



PLECS

*DEMO MODEL*

## **Buck Converter Operating in Boundary Conduction Mode**

**Buck converter with on-line parameter tuning for predictive peak current mode control operating in BCM**

Last updated in PLECS 4.6.1

[www.plexim.com](http://www.plexim.com)

- ▶ Request a PLECS trial license
- ▶ Check the PLECS documentation

# 1 Overview

This demonstration shows a buck converter with an on-line parameter tuning technique for predictive peak current mode control operating in Boundary Conduction Mode (BCM) as explained in [1].

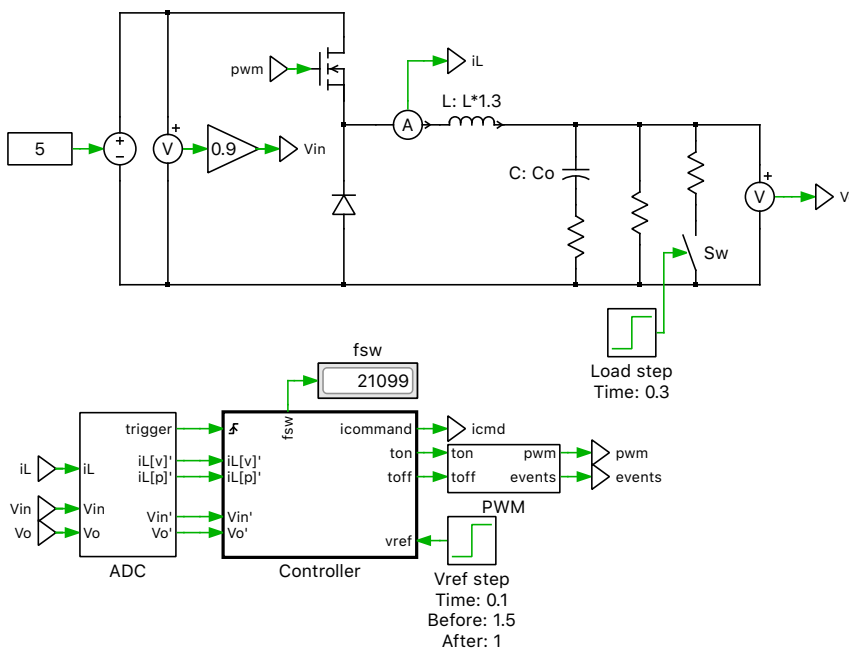
**Note** This model contains model initialization commands that are accessible from:

*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 the power circuit and the controller, as shown in Fig. 1. The power circuit is a simple buck converter with a MOSFET and a resistive load. The controller regulates the output voltage and regulates the converter to operate in BCM. Sensor error in the voltage measurement and parameter deviation in the estimated inductance are modeled as well.



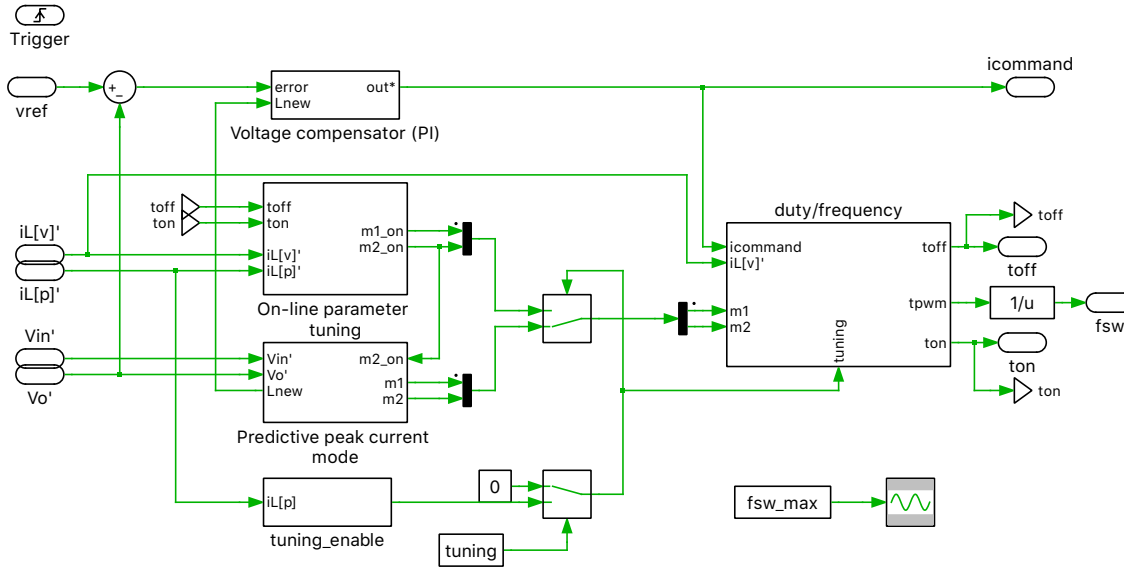
**Figure 1: Top-level schematic of the model**

