electronics

MDPI

Article

Educational Workshop on STM32 Digital Control in Buck Converters: Design, Development, and Online Resources

Fabio Corti [1,*], VipinKumar Shriram Meshram [2], Inmaculada Casaucao [3], Alicia Triviño-Cabrera [3], Alberto Reatti [1] and Francisco Javier López-Alcolea [4]

[1] Dipartimento di Ingegneria dell'Informazione, Università Degli Studi di Firenze, 50139 Florence, Italy
[2] Dipartimento di Ingegneria, Università Della Campania Luigi Vanvitelli, 81031 Aversa, Italy
[3] Departamento de Ingeniería Eléctrica, Universidad de Málaga, 29071 Málaga, Spain
[4] School of Industrial Engineering, University of Castilla-La Mancha, 13001 Ciudad Real, Spain
* Correspondence: fabio.corti@unifi.it

Abstract: This paper presents a comprehensive guide for the design and development of a power electronic converter, specifically focusing on the Buck converter. The guide aims to bridge the gap between theoretical knowledge and practical application, providing engineering students with a hands-on experience in digital control using the STM32 microcontroller. The workshop experience described covers all necessary steps, from the initial design and simulation using tools like MATLAB-Simulink and Plecs to the final implementation and testing of the converter. The educational value of this approach is highlighted, emphasizing the importance of practical engagement in understanding complex concepts related to power electronics.

Keywords: educational; power electronics; Buck converter; programming; experimental results

check for updates

Citation: Corti, F.; Meshram, V.S.; Casaucao, I.; Triviño-Cabrera, A.; Reatti, A.; López-Alcolea, F.J. Educational Workshop on STM32 Digital Control in Buck Converters: Design, Development, and Online Resources. *Electronics* **2024**, *13*, 3207. https://doi.org/10.3390/electronics13163207

Academic Editors: Antonio J. Marques Cardoso and Imed Jlassi

Received: 9 July 2024
Revised: 8 August 2024
Accepted: 12 August 2024
Published: 13 August 2024

1. Introduction

In the dynamic field of power electronics, the gap between theoretical understanding and practical application is often considered an obstacle to overcome, especially for students entering the realm of engineering. The significance of hands-on experience cannot be overstated, as learning becomes more profound when immersed in the intricacies of real-world applications. This paper tries to bridge the gap between theory and practice, providing a comprehensive guide for engineering students to understand a power converter's design, implementation, and control.

The importance of this topic is confirmed by the numerous works currently available in the literature, as shown in Table 1. In [1], a photovoltaic power systems laboratory module integrating MATLAB real-time software and DSP hardware is presented. It focuses on project-based learning to teach design, control algorithms, and hardware fabrication. In [2], the implementation of project-based learning in a power electronics course at the University of Oviedo is proposed. Students engage in designing and constructing a switching-mode power supply and a DC–DC converter topology, applying theoretical knowledge practically. In [3], a MATLAB-based graphic tool for designing AC/DC and DC/DC switching-mode power supplies is proposed. It enhances the learning experience by allowing students to apply theoretical knowledge in practical sessions. In [4], a laboratory course focused on teaching power electronics and renewable energy using industry-standard tools like MATLAB/Simulink and dSPACE is described. It includes ten unique experiments and design projects for practical learning. In [5], the development of a reconfigurable experimental bench for power electronics and drives is detailed. It supports various circuit topologies and is used for postgraduate laboratory courses to enhance learning through hands-on experiments. In [6], an introductory course on the FPGA-based digital control of DC/DC converters is presented. It combines theoretical justifications, CAD tools, and lab experience to teach the design and implementation of a linear compensator based on an FPGA. In [7], the use of design-oriented, project-based learning in a power electronics course at the Technical University of Denmark is explored. Students design power converter applications, including fuel cell converters and LED drivers, to apply their theoretical knowledge. In [8], a method to teach the analysis and simulation of power electronic converters using MATLAB/SIMULINK is described. It focuses on system-level nonlinear state-space models to simulate and design controllers for power converters. In [9], tutorial activities focused on real-world limitations and non-technical aspects of power electronics projects are introduced. It contrasts traditional teaching methods with interactive, self-learning tutorials designed to improve students' abilities to design and troubleshoot power electronic circuits considering real-world constraints. The effectiveness of these new tutorials was validated through their application at two Australian universities, showing significant improvements in students' scores and engagement. Finally, in [10], a comprehensive graduate-level course on magnetic design, addressing gaps in current power electronics education is proposed. The course includes theoretical foundations and hands-on skills, focusing on the design of magnetic components such as inductors, transformers, and wireless inductive power transfer systems. The course content aligns with the needs of the power electronics industry and academia, aiming to enhance students' skills in magnetic design beyond what is typically covered in general power electronics courses.

Table 1. Literature overview.

Reference	Simulation	Coding	Experimental	Supplementary Materials
[1]	Yes	Yes	Yes	No
[2]	No	Yes	Yes	No
[3]	Yes	Yes	No	No
[4]	No	Yes	Yes	No
[5]	No	Yes	Yes	No
[6]	No	Yes	Yes	No
[7]	No	Yes	Yes	No
[8]	Yes	Yes	No	No
[9]	No	No	No	No
[10]	No	Yes	Yes	No
Proposed	Yes	Yes	Yes	Yes

The focal point of this guide is the Buck converter, chosen for its relevance and educational value, as it represents a fundamental building block in power electronics. The approach is to simplify the learning process by breaking down complex concepts into manageable steps, facilitating a more intuitive understanding. Recognizing that practical engagement enhances comprehension, the guide employs practical analyses and demonstrations, steering away from an exclusively theoretical treatment.

The following guide enables students to learn the use of multiple tools relevant to the Power Electronics field, such as MATLAB-Simulink, Plecs, and the STM32 development environment. The guide is divided into the six main steps required for the first design of a power converter:

1. Converter design: The operational principles of the converter and key sizing equations are presented.
2. Small-signal model: The derivation of a small-signal model containing information about the converter's frequency response is demonstrated. The main characteristics of the converter's transfer function, which are essential for controller design, are analyzed.
3. Continuous-time controller design: The process of designing a continuous-time controller to achieve specific speed and closed-loop stability performance is explained.
4. Controller discretization: As the article focuses on digital control design, the principal techniques for discretizing the controller are showcased and compared.

5. Implementation of the controller in the STM32 environment: The guide illustrates how to develop the necessary code for the practical implementation of the controller in the STM32 environment.
6. Experimental setup of the converter: All practical aspects necessary for converter realization, such as component fabrication, and the characteristics of the sensors required for voltage/current measurements, are analyzed.

To facilitate learning, six lessons containing slides and codes are made available. In the following sections, all the points are analyzed in detail.

2. Steady-State Buck Converter Design

2.1. Circuit Topology

The Buck converter circuit is shown in Figure 1.

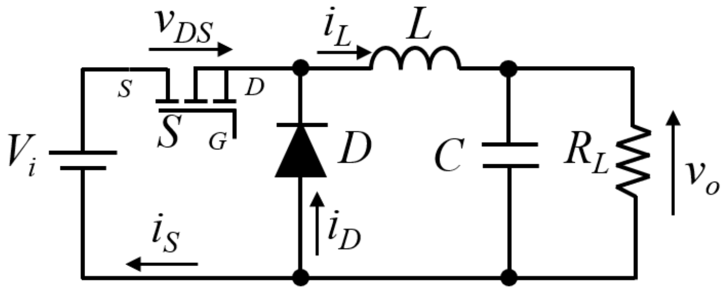


Figure 1. Buck converter general schematic.

The switch S is controlled by a pulse-width modulator and is turned ON and OFF at the switching frequency $f_s = 1/T$ with a duty cycle D. In this paper, the Continuous Conduction Mode (CCM) of the converter is assumed; therefore, the current through the inductor L is always positive. Under this assumption, the main currents/voltages waveforms are shown in Figure 2.

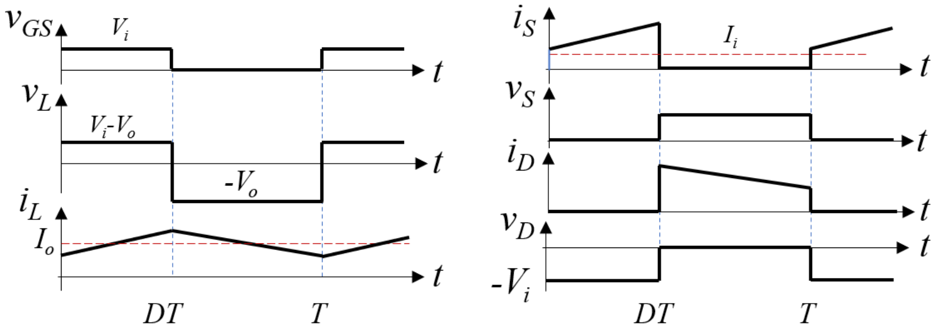


Figure 2. Current and voltage waveforms of the Buck converter.

The design procedure shown in [11] is used as a guide in this paper. First, the constraints are defined in Table 2.

According to [11], the minimum value of inductance that allows operation in CCM is

$$L_{\min} = \frac{R_{L\max}(1 - D_{\min})}{2 f_s} = \frac{50(1 - 0.1)}{2 \cdot 100 \cdot 10^3} \approx 225\ \mu\text{H}. \tag{1}$$

To ensure a higher margin, an inductance L = 650 μH has been chosen.

The voltage transfer function, taking into account the parasitics of the components, is

$$M_{VDC} = \frac{D}{1 + \frac{D r_{DS} + (1-D) R_F + r_L}{R_L} + \frac{(1-D) V_F}{V_O} + \frac{r_C R_L (1-D)^2}{12 f_s^2 L^2}} \approx 0.21 \tag{2}$$

where r_{DS}, R_F, r_L, and r_C are the parasitic resistances of the power MOSFET, the diode, the inductance, and the capacitance, respectively.

Table 2. System constraints.

Parameter	Value
Nominal Input Voltage V_i	5 V
Nominal Output Voltage V_o	1.1 V
Minimum Output Current $I_{o,min}$	0.1 A
Switching Frequency f_s	100 kHz
Maximum Output Voltage Ripple $\Delta_{Vor\%}$	5%

2.2. Simulink Validation

In order to check the validity of the design procedure presented in [11], the circuit shown in Figure 1 has been simulated in Matlab-Simulink, as represented in Figure 3.

The main current and voltage waveforms for the idea case and with parasitics are shown in Figure 4.

Through the Matlab-Simulink environment, the theoretical expected value of the main parameters calculated using [11] can be easily verified through simulations, as shown in Table 3. The effects of the parasitic components can be also studied.

Table 3. Comparison between theoretical and simulated parameters.

Parameter	Theoretical (Ideal)	Simulation (Ideal)	Percentage Difference	Theoretical (Parasitics)	Simulation (Parasitics)	Percentage Difference
M_v	0.220	0.225	2.2%	0.18	0.17	5.5%
Δi_L	23 mA	22 mA	4.3%	20 mA	19 mA	5%
Δv_o	4.1 mV	4.2 mV	2.4%	4.8 mv	4.9 mv	2%

As seen in Table 3, there is a perfect match between theoretical and simulation results. This matching can be easily verified by running the Simulink simulation available as Supplementary Materials.

