NTNU

Kunnskap for en bedre verden

Fakultet for
Informasjonsteknologi og elektroteknikk

ELDI2002 - Reguleringsteknikk for elektriske energisystemer

Block 1 - Control Challenge

Markus Arntzen
Vera Wigum

14.09.2025

Block 1 - Control systems

Contents

Introduction

In this challenge, we focus on modeling and controlling a DC-DC converter, a nonlinear system commonly used in power electronics. The task is divided into three parts. In Part I, we develop a state-space representation, analyze equilibrium conditions, and design a control strategy that clearly states the physical system and the controller. In Part 2, we will test and verify our analysis in MATLAB/Simulink, with emphasis on the system's response to reference changes, disturbances, and modeling uncertainties. Part 3 evaluates closed-loop performance using P and PI controllers—with and without equilibrium control—and in Part 4, we will introduce an outer loop to regulate both state-space variables. The overall goal is to build a robust control framework that performs reliably under both nominal and perturbed conditions.

For this project, we have chosen the model of a common **DC/DC converter** (see Figure 1). We have chosen parameters from a given set of rules from earlier in this course. These parameters are:

```
V0 = 12;                                % input voltage
Vc_nom = 16;                            % (desired) output voltage

D = (Vc_nom-V0)/Vc_nom;                 % Nominal duty cycle
P_nom = 100;                            % output power
I0_nom = P_nom/Vc_nom;                  % Output current
fs = 50000;                             % switching frequency

I0_max = 0.1 * I0_nom;                  % rippel current 10%
Vc_max = 0.05 * Vc_nom;                 % rippel voltage 5%

L_min = (V0*D)/(fs*I0_max);             % Min inductance
C_min = (I0_nom*D)/(fs*Vc_max);         % Min capacitance

n = 0.95;                               % Nominal efficiency

Il_nom = P_nom/(n*V0);                  % nominal input current
P_loss = (1-n)*P_nom;                   % Power loss

R = 3.75/Il_nom^2;                      % 0.75 * P_loss, Resistance
G = 1.25/Vc_nom^2;                      % 25% * P_loss, Conductance
L = 1.05 * L_min;                       %Actual value of the inductor
C = 1.05 * C_min;                       %Actual value of the capacitor

```

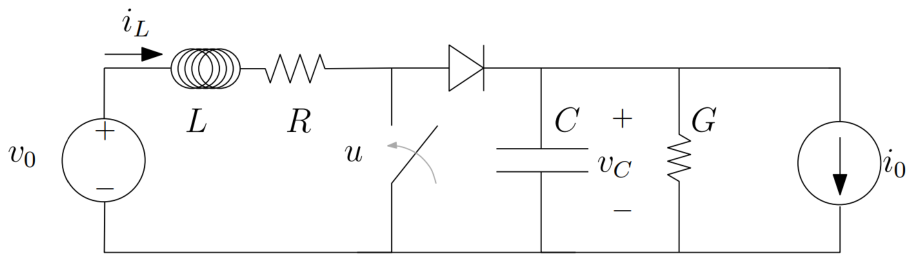


Figure 1: DC/DC converter

1 Analytical System Calculations

1.1 State-space representation

In this section, we will present our model in a **state-space model**. Our chosen model is the **DC/ converter** where we have been given these two equations;

$$L\frac{di_L}{dt} = -Ri_L - (1-u)v_C + v_0 \tag{1}$$

$$C\frac{dv_C}{dt} = -Gv_C + (1-u)i_L - i_0 \tag{2}$$

Defining our state-space variables:

- Inductor current, i_L
- Capacitor voltage, v_C

The system in matrix form becomes:

$$\begin{bmatrix} \dot{i}_L \\ \dot{v}_C \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & -\frac{G}{C} \end{bmatrix} \begin{bmatrix} i_L \\ v_C \end{bmatrix} + u \begin{bmatrix} 0 & -\frac{1}{L} \\ -\frac{1}{C} & 0 \end{bmatrix} \begin{bmatrix} i_L \\ v_C \end{bmatrix} + v_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} - i_0 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Simplified as:

$$\dot{\mathbf{x}} = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & -\frac{G}{C} \end{bmatrix} x + \begin{bmatrix} 0 & -\frac{1}{L} \\ -\frac{1}{C} & 0 \end{bmatrix} ux + \begin{bmatrix} 1 \\ 0 \end{bmatrix} v_0 - \begin{bmatrix} 0 \\ 1 \end{bmatrix} i_0$$

where

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{i}_L \\ \dot{v}_C \end{bmatrix}$$

Nonlinearities. The system is nonlinear because the control u multiplies the states i_L and v_C. This gives terms like $u\,i_L$ and $u\,v_C$, which are not linear.

1.2 Equilibrium Analysis

In this section, we perform a brief **equilibrium analysis**. DC/DC converters are switch–mode converters, which means their dynamics are nonlinear and time–varying. The duty cycle $u(t)$ controls a switch that is either fully ON or fully OFF, leading to two piecewise differential equations: one when the switch is ON, and another when it is OFF. Direct analysis of such a model is complicated, so instead we use the averaged model. In this approach, the duty cycle $u(t)$ is treated as a continuous fraction of the switching period (rather than strictly 0 or 1), which leads to a smoother system representation suitable for equilibrium and stability analysis.

