

RT Box

DEMO MODEL

Boost 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 boost converter with a resistive load and closed-loop current control. The nominal operating condition is given at

- 52 kW power,
- 480 V input voltage and
- 108 A inductor current reference.

This document describes the implementation of the power stage and controls in PLECS and the real-time deployment of the system on a single RT Box. For such a "virtual prototyping" configuration the RT Box has front connection with two 37 pin Sub-D cables to loop back PWM signals and analog measurement of the same Box.

The chosen discretization step sizes and average execution times for each core running the Boost converter model are shown in Tab. 1. The discretization step size parameter specifies the base sample time of the generated code and is used to discretize the state-space equations of the model. The execution time represents the actual time it takes the processor on the RT Box to calculate the corresponding model.

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	1.6 μs / 2 μs	0.9 μs / 50 μs
RT Box 1	1.7 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 `Tdisc.Plant`. Another core (Core 1) runs closed-loop controls in "Controller" task in parallel with a sample time of `Tdisc.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 `Tdisc.Plant`. The "Controller" task is executed as a background task with lower priority at the sample time of `Tdisc.Controller`.

[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

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

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 has analog measurement signals and PWM signals connected in loop-back, so that the offline simulation can also run. Additional delays in the feedback path are also 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.

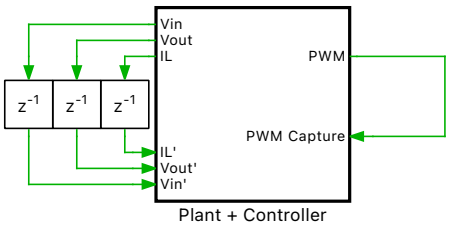


Fig. 1: Top-level schematic of the boost converter model

In the following subsections, the converter stage and the controller part are explained separately.

2.1 Boost Converter

A boost converter is also known as a “step-up” converter since it converts an input DC voltage to a higher DC voltage at its output. The boost converter is modeled using a Boost Converter 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 inductor represents an energy storage element with a parasitic series resistance R_L. At the output a capacitor is used as a filtering component to stabilize the load voltage. An overview of the boost converter is given in Fig. 2.

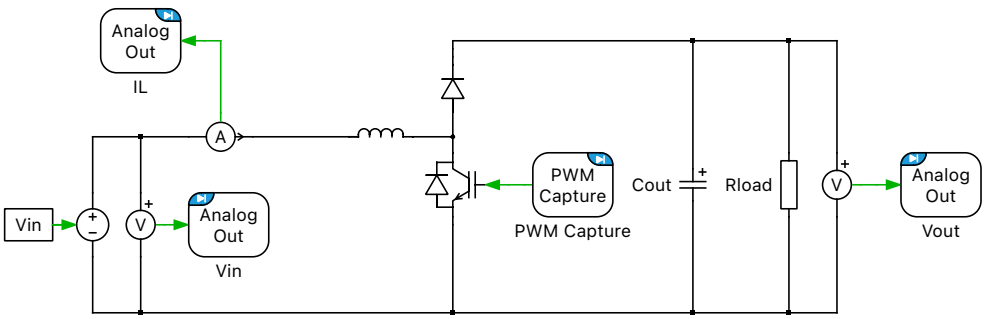


Fig. 2: Boost converter plant model

The plant model uses different blocks from the RT Box Target Support Library to access the physical input and output ports of the RT Box:

- 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.
- Analog Out blocks provide the analog signals required by the controls such as inductor current, input voltage, and output voltage. The Analog Out component contains scale and offset parameters that can be set to avoid saturating the analog outputs of the RT Box and to match IO requirements of the connected hardware or controller. Appropriate scaling factors can be determined with an offline simulation in PLECS. In the case of the inductor current, a closed-loop offline simulation reveals a maximum current value of $110\,\mathrm{A}$. Along with an analog output voltage range configuration of $\pm 5\,\mathrm{V}$, as specified by the user in the **Target** tab of the **Coder Options** window, the Analog Out scaling factor, $I_{L,\mathrm{scale}}$, has been set to:

Note that in this virtual prototyping setup it does not matter which IO channels are configured, but the channel IDs must match between the controller and the plant part due to the usage of loopback cables.

Design

For a boost converter in CCM and with given input and output voltages V_{in} and V_{out}, the required duty-cycle D can be calculated as follows:

$$D = 1 - \frac{V_{\mathrm{in}}}{V_{\mathrm{out}}}$$

The sizing of the input inductor L and output capacitor C are determined using the specifications of current and voltage ripple ($\Delta I_{\mathrm{in,pp}}$ and $\Delta V_{\mathrm{out,pp}}$) in Tab. 2.