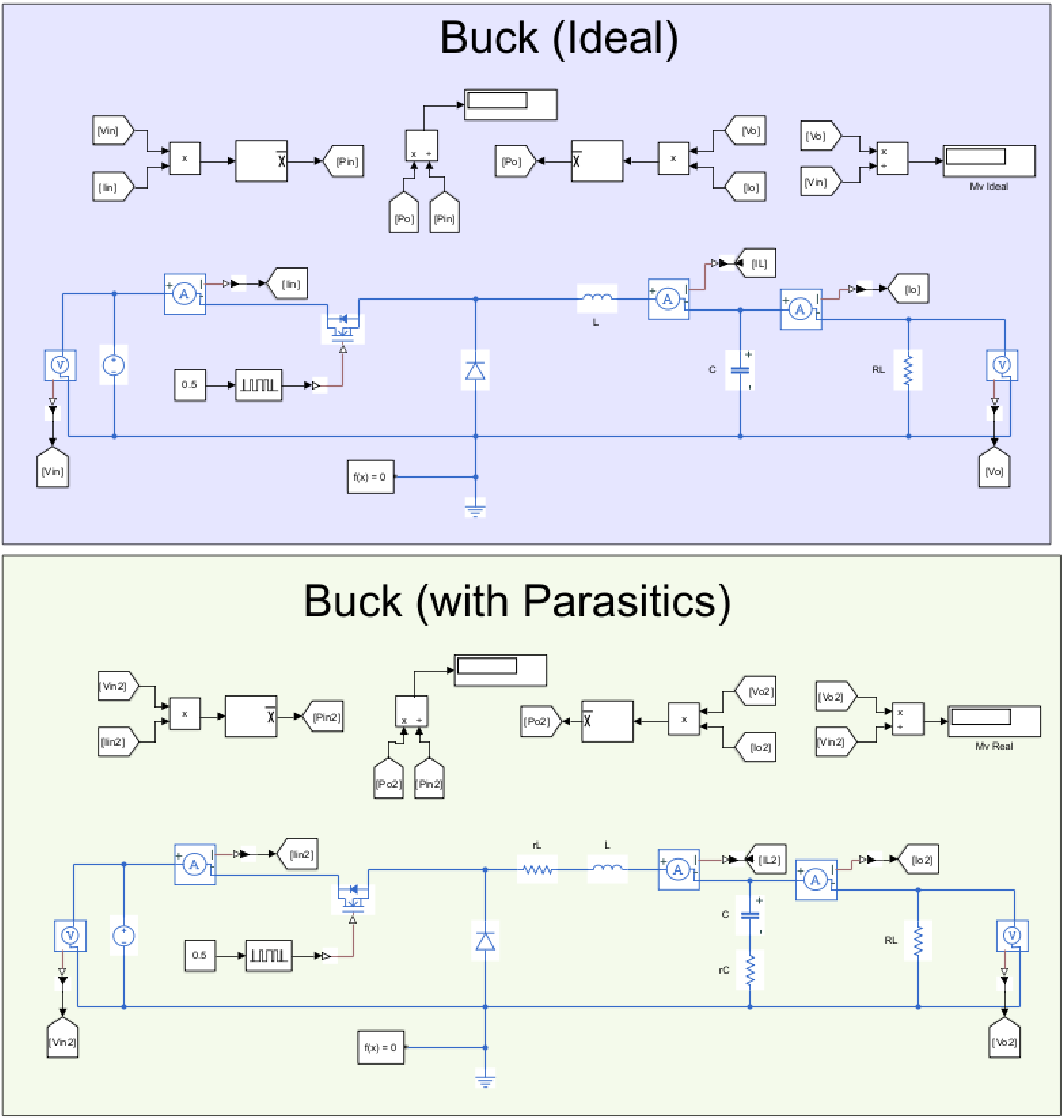


Figure 3. Schematic of Matlab-Simulink simulation.

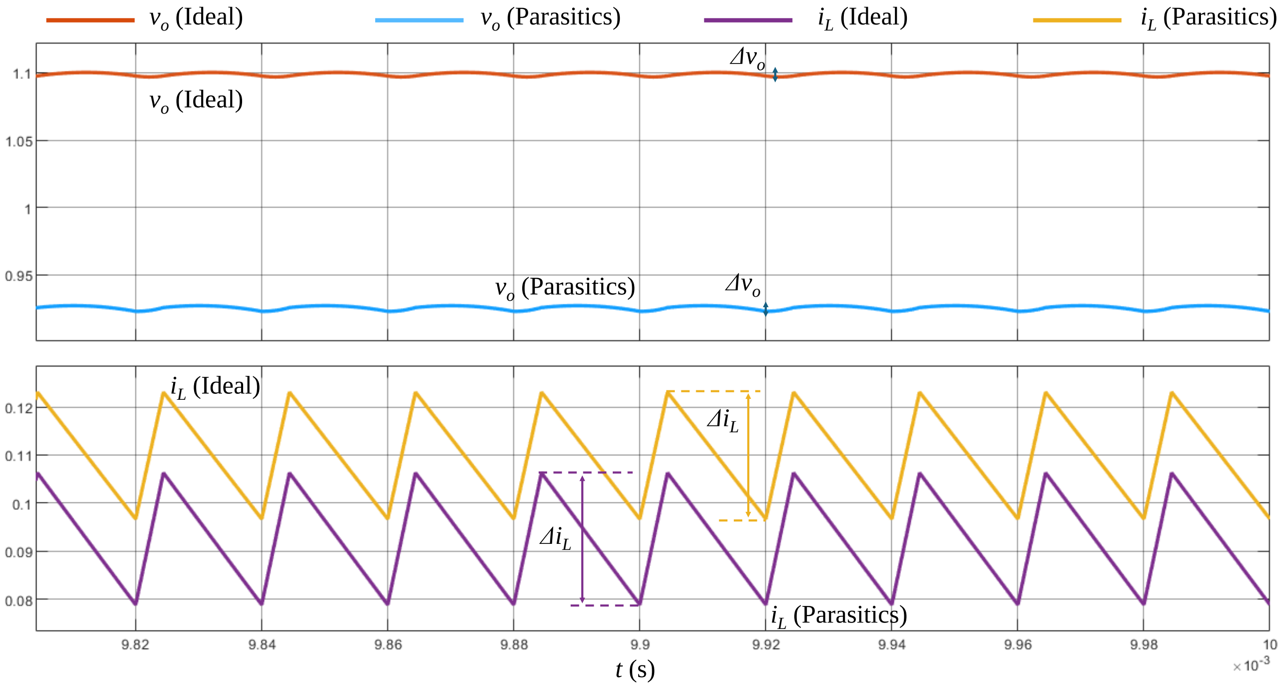


Figure 4. Simulink simulated output voltage v_o and inductor current i_L with and without parasitics.

3. Open Loop Small-Signal Buck Converter Transfer Function

Power converters exhibit highly nonlinear characteristics due to the incorporation of switches. These converters need control circuits to regulate the DC output voltage against fluctuations in operating conditions. Key control aspects, such as frequency response, transient response, and stability, are of significant interest. Although linear control theory offers well-established tools for investigating the dynamic performance of PWM converters, its application requires the averaging and linearization of the nonlinear power stages.

Two distinct methods for averaging PWM converters can be distinguished:

- The state-space averaging method. Power converters typically have different operating modes within a switching cycle (e.g., on and off states of switches). For each mode, a set of linear differential equations describes the system's behavior. This approach involves averaging these piecewise linear state equations over one switching period. This results in a single set of averaged state-space equations that approximate the converter's behavior over the entire period.
- The circuit-averaging method. By averaging the currents/voltages of the circuit elements and their interactions, you create an equivalent circuit that represents the average behavior of the power converter over a switching period. This equivalent circuit can be analyzed using standard circuit analysis techniques, providing insights into the converter's average behavior without needing to consider the detailed switching events.

In this paper, the state space modeling is chosen. This method works well for managing the intrinsic nonlinearities in power converters since it can accurately simulate the nonlinear switching behavior and then linearize around certain operating points for the examination of small signals. Through the state space matrices, this method systematically analyzes stability, transient reactions, and control designs, providing deeper insights into the system dynamics. Furthermore, state-space modeling is based on a strict mathematical framework that makes use of contemporary control theory instruments to improve the variety of analysis and design possibilities, including eigenvalue analysis, state feedback, and observer design.

3.1. Small Signal Modeling with the State Space Averaging Method

A state-space representation is a mathematical model of a physical system, capturing its internal state dynamics and outputs. The given system is described by two main equations: the state equation and the output equation. The state equation is

$$\dot{\mathbf{x}} = A\mathbf{x} + b\mathbf{u} \tag{3}$$

where $\dot{\mathbf{x}}$ represents the time derivative of the state variables, indicating how the state changes over time. The vector $\mathbf{x} = [x_1, x_2, ...x_n]$ represents the state variables of the system; A is the state matrix, which defines the linear relationships between the state variables; $\mathbf{u} = [u_1, u_2, ...u_m]$ represents the input to the system, which can be an external control or disturbance affecting the system; and b defines how the input $\mathbf{u}$ influences the state variables.In this case, the system has two inputs: the duty cycle d and the voltage generator v_i; consequently, $\mathbf{u} = [v_i, d]$.

The output equation is

$$\mathbf{y} = C\mathbf{x} \tag{4}$$

where $\mathbf{y} = [y_1, y_2, ...y_h]$ represents the outputs of the system, which are the quantities of interest that we can measure and C maps the state variables to the outputs, defining which states contribute to the output and how they are combined. In this case, since the aim is to regulate the output voltage, the output of the system is $y = v_o$.

Power converters are time-varying systems. In particular, the switching devices change state periodically between two configurations (e.g., ON and OFF). Therefore, two topologies can be identified depending on the switching status, leading to a change in the matrices A and C, and the vector b. Therefore, the average state space representation can be defined as

$$A = DA_{ON} + (1 - D)A_{OFF} \tag{5}$$

$$b = Db_{ON} + (1 - D)b_{OFF} \tag{6}$$

$$c = DC_{ON} + (1 - D)C_{OFF} \tag{7}$$

where D is the duty cycle, A_{ON}, b_{ON} and C_{ON} are the matrices during the ON state, while A_{OFF}, b_{OFF}, and C_{OFF} are the matrices that represent the topology during the OFF state. Once the averaging model has been identified, the small signal model can be obtained by injecting a small perturbation at the input $\hat{u}$ and analyzing the effect on the state $\hat{x}$ and to the output $\hat{y}$. Each variable can be seen as a sum of an average value and a perturbation term, where the caret denotes a slight perturbation and the capitalized quantities are the steady state values as mentioned below [12]:

$$\mathbf{x} = \mathbf{X} + \hat{\mathbf{x}} \tag{8}$$

$$y = Y + \hat{y} \tag{9}$$

$$\mathbf{u} = \mathbf{U} + \hat{\mathbf{u}} \tag{10}$$

It must be noted that the input is a vector $\mathbf{u} = [v_i, d]$. Therefore $v_i = V_i + \hat{v}_i$ and $d = D + \hat{d}$.