At equilibrium, we impose $\dot{i}_L = 0$ and $\dot{v}_C = 0$. From the state-space model, this leads two algebraic equations:

$$0 = -\frac{R}{L}\bar{i}_L - \frac{1}{L}(1-\bar{u})\bar{v}_C + \frac{v_0}{L} \tag{3}$$

$$0 = \frac{1}{C}(1-\bar{u})\bar{i}_L - \frac{G}{C}\bar{v}_C - \frac{i_0}{C} \tag{4}$$

$\boldsymbol{C}$ and $\boldsymbol{L}$ disappear, leaving us with the equations:

$$-R\bar{i}_L - (1-\bar{u})\bar{v}_C + v_0 = 0 \tag{5}$$
$$(1-\bar{u})\bar{i}_L - G\bar{v}_C - i_0 = 0 \tag{6}$$

Step 1: Solve for $\bar{u}$ in terms of $\bar{v}_C$ and $\bar{i}_L$:

We can substitute $(1-\bar{u}) = a$ and from (6):

$$a\bar{i}_L - G\bar{v}_C - i_0 = 0 \quad \implies \quad a = \frac{G\bar{v}_C + i_0}{\bar{i}_L} \tag{7}$$

Step 2: Substitute a into (5)**:**

$$-R\bar{i}_L - \left(\frac{G\bar{v}_C + i_0}{\bar{i}_L}\right)\bar{v}_C + v_0 = 0 \tag{8}$$

Step 3: Multiplying by $\bar{i}_L$ gives:

$$-R\bar{i}_L^2 - (G\bar{v}_C + i_0)\,\bar{v}_C + v_0\bar{i}_L = 0. \quad \implies \quad R\bar{i}_L^2 - v_0\bar{i}_L + \left(G\bar{v}_C^2 + i_0\bar{v}_C\right) = 0. \tag{9}$$

From this equation, we can see the power flow of the system:

$$v_0\bar{i}_L - R\bar{i}_L^2 - G\bar{v}_C^2 - i_0\bar{v}_C = 0$$

The **first term** is the power delivered by the source, the **second term** the power loss through the resistor, the **third term** the power loss through the conductance, and the **final term** is the power delivered to the load.

Step 4: Solve the quadratic in eq.(9) **for $\bar{i}_L$**

$$R\bar{i}_L^2 - v_0\bar{i}_L + \left(G\bar{v}_C^2 + i_0\bar{v}_C\right) = 0.$$

$$\bar{i}_L = \frac{v_0 \pm \sqrt{v_0^2 - 4R(G\bar{v}_C^2 + i_0\bar{v}_C)}}{2R} \tag{10}$$

—

Step 5: Solve for a (hence $\bar{u}$).

Supplementing $\bar{i_L}$ to eq.(7)

$$(1-\bar{u}) = a = \frac{G\bar{v}_C + i_0}{\frac{v_0 \pm \sqrt{v_0^2 - 4R(G\bar{v}_C^2 + i_0\bar{v}_C)}}{2R}}$$

Simplifying,

$$\bar{u} = 1 - \frac{2R(G\bar{v}_C + i_0)}{v_0 \pm \sqrt{v_0^2 - 4R(G\bar{v}_C^2 + i_0\bar{v}_C)}}$$

Choosing whether $\pm$ will be positive or negative is determined by the value of $\bar{u}$. $\bar{u}$ has to be between 1 and 0, as it defines whether the switch is open or closed. In both cases of $\bar{u}$, the value is between [1,0], which gives us the freedom to choose which value we will use. From the parameter values we defined earlier, we chose that the duty cycle will be 0.25, therefore the $\bar{u}$ closest to our duty cycle will be the natural choice. In this case, minus is the best option.

Therefore,

$$\bar{u} = 1 - \frac{2R(G\bar{v}_C + i_0)}{v_0 - \sqrt{v_0^2 - 4R(G\bar{v}_C^2 + i_0\bar{v}_C)}} \tag{11}$$

Equilibrium Set. The admissible equilibrium set is given by

$$\mathcal{E} = \left\{ (\bar{i}_L, \bar{v}_C, \bar{u}) \;\middle|\; \bar{i}_L = \frac{v_0 - \sqrt{v_0^2 - 4R\big(G\bar{v}_C^2 + i_0\bar{v}_C\big)}}{2R}, \qquad \bar{u} = 1 - \frac{Gv_C \bar{+} i_0}{\bar{i_L}} \right\}$$

subject to the admissibility conditions

$$v_0^2 - 4R\big(G\bar{v}_C^2 + i_0\bar{v}_C\big) \;\geq\; 0, \qquad 0 \;\leq\; \bar{u} \;\leq\; 1.$$

Existence of Equilibrium. The discriminant of u must be equal to or greater than zero:

$$v_0^2 - 4R(G\bar{v}_C^2 + i_0\bar{v}_c) \geq 0.$$

If this condition is violated, no real-valued equilibrium exists. For instance, if the load R is too large or $\bar{v}_C$ chosen too high, then the discriminant becomes negative, preventing equilibrium. In our case, with the chosen parameters from the introduction, we have that the discriminant is a positive number, which indicates that the equilibrium exists.

Notice that both $\bar{i}_L$ and $\bar{v}_C$ cannot be specified independently. Once $\bar{v}_C$ is chosen, $\bar{i}_L$ is uniquely determined through (10), and the control u is then constrained through the quadratic condition above. This coupling highlights the nonlinear dependence of equilibrium points on both states and the control input.

1.3 Overall Control Strategy

The control setup for the DC/DC converter can be divided into two main parts:

- **Physical system:** This is the boost converter itself. In the Simulink model, it is built from the equations from Section 1.1 and includes the inductor current i_L, the capacitor voltage v_C, the input v_0, and the load.
- **Controller:** This block decides the duty ratio u (switching signal). Its job is to keep the output voltage v_C close to a chosen reference value v_C^{ref}. It does this by adjusting u when the load or input voltage changes.