### 2.1 Controller

The control logic consists of three main Subsystems, as shown in Fig. 1. The “PWM” subsystem generates the gate signal based on variable on and off times from the main controller. An event signal is generated by the “PWM” subsystem when the PWM carrier reaches a minimum or maximum. The event signals are used in the “ADC” subsystem to sample the inductor current at the minimum and maximum of the inductor current ripple.

The “Controller” subsystem is a Triggered Subsystem and implements the main control logic. Components within a Triggered Subsystem are executed as a group based on the input trigger signal. The trigger input is aligned with the peak of the current ripple.

The controller subsystem is shown in Fig. 2. The voltage compensator regulates the DC output voltage by comparing the sensed DC output signal against a reference. The output of the voltage compensator is the peak inductor current reference,  $i_{\text{command}}$ . This reference signal is then used in the calculation of on-time and off-time, i.e.,  $t_{\text{on}}$  and  $t_{\text{off}}$  respectively, in the predictive peak current mode control, with on-line parameter tuning technique.



**Figure 2: Controller of the Buck converter in BCM**

## Voltage Compensator

**Plant transfer function** To set the PI controller gain parameters a plant transfer function is needed. The transfer function for the outer voltage loop is  $V(s)$ . The voltage loop is designed to be slower than the current loop so that it does not distort the current reference.

$V(s)$  relates the change of the current through the inductor ( $L$ ),  $I_L$  (the input variable), to the response of the capacitor voltage  $V_C$  (the output variable):

$$V(s) = \frac{V_C}{I_L} \approx \frac{1}{sC} = \frac{1}{sT_2}, \quad \text{where } T_2 := C$$

**Equivalent delay** The control system often introduces several small delays (e.g. from sensors, actuators, sampling, calculation delays, PWM delay). It is generally assumed that these delays are smaller than the time constant of the plant. If  $T_\Sigma$  is the equivalent delay of the control system, then the simplified transfer function of the delay is:

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

The delays present in this model for this specific implementation are:

- a small time constant for control calculation,  $T_{\text{calc}}$ , is  $\frac{1}{2} T_s$
- a small time constant for PWM output generation,  $t_{\text{pwm}}$ , is  $\frac{1}{2} T_s$  where,  $T_s$  is the sample time of the controller.

The equivalent delay,  $T_\Sigma$ , is therefore:

$$T_\Sigma = T_{\text{calc}} + t_{\text{pwm}} = T_s$$

**Calculation of control parameters for the voltage loop** The control parameters of the voltage PI controller ( $K_p$  and  $K_i$ ), since the outer voltage loop of the plant represents a pure integrator, are calculated using the Symmetrical Optimum Criterion. The system's loop gain  $V_{LG}(s)$  is given by the product of the transfer functions of the controller, equivalent time delay and plant:

$$V_{LG}(s) = \frac{1 + sT_n}{sT_i} \cdot \frac{1}{1 + sT_\Sigma} \cdot \frac{1}{sT_2}, \quad \text{where } K_p = \frac{T_n}{T_i} \text{ and } K_i = \frac{1}{T_i}$$

After solving the corresponding closed-loop transfer function, the final coefficients are:

