EE 458/559 – Power Electronics Controls

Experiment 2 Lab Procedure: Closed Loop Control of Boost Converter

Introduction

The goal of this lab is to design and test a controller for the synchronous boost power stage you have designed for your E-Bike. A common control strategy for boost converters is cascaded or multi-loop control comprising an inner current loop control and an outer voltage loop control. The high-bandwidth inner control loop regulates the inductor current, the reference for which is obtained from a lower-bandwidth outer loop controller regulating the output voltage of the power stage. This is shown conceptually in Figure 1. The control design portion of this lab is derived from Sections 1.5, 2, and 4.2 of the the supplemental textbook [1], which is a useful reference during this lab exercise. Automated design techniques are highly recommended for this lab. Note: The R,L,C values for this lab differ than from EE452, since our switching frequency is lower. Use $L = 1.3mH, C = 250, R = 10\Omega, R_L = 60m\Omega$

In this lab you will:

1. Design and test a current controller for your power stage.
2. Design and test a voltage controller for your power stage.

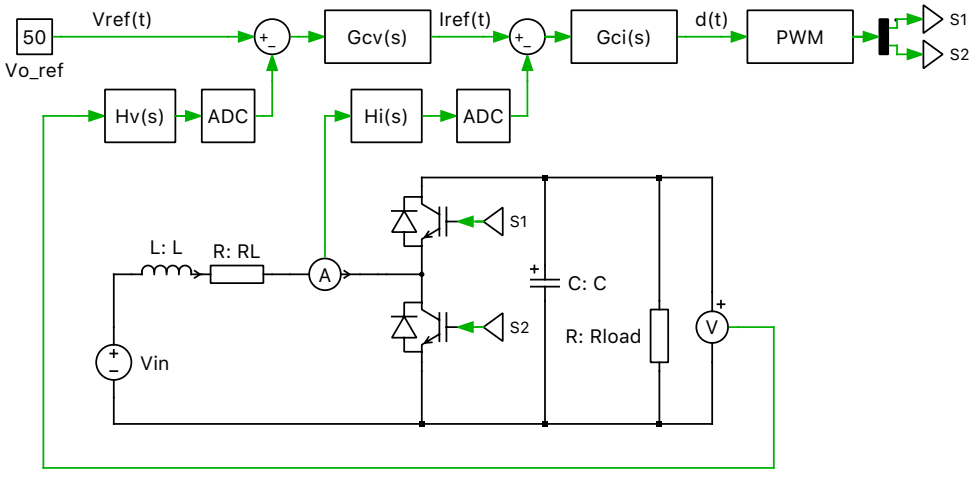

Figure 1: Conceptual diagram of ICOV control for a synchronous boost

1 Continuous Time Average Controller Design

In the Pre-lab exercises, you have designed a boost power stage and derived the converter transfer functions, $G_{id}(s)$, $G_{vd}(s)$, and $G_{vi}(s)$. Using these transfer functions, you will implement Inner Current Outer Voltage (ICOV) control. Your controller will include a current controller, $G_{ci}(s)$, and a voltage controller, $G_{cv}(s)$, for your power stage. Figure 2 shows a representation of ICOV control highlighting the inner and outer current control loops, and the transfer functions of plant, controllers, and sensor circuits. The benefits of this approach are the ability to monitor and limit the inductor current, improved current sharing among multiple parallel inverters, and improved control robustness compared to regulating the output voltage directly [1].

You may design your controllers under the following assumptions:

- Sensing circuits $H_i(s)$ and $H_v(s)$ have unity gain in the range of frequencies considered in the controller design ($H_i(s) = H_v(s) = 1$). The analog sensing circuitry and the ADCs accurately reconstruct the sampled analog signal. In practice, this requires proper sensor design and accounting for scaling along the signal measurement chain.
- ADC and PWM quantization is minimal for a 12-bit ADC and 16-bit PWM. The effect of quantization can be neglected. What is the smallest duty ratio and smallest output voltage that you can sense. **[2 pts]**
- A proportional integral (PI) regulator is suitable for both the current and voltage controllers.