Table 2: Parameters set and design requirements

$V_{\mathbf{in}}$	$V_{\mathbf{out,nom}}$	$\Delta V_{\mathbf{out,pp}}$	$\Delta I_{\mathbf{in,pp}}$	$P_{\mathbf{nom}}$	$R_{\mathbf{L}}$	$f_{\mathbf{sw}}$
$480 - 800\,\mathrm{V}$	$950\,\mathrm{V}$	$10\,\mathrm{V}$	$5\,\mathrm{A}$	$52\,\mathrm{kW}$	$15\,\mathrm{m\Omega}$	$20\,\mathrm{kHz}$

The current increases during the first part of the switching period $t_0 \le t \le t_0 + DT$, when V_{in} is applied over the inductance L. Using the state equation of the inductor and neglecting the inductor series resistance, one finds that:

$$\frac{dI_L}{dt} = \frac{\Delta I_{\mathrm{in,pp}}}{DT_{\mathrm{sw}}} = \frac{V_{\mathrm{in}}}{L} \Rightarrow L = \frac{V_{\mathrm{in}} \cdot D}{\Delta I_{\mathrm{in,pp}} \cdot f_{\mathrm{sw}}}$$

The required inductance L is then found at the nominal output voltage and minimum input voltage:

$$L = \frac{V_{\mathrm{in}} \cdot \left(1 - \frac{V_{\mathrm{in}}}{V_{\mathrm{out}}}\right)}{\Delta I_{\mathrm{in,pp}} \cdot f_{\mathrm{sw}}} = 2.375\,\mathrm{mH}$$

At nominal power, the load current and resistance values are:

$$I_{\mathrm{load}} = \frac{P_{\mathrm{nom}}}{V_{\mathrm{out}}} = 54.7\,\mathrm{A}$$
$$R_{\mathrm{load}} = \frac{V_{\mathrm{out}}}{I_{\mathrm{load}}} = 17.4\,\Omega$$

The value of the output capacitor C is calculated in a similar fashion to the inductor L:

$$C = \frac{I_{\mathrm{load}} \cdot D}{\Delta V_{\mathrm{out,pp}} \cdot f_{\mathrm{sw}}} = 135.4\,\mu\mathrm{F}$$

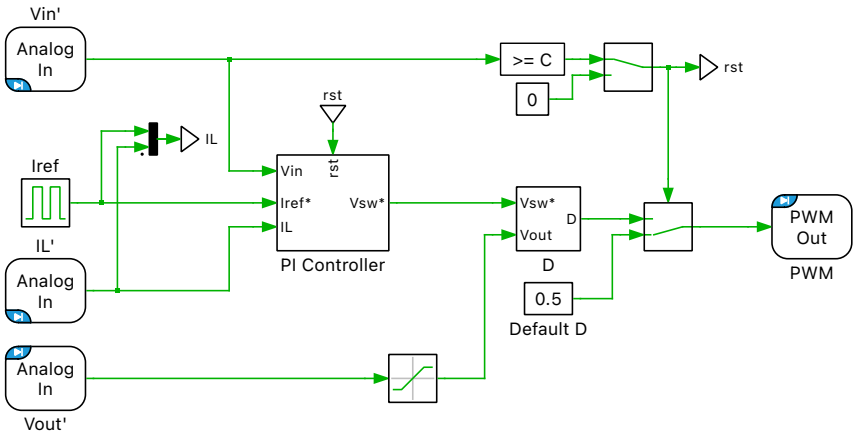


Fig. 3: Schematic of the current controller

2.2 Current Controller

The inductor current of the boost converter is regulated by a PI controller implemented in the controller part, as shown in Fig. 3. The current reference set point will toggle between the minimum and maximum reference values at $50\,\mathrm{Hz}$. The parameters of the PI controller are calculated using the plant transfer function $P(s)$ and the Magnitude Optimum Criterion, as described in more detail below.

