

RT Box

DEMO MODEL

H-Bridge Converter

Last updated in RT Box Target Support Package 3.1.1

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 TSP documentation

1 Overview

This RT Box demo model features a current-controlled H-bridge circuit powering an inductive load. The model is split into plant and controller parts. The plant contains an H-bridge converter with inductive load, and the controller employs a proportional-integral (PI) current control scheme. The following sections provide a brief description of the model and instructions on how to simulate it.

Real-time execution on the RT Box requires the model to execute using a fixed-step solver. The discretization step size parameter specifies the base sample time of the generated code and is used to discretize the physical model and control domain state-space equations. The execution time represents the actual time it takes to execute one calculation step of the PLECS model on the RT Box hardware. The chosen discretization step sizes and average execution times for each core running the H-bridge converter model are shown in Tab. 1.

Table 1: Discretization step size and average execution time of the demo on one RT Box

	Core 0: exec. time / step size	Core 1: exec. time / step size
RT Box 2, 3 or 4	1.9 μs / 2 μs	1 μs / 100 μs
RT Box 1	1.9 μs / 2 μs	N/A

1.1 Requirements

To run this demo model, the following items are needed (available at www.plexim.com[1]):

- One PLECS RT Box[2] and one PLECS[3] and PLECS Coder[4] license
- The RT Box Target Support Package[5]
- Follow the step-by-step instructions on configuring PLECS and the RT Box in the Quick Start guide of the RT Box User Manual[6].
- Two 37 pin Sub-D cables to connect the box in loop-back setup at the front panel.

Note that this demo model primarily showcases the RT Box `multi-tasking` mode. All RT Boxes support multi-tasking, but the implementation differs depending on the model.

- When the target is an RT Box 2, 3 or 4, the main CPU core (Core 0) runs the plant as "Base task" with a sample time of `Ts_plant`. Another core (Core 1) runs closed-loop controls in "Controller" task in parallel with a sample time of `Ts_controller`, which is much slower and usually equals the switching period of the converter. In this way, the multi-core feature of the RT Box 2, 3 or 4 is showcased by splitting the computational effort onto different cores. Besides, the setup can easily transition to a HIL or RCP test later on.
- However if the user has only a single RT Box 1 available, this model can also run with the multi-tasking feature onto the only CPU core of the RT Box 1, but in a pre-emptive multi-tasking fashion. In this case, the "Base task" is doing the plant calculation with the highest priority with a sample time of `Ts_plant`. The "Controller" task is executed as a background task with lower priority at the sample time of `Ts_controller`.

Please check the setting under **Scheduling** tab of the **Coder options...** window.

[1] https://www.plexim.com
[2] https://www.plexim.com/products/rt_box
[3] https://www.plexim.com/products/plecs/plecs_standalone
[4] https://www.plexim.com/products/plecs/plecs_coder
[5] https://www.plexim.com/download/rt_box
[6] https://www.plexim.com/sites/default/files/rtboxmanual.pdf

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 top-level schematic is shown in Fig. 1. It contains one subsystem named “Plant + Controller”. Inside the subsystem both the plant and the controls are modeled. The subsystem is enabled for code generation from the **Edit + Subsystem + Execution settings...** menu. This step is necessary to generate the model code for the RT Box.

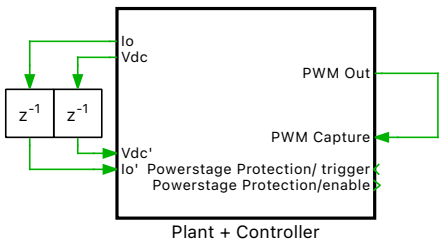


Fig. 1: Top-level schematic of the H-bridge converter model

2.1 Power Circuit

The power circuit, shown in Fig. 2, consists of an H-bridge which drives an inductive load and is supplied by a DC voltage source with $V_{\mathrm{dc}} = 24\,\mathrm{V}$. The H-bridge is modeled using a Full-Bridge Inverter component available in the Nanostep section of the PLECS library. The RT Box's Nanostep solver simulates the converter with time steps in the single-digit nanosecond range. The pulse-width modulated (PWM) switching signals are obtained from the PWM Capture block of the PLECS RT Box library. The PWM Capture block samples incoming switching signals at the speed of the Nanostep solver execution interval. This sampling interval is 7.5 nanoseconds on the RT Box 1 and CE and 4 nanoseconds on the RT Box 2, 3 and 4. The measurements of DC input voltage and output inductor current are sensed and exported out of the subsystem via Analog Out blocks from the PLECS RT Box library. The discretization step size of the plant subsystem is set to $2\,\mu\mathrm{s}$.