Using (8) in (3)–(4), the final state-space averaged equations are produced by expanding the result and eliminating the non-linear (second-order) terms. The average model and the small signal model can be divided. The average model can be obtained, assuming it operates in steady-state conditions, e.g., $\dot{x} = 0$:

$$0 = A\mathbf{X} + b\mathbf{U} \tag{11}$$

$$Y = C\mathbf{X} \tag{12}$$

On the other hand, the small signal model that relates the variation of the output voltage with respect to the duty cycle can be calculated by setting the input voltage variation to zero, $\hat{v}_i = 0$, and assuming it operates at $v_i = V_i$ and $d = D + \hat{d}$. Under this condition, the small signal model can be calculated as

$$\hat{\mathbf{x}} = A\hat{\mathbf{x}} + b\hat{\mathbf{u}} + [(A_{ON} - A_{OFF})\mathbf{X} + (b_{ON} - b_{OFF})V_i]\hat{d} \tag{13}$$

$$\hat{\mathbf{y}} = c\hat{\mathbf{x}} + [(c_{ON} - c_{OFF})\mathbf{X}\hat{d} \tag{14}$$

where $\hat{d}$ is the input of the system that in this case is the perturbation of the duty cycle. By setting $\hat{u} = 0$ in (13) gives the response of the duty cycle variation. Transforming the equation in the Laplace domain, the state-to-transfer function can be calculated as

$$sx(s) = Ax(s) + [(A_{ON} - (A_{OFF})X + (b_{ON} - b_{OFF})V_i]d(s) \tag{15}$$

$$\frac{x(s)}{d(s)} = (sI - A)^{-1}[(A_{ON} - A_{OFF})X + (b_{ON} - b_{OFF})V_i] \tag{16}$$

The output function can always be used to find the transfer function that is being sought if it is not one of the state variables:

$$\hat{y} = c\hat{x} + [(c_{ON} - c_{OFF})X\hat{d} \tag{17}$$

$$y(s) = cx(s) + [(c_{ON} - c_{OFF})Xd(s) \tag{18}$$

$$\frac{y(s)}{d(s)} = c\frac{x(s)}{d(s)} + [(c_{ON} - c_{OFF})X \tag{19}$$

3.2. Small Signal Modeling of the Buck Converter with the State Space Averaging Method

The Buck converter topology is shown in Figure 5.

When the power MOSFET S is ON and the diode D is OFF, the equations are

$$\frac{di_L}{dt} = \frac{V_i - r_L i_L - V_o}{L} \tag{20}$$

$$\frac{dV_{c_o}}{dt} = \frac{V_o - V_{C_o}}{C_o r_{c_o}} \tag{21}$$

$$V_D = V_d c - i_L r_s \tag{22}$$

$$V_b = \frac{R_L(V_{C_o} + i_L r_{C_o})}{R_L + r_{C_o}} \tag{23}$$

when the power MOSFET S is OFF and the diode D is ON, the equations are

$$\frac{di_L}{dt} = \frac{V_i - r_L i_L - V_o}{L} \tag{24}$$

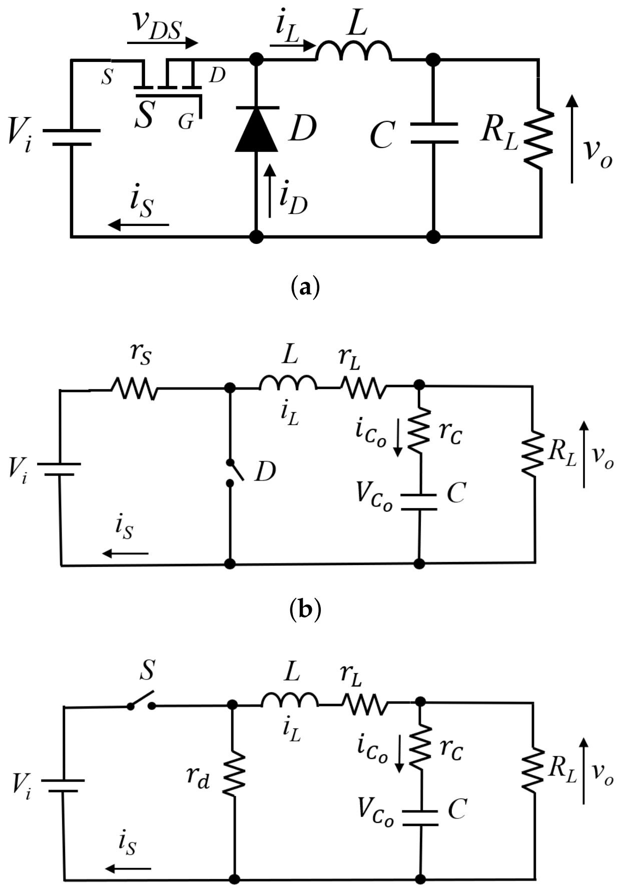
$$\frac{dV_{c_o}}{dt} = \frac{V_o - V_{C_o}}{C_o r_{c_o}} \tag{25}$$

$$V_i = -i_L r_D \tag{26}$$

$$V_o = \frac{R_L(V_{C_o} + i_L r_{C_o})}{R_L + r_{C_o}} \tag{27}$$

$$A_{ON} = \begin{bmatrix} \frac{-r_D + r_L + \frac{R_L r_{C_O}}{R_L r_{C_O}}}{L} & \frac{-R_L}{L(R_L r_{C_o})} \\ \frac{R_L}{C_o(R_L r_{C_o})} & \frac{\frac{R_L}{R_L r_{C_O}} - 1}{C_o r_{C_O}} \end{bmatrix} \quad b_{ON} = \begin{bmatrix} \frac{1}{L} & \frac{-r_D + r_L + \frac{R_L r_{C_O}}{R_L r_{C_O}}}{L} \\ 0 & \frac{R_L}{C_o(R_L r_{C_o})} \end{bmatrix}$$

$$A_{OFF} = \begin{bmatrix} \frac{-r_D + r_L + \frac{R_L r_{C_o}}{R_L r_{C_o}}}{L} & \frac{-R_L}{L(R_L r_{C_o})} \\ \frac{R_L}{C_o(R_L r_{C_o})} & \frac{\frac{R_L}{R_L r_{C_o}} - 1}{C_o r_{C_o}} \end{bmatrix} \quad b_{OFF} = \begin{bmatrix} 0 & \frac{-r_D + r_L + \frac{R_L r_{C_o}}{R_L r_{C_o}}}{L} \\ 0 & \frac{R_L}{C_o(R_L r_{C_o})} \end{bmatrix}$$



(**a**)

(**b**)

(**c**)

Figure 5. Buck converter with all parasitic resistances: (**a**) circuit schematic; (**b**) when the MOSFET is ON and the diode is OFF; (**c**) when the MOSFET is OFF and the diode is ON.

3.3. Simulation Validation through PLECS

To validate the analytical transfer function obtained by the small-signal model, a PLECS simulation has been realized, as shown in Figure 6. The variables involved in the calculation of this transfer function are shown in Table 4.

Table 4. Transfer function parameters considered.

Parameter	Value
V_i	5 V
V_o	1.1 V
L	650 µH
C	20 µF
r_C	5 mΩ
r_L	50 mΩ
R_L	1 Ω
r_{DS}	0.1 Ω
r_F	0.1 Ω

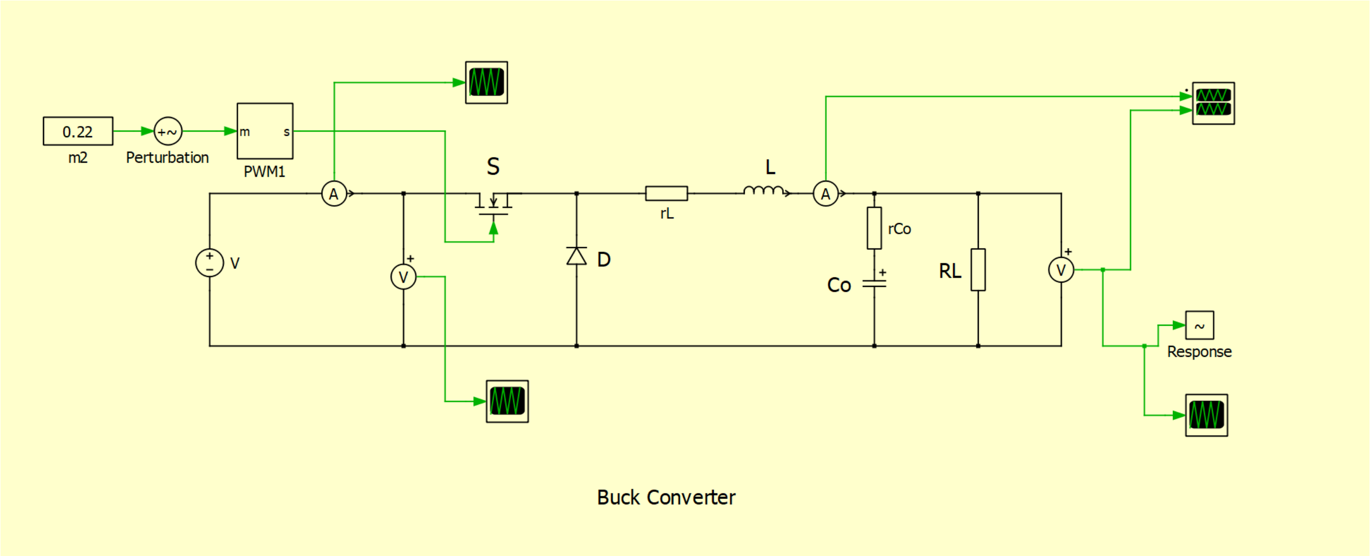


Figure 6. PLECS model of the Buck converter.

As can be seen, a disturbance was introduced on the duty cycle D, and the response on the output voltage v_o was measured. In Figure 7, the comparison of the frequency response between the analytical transfer function is shown. The two models are in good agreement, confirming the validity of the derived transfer function.