The controller part contains the following components from the RT Box Target Support Library:

- Analog In blocks provide the analog signals from the plant model. The input signals can be scaled and offset, similar to the Analog Out block. In this model, the Analog In scaling factors are set to the inverse of the Analog Out scaling factors. The current and voltage values in the controller part will then correspond to unscaled voltage and current measurements in the power stage.
- The PWM Out block is configured to generate a PWM signal with a symmetrical carrier on digital output channel 0. To sample the average inductor current, the sample point of the ADC needs to be in the middle of the switching period for a symmetrical PWM scheme. To ensure consistent point-on-wave sampling, the ADC and the PWM execution need to be synchronized. With the RT Box this is achieved by synchronizing both ADC sampling and PWM update with the model step. More specifically in this multi-tasking demo, it means to synchronize with the task step that the PWM Out block resides in, i.e. the "Controller" task sample time. This behavior is configured by enabling the **Synchronization with model step** parameter of the PWM Out block. The controller runs in a single update mode with the compare value (CMP) updating always on the minimum of the triangular carrier (see **Update** setting of the PWM Out block). In the initialization commands the PWM carrier frequency is set to $20\,\mathrm{kHz}$.

Plant transfer function

To set the PI controller gain parameters the plant transfer function $P(s)$ is needed. $P(s)$ relates the change of the voltage across the RL elements V_{RL} (the input variable) to the response of the inductor currents I_L (the output variable):

$$P(s) = \frac{I_L}{V_{RL}} = \frac{1/R_\mathrm{L}}{1 + s\,L/R_\mathrm{L}} = \frac{K_1}{1 + s\,T_1}$$

where $K_1 = 1/R_L$ and $T_1 = L/R_L$

Time delays

Several types of time delays are present in the virtual prototyping setup of this boost converter real-time model.

- Analog input sampling

- Control calculation time step
- PWM output generation
- PWM capture interface
- Plant calculation time step
- Analog output

Since the plant calculation time step is much smaller than the controller step (factor of 50), only the time delays of the controller are considered in the offline model. These are the control calculation time step itself (T_{sw}) and the average PWM output delay of $T_{\mathrm{sw}}/2$ giving a total delay of $3/2 \cdot T_{\mathrm{sw}}$. For the calculation of the control parameters the delay is approximated with a first-order low pass filter of the form:

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

where $T_{\Sigma} = 3/2 \cdot T_{\mathrm{disc,control}} = 3/2 \cdot T_{\mathrm{sw}}$.

Calculation of control parameters

The control parameters of the PI controller (K_i and K_p) 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:

$$H_{\mathrm{OL}}(s) = \frac{1 + sT_{\mathrm{n}}}{sT_{\mathrm{i}}} \cdot \frac{K_1}{1 + s\,T_1} \cdot \frac{1}{1 + sT_{\Sigma}}$$

where $K_p = \frac{T_n}{T_i}$ and $K_i = \frac{1}{T_i}$. The controller parameter T_{n} is chosen, such that the pole of the plant transfer function is canceled, i.e. $T_{\mathrm{n}} = T_1$. The remaining parameter T_{i} is calculated from the closed-loop transfer function and with the condition $H_{\mathrm{CL}}(j\omega) \approx 1$ at low frequencies $T_{\mathrm{i}} = 2K_1T_{\Sigma}$.

Anti-windup and controller reset

The controller is equipped with an anti-windup mechanism. The anti-windup feedback uses a corrective gain of:

$$K_{\mathrm{bc}} = \frac{K_{\mathrm{i}}}{K_{\mathrm{p}}}\,.$$

The lower saturation level is $0\,\mathrm{V}$ (for a duty cycle of 1) and the upper saturation level is set to V_{out} (for a duty cycle of 0). Note that this assumes that the output capacitor is pre-charged to at least V_{in}. In startup situations where the output capacitor is not pre-charged the saturation level of the controller would need to be variable and adapted to the actual output voltage.

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, an RT Box (referred to as "Plant + Controller") needs to be set up as demonstrated in Fig. 4.

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. 4).

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

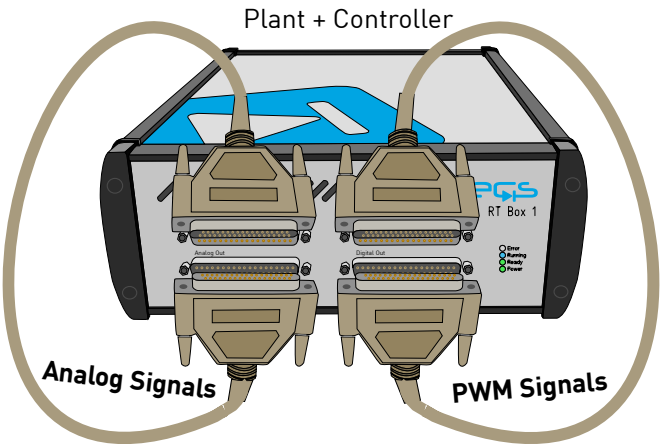
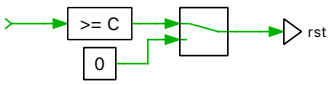