In the Simulink diagram, the physical system and the controller are shown as separate blocks so it is clear which part represents the real converter and which part is the control strategy (see Figure 2 in the next section).

2 Matlab/Simulink

In Simulink, the control function block consists of the equations where we solved for $\bar{\boldsymbol{i}}_{\boldsymbol{L}}$ and $\bar{\boldsymbol{u}}$. The function block in the *Physical System Model* represents the differential equations $\dot{\boldsymbol{x}} = [\frac{d\boldsymbol{i_L}}{dt}, \frac{d\boldsymbol{V_c}}{dt}]$ of the real physical system. The outputs $\boldsymbol{i_L}$ and $\boldsymbol{v_C}$ are integrated to get the actual function variable values before being fed back into the system block alongside $\boldsymbol{u}$ as inputs for the differential equations. This way, we will see how the system will behave as time goes on, starting with initial conditions of the integrator blocks and varying as the control input/signal input influences the system response.

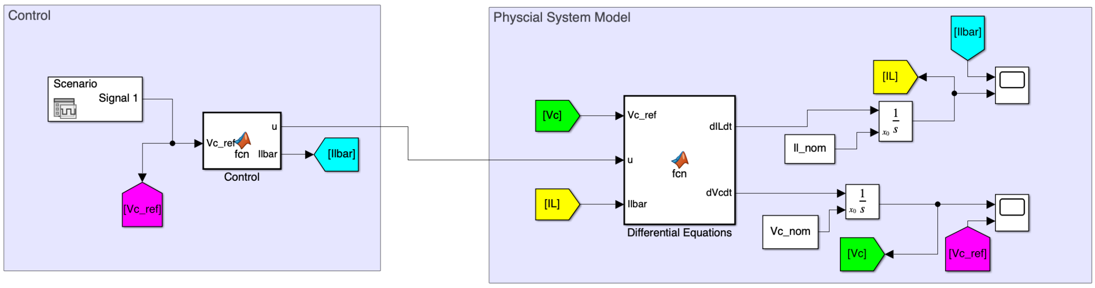


Figure 2: System overview in Simulink for **2.1**, **2.2** and **2.3**

2.1 Computation of $\bar{x}$ and $\bar{u}$

We chose $\boldsymbol{V_C^{ref}}$ as our chosen variable and solved for the other variables $\boldsymbol{i_L}$ and $\boldsymbol{u}$ in the differential equations as shown in 1.2 for equilibrium control. These equations are put directly into the controller (function block called "Control") with $\boldsymbol{u}$ as the output variable which is fed into the physical system as control input. Inside the Control block, $\boldsymbol{V_C^{ref}}$ is set as input because we want to be able to adjust the reference/nominal output voltage of the system. The other variables are constants defined in the Matlab code.

Control

DC_DC_Converter_SimuLink ▸ Control

```
function [u, Ilbar] = fcn(V0, R, G, I0_nom, Vc_ref)

Ilbar = (V0 - sqrt(V0^2 - 4*R*(G*Vc_ref^2 + I0_nom*Vc_ref))) / (2*R);
u = 1 - (G*Vc_ref + I0_nom) / Ilbar;
```

Figure 3: Controller function block with equations solved for $\boldsymbol{i_L}$ and $\boldsymbol{u}$

2.2 Intialization of the system

As seen in the overview image in figure 2, in order to initialize our system at the desired output, we give the respective integrator blocks an input of the constant desired output, as a starting point or initialization (these constants can be taken directly from their variable names defined in Matlab). This means that we can think of the system as being turned on some time ago and given enough time to stabilize. We can therefore choose to look only at what happens when we introduce disturbances like we do in the next subtask.

2.3 Equilibrium application and disturbance introduction

In this subsection, we want to apply some varying inputs into the system represented as step-changes. We choose to use the signal editor for this as it is easy to map out many different steps in one input block. The figure 4 below shows our step-change inputs, at which time and at what values.

