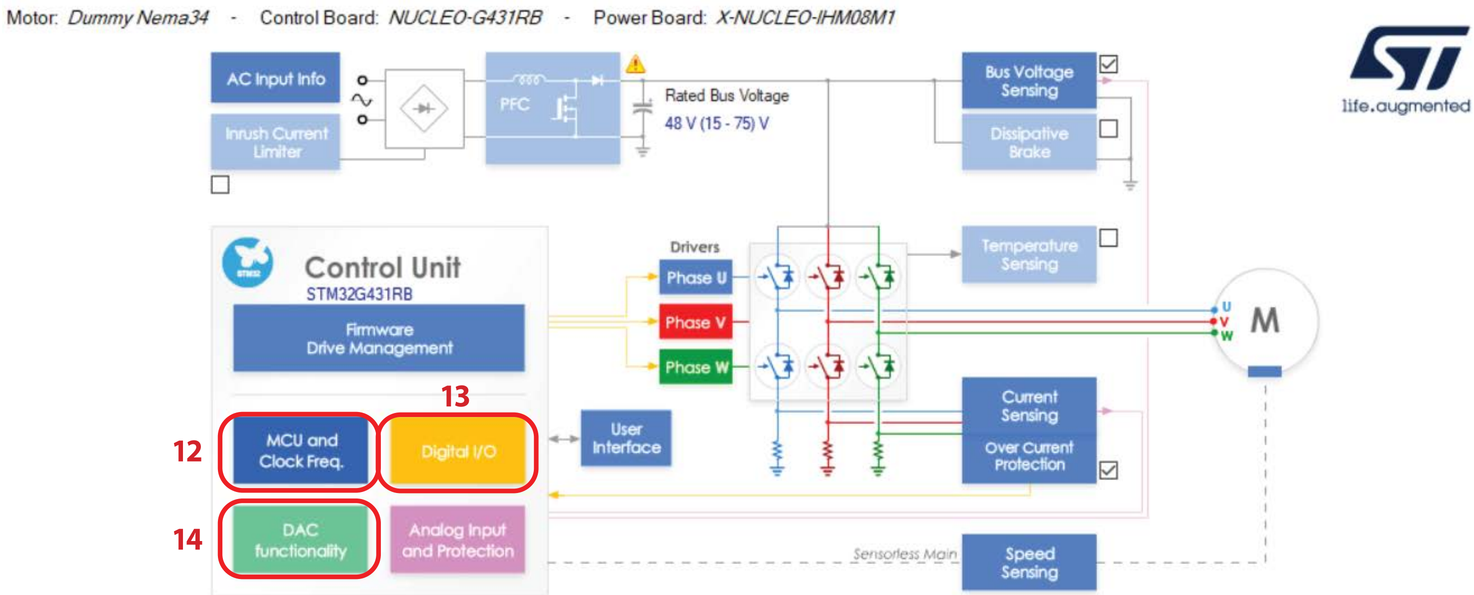
11f



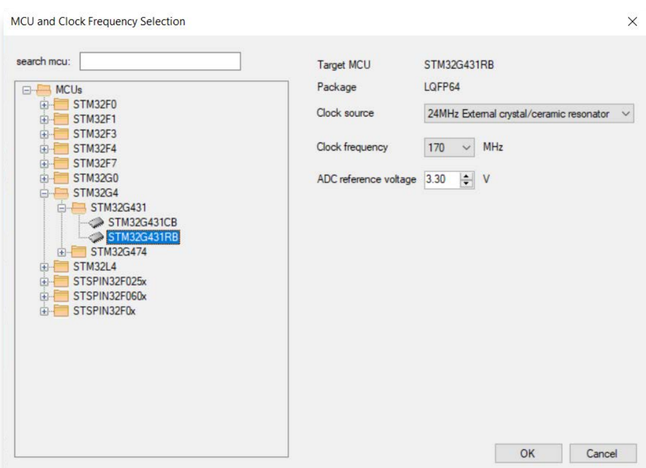
11g



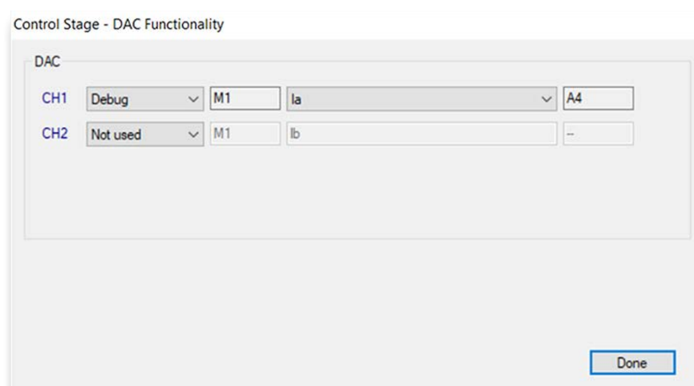
Specific settings for Example MCSDK project for EPC9145 *(continued)*



