

RT Box

DEMO MODEL

Brushless DC Machine Demo Application for LaunchPad Interface Board

Last updated in RT Box Target Support Package 2.2.1

www.plexim.com

- ▶ Request a PLECS and PLECS Coder trial license
- ▶ Get the latest RT Box Target Support Package
- ▶ Check the PLECS and RT Box documentation

1 Overview

The LaunchPad interface board from Plexim ships with a pre-programmed LaunchXL-F28069M so users can quickly start using the RT Box with an example hardware-in-the-loop application. The control logic pre-programmed on the microprocessor (MCU) is a basic trapezoidal control application for a brushless DC (BLDC) machine. This demo includes an RT Box model of the BLDC drive to use with the pre-programmed MCU and shows the basic steps required to use the RT Box.

The discretization step size and average execution times for this model are shown in Fig. 1.

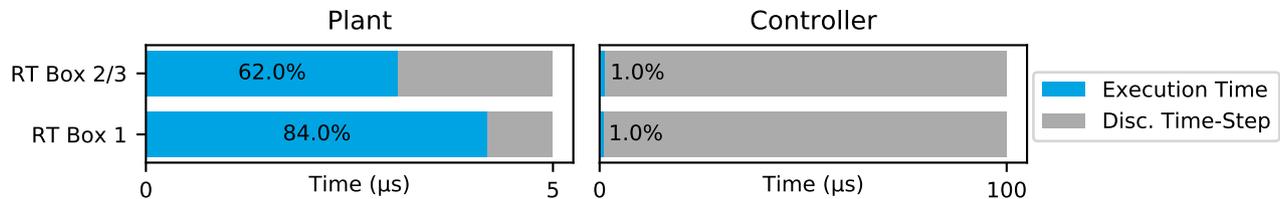


Figure 1: Performance overview for the execution on two RT Boxes

1.1 Requirements

Before you start please make sure you have the following items ready:

- One PLECS RT Box 1, 2 or 3.
- A PLECS Coder license. You can request this license from Plexim at www.plexim.com.
- The RT Box Target Support Library
- A LaunchPad Interface Board
- A TI Launchpad LaunchXL-F28069M. The LaunchPad Interface Board comes with a pre-programmed 28069. In case you are using your own launchpad without the pre-programmed demo application, please follow the instructions given in Section 3.5
- For initial setup of the RT Box, follow the step-by-step instructions on configuring PLECS and the RT Box in the Quick Start guide of the RT Box User Manual.

Note This model contains model initialization commands that are accessible from:

PLECS Standalone: The menu **Simulation + Simulation Parameters... + Initializations**

PLECS Blockset: Right click in the **Simulink model window + Model Properties + Callbacks + InitFcn***

2 Model

The PLECS simulation model is divided into two subsystems titled “Plant” and “Controller”, as shown in Fig. 2. The “Plant” subsystem represents the BLDC drive model that is configured for real-time execution on the RT Box. The pre-programmed MCU will perform closed-loop current control when connected to an RT Box running the “Plant” subsystem. The “Controller” subsystem is configured to run on a second RT Box, if available. The “Controller” subsystem is a representation of the control logic pre-programmed on the MCU and is pin-compatible with the LaunchPad interface connections.

Both subsystems are enabled for code generation from the **Edit + Subsystem + Execution settings...** menu. This step is necessary to generate the model code for a subsystem via the PLECS Coder.

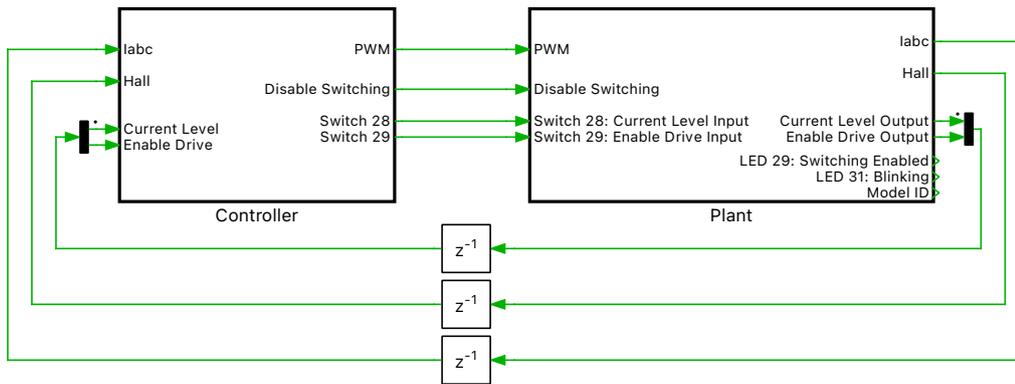


Figure 2: Top level schematic of the plant and the controller subsystems

2.1 Power Circuit

The power circuit consists of a 24 V DC source, a three-phase inverter, and a BLDC machine as shown in Fig. 3. The machine is connected to a mechanical load modeled as an inertia and speed-dependent friction component.

The rotor's position is sensed using three Hall sensors. Hall sensors detect the magnetic field polarity of the permanent magnets in the BLDC machine. The sensors are arranged with 120 electrical degrees of separation, such that one Hall sensor state will change every 60 electrical degrees. In the simulation model, the machine's angle is converted into three Hall sensor states, which are connected to the controller through digital outputs.