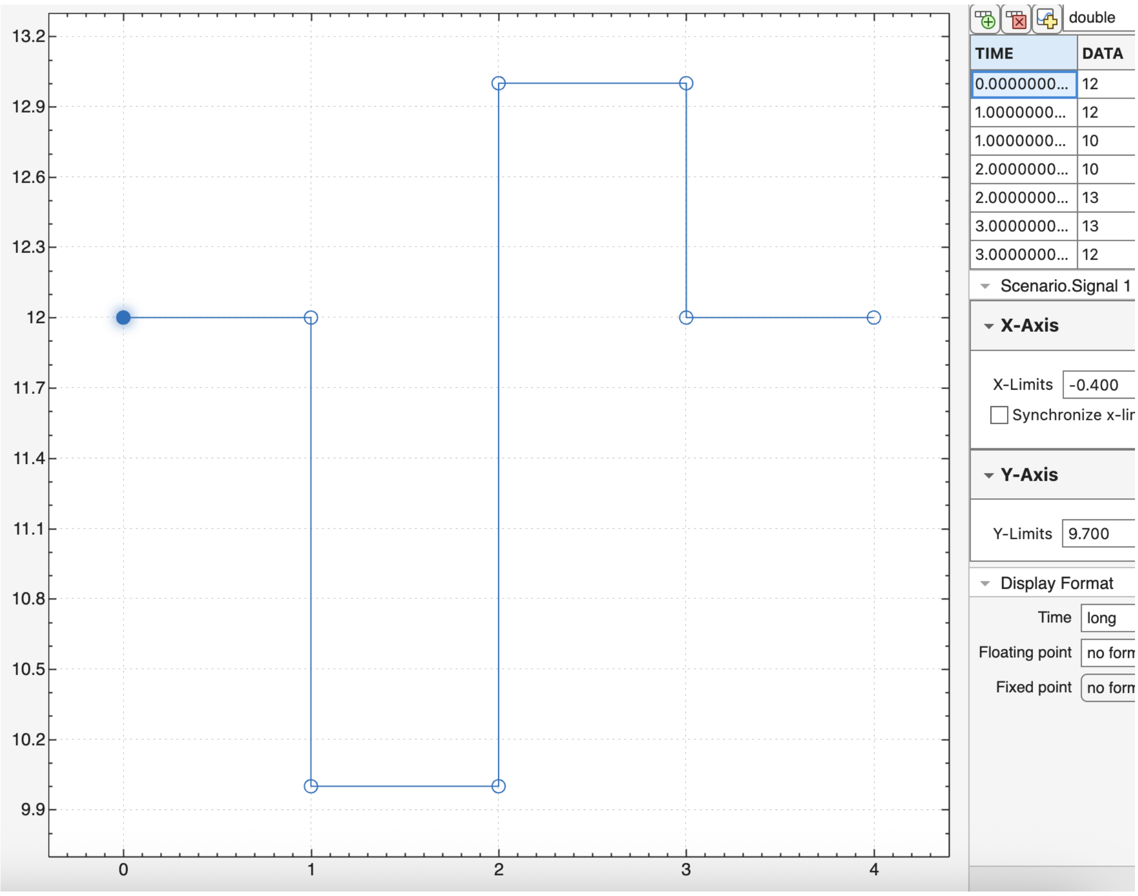
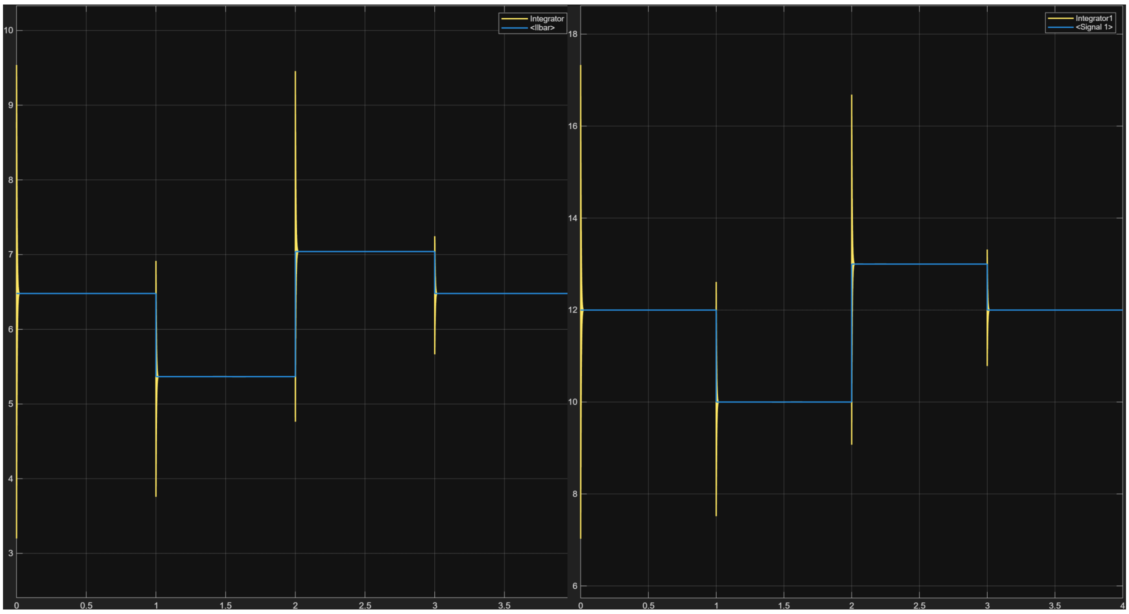


Figure 4: Map of the different input step-changes with desired time and values

When the step-change is applied, we can observe that the system experiences a great deviation and uses some time to adjust to the nominal values. (see figure 5). This is due to the system response with two power storing elements.

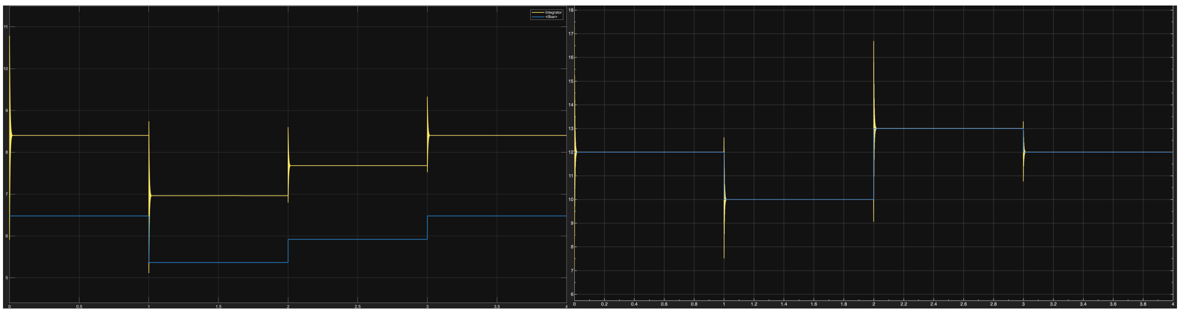


(a) i_L **(b)** V_C

Figure 5: Equilibrium of V_C and i_L

2.4 Repetition of 2.1 with errors

In the physical systems model, we now artificially introduce some errors to see what happens to the system. This means we only change values in the physical systems block and not in the controller, because we still expect that the system is behaving nominally with the values we design for in the control. We chose to adjust R to $R_{new} = 0.8 * R$ and adjust $I0_{nom}$ to $I0_{nom-new} = 1.3 * I0_{nom}$. By looking at the measured values, they differ from the i_{Lnom} and V_{Cnom} that we expect the system to equalize at. The performance, or the "way it equalizes", is still the same because the system is still the same. The controller however, "thinks" we operate at the nominal values of our system, but in reality we have component values that differ a little bit. This will in turn create variance in our measured equilibrium values of I_L as we see in Figure 5 (a). The values for V_C does not differ from V_{Cref} as seen in Figure 5 (b). This makes sense as the controller, and u, forces V_C to be at V_{Cref} and therefore allows i_L to differ from our expected nominal value.