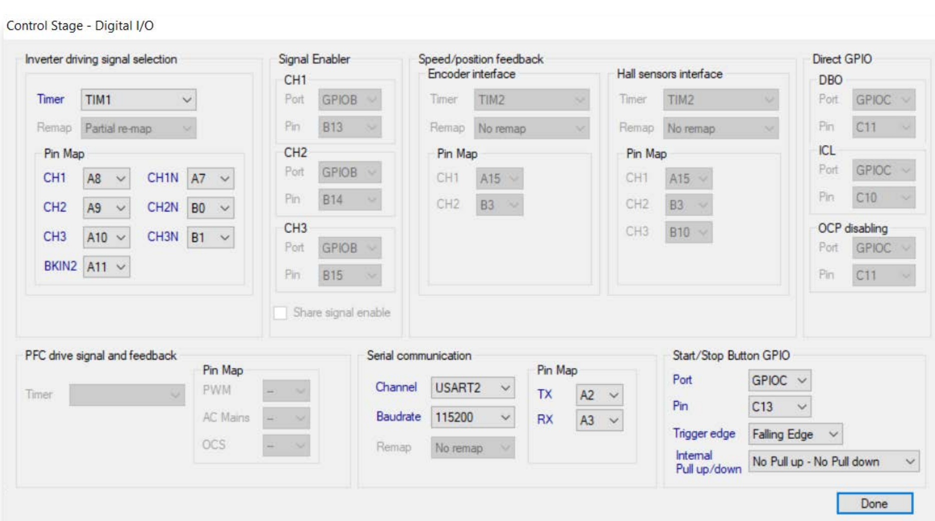
12



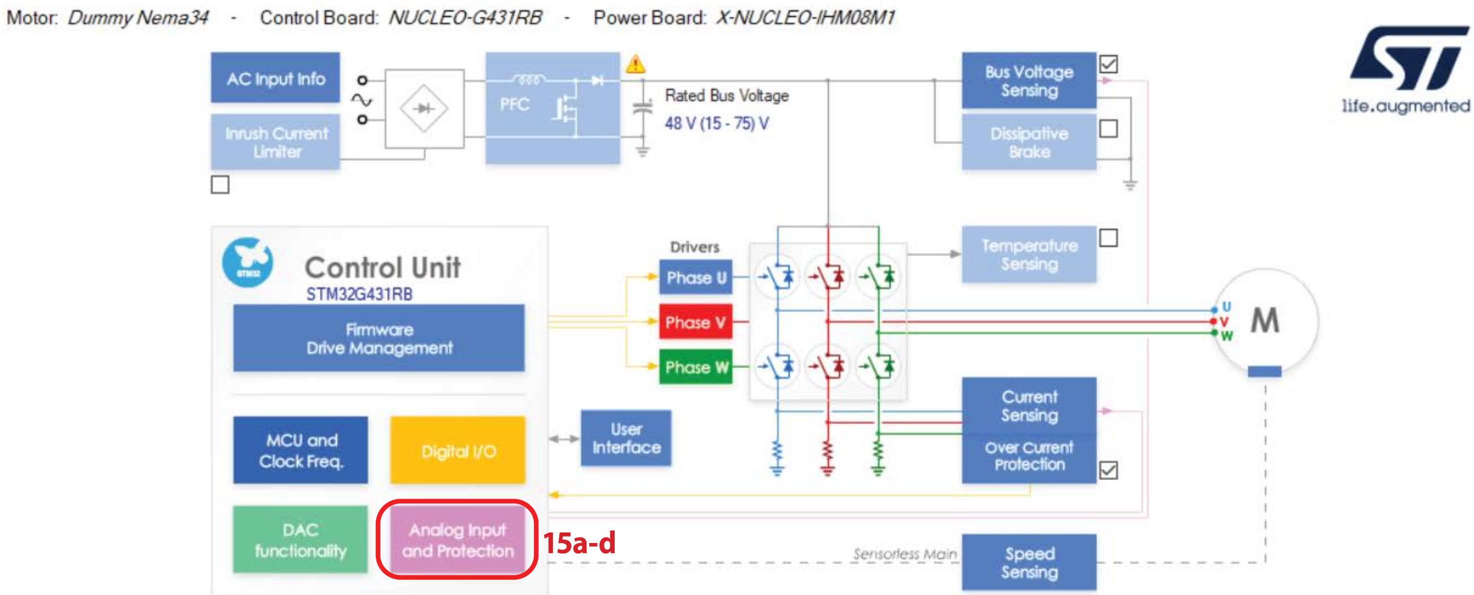
14



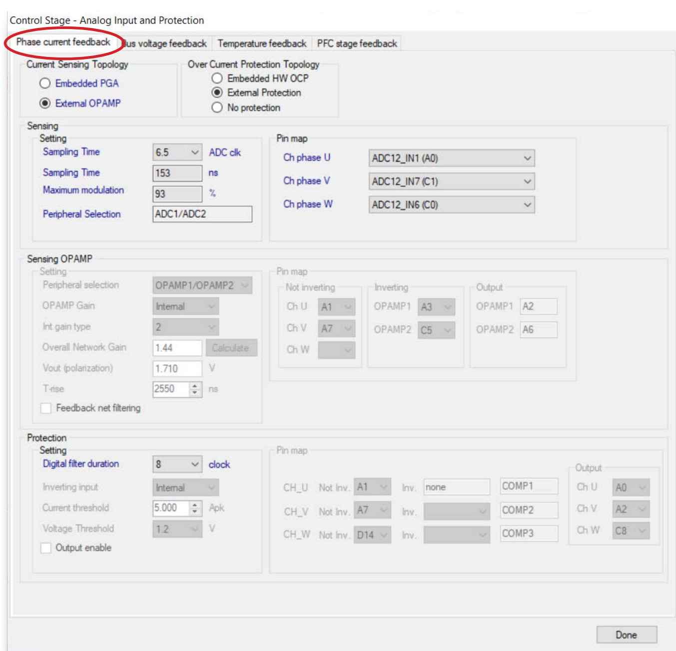
13



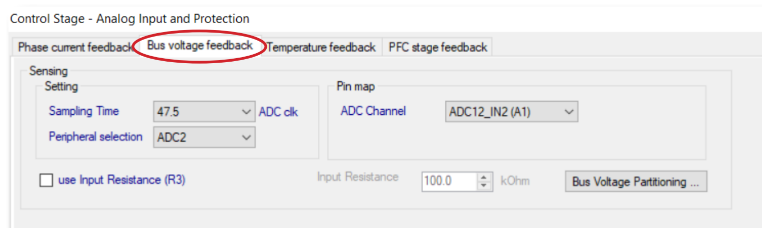
Specific settings for Example MCSDK project for EPC9145 *(continued)*