The current sensing approach models the characteristics of a TI BOOSTXL DRV8301 board [1], which uses low-side shunts to measure the machine stator currents. The measured currents are connected to the analog output pins of the RT Box. The scaling and offset of the RT Box analog output represents the current sensing circuitry of the TI BOOSTXL DRV8301 hardware.

Additional digital IO are used to interface with the LaunchPad controller and display status information on the LaunchPad Interface Board. An active-low PWM enable signal is set via DI22 modeling the gate-enable logic of the TI BOOSTXL DRV8301. Two LEDs on the LaunchPad interface board display the status information, with LED DO29 indicating the switching signal is active and LED DO31 blinking at a rate of 1 Hz while the model is running.

Switch DI29 is used to activate and restart the controller. Switch DI28 changes the reference value of the current control loop. The switch states are sensed via digital inputs on the RT Box and then are relayed to the GPIO on the MCU via RT Box digital outputs DO4 and DO18.

Note Switch DI29 of interface board enables or disables the MCU PWM outputs.

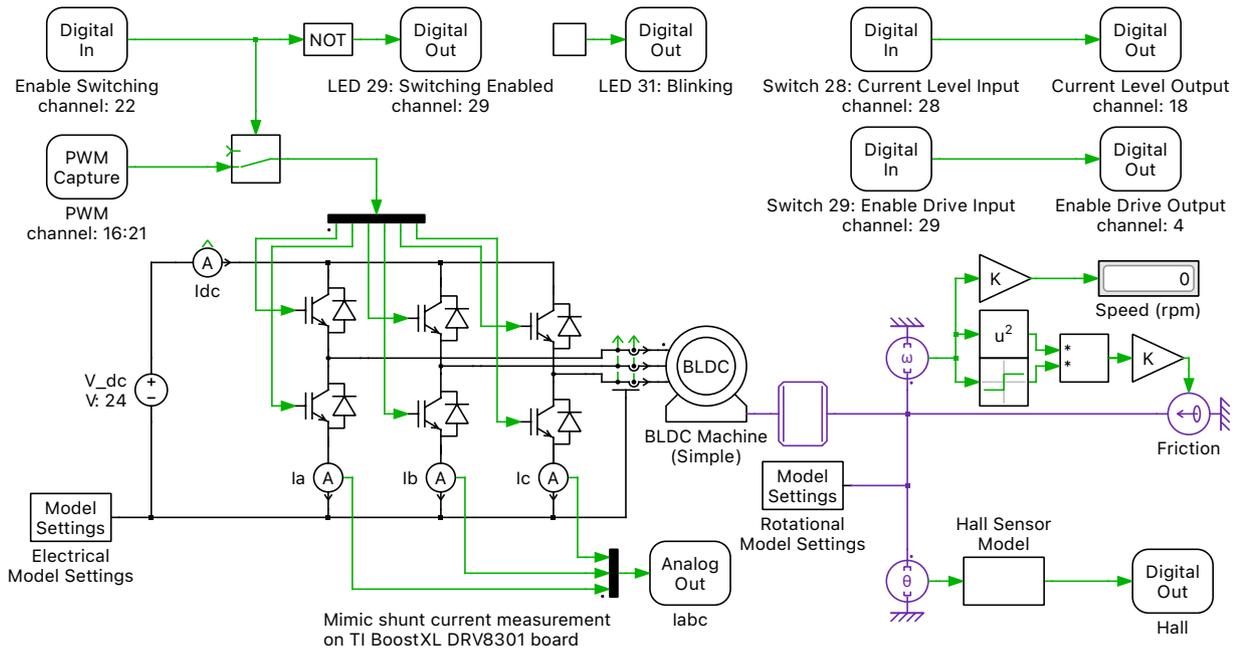


Figure 3: BLDC drive with mechanical load

Table 1 shows the key pin assignments for this demo model.

Table 1: BLDC I/O pin map

Feature	RT Box Channel	LaunchPad Pin
Ia	AO12	J7 67
Ib	AO13	J7 68
Ic	AO14	J7 69
Hall A	DO2	J1 5
Hall B	DO6	J2 13
Hall C	DO7	J2 12
PWM A _H	DI16	J8 80
PWM A _L	DI17	J8 79
PWM B _H	DI18	J8 78
PWM B _L	DI19	J8 77
PWM C _H	DI20	J8 76
PWM C _L	DI21	J8 75

2.2 Control

The control logic is pre-programmed on the provided MCU and that should be connected to the RT Box via the LaunchPad Interface Board. In addition to using the pre-programmed MCU as the controller, the demo model includes an analogous control system that can be deployed to a second RT Box. Fig. 4 shows an overview of the control approach.