(a) i_L **(b)** V_C

Figure 6: Scope measurements of i_L and V_C with respect to their expected values

3 PI Controller

We want to be able to control our system performance and get rid of the big transients we observe in our system before it reaches equilibrium. We will eventually do this with a PI-controller, but build up to it.

3.1 Closed loop

In this section, we will test the system with different conditions. Because we have a non-linear system, our inner loop feedback consists of taking the difference of $\bar{i}_L$ (which is the calculated i_L^{ref} based on V_c^{ref}) and i_L (which is the measured current at any given time).

3.1.1 P - Control without $\bar{u}$

We leave controller output for u empty and use the controller output for $\bar{i}_L$ through a sum block and a gain block before going in as input u into the physical system (see figure 6). The gain effectively creates the portion control.

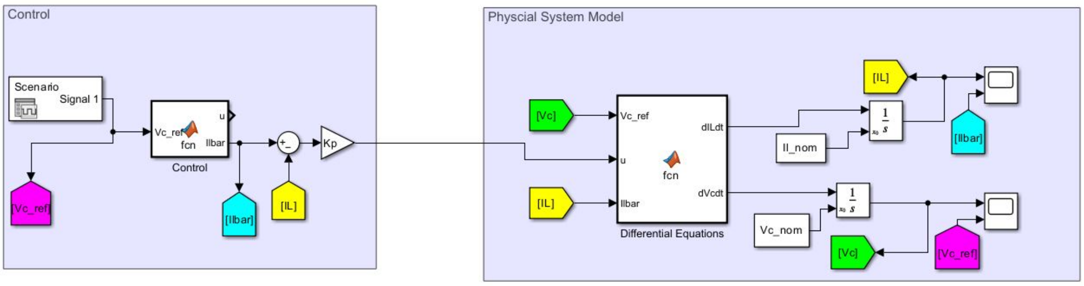


Figure 7: P-controller closed loop without u

By running the simulation and looking at the plot in figure 7, we can see that the measured equilibrium values (yellow graph) for each step input differ by a small amount from the expected equilibrium values (blue graph) and give us a steady-state error.

This happens because at equilibrium

$$e = \bar{i}_L - i_L = 0 \qquad \Longrightarrow \qquad u = Kp \cdot e \neq 0 \tag{12}$$

Since u can not be 0 in our system as it does not make sense to stop switching the circuit and still maintain the desired output voltage, the system will therefore land on a different steady-state value than the reference value to keep the subtraction $\neq 0$. A $K_p = \infty$ would make it mathematically possible to have $\bar{i}_L - i_L = 0$, but this is not possible in reality and thus, not a solution to the problem in Simulink. In addition, by increasing K_p over a certain threshold value, will make the system use the invalid value of i_L from equation 10, making the system unstable.

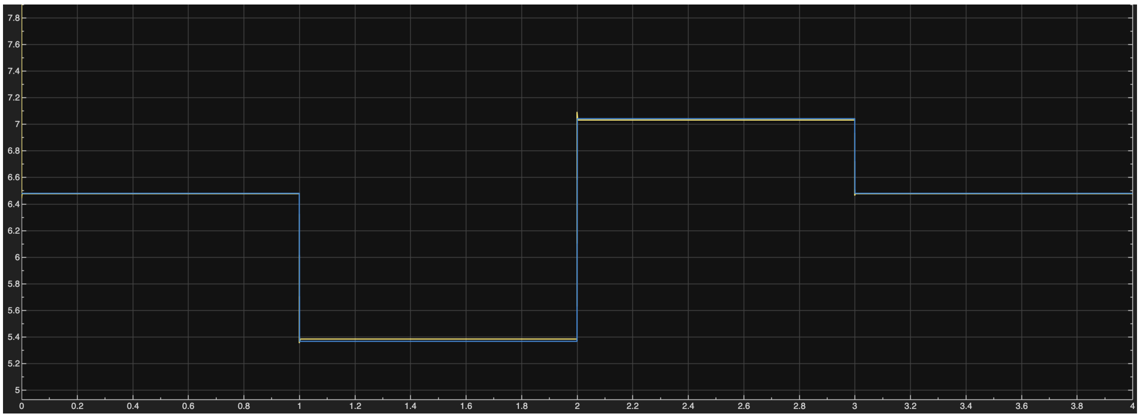


Figure 8: Measured values of i_L for P-controlled circuit **without** u

We do, however, see the P-controller trying to force down the big overshoot response values and oscillations we see when step-changing the input with open-loop only. The graph in figure figure 7 shows results with $K_p = 10$.

3.1.2 P - Control with $\bar{u}$

To eliminate the steady-state error we get when configuring our system like in 3.1.1, we will use a sum block after the K_p gain and u-input on the physical system block where $\bar{u}$ will be summed together with P-controlled $\bar{i}_L$ (figure 8).