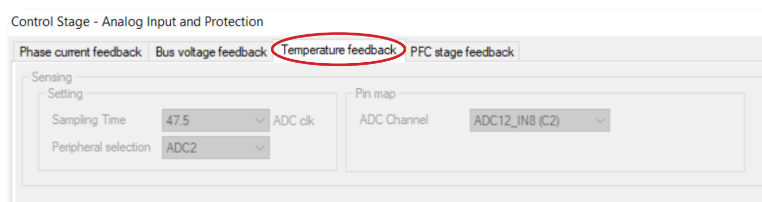
15a



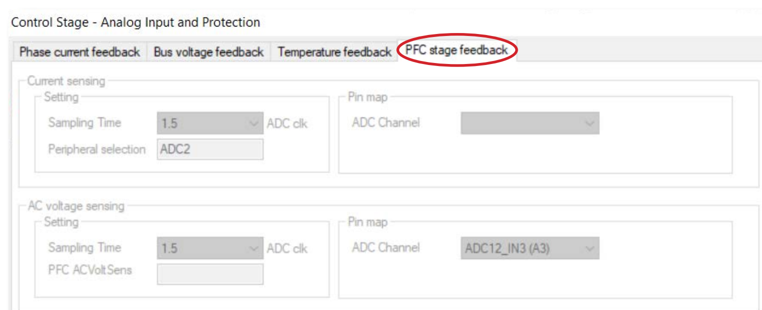
15b



15c

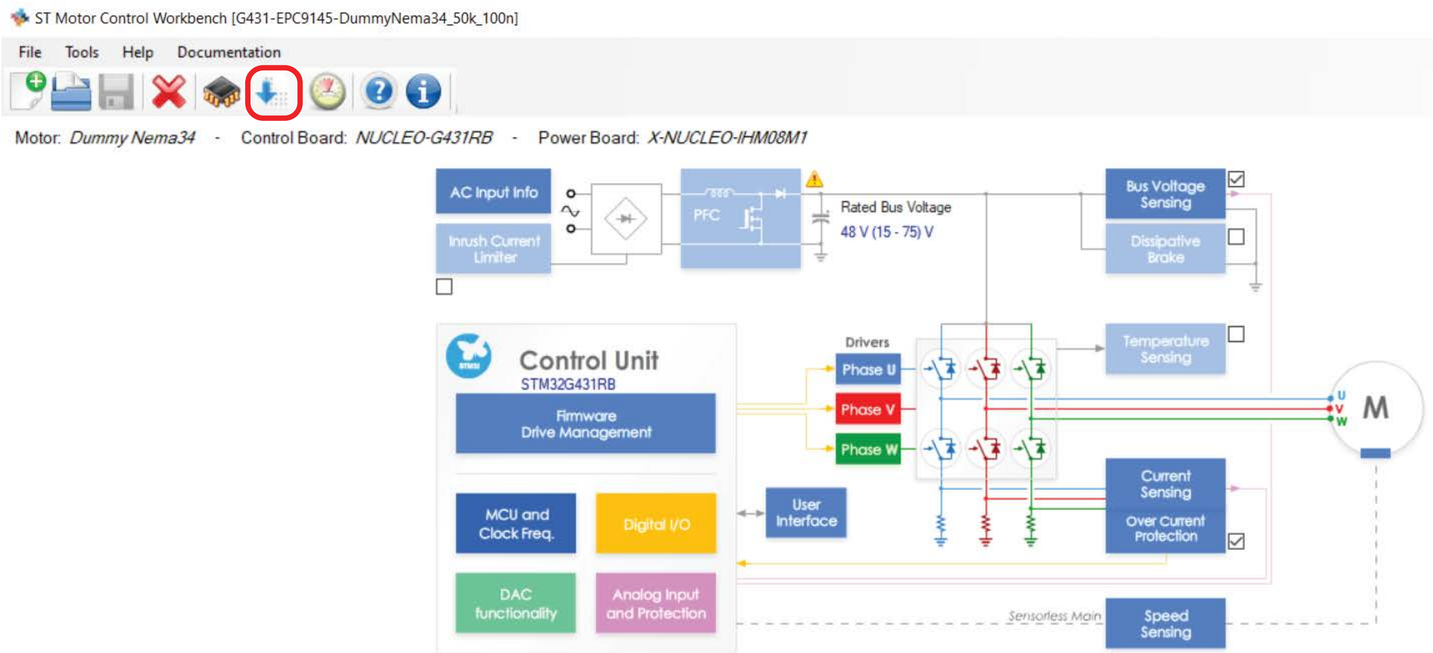


15d



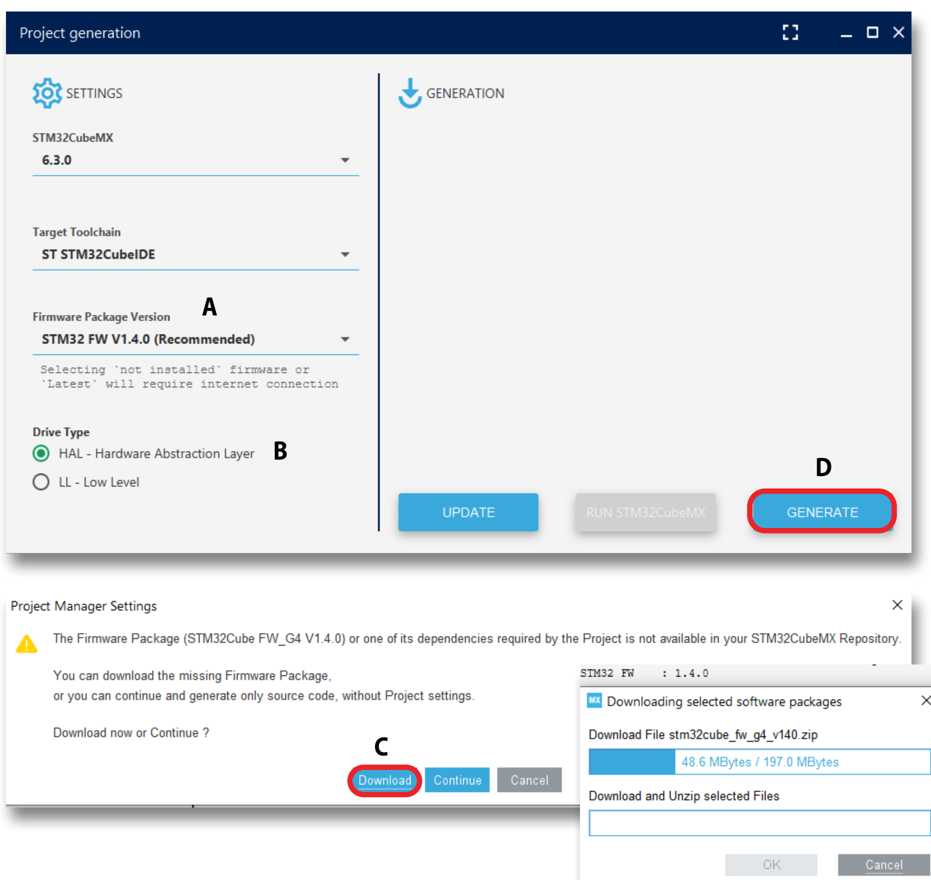
LAUNCH THE CODE GENERATOR

Click on the **Code Generator button.** This will invoke the **STM32CubeMX program.**
Note: depending on the installation, you may have different versions.



STM32CUBEMX – AUTOMATIC CODE GENERATION

1. Select the Recommended firmware version **(A)**
2. Select HAL drive type **(B)**
   Note: At the first-time use, the firmware will not be available, it will be downloaded automatically. Confirm that you want to continue the download **(C)**. This may generate Windows firewall messages. It may also be necessary to disable anti-virus software.
3. Press **Generate Button (D)**
   - The first time it may happen that an error message is generated because the firmware is not yet installed completely. In this case, close the window and restart the process.
   - At the end of code generation, you should not get any error message, if it happens, repeat the process. If it continues, contact a ST local FAE.
4. At the end of the code generation, open the folder containing the code and **close the STM32CubeMX window.**