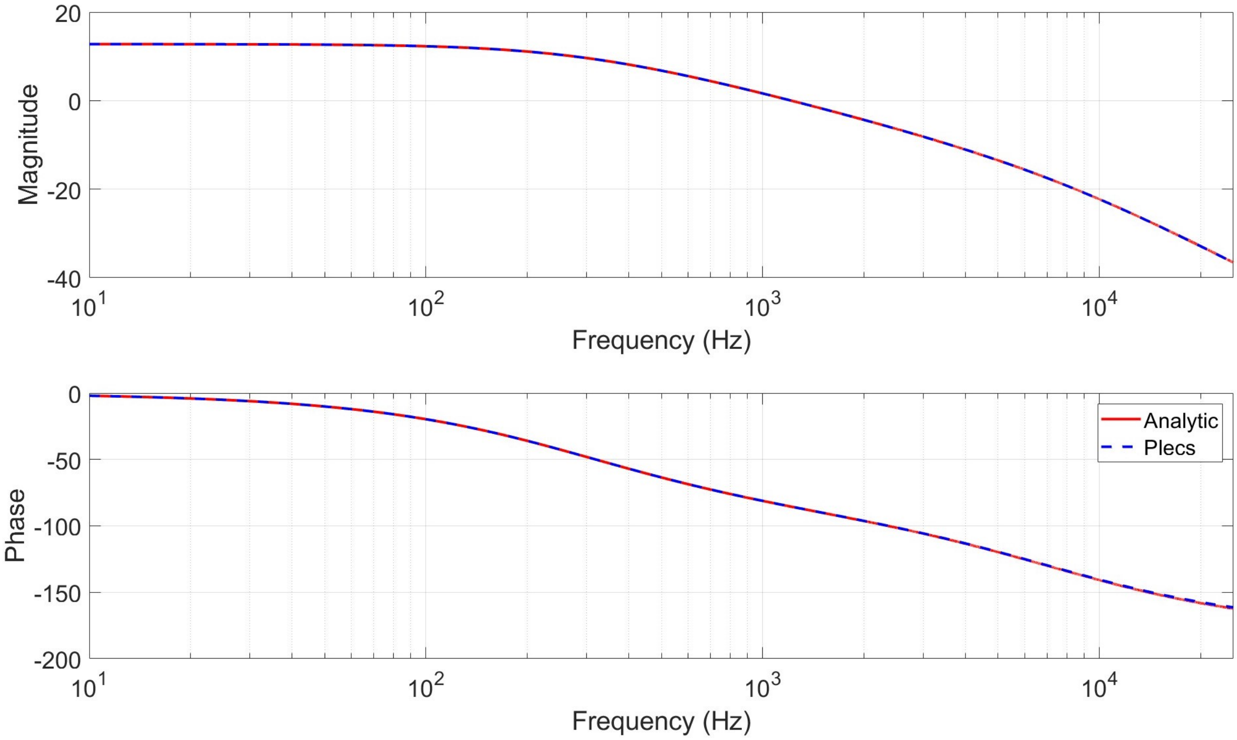


Figure 7. Comparative frequency response performance through a Bode plot representation.

The Bode and the Nyquist diagrams of the control transfer function T_p using the components listed in Table 4 are shown in Figures 8 and 9.

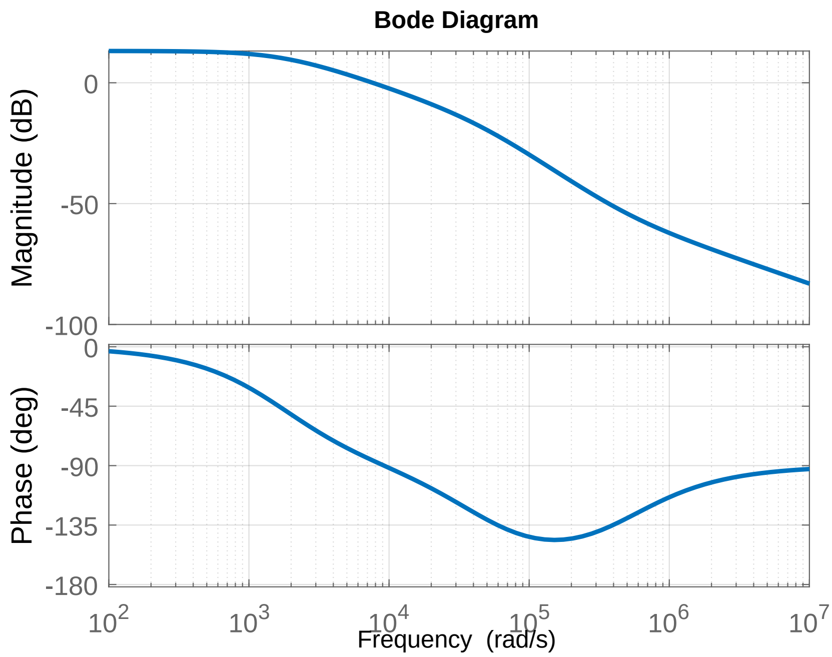


Figure 8. Bode plot representation of the transfer function T_p.

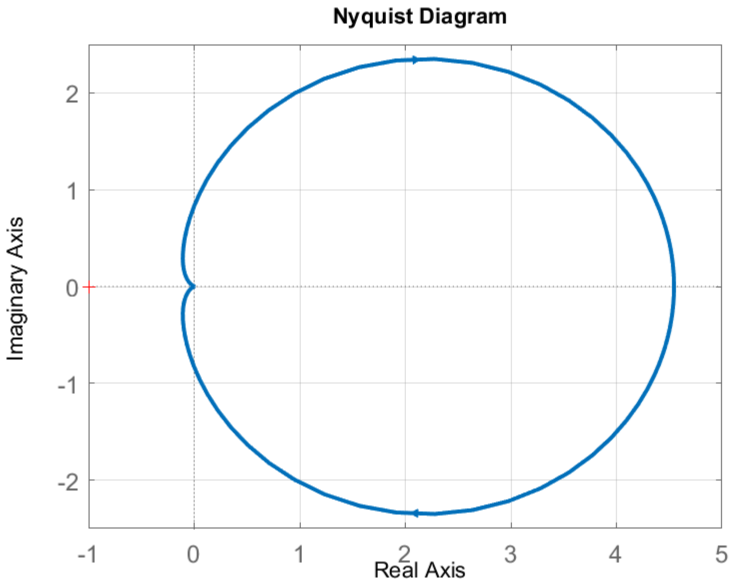


Figure 9. Nyquist diagram representation of the transfer function T_p.

4. Closed-Loop Control Design

The duty cycle of the converter can be manually adjusted to achieve the desired battery current. However, any difference between the mathematical model and the Buck converter will cause this manual tuning of D to introduce a deviation between the obtained and the target i_L. Moreover, any disturbance acting on the system cannot be compensated. To solve this problem, the load current can be fed back into the controller, and a proportional-integral regulator (PI) can be implemented in what is known as closed-loop operation. The implementation of a control loop allows this tracking error to be reduced to zero and gives the system the ability to compensate for disturbances that may affect it.

It should be mentioned that in this specific application, different input and output variables will be considered to perform the control tasks. As inputs, the measured output current and the duty cycle (D) will be used. Then, the system output will be the output voltage found in R_L.

4.1. Continuous Time Controller Design

The design of the controller in the continuous time domain is performed following the recommendations made in [13].

The first step is to obtain the transfer function of the Buck converter, which has been computed in Section III as Equation (16), obtaining

$$P(s) = \frac{I_L(s)}{D(s)} = \frac{(sCr_C + 1)V_i^0}{s^2CL(r_C + R_L) + s[C(r_Lr_C + r_LR_L + r_CR_L) + L] + (R_L + r_L)} = \frac{b_1s + b_0}{a_2s^2 + a_1s + a_0} \tag{28}$$

Furthermore, in order to calculate the phase margin and the correct performance of the PI controller, the following transfer function should be considered:

$$G_{PI}(s) = K_p + \frac{K_i}{s} \tag{29}$$

Then, it is necessary to set the desired value of the phase margin. For our system, the phase margin is required to be at least 75°. Once the phase margin value is established, it is necessary to calculate the region where every possible combination between K_p and K_i gives the desired value of phase margin. To calculate the region, two equations related to K_p and K_i are considered:

$$K_p = \frac{-a_2b_1sin(\phi\omega^3) + (-a_2b_0 + a_1b_1)cos(\phi\omega^2) + (-a_1b_0 + a_0b_1)sin(\phi\omega) + a_0b_0cos(\phi)}{-(b_1^2\omega^2 + b_0^2)} \tag{30}$$

$$K_i = \frac{\omega(-a_2b_1cos(\phi\omega^3) + (a_2b_0 - a_1b_1)sin(\phi\omega^2) + (a_0b_1 - a_1b_0)cos(\phi\omega) - a_0b_0sin(\phi))}{-(b_1^2\omega^2 + b_0^2)} \tag{31}$$

From Equations (30) and (31), a relation between both variables can be obtained. This relation is represented in Figure 10 to visually understand the correlation between both variables. In this figure, it can be observed that the drawn curve represents all the points that result in a 75° phase margin. All the values under this curve have a phase margin higher than 75°.

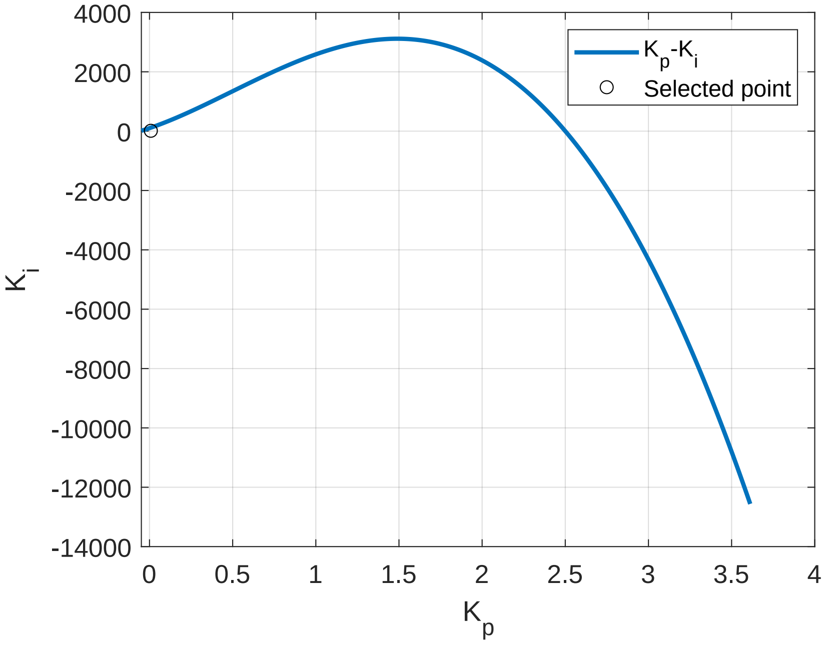


Figure 10. K_p vs. K_i representation.

Considering this restriction, the next step is to choose the optimal values of K_p and K_i, the application of the toolbox "Control System Designer" in Matlab. Equations (30) and (31) provide the sets of proportional and integral gains that satisfy the phase margin requirement, giving a very large number of combinations. By using the *Control System Designer* tool, the transient response of the equivalent closed-loop system can be evaluated, and, therefore, the integral and proportional gains can be tuned to meet transient specifications. In the case of this work, the specifications applied during the design process are rise time (0–80%) = 50 ms and settling time (3%) = 100 ms, without overshoot.

The selected values for K_p and K_i are 0.008 and 12.24, respectively. The result of the application of these controller parameters can be observed in Figure 11.