Fig. 4: Hardware configuration for the real-time operation on a single RT Box

3.1 External Mode

Once the building process is finished, open the **External Mode** tab of the "Plant + Controller" subsystem and click **Connect** and then **Activate autotriggering**. The PLECS Scopes inside the subsystem will now update with real-time data from the RT Box. The "Controller" task has different tunable parameters:

- Manual Signal Switch
- "K_p" Gain block inside the "PI Controller" subsystem
- "K_i" Gain block inside the "PI Controller" subsystem
- "I_ref" Pulse Generator block



Note

A reset option is added to the controller. As shown above, the Manual Signal Switch in the "Controller" task in its default "up" position enables switching.
Changing it to the "down" position resets the integral part of the PI controller to its initial condition and a default duty cycle of 0.5 is applied to the PWM module.

Notice the inductor current reference is toggling between a maximum ($108\,\mathrm{A}$) and minimum ($65\,\mathrm{A}$) value in the "Controller" task. The step response of the current control-loop is shown in Fig. 5. The maximum and minium values for the reference current can be tuned online as well as the control parameters K_i and K_p.

4 Conclusion

This RT Box demo model demonstrates a boost converter under closed-loop control with a continuous PI current controller and resistive load. The demo model can run in both offline simulation and in real-time on a single RT Box. The plant part runs with a discretization step size of $2\,\mu\mathrm{s}$. The controller task runs with a discretization step size of $50\,\mu\mathrm{s}$, which is the length of the switching period.

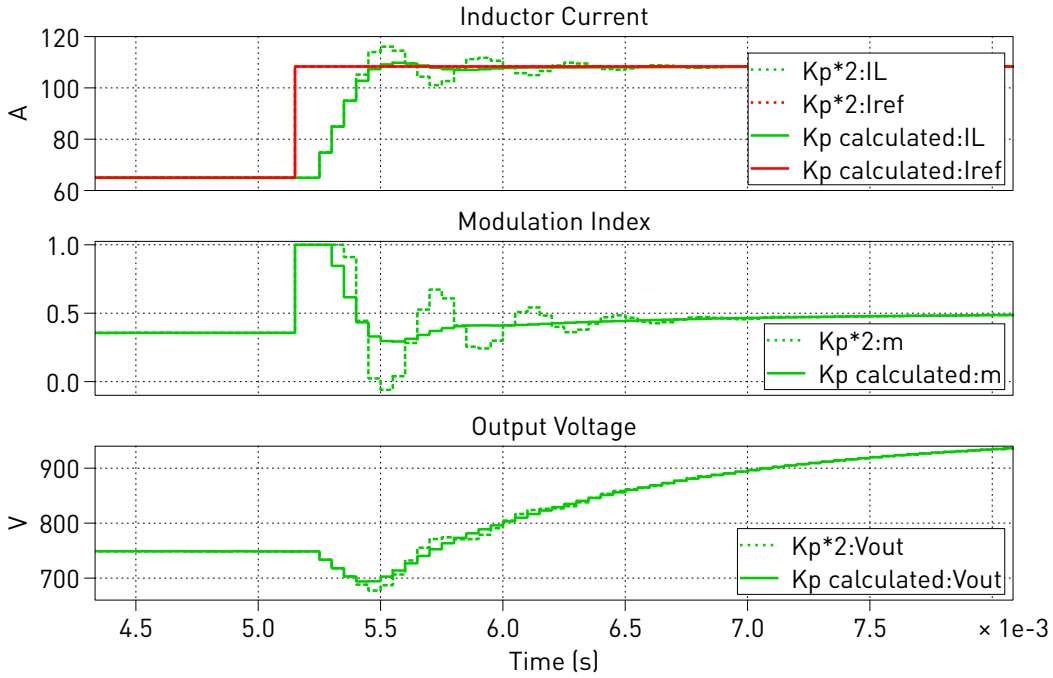


Fig. 5: Current control step response from 64 A to 108 A

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.*

Revision History:

RT Box TSP 1.8.3	First release
RT Box TSP 2.1.7	Use the PI Controller component from the library
RT Box TSP 3.0.1	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.