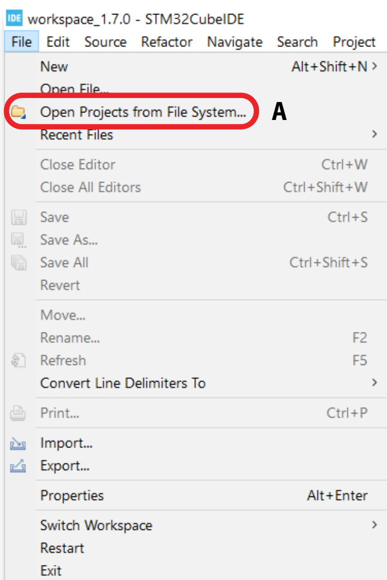


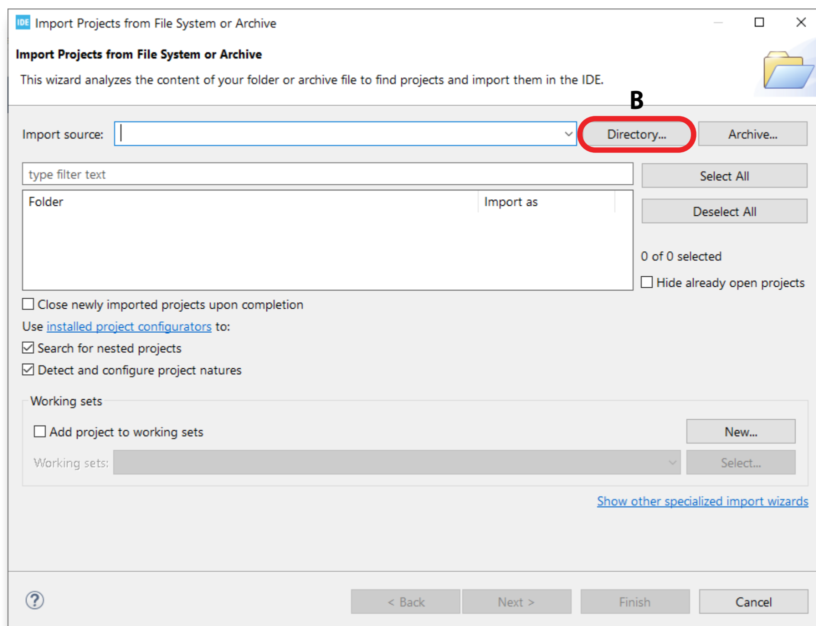
STM32CUBEIDE – IMPORT THE PROJECT

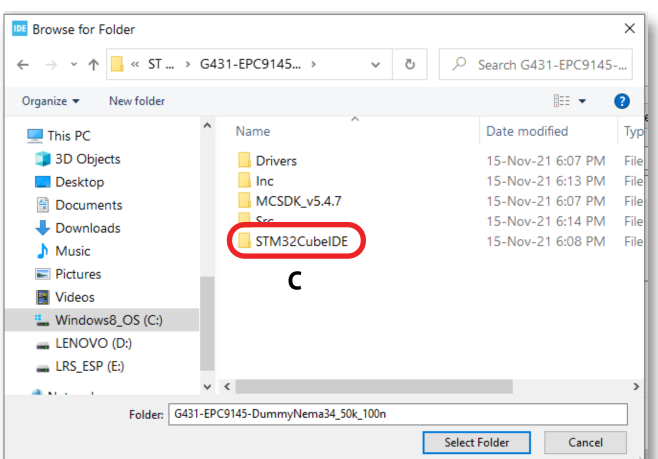
1. Open the **STM32CubeIDE** program from Windows Start Menu
2. Accept the proposed location for the Workspace and select the File menu
3. Under File menu choose **Open Projects from File System (A)**
4. Click on **Directory (B)**
5. Navigate to the generated code directory
6. Choose the **STM32CubeIDE sub-directory (C)** within the generated code directory (note: this directory was generated by STM32CubeMX program)
7. Press **Select Folder (D)**
8. Press **Finish button** on the Import Project window