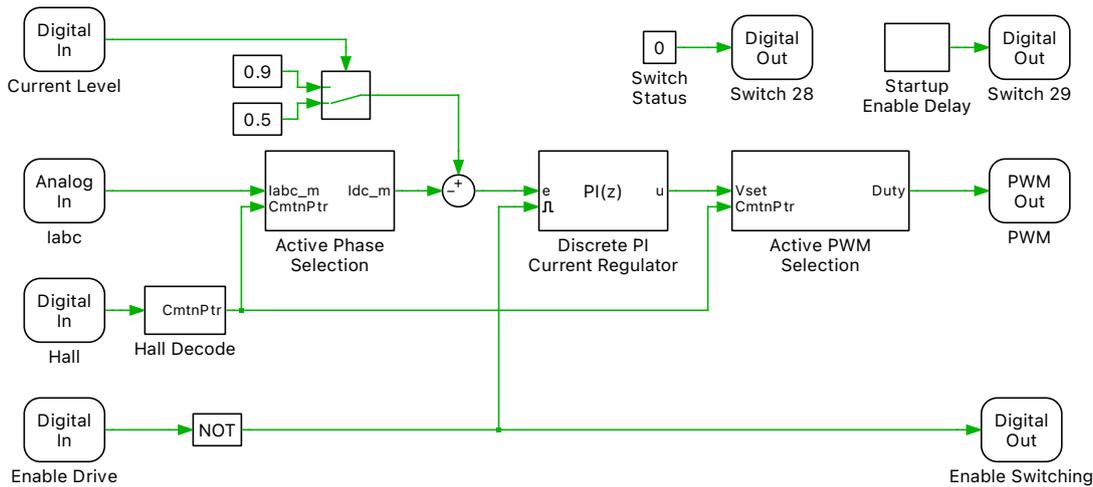


Figure 4: Control logic for the RT Box

The control system implements a trapezoidal control of a BLDC motor [2, 3]. Trapezoidal control is also referred to as six-step control or two-phase on control. The approach divides an electrical revolution of the BLDC into six sectors of 60 electrical degrees. The three Hall sensor readings are decoded into a sector reading that indicates the approximate rotor position. Table 2 shows the relationship between the machine's electrical angle, sector, hall sensor reading, and desired machine phase currents.

In every sector, only two inverter legs are active while the third one is always deactivated. For example, in Sector 1 the upper switch in the B phase half bridge will modulate and the lower switch in the B phase half bridge will remain closed, establishing a positive DC current flowing into the B phase. An equal and opposite current will flow into the A phase of the machine, while phase C remains disconnected.

Table 2: BLDC commutation table

Rotor Electrical Angle (θ_e)	Sector	Hall A	Hall B	Hall C	Ia	Ib	Ic
$210^\circ \leq \theta_e < 270^\circ$	1	0	1	0	-	+	0
$270^\circ \leq \theta_e < 330^\circ$	2	0	1	1	-	0	+
$330^\circ \leq \theta_e < 30^\circ$	3	0	0	1	0	-	+
$30^\circ \leq \theta_e < 90^\circ$	4	1	0	1	+	-	0
$90^\circ \leq \theta_e < 150^\circ$	5	1	0	0	+	0	-
$150^\circ \leq \theta_e < 210^\circ$	6	1	1	0	0	+	-

Energizing the machine's phases in the correct sequence establishes a rotating magnetic field. Fig. 5 shows a conceptual graphic of the full commutation sequence. The A, B, and C coils are identified in the figure, along with the Hall readings within each sector. The phase current vectors are shown as dashed magenta lines, the direction of the magnetic flux generated by the stator coils as a solid magenta vector, and the cyan vector indicates the position of the rotor. When the rotor is in Sector 1, the flux generated by the stator coils will lead the rotor by $90 \pm 30^\circ$ degrees. As the rotor crosses into Sector 2, indicated by the change in Hall state, then the PWM pattern will change to establish a current in the C and A phases. The resulting flux generated by the stator coils will once again lead the rotor by $90 \pm 30^\circ$ degrees.

A discrete PI regulator controls the phase current magnitude. The "Active Phase Selection" subsystem determines which phase current is the reference for the current controller. The correct phase is

selected using a 1D look-up table that depends on the commutation interval. The “Active PWM Selection” subsystem sets which PWMs are active during each commutation interval and uses a similar look-up table based implementation.