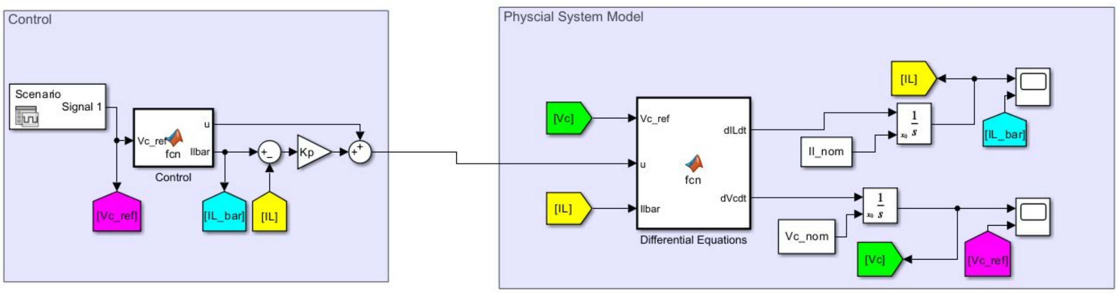


Figure 9: P-controller closed loop with u

As we can see in the plot in figure 9, the steady-state error is eliminated as the system can now mathematically allow $(\bar{i}_L - i_L) = 0$ because we are adding $\bar{u}$, creating the equation

$$\bar{u} = K_p \cdot e + \bar{u} = \underbrace{K_p(\bar{i}_L - i_L)}_{\text{becomes } 0} + \bar{u} \quad \Longrightarrow \quad \bar{u} = \bar{u} \tag{13}$$

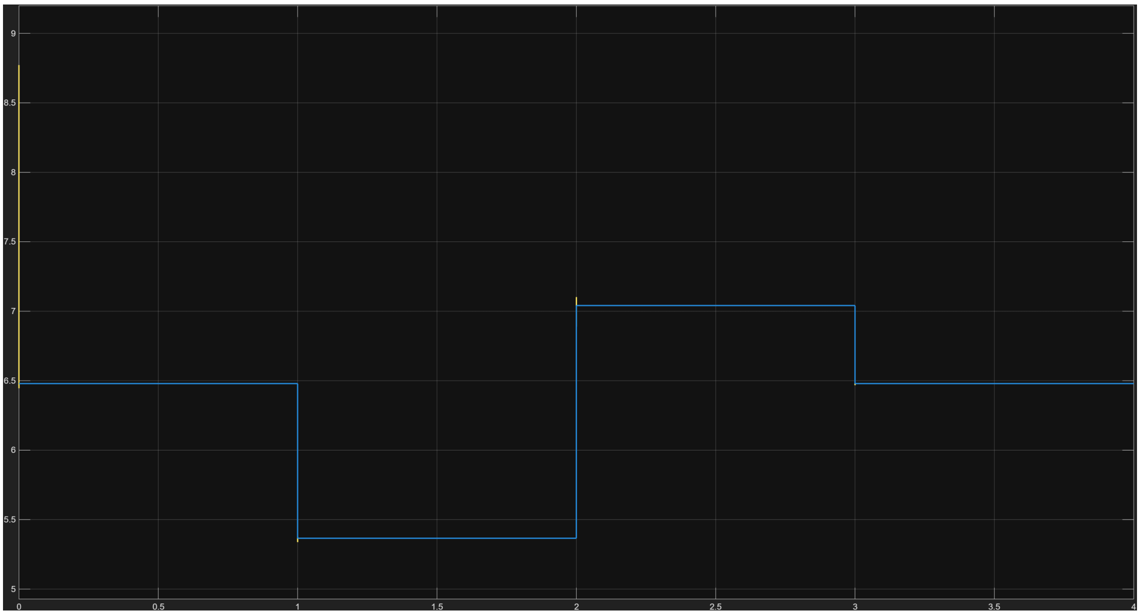


Figure 10: Measured values of i_L for P-controlled circuit **with** u

3.1.3 PI - Control

To better control the over-/under-shoot in the step-change responses, we want to add an integrator. An integrator will keep track of the "history" of the system and work alongside the proportional controller to better force the output value closer to the desired value. It sums up all errors with time, and the longer the time, the more the integrator will build up and try to force the output to the desired value. Figure 10 displays the circuit with a PI controller without output u connected.

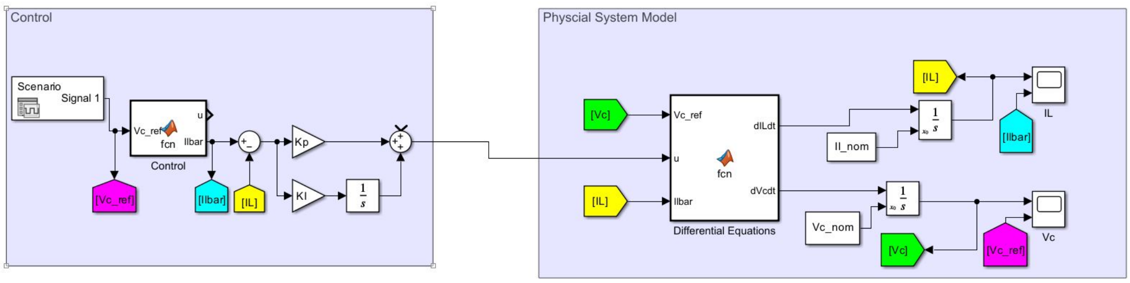


Figure 11: PI-controller without u

The plot in figure 11 displays the system reaching the desired equilibrium values for i_L with the PI-controller. In addition to better transient control, the integrator will add to equation 13 removing the need of $\bar{u}$ in the equation for u.

$$\bar{u} = K_p \cdot e + K_I \cdot \int (e)\, dt = \underbrace{K_p(\bar{i}_L - i_L)}_{\text{still becomes } 0} + K_I \int (\bar{i}_L - i_L)\, dt \tag{14}$$

