



**Embedded
Code Generation**

DEMO MODEL

Boost Converter with Peak Current Control

Peak current control of a boost converter with embedded code generation for TI C2000 MCUs

Last updated in C2000 TSP 1.9.1

www.plexim.com

- ▶ Request a PLECS and PLECS Coder trial license
- ▶ Get the latest TI C2000 and RT Box Target Support Package
- ▶ Check the PLECS, RT Box and TI C2000 TSP documentation

1 Overview

This demo model features a boost converter circuit with peak current mode control operating in continuous conduction mode (CCM). The peak current control is implemented using the Peak Current Control (PCC) component of the TI C2000 Target Support Library. The component integrates several MCU peripherals including a PWM generator, comparator, and a digital-to-analog converter to achieve the desired PCC functionality.

The model is split into two distinct subsystems called “Plant” and “Controller”. The plant contains a boost converter circuit, and the controller uses the TI C2000 PCC component to implement peak current mode control with output voltage regulation. The control logic in the controller subsystem may be built and then flashed to a TI C2000 MCU.

The following sections provide a brief description of the model and instructions on how to simulate the model.

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 contains two separate subsystems representing the controller and plant models, as shown in Fig. 1. The controller subsystem is enabled for code generation from the **Edit + Subsystem + Execution settings...** menu. This configuration is necessary to generate the model code for a subsystem via the PLECS Coder.

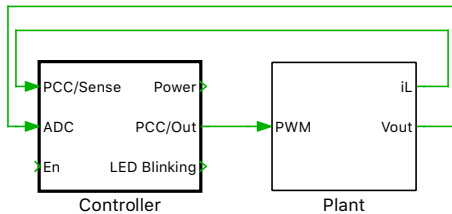


Figure 1: Top level schematic of the plant and the controller subsystems

2.1 Power Circuit

The power circuit is a boost converter powering a resistive load. As shown in Fig. 2, it is supplied by a DC source voltage of $V_{in} = 20\text{ V}$. The DC output voltage and inductor current measurements are connected to Signal Output port components. The inductor current is sensed via a sense resistor R_{Shunt} , connected to the MOSFET source pin, and the output voltage is measured via a voltage divider circuit. These measurements are then fed back to the controller.

2.2 Controls

The controller subsystem is shown in Fig. 3. The “Peak Current Controller” component is responsible for current regulation. An outer voltage control loop supplies the peak current reference to the PCC block. This is implemented using a Type II controller.



In a peak current-mode controller, at the beginning of each switching cycle the output is set (gate signal is turned ON) without a pre-determined duty cycle (although a minimal and a maximal duty cycle is enforced). When the sensed inductor current exceeds the peak current reference value, the output is reset (gate signal is turned OFF). The duty cycle is therefore determined by the rise of the inductor current during the on-time.

Open-loop with peak current determined analytically

For a boost converter, the value of the peak current (I_{pk}) in Continuous Conduction Mode (CCM) is