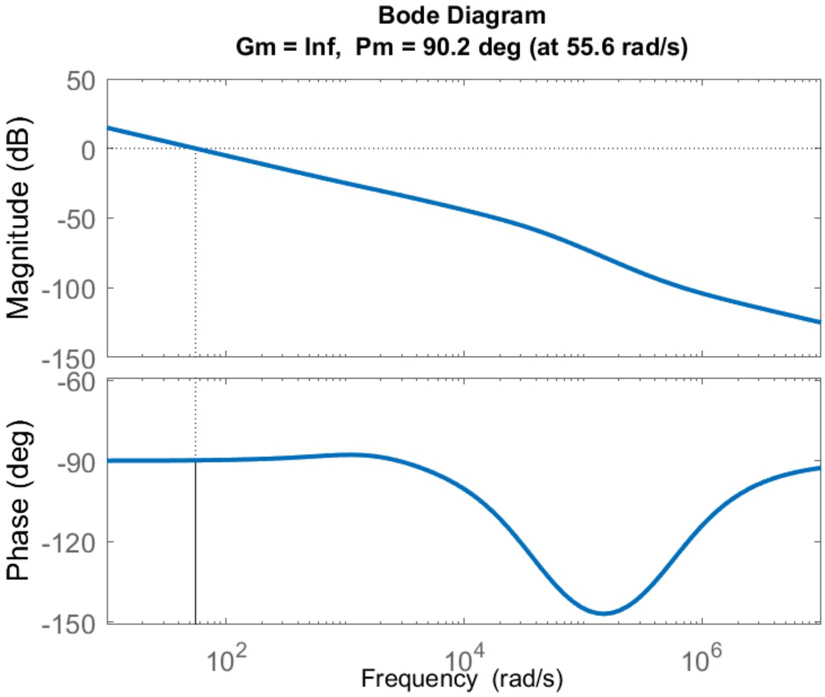


Figure 11. Continuous transfer function of the Buck converter controller.

4.2. Control Discretization and Closed-Loop Performance Evaluation

The discretization of a PI controller is the process of converting a continuous control system into a discrete control system, i.e., a system that operates at discrete time intervals rather than in continuous time. This process is necessary when the physical implementation of the controller is required, as digital processors work with information in discrete form, processing data in finite time intervals.

The Tustin transformation, also known as the bilinear transformation, is a widely used method for discretizing continuous-time transfer functions in the field of control systems. This method is based on the bilinear mapping of the Laplace domain to the Z-domain, where the s-plane is mapped onto the unit circle in the Z-plane. This method preserves the stability of the system and provides a linear mapping between the continuous and discrete domains.

This method has been chosen to perform the discretization of the proposed Buck converter controller. The requirements to apply the Tustin transformation are the continuous-time transfer function, obtained in Section 4.1; the sampling period, set at 10 kHz; and the bilinear transformation formula, which can be seen in (32).

$$z = e^{sT_s} \approx \frac{1 + \frac{sT_s}{2}}{1 - \frac{sT_s}{2}} \tag{32}$$

The application of this method results in the Bode representation shown in Figure 12.

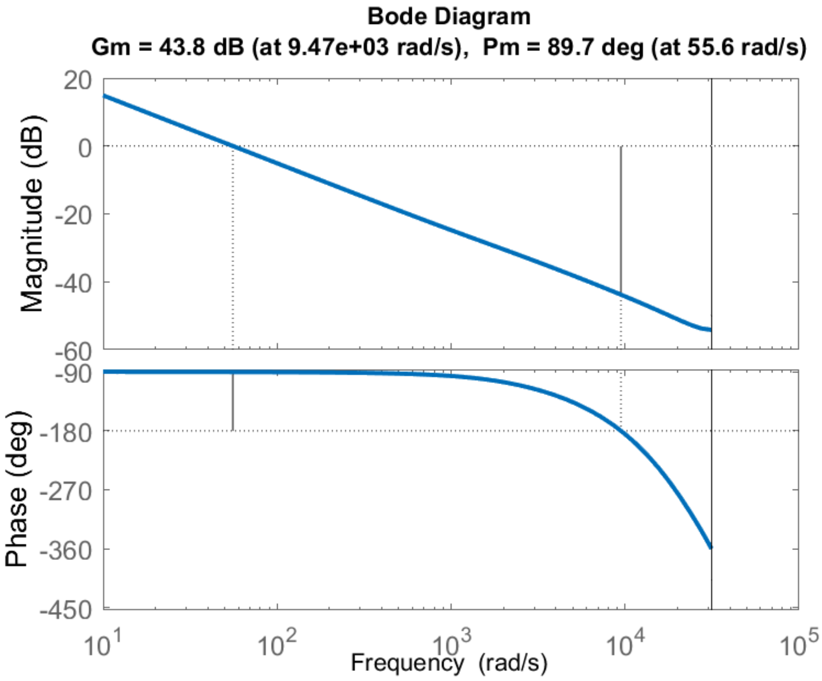


Figure 12. Discrete transfer function of Buck converter controller.

As can be deduced from Figure 13, the differences between the continuous time domain controller and Tustin discrete approximation are not significative.

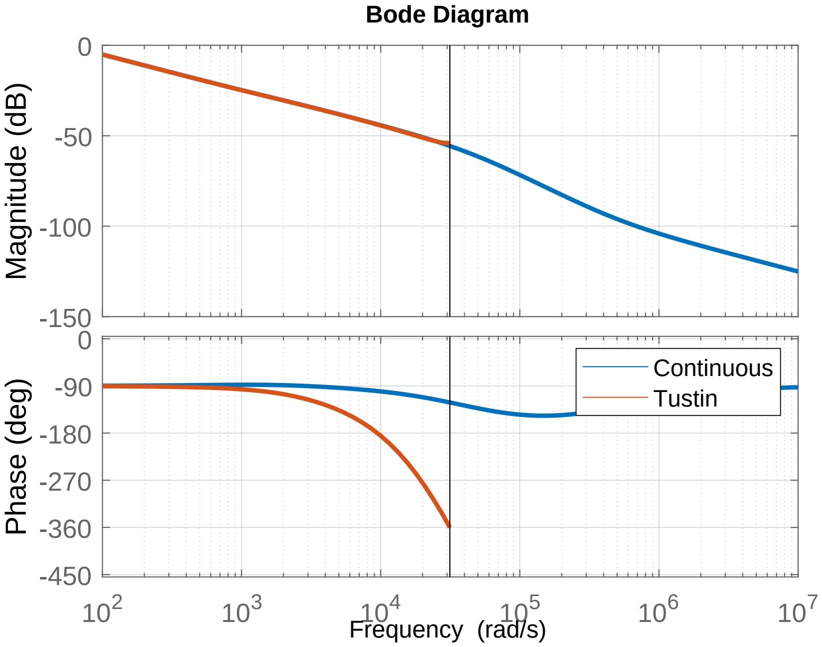


Figure 13. Comparison between continuous and discrete time domain controllers.

5. Step-by-Step Implementation of the Control System in the STM32 Environment

The control loop is implemented on a NUCLEO H7A3ZI-Q STM32 board. This NUCLEO board is based on an STM32H7A3ZI microcontroller, which operates at a clock frequency of 280 MHz. It features 2 MB of flash memory, 1.4 MB of RAM, two 16-bit digital-to-analog converters (ADCs), direct memory access (DMA), and two advanced control timers (ACT). This section aims to summarize the main steps required for the successful implementation of the designed regulator in STM32CubeIDE using the Hardware Abstraction Layer (HAL).

5.1. PWM Generation

The generation of the gate signals to drive the synchronous Buck converter is accomplished using one of the two ACTs. On NUCLEO H7A3ZI-Q STM32 board, the ACTs are Timer 1 (TIM1) and Timer 8 (TIM8). Since both timers operate at the same clock frequency ($f_{TIM} = 280$ MHz) and the PWM signal generation and interaction with the ADC are analogous, its selection does not significantly impact the implementation. In this work, channel 1 of TIM1 has been chosen, but another channel or TIM8 could also be used if desired.

The initial step involves configuring the ACT's behavior. To achieve this, it is necessary to define the *Counter Mode* as *Center Align Mode 1* and specify the *Counter Period* parameter in terms of clock cycles. The value of the counter period is stored in the TIMX_ARR register (TIM1_ARR for timer 1). In this mode, the counter starts from zero in the upward direction until it reaches TIM1_ARR. Subsequently, the counter value decreases until it reaches one, resets, starts counting from zero again, having as a result the behavior illustrated in Figure 14. To determine the appropriate value for TIM1_ARR, the following calculation is necessary.

$$\textbf{TIM1\_ARR} = \frac{f_{TIM}}{2 f_{sw}} \tag{33}$$

To generate the PWM signal, TIM1 will compare the state of the counter with the number of clock ticks stored on the TIMx_CRRx register. When using channel 1 of TIM1, this register is referred to as TIM1_CRR1. Its initial value can be configured through the *Pulse* option in the STM32CubeIDE timer configuration and can be modified during the execution of the code by calling the __HAL_TIM_SET_COMPARE function. The value of the TIM1_CRR1 register depends on the duty cycle as follows:

$$TIM1\_CRR1 = D \cdot TIM1\_ARR \tag{34}$$

and, by comparing the counter output with this value, the PWM signals for both transistors are generated as shown in Figure 14.

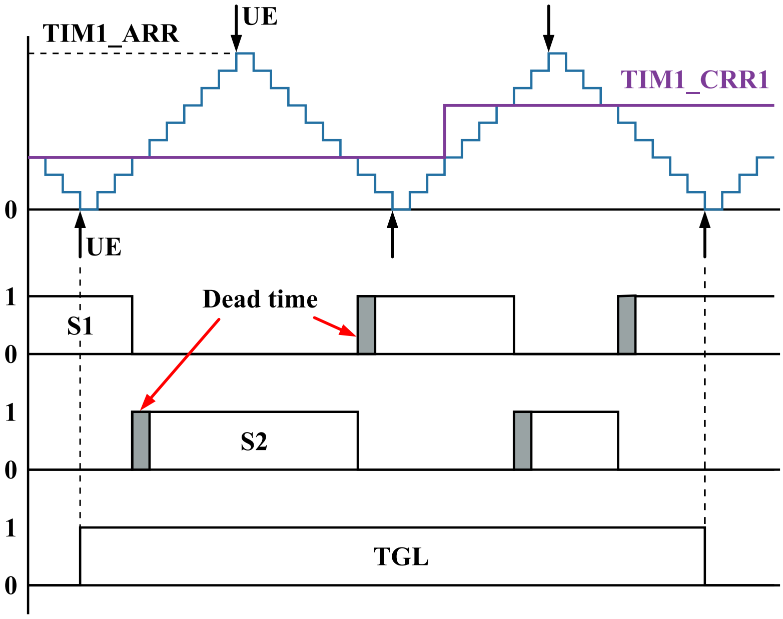
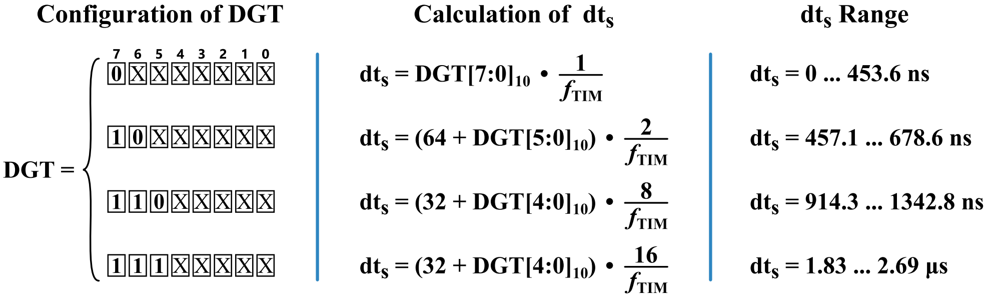


Figure 14. PWM signal generation for TIM1_ARR = 10, $N_{UE} = 4$ and a dead-time of 1 clock tick.