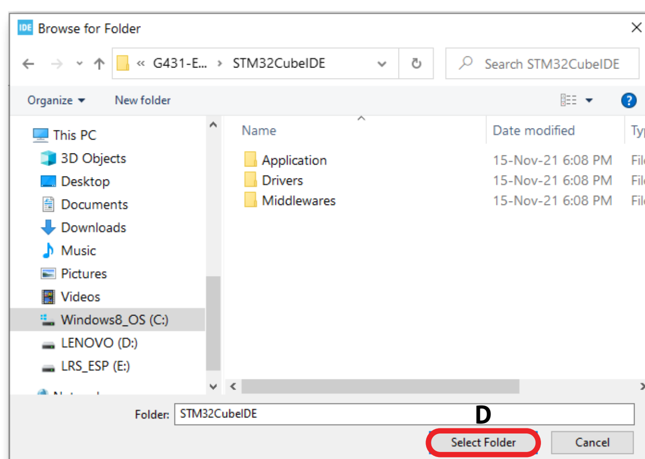
Note: the project must be imported only for the first time. All subsequent code re-generations on the same project will not require to import the folder again.

Note: it is better to have separate projects for different motors







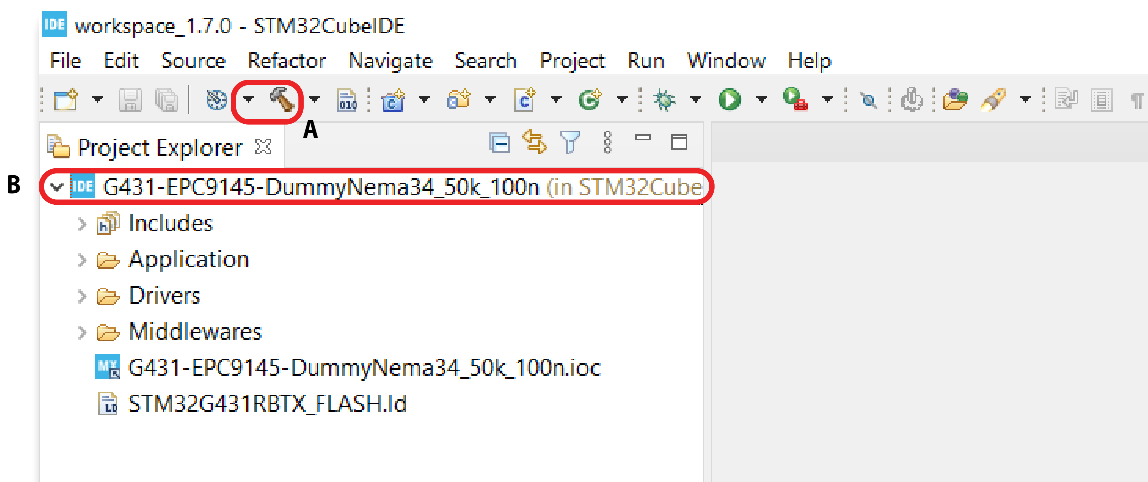


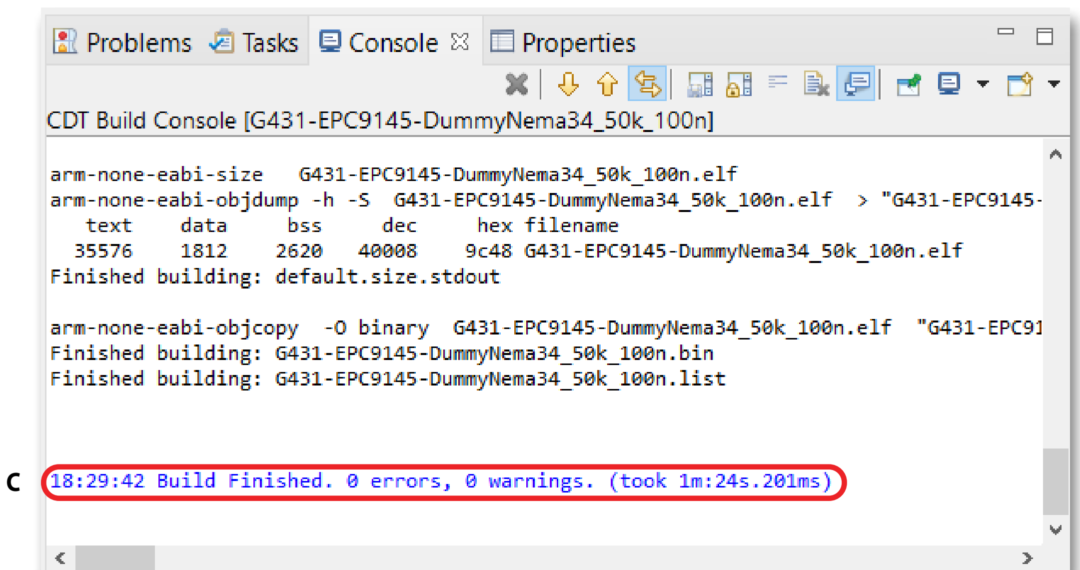
STM32CUBEIDE - BUILD THE CODE

Note: When you open STM32CubeIDE the first time you may land into the info window. In that case close the info window.
If Project Explorer window is not visible, go to the menu **Window**>>**Show View**>>**Project Explorer**