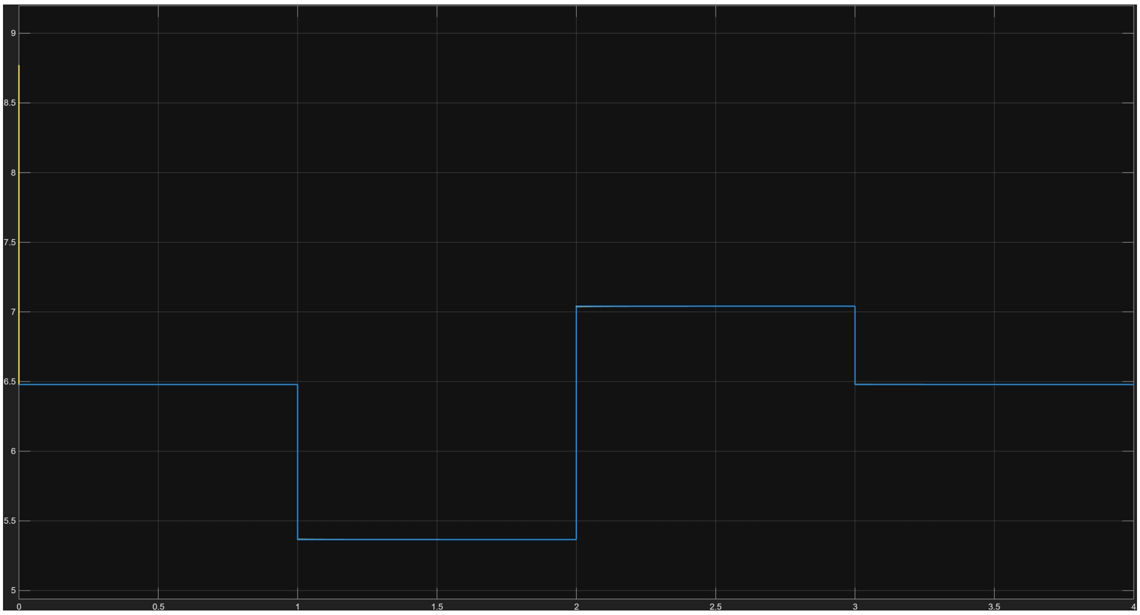
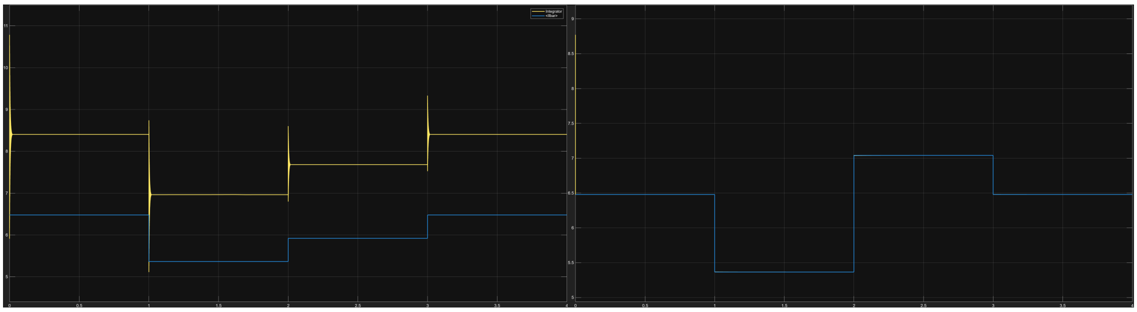


Figure 12: Measured values of i_L for PI-controled circuit without u

3.2 PI compared to equilibrium

Comparing the plot of equilibrium control without feedback, or open loop, and the plot of the PI-controlled circuit, we can see big differences (Figure 12 displays the plots side by side). The open loop equilibrium plot shows the transient oscillations are free to move as they like and only focuses on stabilizing at the desired equilibrium values. The PI-controlled closed loop, however, manages to control the unwanted transient oscillations, when step changing, to a point where they are within our set margin ranges while securing stabilization at the desired equilibrium values.



(a) Open loop **(b)** PI-controled closed loop

Figure 13: Scope measurements of i_L for open loop and closed loop with respect to their expected equilibrium values.

4 Outer Loop

When looking at the measured V_C values, we still experience big spikes in the transient step-change responses with the inner loop PI-controlled circuit. We want to add an outer loop controller where we use the measured V_C values as feedback to try to control the transient responses for V_C, which we want to keep at a stable value. Figure 13 displays our system with the outer loop control.

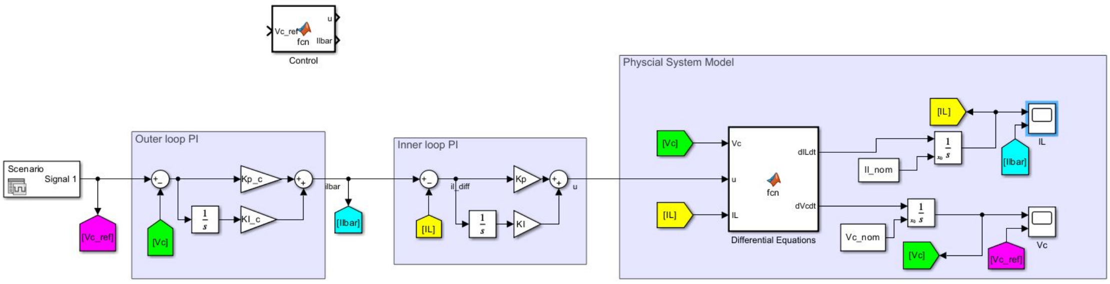


Figure 14: Circuit with outerloop PI-control in Simulink

Looking at the plot of V_C in figure 14, we can see that we managed to achieve nearly perfect control of V_C, which is our actual goal for this system.