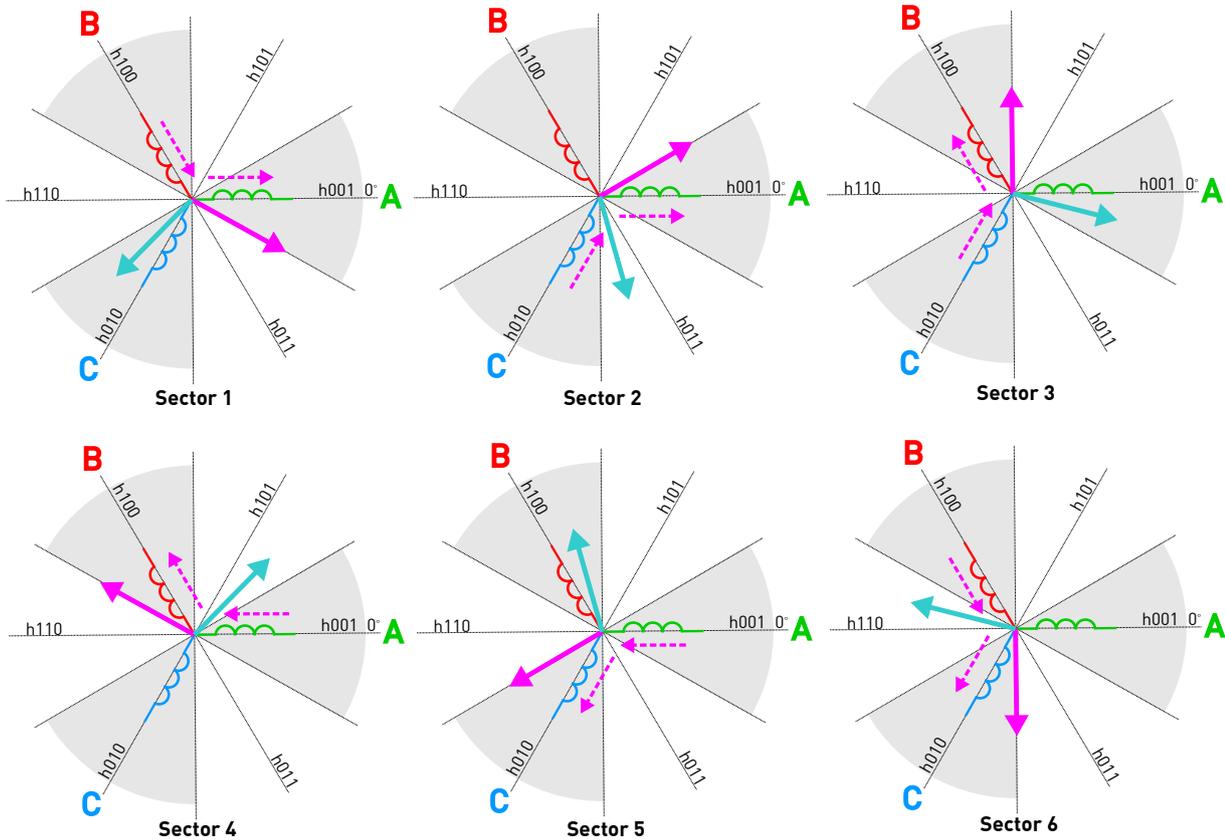


Figure 5: BLDC commutation sequence with current vectors (magenta) and rotor alignment (cyan)

3 Simulation

The simulation model can run offline in PLECS on a desktop PC prior to testing in real-time. For real-time simulation, the plant model of the BLDC and drive system is always deployed to the PLECS RT Box. The controller can be either the pre-programmed LaunchPad device or a second RT Box.

3.1 Program the RT Box with the LaunchPad controller

This section describes the process of uploading a demo model to the RT Box to use with the LaunchPad Interface Board. The connection between the RT Box and LaunchPad is shown in Fig. 6.

Before you begin, verify the following hardware configuration on your Launchpad Interface Board.

- *JP3*, *JP4*, *JP5* and *JP7* on the LaunchPad are closed.
- *JP1*, *JP2* and *JP6* on the LaunchPad are open.
- Dip switches 1 and 2 on the LaunchPad are pointing away from the processor.
- Dip switch 3 on the LaunchPad is pointing towards the processor.
- The *RST* jumper on the LaunchPad Interface Board is open.

Follow the instructions below to run a real-time model on the RT Box.



Figure 6: Hardware setup of the HIL verification with the RT Box

- Click the **Coder** drop-down in the top-menu bar and select **Coder options....** You will see a window similar to Fig. 7.
- In the **System** menu shown on the left side of the window, select **Plant**.
- Switch to the **Target** tab. Select the **Target Device** by clicking the  icon and choosing an RT Box from the list of available devices.
- Click **Accept** and then **Build**. Your model is now compiled and downloaded to the RT Box automatically.
- Verify that the blue **Running** LED on the RT Box 1 is illuminated or the display on the RT Box 2 and 3 updates.