Finally, since two transistors are employed in the Buck converter, it is essential to implement dead time in the PWM signal generation. The switching of a real transistor does not occur immediately. It needs a certain amount of time to transition between ON and OFF states. In a Buck converter, the PWM signal of transistor S_2 is the complementary signal of S_1. When applied to real transistors, there is a moment during transitions when both transistors are in their ON state, potentially short-circuiting the converter. To prevent this, TIM1 and TIM8 can be programmed to introduce a dead time in the PWM signal generation. This delay ensures that the currently active transistor reaches the OFF state before the other starts its transition. As a result, the PWM signals depicted in Figure 14 are obtained. To generate the two complementary PWM signals, Channel 1 must be configured by selecting the *PWM Generation CH1 CH1N* option in the configuration. Subsequently, the desired deadtime can be configured using the *Dead Time* option.

The *Dead Time* option stores a value between 0 and 255 in the eight least significant bits of the *TIMx break and dead-time register* TIMx_BDTR register (TIM1_BDTR for TIM1). This eight-bit section is referred to as DGT, and its configuration allows programming deadtime in four different ranges. The configuration of the DGT bits for $f_{TIM} = 280$ MHz is illustrated in Figure 15. In this context, $\text{DGT}[X : Y]_{10}$ indicates the decimal representation of the bits X to Y within the DGT. If a dead-time of 500 ns is desired, the two most significant bits of DGT must be set to 10, while the six least significant bits must fulfill $dt_s = (64 + \text{DGT}[5 : 0]_{10}) \cdot 2/f_{TIM} = 500$ ns. This condition is met for $\text{DGT}[5 : 0]_{10} = 6$. Consequently, the complete bitstream for DGT is 10000110 in binary form or 134 in decimal form.



Configuration of DGT	Calculation of dt_s	dt_s Range
DGT = 7 6 5 4 3 2 1 0 0 X X X X X X X	$dt_s = \text{DGT}[7{:}0]_{10} \bullet \frac{1}{f_{TIM}}$	dt_s = 0 ... 453.6 ns
1 0 X X X X X X	$dt_s = (64 + \text{DGT}[5{:}0]_{10}) \bullet \frac{2}{f_{TIM}}$	dt_s = 457.1 ... 678.6 ns
1 1 0 X X X X X	$dt_s = (32 + \text{DGT}[4{:}0]_{10}) \bullet \frac{8}{f_{TIM}}$	dt_s = 914.3 ... 1342.8 ns
1 1 1 X X X X X	$dt_s = (32 + \text{DGT}[4{:}0]_{10}) \bullet \frac{16}{f_{TIM}}$	dt_s = 1.83 ... 2.69 µs

Figure 15. Configuration of the DGT bits in the TIMx_BDTR register for $f_{TIM} = 280$ MHz.

5.2. ADC Synchronization

For effective implementation of the digital controller, maintaining a stable sample rate is crucial. The simplest method to trigger ADC conversions is through the use of polling techniques, where the microprocessor manages the process. However, due to variations in code execution from one iteration to another, the execution time can differ. Consequently, the timing of the ADC conversion triggering becomes imprecise. Moreover, when transistors switch between ON and OFF states, an overshoot occurs. If the ADC conversion coincides with transistor switching, this undesired overshoot may affect the measurement. To mitigate this issue, the ADC conversion should be triggered as far as possible from these transitions.

To achieve this, TIM1 uses a trigger line (TGL) to drive the ADC conversions. The ADC monitors changes in this TGL, and when a falling or rising edge is detected, it triggers the ADC conversion. The transitions of the trigger line are controlled by an *Update Event* (UE) that occurs each time the counter state reaches zero or TIM_ARR-1. If this event directly drives the ADC through the TGL, f_{sw} will always be equal to f_s. To allow $f_{sw} \neq f_s$, a *Repetition Counter* can be used to drive the TGL. The *Repetition Counter* counts the number of times that the *Update Event* occurs and, when it reaches the desired number of repetitions, N_{UE}, the state of the TGL changes. This option is managed by the TIMx_RCR register (TIM1_RCR for TIM1), whose value is calculated as

$$TIM1\_RCR = \cdot N_{UE} - 1 \tag{35}$$

Since an *Update Event* occurs every half period, the sampling frequencies that can be selected fulfill

$$f_s = N_{UE} \cdot f_{sw} \tag{36}$$

5.3. Direct Memory Access

After completing the ADC conversion, the measured value is stored in the ADC registers. Then, the microprocessor can be directly used for copying the value of this register to memory until it is needed. For this purpose, the ADC can generate an interruption that will indicate to the microprocessor when the measurement is ready. Then, the execution of the current subroutine in the microprocessor halts in order to handle the interruption, and, after managing this operation, the execution of the code resumes (see Figure 16a). As will be shown later, this strategy would be appropriate for the scope of the work proposed in this paper. However, the implementation of the DMA is proposed to fully understand its potential for reducing computational load.

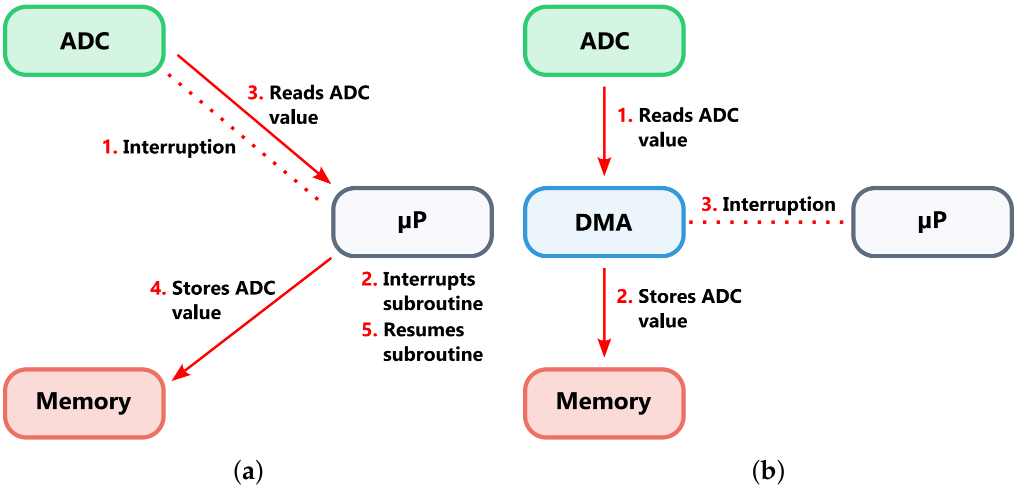


Figure 16. Storage of the measured ADC value in the memory: (**a**) by using ADC interrupt, (**b**) by using the DMA.

The DMA can prevent interruptions in the the subroutine execution when handling such a basic operation (see Figure 16b). It directly copies the values from the ADC registers into memory, bypassing the microprocessor and, thus, reducing its computational load. Once this task is completed, the DMA generates an interruption that can be used to trigger specific subroutines in the microprocessor.

To enable DMA in STM32CubeIDE, we configure *Conversion Data Management Mode* option in the ADC settings to *DMA Circular Mode*. It will make the DMA wait until the measured value is stored on the ADC register. Subsequently, it copies the register values to the specified memory locations and generates an interruption after completing this operation. If the sample rate is very high, the DMA may not have sufficient time to copy all the values before another ADC conversion is triggered. In such cases, new values will overwrite the existing ones in the ADC registers, resulting in the loss of the unsaved measurements.

5.4. Code Description

After configuring the different features of the NUCLEO STM32 board to be used in this development, the description of the code is addressed. A summary of its functionality is depicted in Figure 17. The initial step involves initializing the timer, which serves both for PWM generation and ADC synchronization. Following this initialization, the code enters an infinite “while” loop, as its exit condition will never be met.

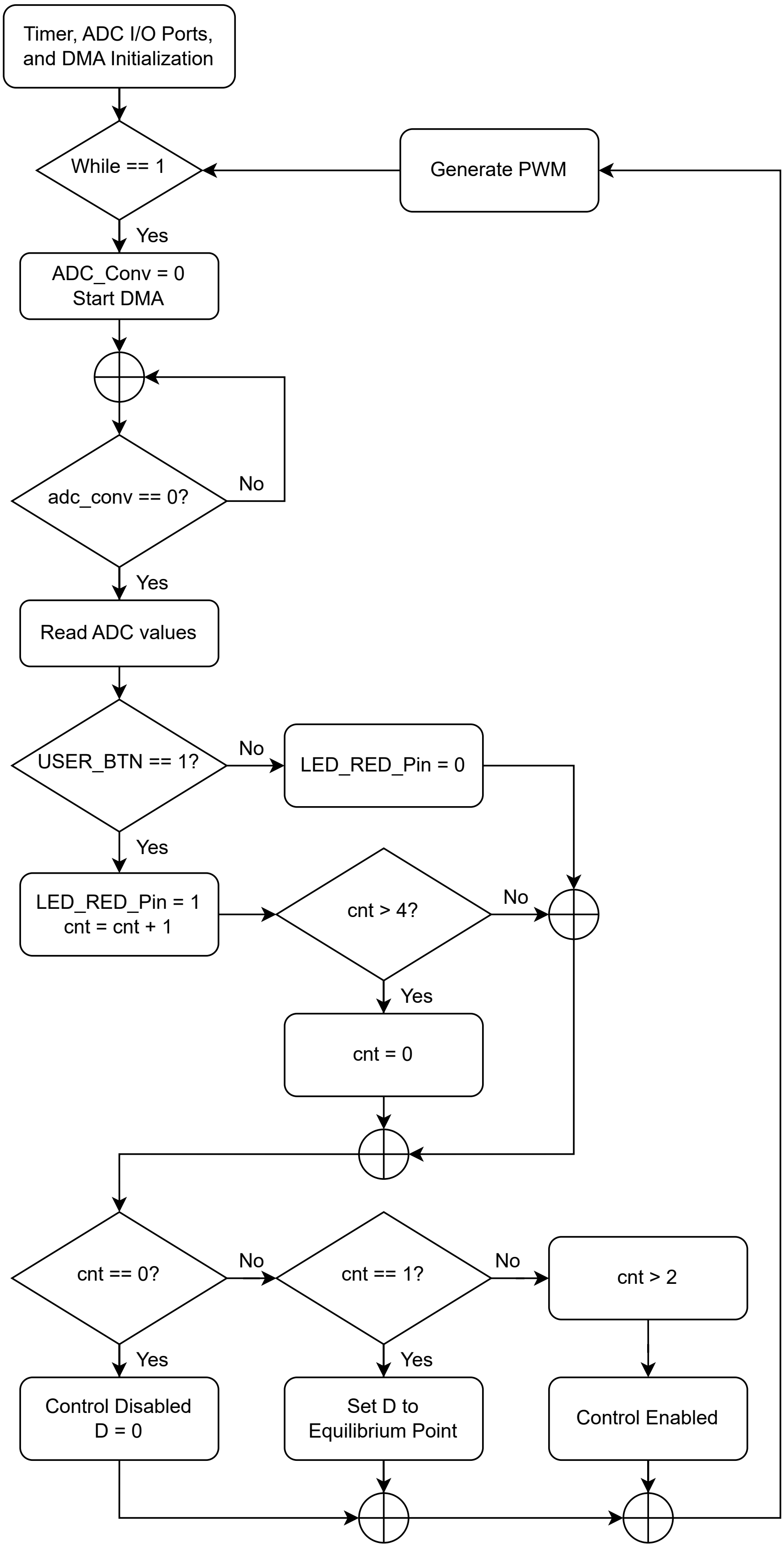


Figure 17. Simplified flowchart of the programmed code.