www.plexim.com

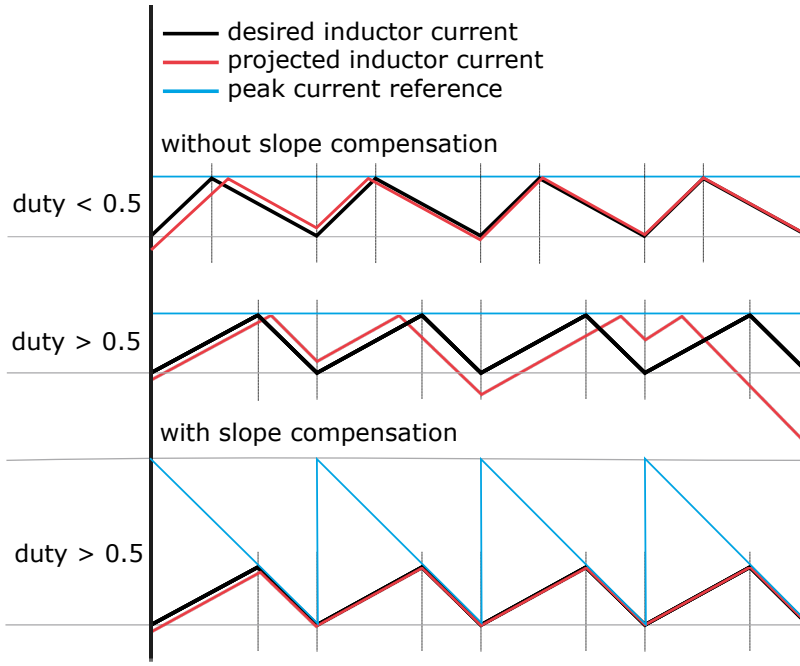


Figure 4: Slope compensation

where, I_{in} is the input current, V_o is the output voltage, D is the duty cycle, T_s is the switching period and L is the inductance of the converter.

The ramp slope is set to

$$\text{ramp} = \frac{V_o - V_{in}}{L}$$

where V_{in} is the input voltage.

To compensate for the slope, the peak current reference needs to be adjusted. An offset of $\text{ramp} \cdot D \cdot T_s$ must be added to I_{pk} , where T_s is the switching period.

On TI C2000 MCUs, the ramp is generated using a 16-bit DAC register that is decremented by a configurable amount (DEC) at each SYSCLK cycle. For a desired ramp, the value for DEC is calculated as

$$\text{DEC} = \text{ceil}\left(\text{ramp} \cdot \frac{R_{\text{shunt}}}{3.3/2^{16}} \cdot \frac{1}{\text{SysClk}}\right)$$

where, SysClk is the system clock frequency.

Therefore, the value of I_{pk_offset} is adjusted accordingly as,

$$I_{pk_offset} = \text{DEC} \cdot \frac{3.3/2^{16}}{R_{\text{shunt}}} \cdot \text{SysClk} \cdot D \cdot T_s$$

Voltage compensator

The voltage compensator is tuned analytically using the K-factor method, based on the plant transfer function from peak current setpoint to the output voltage. The K-factor method is a loop shaping technique, where a controller can be designed accurately for a specified phase margin and crossover frequency. Controller design using the K-factor method is explained in [1].

The plant transfer function for the outer voltage loop is $G_p(s)$. The voltage loop is designed to be slower than the current loop.

$$G_p(s) = \frac{V_o(s)}{i_{pk}(s)} = R_L * (1 - D) * \left(1 - \frac{sL}{R_L(1-D)^2}\right) * \left(\frac{1 + sr_C C_o}{2 + sR_L C_o}\right)$$

where, R_L is the load resistance, C_o is the output capacitance and r_C is the ESR of the output capacitance.

For a desired crossover frequency ω_c , the required phase boost (ϕ_{boost}) is calculated as:

$$\phi_{\text{boost}} = \text{PM} - \phi_{\text{sys}} - 90^\circ$$

where, ϕ_{sys} is the phase of the system and PM is the desired phase margin.

If the required phase boost is smaller than 90° , a Type II controller is used.

The transfer function of a Type II controller is:

$$G_c(s) = \frac{K_c}{s} \frac{1 + \frac{s}{\omega_z}}{1 + \frac{s}{\omega_p}}$$

where K_c is the gain of the controller, ω_z is the location of the zero and ω_p is the location of the pole. The values of k , ω_z and ω_p are calculated from the following expressions:

$$k = \tan\left(\frac{\phi_{\text{boost}}}{2} + 45^\circ\right); \omega_z = \frac{\omega_c}{k}; \omega_p = k \cdot \omega_c$$

$$\frac{1}{K_c} = \text{abs}\left(\frac{1}{j\omega_c} \cdot \frac{1 + \frac{1+j\omega_c}{\omega_z}}{1 + \frac{j\omega_c}{\omega_p}} \cdot G_p(j\omega_c)\right)$$

Configuring TI C2000 Target library components

The controller in Fig. 3 contains several components from the TI C2000 Target library.

- **Peak Current Controller:** The main TI C2000 target component used in the model is the peak current controller (PCC) block. This component implements peak current control with slope compensation.