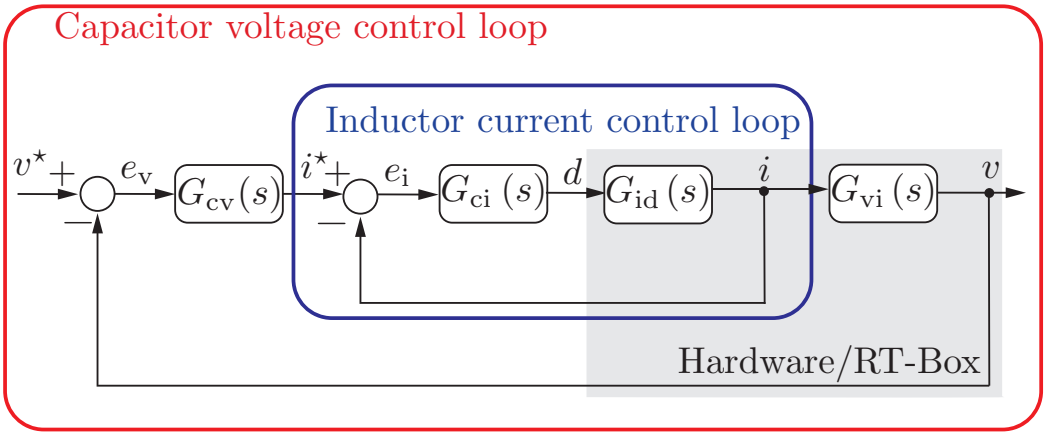

Figure 2: Transfer function diagram of ICOV control

2 Current Controller Design

The design of the current control loop is based on analyzing the inner loop of Figure 2, assuming the outer voltage loop is open and does not contribute the inductor current response. The inner loop consists of an inductor current reference, $I_{ref}(t)$, which is compared the measured inductor current. The error signal between the reference and measurement is the input to the current controller, $G_{ci}(s)$, which determines the converter duty cycle. Based on the transfer functions determined in the Pre-lab, the small signal response of the inductor current to a change in duty cycle is known around the design operating point.

The inner current loop can be summarized in the following equations:

$$\ell_i(s) = G_{ci}(s)\, G_{id}(s)\, H_i(s)$$
$$G_{id}(s) \approx G_{id,approx}(s) = \frac{V_{out}}{sL} \text{ when } s \gg \omega_o$$

$\ell_i(s)$ is the inner current loop gain with the compensator you will design, $G_{ci}(s)$. You need to use this information to approximate $\ell_i(s)$, and design the PI controller.

From the set of transfer functions that you have derived in the Prelab, show how $G_{id}(s)$ can be approximated by V_{out}/sL. When would this approximation not hold true? **[5 pts]** In this task, you will analyze the plant, $G_{id}(s)H_i(s)$, using frequency domain analysis tools. Then you will design a PI regulator, $G_{ci}(s)$, to achieve a desired transient response.

The PI controller you will design should have the from of:

$$G_{ci}(s) = G_{PIi\infty}(1 + \frac{\omega_{PIi}}{s})$$

Where $G_{PIi\infty}$ is the high frequency asymptote and ω_{PIi} is the location of the controller zero. At this stage it may be helpful to sketch the bode plot of a generic PI compensator.**[2 pts]**

Your Task:

1. Select a crossover frequency for the current controller, f_{ci}. This is where we obtain $|\ell_i(j2\pi f_{ci})| = 1$. Specify the phase margin, ϕ_m, ($\angle \ell_i(j2\pi f_{ci}) = -180 + \phi_m$). Note that $f_{ci} \leq f_{sw}/10$ and $60^\circ \leq \phi_m$ are good design practices. **[2 pts]**
2. Determine the gain and phase of the uncompensated open loop gain, $\ell_{u,i}(s)$, at the crossover frequency you have selected using $G_{id}(s)$. The uncompensated open loop gain in this case is just $G_{id}(s)$. Compare this value with the approximation, $G_{id,approx}(s)$. Note that we can use a PI Regulator when $\angle \ell_{u,i}(j2\pi f_{ci}) \geq -180^\circ + \phi_m$. This is because a PI controller can only add additional phase lag. Determine the amount of phase lag you must introduce to achieve your target phase margin in $\ell_i(s)$. **[5 pts]**
3. Write an expression for the open loop gain magnitude, $|\ell_i(j2\pi f_{ci})|$. Write an expression for the open loop phase, $\angle \ell_i(j2\pi f_{ci})$. You may use $G_{id,approx}(s)$ in your design. Refer to 1.5.2 of [1] for additional guidance. Note that formula 1.83 has an error. **[5 pts]**
4. Use your expression for $\angle \ell_i(j2\pi f_{ci}) = -180 + \phi_m$ to derive ω_{PIi}.**[2 pts]**
5. Use your expression for $|\ell_i(j2\pi f_{ci})| = 1$ to derive $G_{PIi\infty}$.**[2 pts]**
6. Build a switched circuit model in PLECS to implement the inner current control of the boost converter. Using the Pulse Generator component, step the current controller set-point from rated input current to 50% of rated input current at a rate of 20 Hz. Based on the phase margin and bandwidth, verify the following time domain quantities : overshoot, rise time, settling time from the simulation. The theoretical values (use Table if needed) should match very closely to the predicted values from simulation. For the parameter verification, show plots of the duty ratio, inductor current, capacitor voltage.**[simulation : 20 pts, each parameter verification: 5 pts]**

7. In the simulation, show that the controller can reject a 10% change in the output resistance. **[10 pts]**

8. In your lab report, include plots of following transfer functions: $G_{id}(s)$, $G_{id,approx}(s)$, $G_{ci}(s)$, and $\ell_i(s)$. **[4X5 pts]**

9. **[Only for 559]**When modeling the load of the boost converter as a resistor, a change in current reference will lead to a change in output voltage. When the reference current is increased, what initially happens to the output voltage? What can be said about the initial trajectory of the output voltage in response to a change in inductor current? What transfer function characteristic does this type of relationship illustrate? **[8 pts]**

3 Voltage Controller Design

In this section the student will design the outer loop voltage control using similar design techniques to the current loop control. Referring to Figure 2, when designing the outer loop controller, the inner current loop regulator appears as a closed loop regulator in the form of $\frac{\ell_i(s)}{1+\ell_i(s)}$.

Referring to Chapter 4.2.3 of [1], obtain the following transfer function. **[5 pts]**

$$G_{vi}(s) = \frac{G_{vd}(s)}{G_{id}(s)} \approx \frac{D' R_{load}}{2} - \frac{R_L}{2D'}$$

In this task, you will specify the cutoff frequency of the voltage control loop, f_{cv} such that we can assume the current regulator is modeled as a constant gain based on the low frequency closed loop bandwidth of the controller (i.e. the inductor current tracks the current reference). Note: a PI controller can add at maximum 90 degrees of phase lag. Choose a positive phase margin and prioritize achieving the desired w_c. w_{cv} should be $w_{ci}/10$.

Your Task:

1. Design a PI regulator for the outer loop voltage controller. Document your design: **[8X2=16pts]**
   (a) Voltage controller crossover frequency, f_{cv}:_____
   (b) Voltage controller phase margin design target, ϕ_m:_____
   (c) Uncompensated open loop gain, $|\ell_{u,v}(j2\pi f_{cv})|$:_____
   (d) Uncompensated open loop phase delay, $\angle\ell_{u,v}(j2\pi f_{cv})$:_____
   (e) PI Controller zero frequency, ω_{PIv}:_____
   (f) PI Controller high frequency asymptote, $G_{PIv\infty}$:_____
   (g) Compensated open loop gain, $|\ell_v(j2\pi f_{cv})|$:_____
   (h) Compensated open loop phase margin, $180^\circ + \angle\ell_i(j2\pi f_{cv})$:_____