$$T_n = 4 \cdot T_\Sigma \text{ and } T_i = 8 \cdot \frac{T_\Sigma^2}{T_2}$$

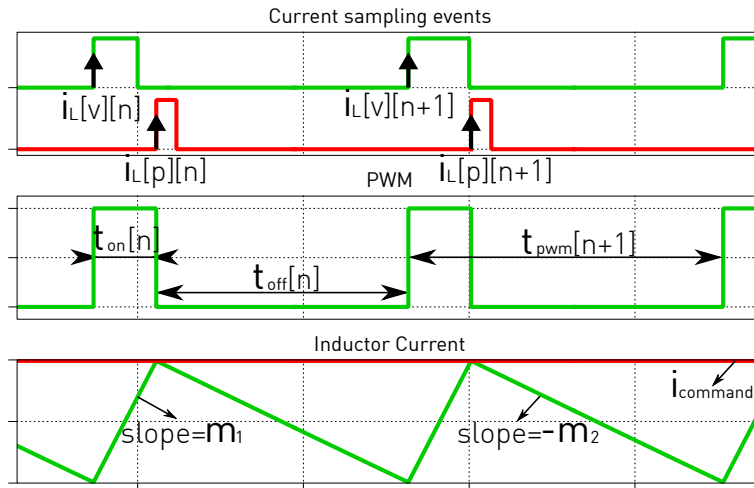
For a more detailed explanation on calculating the controller parameters, refer to [2].

These continuous parameters are converted to discrete parameters in the model. Since the sample time of the controller subsystem varies with the switching period, the gains for the discrete controller are calculated on the fly. The switching period for a boundary mode buck converter is calculated as:

$$t_{pwm} = \frac{2L}{(1-d)R_L}$$

where,  $L$  is the output inductance,  $d$  is the duty ratio and  $R_L$  is the load resistance.

Varying all the parameters on the fly to calculate the sample time required to derive the gains of the controller results in controller instability. Therefore,  $d$  and  $R_L$  are kept constant, and sample time is calculated by using the predicted value of the inductance,  $L_{new}$ , derived in a later section.



**Figure 3: Predictive current control with BCM**

### Predictive peak current mode control

The peak current controller regulates the peak ripple current to the current command,  $i_{command}$ . At the beginning of the switching period the on-time of the PWM signal,  $t_{on}$ , is calculated so as to reach the peak current set point. The current decreases to 0 by controlling the off time of the PWM signal,  $t_{off}$ . The on and off durations are calculated by predicting the inductor current rising slope,  $m_1$ , and falling slope,  $m_2$ , as shown in Fig. 3. These for a Buck converter are calculated as follows:

$$m_1 = \frac{V_{in} - V_o}{L}$$

$$m_2 = \frac{V_o}{L}$$

The on-time and off-time of the  $n$ th switching period are calculated by equations:

$$t_{\text{on}}[n] = \frac{i_{\text{command}} - i_L[v][n]}{m_1}$$

$$t_{\text{off}}[n] = \frac{i_{\text{command}}}{m_2}$$

where,  $i_L[v][n]$  is the inductor current sampled at the beginning of the  $n$ th switching period.

The switching period ( $t_{\text{pwm}}$ ) of the  $n$ th switching period is calculated as follows:

$$t_{\text{pwm}}[n] = t_{\text{on}}[n] + t_{\text{off}}[n]$$

### On-line parameter tuning

To address the sensitivity associated with inductor variation and input voltage drift, on-line parameter tuning is included. This method is based on sampled currents between two PWM outputs. The rising and falling current slopes,  $m_{1\_on}$  and  $m_{2\_on}$ , for  $n$ th switching period are calculated and updated on-line using the following equations:

$$m_{1\_on}[n] = \frac{i_L[p][n-1] - i_L[v][n-1]}{t_{\text{on}}[n-1]}$$

$$m_{2\_on}[n] = \frac{i_L[p][n-1] - i_L[v][n]}{t_{\text{off}}[n-1]}$$

where,  $i_L[p]$  is the inductor current sampled at the peak of inductor current, and  $i_L[v]$  is the inductor current sampled at the beginning of the switching period or the valley of the inductor current. In the PLECS model, all the parameters are delayed by one additional time step.

When on-line tuning is turned on, a small value of  $1e-7$  is subtracted from the calculated off time to create a small disturbance, which pushes the inductor current to almost reach Continuous Conduction Mode (CCM). This ensures that the controller can accurately calculate the rising and falling slopes for the next step, and the inductor current always stays in BCM in steady state.

The on-time and off-time are fed to a State Machine block to generate PWM signals.

Since the on-line tuning block relies on previously calculated values, logic is added to enable the block based on a minimum peak current and switching period, to ensure the on-line tuning block is only enabled after the first few transients.

To ensure that the values of  $m_1$  and  $m_2$  are always up-to-date in the predictive peak current mode block, the inductance value is updated on-line using the updated  $m_{2\_on}$  calculated from the on-line tuning block.

$$L_{\text{new}} = \frac{V_o}{m_{2\_on}}$$

A signal switch monitors the calculated value of the inductance. If it's not within the limits, it switches to using the previously calculated value of  $L$ .