Internally, the PCC block makes use of multiple MCU peripherals, as shown in Fig. 5. The first component is a DAC that provides a peak current set-point including ramp, for controlling the inductor current. The second is a comparator (COMP); the current sensed via the MOSFET source pin in Fig. 2 is fed to the comparator, which is then compared to the peak current reference provided by the DAC. The output of the COMP block is fed to the third component, which is the PWM generator. The PWM generator generates the PWM waveforms at a frequency of 100 kHz.

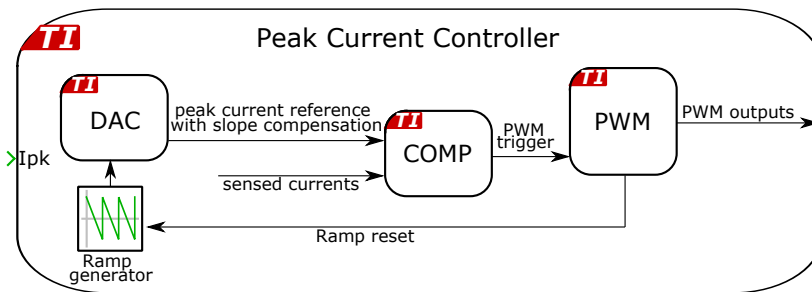


Figure 5: Peak current controller schematic

To prevent the turn-on transient currents from triggering the peak current controller, leading edge blanking time is applied. When leading edge blanking time is applied, the first turn-on transient peak is ignored, and the duty cycle will continue to increase until the sensed inductor current exceeds the desired peak current reference value.

- **ADC:** The measurements of the output voltage are captured by the ADC block of the TI C2000 Target component library. Scaling and offset factors are provided to each channel via the parameter window of the ADC block in order to convert the detected analog voltage into values with physical units to be used for the control algorithm. The **ADC unit** parameters can be modified according to the available resources of different MCUs. The control task is executed once the last ADC channel is converted. This is configured by connecting the Task output of the ADC to the Control Task Trigger block from the TI C2000 Target component library.
- **Powerstage Protection:** The Powerstage Protection block implements an interlock, which is a safety mechanism, to enable or disable all the PWM outputs on the target device. The PWM outputs are disabled unless there is a logical low to high transition on the input signal, labeled en. This prevents the PWM signals from becoming active as soon as the code is executed on the target, thereby ensuring safe operation. For more details, please browse the **Help** section of this block.

3 Simulation

Run the simulation as provided to observe the results from the closed-loop simulation with peak current mode control with output voltage regulation. The PWM signals, inductor current and output voltage measurements can be viewed using the scope found in the “Plant” subsystem. The output voltage measurements can be viewed from the scope in the “Controller” subsystem as well. The output voltage reference is set to 50 V from the “Controller” subsystem.

The PWM signals, inductor current and output voltage measurements for an offline closed-loop simulation are shown in Fig. 6.