Scaling Analog Outputs

The limits on Analog input and output voltage ranges of the of the PLECS RT Box target can be set in between $-10\,\mathrm{V}$ to $10\,\mathrm{V}$ from the **Target** tab of the **Coder options...** window.

However, in a typical HIL simulation, since the Analog Out values exported out of the RT Box are fed to an embedded controller, these values are scaled and offset to be within $0\,\mathrm{V}$ to $3.3\,\mathrm{V}$, to satisfy voltage requirements of of the MCU analog-to-digital converters (ADCs).

The plant model is based on a hardware prototype of the power circuit, comprising two phases of a BOOSTXL-DRV8305EVM BoosterPack [2] connected to an external RL load. Therefore the input voltage and the shunt currents are scaled according to the voltage and current sense circuits of the BoosterPack.

Voltage Sensing

A similar voltage sensing circuit as described in [2] is shown in Fig. 3.

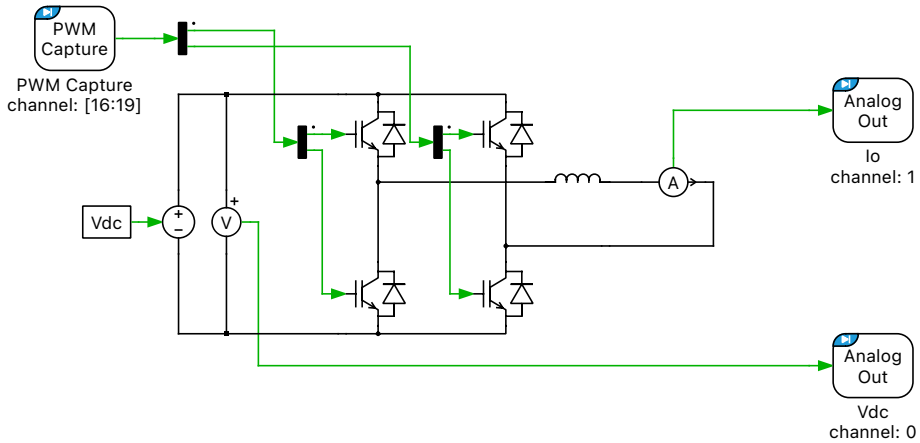


Fig. 2: Power circuit of the H-Bridge with inductive load

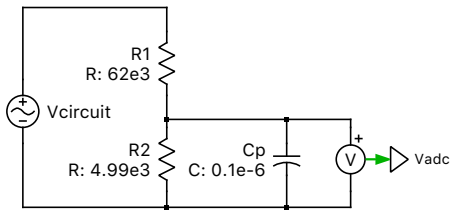


Fig. 3: Schematic of a voltage sensing circuit

The equations below show the response of the voltage sensing circuit. A capacitor, C_p can be selected to create a pole in the sensing circuit to attenuate the switching frequency ripple on the sensed voltage. The pole frequency is denoted as f_p below.

$$\frac{\mathrm{V_{adc}}}{\mathrm{V_{circuit}}} = \frac{\mathrm{R_2}}{(\mathrm{R_1}+\mathrm{R_2}) + s\mathrm{R_1R_2C_p}},\ \mathrm{C_p} = \frac{\mathrm{R_1}+\mathrm{R_2}}{2\pi \mathrm{f_p R_1 R_2}}$$

The voltages are scaled with a gain of

$$\mathrm{Vm.K} = \frac{\mathrm{R_2}}{\mathrm{R_1}+\mathrm{R_2}} = \frac{4990}{62000+4990}$$

Current Sensing

As described in [2], the currents are sensed and amplified with a scaling factor of $Im \cdot K$ and offset with offset factor of $Im \cdot O$ using a current shunt amplifier.

$$\mathrm{Im.K} = 0.007 * 10$$

$$\mathrm{Im.O} = \frac{3.3}{2}$$

2.2 Controls

The controller part, shown in Fig. 4, receives the sensed inductor current and the DC input voltage values in the Analog In blocks of the PLECS RT Box component library. These values are inversely scaled to reverse the scaling applied at the Analog Out blocks of the plant part.