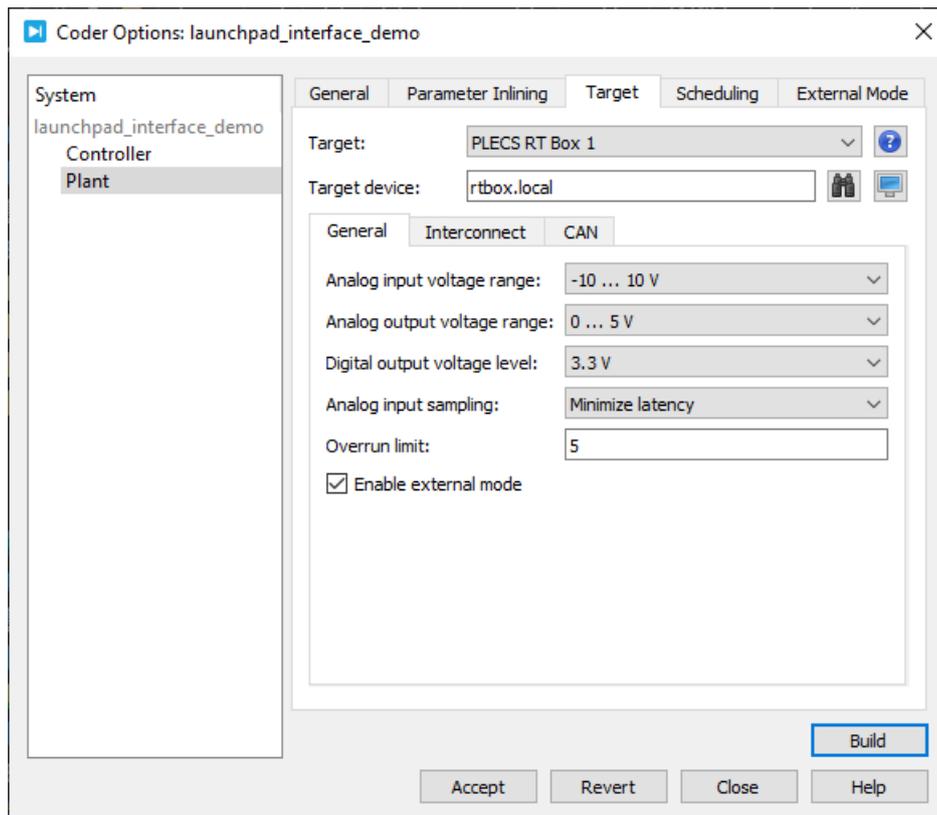


Figure 7: Programming the RT Box

3.2 Connecting the External Mode

The External Mode enables access to the real-time simulation executed on the RT Box. It can be used to visualize simulation signals from the hardware via the model scopes or update and change tunable model parameters.

Switch to the **External Mode** tab in the **Coder options...** window, as shown in Fig. 8. Ensure that the **Plant** is still selected in the left-side menu and then click **Connect** to start communication between PLECS and the model running on the RT Box. **Activate autotriggering** via the appropriate button.

Set Switch *DI-29* to *high* to enable the drive control. Open the Scopes in the Plant model and analyze the control behavior. Fig. 9 shows the data from the *Electrical* Scope in the model, showing the PWM signals and machine phase currents using two-phase on control.

Note Switch *DI-28* can be used to change the current reference of the control system.