2. Build a switched circuit model in PLECS to implement the outer voltage control of the boost converter. Using the Pulse Generator component, step the voltage controller set-point from rated voltage to 80% of rated voltage at a rate of 1 Hz. Based on the phase

margin and bandwidth, verify the following time domain quantities : overshoot, rise time, settling time from the simulation. The theoretical values (use Table if needed) should match very closely to the predicted values from simulation. For the parameter verification, show plots of the duty ratio, inductor current reference, inductor current, capacitor voltage.**[simulation : 20 pts, each parameter verification: 5 pts]**

3. In your lab report, include bode plots of $G_{vi}(s), G_{vi,approx}(s), G_{cv}(s), \ell_v(s)$. **[4X5 pts]**
4. **[Only for 559]** Why do we have an inner current and an outer voltage control loop. Why not the reverse or, maybe the two loops in cascade or any other architecture you could think of?**[4 pts]**

4 Real-Time Implementation of Control of Boost Converter

Once the simulation is working, we will now use the RT Box to emulate the boost converter in real-time and the controllers will be implemented in the TI Controller. The following changes are required both in the hardware and the simulation environment to ensure proper real-time implementation. With proper installation, you will now observe two new libraries in the PLECS and a new tab called "Coder".

4.1 Physical Hardware

The TI launchpad needs to be firmly fixed onto the docking station on one of the interconnection board [link]. Once set, attach the board to the RT-Box to ensure firm connection between the two. The RT-Box needs to be connected to the computer/laptop which has the corresponding PLECS simulation running by an ethernet cable. Power up the RT-Box and take good note of the LEDs and the meaning of the display.

4.2 Software Changes

The communication between the PLECS model running inside the RT-Box and the CCS launchpad is in the following way,

- The plant emulated in the RT-Box generates analog outputs which include measured parameters like inductor current and output voltage. These are then input to the Launchpad through the interconnection board.
- The Launchpad generates digital outputs which include the PWM signals. These signals go inside the RT-Box and it generates real time switching of the power MOSFETS in the model.
- Inside the PLECS library, you can find "RT-Box library". This includes various components which facilitates the interconnections. The components need the knowledge of which pin in the interconnection board is being used to pass the PWM signals from CCS to RT-Box and ADC signals in the opposite direction. This pin information can be found in the document uploaded.
- The ADC need to be scaled properly. The ADC input pins of the launchpad gets damaged if the voltage is beyond 3.3 V. In actual hardware as described in Lab 1, appropriate voltage divide circuits are designed to ensure that feedback voltage signals do not cross the 3.3 V threshold. In real-time emulation, the ADC components in

the RTBox library achieves this same by appropriate scaling and offset. Since we are only dealing with dc quantities, we can apply the correct scale factor. The scale factor would be 3.3 divided by the maximum voltage that we an expect. For the plecs current sensor, we assume that we have a 1V/1A sensor and the plecs voltage sensor the gain is 1V/1V. The scaling factors are κ_i and κ_v. We have used a safety factor of 1.2.

Select appropriate scaling factors. Show calculations for them. **[3X3 = 9 pts]**

Once proper PWM and ADC synchronization is done, we now need to write the CCS codes.

4.3 Writing the C-Code

The controller needs to be implemented in the microcontroller. We will use the same project as in Lab 1 and add minor edits to implement the controller. We will set-up three available ADC channels to input the inductor current, input voltage and the output capacitor voltage. Then follow the steps.

- Configure one channel which will generate the PWM for S2 (Fig 1), and the complement of that PWM will be supplied to S1. Make sure that this channel number corresponds to the channel number in the PLECS module.
- Accept the inductor current and capacitor voltage in two float variables named, i^{adc} and v^{adc} respectively.
- Scale them appropriately and store the actual values of physical current and voltage values in new variables namely, i and v. The formulae relating the two is,

$$i = i^{\text{adc}}(3.3/4096)(1/\kappa_i)$$
$$v = v^{\text{adc}}(3.3/4096)(1/\kappa_v)$$

  where, κ_i, κ_v are factors you have already selected.