The “Right Leg Duty-Cycle” subsystem determines the duty cycle required to maintain an average of $12\,\mathrm{V}$ at the right leg H-bridge output while accounting for variations in the sensed input voltage V_dc.

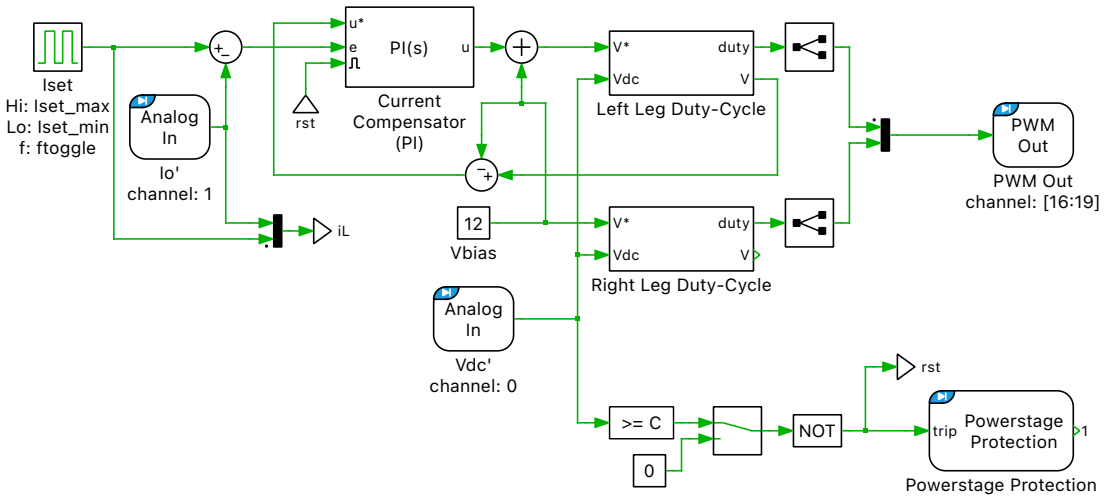


Fig. 4: Controller of the H-Bridge circuit

The "Left Leg Duty-Cycle" subsystem determines the modulation index of left leg of the H-bridge based on a proportional-integral (PI) controller. The sensed inductor current is compared to a setpoint that is toggled between $-3\,\mathrm{A}$ and $3\,\mathrm{A}$. This error is used for current compensation by the digital PI controller, which is equipped with anti-windup logic.

A few of the concepts described in this Section "Controls", related to the controller development, are based on reference [3].

Plant transfer function

To set the PI controller gain parameters a plant transfer function $P(s)$ is needed. $P(s)$ relates the change of the voltage across the inductor (L) and resistor (R), V_{RL} (the input variable), to the response of the inductor current I_{L} (the output variable):

$$P(s) = \frac{I_{\mathrm{L}}}{V_{\mathrm{RL}}} = \frac{1/R}{1 + s\,L/R} = \frac{K_1}{1 + s\,T_1}, \quad \text{where } K_1 = 1/R \text{ and } T_1 = L/R$$

Time constants

For reducing the order of complex systems, it is often useful to distinguish between dominant time constants and small time constants. Typically, the dominant time constants are a part of the plant transfer function, while the control system introduces several small time constants (e.g. sensors, actuators, sampling, calculation delays, fast inner control loops).

If the largest small time-constant is at least 4 times smaller than the smallest dominant time constant, i.e.,

$$min(T_{\mathrm{dominant}}) \geq 4 \cdot max(T_{\mathrm{small}})$$

then it is possible to make an important simplification for modeling of the system, which is to define one equivalent small time constant which is the sum of all small time constants in the system.