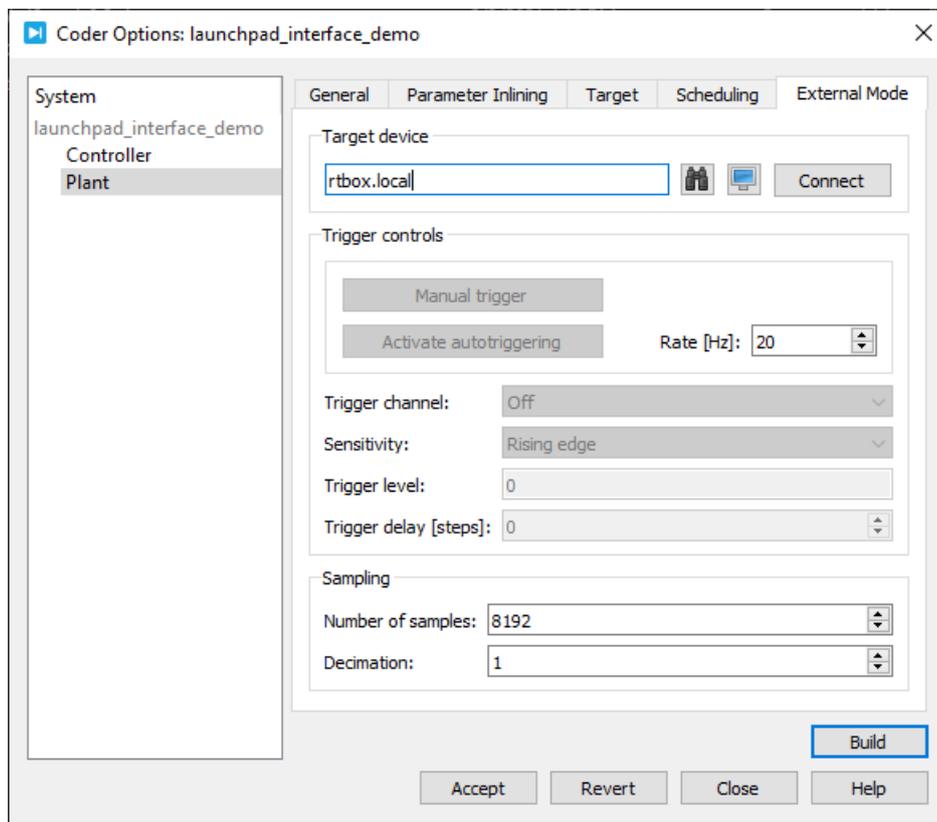


Figure 8: Connecting to the BLDC Model via the External Mode

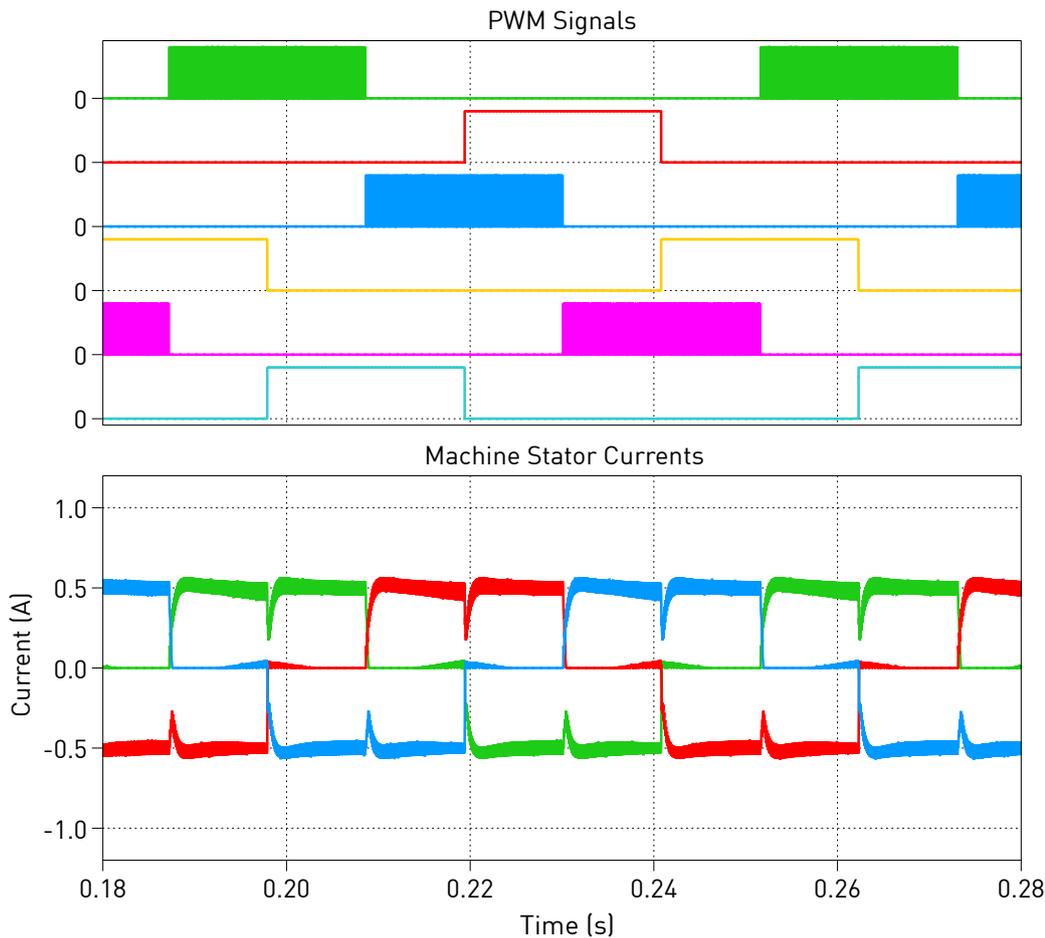


Figure 9: Scope capture showing two-phase on PWM signals and stator currents

3.3 RT Box Web Interface

The Web Interface provides information about the model running on the RT box as well as additional diagnostic options. It can be accessed by clicking on the  icon under the **Target** or the **External Mode** tabs of the Coder Options dialog. The web interface window is shown in Fig. 10.

The processor load statistic reveals information about the time needed to calculate the model and therefore serves as a convenient tool to validate the chosen step size. Do not overload the processor, maintain a safety margin.

Note A model under actuation requires a higher processing time than an idle model. Additional processor load is required using the external mode.

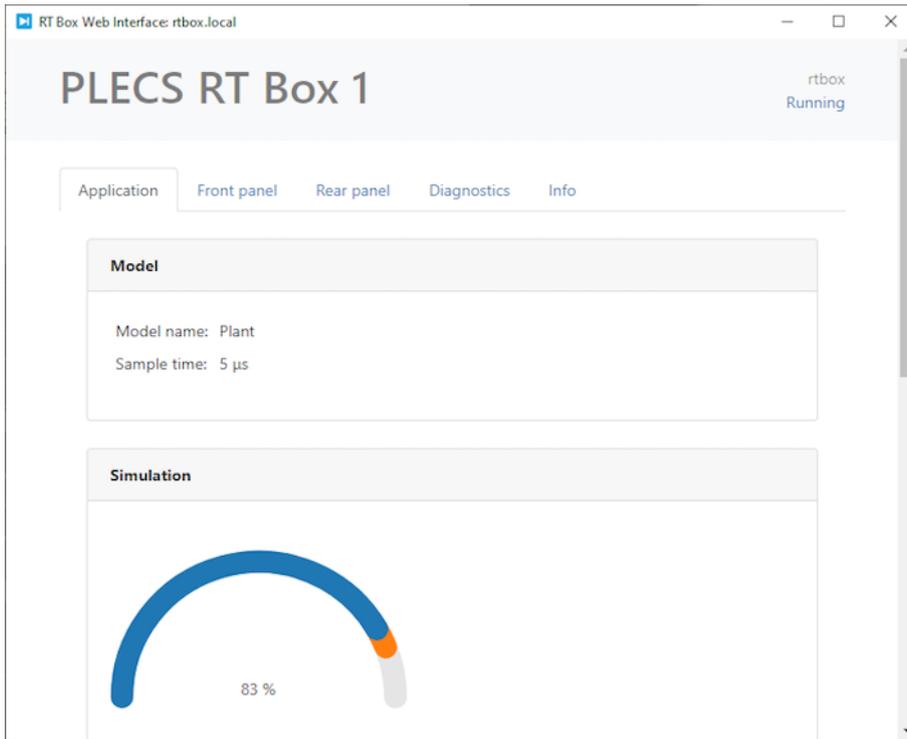


Figure 10: RT Box Web Interface

3.4 Program the RT Box with a second RT Box controller

A second RT Box can be used as the controller instead of the pre-programmed LaunchPad. The two RT Boxes (referred to as “Plant” and “Controller”) need to be set up as demonstrated in Fig. 11 using three DB37 cables. Then, program the second RT Box with the “Controller” model using the steps outlined in Section 3.1.

One can also connect to the Controller RT Box via the External Mode. The *Switch Status* and current reference set points are configured so they can be modified while connected via the External Mode. These constants have been added to the *Exceptions* list found in the **Parameter Inlining** tab of the **Coder options...** window, prior to building the model.