- To test the inner current loop first, set a global variable i_{ref}, which can be changed from watch window. Store the difference between i_{ref} and i in a new variable i_{err}. This will be the input to the PI controller.
- The output of the PI controller, yi is composed of two terms, yi_{pr} and yi_{int}. We obtain yi_{pr} by multiplying i_{err} with a static gain, called proportional gain. We obtain yi_{int} by integrating i_{err} and multiplying with an integrator gain.
- The output, yi is obtianed by adding yi_{pr} and yi_{int}. Note that "duty" in Lab 1 and yi are functionally identical. Follow similar steps to output the PWM from the channel that you have configured in both the c-code as well as the PLECS simulation.
- You should saturate this yi within an upper, yi^{max} and lower value, yi^{min}. Select and report these values. You can select these based on the EE452 lab. **[2 pts]**
- Remember to saturate the output of the integrator to an upper, $yi_{\text{int}}^{\text{max}}$ and lower value, $yi_{\text{int}}^{\text{min}}$. This is required to obtain steady converter control. You can try without these too, and compare the results. These values are not random and depend strongly on your circuit operating conditions. Select these values according to nominal operating conditions and report them. **[4 pts]**

- Once the current controller seems to be working with the real time simulator, we next implement the voltage controller. We follow the same notation for variables, except replacing v for i to signify the voltage control. The major change that we now undertake is that, the output of the voltage controller, yv becomes the current reference, i_{ref}. Make sure to saturate the current reference based on the EE452 lab as well.
- Show waveforms for the inductor current, capcitor voltage and yi for a rated reference, i_{ref} to prove that the inner current control works. **[3X10 pts]**
- Show waveforms for the inductor current, inductor current reference, capcitor voltage and yi for a rated reference, v_{ref} to prove that the outer voltage control works. Note: you may use the CCS graphing feature to graph software variables, and PLECS to plot "analog" waveforms. Other methods are also acceptable. **[4X5 pts]**
- **[Only for 559]** You now need to implement a changing reference for i_{ref} and v_{ref} like you did in the simulations. Record the waveforms as in previous steps showing both the current and voltage control. **[10+10 pts]**
- **[Only for 559]** As a final step, implement an anti-windup to this controller. Considering the output yi can vary between $yi^{\min}$ and $yi^{\max}$, we need to take some corrective action every time it extends beyond that range. Whenever that happens, we saturate it at the limit it hits and then stop the integration part of the PI integrator, so that the huge error which led to saturation is no longer integrated and accumulated. Once the controller stabilizes the plant by only the proportional controller and the value of yi returns in the range $(yi^{\min}, yi^{\max})$, the integrator is reinstated. An easy way of removing the integrator is to multiply the input of the integrator with zero. **[5 pts]**

5 Nomenclature

Most of the symbols will be explained as they come up in text/ if they are from class notes, they will **not** be re-explained here. Some of the new terms are as follows:

1. The uncompensated loop gain $\ell_u(s)$,is defined as the system loop gain when a unity compensation is employed, that is, when $G_c(s) = 1$. If this is not clear, read pg 26 [1]
2. We define ω_o as the natural frequency of the converter. If you do not understand what this is for your converter, refer to Table 8.2 [2]. It should be clear what is the unit of ω_o from the text.
3. If you have problem understanding notation, $\ell(s)$ would mean open loop gain. If this is not clear, read Ch 2 of Skogestad book. The book by Maksimovic and Erickson, [2] refers to $\ell(s)$ as $T(s)$. They are both the same, just different conventions.
4. For any nomenclature or symbol issue in Section 3 and 4, refer to Chapter 4 of [1]. You could alternatively find them from your class note if that chapter has been covered.

References

[1] L. Corradini, D. Maksimovic, P. Mattavelli, R. Zane *Digital control of high-frequency switched-mode power converters.* Hoboken, NJ: John Wiley & Sons, 2015.

[2] Erickson, Maksimovic *Fundamentals of Power Electronics* 2nd Edition