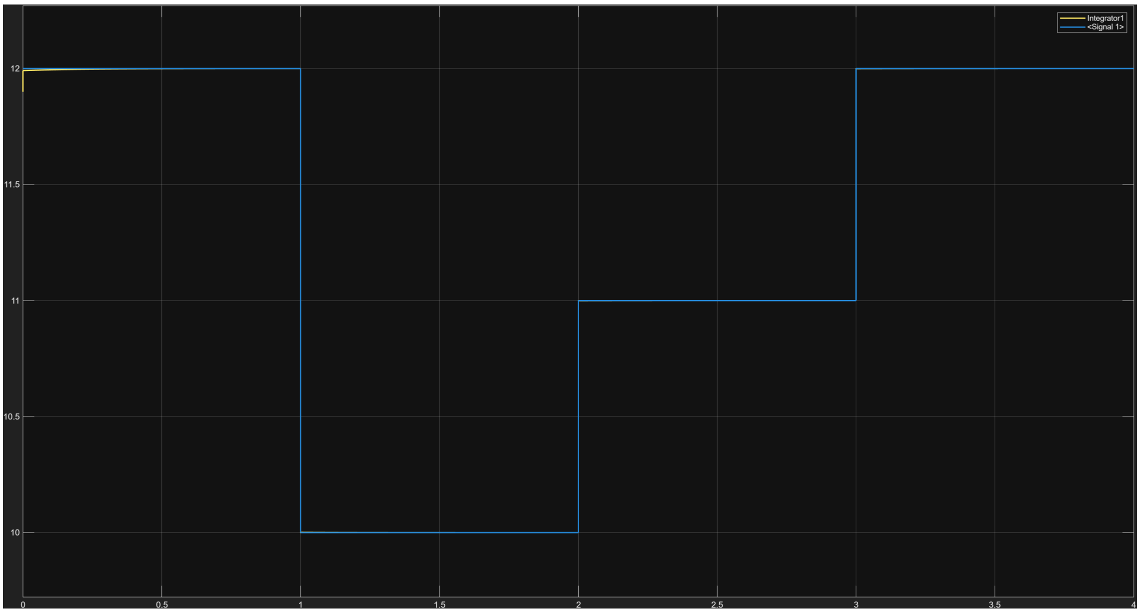


Figure 15: Measured V_c with outer loop PI-controller

We can see i_L spike whenever we step-change the system (figure 15). Even though these spikes are not of unreasonable values, the values necessary for the outer-loop K_P and K_I had to be set to really high in order to satisfy the level of control for V_C in figure 14, in this case, 1000 for both. This also causes the expected values for $\bar{i}_L$ to be very far from the actual values measured whenever the system step-changes.

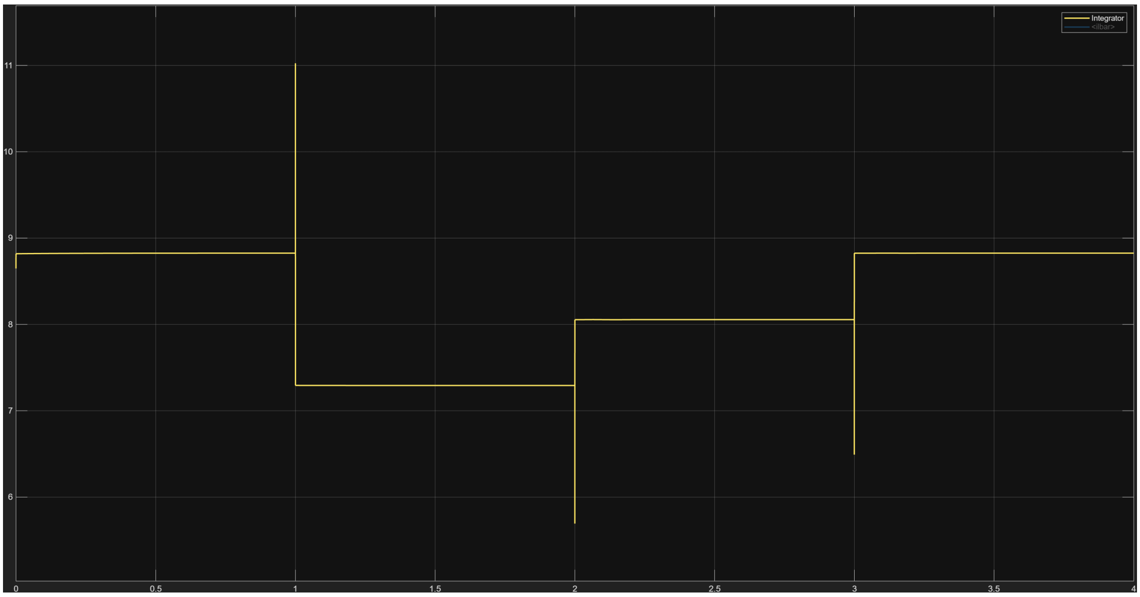


Figure 16: Measured i_L with outer loop PI-controller

In order to achieve a stable control of V_C, we are not entirely certain *why* our outer-loop system requires such high values of K_P and K_I. We had to change our $V_{C_{nom}}$ to 11.9 from our initial 16 and reduce the step change from 10-13 to 10-11 in order to make V_C **actually** stabilize like we see in figure 15. These changed values is what is included in the Matlab/Simulink files attached on Blackboard.