If the controller sampling frequency ($1/T_{\mathrm{s}}$) is same as the switching frequency ($1/T_{\mathrm{sw}}$), the small time constants present in the H-bridge converter model for this specific implementation are:

- small time constant for control calculation,T_{calc},is $\frac{1}{2}\,T_{\mathrm{sw}}$
- small time constant for PWM output generation,T_{pwm},is $\frac{1}{2}\,T_{\mathrm{sw}}$

- small time constant for conversion of continuous parameters to discrete,T_{sh},is $\frac{1}{2}\,T_{\mathrm{sw}}$ (explained in the section below)

The equivalent small time constant,T_Σ,is

$$T_\Sigma = T_{\mathrm{calc}} + T_{\mathrm{pwm}} + T_{\mathrm{sh}}$$

$$D_\Sigma(s) = \frac{1}{1 + sT_\Sigma}$$

Calculation of control parameters

The control parameters of the PI controller (K_{p} and K_{i}) are calculated using the Magnitude Optimum Criterion. The system's open-loop transfer function $H_{\mathrm{OL}}(s)$ is given by the product of transfer functions from the controller, plant and time delays:

$$H_{\mathrm{OL}}(s) = \frac{1 + sT_{\mathrm{n}}}{sT_{\mathrm{i}}} \cdot \frac{K_1}{1 + s\,T_1} \cdot \frac{1}{1 + sT_\Sigma}, \quad \text{where } K_{\mathrm{p}} = \frac{T_{\mathrm{n}}}{T_{\mathrm{i}}} \text{ and } K_{\mathrm{i}} = \frac{1}{T_{\mathrm{i}}}$$

The controller parameter T_{n} is chosen, such that the pole of the plant transfer function is canceled, i.e. $T_{\mathrm{n}} = T_1$. After pole-zero cancellation, the closed-loop transfer function is a second order system. The remaining parameter T_{i} is calculated from setting the damping factor (ζ) of the second order system to $1/\sqrt{2} = \zeta$, resulting in $T_{\mathrm{i}} = 2K_1T_\Sigma$

The outputs from the "Right Leg Duty-Cycle" and the "Left Leg Duty-Cycle" subsystems are fed to the PWM Out block from the PLECS RT Box component library. The switching frequency is $10\,\mathrm{kHz}$.

3 Simulation

This model can run both, in offline mode on a computer or in real-time mode on the PLECS RT Box. For the real-time operation, one RT Box (referred to as "Plant + Controller") needs to be set up as demonstrated in Fig. 5.

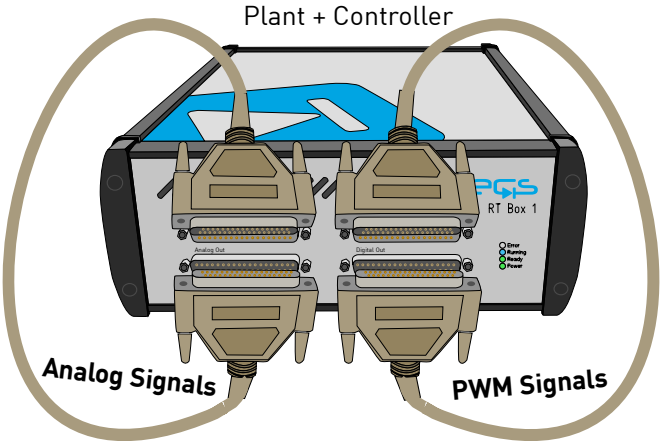


Fig. 5: Hardware configuration for the real-time operation on a single RT Box

Please follow the instructions below to run a real-time model on a single RT Box:

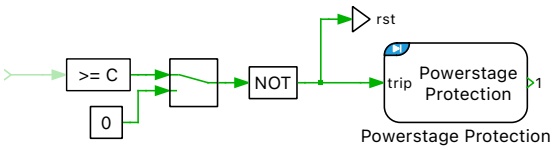
1 Connect the Analog Out interface to the Analog In interface with one DB37 cable, and the Digital Out interface to the Digital In interface with another DB37 cable (as shown in Fig. 5).

2 From the **System** tab of the **Coder options...** window, select the "Plant+Controller" subsystem and **Build** it onto the RT Box.

3 Once the model is uploaded, from the **External Mode** tab of the **Coder options...** window, **Connect** to the RT Box and **Activate autotriggering**.