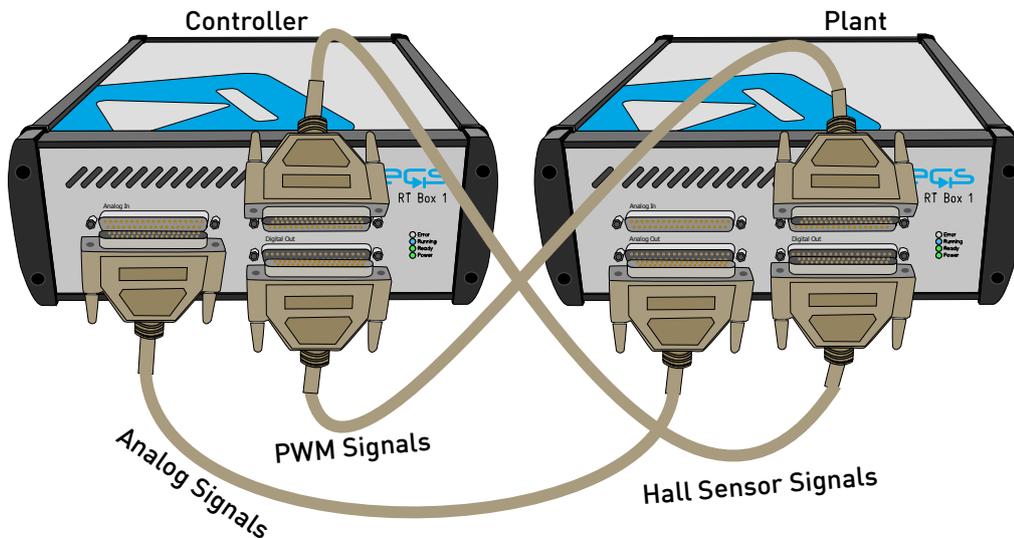


Figure 11: Hardware configuration for the real-time operation of the demo model with two RT Boxes

3.5 Uploading the Firmware

The control required to run the demo models is pre-programmed on the TI Launchpad and is ready to use out of the box. The following section shows how to re-program the MCU with the demo application or perform an update. Otherwise, you can simply skip this section. Please note that this section is applicable for Windows machines only.

Switch off the RT Box. Make sure that all jumpers on the LaunchPad, except *JP6*, are closed and all dip switches are pointing away from the processor.

Connect the JTAG/SCI USB port of the LaunchPad to your PC. Open the Windows Device Manager and confirm that TI Debug Probes are listed.

You may have to install the FTDI drivers if the port is not enumerated.

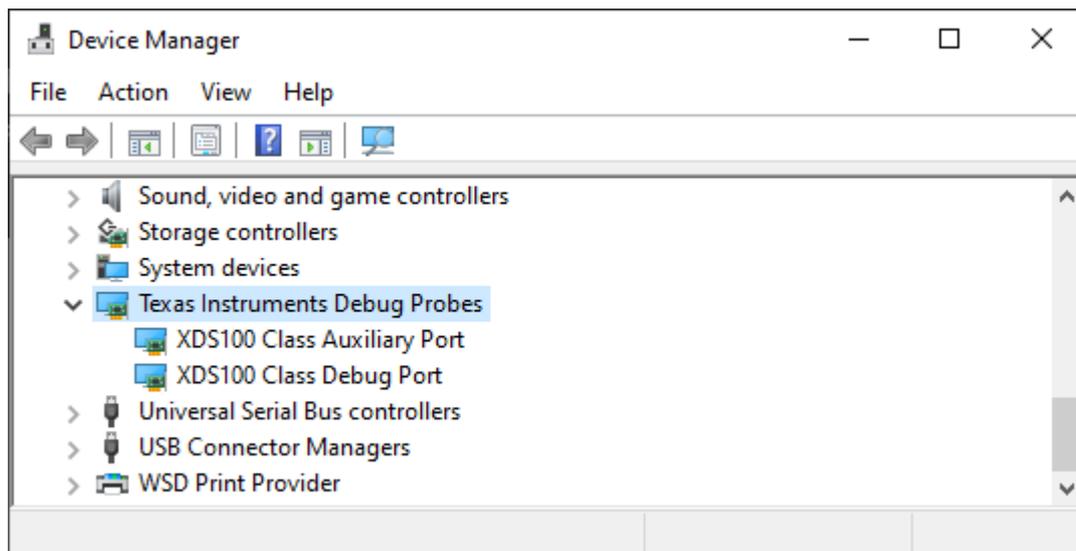


Figure 12: TI debug probes listed in Windows Device Manager

The pre-compiled executable is located in the demo package folder, in the same folder as the PLECS simulation model. Two files are provided. The `launchpad_interface_demo_28069.ehx` file is for use with C2Prog [4] using the steps outlined below. A second file, `launchpad_interface_demo_28069.out`, can be used with TI's Uniflash tool [5] to program the device.

In C2Prog, select the file `launchpad_interface_demo_28069.ehx` and configure the port to XDS100v2. Click the **Program** button.

Once the reflashing completes, disconnect the USB cable and toggle dip switch 3 which should now be pointing towards the processor. Open *JP1* and *JP2* on the LaunchPad. The LaunchPad is now ready for operation.