1. **Select the project (A)** on STM32CubeIDE program. In case of many projects, right click the one you want to select and set it as main project.
2. Press the **Hammer icon (B)**, it will launch the compiler and the builder.

At the end of the build process, **you must get a clean compile message in the CDT Build Console (C)**
Note: if you get any compile error, re-start the entire project from the ST Motor Control Workbench program.
Note: if you still get errors, contact a ST local FAE



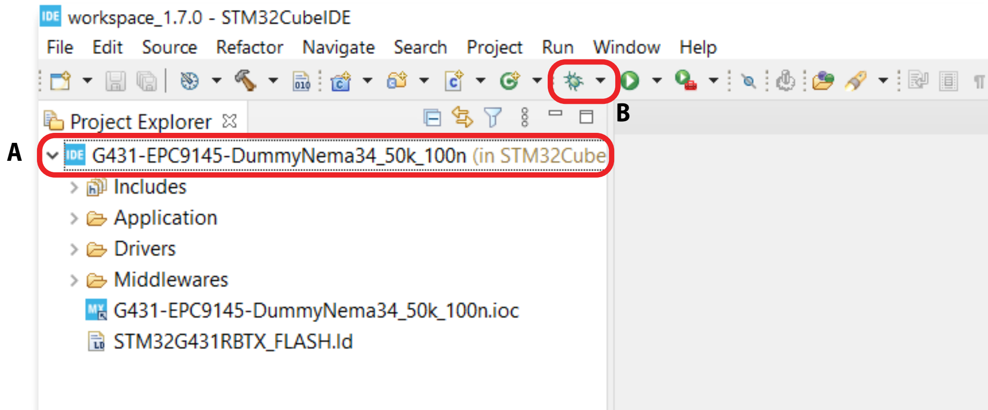


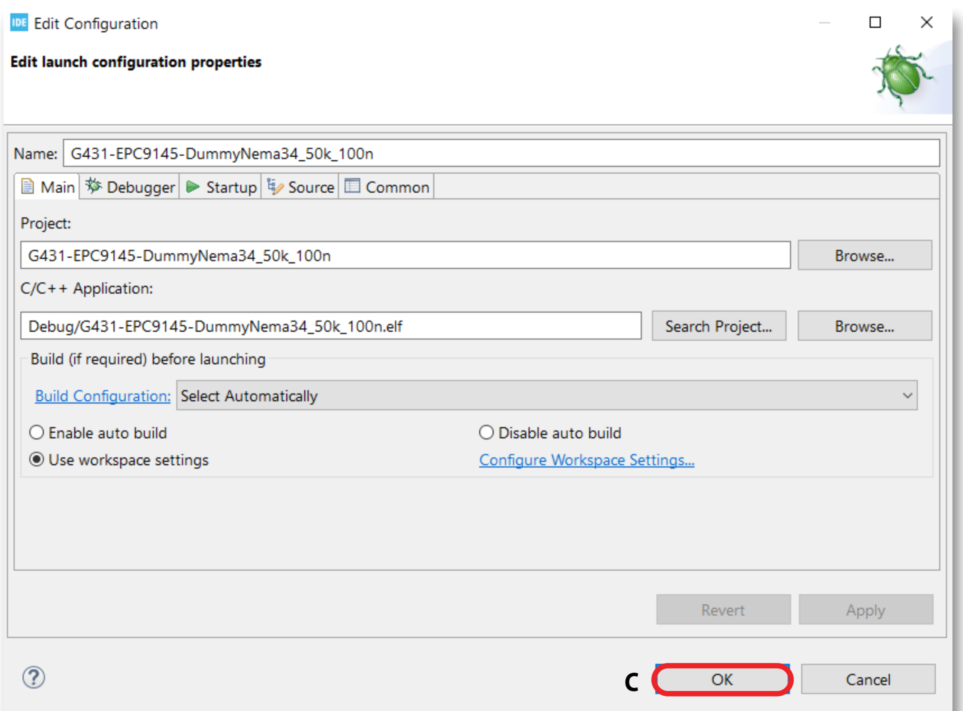
STM32CUBEIDE - FLASH AND DEBUG

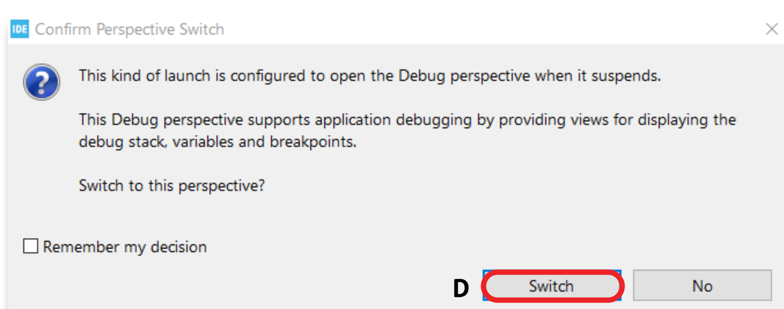
Select the project (A) on STM32CubeIDE program. In case of many projects, right click the one you want to select and set it as main project. Connect the Nucleo board with an USB micro cable to an USB port on the PC. Note: debugger is onboard of the ST Nucleo board.