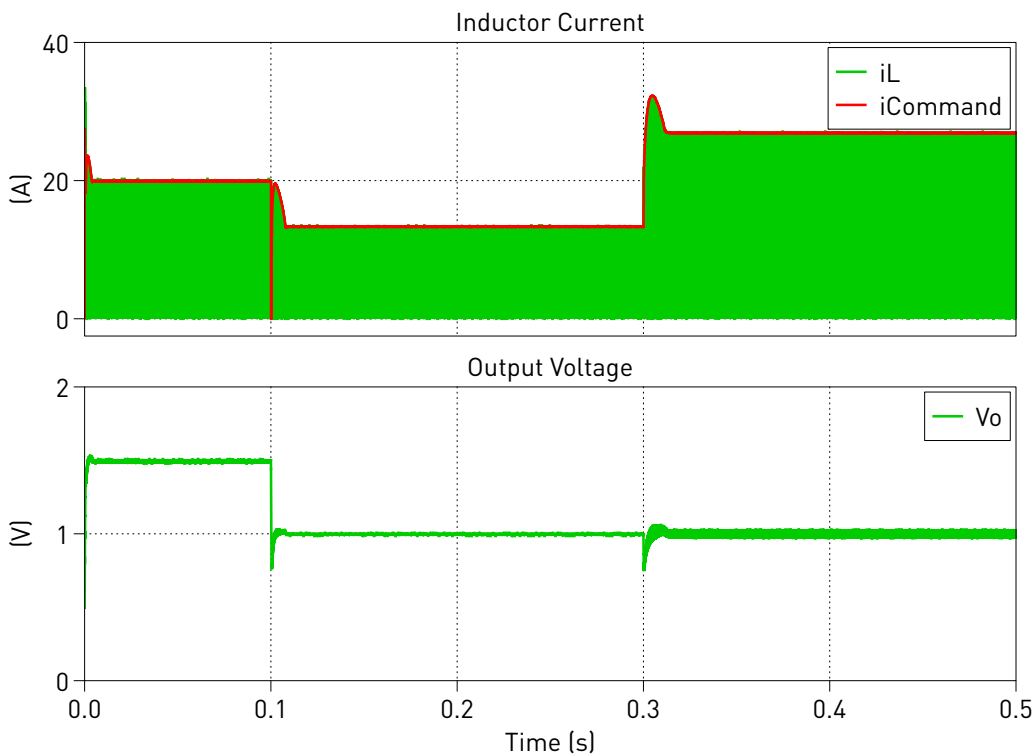
### 3 Simulation

Run the simulation with the model as provided to observe the inductor current command, inductor current, PWM signals, and the triggers where inductor current is sampled.

To show the sensitivity associated with inductor variation and the input voltage drift, the inductor value in the circuit is set to 1.3 of the nominal value by default and the input voltage measurement is changed to 0.9 times the actual value. Additionally, at 0.1 seconds, there is a step change in the voltage reference, and at 0.3 seconds, the load resistance is halved.

To experiment with the model, and to verify the on-line parameter tuning block, run the model as is and save the trace using the “Hold current trace” button. Then, from the “Controller” subsystem set the **online tuning** parameter as off and re-run the model. Compare the two traces. You will observe that when online tuning is disabled, the converter does not achieve BCM. One can also change inductor nominal value to 0.7 rated

The simulation results, with online parameter tuning enabled, are shown in Fig. 4



**Figure 4: Simulation results**

### References

- [1] Y. Chang and Y. Lai, “Online Parameter Tuning Technique for Predictive Current-Mode Control Operating in Boundary Conduction Mode,” in *IEEE Transactions on Industrial Electronics*, vol. 56, no. 8, pp. 3214-3221, Aug. 2009, doi: 10.1109/TIE.2009.2024651. Available: <https://ieeexplore.ieee.org/document/5071286>
- [2] Conception de systèmes automatiques, Hansruedi Bühler, Presses Polytechniques Romandes, Lausanne 1988, ISBN 2-88074-149-1

## Revision History:

PLECS 4.6.1      First release

## 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
	<a href="http://www.plexim.com">http://www.plexim.com</a>	Web

## *PLECS Demo Model*

© 2002–2023 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.