4 Conclusion

The LaunchPad Interface Board arrives with a pre-programmed LaunchXL-F28069M. The control logic on the MCU is for a trapezoidal control application of a BLDC drive. This model shows how to use the RT Box along with the demo application and shows the basic steps required to use the RT Box.

References

- [1] BOOSTXL-DRV8301 Motor Drive BoosterPack featuring DRV8301 and NexFET™ MOSFETs
 URL: <https://www.ti.com/tool/BOOSTXL-DRV8301>

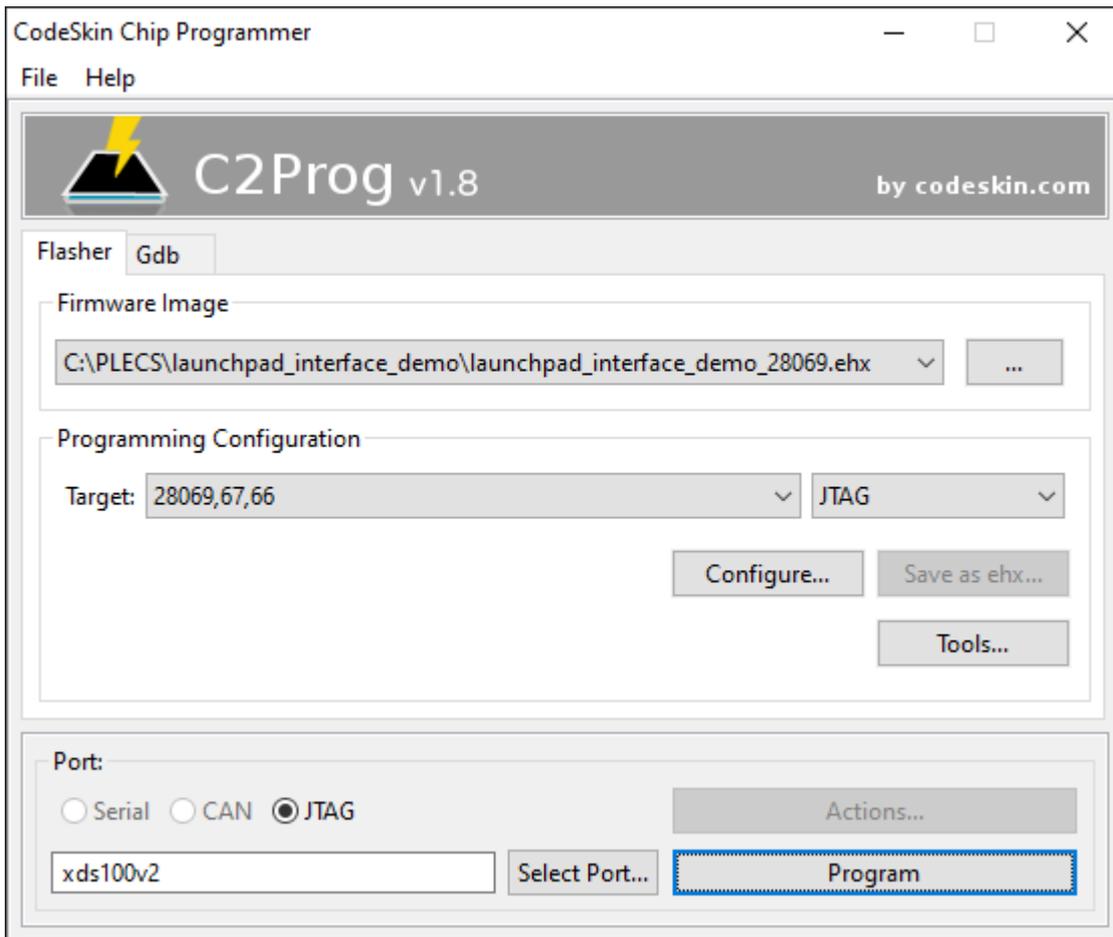


Figure 13: Flashing the LaunchPad with C2Prog

- [2] B. Akin and M. Bhardwaj, Sensorless Trapezoidal Control of BLDC Motors, Texas Instruments, 2015.
- [3] Toshiba Electronic Devices & Storage Corporation, 120° Square-Wave Commutation for Brushless DC Motors, 2018.
- [4] C2Prog Free Flash Programmer for TI MCUs
URL: <https://www.codeskin.com/c2prog-download>
- [5] UNIFLASH UniFlash stand-alone flash tool for microcontrollers, Sitara™; processors and SimpleLink™
URL: <https://www.ti.com/tool/UNIFLASH>

Revision History:

RT Box Target Support Package 1.1.1	First release
RT Box Target Support Package 2.2.1	Resolve PIL dependencies

How to Contact Plexim:

☎	+41 44 533 51 00	Phone
	+41 44 533 51 01	Fax
✉	Plexim GmbH Technoparkstrasse 1 8005 Zurich Switzerland	Mail
@	info@plexim.com	Email
	http://www.plexim.com	Web

RT Box Demo Model

© 2002–2022 by Plexim GmbH

The software PLECS described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from Plexim GmbH.

PLECS is a registered trademark of Plexim GmbH. MATLAB, Simulink and Simulink Coder are registered trademarks of The MathWorks, Inc. Other product or brand names are trademarks or registered trademarks of their respective holders.