1. Press the **Bug icon (B)**, it will launch first the flash downloader process and then the debugger.
2. The first time you will get two dialog windows as the one shown here, **press OK (C)** and then **Switch (D)**

When the download to the Flash is finished, and the controller is in Halt mode (for debug), close STM32CubeIDE and detach the controller from USB port.

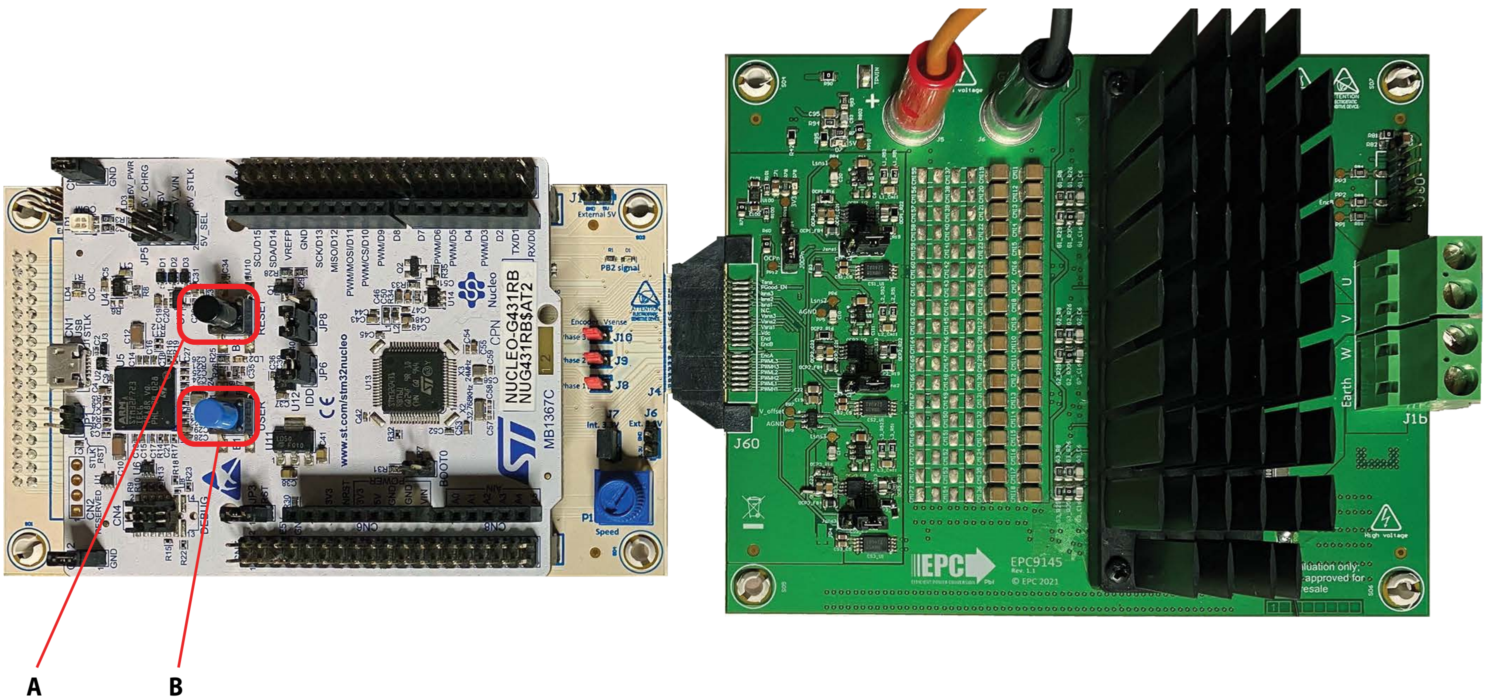






OPERATING THE MOTOR DRIVE SYSTEM DISCONNECTED FROM PC

1. Connect the EPC9147C to a compatible inverter board; e.g. EPC9145
2. Connect the motor to the inverter board. Follow QSG instructions.
3. With power **OFF**, connect the power supply to the inverter board. Make sure the jumpers are set as in the picture.
4. Set the power supply to the correct operating voltage for the inverter board. Make sure the current limit setting is sufficient to operate the motor drive system. For EPC9145 V_{sup} = 48 V and I_{lim} ≥ 2.5 A
5. Power on
6. **Press the black button (A)** on Nucleo board
7. **Press the blue button (B)** on Nucleo board to start and stop the motor