Inside the while loop, the first action consists of starting the DMA and waiting until the transfer of the measurement from the ADC to the memory is completed. For this purpose, the *ADC_Conv* variable is employed. At the beginning of the while loop, its value is set to zero and, while handling the DMA-triggered interrupt, is changed to one. Its value is set to zero at the start of the "while" loop and changes to 1 while handling the interruption triggered by the DMA. The main routine waits until this variable is set to 1 by using a "while" loop where additional code can be executed if desired. Once the ADC measurement is ready, its value is read, and then the state of a digital pin attached to the blue user button included in the NUCLEO board is checked.

When the pulsation of this button is detected, the red LED included in the board is switched on and the variable *cnt*, used as a counter, increases its value to 1. Otherwise, the LED will be set to its OFF state. If the value of this variable is higher than 4, its value resets to 0. After this, depending on the state of the variable *cnt*, three different operating modes are programmed.

When *cnt = 0*, the discrete-time controller is disabled, and the duty cycle is set to 0. This puts the transistor S_2 in the ON state during all the switching periods, resulting in a 0 A current applied to the load. The system enters the second operating mode when *cnt* = 1. In this mode, the duty cycle is set to its equilibrium point value, D^0. This configuration achieves open-loop operation with a constant D and a load current applied I^0.

The current controller will be enabled when the value of the variable *cnt* is between 2 and 4. The implementation of the digital controller begins with the calculation of the tracking error $e(z)$ of the reference signal, $I_{ref}(z)$. For a given time instant *k*, this calculation can be implemented as the following subtraction:

$$e(k) = I_{ref}(k) - I(k) \tag{37}$$

After this, the digital regulator is implemented by employing the definition of its transfer function:

$$R(z) = \frac{\Delta D(z)}{e(z)} K_p + K_I \frac{1}{z-1} = \frac{K_p z + (K_P - K_I)}{z-1} \tag{38}$$

which, by implementing the following expression, gives the $\Delta D(z)$ that must be applied in the time instant *k*:

$$\Delta D(k) = e(k)K_P + e(k-1)(K_I - K_P) + \Delta D(k-1) \tag{39}$$

and, finally, the duty cycle that must be applied to the Buck converter is

$$D(k) = D^0 + \Delta D(k) \tag{40}$$

Once the duty cycle is calculated, its value is updated in the PWM generation, and another iteration of the code is programmed inside the infinite while the loop starts.

6. Experimental Setup

The experimental setup is composed of the KIT-CRD-3DD12P Buck-Boost Evaluation Kit from CREE, the oscilloscope RTM3004, and the current probe RT-ZC15B by Rohde and Schwarz, the NUCLEO H7A3ZI-Q STM32 board, and the demo board EVCS1803-S-05-00A from Monolithic Power Systems. Figure 18 illustrates the schematic of the experimental setup, which includes a trigger signal (TRG) attached to the red LED that is activated when the USER BTN button is pressed.

The EVCS1803-S-05-00A board integrates an MCS1803GS-05 Hall-effect current sensor and provides a voltage proportional to the sensed current that follows the expression:

$$V_{sen} = \frac{V_{dd}}{2} + k_{sen} I_{sen} \tag{41}$$

where V_{dd} is the supply voltage applied to the sensor, I_{sen} is the current flowing to the primary side of the current sensor, and k_p is its gain in V/A units.

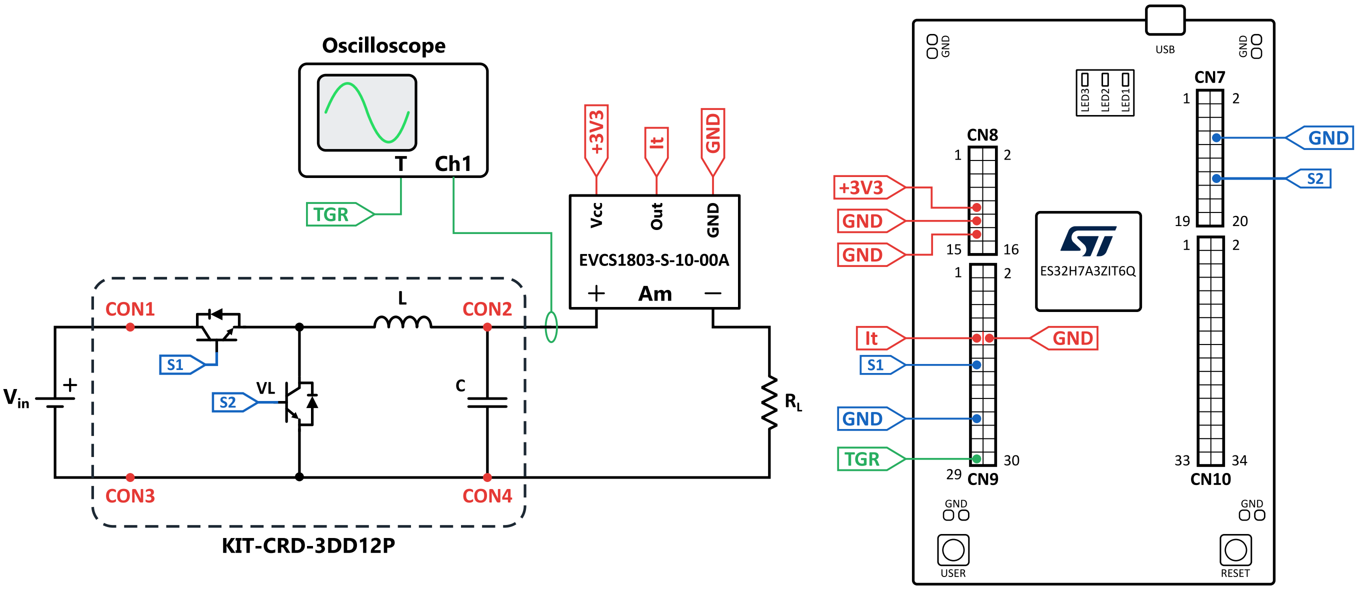


Figure 18. Connection schematic of the proposed setup.

The nominal supply voltage of the EVCS1803-S-05-00A board is 5 V, but a 3.3 V supply is used with the STM32 board so that the voltage V_{sen} is within the 0–3.3 V input range of the STM32 ADC. Therefore, a calibration of the sensor has been made by applying different known currents measured with the Rohde & Schwarz oscilloscope and measuring V_{sen}. As a result, a $k_{sen} = 0.25285249$ V/A has been measured.

The KIT-CRD-3DD12P Buck-Boost Evaluation Kit has been set up in the Buck configuration, with a DC input voltage of 5 V, and the nominal values of the different components of the Buck converter are summarized in Table 5. The switching and sampling frequencies are set to $f_{sw} = 50$ kHz and $f_s = 10$ kHz, respectively. With these parameters, the selected values for K_P and K_I in the continuous- and discrete-time domains are those of Table 6.

Table 5. Nominal values of the Buck converter components.

Parameter	Value
L (µH)	650
C (µF)	20
R_L (Ω)	1
r (mΩ)	50
r_C (mΩ)	5

Table 6. Proportional and integral gains of the controller in the continuous and discrete-time domains.

Gain	Continuous-Time	Discrete-Time
K_P	0.0080	0.0086
K_I	12.2400	0.0012

By using the blue user button integrated into the NUCLEO board, the reference current is changed from 1 ($cnt = 2$) to 3 A ($cnt = 3$) and then set back to 1 A ($cnt = 4$). Figures 19 and 20 show the control response during these changes in the reference current. In both cases, the experimental waveforms demonstrate that the transition met the control requirements, with low current ripple due to the high f_{sw}.

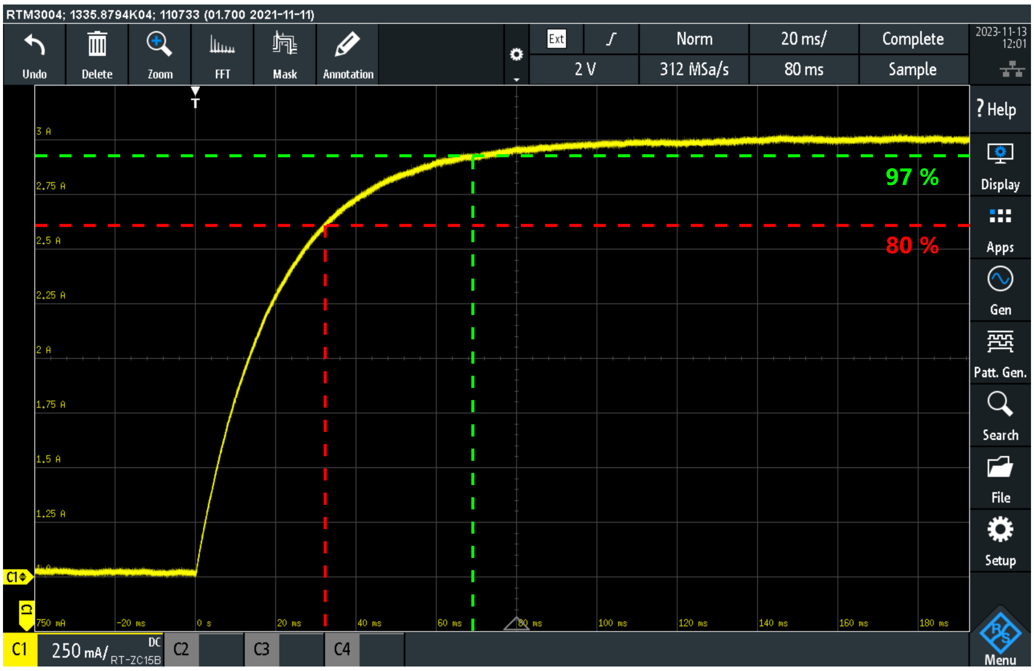


Figure 19. Obtained experimental results when the reference current changes from 1 to 3 A.