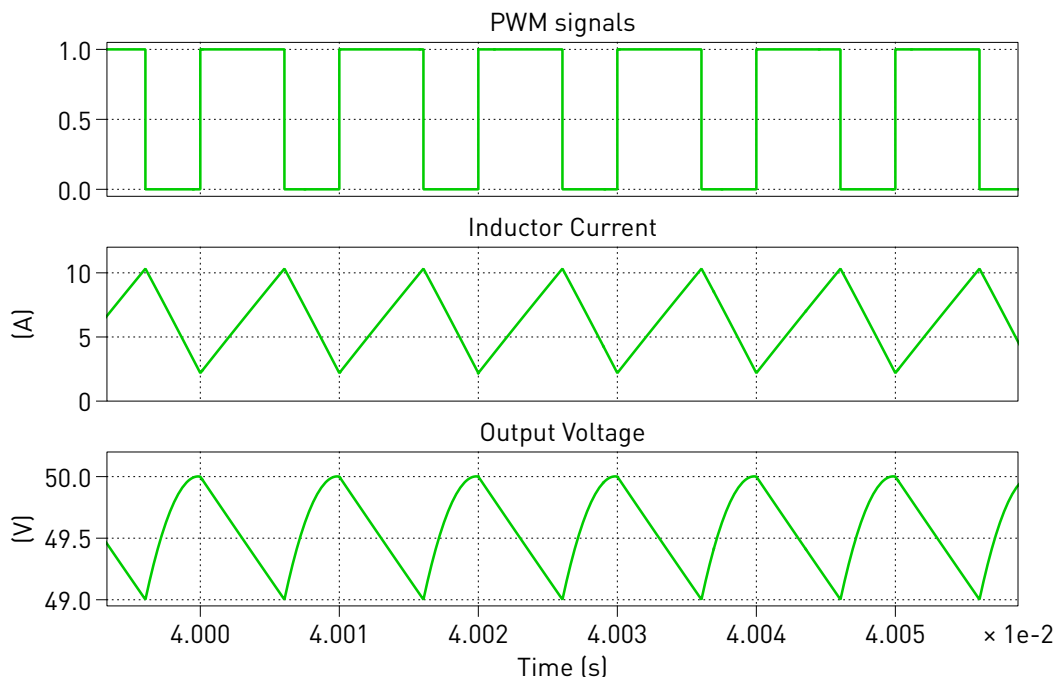


Figure 6: PWM, inductor current and output voltage waveforms for an offline closed-loop simulation

From the “Controller” subsystem, flip the manual switch “Sw” to the sum block to operate the model in open-loop, with the peak current reference determined analytically. The peak current value and the offset are calculated in the initialization commands window.

In addition to running a simulation of this demo model in offline mode on a computer, the “Controller” subsystem can be directly converted into target specific code for the TI C2000 MCUs. The default I/O configuration of all the peripheral blocks (ADC, PWM etc.) supports the TI 280039C [3], TI 280049C [4],

TI 28379D [5], TI 28P550SJ9 [6], TI 28P650DK9 [7] LaunchPads, and the TI 28388D [10] controlCARD. Additionally, the demo model allows for code generation for the TI 280039C [8] and TI 28379D [9] controlCARDS. Refer to “Quick Start” section of the TI C2000 Target Support User Manual [11] for detailed step-by-step instructions.

4 Conclusion

This model demonstrates a boost converter with peak current mode control that supports embedded code generation for TI C2000 MCUs.

References

- [1] Videos on K-factor controller design by Dr. Raja Ayyanar. Click to access online:
k-factor control design method: part 1
k-factor control design method for dc dc part2
- [2] NPTEL lectures from Indian Institute of Science, Bangalore. Click to access online:
Slope compensation for current control
- [3] TI C2000 F280039C LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F280039C>.
- [4] TI C2000 Piccolo MCU F280049C LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F280049C>.
- [5] TI C2000 Delfino MCU F28379D LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F28379D>.
- [6] TI C2000 F28P550SJ9 LaunchPad development kit,
URL: <https://www.ti.com/tool/LAUNCHXL-F28P55X>.
- [7] TI C2000 F28P650DK9 LaunchPad development kit,
URL: <https://www.ti.com/tool/LAUNCHXL-F28P65X>.
- [8] TI C2000 F280039C controlCARD evaluation module,
URL: <https://www.ti.com/tool/TMDSCNCD280039C>.
- [9] TI C2000 F28379D controlCARD development kit,
URL: <https://www.ti.com/tool/TMDSCNCD28379D>.
- [10] TI C2000 F28388D controlCARD evaluation module,
URL: <https://www.ti.com/tool/TMDSCNCD28388D>.
- [11] PLECS TI C2000 Target Support User Manual,
URL: <https://www.plexim.com/download/documentation>.

Revision History:

C2000 TSP 1.4.1	First release
C2000 TSP 1.5.1	Added support for 28388D and 28379D controlCARD targets
C2000 TSP 1.6.1	Added support for 280039C LaunchPad and controlCARD targets, and auto-pin selection
C2000 TSP 1.9.1	Added support for 28P550SJ9 LaunchPad target

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

Embedded Code Generation Demo Model

© 2002–2024 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.