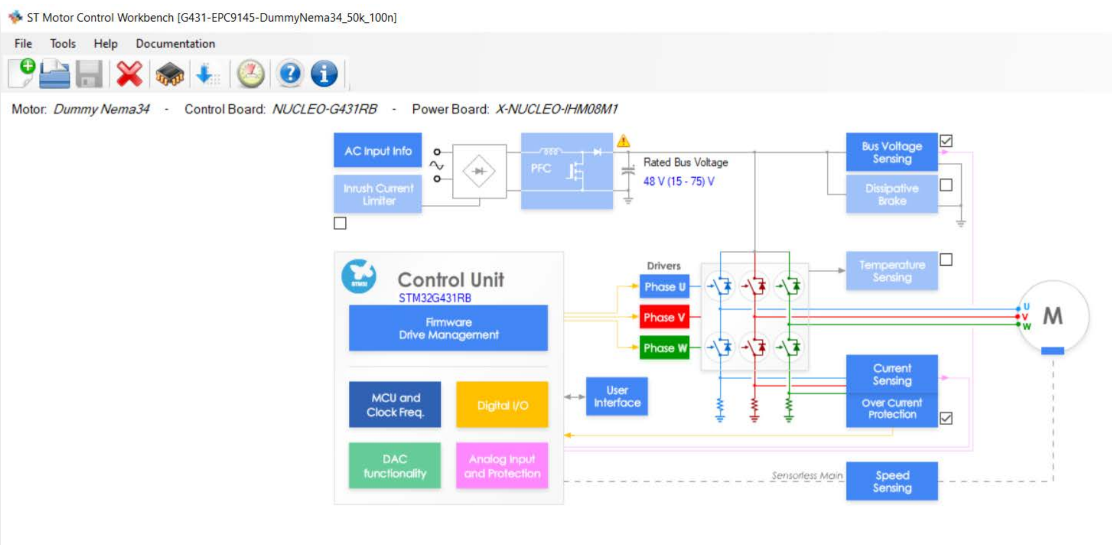
OPERATING THE MOTOR DRIVE SYSTEM CONNECTED TO THE PC AND USING ST MOTOR CONTROL WORKBENCH MONITOR OPTION

Launch the Monitor

When the ST Nucleo board is properly programmed and connected via the EPC9147C to the proper power board, it is also possible to use the ST Motor Control Workbench GUI to change the speed and the direction of the motor.

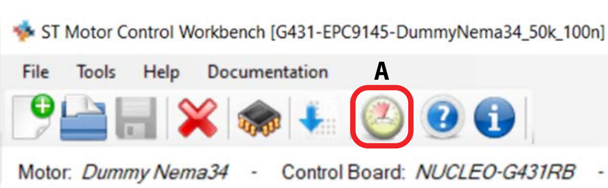
Open the ST Motor Control Workbench and load the proper .stmcx file that is relevant to the project you are working at (e.g. **G431-EPC9145-DummyNema34_50k_100n.stmcx**).

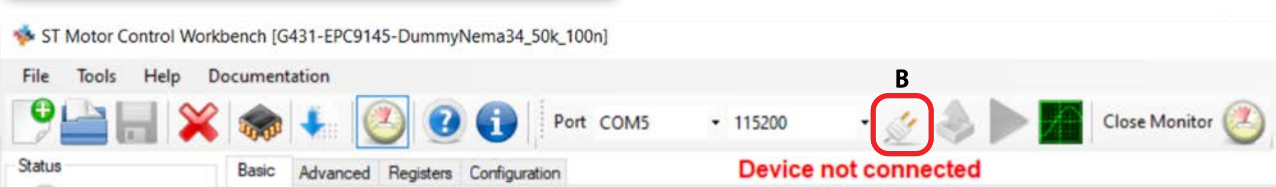
Note: Depending on the installation, you may have different versions.

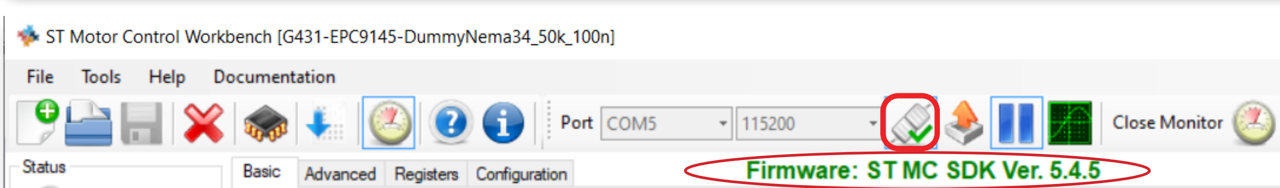


Monitor

1. Click on the **Open Monitor Button (A).**
2. Connect the USB cable to the PC and power up the 48 V to the power board. Click on the **Connect button (B).**
3. Wait for the **successful connection message (C).**







C

Press Fault Ack button (A) if any fault was detected. Then **press Start Motor button (B)**. Motor should spin. It is now possible to move the graphic potentiometer on the GUI to change the speed of the motor and to change the motor direction. Refer to ST user guide manual for more details on how to work with the ST Motor Control Workbench for further customization.

Refer to ST Motor Control Workbench Guide for further information

Note: ST Motor Profiler does NOT work with EPC power boards. If you try, you may damage the EPC power board.

For More Information:

Please contact **info@epc-co.com**
or your local sales representative

Visit our website:
www.epc-co.com

Sign-up to receive
EPC updates at
bit.ly/EPCupdates

EFFICIENT POWER CONVERSION

Demonstration Board Notification

The EPC9147A board is intended for product evaluation purposes only. It is not intended for commercial use nor is it FCC approved for resale. Replace components on the Evaluation Board only with those parts shown on the parts list (or Bill of Materials) in the Quick Start Guide. Contact an authorized EPC representative with any questions. This board is intended to be used by certified professionals, in a lab environment, following proper safety procedures. Use at your own risk.

As an evaluation tool, this board is not designed for compliance with the European Union directive on electromagnetic compatibility or any other such directives or regulations. As board builds are at times subject to product availability, it is possible that boards may contain components or assembly materials that are not RoHS compliant. Efficient Power Conversion Corporation (EPC) makes no guarantee that the purchased board is 100% RoHS compliant.

The Evaluation board (or kit) is for demonstration purposes only and neither the Board nor this Quick Start Guide constitute a sales contract or create any kind of warranty, whether express or implied, as to the applications or products involved.

Disclaimer: EPC reserves the right at any time, without notice, to make changes to any products described herein to improve reliability, function, or design. EPC does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights, or other intellectual property whatsoever, nor the rights of others.