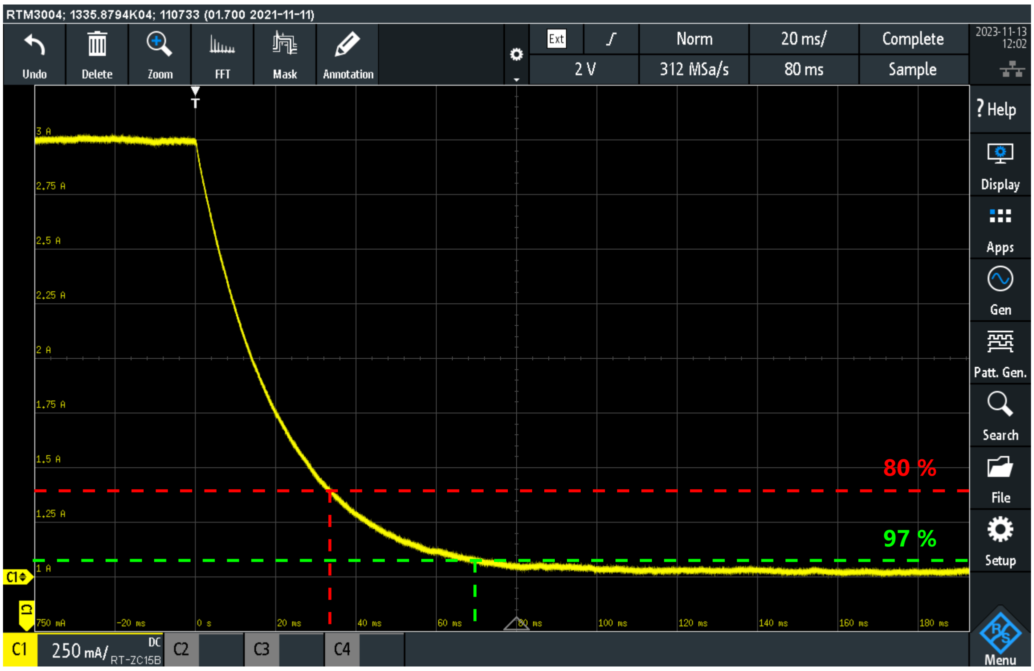


Figure 20. Obtained experimental results when the reference current changes from 3 to 1 A.

The experimental results are compared to those obtained with the block diagram representation of the system in MATLAB Simulink. The comparison is shown in Figure 21, where it is obtained that both transitions are very similar. There is only a small difference in the dynamic response, which can be explained by the fact that for every component of the Buck converter, its nominal value is used. Therefore, if the actual values had been measured, more accurate simulations would have been performed. Nevertheless, it will increase the complexity of the implementation and, since the controller response is satisfactory, it has been decided not to address the components' characterization in this work.

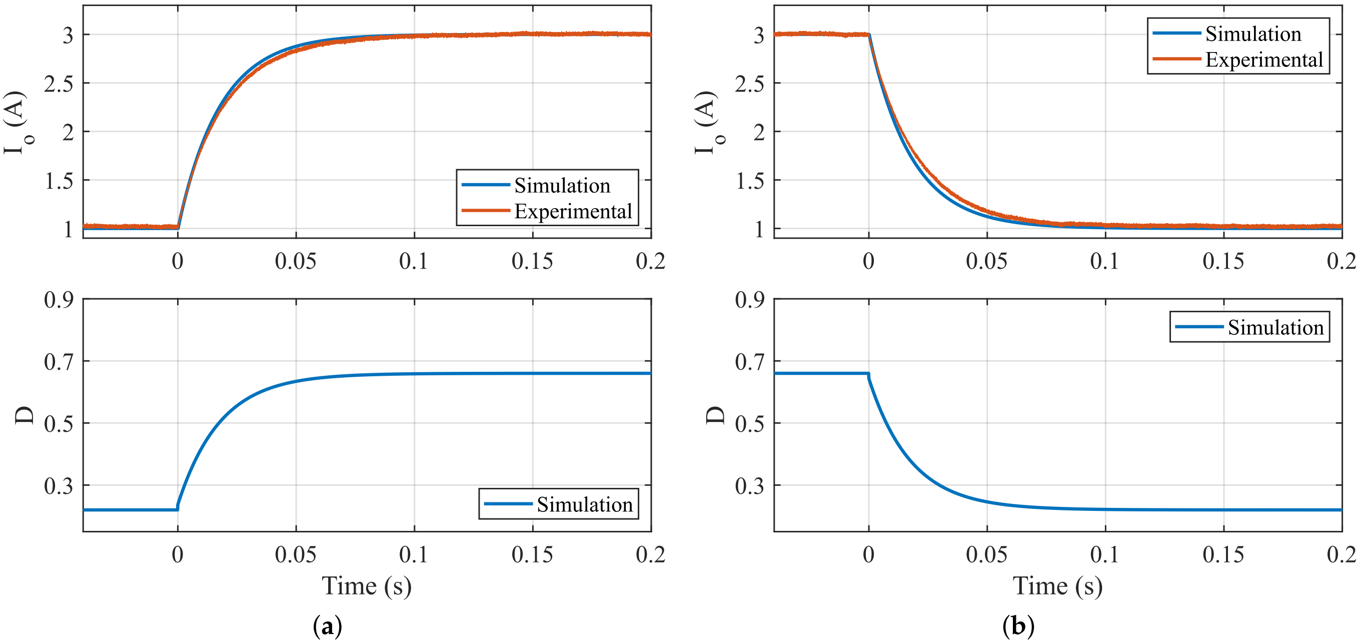


Figure 21. Comparison between simulated and experimental results when the reference current changes from (**a**) 1 to 3 A, and (**b**) 3 to 1 A.

These results highlight that the controller has been correctly designed, since it fulfills the requirements established in the previous sections. In addition, it can be deduced that this guide can be used as an educational material to achieve positive results in a simplified and orderly manner, providing students with a theoretical, simulated, and experimental point of view.

7. Conclusions

The workshop experience detailed in this study has demonstrated the effectiveness of a hands-on approach to learning in the field of power electronics. By guiding students through the complete process of designing, developing, and testing a Buck converter using STM32-based digital control, we have provided a valuable educational tool that enhances theoretical understanding with practical skills. The successful implementation and testing of the converter validate the methodology and highlight the importance of integrating practical exercises in engineering education. All materials, including Simulink and Plecs models, are available online as Supplementary Materials, ensuring that students and educators can easily access and utilize these resources. Future work will focus on expanding the scope of the workshop to include more advanced converter topologies and control techniques, further bridging the gap between classroom learning and real-world applications. In addition, the development of a workshop for using the Buck converter in practical applications such as adjusting the charging process of a battery will be proposed.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/electronics13163207/s1.

Author Contributions: Conceptualization, F.C., I.C. and F.J.L.-A.; methodology, F.C. and F.J.L.-A.; software, I.C. and F.J.L.-A.; validation, I.C., F.J.L.-A. and V.S.M.; formal analysis, F.C., I.C., F.J.L.-A.; investigation, I.C., F.J.L.-A.; resources, A.T.-C.; data curation, F.C., I.C. and F.J.L.-A.; writing—original draft preparation, F.C.; writing—review and editing, F.C. and A.T.-C.; visualization, F.C.; supervision, A.T.-C. and A.R.; project administration, F.C. and A.R.; funding acquisition, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This study was carried out within the MOST - Sustainable Mobility National Research Center and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4 - D.D. 1033 17/06/2022, CN00000023), Spoke 5, Light Vehicle and Active Mobility. This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

Data Availability Statement: The original data presented in the study are openly available in Supplementary Material.

Conflicts of Interest: The authors declare no conflicts of interest

References

1. Muoka, P.I.; Haque, M.E.; Gargoom, A.; Negnevitsky, M. DSP-Based Hands-On Laboratory Experiments for Photovoltaic Power Systems. *IEEE Trans. Educ.* **2015**, *58*, 39–47. [CrossRef]
2. Lamar, D.G.; Miaja, P.F.; Arias, M.; Rodriguez, A.; Rodriguez, M.; Vazquez, A.; Hernando, M.M.; Sebastian, J. Experiences in the Application of Project-Based Learning in a Switching-Mode Power Supplies Course. *IEEE Trans. Educ.* **2012**, *55*, 69–77. [CrossRef]
3. Miaja, P.F.; Lamar, D.G.; de Azpeitia, M.A.P.; Rodriguez, A.; Rodriguez, M.; Hernando, M.M. A Switching-Mode Power Supply Design Tool to Improve Learning in a Power Electronics Course. *IEEE Trans. Educ.* **2011**, *54*, 104–113. [CrossRef]
4. Ochs, D.S.; Miller, R.D. Teaching Sustainable Energy and Power Electronics to Engineering Students in a Laboratory Environment Using Industry-Standard Tools. *IEEE Trans. Educ.* **2015**, *58*, 173–178. [CrossRef]
5. Anand, S.; Farswan, R.S.; Fernandes, B.G. Unique Power Electronics and Drives Experimental Bench (PEDEB) to Facilitate Learning and Research. *IEEE Trans. Educ.* **2012**, *55*, 573–579. [CrossRef]
6. Zumel, P.; Fernandez, C.; Sanz, M.; Lazaro, A.; Barrado, A. Step-By-Step Design of an FPGA-Based Digital Compensator for DC/DC Converters Oriented to an Introductory Course. *IEEE Trans. Educ.* **2011**, *54*, 599–609. [CrossRef]
7. Zhang, Z.; Hansen, C.T.; Andersen, M.A.E. Teaching Power Electronics with a Design-Oriented, Project-Based Learning Method at the Technical University of Denmark. *IEEE Trans. Educ.* **2016**, *59*, 32–38. [CrossRef]
8. Pires, V.F.; Silva, J.F.A. Teaching nonlinear modeling, simulation, and control of electronic power converters using MATLAB/SIMULINK. *IEEE Trans. Educ.* **2002**, *45*, 253–261. [CrossRef]
9. Shahnia, F.; Yengejeh, H.H. Various Interactive and Self-Learning Focused Tutorial Activities in the Power Electronic Course. *IEEE Trans. Educ.* **2019**, *62*, 246–255. [CrossRef]
10. Acero, J.; Carretero, C. A Graduate Magnetic Design Course in a Power Electronics-Oriented Curriculum. *IEEE Trans. Educ.* **2021**, *64*, 223–232. [CrossRef]
11. Kazimierczuk, M.K. *Pulse-Width Modulated DC–DC Power Converters*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2015. [CrossRef]
12. Polivka, W.M.; Chetty, P.R.K.; Middlebrook, R.D. State-Space Average modelling of converters with parasitics and storage-time modulation. In Proceedings of the 1980 IEEE Power Electronics Specialists Conference, Atlanta, GA, USA, 16–20 June 1980; pp. 119–143. [CrossRef]
13. Garg, M.M.; Hote, Y.V.; Pathak, M.K.; Behera, L. An Approach for Buck Converter PI Controller Design Using Stability Boundary Locus. In Proceedings of the 2018 IEEE/PES Transmission and Distribution Conference and Exposition, Denver, CO, USA, 16–19 April 2018. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.