Note

As shown below, the Manual Signal Switch in the “Controller” task in its default “up” position enables switching, once the model is up and running.
Changing it to the “down” position trips all PWMs into the safe state, and resets the integral part of the PI controller to its initial condition.
The safe state is configured in the Protection tab of the PWM Out block.



The inductor current measurements can be viewed using the Scope connected to the plant, as well as the Scope in the “Controller” task. The inductor current reference is toggled between $-3\,\mathrm{A}$ and $3\,\mathrm{A}$ using the “Iset” (Pulse Generator) component of the “Controller” task. These reference values can be changed on the fly, in real-time, since the “Iset” component has been added to the “Exceptions” list found in the **Parameter Inlining** tab of the **Coder options...** window, prior to building the model.

The step response of the inductor current in real-time is shown in Fig. 6.

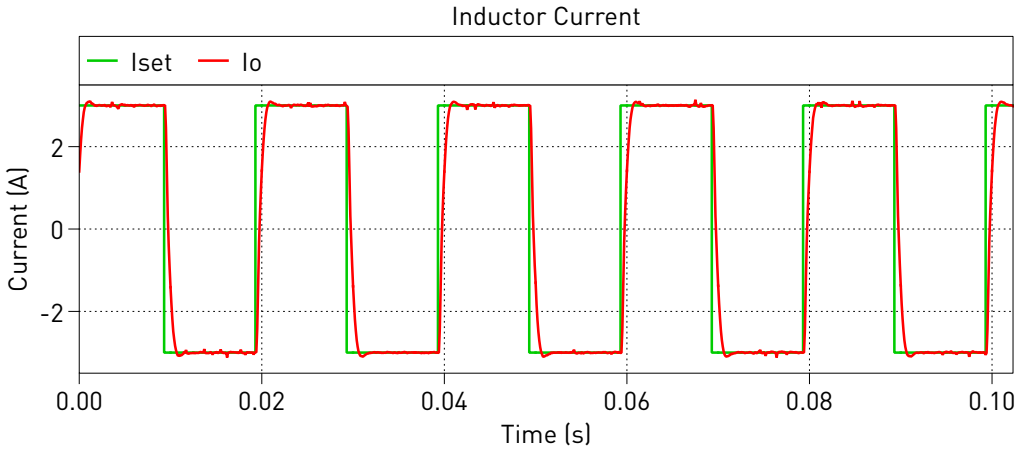


Fig. 6: Step response of the inductor current in real-time

A trigger control for a desired **Trigger channel** can be set from the **External Mode** tab.

4 Conclusion

This model demonstrates an H-bridge converter with a discrete current controller that can be run in both offline mode, as well as in real-time. The model also demonstrates the **Parameter Inlining** feature using a current controller reference that can be changed in real-time.

5 Bibliography

[1] J. Allmeling, and N. Felderer, “Sub cycle average models with integrated diodes for real-time simulation of power converters,” *IEEE Southern Power Electronics Conference (SPEC), 2017*.

[2] TI DRV8305N 3-Phase Motor Drive BoosterPack Evaluation Module newline URL: http://www.ti.com/tool/BOOSTXL-DRV8305EVM.[7]

[7] http://www.ti.com/tool/BOOSTXL-DRV8305EVM

[3] Conception de systèmes automatiques, Hansruedi Bühler, Presses Polytechniques Romandes, Lausanne 1988, ISBN 2-88074-149-1

Revision History:

RT Box TSP 1.8.3	First release
RT Box TSP 2.1.5	Turn on Assertions in IGBT Half Bridges and add deadtime in the PWM Out block
RT Box TSP 2.1.7	Modify the controller subsystem and add the enable/disable switching scheme
RT Box TSP 2.2.1	Use the Powerstage Protection block to enable/disable switching
RT Box TSP 2.3.4	Use one Full Bridge power module instead of two separate Half Bridge power modules
RT Box TSP 3.0.1	Adapt Analog Out/In channel numbers to make the model also runnable on RT Box CE, update the single box model to use the multi-tasking feature on RT Box 1
RT Box TSP 3.0.3	Remove the model with two separate RT Boxes and keep only the single RT Box model using multi-tasking feature
RT Box TSP 3.1.1	Change to Nanostep implementation, update documentation

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
	`https://www.plexim.com`	Web

RT Box Demo Model

© 2002–2026 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.